

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Таврійський державний агротехнологічний університет
імені Дмитра Моторного

Ministry of Education and Science of Ukraine

Dmytro motornyi tavia state agrotechnological university

Матеріали III Всеукраїнської
науково-практичної інтернет-конференції
**«СУЧАСНІ КОМП'ЮТЕРНІ ТА
ІНФОРМАЦІЙНІ СИСТЕМИ І ТЕХНОЛОГІЇ»**

12 - 19 грудня 2022 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**Таврійський державний агротехнологічний університет імені Дмитра Моторного
Український державний університет науки і технологій
Національний технічний університет “Харківський політехнічний інститут”
Харківський національний університет радіоелектроніки
Інститут програмних систем Національної Академії Наук України
Рівненський державний гуманітарний університет**

СУЧАСНІ КОМП'ЮТЕРНІ ТА ІНФОРМАЦІЙНІ СИСТЕМИ І ТЕХНОЛОГІЇ

**МАТЕРІАЛИ ПІ ВСЕУКРАЇНСЬКОЇ НАУКОВО-ПРАКТИЧНОЇ
ІНТЕРНЕТ-КОНФЕРЕНЦІЇ**

02-19 грудня 2022 року

Запоріжжя – 2022

Сучасні комп'ютерні та інформаційні системи і технології: матеріали ІІІ Всеукраїнської наук.-практ. інтернет-конф. (Запоріжжя, 12-19 грудня 2022 р.) / ред. кол.: С.В. Кюрчев, В.М. Кюрчев, А.І. Панченко [та ін.]. Запоріжжя: ТДАТУ, 2022. 456 с.

Редакційна колегія:

Кюрчев С. В. – доктор технічних наук, професор;
Кюрчев В. М. – доктор технічних наук, професор;
Панченко А. І. – доктор технічних наук, професор;
Холодняк Ю.В. – кандидат технічних наук, доцент;
Гнатушенко Вік. В. – доктор технічних наук, професор;
Шоман О.В. – доктор технічних наук, професор;
Дудар З.В. – доктор технічних наук, професор;
Войтович І.С. – доктор педагогічних наук, професор;
Розушина Ю. В. – кандидат фізико-математичних наук, доцент;
Малкіна В. М. – доктор технічних наук, професор;
Прийма С. М. – доктор педагогічних наук, професор;
Галько С.В. – кандидат технічних наук, доцент.

Збірник матеріалів ІІІ Всеукраїнської науково-практичної інтернет-конференції «Сучасні комп'ютерні та інформаційні системи і технології» вміщує результати наукових досліджень співробітників закладів вищої освіти, науково-дослідних установ, здобувачів наукових ступенів, докторантів, аспірантів, здобувачів вищої освіти, фахівців з інформаційних технологій та комп'ютерних наук, розробки програмного забезпечення, комп'ютерної графіки, прикладної математики та цифрового бізнесу. Напрямки роботи конференції: математичне і комп'ютерне моделювання складних процесів; управління; обробка та захист інформації; геометричне моделювання та графічні інформаційні технології; нові інформаційні технології в освіті та управлінні освітнім процесом; проектування інформаційних систем; інтелектуальні інформаційні системи та системи штучного інтелекту, робототехніка.

Відповідальність за зміст наданих матеріалів, точність наведених даних та відповідність принципам академічної доброчесності несуть автори. Матеріали видані в авторській редакції.

© Таврійський державний
агротехнологічний університет
імені Дмитра Моторного, 2022
© Автори, 2022

ЗМІСТ

THE USE OF HIERARCHICAL AGGREGATE ASSESSMENT (HAA) THEORY IN ONLINE DISTRIBUTED COMPUTER SYSTEMS FOR DRONE GUIDANCE	10
<i>Martin Lesage</i>	
ВИКОРИСТАННЯ МЕТОДІВ КОМП'ЮТЕРНОГО ПРОЕКТУВАННЯ ПРИ РОЗРОБЦІ ТЕХНОЛОГІЧНОГО ПРОЦЕСУ ТА УПРАВЛЯЮЧОЇ ПРОГРАМИ ДЛЯ ОБЛАДНАННЯ З ЧПУ.....	28
<i>Мацулевич О.Є., Чаплінський А.П.</i>	
ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО РОЗРАХУНКУ РЕЖИМІВ РІЗАННЯ ПРИ ОБРОБЦІ ДЕТАЛЕЙ НА СВЕРДЛИЛЬНИХ ВЕРСТАТАХ	35
<i>Дереза О.О., Антонова Г.В., Тетервак І.Р.</i>	
МЕТОДИ ТА ЗАСОБИ ОРГАНІЗАЦІЇ МЕНТОРСЬКОЇ ПІДТРИМКИ ПРОЦЕСУ ТЕСТУВАННЯ В ІТ-ПРОЕКТІ.....	45
<i>Читулян В.О.</i>	
СТВОРЕННЯ АЛГОРИТМУ РОЗРОБКИ ВИРОБУ З ВИКОРИСТАННЯМ ПАКЕТІВ ПРОГРАМ В ОБЧИСЛЮВАЛЬНІЙ ТЕХНІЦІ.....	46
<i>Гешева Г.В.</i>	
ІНТЕЛЕКТУАЛЬНА ІНФОРМАЦІЙНА ВЕБ-СИСТЕМА «ВИБІР СМАРТФОНУ».....	51
<i>Коломоєць Д.А.</i>	
ОСОБЛИВОСТІ МОДЕЛЮВАННЯ ТА ДІАГНОСТУВАННЯ ТЕХНІЧНОГО СТАНУ ЕЛЕМЕНТІВ РУЛЬОВОГО КЕРУВАННЯ.....	58
<i>Бондар А.М., Дашивець Г.І.</i>	
ПРО ЗАГАЛЬНІ ПІДХОДИ ДО РОЗВ'ЯЗУВАННЯ АЛГОРИТМІЧНО СКЛАДНИХ ЗАДАЧ МОДЕЛЮВАННЯ.....	63
<i>Поліщук О.Д., Яджак М.С.</i>	
РОЗРОБКА СИСТЕМИ ПРОГНОЗУВАННЯ НАДІЙНОСТІ ПРОГРАМНИХ ПРОДУКТІВ.....	67
<i>Борейченко Г.О., Чижмотря О.В.</i>	
ЗАСТОСУВАННЯ МЕТОДУ ГОЛОВНИХ КОМПОНЕНТІВ ДЛЯ УСУНЕННЯ МУЛЬТКОЛІНЕАРНОСТІ.....	69
<i>Зінов'єва О.Г.</i>	
АЛГОРИТМ МАМДАНИ В СИСТЕМАХ НЕЧІТКОГО ВИВЕДЕННЯ	74
<i>Зінов'єва О.Г., Лубко Д.В.</i>	
ПЛАНУВАННЯ ЕКСПЕРИМЕНТІВ З ІМІТАЦІЙНИМИ МОДЕЛЯМИ.....	80
<i>Кучерков А.О.</i>	
ЗАДАЧІ ОПТИМІЗАЦІЇ З НЕЧІТКИМИ ЗМІННИМИ	83
<i>Мартиць Д.С.</i>	
РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВИКОРИСТАННЯ СИСТЕМИ АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ ТЕХНОЛОГІЧНОЇ ДОКУМЕНТАЦІЇ НА ПІДПРИЄМСТВІ СІЛЬСЬКОГОСПОДАРСЬКОГО МАШИНОБУДУВАННЯ.....	87
<i>Мацулевич О.Є., Дереза О.О., Тетервак І.Р.</i>	

АНАЛІЗ ДОСЛІДНОЇ ЕКСПЛУАТАЦІЇ ПРОГРАМНОГО МОДУЛЮ РОЗРАХУНКУ НОРМ ЧАСУ ОБРОБКИ ДЕТАЛЕЙ СІЛЬСЬКОГОСПОДАРСЬКОЇ ТЕХНІКИ	94
<i>Вершков О.О., Бондаренко Л.Ю., Антонова Г.В., Тетервак І.Р.</i>	
ПОРІВНЯННЯ ГНУЧКИХ МЕТОДОЛОГІЙ AGILE ТА WATERFALL	101
<i>Поплавський В.С.</i>	
ВИКОРИСТАННЯ ТЕХНОЛОГІЇ GOOGLE ANALYTICS В РОЗРОБЦІ ВЕБ-ОРІЄНТОВАНОГО ДОДАТКУ «ПЛАТФОРМА СТВОРЕННЯ ОНЛАЙН ОГОЛОШЕНЬ»	103
<i>Вакалюк Т.А., Кияшенко А.С.</i>	
ОПТИМІЗАЦІЯ ПРОЦЕСУ УПРАВЛІННЯ ПОРТФЕЛЕМ ПРОЕКТІВ В ІТ-КОМПАНІЇ ..	105
<i>Новохацький В.С., Вакалюк Т.А.</i>	
ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ПЛАТФОРМ УПРАВЛІННЯ ОБЧИСЛЮВАЛЬНИМИ СЕРВІСАМИ ПРИ ОРГАНІЗАЦІЇ FOG COMPUTING.....	107
<i>Островська К.Ю., Шерстяних М.О., Стовпченко І.В.</i>	
УПРАВЛІННЯ ВАРТІСТЮ ПРОЄКТА ЗА ТЕХНІКОЮ EVM.....	110
<i>Поплавський В.С.</i>	
ДОСЛІДЖЕННЯ СУЧАСНИХ СИСТЕМ МОНІТОРИНГУ СТАНУ СЕРВЕРНОГО ОБЛАДНАННЯ.....	112
<i>Гольцов В.В.</i>	
ПЕРСПЕКТИВИ ВИКОРИСТАННЯ NOSQL СИСТЕМ КЕРУВАННЯ БАЗАМИ ДАНИХ.	119
<i>Вакалюк Т.А., Кияшенко А.С.</i>	
КІБЕРЗЛОЧИННІСТЬ ЯК ЗАГРОЗА СУЧАСНОМУ ІНФОРМАЦІЙНОМУ СУСПІЛЬСТВУ	121
<i>Хімічук І.С.</i>	
РОЛЬ БЕЗПЕКИ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ СИСТЕМ.....	124
<i>Хімічук І.С.</i>	
THE SEARCH FOR INTERNET CONNECTION UNDER EXTREME CONDITIONS.....	127
<i>Zaitseva A.M.</i>	
ОРГАНІЗАЦІЯ СИСТЕМИ ЗАХИЩЕНОГО ОБМІНУ ДАНИМИ ТРАНСПОРТНОГО ПІДПРИЄМСТВА ІЗ РУХОМИМ СКЛАДОМ.....	129
<i>Хохлов М.О., Єфіменко А.А., Вакалюк Т.А.</i>	
АНАЛІЗ НАЯВНИХ ІНСТРУМЕНТІВ ЗБОРУ ВЕБ-ДАНИХ	132
<i>Дуб А.С.</i>	
ПОРІВНЯННЯ МЕТОДОЛОГІЙ ТЕСТУВАННЯ «ЧОРНОЇ» ТА «БІЛОЇ» СКРИНЬКИ.....	137
<i>Вакалюк Т.А., Курачинська А.Р.</i>	
АНАЛІЗ І КОМП'ЮТЕРНА ОБРОБКА ЗОБРАЖЕНЬ ЗА ДОПОМОГОЮ ГРАФІЧНОГО ІНТЕРФЕЙСУ КОРИСТУВАЧА МАТЛАВ.....	140
<i>Мацулевич О.Є., Поспелов М.А., Тетервак І.Р.</i>	
ПРОГРАМНА РЕАЛІЗАЦІЯ АЛГОРИТМУ ФОРМУВАННЯ МОНОТОННИХ КРИВИХ	148
<i>Гавриленко Є.А., Холодняк Ю.В., Мірошніченко М.Ю.</i>	

ДОСЛІДЖЕННЯ СПОСОБІВ ВІЗУАЛІЗАЦІЇ ТРИВИМІРНИХ СЦЕН У РЕАЛЬНОМУ ЧАСІ	155
<i>Мірошниченко М.Ю.</i>	
ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ СТВОРЕННЯ МОДЕЛЕЙ КРИВОЛІНІЙНИХ ПОВЕРХОНЬ.....	163
<i>Холодняк Ю.В., Мірошниченко М.Ю.</i>	
СИСТЕМНИЙ ДИЗАЙН	170
<i>Стеценко К.О.</i>	
ДОСВІД ВИКОРИСТАННЯ ПРОГРАМИ AFORS-NET ПРИ ПІДГОТОВЦІ ФАХІВЦІВ У ГАЛУЗІ ЕЛЕКТРОЕНЕРГЕТИКИ	172
<i>Дяденчук А.Ф.</i>	
ЗНАЧЕННЯ ІНФОРМАЦІЙНО – КОМУНІКАЦІЙНІ ТЕХНОЛОГІЇ У НАВЧАЛЬНОМУ ПРОЦЕСІ ВИЩОЇ ШКОЛИ	176
<i>Бондаренко Л.Ю., Ускова С.О.</i>	
ІНСТРУМЕНТИ КОМУНІКАЦІЇ ДЛЯ ПІДГОТОВКИ ФАХІВЦІВ ТЕХНІЧНИХ СПЕЦІАЛЬНОСТЕЙ	178
<i>Дережа О.О., Дереза С.В.</i>	
ЕКСПОРТ ЖУРНАЛУ ОЦІНОК З НАВЧАЛЬНОГО ПОРТАЛУ З ВИКОРИСТАННЯМ POWER QUERY	182
<i>Каишкар'єв А.О.</i>	
ІСТОРІЯ УПРАВЛІННЯ ПРОЄКТАМИ	188
<i>Кравченко К.Р.</i>	
ПЕРСПЕКТИВНІСТЬ ПРОВЕДЕННЯ КОМП'ЮТЕРНОГО АВТОМАТИЗОВАНОГО ТЕСТОВОГО КОНТРОЛЮ	190
<i>Лубко Д.В.</i>	
ПЕРЕВАГИ ЗМІШАНОГО НАВЧАННЯ В ЗАКЛАДАХ ВИЩОЇ ОСВІТИ В СУЧАСНИХ РЕАЛІЯХ (НА ПРИКЛАДІ ЛДУ БЖД)	196
<i>Полотай О.І.</i>	
ЕФЕКТИВНІСТЬ ПРОВЕДЕННЯ ЛАБОРАТОРНИХ РОБІТ З ІНЖЕНЕРНОЇ МЕХАНІКИ В РЕЖИМІ КОНФЕРЕНЦІЇ ZOOM	204
<i>Антонова Г.В.</i>	
КОМП'ЮТЕРНА ГРАФІКА. СФЕРА ЗАСТОСУВАННЯ І ПЕРСПЕКТИВИ РОЗВИТКУ .	208
<i>Постол Ю.О.</i>	
ТЕХНОЛОГІЇ BYOD: МОБІЛЬНІ ПРОГРАМИ НА ДОПОМОГУ ВЧИТЕЛЮ	211
<i>Сікора Я.Б., Ляшенко А.І.</i>	
ОГЛЯД ХМАРНИХ СЕРВІСІВ ДЛЯ ОСВІТИ	217
<i>Сіциліцин Ю.О., Семенов Є.О.</i>	
ПРОЄКТУВАННЯ БАЗИ ДАННИХ ДЛЯ НАВЧАННЯ З ПИТАНЬ ОХОРОНИ ПРАЦІ	221
<i>Козирєва Т.А., Дмитренко І.А.</i>	
ВИБІР ОНЛАЙН ЗАСОБІВ ДЛЯ РОЗРОБКИ ПАРАЛЕЛЬНИХ ПРОГРАМ З ВИКОРИСТАННЯМ БІБЛІОТЕКИ MPI	223
<i>Сіциліцин Ю.О.</i>	

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ВОЛОНТЕРСЬКОГО ШТАБУ	226
<i>Вакалюк Т.А., Андрусенко О.М.</i>	
РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ КЛІНІК ТА ЛІКАРЕНЬ	228
<i>Вакалюк Т.А., Скріпченко Д.Г.</i>	
МІКРОСЕРВІСНА АРХІТЕКТУРА У РОЗРОБЦІ КОРПОРАТИВНИХ ВЕБ-ДОДАТКІВ ...	230
<i>Каліберда Ю.О.</i>	
СТВОРЕННЯ ІНФОРМАЦІЙНОЇ ВЕБ-СИСТЕМИ „ІНТЕРНЕТ-МАГАЗИН ПІЦЦЕРІЇ” ...	232
<i>Катане О.Г.</i>	
ВИЗНАЧЕННЯ НЕСПРАВНОСТЕЙ комп'ютера ЗА ДОПОМОГОЮ ПРОЕКТУВАННЯ ІНФОРМАЦІЙНО-ДОВІДКОВОЇ СИСТЕМИ	240
<i>Лубко Д.В., Зінов'єва О.Г.</i>	
МЕТОДИКА РОЗРОБКИ ТА ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ ВЕБ-СИСТЕМИ «БІБЛІОТЕКА»	247
<i>Назаров Є.М.</i>	
ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ ГРОМАДСЬКОЇ ВОЛОНТЕРСЬКОЇ ОРГАНІЗАЦІЇ.....	252
<i>Нарватов О.П., Полозов Д.М., Широкопетлєва М.С.</i>	
ПЕРЕВАГИ ТА НЕДОЛІКИ ГНУЧКОЇ МЕТОДОЛОГІЇ SCRUM ДЛЯ УПРАВЛІННЯ ІТ ПРОЕКТАМИ.....	254
<i>Савчук Ю.В.</i>	
ВИКОРИСТАННЯ ТЕХНОЛОГІЇ DOCKER В РОЗРОБЦІ ВЕБ-ОРІЄНТОВАНОГО ДОДАТКУ «ПЛАТФОРМА СТВОРЕННЯ ОНЛАЙН ОГолошень».....	256
<i>Вакалюк Т.А., Кияшенко А.С.</i>	
ТЕОРЕТИЧНІ АСПЕКТИ ОРГАНІЗАЦІЇ АУТЕНТИФІКАЦІЇ, АВТОРИЗАЦІЇ ТА АУДИТУ В СУЧАСНИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ.....	258
<i>Білявський Н.А.</i>	
АНАЛІЗ МОЖЛИВОСТЕЙ СЕРВЕРНИХ РІШЕНЬ АУТЕНТИФІКАЦІЇ НА БАЗІ ПРОТОКОЛУ RADIUS	260
<i>Білявський Н.А.</i>	
МОДЕЛЬ AAA, ЯК ОСНОВА СУЧАСНИХ СИСТЕМ АУТЕНТИФІКАЦІЇ НА БАЗІ ПРИСТРОЇВ КОМПАНІЇ CISCO	264
<i>Русятинська А.О.</i>	
АНАЛІЗ ФУНКЦІЙ ТА БЕЗПЕКИ ПРОТОКОЛІВ МОДЕЛІ AAA	267
<i>Русятинська А.О.</i>	
ТЕХНІКИ ОЦІНЮВАННЯ ПРОЕКТІВ	271
<i>Савчук Ю.В.</i>	
РОЗРОБКА СУЧАСНОГО ІНТЕРНЕТ-МАГАЗИНУ МЕБЛІВ	273
<i>Фельдшерев Е.О.</i>	
SKEW-SYMMETRIC MATRIX METHOD FOR BALANCING INTRANSITIVE GAMES	281
<i>Yevhenii Krupchak, Yurii Novikov</i>	
АНАЛІЗ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ В УПРАВЛІННІ РЕСТОРАННИМ БІЗНЕСОМ.....	284
<i>Верещага Ю.В.</i>	

ІНТЕЛЕКТУАЛЬНА ІНФОРМАЦІЙНА СИСТЕМА «ПІДБОРУ МОНОБЛОКУ».....	290
<i>Супрун М.В., Холодняк Ю.В.</i>	
РОЗРОБКА ЗАСТОСУНКУ ДЛЯ УПРАВЛІННЯ СИСТЕМАМИ РОЗУМНОГО БУДИНКУ.....	298
<i>Назаров Є.М.</i>	
ОПТИЧНІ-ВОЛОКНА.....	307
<i>Гузюк В.В.</i>	
РОЗРОБКА ІГРОВОГО ЗАСТОСУНКУ ДЛЯ МОБІЛЬНОЇ ПЛАТФОРМИ У ЖАНРІ ПРИГОДНИЦЬКОЇ СТРАТЕГІЇ.....	310
<i>Арієнєков О.М., Новіков Ю.С.</i>	
ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОПЕРАТИВНОГО РОЗРАХУНКУ НОРМ ЧАСУ ВИГОТОВЛЕННЯ ДЕТАЛЕЙ ТА МЕХАНІЗМІВ	313
<i>Вершков О.О., Бондаренко Л.Ю., Гавриленко Є.А.</i>	
РОЗРОБКА УЗАГАЛЬНЕНОГО ПРОЕКТУ КОМП'ЮТЕРНОЇ МЕРЕЖІ.....	319
<i>Білявський Н.А.</i>	
РОЗРОБКА ПРОЕКТУ КОМП'ЮТЕРНОЇ МЕРЕЖІ ТА ПІДСИСТЕМИ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ	323
<i>Русятинська А.О.</i>	
ВИКОРИСТАННЯ LAMP ДЛЯ РОЗРОБКИ ВЕБ-ЗАСТОСУНКІВ	326
<i>Вакалюк Т.А., Кузьмук В.О.</i>	
ХАРАКТЕРИСТИКА ГНУЧКОГО (AGILE) ПІДХОДУ УПРАВЛІННЯ ІТ ПРОЕКТАМИ..	328
<i>Ковальчук О.А.</i>	
ПЕРЕВАГИ ТА НЕДОЛІКИ ВИКОРИСТАННЯ МОВИ JAVASCRIPT.....	331
<i>Вакалюк Т.А., Кузьмук В.О.</i>	
ПРОЕКТУВАННЯ І РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ	333
<i>Величко С.Д.</i>	
РОЗРОБКА ВЕБЗАСТОСУНКУ МАПИ ВИЗНАЧНИХ МІСЦЬ З АУДІОВІДТВОРЕННЯМ ІНФОРМАЦІЇ	338
<i>Перевалова А.Д., Чижмотря О.Г.</i>	
АНАЛІЗ МЕТОДОЛОГІЙ ВИЗНАЧЕННЯ ЛОЯЛЬНОСТІ КЛІЄНТІВ ТА ОЦІНКИ РІВНЯ ЇХ ЗАДОВОЛЕНOSTІ	341
<i>Лейба Я.А., Широкопетлева М.С.</i>	
PECULIARITIES OF LEGACY PROJECTS SUPPORT	344
<i>Oleksii Kucherenko</i>	
РОЗРОБКА СОЦІАЛЬНОЇ МЕРЕЖІ ДЛЯ МЕЛОМАНІВ	346
<i>Гордєєв Р.С., Вакалюк Т.А.</i>	
ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ПРОГРАМНОГО ПРОДУКТУ GPSS WORLD ПРИ МОДЕЛЮВАННІ ІНФОРМАЦІЙНИХ СИСТЕМИ	349
<i>Мацулевич О.Є., Тетєрвак І.Р.</i>	
ОПИС РОБОТИ КОРИСТУВАЧА З ПРОГРАМНИМ МОДУЛЕМ «РОЗРАХУНОК РЕЖИМІВ РІЗАННЯ ПРИ ВИКОНАННІ СВЕРДЛИЛЬНИХ ОПЕРАЦІЙ».....	354
<i>Дєреза О.О., Бондаренко Л.Ю., Антонова Г.В., Тетєрвак І.Р.</i>	

СПЕЦІАЛІЗОВАНИЙ ПРОГРАМНИЙ МОДУЛЬ РОЗРАХУНКУ ОПЕРАЦІЙНИХ НОРМ ЧАСУ ОБРОБКИ ДЕТАЛЕЙ СІЛЬСЬКОГОСПОДАРСЬКОЇ ТЕХНІКИ ЗА УМОВ ІНДИВІДУАЛЬНОЇ ОРГАНІЗАЦІЇ ПРАЦІ	361
<i>Івженко О.В., Антонова Г.В., Чаплінський А.П., Михайленко О.Ю.</i>	
РОЗРОБКА ІНФОРМАЦІЙНО-ТЕСТУВАЛЬНОЇ СИСТЕМИ ДЛЯ ОЦІНЮВАННЯ СТУДЕНТІВ У ЗВО	369
<i>Лубко Д.В.</i>	
РОЗПІЗНАВАННЯ ФІГУР РУКОПИСНИХ ДІАГРАМ ТА СХЕМ ЗА ДОПОМОГОЮ ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ.....	376
<i>Українець М.О., Вакалюк Т.А.</i>	
HANDWRITTEN DIAGRAMS AND SCHEMES OF TEXT RECOGNITION USING CONVOLUTIONAL NEURAL NETWORKS.....	378
<i>Ukrainets M.O., Vakaliuk T.A.</i>	
МЕДИЧНІ ІНФОРМАЦІЙНІ СИСТЕМИ. ОГЛЯД ТА ПОРІВНЯННЯ.....	380
<i>Коломоєць Д.А.</i>	
АНАЛІЗ МЕДИЧНИХ ЕКСПЕРТНИХ СИСТЕМ.....	385
<i>Лубко Д.В.</i>	
ДОСЛІДЖЕННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ ДЛЯ РІШЕННЯ ЗАДАЧ МЕДИЧНОГО ПРОФІЛЮ	392
<i>Островська К.Ю., Мінаєнко А.С.</i>	
ДОСЛІДЖЕННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗРОБКИ СИСТЕМИ АВТОМАТИЧНОЇ КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ ФОТОБАНКУ	395
<i>Островська К.Ю., Рогбак К.С.</i>	
ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В МЕДИЦИНІ.....	398
<i>Вакалюк Т.А., Андрусенко О.М.</i>	
ДОСЛІДЖЕННЯ АЛГОРИТМІВ РЕКОМЕНДАЦІЙНИХ СИСТЕМ.....	400
<i>Гордєєв Р.С., Вакалюк Т.А.</i>	
ВИКОРИСТАННЯ RANDOM FOREST REGRESSOR ДЛЯ ПРОГНОЗУВАННЯ ЦІНИ АВТОМОБІЛЯ	404
<i>Загацький В.В., Вакалюк Т.А.</i>	
ІНФОРМАЦІЙНО-ПОРАДНА СИСТЕМА ВИБОРУ МОВИ ПРОГРАМУВАННЯ.....	406
<i>Засипко В.П.</i>	
СТВОРЕННЯ ЗАСОБУ РОЗУМІННЯ МОВИ ЖЕСТІВ ЗА ДОПОМОГОЮ НЕЙРОННИХ МЕРЕЖ.	412
<i>Ващенко К.Я.</i>	
ЗАБЕЗПЕЧЕННЯ ПРИНЦИПІВ FAIR У СЕМАНТИЧНИХ WIKI-РЕСУРСАХ ДЛЯ ПІДТРИМКИ ВІДКРИТОЇ НАУКИ В УКРАЇНІ.....	415
<i>Рогущина Ю.В.</i>	
ВИКОРИСТАННЯ ПАРАДИГМИ МУЛЬТИАГЕНТНИХ СИСТЕМИ ДЛЯ ПЛАНУВАННЯ ПОВЕДІНКИ КОМАНДИ РОЮ ДРОНІВ.....	423
<i>Рогущина Ю.В., Гладун А.Я.</i>	
ОСОБЛИВОСТІ ПОБУДОВИ СИСТЕМ НАДАННЯ ВІДПОВІДЕЙ	429
<i>Пироженко М.Ю.</i>	

МУЛЬТИАГЕНТНА СИСТЕМА МЕРЕЖІ ДРОНІВ ДЛЯ ЗАХИСТУ КРИТИЧНОЇ ІНФРАСТРУКТУРИ З ОНТОЛОГІЧНИМ ПОДАННЯМ ЗНАНЬ	431
<i>Гладун А.Я., Хала К.О.</i>	
ОГЛЯД СУЧАСНИХ ГЕОІНФОРМАЦІЙНИХ СИСТЕМ	437
<i>Мелешко О.Д.</i>	
ІНТЕЛЕКТУАЛЬНІ ІНФОРМАЦІЙНІ СИСТЕМИ ТА СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ.	441
<i>Пранов Л.І., Вакалюк Т.А.</i>	
ОСОБЛИВОСТІ ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В СИСТЕМІ ВИЩОЇ ОСВІТИ США	448
<i>Кулешов С.О.</i>	
ВАЖЛИВІСТЬ НАПИСАННЯ ЮНІТ ТЕСТІВ ДЛЯ ПРОГРАМНОГО КОДУ	451
<i>Сікайло В.О., Кравченко С.М.</i>	
ПЕРСПЕКТИВИ ВИКОРИСТАННЯ ДИНАМІЧНИХ ПРОТОКОЛІВ КЕРУВАННЯ VLAN - МЕРЕЖАМИ У ХМАРНИХ СЕРВІСАХ.....	453
<i>Сідлецька Д.Р., Єфіменко А.А., Кручинський Я.Т., Вакалюк Т.А.</i>	
ПОКАЖЧИК АВТОРІВ	455

THE USE OF HIERARCHICAL AGGREGATE ASSESSMENT (HAA) THEORY IN ONLINE DISTRIBUTED COMPUTER SYSTEMS FOR DRONE GUIDANCE

Martin Lesage
École Polyvalente de Charny (ESLE)
Québec, Canada

email:lesagelm@hotmail.com

Abstract. The use of Hierarchical Aggregate Assessment (HAA) theory for the guidance of groups of drones is possible. This theory have its roots in the education field but teamwork assessment with several levels of hierarchy algorithms could be extended in the fields of artificial intelligence and operational research for drone guidance, position, target acquisition, target striking, mission assessment procedures using conventional, distributed, cloud or parallel processing, The guidance of drones could use parallel programming because each drone could be assigned to a computer or a processor. Competency based approach paradigm could be used and competencies assessment in the case of a drone mission could be taken as a complex assessment task. In fact, competencies assessment is the presentation of a complex assessment task with assessment criteria. In the drone guidance case, complex assessment tasks could define themselves as to give a group of drones a mission to watch, identify or strike some targets with assessment parameters as group of drone position, target location, number or target to destroy, drone guidance parameters, target acquisition, target identification, target illumination and target striking. The aggregation process is also part of a complex situation that takes a group of drones, forms these drones in teams with a hierarchical structure and assigns to them a hierarchical position with a task description. More research will be done for the development of teamwork with multiple hierarchy levels assessment tasks in the Hierarchical Aggregate Assessment (HAA) field by computer applications that could be implemented in parallel programming, cloud computing and distributed computing paradigms. Aggregation process could be done manually or by computer using game theory, multi-agent systems, ontologies or swarn intelligence. Hierarchical Aggregate Assessment (HAA) computer applications could also be seen as a mean to form groups of drones onto a hierarchical structure, to modify the hierarchical structure if some drones are destroyed or malfunctioning and perform mission assessment in real time with parallel programming having mission assessment parameters as success (% of mission done, % of target destroyed), drone group status (% drones destroyed, % drones malfunctioning) drone position guidance, target acquisition, target identification, target illumination and striking. This paper is a dissertation on the use of Hierarchical Aggregate Assessment (HAA) theory in online distributed computer systems for drone guidance. The paper first introduces the problematics of Hierarchical Aggregate Assessment (HAA) theory and online distributed computer systems for drone guidance. Hierarchical Aggregate Assessment (HAA) definition is then stated and the theoretical framework is introduced presenting concepts as aggregation process and also implementation models in cloud computing, parallel programming and distributed computing. Finally Hierarchical Aggregate Assessment (HAA) for drone guidance is presented with the aggregation process done manually, with ontologies, game theory, multi-agent systems and swarn intelligence.

How to cite this paper: Lesage, M. (2022). The Use of Hierarchical Aggregate Assessment (HAA) in Online Distributed Computer Systems for Drone Guidance

Keywords

Collaborative Work, Distributed Computer Systems, E-Learning, E-Assessment, Management Information Systems (MIS), Online Teamwork assessment, Hierarchical Aggregate Assessment (HAA), E-Assessment, Measurement and Evaluation, Parallel Programming, Game Theory, Multi-Agent Systems, Ontologies, Swarm Intelligence

1. Introduction

Groups of drone guidance could be done manually by drone pilots or automatically with artificial intelligence and computer systems. Sometimes, many drones could take part to a mission and it could be difficult for the computer systems to coordinate the group of drone to accomplish the mission. A solution to this problematic could be to form the drones in groups or teams with multiple levels of hierarchy where some drones could be appointed hierarchical levels as team member, team leader and group manager. The aggregation process, if computerized, could use parallel programming, cloud computing, distributed computing, game theory, multi-agent systems, ontology and swarm intelligence to form groups of drones in teams with several levels of hierarchy. In the conventional assessment paradigm all individuals have the same assessment at the opposite of Hierarchical Aggregate Assessment (HAA) where the assessment changes with the hierarchical level of the team member. That type of assessment gives scores on the task to be done and on leadership soft skills. This brings changes to the theories and paradigms of conventional assessment. A hierarchy is found in the results, the positions assigned to the members and the assessment tasks.

The computerized assessment of teams at several levels of hierarchy has existed for a long time, but in the field of management information systems. The interdisciplinary aspect of the Hierarchical Aggregate Assessment (HAA) gathers the fields of IT, information systems, artificial intelligence, parallel programming, measurement and evaluation. Assessment process is carried out through team assessment tasks in which team members have hierarchical positions such as: team member, team leader as well as group manager or administrator (who lead team leaders) and possibly a very large number of other hierarchy levels.

2. Problematics

Drone formation in teams with several levels of hierarchy could be done manually or with computerized systems. The aggregation and guidance process is slow when done manually and many drones could be destroyed under a fast and coordinated enemy attack. Therefore, it could be important to improve the speed of drone coordination with artificial intelligence and parallel processing to maximize the success of the mission and to minimize the number of drones destroyed.

Drones mission tasks could include the grouping of drones with several levels of hierarchy, target identification, drone location as well as mission results assessment (% of mission done, % of target destroyed), drone group status (% drones destroyed, % drones malfunctioning) drone position guidance, target acquisition, target identification, target illumination and striking. Research on solutions to certain problems considering the implementation at the level of computerization of the aggregation process and the inclusion of this discipline in teaching programs is to be done. Hierarchical Aggregate Assessment (HAA) changes paradigms and current knowledge about assessment in the field of Education. Unlike conventional assessment,

Hierarchical Aggregate Assessment (HAA) implies a hierarchy of team members, assessment tasks and assessment results, as shown in **Figure 1**.

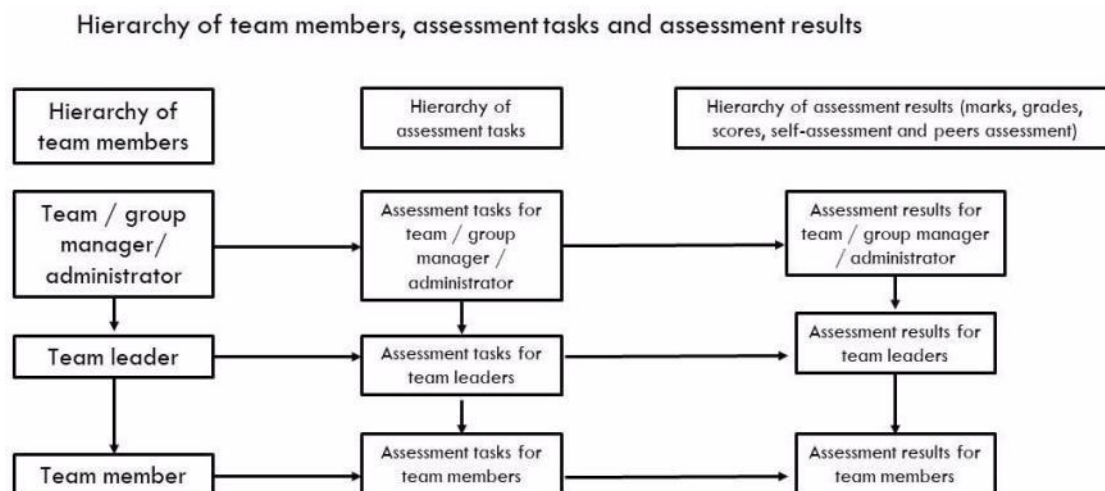


Figure 1. Hierarchy of parameters in the Hierarchical Aggregate Assessment (HAA) field

2.1. Hierarchical Aggregate Assessment Definition

The Hierarchical Aggregate Assessment (HAA) theory is based on the assessment of teams with several levels of hierarchy in education. This assessment process is done on a treelike organization similar as the ones in management information systems. The process is done in three phases, the first phase consists in the team formation and the attribution of hierarchical levels to team members (students) that is the aggregation process, the second phase is the presentation of a test or an assessment task done in team to the team members (students) and the third phase is the team dislocation and the return to the initial phase until the course is done. This iterative process consists of the course curriculum management. While the process iterates, assessment data is collected through the process as summative and formative assessment data that can be used to determine the course success or to guide student for improvement, as shown in **Figure 2**.

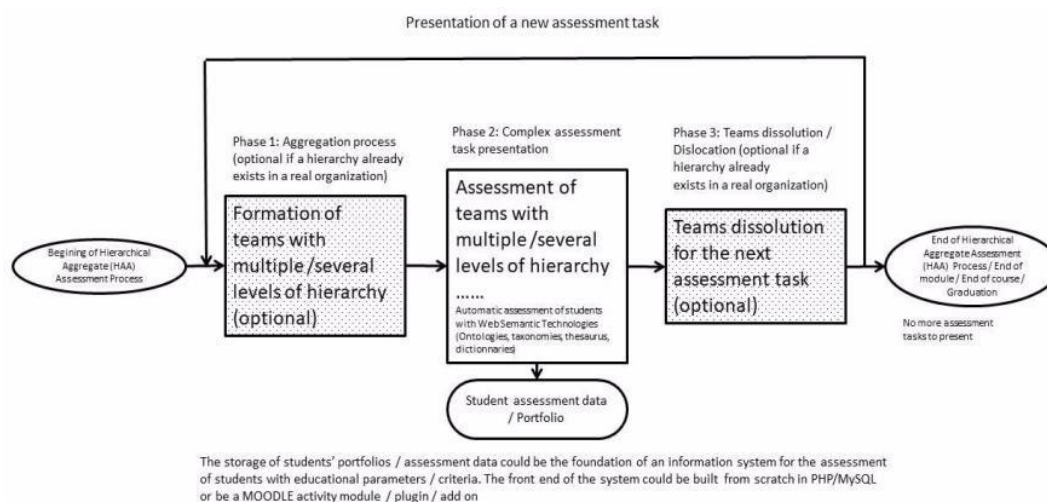


Figure 2. Aggregation and complex assessment tasks presentation process.

An important aspect of HAA's problematic is the occurrence that the hierarchical structure could change during the assessment task because of the probability that some elements or team members quit the task during the assessment process and must be replaced. The replacement process can be done manually by the persons in charge of the task or automatically with Artificial Intelligence (AI) by computer systems. Algorithms and communications between the team members or the elements at different hierarchical positions and also replacements procedures could be implemented in parallel programming, cloud computing and distributed computing paradigms. Aggregation process could be done manually or by computer using game theory, multi-agent systems, ontologies or swarn intelligence, as shown in **Figure 3**.

Aggregation process throughout the assessment task

The hierarchical structure could change during the assessment task because of the probability that some team members quit the task during the assessment process and must be replaced

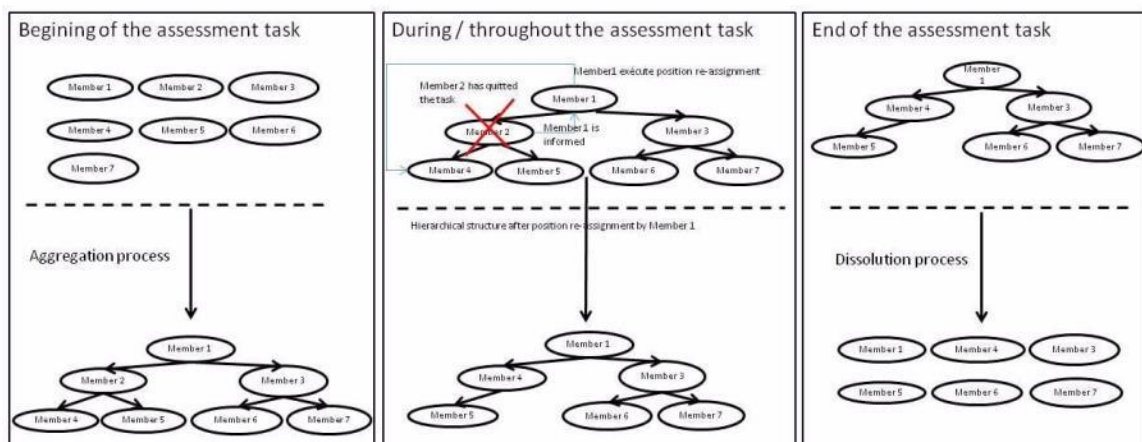


Figure 3. Aggregation process throughout the task.

3. Theoretical framework

The field of Hierarchical Aggregate Assessment (HAA) is the main concern of the actual research project on teamwork assessment which includes the aggregation process that extends the work of Nance (2000, p. 298). The computer applications developed in this research project are presentations engine for teamwork assessment tasks. Teamwork assessment implies collaborative work and collaborative problem based learning on the Internet. Collaborative learning (Boud, Cohen and Sampson, 1999, p. 414; Comeaux, Huber, Kasprzak and Nixon, 1998, p. 3-5; Van Zyl and Massyn, 2011, p. 96), collaborative work assessment (Lingard, 2010, p. 35) and Problem-based learning (Gijbels, Dochy, Van den Bossche and Segers, 2005, p. 29-33) are implemented by the development of complex assessment tasks with multiple levels of hierarchy. The actual research uses teamwork assessment tasks (Saskatchewan Professional Development Unit, 2003; Olivier, 2002, Louis & Bernard, 2004, p. 79; Tardif, 2006, p. 125) where team members could have several hierarchical positions that situates themselves into two educational paradigms that are mastery learning (Legendre, 2005, p. 943) and competency-based learning. The mastery leaning approach (Legendre, 2005, p. 943) could be implemented in this research project by a progress report which indicates the marks of the assignments and exams as well as the progress of the students in the course modules. In the actual research project, competency-based approach implies the assessment of transversal skills and competencies by creating a portfolio of productions

The process of HAA brings together team members into teams that include multiple levels of hierarchy where these people can occupy the hierarchical positions of president, team manager (administrator), team leader as well as team member. The structure of the team is in the form of a pyramid or an inverted tree representing an organizational chart in which each branch is an aggregation of team members. The process of HAA is the action of grouping team members together in a hierarchical organizational structure with several levels and then make an assessment process for each member of the team that is a leaf of the tree or a node of the organizational structure. The HAA process is able to assess team members and also the homework or productions done by the teams and the whole group. In the HAA process, the teams with several levels of hierarchy can be formed manually by the teacher or by computerized algorithms that could use cluster analysis methods.

3.2. HAA implementation in cloud computing

The HAA process is the assessment of teams with several levels of hierarchy. The team organization have a treelike structure that implements the hierarchy. In the case of cloud computing, the HAA application can stay on the HAA institution server like every cloud application that is not implemented on the user's computer but in the cloud where the HAA Internet application resides, as shown in **Figure 5**.

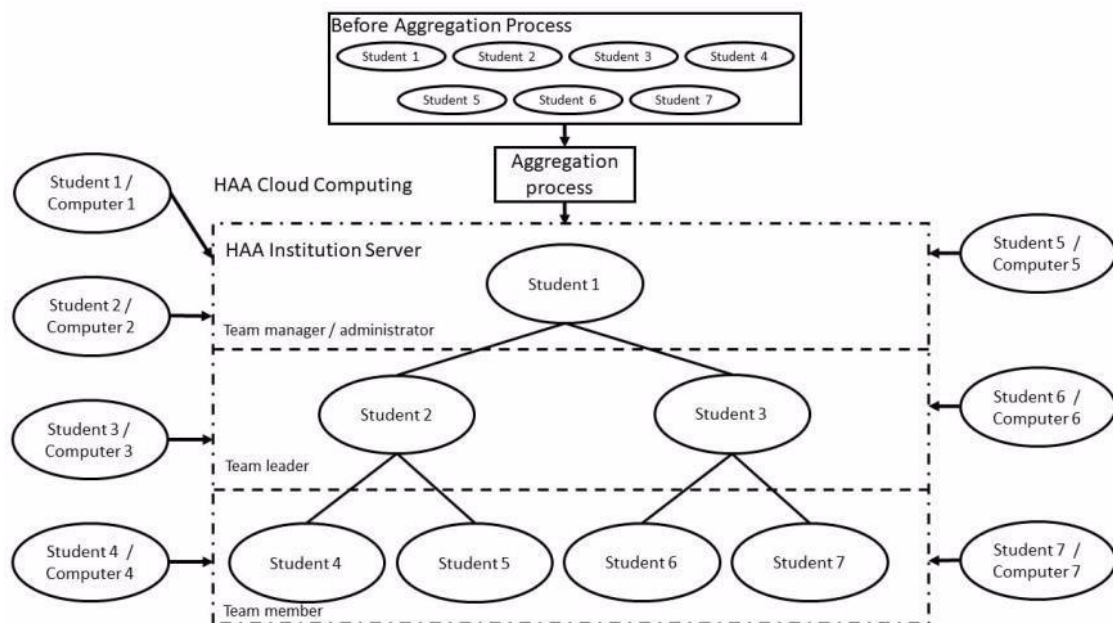


Figure 5. Model of the Implementation of Hierarchical Aggregate Assessment in Cloud Computing

3.3. HAA implementation in parallel programming

The HAA process is the assessment of teams with several levels of hierarchy. The team organization has a treelike structure to implement the hierarchy. In the case of parallel programming, the HAA parallel application can assign one processor by student once the aggregation process is done, as shown in **Figure 6**.

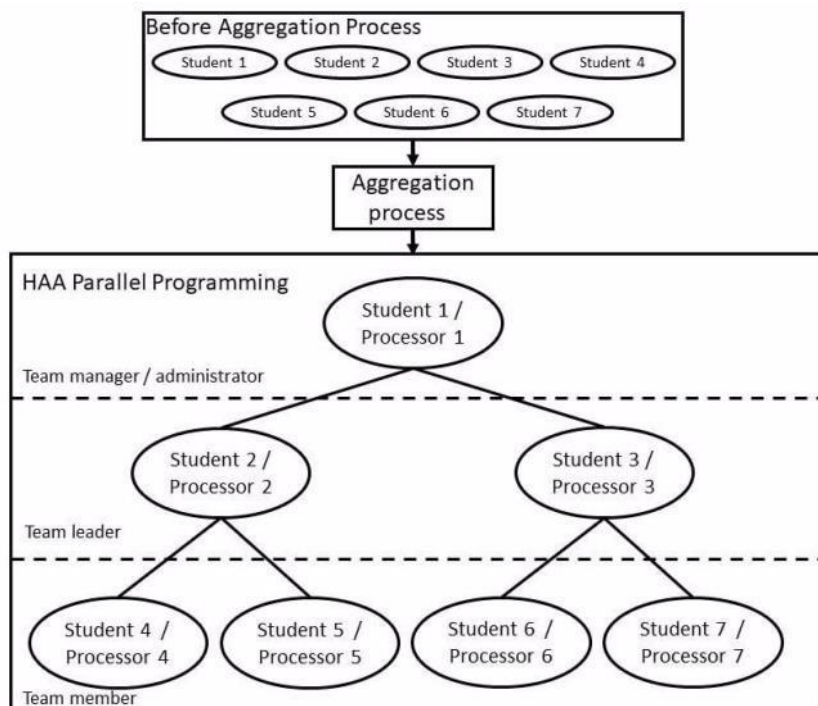


Figure 6. Model of the Implementation of Hierarchical Aggregate Assessment in Parallel Programming

3.4. HAA implementation in distributed computing

The HAA process is the assessment of teams with several levels of hierarchy. The team organization has a treelike structure to implement the hierarchy. In the case of distributed programming, the HAA parallel application can assign one computer by student once the aggregation process is done, as shown in **Figure 7**.

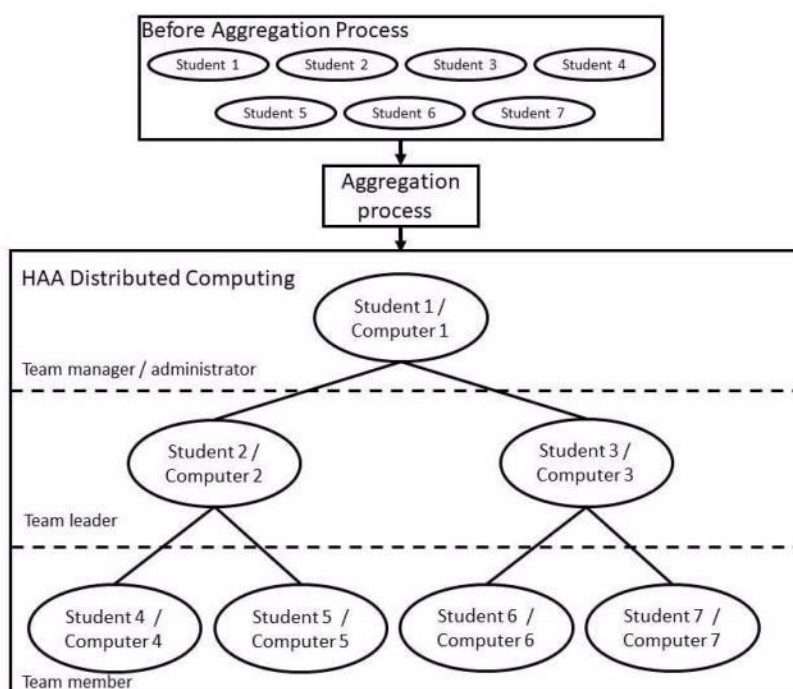


Figure 7. Model of the Implementation of Hierarchical Aggregate Assessment in Distributed Computing

4. Hierarchical Aggregate Assessment for Drone Guidance

The use of Hierarchical Aggregate Assessment (HAA) theory for the guidance of groups of drones is possible. It could be named “The use of Hierarchical Aggregate Assessment (HAA) theory in the artificial intelligence field for drone guidance, position, target acquisition, target striking, mission assessment procedures using conventional, distributed, cloud or parallel processing”. The guidance of drones could use parallel programming because each drone could be assigned to a computer or a processor. Competency based approach paradigm could be used and competencies assessment in the case of mission assessment (% of mission done, % of target destroyed), drone group status (% drones destroyed, % drones malfunctioning) drone position guidance, target acquisition, target identification, target illumination and striking could be taken as complex assessment tasks. In fact, competencies assessment is the presentation of a complex assessment task with assessment criteria. In the drone guidance case, complex assessment tasks could define themselves as to give a group of drones a mission to watch, identify or strike some targets with assessment parameters as group of drone position, target location, number or target to destroy, drone guidance parameters, target acquisition, target identification, target illumination and target striking. The aggregation process is also a part of a complex situation that selects a group of drones, form these drones in teams with a hierarchical structure and assigns to them a hierarchical position with a task description. The aggregation process is done at the beginning of the mission to form teams of drones in a hierarchical structure, as shown in **Figure 8**.

Aggregation process

The aggregation process in the placement of drones in teams with several levels of hierarchy in with a computer (classical programming, game theory, multi-agents systems, ontologies, swarm intelligence or artificial intelligence) or manually by an operator.

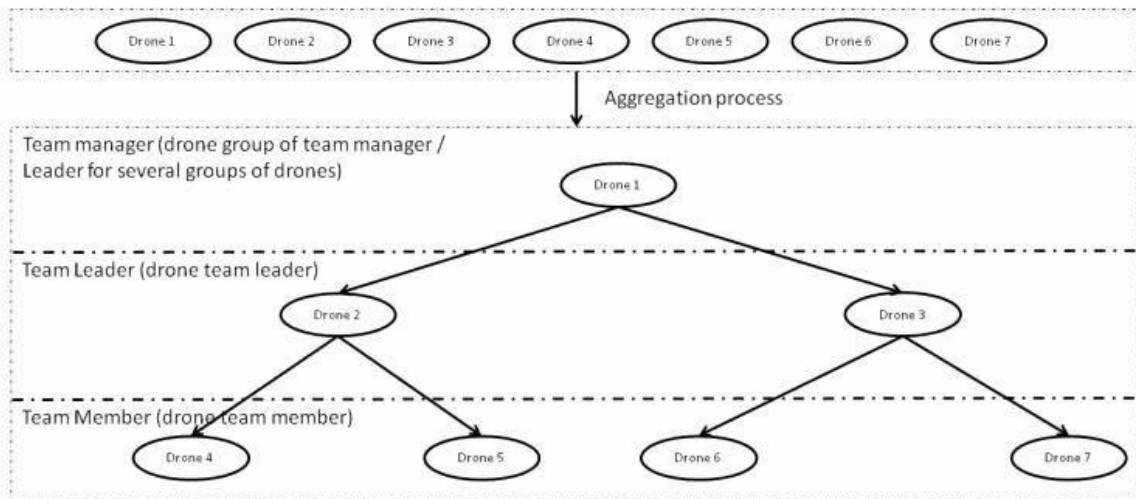


Figure 8. Aggregation process at the beginning of the mission

The aggregation process is also done at the end on the mission to dissolve the teams of drone in order to prepare for the next mission that could need to re-organize the drone teams hierarchical structure, as shown in **Figure 9**.

Dissolution process / Drone teams dissolution

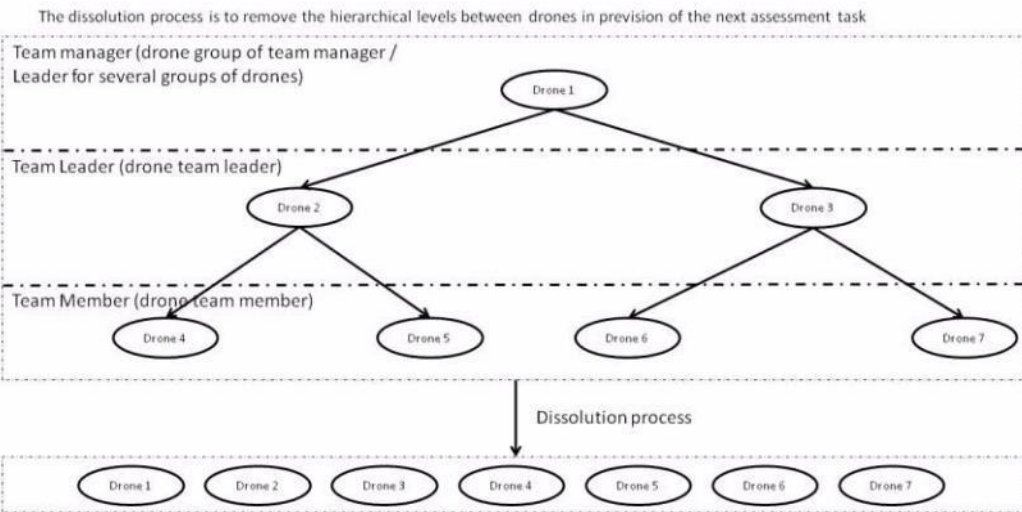


Figure 9. Dissolution process at the end of the mission

Drones team hierarchical structure can change at the end of the mission with drone team dissolution and also during the mission to replace the destroyed or malfunctioning drones. While performing the mission (the assessment task), drones could be destroyed or experience malfunctions. In that case, a drone with a higher hierarchical level could assign a functional (non-destroyed or not malfunctioning) drone a new hierarchical position to replace it and continue the mission. The aggregation process at the beginning and at the end of the mission is shown in **Figure 10**.

Aggregation process throughout the mission

The hierarchical structure could change during the mission because of drones malfunctions, drones destroyed and drone speciality that enables them to lead a drone team during a part of the mission

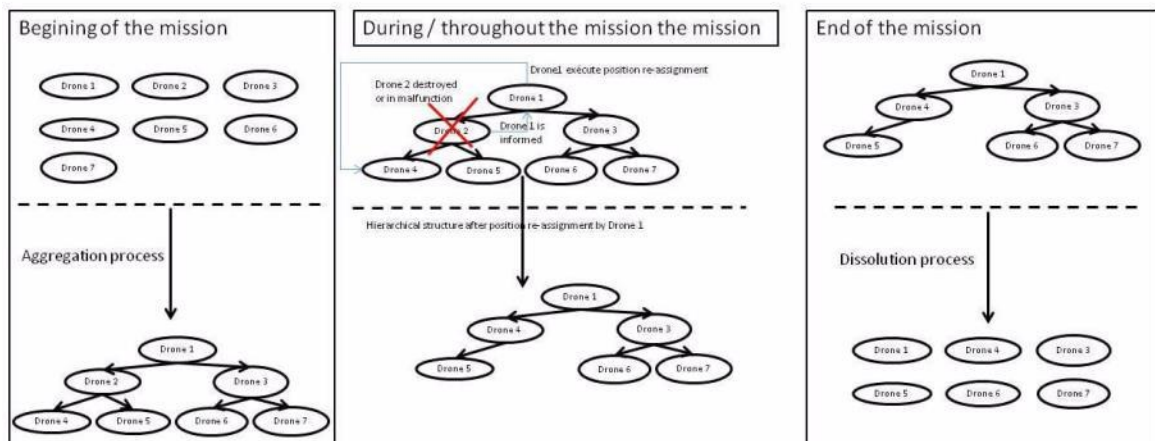


Figure 10. Modification of the hierarchical structure during the mission

The aggregation process while doing the mission is shown in figure 11 where drone 2 is destroyed and drone 1 become aware that drone 2 is down. Drone 1 then reassign drone 4 at the drone 2 hierarchical position with manual human interference or automatically with artificial intelligence heuristics so that the number of drones with leadership and higher command positions is maintained to keep a higher rate of

leadership and mission success because these two parameters are related and that the mission could achieve the maximum of success even if drones are destroyed or experience malfunctions during the mission (complex assessment task), as shown in **Figure 11**.

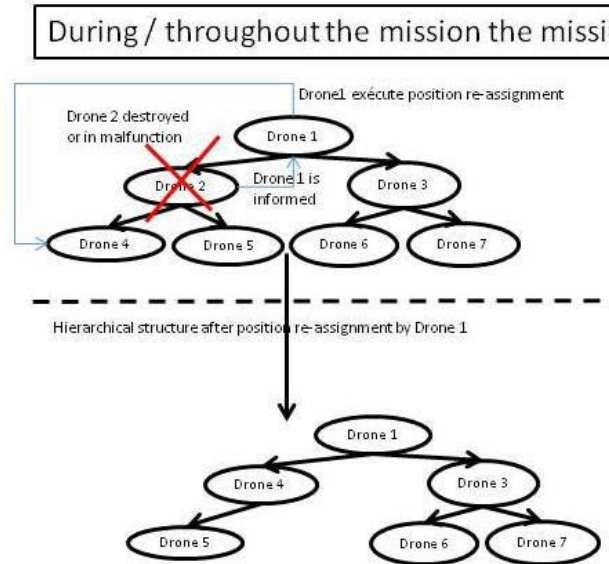


Figure 11. Modification of the hierarchical structure due to drones destruction or malfunction

The aggregation and hierarchical levels in a drone combat mission is shown in figure 23 where the hierarchical levels and the assessment parameters are described. Hierarchical levels are for team member, team leader and group manager: drone team member (team member), drone team leader (team leader) and drone group team manager that is a leader of several groups of drones (group manager). The assessment criteria could be mission assessment (% of mission done, % of target destroyed), drone group status (% drones destroyed, % drones malfunctioning) drone position guidance, target acquisition, target identification, target illumination, communication messages, drone hierarchical position reassignment, missile guidance, location attained, shots / missiles fired, as shown in **Figure 12**.

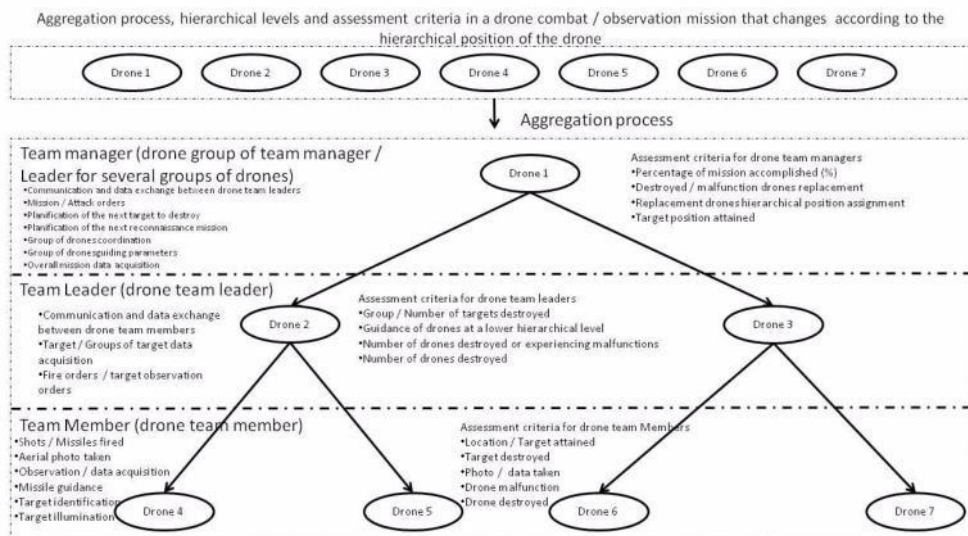


Figure 12. HAA process, hierarchical levels and assessment criteria in a drone mission

4.1 Aggregation process done manually by the drone pilot

The formation of drones in groups or teams with several levels of hierarchy could be done by a team of drone pilots. However, if these drone pilots have to work in teams, the pilots will have to share communication and decisions at a fast rate level if the drones come under attack and they have to replace broken drones. A lot of information will have to be analyzed in real time with a short notice. It might be preferable to use online computer systems to form drones in teams with specific algorithms or artificial intelligence heuristics due to the fast processing of the information and rapid decision making under attack.

2 Aggregation process with ontologies

In the education domain, a lot of experimentation, research, work and paper publication focused on teamwork assessment in the education field, most of it done with assessment grids. However, the assessment process is occurring at two levels of hierarchy and concerns a team leader assessing/evaluating some team members. What happened if the assessment has to be done on organizations of teams that have several/multiple levels of hierarchy and that the assessment process is implemented on computer systems residing in the Internet and also could use cloud computing, parallel processing, distributed processing and also assessment with Web semantics using ontologies. Hierarchical Aggregate Assessment (HAA) field is also very suited to assess transversal competencies and leadership soft skills at team leader and team manager levels of hierarchy. So an information system designed for assessment in computerized/Internet modes is a complex assessment task presentation engine that could use knowledge objects/assessment objects that could be coded in XML standards as SCORM and the complex assessment tasks presentation engine could use cloud computing, parallel processing, distributed processing and also assessment with Web semantics using ontologies. Drone guidance in the Hierarchical Aggregate Assessment paradigm resides in the aggregation process where drones are assigned a hierarchical position. Drone guidance ontology must have now a hierarchy because every drone hierarchical level is producing different data and has different assessment parameters, as stated in **Table 1**.

Hiera	D	Ex	Ontology
Team member	Drone team member	Conventional drone with team member guidance (done by drone pilot or by computer systems with parallel programming)	Drone team member Ontology
Team leader	Drone team leader	Conventional drone with team leader guidance (done by drone pilot or by computer systems with parallel programming and team leader aggregation)	Drone team leader Ontology
Team Manager	Drone group of team manager / Leader for several groups of drones	Conventional drone with team manager guidance (done by drone pilot or by computer systems with parallel programming and group of team manager aggregation)	Drone group of team manager / Leader for several groups of drones Ontology

Table 1. Hierarchical Aggregate Assessment (HAA) ontology hierarchy for drone guidance (Pryima, Strokan, Rogushina, Gladun, Lubko & Malkina, 2020).

Drone guidance in the Hierarchical Aggregate Assessment paradigm resides in the aggregation process where drones are assigned a hierarchical position. Drone guidance ontology must have now a hierarchy because every drone hierarchical level is producing different data and has different assessment parameters. The drone mission or complex

assessment task in authentic context can divide the task or mission competencies in trade hard skills that are essential to accomplish the mission and also the leadership soft skills that are implemented by the aggregation process either done manually or by artificial intelligence with ontologies, game theory, multi-agent systems and swarn intelligence, as stated in Table 2 (Pryima, Strokan, Rogushina, Gladun, Lubko & Malkina, 2020).

S	Desig	Desc
Trade Hard skills (essential to accomplish the mission)	- Data acquisition - Target destruction Drone location / position	- Data acquisition - Shots fired - Target acquisition / Target destroyed - Malfunction detection - Location / target attained
Leadership soft skills (implemented by the assessment process)	- Mission coordination - Drone group guidance - Broken drone replacement algorithms - Mission success assessment - Next target / Next mission planification	3 Mission / Attack orders 4 Communication between drones 5 Destroyed / broken drone replacement 6 Drone coordination 7 Mission realization assessment (% done)

Table 2. Hierarchical Aggregate Assessment (HAA) ontology hierarchy hard skills and soft skills assessment (Pryima, Strokan, Rogushina, Gladun, Lubko & Malkina, 2020).

The Hierarchical Aggregate Assessment (HAA) Paradigm could be use for drone competencies assessment in a mission using drone guidance. Competencies assessment could be realized by the presentation of a complex assessment task in authentic context. In the present case the complex assessment task in authentic context is a drone mission where a group of drones are formed in teams with multiple hierarchical positions. In the Hierarchical Aggregate Assessment (HAA) paradigm, assessment parameters vary in function of the hierarchical position. However, a baseline is common at all the assessment levels and is the trade hard skills that are essential to accomplish the mission. The grouping of drone in teams with several levels of hierarchy implies that drones that are at a higher level that team member could also be assessed on their leadership soft skills that are implemented by the assessment process and also could vary according if the drones have a lower or higher leadership position as shown in Figure 13 (Pryima, Strokan, Rogushina, Gladun, Lubko & Malkina, 2020).

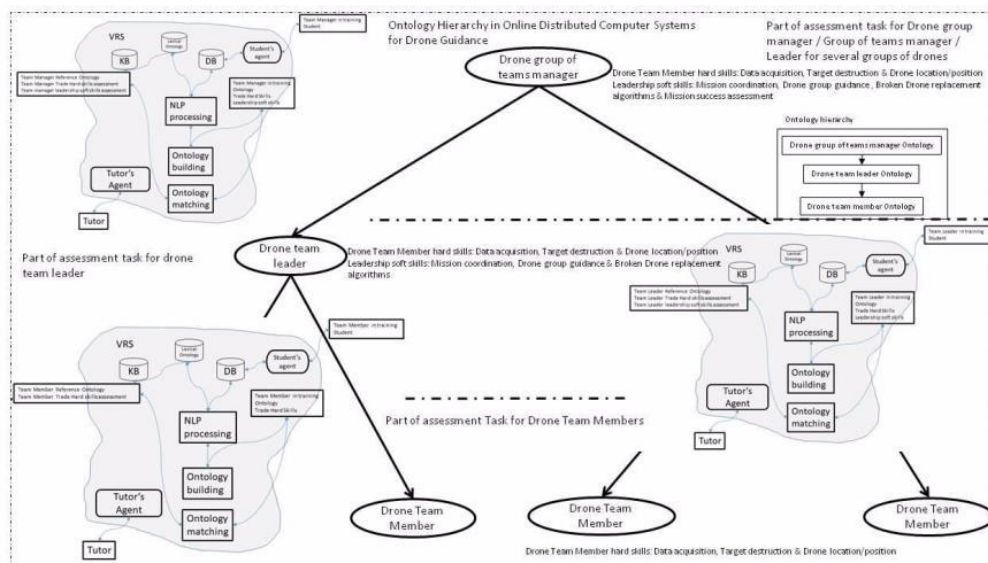


Figure 13. Ontology Hierarchy in Online Distributed Copmputer Systems for Drone Guidance (Pryima, Strokan, Rogushina, Gladun, Lubko & Malkina, 2020).

4.3 Aggregation process with Game theory

Game theory systems consists of players or agents that are evolving in a game environment. In drone guidance case, software players or agents are the drones and these players within the aggregation process are team member agents, team leader agents and team manager agents. The game is the drone mission, the players are the drones, the information is the targets, the drone locations, the drone hierarchical positions and the number of drones with malfunctions. Finally, the actions are the drone movements, the target acquisition and the shots fired, as shown in **Figure 14**.

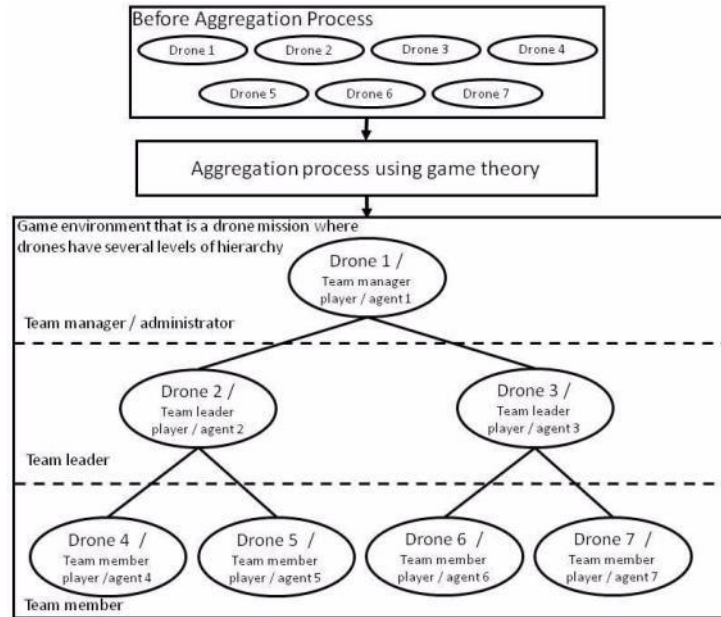


Figure 14. Aggregation process using game theory

4.4 Aggregation process with Multi-Agent systems

Multi-agent systems consists of agents and their environment. In drone guidance case, software agents are the drones and these agents within the aggregation process are team member agents, team leader agents and team manager agents. The agent environment is the drone mission, as shown in **Figure 15**.

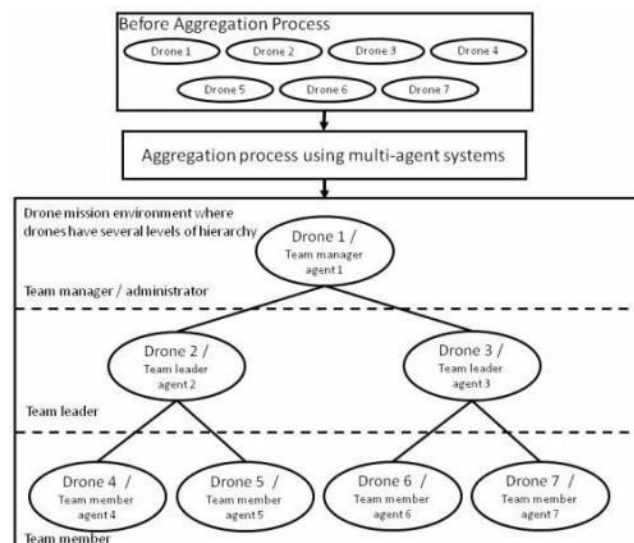


Figure 15. Aggregation process using multi-agent systems

4.5 Aggregation process with Swarn intelligence

Swarn intelligence systems consists of agents and their environment. In drone guidance case, software agents are the drones and these agents within the aggregation process are team member agents, team leader agents and team manager agents. The agent environment is the drone mission, as shown in **Figure 16**.

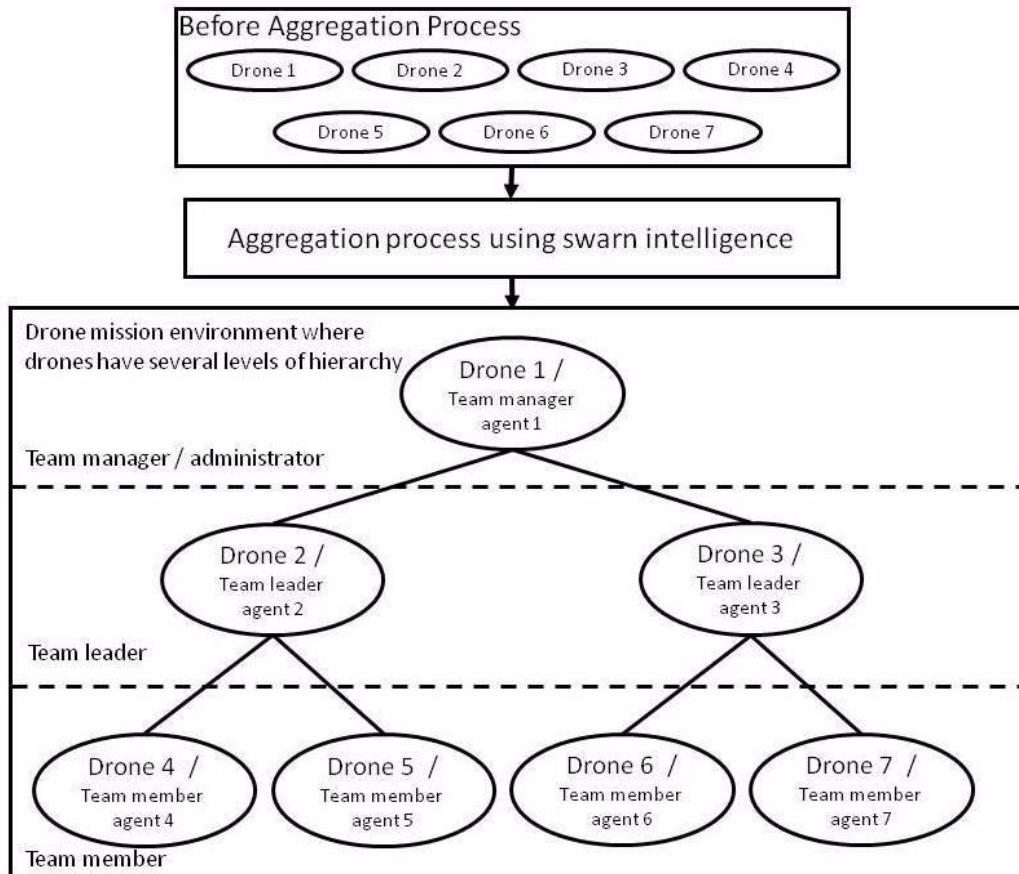


Figure 16. Aggregation process using swarn intelligence

5 Present and future challenges

This paper has shown that Hierarchical Aggregate Assessment (HAA) paradigm can encapsulates the aggregation process so that it could be isolated, modeled and be implemented in cloud computing, parallel programming and distributed computing that could use artificial intelligence paradigms as ontologies, game theory, multi-agent systems and swarn intelligence. However, this paper has only stated a theoretical framework. To implement these theories, the experimenters will have to choose an online server or computer and implement the aggregation process for online drone guidance in conventional or parallel programming that could include the artificial intelligence ontologies, game theory, multi-agent systems and swarn intelligence. A lot of work for the programming of these algorithms is to be done that could need funding for computer hardware and to pay the salary of the experimenters or the computer scientists that will work on that project. Once that the drone guidance system will be developed, this could lead to a lot of results and paper publishing.

6. Conclusion

This paper is a dissertation on the use of Hierarchical Aggregate Assessment (HAA) theory in online distributed computer systems for drone guidance. Conventional drone guidance methods could be manually operated with drones pilots or computerized. However, none of these methods are able to form drones into teams with several levels of hierarchy to perform a task. This paper has given a proof that Hierarchical Aggregate Assessment (HAA) paradigm can encapsulates the aggregation process so that it could be isolated, modeled and be implemented in cloud computing, parallel programming and distributed computing that could use artificial intelligence paradigms as ontologies, game theory, multi-agent systems and swarn intelligence. This research work could create writing material for a lot of scientific papers and also that the Hierarchical Aggregate Assessment (HAA) could be recognized worldwide.

References

1. Allal, L. (2002). Acquisition et évaluation des compétences en milieu scolaire. Dans J., Dolz, et E., Ollagnier (Dirs.). *L'énigme de la compétence en éducation* (p. 77-94). Bruxelles : de Boeck.
2. Blackboard (2013). *Blackboard Web Site*. [On Line] <http://www.blackboard.com>
3. Boud, D., Cohen, R. et Sampson, J. (1999). Peer Learning and Assessment. *Assessment & Assessment in Higher Education*, 24 (4), 413-426.
4. Burch, J. G., & Grudnitski, G. (1989). *Information Systems: Theory and Practice* (5th ed.). New York: John Wiley.
5. Comeaux, P., Huber, R., Kasprzak, J. et Nixon, M. A. (1998). *Collaborative learning in Web-based instruction*. Communication présentée à la conference WebNet 1998, Orlando, Floride.
6. Davis, G. B., & Olson, M. H. (1985). *Management Information Systems: Conceptual Foundations, Structure, and Development*. New York: McGraw-Hill.
7. Davis, G. B., Olson, M. H., Ajenstat, J., & Peaucelle, J. L. (1986). *Information Systems for Management. Volume I: Thebases*. Boucherville: G. Vermette Inc. Editors.
8. Eisen, M. B., Spellman, P. T., Browm, P. O., & Botstein, D. (1998). Cluster Analysis and Display of Genome-Wise Expression Patterns. *Proceedings of the National Academy of Sciences of the United States of America*, 95, 14863-14868. <http://dx.doi.org/10.1073/pnas.95.25.14863>
9. Endrizzi, L. et Rey, O. (2008). *L'évaluation au cœur des apprentissages*. Dossier d'actualité n°39, Service de veille scientifique et technologique INRP. Accès : <http://www.scribd.com/doc/8574511/lassessment-au-coeur-des-apprentissages>.
10. Fidalgo-Blanco, A., Sein-Echaluce, M. L., Garcfa-PeFialvo, F. J., et Conde, M. A. (2015). Using Learning Analytics to improve teamwork assessment. *Computers in Human Behavior*, 47, 149- 156.
11. Fraley, C., & Raftery, A. E. (1998). How Many Clusters? Which Clustering Method? Answers via Model-Based Cluster Analysis. *The Computer Journal*, 41, 578-588. <http://dx.doi.org/10.1093/comjnl/41.8.578>
12. Freeman, M., & McKenzie, J. (2000). Self and Peer Assessment of Student Teamwork: Designing, Implementing and Eva-luating SPARK, a Confidential, Web Based System [On Line]. In L. Richardson, & J. Lidstone (Eds.), *Flexible Learning for a Flexible Society, Proceedings of ASET-HERDSA 2000 Conference*. University of

Southern Queensland, Toowoomba: ASET and HERDSA. <http://ascilite.org/archived-journals/aset/confs/aset-herdsa2000/procs/freeman.html>

13. Freeman, M., & McKenzie, J. (2002). SPARK, a Confidential Web-Based Template for Self and Peer Assessment of Student Teamwork: Benefits of Evaluating across Different Subjects. *British Journal of Educational Technology*, 33, 551-569. <http://dx.doi.org/10.1111/1467-8535.00291>

14. Gladun, A. (2022). *Ontological analysis of outcomes of non-formal and informal learning: agro-advisory system AdvisOnt – Paper & PowerPoint presentation*

15. Gladun, A., Rogushina, J. Garcia-Sanchez, F., Martinez-Bejar, R. & Fernandez-Breis. (date). *An application of intelligent techniques and semantic Web technologies in e-learning environments*

16. Gladun, A., Rogushina, J. (date). *An Ontology-based approach to student skills in multiagent E-Learning systems*

17. Gladun, A., Rogushina, J. (date). *Use of semantic Web technologies and multilingual thesauri for knowledge-based access to biomedical resources.*

18. Goldfinch, J., et Raeside, R. (1990). Development of a Peer Assessment Technique for Obtaining Individual Marks on a Group Project. *Assessment & Assessment in Higher Education*, 15 (3), 210-231.

19. Goldfinch, J. (1994). Further Developments in Peer Assessment of Group Projects. *Assessment & Assessment in Higher Education*, 19 (1), 29-35.

20. Gijbels, D., Dochy, P., Van Den Bossche, P. et Segers, M. (2005). Effects of Problem-Based Learning: A Meta-Analysis from the Angle of Assessment. *Review of Educational Research*, 75 (1), 27-61.

21. Harvey, S. et Loisele, J. (2009). Proposition d'un modèle de recherche-développement. *Recherches qualitatives*, 28 (2), 95-117.

22. Havyer, R. D., Wingo, M. T., Comfere, N. I., Nelson, D. R., Halvorsen, A. J., McDonald, F. S., et Reed, D. A. (2014). Teamwork assessment in internal medicine: a systematic review of validity evidence and outcomes. *Journal of General Internal Medicine*, 29 (6), 894-910.

23. Howe, R. (1991). Formules pédagogiques et évaluation formative: une combinaison gagnante. *Pédagogie collégiale*, 4 (4) 8-13.

24. Illingworth, V. (Ed.) (1996). *Dictionary of Computing*. New York: Oxford University Press.

25. Kanter, J. (1984). *Management Information Systems* (3rd ed.). Upper Saddle River, NJ: Prentice Hall.

26. Kaye, W. & Mancini, M. E. (1986). Use of the Mega Code to evaluate team leader performance during advanced cardiac life support. *Critical Care Medicine*, 14 (2), 99-104.

27. Ketchen Jr, D. J., & Shook, C. L. (1996). The Application of Cluster Analysis in Strategic Management Research: An Analysis and Critique. *Strategic Management Journal*, 17, 441-458. [http://dx.doi.org/10.1002/\(SICI\)1097-0266\(199606\)17:6<441::AID-SMJ819>3.0.CO;2-G](http://dx.doi.org/10.1002/(SICI)1097-0266(199606)17:6<441::AID-SMJ819>3.0.CO;2-G)

28. Laudon, K. C., & Laudon, J. P. (2000). *Management Information Systems: Organization and Technology in the Networked Enterprise* (6th ed.). Upper Saddle River, NJ: Prentice Hall.

29. Laudon, K. C., Laudon, J. P., & Brabston, M. E. (2011). *Management Information Systems: Managing the Digital Firm* (5th Canadian ed.). Toronto, ON: Pearson Education Canada Inc.

30. Legendre, R. (2005). *Dictionnaire actuel de l'éducation* (3e ed.). Montréal : Guérin.

31. Lingard, R. W. (2010). Teaching and assessing teamwork skills in engineering and computer science. *Journal of Systemics, Cybernetics and Informatics*, 18 (1), 34-37.
32. Louis, R. et Bernard, H. (2004). *L'évaluation des apprentissages en classe : théorie et pratique*. Montréal : Groupe Beauchemin, éditeur ltée. Marshall-Mies, J. C., Fleishman, E. A., Martin, J. A., Zaccaro, S. J., Baughman, W. A., & McGee, M. L. (2000). Development and Assessment of Cognitive and Metacognitive Measures for Predicting Leadership Potential. *The Leadership Quarterly*, 11, 135-153. [http://dx.doi.org/10.1016/S1048-9843\(99\)00046-6](http://dx.doi.org/10.1016/S1048-9843(99)00046-6)
33. Moodle (2013). *Moodle Web Site*. [On Line] <http://www.moodle.org> MySQL (2013). *MySQL Web Site*. [On Line] <http://www.mysql.com>
34. Nance, W. D. (2000). Improving Information Systems Students' Teamwork and Project Management Capabilities: Experiences from an Innovative Classroom. *Information Technology and Management*, 1, 293-306. <http://dx.doi.org/10.1023/A:1019137428045>
35. Northrup, S. G. et Northrup, D. A. (2006). *Multidisciplinary teamwork assessment: individual contributions and interdisciplinary interaction*.
36. Communication présentée à la 36ième conférence ASEE/IEEE Frontiers in education, Octobre 28-31, San Diego.
37. Olivier, L. (2002). *L'évaluation des apprentissages* [En ligne]. Commission Scolaire de Montréal. Accès: http://www.csdm.qc.ca/Csdm/Administration/pdf/reforme_vol1_no2.pdf Palm, T. (2008). *Performance Assessment and Authentic Assessment: A Conceptual Analysis of the Literature* [En Ligne]. Practical Assessment, Research & Assessment, 13 (4). Accès: <http://pareonline.net/pdf/v13n4.pdf>.
38. Petkovic, O., Thompson, G., Todtenhofer, R., Huang, S., Levine, B., Parab, S., & Shrestha, S. (2010, October). Work in progress-e-TAT: Online tool for teamwork and "soft skills" assessment in software engineering education. Dans *Frontiers in Education Conference (FIE)*, 2010 IEEE (pp. S1G-1). IEEE.
39. Pryima, S. M., Rogushina, J. V. & Strokan, A. V. (2021). *Use of semantic technologies in the process of recognizing the outcomes of non-formal and informal learning*. CEUR-WS.org/Vol-2139/226-235
40. Pryima S.M., Strokan, O. V., Rogushina, J. V., Gladun, A. Y., Lubko, D. V. & Malkina, V. M. (2020). Ontological Analysis of Outcomes of Non-formal and Informal Learning for Agro-Advisory System AdvisOnt. *CITI 2020*, (3) 17.
41. Rogushina, J., Gladun, A., Pryima, S. & Strokan, O. (2019). *Ontology-based Approach to Validation of Learning Outcomes for Information Security Domain*. CEUR-WS.org/Vol-2577/Paper3.pdf
42. Rogushina, J., Gladun, A. & M. Salem A.-B. (date). *Usage of Ontology-based semantic analysis of complex information objects in virtual research environments*
43. Rogushina, J., Gladun, A., Shreurs, J. & Salem A.-B. (date). *Use of Ontological analysis as an instrument of knowledge recognition in complex virtual research environments*
44. Saskatchewan Professional Development Unit (2003). *Performance Assessments : A wealth of possibilities* [En ligne]. Accès: http://www.sasked.gov.sk.ca/branches/aar/afl/docs/assessment_support/perfassess.pdf
45. Scallon, G. (1999). *L'évaluation sommative et ses rôles multiples* [En Ligne]. Accès: <http://fse.ulaval.ca/gerard.scallon/fascicules/sommative.htm> Scallon, G. (1999). *L'harmonisation des fonctions formative et sommative de l'évaluation* [En Ligne]. Accès: <http://fse.ulaval.ca/Gerard.Scallon/fascicules/harmoni.pdf>

46. Sigler, T. H., et Rhee, K. S. (2014). Unlocking learning: Discovering the keys to effective assessment. *Journal of Management Education* 38 (3), 303-312.
47. Tardif, J. (2006). *L'évaluation des compétences. Documenter le parcours de développement*. Montréal : Chenelière Éducation.
48. Undre, S., Sevdalis, N., Healey, A. N., Darzi, S. A., & Vincent, C. A. (2007). Observational Teamwork Assessment for Sur-gery (OTAS): Refinement and Application in Urological Surgery. *World Journal of Surgery*, 31, 1373-1381. <http://dx.doi.org/10.1007/s00268-007-9053-z>
- Van Zyl et Massyn, L. (2011). Integrated Assessment: A Learning Adventure And Growth Opportunity For Adult Learners [En Ligne]. *American Journal Of Business Education (AJBE)*, 1 (2), 95-104.
49. Acces:
<http://cluteonline.com/journals/index.php/AJBE/article/view/4628/4717>. Wiggins, G. (1990). The case for authentic assessment [En Ligne]. *Practical Assessment, Research & Assessment*, 2 (2). Accès: <http://PAREonline.net/getvn.asp?v=2&n=2>.
50. Willey, K. et Freeman M. (2006). *Completing the learning cycle: The role of formative feedback when using self and peer assessment to improve teamwork and engagement*. Proceedings of the 17th Annual Conference of the Australasian Association for Engineering Education, 10 -13 décembre 2006, Auckland, New Zealand.
51. Willey, K., et Freeman, M. (2006). Improving teamwork and engagement: the case for self and peer assessment [En Ligne]. *Australasian Journal of Engineering Education*, 02. Accès : <http://www.aace.com.au/journal/2006/willey0106.pdf>.
52. Yilmaz, H., et Cetinkaya, B. (2007). Using an online portfolio course in assessing students' work. *The Turkish Online Journal of Educational Technology (TOJET)*, 6 (4), 1-6.

Технологічний процес буде розроблятися для виготовлення деталі «Втулка» (рис. 2), яка відноситься до простих деталей з погляду технологічності.

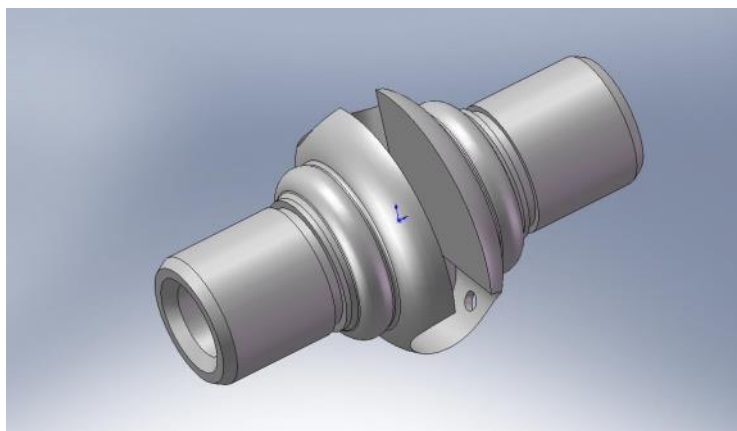


Рисунок 2. Зовнішній вигляд тривимірної моделі деталі «Втулка»

Особливістю деталі є відповідальними внутрішній паз, що вимагає високої точності та низької шорсткості. Складність виготовлення пов'язана з кількома причинами: трудомісткий, глибокий. Розроблена технологія завдяки FeatureCAM є оптимальною та високопродуктивною для обробки даного паза.

Технологічний процес обробки:

Заготовка - пруток круглого перерізу діаметром 110 мм згідно з ГОСТ 21488-97.

Операція 005 «Токарна з ЧПУ». Для її виконання застосовуємо токарно-гвинторізний верстат Spinner TC 400, який є у наявності.

Для виготовлення деталі підібрано комплект різального інструменту для обробки сталей фірми Sandvik Coromant та розраховано режими різання за каталогом.

При розробці технологічного процесу (ТП) та керуючих програм (УП) для обладнання з ЧПУ на кафедрі «Інженерна механіка та комп'ютерне проектування» застосовують різні методи комп'ютерного проектування – CAD/CAM – системи, такі як:

- SolidWorks 2009 (Компанія «SolidWorks») – ядро інтегрованого програмного комплексу автоматизації підприємства, за допомогою якого здійснюється підтримка життєвого циклу виробу відповідно до концепції CALS-технологій.

- Feature CAM 2009 (Компанія «Delcam plc.») – системи автоматизованої підготовки програм для верстатів із ЧПУ.

Всі ці програмні продукти скорочують час розробки технологічного процесу та керуючих програм (УП) для верстата з ЧПУ, підвищують продуктивність роботи інженерів-технологів та операторів з ЧПУ, а також знімають зайве навантаження. Впровадження даних програм виробництва дозволить скоротити витрати виробництва, пов'язані з невиправданим простоєм дорогих сучасних металорізальних верстатів.

Також, однією з основних важливих особливостей FeatureCAM – є можливість автоматизованого складання програм для типових елементів, тобто, при завантаженні моделі в систему, вона, на основі вбудованих технологічних знань, автоматично вибирає інструмент, режими та стратегію обробки. Далі

програміст переглядає отриманий результат, за необхідності вносить зміни та відправляє кінцеву програму на верстат із ЧПУ.

Програмний продукт Feature CAM чудово зістикується із CAD-системами такі, як Компас-3D, SolidWorks, AutoCAD та іншими. FeatureCAM дозволяє імпортувати моделі всіх форматів (наприклад, IGES, DWG/DXF, STEP, VDA, 3D Parasolid XMT files), що робить її універсальною і затребуваною на ринку CAD/CAM-систем.

Текст УП зберігається в електронному вигляді, а також відображається в інтерфейсі FeatureCAM.

Послідовність розробки керуючої програми у FeatureCAM:

1. Створюємо або імпортуємо твердотільну модель деталі (рис. 3);

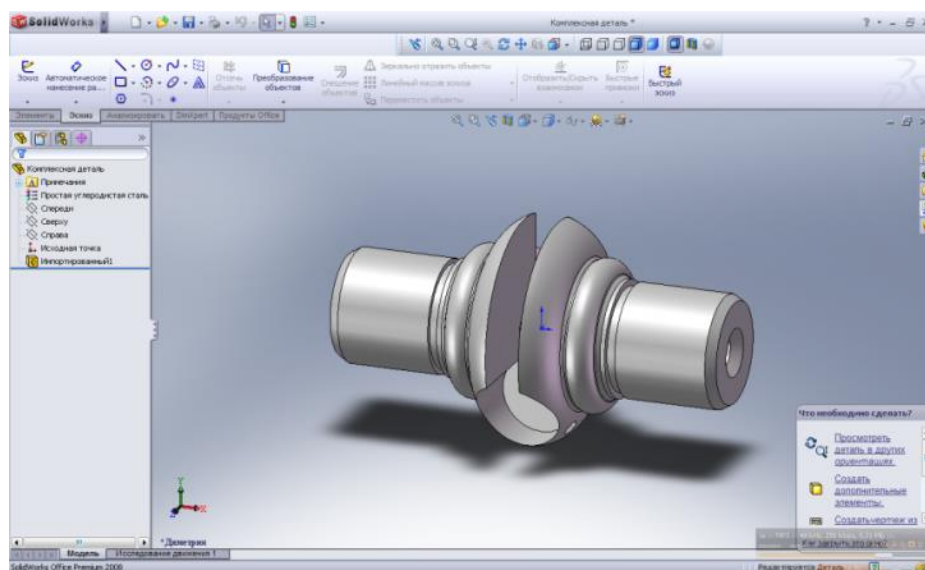


Рисунок 3. Твердотільна модель комплексної деталі

2. Задаємо властивості заготовки (рис. 4). Тип заготовки - прокат круглого перерізу діаметром 72 мм. Залишаємо припуски по діаметру та торцям.

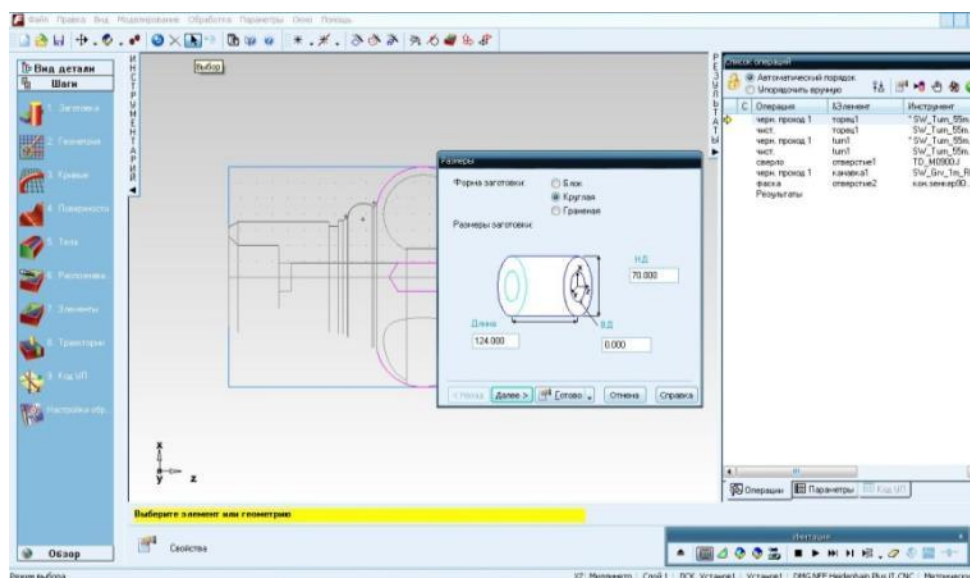


Рисунок 4. Діалогове вікно завдання властивостей заготовки

3. Задаємо інструмент, що використовується;
4. Створюємо першу робочу систему координат так, як потрібно для обробки у першій позиції (вісь X – вздовж осі заготівлі);
5. Призначаємо режими різання. Встановимо необхідну швидкість різання, частоту обертання шпинделя та глибину різання.
6. Зробимо чорнову, а потім чистову обробку поверхні. Використовуємо чорновий різець з кутом при вершині 55 градусів начорно, а потім прохідний чистий різець з кутом при вершині ріжучої кромки рівної 35 градусів.
7. Далі проводиться обробка глибокого паза кінцевим інструментом за допомогою приводного інструменту.

Взаємодіючи з базою знань та послідовністю роботи у FeatureCAM призначаємо всі необхідні параметри для обробки: атрибут обробки, вибір різального інструменту, послідовність обробки, призначення матеріалів та режимів різання.

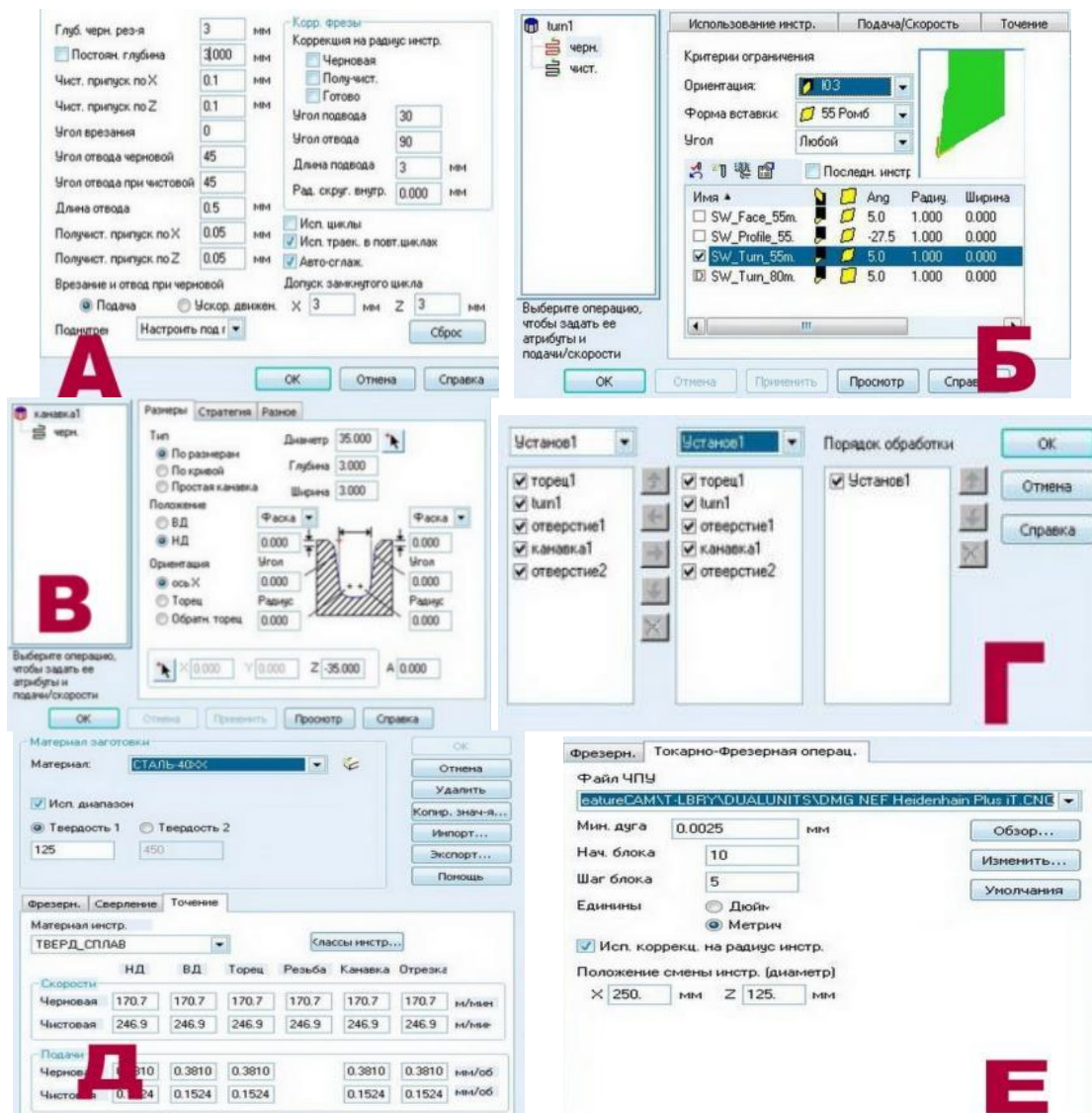


Рисунок 5. Взаємодія з базою знань та послідовності роботи у Feature CAM:
А – завдання атрибутів обробки; Б – менеджер вибору інструмента; В – призначення параметрів обробки; Г – завдання послідовності виконання операцій;
Д – призначення матеріалів та режимів різання;
Е – вибір постпроцесора та генерація УП.

Отримана траєкторія руху інструмента показано на Рисунок 6.

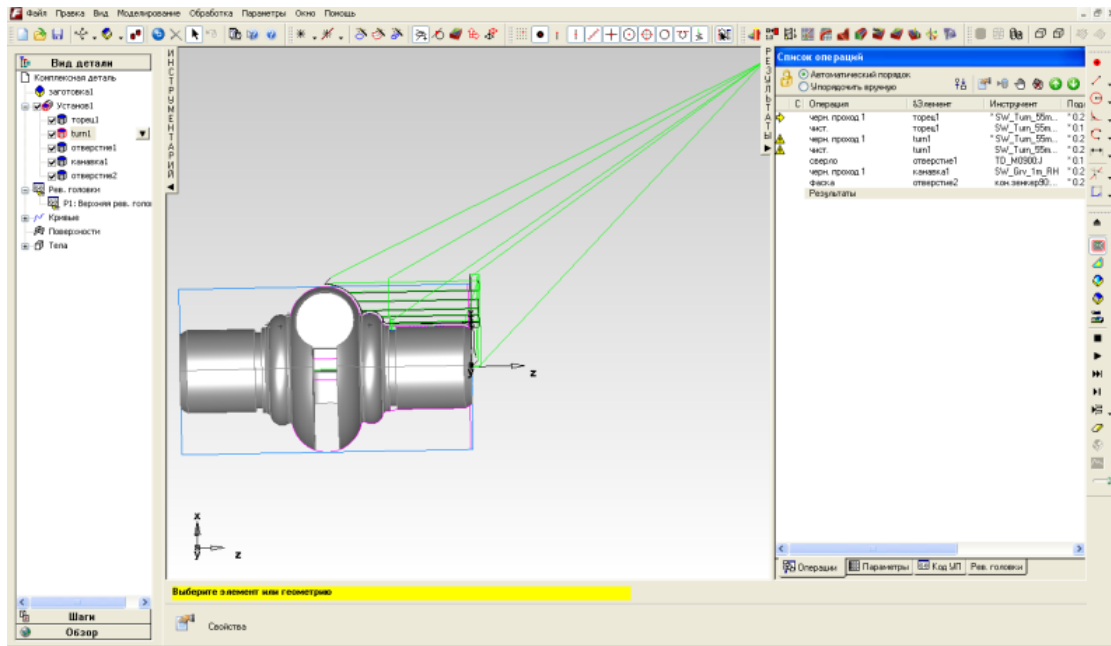


Рисунок 6. Траєкторія чорнової обробки деталі.

Нижче представлена частина технологічного процесу обробки (рис. 7).

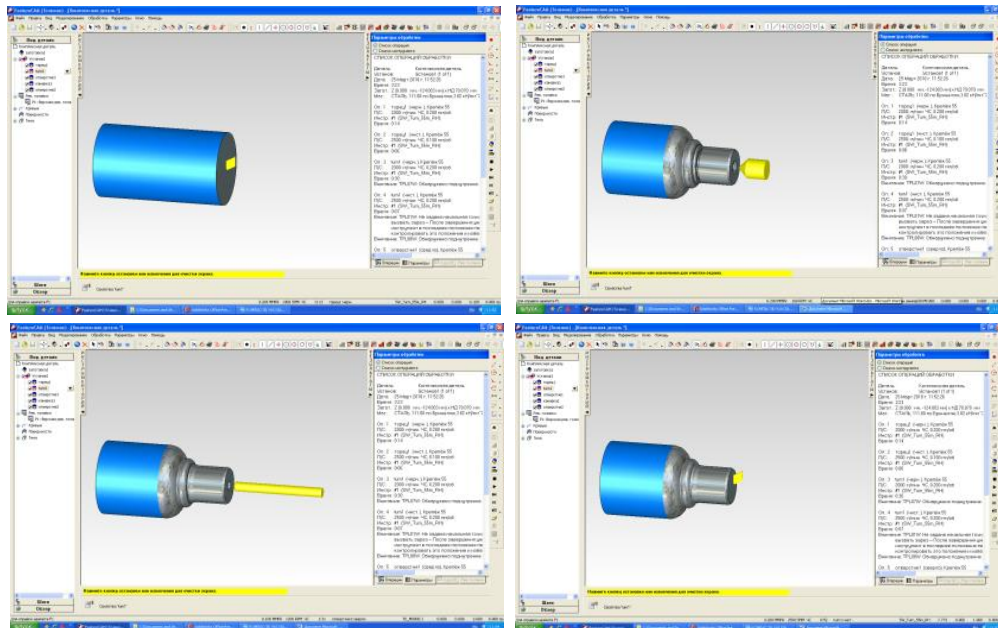


Рисунок 7. Фрагмент обробки деталі в САМ-системі FeatureCAM

Нижче представлена фотографія деталі, після чистової обробки з обох боків (рис. 8).

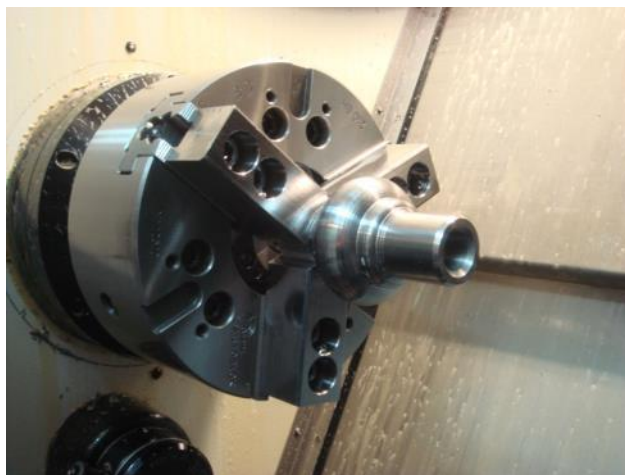


Рисунок 8. Фотографія виробу після завершення токарної операції

Висновки. В результаті виконаної роботи за допомогою програмного продукту Feature CAM 2009, а також SolidWorks 2009 було отримано:

- Комп'ютерну модель комплексної деталі;
- Розроблено технологію механообробки та на основі моделі створено управляючу програму (УП) для обробки на верстатах з ЧПУ;
- Оформлено технологічну документацію згідно з ЄСКД, ЄСТД.

Ці програми були випробувані в умовах лабораторій кафедри «Інженерна механіка та комп'ютерне проектування» Таврійського державного агротехнологічного університету імені Дмитра Моторного та отримано дослідний зразок деталі.

Завдяки створеній базі знань FeatureCAM та інтуїтивно зрозумілому інтерфейсу, працювати в цьому пакеті швидко та легко, що, у свою чергу, дозволяє інженеру-технологу та оператору з ЧПУ оперативно створювати та коригувати УП.

Список використаних джерел:

1. Гавриленко Е.А., Холодняк Ю.В., Антонова Г.В., Чаплинский А.П. Разработка алгоритма программного обеспечения для формирования обводов по заданным геометрическим условиям. *Праці Таврійського державного агротехнологічного університету імені Дмитра Моторного.* – Мелітополь: ТДАТУ ім. Д. Моторного, 2020. – Вип. 20, т. 3. (с.293-303).-DOI: 10.31388/2078-0877-2020-20-3-293-303.

2. Мацулевич О.Є., Щербина В.М., Бондаренко Л.Ю., Малюта С.І., Антонова Г.В. Програмне забезпечення для автоматизованого визначення параметрів різального інструменту фрезерної обробки корпусних деталей. *Праці Таврійського державного агротехнологічного університету.* Мелітополь: ТДАТУ, 2020. Вип. 20, т. 3. С. 275-281.

3. Інженерна та комп'ютерна графіка. Частина 1: навчальний посібник / В.М. Щербина, О.Є. Мацулевич, Є.А. Гавриленко, Ю.В. Холодняк, О.В. Івженко, І.В. Пихтєєва, О.О. Вершков, С.В. Галько, А.П. Чаплінський. Мелітополь: Люкс, 2020. 238 с.

4. Проектування керуючих програм для верстатів токарної групи з пристроєм числового програмного керування: навчальний посібник / Дмитрієв

Ю.О., Мацулевич О.Є., Щербина В.М., Холодняк Ю.В. Мелітополь, Люкс, 2018. 132 с.

5. Теоретична механіка. Динаміка матеріальної точки: навчально – методичний посібник / О.М. Леженкін, Г.В. Антонова, О.О. Вершков, Л.Ю. Бондаренко, О.Є. Мацулевич, А.О. Смєлов, О.Ю. Михайленко. Мелітополь: Люкс, 2021. 160с.

6. Інженерна механіка: практикум / Г.В. Антонова, О.Є. Мацулевич, О.Ю. Михайленко, І.В. Пихтєєва, О.В. Івженко, Ю.В. Холодняк, В.М. Щербина, Ю.О. Дмитрієв. Мелітополь: Люкс, 2021. 147 с.

7. Нарисна геометрія та креслення: навчально–методичний посібник для підготовки бакалаврів зі спеціальностей 131 “Прикладна механіка” та 133 “Галузеве машинобудування” / Івженко О.В., Пихтєєва І.В., Гавриленко Є.А., Мацулевич О.Є., Щербина В.М., Холодняк Ю.В., Бондаренко Л.Ю., Михайленко О.Ю. Мелітополь: Люкс, 2020. 217 с.

8. Програмування автоматизованих процесів обробки деталей: навчально–методичний посібник. Лабораторний практикум /Ю.О. Дмитрієв, О.Є. Мацулевич, Є.А. Гавриленко, Ю.В. Холодняк, Г.В. Антонова – Мелітополь (протокол №8 від 28.04.2022р.), 2022.–170с.

9. Мацулевич О.Є., Вершков О.О., Холодняк Ю.В., Дмитрієв Ю.О., Чаплінський А.П.Розробка мурашиного алгоритму для оптимізації оперативного планування робіт по збиранню врожаю кісточкових. *Плодовий сад – новітнє в теорії та практиці*: матеріали VВсеукр. наук.-практ. інтернет-конференції. Мелітополь: ТДАТУ, 2021. С. 106-110.

10. Спірінцев В.В., Мацулевич О.Є., Холодник Ю.В, Чаплінський А.П. Застосування графічного редактора ARCHICAD при вивченні дисципліни «Комп'ютерне проектування простору інженерних споруд» *Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації*: матеріали Міжнар. наук.-практ. інтернет-конф. Мелітополь: ТДАТУ, 2020. С. 262-266.

УДК 004.9

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗОВАНОГО РОЗРАХУНКУ РЕЖИМІВ РІЗАННЯ ПРИ ОБРОБЦІ ДЕТАЛЕЙ НА СВЕРДЛИЛЬНИХ ВЕРСТАТАХ

Дережа О.О., к.т.н., доцент
Антонова Г.В., ст. викладач
Тетервак І.Р., асистент
Таврійський державний агротехнологічний університет імені Дмитра
Моторного

e-mail: olena.dereza@tsatu.edu.ua
e-mail: galina.antonova@tsatu.edu.ua
e-mail: illia.tetervak@tsatu.edu.ua

Актуальність дослідження та постановка проблеми. На підприємствах машинобудування на сьогодні активно впроваджуються підсистеми автоматизованого проектування технологічних процесів.

В структурі металорізючого обладнання підприємств свердлильні верстати займають 20%.

В наявних пакетах програм для автоматизованого проектування відсутній модуль для розрахунку режимів різання свердел. У зв'язку з цим на підприємствах стає задачею проектування програмного модуля для розрахунку режимів різання свердел.

Основними технологічними способами обробки отворів різного ступеня точності і з різною шорсткістю обробленої поверхні є свердління, зенкування і розгортання.

Свердлінням отримують наскрізні і глухі циліндричні отвори. Шорсткість поверхні після свердління $R_a = 12,5 - 6,3$ мкм, точність по 11-14 квалітету. Отвори діаметром більше 30 мм в суцільному матеріалі зазвичай свердлять двома свердлами (перше - діаметром 0,2-0,4 заданого, друге - в розмір отвору).

Розсвердлювання спіральним свердлом виробляють для збільшення діаметра отвори. Діаметр отвору під розсвердлювання вибирають таким чином, щоб поперечна ріжуча кромка в роботі не брала участь.

Зенкування - технологічний спосіб обробки попередньо просвердлених отворів або отворів, виготовлених литтям або штампуванням. Точність зенкування 10-11 квалітет, шорсткість поверхні $R_a = 6,3 - 3,2$ мкм. Зенкування може бути і остаточною операцією при обробці просвердлених отворів по 11-13 квалітетами або для напівчистої обробки перед розгортанням.

Розгортанням отримують отвори підвищеної точності (5-7 квалітет) з низькою шорсткістю до $R_a = 0,4$ мкм. Розгортають циліндричні і конічні отвори. Для розгортання конічних отворів циліндричні отвори в заготівлі спочатку обробляють конічним ступінчастим зенкером, потім конічної розгорткою зі стружко-розділювальними канавками і остаточно - конічної розгорткою (рис. 1) з гладкими ріжучими крайками.

Отвори діаметром до 10 мм розгортають після свердління, а понад 10 мм - після свердління і зенкування. Перед розгортанням необхідно ретельно обробити торцеву поверхню деталі, щоб розгортка увійшла в отвір без перекосу.

При розгортанні велике число зубів одночасно бере участь в різанні. Розгортання характеризується невеликою глибиною різання ($t = 0,05 - 0,3$ мм), що сприяє малої шорсткості й високій якості обробки.

Зенкуванням обробляють циліндричні і конічні поглиблення під головки болтів і гвинтів. Обробку ведуть зенкерами спеціальної конструкції, званими

зенківки. Деякі зенківки мають напрямну частину, яка забезпечує співвісність поглиблення та основного отвори.

Цекуванням обробляють торцеві площини, які є опорними поверхнями головок гвинтів, болтів, гайок. Перпендикулярність торця основного отвору досягається наявністю направляючої частини у цековки.

Нарізування різьби виробляють мітчиком.

Комбінованим інструментом отримують складні поверхні.

При свердлінні, зенкеруванні і розгортанні зазвичай різальному інструменту повідомляють головний рух різання - обертальний рух ріжучого інструменту і рух подачі - осьове переміщення різального інструменту. При нарізуванні різьблення мітчиками інструмент отримує тільки обертальний рух, а примусова подача відсутня, тому мітчик – інструмент, який подається самостійно.

Елементами режиму різання при осьовій обробці отворів є:

- Швидкість різання V - це окружна швидкість точки різальної крайки, найбільш віддаленої від осі інструменту

$$V = \frac{\pi D n}{1000}, \text{ м / хв,}$$

де: D - діаметр інструменту, мм; n - частота обертання інструменту, об / хв;

- подача S дорівнює величині переміщення інструменту уздовж осі в мм за один оберт;

- глибина різання t (мм) при свердлінні дорівнює половині діаметра свердла $t = 0,5D$, а при розсвердлюванні, зенкеруванні, розгортанні $t = 0,5(D - d)$.

Основні матеріали дослідження. Для обробки деталей на свердлильних верстатах застосовують свердла, зенкери, розгортки, мітчики і комбіновані інструменти.

Свердла по конструкції поділяють на спіральні, центрові та спеціальні. Найпоширеніші з них спіральні, призначені для свердління і розсвердлювання отворів, глибина яких не перевищує 10 діаметрів свердла.

Конструкція спірального свердла представлена на рисунку 2. Робоча частина свердла має дві спіральні канавки і закінчується збірними конусом - ріжучою частиною. У перетині гвинтових канавок з конусом (передній і головний задній поверхні) утворюються дві головні ріжучі кромки, що виконують основну роботу різання.

Головні ріжучі кромки при сполученні один з одним утворюють поперечний лезо - перемичку (допоміжна ріжуча кромка). Перемичка розташовується відносно головних різальних крайок під кутом $\varphi = 50 - 55^\circ$ і ріже метал з ускладненням. У зв'язку з цим отвори діаметром більше 30 мм просвердлюють в два прийоми. Спочатку свердлять отвір діаметром, який трохи перевищує довжину перемички свердла, а потім отвір розсвердлюється до необхідного діаметра.

Для зменшення тертя направляючої частини свердла об стінки просвердлюваного отвору його діаметр має змінний перетин, зменшуючийся до хвостовика. У цих же цілях зовнішня поверхня направляючої частини свердла профрезерована і залишені дві виступаючі стрічки, розташовані вздовж гвинтових канавок. Кромки стрічок зачищають циліндричну поверхню просвердлюваного отвору, тому їх вважають допоміжними ріжучими крайками. Таким чином, у спірального свердла є п'ять різальних крайок - дві головні, поперечна і дві допоміжні. Дві головні ріжучі кромки утворюють кут при вершині (кут в плані) .

Для свердління м'яких матеріалів $\alpha = 80-90^\circ$, для твердих і крихких $\alpha = 130-140^\circ$. Стандартні свердла розраховані на свердління конструкційних сталей і мають кут $\alpha = 116-118^\circ$. Кут нахилу гвинтової канавки визначає положення передньої поверхні свердла, а, отже, і передній кут різання.

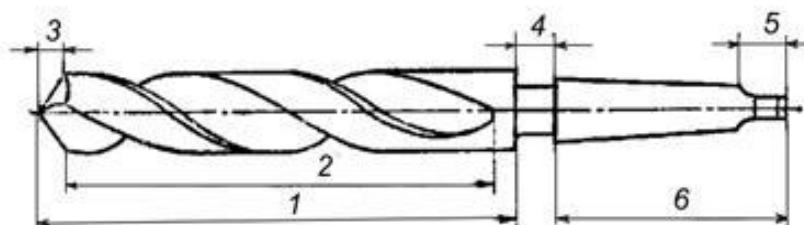


Рисунок 1. Конструкція спірального свердла

1 - робоча частина; 2 - напрямна частина; 3 - ріжуча частина; 4 - шийка; 5 - лапка; 6 - хвостовик (конічний або циліндричний).

Робоча частина розгортки (рис. 3) складається з вхідного конуса, ріжучої і калібрує частин.

Калібруюча частина складається з циліндричної і конусної частини. Циліндрична частина служить для направлення розгортки в процесі різання і калібрування отвори. Зворотний конус (3-6 мкм) робиться для зменшення тертя розгортки про оброблену поверхню і зменшення розбивки отвору.

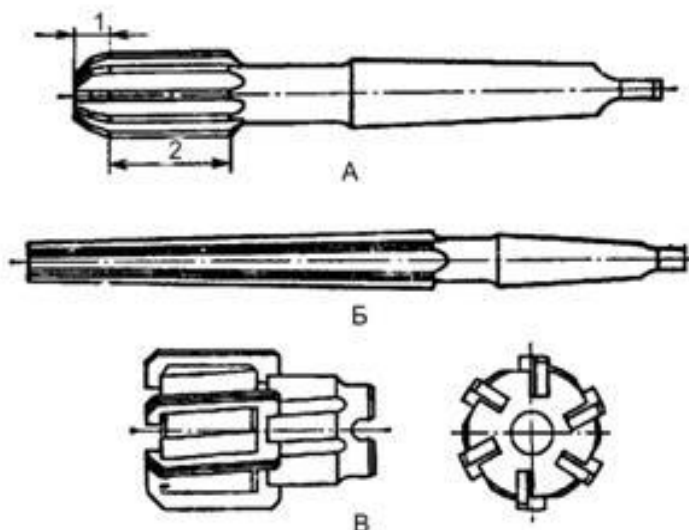


Рисунок 2. Розгортки: А- циліндрична; Б конічна; В- машинна насадна

Мітчики застосовують для нарізування внутрішніх різьблень. Вони являють собою гвинт з прорізними прямими або спіральними канавками, що утворюють ріжучі леза, і складаються з робочої і хвостової частин (рис. 4).

Робоча частина мітчика має ріжучу і калібруючу частину. Ріжуча частина виробляє основну роботу по нарізці різьблення, а калібруюча зачищає її.

Профіль різьби мітчика повинен відповідати профілю нарізаємої різьби. Розрізняють гайкові, машинні і ручні мітчики.

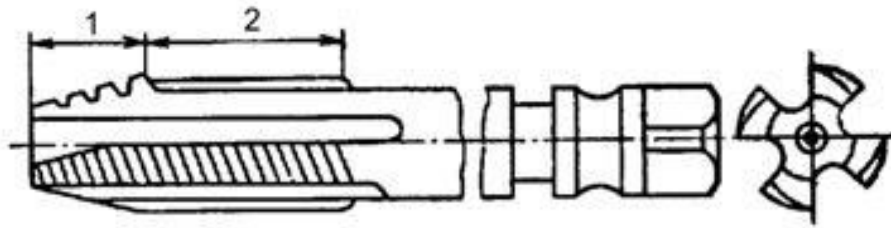


Рисунок 3. Конструкція мітчика: 1 ріжуча (огорожуюча) частина;
2 - калібруюча частина

Отже, враховуючи всю представлену інформацію, було прийнято рішення розробити програмний модуль, який буде вважатиме всі ці фактори. Крім цього програмний модуль повинен забезпечувати розрахунки режимів різання, для операцій, які виконуються на свердлильних станках:

- зенкерування;
- зенкування;
- розгортання.

Програма повинна забезпечувати розрахунок режимів різання з урахуванням фізико-механічних властивостей матеріалу, і геометричних параметрів інструмента.

В програмі необхідно передбачити наповнення бази даних: станків (з урахуванням технічних характеристик) й ріжучого інструмента (з урахуванням геометричних параметрів).

У даний час все частіше використовуються візуальні мови програмування. Дані мови програмування зручні через досить багаті бібліотеки, що накопичувалися роками. Вони дають можливість досить швидко і без особливих зусиль створити робочий додаток зі звичним для користувачів інтерфейсом. Візуальні компоненти звільняють програміста від довгострокової роботи над інтерфейсом програми і дають можливість цілком зосередитися на поставленій задачі.

Тепер пропонується кілька систем візуального програмування. У першу чергу це Delphi XE, C++ Builder, VisualBasic, Visual C++. Найбільш повними, універсальними і часто використовуваними системами є Delphi XE і Builder C++ від Borland. Ці мови мають найбільшу і наймогутнішу бібліотеку візуальних компонентів. Крім того, ця бібліотека постійно поповнюється за рахунок інших компаній, що створюють програмне забезпечення. Безліч нових компонентів можна знайти у всесвітній мережі Internet. Система Delphi є ще й однією з найпростіших у вивченні, що дає їй перевагу над іншими візуальними мовами. Delphi має прекрасні засоби для обробки і збереження як локальних так і мережових баз даних. Виходячи із цього зупиняємо вибір на системі Delphi-7.0. Ця версія продукту фірми Borland є однією з розповсюджених розробок і має всі необхідні компоненти для розробки програмного модуля[2].

Як операційне середовище для функціонування програмного комплексу була обрана платформа win64 (їй відповідають операційні системи Windows 7, Windows 8), що обумовлено наступними її особливостями:

- орієнтація ВНЗ на цю платформу;
- розвинені засоби створення користувацького інтерфейсу;

- достатня масштабуємість, тобто здатність працювати на широкому діапазоні комп'ютерного устаткування, починаючи від машин початкового рівня Pentium до багатопроцесорних систем;

- наявність драйверів для підтримки широкого спектру периферійних пристроїв (відеоадаптерів, мережних адаптерів, принтерів, дисководів CD-ROM і ін.);

- надзвичайно широке поширення цієї платформи;

Вибір пакету BorlandDelphi 7.0 обумовлений наступними його особливостями:

- можливість повторного використання готових програмних компонентів;

- наявність великої кількості стандартних компонентів, а також достатня кількість бібліотек компонент від сторонніх фірм, що розширюють і доповнюють можливості стандартних;

- можливість генерації коду під платформу win64;

- підтримка технологій ActiveX, OLE, COM, Internet-технологій;

- досить висока швидкість і надійність роботи скомпільованих програм у порівнянні з інтерпретуючими системами;

- орієнтація на «візуальні» методи розробки програм, що дозволяє швидко і якісно спроектувати і реалізувати стандартний користувацький інтерфейс;

- перспективність, популярність і широка поширеність середовища розробки у світі.

Як було згадано раніше, для реалізації програми ми використали середовище для швидкої розробки інтерфейсів й додатків Delphi 7.

Інтерфейс даної програми містить достатню кількість елементів для того, щоб повністю відобразити всі дані необхідні для перегляду інформації, що цікавить. Інтерфейс програми не перевантажений зайвими компонентами.

Контейнером для розміщення елементів інтерфейсу є форма, що являє собою вікно додатка на етапі розробки й забезпечує створення інтерфейсу користувача.(рис. 5)

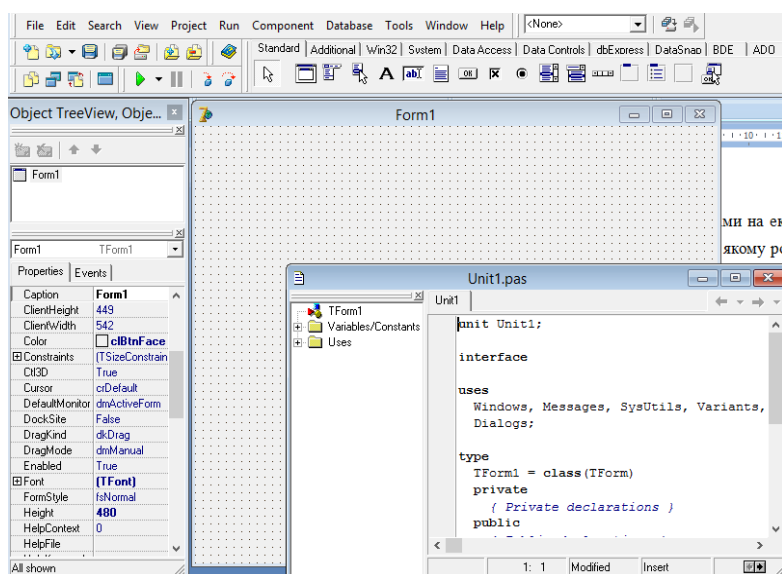


Рисунок 5. Головне вікно Delphi 7

Для створення форми використовується пункт меню File→New Form головного меню Delphi. Після створення форми на ній можна розміщати елементи управління. При запуску програми на екрані монітора з'являється головне вікно програми (рис. 5), на якому розташоване головне меню програми. Меню скомпоновано за змістовою ознакою й дає опис всім запропонованим користувачеві можливостям.

Запустивши Delphi, ми безпосередньо можемо почати розробляти наш програмний продукт. Всі необхідні нам компоненти розташовані на панелі компонентів (рис. 6)



Рисунок 6. Панель компонентів і вкладок

Нас цікавлять такі компоненти:

- Edit (3 штуки) - являє собою однорядкове текстове поле, що служить для введення даних користувачем.
- ComboBox (2 штуки) – являє собою комбінацію списку рядків ListBox з рядком введення Edit. При цьому "список рядків" компонента ComboBox спочатку прихований, і розкривається при клацанні мишкою по трикутнику розкриття, який знаходиться праворуч в рядку введення.
- Label (7 штук) – призначений для відображення статичного тексту, тобто написів і позначок на Формі, які не змінюються протягом усього часу роботи програми
- Button (1 штука) – використовується для реалізації в програмі команд за допомогою обробника події OnClick цього компонента.

Розміщуємо їх на формі таким чином, щоб вимальовувався початковий інтерфейс програми (рис. 7).

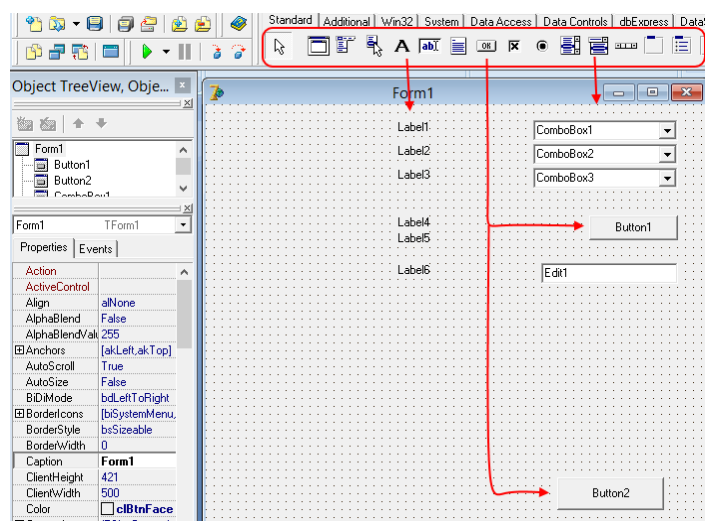


Рисунок 7. Додавання компонентів на форму

Тепер додамо на форму відсутні компоненти (рис 8):

- Мемо (1 штука) - простий текстовий редактор. DelphiMemo дозволяє вводити багаторядковий текст з клавіатури, завантажувати його з файлу, редагувати і зберігати в файл текстового формату.
- RadioGroup (2 штуки) - група залежних перемикачів. Містить спеціальні властивості для обслуговування декількох пов'язаних між собою залежних перемикачів.
- RadioButton (2 штуки) - компонент який служить для "перемикання каналів", як і в сьогоденні радіоприймачі.

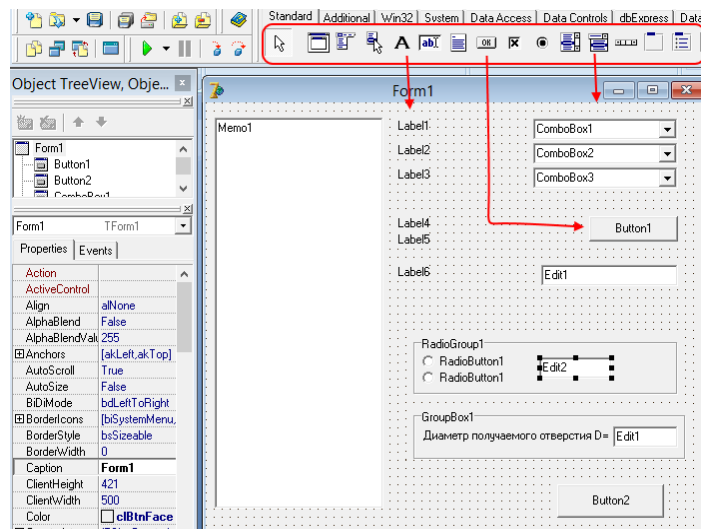


Рисунок 8. Додавання компонентів на форму

Підписуємо компоненти «label» для більш зручного користування (рис. 9)

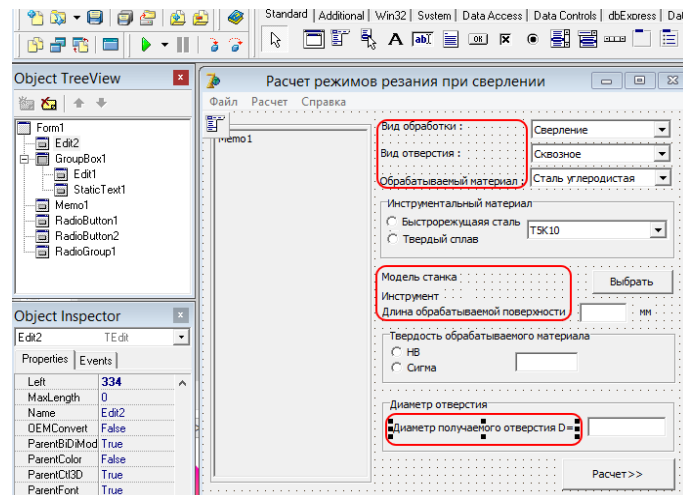


Рисунок 9. Підписання компонентів «Label»

Заповнюємо компоненти ComboBox (рис. 10).

- Вид обробки: свердління, розсвердлювання, зенкерування, розсорткування.
- Вид отвору: глухе, наскрізне.
- Обробляємий матеріал : сталь, чугун.

- Інструментальний матеріал: P6M5, P18, T5K10, T15K6, T30K4.

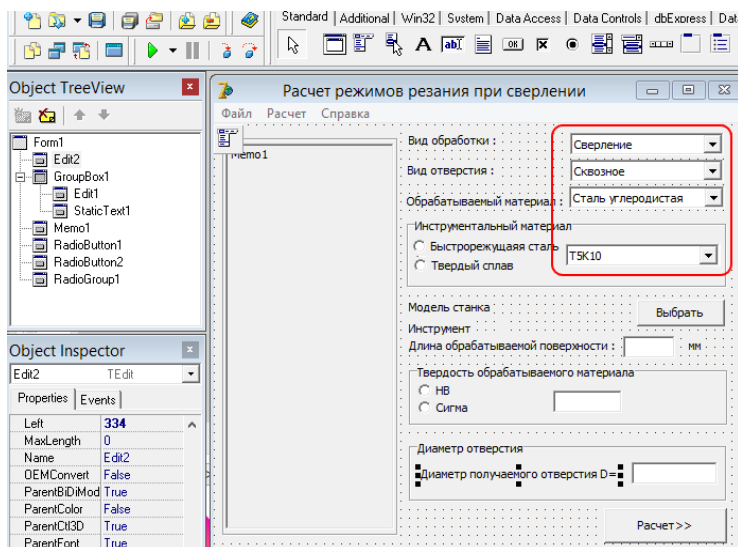


Рисунок 10. Заповнення компонентів ComboBox

Тепер створюємо нову форму, яка міститиме бази даних зі станками, і ріжучим інструментом (рис. 11). Додамо на неї такі компоненти:

- DbGrid (2 штуки) – компонент, який відображає набір даних у форматі електронної таблиці.
- Button (3 штуки).
- AdoConnection(2 штуки) – компонент, який зв'язує між собою всі компоненти з приставкою ADO.
- AdoTable(2 штуки) – таблиця ADO.
- DataSource (2 штуки) – не візуальний компонент, який забезпечує зв'язок з зовнішньою БД.

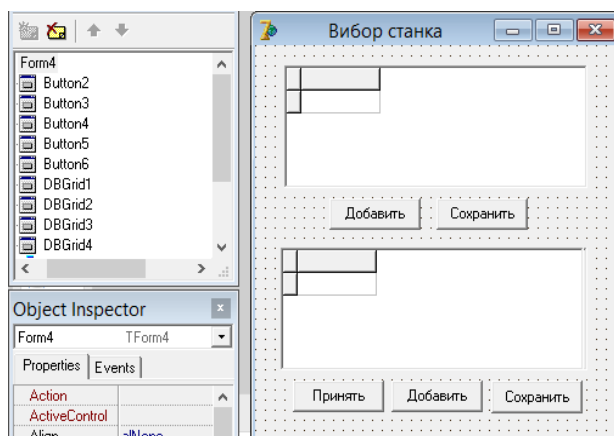


Рисунок 11. Форма вибору станка і ріжучого інструмента із БД

Аналогічними методами створюємо другу форму, яка міститиме вихідні, розрахункові дані (рис. 12). Додаємо такі компоненти: два ComboBox, двадцять один Label, дві Button і одне Memo.

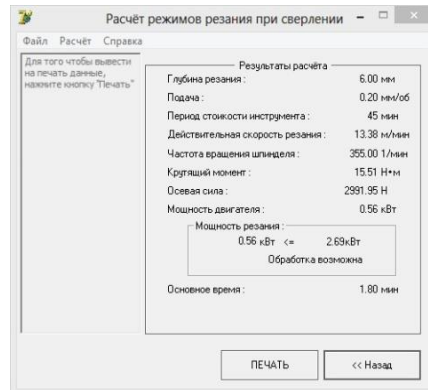


Рисунок 12. Форма з вихідними даними

Останніми створимо форми авторизації та створення нового користувача (рис. 13). Додаємо форму та добавляємо на неї такі компоненти: Label (2 штуки), Edit (2 штуки), Button (2 штуки).

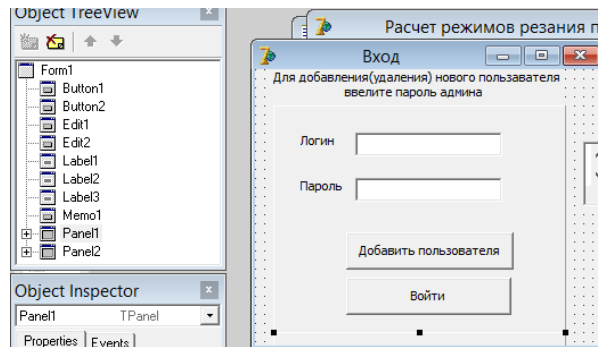


Рисунок 13. Вікно авторизації користувача

На форму додавання нового користувача додатково необхідно додати компонент Мемо (рис. 14), який додаватиме і зчитуватиме інформацію з текстового документа usr.txt, який розташовано в каталозі програми і який містить базу даних існуючих користувачів.

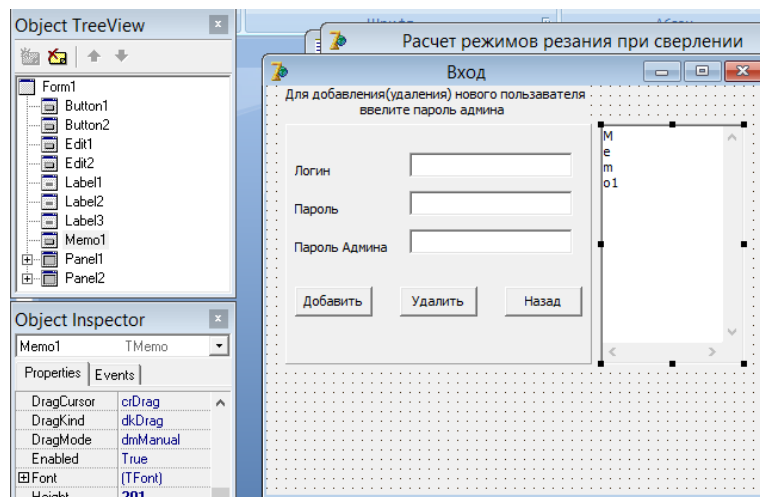


Рисунок 14. Форма додавання нового користувача

Висновки. В роботі запропоновано програмне забезпечення автоматизованого визначення режимів обробки деталей сільськогосподарського призначення на свердлильних верстатах з числовим програмним управлінням. Враховуючи всю представлену інформацію, та з використанням розглянутого програмного забезпечення, було прийнято рішення розробити технічний опис роботи користувача програми «Розрахунок режимів різання свердел», що буде подальшими дослідженнями даної тематики.

Список використаних джерел:

1. Аверченко В.И. «САПР технологічних процесів, пристроїв та ріжучих інструментів» – К. Виш. шк., 1993 – 288 с.
2. Мацулевич О.Є., Вершков О.О., Холодняк Ю.В., Дмитрієв Ю.О., Чаплінський А.П. Розробка мурашиного алгоритму для оптимізації оперативного планування робіт по збиранню врожаю кісточкових. *Плодовий сад – новітнє в теорії та практиці*: матеріали V Всеукр. наук.-практ. інтернет-конференції. Мелітополь: ТДАТУ, 2021. С. 106-110.
3. Мацулевич О.Є., Щербина В.М., Бондаренко Л.Ю., Малюта С.І., Антонова Г.В. Програмне забезпечення для автоматизованого визначення параметрів різального інструменту фрезерної обробки корпусних деталей. *Праці Таврійського державного агротехнологічного університету*. Мелітополь: ТДАТУ, 2020. Вип. 20, т. 3. С. 275-281.
4. Інженерна та комп'ютерна графіка. Частина 1: навчальний посібник / В.М. Щербина, О.Є. Мацулевич, Є.А. Гавриленко, Ю.В. Холодняк, О.В. Івженко, І.В. Пихтєєва, О.О. Вершков, С.В. Галько, А.П. Чаплінський. Мелітополь: Люкс, 2020. 238 с.
5. Проектування керуючих програм для верстатів токарної групи з пристроєм числового програмного керування: навчальний посібник / Дмитрієв Ю.О., Мацулевич О.Є., Щербина В.М., Холодняк Ю.В. Мелітополь, Люкс, 2018. 132 с.
6. Теоретична механіка. Динаміка матеріальної точки: навчально – методичний посібник / О.М. Леженкін, Г.В. Антонова, О.О. Вершков, Л.Ю. Бондаренко, О.Є. Мацулевич, А.О. Смілов, О.Ю. Михайленко. Мелітополь: Люкс, 2021. 160с.
7. Інженерна механіка: практикум / Г.В. Антонова, О.Є. Мацулевич, О.Ю. Михайленко, І.В. Пихтєєва, О.В. Івженко, Ю.В. Холодняк, В.М. Щербина, Ю.О. Дмитрієв. Мелітополь: Люкс, 2021. 147 с.
8. Нарисна геометрія та креслення: навчально-методичний посібник для підготовки бакалаврів зі спеціальностей 131 “Прикладна механіка” та 133 “Галузеве машинобудування” / Івженко О.В., Пихтєєва І.В., Гавриленко Є.А., Мацулевич О.Є., Щербина В.М., Холодняк Ю.В., Бондаренко Л.Ю., Михайленко О.Ю. Мелітополь: Люкс, 2020. 217 с.
9. Програмування автоматизованих процесів обробки деталей: навчально-методичний посібник. Лабораторний практикум / Ю.О. Дмитрієв, О.Є. Мацулевич, Є.А. Гавриленко, Ю.В. Холодняк, Г.В. Антонова – Мелітополь (протокол №8 від 28.04.2022р.), 2022.–170с.
10. Спирінцев В.В., Мацулевич О.Є., Холодняк Ю.В., Чаплінський А.П. Застосування графічного редактора ARCHICAD при вивченні дисципліни «Комп'ютерне проектування простору інженерних споруд» *Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації*: матеріали Міжнар. наук.-практ. інтернет-конф. Мелітополь: ТДАТУ, 2020. С. 262-266.

МЕТОДИ ТА ЗАСОБИ ОРГАНІЗАЦІЇ МЕНТОРСЬКОЇ ПІДТРИМКИ ПРОЦЕСУ ТЕСТУВАННЯ В ІТ-ПРОЕКТІ

Читулян В.О.

e-mail: chytulianlessons@gmail.com

Науковий керівник: Золотухіна О.А. к.т.н., доцент

Державний університет телекомунікацій, м. Київ

У роботі описано аспекти прийому на роботу фахівців спеціальності Quality Assurance. Особливості проходження інтерв'ю та закріплення менторів за стажерами, відповідно до їх рівня знань. Висвітлено відомості про модель підбору наставників відповідно до знань фахівця.

Ключові слова. Quality Assurance, ментор, trainee, тестування ПЗ, продукційна модель знань, співбесіда.

На посаду Quality Assurance приходять як досвідчені фахівці, так і початківці, які раніше вивчали процеси тестування на різних практикумах або курсах. Як показує практика, менторська підтримка необхідна кожному, незалежно від рівня і кваліфікації, оскільки вона допомагає створити у нового співробітника загальне бачення всіх процесів і занурити в роботу, а специфіка компанії та її культура розробки можуть відрізнитися від того, з чим фахівець мав справу до цього.

Під час інтерв'ю технічний фахівець і керівник відділу з'ясовують сильні та слабкі сторони фахівця, його уподобання щодо розвитку, досвід роботи та інші важливі моменти. Усі ці дані враховуються під час підбору ментора. Основним завданням інтерв'ю є – максимально корисна і комфортна взаємодія ментора та нового співробітника, на період випробувального терміну [**Ошибка! Источник ссылки не найден.**].

Залежно від базових знань визначається зміст програми. Якщо новий співробітник не має досвіду і глибоких знань, він проходить усю програму – від початку і до кінця. Якщо ж працівник проходив певні курси чи має досвід у QA, то спочатку йому потрібно написати тест. Згідно результатів тесту потрібно підібрати набір матеріалів, що співробітник не знає або має певні прогалини.

Наприклад, якщо приходить людина з сильними технічними навичками, то і наставник у неї буде високого рівня компетентності, щоб їм було цікаво спілкуватися один з одним. Якщо ж фахівець має початкові знання у певній галузі то і ментор має володіти знаннями щодо роботи з trainee.

Тому необхідно розробити модель для аналізу знань, умінь та навичок кандидата, що влаштовується на роботу для швидкого підбору ментора відповідно до можливостей фахівця. Для розробки такої моделі буде використовуватися принцип продукційної моделі, що дозволяє представити перелік змінних у вигляді речень виду «Якщо (умова) то (дія)» [2], де змінними будуть уміння та навички співробітника.

Список використаних джерел:

1. Бути ментором. Що це таке та чому корисно для спеціаліста? | GlobalLogic Ukraine. URL: <https://www.globallogic.com/ua/insights/blogs/be-a-mentor/>. (Дата звернення: 19.11.2022р.).

2. Продукційна модель. URL: https://www.wiki.uk-ua.nina.az/%D0%9F%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%86%D1%96%D0%B9%D0%BD%D0%B0_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C.html. (Дата звернення: 19.11.2022р.).

УДК [004.421:002]

СТВОРЕННЯ АЛГОРИТМУ РОЗРОБКИ ВИРОБУ З ВИКОРИСТАННЯМ ПАКЕТІВ ПРОГРАМ В ОБЧИСЛЮВАЛЬНІЙ ТЕХНІЦІ

Гешева Г.В.

e-mail: hanna.hesheva@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність теми. При проектуванні замкнутого циклу виготовлення моделі постає необхідність в автоматизації процесу проектування та керівництві виробництва за допомогою програмних засобів.

Мета роботи – моделювання замкнутого циклу виготовлення деталі.

Ключові слова – автоматизоване проектування конструкторської документації, технологічна документація, числове програмне керування, технологічний процес.

Основні матеріали дослідження. Так як невирішеність питання впровадження сучасних технологій автоматизації проектування та підготовки виробництва та перспективи розвитку підприємства, пов'язані з розширенням номенклатури пристроїв, освоєнням випуску продукції, а це в значній мірі визначає розвиток і технічний рівень усіх галузей народного господарства, постає необхідність ув'язнення всіх автоматизованих процесів, а також своєчасне виконання експортних контрактів в міжнародному поділі праці [1].

Новизна отриманих результатів полягає в наступному: створення моделі «Декоративна плита», розроблений технологічний процес виготовлення моделі, розроблена керуюча програма на устаткування з ЧПК, розроблена API програма, що дозволяє змінювати будь-який геометричний параметр моделі, зроблена експертна оцінка обраних автоматизованих систем, спроектоване технологічне устаткування.

Для досягнення поставлених цілей в роботі вирішуються наступні задачі: розроблення комплексу конструкторської документації виробництва; розроблення комплексу технологічної документації виробництва; розроблення API програми моделі «Декоративна плита».

1.1 Розрахунки на міцність

Комплекс пакетів CosmosWorks, інтегрований в інтерфейс SolidWorks, створений для потреб аерокосмічної промисловості і дозволяє вирішувати будь-які інженерні завдання.

Створений для потреб аерокосмічної промисловості, COSMOSWorks дозволяє вирішувати будь-які інженерні завдання. COSMOSWorks є "золотим партнером" SolidWorks, тому Ви просто підключаєте модуль і проводите розрахунок. Використовуючи інтегроване рішення SolidWorks - COSMOSWorks, користувачі мають ефективне дороге рішення своїх задач. Використовуючи масштабований підхід, Ви можете вибрати тільки необхідні інструменти.

COSMOSWorks аналізує міцність деталі методом кінцевих елементів, тобто чисельним методом аналізу завдань по проектуванню, з допомогою якого розв'язуються рівняння, що керують поведінкою елементів, враховують їх зв'язки між собою та встановлюють взаємозв'язок між обмеженнями й навантаженнями, переміщеннями і властивостями матеріалів. Програма виявляє переміщення в напрямках X, Y, Z у кожному вузлі, таким чином вона розраховує навантаження, що діють у різних напрямках. Також, програма використовує математичні

формули і вирази для розрахунку напружень. Аналізуючи напруження на основі завдання матеріалу, обмежень і навантажень, можна розраховувати навантаження, переміщення і напруження в деталі. Коли напруження досягає певного рівня, деталь руйнується, це зумовлено властивостями матеріалу, з якого вона виготовляється.

Застосовуються наступні способи обмежень для деталі:

- обмеження для кромок і вершин деталі;
- обмеження в певному напрямку;
- використати симетрію для аналізу частини деталі;
- вказати тверді зв'язки, болти, пружини тощо;
- в різних областях деталі вказати різні розміри елемента для підвищення точності результатів.

Для початку роботи в COSMOSWorks завантажується 3D модель. Розрахунки в COSMOSWorks виконуються у вигляді Вправ. В Менеджері COSMOSWorks за допомогою контекстного меню вибирається команда Вправа, вказуються наступні параметри: Тип аналізу → Статичний, Тип сітки → Сітка на твердому тілі. Наступним етапом задаються вихідні параметри розрахунку на міцність.

У контекстному меню твердотілого елемента "Декоративна плита" в Менеджері COSMOSWorks вибирається команда Застосувати/редагувати матеріал. У діалоговому вікні "Матеріал" обираються матеріали, наявні в базах SolidWorks.

Для фіксації обираються нижню та верхню грань моделі.

Щоб накласти навантаження, у контекстному меню елемента "Навантаження / Обмеження" в дереві Менеджера обирається команда "Тиск", задаються верхню грань деталі "Декоративна плита" на яку діє тиск.

Оскільки метою проекту є удосконалення моделі "Декоративна плита", то буде доцільним провести ряд іспитів з існуючою та модернізованою моделлю, щоб зробити порівняльний аналіз та визначитись, яка з представлених моделей є оптимальною.

Найважливішою задачею інженерних розрахунків є пошук коефіцієнта запасу міцності деталі.

Коефіцієнт запасу міцності – вказує у скільки разів допустима напруга менше небезпечного та залежить від стану матеріалу, характеру прикладання навантаження тощо.

В результаті необхідно знайти таку силу при якій деталь може не витримас прикладеної до неї сили, тоді коефіцієнт запасу міцності буде дорівнювати приблизно одиниці.

Коефіцієнти запасу міцності автоматично визначаються в кінці розрахунку програми CosmosWorks.

Елемент дерева "Звіт" дозволяє одержати повний звіт про розрахунок на міцність деталі у вигляді HTML-файлу. Звіт можна одержати, натиснувши кнопку —"Звіт" у панелі "Інструменти" результатів COSMOSWorks.

1.2 Основи розробки спеціалізованих систем проектування (API)

Обґрунтування вибору CAD-системи Компас-3D V13.

При обґрунтуванні вибору Компас-3D V13 використовувався метод "Дельфі". Метод "Дельфі" - багатоетапний метод, який передбачає початкове ізолювання винесення експертами своїх суджень і подальше багаторазове корегування на базі ознайомлення кожного експерта з судженнями інших експертів, доки інтервали між оцінками стануть сприятливими.

Були проведені наступні аналітичні розрахунки:

- середньо групова оцінка обізнаності (сума усіх оцінок обізнаності поділених на кількість експертів);
- середнє значення вибору САПР (сума усіх індивідуальних оцінок поділених на кількість експертів);
- медіана (при парному числі експертів розраховується як середньоарифметичне значення між серединами індивідуальних оцінок);
- область довірливості (визначається мінімальна індивідуальна оцінка експертів та максимальна оцінка експертів)
- квартиль (визначається як сума min та max оцінки експертів поділені на чотири)

Правильність нашого вибору САПР для проектування моделі "Декоративна плита" дорівнює 80%.

Характеристика можливостей використаної CAD-системи.

У даному проєкті використовується CAD-система Компас-3D V13. Існує два типи параметризації тривимірної моделі в КОМПАС-3D - варіаційна і ієрархічна, поєднання яких дозволяє широко варіювати параметри створюваної моделі, не змінюючи її топологію.

Варіаційна параметризація має два прояви: параметризація графічних об'єктів в ескізі і сполучення між собою компонентів збірки.

Ієрархічні параметричні зв'язки виникають автоматично по мірі виконання команд створення елементів моделі.

Для "Компас-3DV13" багата база даних стандартних елементів редукторів, що дозволяє створювати креслення компонувань редукторів, користуючись стандартними елементами (зубчасті колеса, вали, болти, підшипники).

Розглянемо основи роботи з API-інтерфейсом системи автоматизованого проектування "КОМПАС 3D" версії 13. Для використання API-інтерфейсу з Delphi необхідно перш за все обзавестися файлами, що зберігають прототипи (заголовки) процедур і функцій API. Ці файли мають назви ksAuto.pas, ksTLB.pas, LDefin2D.pas, LDefin3D.pas. Вони входять в стандартну поставку КОМПАС 3D V13 і за замовчуванням розташовані в папці Program Files \ Ascon \ KOMPAS 3DV13 \ SDK \ Include.

Для того щоб працювала API технологія необхідно:

- створити деталь;
- обміряти в кожному ескізі необхідні розміри і назвати їх згідно з призначенням та конструктивними особливостями;
- зробити необхідні розміри зовнішнім. А саме запустити функцію в КОМПАС-3D V13 змінні і в спадному вікні виділити розмір, і вибрати "Зовнішня";
- далі необхідно створити збірку, в якій буде вставлена деталь. Зберегти збірку під розпізнаваним ім'ям;
- запустити розрахунковий модуль, через функцію відкрити вибрати необхідну деталь;
- внести корективи і натиснути перебудувати.

Програмний модуль API виконує функцію - має всі зовнішні змінні разом з їх значеннями з файлу збірки, користувач програми присвоює їм нові значення, натискає кнопку "Перебудувати" і всі задані значення змінних-розмірів повертаються назад у збірку після чого починається процес перестроювання збірки.

Для чого потрібен цей модуль: якщо нам необхідно підібрати необхідні геометричні параметри деталі, але ми точно не знаємо які вони потрібні. У таких випадках нам доведеться неодноразово перебудовувати деталь і перевіряти її на міцність в спеціальних програмних пакетах. Робити це стандартними засобами КОМПАС або інших графічному пакетів незручно, оскільки це вимагає значного часу на пошук та зміну розмірів, які необхідно змінювати. Створений модуль в сукупності з параметричним складанням істотно спрощує цей процес і економить час.

1.3 Розробка API програми

В процесі виконання проекту було розроблено API програму, яка дозволяє змінити будь-який параметр деталі "Декоративна плита".

Нижче представлений фрагмент коду програми.

```
...procedure TVal.FormShow(Sender: TObject);
var i,j: integer;
c: integer;
begin
  c := 1;
  StringGrid1.RowCount := c;
  if (OpenDialog1.Execute) then
    if (FileExists(OpenDialog1.FileName)) then
      StartKompas({FILENAME}{C:\Temp\СТАКАН.a3d'}OpenDialog1.FileName)
    else Application.Terminate;
  s := TStringList.Create();
  ReadParts(s);
  FOR i:=1 TO s.Count-1 DO
    begin
      inc(c);
      StringGrid1.RowCount := c;
      StringGrid1.Cells[0, c-1] := mas[j].varname;
      StringGrid1.Cells[1, c-1] := FloatToStr(mas[j].varvalue);
      StringGrid1.Cells[2, c-1] := mas[j].VarNote;
      StringGrid1.Cells[0, c-1] := s[i];
      StringGrid1.Cells[1, c-1] := ";";
      mas := GetPartVars(s[i]);
      for j := 0 to Length(mas)-1 do
        begin
          .....
        end.
```

Висновок. Була поставлена задача створення алгоритму розробки виробу з використанням пакетів програм в обчислювальній техніці. Для проведення подальшої роботи була створено тривимірну твердотільну модель та розроблено технологічний процес на її виготовлення та автоматизовано процес проектування при керівництві замкнутого циклу виробництва за допомогою програмних засобів.

Обрано необхідні ріжучі інструменти для обробки деталі, проведено їх аналіз, а також передбачені вимірювальні інструменти, які підтримують якість та точність конструкції.

Була розроблена 2D та 3D модель деталі в програмі «Компас 3DV13».

Для зменшення затрат на моделювання деталі та програмної реалізації автоматизації її перебудови було створено модуль розрахунку в програмі «Delphi7». Даний модуль інтегрований в систему проектування «Компас 3DV13», що дозволяє візуально спостерігати зміни структури деталі при її перебудові та вносити в неї зміни.

Для обробки деталі на верстаті з ЧПК була розроблена керуюча програма за допомогою програми PowerMill, для цього були використані тривимірна модель та заготовка твердотілої деталі. Обрані стратегії для обробки даної деталі є оптимальними.

За допомогою «платіжної матриці» та функції корисності в умовах ризику був обран верстат вертикально – фрезерувальний 6Р11Ф3-1.

Список використаних джерел:

1. Анур'єв В.І. Довідник конструктора-машинобудівника. У 3-х т./В.І. Анур'єв. Т. 2. - 5 - е вид., Перероб. та дод. - М.: Машинобудування, 1980. - 559 с.
2. Гжиров Р.І., Серебрицький П.П.. Програмування обробки на станках з ЧПУ/Р.І. Гризов, П.П. Серебрицький з ЧПЗ: Довідник. - Л.: Машинобудування, 1990.-588 с.
4. Потьомкін А. Тривимірне твердотільне моделювання. / А. Потьомкін. - М: Комп'ютерПрес, 2002.-296 с.
5. Скворцов А. В., Схиртладзе А. Г. Основи технології автоматизованих машинобудівних виробництв. / О.В. Скворцов, А.Г. Схиртладзе. – М.: Вища школа, 2010. – 589 с.

УДК 004.891

ІНТЕЛЕКТУАЛЬНА ІНФОРМАЦІЙНА ВЕБ-СИСТЕМА «ВИБІР СМАРТФОНУ»

Коломоєць Д.А.

e-mail: dany1a20011@gmail.com

Науковий керівник к.т.н, доцент Холодняк Ю.В.

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. В останні роки в усьому світі спостерігається зростаючий інтерес до систем, які в автоматизованому виді видають товар клієнту за набором ознак та характеристик. Власноруч проводити обробку такої кількості інформації неможливо. Тому виникає необхідність в створенні комп'ютерних засобів автоматизації обробки, пошуку, збереження, систематизації даних, при цьому здатних аналізувати дані, пристосовуватися до змін інформації, робити власні висновки на основі аналізу.

Метою статті є розробка інтелектуальної інформаційної системи, яка дасть можливість враховувати побажання користувача по вибору смартфона.

Основні матеріали дослідження. Інтелектуальна система (ІС) – це технічна або програмна система, яка забезпечує розв'язування неформалізованих задач користувача в деякій предметній галузі та організовує його взаємодію з комп'ютером у звичних поняттях, термінах, образах, що вважалися творчими, які традиційно належать конкретним предметним областям, інформація зберігаються в пам'яті інтелектуальної системи. Структура інтелектуальної системи включає три основні блоки - базу знань, механізм виведення рішень і інтелектуальний інтерфейс. Властивість ІС виконувати творчі завдання називається штучним інтелектом [1].

Штучний інтелект – здатність інженерної системи обробляти, застосовувати та вдосконалювати здобуті знання та вміння.

Найчастіше, інформаційні інтелектуальні системи (ІС) в мережі інтернет використовуються в якості рекламно-інформаційної функції. В свою чергу, спеціалізовані інтелектуальні системи використовуються локально, тому що, виконують конкретні вузкопрофільні завдання.

Інтелектуальна система має виконувати наступні функції:

- обробка інформаційних ресурсів;
- надання інформаційних ресурсів користувачам. Підтримка в інформаційній системі інформаційних ресурсів, що дозволяють моделювати стан та поведінку предметної галузі;
- збір та реєстрація інформаційних ресурсів. Ця функція забезпечує "фотографування" предметної області, формування та підтримку на цій основі моделі предметної області;
- зберігання інформаційних ресурсів. Ця функція інформаційних систем пов'язана з необхідністю управління двома видами ресурсів: ресурсами даних і ресурсами пам'яті, що зберігаються;
- актуалізація інформаційних ресурсів. Призначення інформаційної системи полягає у підтримці динамічної інформаційної моделі її предметної галузі.

Інтелектуальні інформаційні системи поділяються на наступні різновиди:

- інтелектуальні бази даних (БД);

- природно-мовний інтерфейс;
- гіпертекстові системи;
- системи контекстної допомоги;
- системи когнітивної графіки.

Інтелектуальна база даних (або Інтелектуальна БД) - база даних, в якій для відповіді на запит використовуються як факти, що зберігаються безпосередньо, так і факти, які отримуються шляхом логічного виведення, а користувачі звертаються до даних у звичних поняттях, термінах, образах.

Інтелектуальні БД відрізняються від звичайних експертних систем можливістю вибірки по запиту інформації, що не зберігається явно в базі.

Інтелектуальні бази даних відрізняються від звичайних баз даних можливістю вибірки за запитом необхідної інформації, яка може явно не зберігатися, а виводитися з наявною в базі даних. Прикладами таких запитів можуть бути наступні:

- "Вивести список товарів, ціна яких вище середньогалузевої";
- "Вивести список товарів-замінників деякої продукції";
- "Вивести список потенційних покупців деякого товару" і т. д.

Природно-мовний інтерфейс передбачає трансляцію природно-мовних конструкцій на внутрішньо машинний рівень представлення знань. Для цього необхідно вирішувати завдання морфологічного, синтаксичного і семантичного аналізу та синтезу висловлювань на природній мові. Так, морфологічний аналіз передбачає розпізнавання і перевірку правильності написання слів за словниками, синтаксичний контроль - розкладання вхідних повідомлень на окремі компоненти (визначення структури) з перевіркою відповідності граматичним правилам внутрішнього представлення знань і виявлення відсутніх частин, а семантичний аналіз - встановлення смислової правильності синтаксичних конструкцій. Синтез висловлювань вирішує зворотну задачу перетворення внутрішнього подання інформації в природно-мовну.

Природно-мовний інтерфейс використовується для:

- доступу до інтелектуальних баз даних;
- контекстного пошуку документальної текстової інформації;
- голосового введення команд в системах управління;
- машинного перекладу с іноземних мов.

Гіпертекстові системи призначені для реалізації пошуку за ключовими словами в базах текстової інформації. Інтелектуальні гіпертекстові системи відрізняються можливістю більш складної семантичної організації ключових слів, яка відображає різні смислові відносини термінів. Таким чином, механізм пошуку працює, перш за все, з базою знань ключових слів, а вже потім безпосередньо з текстом. У більш широкому плані сказане поширюється і на пошук мультимедійної інформації, що включає крім текстової та цифрової інформації графічні, аудіо-та відео-образи.

Системи контекстної допомоги можна розглядати, як окремий випадок інтелектуальних гіпертекстових і природно-мовних систем. На відміну від звичайних систем допомоги, які нав'язують користувачеві схему пошуку необхідної інформації, в системах контекстної допомоги користувач описує проблему (ситуацію), а система за допомогою додаткового діалогу її конкретизує і сама виконує пошук відносяться до ситуації рекомендацій. Такі системи відносяться до класу систем поширення знань (Knowledge Publishing) і створюються як додаток до систем документації (наприклад, технічної документації по експлуатації товарів).

Когнітивні системи дозволяють здійснювати об'єднання інтерфейсу користувача з інформаційно-вимірювальних систем за допомогою графічних образів, які генеруються відповідно до подій. Такі системи використовуються в моніторингу і управлінні оперативними процесами. Графічні образи в наочному і інтегрованому вигляді описують безліч параметрів досліджуваної ситуації. Наприклад, стан складного керованого об'єкта відображається у вигляді людського обличчя, на якому кожна риса відповідає за будь-якої параметр, а загальний вираз обличчя дає інтегровану характеристику ситуації.

Системи когнітивної графіки широко використовуються також в навчальних і тренажерних системах на основі використання принципів віртуальної реальності, коли графічні образи моделюють ситуації, в яких хто навчається необхідно приймати рішення і виконувати певні дії.

1. Інтерпретація даних. Це одне з традиційних завдань для експертних систем. Під інтерпретацією розуміється процес визначення змісту даних, результати якого мають бути погодженими і коректними. Зазвичай передбачається багатоваріантний аналіз даних.

2. Діагностика. Під діагностикою розуміється процес співвідношення об'єкта з деяким класом об'єктів і виявлення несправності в деякій системі. Несправність - це відхилення від норми. Таке трактування дозволяє з єдиних теоретичних позицій розглядати і несправність устаткування в технічних системах, і захворювання живих організмів, і всілякі природні аномалії. Важливою специфікою є тут необхідність розуміння функціональної структури («анатомії») діагностуючої системи.

3. Моніторинг. Основне завдання моніторингу - безперервна інтерпретація даних у реальному масштабі часу і сигналізація про вихід тих або інших параметрів за допустимі межі. Головні проблеми - «пропуск» тривожної ситуації і інверсне завдання «помилкового» спрацьовування. Складність цих проблем полягає в розмитості симптомів тривожних ситуацій і необхідності обліку тимчасового контексту.

4. Проектування. Проектування полягає в підготовці специфікацій на створення «об'єктів» із заздалегідь визначеними властивостями. Під специфікацією розуміється весь набір необхідних документів - креслення, записка пояснення і так далі. Основні проблеми - це здобуття чіткого структурного опису знань про об'єкт і проблема «сліду». Таким чином, в завданнях проектування тісно зв'язуються два основні процеси, виконуваних в рамках відповідної ЕС: процес виведення рішення і процес пояснення.

5. Прогнозування. Прогнозування дозволяє передбачати наслідки деяких подій або явищ на підставі аналізу наявних даних. Прогнозуючі системи логічно виводять ймовірні наслідки із заданих ситуацій. У прогнозуючій системі зазвичай використовується параметрична динамічна модель, в якій значення параметрів «підганяються» під задану ситуацію. Висновки, що виводяться з цієї моделі, складають основу для прогнозів з ймовірними оцінками.

6. Планування. Під плануванням розуміється знаходження планів дій, що відносяться до об'єктів, здатних виконувати деякі функції. У таких ЕС використовуються моделі поведінки реальних об'єктів з тим, аби логічно вивести наслідки планованої діяльності.

7. Навчання. Під навчанням розуміється використання комп'ютера для навчання деякої дисципліни або предмету. Системи навчання діагностують помилки при вивченні якої-небудь дисципліни за допомогою ЕОМ і підказують правильні рішення. Вони акумулюють знання про гіпотетичного «учня» і його

характерні помилки, потім у роботі вони здатні діагностувати слабкості в знаннях учнів і знаходити відповідні засоби для їхньої ліквідації. Крім того, вони планують акт спілкування з учнем залежно від успіхів учня з метою передачі знань.

8. Керування. Під керуванням розуміється функція організованої системи, що підтримує певний режим діяльності. Такого роду ЕС здійснюють управління поведінкою складних систем відповідно до заданих специфікацій.

9. Підтримка прийняття рішень. Підтримка прийняття рішень — це сукупність процедур, що забезпечує особу, що приймає рішення, необхідною інформацією і рекомендаціями, що полегшують процес ухвалення рішення. Ці ЕС допомагають фахівцям вибрати і сформулювати потрібну альтернативу серед безлічі виборів при ухваленні відповідальних рішень.

У загальному випадку всі системи, засновані на знаннях, можна підрозділити на системи, що вирішують завдання аналізу, і на системи, які вирішують завдання синтезу. Основна відмінність завдань аналізу від завдань синтезу полягає в тому, що якщо в завданнях аналізу безліч рішень може бути перераховане і включене в систему, то в завданнях синтезу безліч рішень потенційно не обмежена і будується з рішень компонентів або проблем. Завданнями аналізу є: інтерпретація даних, діагностика, підтримка ухвалення рішення. До завдань синтезу відносяться проектування, планування, управління. Комбіновані: вчення, моніторинг, прогнозування.

Предметна область системи, яка розробляється, описується діаграмою варіантів використання (прецедентів) (рис. 1). Дана діаграма дозволяє побачити ролі кожного актора у системі. Проектована система представляється у вигляді наборів факторів, що взаємодіють з системою за допомогою так званих варіантів використання. Система має двох акторів: Системного адміністратора ЕС та Користувача ЕС, діаграма дає уявлення можливостей та зобов'язань кожного з акторів.

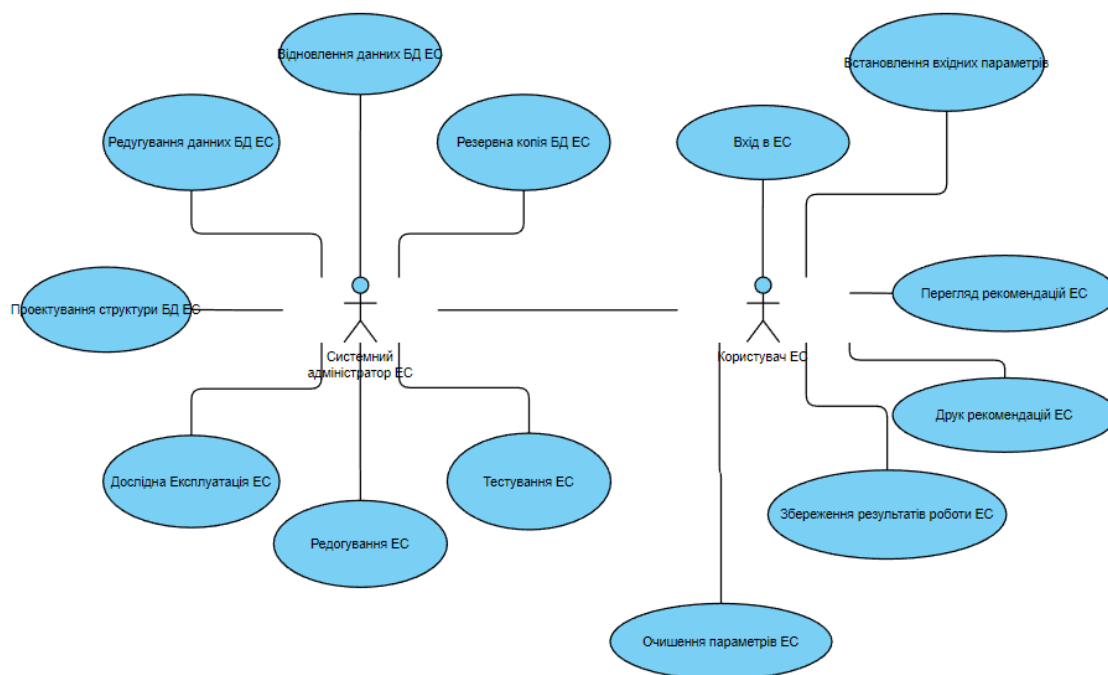


Рисунок 1. Діаграма варіантів використання

Задачі Системного адміністратора ЕС:

- Резервне копіювання БД ЕС;
- Відновлення даних БД ЕС;
- Редогування даних БД ЕС;
- Тестування ЕС;
- Редогування ЕС і тд.

Задачі Користувача ЕС:

- Вхід в ЕС;
- Перегляд рекомендацій ЕС;
- Друк рекомендацій ЕС;
- Очищення параметрів ЕС.

На головній сторінці інтелектуальної інформаційної системи (рис. 2) знаходиться загальний перелік доступних смартфонів, меню з базовими параметрами пошуку смартфона, авторизація та поле пошуку дивайсу.

Дана інтелектуальна система має наступні переваги:

- простий інтерфейс;
- велика кількість параметрів для підбору;
- великий товарний асортимент;
- інтуїтивно зрозумілий принцип роботи системи.

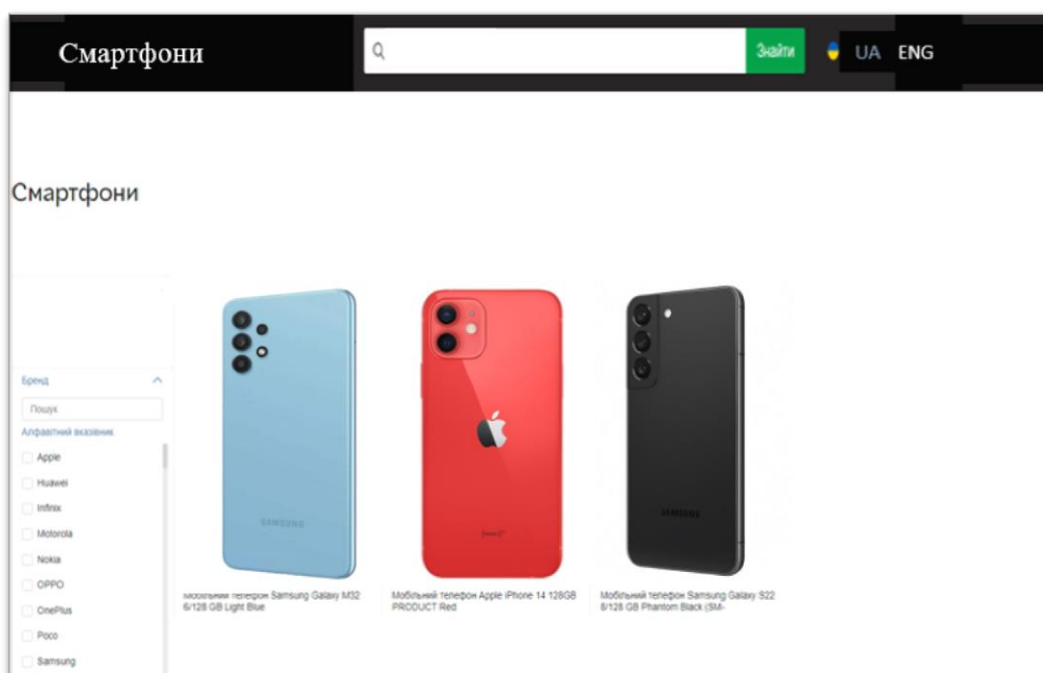


Рисунок 2. Головна сторінка інтелектуальної інформаційної системи

Користувач має можливість пошуку товару за назвою через поле пошуку, або через параметри моделей (рис. 3), такі як:

- ціна;
- виробник;
- ємність акумулятора;
- об'єм вбудованої пам'яті;
- об'єм оперативної пам'яті;
- розмір екрану;
- операційна система та ін.

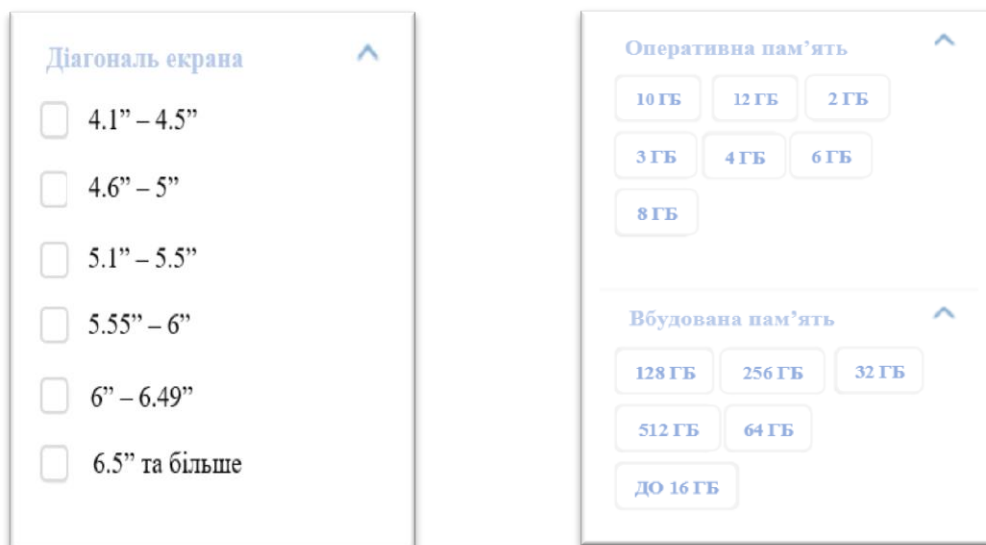


Рисунок 3. Поле параметрів моделей

Крім того додано можливості системі виведення популярних моделей, що в першу чергу відображаються на головній сторінці.

На рис. 4 наведено результат роботи інформаційної системи.

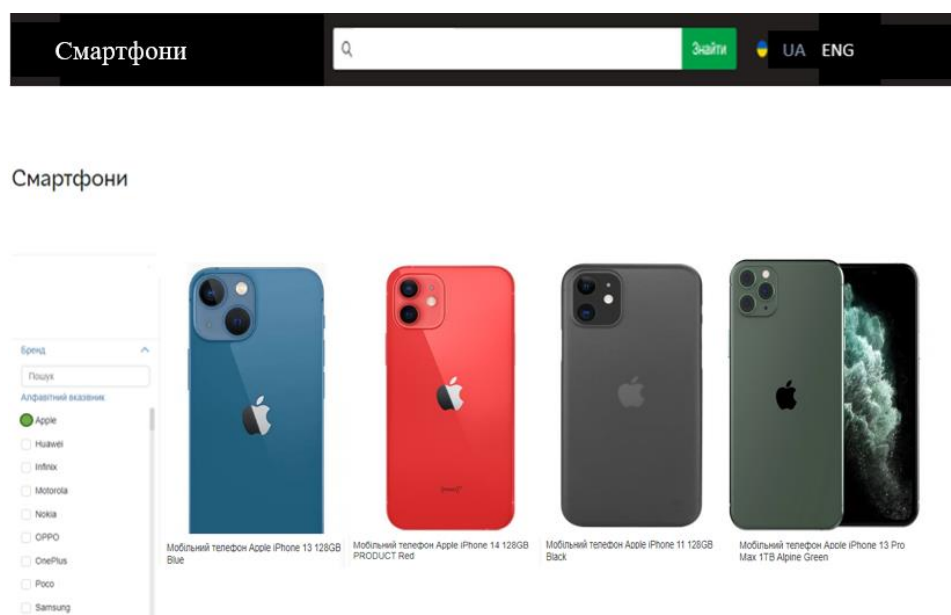


Рисунок 4 – Приклад роботи інформаційної системи

Серверна частина ресурсу була розроблена за допомогою мов програмування PHP та MySQL.

PHP, попередня назва: Personal Home Page Tools — скриптова мова програмування, була створена для генерації HTML-сторінок на стороні вебсервера. PHP є однією з найпоширеніших мов, що використовуються у сфері веброзробок та підтримується переважною більшістю хостинг-провайдерів.

PHP інтерпретується веб-сервером у HTML-код, який передається на сторону клієнта. На відміну від скриптової мови JavaScript, користувач не бачить PHP-коду, тому що браузер отримує готовий html-код. Це є перевагою з точки

зору безпеки, але погіршує інтерактивність сторінок. Існує можливість використовувати PHP для генерування JavaScript-кодів, які виконуються на стороні клієнта.

MySQL - одна з систем управління базами даних, що найбільш використовуються. MySQL управляє реляційними базами даних, тобто такими, у яких таблиці пов'язані між собою. MySQL працює за принципом клієнт-сервер. Комп'ютер користувача надсилає запит. Сервер баз даних його обробляє та надає відповідь. Це сервер, де зберігається база даних. Система MySQL написана мовами програмування C та C++. Для роботи MySQL використовується мова структурованих запитів SQL.

Дана частина системи підключається до бази даних з переліком всіх доступних товарів та фільтрує їх за заданими параметрами.

Код підключення до БД виглядає наступним чином:

```
<?php
// Підключення до серверу MySQL
$host = ' localhost ';
$user = ' rootuser ';
$password = '01012001';
$db_name= 'danil01012001';
$db = mysqli_connect($host, $user, $password)
or die('connect to database failed');

// Інсталювання кодування
mysqli_set_charset($db, "utf8");
$select_db = mysqli_select_db($db, $db_name)
or die('db not found');
?>
```

Висновок. В результаті проведеної роботи було створено інтелектуальну інформаційну веб-систему «Вибір смартфона» відповідно до бажань клієнта та було виконано розробку технічного завдання, в якому наведена мета, параметри системи її призначення, вимоги до створення цієї системи. Була проведена робота по огляду та класифікації інтелектуальних систем, були описані завдання, які ними мають бути виконуються.

Список використаних джерел:

1. Качмар В. О. Медичні інформаційні системи – стан розвитку в Україні. *Український журнал телемедицини та медичної телематики*. 2010. Т.8, №1. С. 12-17.
2. Медичні інформаційні системи: огляд можливостей і приклади використання. URL: <https://evergreens.com.ua/ua/articles/medical-information-systems.html>.
3. Злепко С.М., Овчарук Т.І., Овчарук А.А. Огляд медичних інформаційних систем. Системи обробки інформації. 2011. № 3. С. 189-192.
4. Методи та системи штучного інтелекту: навч. посіб. / укл. Д.В. Лубко, С.В. Шаров. Мелітополь: ФОП Однорог Т.В., 2019. 264 с.
5. IOM. To Err is Human: Building a Safer Health System: Institute of Medicine (IOM); 1999.
6. Wang, S., et al. A Cost-Benefit Analysis of Electronic Medical Records in Primary Care. *The American Journal of Medicine*. 2003. Vol. 114. P. 397–403.

УДК 635.36

ОСОБЛИВОСТІ МОДЕЛЮВАННЯ ТА ДІАГНОСТУВАННЯ ТЕХНІЧНОГО СТАНУ ЕЛЕМЕНТІВ РУЛЬОВОГО КЕРУВАННЯ

Бондар А.М., к.т.н.,

e-mail: andriy.bondar@tsatu.edu.ua

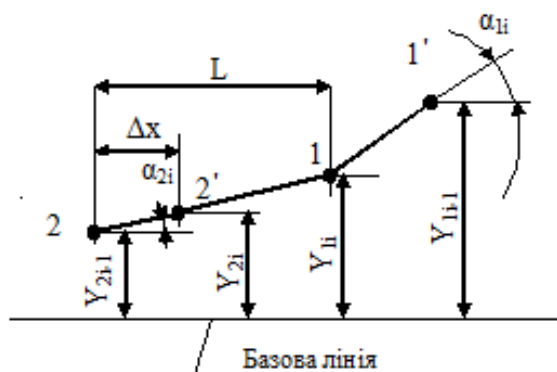
Дашивець Г.І., к.т.н.

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Рульове керування повинне забезпечувати рух транспортного засобу по заданому оператором напрямку. Його конструкція повинна забезпечувати точність керування, надійність роботи всіх вузлів і деталей та не вимагати від оператора витрати великих зусиль і не передавати на рульове колесо поштовхи, які сприймаються колесами транспортного засобу, а також дотримуватись агротехнічних вимог під час виконання технологічних операцій. Тому актуальними є дослідження пов'язані з роботою рульового керування під час виконання транспортних та технологічних операцій [1].

Основні матеріали дослідження. Найбільш важливою та складною ланкою процесу керування є людина, тобто оператор. Це пояснюється тим, що йому необхідно слідкувати за зовнішніми умовами на дорозі та навколо, підтримувати роботу транспортного засобу у оптимальному режимі, не дивлячись на те, що постійно змінюється швидкість та навантаження на робочі органи. Також необхідно слідкувати за робочими органами та приборами керування та вчасно виконувати корегуючі дії у відповідності до технологічного процесу та зовнішніх умов. І звичайно, виникає додаткове психологічне та фізичне навантаження на оператора, яке він отримує під час роботи у незручних умовах (нічний час, схили та ін.).

З метою пояснення роботи транспортного засобу та оператора під час виконання транспортних та технологічних операцій була розроблена схема переміщення вздовж базової лінії [1,2]:



L – довжина колісної бази трактора; α_1 – кут повороту передніх керуючих коліс; α_2 – поточне значення курсового кута; Y_1 , Y_2 – поточне значення відхилень від базової лінії переднього та заднього коліс трактора відповідно; Δx – крок квантування моделі по переміщенню.

Рисунок 1. Схема переміщення транспортного засобу вздовж базової лінії

Відомі намагання математичного описання поведінки оператора, однак це пов'язано з не виправданими спрощеннями можливостей людини. Крім того, в сучасній, складній техніці використовуються комп'ютерні симулятори та тренажери. З одного боку вони використовуються для тренування навичок оператора, з іншого боку – широко використовуються для отримання оптимальних експлуатаційних показників систем керування та загальної оцінки можливостей нових типів систем керування.

Такий підхід дозволяє отримати достовірні оцінки керованості складних систем управління, синтезувати таку систему керування, яка відповідає можливостям людини оператора та забезпечує високі показники керованості.

Розглянемо деякі математичні моделі рульового керування:

Традиційне рульове керування.

Базовий варіант – рульове керування транспортного засобу з передніми управляючими колесами і з постійним передатним відношенням рульового приводу являє собою дискретну модель.

Дискретна математична модель в даному випадку має вигляд:

$$\left\{ \begin{array}{l} \alpha_{1i} = \frac{\alpha_i}{W} \\ \Delta x = V_i \cdot \Delta t; \\ \alpha_{3i} = \frac{(Y_{1i-1} - Y_2)}{L} \\ Y_{1i} = Y_{1i-1} + (\alpha_{3i} + \alpha_{1i} + \delta_i) \cdot \Delta x \\ Y_{2i} = Y_2 + \alpha_{3i} \cdot \Delta x \end{array} \right. \quad (1)$$

де: V – швидкість руху МТА;

L – довжина колісної бази транспортного засобу;

W – передаточне відношення рульового механізму;

α – кут повороту рульового колеса;

α_1 – кут повороту передніх керуючих коліс;

α_3 – поточне значення курсового кута;

Δt – шаг квантування по часу (0.05 с);

Y_1, Y_2 – поточне значення відхилень від базової лінії переднього та заднього коліс транспортного засобу відповідно;

Δx – шаг квантування моделі по переміщенню;

δ – значення збурюючого кута уводу еластичних шин.

Рульове керування постійної чутливості [3-5]:

Конструкція такого рульового керування, розроблена автором, забезпечує сумісність високої маневреності і стабільності руху МТА на підвищених швидкостях.

Запропонована методика інтерактивного моделювання процесу керування МТА на комп'ютерному тренажері дає можливість при мінімальному апаратному забезпеченні отримати адекватні оцінки керованості.

$$\left\{ \begin{array}{l} \Delta x = V_i \cdot \Delta t \\ W_i = W \left(1 + \frac{V_i}{V_0} \right) \\ \alpha_{1i} = \frac{\alpha_i}{W_i} \\ \alpha_{3i} = \frac{(Y_{1i-1} - Y_2)}{L} \\ Y_{1i} = Y_{1i-1} + (\alpha_{3i} + \alpha_{1i} + \delta_i) \cdot \Delta x \\ Y_{2i} = Y_2 + \alpha_{3i} \cdot \Delta x \end{array} \right. \quad (2)$$

Правомірність такого підходу є в наступному [3, 4].:

- крізь зоровий канал оператор отримує біля 90% інформації відносно зовнішньої обстановки;

- час реакції оператора на відхилення процесу, який контролюється, складає від 0,5 до 2,5 секунд, що значно перевищує шаг квантування по часу при моделюванні.

Невелике значення кутів α_1 , α_3 і δ дозволило уникнути застосування тригонометричних формул та визначило простоту математичних моделей, конкретний вид яких залежить від принципу дії особливостей рульових керувань моделюємих транспортних засобів (рис. 1).

Діагностування елементів рульового керування

Перевірку рульових механізмів проводять за допомогою візуального огляду систем, для цього транспортний засіб встановлюють на естакаді або використовують оглядову яму. Для перевірки рульового управління передні колеса автомобіля виставляють в режимі прямолінійного руху.

Підготувавши машину до технічного огляду, насамперед перевіряють вільний хід рульового колеса, для чого його починають повертати спочатку в одну, а потім в іншу сторону. У нормі вільний хід колеса до початку повороту передніх коліс не повинен перевищувати 5° , обід колеса при цьому зміщується не більше ніж на 20 мм.

Якщо важко визначити вільний хід рульового колеса, можна зробити відповідні заміри та обчислення. Для проведення заміру буде потрібна лінійка, яку потрібно поставити вузькою стороною в упор до панелі приладів, при цьому площина лінійки повинна щільно прилягати до зовнішньої поверхні рульового колеса. Потім повертають кермо до початку повороту коліс і роблять мітку на кермі, для цього підійдуть тонкий дріт, фломастер або крейда. Потім кермо повертають в інший бік також до початку повороту коліс і роблять другу мітку.

Якщо після виставлення коліс на прямолінійний рух спиці керма не займають строго горизонтального положення, а зміщені, необхідно провести регулювання кутів установки коліс, перевірити системи рульового управління і підвіски.

Після цього лінійкою заміряють відстань між двома мітками і порівнюють його з розрахунковим, виробленим за формулою:

$$L = (5^\circ/360^\circ) \pi D, \quad (3)$$

де L - це люфт рульового колеса, мм,

$\pi = 3,14$,

D - це зовнішній діаметр рульового колеса, мм.

входу керуючого логічного елемента 8, інформаційний вхід якого через нормуючий підсилювач 4 підключений до датчика 1 кутових переміщень. П'ятий вихід керуючого ключа 10 підключений до входу керуючого логічного елемента 9, інформаційний вхід якого з'єднаний з виходом перетворювача "аналог - частота" 7. Вхід перетворювача "аналог - частота" підключений до виходу нормуючого підсилювача 5, вхід якого з'єднаний з датчиком 3 зусиль.

Виходи логічних елементів 8 і 9 з'єднані з входами логічного елемента АБО 11, вихід якого підключений до лічильного входу лічильника імпульсів 12. До виходу лічильника імпульсів підключені інформаційний вхід цифрового індикатора 15 і один з входів компаратора 14 сигналів, а до виходу компаратора підключено індикатор "Перевищення" 17.

Як датчик 3 зусилля, можна використовувати тензо- або п'єзодатчик мікропереміщень, що має на виході електричний сигнал. Цей датчик встановлений на корпусі, що закріплюється на рульовому колесі за допомогою самоцентруючого захоплення. З корпусом шарнірно пов'язана навколо осі рульового колеса штанга, що повертається відносно нього, взаємодіє з датчиком зусиль. Зверху корпус закритий прозорим диском, що мають радіальні світловідбиваючі штрихи.

Висновок. Наведена методика моделювання руху транспортного засобу вздовж базової лінії. З'ясовані основні несправності рульового керування, які виникають під час експлуатації. Проаналізована послідовність перевірки елементів рульового керування з наведенням необхідних значень для усунення недоліків. Запропонована блок-схема електронного пристрою для контролю зусиль та люфту рульового управління.

Список використаних джерел:

1. Бондар А.М. Технічний сервіс мехатронних систем: навчально-методичний посібник до самостійної роботи. Мелітополь: ВПЦ «Люкс», 2021. 141 с.
2. Журавель Д.П., Бондар А.М. Перспективний спосіб підвищення керованості колісних машин Праці Таврійського державного агротехнологічного університету. – Мелітополь: ТДАТУ, 2020. – Вип. 20, т. 4. – 279 с.
3. Журавель Д.П., Бондар А.М. Технологія ремонту рульових рейок Технічне забезпечення інноваційних технологій в агропромисловому комплексі: матеріали II Міжнар. наук.-практ. Інтернет-конференції. Мелітополь, 02-27 листопада 2020 р. С.537-538.
4. Журавель Д.П. Бондар А.М. Несправності рульового керування та їх наслідки. Технічне забезпечення інноваційних технологій в агропромисловому комплексі: матеріали II Міжнар. наук.-практ. Інтернет-конференції. Мелітополь, 02-27 листопада 2020 р. С.478-479.
5. Патент України на корисну модель UA № 146117 МПК B62 D 1/18. Тактильне рульове керування транспортного засобу з GPS корекцією напрямку руху. / Д.П. Журавель, А.М. Бондар, В.О. Петров, Г.І. Дашивець - № u 2020 05305. Дата публікації відомостей про видачу патенту 21.01.2021, бюл. №3.

УДК 004.3+519.681.5+539.3

ПРО ЗАГАЛЬНІ ПІДХОДИ ДО РОЗВ'ЯЗУВАННЯ АЛГОРИТМІЧНО СКЛАДНИХ ЗАДАЧ МОДЕЛЮВАННЯ

Поліщук О.Д.¹, д. т. н.

e-mail: od_polishchuk@ukr.net

Яджак М.С.^{1,2}, д. ф.-м. н.

e-mail: yadzhak_ms@ukr.net

¹Інститут прикладних проблем механіки і математики ім. Я. С. Підстригача
НАН України

²Львівський національний університет імені Івана Франка

Актуальність та формулювання проблеми. Зараз використання паралельних обчислень є одним із найважливіших факторів прискорення обчислювального процесу під час розв'язання алгоритмічно складних задач (АСЗ) [5, 9]. До АСЗ будемо відносити перш за все ті, що потребують значних обчислювальних ресурсів, а також задачі, розв'язок яких потрібно одержувати в режимі реального часу [8].

Для ефективної реалізації паралельних методів та алгоритмів обчислень розвиваються, поширюються та стають загальнодоступними програмні (мови програмування високого рівня Fortran, Delphi, C, C++, C#, Java, Python, Go, Ada, Ruby, Eiffel тощо; сервіси для паралельного програмування MPI, OpenMP, TBB, CSP, CUDA; бібліотеки виконання паралельних завдань) та апаратні (комп'ютери з багатоядерним процесором, кластерні системи, гібридні архітектури з використанням прискорювачів NVIDIA GPUs, співпроцесорів Intel Xeon Phi; розподілені обчислювальні середовища [11]) засоби. Головним інтеграційним напрямком розвитку високопродуктивних обчислювальних систем та середовищ в усьому світі і в Україні зокрема є розвиток grid-інфраструктури. Більшість обчислювальних систем суперкомп'ютерного класу, об'єднаних grid, є кластерного типу. Останнім часом здійснюється їх модернізація шляхом переходу на процесори з більшою кількістю ядер, залучення графічних процесорів та інших прискорювачів спеціалізованих обчислень; збільшення обсягів сховищ даних та пропускної здатності комунікаційної мережі, що з'єднує обчислювальні вузли. Пропускна здатність комунікаційної мережі суттєво впливає на продуктивність всієї обчислювальної системи. Як впливає із останньої редакції (листопад 2022 року) списку найпотужніших обчислювальних систем світу [7] зараз найбільш поширеними є такі комунікаційні технології, як Gigabit Ethernet, InfiniBand та Intel Omni-Path, а також використовуються, наприклад, Tofu interconnected D та Aries interconnect.

Алгоритмічно складними є задачі моделювання різноманітних процесів у механіці зв'язаних полів, оскільки зазвичай вони є нелінійними, тривимірними, нестационарними та залучають до розгляду об'єкти зі складною геометрією. Прикладами таких задач є задачі електромагнітотермомеханіки для моделювання термомеханічних процесів [4], задачі моделювання процесів деформування елементів конструкцій за інтенсивних термосилових навантажень [2], задачі аналізу напружено-деформованого стану плоских середовищ з включеннями різної жорсткості [1] тощо. Нелінійність, складна геометрична форма досліджуваних об'єктів та їх включень суттєво утруднюють використання аналітичних підходів під час розв'язання таких задач, тому потрібно застосовувати універсальні чисельні методи, зокрема метод скінченних елементів (МСЕ), метод граничних елементів або метод скінченних різниць (МСР). Також тут слід зазначити про певні труднощі з використанням ліцензійних пакетів (висока вартість, складність етапу підготовки даних для обчислень). Однак, поряд із розробкою та застосуванням ефективних пос-

лідовних обчислювальних схем потрібно широко використовувати і можливості розпаралелювання та реалізації обчислень на сучасних комп'ютерах і суперкомп'ютерах.

Основні матеріали дослідження. Розглянемо доволі загальні підходи до розпаралелювання, які ґрунтуються на використанні паралелізму за даними, а також геометричного та функціонального паралелізму.

Підхід до розпаралелювання на підставі паралелізму даних називається декомпозицією за даними. Він є доволі тривіальний і може бути використаний у разі, коли нам потрібно розв'язати одну і ту ж задачу при різних наборах вхідних даних. Такий підхід дозволяє за час розв'язання одного варіанту задачі одержати розв'язок для декількох її варіантів. Декомпозиція за даними може бути статичною і динамічною. У разі статичної декомпозиції за даними кожному процесорові (ядру, обчислювальному вузлу) наперед призначається його частка даних. У випадку ж динамічної декомпозиції блоки даних розподіляються по процесорах під час розв'язання задачі в порядку надходження даних і звільнення відповідних процесорів. Алгоритм обчислень, побудований на основі паралелізму даних, зазвичай є крупнозернистим (крупноблочним) і при цьому не виникає особливих проблем із його відображенням на архітектуру сучасних паралельних систем. Описаний підхід певною мірою використано в [5, 9] під час розпаралелювання методики комплексного оцінювання стану та процесу функціонування складних динамічних систем з ієрархічно-мережевою структурою.

Геометричним паралелізмом володіють, наприклад, задачі, що описуються диференціальними рівняннями в частинних похідних, зокрема, крім наведених вище, і задачі механіки суцільного середовища, теорії поля, переносу зарядів у напівпровідникових структурах [10] тощо. Такі задачі зазвичай розв'язуються з допомогою МСЕ або МСР. Для дискретних аналогів цих задач характерними є локально-обмежені взаємодії між вузлами сітки, що покриває область розв'язання. А це дозволяє розбити цю область на підобласті і обчислення в кожній з них «доручити» окремому процесору. Відмінність геометричного паралелізму від паралелізму за даними полягає в тому, що підзадачі опрацювання кожної із підобластей взаємозв'язані між собою (потрібен обмін даними між цими підзадачами). Підхід до розпаралелювання на основі геометричного паралелізму називається декомпозицією області розв'язання і є ефективним за умови, що обчислювальна складність кожної із підзадач є приблизно однаковою. Крім цього, для ефективності такого підходу фрагмент обчислень, що виконується на деякому процесорі P_i , повинен використовувати лише невеликий обсяг даних, розташованих на інших процесорах. Також бажано, щоб ці не локальні дані були розташовані на невеликій кількості сусідніх процесорів. Можна виділити статичну декомпозицію області розв'язання і динамічну. Якщо обчислювальні складності підзадач опрацювання кожної із підобластей змінюються в процесі обчислень, то статична декомпозиція області може виявитись малоефективною. У цьому випадку може бути більш доцільною динамічна декомпозиція області, за якої межі між підобластями змінюються під час обчислень. Згідно з висновками, зробленими у праці [3], ефективність розпаралелювання під час розв'язання багатовимірних дискретних задач визначається перш за все способом декомпозиції області і її відображенням на архітектуру обчислювальної системи. При цьому збільшення обчислювальної складності задачі (нелінійність, нестационарність, оптимальне керування) не змінює принципово прискорення та ефективність розпаралелювання, а збільшення логічної та «інтелектуальної» складності алгоритмів (адаптивні сітки з динамічними локальними згущеннями, необхідність візуального контролю за обчислювальним проце-

сом і його корегування) може значно знизити очікуваний ефект від розпаралелювання.

Функціональний паралелізм – це паралелізм груп операцій, об'єднаних за функціональною ознакою. Підхід до розпаралелювання на підставі функціонального паралелізму називається функціональною декомпозицією. Тривіальним прикладом функціональної декомпозиції є декомпозиція задачі на чотири такі підзадачі: введення початкових даних, опрацювання їх, виведення результатів, візуалізація результатів. Паралелізм при цьому досягається унаслідок одночасного виконання чотирьох вказаних підзадач і створення “конвеєра” між ними. Зазначимо, що кожна із вказаних підзадач може володіти паралелізмом за даними. У разі функціональної декомпозиції задачі кількість використовуваних процесорів визначається кількістю підзадач. Збільшити кількість процесорів з метою збільшення швидкості розв'язання задачі за такого підходу є проблематичним. Функціональна декомпозиція була частково використана під час побудови високопаралельних алгоритмів розв'язання задач цифрової фільтрації різної розмірності на суперкомп'ютерах зі структурно-процедурною організацією обчислень [12] та спеціалізованих обчислювальних засобах – квазісistolічних структурах [8].

Необхідно зазначити, що більшість задач моделювання зводяться до розв'язання СЛАР великої розмірності з розрідженими матрицями. У [6] розроблено алгоритми структурної регуляризації розріджених матриць і формування їх блочно-розрідженої структури (БРС) та побудовано блочні та блочно-циклічні схеми декомпозиції і розподілу між процесорами елементів матриць БРС. Оскільки під час розвинення матриць виконується значно більше арифметичних операцій, ніж під час розв'язання СЛАР з трикутними матрицями розвинення, то у згаданій праці запропоновано блочні варіанти методів розвинення Гаусса і Холецкого, які враховують структуру матриці. Для розв'язування СЛАР на високопродуктивних гетерогенних комп'ютерах розроблено цілу низку алгоритмів, які використовують багаторівневу модель паралельних обчислень, зокрема блочно-циклічні алгоритми для систем з симетричними матрицями БРС (стрічковими, профільними, хмарочосними), блочний алгоритм для систем з симетричною додатно визначеною блочно-діагональною матрицею з обрамленням, циклічний алгоритм для систем з блочно-стрірковою несиметричною матрицею тощо.

Висновок. Отже, нами зроблено огляд деяких загальних підходів до розпаралелювання обчислень під час розв'язання задач моделювання на сучасних та перспективних обчислювальних системах суперкомп'ютерного класу зі спільною та розподіленою пам'яттю [7]. Ці підходи можна застосовувати окремо, виходячи з особливостей розв'язуваної задачі, а також у допустимому поєднанні. Зауважимо, що описані підходи до розпаралелювання є доволі універсальними і можуть бути використані для прискорення обчислень під час моделювання у режимі реального часу складних процесів (явищ), які протікають (спостерігаються) в інших предметних галузях з використанням сучасних технологій програмування та апаратних засобів.

Список використаних джерел:

1. Гудрамович В. С., Гарт Э. Л. Конечноэлементный анализ процесса рассеянного разрушения плоскодеформированных упругопластических сред с локальными концентраторами напряжений. *Упругость и неупругость: матер. междунар. симп., посв. 105-летию со дня рожд. А. А. Ильюшина* (Москва, 20–21 янв. 2016). М.: Изд. МГУ, 2016. С. 158–161.

2. Дробенко Б., Бурик О. Напружений стан елементів конструкцій за інтенсивних термосилових навантажень. *Сучасні проблеми термомеханіки: зб. наук. праць / за заг. ред. Р. М. Кушніра [Електронний ресурс]*. Інститут прикладних проблем механіки і математики ім. Я. С. Підстригача НАН України, 2016. С. 85. Режим доступу: www.iapmm.lviv.ua/MPT2016.
3. Ильин В. П. О стратегиях распараллеливания в математическом моделировании. *Программирование*. 1999. № 1. С. 41–46.
4. Подстригач Я. С., Бурак Я. И., Кондрат В. Ф. Магнитотермоупругость электропроводных тел. Киев: Наук. думка, 1982. 293 с.
5. Поліщук О. Д., Яджак М. С. Мережеві структури та системи: IV. Паралельне опрацювання результатів неперервного моніторингу. *Системні дослідження та інформаційні технології*. 2019. № 2. С. 105–114.
6. Попов О. В. Комп'ютерні методи дослідження математичних моделей з розрідженими структурами даних. *Автореф. дис. ... д. ф.-м. н., спец. 01.05.02–математичне моделювання та обчислювальні методи*. Київ: Інститут кібернетики ім. В. М. Глушкова НАН України, 2021. 32 с.
7. Список найпотужніших обчислювальних систем світу. [Електр. ресурс]. Режим доступу: <http://www.top500.org>.
8. Яджак М. С. Високопаралельні алгоритми та засоби для розв'язання задач масових арифметичних і логічних обчислень. *Автореф. дис. ... д. ф.-м. н., спец. 01.05.03–математичне та програмне забезпечення обчислювальних машин і систем*. Київ: КНУ імені Тараса Шевченка, 2009. 33 с.
9. Яджак М. С., Поліщук О. Д., Тютюнник М. І. Оптимізація методики комплексного оцінювання складних систем на підставі паралельних обчислень. *Інформатика та математичні методи в моделюванні*. 2016. 6, № 4. С. 347–356.
10. Gorbal' S. I., Yadzhak M. S. Numerical solution of the transport equations for charges in semiconducting devices. *Journal of Soviet Mathematics*. 1993. 66(6). P. 2637–2641.
11. Polishchuk O., Polishchuk D., Tyutyunnyk M., Yadzhak M. Big Data Processing in Complex Hierarchical Network Systems II: Computer Environments and Parallelization. *AASCIT Communications*. 2016. 3(3). P. 119–124.
12. Yadzhak M. S., Tyutyunnyk M. I. An optimal algorithm to solve digital filtering problem with the use of adaptive smoothing. *Cybernetics and Systems Analysis*. 2013. 49. P. 449–456.

УДК 004.052

РОЗРОБКА СИСТЕМИ ПРОГНОЗУВАННЯ НАДІЙНОСТІ ПРОГРАМНИХ ПРОДУКТІВ

Борейченко Г.О., студент

e-mail: boreichenkoglib@gmail.com

Чижмотря О.В., ст. викл.

e-mail: chov@ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. У сучасному життєвому циклі програмного забезпечення тестування є найважливішим засобом перевірки його надійності. Однак, тестування є трудомістким процесом і потребує багато часу та бюджетних ресурсів. Сьогодні вартість розробки програмного забезпечення та витрати пов'язані з відмовами програм є однією з найбільших категорій витрат у розробці складних комп'ютерних систем. Надійність програмного забезпечення залишається актуальною тематикою для досліджень. Саме тому було вирішено реалізовувати систему прогнозування надійності програмних продуктів.

Основні матеріали дослідження. Для розробки системи було вибрано дві ймовірнісні моделі надійності ПЗ, а саме які базуються на неоднорідному пуассоновому процесі. Це S-подібна модель з затримкою та модель Гоеля-Окумото. Для того, щоб отримати точніший результат під час роботи майбутньої системи використано показник покриття коду тестами. Цей показник можна отримати шляхом відношення кількості відтестованих рядків коду (TLOC) до загальної кількості рядків коду в програмі (LOC). В результаті додавання показника покриття коду тестами отримано наступні формули розрахунку кількості програмних дефектів для моделей Гоеля-Окумото та S-подібної моделі з затримкою.

$$\alpha = \frac{n}{1 - e^{-bs_n}} \left(2 - \frac{TLOC}{LOC} \right) \quad (1)$$

$$\alpha = \frac{y_n \left(2 - \frac{TLOC}{LOC} \right)}{1 - (1 + bt_n)e^{-bt}} \quad (2)$$

Якщо загальна кількість рядків коду дорівнює нулю, то функція немає розв'язків тому, що програма має складатись принаймні з одного рядка коду. Подібні уточнення можливо застосовувати й до інших моделей надійності ПЗ, оскільки вони побудовані на статистичних даних про кількість програмних дефектів виявлених упродовж певного проміжку часу. Внесені зміни загалом не впливають на алгоритм розрахунку надійності програмних продуктів, а лише покликані збільшити точність отриманих результатів дослідження.

Для розроблюваної системи прогнозування надійності програмних продуктів було складено алгоритм роботи з використанням згаданих вище функцій (1, 2). Спершу задано точність необхідних розрахунків. Далі проведено розрахунок інтенсивності виявлення відмов. Проведено розрахунок функції кількості виявлення помилок. Наприкінці алгоритму проведено розрахунок кумулятивної кількості помилок.

Для організації розрахунків показника надійності розроблено програмний застосунок для визначення і прогнозування загальної кількості дефектів у протестованому ПЗ. Під час розробки було використано вебфреймворк Angular 14 та технології HTML, CSS. Застосовано бібліотеку web-worker вбудовану у Angular для математичних розрахунків, бібліотеку Angular Material для швидкої розробки компонентів програми та бібліотеку Ngx-charts для коректного відображення графіків у програмі. Для запуску програмного застосунку застосована платформа Node.js. Розроблювана система прогнозування надійності ПЗ складається з серверної та клієнтської частин. Розробка програми здійснювалась згідно з технологією розробки керованої тестами (Test Driven Development). Після застосування даної технології вдалося забезпечити повне покриття коду тестами.

Під час дослідження було виявлено, що показник покриття коду тестами дозволяє істотно збільшити точність розрахунків надійності ПЗ. Підсумовуючи результати дослідження, можна дійти висновку, що використані моделі надійності ПЗ дозволяють певніше надавати показники надійності ПЗ користувачу для оптимального розподілу бюджету проєкту. Хоча зазначити, що для кращого розрахунку показників надійності ПЗ варто додати до формул моделей показник складності проєкту або кількість потрібного часу на тестування коду програми. У майбутньому це уможливить прогнозування дефектів з ще вищою точністю.

Висновок. Отже, було досліджено проблему надійності програмного забезпечення у сучасних комп'ютерних системах та засоби розрахунку показників надійності програмних продуктів. Для розв'язання проблеми було вирішено створити програмний додаток для розрахунку показника надійності. Було окреслено структуру програмного застосунку, основні інструменти розробки та технології для втілення. Наприкінці було проаналізовано напрями удосконалення програмного застосунку та підведено підсумки дослідження.

Підсумовуючи, можна зазначити, що попри численні напрацювання у тематиці надійності програмного забезпечення проблема залишається актуальною і на сьогодні. І оскільки з кожним днем збільшується складність програмних систем, збільшується і нагальність рішення за даною проблематикою. Розроблений програмний застосунок може проводити розрахунки показників надійності програмного забезпечення на базі статистики про помилки та дефекти знайдені у протестованому коді програми і надавати їх розробникам у доступній формі. Такий програмний застосунок дозволить зменшити витрати на розробку та тестування програмного забезпечення, підвищити продуктивність розробки та збалансувати розподіл бюджетних ресурсів.

Список використаних джерел:

1. Бобало, Ю. Я., Волочій Б. Ю., Лозинський О. Ю., Менді Б. А. Озірковський Л. Д., Федасюк Д. В., Щербовський С. В., Яковина, В. С. (2013). Математичні моделі та методи аналізу надійності радіоелектронних, електричних і програмних систем: монографія. Львів: Видавництво Львівської політехніки, 300 с.
2. Яковина, В. С., і Сенів М. М. (2020). Основи теорії надійності програмних систем: навчальний посібник. Львів: Видавництво Львівської політехніки, 300 с.
3. Яковина В. С., Федасюк Д. В., Сенів М. М. та Нитребич О. О. (2015). Моделі, методи та засоби аналізу надійності програмних систем: монографія. Львів: Видавництво Львівської політехніки, 220 с.

УДК 51-77

ЗАСТОСУВАННЯ МЕТОДУ ГОЛОВНИХ КОМПОНЕНТІВ ДЛЯ УСУНЕННЯ МУЛЬТКОЛІНЕАРНОСТІ

Зінов'єва О.Г., ст. викл.

e-mail: olha.zinovieva@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. На практиці при кількісній оцінці параметрів економетричної моделі досить часто стикаються з проблемою мультиколінеарності пояснювальних змінних. Тобто коли взаємозв'язок між цими змінними доволі тісний, то оцінка параметрів моделі може мати велику похибку та результати побудови моделі можуть бути некоректними.

Для усунення мультиколінеарності розроблено декілька методів. Одним з найбільш простих є виключення однієї або декількох змінних. Але це може бути досить складним при реалізації. Іншим методом є метод перетворення змінних. Але цей спосіб також не може повністю гарантувати усунення мультиколінеарності. Тому для усунення мультиколінеарності пропонується використовувати метод головних компонент, де замість вхідних змінних використовуються головні компоненти цих змінних. Так як головні компоненти не корелюють одна з одною, то в результаті отримана регресійна модель відповідна моделі, яка отримана за методом найменших квадратів. В роботі показано як при побудові регресійної моделі усувається мультиколінеарність за допомогою методу головних компонент.

Основні матеріали дослідження. Метод головних компонентів – це один з методів багатомірної математичної статистики. Він призначений для оцінювання параметрів моделей з великою кількістю незалежних змінних та при наявності мультиколінеарних змінних. Суть методу полягає у зведенні числа факторних змінних до найбільш істотно впливаючих факторів. Це досягається шляхом лінійного перетворення всіх факторних змінних у нові змінні, які називаються головними компонентами. При цьому висувається вимога, щоб перший головний компоненті відповідав максимум загальної дисперсії всіх змінних, другій компоненті – максимум дисперсії, що залишилась, після виключення впливу першої компоненти і т.д. k-й головною компонентою ($k = 2, \dots, p$) досліджуваної системи показників X називається така нормовано-центрована лінійна комбінація цих показників, яка не корельована з $(k - 1)$ попередніми головними компонентами і серед інших нормовано-центрованих і не корельованих з попередніми $(k - 1)$ головними компонентами лінійних комбінацій змінних $x(1), x(2), \dots, x(p)$ має найбільшу дисперсію. Декілька перший головних компонент пояснюють майже всю мінливість даних, а інші відображають випадкові похибки і можуть далі не розглядатися. Знаходження основних компонентів зводиться до обчислення власних векторів та власних значень коварійної матриці вихідних даних.

Алгоритм методу головних компонентів складається з наступних кроків:

Крок 1. Нормування змінних. Одиниці виміру ознак можуть бути непорівнянні, тому переходять до нормованих змінних:

$$x_{ij}^* = \frac{x_{ij} - \bar{x}_j}{\bar{\sigma}_{ij}}, \quad i = \overline{1, n}; \quad j = \overline{1, m},$$

де $\bar{x}_j$ ($j = \overline{1, m}$) - середнє значення змінної x_{ij} $i = \overline{1, n}$; $j = \overline{1, m}$;
 $\bar{\sigma}_{ij}$ - середньоквадратичне відхилення змінної x_{ij} $i = \overline{1, n}$; $j = \overline{1, m}$.

Крок 2. Обчислення кореляційної матриці R , елементами якої є парні коефіцієнти кореляції $r_{x_i x_j}$. Мета цього кроку – визначити, чи є взаємозв'язок між факторами у вхідних даних.

$$R = \begin{pmatrix} 1 & r_{x_1 x_2} & r_{x_1 x_3} & \dots & r_{x_1 x_n} \\ r_{x_2 x_1} & 1 & r_{x_2 x_3} & \dots & r_{x_2 x_n} \\ r_{x_3 x_1} & r_{x_3 x_2} & 1 & \dots & r_{x_3 x_n} \\ \dots & \dots & \dots & \dots & \dots \\ r_{x_n x_1} & r_{x_n x_2} & r_{x_n x_3} & \dots & 1 \end{pmatrix}$$

Крок 3. Знаходження характеристичних чисел матриці R , тобто визначення коренів $\lambda_1, \lambda_2, \dots, \lambda_m$ рівняння m -го порядку

$$|R - \lambda_j E| = 0, \quad (j = \overline{1, m})$$

де E - одинична матриця розмірності $m \times m$

Крок 4. Ранжування власних значень $\lambda_1, \lambda_2, \dots, \lambda_m$ за абсолютним внеском кожного головного компонента в загальну дисперсію

Крок 5. Обчислення власних векторів $\vec{a}_j$ для кожного кореню характеристичного рівняння $\lambda_1, \lambda_2, \dots, \lambda_m$ за допомогою системи лінійних рівнянь

$$|R - \lambda_j E| \vec{a}_j = 0,$$

де $\vec{a}_j$ - власні вектори матриці R ;

E - одинична матриця розмірності $m \times m$;

λ_j - характеристичні числа матриці R .

Крок 6. Знаходження матриці головних компонентів

$$Z = AX^*,$$

де A - матриця, складена з власних векторів $\vec{a}_j$;

X^* - матриця нормалізованих змінних.

Визначивши всі головні компоненти і відкинувши ті з них, які відповідають невеликим значенням характеристичних чисел, знаходимо рівняння регресії з головними компонентами $y = b_0 + b_1 z_1 + \dots + b_j z_j, (j = \overline{1, m})$, а далі за допомогою оберненого перетворення переходимо від параметрів моделі з головними компонентами до моделі зі змінними X_{ij} .

Нижче наведений приклад розрахунку методу головних компонентів в пакеті символьної математики Maple.

```
> with(linalg):
> with (stats):
> x1:=[80,100,90,75,120,85,100,70,120,75]:
> x2:=[25,40,10,0,60,80,10,0,15,5]:
> x3:=[15,18,10,17,14,26,11,25,6,12]:
> x1sr:=describe[mean](x1):
> x2sr:=describe[mean](x2):
> x3sr:=describe[mean](x3):
> sigma1:=evalf(describe[standarddeviation](x1)):
> sigma2:=evalf(describe[standarddeviation](x2)):
> sigma3:=evalf(describe[standarddeviation](x3)):
> for i from 1 to 10 do x1n[i]:=(x1[i]-x1sr)/sigma1 od:
> for i from 1 to 10 do x2n[i]:=(x2[i]-x2sr)/sigma2 od:
> for i from 1 to 10 do x3n[i]:=(x3[i]-x3sr)/sigma3 od:
>
Xn:=matrix([[x1n[1],x1n[2],x1n[3],x1n[4],x1n[5],x1n[6],x1n[7],x1n[8],x1n[9],x1n[10]],
[x2n[1],x2n[2],x2n[3],x2n[4],x2n[5],x2n[6],x2n[7],x2n[8],x2n[9],x2n[10]],
[x3n[1],x3n[2],x3n[3],x3n[4],x3n[5],x3n[6],x3n[7],x3n[8],x3n[9],x3n[10]]]);
> Rx1x2:=evalf(describe[linearcorrelation](x1,x2)):
> Rx1x3:=evalf(describe[linearcorrelation](x1,x3)):
> Rx2x3:=evalf(describe[linearcorrelation](x2,x3)):
>
R:=matrix([[1,Rx1x2,Rx1x3],[Rx1x2,1,Rx2x3],[Rx1x3,Rx2x3,1]]):
):
> L:=eigenvals(R):
> sort([L[1],L[2],L[3]]):
> L[3]/(L[1]+L[2]+L[3]):
> L[2]/(L[1]+L[2]+L[3]):
> L[1]/(L[1]+L[2]+L[3]):
> lambda:=eigenvectors(R):
> a1:=lambda[1][3]:
> a2:=lambda[2][3]:
> a3:=lambda[3][3]:
>
A:=matrix([[a1[1][1],a2[1][1],a3[1][1]],[a1[1][2],a2[1][2],a3[1][2]],
[a1[1][3],a2[1][3],a3[1][3]]]):
> At:=transpose(A):
> Xnt:=transpose(Xn):
> Z:=multiply(Xnt,A);
```



```
[.3584062738 .3601833464 -.4410258932]
[.1649217298 -.8409851757 .2362689300 ]
[-.7569543582 .6757316779 -.3055262459]
[.5698586122 1.247378874 .1060535340 ]
[-.8710573498 -1.981322425 .0958560360 ]
Z:= [2.053602705 -1.884209614 -.274462491 ]
[-.9822780234 .3283375213 .1357049149 ]
[1.758406550 1.154920077 .7539662739 ]
[-2.275419889 -.3056195187 .2004595195 ]
[-.0194862510 1.245585235 -.5072945783]
```

При розв'язанні задачі отримали нові змінні Z_j – головні компоненти.

Визначимо кількість власних чисел, які «пояснюють» задану частку дисперсії. Пропонований алгоритм використовує одну з властивостей власних чисел: чим більша частка власного числа в сумі елементів вектора власних чисел, тим більшу частку інформації здатна пояснити відповідна йому головна компонента кількість власних чисел, які «пояснюють» задану частку дисперсії. Пропонований алгоритм використовує одну з властивостей власних чисел: чим більша частка власного числа в сумі елементів вектора власних чисел, тим більшу частку інформації здатна пояснити відповідна йому головна компонента.

При упорядкуванні характеристичних коренів рівняння за внеском у загальну дисперсію

```
> sort([L[1],L[2],L[3]]);
      [.1239153929, 1.325048957, 1.551035650]

> L[3]/(L[1]+L[2]+L[3]);
      .5170118833

> (L[3]+L[2])/(L[1]+L[2]+L[3]);
      .9560810917
```

можна зробити висновок, що останньою головною компонентою можна знехтувати. Таким чином ми скоротили число змінних до найбільш суттєво впливаючих факторів.

Використовуючи обернене перетворення $X^* = A^T Z$ переходимо до відновлюваних вихідних факторів.

```
> X:=multiply(submatrix(Z,1..10,1..2),submatrix(At,1..2,
1..3));
> x1:=evalm(col(X,1)*sigma1+x1sr);

      x1 := [79.99999999, 99.99999999, 90.00000003, 75.00000001,
120.0000000, 84.99999993, 100.0000000, 70.00000002, 120.0000000,
75.00000001]
> x2:=evalm(col(X,2)*sigma2+x2sr);
```

```
x2 := [25.000000003, 40.00000001, -.4 10 , 10.00000001, 59.99999999,  
      80.00000016, 9.99999995, 0, 14.99999991, 4.99999998]  
> x3:=evalm(col(X,3)*sigma3+x3sr);  
  
x3 := [15.00000000, 18.00000000, 10.00000000, 17.00000000,  
      14.00000000, 26.00000000, 11.00000000, 25.00000000, 6.000000002,  
      12.00000000]
```

Висновки. Розглянуто методику використання методу головних компонентів при побудові регресійної моделі з наявністю мультиколінеарності факторів. Процедура усунення мультиколінеарності факторів за допомогою методу головних компонентів описано в пакеті символьної математики Maple. Надано пояснення цієї процедури з позицій теоретичних положень.

Список використаних джерел:

1. Наконечний С.І., Терещенко Т.О., Романюк Т.П. Економетрія: Підручник. Вид. 2-ге, допов. та перероб. К.: КНЕУ, 2000. 296 с.
2. Ліщинський О.Л., Рязанцева В.В., Юнькова О.О. Економетрія: Навч. посібник для студ. вищ. навч. закл.. К.: МАУП, 2003. 208 с.
3. Касярум О.П., Касярум Я.О. Обчислення у таблицях «EXCEL» головних компонентів факторного аналізу/Фінансовий простір. №4 (24). 2016, С.29-35.
4. Principal Component Analysis (PCA) and Partial Least Squares (PLS) Technical Notes // STATISTICA Help. URL: <http://documentation.statsoft.com>

УДК004.825

АЛГОРИТМ МАМДАНІ В СИСТЕМАХ НЕЧІТКОГО ВИВЕДЕННЯ

Зінов'єва О.Г., ст. викл.

e-mail: olha.zinovieva@tsatu.edu.ua

Лубко Д.В., к.т.н.

e-mail: di79ma@gmail.com

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Розв'язання завдань у прикладних областях, де використовуються інформаційні системи, вимагає обробки неповної, неточної та розмитої інформації та призводить до необхідності використання знань людини-експерта.

Перспективним методом вирішення таких завдань є застосування апарату нечіткої логіки, що також визначається можливістю реалізації довільного нелінійного перетворення вхідних даних у поєднанні із простотою їх лінгвістичного опису.

Використання систем нечіткого логічного виведення почало розповсюджуватися з 1980 року. Ця система була розроблена професором Л. Заде. Його теорія нечіткої логіки зараз повсюди застосовується у всіх цифрових системах. Математична теорія нечітких множин дозволяє описувати ... Пізніше Л. Заде та його послідовники ввів поняття операцій над нечіткими множинами, нечіткого висновку та поняття лінгвістичної змінної, значення якої є нечіткими множинами. Подібний підхід використовується як для моделювання складних технічних об'єктів і для створення реальних пристроїв автоматичного керування такими об'єктами [2-4]. У цій статті обговорюється застосування пакету прикладних програм Mathcad для методів нечіткої логіки.

Основні матеріали дослідження. В рамках даної статті розглянемо реалізацію алгоритму нечіткого виведення Мамдані для прийняття рішень в середовищі Mathcad.

Одним з найбільш зручних засобів програмування в галузі нечіткої логіки є середовище Fuzzy Logic пакету прикладних програм Matlab. Основна перевага цього методу – візуальне моделювання, коли можна створювати досить складні програми без написання програмного коду. Але цей пакет є досить коштовним і тому не завжди може бути доступним для приватних осіб. До того при створенні програм можна використовувати тільки алгоритм Мамдані або Сугено. Використання інших алгоритмів потребує більш складних методів розробки. В цій статті показано реалізацію алгоритма Мамдані в прикладному математичному пакеті Mathcad. На основі цього підходу можна реалізувати будь-який обчислювальний алгоритм нечіткої логіки

Як було зазначено вище, концепція нечіткої логіки базується на поняттях нечітких множин, нечітких тверджень, тощо. Методи нечіткої логіки відносяться до так званих «м'яких» обчислень, заснованих на імовірнісному підході. У рамках цього підходу було введено поняття нечіткої лінгвістичної змінної, її значення можна виразити через типи «низький», «помірний», «великий», «дуже», «висока» тощо. Сукупність лінгвістичних значень нечіткої змінної утворює так званий терм-набір.

Для реалізації систем з урахуванням нечітких правил розроблено безліч алгоритмів нечіткого виведення. Алгоритми нечіткого виведення різняться переважно видом використовуваних правил, логічних операцій та різновидом

способу дефазифікації. Розглянемо роботу процедур нечіткої логіки на прикладі алгоритму Мамдані для задачі оцінки ризику використання інформаційної системи.

Основна ідея алгоритму Мамдані: база нечітких правил містить лінгвістичні правила, що використовують функції приладдя для опису понять.

Механізм логічного виведення складається з наступних етапів:

1. Введення нечіткості (фазифікація). На даному етапі здійснюється розробка вхідних та вихідних лінгвістичних змінних.

2. Оцінка правил, яку можна розуміти як часткове неточне знання про якусь чітку функцію, так звані нечіткі правила ЯКЩО-ТО. В якості правил логічного висновку в алгоритмі Мамдані використовується операція мінімум (min), що «відсікає» функцію належності укладання правила за висотою, що відповідає обчисленому ступеню істинності передумови правила

3. Агрегація (композиція). На цьому етапі «відсікаються» функції належності змінної і створюються вихідні дані, які потім складаються разом, щоб утворити один нечіткий набір, при цьому використовується функція максимум.

4. Дефазифікація, яка передбачає процес транспонування нечітких виходів у чіткі виходи. На цьому етапі отримують чітке значення вихідної змінної. Дефазифікація може бути виконана:

- методом центру тяжкості;
- методом центру тяжкості для однокрапкових множин;
- методом центру площі;
- методом лівого модального значення;
- методом правого модального значення;
- методом першого максимуму;
- методом центрального максимуму

Для задачі оцінки ризику використання інформаційної системи були введені вхідні лінгвістичні змінні: надійність адміністрування інформаційної системи і швидкість знаходження виникаючих помилок. Надійність адміністрування інформаційної системи оцінюється як висока, середня, низька і визначається за 10-бальною шкалою (рис. 1).

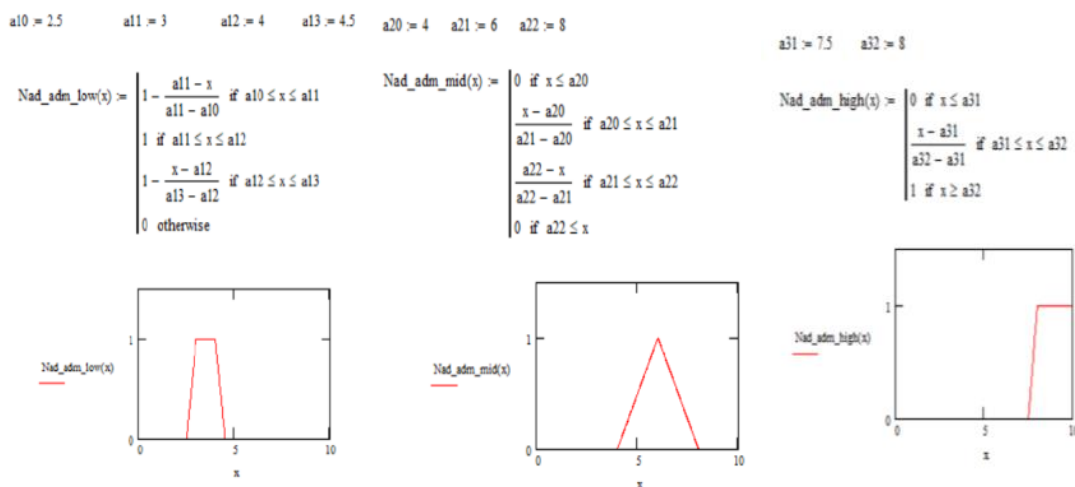


Рисунок 1. Функції належності лінгвістичної змінної «Надійність адміністрування»

Швидкість знаходження виникаючих помилок оцінюється за 7-бальною шкалою як: висока, середня, низька і визначається (рис. 2)

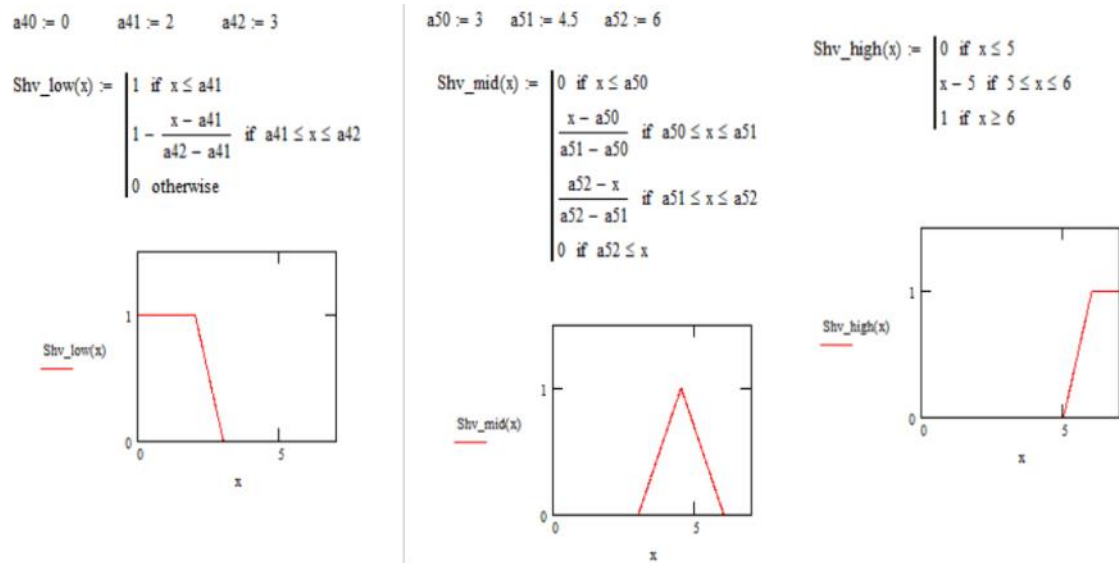


Рисунок 2. Функції належності лінгвістичної змінної «Швидкість знаходження виникаючих помилок»

Вихідною лінгвістичною змінною задачі є «Ризик використання інформаційної системи», яка оцінюється як високий, помірний, низький і визначається за 10-бальною шкалою

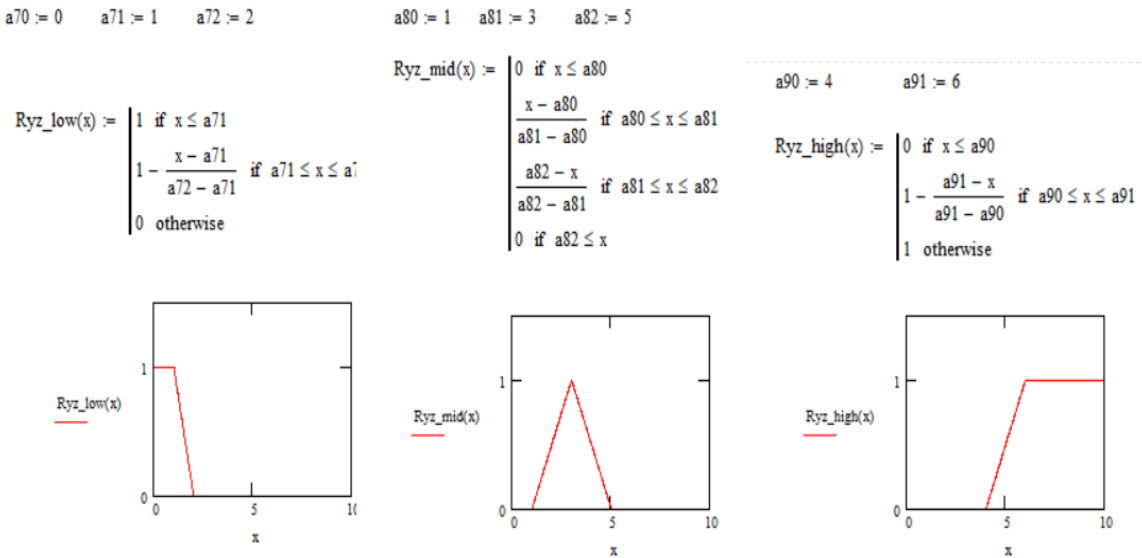


Рисунок 3. Функції належності лінгвістичної змінної «Ризик використання інформаційної системи»

З урахуванням лінгвістичних змінних визначаються нечіткі правила:

- Якщо Надійність адміністрування = «низька» або Швидкість знаходження помилок = «низька», то Ризик = «високий».
- Якщо Надійність адміністрування = «середня», то Ризик = «помірний».
- Якщо Надійність адміністрування = «висока» і Швидкість знаходження помилок = «середня», то Ризик = «помірний».

– Якщо Надійність адміністрування = «висока» і Швидкість знаходження помилок = «висока», то Ризик = «низький».

На етапі фазифікації визначаються значення функцій належності, що відповідають вхідним змінним (рис. 4)

Значення функцій належності, що відповідають вхідним змінним

$x_0 := 8$ $x_1 := 5.5$

$a_1 := \text{Nad\_adm\_high}(x_0)$ $b_1 := \text{Shv\_high}(x_1)$

$a_1 = 1$ $b_1 = 0.5$

$a_2 := \text{Nad\_adm\_mid}(x_0)$ $b_2 := \text{Shv\_mid}(x_1)$

$a_2 = 0$ $b_2 = 0.333$

$a_3 := \text{Nad\_adm\_low}(x_0)$ $b_3 := \text{Shv\_low}(x_1)$

$a_3 = 0$ $b_3 = 0$

Рисунок 4. Значення функцій належності, обчислені при заданих значеннях вхідних змінних

За алгоритмом Мамдані підсумкова нечітка підмножина для змінної виведення «Ризик використання інформаційної системи» з функцією належності визначається формулою:

$$\mu_{\Sigma} = \max\{c_1(z); c_2(z); c_3(z); c_4(z)\},$$

де $c_1(z), c_2(z), c_3(z), c_4(z)$ - знайдені усічені функції, отримані у відповідності з правилами:

$$c_1(z) = \min\{\alpha_1, \gamma\_h(n)\},$$

де α_1 - значення функції належності $\gamma\_h(n)$, що відповідає значенню вхідного параметру

$$c_2(z) = \min\{\alpha_2, \gamma\_m(n)\},$$

де α_2 - значення функції належності $\gamma\_m(n)$, що відповідає значенню вхідного параметру

$$c_3(z) = \min\{\alpha_3, \gamma\_m(n)\},$$

де α_3 - значення функції належності $\gamma\_m(n)$, що відповідає значенню вхідного параметру

$$c_4(z) = \min\{\alpha_4, \gamma\_l(n)\},$$

де α_4 - значення функції належності $\gamma\_l(n)$, що відповідає значенню вхідного параметру.

Дефазифікація здійснювалась на основі методу центру ваги
Результати виконання даного алгоритму показані на рисунку 5

Композиція

$$C1(z) := \min(\alpha1, Ryz\_high(z))$$

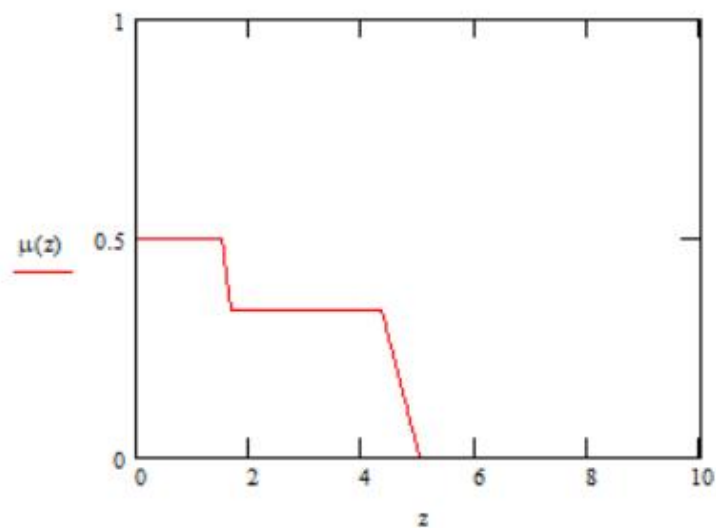
$$C2(z) := \min(\alpha2, Ryz\_mid(z))$$

$$C3(z) := \min(\alpha3, Ryz\_mid(z))$$

$$C4(z) := \min(\alpha4, Ryz\_low(z))$$

$$\mu(z) := \max(C1(z), C2(z), C3(z), C4(z))$$

Новий терм вихідної змінної "Ризик"



Дефазифікація за методом "центру ваги"

$$Ryz := \frac{\int_1^4 z \cdot \mu(z) dz}{\int_1^4 \mu(z) dz}$$

$$Ryz = 2.393$$

Рисунок 5. Результат виконання алгоритму

Таким чином ми отримуємо, що ризик використання інформаційної системи при оцінці надійності адміністрування інформаційної системи 8 балів та швидкості знаходження помилок становить 2,39 бали.

Висновки. Результати нашого дослідження показали, що алгоритми нечіткої логіки можуть бути успішно реалізовані в пакет програм Mathcad. Реалізація алгоритму Мамдані в цьому математичному пакеті забезпечує простоту та зручність його використання.

Розглянутий алгоритм розрахунків можна легко адаптувати до інших алгоритмів нечіткого виведення

Список використаних джерел:

1. Желдак Т.А., Коряшкіна Л.С., Ус С.А. Нечіткі множини в системах управління та прийняття рішень: навч. посіб./М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». Дніпро : НТУ «ДП», 2020. 387 с.
2. Зайченко, Ю.П. Нечіткі моделі і методи в інтелектуальних системах. К .: Слово, 2008. 344 с.
3. Матвійчук А.В. Моделювання економічних процесів із застосуванням методів нечіткої логіки: Монографія.– К.: КНЕУ, 2007.– 264 с.
4. Субботін С. О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень : навчальний посібник. Запоріжжя: ЗНТУ, 2008. 341с.
5. Волков В.Е., Макоед Н.О. Нечітка логіка та керування складними системами. Зб. тез доп. 75-ої наук. конф. викл. акад., Одеса, 20-24 квіт. 2015 р. Одес. нац. акад. харч. технологій; під заг. ред. Б.В. Єгорова. Одеса, 2015. С. 211-212.

УДК004.94

ПЛАНУВАННЯ ЕКСПЕРИМЕНТІВ З ІМІТАЦІЙНИМИ МОДЕЛЯМИ

Кучерков А.О., студент

e-mail: kucherkov.artem.1599@gmail.com

Науковий керівник: ст. викладач Зінов'єва О.Г.

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Для імітаційного моделювання дуже суттєве питання його ефективності, яка моделювання може оцінюватися рядом критеріїв, у тому числі точністю та достовірністю результатів моделювання, часом побудови та роботи з моделлю, витратами машинних ресурсів. Ефективність використання ресурсів істотно залежить від вибору плану експерименту. Основна мета експериментальних досліджень за допомогою імітаційних моделей полягає в найбільш глибокому вивченні поведінки системи, що моделюється. І тому потрібно планувати і проектувати як саму модель, а й процес її використання, тобто. проведення з нею експериментів на ЕОМ

Основні матеріали дослідження. Імітаційне моделювання є за своєю суттю машинним експериментом із моделлю досліджуваної або проектованої системи. Планування експерименту є процедурою вибору числа та умов проведення дослідів, необхідних і достатніх для вирішення поставленого завдання з необхідною точністю

Існує ще таке поняття як теорія планування експерименту, вона дозволяє за мінімальної кількості дослідів отримати математичну модель процесу і визначити оптимальні шляхи його розвитку. Основою теорії планування становлять математична статистика і теорія ймовірностей, оскільки результати експерименту переважно є випадковими величинами чи процесами. Причиною цього можуть бути неконтрольовані умови проведення експерименту, помилки спостережень, вимірювань тощо.

В даній роботі ми розглянемо як провести факторний експеримент з моделлю в пакеті імітаційного моделювання GPSS.

Система GPSS World забезпечує два види експериментів, що автоматично проводяться: розроблені для користувача та розроблені користувачем. До експериментів, розроблених для користувача, відносяться відсіваючий експеримент або дисперсійний аналіз, та оптимізаційний експеримент.

Відсіваючий експеримент використовується для визначення факторів, що істотно впливають на систему. Отримана інформація важлива для проведення подальших експериментів з метою створення більш ефективних оптимізаційних експериментів.

Сутність цього експерименту полягає у проведенні багатофакторного дисперсійного аналізу з метою виявлення ступеня впливу різних факторів та їх комбінацій (взаємодій) на значення цільової функції (функції відгуку, представленої у вигляді рівняння регресії).

Розглянемо проведення експерименту на прикладі системи обробки інформації. В системі є сервер і декілька терміналів, з яких поступають запити тільки після відповіді на попередній запит. За умовами задачі необхідно дослідити залежність середнього часу відповіді на запит від вхідних факторів при заданих їх мінімальних та максимальних значеннях. Вхідними факторами є кількість терміналів (N_TERM), середній час обробки запиту сервером (T_SERV) та

середній час формування запиту в терміналі (T_TERM) при заданих їх мінімальних та максимальних значеннях (рис. 1).

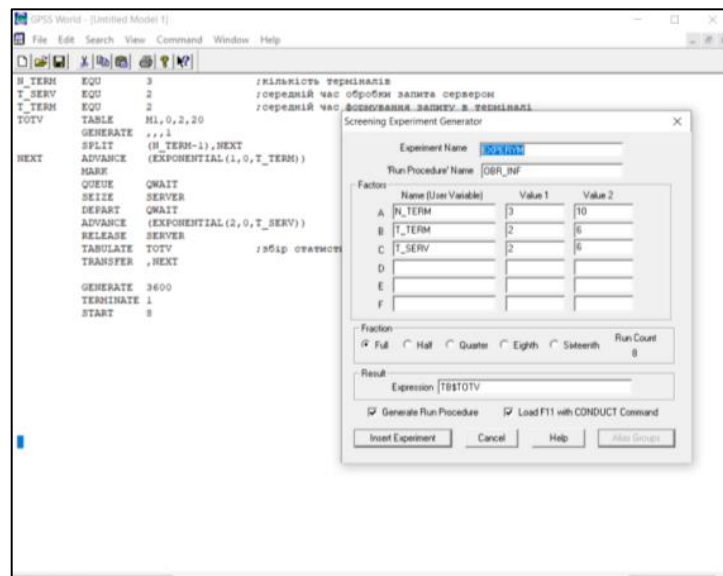


Рисунок 1. Налаштування параметрів експерименту в GPSS World

Для організації експерименту необхідно мати налагоджену імітаційну модель, яка повинна бути перевірена за всіляких значень вхідних змінних. Необхідно гарантувати, що модель правильно працюватиме при будь-якій комбінації рівнів факторів.

В результаті заповнення полів діалогового вікна до моделі автоматично буде додано код мовою PLUS.

Після заповнення діалогу в модель будуть вставлені такі процедури:

ScreenNetwork - власне експеримент, що відсіює;

ScreenNetworkGetResult - процедура виконання прогонів, що служить для фіксації результатів кожного прогону та переходу до наступного прогону;

DoTheRun - процедура прогону, яка служить для встановлення початкових значень генераторів випадкових чисел та запуску кожного нового прогону

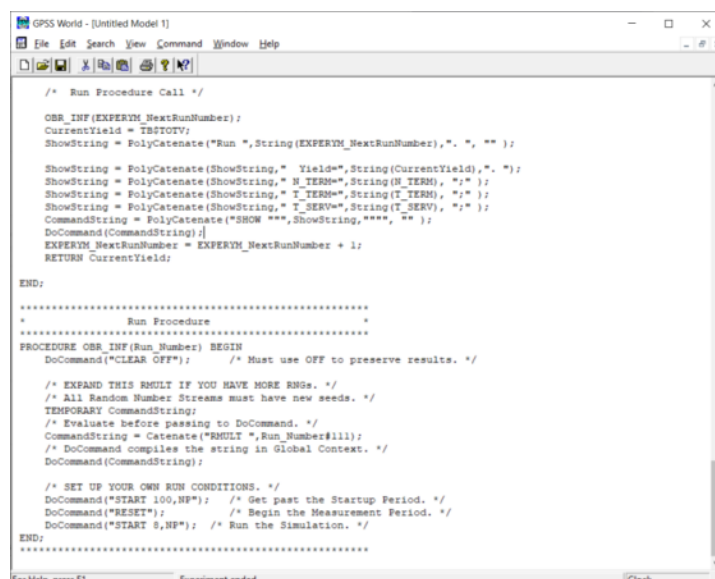


Рисунок 2. Фрагмент процедури PLUS

Після трансляції моделі та запуску згенерованого експерименту автоматично створюється звіт.

Остаточні результати моделювання після статистичної обробки будуть виведені у вигляді таблиці Anova (рис. 3)

12/08/22 17:50:45	Alias Group	Effect	Sum of Squares	Degrees of Freedom	F - for Only Main Effects	Critical Value of F [p=.05]
12/08/22 17:50:45						
12/08/22 17:50:45	A	26.883	1445.425	1	13.553	7.71
12/08/22 17:50:45	B	-3.083	19.007	1	0.178	7.71
12/08/22 17:50:45	AB	-1.126				
12/08/22 17:50:45	C	25.739	1325.002	1	12.424	7.71
12/08/22 17:50:45	AC	14.553				
12/08/22 17:50:46	BC	-0.464				
12/08/22 17:50:46	ABC	0.186				
12/08/22 17:50:46	Error		426.595	4		
12/08/22 17:50:46	Total		3216.029	7		
12/08/22 17:50:46	Grand Mean		22.719			
12/08/22 17:50:46						
12/08/22 17:50:46 Experiment ended.						

Рисунок 3. Результати дисперсійного аналізу

У таблиці кожен фактор та взаємодія факторів представлені окремим рядком. У кожному рядку для всіх факторів зазначені коефіцієнти, із якими вони входять у цільову функцію (стовпець Effect), а головних ефектів (A, B, C) - суми квадратів відхилень - стовпець Sum of Squares.

Чим більше значення F-статистики (F-for Only Main Effects), тим більше ефект. Ефект, отже, і чинник, вважається значним, якщо перевищує критичне значення (Critical Value of F(p=.05)).

У цьому прикладі для фактору фактори A і C є значущими, тому що їх F-статистики більше критичного значення, що дорівнює 7.71

Висновки. Таким чином, за результатами моделювання можна дійти невтішного висновку, що з даному потоці і характеристиці сервера середній час виконання запитів становить 22,791 ч. Для зменшення втрат часу необхідно продовжити дослідження кожного значущого чинника.

Список використаних джерел:

1. Томашевський В.М. Моделювання систем. К.: Видавнича група BVH, 2005. 352 с.
2. Ситник В. Ф., Орленко Н.С. Імітаційне моделювання: навч. посіб. Київ : КНЕУ, 1998. 232 с.
3. Кельтон В., Лоу А. Имитационное моделирование. Киев: Издательская группа BVH, 2004. 847 с.
4. Братушка С.М. Імітаційне моделювання як інструмент дослідження складних економічних систем// Вісник Української академії банківської справи, 2009. № 2(27), С. 113–118.
5. Жерновий Ю.В. Імітаційне моделювання систем масового обслуговування: практикум. Львів : ЛНУ імені Івана Франка, 2007. 307 с

УДК519.816

ЗАДАЧІ ОПТИМІЗАЦІЇ З НЕЧІТКИМИ ЗМІННИМИ

Мартиць Д.С., студент

e-mail: danielmartits@gmail.com

Науковий керівник: Зінов'єва О.Г., ст. викл.

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Однією з умов для побудови математичної моделі є вимога адекватності, тобто відповідності моделі процесу, який розглядається. Але не завжди існують моделі, де дані представлені однозначно. Є системи, де людина грає важливу роль і вихідні дані є досить суб'єктивними. На цей час одним з напрямків в області аналізу та побудові математичних моделей економічних явищ є нечітка логіка. Головна мета нечіткого математичного програмування - допомогти особі, що ухвалює рішення, прийняти оптимальне та адекватне рішення. Нечіткий підхід полегшує завдання, дозволяючи не формулювати точні цілі та обмеження, які для реальних завдань, як правило, точно не відомі.

Основні матеріали дослідження. При моделюванні задач прийняття оптимальних рішень цільова функція та її параметри можуть бути представлені в нечіткій формі. Тобто задача стандартного математичного програмування перетворюється в задачу нечіткого математичного програмування.

Стандартна задача нечіткого математичного програмування – це задача максимізації (або мінімізації) заданої функції на заданій множині допустимих альтернатив, яке описується системою нерівностей або рівнянь.

Наприклад,

$$\begin{aligned} f(x) &\rightarrow \max, \\ \begin{cases} \varphi_i(x) \leq 0, & i = \overline{1, m}, \\ x \in X, \end{cases} \end{aligned}$$

де $f(x)$ - цільова функція,

$\varphi_i(x)$ - задана функція обмежень,

X - задана множина альтернатив.

Припустимо a - це задане значення цільової функції $f(x)$, досягнення якого є достатнім для прийняття оптимального рішення, b - нижній рівень, такий що нерівність $f(x) < a - b$ означає сильне порушення нерівності $f(x) \geq a$. Тоді функція приналежності для нечіткої цільової функції визначається наступним чином

$$c(x) = \begin{cases} 0, & \text{якщо } f(x) \leq a - b, \\ c_a(x), & \text{якщо } a - b < f(x) < a, \\ 1, & \text{якщо } f(x) \geq a, \end{cases}$$

яка описує ступені виконання відповідної нерівності для прийняття оптимального рішення.

Алгоритм розв'язання задачі з нечіткими коефіцієнтами має наступний вигляд:

1. Початкове завдання.
2. Вводимо дискретні α -рівні.
3. Обмеження приймають інтервальний вигляд.
4. Записуємо нерівності окремо по лівому і правому краях з урахуванням знаків нерівності (при цьому розмірність збільшується).
5. Отримуємо задачу лінійного програмування з чіткими коефіцієнтами.
6. Вирішуємо отриману задачу симплекс-методом.

Задачу оптимізації з нечіткими змінними можна розглянути на прикладі задачі про відбір кандидатів на вакантну посаду.

Відбір персоналу зазвичай проводиться поетапно. на кожному етапі відкидається частина кандидатів.

На першому і другому етапах проводиться тест, що дозволяє оцінити ступінь особистих якостей, і класифікація професій.

На третьому етапі за допомогою методу експертного оцінювання визначені оптимальні ступені володіння особистими якостями для кожної сфери діяльності. Для кожної групи професій обчислюються середні показники загальної думки експертів, і далі в математичній моделі вибору кандидата використовують медіану.

На базі отриманих результатів із застосуванням апарату нечітких множин розробляється математична модель оптимізації вибору кандидату на вакансію.

В даній моделі змінні x_{ij} - особисті фактори i -го кандидату (тобто нечіткі змінні: товариськість, емоційна стійкість, інтелект і т.п.). Використовуючи визначену експертним методом оптимальну ступінь володіння особистими якостями, формують обмеження – нечіткі множини, які є вимогами до кандидата. Обчислив цільову функцію, отримують оптимальне розв'язання відповідно до того, хто з претендентів найбільш підходить до вакансії за своїми особистими якостями.

Ступені володіння особистими якостями складають множину можливих значень x_{ij} ($i = \overline{1, n}; j = \overline{1, m}$). Дані нечіткі змінні характеризуються трьома характеристиками $(x_{ij}, X, \varphi(x_{ij}))$, $i = \overline{1, n}; j = \overline{1, m}$, де x_{ij} - найменування особистого фактору, X - множина значень x_{ij} (в діапазоні від 1 до n).

Формуємо обмеження – нечіткі множини $\varphi(x_{ij})$ ($i = \overline{1, n}; j = \overline{1, m}$) - вимоги до кандидату. Міра, з якою задовольняються обмеження $\varphi(x_{ij})$ називається сумісністю значення x_{ij} з $\varphi(x_{ij})$ та позначається $c_{x_{ij}}$ ($i = \overline{1, n}; j = \overline{1, m}$).

Позначимо оптимальну ступінь володіння j -ю якістю як u_j^* . Тоді нечітке обмеження для i -го кандидату та j -го особистого фактору визначається з наступного виразу:

$$c_{x_{ij}} = \begin{cases} 1 - \frac{|x_{ij} - u_j^*|}{(n+1) - x_{ij}}, & x_{ij} < \frac{n}{2}, \\ 1 - \frac{|x_{ij} - u_j^*|}{x_{ij}}, & x_{ij} > \frac{n}{2}. \end{cases}$$

Ця рівність дозволяє визначити значення $c_{x_{ij}}$ - сумісність значення x_{ij} з обмеженням за допомогою підстановки відповідних x_{ij} та u_j^* для кожного фактору.

Цільова функція $f(x) = \sum_{j=1}^n c_{x_{ij}} w_j \quad (i = \overline{1, n}; \quad j = \overline{1, m})$ - ступінь

відповідності ідеальному працівнику, де $w_j \quad (j = \overline{1, m})$ - вагові коефіцієнти кожного з компонентів X , отримані за допомогою експертних оцінок.

Максимальне значення $f(x) = 1$, тобто чим ближче цільова функція $f(x)$ до 1, тим більше кандидат відповідає вимогам.

На основі отриманих результатів кадровою службою підприємства приймається рішення по вибору кандидата на вакантну посаду.

На рис. 1 наведено отримання результату для першого кандидата на вакансію, де X - це набір ступенів володіння особистими якостями цього кандидату.

	A	B	C	D	E	F
1	Найменування фактору x_{ij}	X	u_j^*	w_j	$c_{x_{ij}}$	
2	A - замкнутість/товариськість	7	8	0,0762	0,85714	
3	B - інтелект	8	9	0,0857	0,875	
4	C - емоційна нестійкість	10	9	0,0857	0,9	
5	E - підлеглість	6	7	0,0667	0,83333	
6	F- стриманість	4	3	0,0286	0,5	
7	G - чуйність	9	9	0,0857	1	
8	H - боязливість/сміливість	7	6	0,0571	0,85714	
9	I - жорсткість	5	5	0,0476	1	
10	L - довірливість	3	5	0,0476	0,33333	
11	M - практичність	6	5	0,0476	0,83333	
12	N - прямолінійність	6	8	0,0762	0,66667	
13	O - впевненість в собі	3	2	0,0190	0,66667	
14	Q - консерватизм	6	6	0,0571	1	
15	P - конформізм	5	6	0,0571	0,8	
16	R - низький/високий самоконтроль	10	9	0,0857	0,9	
17	S - розслабленість/напруженість	4	8	0,0762	-1	
18		$\Sigma =$	105			
19				F=	0,69246	
20						
21						
22						

Рисунок 1. Результат розв'язання задачі із використанням табличного процесору Excel

Аналогічним чином проводимо обчислення для останніх кандидатів, і при порівнюванні цільової функції $f(x)$ отримуємо рішення відносно того, хто з претендентів найбільше підходить для даної вакансії по своїм особистим якостям.

Висновки. Таким чином процедура оцінки особистих якостей кандидата спрощується до визначення ступенів володіння якостями та підстановки результатів тестування в математичну модель.

Список використаних джерел:

1. Кондратенко Ю. П. Методи обробки нечіткої лінгвістичної інформації в задачах багатокритерійного прийняття рішень / Ю. П. Кондратенко, Є. В. Сіденко // Матер. 6-ї Міжн. наук.-практ. конф. Сучасні інформаційні та інноваційні технології на транспорті MINTT2014. – Херсон, Травень 2014. – С. 161–163..
2. Ямпольський Л. С. Системи штучного інтелекту в плануванні, моделюванні та управлінні / Л. С. Ямпольський, Б. П. Ткач, О. І. Лісовиченко. – Київ : ДП «Видавничий дім «Персонал», 2011. – 544 с..
3. Волошин О. Ф. Моделі та методи прийняття рішень : навч. посіб. Для студ. вищ. навч. закл. / О. Ф. Волошин, С. О. Мащенко. – 2-ге вид., перероб. та допов. – К. : Видавничо-поліграфічний центр "Київський університет", 2010. – 336 с.

УДК 515.8 + 004.9

РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВИКОРИСТАННЯ СИСТЕМИ АВТОМАТИЗОВАНОГО ПРОЕКТУВАННЯ ТЕХНОЛОГІЧНОЇ ДОКУМЕНТАЦІЇ НА ПІДПРИЄМСТВІ СІЛЬСЬКОГОСПОДАРСЬКОГО МАШИНОБУДУВАННЯ

Мацулевич О.Є., к.т.н., доцент

e-mail: oleksandr.matsulevych@tsatu.edu.ua

Дереза О.О., к.т.н., доцент

e-mail: olena.dereza@tsatu.edu.ua

Тетервак І.Р., асистент

e-mail: illia.tetervak@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність дослідження та постановка проблеми. Основними чинниками, що визначають економічний ефект підсистеми автоматизованого проектування, є: можливість комплексного вирішення всіх завдань технологічної підготовки виробництва в автоматизованому режимі. Проте важливу роль грає і людський чинник - фахівці, які, використовуючи програмно - апаратні засоби, реалізують в підсистемі автоматизованого проектування свої ідеї і творчі здібності. В цьому відношенні успіх залежить від правильності підбору і розстановки кадрів, їх кваліфікації, а також від ретельності опрацювання організаційно - системних питань функціонування підсистеми.

Основні матеріали дослідження. Річний економічний ефект E , що отримується при функціонуванні підсистеми автоматизованого проектування технологічної документації визначається за формулою:

$$E = E_p - E_n \cdot K_a, \quad (1)$$

де E_p – річна економія після впровадження проекту;

E_n – нормативний коефіцієнт ефективності капітальних вкладень (для автоматизованих систем управління і проектування $E_n=0.333$. Він визначається за формулою;

$$E_n = 1 / T_{\text{нок}}, \quad (2)$$

де $T_{\text{нок}}$ – нормативний термін окупності капітальних вкладень (для засобів автоматики і обчислювальної техніки рівний 3 роки);

K_a – повні одноразові витрати на створення спроектованої системи.

$$K_a = K_n + K_k, \quad (3)$$

де K_n – витрати на розробку системи;

K_k – капіталовкладення в комплект програмних і технічних засобів.

$$K_n = Z_n \cdot T_n \quad (4)$$

де Z_n – місячна заробітна плата розробнику проекту підсистеми автоматизованого проектування (розраховується як сума оплати праці розробника);

T_n – тривалість проектування ($T_n = 6$ місяців – тривалість виконання дипломного проекту).

Місячна заробітна плата розробнику проекту підсистеми автоматизованого становить 0 гривень. Тому по $K_n = 0$ гривень.

$$K_K = B_T + B_{II}, \quad (5)$$

де B_T – витрати на придбання комплексу технічних засобів (розраховується, як сума ринкових цін на технічне забезпечення);

B_{II} – витрати на придбання комплексу програмних засобів (розраховується, як сума ринкових цін на програмне забезпечення).

Повні витрати на придбання технічних засобів підсистеми автоматизованого включають:

- витрати на технічне забезпечення кожного автоматизованого робочого місця (АРМ) ($B_{ТАРМ}$);
- витрати на технічне забезпечення обчислювальної мережі ($B_{ТМ}$);
- витрати на монтаж технічних засобів та мережі ($B_{Тмонт}$).

Вартість технічного забезпечення одного АРМ відділу головного технолога підприємства ГРП «АвтоЗАЗ - Мотор» становить 6300 гривень. Всього автоматизованих робочих місць 20, тому $B_{ТАРМ} = 126000$ гривень. Вартість технічного забезпечення обчислювальної мережі ($B_{ТМ}$) складає 18879 грн. Витрати на монтаж технічних засобів та мережі ($B_{Тмонт}$) становлять 2500 гривень. Таким чином, витрати на придбання технічного забезпечення

$$B_T = B_{ТАРМ} + B_{ТАРМ} + B_{Тмонт} = 147379 \text{ гривень.}$$

Повні витрати на придбання програмного забезпечення підсистеми автоматизованого включають:

- витрати на програмне забезпечення кожного АРМ ($B_{ПАРМ}$);
- витрати на програмне забезпечення мережі ($B_{ПМ}$);
- витрати на встановлення та налагодження програмного забезпечення ($B_{Пінст}$);

Витрати на кожен комплект програмного забезпечення наведено у таблиці 1.

Таблиця 1 - Витрати на програмне забезпечення по кожному АРМ

Комплектація пакету	Вартість, гривень	Кількість, штук
АРМ технолога: «Механообробка» Включає «ВЕРТИКАЛЬ – технологія 3»; спеціалізований клієнт «ЛОЦМАН:PLM» для «ВЕРТИКАЛЬ – Технологія 3»; корпоративний довідник «Матеріали і сортаменти 2.0» (клієнт); систему	21690	11

Комплектація пакету	Вартість, гривень	Кількість, штук
розрахунку режимів різання.		
АРМ технолога: «Зварювання» Включає «ВЕРТИКАЛЬ – технологія 3»; спеціалізований клієнт «ЛОЦМАН:PLM» для «ВЕРТИКАЛЬ – Технологія 3»; корпоративний довідник «Матеріали і сортаменти 2.0» (клієнт); режими зварювання: дугове зварювання покритим електродом; система адміністрування зварювальних конструкторсько – технологічних елементів.	18975	1
АРМ технолога: «Збирання» Включає «ВЕРТИКАЛЬ – Технологія 3», спеціалізований клієнт «ЛОЦМАН:PLM» для «ВЕРТИКАЛЬ – Технологія 3»	24420	2
АРМ конструктора: «Оснастка стандарт» Включає «Систему проектування специфікацій», «Бібліотеку стандартних виробів», «Бібліотеку стандартних виробів: елементи верстатних пристосувань»	14580	4
АРМ конструктора: «Оснастка професіонал» Включає «Систему проектування специфікацій», «Бібліотеку конструктора штампів», «Бібліотеку конструктора прес-форм».	22350	1
АРМ технолога: «Ливарна оснастка» Включає «ВЕРТИКАЛЬ – Технологія 3», спеціалізований клієнт «ЛОЦМАН:PLM» для «ВЕРТИКАЛЬ – Технологія 3», «LVMFlow 2009».	29456	2

Таким чином загальні витрати на програмне забезпечення всіх АРМ у підсистемі ($B_{ПАРМ}$) становлять 445987 гривень.

Витрати на програмне забезпечення мережі ($B_{ПМ}$) на пропонованому підприємстві становлять 7420 гривень. Витрати на встановлення та налагодження програмного забезпечення становлять 1600 гривень.

Повні витрати на програмне забезпечення:

$$B_{П} = B_{ПАРМ} + B_{ПМ} + B_{Пінст} = 455007 \text{ гривень.}$$

Капіталовкладення в комплект програмних і технічних засобів (K_K) визначаються за формулою (5) і дорівнюють 378278 гривням.

Повні одноразові витрати на створення підсистеми автоматизованого проектування (K_a) визначаються за формулою (3) і дорівнюють 602386 гривням.

Розрахунок річної економії після впровадження проекту підсистеми автоматизованого проектування технологічної документації на підприємстві ГРП «АвтоЗАЗ - Мотор» визначається за формулою:

$$E_p = P_2 - P_1, \quad (6)$$

де P_1 , – прибуток від проектних робіт до впровадження підсистеми;
 P_2 – прибуток від проектних робіт після впровадження підсистеми.

$$P_1 = B_1 - ВСП_1, \quad (7)$$

$$P_2 = B_2 - ВСП_2, \quad (8)$$

де $ВСП_1$ – виробнича собівартість проекту до впровадження підсистеми;
 $ВСП_2$ – виробнича собівартість проекту після впровадження підсистеми;
 B_1 – річні надходження виплат за проведення проектних робіт до впровадження підсистеми;
 B_2 – річні надходження виплат за проведення проектних після впровадження підсистеми.

$$B_1 = m_1 \cdot ДВ, \quad (9)$$

$$B_2 = m_2 \cdot ДВ, \quad (10)$$

де $ДВ$ – договірна вартість одного проекту;
 m_1 – кількість проектів, що розробляється до впровадження підсистеми;
 m_2 – кількість проектів що розробляється, після впровадження підсистеми;
З попереднього обстеження ВГТ кількість проектів, що розробляється у існуючій підсистемі проектування технологічної документації становить $m_1=430$ проектів на рік.

Після впровадження системи автоматизованого проектування технологічної документації на запропонованому підприємстві, згідно з технічним завданням, кількість розроблених проектів повинна збільшитися на 25%, тобто $m_2 = 527$ проектів на рік. Договірна ціна одного проекту становить 2000 гривень. Таким чином, за формулою (9) вартість проектів до впровадження (B_1) складає 840000 гривень, а вартість проектів після впровадження (B_2) – 1050000 гривень.

Розрахуємо виробничу собівартість проекту до і після впровадження підсистеми автоматизованого проектування технологічної документації.

$$ВСП_1 = \Phi ОП_1 + О_{ТЧ} \cdot \Phi ОП_1 + M_1 + A_{нрим1} + C_1 + H_1, \quad (11)$$

$$ВСП_2 = \Phi ОП_2 + О_{ТЧ} \cdot \Phi ОП_2 + M_2 + A_{ном2} + C_2 + H_2, \quad (12)$$

де $\Phi ОП_1$ – річний фонд оплати праці до впровадження підсистеми;
 $\Phi ОП_2$ – річний фонд оплати праці після впровадження підсистеми;
 $О_{ТЧ}$ – нормативні відрахування від фонду оплати праці (34%);
 $М_1$ – річна вартість матеріалів, що витрачаються, і оплата електроенергії до впровадження підсистеми ($М_1 = 3800$ гривень);

$М_2$ – річна вартість матеріалів, що витрачаються, і оплата електроенергії після впровадження ($М_2 = 2300$ гривень);

$А_{прим\_1}$ – річна вартість експлуатації приміщень для працівників ВГТ до впровадження підсистеми (площа, необхідна для розміщення групи проектувальників, що виконують роботи в ручному режимі, більше, ніж для розміщення проектувальників тих, що експлуатують підсистему автоматизованого проектування);

$А_{прим\_2}$ – річна вартість експлуатації приміщень для працівників ВГТ після впровадження підсистеми;

$С_1$ – річні витрати на обслуговування комплексу технічних засобів до впровадження підсистеми ($С_1 = 7000$ гривень);

$С_2$ – річні витрати на обслуговування комплексу технічних засобів після впровадження підсистеми ($С_1 = 2400$ гривень);

$Н_1$ – накладні витрати ($Н_1 = 0,4 \cdot \Phi ОП_1$) до впровадження підсистеми;

$Н_2$ – накладні витрати ($Н_2 = 0,4 \cdot \Phi ОП_2$) після впровадження підсистеми.

Штат працівників ВГТ становить 20 працівників. Ставка головного технолога становить 2200 гривень, ставка всіх інших працівників становить 1500 гривень, ставка замісників головного технолога становить 2000 гривень. Таким чином, річний фонд оплати праці діючої підсистеми становить

$$\Phi ОП_1 = (2200 + 19 \cdot 1500 + 2 \cdot 2000) \cdot 12 = 416400 \text{ гривень.}$$

Впровадження підсистеми автоматизованого проектування технологічної документації приводить до зменшення штату ВГТ, а саме до скорочення посад замісників головного технолога. Відповідно заробітна плата головного технолога повинна підвищитися до 3000 гривень, а спеціалістів до 1700 гривень.

Річний фонд заробітної плати після впровадження підсистеми становить

$$\Phi ОП_2 = \Phi ОП_1 = (2800 + 19 \cdot 1600) \cdot 12 = 398400 \text{ гривень.}$$

Розрахуємо річну вартість експлуатації приміщень для працівників ВГК до і після впровадження підсистеми:

$$A_{\text{прим}_1} = S \cdot B_1 \cdot C_a, \quad (13)$$

$$A_{\text{прим}_2} = S \cdot B_2 \cdot C_a, \quad (14)$$

де S – площа, що використовується для розміщення одного працівника;

C_a – вартість використання 1 м²;

B_1 – число працівників ВГТ до впровадження підсистеми ;

B_2 – число працівників ВГТ після впровадження підсистеми;

На даний момент площа приміщення ВГТ, що використовується становить 180м². Вартість використання одного квадратного метра на рік становить $C_a = 276$ гривні. Тобто, витрати на приміщення за формулою (13) становлять $A_{\text{прим}_1} = 49680$ гривень.

Впровадження автоматизованої системи проектування технологічної документації передбачує зменшення площі, необхідної для розміщення одного працівника до 6 м², тобто до 120 м² для розміщення всіх працівників. Таким чином, за формулою (14) витрати на приміщення після впровадження підсистеми становлять $A_{\text{прим}_2} = 33120$ гривень.

За формулою (11) виробничу собівартість проекту до впровадження підсистеми автоматизованого проектування технологічної документації становить:

$$BC\Pi_1 = 793344;$$

Виробничу собівартість проекту після впровадження підсистеми за формулою (12) становить:

$$BC\Pi_2 = 739004.$$

Таким чином, річна економія після впровадження проекту підсистеми автоматизованого проектування технологічної документації за формулою (6) дорівнює $E_p = 264340$ гривень.

Час окупності одноразових витрат на підсистему проектування технологічної документації розраховується за формулою.

$$T_{\text{фок}} = K_a / E_p, \quad (15)$$

де $T_{\text{фок}}$ – фактичний термін окупності капітальних вкладень, причому підсистема є ефективною, якщо $T_{\text{фок}} \leq 3$ років.

Час окупності підсистеми, визначений за формулою (15) становить 2,28 року.

Висновки. За пропонованою у роботі методикою визначено ефективність розробленої підсистеми автоматизованого проектування технологічної документації та строки її окупності.

Список використаних джерел:

1. ГОСТ 34.602 - 89 «Інформаційна технологія. Комплекс стандартів на автоматизовані системи. Технічне завдання на створення автоматизованої системи».

2. Мацулевич О.Є., Вершков О.О., Холодняк Ю.В., Дмитрієв Ю.О., Чаплінський А.П. Розробка мурашиного алгоритму для оптимізації оперативного планування робіт по збиранню врожаю кісточкових. *Плодовий сад – новітнє в теорії та практиці*: матеріали V Всеукр. наук.-практ. інтернет-конференції. Мелітополь: ТДАТУ, 2021. С. 106-110.

УДК 515.8 + 004.9

АНАЛІЗ ДОСЛІДНОЇ ЕКСПЛУАТАЦІЇ ПРОГРАМНОГО МОДУЛЮ РОЗРАХУНКУ НОРМ ЧАСУ ОБРОБКИ ДЕТАЛЕЙ СІЛЬСЬКОГОСПОДАРСЬКОЇ ТЕХНІКИ

Вершков О.О., к.т.н., доцент

e-mail: oleksandr.vershkov@tsatu.edu.ua

Бондаренко Л.Ю., к.т.н., доцент

e-mail: larysa.bondarenko@tsatu.edu.ua

Антонова Г.В., ст. викладач

e-mail: galina.antonova@tsatu.edu.ua

Тетервак І.Р., асистент

e-mail: illia.tetervak@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність дослідження та постановка проблеми. Автоматизація розрахунку норм праці при проектно-конструкторських роботах із застосуванням найсучасніших комп'ютерних засобів проводиться, в даний час, багатьма організаціями. Це обумовлено вимогами сучасного ринку та напряму пов'язане з конкурентним протистоянням між виробниками. Частіше створюються системи розрахунку операційних норм часу для умов індивідуальної організації праці. Але, на теперішній час, вкрай необхідні такі інформаційні системи, які в змозі забезпечити відповідні розрахунки для серійних та масових виробництв. Подібні системи створюються іноді як автономні системи нормування на базі заданого трудового і технологічного процесу.

Основні матеріали дослідження. Розроблений авторами програмний модуль визначення норм часу та заробітної плати деталей надає можливість автоматизувати процес розрахунку норм часу та заробітної плати на підприємстві навіть для станків з ручним управлінням.

Слід зазначити, що в основі структури пропонованого програмного забезпечення покладено використання структурованої бази 1С даних, яку було надано керівництвом підприємства ТОВ «Мелітопольський завод турбокомпресорів – ТУРБОКОМ» та представлено на рисунку 1.

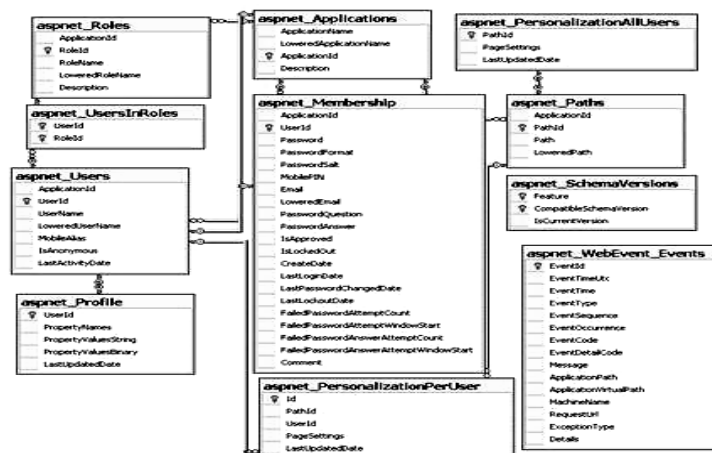


Рисунок 1. Структурована база даних

Програмний модуль створено таким чином, що дозволяє нормувальнику та бухгалтеру отримати більш достовірні розрахунки не використовуючи ні яких інших програмних продуктів.

Для початку роботи необхідно загрузити до програми файл із вхідними даними. На рисунку 2 наведено головне вікно програми.

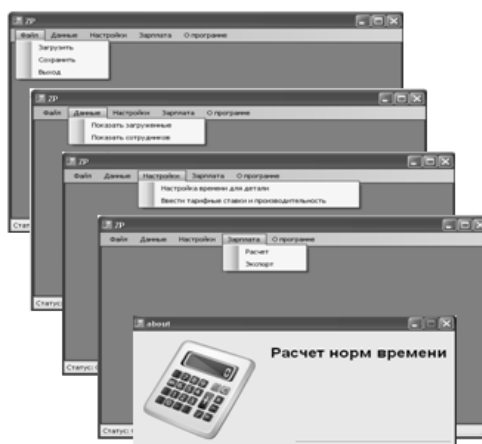


Рисунок 2. Головне вікно програми

Для того, щоб загрузити дані в програму, необхідно відкрити головне вікно і виконати наступну послідовність дій: Файл -> Загрузить та обрати дані для завантаження. В нашому випадку ми обрали імпортований з 1С файл Microsoft Excel з переліком співробітників та інформацією про їх працю.

Після завантаження отримані дані необхідно вивести на екран, для цього виконаємо команду: Данные -> Показать загруженные і одразу ж бачимо, що дані мають неупорядковану форму, щоб привести їх до потрібного виду ми виконаємо наступну послідовність дій: Данные -> Показать сотрудников, після чого програма автоматично відбере інформацію, яка стосується лише співробітників та їх графіку праці.

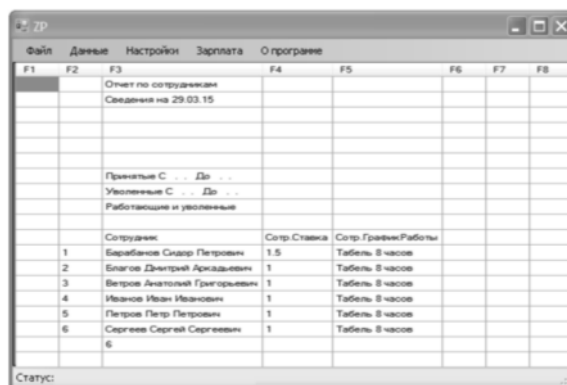


Рисунок 3. Імпортування даних у програму

Користувач має змогу корегувати вхідні дані та зберегти відредагований файл даних використовуючи кнопки «Сохранить». Операції для роботи з файлами продубльовано в розділі «Файл» головного меню програми. Для подальшої роботи необхідно натиснути вкладку : Настройки -> Настройки времени для детали, після чого з'явиться нове вікно для вибору виду обробки. У цьому вікні є можливість обрати, як саме користувач хоче розрахувати норми часу.

Файл	Данные	Настройки	Зарплата	О программе
№	ФИО	Ставка	Смена, ч.	
1	Барабанов Сидор Петрович	1.5	8	
2	Благов Дмитрий Аркадьевич	1	8	
3	Ветров Анатолий Григорьевич	1	8	
4	Иванов Иван Иванович	1	8	
5	Петров Петр Петрович	1	8	
6	Сергеев Сергей Сергеевич	1	8	

Рисунок 4. Упорядкована таблиця даних

Програмний модуль надає три можливості для вирішення поставленої задачі, а саме:

- розрахувати норми часу у програмному модулі;
- завантажити необхідні дані з операційної карти;
- ввести дані вручну.

Користувач обирає, який з наведених варіантів йому підходить найбільше та приступає до наступного етапу розрахунків.

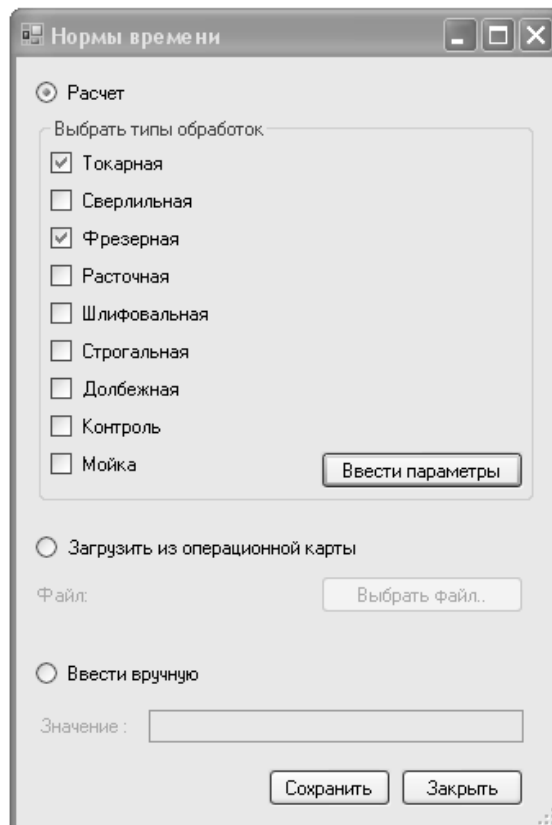


Рисунок 5. Вікно вибору виду обробки

Якщо розрахунок норм часу буде вирішено робити у самій програмі, то для цієї операції необхідно використати ще одне додаткове вікно для вводу параметрів. Користувач має обрати, для якого виду обробки йому потрібно розрахувати норми часу, виділити обрані поля та приступати до вводу параметрів вручну.

Параметры операции

Токарная

$T = T_o + T_v + T_{доп} + T_{пз} / N_{шт}$

Диаметр, d :

Длина, L :

Число проходов, i :

Скорость резания, V : 57.8

Подача, S : 0.5

Вспомогательное время, Тв : 0.042

Дополнительное время, Тв : 0.009

Подготов.-Заклуч. время, Тв : 16

OK

Рисунок 6. Вікно вводу параметрів розрахунку

Після того, як усі необхідні поля були заповнені, необхідно натиснути кнопку «Сохранить», щоб перейти до наступного кроку розрахунків. Програма повертає нас до початкової таблиці з даними, де у головному вікні програми треба натиснути: Настройки → Ввести тарифные ставки и производительность.

У новому вікні спочатку треба визначитися, яким чином буде розрахована заробітна плата. Можна ввести параметри для кожного співробітника окремо, а можна ввести однакові значення для всіх. Після підтвердження вибору необхідно натиснути кнопку «Сохранить».

Після цього можна переходити до розрахунку заробітної плати. Для цього тиснемо: Зарплата → Расчет. Перед нами з'явиться ще одне вікно для вводу параметрів, де необхідно буде ввести кількість робочих днів у місяці та процент за перевиконання. Після цього натискаємо кнопку «Рассчитать» і через декілька секунд ми побачимо нове вікно з розрахованою таблицею результатів.

ZP

Файл Данные Настройки Зарплата О программе

№	ФИО	Ставка	Смена, ч.
1	Барабанов Сидор Петрович	1.5	8
2	Благов Дмитрий Аркадьевич	1	8
3	Ветров Анатолий Сергеевич	1	8
4	Иванов Иван Иванович	1	8
5	Петров Петр Петрович	1	8
6	Сергеев Сергей Сергеевич	1	8

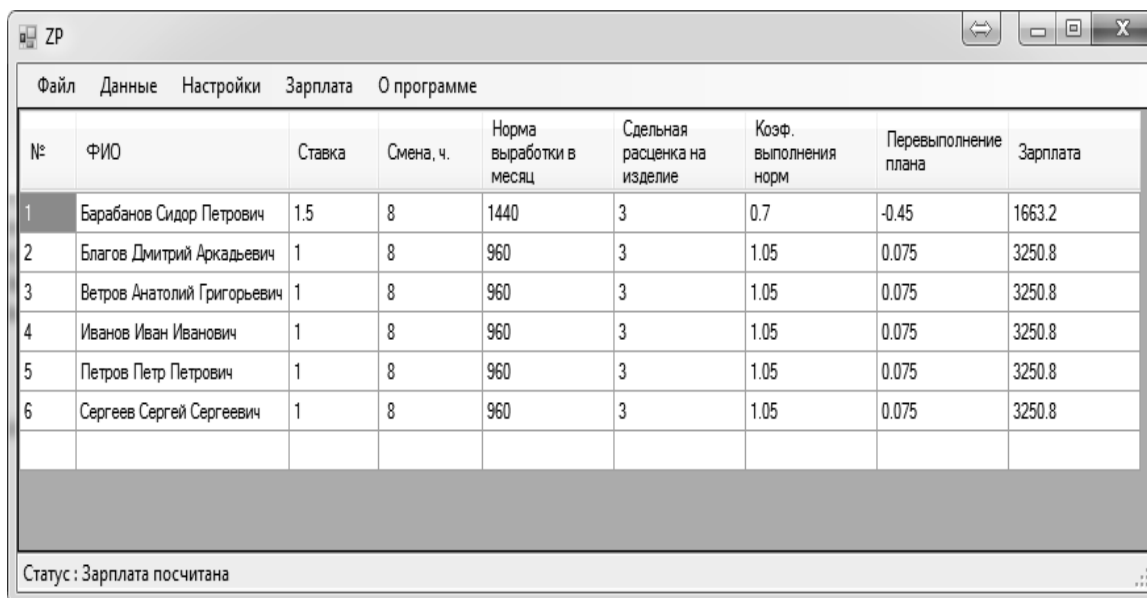
Расчет ЗП

Рабочих дней в месяце : 24

Процент за перевыполнение : 1.5

Рассчитать зарплату

Рисунок 7. Вікно введення параметрів розрахунку



№	ФИО	Ставка	Смена, ч.	Норма выработки в месяц	Сдельная расценка на изделие	Коэф. выполнения норм	Перевыполнение плана	Зарплата
1	Барабанов Сидор Петрович	1.5	8	1440	3	0.7	-0.45	1663.2
2	Благов Дмитрий Аркадьевич	1	8	960	3	1.05	0.075	3250.8
3	Ветров Анатолий Григорьевич	1	8	960	3	1.05	0.075	3250.8
4	Иванов Иван Иванович	1	8	960	3	1.05	0.075	3250.8
5	Петров Петр Петрович	1	8	960	3	1.05	0.075	3250.8
6	Сергеев Сергей Сергеевич	1	8	960	3	1.05	0.075	3250.8

Статус: Зарплата посчитана

Рисунок 8. Виведення результатів розрахунку

Заключним етапом процесу розробки програмного модулю є його тестування, тобто дослідження з метою отримання інформації про якість програмного продукту, а саме відповідність специфікації, технічному завданню, або вимогам замовника програмного модуля.

Серед видів тестування програмного забезпечення слід виділити:

- функціональне тестування (Functional Testing) розглядає заздалегідь зазначену поведінку і ґрунтується на аналізі специфікацій функціональності компонента або системи в цілому. Функціональні тести ґрунтуються на функціях, виконуваних системою, і можуть проводитися на всіх рівнях тестування (компонентному, інтеграційному, системному, приймальному). Як правило, ці функції описуються у вимогах, функціональних специфікаціях або у виді випадків використання системи (use cases);

- тестування взаємодії (Interoperability Testing) – це функціональне тестування, що перевіряє здатність додатка взаємодіяти з одним і більш компонентами або системами, що включає в себе тестування сумісності (compatibility testing) і інтеграційне тестування (integration testing).

- тестування безпеки (Security and Access Control Testing) – це перевірка безпеки системи, а також аналіз ризиків, пов'язаних із забезпеченням цілісного підходу до захисту додатка, атак хакерів, вірусів, несанкціонованого доступу до конфіденційних даних. Тестування безпеки може виконуватися як автоматизовано так і в ручну, включаючи перевірку як позитивних, так і негативних тестових випадків. Ґрунтується на трьох основних принципах - це конфіденційність, цілісність і доступність (confidentiality, integrity, availability);

При тестуванні програмного забезпечення необхідно завести штат співробітників та заповнити довідники програми, провести тестування функціоналу програми на етапі впровадження та виконати дослідну експлуатацію, під час якої відстежити й усунути всі відмови та недоліки ПЗ.

Розроблене програмне забезпечення моделювання функціональних поверхонь газорозподільних механізмів двигунів внутрішнього згоряння було протестоване спільно фахівцями центру інформатизації та відділу персоналу у

режимі тестової експлуатації. У таблиці 4.1 наведено набір тестів і результати тестування розробленого програмного продукту.

На першому етапі тестування було проведено спостереження за роботою програми з коректними вхідними даними, тобто за наявності файлу, в якому таблично задана вся необхідна інформація для розрахунків. Робота програми була перевірена для декількох різних файлів. Помилки у роботі системи виявлено не було.

Таблиця 1 – Результати тестування розробленого програмного модуля

Набір тестів	Очікуваний результат
Вибір файлу вхідних даних	Відкривається відповідна інформація
Упорядкування даних	Відкривається відповідна інформація
Настройка часу для деталі	Відкривається відповідна інформація
Вибір виду обробки	Відкривається відповідна інформація
Введення тарифних ставок та продуктивності	Відкривається відповідна інформація
Розрахунок норм часу	Відкривається відповідна інформація
Розрахунок заробітної плати	Відкривається відповідна інформація

Як видно з наведеної вище таблиці 1 тестування розробленого програмного модуля, в процесі роботи програми не було виявлено ніяких помилок. Робота програмного модуля була перевірена декілька разів.

У таблиці 2 наведено результати тестування кнопок переходу.

З даної таблиці також можна побачити, що результати тестування кнопок переходу не виявили ніяких помилок у системі.

Таблиця 2 – Результати тестування кнопок переходу

Тестування кнопок переходу	
Загрузить	Перехід на відповідну сторінку
Сохранить	Перехід на відповідну сторінку
Выход	Перехід на відповідну сторінку
Показать загруженные	Перехід на відповідну сторінку
Показать сотрудников	Перехід на відповідну сторінку
Настройки	Перехід на відповідну сторінку
Зарплата	Перехід на відповідну сторінку
О программе	Перехід на відповідну сторінку

Наступним етапом тестування було спостереження за роботою програми з некоректними вихідними даними. Програма успішно пройшла і цей вид тестування. При відсутності або пошкодженні будь-якого з необхідних системі файлів, користувачеві видавалося повідомлення про виниклу ситуацію, в разі критичної помилки програма так само повідомляла про це користувачеві і коректно завершувала роботу. У всіх випадках система успішно впоралася із завданням.

Висновки. Тестування показало, що розроблене програмне забезпечення, є ефективним і надійним, яке надає можливість розрахунку норм часу та заробітної

плати для підприємства. Розроблене програмне забезпечення має простий та зрозумілий інтерфейс, що дозволяє скоротити терміни на підготовку фахівців, які, використовуючи пропонований програмний продукт, будуть займатися проектуванням та моделюванням функціональних поверхонь газорозподільних механізмів двигунів внутрішнього згоряння.

Список використаних джерел:

1. Мацулевич О.Є., Щербина В.М., Бондаренко Л.Ю., Малюта С.І., Антонова Г.В. Програмне забезпечення для автоматизованого визначення параметрів різального інструменту фрезерної обробки корпусних деталей // Праці Таврійського державного агротехнологічного університету. – Мелітополь: ТДАТУ, 2020. – Вип. 20, т. 3. С 275-281.
2. Холодняк Ю.В., Гавриленко Е.А., Івженко А.В., Найдыш А.В. Моделирование участка пространственной монотонной кривой линии /Сучасні проблеми моделювання: наукове фахове видання. – Мелітополь: МДПУ ім. Б. Хмельницького, 2020. Вип.17. С. 131-137.
3. Мацулевич О.Є., Вершков О.О., Холодняк Ю.В., Дмитрієв Ю.О., Чаплінський А.П. Розробка мурашиного алгоритму для оптимізації оперативного планування робіт по збиранню врожаю кісточкових. *Плодовий сад – новітнє в теорії та практиці*: матеріали VВсеукр. наук.-практ. інтернет-конференції. Мелітополь: ТДАТУ, 2021. С. 106-110.
4. Вершков О.О., Мацулевич Ю.О. Визначення шорсткості поверхонь із застосуванням програмного забезпечення COPYCAD ф. DELCAM plc // Сучасні комп'ютерні та інформаційні системи і технології, Матеріали і всеукраїнської науково-практичної інтернет-конференції, Мелітополь 7-25 грудня 2020р. С. 17-23
5. Інженерна механіка: практикум / Г.В. Антонова, О.Є. Мацулевич, О.Ю. Михайленко, І.В. Пихтєєва, О.В. Івженко, Ю.В. Холодняк, В.М. Щербина, Ю.О. Дмитрієв. Мелітополь: Люкс, 2021. 147 с.

ПОРІВНЯННЯ ГНУЧКИХ МЕТОДОЛОГІЙ AGILE ТА WATERFALL

Поплавський В.С., студент

e-mail: uitpm221_pvs@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. ІТ сфера дуже активно розвивається на сьогоднішній день. Компанії постійно залучають нових клієнтів, створюють нові продукти, розвивають та підтримують проекти які вже створені... Саме через це часто виникає необхідність до покращення процесів пов'язаних з розробкою чи підтримкою програмного забезпечення. Існує багато різних методів та підходів, але найпопулярнішими серед них є - Agile і Waterfall. В кожній методології є свої плюси та мінуси.

Основні матеріали дослідження. **Agile** - гнучка методологія розробки програмного забезпечення, основним принципом якої є розробка на основі ітерацій(цикли, спринти). В кінці кожного циклу ми можемо мати на виході якийсь робочий функціонал(можливо трохи урізаний за функціоналом). На основі agile побудовано багато методів, найпопулярніші це Scrum та Kanban.

Перевагами Agile підходу є:

- Короткі та зрозумілі ітерації - розробка ділиться на цикли(тривалість від двох тижнів до двох місяців). Результатом такого циклу буде готова версія продукту;
- Високий ступінь залучення команди розробки та замовника. Так як розробка виконується циклічно, то є можливість планування кожного циклу окремо та змінювати пріоритети того що необхідно виконати в першу чергу;
- Мінімізація ризиків. Завжди можна перепланувати наступну ітерацію(внести поправки до того що необхідно зробити, змінити пріоритети...);
- Має велику популярність серед розробників;

Недоліками Agile є:

- Важкість підрахунку кінцевої вартості. Від того що є можливість гнучко вносити коригування до проекту буде важко підрахувати вартість виконаних робіт;
- Пoviшені вимоги до досвіду та кваліфікації команди. Команда повинна вміти аналізувати та покращувати ефективність своєї роботи, та постійно обмінюватися інформацією стосовно проекту;
- Ця методологія не є чіткою інструкцією. Вона є цілою філософською концепцією, потрібно прийняти ключові принципи системи;

Методологію слід використовувати коли:

- Над проектом працює висококваліфікована та досвідчена команда;
- Потрібно швидко отримувати робочі версії продукту;
- Продукт розвивається в сфері в якій можливі постійні зміни;
- Робота проходить над стартапом;

Методологію не слід використовувати якщо:

- Немає можливості витратити додаткові ресурси для постійної комунікації по проекту;
- Є конкретний термін коли продукт повинен бути виконаним;
- Є конкретний бюджет продукту;
- Коли потрібна детальна інформація по всім процесам розробки; [1].

Waterfall - методика, в основі якої лежить послідовний перехід від одного етапу до іншого без пропусків та повернень на попередні етапи. В даному підході важливим є те, що перед початком роботи необхідно мати чітке розуміння того що потрібно буде зробити, так як не буде можливості повернутись назад і щось переробити.

Перевагами Waterfall методології є:

- Зрозуміла та проста структура розробки. Дуже спрощується поріг входження для команд;
- Зручна звітність. Можна легко відслідковувати ризики, витрачений час, фінанси та виділені ресурси завдяки строгій поетапності роботи та добре задукоментованим вимогам до проекту;
- Не змінюються вимоги до проекту. Всі вимоги зрозумілі від початку проекту та не змінюються в процесі розробки;
- Вартість та час на розробку може бути вирахованою ще до початку розробки проекту;

Недоліками Waterfall є:

- Стійкість до змін. При розробці проекту у нас не має можливості вносити поправки;
- Інерційність. Немає можливості змінити проект в сторону оптимізації затрат, зміни концепції чи функціоналу до випуску готового продукту;
- У випадку недостатньої кількості ресурсів втрачається якість кінцевого проекту;

Методологію слід використовувати коли:

- Є чітка концепція продукту, який повинен бути на виході;
- Немає обмежень в часі та в ресурсах для створення продукту;
- Створення продукту створене на строгій послідовності задач;

Методологію не слід використовувати якщо:

- Йде створення інноваційного продукту, або великого проекту;
- Немає впевненості в кінцевій концепції продукту; [2].

Висновок. Кожна з моделей які ми переглянули має набір характеристик які підходять для проектів різної направленості. Agile та Waterfall допоможуть в створенні практично будь-якого продукту. Проте при виборі методології слід обирати ту методологію, яка допоможе максимально якісно та ефективно реалізувати саме ваш проект. Необхідно звертати увагу на всі вхідні параметри такі як час, бюджет, професійний рівень команди та інші важливі критерії майбутнього проекту. Потрібно пам'ятати, вибираючи методологію, що головною метою є створення якісного продукту, який може закрити потреби бізнесу.

Список використаних джерел:

1. Гнучка методологія розробки ПЗ – AGILE. URL: <https://training.qatestlab.com/blog/technical-articles/flexible-software-development-methodology-agile/>
2. Agile Vs Waterfall: яка найкраща методологія для вашого проекту. URL: <https://uk.myservername.com/agile-vs-waterfall-which-is-best-methodology>

УДК-004.8

ВИКОРИСТАННЯ ТЕХНОЛОГІЇ GOOGLE ANALYTICS В РОЗРОБЦІ WEB-ОРІЄНТОВАНОГО ДОДАТКУ «ПЛАТФОРМА СТВОРЕННЯ ОНЛАЙН ОГОЛОШЕНЬ»

Вакалюк Т.А., д. пед.н., професор

e-mail: kik_vta@ztu.edu.ua

Кияшенко А.С., магістрант

e-mail: ipzm221_kas@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Найбільш важливими показниками для комерційного ресурсу є число унікальних відвідувачів, кількість замовлень чи заявок, рівень доходу, поведінка користувача в додатку та інші показники ефективності. Ці дані неможливо зібрати самотійно, проте для цього існують спеціальні засоби збору статистики. Один з них – це Google Analytics.

Основні матеріали дослідження. Google Analytics – це фірмовий сервіс збору та аналізу відвідувачів сайту від Google. Його використання дає можливість отримати велику кількість детальних даних про трафік на сайті. Можливості Google Analytics дуже широкі. За допомогою цього сервісу можна:

- відслідковувати поведінку користувачів на сайті;
- аналізувати конверсію;
- виконувати А/В тестування;
- отримувати інформацію ефективності використання рекламних каналів;
- відслідковувати події на сайті в режимі реального часу;
- формувати статистику відвідувачів на основі демографічних даних;
- формувати звіти в залежності від того, як дані будуть оброблюватись.

Google Analytics один з провідних сервісів збору аналітики в світі, що має цілий ряд переваг:

1. Функціональна безкоштовна версія, якої буде достатньо навіть для великих проектів.
2. Якісна документація від Google.
3. Велика спільнота користувачів – сервіс присутній на ринку довгий час.
4. Інтеграція з іншими фірмовими рішеннями від Google: Google Adwords, Google Tag Manager та інші.
5. Регулярні оновлення та додавання нових функцій.
6. Стабільність.

Всі відвідування сайту відслідковуються Google Analytics. Це один з найважливіших показників ефективності роботи веб-додатку. На основі відвідувань формуються звіти, в яких присутні такі показники, як час відвідування, цінність користувача, когортний аналіз тощо. Важлива також не лише кількість користувачів, а і географічне розположення. Якщо веб-сайт створений для користувачів з України, то навряд чи вас будуть цікавити користувачі з інших країн. Проте такі дані дадуть вам інформацію про те, що в певній місцевості є зацікавленість у вашому сервісі.

Google Analytics дозволяє не тільки дізнатися об'єм трафіку, але і його джерело, будь то пошуковий сервіс, контекстна реклама чи партнерський ресурс. Це дозволяє оцінити ефективність просування сайту та у разі необхідності змінити підхід до просування.

Поведінка користувача, яку також відслідковує Google Analytics дозволяє оцінити, що саме на сайті більше всього цікавить людей. Існує багато підходів для

стимуляції користувачів зробити певну дію, наприклад купити продукт, проте ситуації бувають різні і ця інформація дасть можливість розробити унікальний підхід для вашого сайту. Цьому також допомагає вбудована функція створення А/В тестів.

Важливою функцією є розділ «Час завантаження сторінок». Зазвичай користувачі не люблять, коли сайт завантажується занадто довго. Це спонукає їх покинути такий сайт, що шкодить конверсії. Цей розділ показує наскільки швидко працює кожна сторінка та дає рекомендації для пришвидшення.

Висновок. Аналіз аудиторії – найважливіший процес для просування і покращення веб-додатку. Для комерційних сервісів це взагалі обов'язково. Google Analytics не єдиний сервіс, що дозволяє збирати подібну статистику. Проте це один з найпоширеніших сервісів для цього. А враховуючи популярність Google як пошукової системи, його дані будуть доволі точні. Тому навіть у разі використання іншого сервісу для аналітики, Google Analytics є рекомендований для встановлення.

Список використаних джерел:

1. <https://www.techtarget.com/searchbusinessanalytics/definition/Google-Analytics>

УДК 004.4

ОПТИМІЗАЦІЯ ПРОЦЕСУ УПРАВЛІННЯ ПОРТФЕЛЕМ ПРОЕКТІВ В ІТ-КОМПАНІЇ

Новохацький В.С., студент

e-mail: uitpm221_nvs@student.ztu.edu.ua

Вакалюк Т.А., д.пед.н., професор

e-mail: kik_vta@ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність управління портфелем ІТ-проектів важко переоцінити, адже його важливість та затребуваність суттєво підвищились за останній час та продовжують стрімко набирати оберти. В ІТ-індустрії, а саме у більшості її компаній, існує стійка тенденція стосовно збільшення загального об'єму роботи над проектами при порівняно незначних змінах у кількості людських ресурсів та коштів, які залучені для реалізації цих проектів. Саме ця проблематика спонукає ІТ-підприємства до формування портфелю проектів з подальшою організацією його ефективного управління.

Портфель проектів - це сукупність проектів, програм та інших робіт, об'єднаних разом з метою ефективного управління цими роботами для досягнення стратегічних цілей (визначення Project Management Body Of Knowledge (PMBoK) Guide 3rd Edition). Завданням **управління портфелями проектів** (*project portfolio management*) є вибір "правильних проектів" для виконання "правильної роботи", тоді як управління проектами сфокусоване на "правильному виконанні роботи". Результатами використання проектного підходу до управління компанією є забезпечення 100% прозорості процесів, що відбуваються, до 25% економії часових і матеріальних ресурсів і збільшення прогнозованості ризиків до 70% [1].

Управління портфелями проектів, загалом, є складною комплексною дисципліною, що передбачає відповідний рівень організаційної зрілості, а також налагодження автоматизованої підтримки, тому, в першу чергу, **PPM** вкрай необхідне для великих ІТ-організацій, в яких ведеться велика кількість проектів, а можливість їх консолідованого планування та виконання стає одним з найважливіших факторів продуктивності бізнесу. Переваги формування портфелю проектів базуються, перш за все, на безпосередньому зв'язку між проектами та на можливості розставити пріоритети в межах конкретного портфелю, а також на спроможності більш чіткого розподілу ресурсів. Належний контроль витрат, забезпечення високої якості продукту, виконання проектів у запланований термін – це неповний перелік переваг та вигод, які отримує компанія при належному управлінні портфелями проектів, основною метою якого являється настання матеріальних та нематеріальних наслідків, відповідальність за які несуть менеджери портфелів, тому вкрай важливо та необхідно збалансувати вигоди, ризики та ресурси шляхом оптимізації.[2]

Абсолютно природнім є той факт, що різні компанії мають різні проекти, тому проводячи оптимізацію управління, варто керуватися наступними універсальними правилами:

- документування процесу управління портфелем проектів;
- портфель проектів має складатися з упорядкованого набору компонентів: проектів, програм проектів та інших робіт, портфелів бізнес-напрямів - тобто має бути зрозуміла та помітна чітка ієрархія всіх процесів, які відбуваються в компанії;

- за портфелем і за його компонентами має бути закріплено відповідальних осіб, які мають повне та чітке розуміння процесу, визначено основних зацікавлених сторін, їх обов'язки та зони відповідальності;
- управління портфелем проектів має передбачати прийняття важливих рішень - визначення пріоритетів та розміщення ресурсів і напрямку інвестицій;
- створення можливостей для зручної та ефективної комунікації всередині компанії з питань управління портфелем проектів;
- звітування перед керівництвом щодо статусу проектів в режимі реального часу.[3]

Дотримання цих, на перший погляд, типових, простих правил створить стабільне підґрунтя для організаційної структури компанії в сфері управління портфелями проектів.

Список використаних джерел:

1. Анна Старинська, журнал "Корпоративні системи" (№5, 2007) [Електронний ресурс] Режим доступу: <http://www.management.com.ua/ims/ims141.html>
2. Стандарт з управління портфелем проектів. Четверте видання. Project Management Institute, Inc., 2017, ст. 39-40
3. Герасимов А., Жданов В. Бізнес-проект з ІТ-складової [Електронний ресурс] // Директор інформаційної служби. - 2005. - № 03, URL: <http://www.osp.ru/cio/2005/03/173879/>

УДК 004.75

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ПЛАТФОРМ УПРАВЛІННЯ ОБЧИСЛЮВАЛЬНИМИ СЕРВІСАМИ ПРИ ОРГАНІЗАЦІЇ FOG COMPUTING

Островська К.Ю., к.т.н.

e-mail: kuostrovskaya@gmail.com

Шерстяних М.О., магістр

e-mail: n.sherstianykh@gmail.com

Стовпченко І.В., ст.викладач

e-mail: istovpchenko@gmail.com

Український державний університет науки і технологій

Актуальність та постановка проблеми. На сьогоднішній день, коли технології інтернету речей та надання хмарних обчислень як послуг впевнено знаходять своє застосування, розвиваються нові обчислювальні концепції, зокрема технології туманних обчислень. Даний вид обчислень досі є новою технологією, яка тільки починає набирати свою популярність.

Парадигма хмарних обчислень сприяла розвитку надання обчислювальних сервісів та інфраструктур як послуг, що дозволило заощадити кошти на створення та обслуговування власної обчислювальної інфраструктури. У цьому середовищі спрощено та автоматизовано масштабування додатків для задоволення потреб в умовах високого навантаження. Віртуалізація є ключовою технологією, що забезпечує ці можливості. В даний час контейнеризація стала популярною альтернативою віртуальним машинам і набула широкого застосування, у результаті інструменти оркестрації стали невід'ємною частиною хмарних обчислень. Незважаючи на успішне застосування технології контейнеризації, досі хмарні обчислення не забезпечують належного відповідності критеріям технологій Інтернету речей [1]. Управління службами, розгорнутими в туманному обчислювальному середовищі, є складним завданням, а інструменти контейнеризації та оркестрації реалізують її безпроблемне впровадження та використання [2].

Таким чином, дослідження ефективності платформ контейнерної оркестрації при організації туманних обчислень є актуальним завданням розвитку концепції туманних обчислень.

В рамках роботи організується розгортання Docker-контейнерів [3]. Для створення кластера використовується Docker Swarm. Вирішуються завдання вимірювання наступних параметрів: час розгортання одного контейнера, час розгортання групи контейнерів, час відгуку задачі горизонтального масштабування, час затримки передачі. Проводиться аналіз одержаних результатів випробувань.

Основні матеріали дослідження. Для достовірного дослідження для різних інструментів контейнерної оркестрації необхідно створити ідентичні умови виконання тестувань і провести ряд повторень кожного випробування для досягнення достовірних результатів. Для цього скористаємося віртуальними машинами, що надаються Amazon Web Services.

Для достовірності результатів проведення випробування необхідно розробити утиліту автоматичного проведення випробувань, яка має забезпечувати наступний функціонал:

- можливість налаштування, вибір та запуск випробування;
- моніторинг поточного випробування;
- демонстрація результатів випробування.

Для опису варіантів використання утиліти розроблено діаграму варіантів використання (див. рис. 1) – діаграма, що відображає відносини між акторами та прецедентами.

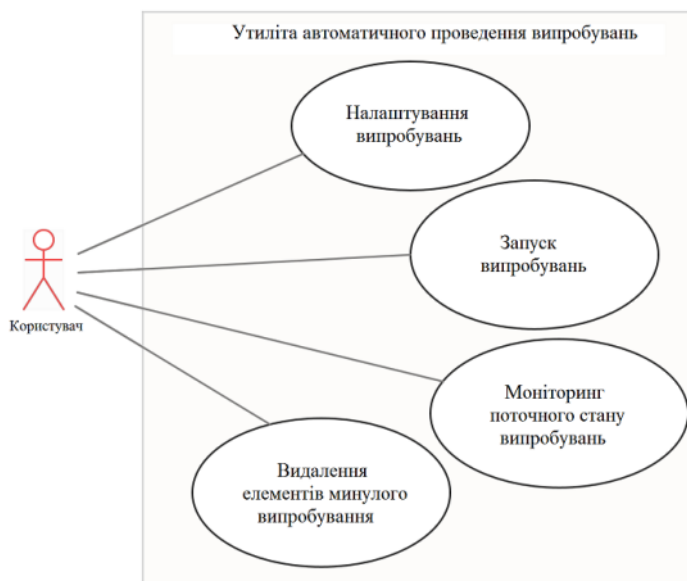


Рисунок 1. Діаграма прецедентів утиліти автоматичного проведення випробувань

Для того, щоб виміряти час проведення випробувань, було обрано принцип зворотного відгуку кожного розгорнутого контейнера [4], який полягає в наступному:

- запускається слухач (сервер), який приймає повідомлення від контейнерів (клієнтів);
- слухач фіксує час початку випробування та приймає всі повідомлення, надіслані контейнерами під час розгортання, та показує час відгуку.

Щоб отримати час випробування, необхідно від часу останнього відгуку відняти час початку випробування.

Наочно принцип зворотного відгуку кожного контейнера можна побачити на рисунку 2.

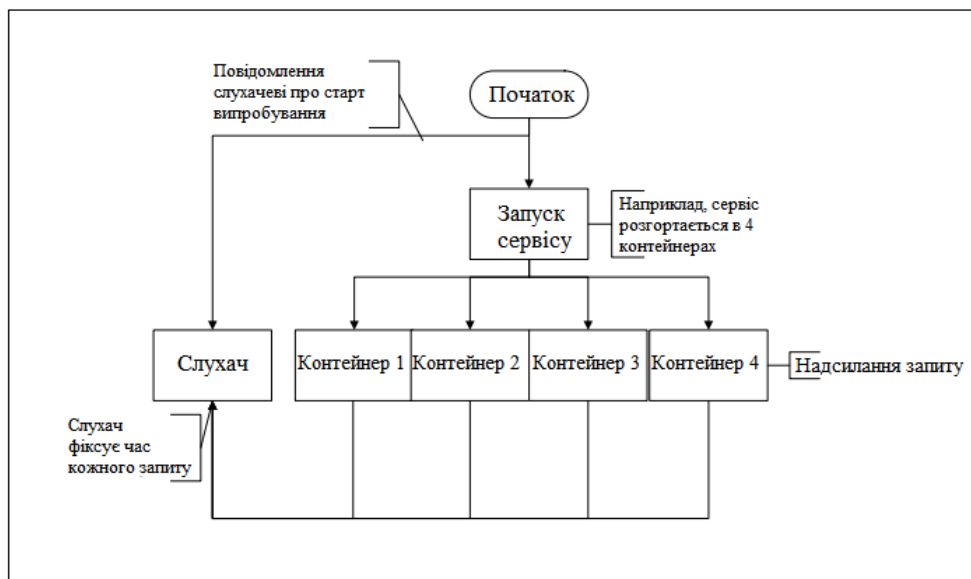


Рисунок 2. Принцип зворотного відгуку кожного контейнера

Висновок. У ході роботи було розроблено утиліту автоматичного проведення випробувань. Також було проведено випробування «Один контейнер», «Група контейнерів», «Горизонтальне масштабування», «Latency».

Для виконання поставленої мети було вирішено такі завдання:

- здійснено підбір літератури, наукових публікацій та інтернет статей, необхідних для проведення дослідження;
- виконано огляд платформ контейнерної оркестрації;
- визначено ключові вимоги та критерії проведення дослідження;
- спроектовано та реалізовано утиліту автоматичного проведення випробувань;
- виконано дослідження ефективності платформ контейнерної оркестрації з організацією туманних обчислень

Список використаних джерел:

1. Что такое интернет вещей и как он работает? [Електроний ресурс]
URL: <https://server-shop.ua/the-internet-of-things-and-the-scope-of-its-use.html>
(дата звернення 06.12.2022р.)
2. IoT, туман и облака: поговорим про технологии? [Електроний ресурс]
URL: <https://habr.com/ru/company/cloud4y/blog/467711/> (дата звернення 06.12.2022р.)
3. *Моуэт Э.* Использование Docker / Э. Моуэт; пер. с англ. А.В. Снастина; науч. ред. А. А. Маркелов. – М.: ДМК Пресс, 2017. – 354 с.
4. Evaluating Container Platforms at Scale. [Електроний ресурс] URL: <https://medium.com/on-docker/evaluating-container-platforms-at-scale-5e7b44d93f2c>
(дата звернення 06.12.2022р.)
5. *Колісник Д.Р., Місевич К.С., Коваленко С.В.* Системна архітектура IoT-Fog-Cloud для систем аналізу великих даних і кібербезпеки: огляд туманних обчислень, впровадження аудиту інтернету речей. Сучасний захист інформації. 2020. № 3. С. 34–38.

УПРАВЛІННЯ ВАРТІСТЮ ПРОЄКТА ЗА ТЕХНІКОЮ EVM

Поплавський В.С., студент

e-mail: uitpm221_pvs@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Кожен керівник проєкту хоче чітко знати відповіді на питання чи відстає його проєкт від плану виконання, наскільки ефективно використовують ресурси, йде перевищення бюджету, або ж навпаки його економія та яка вартість робіт які залишилось виконати? Для того щоб проєкт був успішним потрібно уважно керувати кожним його компонентом та процесом. Керівнику проєкту потрібен постійний зворотній зв'язок, щоб вчасно виявляти та вирішувати можливі проблеми, для своєчасного завершення проєкту та вкладання у запланований бюджет.

Основні матеріали дослідження. Техніка EVM (Earned Value Management - управління освоєним об'ємом) допомагає виконати поставлені задачі. Ця техніка містить в собі аналіз всього об'єму робіт з планом виконання та вартістю. Завдяки їй керівник має можливість бачити стан проєкту, оцінювати його реальний стан та вносити корективи. [1].

Для інтеграції EVM в проєкт, керівнику проєкту потрібно виконати декілька кроків:

- **Планування.** Потрібно вирішитися з об'ємом робіт (що повинно бути зроблене та поставлене замовнику). Кожна робота повинна бути розбита на задачі, це дасть змогу краще контролювати виконання плану. Вирішуємо питання з виконавцями та визначаємо їх зону відповідальності (готуємо організаційну структуру команди);
- **Складання бюджету.** Потрібно всі заплановані роботи прив'язати до часових інтервалів та накласти на роботи бюджет який буде йти на виконання кожної роботи. Це дасть можливість керівнику бачити скільки коштів повинно бути використано та в який момент;
- **Вимірювання прогресу.** На цьому етапі потрібно вирахувати ключові показники. На основі цих показників буде можливість виміряти відхилення та спрогнозувати рух проєкту;
- **Аналізуємо відхилення.** EVM дає можливість виявити відхилення на основі аналізу своїх основних елементів - виконаного об'єму, плану робіт та фактичних витрат;
- **Прогнозування.** На основі отриманих даних керівник має можливість бачити скільки робіт залишилось виконати, яка кількість бюджету залишилась та зіставити чи співпадають прогнози з тією картиною яку маємо на даний момент;

Аналізуючи відхилення за обраний період можна виявити наступні показники:

- Відхилення по вартості. Можна побачити що в нас відбувається з бюджетом, йде перевищення, або йде економія;
- Відхилення у часі. Цей параметр дає змогу бачити чи встигаємо ми виконати запланований об'єм робіт у визначені строки;
- Індекс вартості виконання. Показує наскільки ефективно команда проєкту використовує виділені ресурси;
- Індекс строків виконання. Дає можливість побачити наскільки ефективно команда витрачає свій час; [2].

У випадку коли один з двох перших показників (або відразу обидва) мають від'ємне значення, то на це потрібно відразу реагувати. Це свідчить про те що на проєкті є відхилення не в кращу сторону.

У випадку коли два останні показники рівні або дорівнюють одиниці, то проєкт знаходиться в гарній формі. А якщо навпаки (індекс матиме значення менше одиниці), то це свідчить про те що є проблема в строках, або у вартості, чи і в строках і в вартості.

Отримавши дані та проаналізувавши їх можна робити певний прогноз. EVM дозволяє отримати наступні прогнозовані дані:

- **Прогноз по завершенню проєкта.** Дає змогу прогнозувати коли закінчиться проєкт та скільки коштів на нього буде витрачено;

- **Відхилення по завершенню.** Можна буде визначити чи ми економимо кошти які були сплановані, чи у нас буде перевикористання запланованого бюджету;

- **Індекс продуктивності до завершення.** Цей індекс показує ефективність з якою потрібно використовувати ресурси для того щоб в кінці вартість відповідала тим значенням які були сплановані;

- **Прогноз до завершення.** Показує очікувані витрати на виконання робіт, які залишилися по проєкту. [2].

По результатам отриманих показників та прогнозування рекомендується фіксувати результати прогнозування та результати вичислювальних процедур в графіках або зведених таблицях. Це дасть можливість бачити рух проєкта та його історію. А також потрібно не забувати вносити корективи для успішного виконання проєкту.

Висновок. Дуже важливим для керівників проєктів є можливість бачити поточний стан проєкту, які роботи виконуються, чи встигаємо по часу та чи вкладаємось в запланований бюджет, бачити та керувати можливими ризиками. Для таких цілей існує багато інструментів. Одним з них є техніка управління освоєним об'ємом (EVM). EVM - це техніка яка дає змогу керівнику бачити ситуацію яка є на проєкті в конкретний момент в числовому представленні, дає можливість виявити ризики пов'язані з витратами та часом затраченим на розробку проєкту. Проведений аналіз факторів, які впливають на проєкт надасть можливість керівнику побачити що йде не за планом та застосувати корективи які можуть виправити ситуацію на ранніх етапах роботи.

Список використаних джерел:

1. Що таке система управління заробленою вартістю (EVM)? URL: <https://uk.myservername.com/pecerna-divcina-ci-majbutnij-hlopcik-aki-vidminnosti-miz-pokemon-scarlet-i-violet>

2. ДСТУ ISO 21508:2022 Управління здобутою цінністю в управлінні проєктами та програмами. URL: <https://pmdoc.ua/iso/iso21508/>

УДК 004.891

ДОСЛІДЖЕННЯ СУЧАСНИХ СИСТЕМ МОНІТОРИНГУ СТАНУ СЕРВЕРНОГО ОБЛАДНАННЯ

Гольцов В.В.

e-mail: vovladik141@gmail.com

Науковий керівник к.т.н., доц. Холодняк Ю.В.

Таврійський державний агротехнологічний університет імені Дмитра
Моторного

Постановка проблеми. Компаніям часто необхідно відстежувати які пристрої та користувачі автентифікуються та мають доступ до корпоративної мережі. Головна мета цього – контроль трафіку та виявлення різних кібератак на сервісне обладнання, що можуть завдати значних збитків і привести до крадіжки даних користувачів, і також завдати репутаційних збитків. Зараз багато різних компаній, що працюють в Україні, мають великий шанс, що їх сервісне обладнання зазнає один або декілька з видів кібератак. Щоб уникнути цього, першим шагом пропонується простежувати всю інформацію, яку отримують та надсилають сервера, а другим – блокувати дані або попереджувати працівників, що працюють з сервісним обладнанням. Програма моніторингу повинна містити простий, але масштабований інтерфейс, що дозволить швидко опанувати та налаштувати її під власні потреби.

Зараз у зв'язку з надзвичайною ситуацією в країні, розробляється багато програмного забезпечення, що потребує зв'язок з сервером. З іншого боку стали частіше проходити різні кібератаки, які направлені на перевантаження баз даних, або виводу мережевих пристроїв з ладу. На цей час майже не існує статей, що детально досліджують тему мережевих кібератак. Також розробляється багато вузькоспеціалізованих додатків, що призначенні для захисту від зовнішньої загрози, але опис цих програм майже не публікуються за для застереження їх від програмного злому.

Метою роботи є аналіз існуючих систем моніторингу стану серверного обладнання для розподілених та централізованих мереж з метою подальшої розробки власної. Для досягнення поставленої мети необхідно вирішити наступні завдання:

- проаналізувати найпоширеніші кібератаки на серверне обладнання в Україні;
- дослідити оптимальний шлях моніторингу трафіку, та оптимізації роботи з ним;
- визначити які дані треба відокремлювати від поточної інформації з подальшої ізоляцією.

Основні матеріали дослідження. Працюючи з комплексом програмно-технічних засобів, а саме з сервером та його архітектурою, програмуючи СУБД і, змінюючи інформацію у БД, то відповідно для цього повинен надаватись захист як фізичний, для охорони сервісного обладнання та його компонентів, так і інформаційний [1].

Стосовно проектування архітектури сучасних центрів обробки даних (ЦОД) необхідно здійснювати виходячи з розрахунку передбачення можливих відмов устаткування.

Інформаційні системи допускають розпаралелювання процесів збору, обробки, зберігання і надання даних користувачам, а також передбачають

механізми захисту від апаратних збоїв. Кластерні рішення і балансування навантаження ще більше підвищують надійність і доступність сервісів ЦОД. Механізми самокорекції, а також розвинені механізми моніторингу стану дозволяють зменшити вплив відмови окремих елементів на систему в цілому.

Сучасний центр обробки даних є комплексною інженерною спорудою, що забезпечує функціонування бізнес-процесів. Його основу складають три функціональні блоки:

— *телекомунікаційна система*, що реалізує взаємозв'язок елементів ЦОД, а також прийом/передачу даних між центром і користувачами його сервісів;

— *технічна архітектура* (сервери доступу, сервери додатків, сервери СУБД, сховища даних), що підтримує функціонування інформаційних систем, доступ користувачів до додатків і зберігання даних;

— *інженерна інфраструктура*, що забезпечує оптимальні умови для функціонування вищенаведених систем і діяльності обслуговуючого персоналу [2].

Кожен з цих блоків потребує захист, але саме технічна архітектура потребує більшого оснащення та перевірки всіх даних, що поступають. Зважаючи на це, існують програмні додатки, що допомагають відстежувати поточну інформацію. Можна навести декілька актуальних прикладів існуючих програм моніторингу стану серверів (табл. 1).

Таблиця 1. Порівняння програм моніторингу стану серверів

Програма Критерій оцінки	Server and Application Monitor	Site24x7	Sematext Monitoring
Наявність документації користувача	Так	Ні	Так
Можливість розширення додатку за допомогою зовнішніх модулів	Так	Ні	Так
Робота зі хмарними технологіями	Так	Так	Ні
Кількість елементів керування (за досвідом користувачів)	Багато	Середнє	Середнє
Автоматичне відстежування протоколів та роботи серверу	Так	Так	Так
Створення звітів відстежуваної інформації	Так	Так	Ні

1) Server and Application Monitor. Це потужний додаток що відстежує продуктивність сервера та програми, сповіщаючи про будь-які проблеми та повідомляючи про них [3].

Розробник SolarWinds інтегрував можливість моніторингу віртуальних серверів як локальних, так і хмарних для понад 1200 різних програм. Він забезпечує розуміння продуктивності програми та показників використання сервера (рис. 1).



Рисунок 1. Візуальний інтерфейс «Server and Application Monitor»

Він має такі функції, як рішення для віддаленого моніторингу серверів, моніторинг інвентаризації серверів, моніторинг процесів сервера, автоматизований моніторинг мережі серверів і моніторинг справності серверів.

Мінусом програми можна зазначити великий масив функцій та модулів, що погано сприймаються нетехнічними користувачами.

2) Site24x7 – це хмарний інструмент моніторингу мережі, який цілодобово відстежує стан сервера та кінцевих веб-точок.

Додаток контролює мережеву інфраструктуру компанії. Це робиться шляхом контролю всього апаратного забезпечення та віртуальних ресурсів, як локальних, так і хмарних (рис. 2).

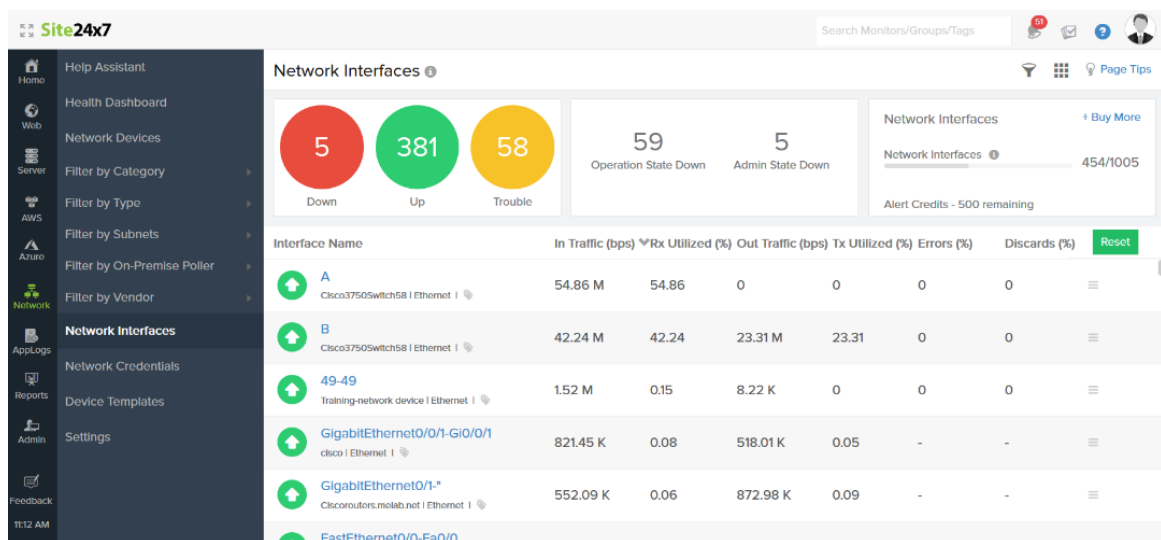


Рисунок 2. Візуальний інтерфейс «Site24x7»

Site24x7 спостерігає за справністю праці веб-сайту та сповіщає, коли виявляє конкретну аномалію. Рішення для моніторингу спостерігає за діяльністю таких служб, як протоколи передачі файлів (FTP - File Transfer Protocols), порогові значення подій, прості протоколи передачі пошти (SMTP - Simple Mail Transfer Protocols), керування продуктивністю додатків (APM - Application Performance Management) і моніторинг записів систем доменних імен (DNS - Domain Name Systems).

Мінуси: налаштування реєстратора транзакцій займає багато часу та розширений інтерфейс може бути складним для навігації у структурі сайту.

3) Sematext Monitoring [4]. Це єдиний інструмент моніторингу для серверів, програм і стеків у багатохмарних середовищах. Програма відстежує серверні процеси, а також усі системні пакети, їх версії, встановлення, видалення і тощо.

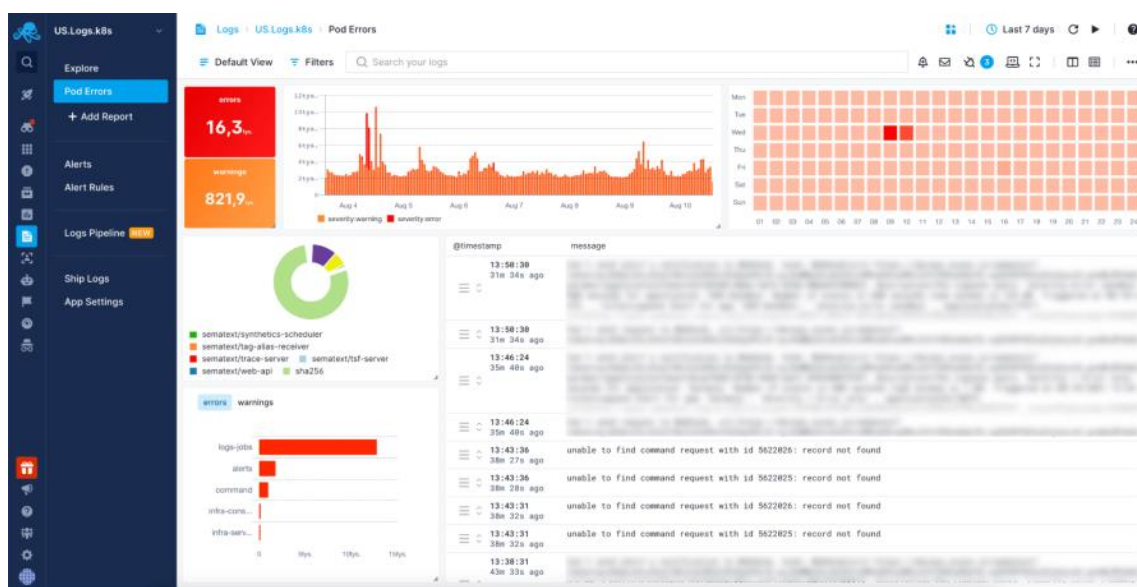


Рисунок 3 - Візуальний інтерфейс «Sematext Monitoring»

Також, рішення для моніторингу надає все необхідне в одному місці для кращого та швидшого усунення несправностей із виявленням аномалій, попередженням і кореляцією між усіма компонентами, кластерами та контейнерами. Усі показники, що стосуються вашого серверу, надаються для перегляду, включаючи ЦП, пам'ять, використання диска, введення-виведення, мережу, навантаження тощо.

З іншого боку програма не нова, хоча все ще оновлюється і відсутність документації для старих агентів робить вивчення інтерфейсу складнішим, і не ідеальність агентів для певної інтеграції, які повинні надсилати власні показники. Ще можна визначити обмежену підтримку відстеження транзакцій.

Усі вищезазначені приклади стосуються як централізованих систем, так і розподілених. Кожна програма передбачає подібне рішення, що можна налаштувати через інтерфейс додатка. Для нормального стану роботи сервера пропонується використовувати ряд основних показників, що потрібні для моніторингу ресурсів:

- **Моніторинг здоров'я** – це загальна інфраструктура, яка вимагає моніторингу стану апаратного забезпечення фізичного сервера, стану гіпервізора,

стан віртуальної машини, фізичної та віртуальної мережі, комутатори та маршрутизатори, а також системи зберігання.

• **Моніторинг продуктивності** – базова продуктивність, перевіряє стан ЦП, внутрішньої пам'яті і показники продуктивності мережі від гостьової віртуальної машини ОС, а також з гіпервізора. Ці показники зазвичай контролюються навіть у невіртуалізованих середовищах. Метрики віртуалізації можуть бути для конкретних сутностей, які вводяться через різні технології віртуалізації, наприклад, кластері концепції центрів обробки даних у **VMware**. Поведінки інших функцій віртуалізації також можна виміряти такими показниками, як частота міграцій віртуальної машини, або у задіяних функціях масштабованості. Є спеціалізовані програми, створені за допомогою віртуалізації як віртуалізація робочого столу (VDI). Моніторинг для такого рішення потребує збору додаткових параметрів з віртуальної машини, а також рівня гіпервізора, наприклад час обслуговування для віртуальної машини, наданої у відповідь на запит на підключення до робочого столу кінцевим користувачем.

• **Моніторинг потенціалу**. Сучасні організації справді динамічні, а використання ресурсів/вимоги до них постійно розвиваються. Отже, потрібно постійне планування різноманітних ресурсів, таких як сервери, робочі столи, мережа та сховище. Ця потреба вимагає періодичних перевірок як фізичних, так і віртуальних ресурсів. Моніторинг ємності вимагає наскрізного безперервного моніторингу ємності за такими ключовими показниками:

1) **Використання пам'яті**: використання пам'яті на кожному сервері, вузькі місця ємності та зв'язок із кількістю користувачів.

2) **Використання сервера**: пікове/середнє використання ресурсів сервера – пам'ять/ЦП/ресурс, вузькі місця сервера та кореляція з показниками робочого навантаження, такими як кількість користувачів.

3) **Використання мережі**: пікове/середнє використання мережі, пропускна здатність/вузькі місця та зв'язок із кількістю користувачів/віртуальної машини.

4) **Використання мережі**: пік/середнє використання мережі, пропускна здатність/вузькі місця та зв'язок із кількістю користувачів.

• **Безпека та моніторинг відповідності** - представляє новий набір ризиків для безпеки через розповсюдження віртуальних машин та автономних віртуальних машин. Жива міграція віртуальних машин потенційно може конфліктувати зі способом керування доступом і застосуванням ряду правил. ІТ-безпека та моніторинг відповідності стають критично важливими для захисту віртуалізованого середовища. Моніторинг безпеки та його відповідності потребують наскрізної перевірки активності через віртуалізацію, а саме:

1) **Розповсюдження віртуальної машини**: метрики для моніторингу активності віртуальної машини під час її клонування, копіювання чи міграції в межах мережі або навіть у інше місце зберігання.

2) **Показники конфігурації**: моніторинг конфігурації віртуального сервера, щоб переконатися, що вони відповідають стандартам і вказівкам щодо посилення, моніторинг конфігурації віртуальної машини для застосування політики ліцензування програмного забезпечення.

3) **Події віртуалізації**, які допомагають застосовувати/виявляти порушення ІТ-політики. Це включає моніторинг індивідуальної безпеки, політики безпеки організації.

4) **Контроль доступу**: моніторинг контролю доступу та звіти для примусового контролю доступу на основі ролей.

5) **Моніторинг відповідності:** Метрики для перевірки/аудиту ІТ-налаштувань і процесів для стандартів і такі нормативні акти, як HIPAA, SOX, GLBA.

Безпосередньо, архітектура програми повинна бути відкритою, щоб її було легко розширити, підключивши нові модулі на будь-якому рівні. На найнижчому рівні повинен знаходитися моніторинг збору даних, який взаємодіє з різними джерелами даних загального моніторингу. Він призначений для збору даних з усіх доступних джерел. Віртуальна інфраструктура являє собою апаратне забезпечення (сервер, мережа та сховище) та програмні компоненти (гіпервізори та програмне забезпечення для керування). Інші компоненти, з яких збирач даних моніторингу отримує дані, — це база даних конфігурації та наявні інструменти для моніторингу продуктивності програми. Верхній рівень моніторингу займається аналітикою Рівень аналітики обробляє зібрані дані та створює різноманітні аналітичні результатів. Основна перевага можливості збирати дані з різних джерел полягає в тому, щоб гарантувати, що отримані аналітичні результати є точними та дієвими. Тут наголошується на тому, що аналіз не повинен давати рекомендації щодо розміщення, які не відповідають певній безпеці чи мережевій політиці.

Платформа повинна надавати два зовнішні інтерфейси. Один — інтерфейс користувача, який відображає звіти на основі аналітичних модулів, інтегрованих із структурою. Інші можливості інтерфейсу користувача дозволяють адміністраторам налаштовувати параметри, що керують аналітичною обробкою. Наприклад, адміністратор може встановити порогові значення для використання в обчисленнях. Сповіщення та попередження можуть відображатися в інтерфейсі користувача або доставлятися до налаштованих кінцевих точок. Інший інтерфейс — це інтерфейс API, який дозволить іншим стороннім рішенням створювати можливості цієї структури. Інтерфейс API забезпечить доступ до необроблених даних, а також до результатів аналітичних обчислень.

Усі рівні інфраструктури повинні дотримуватися архітектури, що підключається з інтерфейсами, визначеними таким чином, що модулі можуть створюватися незалежно для розширення його можливостей, наприклад, незалежний модуль може бути написаний для додавання можливості збору даних для підтримки нового гіпервізора або надання додаткових аналітичних можливостей.

Висновки. Сьогодні існують різні програми для збору та аналізу даних, але кожна з них має певні складності, з якими може зіткнутися інженер або системний адміністратор, що може зменшити якість моніторингу даних, та піддати ризику роботу серверу. До розробки власної програми моніторингу у віртуалізованому середовищі треба мати широке рішення для підходу моніторингу та аналізу поточних даних. Для впровадження цього треба запропонувати відкриту систему моніторингу, яка призначена для подолання деяких із цих проблем. Є потреба у розробці консолідованої основи для моніторингу з додаванням модулів для розширення загальних можливостей. Цікавим є рішення повної віртуалізації процесів моніторингу з його подальшим використанням на практиці. Це потребує особливого рішення для моніторингу, що буде здатний автоматизувати виявлення та вирішення проблем з сервісним обладнанням. Проблема сьогодення полягає в тому, що рішення для моніторингу спеціалізуються лише на кількох аспектах загального моніторингу, і, отже, аналіз не можна підтвердити для аспектів, які не охоплюються системою, що вимагає ручного втручання для затвердження рекомендацій щодо конфігурації.

Список використаних джерел:

1. Некряч Д. О. Система захисту сервера автентифікації RADIUS від DoS атак: магістерська дис. техн. наук: 121 Програмна інженерія. Київ, 2020. 107 с.
2. Маноха Л. Ю., Гаркуша І. Д. Дослідження і обґрунтування доцільності створення центрів обробки даних в організаціях з розгалуженою інфраструктурою. *Наукові праці НУХТ*. К.: НУХТ, 2008. № 24. С. 5-6.
3. Nilesh Jayanandana. 10 Best Server Performance Monitoring Tools & Software in 2022: презентація. URL: <https://sematext.com/blog/server-monitoring-tools/> (дата публікації: 06.06.2022).
4. Lawrence Williams. 10 BEST Server Monitoring Tools & Software (Nov 2022): презентація. URL: <https://sematext.com/blog/server-monitoring-tools/> (дата публікації: 07.11.2022).
5. Chandran M., Walvekar J. Monitoring in a Virtualized Environment. *GSTF Journal on Computing*. 2014. Т. 1. №. 1.
6. Лупина І. Б. Пристрій багатокритеріального моніторингу стану металообробного обладнання: дис. техн. наук: 151 Автоматизація та комп'ютерно-інтегровані технології, 2021. 147 с.
7. Суховерша В. О. Розробка веб-сервісу для моніторингу роботи серверного обладнання: дис. комп'ют. наук: 122 Комп'ютерні науки, 2021. 169 с.
8. Kadir E. A. et al. Wireless monitoring for big data center server room and equipments. *2015 International Conference on Science in Information Technology (ICSITech)*. IEEE, 2015. С. 187-191.

УДК-004.8

ПЕРСПЕКТИВИ ВИКОРИСТАННЯ NOSQL СИСТЕМ КЕРУВАННЯ БАЗАМИ ДАНИХ

Вакалюк Т.А., д. пед.н, професор

e-mail: kik_vta@ztu.edu.ua

Кияшенко А.С., магістрант

e-mail: ipzm221_kas@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Реляційні бази даних – це перевірений спосіб зберігання даних, що відповідає вимогам ACID (атомарність, узгодженість, ізолюваність, довговічність). Проте такий тип баз даних має лише один спосіб масштабування – вертикальний. Це означає, що масштабувати базу даних можна лише збільшивши обчислювальні можливості апаратного забезпечення. Це дорогий та доволі обмежений процес. Тож поява таких високомасштабованих веб-додатків як пошукові системи і необхідність створення паралельних розподілених систем для них призвела до появи нереляційних систем керування базами даних нового покоління.

Основні матеріали дослідження. NoSQL не є назвою якоїсь певної системи керування. Насправді це неофіційна назва широкого класу нереляційних систем керування базами даних, що суттєво відрізняються від традиційних реляційних СКДБ, які використовують мову SQL. Назва так і розшифровується – Not only SQL (Не тільки SQL). В базах даних цього типу вирішується проблема масштабованості та доступності за рахунок повної чи часткової відмови від атомарності та узгодженості. Це робить їх чудовим вибором для сучасних мобільних додатків та ігор, коли є необхідність швидкого доступу до гнучкої та масштабованої бази даних. Проте це робить неможливим їх використання в транзакційних додатках, таких як банківські системи, біржі і тд.

NoSQL бази даних діляться на декілька типів:

- Ключ-значення. В сховищах такого типу пари ключів та значень формуються за допомогою хеш-таблиці. Такі бази даних мають найбільший потенціал горизонтального масштабування. Приклади: Berkeley DB, Redis, Amazon DynamoDB;
- Документноорієнтовані. Служать для зберігання ієрархічних структур даних. Документом – це JSON подібний об'єкт, аналог якого в SQL СКБД називається ряд. Документи зберігаються в колекціях. Підходять для систем керування вмістом (CMS). Приклади: CouchDB, MongoDB.
- Графові. База даних такого типу має вигляд графу. Застосовуються в сферах, в яких дані потребують великої кількості зв'язків, оскільки, на відміну від з'єднань (join) в SQL, обхід графу не потребує додаткових обчислень. Приклади: Neo4j, OrientDB.
- Стовбчаті. В таких системах дані зберігаються у вигляді розрідженої матриці, рядки і стовбці якої використовуються як ключі. Застосовуються для задач, пов'язаних з обробкою великих даних, таких як веб-індексування для пошукових систем. Приклади: Google BigTable, Apache HBase, Apache Cassandra.

Висновок. Бази даних NoSQL мають широку сферу застосування. За рахунок підтримки гігантів індустрії як Google, Facebook, IBM та інших технології для обробки великих даних широко розповсюдились, оскільки бази

даних реляційного типу не дають потрібної продуктивності через надмірну нормалізацію даних. Також NoSQL чудово підходить і для невеликих додатків, в яких важливою є швидкість відгуку. Проте повністю замінити реляційну систему вони не здатні, оскільки існують і будуть існувати системи, в яких необхідно зберігати цілісність посилальних даних, а також використовувати складні запити.

Список використаних джерел:

1. <https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-nosql-database/>

УДК 004.4

КІБЕРЗЛОЧИННІСТЬ ЯК ЗАГРОЗА СУЧАСНОМУ ІНФОРМАЦІЙНОМУ СУСПІЛЬСТВУ

Хімічук І.С.

e-mail: kbm221_khis@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Використання інформаційних технологій розширює свою дію на нові сфери людської діяльності. Інформація, як один з основних елементів даного процесу, відіграє все більш істотну роль, як у житті окремої людини, так і в житті всього суспільства та кожної держави. Фактично розвиток науково-технічного прогресу, пов'язаний із впровадженням сучасних інформаційних технологій, призвів до появи нових видів злочинів, зокрема, до незаконного втручання в роботу персональних комп'ютерів, систем та комп'ютерних мереж, розкрадання, присвоєння, вимагання комп'ютерної інформації, небезпечного антисоціального явища, що отримало поширену назву - кіберзлочинність.

Ситуація з кіберзлочинністю з кожним роком лише погіршується. Маштаби кіберзлочинності досягли таких розмірів, що дають змогу називати їх загрозою національній безпеці. Зловмисники діють на випередження і завжди на кілька кроків попереду тих, хто їм протидіє. Кожен четвертий злочин скоюється з використанням ІТ-технологій і за даними Національного відділення ФБР США з комп'ютерних злочинів, від 85 до 97% кіберзлочинів залишаються невиявленими.

Основні матеріали дослідження. Кіберзлочинність - це злочинна діяльність, в рамках якої використовуються або атакуються комп'ютер, комп'ютерна мережа або мережний пристрій. Більшість кібератак здійснюється кіберзлочинцями чи хакерами з метою отримання фінансового прибутку. Однак метою кібератак може бути і виведення комп'ютерів чи мереж із ладу – з особистих чи політичних мотивів.

Кіберзлочини вчиняють приватні особи та організації – від хакерів-початківців до злагоджених угруповань, які використовують просунуті методики і добре підковані технічно.

Кіберзлочинці, метою яких є атака на комп'ютери, можуть заражати їх шкідливими програмами, щоб пошкодити або повністю вивести з ладу, а також видалити або викрасти дані. Також метою кіберзлочинців може бути DoS-атака (атака типу "відмова в обслуговуванні"), через яку користувачі або клієнти компанії не зможуть користуватися веб-сайтом, комп'ютерною мережею або програмними сервісами.

Кіберзлочини, в рамках яких комп'ютери використовуються для скоєння інших злочинів, можуть бути націлені на поширення шкідливого програмного забезпечення, забороненої інформації або зображень за допомогою комп'ютерів або комп'ютерних мереж.

Найчастіше кіберзлочинці одночасно і використовують, і атакують комп'ютери. Наприклад, вони можуть спочатку атакувати комп'ютери за допомогою вірусу, а потім використовувати їх для поширення шкідливого програмного забезпечення далі по мережі. У деяких країнах класифікація кіберзлочинів передбачає ще одну, третю категорію: використання комп'ютера як допоміжного засобу скоєння злочину. Прикладом може бути зберігання викрадених даних на комп'ютері.

Конвенція Ради Європи про кіберзлочинність говорить про чотири типи комп'ютерних злочинів, визначаючи їх як злочини проти конфіденційності, цілісності та доступності комп'ютерних даних та систем:

- Незаконний доступ – протиправний навмисний доступ до комп'ютерної системи чи її частини;

- Нелегальне перехоплення – протиправне навмисне перехоплення не призначених для громадськості передач комп'ютерних даних на комп'ютерну систему, з неї або в її межах);

- Втручання у дані – протиправне ушкодження, видалення, порушення, зміна чи припинення комп'ютерних даних;

- Втручання у систему – серйозне протиправне перешкоджання функціонуванню комп'ютерної системи шляхом введення, передачі, пошкодження, видалення, порушення, зміни чи припинення комп'ютерних даних.

Саме ці чотири види злочинів є власне «комп'ютерними», решта — це пов'язані з комп'ютером, або злочини, що скоюються за допомогою комп'ютера.

Існує безліч видів кіберзлочинів, але основними з них є:

Атаки з використанням шкідливих програм - комп'ютерна система чи мережа заражаються комп'ютерним вірусом чи іншим шкідливим ПЗ. Після цього кіберзлочинці можуть використовувати комп'ютер для розкрадання конфіденційної інформації, пошкодження даних та інших злочинних дій.

Фішингова атака – відправлення спаму (в електронних листах або іншими каналами), щоб обманним шляхом змусити користувачів зробити щось, що послабить їхню безпеку. Фішингові повідомлення можуть містити заражені вкладення, посилання на шкідливі сайти або прохання надати конфіденційну інформацію.

DDoS-атака - відправка множинних запитів на підключення по стандартним протоколам зв'язку, що призводить до перевантаження системи. Кіберзлочинці можуть загрожувати DDoS-атакою в рамках кібершантажу, вимагаючи гроші. Також DDoS-атака може застосовуватися як маневр, що відволікає, під час іншого кіберзлочину.

На протипагу кібератакам можна зазначити декілька розповсюджених спеціалістами з кібербезпеки порад:

- Регулярне оновлення ПЗ та операційної системи;
- Використання антивірусних програм та їх регулярне оновлення;
- Використання надійних паролів;
- Не відкривати вкладені файли у листах;
- Не переходити за посиланнями у спам-листах та на недовіrenих веб-сайтах;
- Обережність під час передачі особистої інформації;
- Спілкування офіційними каналами зв'язку;
- Уважність при відвідуванні веб-сайтів.

Висновки. З кожним роком кількість кіберзлочинів стрімко продовжує рости. Кібератаки заподіюють шкоду не тільки певним компаніям, а і звичайним людям, оскільки багато організацій зберігають конфіденційну інформацію та особисті дані своїх клієнтів і користувачів. Одна така атака, чи це витік даних, чи DDoS-атака, зараження програмою-вимагачем або іншим шкідливим ПЗ, в середньому обходиться компанії в 200 000 доларів, а то й призводить до повного припинення її роботи. Також кіберзлочини становлять величезну загрозу на державному рівні. Отже, необхідність захисту від кіберзлочинців є очевидною.

Список використаних джерел:

1. Боднар І. Р. Інформаційна безпека як основа національної безпеки / Механізм регулювання економіки. Суми. 2014. С. 68-75.
2. Ліпкан В.А., Максименко Ю.С., Желіховський В.М. Інформаційна безпека України в умовах євроінтеграції: Навчальний посібник. — К.: КНТ, 2006. — 280с.
3. Банк Р. О. Інформаційний тероризм як загроза національній безпеці України: теоретико-правовий аспект / Інформація і право 1.16. 2016 С.110-116 URL: <http://ippi.org.ua/sites/default/files/17.pdf> (дата звернення: 11.06.2022).

УДК 004.4

РОЛЬ БЕЗПЕКИ ІНФОРМАЦІЙНО–КОМУНІКАЦІЙНИХ СИСТЕМ

Хімічук І.С., магістр

e-mail: kbm221_khis@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Сьогодні інформаційні технології розвиваються дуже швидко, проте люди не завжди замислюються над цінністю інформації. Тому її крадіжка – це одна з головних цілей кіберзлочинців. З інформацією, отриманою під час атаки, злочинці можуть заволодіти вашими логінами, пароллями, документами, коштами та іншим.

Отже, система захисту інформації зараз важлива, як ніколи раніше, тому що всі організації збирають, обробляють та зберігають величезні обсяги даних на комп'ютерах та інших пристроях. Більшість цих даних може бути конфіденційними, для них несанкціонований доступ або викриття зовсім недопустимі. Кібератаки стають все серйознішими та витонченішими, компаніям та організаціям потрібно вжити відповідних заходів для гідної відсічі та забезпечення якісного захисту конфіденційної інформації.

Основні матеріали дослідження. Захист інформації є однією з серйозних та важливих проблем. Протягом усієї історії людства шляхи вирішення цієї проблеми повністю залежали від рівня розвитку технологій. У нашому інформаційному суспільстві технологія відіграє роль активатора цієї проблеми – комп'ютерні злочини стали візитівкою сьогодення.

Комп'ютерними злочинами називають ті, які пов'язані з втручанням у роботу комп'ютера, та злочини, у вчиненні яких комп'ютери використовуються як необхідні технічні засоби.

Серед причин комп'ютерних злочинів та пов'язаних з ними розкрадань інформації можна виділити такі:

- швидкий перехід від традиційних паперових технологій зберігання та передачі інформації до електронних при одночасному відставанні від технологій захисту інформації, записаної на машинних носіях;
- широке використання локальних мереж, створення глобальних мереж та розширення доступу до інформаційних ресурсів;
- постійне ускладнення програмного забезпечення, що значно знижує їхню надійність та збільшує кількість вразливостей.

Сьогодні ніхто не може назвати точну цифру загальної шкоди від комп'ютерних злочинів, але експерти сходяться на думці, що відповідні суми вимірюються мільярдами доларів. Серед основних чинників можна назвати такі:

- збитки, спричинені ситуацією, коли працівники організації, що неспроможні виконувати свої обов'язки через непрацездатність системи (мережі);
- вартість викрадених та скомпрометованих даних;
- витрати на відновлення системи, на перевірку її цілісності, на усунення вразливостей тощо

Також слід враховувати морально–психологічні наслідки для користувачів, персоналу і власників ІС та інформації. Що ж до порушення безпеки критичних додатків у державному і військовому управлінні, атомній енергетиці, медицині, космічній галузі та фінансах, то воно може мати серйозні наслідки для навколишнього середовища, економіки й безпеки держави, здоров'я і навіть людського життя.

Економічні та правові питання, приватна та комерційна таємниця, національна безпека – усе це зумовлює необхідність захисту інформації та ІС.

Під інформаційно-комунікаційною системою (ІКС) розуміємо організаційно-технічну систему, що виконує функції інформаційної системи, тобто такої організаційно-технічної системи, що реалізує певну технологію (або сукупність технологій) обробки інформації, та (або) комунікаційної системи – технічної системи, що реалізує певну технологію (або сукупність технологій) передачі даних шляхом їхнього кодування у вигляді фізичних сигналів.

Зупиняючись на термінах, які безпосередньо пов'язані із питаннями інформаційної безпеки в ІКС. Перше питання, яке виникає, – що саме ми захищаємо і яка мета цього захисту.

Ціллю захисту має бути збереження цінності інформаційних ресурсів для їхнього власника. Прямі заходи захисту спрямовані не так на самі інформаційні ресурси, як на збереження певних технологій їх створення, оброблення, зберігання, пошуку та надання користувачам. Ці технології повинні враховувати особливості інформації, що роблять її цінною, а також дозволяти користувачам різних категорій працювати з інформаційними ресурсами (створювати, знаходити, копіювати, узагальнювати, порівнювати, модифікувати, перетворювати, знищувати тощо).

Слід розуміти, що не впровадження інформаційно-комунікаційної системи дозволяє користувачам різних категорій отримувати доступ до тих чи інших інформаційних ресурсів, а фактори, до яких відноситься Закони України. Отже найважливіше для визначення ролі захисту інформації поняття – політика безпеки.

Політика безпеки інформації – це сукупність законів, правил, обмежень, рекомендацій, інструкцій тощо, які регулюють обробку інформації в ІКС. Отже, саме політика інформаційної безпеки визначає застосування тих чи інших заходів захисту, які дозволяють забезпечити безпеку інформації.

Інформаційна безпека – це стан інформації, в якому властивості інформації, визначені політикою безпеки. Багаторічний досвід захисту в ІКС дозволяє визначити головні властивості інформації, збереження яких дозволяє зберегти цінність інформаційних ресурсів. Цими властивостями є конфіденційність, цілісність і доступність інформації.

Конфіденційність – властивість інформації, завдяки якій лише авторизовані користувачі мають змогу її отримувати (тобто ознайомлюватися з інформацією).

Цілісність – властивість інформації, яка дає можливість лише авторизованим користувачам її модифікувати.

Доступність – властивість інформації, завдяки якій авторизовані користувачі можуть використовувати її згідно з правилами, встановленими політикою безпеки, не очікуючи довше заданого (короткого) періоду часу. Тобто, інформаційний ресурс має необхідний користувачу вигляд, перебуває у тому місці, де він потрібен користувачу, і тоді, коли це йому потрібно.

Термін доступність використовується не тільки, коли йдеться про інформаційні ресурси, а й про ІКС у цілому, про її компоненти або окремі ресурси. Наприклад, коректно вживати цей термін до сервера, сегмента мережі, служби електронної пошти тощо.

Висновки. Зараз інформаційно-комунікаційні системи являються одною із основних систем оброблення даних засобами накопичення, зберігання, оновлення та їх пошуку і відображення, а отже її захист дуже важливий, бо

втрата цієї інформації може завдати неповторних збитків, а також тягти за собою важку відповідальність.

Список використаних джерел:

1. Гончарова Л.Л., Возненко А.Д., Стасюк О.І., Коваль Ю.О. Основи захисту інформації в телекомунікаційних та комп'ютерних мережах. – К., 2013. 435с.
2. Голев Д.В., Кільдишев В.Й., Кононович В.Г. Інформаційна безпека інформаційно-комунікаційних систем. – О., 2010. 168с.

УДК 004.738.5

THE SEARCH FOR INTERNET CONNECTION UNDER EXTREME CONDITIONS

Zaitseva A.M., student

e-mail:azaitseva938@gmail.com

Dmytro Motorny Tavria State Agrotechnological University

Problem statement. Learning during the war is very difficult, but it is possible. At this time we apply the knowledge and skills that we got before on practice.

In February 2022 the city of Melitopol was occupied by Russian military forces. One of the problems in occupation has been the search of communication.

Knowledge of physics and the probability theory came in handy. Of course, there are days, when there is neither communication nor electricity, or signal suppression devices are working. Before full-scale invasion into the city of Melitopol there were nearly 10 internet-providers working and three operators of mobile communication functioning. During the first weeks a few of them had stopped broadcasting, and in a few months the rest joined them. For a while it was possible to find communication. Now there are only three providers and two operators of mobile communication which work. But the quality of communication is poor.

Main research. Generally, the reliability calculation scheme of internet-communication can be presented in Figure 1.



Figure 1. The reliability calculation scheme of internet-communication under general conditions

Previously, the probability of failure of each of the elements was low, whereas now (from empirical experiments) it is 50%. So, we look for some sources of signal, so that reliability scheme can look this way: (Fig.2).

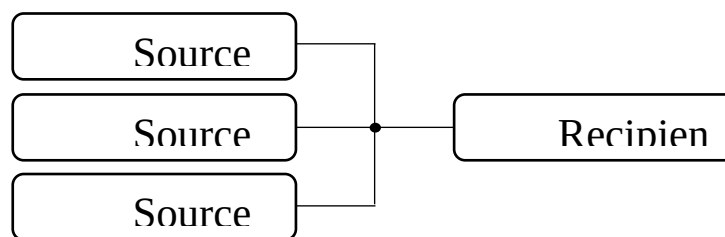


Figure 2. The reliability calculation scheme of internet-communication under extreme conditions

On practice this means that we need to connect to Wi-Fi from neighbours, relatives or a router without a password of an establishment and, preferably, they should be from different providers or be located in different buildings. This helps to find communication when a part of the city is shut down or when signal suppression devices are on. The mobile internet can also be used.

If three independent sources with likelihood of operation of 50% (0,5) are available, the probability that at least one of them works is:

$$P = 1 - 0,53 = 0,875.$$

All modern transmitters of the Internet use the ultra high frequency range. They are substantially absorbed by atmosphere, and therefore require repeaters, that work within a line of sight. Radiation that Wi-Fi devices emit falls under the radio frequency category. Phones of the second, third and fourth generations (2G, 3G, 4G) emit radio frequencies in the range of 700-2700 MHz, whereas the fifth generation phones (5G) emit up to 80 GHz [1]. Modern routers work on 2.4 GHz and 5 GHz frequencies. All ranges listed above refer to ultra-high frequencies – a part of electromagnetic spectrum with oscillation frequencies that lay approximately between 30 MHz and 3000 GHz [2].

Routers and 2G, 3G, 4G communication work in the range of decimeter and centimeter long waves [4]. Therefore, Internet receivers (mobile phone, PC or laptop) receive radio waves of the ultrashort range. These waves are not absorbed by the ionosphere, but, like sun rays, they permeate it and head into space. That is why communication, using ultrashort waves is only possible on distances, where there is nothing between recipient and transmitter antennas [2].

For the approximate calculation of the range of propagation of ultrashort waves within a line of sight (S , km), the following formula is used [3]:

$$S = 4,12 \cdot (\sqrt{H} + \sqrt{h})$$

where H – the height of transmitter antenna (meters),

h – the height of recipient antenna (meters).

If the height of a communication tower is 25 m, then at the ground level the range of the signal will reach 20 km in the absence of obstacles. On the ninth floor (the height is approximately 27 m) it can reach 42 km.

But, like all waves, radio waves can undergo diffraction, interference, reflection, absorption and attenuation. The main factor here is dielectric properties of the environment. In a city building the signal might be reflected multiple times by metal roofs. Therefore, distribution trajectories of radio waves can be the most unexpected [4]. At the same time big trees and buildings create signal shadow zones and weaken the signal. Metallic sunscreens on windows, metal roller shutters also influence the signal propagation.

Conclusions. On practice the search of the signal on the upper floors of high buildings, or watching for the appearance of the signal when moving around the city would be productive. Even within one room until mid-July of 2022 it was possible to find places with signals from Kyivstar and Vodaphone, despite the fact that local communication towers had been turned off at the end of May.

References:

1. Електромагнітне випромінювання – безпечно чи ні?
URL: <https://www.uatom.org/2022/01/17/elektromagnitne-viprominyuvannya-bezpechno-chi-ni.html>.

2. Радіохвилі. URL: <https://uk.wikipedia.org/Радіохвилі>.

3. Гордієнко І.В. Конспект лекцій з дисципліни «Поширення радіохвиль та антенно-фідерні пристрої». Дніпропетровський радіоприладобудівний коледж. Дніпро, 2016. 228 с.

4. Кравчук С.О. Поширення радіохвиль в зоні покриття безпроводових мереж зв'язку: теоретичні основи та приклади розв'язання задач [Електронний ресурс]: навч. посіб. для студ. спеціальності 172 «Телекомунікації та радіотехніка». Київ: КПІ ім. Ігоря Сікорського, 2020. 107с.
URL: https://ela.kpi.ua/bitstream/123456789/36935/1/Poshyrennia_radiokhvyl_NP.pdf

Language adviser: Symonenko S.V., Candidate of Pedagogics, Associate Professor, Dmytro Motornyi Tavria State Agrotechnological University

УДК 004.891.2

ОРГАНІЗАЦІЯ СИСТЕМИ ЗАХИЩЕНОГО ОБМІНУ ДАНИМИ ТРАНСПОРТНОГО ПІДПРИЄМСТВА ІЗ РУХОМИМ СКЛАДОМ

Хохлов М.О.

e-mail: kbm221_khmo@student.ztu.edu.ua

Єфіменко А.А., к.т.н., доцент,

e-mail: yefimenko.andrii@ztu.edu.ua

Вакалюк Т.А., д. пед. н, професор

e-mail: kik_vta@ztu.edu.ua

Державний університет «Житомирська політехніка»

Постановка проблеми. Прогрес у сфері інформаційних технологій надав можливість людству користуватись багатьма пристроями та системами, які так чи інакше, поліпшили рівень життя та комфорту.

Транспорт – один із головних видів пересування людства, з яким майже кожен з нас стикається повсякденно. Це не просто спосіб пересування із одного пункту до іншого, а вагома частина у ланцюгу постійного розвитку суспільства, його мобільності, індустріалізації та пізнанні світу.

Сучасні технології оточують кожного з нас, мобільні пристрої, різноманітні додатки Інтернету речей, вже стали невід'ємною частиною нашого життя. Не відстали від розвитку і інформаційні технології на рухомому транспорті. Намагаючись забезпечити максимальний комфорт під час подорожей, транспортні компанії почали впроваджувати комплексні проєкти IoT інформаційних систем для пасажирів.

Насамперед, це облаштування рухомого складу різноманітними пристроями, такими як IP-камери, інформаційні дисплеї, системи оплати проїзду, датчики контролю палива, електронні тахографи та багато чого іншого.

Усі вони можуть бути підключені до Інтернет-мережі підприємства, через що одразу виникло питання, чи можливо захистити дані що передаються по такій мережі.

Основні матеріали дослідження. Найбільш поширені типи атак на інформаційні системи, які сьогодні застосовують кіберзлочинці [1]:

- отримання несанкціонованого доступу до сервера, де зберігались конфіденційні дані;
- атаки на бази даних власників транспортних пластикових карт;
- атаки на інтернет-сайти транспортних компаній різних країн;
- поширення шкідливого ПЗ, навмисно перешкоджаючи роботі мереж, і як наслідок – зупинка роботи потягів, літаків та ін.;
- Dos-атаки (Denial of Service - "відмова в обслуговуванні") з метою погіршення трафіку або взагалі його припинення;

Провідні світові компанії із організації та захисту мереж вже сьогодні пропонують свої рішення даного питання. Для організації та подальшого забезпечення захисту мережі між рухомим транспортом та організацією, необхідно врахувати два критерії:

- 1) наявність відповідного мережевого обладнання із вбудованими функціями безпеки;
- 2) налаштування надійних протоколів шифрування та захисту передачі даних;

Проблематика. Проєкт від компанії Cisco "Connected Mass Transit" забезпечує скрізну, захищену мережу, від транспортних засобів до зупинок громадського транспорту та ремонтних майданчиків. Вона замінює надлишкові,

запатентовані та єдині прикладні рішення з обмеженою взаємодією або взагалі без неї. Це призводить до зниження капітальних витрат, збільшення пасажиропотоку та покращення безпеки для громадського транспорту.

Основні переваги системи:

- налаштування шаблонів, використовуючи зрозумілий та доступний інтерфейс оператора. Прив'язка шлюзів до ідентифікатору транспортного засобу;
- технік розгортає шлюзи за допомогою мобільного або веб-додатку;
- єдина панель для активації/деактивації SIM-картки та керування тарифним планом за допомогою центру керування Cisco (пропонується вибраними постачальниками послуг)
- безпеку даних забезпечує сервіс Cisco Cloud;

Проте варто відмітити, що така система має і недоліки:

- може бути занадто витратною для невеликих за розміром організацій;
- потребує заміну або переоснащення вже встановленого обладнання та додатків від сторонніх виробників;

Саме тому необхідно обрати оптимальний варіант організації мережі, при цьому зберігаючи безпеку інформації як основну мету.

Основні вимоги до обладнання на транспортному засобі можуть бути:

- стабільне, швидке та бездротове з'єднання із сервером організації за технологією LTE;
- підтримка можливості налаштування захищеної точки доступу Wi-Fi для пристроїв та датчиків транспортного засобу;
- підтримка протоколів захисту інформації;
- можливість дистанційної конфігурації;

Вибір обладнання може бути досить тривалим процесом, оскільки багато компаній пропонують своє власне рішення, але серед них є обмежена кількість саме тих, які забезпечать надійний захист і водночас довготривалий строк служби. Cisco IR829 – маршрутизатор нового покоління для використання у сучасній індустрії, зокрема на громадському транспорті. Компактний за розміром (18 x 6.5 см), маршрутизатор може бути встановлений навіть у невеликих відсіках та забезпечує усі необхідні вимоги, зокрема: наявність 4G зв'язку, Wi-Fi та 4 LAN 10/100/1000 для підключення локальних пристроїв, контроль доступу, приватність та конфіденційність даних, вбудований Cisco IOS Firewall, рішення VPN та наявність шифрування даних[2]. Обладнавши кожний транспортний засіб даним маршрутизатором, забезпечить організації надійний та швидкий зв'язок з кожним із них, що в свою чергу надасть безперервну передачу даних до серверу організації, віддалене оновлення конфігурацій додатків та пристроїв транспортного засобу.

Враховуючи постійну мобільність транспортних засобів, а також їх підключення до мережі LTE, варто обрати оптимальну модель захисту даних, яка не додасть додаткових витрат. Технології VPN, на мою думку, найкраще рішення для організації захищеної мережі транспортної компанії.

Розглянемо 4 основних відомих VPN-протоколів:

- PPTP
- L2TP/IPsec
- OpenVPN
- IKEv2

PPTP (point to point tunneling protocol) – сумнозвісний протокол типу точка-точка. Хоча й за замовчуванням використовується 128-бітове шифрування, є певні

вразливості безпеки, і одна з критичних – неінкапсульована автентифікація MS-CHAP v2.

L2TP та L2TP/IPsec - (англ. Layer 2 Tunneling Protocol). Головна перевага L2TP у тому, що цей протокол дозволяє створювати тунель не тільки в мережах IP, а й у таких, як ATM, X.25 і Frame Relay. Вбудований у всіх сучасних пристроях та системах з підтримкою VPN. Про критичні вразливості поки не було відомо.

OpenVPN — використовує протоколи SSLv3/TLSv1 та бібліотеку OpenSSL. Базово воно не підтримується на будь-яких платформах. Більш складне налагодження, в порівнянні з IPsec або L2TP.

IKEv2 – Internet key exchange version 2 - протокол тунелювання, версія 2. Результат спільної роботи компаній Cisco та Microsoft. На відміну від попередника IKEv1 не має вразливостей щодо main та aggressive режимів. В IKEv2 був доданий додатковий механізм захисту від DoS-атак. Перш ніж відповідати на кожен запит під час встановлення захищеного з'єднання (IKE_SA_INIT), VPN-шлюз надсилає хостові такого запиту дані і чекає відповіді. Якщо джерело відповіло - все гаразд, можна починати з ним генерацію ключів Діффі-Хеллмана. Якщо ні – генерація ключів не відбуватиметься, таким чином економлячи ресурси обладнання. Виходячи із наведеного, цей протокол повністю задовольняє вимоги щодо безпеки та ідеально підходить для системи захисту даних на рухомому транспорті.

Висновок. Отже, розглянувши та проаналізувавши обладнання, протоколи та, враховуючи специфіку майбутнього застосування, було прийнято рішення обрати пристрій компанії Cisco як основу, та протоколи IPsec з IKEv2 як найбільш надійні та доступні на обраному обладнанні. Таким чином, при застосуванні технології VPN при передачі конфіденційної інформації, транспортні компанії суттєво зменшать ризик кібератак на свої мережі, при цьому уникаючи додаткових витрат на розгортання супутніх додатків.

Підводячи підсумок, хотілося б зазначити, що атаки на мережі транспортних компаній, є загрозою нашого сьогодення яка зростає дедалі більше. Щоб убезпечити інформацію від зловмисників, необхідно впроваджувати нові пристрої, розробки та методи захисту у галузі кібербезпеки. Варто не забувати також і про людський фактор, своєчасно проводити інструктажі з інформаційної безпеки підрозділам транспортних компаній, обмежувати доступ стороннім особам до серверів та іншого критичного обладнання. Особливу увагу необхідно приділити розробкам моделей загроз, які у майбутньому допоможуть суспільству покращити кібербезпеку бездротових мереж зв'язку та захисту персональних даних кожного з нас.

Список використаних джерел:

1. Common cyber attacks: reducing the impact - Cyber Attacks White Paper - January 2016 [Електронний ресурс] – Режим доступу до ресурсу: [https://www.ncsc.gov.uk/files/common_cyber_attacks_ncsc%20\(1\).pdf](https://www.ncsc.gov.uk/files/common_cyber_attacks_ncsc%20(1).pdf)
2. Cisco IR829 Industrial Integrated Services Routers Data Sheet [Електронний ресурс] – Режим доступу до ресурсу: <https://www.cisco.com/c/en/us/products/collateral/routers/829-industrial-router/datasheet-c78-734981.html>

УДК 004.93+004.6

АНАЛІЗ НАЯВНИХ ІНСТРУМЕНТІВ ЗБОРУ ВЕБ-ДАНИХ

Дуб А.С., магістрант

e-mail: asdub1984@gmail.com

Український державний університет науки і технологій

Актуальність та постановка проблеми. Ринок праці як підсистема ринкової економіки взаємопов'язаний з іншими ланками, зокрема ринком товарів та послуг. Сектор інформаційних технологій постійно розвивається, зростає попит на новітні технологічні товари, на послуги фахівців з їх створення, обслуговування тощо. Функціонування індустрії зумовлює зростання і розвиток національної та світової економіки. У свою чергу, обсяги самої інформації експоненціально зростають. З'явилося поняття «великі дані», аналіз яких потребує налаштування окремих процесів та створення відповідних інструментів. Утім, наразі немає єдиної методики чи стандартизованого рішення для збору даних. Проблемі систематизації інструментарію і присвячена наведена робота.

Основні матеріали дослідження. Для збору даних застосовуються такі техніки: Копіювання та вставка вручну (copy-paste, «копіпаст») – іноді може бути єдиним задовільним варіантом, коли на сайтах навмисно встановлюють перешкоди автоматизованій обробці (CAPTCHA, відсутність можливості копіювання). Щодо перегляду сайтів оголошень про вакансії, для звичайних користувачів цей метод зазвичай є основним для збереження публікацій, які зацікавили (копіювання даних про вимоги, контактних даних, лінків на апікаційні форми у текстові файли/ хмарні сервіси/ месенджери тощо). Ручний перебір сайтів з вакансіями неефективний, фільтрація результатів недосконала. На всіх сайтах методологія пошуку Boolean search (використання +AND -NOT ||OR) не працює як належить, навіть якщо прямо вказана її наявність (Indeed, LinkedIn).

Зіставлення текстових шаблонів (regular expressions, regex, регулярні вирази) – є одним з простих, але ефективних підходів отримання даних з сторінок. Функції пошуку тексту відповідно до шаблонів (з UNIX grep чи співставленні регулярних виразів), підтримуються багатьма мовами програмування (наприклад, Perl чи Python). Утім regex не рекомендується застосувати для скрейпінгу даних, тому що при будь-яких змінах сайту ця техніка одразу стає неробочою. Крім того, regex не розпізнає різні варіації написання, які може прочитати людина. Розробник не в змозі передбачити усі варіації представленого тексту. Техніка прийнятна лише для скрейпінгу даних у визначених межах, відомої структури і оформлення, як одноразове завдання [1].

HTTP-програмування – вміст статичних і динамічних веб-сторінок може бути видобутий шляхом надсилання HTTP-запитів на віддалений веб-сервер за допомогою сокетів (обміну даними між процесами).

Синтаксичний аналіз HTML (HTML-аналізатори/парсери) – велика кількість сайтів містять великі збірники сторінок, кожна з яких формується динамічно з базового структурованого джерела (бази даних). Дані однієї категорії, як правило, кодуються на подібних сторінках з допомогою загального скрипта або шаблону. В інтелектуальному аналізі даних використовують оболонку (wrapper) – програму, що виявляє такі закономірності у конкретних джерелах інформації, витягує вміст і представляє його в реляційній формі.

Алгоритми генерації wrapper'а базуються на припущенні, що сторінки, їх атрибути і адреси в системі відповідають загальному зразкові, і що вони можуть бути визначені за спільною схемою URL. Також для аналізу HTML сторінок, отримання (retrieve) і перетворення їх вмісту можуть бути використані напівструктуровані мови запитів даних (як XQuery і HTQL).

DOM-парсинг – програми, які вбудовуючись у веб-браузер, можуть отримувати динамічний вміст як результат генерації клієнтських сценаріїв. Контрольні панелі переглядачів також інтерпретують сторінки в DOM-дереві, на підставі чого програми можуть отримати дані з окремих частин сторінок. Такі мови як Xpath можуть використовуватися для синтаксичного аналізу отриманого дерева DOM.

Платформи вертикальної агрегації даних – створюються і контролюються множиною «ботів» без втручання людини і попередньої роботи з конкретним цільовим сайтом. Для всієї вертикалі попередньо встановлюють базу знань, і згодом платформа автоматично створює ботів. За якістю інформації (кількість полів) і масштабованістю (швидкістю розширення кількості аналізованих сайтів від десятків до сотень чи тисяч) вимірюють надійність платформи. Здебільшого техніка використовується для сайтів, де загальним агрегаторам дані знайти складно або процес збирання контенту є занадто трудомістким.

Розпізнавання семантичних анотацій – після проведення скрейпінгу набори можуть містити метадані / семантичні позначки / анотації. Ці елементи можуть додатково використовуватись для пошуку конкретних даних. Якщо анотації, вбудовані в сторінки, як Microformat (набір визначених класів HTML), то цей метод вважається окремим випадком DOM-розбору. В іншому разі організовані в семантичний шар анотації підлягають зберіганню чи управлінню як окремий елемент від веб-сторінок, а скрейпери перед початком роботи використовують цей шар як вхідну інформацію для зчитування схеми даних, як інструкцію.

Аналізатори веб-сторінок (з використанням систем штучного інтелекту, машинного навчання, комп'ютерного зору) – у процесі цієї техніки AI намагається прочитати структуру, ідентифікувати потрібні поля, зібрати та інтерпретувати інформацію зі сторінок. Вимагається тренування моделі на тестовому зразкові достатнього обсягу.

Звернення до проксі-сервера – якщо сайт є HTML- або XML-документом і до нього дозволені крос-доменні запити, то можна отримати вміст документа з допомогою одного з існуючих проксі-сервісів. У тій чи іншій мірі вищезазначені техніки веб-скрейпінгу використовують програмні інструменти (деякі з них у табл.1). Прикладна реалізація може бути у вигляді:

- бібліотек для парсингу на різних мовах (Nokogiri для Ruby – підтримує HTML та XML; Puppeteer для Node.js для сайтів з Javascript, коли на сторінці користувач для відображення даних має провести певні ітерації; XPath як елемент мови XSLT дозволяє обирати гілки в документі XML; Requests, Requests-HTML, Selenium як headless browser для тестування, автоматизації, скрейпінгу);

- додатків (PHP-Crawler для найпростішого пошуку на локальному веб-сайті; pyspider Web Crawler System на Python; Larm Lucene Advanced Retrieval Machine на Javascript, не дуже гнучкий, останнє оновлення у 2003; Heritrix – open-source на Java від Internet Archive'y; Proxy Crawler на Python – збір даних з Github);

- сервісів (онлайн-, хмарних, SaaS (Software as a Service: DataHen), DaaS (Desktop as a Service);

- розширень/плагінів до браузерів.

Таблиця 1. Вибірка ПЗ для веб-скрейпінгу

Програмний засіб	Характеристики
Apify	веб-сервіс; перетворює сайти на API – збір даних, кроулер, автоматизація процесів; фріміум, відкритий код
BibSonomy	система обміну закладками та списками літератури (формат BibTex); безкоштовно, відкритий код
Data Miner	розширення для Chrome, зберігає в .xls або .csv; фріміум, пропрієтарний
Datahut	веб-сервіс для компаній, використовує технології відкритого коду; платний, пропрієтарний
DataScraping.co	рішення для малого та середнього бізнесу; зберігає дані у хмарі; платний, пропрієтарний
DataStock	завантаження готових до використання попередньо просканиваних веб-наборів даних з різних країн і галузей; платний, пропрієтарний
dexi.io	доступний з браузера; засіб для процесингу даних; платний, пропрієтарний
DiffBot	перетворює веб у БД, використовує комп'ютерне бачення, автоматичне отримання структурованого вмісту зі статей, сторінок продуктів, ін.; платний, пропрієтарний
import.io	веб-сервіс; не вимагає кодування, інтерфейс drag'n'drop – візуальний вибір потрібних полів; платний, пропрієтарний
JobsPikr	платформа збору даних про роботу, яка витягує дані безпосередньо з веб-сайтів компаній, використовуючи методи машинного навчання на основі AI
Marketing Miner	інструмент для SEO спеціалістів і маркетологів, фріміум, пропрієтарний
morph.io	використовує технології відкритого коду; безкоштовний, пропрієтарний
Mozenda	перетворює контент веб-сторінок у структуровані дані; використовує додаток у Windows; фріміум, пропрієтарний
Newspaper	витяг метаданих новин, повнотекстових текстів та статей, побудований за допомогою Python; безкоштовний, відкритий код
Number Analytics	статистичний пакет онлайн, з побудовою графіків, платний, пропрієтарний
Octoparse	візуальне програмне забезпечення для збору веб-даних; фріміум, пропрієтарний
ParseHub	фріміум, пропрієтарний
Portia	візуальне програмне забезпечення для збору веб-даних; від розробників Scrapy; безкоштовний, відкритий код
OutWit Hub	збір з інтернету або локальних ресурсів, розпізнає та розподіляє посилання, зображення, документи, контакти, повторюваний словниковий запас та фрази, RSS-канали та перетворює структуровані та неструктуровані дані у відформатовані таблиці, які можна експортувати до електронних таблиць або баз даних; пропрієтарний
Phantombuster	автоматизація маркетингу (у LinkedIn, Twitter, Facebook, Instagram), створення API інших сайтів; фріміум, відкритий код

Програмний засіб	Характеристики
Product API by Fetcher	автоматичний API на вимогу забезпечує стартапам та підприємствам послугу збору точних даних про електронну комерцію; фріміум, пропрієтарний
ScrapeHero	без кодування; платний, пропрієтарний
ScrapeHunt	для побудови SaaS/ пошуку нових клієнтів/ автоматизації маркетингу; платний, пропрієтарний
Scraper	розширення для браузера, працює через праву клавішу миші і вибір полів автоматично, коригування через XPath
ScrapeStorm	візуальне програмне забезпечення на основі AI, без написання коду; фріміум, пропрієтарний
ScrapingBee	декларують що є найлегший API для веб-скрейпінгу; платний, пропрієтарний
Scrapinghub	платформа для розгортання та запуску веб-сканерів (crawlers); платний, пропрієтарний
Scrapy	програмний каркас (framework) для спільної роботи з добування необхідних даних із веб-сайтів. Швидкий, простий, але при цьому розширюваний спосіб; безкоштовний, відкритий код. Вимагає написання коду, не може зчитувати JavaScript, треба надсилати запити AJAX чи використовувати інший інструмент (як Selenium)
SEOMagnifier	платформа з різними сервісами для веб-майстрів (перефразування, підрахунок кількості слів, спінінг статей [заміна слів, речень, абзаців на подібні версії], перевірка на плагіат); безкоштовний, пропрієтарний
TheWebMiner	пошук потенційних клієнтів або проведення дослідження ринку; фріміум, пропрієтарний
WhatsApp	у політиці приватності з 2021 року зазначено про добування даних; безкоштовний (“користувачі стали продуктом”); пропрієтарний
Visallo	застосунок для аналітики та криміналістики; платний; пропрієтарний
WinTask	автоматизація заповнення веб-форм і скрейпінг сайтів, десктопний застосунок Windows і розширення Chrome
Wintr	проксі-сервіс та веб-API для скрейпінгу, який дозволяє вам зібрати та проаналізувати HTML-код будь-якої веб-сторінки за допомогою Cheerio (бібліотека JQuery), щоб перетворити його на персоналізований набір даних; безкоштовний, пропрієтарний
YabTab	сервіс на основі AI; безкоштовний для особистого, пропрієтарний

Джерело: узагальнено автором на підставі [2; 3, с.1027; 4]

Подібне програмне забезпечення може автоматично розпізнавати структуру даних сайту за тегами або слугувати інтерфейсом збору. Це усуває необхідність писати код або скрипти для витягування вмісту, інтерфейсів баз даних, які можуть зберігати зібрані дані локально.

Бачимо, що наведені 35 інструментів і подібні до них мають низку обмежень: часто є пропрієтарними, платними (або фріміум), налаштування обмежено функціоналом сервісів чи додатків. Як часто в таких випадках, найкращим рішенням в сучасних умовах є розробка власного програмного продукту визначеної специфікації.

Для прикладного вирішення завдання – формування інформаційно-довідкової системи для аналізу ринку праці IT-фахівців – обираємо мову Python з відповідними бібліотеками для скрейпінгу, парсингу та візуалізації.

На рис. 1 визначена різниця між різними засобами для веб-скрейпінгу у Python.

Scrapy	Selenium	BeautifulSoup
фреймворк, написаний на Python	інструмент автоматизації	модуль для статичних сторінок
використовує XPath	на динамічних сайтах симулює поведінку веб-браузера, кліки, введення логінів	залежить від бібліотек
найкращий з точки зору продуктивності і насправді найшвидший серед 3-х інструментів	повільніший за BeautifulSoup HTTPS requests	повільний, може бути пришвидчений через багатопотоковість
має вбудований парсер	використання парсерів BeautifulSoup чи lxml для пришвидчення	використовує сторонні парсери

Рисунок 1. Засоби скрейпінгу в Python

Джерело: систематизовано автором на підставі [3, с.1028; 5; 6, с.9; 7]

Як було зазначено на рис. 1, Scrapy не відображає JavaScript; потрібно надсилати запити AJAX, щоб отримати дані, приховані за процесами JavaScript, або скористатися стороннім інструментом, таким як Selenium. Подальший вибір засобів буде ґрунтуватися на характеристиках сайтів оголошень про роботу – статичні чи динамічні і який обсяг даних вони містять.

Висновок. Розглянуті методи та інструменти підтвердили гіпотезу про відсутність на даний момент уніфікованої технології збору даних з мережі і підтвердили необхідність розробки власного програмного продукту для вирішення специфічних завдань. Проведений аналіз є одним з етапів розроблення програмного аналітичного комплексу про ринок праці IT-фахівців.

Список використаних джерел:

1. Stack Overflow. RegEx Match Open Tags except XHTML Self-Contained Tags. 12 Nov. 2020. URL: <https://stackoverflow.com/questions/1732348/regex-match-open-tags-except-xhtml-self-contained-tags>
2. AlternativeTo. 2022. Apps with 'Data Mining' feature. URL: <https://alternative.to.net/feature/data-mining/>
3. Luscombe A., Dick K., Walby K. Algorithmic thinking in the public interest: navigating technical, legal, and ethical hurdles to web scraping in the social sciences. Quality & Quantity. 2021. 56(3), pp. 1023-1044. <https://doi.org/10.1007/s11135-021-01164-0>
4. Farley E. J., Pierotte L. Web scraping – an emerging data collection method for criminal justice researchers. JRSA fact sheet. 2017, pp.1-9.
5. Grimes J. Scrapy Vs. BeautifulSoup Vs. Selenium for Web Scraping. 13 Apr. 2021. URL: <https://www.bestproxyreviews.com/scrapy-vs-selenium-vs-beautifulsoup-for-web-scraping/>
6. Nguyen V., Sukunesan S., Huynh M. Web scraping Instagram Pre and During Covid-19: Examining customer engagement on Australian SMEs accounts. Australasian Journal Of Information Systems. 2022. Vol. 26. <https://doi.org/10.3127/ajis.v26i0.3447>
7. Vanden Broucke S., Baesens B. Practical Web Scraping for Data Science Best Practices and Examples with Python. 2018.

УДК 004

ПОРІВНЯННЯ МЕТОДОЛОГІЙ ТЕСТУВАННЯ «ЧОРНОЇ» ТА «БІЛОЇ» СКРИНЬКИ

Вакалюк Т.А., д.пед.н., професор

e-mail:kik_vta@ztu.edu.ua

Курачинська А.Р., студентка

e-mail:kbm221_kar@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Тестування – це один із важливих етапів життєвого циклу розробки програмного забезпечення. Воно необхідне для забезпечення безпеки, надійності та якості продукту. На сьогоднішній час існує безліч ризиків для програмного забезпечення, які можуть вплинути на цілісність даних. Тестування на проникнення забезпечує тестування, при якому відбувається симуляція реальної атаки на систему. Даний тип тестування забезпечує оцінювання здатності наявних засобів безпеки протистояти загрозі з боку зловмисника та дозволяє виявити вразливість нульового дня. Вразливість нульового дня – це раніше невідома вразливість для розробників чи користувачів.

Основні матеріали дослідження. Тестування методом «чорної скриньки» - це процес тестування системи та її поведінки незалежно від її внутрішньої структури, архітектури та реалізації [2]. Цей метод може застосовуватися до всіх рівнів тестування ПЗ, але в основному використовується на вищому рівні. Тестування методом «чорної скриньки» схоже на реальну атаку зловмисника, оскільки він демонструє, як злочинці, які не розуміють систему, можуть отримати конфіденційну інформацію.

Метод «чорної скриньки» поділяється на наступні типи:

- Функціональне тестування. Форма тестування в закритому середовищі досліджує, як програмне забезпечення працює відповідно до конкретних функціональних вимог [3]. Тести перевіряють базову функціональність програмного забезпечення, надаючи певні вхідні дані та порівнюючи його вихідні дані з очікуваною поведінкою;
- Нефункціональне тестування. Тип тестування при якому перевіряється нефункціональні вимоги програми: продуктивність, надійність, зручність у використанні;
- Регресивне тестування. Форма тестування, що використовується для перевірки впливу нещодавнього оновлення на існуючий функціонал програми. Підхід включає в себе частковий тест динамічного аналізу, який повторно виконує тестові кейси на існуючих функціях, щоб переконатися, що нові зміни не призведуть до небажаних побічних ефектів [3].

Переваги методу «чорної скриньки»:

- Пентестер не обов'язково повинен мати технічну освіту, бо важливо перевіряти з точки зору користувача [1];
- Тестування розпочинається одразу після завершення розробки проекту, оскільки розробники і тестувальники працюють незалежно один від одного;
- Метод є більш ефективним для великих і складних проектів;
- Дефекти можуть бути виявленими на ранніх стадіях тестування.

Недоліки методу «чорної скриньки»:

- Деякі тест-кейси можуть дублюватися або пропускатися;
- Є можливість, що певні функції додатку залишаться непротестовані, тому що тестувальник немає знання про внутрішню структуру;
- Метод є більш ефективним для великих і складних проєктів;
- Потреба в чітких і явних специфікаціях тестування.

Метод «чорної скриньки» слід використовувати коли:

- Під час приймального тестування для розуміння впливу сторонніх рішень на безпеку додатків;
- Під час забезпечення якості, щоб переконатися у коректній роботі додатку перед розгортанням;
- Для інтеграційного тестування нових функцій під час розгортання та впровадження додатку;
- У розробці для розуміння зручності та користувацького досвіду.

Тестування методом «білої скриньки» - це процедура написання чи вибору тест-кейсів з урахуванням аналізу внутрішнього пристрою системи чи компонента. Метод «білої скриньки» допомагає заощадити час і гроші, також він корисний для імітації цілеспрямованої атаки на певну систему з використанням якомога більшої кількості векторів атак [4].

Метод «білої скриньки» поділяється на наступні типи:

- Модульне тестування. Виконується на кожному модулі або блоці коду в міру його розробки. Зазвичай розробник виконує дане тестування, коли пише кілька рядків коду, одну функцію або об'єкт і тестує їх, щоб переконатись у їх роботі. Допомагає виявити більшість помилок на ранніх етапах розробки програмного забезпечення;
- Тестування на витік пам'яті. Витік пам'яті – це одна з основних причин повільної роботи додатків, тому даний тип тестування потрібно виконувати, коли повільно працює додаток;
- Тестування на проникнення. Тестувальник отримує повну інформацію про вихідний код програми, детальну інформацію про мережу, пов'язані IP – адреси та всю інформацію про сервер на якому працює додаток. Мета полягає, щоб виявити загрози безпеці, атакуючи з різних шляхів.
- Тестування мутацій. Часто використовується для виявлення найкращих методів кодування, які використовуються для розширення програмного рішення.

Переваги методу «білої скриньки»:

- Тестування можна виконувати на ранніх етапах, коли навіть не створено користувацького інтерфейсу;
- Тестування є більш ретельним, оскільки зазвичай покриваються всі шляхи коду;
- Тест-кейси легко автоматизуються;

Недоліки методу «білої скриньки»:

- Тестувальник повинен мати певні технічні навички та детально розуміти програмування і реалізацію;
- Якщо розробник виконує тестування недетально, то це призведе до багів;
- У разі автоматизації тестування на цьому рівні, підтримка тестів може виявитись досить накладною, якщо програма часто змінюється [5].

Метод «білої скриньки» слід використовувати коли:

- Тестувальник хорошо знає архітектуру продукту, код і може орієнтуватися в самій програмі;

- Спеціаліст з якості має необхідні знання коду, для того щоб вибрати вхідні дані, які будуть оброблятися;

- Наявна проектна документація для виконання тестування.

Для кращого порівняння методів «чорної» та «білої» скриньок нижче наведено таблицю 1 з порівнянням даних методик [5].

Таблиця 1 – Порівняння методів тестування

Критерій	Black Box	White Box
Визначення	Тестування, як функціональне так і не функціональне, не передбачає знання внутрішнього складу компоненту або системи	Тестування засноване на аналізі внутрішньої структури компонента або системи
Рівні, до яких можна застосувати техніку	Приймальне та Системне тестування	Юніт та Інтеграційне тестування
Хто виконує	Тестувальник	Тестувальник або розробник
Знання програмування	Не потрібно	Необхідно
Знання реалізації	Не потрібно	Необхідно
Основа для тест-кейсів	Специфікація	Проектна документація

Висновок. Отже було розглянуто два методи тестування, які мають певні переваги та недоліки. Тестування як методом «чорної скриньки», так і методом «білої скриньки» - важлива частина в арсеналі тестувальника і не повинна бути єдиною. При виборі методу тестування слід дивитися на час, який виділяється для тестування, на знання, які наявні у тестувальника. Також слід звернути увагу на рівні для яких потрібно виконати тестування і не забувати чи потрібно автоматизовувати тести чи ні. Та найголовнішим в тестуванні є саме якість кінцевого продукту та задоволення користувачів.

Список використаних джерел:

1. Black Box Testing: An In-Depth Tutorial With Examples And Techniques [Електронний ресурс] // Software Testing Help. – 2022. – Режим доступу до ресурсу: <https://www.softwaretestinghelp.com/black-box-testing/>.
2. Olomu M. What Is Black Box Testing? [Електронний ресурс] / Mario Olomu. – 2022. – Режим доступу до ресурсу: <https://dzone.com/articles/what-is-black-box-testing>
3. Sudip S. Black-Box Penetration Testing in Cybersecurity [Електронний ресурс] / Sengupta Sudip // Crashtest Security. – 2022. – Режим доступу до ресурсу: <https://crashtest-security.com/black-box-testing/>.
4. Types of Pen Testing: Black Box, White Box & Grey Box [Електронний ресурс] // The Redscan Team. – 2022. – Режим доступу до ресурсу: <https://www.redscan.com/news/types-of-pen-testing-white-box-black-box-and-everything-in-between/>.
5. White/black/grey box-тестування [Електронний ресурс] // QaLight. – 2022. – Режим доступу до ресурсу: <https://qalight.ua/baza-znaniy/white-black-grey-box-testuvannya/>.

УДК 514.1+528.7

АНАЛІЗ І КОМП'ЮТЕРНА ОБРОБКА ЗОБРАЖЕНЬ ЗА ДОПОМОГОЮ ГРАФІЧНОГО ІНТЕРФЕЙСУ КОРИСТУВАЧА MATLAB

Мацулевич О.С., к.т.н., доцент

e-mail: oleksandr.matsulevych@tsatu.edu.ua

Поспелов М.А., ст.. викладач

e-mail: mykyta.pospielov@tsatu.edu.ua

Тетервак І.Р., асистент

e-mail: illia.tetervak@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність дослідження та постановка проблеми. Використання засобів MATLAB дає широкі можливості для обробки двовимірних сигналів, де, як наочна область, розглядаються цифрові зображення, їх обробка і аналіз. Важливою гідністю системи MATLAB є програми, що реалізовані у вигляді графічних інтерфейсів користувача (GUI) і забезпечують вирішення різноманітних наочно-орієнтованих завдань в простій, наочній і зручній формі. Загальна постановка таких завдань — це аналіз і поліпшення якості зображень. Подібні завдання вирішуються з використанням різноманітних процедур (функцій), зокрема тих, що реалізують різні методи обробки сигналів: лінійної і нелінійної фільтрації, препарування зображень, спектрального аналізу (обчислення і дослідження двовимірного Фур'є-спектру просторових частот зображення) та інші. Для збільшення інформативності представлення результатів застосовуються засоби демонстрації поперечних перетинів зображення і об'ємних зображень поверхонь з використанням підсвічування.

До програм, спроможних забезпечувати якісну обробку цифрових зображень, відносяться програми WAVEMENU (GUI, яка дозволяє проводити аналіз одновимірних і двовимірних сигналів за допомогою wavelet – перетворення) [1], SPTOOL (GUI для фільтрації і спектрального аналізу одновимірних сигналів) [2] та інші. MATLAB також надає широкий набір інструментальних засобів обробки зображень, проте готового GUI в цій області в існуючих версіях MATLAB немає. Програмою, яка використовує багато стандартних функцій MATLAB, що входять до складу Toolbox обробки сигналів є програма IMAGEGUI.

Основні матеріали дослідження. Програма IMAGEGUI підтримує стандартні графічні формати (jpg, bmp, tif, psx та інші) реалізує паралельну роботу з різними зображеннями, багатократної обробки одного зображення, а також його фрагментів.

Головний модуль IMAGEGUI містить опис головного графічного вікна, вид якого представлений на рис. 1. Всі виконувані функції є окремими модулями. Далі описуються елементи, розташовані на панелі головного вікна, і відповідні ним процедури (рис. 1).

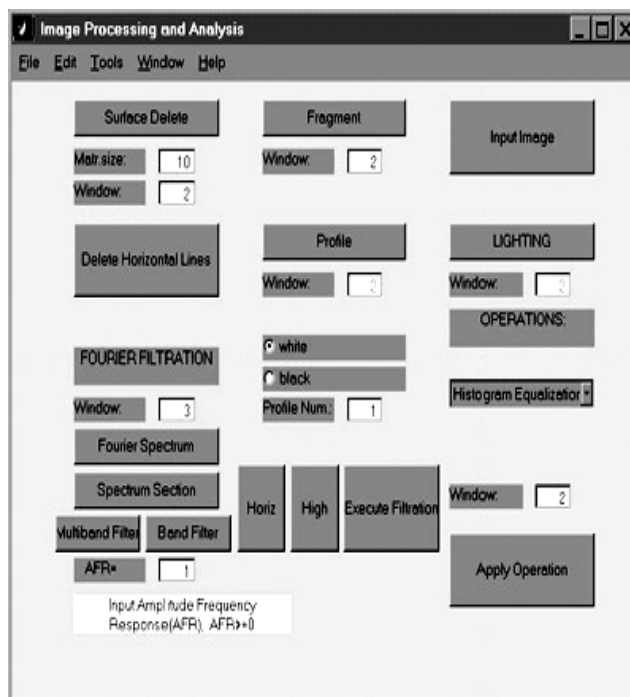


Рисунок 1. Введення зображення

Програма підтримує введення зображень, що мають стандартний графічний формат (кнопка «Input Image»). Допустиме введення кольорових зображень, проте при обробці вони конвертуються в чорно-білі.

Для того, щоб почати роботу, необхідно натиснути кнопку «Input Image» і вибрати файл в тому, що з'явився після цього стандартному вікні діалогу введення. Введений графічний файл відображається в окремому вікні, і стають доступними операції його обробки, описані нижче.

Видалення горизонтальних смуг.

Ця функція виконується при натисненні кнопки «Delete Horizontal Lines» головного вікна програми (рис. 1). Горизонтальні смуги різної яскравості є типовим видом зашумлення зображень, зокрема, отриманих за допомогою скануючих приладів, оскільки при переході на новий рядок можуть змінитися деякі умови в зовнішньому середовищі або в експериментальній установці. Для усунення смуг реалізується алгоритм, що здійснює приведення середніх значень яркостей точок зображення по кожному рядку до одного середнього значення. Нехай Q — середнє значення всіх точок двовимірної матриці зображення A_{ij} , а P_i — середнє значення точок в i -тому рядку. Тоді значення в кожній точці перетвориться відповідно до формули:

$$a'_{ij} = a_{ij} + (Q - P_i). \quad (1)$$

У описуваній версії програми цей алгоритм працює з повним масивом початкових даних.

Виділення фрагмента зображення.

Фрагмент може бути виділений із зображення, що знаходиться в будь-якому з відкритих графічних вікон. Для виділення фрагмента необхідно натиснути кнопку «Fragment» (рис. 1), якщо ця функція виконується вперше після запуску програми IMAGEGUI, а потім у віконці «Window», розташоване під цією кнопкою, ввести номер графічного вікна із зображенням, після чого натиснути «Enter». Фрагмент зображення визначається шляхом вказівки мишею кінцевих

точок будь-якої його діагоналі. Виділений фрагмент виводиться в новому графічному вікні, а на початковому зображенні промальовувалася рамка, відповідна його межах. При повторних викликах цієї функції досить вводити номер вікна із зображенням.

Блок «Operations» — виконання функцій корекції гистограми зображення і стандартної фільтрації

Цей блок знаходиться в правому нижньому кутку головного вікна програми (рис. 1) і містить випадний список, що містить функції, які можуть бути використані; вікно редагування «Window», в якому необхідно вказати номер графічного вікна, що містить зображення і кнопку «Apply Operation». При натисненні цієї кнопки вибрана із списку функція застосовується до зображення, що знаходиться в графічному вікні з вказаним номером, і результат відображається в новому графічному вікні.

У цьому блоці програми реалізовані функції побудови гистограми зображення, збільшення і зменшення контрасту і яскравості зображення, згладжування, знищення локальних дефектів, медіанної фільтрації і вінерівської фільтрації.

Результати виконання деяких функцій цього блоку продемонстровані на прикладі зображення зрізів мозку щура (рис. 2). Зображення отримане за допомогою цифрової відеокамери. За допомогою операції «Boundary interpolation» можна видаляти локальні дефекти. На, побудований в процесі виконання «Boundary interpolation» інтерактивно, за допомогою миші. Результат виконання цієї операції (що з'являється після натиснення правої кнопки миші) приведений на рис. 3.

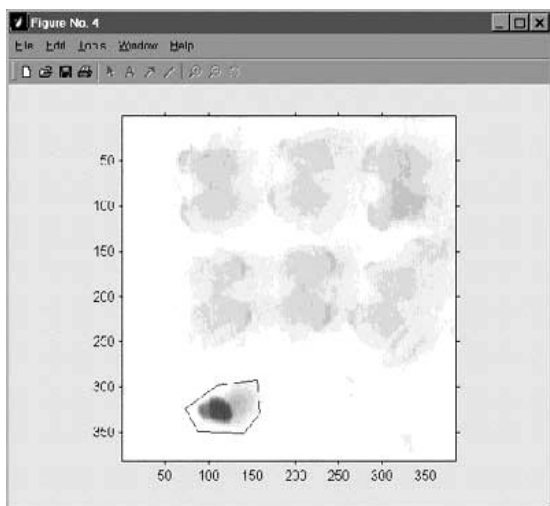


Рисунок 2. Дефект поміщений в багатокутник

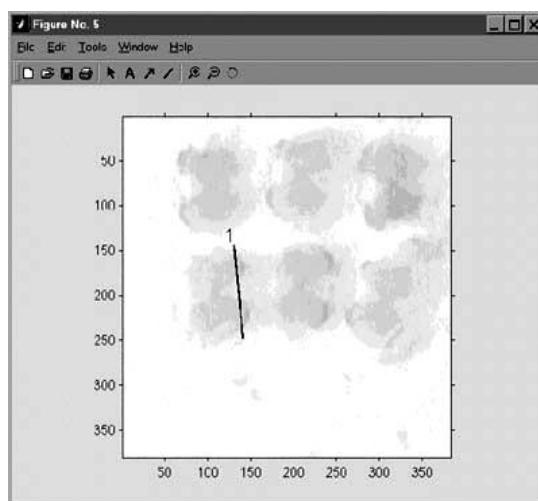


Рисунок 3. Результат виконання операції.

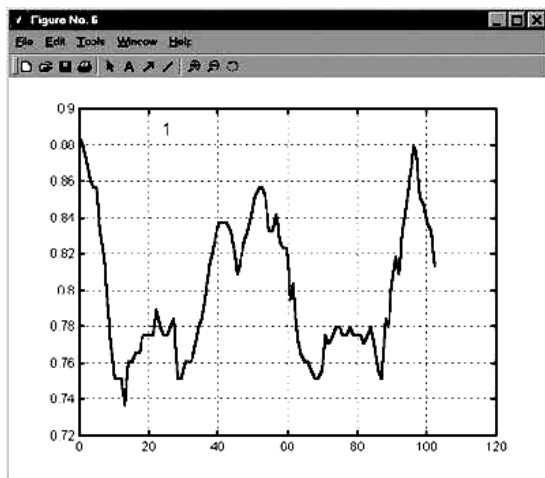


Рисунок 4

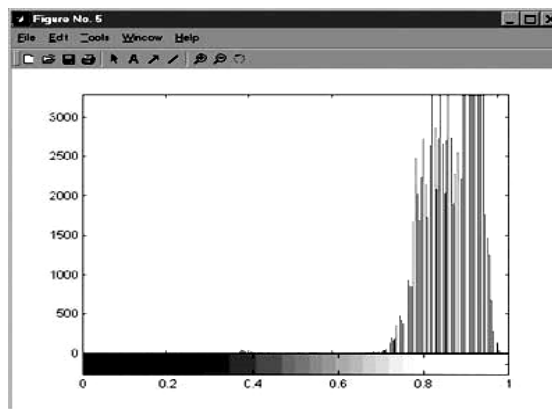


Рисунок 5. Гістограма зображення, при виконанні операції «histogr».

Гістограма зображення, показаного на рис. 2, побудована при виконанні операції «histogr» і приведена на рис. 4.

Побудова профілю поверхні.

Ця функція програми реалізується за допомогою елементів, розташованих в середній частині головного вікна «IMAGEGUI»: кнопка. «Profile» і елементи, що знаходяться нижче за неї (рис. 1). Перед прома-льовуванням профілю необхідно занести його номер у віконце «Profile Num.». Цей номер призначається користувачем, він буде поміщений на зображенні поряд з лінією перетину, а також написаний на графіці перетину в місці, вказаному натисненням лівої кнопки миші. Колір лінії перетину, яка наноситься на зображення, регулюється за допомогою перемикача «white/black».

Для побудови профілю зображення необхідно натиснути кнопку «Profile», потім ввести номер графічного вікна з цим зображенням у віконце «Window», розташоване під цією кнопкою «Profile» і натиснути «Enter». Після цього на вибраному зображенні за допомогою лівої кнопки миші указуються кінцеві точки, через які проводиться перетин. При подальших викликах цієї функції натискати кнопку «Profile» не обов'язково, досить вказати номер вікна із зображенням, а потім натиснути «Enter». Робота цієї функції продемонстрована на прикладі рис. 3 де чорною лінією показано вибране місце перетину, а на рис. 5 приведено самоперетин.

Тривимірне представлення зображення з підсвічуванням.

Для представлення вибраного зображення в аксонометрії, із затінюванням і підсвічуванням, необхідно натиснути кнопку «Lighting» (рис. 1), потім ввести номер графічного вікна з вибраним зображенням і натиснути «Enter». При повторних зверненнях до цієї функції натискати кнопку «Lighting» немає необхідності. Побудова тривимірної поверхні і її інтерактивне обертання здійснюється за допомогою набору засобів, що надаються системою MATLAB. Приклад такого зображення приведений на рис. 6.

Видалення нерівностей фону.

Операція видалення нерівності фону (тренда) на зображенні виконується при натисненні кнопки «Surface Delete». Перед цим у віконця, розташовані під цією кнопкою, необхідно ввести номер графічного вікна із зображенням, підметом обробці, а також розмір квадратної матриці $T(x,y)$ (у пікселях), операції, що є оперантом, згортки. Видалення тренда в даній програмі здійснюється шляхом фільтрації з використанням згортки. Спочатку обчислюється матриця згладженої поверхні $U(x,y)$ методом ковзної середньої, а потім цей результат віднімається з початкового зображення:

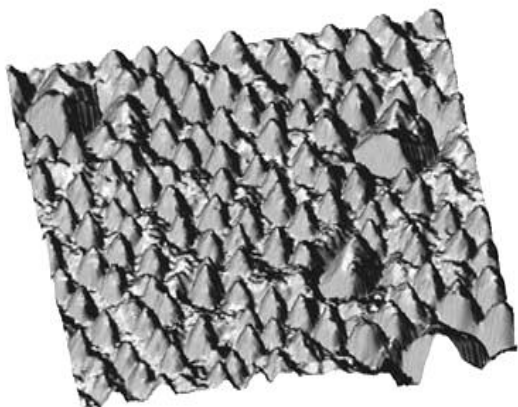


Рисунок 6. Приклад зображення

$$U(x,y) = A(x,y) ** T(x,y),$$

$$A'(x,y) = A(x,y) - U(x,y),$$

де: $**$ — операція згортки; $A(x,y)$ — початкове зображення, $T(x,y)$ — одинична матриця, з розміром, відповідним розміру найбільших

елементів на зображенні, які необхідно зберегти.

При виконанні операції згортки для запобігання спотворенням на краях зображення здійснюється його парне продовження.

Виконання цієї операції продемонстроване на рис. 7. Початкове зображення поверхні, отримане за допомогою скануючого силового мікроскопа, а результат видалення нерівності фону за допомогою описаного вище методу.

Двовимірні перетворення Фур'є і лінійна фільтрація зображень.

Елементи, за допомогою яких в програмі здійснюються функції, пов'язані з перетворенням Фур'є, розташовані під написом рис. 1

Обчислення двовимірного дискретного перетворення Фур'є служить основою для побудови фільтрів в області просторових частот і здійснення лінійної фільтрації (кнопка «Fourier Spectrum»). Побудова двовимірного спектру для зображення, номер якого заздалегідь введений у віконце «Window», розташоване над кнопкою «Fourier Spectrum», проводиться після натиснення цієї кнопки.

Дослідження спектру зображення проводиться за допомогою побудови його перетинів площинами, перпендикулярними площині просторових частот. Результатом є кількісна інформація про досліджувані зображення: величини просторових періодів і інформація про орієнтацію періодів в координатах зображення. Для виконання цієї функції необхідно натиснути кнопку «Spectrum Section», а потім за допомогою миші вказати напрям перетину. Величини просторових частот виводяться в командному вікні при натисненні лівої кнопки миші в полі графіка перетину. Ця операція закінчується після

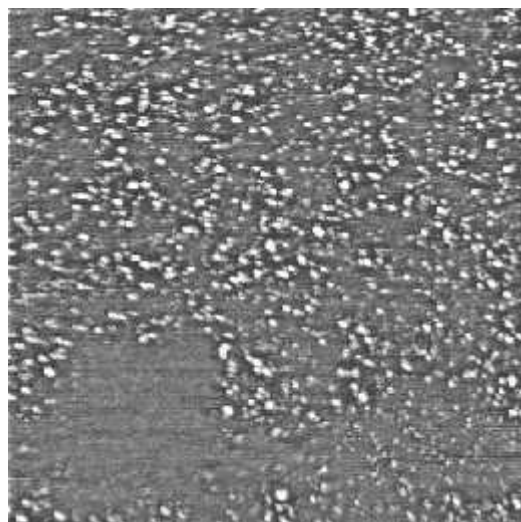


Рисунок 7. Початкове зображення поверхні

натиснення правої кнопки миші. Після цього можна провести новий перетин, або перейти до побудови фільтру.

Поліпшення якості зображення або його препарування з метою виділення характерних ознак здійснюється шляхом лінійної фільтрації. Побудова частотної характеристики фільтру відбувається в інтерактивному режимі. Вибирається певний вид фільтру: смуговий, нижніх або верхніх частот, режекторний, багатосмуговий або фільтр-коректор, що підкреслює дрібні деталі зображення. Фільтри, використовувані в GUI, розраховуються методом вікон.

Частотна характеристика фільтру, що ідеалізується, задається у вигляді двовимірної маски в координатах просторових частот. При цьому програма побудована таким чином, що в основній частині апертури задаються одиниці, а в робочій частині записуються числа, великі або менші одиниці, залежно від характеру фільтру. Значення амплітудно-частотної характеристики (на панелі GUI позначене як AFR — Amplitude Frequency Response повинне бути введене до проведення перетину. Можливе використання різних значень АЧХ для різних смуг. Виділена область частот визначається натисненням лівої кнопки миші спочатку в центрі смуги, а потім на одній з її меж. На зображенні двовимірного спектру здійснюється контрольне промальовування характеристик фільтру, що синтезується. Тривимірне представлення фільтру виводиться в окремому графічному вікні. Фільтрація виконується після натиснення кнопки «Execute Filtration».

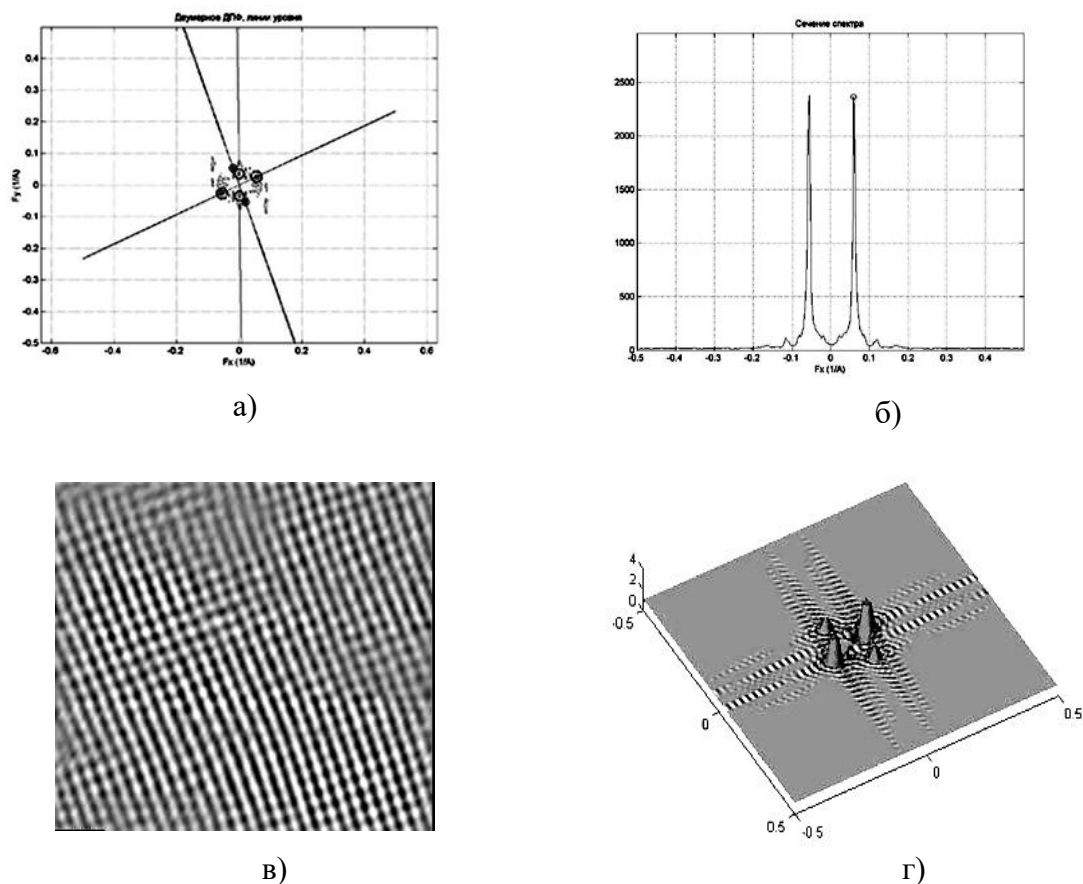


Рисунок 8

На рис. 8 показані різні етапи фільтрації зображення кристалічної решітки GaAs, отриманого в скануючому тунельному мікроскопі. На рис. 8, а приведені

початкове, сильно зашумлене зображення, на рис. 8, б — модуль його двовимірного Фур'є-перетворення. На цьому ж рисунку лініями показані перетини спектру, на яких відмічені області частот, для яких було здійснено побудову багатосмугового фільтру. Один з перетинів спектру показаний на рис. 8, в. Тривимірне зображення побудованого фільтру — на рис. 8, г.

Приклад обробки зображення в програмі IMAGEGUI (рис. 9).

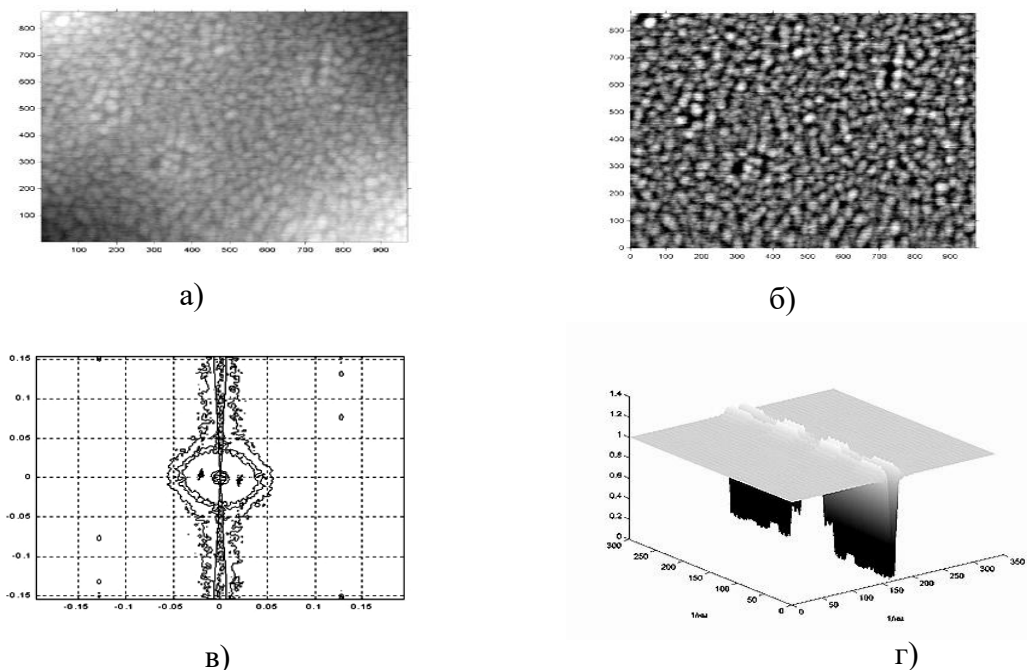


Рисунок 9.

Першою повинна бути виконана операція введення зображення за допомогою кнопки «Input Image» (рис. 9, а).

Далі зображення піддається обробці, проведеній за декілька кроків:

1. Видалення горизонтальних смуг за допомогою команди «Delete Horizontal Lines» (рис. 9, б);

2. Видалення нерівності фону за допомогою команди «Surface Delete» з параметром «Matr. size», рівним 22 квадратним пікселям (рис. 9, в).

3. Побудова двовимірного Фур'є-спектру зображення (рис. 9, г) за допомогою команди «Fourier Spectrum» модуль якого приведений на рис. 9, г.

4. Для видалення вузьких горизонтальних ліній, помітних на зображенні рис. 9, в, використано режекторний фільтр, що пригнічує частоти у вузькому секторі. Межі смуг фільтру, що ідеалізується, показані лініями на рис. 9, г. Для побудови цього фільтру необхідно натиснути кнопку «Horiz», а після цього — «Execute Filtration».

Висновки. У статті представлена і описана графічна інтерактивна програма обробки зображень IMAGEGUI, створена в середовищі MATLAB. Ця програма містить різні функції обробки і аналізу зображень, включаючи лінійну і нелінійну фільтрацію, аналіз двовимірного спектру просторових частот зображення, корекцію гістограми,

дослідження локальних властивостей зображення шляхом виділення фрагментів і проведення перетинів, тривимірне уявлення з підсвічуванням.

Список використаних джерел:

1. Андреев И.В. Ланне А.А. MATLAB для DSP: SPTool інструмент для расчета цифровых фильтров и спектрального анализа сигналов // Цифровая обработка сигналов. 2000. № 2. С. 6–13.
2. Бинниг Р., Рорер Г. Сканирующая тунельная микроскопия — от рождения до юности. Нобелевська лекція // Успехи физических наук. том 154. вып. 2. 1988. С. 261–277.
3. Дуда Р., Харт П. Распознавание образов и анализ сцен. //Москва: Мир, 1976. 511 с.
4. Даджион Д., Мерсеро Р. Цифровая обработка многомерных сигналов. // Москва: Мир, 1988. 488 с.
5. Щербина В.М., Мацулевич О.Є. Спосіб побудови дотичних у вузлах спиралеподібних дискретно представлених кривих із використанням спеціальної функції // Праці Таврійського державного агротехнологічного університету імені Дмитра Моторного. Мелітополь: ТДАТУ імені Дмитра Моторного, 2019. Вип. 19, т. 2. С. 278-293
6. Вершков О.О., Мацулевич Ю.О. Визначення шорсткості поверхонь із застосуванням програмного забезпечення COPYCAD ф. DELCAM plc // Сучасні комп'ютерні та інформаційні системи і технології, Матеріали і всеукраїнської науково-практичної інтернет-конференції, Мелітополь 7-25 грудня 2020р. С. 17-23
7. Мацулевич О.Є., Вершков О.О., Холодняк Ю.В., Дмитрієв Ю.О., Чаплінський А.П. Розробка мурашиного алгоритму для оптимізації оперативного планування робіт по збиранню врожаю кісточкових. *Плодовий сад – новітнє в теорії та практиці*: матеріали VВсеукр. наук.-практ. інтернет-конференції. Мелітополь: ТДАТУ, 2021. С. 106-110.

УДК 004.942

ПРОГРАМНА РЕАЛІЗАЦІЯ АЛГОРИТМУ ФОРМУВАННЯ МОНОТОННИХ КРИВИХ

Гавриленко Є.А., д.т.н.

e-mail: yevhen.havrylenko@tsatu.edu.ua

Холодняк Ю.В., к.т.н.

e-mail: yuliya.kholodnyak@tsatu.edu.ua

Мірошніченко М.Ю., к.т.н.

e-mail: mykola.miroshnychenko@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Конструювання поверхонь з підвищеними динамічними властивостями вимагає розробки методів геометричного моделювання, які забезпечують задану точність, контроль диференціально-геометричних характеристик, відсутність осциляцій.

При оцінюванні функціональних якостей виробу динамічні властивості поверхні визначаються характеристиками потоку середовища, що виникає у пограничному шарі уздовж поверхні. Основною вимогою є закономірний, стійкий, регламентований характер обтікання. З геометричної точки зору підвищені динамічні властивості забезпечують поверхні із закономірною, плавною зміною диференціально-геометричних характеристик.

Підвищені динамічні властивості необхідні поверхням, які обмежують корпусні вироби авіа-, автомобіле-, судобудівництва, лопати турбін, змішувачі, канали двигунів внутрішнього згорання, трубопроводи, робочі органи сільськогосподарських машин. Постійне підвищення вимог до якості моделювання таких поверхонь обумовлено збільшенням швидкостей взаємодії із середовищем.

Заданий характер зміни диференціально-геометричних характеристик поверхні можна забезпечити за рахунок характеристик кривих, що входять у визначник поверхні. До таких характеристик належить другий порядок гладкості та монотонна зміна радіусів кривини уздовж обводу.

Одновимірні обводи можна моделювати по заданим умовам методами неперервного (НГМ) та дискретного геометричного моделювання (ДГМ). Методи НГМ передбачають отримання на основі точкового ряду обводу, що складається із ділянок алгебраїчних кривих, які стикаються у вихідних точках із заданим порядком гладкості. Вихідні дані (координати точок, положення дотичних та значення радіусів кривини обводу у цих точках) визначають ділянку кривої. При цьому забезпечити бажаний характер зміни характеристик кривої всередині ділянки складно або неможливо.

Більш широкі можливості для моделювання обводів по заданим умовам надають методи ДГМ, що враховують внутрішню геометрію кривої та дозволяють накладати на розв'язок будь-яку кількість додаткових умов. Однак ДГМ як напрям прикладної геометрії виник відносно нещодавно та методи, які забезпечують монотонну зміну кривини розвинені недостатньо.

Серед методів НГМ, які забезпечують другий порядок гладкості, найбільше застосування отримало формування обводів на основі В-сплайнів. В-сплайн визначається контрольними токами, кожний із яких відповідає функція сполучення. Дискретний характер вихідних даних забезпечує гнучкість управління формою кривої. Порядок гладкості обводу забезпечується ступенем функцій сполучення. При збільшенні порядку гладкості знижується можливість

локального корегування форми кривої. Одночасно зростає ймовірність виникнення осциляції. Ці особливості обмежують можливість забезпечення заданих характеристик обводів, що формуються на основі В-сплайнів.

Серед методів ДГМ найбільш широкі можливості локального управління формою кривої при забезпеченні граничних умов надають методи варіативного дискретного геометричного моделювання (ВДГМ). Основною властивістю методів ВДГМ є вибір розв'язку всередині деякого діапазону можливих, за умовами задачі, значень. Наявність діапазону дає можливість локального моделювання та корегування розв'язку, дозволяє враховувати додаткові умови задачі. Обмеженість діапазону розв'язку дозволяє контролювати відсутність осциляції та забезпечувати необхідні вимоги до характеристик та гладкості обводу. Особливістю методів є багатократне повторювання розрахункових алгоритмів (послідовне згущення), яке призводить до заміни із заданою точністю вихідного геометричного образу супровідною ламаною лінією.

Методика, яка дозволяє визначити можливість забезпечення монотонної зміни кривини (зростання або убуття) уздовж дискретно представленої кривої (ДПК) розроблена в [1]. Вихідними даними для проведення аналізу є положення вузлів вихідного точкового ряду.

Геометрична схема формування ДПК з закономірною зміною кривини запропонована у роботі [2]. ДПК формується на основі наперед сформованої дискретно представленої еволюти (ДПЕ). ДПЕ визначається положеннями нормалей і центрів кривини, призначених у вихідних вузлах, та умовами, що на неї накладаються.

Задача визначення області можливого розташування нормалей (n_i) та центрів кривини (C_i) у вихідних точках ДПК розв'язується у роботі [2]. Областю розташування нормалей є сектор, що обмежений граничними, за умовами задачі, положеннями. Діапазоном розташування центрів кривини є відрізок, що належить нормалі. Після призначення положень нормалей та центрів кривини уздовж вихідного точкового ряду отримується ланцюг базисних трикутників, який обмежений нормальми у вихідних точках та хордами, що з'єднують центри кривини, призначені на цих нормальх.

ДПЕ формується на основі ланцюга базисних трикутників за методикою, розробленою в [3-5]. Вихідними даними для формування еволюти є координати вузлів, положення дотичних до ДПЕ (нормалей до ДПК), порядок гладкості кривої та її довжина. Область розташування центра кривини, що відповідає точці згущення, визначається на нормалі згущення, положення якої призначається паралельно відрізку, що з'єднує два послідовних центра кривини ДПК. Діапазон розташування точки згущення визначається як відрізок на нормалі згущення. Довжина відрізка залежить від положення нормалі.

При моделюванні монотонної кривої на основі еволюти необхідно забезпечити контроль наступних умов: порядок гладкості кривої, відсутність осциляції, задану довжину еволюти. Крім того для досягнення оптимального розв'язку задачі необхідно забезпечити плавну зміну характеристик уздовж обводу. Аналітично забезпечити виконання вказаних вимог складно або неможливо. Тому розв'язання поставленої задачі доцільно виконувати із застосуванням спеціалізованого програмного забезпечення.

Метою дослідження є програмна реалізація алгоритму, що дозволяє сформувати на основі вихідного точкового ряду криву із монотонною зміною диференціально-геометричних характеристик.

Основні матеріали дослідження. Монотонна крива, яка задана координатами вузлів, моделюється на основі алгоритму, який передбачає попереднє формування еволюти.

Еволюта монотонної кривої має відповідати наступним вимогам:

- еволюта є випуклою кривою;
- нормалі до кривої є дотичними до еволюти, що її визначає;
- довжина еволюти дорівнює різниці радіусів кривини у точках, що обмежують відповідну ділянку кривої.

Нехай вихідний точковий ряд дозволяє забезпечити монотонне зростання радіусів кривини уздовж ДПК [3]. Еволюта ДПК моделюється як випуклий обвід першого порядку гладкості, що не містить особливих точок. Виконання вказаних умов забезпечує другий порядок гладкості ДПК при монотонній зміні радіусів кривини уздовж кривої.

Положення нормалей у вихідних точках $\dots n_{i-1}, n_i, n_{i+1} \dots$ призначаються всередині відповідних діапазонів їх можливого розташування. В результаті отримуємо ламану лінію $\dots T_{i-1}, T_i \dots$, яка визначає ДПЕ в першому наближенні (рис. 1). Центри кривини $\dots C_{i-1}, C_i, C_{i+1} \dots$ є точками, в яких еволюта торкається вказаної ламаної лінії. Положення центрів кривини визначаються всередині відповідних ланок ламаної лінії (C_i належить відрізку $[T_{i-1}, T_i]$). В результаті послідовних згущень кількість ланок вказаної ламаної лінії збільшується та в граничному положенні отримаємо обвідну лінію нормалей, яка є еволютою кривої.

Для того, щоб сформувати ДПЕ, яка визначає монотонну криву, що відповідає умовам задачі, необхідно розв'язати наступне:

- визначити положення нормалей та центрів кривини у вихідних точках ДПК та отримати ланцюг базисних трикутників, на основі якого можливо сформувати обвід першого порядку гладкості заданої довжини;

- всередині кожного базисного трикутника сформувати ділянку еволюти C_i, C_{i+1} , яка дотична до n_i та n_{i+1} в точках C_i та C_{i+1} . Довжина ділянки еволюти має дорівнювати $d_i = |i+1; C_{i+1}| - |i; C_i|$ (рис. 1).

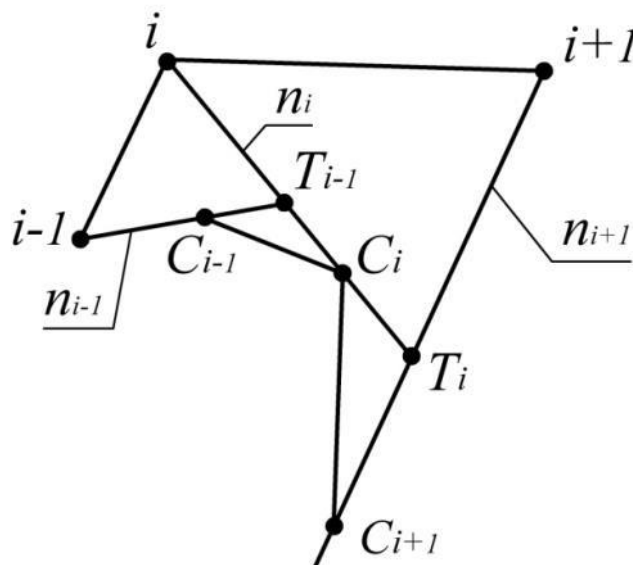


Рисунок 1. Вихідний ланцюг базисних трикутників

Для розв'язання поставлених задач розроблено спеціалізоване програмне забезпечення, що реалізує алгоритми визначення положень центрів кривини у вихідних точках ДПК та центрів кривини, які відповідають точкам згущення. Розглянемо алгоритм визначення положень центрів кривини ДПК у вихідних точках.

Попередньо ДПЕ формується як ламана лінія $A_{i-1}^2, A_i^1, A_i^2, A_{i+1}^1$ (рис. 2, а) таким чином, щоб виконувались наступні умови:

$$|i+1; A_{i+1}^1| = |A_i^2; A_{i+1}^1| + |A_i^2; i|, \quad (1)$$

$$|A_{i-1}^2; T_{i-1}| = |T_{i-1}; A_i^1|, \quad |A_i^2; T_i| = |T_i; A_{i+1}^1|. \quad (2)$$

Отримана ДПЕ визначає евольвенту як коробову лінію кіл, радіуси яких монотонно зростають уздовж обводу. Виконання умов (1) та (2) однозначно визначає розташування точок $A_{i-1}^2, A_i^1, A_i^2, A_{i+1}^1$ при зафіксованих положеннях нормалей n_{i-1}, n_i, n_{i+1} .

Необхідною умовою існування розв'язку задачі є розташування нормалей, при якому виконується умова $|T_{i-1}; T_i| > |T_{i-1}; A_i^1| + |A_i^2; T_i|$. В іншому випадку (рис. 2. б) задача не має розв'язку.

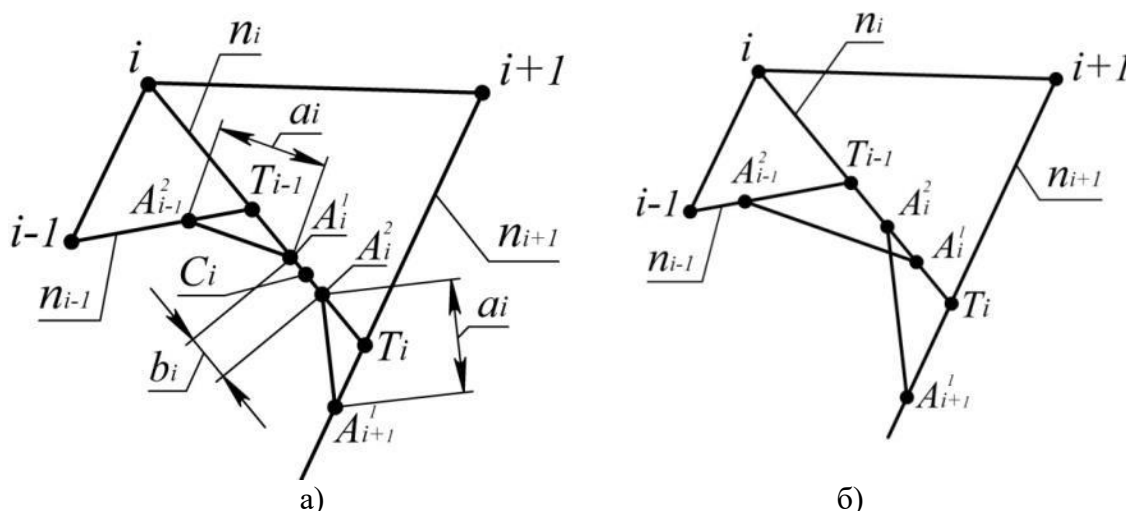


Рисунок 2. Визначення положень центрів кривини у вихідних точках

Призначення центра кривини C_i всередині відрізка $[A_i^1; A_i^2]$ є необхідною умовою формування еволюти як обводу першого порядку гладкості заданої довжини.

Виконання вказаних умов контролюється за допомогою наступного коефіцієнту:

$$e_i = \frac{2b_i}{a_{i-1} + a_i}, \quad (3)$$

де $a_{i-1} = |A_{i-1}^2; A_i^1|$, $a_i = |A_i^2; A_{i+1}^1|$ та $b_i = |A_i^1; A_i^2|$ – довжини ланок ламаної лінії, яка визначає ДПЕ.

Обов'язковою умовою існування розв'язку є додатне значення введеного коефіцієнту, тобто $e_i > 0$. Крім того за допомогою контролю значень e_i можна забезпечити оптимальний розв'язок задачі. Оптимальним розв'язком будемо вважати таку ДПЕ, яка забезпечує плавність зміни радіусів кривини уздовж

евольвенти. Область оптимального розв'язку визначається виконанням умови $e_i = 1 \pm \varepsilon$, де ε – задане значення відхилення від оптимального значення.

Досягнення оптимальних значень e_i уздовж усього точкового ряду виконується в автоматичному режимі. Для цього виконуються наступні дії.

1) Уздовж точкового ряду призначаються положення n_i по центру діапазонів їх можливого розташування.

2) Розраховуються значення e_i та визначається $e_i^{\max}$ – максимальне значення e_i із отриманих.

3) Положення n_i , яке відповідає $e_i^{\max}$, корегується з метою забезпечення його наближення до оптимального. Положення n_i змінюється із заданим кроком з контролем значення e_i на кожному кроці.

4) При зміні положення n_i змінюються значення e_{i-1} , e_i та e_{i+1} . На кожному кроці перераховані значення e_{i-1} , e_i та e_{i+1} порівнюються із значеннями e_i на інших ділянках. Якщо e_i стає максимальним на іншій ділянці, то корегується положення відповідної нормалі.

Після забезпечення на всіх ділянках розташування значень коефіцієнту e_i всередині області оптимального розв'язку положення n_i сформовано. Положення C_i призначаються по центру відповідних відрізків b_i .

Таким чином отримано положення нормалей та центрів кривини у вихідних точках ДПК, які визначають ланцюг базисних трикутників. Отриманий ланцюг базисних трикутників визначає вихідну ДПЕ як монотонну криву лінію.

На наступному етапі всередині кожного базисного трикутника формується ділянка еволюти кривої у вигляді точкового ряду, що складається із будь-якої кількості центрів кривини ДПК. Розглянемо алгоритм визначення положення центрів кривини, які відповідають точкам згущення (C_{32}) та самих точок згущення (i_{32}).

Положення нормалі згущення (n_{32}) призначається паралельно стороні базисного трикутника $[C_i; C_{i+1}]$ на відстані h від неї (рис. 3). Точки перетину n_{32} з нормальми n_i та n_{i+1} позначимо як T_1 та T_2 відповідно.

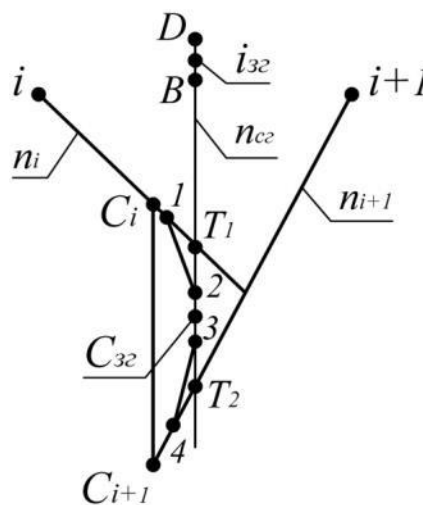


Рисунок 3. Формування ділянки еволюти ДПК

На нормалі n_{32} визначається діапазон можливого розташування i_{32} – відрізок $[B; D]$. Положення точок B та D визначається згідно з наступними співвідношеннями:

$$[B; T_1] = [T_1; i], \quad (4)$$

$$[D; T_2] = [T_2; i + 1]. \quad (5)$$

При переміщенні n_{32} довжина відрізка $[B; D]$ змінюється. Якщо n_{32} займає граничні із можливих за умовами задачі положення, то точки B та D співпадають. Діапазон можливого розташування n_{32} визначається виконанням умови $[T_1; D] > [T_1; B]$.

ДПЕ формується як ламана лінія $C_i, 1, 2, 3, C_{i+1}$ таким чином, щоб виконувались наступні вимоги:

$$|i_{32}; 2| = |i; 1| + |1; 2|, \quad |i + 1; 4| = |4; 3| + |3; i_{32}| \quad (6)$$

$$|1; T_1| = |T_1; 2|, \quad |3; T_2| = |T_2; 4|. \quad (7)$$

Виконання вимог (6) та (7) однозначно визначає розташування точок 1, 2, 3, 4 при зафіксованих положеннях нормалей. Призначення центра кривини, який відповідає i_{32} , всередині відрізка $[2; 3]$ є необхідною умовою формування еволюти як обводу першого порядку гладкості заданої довжини. Отримана ділянка ДПЕ визначає евольвенту як коробову лінію кіл, радіуси яких монотонно зростають уздовж обводу.

В результаті послідовних згущень кількість ланок ДПЕ збільшується, в результаті чого отримується точковий ряд, який із заданою точністю представляє еволюту кривої. Точність представлення кривої на кожному кроці згущення визначається довжиною відрізка $\delta = |B; D|$.

Попередні положення n_{32} та i_{32} визначаються по центру відповідних діапазонів. В процесі згущення існування розв'язку задачі та плавність зміни характеристик уздовж ділянки кривої контролюється за допомогою коефіцієнта, аналогічного (3).

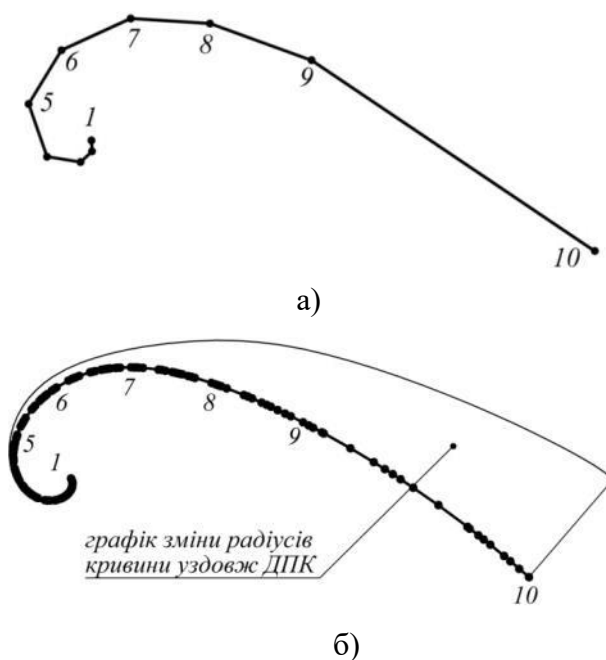


Рисунок 4. Контрольний приклад

Розглянемо контрольний приклад. Вихідний точковий ряд складається із 10 точок (рис. 4, а). Координати точок призначені довільно. Контроль можливості моделювання монотонної кривої виконано згідно зі способом, розробленим в [3].

Призначено точність представлення кривої $\delta = 10^{-5}$. В результаті згущення отримано новий точковий ряд, який складається із 172 точки (рис. 4, б). Максимальна абсолютна похибка визначення точки згущення дорівнює $\delta = 9,8 \cdot 10^{-6}$. Розрахункові дані, отримані на вихідній ділянці (6...7) наведені у таблиці 1 (R_i – радіуси кривини у точках i , отримані у процесі моделювання ДПК).

Таблиця 1. Дані, отримані на ділянці 6...7 після згущення

i	R_i (мм)	δ_i (мм)	i	R_i (мм)	δ_i (мм)
6.0	57.30	$8.1 \cdot 10^{-6}$	6.8	62.12	$8.9 \cdot 10^{-6}$
6.1	58.31	$7.6 \cdot 10^{-6}$	6.9	62.62	$8.6 \cdot 10^{-6}$
6.2	58.57	$6.3 \cdot 10^{-6}$	6.10	62.68	$7.5 \cdot 10^{-6}$
6.3	59.29	$6.8 \cdot 10^{-6}$	6.11	63.17	$8.3 \cdot 10^{-6}$
6.4	59.61	$8.2 \cdot 10^{-6}$	6.12	64.00	$6.6 \cdot 10^{-6}$
6.5	59.64	$8.1 \cdot 10^{-6}$	6.13	64.37	$7.1 \cdot 10^{-6}$
6.6	60.06	$7.7 \cdot 10^{-6}$	6.14	65.18	$9.4 \cdot 10^{-6}$
6.7	60.98	$6.8 \cdot 10^{-6}$	6.15	66.75	-

Висновки. У роботі запропоновано алгоритм та розроблено програмне забезпечення для моделювання кривої другого порядку гладкості із монотонною зміною диференціально-геометричних характеристик. Граничними умовами при моделюванні кривої є призначені положення нормалей та центрів кривини вихідної ДПК. Крива моделюється на основі наперед сформованої еволюти. Запропоновано критерії оптимальності розв'язку задачі. У процесі подальших досліджень планується розробити алгоритми та програмне забезпечення для моделювання ДПК на ділянках, які містять особливі точки (точки зміни зростання та убуття радіусів кривини, точки зміни опуклості та увігнутості кривої). Це дасть можливість моделювати криву із закономірною зміною диференціально-геометричних характеристик на основі будь-якого точкового ряду.

Список використаних джерел:

1. Гавриленко Е. А., Холодняк Ю. В., Антонова Г. В., Михайленко Е. Ю. Моделирование участков одномерных обводов по заданным условиям. *Прикладна геометрія та інженерна графіка*. К.: КНУБА, 2021. Вип. 101. С. 34-44.
2. Kholodniak Yu., Havrylenko Ye., Ivzhenko O., Pykhtieieva I. *Modeling surfaces according to specified conditions based on array of points*. Modern modeling problems. Melitopol: MSPU, 2021. Vol. 24. P. 175-182.
3. Гавриленко Е.А., Холодняк Ю.В., Найдыш А.В., Лебедев В.А. Создание САД-моделей поверхностей с использованием специализированного программного обеспечения. *Прикладні питання математичного моделювання*. Херсон: ХНТУ, 2020. Т. 3, № 2.2. С. 66-75.
4. Гавриленко Е.А., Холодняк Ю.В., Антонова Г.В., Чаплинский А.П. Разработка алгоритма программного обеспечения для формирования обводов по заданным геометрическим условиям. *Праці Таврійського державного агротехнологічного університету*. Мелітополь: ТДАТУ, 2020. Вип. 20, т. 3. С. 293-303
5. Холодняк Ю.В., Гавриленко Е.А., Ивженко А.В., Найдыш А.В. Моделирование участка пространственной монотонной кривой линии. Сучасні проблеми моделювання: наукове фахове видання. Мелітополь: МДПУ, 2020. Вип.17. С. 131-137.

УДК 515.2:528.71

ДОСЛІДЖЕННЯ СПОСОБІВ ВІЗУАЛІЗАЦІЇ ТРИВИМІРНИХ СЦЕН У РЕАЛЬНОМУ ЧАСІ

Мірошніченко М.Ю., к.т.н.

e-mail: mykola.miroshnychenko@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Візуалізація - це завершальний і, безумовно, найвідповідальніший етап створення тривимірної сцени. Редактор тривимірної графіки прораховує зображення, враховуючи геометрію об'єктів, властивості матеріалів, з яких вони виготовлені, розташування та параметри джерел світла тощо. Невдало виконана візуалізація може звести нанівець усі багатоденні зусилля щодо моделювання, освітлення та текстурювання сцени. Створення сцени з реалістичним освітленням — це ще одне завдання, яке вирішують для того, щоб надати кінцевому зображенню більшої реалістичності.

У реальному світі світлові промені багаторазово відбиваються і переломлюються в об'єктах, внаслідок чого тіні, що відкидаються об'єктами, здебільшого мають нечіткі, розмиті межі. За якість відображення тіней, в основному відповідає апарат візуалізації. До тіней, що відкидають у сцені, пред'являються окремі вимоги. Тінь, що відкидається від об'єкта, може сказати багато про що — як високо він знаходиться над землею, яка структура поверхні, на яку падає тінь, яким джерелом освітлений об'єкт і т.д. Крім цього, тінь може підкреслити контраст між переднім та заднім планом, а також «видати» об'єкт, який не потрапив у поле зору об'єктива віртуальної камери. І тут глядачеві дається можливість самому домислити навколишнє оточення сцени.

Основна проблема побудови тіньового проектування полягає в тому, що апаратна частина для опису поверхні використовує в якості універсальної одиниці трикутники. Трикутники є елементарною поверхнею, для побудови якої використовується мінімальне число вершин. Описувані алгоритми побудови поверхонь апаратною частиною виділяються у загальну групу триангулятивних (що використовують у своїй основі універсальні одиниці трикутники). Сучасні графічні прискорювачі працюють на рівні не вище за триангулятивні алгоритми і нічого не знають про наявність сцени, тому тіні ними безпосередньо не підтримуються і вся підготовча робота лягає на плечі програміста. Проте в найпоширеніших алгоритмах практично всі стадії побудови тіні реалізуються саме через графічний прискорювач. Крім того, за останні роки були внесені значні зміни як до обсягу, так і до складових динамічних бібліотек, OpenGL, Direct3D тощо. у напрямку опису тіні. Поліпшення алгоритмів побудови тіней сприяли високим результатам реалістичності тривимірної візуалізації. Поява таких функцій і понять, як Decay, ShadowBias, ShadowType, Near/Far Attenuation і т.д., свідчать про збільшення гнучкості в керуванні тінню, що відкидається об'єктами.

Метою даної статті є дослідження питань вибору алгоритму побудови тіні, що є важливим етапом тривимірної візуалізації сцени.

Основні матеріали дослідження. Тіні бувають чіткі (*hard shadows*) та м'які (*soft shadows*). Точні тіні виходять, коли є точкове джерело або джерело спрямованого паралельного світла. Згідно з геометричною оптикою Френеля, у разі наявності протяжного джерела світла, від об'єкта виходить не одна тінь, а ціла серія тіней, які накладаються один на одного і утворюють більш-менш

затемнені області, які становлять основну тінь (*umbra*) і область півтіні (*penumbra*). Усі існуючі алгоритми побудови тіней дозволяють будувати чіткі тіні, але з них дозволяють розмити отриману тінь, створюючи псевдо-м'яку тінь.

Існує кілька основних підходів до побудови тіней:

- Перетворення моделі «на землю» та малювання її як тіні.
- Побудова тіньової маски об'єкта та проєктивне накладення її на інші об'єкти.
- Тіньові обсяги.
- Використання інформації про глибину (*depth buffer*).

1. *Просціювання (перетворення "на землю")*. Фактично це перший алгоритм побудови тіні, який був застосований в іграх. Він відрізняється простотою реалізації та гарною якістю одержуваної тіні. Цей алгоритм був вперше описаний Д. Бліном [1], який описав рівняння для проєктування полігону «на землю», тобто. на площину $z = 0$, у напрямку джерела світла.

Розглянемо два випадки:

- 1) Джерело на нескінченності.
- 2) Локальне джерело (точкове джерело неподалік об'єкта).

При цьому використовується геометричне взаємовідношення джерела світла та полігону для обчислення проєкції кожного полігону моделі «на землю».

«Тіньові полігони» мають бути розраховані кожному із джерел світла, тобто, якщо об'єкт висвітлюється N джерелами світла, необхідно розрахувати N його «тіньових проєкцій».

Розглянемо випадок, коли джерело світла розташовано на нескінченності відносно до об'єкта. У разі нескінченно віддаленого джерела світла ми припускаємо, що промені світла, що приходять до об'єкта, є паралельними (рис. 1). Це дозволить нам розв'язати рівняння проєкції лише раз і застосовувати отримане рішення до всіх вершин об'єкта.

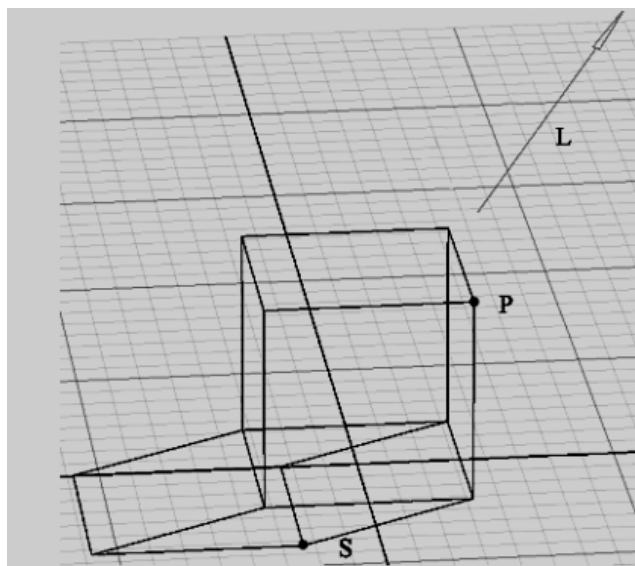


Рисунок 1. Схема розташування об'єкта та джерела світла при нескінченній відстані між ними

Сформулюємо загальну постановку задачі: маючи координати точки джерела світла $(x_l; y_l; z_l)$ і координати вершини об'єкта $(x_p; y_p; z_p)$ необхідно

отримати координати проекції вершини об'єкта на площину $z = 0$, тобто точку тіні $(x_s; y_s; z_s)$.

З подібних трикутників отримуємо трикутників (рис. 1) отримуємо:

$$\frac{x_p - x_s}{z_p - z_s} = \frac{x_l - x_p}{z_l - z_p} \quad (1)$$

Розв'язавши це рівняння щодо x_s , отримуємо:

$$x_s = x_p - (z_p - z_s) \cdot \left(\frac{x_l - x_p}{z_l - z_p} \right) \quad (2)$$

Якщо прийняти, що L це вектор із точки P до джерела світла, то точку S можна висловити як

$$S = P - \alpha \cdot L. \quad (3)$$

Оскільки ми проектуємо на площину $z = 0$, то рівняння (3) можна переписати у такому вигляді:

$$0 = z_p - \alpha \cdot z_l \quad \text{або} \quad \alpha = \frac{z_p}{z_l}. \quad (4)$$

Розв'язавши рівняння (3) відносно x_s и y_s отримаємо

$$x_s = x_p - \frac{z_p}{z_l} \cdot x_l \quad \text{та} \quad y_s = y_p - \frac{y_p}{y_l} \cdot y_l. \quad (5)$$

або у матричному вигляді

$$M_s = \begin{vmatrix} 1 & 0 & -x_l/z_l & 0 \\ 0 & 1 & -y_l/z_l & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}. \quad (6)$$

Тепер, маючи координати точки P у світовому координатному просторі, можна отримати її проекцію на площину $z = 0$ просто шляхом множення на матрицю M_s

$$S = M_s \cdot P. \quad (7)$$

Рівняння (5) для віддаленого джерела світла може бути узагальнено для випадку, коли джерело світла знаходиться на кінцевій відстані від об'єкта (локальне джерело). І тут нам знадобляться додаткові обчислення для кожної вершину, так як кожна вершина має власний напрям на джерело світла. Тим не менш, у цьому випадку ми теж можемо перенести більшу частину обчислень у

матрицю M_s . Якщо L — точка розташування джерела світла, то рівняння (3) набуває вигляду:

$$S = P + \alpha \cdot (P - L). \quad (8)$$

Необхідно зробити проекцію на площину

$$\alpha = \frac{-z_p}{z_p - z_l}. \quad (9)$$

Отримуємо координати

$$x_s = \frac{x_l z_p - x_p z_l}{z_p - z_l} \quad \text{та} \quad y_s = \frac{y_l z_p - y_p z_l}{z_p - z_l}. \quad (10)$$

Якщо застосувати гомогенізацію після перетворення, то (10) можна записати у вигляді матриці

$$M_{sh} = \begin{vmatrix} -z_l & 0 & x_l & 0 \\ 0 & z_l & 0 & 0 \\ 0 & 0 & y_l & 0 \\ 0 & 0 & 1 & -z_l \end{vmatrix}. \quad (11)$$

Тоді

$$S_h = M_{sh} \cdot P. \quad (12)$$

Після цього після чого необхідно провести гомогенізацію точки S_h для получения проекции точки P на площину $z = 0$.

Побудова тіньової маски та проєктивне накладання. Цей метод є логічним продовженням попереднього, але, на відміну від нього, має цілу низку переваг. Об'єкт, який відкидає тінь (*shadow caster*) у той чи інший спосіб будується з погляду джерела світла чорним кольором в білу текстуру. Розмір текстури залежить від того, наскільки дрібні елементи об'єкта ми хочемо бачити в тіні. Уявімо собі площину, перпендикулярну напрямку на джерело світла і розташовану відразу за об'єктом по променю світла - це і є текстура, в яку проводиться побудова об'єкта. Щоб отримати проєкцію об'єкта на цю текстуру, застосуємо елементи стандартного конвеєра трансформації.

Припустимо, що текстура, яку нам необхідно побудувати об'єкт, це екран. Тоді послідовність перетворень стає просто очевидною (стандартний конвеєр перетворення): *простір об'єкта* => *світовий простір* => *простір камери* (у нас це джерело світла) => *проєктивний простір* => *нормалізований проєктивний простір* => *екранний простір*. Для наших цілей необхідно модифікувати матриці переходів у простір камери, проєктивний простір та екранний простір.

Простір камери — це практично звичайна матриця камери, але тільки розташована в точці знаходження джерела світла і спрямована на *shadow caster*. У разі нескінченно віддаленого джерела світла віртуальна камера може розташовуватися в будь-якій точці на прямій між *shadow caster* та джерелом світла (зазвичай її вибирають близько до *shadow caster*).

Проектний простір. Ця матриця суттєво залежить від того, чи джерело світла є локальним або нескінченно віддаленим. Для нескінченно віддаленого джерела світла ця матриця має вигляд:

$$M_{pj} = \begin{vmatrix} \frac{2}{width} & 0 & 0 & 0 \\ 0 & \frac{2}{height} & 0 & 0 \\ 0 & 0 & -\frac{1}{depth} & 0 \\ 0 & 0 & 1 & 1 \end{vmatrix}. \quad (13)$$

де *width*, *height* та *depth* – ширина, висота та глибина проєктивного об'єма. В цьому випадку не потрібна гомогенізація.

Для локального джерела світла потрібна матриця перспективного перетворення

$$M_{pj} = \begin{vmatrix} \cos\left(\frac{\varphi}{2}\right) & 0 & 0 & 0 \\ 0 & aspect \cdot \cos\left(\frac{\varphi}{2}\right) & 0 & 0 \\ 0 & 0 & \frac{\sin\left(\frac{\varphi}{2}\right) \cdot far}{far - near} & \sin\left(\frac{\varphi}{2}\right) \\ 0 & 0 & -\frac{\sin\left(\frac{\varphi}{2}\right) \cdot far \cdot near}{far - near} & 0 \end{vmatrix},$$

де φ – кут зору, *aspect* - відношення ширини до висоти екрану (при побудові в квадратну текстуру *aspect* = 1), *far* і *near* - дальня та ближня стінки проєктивного об'єму відповідно. Після перспективного перетворення потрібна гомогенізація, як і у разі локального джерела при перетворенні «на землю».

Екранний простір. Модель із нормалізованого проєктивного простору має лягти у квадратну текстуру з мінімальними зазорами. Після того, як *shadow caster* промальований у текстурі, її необхідно накласти на затінені об'єкти (*shadow receiver*). Для цього можна скористатися технологією проектування текстур. Щоб уникнути повторення тіньової текстури, необхідно виставити *clamp* або *border color*. В результаті ми отримуємо тінь від одного об'єкта (*shadow caster*) на іншому (*shadow receiver*). Далі можна зробити з цією тінню:

1. Розмити (отримати псевдо-м'яку тінь). Зробити це можна кількома способами:

- «Вручну» шляхом накладання якогось *blur*-фільтра. Такий підхід зручний при програмному малюванні самої тіньової текстури.

- Використовувати "залізо". Можна відмалювати текстуру саму на себе зі зміщенням, або згенерувати *mip-level*.

2. Вводити тінь у прозорість на основі відстані до джерела світла. На жаль, реалізувати це без *vertex shader* за допомогою апаратної частини неможливо.

Єдине, що можна зробити для поліпшення виду тіні, так це зробити її не вугільно-чорною, а рівномірно напівпрозорою.

3. Мінімізувати витрати на побудову тіні. Це досягається при використанні технології LOD.

Тіньові об'єми. При освітленні сцени в тіні виявляються об'єкти, які потрапляють всередину тіньового об'єму [2]. Для того, щоб точка була затінена, промінь світла повинен увійти в тіньові обсяги більше разів, ніж вийти з них. Для спрощення завдання можна прийняти, що не промінь світла від джерела повинен пройти цей шлях, а промінь від спостерігача. За такого спрощення виникає кілька неприємних випадків, які розглянемо окремо.

Алгоритм можна розділити на дві явні дії: побудова «тіньової маски» (маски освітлених та затінених областей 2-мірної картинки) та відтворення сцени з використанням тіньової маски. Пропонується реалізація, яку можна розбити на три стадії.

Перша стадія. На цій стадії просто малюємо всю сцену таким чином, ніби вона затінена. Усі наступні стадії служать додавання освітлення в картинку. Ділянки картинки, які так і не будуть освітлені у наступних проходах, залишаються затіненими.

Друга стадія. На цій стадії будуємо тіньову маску, використовуючи інформацію про тривимірні тіньові об'єми. Для створення та зберігання тіньової маски нам знадобиться буфер шаблонів (*stencil buffer*). Побудова оптимальних тіньових об'ємів є досить складним завданням, яке виходить за рамки даної роботи. Опис одного з алгоритмів побудови силуету, необхідний для побудови оптимального тіньового обсягу, можна знайти, наприклад, в [3]. Отже, нам необхідно підрахувати для кожної точки різницю того, скільки разів промінь від спостерігача перетнув позитивно та негативно орієнтовані полігони тіньового об'єму. Це робиться за два проходи. На першому проході будуємо всі тіньові об'єми з нормальним (для нашого застосування) відкиданням тіні поверхні, алгоритм *back face culling*. У процесі виконання алгоритмів тінеутворення, наприклад, *z-buffer* для прискорення і спрощення прорахунку модель зберігається в триангулятивному вигляді. Тому часто йдеться про відкидання тіні набору трикутних поверхонь. Кожна точка, яка проходить тест по глибині, збільшує значення в буфері шаблонів на 1. На другому проході міняємо *back face culling* на протилежний – малюватимуться задні стінки тіньових об'ємів. Тепер кожна точка, яка проходить тест по глибині, зменшуватиме значення в буфері шаблонів на 1. У результаті після цих двох проходів ми маємо в буфері шаблонів нульові значення для освітлених областей і значення більше за нуль для затінених.

Третя стадія. На цій стадії вже проведено всі попередні приготування - є і затінена сцена, і тіньова маска. Залишається відмалювати ще раз всю сцену освітлення, використовуючи перевірку по буферу шаблонів.

Використання інформації про глибину. Метод базується на ідеї про те, що затінені точки – це ті, що «приховані» від джерела світла. Інакше кажучи, це «*hidden surfaces*» з погляду джерела світла. Щоб визначити невидимі поверхні з погляду джерела світла, необхідно відмалювати всю сцену з погляду джерела світла в буфер глибини. Далі необхідно відмалювати всю сцену з погляду спостерігача, проробляючи таку перевірку: якщо глибина точки, переведена в систему координат джерела світла, більше значення, яке ми зберегли з буфера глибини попередньої стадії, отже, точка затінена. В іншому випадку точка освітлена.

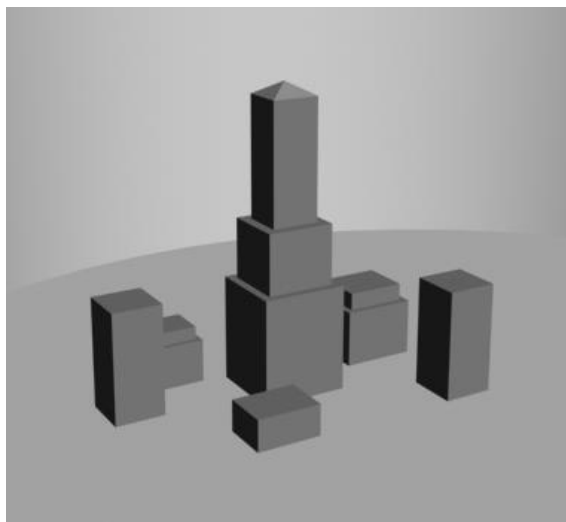


Рисунок 2. Сцена, побудована з погляду джерел світла

Якщо «залізо» не підтримує специфічних функцій (збереження та доступу до буфера глибини, порівняння глибини зі збереженим значенням тощо), то реалізувати його на пряму практично неможливо. Але через те, що метод дуже привабливий, необхідно «обійти» апаратну частину. Припустимо, що апаратна частина не дає доступу до буфера глибини, у нього немає жодних можливостей порівнювати значення, крім порівняння по альфа-каналу. Єдине, що потрібно від апаратної частини, так це можливість зміни колірної буфера для поточного відмальовування (*render target*) та 2-текстурний растеризатор. На рис. 2. зображена сцена з погляду джерел світла, але на рис. 3. - сцена з погляду спостерігача з побудованими тінями.

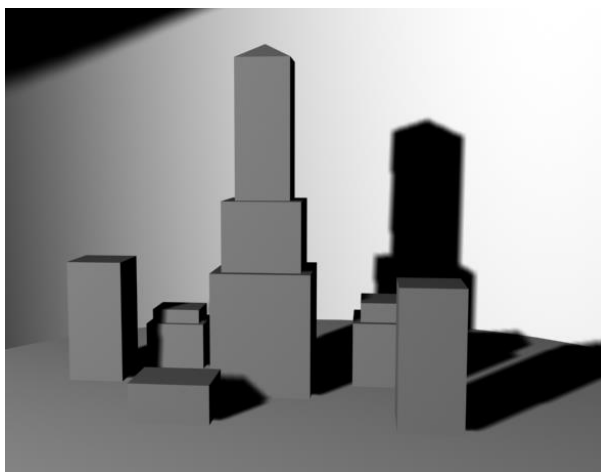


Рисунок 3. Сцена з побудованими тінями, зображена з боку спостерігача

Як і в попередньому методі, алгоритм можна розділити на дві явні дії: побудова тіньової маски та відтворення сцени з використанням тіньової маски.

Для отримання глибини ми скористаємося текстурою 256x1x32, в якій значення розміщені від 0 до 255 в альфа-каналі (колірні значення нас не цікавлять). Щоб текстура лягла необхідним чином, згенеруємо текстурні координати з подальшою трансформацією

$$S_h = (M_{pj} * M_{sel}) \cdot P_{cam}. \quad (14)$$

де M_{pj} – це матриця проєктивного перетворення, а M_{sel} – матриця вибору

$$M_{sel} = \begin{vmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{vmatrix}. \quad (15)$$

Після такої побудови, у текстурі ми маємо образ глибини сцени з дискретом 1/256. Далі створюємо тіньову маску, базуючись на порівнянні глибини. Для цього ми накладемо першу текстуру (256x1x32) на об'єкт способом, подібним до першої стадії, і накладемо другу текстуру, отриману на першій стадії, проєктивним методом. Після чого, у кожній точці віднімемо значення альфа-каналу першої текстури з другої та побудуємо в тіньову маску з перевіркою по альфа-каналу. Точки, які пройдуть цю перевірку, запишемо в тіньову маску як 1, а які не пройдуть залишимо в 0. Після цього залишається лише побудувати сцену, проставляючи значення у буфер шаблонів (рис. 4).

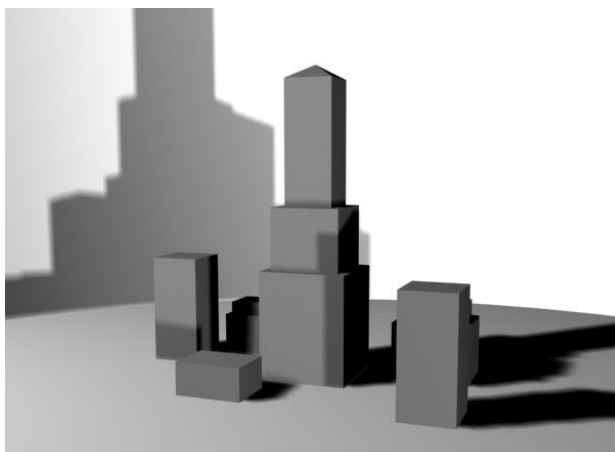


Рисунок 4. Результат створення тіньової маски

Висновки. У статті розглянуті та проаналізовані основні принципи і алгоритми побудови тіней в тривимірній графіці. Представлені алгоритми були розділені на дві явних дії: побудова тіньової маски та побудова сцени з використанням тіньової маски, і розділені на три стадії. Досліджена проблема побудови тіней в тривимірних сценах, в реальному часі. Перспективою подальших досліджень є розробка способу усунення помилкового самозатінення, яке виникає через недостатню точність уявлення глибини через альфа-канал.

Список використаних джерел:

1. Blinn James. Me and my (fake) shadow. *IEEE Computer Graphics and Applications*. 1988. 8(1). P. 82-86.
2. Franklin C. Crow. Shadow algorithms for computer graphics. In *Computer Graphics (SIGGRAPH '77 Proceedings)*. 1977. P. 242-248.
3. Sander P., Gu X., Gortler S., Hoppe H., Snyder J. Silhouette Clipping, *Computer Graphics (SIGGRAPH 2000 Proceedings)*. 2000. P. 327-334.

УДК 004.942

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ СТВОРЕННЯ МОДЕЛЕЙ КРИВОЛІНІЙНИХ ПОВЕРХОНЬ

Холодник Ю.В., к.т.н.

e-mail: yuliya.kholodnyak@tsatu.edu.ua

Мірошніченко М.Ю., к.т.н.

e-mail: mykola.miroshnychenko@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Комп'ютерне моделювання є потужним інструментом розв'язання багатьох науково-виробничих задач. Особливо це стосується проектування функціональних поверхонь із підвищеними динамічними якостями. Підвищені динамічні якості необхідні поверхням, що обмежують корпусні вироби авіа-, авто-, суднобудування, лопатки турбін і змішувачів, канали двигунів внутрішнього згоряння, робочі органи сільськогосподарських машин.

До появи верстатів із числовим програмним управлінням (ЧПУ) вироби, обмежені такими поверхнями, оброблялися за допомогою універсального обладнання, на яке встановлювалися індивідуальні обкатувальні пристрої. Така технологія призводить до виникнення похибок під час виготовлення деталі щодо геометричної моделі. Не менш важливим питанням є фізичне зношування універсального обладнання та наявність професійних кадрів, які здатні виконати роботу на таких верстатах.

Завдання виготовлення виробів, обмежених складними функціональними поверхнями, з високим ступенем точності вирішують технології, які потребують використання верстатів з ЧПУ. Виготовлення виробу на верстаті з ЧПУ забезпечує мінімальну похибку щодо комп'ютерної моделі. Обов'язковим етапом такої технології є створення тривимірної моделі виробу з використанням CAD - системи.

Більшість пакетів містять обмежену кількість кривих ліній, які можуть бути використані для створення динамічних поверхонь. Коли виникає необхідність побудови кривих ліній, яких немає в CAD-системі (астроїди, евольвенти, епітрохоїди та ін), найчастіше формується набір точок, розташованих на кривій, після чого цей точковий ряд інтерполюється за допомогою сплайну.

Коли до моделі поверхні висувають високі вимоги точності, виникає необхідність задавати В-сплайн, який апроксимує криву, великою кількістю вузлів. Кількість цих вузлів може налічувати сотні або іноді тисячі. Вручну розташувати та об'єднати таку кількість точок є трудомістким завданням. Виникає проблема автоматизації цього процесу за допомогою програмних засобів.

Складні геометричні поверхні можуть бути сформовані на основі каркасу, елементами якого є криві плоскі.

Нами розроблено методи формування обводів із закономірною зміною кривизни на основі вихідного точкового ряду [1, 2]. Метод ґрунтується на аналізі вихідного точкового ряду, в результаті якого визначається область можливого розташування кривої та діапазони можливих значень її характеристик – положень нормалей (дотичних) та центрів кривизни (значень радіусів кривизни).

Вихідними даними для моделювання кривої є упорядкований точковий ряд. Цей точковий ряд називатимемо дискретно представленою кривою (ДПК). Крім вихідного ряду до визначника ДПК входять її геометричні характеристики, які

необхідно забезпечити в процесі моделювання. Врахування геометричних характеристик при поданні кривої значно збільшує точність її моделювання. Це насамперед положення дотичних і значень радіусів кривизни.

В результаті попереднього аналізу вихідного крапкового ряду визначаються ділянки, на основі яких можна сформувати монотонну криву. Отримані монотонні криві формуються окремо та стикаються з другим порядком гладкості.

Кожна монотонна крива моделюється окремо на ділянках, які обмежені двома сусідніми вихідними точками. При цьому забезпечується монотонність зміни кривизни на кожній ділянці та стикування ділянок із заданим порядком гладкості.

При формуванні ділянки кривої точка згущення і дотична, яка через неї проходить, визначаються всередині базисного трикутника. Базовий трикутник обмежений дотичними, які проходять через два сусідні вузли та хорду, яка ці вузли з'єднує. Призначення точок згущення всередині базисних трикутників забезпечує запобігання осциляцій кривої, що формується.

Розроблений метод пройшов апробацію під час виготовлення роторів компресорів. До динамічних якостей робочих поверхонь роторів висуваються підвищені вимоги. Для забезпечення цих вимог потрібна висока точність моделювання та обробки поверхонь. До цього часу ротор оброблявся на стругальних верстатах з ручним керуванням із застосуванням обкатувального пристрою. В основі процесу обкатки лежить переміщення стругального різця траєкторією відповідної лекальної кривої. Лекальна крива є обведенням першого порядку гладкості.

При використанні такої методики подання поверхні забезпечити високу точність виготовлення поверхні неможливо. Крім того, дана методика передбачає використання обкатувального пристрою, який вже не виробляється, а ті одиниці, які залишилися, практично вичерпали свій ресурс.

Обробка робочих поверхонь ротора на верстатах із ЧПУ вимагає створення якісної тривимірної моделі виробу. Геометрична модель поверхні створена за розробленою нами методикою з використанням оригінального програмного забезпечення. Вихідними даними для моделювання є масив точок, координати яких отримані в результаті вимірювання на виробі.

Робоча поверхня ротора являє собою, циліндричну поверхню утворену поступовим переміщенням твірної лінії – евольвенти кола. Евольвента кола – це крива, центри кривини якої утворюють коло. Лінія, утворена центрами кривини називається еволютою.

Утворення евольвенти кола можна представити як траєкторію руху кінця натягнутої нитки, яку розмотують з кола – еволюти, що визначає евольвенту.

Параметричні рівняння евольвенти кола

$$x = r \cos \varphi + r \sin \varphi$$

$$y = r \sin \varphi - r \cos \varphi$$

де r - радіус кола, яке визначає евольвенту;

φ - кут відповідної дуги евольвенти

Початковими даними для побудови точкового ряду є еволюта – коло діаметра d .

Перша точка точкового ряду визначається на вказаному колі. Для побудови поточної точки ряду, необхідно виконати наступні дії:

- на даному колі (еволюті) призначається точка, що відповідає точці ряду;
- через обрану точку проводимо пряму, дотичну до кола;

- точку отримаємо на дотичній, відклавши від точки дотику довжину, яка дорівнює довжині дуги кола, обмеженої початковою точкою евольвенти і точкою на колі, що відповідає до точці ряду, яка визначається.

Довжину дуги між точками i та $i + 1$ визначаємо по формулі:

$$a = \frac{\pi d}{m},$$

де d - діаметр кола;

m - число частин, на які розділено коло.

Щоб отримати весь точковий ряд на дузі кола, що відповідає усю дузі евольвенти, визначаються точки, які розділяють дугу кола на рівні частини (рис. 1).

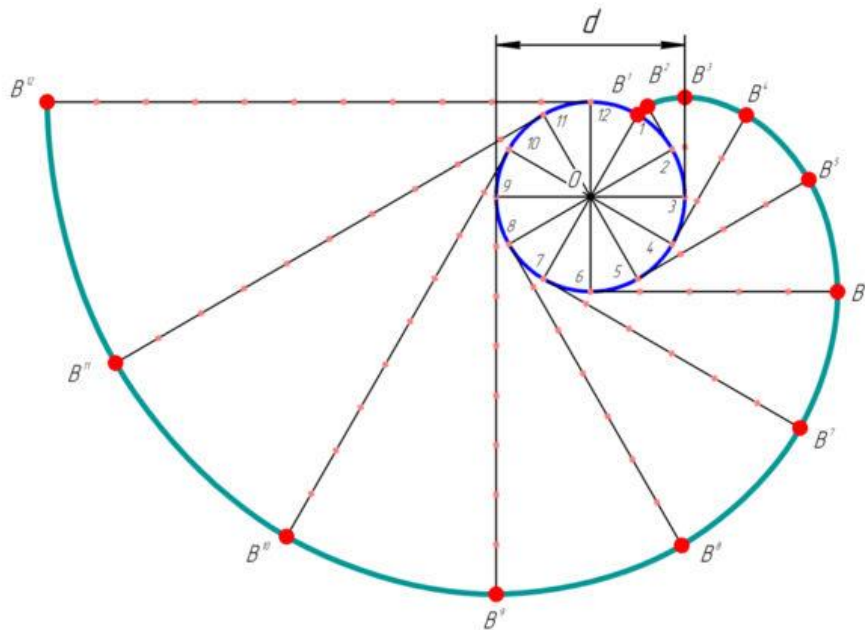


Рисунок 1. Схема побудови вихідного точкового ряду

Кількість точок на колі відповідає кількості вузлів утвореного точкового ряду. Для кожної точки еволюти, визначається положення вузла, розташованого на евольвенті.

Розроблене програмне забезпечення дозволяє провести аналіз, коригування та розрахунок робочої поверхні ротора. Коригування положення вихідних точок здійснюється відповідно до умови закономірної зміни кривизни вздовж ДПК, розташованих на поверхні. Для покращення динамічних якостей ротора елементи каркасу формуються як обводи другого порядку гладкості з монотонною зміною кривизни.

Розроблене програмне забезпечення складається з двох частин:

- програма для розрахунку точкових рядів, на основі яких формуються лінії, що входять до визначника поверхонь;
- програма для побудови тривимірних моделей елементів каркасу поверхні в пакеті тривимірного моделювання.

Розрахунок координат вузлів кривої, що формується, проводиться в символічному пакеті Maple. Результатом роботи програми є координати вузлів точкових рядів, що згущують, що представляють із заданою точністю криві, які утворюють каркас поверхні ротора.

Програма для побудови тривимірних моделей поверхонь розроблена мовою C#. З використанням інструменту API (Application Program Interface) програма інтегрована з базовим CAD-пакетом - SolidWORKS.

Вихідними для роботи створеної програми є отримані в Maple координати масиву точок. У SolidWORKS отримані точкові ряди в автоматичному режимі інтерполуються кубічним сплайном. На основі одержаного каркаса формується модель поверхні.

Похибка відповідності сформованої моделі існуючому об'єкту залежить від кількості вихідних точок, отриманих у результаті вимірів на виробі. Метод, що розробляється, дозволяє визначити кількість точок, необхідну для забезпечення заданої точності. Задаючи різну кількість вихідних точок, можна забезпечити будь-яку наперед задану точність уявлення поверхні, що моделюється. При вимірюванні ротора компресора отримано точковий ряд, який забезпечує подання поверхні з похибкою, яка не перевищує 10^{-4} мм.

Основним критерієм якості апроксимації евольвенти кола, є монотонний графік зміни кривини вздовж отриманого сплайну. Встановлено, що значення кривини, вздовж сплайну, монотонно зростає (рис. 2). Отриманий сплайн повністю задовольняє вимогам до твірної лінії динамічної кінематичної поверхні.

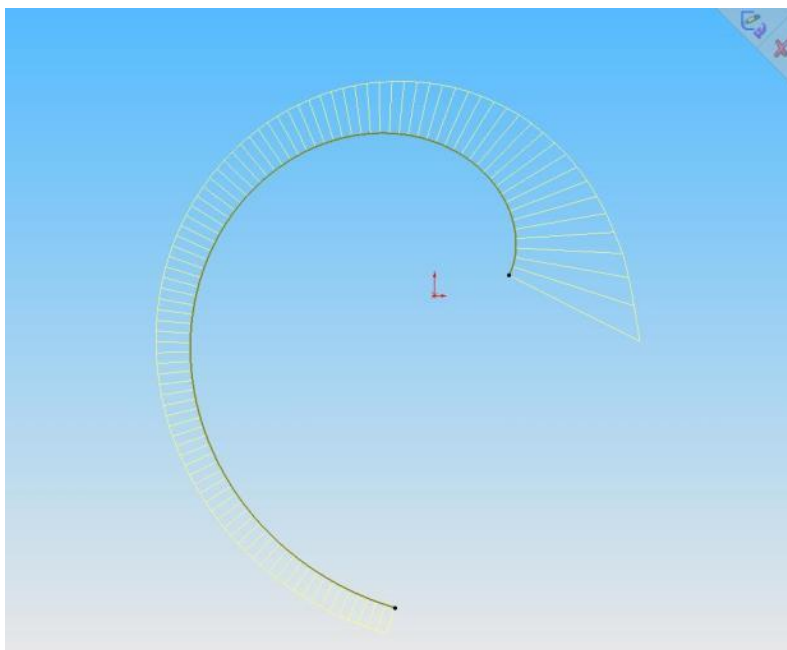


Рисунок 2. Графік кривини отриманого сплайна

Властивості сформованої моделі поверхні досліджені у пакетах SolidWORKS та SolidWorks Flow Simulation. Результати дослідження SolidWORKS показали, що значення кривизни монотонно зростають вздовж ліній різних напрямів, що належать поверхні (рис. 3). Монотонна зміна кривизни забезпечує підвищені динамічні якості поверхні. Результати дослідження моделі SolidWorks Flow Simulation показали, що сформована поверхня забезпечує стійкий характер обтікання поверхні повітряним потоком.

Отримана модель поверхні ротора є вихідними даними для розробки програми для верстата з ЧПУ в пакеті PowerMill. Дослідження виготовленого ротора компресора показали, що відхилення точок робочої поверхні від геометричної моделі не перевищує 10^{-4} мм.

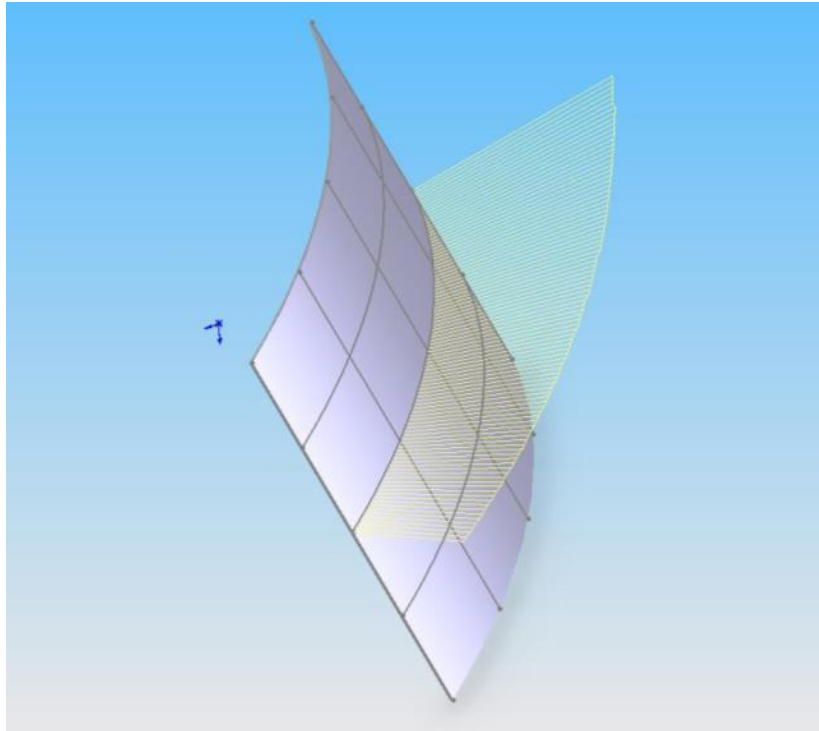


Рисунок 3. Графік кривини сплайну на робочій поверхні ротора

При збільшенні значень кривини колір змінюється від чорного до блакитного, зеленого і червоного. На рис. 4 наведений графік зміни кривини по всій площині робочої поверхні ротора за допомогою кольору.

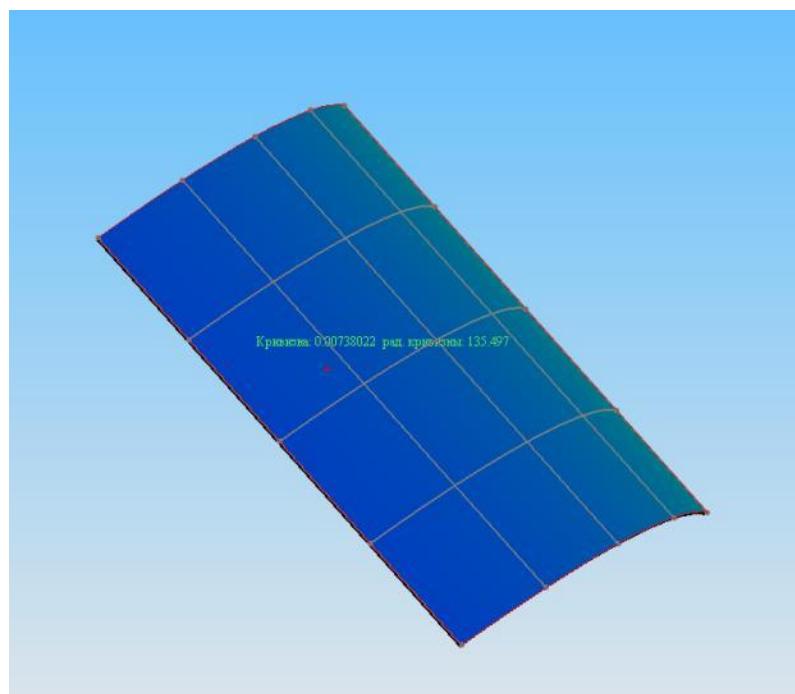


Рисунок 4. Графік кривини поверхні робочого органа в кольорі

Згідно з графіка відображеного на рисунках 3 та 4, кривина вздовж отриманої поверхні змінюється закономірно, що відповідає умовам, які висуваються до динамічних поверхонь.

Технології, базовані на використанні ЧПУ, дозволяють виготовляти деталі, з мінімальною попередньо заданою похибкою, відносно відповідної комп'ютерної моделі, точність формування в CAD – пакеті забезпечує функціональні якості виробу.

При побудові моделі ротору газодувки головною задачею є точне формування сплайну, який використовується як твірна лінія робочої поверхні ротору. Точність формування сплайну можна оцінити похибкою з якою сплайн апроксимує теоретичний контур твірної лінії – евольвенту кола.

Похибка апроксимації залежить від кількості узлів призначених на евольвенті та визначають сплайн. Чим щільніший вхідний точковий ряд тим точніше отриманий сплайн представляє вихідний контур.

Для оцінки максимальної абсолютної похибки апроксимації ми використовуємо методику, яка надає можливість визначити межі всередині яких розташована крива з монотонною зміною значень кривини, яка інтерполює заданий точковий ряд. Максимальна абсолютна похибка визначається наступним чином.

Через кожні три послідовні точки ряду проводимо, обумовлено ними коло. Ці кола будемо називати стичними. Якщо точковий ряд представляє криву з монотонною зміною кривини, то радіус стичних кіл буде монотонно змінюватися уздовж точкового ряду (рис. 5).

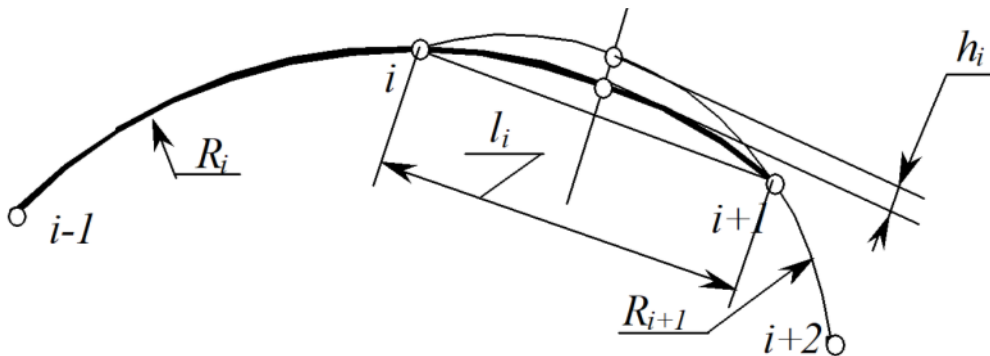


Рисунок 5. Стичні поля, обумовлені вхідним точковим рядом

Будь яка гладка крива лінія, що проходить через точки нашого ряду, у якій значення кривини монотонно змінюються в тому ж напрямку, що й значення радіусів стичних кіл, пройде у середині смуги стичних кіл.

Величину погрішності, з якою створюваний нами обвід апроксимує евольвенту можна оцінювати відстанню між сусідніми стичними колами. Максимальна абсолютна похибка апроксимації на ділянці обводу $(i:i+1)$ (h_i) приймаємо рівний довжині відрізка, обмеженого точками перетину перпендикуляра до середини хорди супровідної ламаної лінії $[i:i+1]$ з стичними колами, обумовленими точками $i-1; i; i+1$ та $i; i+1; i+2$.

Якщо величина h_i на всіх ділянках ДПК не перевищує заданої точності апроксимації, а створений нами обвід проходить усередині смуги стичних кіл, то можна вважати завдання апроксимації вирішеною.

Максимальна абсолютна похибка апроксимації всього відрізка евольвенти визначається за формулою:

$$\Delta_{\max} = \delta \cdot l_{\max},$$

де $l_{\max}$ - максимальна довжина хорди супровідної ламаної лінії,

$\delta = \left(\frac{h_i}{l_i} \right) \min$ - мінімальне за значенням відношення довжини між стичними

колами до довжини відповідної хорди супровідної ламаної лінії.

Алгоритм визначення $\Delta \max$ виглядає наступним чином:

- виконується аналіз точкового ряду, розташованого на евольвенті з метою

визначення величини δ , як мінімального за значенням відношення $\frac{h_i}{l_i}$;

- визначається величина $\Delta \max$.

Підвищити точність апроксимації можливо ущільнивши точковий ряд. Так ущільнивши точковий ряд в два рази значення δ зменшиться приблизно в чотири рази.

Висновки. У роботі представлено методику, яка дозволяє в автоматизованому режимі створювати програми для верстатів з ЧПУ для обробки виробів, обмежених складними геометричними поверхнями. Методика включає наступні етапи:

- розрахунок ДПК, що є елементами каркасу поверхні, за заданими геометричними умовами;

- формування безперервних ліній, що інтерполують отримані точкові ряди;

- створення тривимірної моделі поверхні у пакеті SolidWorks;

- створення програми для верстата з ЧПУ у САМ-системі PowerMill для обробки отриманої моделі.

Розроблена методика дозволяє конструювати поверхні як на основі технічної документації, так і за результатами вимірювань на виробі. При формуванні моделі поверхні можуть використовуватись будь-які криві ліній, у тому числі ті, яких немає в меню CAD-пакетів.

Список використаних джерел:

1. Гавриленко Е.А., Холодняк Ю.В., Антонова Г.В., Чаплинский А.П. Разработка алгоритма программного обеспечения для формирования обводов по заданным геометрическим условиям. *Праці Таврійського державного агротехнологічного університету*. Мелітополь: ТДАТУ, 2020. Вип. 20, т. 3. С. 293-303

2. Холодняк Ю.В., Гавриленко Е.А., Ивженко А.В., Найдых А.В. Моделирование участка пространственной монотонной кривой линии. Сучасні проблеми моделювання: наукове фахове видання. Мелітополь: МДПУ, 2020. Вип.17. С. 131-137.

3. Kholodniak Yu., Havrylenko Ye., Ivzhenko O., Pykhtieva I. *Modeling surfaces according to specified conditions based on array of points*. Modern modeling problems. Melitopol: MSPU, 2021. Vol. 24. P. 175-182.

4. Kholodniak Yu., Havrylenko Ye., Pykhtieva I., Shcherbyna V. Design of Functional Surfaces in CAD System of SolidWorks via Specialized Software. Modern Development Paths of Agricultural Production. Cham: Springer International Publishing, 2019. P. 63-74.

5. Havrylenko Y., Kholodniak Y., Vershkov O., Naidysh A. Development of the method for the formation of one-dimensional contours by the assigned interpolation accuracy. *Eastern-European Journal of Enterprise Technologies*. 2018. Vol. 1, Iss. 4(91). P. 76-82.

УДК 004.92

СИСТЕМНИЙ ДИЗАЙН

Стеценко К.О.

e-mail: stecenkokaterina8@gmail.com

Науковий керівник асистент Гешева Г.В.

Таврійський державний агротехнологічний університет імені Дмитра
Моторного

Актуальність та постановка проблеми. Актуальність системного дизайну є дуже значимою в сучасному світі. Адже він відіграє важливу роль в дизайні в цілому.

Цілісне структуроутворення змісту і форми предметної системи або програми діяльності, яка отримує предметне оснащення, з одночасною і взаємозалежною розробкою всіх структурних елементів і керуючих моментів, з «виходом» на формування художнього образу, називається **системним дизайном**.

Сучасний стан справ у галузі розробки вбудованих систем вимагає швидкого виходу на ринок. Втрачений час виливається у недоотриманий прибуток, або прямі збитки від конкурентів, які займають нішу і знімають вершки. Одним із оптимальних рішень є аналіз системи із розбиттям на програмно-апаратні модулі і подальший їх синтез та інтеграція в єдине ціле.

Специфіка системного підходу в дизайні. Художнє конструювання «штучних» виробів склалося і здійснювалося тому, що в проектній діяльності, не було системного підходу. Але проєктовані окремі речі все одно «прямували» в об'єктивно існуючу систему штучної предметного середовища, значить продукт і одиничного художнього конструювання може бути розглянутий з позицій системного підходу для з'ясування ступеня його відповідності системі, в яку він буде включений. При цьому необхідно мати на увазі, що «чуттєва оболонка, в яку наділяє предмет натуралістичне свідомість, ізолює даний предмет від справжнього контексту і тим самим робить його принципово „штучною” річчю.

Основні матеріали дослідження. Основні поняття і структура системного дизайну. Істота системного дизайну, розкривають його основні поняття - «Суб'єкт», «об'єкт», «зв'язки» і ін. Ці структурні елементи (підсистеми) мають одночасно багатьма різними характеристиками і виконують різні функції.

У реальному проєктуванні до типових, різноманітним, але взаємопов'язаним об'єктів системного дизайну відносяться:

- предметна система (Тематичне єдність речей і зв'язків між ними),
- програма і сценарій (В тому випадку, коли вони являють собою тексти-зафіксовану вербальну основу майбутніх реальних дій, актів),
- аудіовізуальна комунікація (Предметно-програмна система, що забезпечує найрізноманітніші види інформаційних зв'язків),
- фірмовий стиль (Формальне і змістовне єдність предметного середовища, аудіовізуальних комунікацій і процесів, які стосуються певної функціональної цілісності)

Висновок. При постійному ускладненні пристроїв лінійний ріст розміру коду веде до того, що час розробки збільшується по експоненціальному закону. В той же час сучасний розвиток мікроелектроніки, коли вартість кремнієвих чипів набагато порядків дешевша за програмне забезпечення, дозволяє розбивати систему на слабо зв'язані між собою блоки і модулі та створювати

багатопроцесорні системи, в яких код розділений на ізольовані частини. Такий дизайн і архітектура розподілення коду, функцій і бізнес логіки між апаратною та програмною частиною дозволяє паралельно працювати над алгоритмами, програмним забезпеченням і кодом, що описує апаратуру та електричними схемами і електронними платами. В такому випадку системний дизайн із ланкою процесор-конфігурована логіка є досить ефективним рішенням при розробці як простих так і складних вбудованих систем.

Список використаних джерел:

1. <https://dou.ua/lenta/articles/esd/>
2. <http://um.co.ua/8/8-19/8-196277.html>

УДК 378:539.23

ДОСВІД ВИКОРИСТАННЯ ПРОГРАМИ AFORS-HET ПРИ ПІДГОТОВЦІ ФАХІВЦІВ У ГАЛУЗІ ЕЛЕКТРОЕНЕРГЕТИКИ

Дяденчук А.Ф., к.т.н

e-mail: dyadenchukalena@gmail.com

Таврійський державний агротехнологічний університет імені Дмитра
Моторного

Актуальність та постановка проблеми. Використання досягнень сучасної електроніки – ознака успішного розвитку будь-якого підприємства. Не випадково останнім часом на промислових підприємствах приділяється особлива увага модернізації та розвитку власної енергетики. Розвиток об'єктів сонячної енергетики дає суттєві соціально-економічні та екологічні ефекти. Ефективність фотоелектричних компонентів, що зазвичай використовуються, швидко наближається до своїх технологічних меж. Таким чином, майбутні розробки залежатимуть від досвідчених інженерів з моделювання, які добре знаються на використанні передових та нових матеріалів, розробці інтерфейсів та архітектурі пристроїв. На стадії підготовки інженера-енергетика необхідно передбачити вивчення таких областей знань, як проектування та функціонування енергооб'єктів, електроенергетичного й електротехнічного обладнання [1], вибір оптимальних рішень завдань із врахуванням вимог якості, надійності, безпечності та екологічної чистоти виробництва [1], теоретичні основи фізики функціонування електронних приладів [2], комп'ютерне моделювання цифрових компонентів техніки тощо. Для освоєння даних питань та здобуття практичних навичок необхідно використовувати спеціалізовані програмні продукти.

Зараз існує достатня кількість спеціалізованих систем проектування енергетики, але часто практичні завдання успішно вирішують на базі універсальних систем. Актуальним є питання ознайомлення здобувачів вищої освіти із різноманітними програмними засобами, що використовуються у професійній діяльності. На етапі початкового освоєння питання моделювання напівпровідникових фотоелектричних перетворювачів і вивчення процесів, що протікають в них, найбільш прийнятним засобом є програма AFORS-HET (розробник – Hahn-Meitner Institut).

У даній роботі представлено досвід застосування програми для чисельного моделювання сонячних елементів AFORS-HET v2.5 для моделювання електричних характеристик гетероструктурних сонячних елементів.

Основні матеріали дослідження. На сьогодні широкого поширення набуло комп'ютерне моделювання в таких програмних середовищах, як Analysis of Microelectronic and Photonic Structure (AMPS-1D), PC1D, Solar Cell Capacitance Simulator (SCAPS-1D) та інші. Програмою, яка дозволяє розуміти вплив геометричних й оптоелектричних властивостей на продуктивність фотоперетворювачів та оволодіти методикою управління цим причинно-наслідковим зв'язком, і при цьому маючи нескладний інтерфейс, є симулятор AFORS-HET.

Вибір даної програми обумовлений такими причинами:

- ✓ виявлення значень параметрів, що впливають на продуктивність пристрою сонячної енергетики з подальшим застосуванням отриманої інформації для підвищення продуктивності сонячних батарей;

- ✓ точне співвідношення результатів оптичного та електричного моделювання, чим забезпечується достовірність отриманих результатів;
- ✓ система має достатній набір функцій моделювання, що дозволяють вирішувати поставлене завдання;
- ✓ система досить легко освоюється користувачами, надаючи багаті можливості модифікації проектів, які розробляються.

AFORS-HET є програмою з багатовіконним графічним інтерфейсом, що дозволяє будувати та редагувати схеми, моделі та зображення компонентів (рис. 1), а також представляти результати розрахунків у зручному графічному вигляді (рис. 2).

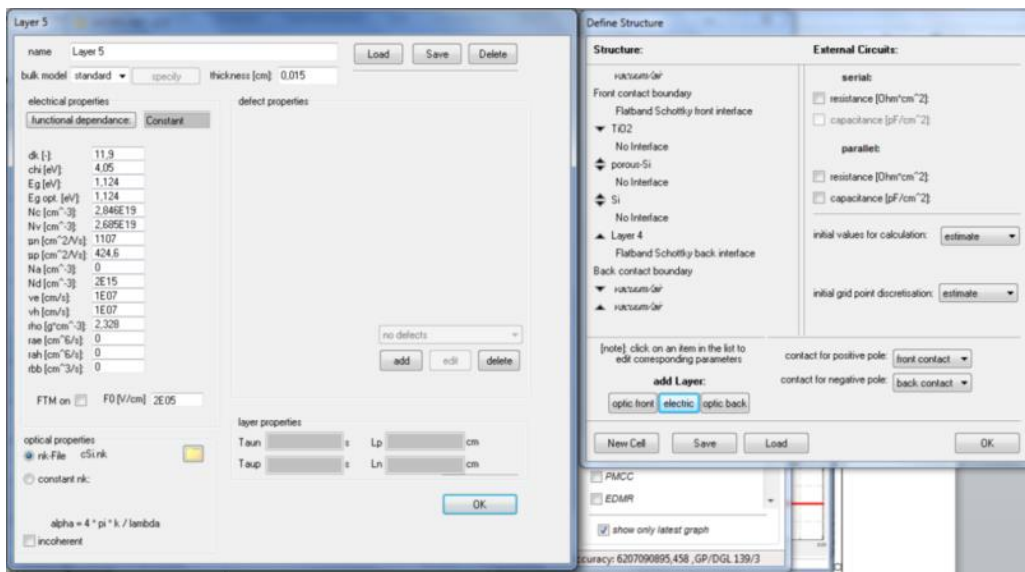


Рисунок 1. Вікно для внесення параметрів шарів у програмі AFORS-HET v2.5

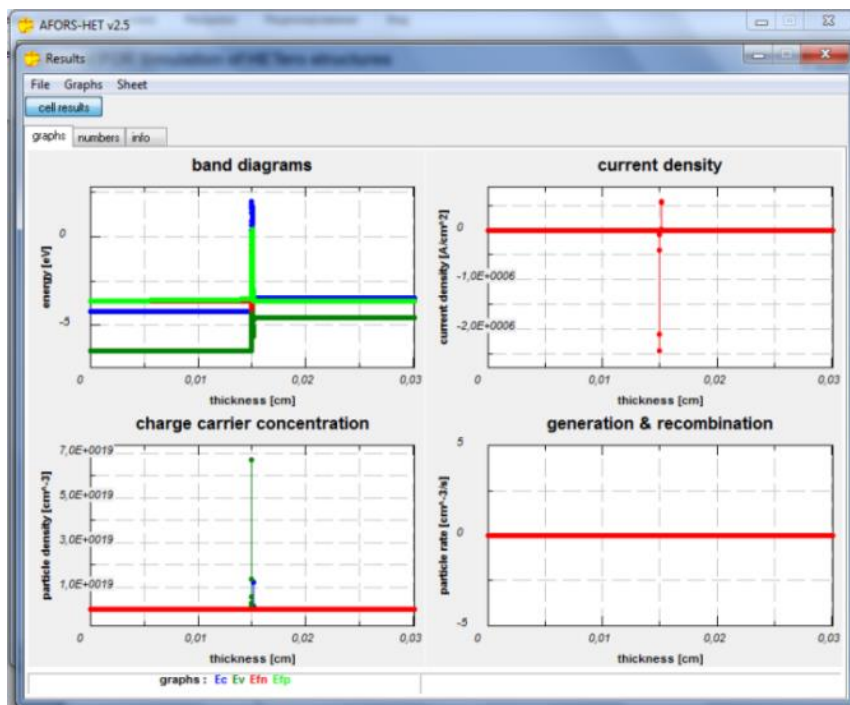


Рисунок 2. Графіки, побудовані в програмі AFORS-HET v2.5

На базі програмного середовища AFORS-HET v2.5 здобувачами вищої освіти спеціальності 141 «Електроенергетика, електротехніка та електромеханіка» було виконано ряд наукових проєктів. Так, в роботі «Моделювання функціональних характеристик фотоелектричних перетворювачів на основі TiO_2 », за допомогою програми AFORS-HET 2.5 змодельовано робочі характеристики фотоперетворювачів, виконаних на базі напівпровідникових гетеропереходів TiO_2/Si [3]. Здобувачем вищої освіти в тезах «Моделювання фотоелектричних параметрів напівпровідникових фотоелементів» досліджено вплив температури на вигляд вольт-амперної характеристики фотоелектричного перетворювача [2].

У процесі виконання вищезгаданих та аналогічних проєктів з використанням програми AFORS-HET здобувачі вищої освіти вчать аналізувати виробничі аспекти, оптимізувати робочі процеси, а також знаходити способи удосконалення та покращення архітектури сонячних елементів. Завдяки цим новим навичкам та знанням вони можуть покращити продуктивність та ефективність, а також модернізувати сучасні та майбутні пристрої.

Висновок. Таким чином, використання в освітньому процесі симулятора AFORS-HET дозволяє:

- отримати та закріпити теоретичні знання відносно функціонування напівпровідникових приладів;
- поглибити розуміння та розширити межі використання інструментів та методів для точної характеристики та моделювання характеристик фотогальванічних матеріалів та пристроїв;
- освоїти практичні навички комп'ютерного моделювання електричних характеристик гетероперехідних сонячних елементів;
- оволодіти практичними вміннями та навичками ефективного використання електронних ресурсів.

Особливістю програми AFORS-HET є простий наочний інтерфейс, потужні засоби графічного аналізу результатів моделювання, наявність віртуальних вимірювальних приладів, програма дозволяє працювати з довільною послідовністю напівпровідникових шарів та різноманітними граничними умовами [4]. За допомогою даного симулятора студенти можуть з легкістю перейти від теорії до практики, розглядаючи не тільки сонячні батареї, які є у виробництві, але й оптимізувати параметри інших гетероструктур та приладів на їх основі.

Список використаних джерел:

1. Дяденчук А. Ф. Формування енергоефективної компетентності майбутніх фахівців енергетичної галузі. Матеріали XVI Міжнародної науково-практичної конференції аспірантів і студентів «Молода наука Волині: пріоритети та перспективи досліджень» (17 травня 2022 року). Луцьк: ВНУ ім. Лесі Українки, 2022. С. 771-773.
2. Іванов В.С., Дяденчук А.Ф. Моделювання фотоелектричних параметрів напівпровідникових фотоелементів. *Автоматизація та комп'ютерно-інтегровані технології у виробництві та освіті: стан, досягнення, перспективи розвитку*: матеріали Всеукраїнської науково-практичної Internet-конференції. Черкаси, 2022. С. 89-91.
3. Шквиря В.В. Моделювання функціональних характеристик фотоелектричних перетворювачів на основі TiO_2 . IX Всеукраїнська науково-технічна конференція здобувачів вищої освіти ТДАТУ. Факультет енергетики і

комп'ютерних технологій: матеріали IX Всеукр. наук.-техн. конф., 10-25 листопада 2021 р. Мелітополь: ТДАТУ, 2021. С. 44-47.

4. Дяденчук А.Ф. Використання інформаційних технологій у науково-дослідній діяльності майбутніх фахівців енергетичної галузі. *Інформаційні технології – 2022*: зб. тез IX Всеукраїнської науково-практичної конференції молодих науковців, 19 трав. 2022 р., м. Київ / Київ. ун-т ім. Б. Грінченка; Відповід. за вип.: М.М. Астаф'єва, Д.М. Бодненко, О.М. Глушак, Г.А. Кучаковська, О.С. Литвин, В.В. Прошкін, С.М. Шевченко. К. : Київ. ун-т ім. Б. Грінченка, 2022. С. 29-31.

УДК 376

ЗНАЧЕННЯ ІНФОРМАЦІЙНО – КОМУНІКАЦІЙНІ ТЕХНОЛОГІЇ У НАВЧАЛЬНОМУ ПРОЦЕСІ ВИЩОЇ ШКОЛИ

Бондаренко Л.Ю., к.т.н., доцент

e-mail: larysa.bondarenko@tsatu.edu.ua

Ускова С.О., студент

e-mail: svetlanauskova2003@gmail.com

*Таврійський державний агротехнологічний університет імені Дмитра
Моторного*

Актуальність та постановка проблеми. У світі, тяжко уявити функціонування будь-якої сфери людської діяльності без комунікації. Що стосується сучасності, інформаційно-комунікація, все частіше, відбувається за допомоги комп'ютерів, смартфонів чи ще чогось.

Інформаційно-комунікаційні технології (ІКТ) – сукупність технологій, що забезпечують фіксацію, її обробку і обмін інформацією. Інформаційні технології – це методи і засоби отримання, перетворення, передачі, зберігання і використання інформації [1]. У світі, що оточує нас, все частіше спостерігаються позитивні тенденції поширення інформаційно-комунікаційних технологій. Стрімкий розвиток інформаційних технологій, конвергенція комп'ютерних систем, різних видів комунікацій створюють необхідність переглянути уявлення про інформаційну індустрію, а особливо, її значення і місце в суспільстві.

Проблему інформатизації суспільства часто відносять до технологічної. І дійсно, матеріальною основою ІКТ служить глобальний перехід до безпаперової інформатики, тобто, основна маса інформаційних даних, циркулюючих в суспільстві, зберігатиметься і оброблятиметься, за допомогою комп'ютерних систем, а передаватиметься по автоматизованих каналах супутникового або телекабельного зв'язку. Завдяки чому, сполучаються окремі комп'ютери і автоматизовані робочі місця в інформаційні мережі, при цьому, забезпечуючи звернення до баз даних. В основу цих досягнень закладено ряд фундаментальних винаходів і відкриттів, таких як: персональні комп'ютери, оптичні диски на десятки і сотні гігабайт, завдячуючи яким, є можливість, в компактному виді, зберігати вміст цілих бібліотек. Все це сприяє створенню, в недалекому майбутньому, високоавтоматизованого інформаційного середовища, що дозволяє отримати доступ до будь-якого знання, наданого у вигляді інформації. Фактично це означає, що інформаційне середовище дозволяє отримати доступ до знань у кожній точці світу у будь-який час. Судячи з цього, можна зробити висновок, що проблема інформатизації є, значною мірою, соціокультурною.

Основні матеріали дослідження. Для більш детального огляду ІКТ, розглянемо сферу освіти.

До головних переваг використання ІКТ можна віднести такі пункти:

1. Індивідуалізація навчання.
2. Інтенсифікація роботи.
3. Розширення інформаційних потоків.

Якщо мова йде про ситуацію з викладачем, то цей метод допомагає їм краще оцінити здібності і знання студента, зрозуміти його, спонукає шукати нові форми і методи навчання, стимулює його професійний ріст і все подальше освоєння комп'ютера.

Під час дистанційного навчання широко використовуються мережеві технології, які забезпечують доступ до навчальних матеріалів та консультації з викладачами за допомогою телекомунікаційних технологій (рис. 1).

До позитивних моментів такої системи можна віднести посилення інтелекту студента за рахунок залучення його до вирішення складніших завдань, а також розвиток логічного, прогностичного та оперативного мислення. Творчий підхід до застосування комп'ютерних технологій підвищує самооцінку студента, його впевненість у спроможності вирішувати професійні завдання.

Разом із розповсюдженням комп'ютеризації, у закладах освіти можна відслідкувати тенденцію застосування на заняттях тестів чи діагностичних комплексів, котрі студенти виконують на своїх чи університетських гаджетах. Результати комп'ютерного тестування студентів на основі спеціально розроблених програм навчання, особливо в умовах, коли прийняте рішення не підлягає скасуванню, сприяють формуванню таких рис, як точність, акуратність, відповідальність, впевненість у собі.



Рисунок 1. Види гаджетів, що дозволяють освоїти комп'ютерний простір

Після проходження, наприклад, тестування, викладач, одразу, може отримати об'єктивну картину рівня засвоєння матеріалу і, своєчасно, його скоректувати.

За останні 80 років людство сягнуло неймовірних вершин в інформаційному розвитку. Тепер, ми маємо можливість миттєво комунікувати з тими, хто за сотні кілометрів від нас, та зберігати величезну кількість інформації, без ризику її фізичного пошкодження і, за необхідності, легко знаходити потрібну інформацію, за допомоги пошукових систем, типу Google.

Висновок. Розвиток інформаційно-комунікаційних технологій суттєво полегшує зберігання та передачу різноманітних типів інформації, завдяки чому, людина може швидше оволодівати новими знаннями та сприяти науковому прогресу.

Список використаних джерел:

1. Інформаційні й комунікаційні технології. URL: <https://ua5.org/svit/281-nformacjijn-jj-komunkacjijn-tekhnolog.html>
2. Бондаренко Л. Ю., Тетервак І. Р. Інтерактивне навчання у вищому навчальному закладі. *Сучасні комп'ютерні та інформаційні системи і технології: матеріали II Всеукр. наук.-практ. інтернет-конф.* (Мелітополь, 01-12 грудня 2021р.) Мелітополь: ТДАТУ, 2021. С. 53-58.
3. Бондаренко Л. Ю., Тетервак І. Р. Впровадження та використання комп'ютерних технологій для вирішення задач опору матеріалів. *Сучасні комп'ютерні та інформаційні системи і технології: матеріали I Всеукр. наук.-практ. інтернет-конф.* (Мелітополь, 07-25 грудня 2020р.) Мелітополь: ТДАТУ, 2021. С.82-83.

УДК 37.091.33

ІНСТРУМЕНТИ КОМУНІКАЦІЇ ДЛЯ ПІДГОТОВКИ ФАХІВЦІВ ТЕХНІЧНИХ СПЕЦІАЛЬНОСТЕЙ

Дереза О.О., к.т.н.,
Дереза С.В., ст. викл.

e-mail: olena.dereza@tsatu.edu.ua
e-mail: derezasv2017@gmail.com

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Сучасне життя розвивається швидкими темпами, тому система вищої освіти також повинна розвиватися, змінюватися. Традиційне навчання з його авторитаризмом, орієнтацією на середнього учня, перевагою репродуктивної діяльності над пошуковою не відповідає вимогам часу.

Тому в Україні формується нова система освіти, яка орієнтована на входження у світовий освітній простір, супроводжується істотними змінами в педагогічній теорії і практиці освітнього процесу [1].

Сьогодні багато студентів вміло поєднують роботу і навчання, причому, коли останнім приносить стабільність і матеріальну незалежність, сенс дорогоцінної «корочки» і зовсім втрачається. Виникає інше питання: «А навіщо мені це навчання, коли я вже знаю свою майбутню спеціальність і маю конкретне робоче місце»? Повернути інтерес до вивчення спеціальних предметів, зробити навчання цікавим, посилити бажання учитися спонукає до пошуків. Сприятливі вирішення поставлених перед освітою завдань мають інноваційні технології навчання.

Основні матеріали дослідження. Дуже швидкі темпи розвитку інформатизації суспільства вимагають широкого застосування нових технологій у вивченні технічних дисциплін. За короткий час створено ряд комп'ютерних засобів навчання. Вони дозволяють суттєво перебудувати і вдосконалити навчальний процес. У наш час комп'ютер дає нові можливості для творчого розвитку, дозволяє звільнитися від нудного традиційного навчання і розробити нові ідеї, дає можливість вирішувати цікавіші проблеми. Все це неодмінно передбачає широке застосування комп'ютерної техніки [2, 3].

Впровадження у навчально-виробничий процес електронних навчальних посібників сприяє розвитку самостійної діяльності, підвищенню їх пізнавального та професійного інтересу. Таке навчання орієнтоване на самоосвіту, розвиток особистості відповідно до індивідуальних потреб. При цьому, посібник є самим терплячим педагогом, здатним скільки завгодно повторювати будь-які завдання, домагаючись вірної відповіді й, у кінцевому рахунку, автоматизації навичок, що відпрацьовуються. Використання електронних підручників і тестування звільняє викладача від дуже нудної і втомливої процедури письмової перевірки знань.

Під час вивчення технічних дисциплін майбутніми фахівцями на перший план виступає технологія розгляду об'єкта, його робочої схеми, взаємодії його вузлів деталей і механізмів. При опануванні нової техніки, яка відсутня в лабораторіях вишу, в ситуації коли здобувачі обмежені у безпосередньому спостереженні реальних технічних об'єктів, опрацювання наочності стає особливим видом пізнавальної діяльності. Також цей вид навчальної роботи дає змогу отримувати конкретні і певні уявлення про будову вузлів і механізмів нових машин вітчизняного і зарубіжного виробництва.

Переходу суспільства до дистанційної форми роботи, навчання, соціального спілкування та інших видів онлайн взаємодії сприяли карантинні заходи під час пандемії Covid-19. Все більш популярним стає використання хмарних технологій в освіті, позаяк це відкриває багато можливостей як для навчальних закладів, так і для вчителів та студентів. Завдяки хмарним сервісам і наявності підключення до Інтернету студенти можуть отримати та виконувати завдання, незалежно від місця фізичного їх розташування. Впровадження хмарних сервісів у навчальний процес дає змогу об'єднати викладачів та учнів в єдиній онлайн-платформі.

Багато особливостей навчання є під час війни. Відбуваються зміни в організації самого процесу, зазвичай вони мають негативний характер, так як люди вимушені багато часу перебувати у сховищах, або взагалі покидати свої домівки. Деякі були вимушені змінити своє місце проживання, а деякі змушені були виїхати навіть за кордон. Іноземці, які навчались в вищих навчальних закладах до початку воєнних дій вимушені були евакуюватись задля своєї безпеки. Війна позбавила можливості навчатись для деяких з них взагалі. Іще однією проблемою виступає обмеженість або відсутність мобільного зв'язку та Інтернету.

Є безліч платформ для дистанційного навчання, які надають освітні онлайн-послуги і відрізняються набором можливостей. Легко вести облік, легко давати завдання, легко ці завдання перевіряти, легко створювати новий контент. Наприклад, програма Moodle зручна в експлуатації, тут безліч інструментів для роботи, вона безкоштовна, але для того щоб встановити систему і користуватися нею, необхідно мати хостинг і домен.

В складне становище потрапили ті ВНЗ, які опинилися на окупованій території. Навіть перемістившись до підконтрольної території України, заклади мають певні складності з-за вимикання світла та Інтернету. Також великий ризик зламу сайтів ворожими хакерами. Відсутність зв'язку із сервером не дає можливість доступу студентам до матеріалів, викладеним на платформі.

Як альтернативний варіант, можна організувати онлайн навчання за допомогою Google Classroom - безкоштовна платформа для організації роботи з групами. Досить мати поштову скриньку на Google.

У додатку для Android з'явилася можливість працювати з класом без інтернету або з нестабільним підключенням. Учні можуть в офлайн-режимі переходити до своєї роботи, переглядати завдання, відкривати прикріплені файли Диска і писати в Google Документах. У режимі онлайн пристрій автоматично синхронізує файли кожні 24 години у фоновому режимі. Таким чином вам завжди доступні актуальні файли та стрічка курсу.

Сучасні хмарні технології в парі зі смартфонами дають можливість суттєво полегшити роботу викладача та зменшити час на опитування. Для перевірки знань після вивчення певної теми навчальної програми можливо застосовувати сервіс Гугл форми від Google. Сервіс простий у застосуванні, дає можливість швидко створювати тестові матеріали на основі існуючих шаблонів та форм з можливістю надсилання тестових матеріалів кожному учню або групі учнів, контролювати їх виконання і в автоматичному режимі перевіряти правильність рішення даного тесту з автоматичним виставленням оцінки кожному студенту або групі [2, 5].

Портал Google Forms є частиною офісного інструмента Google Drive. Це, мабуть, один із найшвидших і найпростіших способів створити власне опитування чи тест. Отриманий тест можна надіслати студентам електронною поштою або вставити на свій веб-сайт за допомогою спеціального коду.

Завантаження рисунків та формул для тестування у Google Forms значно простіше ніж на платформі Moodle (рис. 1).

Для створення та зберігання навчальних матеріалів тестового характеру, різноманітної наочності, інструктивних карт доцільно буде створювати посилання для студентів на відповідний контент хмарного сховища з можливістю кодування та швидкого доступу до нього за допомогою технології **QR-кодів** для запобігання кібератак.

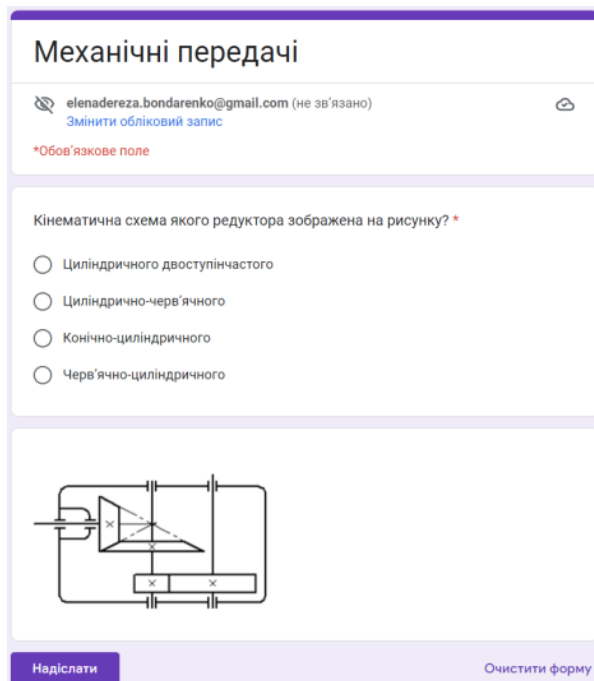


Рисунок 1. Вигляд тесту у формі від Google

Саме технічні дисципліни дають багатий матеріал для відпрацювання найрізноманітніших методів і прийомів роботи з інформацією. Окремі деталі та збірки можна створити за допомогою графічних програм і демонструвати онлайн [4]. Крім вивчення будови вузла або машини необхідно також знати принципи роботи, регулювання, технологічні налагодження тощо. Викладання таких предметів пов'язане з використанням величезного обсягу найрізноманітнішої інформації, включаючи ознайомлення з іноземними машинами та технологіями. Застосування електронних підручників та посібників не завжди вирішує цю проблему, а створення анімації вимагає певного часу.

В процесі навчання доцільним є використання навчального відео, оскільки слуховий та зоровий аналізатори є домінуючими каналами сприйняття навчального матеріалу. Використання відеоматеріалів значно покращує ефективність навчального процесу, спонукає до комунікативної діяльності та дозволяє засвоїти до 65% матеріалу. А інколи майже єдиний спосіб зрозуміти принцип дії механізму в умовах дистанційного навчання без можливості займатися з лабораторним обладнанням.

Відео є мультимодальним, тобто задіює різні органи чуття, завдяки чому покращує сприйняття та засвоєння навчальних матеріалів. Об'єктивна суспільна значущість відео підтверджує доцільність його використання в навчальному процесі. Тому ще одним із ефективних компонентів інформаційних технологій є демонстрація навчальних відеофільмів. Створивши із завантажених відео список відтворення, можна надавати студентам посилання і доступ до перегляду певних відеофільмів для ознайомлення з матеріалами дисциплін (рис. 2).

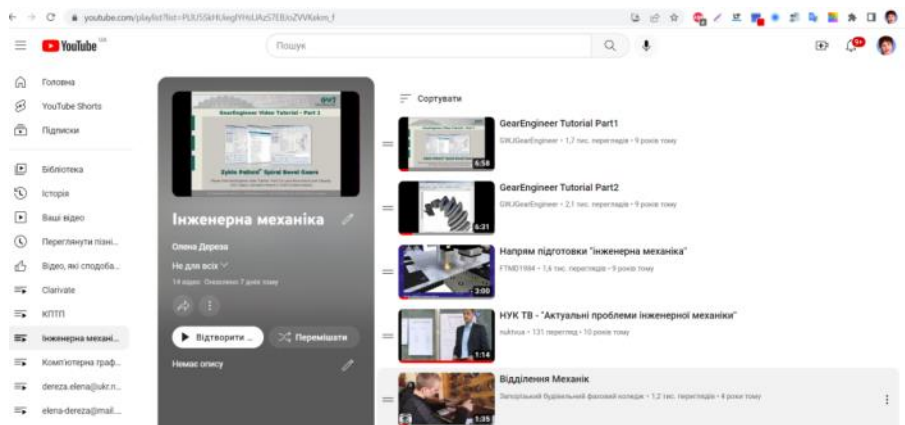


Рисунок 2. Сторінка YouTube на Google Диску

Сумісність хмарних програм з будь-яким пристроєм та операційною системою дає змогу програмам безперешкодно працювати в інтернет-браузерах, а мінімальні вимоги до апаратних ресурсів дозволяють підключатися до відповідних академічних програм навіть з дешевого смартфона. Тому студенти з низьким рівнем доходу можуть отримати доступ до високої якості навчання, тобто впровадження хмарних технологій усуває фінансову нерівність, ставлячи студентів усіх станів до одного освітнього середовища. Це значуща перевага дистанційного навчання, особливо для здобувачів із сільської місцевості з невеликим рівнем доходу.

Висновок. В сучасних, а особливо в умовах воєнного стану, для підготовки фахівців технічних спеціальностей необхідно впровадження інноваційних інструментів комунікації. При цьому створюються додаткові умови, які сприяють досягненню поставлених цілей навчання, зростають їхні дидактичні можливості щодо формування в здобувачів професійних компетентностей незалежно від місця їх перебування. Перенесення навчального середовища в хмару – це гарний інструмент для оптимізації освітнього процесу і підготовки навчального закладу до майбутнього.

Список використаних джерел:

1. Освіта України в умовах воєнного стану. Інноваційна та проєктна діяльність: Науково-методичний збірник/ за загальною ред. С. М. Шкарлета. Київ-Чернівці «Букрек». 2022. 140 с.
2. Олена Дереза Цифрові інструменти для навчання і роботи. *Матеріали III Міжнародної науково-практичної інтернет-конференції “Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації”* 30 вересня 2022 р. 5с.
3. Дереза О.О., Дереза С.В. Аналіз процесу впровадження дистанційної форми навчання у закладі вищої освіти. *Зб. наук.-метод. пр. ТДАТУ «Удосконалення освітньо-виховного процесу в закладі вищої освіти»*. 2021. Випуск 24 С. 454-461.
4. Дереза О.О., Дереза С.В. Моделювання механічних передач / Науковий вісник Таврійського державного агротехнологічного університету імені Дмитра Моторного [Електронний ресурс]. Мелітополь: ТДАТУ, 2020. Вип. 10, том 1. URL: <http://oj.tsatu.edu.ua/index.php/visnik>. DOI: 10.31388/2220-8674-2020-1-30.
5. Дереза О.О., Дереза С.В. Засоби і методи навчання професійним дисциплінам в технічному закладі вищої освіти. *Зб. Наук.-метод. Пр. ТДАТУ «Удосконалення освітньо-виховного процесу в закладі вищої освіти»*. 2019. Вип. 22 С. 202-210.

УДК 004.62

ЕКСПОРТ ЖУРНАЛУ ОЦІНОК З НАВЧАЛЬНОГО ПОРТАЛУ З ВИКОРИСТАННЯМ POWER QUERY

Кашкар'ов А.О., к.т.н.

e-mail: anton.kashkarov@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені
Дмитра Моторного

Актуальність та постановка проблеми. Інформаційні технології виявились максимально адаптованими до вимог дистанційного навчання, особливо для України у 2022 році. Для того, щоб підготуватись до викликів військового стану та відповідати ним науково-педагогічний склад Таврійського державного агротехнологічного університету імені Дмитра Моторного розробляв і використовує, як локальні рішення [2, 6] так і інструменти та методики на основі типових програмних засобів [3, 5]

Освітній портал Таврійського державного агротехнологічного університету (ОП ТДАТУ) [4] беззаперечно відповідає викликам сьогодення та допомагає в отриманні нових знань здобувачами вищої освіти з урахуванням національних освітніх традицій. Але внутрішні вимоги університету до компонування та наповнення дисциплін дещо ускладнюють комплексне оцінювання роботи здобувачів. Мова йдеться про візуальне сприйняття журналу оцінок, який розтягується в екрані браузеру відповідно до кількості завдань, особливо, якщо їх назва повністю або частково відповідає назві матеріалу до якого вони відносяться (питання лекції, назва лабораторної або практичної роботи, назва модулів та видів самостійної роботи). Якщо експортувати журнал оцінок інструментами ОП ТДАТУ, то візуальне сприйняття можливо покращити додатковими маніпуляціями, але усі дії необхідно постійно повторювати при оновленні оцінок.

Таке представлення результатів роботи здобувачів ускладнює моніторинг їх успішності та синхронізацію роботи на порталі і електронного журналу, а також сприяє неефективному використанні робочого часу навчально-педагогічного персоналу.

У статті пропонується робота з журналом оцінок освітнього порталу із використанням пакету Power Query [1] при експорті журналу оцінок як таблиці Excel.

Основні матеріали дослідження. У кінцевому результаті, для роботи із журналом оцінок ОП ТДАТУ створено запит отримання даних з експортованого журналу оцінок у вигляді таблиці Excel з наступною зміною форматів комірок інструментами Excel. Це дозволило зробити таблицю успішності (рис. 1), яка зручна для візуального сприйняття.

1	2																			+				
		A	B	G	M	Q	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AU	AV	AW	
1	2	ІС		Лекція 1	Лекція 2	Лекція 3	Лекція 4	Лекція 5				ПЗ 1		ПЗ 2		ПЗ 3		ПЗ 4						
		Ім'я	Прізвище	Л1	Л2	Л3	Л4	Л5-1	Л5-2	Л5-3	Л5	Л5	Л1	Л1	Л2-1	Л2-2	Л2	Л3	Л3	Л4	Л4	Модуль 1 (ПЗ)	Модуль 1 (Л)	мод
6		Анастасія	Базельчук	1,65	1,92	1,9	1,2	1	1	0,88	1,92	6	0,76	3	2,85	1	6	2	6	0,92	28,53	3,57	9	
7		Ангеліна	Базельчук	1,83	0,952	1,7	1,2	1	1	0,88	1,92	4,5	1	2,85	2,85	0,76	6	1,888	6	0,92	26,768	2,782	7,33	
8		Анна	Бідна	2	2	2	2,6	1,3	1	1	1	2	6	1	3	3	1	6	2	6	1	29	4	8,67
9		Катерина	Васютіна	1,75	1,868	1,9	1,3	1	1	1	2	4,8	1	0	3	0,76	6	1,556	6	1	24,116	3,618	8	
10		Каріна	Віщаренко	0,745	1,6188	2,6	0,9	1	1	1	2	5,1	0,5	2,55	2,55	0,5	6	2	6	0,84	26,04	2,3638	5,33	
11		Тетяна	Віщаренко	1,085	1,753	2,6	1,3	1	1	1	2	4,8	0	2,55	2,55	0	6	2	6	0	21,9	2,837	5,33	
31																								
32	4	Максимальна оцінка		2	2	2	2	2	1	1	1	2	6	2	3	3	2	6	2	6	2	32	8	10

Рисунок 1. Результат запити до експортованого журналу оцінок з ОП ТДАТУ

Для форматування представленого журналу успішності (рис. 1) студентів використано:

- групування за видом роботи (питання лекції, за якими наявні тести, об'єднанні за кожною лекцією окремо) – блок 1;
- найменування видів робіт приведено у відповідність з форматом електронного журналу – блок 2;
- групування оцінок за етапами навчального процесу (проміжні та загальний модулі) – блок 3;
- умовне форматування комірок (кольорова заливка відносно максимальної оцінки за вид роботи) – таблиця оцінок та блок 4.

Роботу можна розділити на певні етапи, які включають отримання даних, створення запиту за допомогою Power Query та форматування даних інструментами Excel.

Етап 1. З навчального порталу експортуємо журнал оцінок як таблицю Excel.

Етап 2. На вкладці Power Query обираємо групу «Отримати зовнішні дані» - «З файлу» (рис. 2). У діалоговому вікні, яке відкриється обираємо файл, який експортували з освітнього порталу на етапі 1.

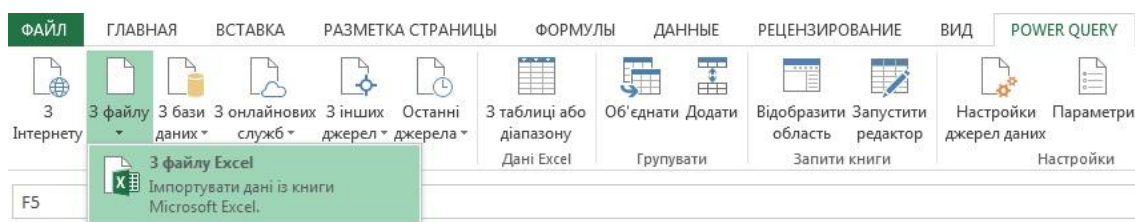


Рисунок 2. Завантаження журналу оцінок до Excel

Якщо файл не пошкоджено та не має помилок, то відкриється навігатор (рис. 3), на лівій частині якого обираємо лист книги Excel, на якому розташовані дані, а з правої сторони відображається зміст листу. Візуально оцінюємо відповідність даних. Якщо дані правильні – натискаємо кнопку «Змінити»

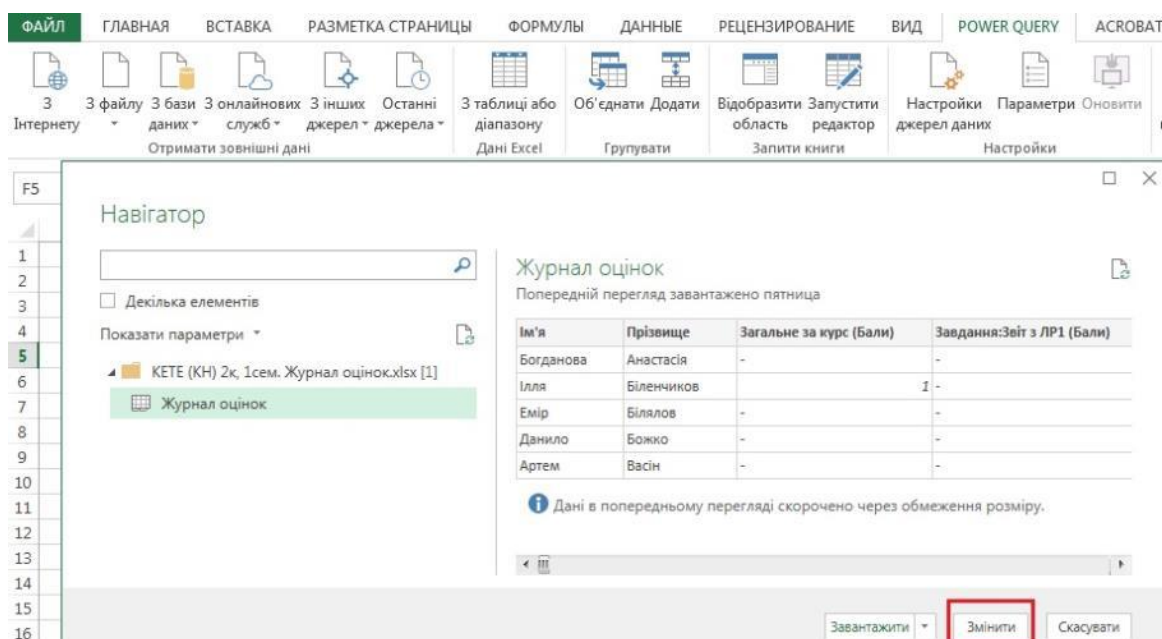


Рисунок 3. Навігатор для попередньої перевірки змісту даних, які експортуються

Етап 3. Створення та форматування запиту.

Етап 3.1. Зміна типу даних (рис. 4). Необхідність цього етапу пов'язана з тим, що дані з освітнього порталу експортуються у текстовому форматі. Excel та інструмент Power Query має можливість автоматичного розпізнавання формату, але при виконанні арифметичних операцій з комірками, які містять короткого тире «-» (наявне при відсутності оцінки, якщо студент не проходив вид контролю), результуюча комірка матиме помилку - «Error» (рис. 5). Щоб уникнути таких помилок та забезпечити можливість арифметичних операцій з цими даними необхідно змінити тип даних у комірках. Для цього, на вкладці «Додавання стовпця» додаємо «Умовний стовпець» та у діалоговому вікні, яке відкривається, вказуємо правила (рис. 4):

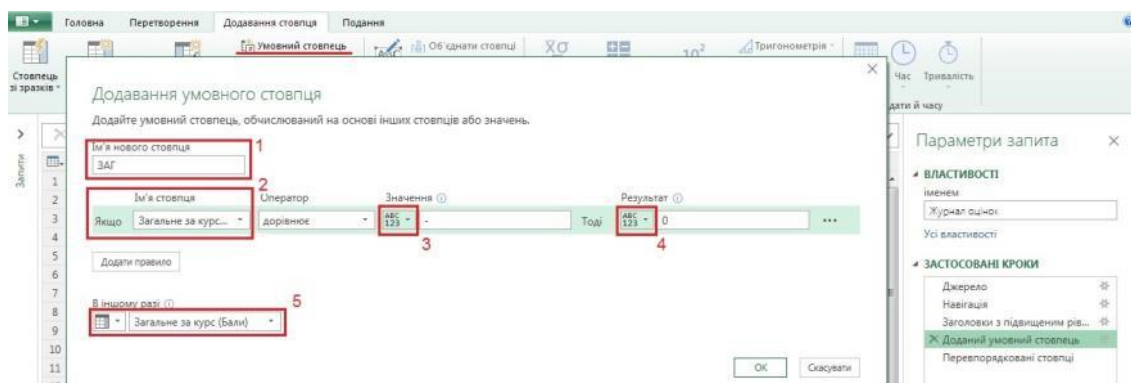


Рисунок 4. Навігатор додавання умовного стовбця (використано для узгодження форматів даних)

- блок 1: введення заголовку стовпця (його можна змінити за необхідністю, шляхом натискання правою кнопкою миші на найменуванні стовпця та вибору «Перейменувати»)
- блок 2: найменування початкового стовпця, значення якого порівнюється з правилом, яке встановлюється вибором оператора із списку, який розташовано з права (для даної задачі обрано «дорівнює»);
- блок 3: зазначається тип даних в умові та його значення, з яким порівнюються комірки стовпця з блоку 2 (відповідно до роботи порталу знак «-» відповідає відсутності оцінки за вид діяльності у користувача);
- блок 4: зазначається тип даних та його значення, що замінює знак з блоку 3 (у нашому випадку коротке тире «-», що відповідає «0» балів за певний вид діяльності або завдання);
- блок 5: визначається тип даних або джерело з якого додаються данні у випадку не виконання умови зазначеної у блоках 2 та 3 (у нашому випадку, це наявність оцінок у колонці яка вказана у блоці 2, які необхідно залишити без змін).

За представленими блоками необхідно обробити необхідні або усі експортовані стовпці. Усі операції будуть відображені як «Застосовані кроки» у вікні «Параметри запиту» з правої сторони (рис. 4). Після чого, виділити додані стовпці, на будь-якому найменуванні стовпців викликати контекстне меню та змінити їх тип на «Десяткове число» (рис. 5). При правильному виконанні заміни текстових значень тип буде відображено біля найменувань стовпців (рис. 5, блоки 1 та 2), а у контекстному меню біля типу «Десяткове число» з'явиться відповідна позначка (рис. 5, блок 3).

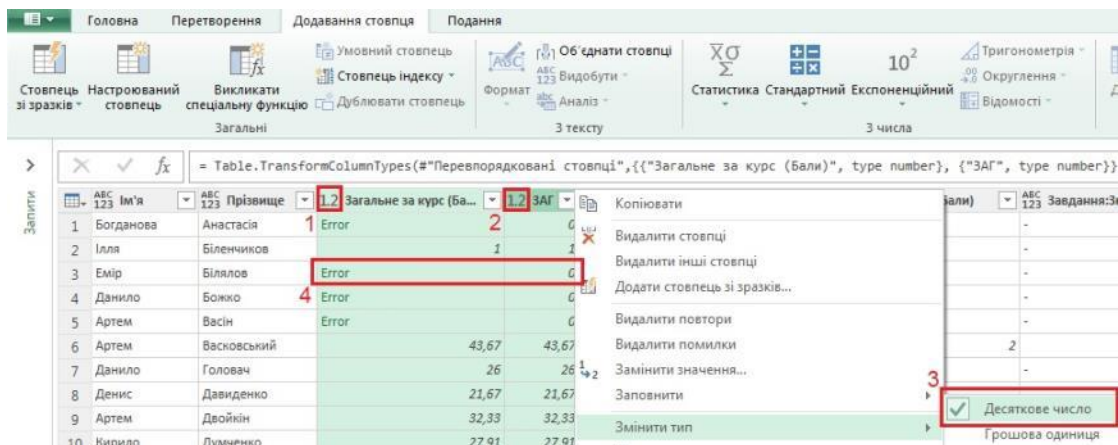


Рисунок 5. Процедура зміни типу експортованих даних

У випадку наявності помилок у доданих стовпцях (рис. 5, блок 4) необхідно звернути увагу на відповідну колонку експортованого файлу з метою пошуку інших текстових символів крім короткого тире «-». Такі помилки можуть виникнути при ручному заповненні журналу оцінок на освітньому порталі. Якщо доцільно, то необхідно у журналі оцінок освітнього порталу зробити відповідні коригування.

При відсутності помилок, необхідно виділити початкові дані та видалити стовпці початкових даних. Таким чином, залишаться тільки стовпці необхідні для загального журналу оцінок.

Етап 3.2. Узгодження оцінок за виконання завдання із структурою навчальної дисципліни з робочої програми. При створенні завдань на освітньому порталі (тести, звіти та ін.) бажано вказувати кількість балів за його виконання. Дуже часто оцінка залишається за замовченням, а саме: виконано чи не виконано («1» або «0») або 100%. Це не відповідає структурі дисципліни. Тому необхідно додати стандартну арифметичну операцію (рис. 6). Для представленого прикладу (рис. 6): 1 – меню «Перетворення»; 2 – група «Група числовий стовпець», блок «Стандартний»; 3 – функція операція «Помножити». Наступним кроком буде введення коефіцієнту множення через інтуїтивно зрозуміле діалогове вікно.

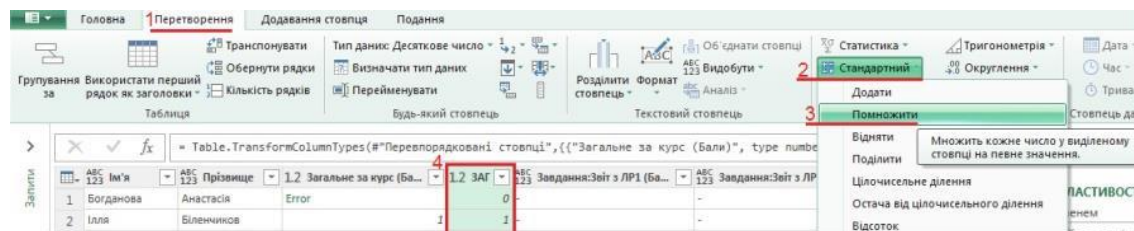
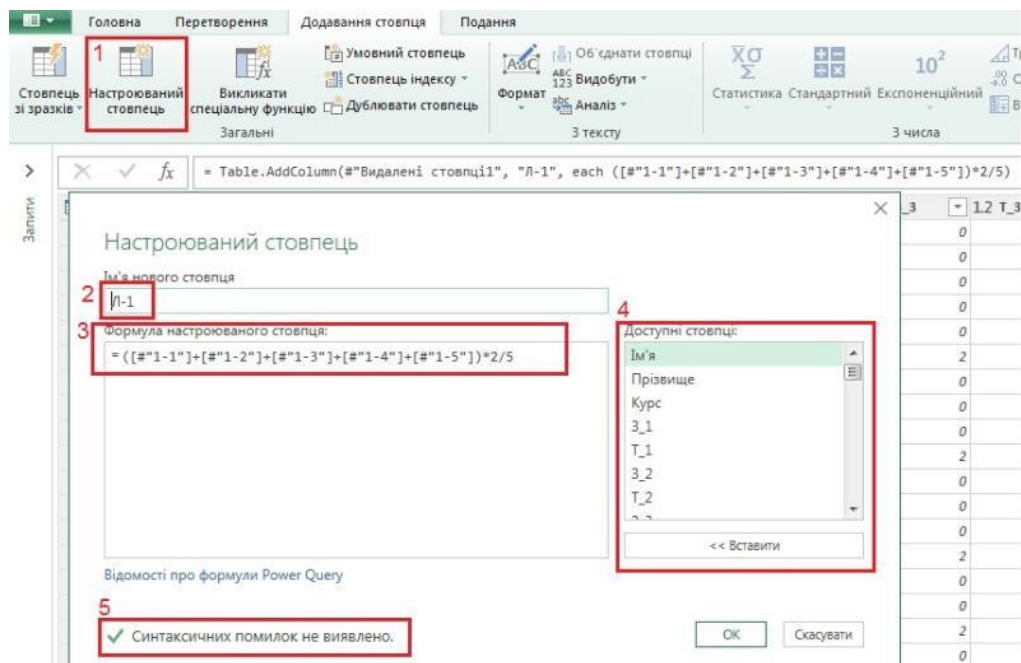


Рисунок 6. Стандартні арифметичні операції без додавання нового стовпця

При формуванні запиту є можливість створення додаткових стовпців, які будуть обчислювати певний загальний показник. Для прикладу (рис. 7): є 5 тестових завдань по кожному пункту лекційного матеріалу, котрі оцінюються у діапазоні [0, 1], необхідно визначити середню оцінку по лекційному матеріалу, враховуючи, що за робочою програмою максимальна оцінка складає 2 бали.

Для цього додаємо настроюваний стовпець (рис. 7, блок 1). Вказуємо ім'я нового стовпця (рис. 7, блок 2), яке можна коригувати за необхідністю. За допомогою менеджера стовпців (рис. 7, блок 4) генеруємо формулу обчислення загальної оцінки (рис. 7, блок 3). При відсутності помилок у виразі - буде наявна

відповідна позначка (рис. 7, блок 5). Аналогічним чином можна групувати оцінки за модулями, практичними та лабораторними роботами.



без колонки належності до певної групи (у налаштуваннях користувача додати таке поле не є можливим);

- при видаленні або додаванні тестових завдань їх необхідно також додати або видалити до параметрів запиту або видалити з усіх кроків запиту;
- при зміні назви тесту або форми контролю, необхідно шукати відповідний крок параметру запиту та корегувати його зміст;
- не можливо автоматично масштабувати запит на інші дисципліни;
- відсутня можливість експорту результатів запиту з освітнього порталу до електронного журналу.

Перший недолік можливо виправити, якщо використовувати довідкові таблиці із списком студентів кожної групи та функцій Excel з категорії «Посилання та масиви».

Список використаних джерел:

1. What is Power Query? URL: <https://learn.microsoft.com/en-us/power-query/power-query-what-is-power-query>
2. Бондаренко Л.Ю., Вершков О.О., Караєв О.Г., Холодняк Ю.В., Гавриленко Є.А. Використання Zoom як додаткової платформи для навчання під час воєнних дій на території України. Удосконалення освітньо-виховного процесу в закладі вищої освіти. Мелітополь: ТДАТУ, 2022. Вип. 25. С. 64-69.
3. Лубко Д.В. Інформаційна система кафедри вишу з підсистемою підтримки контролю дистанційного навчання та обліку успішності / Д.В. Лубко, Ю.О. Литвин // Фундаментальна підготовка фахівців у природничо-математичній, технічній, агротехнологічній та економічній галузях. Матеріали Всеукраїнської науково-практичної конференції з міжнародною участю. (11-13 вересня). – Мелітополь: ТОВ „Колор Принт”, 2017. С. 85-87.;
4. Освітній портал Таврійського державного агротехнологічного університету URL: <http://op.tsatu.edu.ua>
5. Прийма С.М., Строкань О.В., Лубко Д.В., Литвин Ю.О., Білик Н.І. Розробка інформаційної системи семантичної ідентифікації і документування результатів неформального й інформального навчання. Проблеми програмування: Інформаційні системи. 2020. № 2-3. С. 11-21.
6. Стаття «Валідація результатів навчання». Автори: Прийма С.М., Аніщенко О.В. (свідectvo про реєстрацію авторського права на твір № 110833 від 06.01.2022 р.).

УДК: 005.4

ІСТОРІЯ УПРАВЛІННЯ ПРОЄКТАМИ

Кравченко К.Р.

e-mail: uitpm221_kkr@student.ztu.edu.ua

Науковий керівник: Вакалюк Т.А., д. пед. н., професор
Державний університет "Житомирська політехніка"

Управління проєктами існує протягом декількох тисяч років. Зрештою, все, що вимагає підходу, коли люди ефективно організовуються за планом і досягають конкретних цілей, можна умовно визначити як проєкт. Історія управління – це історія мегапроєктів за останні 4500 років, серед яких піраміда Гізи, Парфенон, Колізей, готичні собори Європи, Тадж-Махал і Трансконтинентальна залізниця. Це не були аномалії в історії, а проєкти, які реалізовувалися систематично з характеристиками, дуже схожими на сучасні.

До 1900 року проєктами будівництва та цивільної інженерії керували архітектори, інженери та майстри-будівельники, наприклад Вітрувій (перше століття до нашої ери), Крістофер Рен (1632–1723), Томас Телфорд (1757–1834) та Ізамбард Кінгомд Брунель (1806). –1859). Саме в 1950-х роках організації почали систематично застосовувати інструменти та методи управління для складних інженерних проєктів.[1]

Видатною особистістю у галузі управління проєктами є Генрі Гант (1861–1919), який вважається батьком сучасного планування та контролю техніки.

У 1884 році він почав працювати креслярем на чавуноливарній та машинній майстерні Poole & Hunt у Балтиморі. У 1887 році він приєднався до Фредеріка В. Тейлора у застосуванні принципів наукового менеджменту до роботи в Midvale Steel і Bethlehem Steel, працюючи там з Тейлором до 1893. У своїй подальшій кар'єрі консультанта з питань управління та після винаходу діаграми Ганта, він розробив систему оплати праці «завдання та бонуси» та додаткові методи вимірювання ефективності й продуктивності робітників. У 1916 році під впливом Торстена Веблена він створив асоціацію, яка прагнула застосувати усуперечення для промислового виробництва під контролем менеджерів.[2]

1950-ті роки ознаменували початок сучасної ери управління проєктами, коли ключові галузі інженерії об'єднуються, щоб працювати як одне ціле, тому управління проєктами було визнано окремою дисципліною. У Сполучених Штатах до 1950-х років управління здійснювалося при використанні переважно діаграми Ганта. У той час було розроблено дві математичні моделі планування проєктів. «Метод критичного шляху» (CPM) був розроблений як спільне підприємство DuPont Corporation і Remington Rand Corporation для управління проєктами технічного обслуговування заводів. А «Техніка оцінки та перегляду програми» або PERT була розроблена Буз Аллен Гамільтон у рамках програми ракетних підводних човнів Polaris ВМС США (спільно з корпорацією Lockheed).[3]

У програмах управління проєктами 1980-х років здебільшого ґрунтувалися на моделі Projects Resource Management Planning Technique II (PROMPT II), яка пізніше була вдосконалена та стала називатись PROjects In Controlled Environments (PRINCE).[4]

Іншою важливою подією того часу стала теорія обмежень (ТОС), яка є філософією управління, представленою Еліяху М. Голдраттом. Філософія спрямована на те, щоб допомогти організаціям постійно досягати своїх цілей,

використовуючи передумову, що швидкість досягнення мети цільовою системою обмежена принаймні одним обмеженням. [5]

У 1987 році був опублікований PMBOK®, книга здебільшого ґрунтувалася на книзі, опублікованій у 1983 році під назвою «Заключний звіт комітету з етики, стандартів і акредитації». Метою книги було задокументувати, стандартизувати правила управління проєктами, з урахуванням досвіду фахівців. Посібник PMBOK® став світовим стандартом для цієї галузі.[6]

Розуміння минулого дає нам шанс краще зрозуміти майбутнє. Вивчаючи історію, можна зрозуміти, що управління проєктами розвивалося тисячоліттями. Тому дана сфера є фундаментом для створення кожного майбутнього відкриття, яке в змозі змінити весь світ.

Список використаних джерел:

1. Y.C. Chiu: An Introduction to the History of Project Management: From the Earliest Times to A.D.1900, 2010.
2. Young Hoon Kwak, Frank T. Anbari: The Story of Managing Projects: An Interdisciplinary Approach, 2005.
3. Mark Kozak-Holland: The The History Of Project Management (Lessons from History), 2011.
4. Dr. Mark Foster: Academy of Business Research Journal Volume III, 2012.
5. Детмер Уильям: Теория Ограничений Голдрата, Системный подход к непрерывному совершенствованию, 2015.
6. Project Management Institute (Global Standard): A Guide to the Project Management Body of Knowledge (PMBOK® Guide) 7th edition, 2021.

УДК 378.147.227

ПЕРСПЕКТИВНІСТЬ ПРОВЕДЕННЯ КОМП'ЮТЕРНОГО АВТОМАТИЗОВАНОГО ТЕСТОВОГО КОНТРОЛЮ

Лубко Д.В., к.т.н.

e-mail: dmytro.lubko@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Реформування освіти в нинішніх умовах становить чимало нових завдань, пов'язаних, насамперед, з покращенням якості підготовки фахівців для дошкільних навчальних закладів. Одним із шляхів формування якості професіоналізму майбутніх педагогів є створення і вдосконалення стимулюючої системи педагогічного контролю навчальної роботи студентів. Актуальність питань теорії і практики тестового контролю в теперішній час зумовлена вимогами модернізації вищої освіти в Україні в контексті Болонського процесу, коли «новими концепціями освітнього процесу у вищій школі позначаються тенденції розвитку виміру навчальних досягнень за допомогою тестів» [1].

Порівняно з традиційними формами контролю, комп'ютерне тестування має ряд переваг [2]: швидке отримання результатів випробування і звільнення викладача від трудомісткої роботи з обробки результатів тестування; - об'єктивність оцінки; - тестування на комп'ютері цікавіше порівняно з традиційними формами опитування, що створює позитивну мотивацію у студентів; підвищення ефективності роботи викладацького складу [3].

Значення тестів як інструменту оцінювання ефективності самостійної роботи порівняно з іншими формами контролю полягає в тому, що, по-перше, тести значно якісніший і об'єктивніший спосіб оцінювання і, по-друге, показники тестів орієнтовані на вимірювання ступеня, визначення рівня засвоєння ключових понять, тем і розділів навчальної програми, умінь і навичок, а не на констатацію наявності у студента певної сукупності формально засвоєних знань [5].

В свій час питаннями розробки і впровадження тестових технологій займалися і досліджувались у наукових працях таких науковців В.Аванесов, В.Євдокимов, В.Лозова, І.Прокопенко, Л.Гриневич та інших.

Проведений аналіз наукової літератури (В.С. Аванесов, В.Г. Кузнецов, А.Н. Майоров, тощо) показав [4, 7, 8], що основними умовами, необхідними для організації ефективного автоматизованого тестового контролю є:

- врахування класичної і сучасної тестової теорії: тільки на базі тестової теорії і сучасних методик розробки тестів можна забезпечити надійність, валідність і ефективність контролю; важливо також, щоб тестовий контроль не зводився виключно до перевірки знань студентів; у процесі тестового контролю велике значення має комплексна перевірка всієї навчальної діяльності студента, зокрема динаміки його загального розвитку, формування спеціальних умінь і навичок, активності, пізнавальних інтересів, творчих здібностей;

- професійно зацікавлене, творче ставлення викладача до організації і управління навчальним процесом. На думку В. С. Аванесова, «тести можуть бути ефективними тільки в такому навчальному процесі, в якому викладач перетворюється на розробника нових програмно-методичних засобів, в організатора процесу самостійного навчання студентів. Навчання має починатися

з вхідного тестового контролю, супроводжуватися самоконтролем і закінчуватися підсумковим тестуванням”;

- активна участь студентів в організації і здійсненні власної навчально-пізнавальної діяльності в процесі навчання шляхом самоконтролю за допомогою тестування результатів своєї навчальної роботи.

Саме тому розгляд перспективності проведення комп'ютерного тестового контролю на сьогодні є актуальним завданням для усіх ВНЗ країни.

Основні матеріали дослідження. Система навчання у вищій школі – це багатогранний процес, який складається з цілого ряду взаємопов'язаних елементів. Серед них важливе місце посідає контроль знань, тобто організація зворотного зв'язку як засобу управління навчально-виховним процесом. Ця проблема особливо актуальна у наш час у зв'язку з тим, що вся система вищої освіти в Україні підлягає повній організаційній перебудові. Перевірка і оцінка знань виконують вісім функцій: контрольну, навчальну, виховну, організаторську, розвиваючу і методичну, діагностичну, стимулюючу.

Термін “тестування” (від англійського слова testing – випробування) уперше використав Дж. Фішер для перевірки рівня знань, умінь та навичок учнів за допомогою оригінальних спеціальних книг (scalebooks), які з'явилися в 1864 р. у Великобританії. На сьогодні тестування в системі освіти знаходиться в стані постійного вдосконалення і періодичних змін напрацьованих і широко апробованих методик.

Тести виконують такі навчальні функції, як: коригування та вдосконалення контрольованого матеріалу; формування мовленнєвих навичок та вмінь; стимулювання розвитку пам'яті та мислення.

Специфічні характеристики тесту дозволяють визначити генеральну задачу тестового контролю – перевірку відповідності сформованих якостей вчителя ВНЗ цілям навчання на визначеному етапі підготовки, аналіз і розробку студентом і викладачем на цій основі відповідних рішень по вдосконаленню навчання. Таким чином, за допомогою тестів відбувається контроль за рівнем засвоєння знань, рівнем опанування способів діяльності, рівнем професійного становлення.

У навчальному процесі у ВНЗ найчастіше застосовується вихідний, поточний та підсумковий контроль.

Вихідний контроль має на меті визначити початковий рівень знань студентів і дібрати відповідну літературу. Цей вид тестування, як правило, проводиться на вступній лекції [1].

За результатами поточного контролю викладач оцінює ефективність своєї методичної майстерності та доцільності вивчення педагогічного досвіду минулих поколінь, а студенти мають можливість спостерігати за власним процесом у засвоєнні навчальної програми з певної дисципліни.

При цьому слід наголосити, що результати поточного тестування не можуть бути використані для офіційної оцінки рівня сформованості вмінь студента, оскільки поточний контроль спрямовується на визначення проміжних, а не кінцевих результатів. Під час поточного контролю тести застосовуються в основному на аудиторних заняттях. На основі отриманої інформації викладачем здійснюється коригування навчальної діяльності студентів. Тести за такого виду контролю стимулюють у студентів прагнення до систематичної самостійної роботи, виконання завдань, підвищення інтересу до навчання, формування почуття відповідальності. Обов'язковим видом контролю навчальних досягнень є підсумковий тест.

За допомогою підсумкового тестового контролю вивчається рівень володіння студентами навчального матеріалу з певної дисципліни в межах теми, що вивчається. Його мета – встановити успішність оволодіння системою знань та умінь з окремої теми або розділу, відповідність рівня їх засвоєння програмовим вимогам. За результатами цього тестування студентам виставляється оцінка [1].

Встановлено, що тестовий контроль знань за допомогою комп'ютера має (в порівнянні з традиційними методиками) такі переваги, як:

1. Висока технологічність (операція тестового контролю знань є набором нескладних дій з схеми "питання-відповідь-аналіз відповіді, що дозволяє розробити досить прості і універсальні програмні засоби контролю);

2. Скорочення тимчасових витрат, можливість не лише швидкої і одночасної перевірки знань у великих груп студентів, але і включення великого об'єму навчального матеріалу; доступність результатів тестування відразу після закінчення теста;

3. Відповідність вимогам максимальної об'єктивності контролю, оскільки усі учні опиняються в рівних умовах (час, об'єм і складність матеріалу однакові для усіх, що виключає вплив такого чинника, як везіння / невдача);

4. Можливість застосування в усіх видах контролю, у тому числі і для самоконтролю студентів та оцінювання ефективності функціонування освітнього середовища в цілому;

5. Придатність для використання в друкованому варіанті, тобто для проведення традиційного контролю;

6. Практично повне виключення можливості використання шпаргалок (обмеження часу на відповідь, розміщення завдань для кожного користувача в різному порядку).

До переваг тестового опитування слід віднести:

- велику об'єктивність і, як наслідок більша позитивна стимулююча дія на пізнавальну діяльність студента;

- виключається дія негативного впливу на результати тестування таких чинників як настрій, рівень кваліфікації та інші характеристики конкретного викладача;

- орієнтованість на сучасні технічні засоби на використання в середовищі комп'ютерних (автоматизованих) навчальних систем;

- універсальність, охоплення усіх стадій процесу навчання.

Крім того, тестове опитування багатofункціональне. Воно дозволяє швидше зрозуміти, як далі працювати з цим студентом, а також допомагає лекторові скоригувати курс.

Значною перевагою тестування є можливість широкого охоплення матеріалу в порівнянні з усним опитуванням (у тести можна включити багато питань, що охоплюють усю тематику), що дає змогу оцінювати знання кожного студента не за одне конкретне питання, а за знання всієї теми [9].

Серед позитивних сторін слід указати на те, що тестування на кожному занятті спонукає студентів до систематичного вивчення матеріалу, оскільки отримання негативної оцінки призводить до зниження середнього бала і, відповідно, рейтингу студента.

До переваг тестування також належать: однакові критерії оцінювання для всіх студентів; відсутність психоемоційного напруження студентів, що може виникати на усному опитуванні; об'єктивність оцінки результатів тестування (в порівнянні з усною відповіддю); стандартизованість і уніфікованість тестового контролю.

Однак тестування, крім перерахованих переваг, має певні недоліки. До недоліків слід віднести досить часте використання некоректно складених тестів або тестів, що не повною мірою висвітлюють тему, яка вивчається. Якість складання тестових завдань залежить від викладача, його знань даного предмета, вміння правильно подати матеріал і побудувати тестове завдання.

Викладачі, готуючи завдання для тестового контролю навчальних досягнень студентів, мають дотримуватися таких основних правил:

- з метою максимального забезпечення самостійного виконання завдань розробляється достатня кількість варіантів (їх має бути не менше 4);
- варіант містить достатню кількість запитань (для поточного контролю – 6-8, для підсумкового – 10-20);
- у завдання не включаються запитання, що повторюють формулювання підручника (за винятком дефініцій понять, формулювання положень теорій чи законів);
- завдання у варіанті не мають бути однотипними (у кожному варіанті слід передбачити виконання різних за складністю завдань);
- недоцільно включати відповіді, неправильність яких на час тестування студент не може обґрунтувати.

Заліки, іспити, колоквіуми, тестування виконують переважно діагностичну функцію контролю. При застосуванні програмованого контролю проявляється його навчальна та контрольна функції [10].

Контроль знань студентів – це невід'ємна і важлива частина процесу навчання, відповідальний етап на шляху від незнання до знання, від неповного знання до більш точного і більш повного.

Кожному викладачеві необхідно напружувати таку систему контролю знань, щоб у цілому вона найкращим чином слугувала виконанню всіх його функцій.

Завдання викладача і студентів полягає у тому, щоб у процесі перевірки оцінки знань виявити справжній стан знань, умінь і навичок і тим самим допомогти студентам раціонально організувати навчальну роботу в подальшому. Успіх вирішення цього завдання безпосередньо залежить від суворого дотримання викладачем дидактичних принципів контролю знань.

Виявлення, контроль і оцінка знань студентів – важлива проблема теорії і практики навчання. Без перевірки або самоперевірки засвоєних знань, набутих умінь і навичок неможливе якісне здійснення цієї проблеми. Тому контроль знань студентів завжди був, є і буде важливою складовою частиною навчального процесу, хоч і ставлення до нього зазнавало певних змін. Міняються окремі форми і способи контролю знань, але його головна суть – знати, наскільки вдало відбувся процес засвоєння вивченого матеріалу, – залишається незмінною. Вона визначається самою природою процесу навчання. Проблема ефективної організації контролю під час навчання є однією з найактуальніших у методиці викладання іноземних мов. Контроль як органічну складову навчального процесу більшість методистів розглядає як дидактичний засіб управління навчанням, мета якого – “забезпечити ефективність навчання шляхом приведення до системи знань, умінь, навичок студентів, самостійного застосування ними здобутих знань на практиці” [10]. У його завдання входить також стимулювати студентів старанно навчатися, формувати в них прагнення до самоосвіти.

Результати тестів вказують на недоліки або досягнення в роботі викладача зі студентами, допомагають передбачити проблемні ситуації у вивченні певних тем та спрямувати зусилля у правильне русло для отримання високих результатів.

Використання тестів у процесі навчання є одним із раціональних доповнень до методів перевірки знань, вмінь та навичок студентів. Тести є також відмінним засобом індивідуалізації навчання, оскільки враховують психологічні особливості студентів.

Тестування є одночасно і навчальною вправою, і засобом контролю. Використання тестових завдань як засобу навчання є ефективним і виправданим з точки зору дидактики та психології. Чітка регламентованість процедури тестування та наявність еталона дозволяють оптимізувати навчальний процес, а спрямованість на активну розумову діяльність та об'єктивність результатів створюють у студентів позитивне ставлення до певної дисципліни

Висновок. Підсумовуючи вищезазначене, можна стверджувати, що застосування комп'ютерного тестового контролю для оцінювання знань, умінь і навичок студентів забезпечує високу технологічність проведення контролю та об'єктивність його результатів, дозволяє використовувати гнучку шкалу оцінювання знань. Водночас систематичне комп'ютерне тестування є ефективним засобом підвищення мотивації студентів до системного активного самостійного навчання впродовж семестру та навчального року, підвищення рівня індивідуалізації навчання, тобто є ефективним засобом адаптації навчального процесу до вимог, визначених Європейською системою залікових кредитів (ECTS), особливостей кредитно-модульної системи організації навчального процесу (КМСОНП). Однією з найважливіших функцій комп'ютерного тестування є функція забезпечення зворотного зв'язку в процесі самостійного навчання, що створює можливість гнучкого управління цим процесом.

Організація успішного впровадження тестової технології контролю потребує в першу чергу відповідної теоретико-методичної підготовки викладачів та методологічної компетентності студентів. Вона пов'язана з необхідністю використання студентами професійно-орієнтованих наукових знань з педагогіки, психології, інформаційних технологій для аналізу і вдосконалення своєї роботи, самостійного вирішення різноманітних навчальних завдань. В свою чергу, реформування навчального процесу, збільшення частки самостійної та індивідуальної роботи, зміна форм контролю знань потребує якісно нового науково-методичного забезпечення.

Подальшого дослідження потребує з'ясування додаткових умов організації самостійної роботи в паралель проведенню комп'ютерного автоматизованого тестового контролю. А ця суміжність цих двох напрямків набуває особливого значення для підготовки висококваліфікованих спеціалістів в умовах функціонування кредитних технологій навчання.

Список використаних джерел:

1. Танько Т.П. Тестовий контроль знань студентів як умова модернізації освіти в Україні. Витоки педагогічної майстерності. Серія: Педагогічні науки, №7, 2010. С. 14-18.
2. Аванесов В.С. Композиция тестовых заданий. Учебная книга для преподавателей вузов, учителей школ, аспирантов и студентов педвузов. – 2 изд., испр. и доп. – М.: Адепт, 1998.
3. Козаков В.А. та інші. Рейтингова система оцінки успішності навчання студентів: Збірник. – УМК ВО, 1992.
4. Шимкова І.В. Використання автоматизованого тестового контролю знань для організації самостійної навчально-пізнавальної діяльності студентів. Збірник наукових праць «Педагогічні науки», №46, 2007. С. 408-411.

3. Алексюк А. М. Педагогіка вищої освіти України: Історія. Теорія: [підручник для студентів, аспірантів та молодих викладачів вузів] / Міжнародний фонд «Відродження»; А. М. Алексюк. – К. : Либідь, 1998. – 558 с.
4. Грабар Е. В. Історичні передумови виникнення та розвиток форми тестового контролю навчальних досягнень учнів у педагогіці США / Е. В. Грабар // Освіта Донбасу. – 2010. – № 2 (139). – С. 105–110.
5. Майоров А.Н. Тесты школьных достижений: конструирование, проведение, использование. – С-Пб.: Образование и культура, 1997. – 304 с.
6. Булах І. Є. Історія розвитку та сучасний стан педагогічної тестології / І. Є. Булах. – К. : ЦМК МОЗ України, 1994. – 21 с.
7. Lubko D.V., Zolotukhin Y.O. Розробка експертної системи для вивчення дисципліни «Проектування інформаційних систем». Ukrainian Journal of Educational Studies and Information Technology, Volume 5(1), 2017, p. 182-186.
8. Шаров С.В. Використання адаптивного тестування при тематичному контролі знань учнів. Теорія і методика навчання інформатики та математики: зб. наук. пр. Том 3. 2004. – С. 169-171.
9. Колісник І.А. Тестовий контроль студентів: переваги і недоліки. Український стоматологічний альманах, №1, 2017. С. 83-85.
10. Гарматюк Н.Д., Марценюк В. П. Особливості застосування тестового контролю при вивченні іноземної мови у вищих навчальних закладах. *Медична освіта*, №3, 2013. С. 17-24.

УДК 378.14:004.02

ПЕРЕВАГИ ЗМІШАНОГО НАВЧАННЯ В ЗАКЛАДАХ ВИЩОЇ ОСВІТИ В СУЧАСНИХ РЕАЛІЯХ (НА ПРИКЛАДІ ЛДУ БЖД)

Полотай О.І., к.т.н., доцент

email: orest.polotaj@gmail.com

Львівський державний університет безпеки життєдіяльності

Актуальність та постановка проблеми. Львівський державний університет безпеки життєдіяльності (ЛДУ БЖД) – провідний український заклад вищої освіти в сфері безпеки людини. Належить він до державної служби України з надзвичайних ситуацій. Всі учасники навчального процесу поділяються на дві групи: студенти та курсанти. Для курсантів, навчання в університеті аналогічне службі в армії, виконання нарядів і одночасно власне навчання.

На жаль, так склалося, що останні пару років процес навчання підпадає під вплив непередбачуваних форс-мажорних обставин. Пандемія COVID-19 активізувала пошук нових методів та засобів навчання. Низка нормативно-правових документів з весни 2020 р., зокрема: Постанова Кабінету Міністрів України «Про запобігання поширенню на території України гострої респіраторної хвороби COVID-19, спричиненої коронавірусом SARS-CoV-2» (від 11.03.2020 №211, від 02.04.2020 №255); Закон України «Про внесення змін до деяких законодавчих актів України, спрямованих на забезпечення додаткових соціальних та економічних гарантій у зв'язку з поширенням коронавірусної хвороби (COVID-19)» (від 30.03.2020 №540-IX), Наказ Міністерства освіти і науки України «Про організаційні заходи для запобігання поширенню коронавірусу COVID-19» (від 16.03.2020 №406) та інші змінили вектор діяльності закладів освіти в Україні. Також, свої корективи внесла і війна в Україні.

Виходячи з цього, виникає потреба у змішаній формі навчання, яка би містила в собі елементи дистанційного навчання.

Отже, доцільним є опис віртуального навчального середовища ЛДУ БЖД, яке використовується як допоміжний засіб організації змішаного навчання.

Основні матеріали дослідження. Віртуальне навчальне середовище ЛДУ БЖД [2], має назву Віртуальний університет ЛДУ БЖД (ВУ ЛДУ БЖД) та працює він на базі оболонки Moodle.

Перед початком роботи з даною системою було організовано процес створення електронних курсів. Для цього було проведено навчання відповідальних працівників кафедри, деканату та за бажанням викладачів-тьюторів на семінарах тренінгах. Було проведено два семінари тренінги, щодо роботи навчально-педагогічного персоналу кафедр. Також було розроблено два допоміжні електронні курси, один з яких був зразком для створення нових курсів, інший для онлайн навчання майбутніх викладачів-тьюторів створювати власні електронні курси.

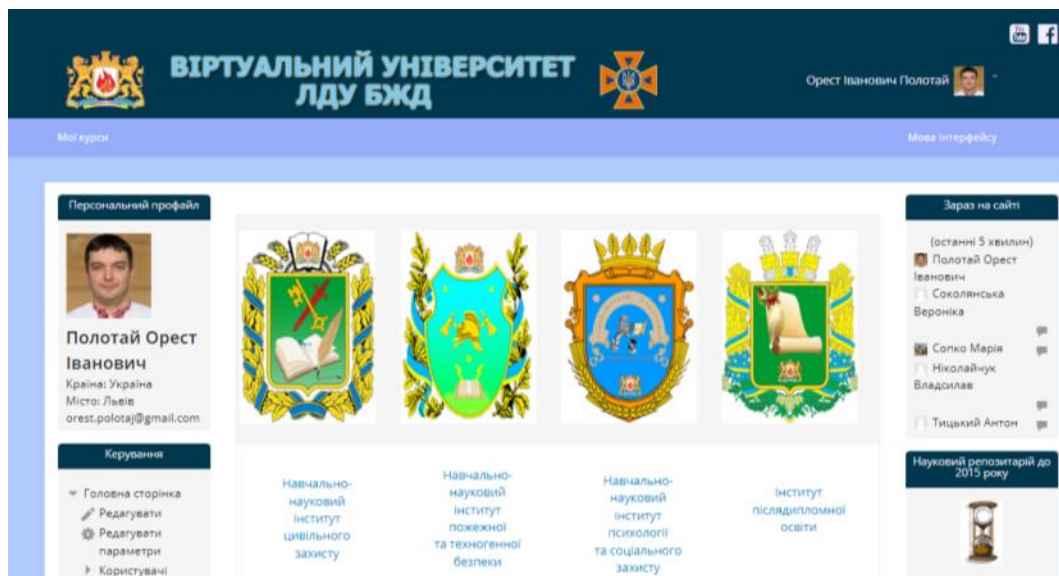


Рисунок 1. Головна сторінка навчального середовища ЛДУ БЖД

Для покращення засвоєння викладеного на семінарах-тренінгах було опубліковано посібник для викладача, щодо роботи у ВУ ЛДУ БЖД [4].

В процесі запровадження змішаного навчання у ЛДУ БЖД було узгоджено структуру електронного курсу.

Структура ЕК включає такі елементи:

Інструктивний блок: назва ЕК; Мета ЕК; Передмова курсу; Зворотній зв'язок; Робоча програма; Навчальна програма; Глосарій; Е-посібники та додаткові електронні ресурси; Рекомендована література.

Теоретична частина:

- електронний конспект лекцій;
- мультимедійні презентації лекцій;
- за потреби аудіо-, відео-, анімаційні навчальні ресурси;
- інші додаткові матеріали;

Практична частина:

–завдання на практичні, лабораторні, самостійні та розрахунково-графічні роботи (в залежності від специфіки курсу), перелік індивідуальних варіантів завдань;

–методичні вказівки для виконання практичних, лабораторних, самостійних чи розрахунково-графічних робіт (в залежності від специфіки курсу);

Контрольна частина:

–виконання практичних, лабораторних, самостійних чи розрахунково-графічних робіт (в залежності від специфіки курсу);

–тестування.

Підсумковий контроль:

–Тестування;

–Перелік питань на екзамен чи диференційований залік.

Детальніше опис кожного елементу електронного курсу описано нижче.

Загальна інформація про курс. Розташовується в інструктивному блоці (нумерованому) розділі курсу:

Назва курсу (відповідно до назви дисципліни);

Передмова курсу містить стислу інформацію про дисципліну, з вказанням

обсягу дисципліни, форми навчання, форми контролю, викладачів курсу.

Робоча програма. Робоча програма містить: назву та загальну характеристику навчальної дисципліни, яка включає кількість годин (кредитів); шифр та назву напрямку та спеціальності; кількість модулів разом з самостійною та індивідуальною роботою студента; кількість семестрів викладання дисципліни; кількість змістових модулів; загальну кількість годин, передбачених навчальним планом для лекційних, лабораторних, практичних та інших видів робіт; види контролю по семестрах тощо.

Е-посібник. Посібник з відповідної дисципліни в електронному форматі, автором якого є розробник та викладач ЕК.

Глосарій. Містить основні терміни, поняття дистанційного курсу та їх визначення.

Рекомендована література. У цьому ресурсі пропонуються основні, додаткові друковані джерела з дисципліни.

Теоретична (лекційна) частина. Містить обов'язкові навчальні ресурси, які направлені на засвоєння теоретичних основ за темами навчального модуля та відображає логіку навчання за курсом і надає студенту теоретичні відомості з модуля у повному обсязі.

Крім адаптованого до самостійного вивчення статичного теоретичного матеріалу, тут можна розміщувати мультимедійні презентації лекцій, аудіо- та відео-матеріали, призначені для вивчення теоретичної частини курсу, флеш-ролики; довідкові та нормативні документи (форми, шаблони, стандарти, нормативні акти, закони) тощо.

Практична частина. Містить завдання методичні вказівки для виконання практичних, лабораторних, самостійних чи розрахунково-графічних робіт, які охоплюють навчальну діяльність курсанта та студента за участю викладача, направлену на вирішення практичних завдань згідно тем навчального модуля. Сюди можуть входити лабораторні та практичні заняття, які виконуються з участю викладача, семінари, дискусії тощо. Дані матеріали необхідно розробляти у форматі *.pdf, та за необхідності у інших форматах (*.xls, *.jpeg та ін).

До кожного завдання необхідно вказати часові рамки для виконання завдання та максимальну кількість балів, які курсанти та студенти зможуть отримати.

Контрольна частина. Містить форму для передачі виконаних завдань на перевірку викладачу та електронне тестування по матеріалах відповідної теми. Результат виконання лабораторної (практичної) роботи студенти можуть надіслати викладачу-тьютору в електронній формі.

Після перевірки та оцінювання виконаних завдань, викладач-тьютор має виставити бали до журналу оцінок ЕК та надати до кожної надісланої роботи коментар та зауваження. Бали за виконання тестування виставляються в журнал ЕК автоматично. Максимальну кількість балів за тестування вказує викладач.

Підсумковий контроль – передбачає наявність матеріалів для підготовки студентів до складання заліків та іспитів (контрольні завдання для практичного виконання або підсумковий тест, який включає в себе випадкову вибірку з питань тестів поточних модульних робіт, або одночасно декілька видів контролю). Також можливе проведення підсумкового контролю шляхом спілкування студента та викладача-тьютора в режимі он-лайн, за умови наявності переліку питань для підсумкового контролю.

Важливим кроком при підготовці віртуального середовища до роботи в умовах змішаного навчання є розподіл обов'язків учасників та їх можливості у

ВУ ДЖУ БЖД. Отже, було визначено такі групи користувачів:

1. Адміністратор сайту:
 - реєстрація нових користувачів, видалення старих користувачів;
 - створення резервних копій курсів;
 - відновлення курсів із резервної копії;
 - міграція курсів між спеціальностями та навчально-науковими інститутами згідно навчальних планів;
 - ведення глобальної статистики активності користувачів;
 - при необхідності «чистка» курсів в кінці навчального року від зайвих користувачів;
 - часткова модернізація сайту;
 - надання консультативної допомоги користувачам ВУ ЛДУ БЖД.
2. Адміністратор серверу (технічний):
 - періодичне створення резервних копій сайту на сервері.
 - періодичне оновлення версій Moodle.
 - слідкування за технічним станом серверу та його програмно-апаратним забезпеченням.
3. Керівництво трьох навчально-наукових інститутів (ННІ/факультетів):
 - моніторинг стану всіх курсів в межах інститутів;
 - моніторинг роботи з курсами.
4. Відповідальний працівник деканату:
 - збір даних про нових курсантів/студентів 1 курсу ННІ та передача цієї інформації адміністратору сайту;
 - моніторинг змін у навчальних планах та передачах цієї інформації адміністратору сайту.
5. Відповідальний працівник спеціальності:
 - створення нових курсів в межах спеціальності;
 - моніторинг стану всіх курсів в межах спеціальності;
 - моніторинг роботи з курсами в межах спеціальності.
6. Відповідальний працівник кафедри:
 - створення нових курсів в межах кафедри;
 - призначення викладачів на курси;
 - надання консультативної інформації працівникам кафедр, щодо роботи у ВУ ЛДУ БЖД;
 - ведення кафедральної статистики роботи викладачів кафедр у ВУ.
7. Викладач:
 - наповнення електронних курсів навчально-методичними матеріалами;
 - призначення та видалення користувачів з курсу;
 - періодичне оновлення електронних курсів;
 - супровід роботи курсантів/студентів у електронних курсах;
 - аналіз роботи курсантів/студентів у електронних курсах.
8. Студент/курсант – слухач курсу.

Останнім етапом впровадження системи змішаного навчання було розроблення політики доступу до електронного курсу ВУ ЛДУ БЖД.

Доступ до ЕК ЛДУ БЖД – персоніфікований. Логін та пароль доступу курсанти та студенти отримують у адміністратора сайту на початку навчання. Викладачі мають можливість самостійно зареєструватись на сайті. Обов'язковою умовою реєстрації учасника на сайті є наявність облікового запису внутрішньої електронної пошти ЛДУ БЖД.

Кожний студент та викладач-тьютор має доступ лише до тих ЕК, на яких він

zareestrovaniy dlya uchasti y navchal'nomu protsesi. Rol' vikleдача kursu nadaetsya administratorem saytu. Zapis studentiv na kurs zdidsnyetsya takimi sposobami:

- викладачем цього курсу з обов'язковим вказанням терміну перебування курсанта та студента на електронний курс;
- адміністратором сайту з обов'язковим вказанням терміну перебування курсанта та студента на електронний курс;
- відповідальним працівником деканату з обов'язковим вказанням терміну перебування курсанта та студента на електронний курс;
- студенти записуються на курс самостійно (в цьому випадку електронний курс повинен бути захищений паролем і вказано дату відписки користувачів з курсу).

Nezvažajuchi na pereraxovani višče переваги використання, VU y LDU BJD vikoristovuvavsa ne v povnij miri. Virtual'nyj uniwersitet služiv dopomizhnyj (zapasnyj) instrumentom organizacii navčannya.

Sprubujemo porivnyati statystичні дані активності virtual'noho uniwersitetu LDU BJD do pandemii Covid-19 ta pid čas neї.

V tablicі 1 miŭstitsya informacija pro nayavnist' distancijnyh kursiv v mežax navchal'no-naukovyh institutiv LDU BJD stanom na berezen' 2017 roku, tobo za tri roki do pandemii i vojny v Ukraїni.

Таблиця 1

Наповненість virtual'noho uniwersitetu distancijnyjmi kursami y 2017 p.

Категорія курсів	Кількість курсів
Навчально-науковий інститут психології та соціального захисту	86
Навчально-науковий інститут цивільного захисту	224
Навчально-науковий інститут пожежної та техногенної безпеки	299
Інститут післядипломної освіти	9
Всього	618

З tablicі 1 vidno, ŭo y virtual'nomu uniwersiteti LDU BJD y peršomu kvartali 2017 roku narahovuvavlos' 618 distancijnyh kursiv.

Takoz možna proanalizuvati vidviduvanist' saytu korystuvachami za цей період. Taku informaciju možna pobachiti y tablicі 2.

Таблиця 2

Відвідуваність сайту за березень-травень 2017p.

03.03.2017	06.04.2017	11.05.2017
88	139	72

З tablicі 2 vidno, ŭo y berezni-travni 2017 roku ŭodenno vidviduvanist' virtual'noho uniwersitetu LDU BJD korystuvachami kolivavalsja v mežax 72-139 чоловік в день. Najmenša kіlkіst' pripadala na vихідні dні.

Цю динаміку možna продемонструвати на графіку, ŭo представлений na рис. 2.

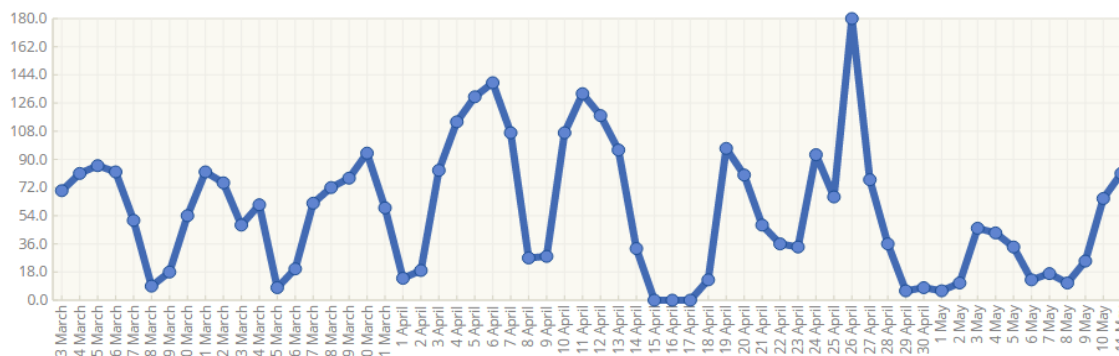


Рисунок 2. Відвідуваність віртуального університету ЛДУ БЖД за березень-травень 2017р.

Під відвідуваністю сайту мається на увазі кількість різних користувачів, які входили на сайт.

В таблиці 3 представлена інформація про статистику основних типів дій користувачів у віртуальному університеті ЛДУ БЖД за один тиждень з понеділка по п'ятницю в грудні 2018 року, тобто до пандемії Covid-19. Під статистикою розуміється кількість лог-записів конкретної події, які можна отримати за допомогою відповідного вбудованого інструментарію СДН Moodle.

Таблиця 3
Статистика подій у віртуальному університеті ЛДУ БЖД у 2018 році

	Робота з тестами	Робота з лекціями	Робота з завданнями	Всього
3 грудня 2018	3245	1243	1001	23794
4 грудня 2018	5314	1226	570	26389
5 грудня 2018	5523	1535	790	24730
6 грудня 2018	9802	1712	706	29828
7 грудня 2018	7546	1338	1351	26334
Всього	31430	7054	4418	131075

В таблиці показана статистика по основних типах дій користувачів під час роботи у дистанційному курсі віртуального університету ЛДУ БЖД та загальна кількість подій. До таких подій відносяться перегляд лекційних матеріалів, виконання завдань та тестів. Числові дані складають враження про велику активність, але насправді система записує в логи кожен клік миші користувача.

Для порівняння, кількість подій за 4 листопада 2022 року становила 42785 логів, що майже вдвічі більше за дані таблиці 3. А це ще навіть не заліково-екзаменаційний період.

В таблиці 4 міститься інформація про наявність дистанційних курсів в межах навчально-наукових інститутів ЛДУ БЖД станом на листопад 2022 року.

Таблиця 4
Наповненість віртуального університету дистанційними курсами на кінець 2022 року

Категорія курсів	Кількість курсів
Навчально-науковий інститут психології та соціального захисту	460
Навчально-науковий інститут цивільного захисту	851

Категорія курсів	Кількість курсів
Навчально-науковий інститут пожежної та техногенної безпеки	680
Інститут післядипломної освіти	20
Всього	2011

З даної таблиці можна зробити висновок, що сучасні умови призвели до значного зростання кількості дистанційних курсів. З'явилося близько 1500 нових дистанційних курсів, тобто їх кількість зросла практично втричі.

Щодо відвідуваності віртуального університету ЛДУ БЖД, то станом на вересень 2021 року вона становила більше 700 користувачів за день. Станом на листопад 2022 року ця кількість зросла до більше 6000. Цю динаміку можна побачити на рис. 3-4.

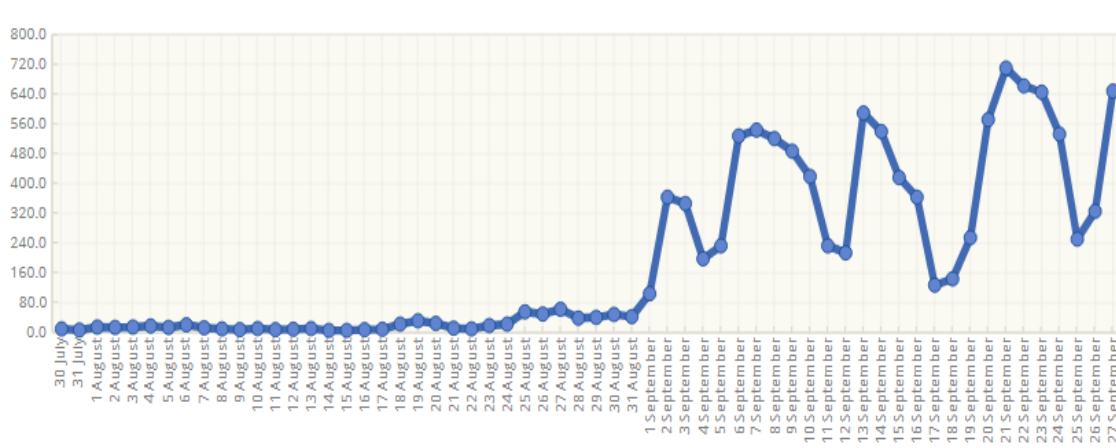


Рисунок 3. Відвідуваність віртуального університету ЛДУ БЖД у вересні 2021 р.

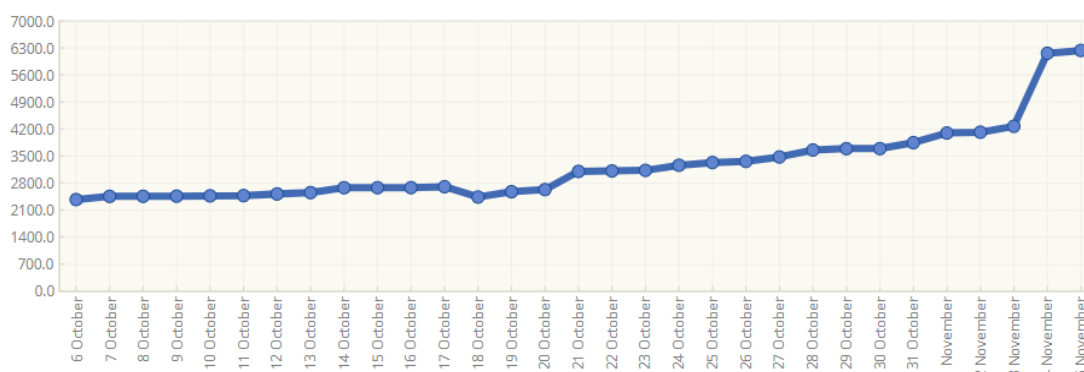


Рисунок 3. Відвідуваність віртуального університету ЛДУ БЖД у листопаді 2022 р.

Відвідуваність віртуального університету ЛДУ БЖД та робота в ньому під час пандемії у порівнянні з «докарантинним» періодом зросла практично у 4 рази. А якщо враховувати період війни, то ця кількість зросла ще майже у 10 раз.

В сучасних умовах зросло виконання всіх типів робіт у віртуальному університеті. Хоча варто виділити суттєве зростання (більше ніж у 10 разів) виконання тестових завдань, що пояснюється тим, що тести виступають, по-

перше, основним найбільш доступним для викладачів засобом перевірки знань студентів, по-друге вимогою керівництва щодо наявності тестів до кожної навчальної лекції курсу.

Висновки.

В результаті впровадження в освітній процес ЛДУ БЖД віртуального середовища, здобувачі вищої освіти мають змогу самостійно опрацювати теоретичний матеріал та візуально ознайомитись з особливостями та тонкощами таких процесів як пожежогасіння та ін. Отже, можна зробити висновок, що віртуальний університет ЛДУБЖД, виступає потужним інструментом для ефективного забезпечення освітнього процесу в умовах

Список використаних джерел:

1. Анисимов А.М. Работа в системе дистанционного обучения Moodle. Учебное пособие. – Харьков, ХНАГХ, 2008. – 275 стр.
2. Веб-сайт Віртуальний університет Львівського державного університету безпеки життєдіяльності. [Електронний ресурс] – Режим доступу з <http://virt.ldubgd.edu.ua>
3. Полотай О.І. Особливості запровадження змішаного навчання в закладах вищої освіти з особливими умовами навчання. «Змішане навчання – інновації ХХІ сторіччя»: матеріали міжнародної науково-практичної конференції. – Харків, ХПІ, 2018. – С. 79-84.
4. Полотай О.І., Кухарська Н.П. Розроблення електронних курсів у віртуальному навчальному середовищі. Навчальний посібник. - Львів : СПОЛОМ, 2021. – 172 с.
5. Полотай О.І., Меньшикова О.В. Використання технологій дистанційного навчання в умовах Covid-19 в ЛДУ БЖД. VI Зб. тез доп. VI Міжнар. наук.-практ. конф. “Інформаційно-комунікаційні технології в сучасній освіті: досвід, проблеми, перспективи”. (м. Львів, 04 листопада 2021 р.). Львів : ЛДУБЖД, 2021. С. 102-110.
6. Полотай О.І. Напрями вдосконалення управління проектами запровадження дистанційного навчання у вищому навчальному закладі. Управління розвитком складних систем. – 2013. – Вип. 13. – С. 40–44.
7. Полотай О.І., Ноздріна Л.В. Дослідження передумов запровадження інноваційних освітніх проектів у ВНЗ. CD - ROM : ISBN 978-966-593-624-4. Міжнародна науково-практична конференція “Е-навчання у вищій школі – проблеми й перспективи” (INCEL-08) (м. Одеса, 13–15 травня 2008 р.). Одеса : NTU "KhPI", 2008. С. 1–6.
8. Полотай О.І. Зростання індексів розвитку економіки знань – основа ефективного управління освітніми проектами інформатизації. Управління проектами та розвиток виробництва : зб. наук. пр. Луганськ : СНУ ім. В. Даля, 2012. № 3 (43). С. 62–69.

УДК 378:002

ЕФЕКТИВНІСТЬ ПРОВЕДЕННЯ ЛАБОРАТОРНИХ РОБІТ З ІНЖЕНЕРНОЇ МЕХАНІКИ В РЕЖИМІ КОНФЕРЕНЦІЇ ZOOM

Антонова Г.В., ст. викладач

e-mail: galina.antonova@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Одним з найважливіших завдань, що стоять перед освітою, є оволодіння студентами інформаційними та телекомунікаційними технологіями для формування професійних та загальних компетенцій. Цілком очевидно, що, використовуючи лише традиційні методи навчання, вирішити це завдання неможливо.

Сьогодні необхідно, щоб кожен викладач з будь-якої дисципліни міг підготувати та провести заняття з використанням інформаційних технологій, тому що це наочно, інформативно та інтерактивно, економить час викладача та студента, дає можливість викладачеві працювати зі студентами диференційовано та індивідуально, оперативно контролювати та оцінювати результати навчання.

За період пандемії отримано та накопичено досвід активного застосування дистанційних методів навчання на кафедрі ІМКП. Як студенти, так і викладачі висловили думку про існування певних переваг та недоліків даної системи навчання. До переваг віднесено: зручність проведення занять, використання сучасних цифрових освітніх ресурсів, здатність студентів навчатися самостійного пошуку інформації. Негативними ефектами стали відсутність безпосереднього контакту та взаємодії викладача зі студентами, неможливість контролю під час занять та іспитів, а також відсутність виконання практичних навичок за спеціальністю студентами, що також є значущою метою навчального процесу.

Лекційний матеріал можна швидко сформувати з електронних підручників та навчальних посібників, яких достатньо в Інтернеті з загальнотехнічних дисциплін [1, 2, 3, 4]. Практичні завдання з таких предметів, як механіка матеріалів та конструкцій, інженерна механіка, теоретична механіка, якщо викладач не мав своїх напрацювань, довелося терміново оформляти у зручний електронний формат.

Але найгірше становище виявилось з проведенням лабораторних робіт. За загальнотехнічними дисциплінами потрібна наочна демонстрація принципів роботи пристроїв та фізичних процесів, що відбуваються в процесі експерименту. Тому без навчальних програм, виконаних у віртуальній реальності, проведення лабораторних робіт малоефективне для навчання.

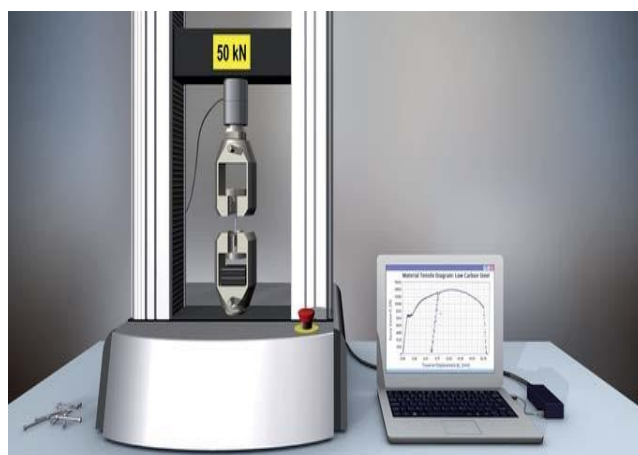
Основні матеріали дослідження. Випускник ВНЗ повинен мати професійний менталітет, тобто компонент професійної свідомості. Сформувати професійний менталітет майбутнього інженера неможливо без комплексного і системного підходу до освоєння загальнопрофесійних дисциплін. Щоб йшов процес формування необхідних професійних рис особистості повинна бути створена система вимог, що об'єктивно пред'являються до студентів та визначають відповідний рівень знань.

Проблематика. Аналіз, запропонований різними розробниками віртуальних лабораторних робіт з інженерної механіки (ММК) [5], показав, що вони практично не придатні для самостійного освоєння навчального матеріалу. Перегляд віртуальної лабораторної роботи призначений створення образу об'єкта

вивчення у свідомості студента. Під час цієї теоретичної діяльності студента відбувається процес формування розумових дій. У цей момент головним для ефективного створення матеріального образу та переходу його до розумової скарбнички людини є слово [9]. Правильно, професійно складені фрази, що описують процеси, що відбуваються на екрані, сприяють засвоєнню навчальної інформації та процесу інтеріоризації (від лат. *interior* - внутрішній, поняття, що означає формування розумових дій і внутрішнього плану свідомості через засвоєння індивідом зовнішніх дій з предметами спілкування). Але в розроблених віртуальних лабораторних роботах з інженерної механіки (ММК) відсутній головний інструмент інтеріоризації – слово, яке фіксує основні процеси, що досліджуються у лабораторній роботі. При випробуванні зразка на розтяг звуковим супроводом є гул двигуна протягом 30 с, до розриву зразка. Реєстрація залежності між абсолютним подовженням і прикладеною силою, що розтягує, так само не озвучена, а на екрані не чітко видно одиниці вимірювання осей координат, в яких будується графік. Віртуальні лабораторні роботи виконані засобами програмування тривимірної графіки та зображення точно копіює реальний об'єкт, можливість розгляду установки з різних сторін створює ефект присутності. Так під час проведення випробувань зразка на розтягуванні використовується універсально, показаний рисунку 1а.



а)



б)

Рисунок 1. Випробувальна установка:
а) реальний об'єкт; б) тривимірна модель

Об'єкт дослідження, в даному випадку зразок, що розтягується, піддається деформації розтягу. Так як абсолютне подовження реального зразка становить 15-20% від його робочої довжини, то при анімації без звукового пояснення процесу, що відбувається, це подовження малопомітно.

Обробка діаграми розтягування здійснюється в режимі реального часу, у міру розтягування зразка на екрані монітора будується графік. За графіком, отриманим у результаті віртуальної лабораторної роботи, студенти повинні розрахувати основні механічні характеристики матеріалу, з якого виготовлено віртуальний зразок. Але на екран не виведені формули для необхідного розрахунку, не дано пояснення до заданих механічних характеристик (рис. 2).

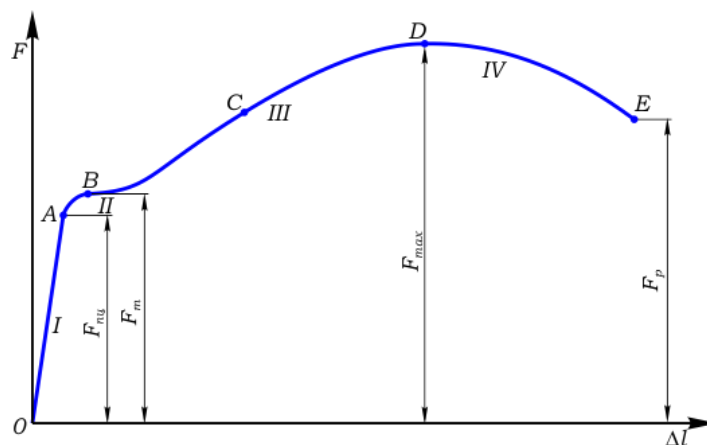


Рисунок 2. Діаграма розтягу (діаграма навантажень)

Ці комп'ютерні лабораторні роботи потребують повного попереднього освоєння теоретичного матеріалу. Необхідно під час проектування лабораторних робіт передбачити можливість поетапного занурення студента в теорію. Так, при розтягуванні зразка виділяються п'ять зон. Кожна з них характеризується змінами на міжмолекулярному рівні, які неможливо побачити під час реального експерименту. На ділянці BC відбувається зміни в кристалічній решітці металу і на поверхні зразка з'являються більш менш різкою мірою лінії Людерса-Чорнова. Якщо розвантажити зразок на цій ділянці, то спостерігатиметься явище «наклепу», тобто виникає ефект самозміцнення. Цей процес дуже важливий, оскільки він призводить до підвищення міцності та твердості металу, але водночас знижує стійкість до корозії. Майбутній інженер повинен знати і враховувати цю властивість деталей, виготовлених з маловуглецевої сталі, при розрахунку допустимих навантажень. Тому при вивченні діаграми розтягу необхідно не тільки коментувати процеси, що відбуваються на екрані, а й супроводжувати їх мультимедійними картинками. Таким чином, комп'ютерна графіка дозволяє зробити лабораторну роботу наочнішою, а значить запам'ятовується.

Класична методика проведення лабораторного практикуму містить усний звіт студента з цієї роботи після закінчення експериментальної та теоретичної частини. Віртуальна лабораторна робота також повинна містити засоби контролю та засвоєння теоретичного матеріалу. Це може бути система тестів, або он-лайн спілкування з викладачем через інтернет.

За рівнем пізнавальної активності, існуючі комп'ютерні роботи з інженерної механіки (ММК) значно відстають від натурних лабораторних робіт. Студент бачить і чує, але це не означає, що він повністю все розуміє, усвідомлює і може відтворити почуте у розмові з однокурсниками та викладачами. В аудиторії завжди йде спільна робота: вимірюють штангенциркулем геометричні розміри зразка, до та після проведення випробування, проводять розрахунки, результати вимірювань діаграми та розрахунки заносяться до зошита лабораторних робіт. Потім ведеться колективне обговорення і робляться висновки, у разі визначається з якого марки стали виготовлений зразок, взятий на випробування. Професійна особистість може сформуватися лише у спілкуванні з колегами з навчання чи роботі. Засвоєння нового матеріалу завжди відбувається лише через власну діяльність людини, тому навіть віртуальна лабораторна робота має бути організована таким чином, щоб студент брав участь у ній. Необхідно, щоб кожен

студент самостійно обробив діаграму розтягування: проставив на діаграмі характерні точки і розрахував механічні характеристики матеріалу, з якого виготовлений зразок.

Висновок. Таким чином, віртуальні лабораторні роботи з опору матеріалів, що знаходяться у вільному інтернет-доступі, не є повноцінним навчальним матеріалом. Але можуть бути допоміжною частиною для навчання в режимі конференції Zoom або на освітньому порталі, де присутність викладача є невід'ємною частиною проведення заняття. Потрібно до кожної лабораторної роботи розробити нову методику організації самостійної роботи студентів з урахуванням поетапного освоєння та самоконтролю теоретичного матеріалу, що вивчається. Лабораторні роботи з інженерної механіки (ММК) призначені для студентів, які навчаються з технічних спеціальностей, тому необхідно провести зв'язок між теоретичними знаннями та майбутньою професією інженера.

Список використаних джерел:

1. Лабораторний практикум з механіки матеріалів і конструкцій: навчальний посібник для підготовки бакалаврів зі спеціальностей: 208 «Агроінженерія», 133 «Галузеве машинобудування», 122 «Комп'ютерні науки»/ Л.Ю. Бондаренко, О.О. Вершков, Г.В. Антонова. Мелітополь: Люкс, 2020. 243 с.
2. Прикладна механіка: навчальний посібник / Г.В. Антонова, Л.Ю. Бондаренко, О.О. Вершков. Мелітополь: Люкс, 2019. 202с.
3. Механічні випробування матеріалів: навчально – методичний посібник для виконання лабораторних робіт з дисципліни «Технічна механіка»: для здобувачів ступеня вищої освіти «Бакалавр» зі спеціальності 263 «Цивільна безпека» / Г.В. Антонова, О.О. Вершков, Л.Ю. Бондаренко, А.П. Чаплінський. Мелітополь: Люкс, 2020. 112 с
4. Бондаренко Л.Ю. Лабораторний практикум з механіки матеріалів і конструкцій [Електронний ресурс]: навч. посіб. / Л.Ю. Бондаренко, О.О. Вершков, Г.В. Антонова; Таврійський державний агротехнологічний університет. – Електрон. текст. дані. – Мелітополь: ТДАТУ, 2017. 183 с. – 1 електрон. опт. диск (CD-R). – Систем. вимоги: Pentium-II/300; 64 Mb RAM; Microsoft Windows XP; 60 Mb вільного дискового простору; NET Framework 2.0. – Лабораторний практикум з механіки матеріалів і конструкцій.
5. Віртуальні лабораторні роботи. https://m.apkpure.com/virtual-lab-strength-of-materials-free/com.virtlab.som_demo/download?from=details
6. Бондаренко Л.Ю., Вершков О.О. Використання відкритого програмного забезпечення для навчання здобувачів вищої освіти інженерних спеціальностей. *Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації*: матеріали Міжнар. наук.-практ. інтернет-конф. Мелітополь: ТДАТУ, 2020. С.220-224.
7. Бондаренко Л.Ю., Вершков О.О., Холодняк Ю.В., Гавриленко Є.А. Використання технологій візуалізації навчального матеріалу в інтелектуальних освітніх системах. Удосконалення освітньо-виховного процесу в закладі вищої освіти. Мелітополь: ТДАТУ, 2021. Вип. 24. С. 236-242.
8. Бондаренко Л. Ю., Тетервак І. Р. Впровадження та використання комп'ютерних технологій для вирішення задач опору матеріалів. Сучасні комп'ютерні та інформаційні системи і технології: матеріали I Всеукр. наук.-практ. інтернет-конф. (Мелітополь, 07-25 грудня 2020р.) Мелітополь: ТДАТУ, 2021. С.82-83.

УДК 004.92

КОМП'ЮТЕРНА ГРАФІКА. СФЕРА ЗАСТОСУВАННЯ І ПЕРСПЕКТИВИ РОЗВИТКУ

Постол Ю.О., к.т.н.

e-mail: yulia.postol@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Постановка проблеми. Одне із застосувань зростаючих обчислювальних потужностей персональних комп'ютерів – графічне уявлення складних процесів, моделювання реальних об'єктів, візуалізація даних, у суттєвій частці цифрові класичні інженерні функції, як проектні, і механічні. Всі ці завдання поєднуються однією назвою - комп'ютерна графіка.

Цифрове графічне уявлення знаходить своє застосування у різних галузях діяльності. Серед них можна виділити: дослідників у різних наукових та прикладних галузях; спеціалістів з комп'ютерної верстки; художників; конструкторів; дизайнерів; розробників рекламної продукції, авторів мультимедіа презентацій; медиків; модельєрів тканин та одягу; фотографів; фахівців у галузі теле- та відеомонтажу та ін. [1].

Основні матеріали дослідження. Вже неможливо уявити своє життя без використання комп'ютерної графіки практично у всіх сферах діяльності, у тому числі і за візуалізації різних видів даних. Наприклад можна назвати медицину, де використовується комп'ютерна томографія, рентгенівські знімки. Також комп'ютерна графіка використовується і в галузі медицини, як стоматологія, де проводиться проектування зубних протезів, моделювання майбутньої посмішки пацієнта після проходження лікування. Іншим прикладом є виробництво ювелірних виробів: за допомогою комп'ютерної графіки можна повністю сконструювати виріб, встановити його розміри, колір, вказати які метали будуть використовуватися при виготовленні кінцевого продукту.

І останнім прикладом можна розглянути дизайн інтер'єрів. За допомогою спеціалізованого програмного забезпечення можна не тільки сконструювати окремі види меблів, але й створити повністю готовий проект з усіма деталями: меблями, розташуванням світла, аксесуарів і навіть повністю оформити стіни і стелю.

Таким чином, можна описати практично всі галузі, де використовується комп'ютерна графіка. Вона значно полегшує створення того чи іншого продукту, скорочує час виробництва, мінімізує появу помилок при створенні кінцевого виробу.

Для подальшого обговорення слід сформулювати основні пункти, якими класифікують підходи у побудові графічних зображень інформаційними засобами: традиційно виділяють растрову, векторну і фрактальну графіку.

Відмінності між ними полягають у тому, що вони по-різному підходять до формування вихідних сигналів відображення бажаного результату.

Існує ще один підхід, який має достатню кількість відмінностей для обґрунтування розгляду його як окремої категорії. Йдеться про просторове представлення графічних зображень, а саме про тривимірну графіку. З погляду функціональних особливостей її слід розглядати як композицію векторного та растрового підходу. Тривимірне моделювання є унікальною технологією, що надає у майбутньому чудові можливості втілення всіх ідей для створення того чи

іншого зображення для багатьох бізнес-процесів, навчання та іншого, спрощуючи роботу та розширюючи можливості [2].

Крім конкретних прикладів, варто розглянути окремі області комп'ютерної графіки.

Графіка у наукових дослідженнях.

Як і багато інших технологій, комп'ютерні цифрові методи обробки формування графічних зображень знайшли своє перше практичне застосування у сфері наукових досліджень про. Наявність можливості швидкого отримання візуальної презентації результатів експериментів суттєво збільшила швидкість та ефективність роботи науково-дослідних колективів та організацій.

Графіка у корпоративному середовищі.

Наступним логічним кроком розвитку візуальної формалізації представленої інформації були економетричні показники та параметри планування програм розвитку. Величезний внесок у розвиток інструментів подібної роботи з інформацією зробили організації фінансової торгівлі, банки та державні структури, що їх контролюють.

Графіка в інженерному проектуванні.

Незамінний інструмент у сучасних проектних та конструкторських бюро. Високий рівень автоматизації дозволяє суттєво оптимізувати час розробки нових проектів та адаптації типових проектів під нові завдання. Сукупність таких інструментів називається системою автоматизації проектування (САПР).

Зауважимо, що такі системи функціонують не тільки в конструкторсько-інженерних задачах та проектах, а й застосовуються у процесі розробки сучасних обчислювальних систем. Незамінними є інструменти, що дозволяють оптимізувати розташування елементів на друкованій платі, автоматизовано прорахувати відстані між елементами на відповідність стандартам, що діють в індустрії. Окремо зауважимо, що наочність результатів такого моделювання дозволяє використовувати результати з погляду візуального дизайну пристроїв.

Інструменти для графічного дизайну.

Один із найпопулярніших кар'єрних напрямів на даний момент – графічний дизайнер, який використовує інструменти ілюстративної графіки для отримання малюнків, аналогічних творчості традиційних художників, лише у цифровому форматі. Такі інструменти можуть бути використані в будь-якій області, яка передбачає використання зображень.

Рекламний дизайн.

Маркетинговий ринок дає один із наймасштабніших запитів на комп'ютерну графіку, вимагаючи все більш складних програм, але й простих у застосуванні інструментів. Зараз завдання отримання рекламного банера за допомогою спеціалізованих програмних продуктів вкрай просте для дизайнера та вкрай складне для програміста. Розроблені графічні пакети вимагають великих обчислювальних потужностей, набувають більш зрозумілого обивателю інтерфейсу, отримують дедалі ширші можливості для отримання відеоряду, а не лише статичних зображень.

Комп'ютерна анімація.

Анімація – це створення рухомих зображень шляхом зміни властивостей нерухомих зображень, що швидко змінюють одне одного. Нещодавно художникам-мультиплікаторам доводилося створювати свої фільми та мультфільми вручну. Щоб передати рух, доводилося витратити дуже багато часу, щоб створити систему послідовних ітераційних змін у статичному зображенні, таким чином формуючи ілюзію динаміки процесу. Отримувані зображення потім

дублювалися на спеціальну плівку для подальшого відтворення. Сучасні системи подання візуальної інформації багато в чому автоматизували і вивели на новий рівень ці функції. Художнику анімації сьогодні достатньо лише вказати бажаний шлях статичної картинки, а програмне забезпечення самостійно сформує складну динаміку послідовності рухів на основі вже наявного зображення.

У наш час різноманітність інструментів для графічних завдань винятково велика, що дозволяє знайти як спеціалізовані пакети, так і універсальні рішення.

Існує безліч як платних, так і безкоштовних програм, а також online-редактори різної складності та спрямованості. Прикладами таких програм можуть бути: Adobe Photoshop, CorelDRAW, Autodesk AutoCAD, Autodesk 3ds Max, Blender та багато інших, створених для різних напрямків [3]. Наприклад, такі програми часто застосовуються для конструювання меблів, у створенні комп'ютерних ігор, креслень будівель та споруд, кіноіндустрії та багато іншого.

Для навчання комп'ютерної графіки в будь-якій галузі можна скористатися відео-уроками, викладеними у вільний доступ, різними курсами, як платними, так і безкоштовними, але в будь-якому випадку необхідний досвід приходить тільки з постійною практикою, а якщо враховувати, що програмні продукти постійно виходять оновлення, з додаванням нових функцій, покращенням зовнішнього вигляду програми, то вчитися потрібно постійно.

Якщо говорити про перспективи розвитку комп'ютерної графіки, то тут комп'ютерні ігри, які зараз розробляються з дуже якісною графікою, якнайкраще відображають поточний стан та розвиток комп'ютерної графіки. В ігровій промисловості комп'ютерна графіка займає дуже важливе місце. Будь-який ігровий проект, незалежно від його цілей та аудиторії, вимагає обов'язкового її опрацювання, чи це 2D, 3D або VR гра. У наш час комп'ютерна графіка в іграх у більшості випадків досягла своєї максимальної реалістичності, що, у свою чергу, вимагає підвищених потужностей, як для її створення, так і для її подальшої обробки. Робота над комп'ютерною графікою для ігор відбувається в процесі створення самих ігор фахівцями цієї сфери, на основі реальних або вигаданих об'єктів живої і не живої природи. Спочатку це лише проекти, які надалі вимагають ретельного опрацювання всіх найменших деталей, як геометричних, так і візуальних, від чого залежить ступінь реалістичності того чи іншого об'єкта в іграх. Крім того, моделюється і сам ігровий світ, який також має відповідати проектному як геометрично, так і візуально.

Висновки. Сучасний рівень розвитку графічного представлення інформації далекий від досягнення своєї межі, і те, що є зараз, здавалося фантастичним ще двадцять років тому. Зі зростанням обчислювальних потужностей, ускладнень внутрішньої архітектури графічних процесорів, розвитком математичного апарату геометрії, з'являтимуться дедалі досконаліші методи і підходи отримання графічних об'єктів. А це означає розвиток та зростання компаній, які такі продукти розробляють, зростання пропозицій системи освіти для нових напрямів підготовки як інженерної, так і в дизайнерській сфері.

Список використаних джерел

1. Комп'ютерна графіка. <https://veryimportantlot.com/news/blog/chto-takoe-kompyuternaya-grafika>
2. Огляд програм комп'ютерної графіки. URL: <http://cpu3d.com/grapcat/>
3. http://stud.com.ua/43376/informatika/inshi_vidi_kompyuternoyi_grafiki
4. http://studme.org/43369/informatika/kompyuternaya_grafika

ТЕХНОЛОГІЇ BYOD: МОБІЛЬНІ ПРОГРАМИ НА ДОПОМОГУ ВЧИТЕЛЮ

Сікора Я.Б., к.пед.н.

e-mail: iaroslava.sikora@gmail.com

Ляшенко А.І., студент

e-mail: musikalina14@gmail.com

Житомирський державний університет імені Івана Франка

Актуальність. Цифрова трансформація – це те, що сьогодні виділяє Україну з-поміж інших держав світу. Саме вона дає можливість забезпечити рівний доступ українців до будь-яких необхідних ресурсів. Тож в даному контексті особливої актуальності набуває цифрова трансформація освіти, а саме активне впровадження в освітній процес інноваційних технологій, покликаних полегшити навчання сучасним школярам, поглибити їхні знання та водночас зробити доступ до інформації швидким і зручним.

Постановка проблеми. За такими принципами наразі в українській освіті презентує себе BYOD-технологія. Впроваджуючи в освітній процес BYOD, педагоги по-новому презентують навчальний процес з методичної точки зору, формуючи в учнів нові компетентності, необхідні для суспільної адаптації в цифрову епоху. Тож об'єктивним наразі є й дослідження перспектив реалізації концепції Bring Your Own Device.

Методи вирішення проблеми. Під час написання даної статті ми керувалися методами аналізу й порівняння, що були зумовлені вивченням наукових та публіцистичних матеріалів із нашої тематики.

Результати. У статті досліджено та розкрито зміст поняття та концепції Bring Your Own Device, зокрема BYOD-технології в освітньому процесі. Зроблено висновок про те, що використання технології BYOD в освіті дає можливість вирішити кілька важливих проблем, які актуальні в сучасній освіті: за складних умов зробити навчання легкодоступним та підвищити вмотивованість здобувачів, шляхом «легалізації» гаджетів. Тобто, BYOD надає більше можливостей для навчання, оцінювання, покращення успішності здобувачів освіти, сприяє більш чіткій та своєчасній комунікації вчителя з учнями.

Висновки. Принцип BYOD є новим напрямком в області інформаційних і комунікаційних технологій в навчанні. Очевидні переваги використання цієї технології полягають в наступному: дозволяє поєднувати мобільне навчання разом із традиційними методами, сприяє розробці нових способів донесення інформації та формування діяльнісного і ціннісного компонентів змісту освіти із застосуванням мобільних пристроїв, допомагає поєднувати мобільне навчання з іншими компетентнісно орієнтованими технологіями, що в підсумку забезпечують максимальний ефект.

Ключові слова: цифровізація, Bring Your Own Device, BYOD-технологія, гаджети, сучасна освіта, нові компетентності, комунікація, інформаційні технології.

Relevance. Digital transformation is what distinguishes Ukraine from other countries of the world today. It makes it possible to ensure equal access of Ukrainians to any necessary resources. Therefore, in this context, the digital transformation of education is of particular relevance, namely the active introduction of innovative technologies into the educational process, designed to facilitate learning for modern students, deepen their knowledge and at the same time make access to information fast and convenient.

Statement of the problem. According to these principles, BYOD technology is currently presenting itself in Ukrainian education. By introducing BYOD into the educational process, teachers present the educational process in a new way from a methodological point of view, forming in students new competencies necessary for social adaptation in the digital age. Therefore, it is now objective to study the prospects for implementing the Bring Your Own Device concept.

Methods of solving the problem. When writing this article, we were guided by the methods of analysis and comparison, which were due to the study of scientific and journalistic materials on our topic.

Results. The article investigates and reveals the content of the concept and concept of Bring Your Own Device, in particular BYOD-technology in the educational process. It is concluded that the use of BYOD technology in education makes it possible to solve several important problems that are relevant in modern education: to make learning easily accessible in difficult conditions and to increase the motivation of students by "legalizing" gadgets. That is, BYOD provides more opportunities for learning, assessment, improving the performance of students, promotes clearer and more timely communication between teachers and students.

Conclusions. The BYOD principle is a new direction in the field of information and communication technologies in education. The obvious advantages of using this technology are as follows: it allows you to combine mobile learning with traditional methods, promotes the development of new ways of delivering information and the formation of activity and value components of educational content using mobile devices, helps to combine mobile learning with other competency-based technologies, which ultimately provide maximum effect.

Keywords: digitalization, Bring Your Own Device, BYOD-technology, gadgets, modern education, new competencies, communication, information technology.

Постановка проблеми. Нині основним пріоритетом розвитку країни та її стратегії є покращення якості освіти. Першочерговим в цьому плані є ефективне впровадження інформаційних та комунікаційних технологій в освітній процес. Сучасний рівень розвитку інформаційних технологій відкриває перспективи використання користувачами принципово нових засобів, зокрема мобільного навчання. Якщо дистанційне навчання дозволило слухачам навчатися віддалено від самого навчального закладу, то мобільне навчання пропонує можливість навчатися незалежно від місця і часу, забезпечуючи неперервність і максимальну гнучкість навчального процесу. Одним із шляхів запровадження мобільного навчання у школі є використання технології BYOD: Bring Your Own Device.

Ми живемо у період цифрового суспільства, де наявність в учнів різноманітних девайсів вже нікого не дивує, а навпаки може покращити навчання, зробити його цікавішим та ефективним.

Аналіз актуальних досліджень. BYOD відносно нова технологія, про яку вперше було заявлено в праці Рафаела Беллагаса (Rafael Bellagas) у 2005 році, та перекладається як «візьми свої девайси з собою». Це технологія, за якої активно використовуються мобільні телефони, планшети, ноутбуки та інші пристрої у навчанні.

Технологію використання засобів мобільного навчання висвітлювали такі науковці: І. Золотарьова І., О. Мардаренко, Т. Сараєва, А. Труш та ін. Зокрема, застосування технології BYOD розглядали Р. Браян (Byrne R.), Р. Мілман (Millman R.), М. Зільберман.

Також питаннями поширення BYOD в освітньому процесі займалися такі науковці, як Е. Чедбанд (E. Chadband), В. Копітоф (V. Kopytoff), П. Лінгвінстон (P. Livingston), Б. Стабнік (B Sztabnik), К. Хегарт (K Heggart), Т. Еріксон (T Erickson), С. Рос Померой (S. Ross Pomeroy), П. Де-Віт (P. DeWitt), Т. Балдасаро (T. Baldasaro) та ін.

Огляд наукової літератури дозволяє нам констатувати, що питанням BYOD-технології почали займатися в XX столітті в Західній Європі, Австралії та Америці, активніше його впровадження почало відбуватися вже на початку XXI в умовах розвитку новітніх технологій та поширення STEAM-освіти.

Отже, BYOD-технологія (*Bring Your Own Device*, перекл. як «принеси свій пристрій») – це технологія, що наразі набирає все більшої популярності серед освітян. Вона дозволяє педагогам контролювати використання здобувачами освіти особистих інформаційних пристроїв на уроках, демонструючи школярам можливості їх використання не тільки для спілкування чи ігрової діяльності, але й у якості навчальних засобів. Натомість у дітей вже з раннього віку починає формуватися цифрова компетентність [1].

Виклад основного матеріалу: Основною причиною використання девайсів у школах є потреба у підвищенні результатів навчання та мотивації сучасних дітей, тому що епоха цифрового суспільства вимагає докорінної зміни середовища навчання. До того ж BYOD-технологія, на думку педагогів, дозволяє вчителю і батькам не лише сприяти використанню девайсів саме з конкретною навчальною метою, а й застерігає дітей від марного витрачання часу в процесі їх використання та вчить основам актуальної на просторах Інтернету кібербезпеки.

Сучасні смартфони мають потужний процесор, швидку оперативну та велику пам'ять, розширений функціонал, камеру, що може фіксувати достатньо чіткі зображення. Цей пристрій може виконувати багато функцій: сьогодні його використовують для спілкування, обміну інформацією та документами, різноманітними файлами, для пошуку інформації, відео- та фотофіксації різних заходів, самостійних робіт, для прослуховування музичних файлів або перегляду відеофайлів, для зберігання потрібної інформації, роботи з додатками, тощо [3]. Тобто, це дуже потужний пристрій, що в сучасному світі позиціонується не тільки як засіб зв'язку, але й як необхідний засіб для виконання багатьох банківських та державних послуг, ідентифікації особистості, знаходження довідкових матеріалів, запису до електронних черг у медичні, дитячі, державні установи, оплати у транспорті та магазинах, тощо. Відтак, відкриває широкі можливості й для освітнього процесу. Суть технології BYOD полягає в тому, що учні приносять свої мобільні пристрої та з їх допомогою відбувається певна запланована учителем робота на уроці [1].

Використання BYOD-технології допоможе у:

- покращенні мотивації та розвитку активних учнів, з високими можливостями у навчанні та гіперактивних учнів;
- мотивуванні тих учнів, які можуть вважати традиційні методи навчання та сталі форми роботи нудними;
- розвитку інклюзивних дітей у навчанні: кращій підтримці менш працездатних учнів та учнів з обмеженими можливостями та спеціальними освітніми потребами [4, С. 148].

BYOD-технологія дозволяє окремим учням отримати доступ до цифрових підручників та інших навчальних ресурсів у багатьох місцях. Учні, які використовують власні пристрої, а не шкільні комп'ютери, мають більш розширений та персоналізований досвід. Вони можуть швидше виконувати

завдання та контролювати своє навчання, оскільки мають власне програмне забезпечення, з яким вони вже ознайомлені і це не потребує додаткових витрат часу на організаційні моменти типу: «як вимкнути, вимкнути прилад», «як користуватися ним» тощо. Вони можуть більше зосереджуватися на змісті навчальної діяльності, а не на технології, що використовується для його підтримки.

Використання мобільних пристроїв, особливо використання власних пристроїв учнів, надає більше можливостей для навчання школярів створювати власні навчальні матеріали на додаток до використання пристроїв для доступу до навчального контенту, створеного іншими. Це полегшує вбудовані функції збору даних на мобільних пристроях, включаючи можливість фотографувати та записувати відео, звуки, текст та знаходити інформацію про місцезнаходження. Різноманітні функції та додатки можна додатково використовувати на пристроях учнів для збору інших типів даних, наприклад, інформацію про температуру повітря або про різноманітний світ природи в будь-якому місці планети для використання на різноманітних уроках. Учні можуть поєднувати, редагувати, ділитися та додавати до даних, зібраних та створених, сприяючи розширенню спілкування, співпраці, партнерського навчання та для роботи проектною групою [5, С. 290].

Використання власних мобільних пристроїв учнів надає більше можливостей для формувального оцінювання, а їх поєднання із системами відповідей в Інтернеті забезпечує швидкий цифровий зворотний зв'язок щодо успішності учнів на теми, які потребують додаткових пояснень учителем. BYOD також може дозволити використання мобільних пристроїв для навчання та реалізації короткотермінових проєктів чи пілотних проєктів. Коли учні використовують власні пристрої для навчання, це допомагає розвинути їх цифрові компетенції. Крім того, ця технологія дозволяє користуватися мобільними пристроями, які можуть бути навіть досить примітивними і мати лише невелику кількість доступних функцій та можливостей.

Цифрові пристрої підтримують навчання дітей у школі та поза нею, допомагають їм у розвитку таких навичок XXI століття, як: спілкування, співпраця та творчість, засвоєння знань, отримання знань з медіаграмотності та оволодіння технологіями [2].

Існує достатня кількість мобільних додатків, платформ та ресурсів, які можна використовувати для навчання, зокрема Google Forms, Survey Monkey, Kahoot it!, Plickers, Grand Tools, Prompt offline translator, Education App For Kids, Linear X, Quick quadratics, Prezi, PowToon та багато інших [1]. За допомогою цих додатків учитель має можливість швидко оцінити знання та уміння, створювати навчальні матеріали в електронному вигляді, враховуючи принцип інтерактивності. Як правило ці програми працюють з різними операційними системами, а саме: Windows, Linux, Android, BlackBerry, iOS, тому можна відповідати на контрольні запитання або проходити тести, розв'язувати рівняння, створювати презентації, будувати графіки та діаграми із власного мобільного пристрою, а не використовувати стаціонарний персональний комп'ютер [3].

Завдяки BYOD покращується спілкування між учнями та батьками, батьками і вчителями, учнями та вчителями в результаті інтегрованого використання власних мобільних пристроїв учнів із використанням систем управління навчанням для обміну розкладами, описом уроків, навчальними ресурсами, завданнями, оцінками та інформацією. Але через специфічну організацію в кожній школі деякі з них можуть зіткнутися зі значними

проблемами у впровадженні BYOD. Школа повинна забезпечити, щоб усі пристрої мали сумісність з мережевими комп'ютерами та доступ до мережі інтернет, всім вистачало місця для підзарядки, а класи для фронтальної роботи були оснащені проекторами. Але, незважаючи на це, вчитель керує і організовує процес навчання і повинен пам'ятати, що девайси відіграють тільки допоміжну роль.

Використання технології BYOD у навчанні, дає змогу працювати в онлайн режимі, швидко давати відповіді, проходити тести, ознайомлюватись з навчальним матеріалом, створювати різні закладки. За рахунок виконання вказаними вище мобільними платформами і додатками трудомістких логічних операцій, учитель більше часу приділятиме безпосередній роботі з учнями (дискусії, круглі столи тощо). Тоді важливим аспектом буде ефективне, раціональне використання навчального часу на уроці [1].

Основними завданнями, які можна вирішити за допомогою BYOD-технологій в освітньому процесі є:

- **активізація пізнавальної діяльності:** навчальні можливості смартфона можна використовувати на будь-якому етапі заняття – чи то для ознайомлення з новим матеріалом, чи для закріплення раніше вивченого, або для того, щоб підсумувати вивчену тему;

- **оптимізація часу:** використання мобільних технологій дозволяє більш якісно впроваджувати діяльнісний підхід до навчання. Педагогу краще надати учням посилання на відеоматеріал, який мають наслідувати діти, ніж самому намагатися описати необхідні дії словами, витрачаючи на це багато часу. Те ж саме стосується і пошуку додаткової інформації, що може зацікавити учнів під час занять, у такому випадку педагог, використавши можливості смартфона, не залишає питання відкритим до наступного разу і не витрачає час на його роз'яснення на наступних заняттях, до того ж, демонструє свою мобільність та гнучкість. Деякі теми вчитель може пропонувати вивчити самостійно, надаючи посилання на необхідний ресурс;

- **забезпечення зворотного зв'язку:** завдяки використанню різних мобільних додатків, педагог може одночасно проводити опитування чи тестування, організовувати у чаті діалог з будь-якої теми для з'ясування ставлення дітей до тієї чи іншої події, для отримання запитань та спільного знаходження відповідей, тощо.

Висновок. Отже, мобільні технології стають невід'ємними освітніми інструментами, відкривають широкі можливості для використання інформаційно-комунікаційних технологій в навчанні. Тому вчителям потрібно розробляти нові способи донесення інформації до учнів із використанням мобільних пристроїв та поєднувати їх із традиційними методами навчання. Використання сучасних інформаційних технологій в навчально-виховному процесі стає для вчителя важливою частиною його професії. Застосування технології BYOD за умов вирішення проблем, що можуть виникати за умов використання мобільних технологій у навчальному процесі основної школи, є доцільним засобом для удосконалення навчального процесу у частині його автоматизації та підвищення інтересу учнів до навчальних предметів.

Сучасність формує до педагогів такі вимоги до організації освітньої діяльності, де неможливо обійтися без використання сучасних інформаційних засобів. Тому їх необхідно опановувати, вивчати можливість їх впровадження на уроках та виховних заходах.

Список використаних джерел:

1. Використання технології BYOD [Електронний ресурс]. Режим доступу: <http://didaktika.org/2014/p/ispolzovanie-mobilnyh-tehnologij-vobrazovatelnomprocesse/>.
2. Голуб Н. Компетентнісне навчання – це актуально/ Н. Голуб//Дивослово. 2012. №6. С.60-62.
3. Далингер. В.А. Компьютерная компетентность – основа профессионализма современного учителя математики // Материалы конференции «Информационные технологии в образовании-2003». [Електронний ресурс]. Режим доступу: <http://ito.edu.ru/2003/II/3/II-3-1788.html>.
4. Золотарьова І.О. Застосування мобільного навчання в системі освіти / І.О.Золотарьова, А. М. Труш // Системи обробки інформації. 2015. Вип. 4. С. 147-150. [Електронний ресурс]. Режим доступу: http://nbuv.gov.ua/UJRN/soi_2015_4_32.
5. Мардаренко О.В. Інтерактивні комунікативні технології освіти: мобільне навчання як нова технологія в підвищенні мовної компетенції студентів немовних ВНЗ / О.В. Мардаренко // Інформатика та математичні методи в Використання технології BYOD у процесі навчання в основній школі 4 моделюванні. 2013. Т. 3, № 3. С. 288-293.

УДК 378.091.33

ОГЛЯД ХМАРНИХ СЕРВІСІВ ДЛЯ ОСВІТИ

Сіциліцин Ю.О.¹, к.пед.н.

e-mail:yurarud@gmail.com

Семенов Є.О.², магістрант

e-mail:fermer1abdula@gmail.com

¹ Мелітопольський державний педагогічний університет імені Богдана Хмельницького

² Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Хмарні обчислення - це модель для забезпечення зручного, контрольованого мережевого доступу до спільного пулу обчислювальних ресурсів. За звичай хмарний ресурс користувач отримує заздалегідь налаштованим наприклад мережу, сервер, сховище, мережеві додатки та послуги. Такі ресурси можуть бути швидко надані та випущені з мінімальними зусиллями управління або взаємодією з постачальником послуг. Хмарні обчислення стали популярним рішенням для забезпечення дешевого та легкого доступу до екстериторіальних ІТ (інформаційних технологій) ресурсів. Все більша кількість організацій (наприклад, дослідницькі центри, підприємства) отримують вигоду від хмарних обчислень для розміщення своїх додатків. Завдяки віртуалізації, хмарні обчислення здатні вирішити з однією і тією ж фізичною інфраструктурою велику клієнтську базу з різними обчислювальними потребами. На відміну від попередніх парадигм (Clusters і Grid computing), хмарні обчислення не орієнтовані на додаток, а орієнтовані на послуги; вони пропонують на вимогу віртуалізовані ресурси як вимірювані та оплачувані утиліти [1]. Використання таких хмарних сервісів значно спрощує процес навчання у сучасних умовах, коли здобувачі освіти можуть знаходитись у будь-якому місці Земної кулі, а навчання проходить тільки дистанційно.

Основні матеріали дослідження. Cloud computing має різноманітні характеристики, основними з яких є:

- спільна інфраструктура – використовує віртуалізовану модель програмного забезпечення, що дозволяє обмінюватися фізичними службами, сховищем і мережевими можливостями. Хмарна інфраструктура, незалежно від моделі розгортання, прагне максимально використовувати доступну інфраструктуру для всіх зареєстрованих користувачів.

- динамічне забезпечення – дозволяє надавати послуги на основі поточних вимог попиту. Це робиться автоматично за допомогою програмної автоматизації, що дозволяє розширювати і скорочувати можливості обслуговування, в міру необхідності. Таке динамічне масштабування необхідно виконувати, зберігаючи високий рівень надійності та безпеки.

- доступ до мережі – доступ до Інтернету можливо отримати з широкого кола пристроїв, таких як ПК, ноутбуки та мобільні пристрої, використовуючи API на основі стандартів (наприклад, на основі HTTP).

- кероване вимірювання – використовує вимірювання для керування та оптимізації послуги, а також для надання звітної та платіжної інформації. Таким чином, споживачам виставляються рахунки за послуги відповідно до того, скільки вони фактично використали протягом розрахункового періоду.

Виходячи з вищезазначеного, хмарні обчислення є основою для спільного використання та масштабованого розгортання необхідних послуг майже з будь-

якого місця, і за які клієнт може отримувати рахунок на основі фактичного використання [1,2].

Розглянемо основні моделі розгортання хмарних сервісів. Моделями розгортання, визначеними хмарною спільнотою, є публічна хмара, приватна хмара, гібридна хмара та хмара спільноти.

Публічна хмара: одна з провідних форм поточної моделі розгортання обчислень. В основному використовується споживачем публічної хмари, а політика, вартість та вартість визначаються постачальником послуг. Популярними загальнодоступними хмарними сервісами є Amazon EC2, S3, Google App Engine та Force.com.

Приватна хмара: Це хмарна модель для однієї організації, керована організацією або третьою стороною. Інфраструктура може бути розташована на території або поза приміщенням. Основною причиною впровадження приватної хмари є максимізація та використання існуючих власних ресурсів. Другорядні причини включають конфіденційність даних та довіру до безпеки. Нарешті, вартість передачі даних і повний контроль над критично важливими діями за брандмауерами. Академічні установи створюють приватну хмару для дослідницьких та навчальних цілей.

Гібридна хмара: Це комбінація двох або більше хмар, а саме: приватної, хмари спільноти або публічної. Для оптимізації ресурсу та використання основних компетенцій публічних хмарних організацій використовують гібридну хмару. Віртуальна приватна хмара (VPC) - це модель розгортання Amazon Web Services (AWS). Використовуючи VPC, можна створити безпечний та безперебійний міст між IT-інфраструктурою організації та загальнодоступною хмарою Amazon. Гібридна хмара - це поєднання публічної та приватної хмари.

Під хмарою спільноти розуміємо хмару де кілька організацій однієї групи діляться своїми хмарними ресурсами та спільно будують політики та вимоги. Інфраструктура хмари може бути розміщена стороннім постачальником або в межах однієї з організацій спільноти[2].

Розглянемо моделей сервісів хмарних обчислень. Моделі хмарних сервісів описують, як хмарні сервісистають доступними для клієнтів. Більшість фундаментальних моделей обслуговування включають комбінацію IaaS (інфраструктура як послуга), PaaS (платформа як послуга) і SaaS (програмне забезпечення як послуга). Ці сервісні моделі можуть мати синергію між собою і бути інтуїтивнозалежними – наприклад, PaaS залежить від IaaS, оскільки платформи додатків вимагають фізичної інфраструктури.

Модель IaaS (Інфраструктура як послуга) надає клієнтам компоненти інфраструктури. Компоненти можуть включати віртуальні машини, сховища, мережі, брандмауери, балансувальники навантаження і так далі. З IaaS клієнти мають прямий доступ до програмного забезпечення найнижчого рівня в стеку – тобто до операційної системи на віртуальних машинах, або до інформаційної панелі управління брандмауера або балансувальника навантаження. Amazon Web Services є одним з найбільших постачальників IaaS.

Модель PaaS (Платформа як послуга) надає клієнту попередньо створену платформу додатків; клієнтам не потрібно витрачати час на створення базової інфраструктури для своїх додатків. На бекенді PaaS автоматично масштабує та вимагає положень компоненти інфраструктури залежно від вимог до приєднання. Як правило, рішення PaaS надають API, який включає в себе набір функцій для програмного управління платформою і розробки рішень. Google AppEngine є

популярним постачальником PaaS, і Amazon Web Services також надає деякі пропозиції PaaS надодаток до пропозицій IaaS.

SaaS (Програмне забезпечення як послуга) надає готові онлайн-програмні рішення. Постачальник програмного забезпечення SaaS має повний контроль над прикладним програмним забезпеченням. Приклади додатків SaaS включають онлайн-пошту, систему управління проектами, CRM та платформи соціальних мереж.

Основна відмінність SaaS від PaaS полягає в тому, що PaaS зазвичай являє собою платформу для розробки додатків, в той час як SaaS надає онлайн-додатки, які вже розроблені[3].

Розглянемо основних постачальників хмарних обчислень.

Microsoft Live@edu для освіти. Microsoft Live@edu призначена для освітніх потреб. Він надає набір розміщених послуг співпраці для навчальних закладів. Розміщена послуга включає послуги співпраці, засоби зв'язку, мобільні, настільні та веб-програми. Він має функцію можливостей зберігання даних. Робоча область Office Live, служба Windows Live SkyDrive, Windows Live Spaces, бета-версія спільного подання Microsoft, Microsoft Outlook Live, програма Windows Live Messenger і оповіщення Windows Live є частиною програмного комплексу Live@edu. За допомогою процесу безкоштовної реєстрації університети, коледжі та школи можуть записатися на програму [4]. Microsoft Live@edu в основному призначений для установ для надання можливостей для їх академічної діяльності.

Google Додатки для освіти. Google Apps – це колекція веб-програм і сховищ файлів, які запускаються у веб-браузері, не вимагаючи від користувачів купувати або встановлювати програмне забезпечення. Користувачі можуть просто увійти в службу, щоб отримати доступ до своїх файлів та інструментів для маніпулювання ними. Інструментами спілкування Google Apps є Gmail, Google Talk і Календар Google, а інструментами підвищення продуктивності є Документи Google: текстові документи, електронні таблиці та презентації, iGoogle і Google Sites для розробки веб-сторінок [5]. Інструменти безкоштовні, або користувачі можуть заплатити за Преміум-видання, яке додає більше місця для зберігання та інші функції. Освітнє видання включає більшість додаткових послуг у Premium Edition і пропонується безкоштовно до К-12 (позначення суми початкової та середньої освіти та вищої освіти). Google Apps дозволяє установам використовувати власне доменне ім'я зі службою та налаштувати інтерфейс, щоб відобразити брендинг цієї установи. Таким чином, коледж або університет може запропонувати функціональність Google Apps у пакеті[6].

Веб-сервіси Amazon для освіти (AWS). Amazon Web Services надає хмарні сервіси в категоріях «Обчислення», «Програмне забезпечення», «Доставка контенту», «База даних», «Сховище», «Розгортання та управління», «Сервіси додатків» і «Робоча сила» [6].

Обчислювальна послуга включає в себе Amazon Elastic Computer Cloud (EC2), Amazon Elastic MapReduce, автоматичне масштабування та пружне балансування навантаження. Amazon Elastic Compute Cloud забезпечує масштабовану обчислювальну ємність з оплатою по мірі використання в хмарі. Amazon Elastic MapReduce – це веб-сервіс, який дозволяє підприємствам, дослідникам, аналітикам даних і розробникам легко та економічно ефективно обробляти величезні обсяги даних. Автоматичне масштабування дозволяє користувачеві автоматично масштабувати ємність Amazon EC2 вгору або вниз відповідно до умов. Elastic Load Balancing автоматично розподіляє вхідний трафік додатків між кількома екземплярами Amazon EC2.

У програмному забезпеченні AWS Marketplace - це інтернет-магазин, який допомагає клієнтам знайти, придбати та негайно почати використовувати програмне забезпечення, яке працює в хмарі AWS. Він включає програмне забезпечення від надійних постачальників, таких як SAP, Zend, Microsoft, IBM, Canonical та 10gen, а також багато широко використовуваних пропозицій з відкритим кодом, включаючи Wordpress, Drupal та MediaWiki.

У доставці вмісту Amazon CloudFront — це веб-сервіс, який дозволяє легко розповсюджувати вміст із низькою затримкою через глобальну мережу крайових розташувань.

У базі даних він має категорію реляційних баз даних Amazon (RDS), Amazon DynamoDB, Amazon SimpleDB та Amazon Elastic Cache.

Служба реляційних баз даних Amazon - це веб-сервіс, який дозволяє легко налаштувати, керувати та масштабувати реляційну базу даних у хмарі. Amazon DynamoDB - це повністю керована, високопродуктивна служба баз даних NoSQL, яка проста в налаштуванні, експлуатації та масштабуванні. Amazon SimpleDB — це керована служба баз даних NoSQL, розроблена для менших наборів даних. Amazon ElastiCache — це веб-сервіс, який дозволяє легко розгортати, керувати та масштабувати кеш у пам'яті в хмарі.

У мережі класифікаціями є Amazon Route S3, Віртуальна приватна хмара Amazon (VPC) та AWS Direct Connect.

У сховищі, залежно від потреб, послугами, що надаються AWS, є служба зберігання Amazon Simple (S3), льодовик Амазонки, магазин еластичних блоків Amazon (EBS), імпорт/експорт AWS та шлюз зберігання AWS.

Висновок. Хмарні обчислення швидко зростають. Розвиваються моделі, що пропонують значні переваги, але також потенційні недоліки. Хмарні обчислення можуть стати у пригоді всім, від малого бізнесу та великих підприємств до елітних університетів та онлайн-коледжів. Хмарні обчислення, що мають як сильні, так і слабкі аспекти, можна сказати, що масштабована, портативна, платіжна модель за використання і політика управління ризиками і безпекою, ефективність, доступність в будь-який час і ряд інших аспектів представляють позитивні фактори при прийнятті рішення про використання хмарних обчислень. Хмарні обчислення дозволять учням отримувати освітні послуги дистанційно що є дуже важливим у теперішній час.

Список використаних джерел:

1. J.Srinivas, K.Venkata Subba Reddy, A. Moiz Qyser. Cloud Computing Basics, *International Journal of Advanced Research in Computer and Communication Engineering*. 2012. Vol. 1,
2. B.V. Pranay Kumar , S. Kommareddy , N. Uma Rani, Effective ways cloud computing can contribute to education success. *Advanced Computing: An International Journal (ACIJ)*. 2013. Vol.4
3. E. Gorelik, Cloud Computing Models. *MIT Sloan School of Management and the MIT Engineering Systems Division*. 2013. URL: <https://dspace.mit.edu/handle/1721.1/7582>
4. M. Sirelkhtem Adrees, M. Kamal Aldein Omer, O. E. Sheta. Cloud computing architecture for higher education in the third world countries (republic of the sudan as model. *International Journal of Database Management Systems (IJDMS)*. 2015. Vol.7, No.3.
5. T. Erl, Z. Mahmood, R. Puttini. Cloud Computing: Concepts, Technology & Architecture. ServiceTech Press , September 2013.

УДК 004.65

ПРОЕКТУВАННЯ БАЗИ ДАНИХ ДЛЯ НАВЧАННЯ З ПИТАНЬ ОХОРОНИ ПРАЦІ

Козирєва Т.А., студентка

e-mail: zipzm221_kta@student.ztu.edu.ua

Дмитренко І.А., ст. викл.

e-mail: iryna.dmytrenko@ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Важливою складовою системи управління охороною праці на об'єкті господарювання (підприємстві, установі, організації) є навчання та інструктаж працівників з питань охорони праці. Своєчасне та якісне навчання з питань охорони праці є гарантією безпечного та безаварійного ведення робіт підвищеної небезпеки та експлуатації устаткування підвищеної небезпеки, що в свою чергу сприяє збереженню здоров'я та життя працівників. В залежності від виконуваної роботи, періодичність та кількість необхідних напрямків навчання та інструктажів для кожного працівника може варіювати у значних межах. Перед відділом охорони праці стоїть завдання забезпечити своєчасне проведення навчання для кожного працівника. Тому автоматизація обліку проведення інструктажів є вкрай важливою та призначена для зниження трудомісткості та скорочення термінів обробки інформації відділом охорони праці підприємства. Відсутність системи призводить до повільних та незручних процедур оформлення документів, низької ефективності обробки інформації.

Основні матеріали дослідження. Сьогодні на ринку є програмні продукти за цим напрямком 1С: Підприємство 8: Виробнича безпека. Промислова безпека та AVELife TestGold Agent 5 (таблиця 1). Але продукти з російською локацією не продаються в Україні з 24.02.2022 р.

Таблиця 1

Порівняння систем автоматизації

Функції та характеристики	1С: Підприємство 8. Виробнича безпека. Промислова безпека	AVELife TestGold Agent 5
Реєстрація та облік інструктажів	+	-
Збереження даних про інструктажі	+	-
Реєстрація та облік перевірок знань	+	+
Ведення електронних журналів обліку проведення інструктажів	+	+
Перелік необхідних напрямків навчання для працівника	-	-
Контроль проведення навчання та перевірок знань	-	+
Низька вартість розробки	-	-
Модернізація під потреби підприємства	+	-

Розглянуті системи не відповідають усім функціональним вимогам, а також мають високу собівартість придбання та обслуговування.

Після проведення порівняльного аналізу наявних систем автоматизації, можна зробити висновок про необхідність розробки власного проекту автоматизації.

Моделювання системи повинно відбуватися в рамках системи стандартів IDEF. Одним з перших етапів є проектування бази даних та побудова моделі даних. Від того на скільки буде прорахована структура бази та визначені зв'язки між її елементами, залежить продуктивність системи та її інформаційна насиченість.

Після вивчення нормативно-правових актів з питань охорони праці можемо виділити такі основні об'єкти БД:

- *працівник* - ПІБ працівника, дата народження, стать;
- *відділ, цех* – підрозділ підприємства;
- *основна посада* – назва посади ;
- *посада за суміщенням* – назва посади;
- *роботи підвищеної небезпеки* – назви робіт;
- *перевірка знань* – дата поточної перевірки знань, номер протоколу, код нормативно-правового акту з питань охорони праці за Показником,
- *дата наступної перевірки знань* - дата перевірки знань в залежності від періодичності навчання;
- *особи, що проводять інструктаж* – ПІБ, посада.
- *особи, що проводять навчання з питань охорони праці* – ПІБ, посада.

В системі повинно здійснюватися планування і фіксуватися результати проведення навчання та атестації співробітників в галузі охорони праці, тому раціональним вважаємо розподіл на розподіли:

- Планування навчання з питань охорони праці.
- Результати навчання та допуски до роботи.
- Аналітичні звіти.

Необхідно реалізувати формування наступних документів:

- список професій і посад, які підлягають навчанню;
- графік проведення навчання та атестації працівників;
- результати навчання та атестації працівників;
- протоколи перевірки знань з питань охорони праці;
- графік проведення інструктажів з питань охорони праці;
- журнал проведення інструктажів з питань охорони праці
- журнал проведення вступного інструктажу з питань охорони праці.

У системі слід передбачити контроль термінів дії документації дозвільного характеру, допусків до роботи, прострочення черговості навчання або інструктування за допомогою автоматичних нагадувань і повідомлень.

Висновок. Отже, автоматизація обліку дозволить оперативно отримати інформацію про підготовку і навчання співробітників для прийняття ефективних управлінських рішень та покращити планування та здійснення з роботи охорони праці в цілому.

Список використаних джерел:

1. Матвієнко О., Цивін М. Основи організації електронного документообігу: Навчальний посібник.-К.: Центр учбової літератури, 2008. – С. 17-22.
2. Електронний документообіг як інструмент управління інформаційним забезпеченням діяльності підприємства // Збірник наукових праць учасників IX Всеукраїнської науково-практичної конференції «Сучасні підходи до управління підприємством». Київ: НТУУ «КПІ імені Ігоря Сікорського», 2018.

УДК 378.091.33

ВИБІР ОНЛАЙН ЗАСОБІВ ДЛЯ РОЗРОБКИ ПАРАЛЕЛЬНИХ ПРОГРАМ З ВИКОРИСТАННЯМ БІБЛІОТЕКИ MPI

Сіциліцин Ю.О., к.пед.н.

e-mail :yurarud@gmail.com

Мелітопольський державний педагогічний університет імені Богдана Хмельницького

Актуальність та постановка проблеми. Події останніх років суттєво змінили всю систему освіти в Україні та світі. Вітчизняні та світові системи зіткнулися з пандемією коронавірусу, до чого досить швидко змогли адаптуватися завдяки попереднім напрацюванням щодо формату дистанційного навчання. Тепер вітчизняна освітня система зіткнулася з викликами, що постали внаслідок війни з Росією на Сході України. Більшість навчальних закладів, яких безпосередньо зачепили воєнні дії на сході продовжили працювати у дистанційному режимі навчання. Ще один виклик – це окуповані учбові аудиторії та обладнання для проведення лабораторних занять. Є багато практичних дисциплін, які потребують наявності аудиторного обладнання для проведення практичних занять. Одна з таких дисциплін це «Технології розподілених систем та паралельних обчислень». Під час аудиторних занять здобувачі, які вивчають цю дисципліну проводять ряд практичних робіт у ході яких необхідно розробити розподілену програму для учбового кластеру або розробити таку програму для запуску у локальній комп'ютерній мережі. Головним обладнанням під час таких практичних робіт є наявність декілька комп'ютерів, які поєднані у комп'ютерну мережу [1].

Основні матеріали дослідження. При викладанні паралельних та розподілених обчислень для практичних робіт зазвичай використовують мову C++ разом з бібліотекою MPI [2]. Інтерфейс передачі повідомлень (MPI) - це стандартизована бібліотека передачі повідомлень. Парадигма передачі повідомлень була розроблена для архітектури розподіленої пам'яті. Вона пропонує різні технології передавання повідомлень, які забезпечують спільну пам'ять. У програмі передачі повідомлень процеси з'єднуються через набір примітивів точка-точка та колективних комунікацій. MPI є фактичним стандартом розробки високопродуктивних наукових програм [2]. MPI пропонує API для розробки масштабованих програм у C ++ які є портативними, ефективними та гнучкими. У якості методів та засобів викладання розподіленого програмування за допомогою бібліотеки MPI використовують навчальний комп'ютерний кластер або локальну комп'ютерну мережу навчальної аудиторії. Але під час дистанційного навчання ці засоби не доступні.

Вирішення цієї проблеми можливо декілька підходів:

1. Встановлення та розробка програм MPI на локальному комп'ютері здобувача.
2. Розгортання на локальному комп'ютері здобувача декілька віртуальних комп'ютерів та об'єднання їх у кластер.
3. Використання онлайн засобів для розробки паралельних програм з використанням бібліотеки MPI.

Розглянемо переваги та недоліки кожного з підходів з точки зору онлайн викладання дисципліни «Технології розподілених систем та паралельних обчислень».

Встановлення та розробка програм MPI на локальному комп'ютері здобувача є самим простішим з розглянутих підходів. Його основні переваги – відносна простота дій по встановленню бібліотеки та створенню MPI програми. Недоліками такого підходу є не можливість здобувача запускати MPI саме у розподіленому середовищі що значно знижує навчальний ефект від розробки програми, та можливі проблеми з встановленням бібліотеки MPI на старі та одноядерні комп'ютери. Також неможливість провести практичну, якщо у здобувача зовсім відсутня можливість доступу до комп'ютерної техніки.

Розгортання на локальному комп'ютері здобувача декілька віртуальних комп'ютерів та об'єднання їх у кластер – є оптимальним підходом при наявності у здобувача потужного багатоядерного комп'ютеру з великим об'ємом оперативної пам'яті. При цьому підході можна розробляти MPI-програми та відразу їх тестувати у віртуальному кластері або локальній мережі. Недоліки такого підходу це наявність у здобувача високого рівня знань з архітектури комп'ютерів та комп'ютерних мереж та наявність потужного комп'ютерного обладнання.

Використання онлайн засобів для розробки паралельних програм з використанням бібліотеки MPI це організація розробки розподілених програм за допомогою онлайн сервісів, які дозволяють розгорнути віртуальну комп'ютерну мережу за допомогою хмарних технологій. Після встановлення та налагодження такої мережі та засобів розробки розподілених програм здобувач може підключитися до мережі за допомогою будь-якого пристрою. Переваги такого метода це відсутність підготовчих заходів з боку здобувача перед початком практичної роботи, не має великих вимог до апаратних засобів здобувача, не має вимог до кваліфікації здобувача по встановленню та налагодженню комп'ютерних мереж. Єдиний недолік це наявність у здобувача стійкого інтернет з'єднання.

Розглянемо деякі онлайн засоби за допомогою яких можна розробити онлайн кластер для викладання паралельних та розподілених обчислень:

VPS (Virtual Private Server) – онлайн ресурс який надає послуги оренди віртуального серверу для будь-яких цілей. На основі такого серверу (або декількох серверів) можна організувати початковий кластер для дистанційного навчання. Недоліки – доволі висока ціна, віртуальні процесори серверу не дозволяють у повній мірі дослідити розробку розподілених програм.

Виділений сервер – онлайн ресурс у вигляді фізичного серверу, який можна взяти в оренду для будь яких цілей. Завдяки фізичному доступу до обладнання на його основі можна організувати віртуальний обчислювальний кластер який буде задовольняти всі потреби при розробці розподілених програм. Головний недолік такого засобу – дуже велика ціна оренди. В основному такий сервер використовують комерційні бізнес-структури.

Amazon Web Services (AWS) це платформа для хмарних обчислень, яка надає свої потужності в оренду на основі платної підписки. [3] Існує безкоштовна підписка, яка доступна протягом перших 12 місяців. За допомогою AWS ми отримуємо повноцінний віртуальний кластер комп'ютерів, який доступний при наявності мережі Інтернет. Віртуальні комп'ютери AWS мають усі необхідні апаратні компоненти для розробки на їх основі віртуального навчального кластеру. Переваги такого підходу це можливість розгорнути віртуальну комп'ютерну мережу за допомогою хмарних технологій. Після встановлення та налагодження такої мережі та засобів розробки розподілених програм здобувач може підключитися до мережі за допомогою будь-якого пристрою та використовувати ресурси для роботи по створенню розподілених програм за допомогою технології MPI.

Висновок. Були розглянуті проблеми, які виникають при організації практичних робіт з предмету «Технології розподілених систем та паралельних обчислень» під час дистанційного навчання. Було виявлено, що шляхом для вирішення таких проблем є організація віртуального навчального кластеру засобами онлайн сервісів.

Вважаємо, що з розглянутих варіантів онлайн засобів розробки віртуального навчального кластеру за відношенням переваг та недоліків оптимальним варіантом є хмарна платформа Amazon Web Services.

Список використаних джерел:

1. https://uk.wikipedia.org/wiki/Amazon_Web_Services
2. Sitsylitsyn Yu. Methods, tools for teaching parallel, distributed computing in universities: a systematic review of the literature. *SHS Web of Conferences*. 2020. Vol. 75. Article 04017. DOI: 10.1051/shsconf/20207504017
3. Marowka A., Think Parallel: Teaching Parallel Programming Today. *IEEE Distributed Systems Online* 9(8) (2008). doi:10.1109/MDSO.2008.24

УДК-004.8

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ВОЛОНТЕРСЬКОГО ШТАБУ

Вакалюк Т.А., д. пед.н, професор

e-mail : kik_vta@ztu.edu.ua

Андрусенко О.М., магістрант

e-mail: kim221_aom@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Веб-орієнтовані інформаційні системи широко застосовуються в різноманітних сферах діяльності людини, а саме в інтернет-магазинах, лікарнях, навчальних закладах, громадських організаціях тощо.

Зважаючи на сьогоденну ситуацію в Україні кількість людей, яким потрібна допомога, зростає дуже стрімко, і люди потребують способу, який дозволить швидко та зручно знайти волонтерський штаб, де вони зможуть отримати речі першої необхідності. В основному наявні варіанти веб-ресурсів волонтерських організацій мають ряд недоліків. Їх усунення вимагає вдосконалення наявних систем, алгоритму їх роботи та взаємодії з користувачем. Зокрема, необхідно зробити взаємодію користувача з системою максимально зручною, простою та зрозумілою, оскільки більшість наявних організацій взаємодіють з користувачами через соціальні мережі, що значно зменшує її ефективність. Також необхідно розробити функціональну частину програми так, щоб робота з нею була максимально простою з боку користувача та максимально ефективною з боку серверної частини.

Тому створення веб-орієнтованої системи волонтерського штабу, у якій реалізовано зручний та швидкий інтерфейс користувача для зручного отримання необхідної допомоги, має стати інструментом для ефективної організації передачі допомоги людям, які потребують її.

Основні матеріали дослідження. Веб-сайт волонтерського штабу обов'язково повинен містити перелік доступних речей, які є в наявності в тому, чи іншому волонтерському центрі, їх короткий опис та можливість забронювати необхідні речі, щоб забрати їх в зручний час.

Оскільки список доступних речей на веб-сайті може бути досить великим, то необхідно організувати швидку пошукову систему та можливість фільтрування за певними критеріями. Дані механізми зроблять пряму двосторонню взаємодію із сайтом більш зручною та зрозумілою та дозволять пришвидшити процес формування списку необхідних речей користувачем.

На веб-сайті користувач повинен мати змогу здійснювати формування такого списку речей за допомогою зручного інтерфейсу, після чого на його основі створити заявку на отримання, в якій вказати час отримання у відповідному центрі видачі. Також необхідно, щоб веб-сайт містив детальну інформацію про ці центри, а саме про їх точне розташування, контактний номер телефону та години роботи.

Важливо теж реалізувати можливість створення заявки на допомогу самим волонтерським центрам, тобто користувач повинен мати змогу створити список речей, які він може принести в центр, відповідно обравши його зі списку, та вказати час, коли ці речі прибудуть до пункту прийому цього центру. Це дасть можливість людям допомагати одне одному в складний час за допомогою створеної інформаційної системи.

Головне, щоб веб-сайт був адаптивним та оптимізованим для швидкого завантаження на всіх пристроях, а також мати зручний, швидкий та інтуїтивно зрозумілий для користувача інтерфейс, щоб зробити користування сайтом максимально простим та доступним.

Висновки. Розробка та покращення інформаційних систем волонтерських штабів в наш час має бути одним з пріоритетних напрямків розвитку у сфері благодійної діяльності та волонтерства. Така система повинна не лише забезпечувати можливість отримання інформації про допомогу, а й максимально спрощувати процес її отримання, а також надавати можливість користувачам робити свій внесок в роботу волонтерських організацій шляхом реалізації можливості формування заявок на додавання речей у волонтерські центри.

Список використаних джерел:

1. Інформаційні системи та їх види. www.kievoit.ippo.kubg.edu.ua.
URL: <http://www.kievoit.ippo.kubg.edu.ua/kievoit/2013/95/95.html>.

УДК-004.8

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ КЛІНІК ТА ЛІКАРЕНЬ

Вакалюк Т.А., д. пед.н., професор

e-mail : kik_vta@ztu.edu.ua

Скріпченко Д.Г., магістрант

e-mail: ipzm221_sdg@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Хоча медицина тривалий час розглядалася як професія, що базується виключно на живому спілкуванні, останнім часом стає все більш очевидним, що використання ІТ загалом, та зокрема електронних комунікацій, забезпечує не тільки економічну ефективність діяльності в медичній сфері, але й також спосіб забезпечення дистанційного зв'язку між медичним персоналом і пацієнтами або особами, які за ними доглядають.

Використання Інтернету в лікувальній діяльності стало реальністю останніх років, а телемедицина або віртуальна лікарня перейшли етап наукової фантастики в реальність. Розвиток інформаційного суспільства, збільшення кількості людей, які мають доступ до комп'ютера, навіть людей похилого віку (включаючи розробку для них вступних курсів з комп'ютера), диверсифікація інформації та необхідність якнайшвидшої взаємодії зумовили адаптацію медичних підрозділів до засоби передачі інформації, щоб зробити їх більш привабливими, простішими в експлуатації та використанні. Розробка систем із гарантованою якістю інформації має бути постійною турботою спеціалістів у сфері громадського здоров'я, які мають працювати разом із ІТ-фахівцями та створювати власні веб-сторінки.

Основні матеріали дослідження. Багато інтернет-сайтів намагаються зарекомендувати себе як джерела медичної інформації та засоби для зв'язку пацієнтів з лікарями. Кількість відвідувачів цих сайтів, і величезна кількість сайтів свідчить про громадську інтерес до цих послуг [1]. Тому їм потрібно використовувати наявні технології для ефективного спілкування з ними пацієнтів, а також продовжувати покращувати їх і створювати нові.

Веб-сайт лікарні або поліклініки обов'язково повинен містити посилання на послуги та спеціальності клініки зі статтями про всі захворювання, їх симптоми та сучасні терапевтичні підходи разом із відповідною медичною ілюстрацією чи фотографією, а також описом її наукового обладнання.

Також необхідно, щоб веб-сайт містив детальне та актуальне резюме лікаря, якщо це веб-сайт для кабінету лікаря, або резюме асоційованих лікарів і медичних працівників, якщо розробляється веб-сайт для клініки чи лікарні.

Необхідно подбати про візуальне оформлення сайту, зокрема наявність фотографій внутрішнього та зовнішнього простору та обладнання кабінету чи клініки. Відповідні медичні зображення, фотографії та відео мають бути доступними на видних місцях, щоб привернути увагу потенційних пацієнтів і сприяти дружньому середовищу для відвідувачів веб-сайту.

У той час, як у багатьох відвідувачів веб-сайту лікарні можуть виникати загальні питання про стан здоров'я або потреби в медичних послугах, існує велика неоднорідність серед усіх відвідувачів. Отже, відвідувачам потрібна можливість пошуку, яка забезпечує пряму двосторонню взаємодію із самим сайтом.

Так як кожен пацієнт має унікальний набір потреб, що залежить від його медичних умов. У той же час, лікарня або клініка повинна знайти способи ефективного лікування цих дуже різноманітних захворювань, тому також необхідно, щоб на сайті був дискусійний форум, де відвідувачі можуть ставити запитання та отримувати відповіді, до яких інші відвідувачі також можуть легко отримати доступ, а також залишити відгуки та подяки.

Перетворення інтересу відвідувачів до лікарні у запис на прийом є однією з найважливіших цілей веб-сайту лікарні чи клініки. Ця можливість відрізняє веб-сайти лікарень та клінік від сайтів загальної інформації про здоров'я, наприклад Drkoop, WebMD тощо. Тому необхідно, щоб веб-сайт містив можливість для відвідувачів планувати відвідини лікарів онлайн. Наприклад, запис на прийом електронною поштою за допомогою онлайн-форми. Також слід включити графік відвідувань. Відвідувач може здійснювати пошук у календарі подій здоров'я за темою, за датою або місцем [1].

Соціальні мережі, контактна форма та контактна інформація мають бути на видних місцях разом із картою місцевості та інформацією про те, як дістатися до офісу, клініки чи лікарні. Важливо також розмістити інформацію про вартість різноманітних послуг, зокрема запису на прийом, а також розмістити список речей, які потрібно мати з собою у разі необхідності.

Багато комерційних сайтів використовують технології, які допомагають споживачам визначити, який продукт чи послуга найкраще відповідає їхнім конкретним потребам, а також застосовують певні технології для двостороннього спілкування з користувачами. Тому необхідно, щоб веб-сайт клініки або лікарні застосовував схожі технології для покращення та полегшення користування ресурсом для відвідувачів, наприклад, розміщення контактної інформації гарячої служби, чат-бот тощо [1].

Також важливим доповненням для сайту може бути регулярно оновлюваний блог із порадами щодо здоров'я та новинами в галузі медицини, конференціями, в яких беруть участь медичні працівники, публікаціями тощо.

Нарешті, але все ж важливо, медичний веб-сайт має правильно відображатися на всіх типах пристроїв і бути належним чином оптимізований, бути зручним та інтуїтивно зрозумілим для відвідувачів, і зручним для пошукових систем, щоб зробити його доступним для всіх.

Висновок. Розробка та вдосконалення веб-сайтів клінік і лікарень має бути в центрі уваги фахівців у сфері охорони здоров'я та інформатики.

Веб-сайти клінік та лікарень повинні не тільки забезпечувати дотримання етичних правил, бути функціональними та професійними з точки зору інформатики, але й надавати адекватну інформацію потенційним клієнтам медичних підрозділів. Веб-сайти – це не лише засіб спілкування, але й зручний ІТ-інструмент для кращого управління цими структурами.

Список використаних джерел:

1. <https://www.sciencedirect.com/science/article/pii/S0148296302003491>

УДК 004.4

МІКРОСЕРВІСНА АРХІТЕКТУРА У РОЗРОБЦІ КОРПОРАТИВНИХ ВЕБ-ДОДАТКІВ

Каліберда Ю.О.

e-mail: yuraklb@gmail.com

*Інститут промислових та бізнес технологій українського державного
університету науки і технологій*

Актуальність та постановка проблеми. Зростаючі вимоги до сучасних корпоративних веб-додатків, такі як можливість надання API, інтеграція з іншими веб-додатками через різні веб-служби, масштабованість, обробка великого кількості запитів, забезпечення необхідної швидкості доступу до інформації, виключення та мінімізація ризиків витоку корпоративних даних, призводять до того, що корпоративні веб-додатки з монолітною архітектурою стають незручними у розробці, складно тестуються та вводяться в експлуатацію з великими часовими затримками.

Мікросервісна архітектура полягає в розбитті моноліту системи на окремі компоненти, які є незалежними атомарними сервісами. Кожен окремих сервіс покликаний виконувати щодо нескладні функції. Сервіси мають чіткі фізичні межі, що робить їх масштабованими, дає можливість використовувати для написання різні технології та мови програмування. Необхідні зміни вносяться, не торкаючись весь додаток повністю.

Концепція декомпозиції складної системи корпоративного веб-додатку на мікросервіси - це спосіб створення веб-програми у вигляді набору невеликих сервісів, кожен із яких виконується як окремих процес, який обмінюється даними, використовуючи або синхронні протоколи, такі як HTTPs (HTTP) або асинхронні протоколи, такі як AMQP.

Мікросервіси можуть розроблятися та розгортатися незалежно один від одного друга. Кожен мікросервіс може мати свою власну базу даних, щоб бути відокремленою від інших служб. Мікросервіси інкапсулюють певний функціонал системи та створюються виходячи з особливостей предметної області корпоративного веб-додатку.

Основні матеріали дослідження. Основними перевагами мікросервісної архітектури є:

- Використання різних мов програмування та програмних засобів, оптимальні для реалізації кожного мікросервісу;
- Взаємозамінність мікросервісів;
- Незалежність мікросервісів один від одного, кожен мікросервіс може бути розгорнуто незалежно від інших служб;
- Спрощення процесу масштабування веб-додатку, що розробляється;
- Організація мікросервісів як модулів навколо окремих функцій;

До недоліків можна віднести наступне:

- Повідомлення між самими сервісами складне. Так як кожен функціональний елемент ізольований, потрібна особлива ретельність при побудові між ними грамотної комунікації, адже їм у будь-якому випадку доведеться обмінюватися запитами та відповідями один з одним.
- Складність тестування виявляється у тому, що спочатку потрібно окремо розбиратися з кожним сервісом, а потім тестувати коректну взаємодію

його з іншими мікросервісами.

- Мікросервіси гірше підходять для використання всередині окремих організацій, для них вони можуть виявитися невиправдано складними у застосуванні, тоді як для масових інтернет-сервісів вони підходять чудово.

Особливості експлуатації додатків із мікросервісною архітектурою. Згодом невеликі проблеми можуть призвести до серйозних. Тому своєчасний, а у разі мікросервісів, безперервний, моніторинг – необхідний. Моніторинг мікросервісної архітектури набагато складніший у налаштуванні і вимагає більших витрат за свою підтримку, ніж моніторинг монолітного докладання.

Як монолітна, так і мікросервісна архітектура, неспроможні уникнути відмов під час роботи додатку. Важливим принципом при створенні програми на мікросервісах є встановлення на те, що відмови системи рано чи пізно відбудуться. І з таким початковим припущенням і створюється програма. Перед випуском у продакшн код має бути ретельно протестований. Тим не менш, завжди є ймовірність, що якусь помилку буде втрачено. Саме для таких випадків придумано систему хаос-тестування. Помилки та проблеми створюються автоматично та навмисно – щоб заздалегідь вивчити їх та придумати швидке рішення або спосіб їх запобігти. Хмарна природа мікросервісів та віртуалізація, що лежить у їх основі, головною перевагою мають масштабування. Кожен сервіс виконує певну функцію, всі сервіси відносно невеликі – всі ці характеристики архітектури роблять масштабування більш простим і безболісним.

Висновок. У порівнянні з монолітами мікросервіси відкривають наступний рівень гнучкості операцій через створення нових бізнес-процесів навколо існуючих систем. Саме тому сьогодні багато хто націлений на мікросервісну трансформацію – її переваги надто очевидні. Оскільки мікросервіси незалежні один від одного, збої та дефекти в одному з них ніяк не впливають на роботу решти, система функціонує з мінімальними збоями та простоями.

Список використаних джерел:

1. https://ru.wikipedia.org/wiki/Микросервисная_архитектура;
2. Ньюмен С. Создание микросервисов. — СПб.: Питер, 2016. — 304 с.
3. Фаулер М. Архитектура корпоративных программных приложений.: Пер. с англ. — М.: Издательский дом "Вильямс", 2006. — 544 с.

УДК 004.43

СТВОРЕННЯ ІНФОРМАЦІЙНОЇ ВЕБ-СИСТЕМИ „ІНТЕРНЕТ-МАГАЗИН ПІЦЦЕРІЇ”

Катане О.Г.

e-mail: akatane18.19@gmail.com

Науковий керівник: к.т.н., доцент Лубко Д.В.

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Інтернету притаманний імідж «середовища безмежних можливостей». Однак у ньому зовсім небагато напрямків, де можна дійсно заробляти гроші. Один із них - інтернет-магазини. Вони дозволяють без великих витрат збільшити кількість покупців, запропонувавши їм можливість дистанційного замовлення та доставки товарів за місцем вимоги. Інтернет-магазини швидше прижилися в тих країнах, де вже давно існує система торгівлі за каталогами. Що таке Інтернет-магазин? Основна відмінність Інтернет-магазину від традиційного – у типі торгового майданчика. Якщо звичайному магазину потрібен торговий зал, вітрини, цінники, а також продавці, касири та досвідчені консультанти, то у його онлайнового «тезки» вся інфраструктура реалізована програмно. Іншими словами, Інтернет-магазин – це сукупність програм, що працюють на веб-сайті, які дозволяють покупцеві дистанційно вибрати товар із каталогу та оформити його замовлення [1]. Функції вітрини та торгового залу виконують «сторінки» з ілюстрованим каталогом товарів, а консультантів – підказки, інструкції та описи. Решта - як у звичайному магазині. Навіть в інтерфейсі інтернет-магазину зберігаються звичні елементи, наприклад віртуальний «візок» («кошик»), куди ми по дорозі до каси складаємо вибрані товари.

Метою даної роботи є створення інтернет-магазину піццерії з використанням сучасних веб-технологій, що наразі є дуже актуальною темою для того щоб у дані часи збільшити продаж продукції та підвищити прибутки таких фірм.

Основні матеріали дослідження. Виконаємо проектування інформаційної веб-системи (в подальшому сайту) на тему - „Інтернет-магазин Піццерії”. Першим етапом при створенні даного тематичного сайту буде розробка моделі варіантів використання системи.

Розробка моделі варіантів використання системи .

На рисунку 1 відображені актори та варіанти дії, які можуть виконувати актори в даній інформаційній системі в моделі варіантів.

Клієнт – це потенційний покупець, який може, переглядає вміст магазину і при бажанні робить замовлення. Адміністратор – це особа співробітників магазину, який допомагає клієнтові у виборі товару, отримує інформацію про зроблених замовленнях і інформує склад про статус замовлення.

Створення даного Інтернет-магазину ми проводимо за допомогою мов програмування PHP, HTML, CSS та фреймворку Bootstrap [2-4].

Наступним етапом при створенні даного сайту є розробка бази даних (БД).

Проведемо опис моделі даних та типів даних при створенні БД системи.

При описі предметної області набагато зручніше і ефективніше використовувати таке її уявлення, яке було б зрозумілим для більшості категорій користувачів, такий узагальнений погляд на базу даних і на її функціонування реалізується через концептуальне проектування [5].

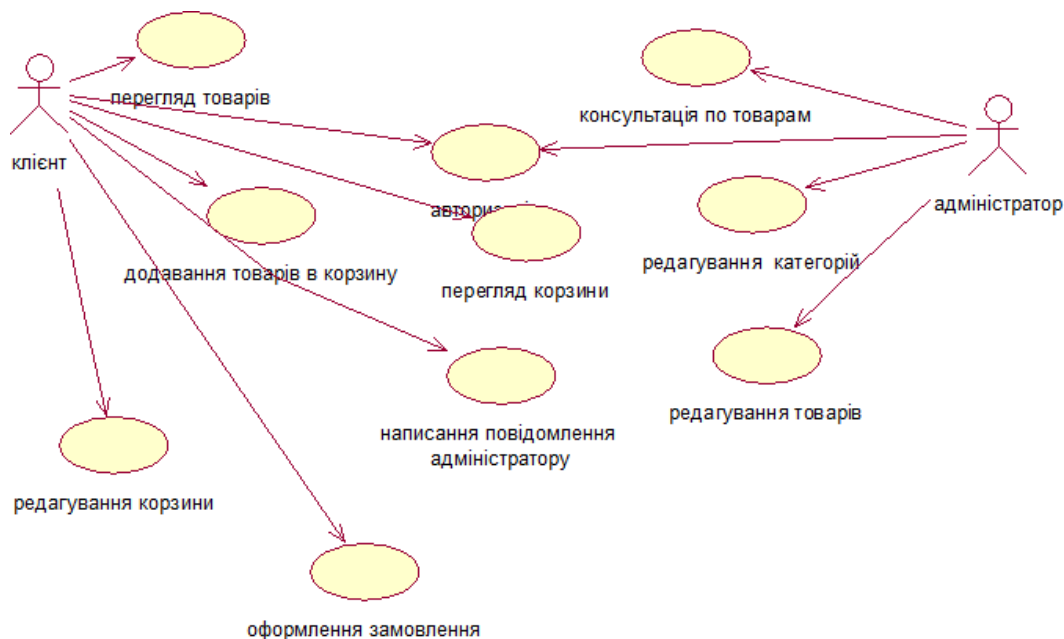


Рисунок 1. Діаграма варіантів використання системи

Концептуальна модель представляє об'єкти предметної області та їх взаємозв'язку без вказівки способів їх фізичного зберігання. Таким чином, концептуальна модель є, по суті, моделлю предметної області. При розробці таблиці «Категорії» в базі даних «Інтернет-магазин», база даних повинна містити дані які вказані на рисунку 2 [6].

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int(11)			Нет	Нет		AUTO_INCREMENT	Изменить Удалить Первичный
2	name	varchar(255)	utf8_general_ci		Нет	Нет			Изменить Удалить Первичный
3	sort_order	int(11)			Нет	0			Изменить Удалить Первичный
4	status	int(11)			Нет	1			Изменить Удалить Первичный

Рисунок 2. Структура БД таблиці Категорії

Сутність - будь-який конкретний або абстрактний об'єкт в аналізованій предметній області. Суті - це базові типи інформації, які зберігаються в базі даних (в реляційній базі даних, кожній суті призначається таблиця). До сутностей можуть ставитися: клієнти, підприємства і т.д. При складанні списку сутностей потрібно постаратися виділити всілякі, проаналізувати їх і видалити повторювані.

На основі аналізу предметної області, виділимо суті таблиці «Товари» БД системи (рис. 3).

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int(11)			Нет	Нет		AUTO_INCREMENT	Изменить Удалить
2	name	varchar(255)	utf8_general_ci		Нет	Нет			Изменить Удалить
3	category_id	int(11)			Нет	Нет			Изменить Удалить
4	code	int(11)			Нет	Нет			Изменить Удалить
5	price	float			Нет	Нет			Изменить Удалить
6	availability	int(11)			Нет	Нет			Изменить Удалить
7	brand	varchar(255)	utf8_general_ci		Нет	Нет			Изменить Удалить

Рисунок 3. Структура БД таблиці Товари

У таблиці нижче (рис. 4) представлено опис таблиці «Замовлення» БД.

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int(11)			Нет	Нет		AUTO_INCREMENT	Изменить Удалить
2	user_name	varchar(255)	utf8_general_ci		Нет	Нет			Изменить Удалить
3	user_phone	varchar(255)	utf8_general_ci		Нет	Нет			Изменить Удалить
4	user_comment	text	utf8_general_ci		Нет	Нет			Изменить Удалить
5	user_id	int(11)			Да	NULL			Изменить Удалить
6	date	timestamp			Нет	CURRENT_TIMESTAMP			Изменить Удалить
7	products	text	utf8_general_ci		Нет	Нет			Изменить Удалить
8	status	int(11)			Нет	1			Изменить Удалить

Рисунок 4. Структура БД таблиці Замовлення

У таблиці нижче (рис. 5) представлено опис таблиці «Користувачі» БД.

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int(11)			Нет	Нет		AUTO_INCREMENT	Изменить Удалить Первичный
2	name	varchar(255)	utf8_general_ci		Нет	Нет			Изменить Удалить Первичный
3	email	varchar(255)	utf8_general_ci		Нет	Нет			Изменить Удалить Первичный
4	password	varchar(255)	utf8_general_ci		Нет	Нет			Изменить Удалить Первичный
5	role	varchar(50)	utf8_general_ci		Нет	Нет			Изменить Удалить Первичный

Рисунок 5. Структура БД таблиці Користувачі

Далі створюємо структуру Інтернет-магазину.

Була виконана розробка та визначена структура Інтернет-магазину, яка приведена рисунку 6 [7-8].

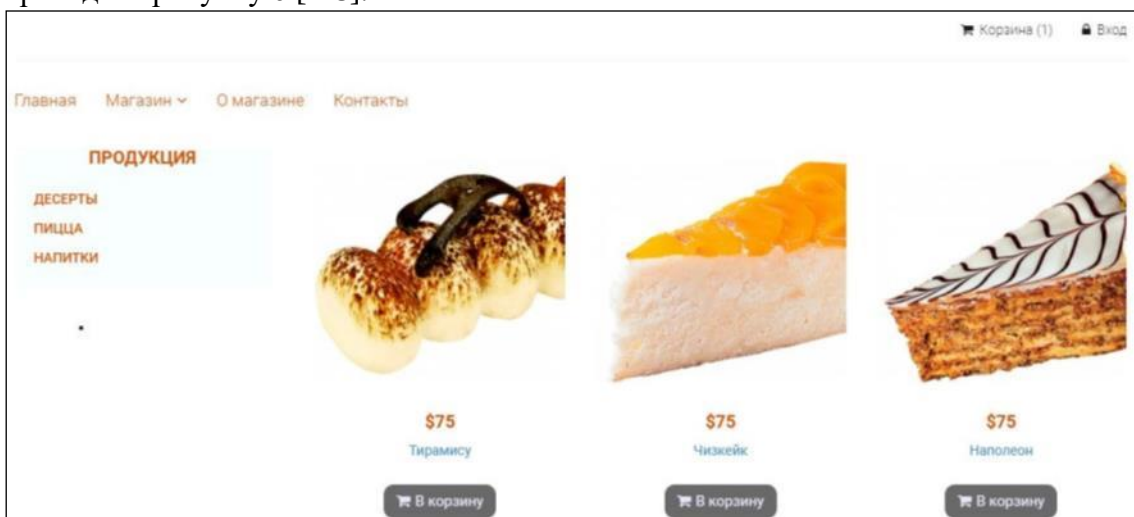


Рисунок 6. Структура Інтернет-магазину

Наступний етап - це проектування макетів сторінок системи.

На рисунку 7 відображено сторінку авторизації користувача.

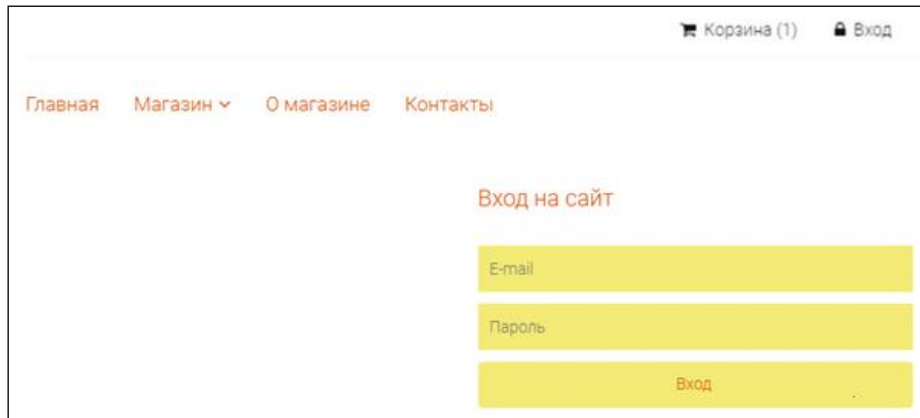


Рисунок 7. Авторизація користувача

На рисунку 8 відображено сторінку головну сторінку користувача після авторизації.



Рисунок 8. Головна сторінка після авторизації користувача

На рисунку 9 відображено головну сторінку редагування даних користувача.

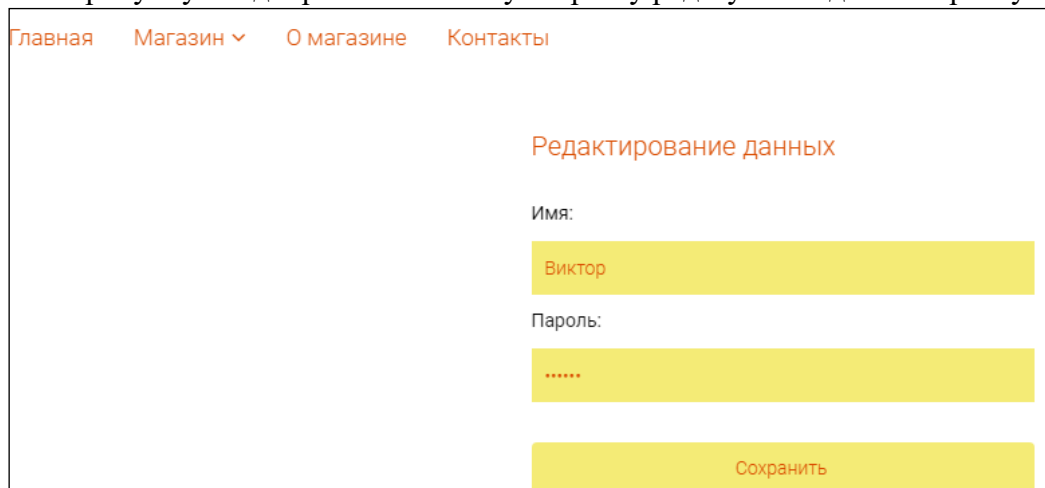


Рисунок 9. Редагування даних користувача

На рисунку 10 відображено головну сторінку товарів нашої піццерії.



Рисунок 10. Сторінка товарів магазину

На рисунку 11 відображено сторінку сортування товарів по категоріям сайту.

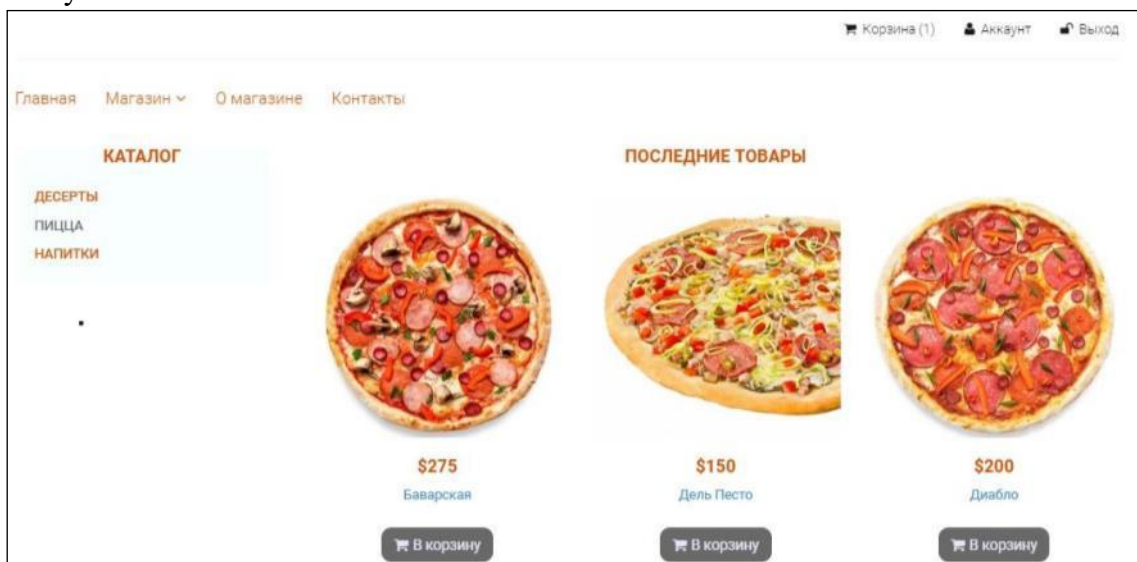


Рисунок 11. Сортування товарів по категорії

На рисунку 12 відображено сторінку корзини користувача сайту.

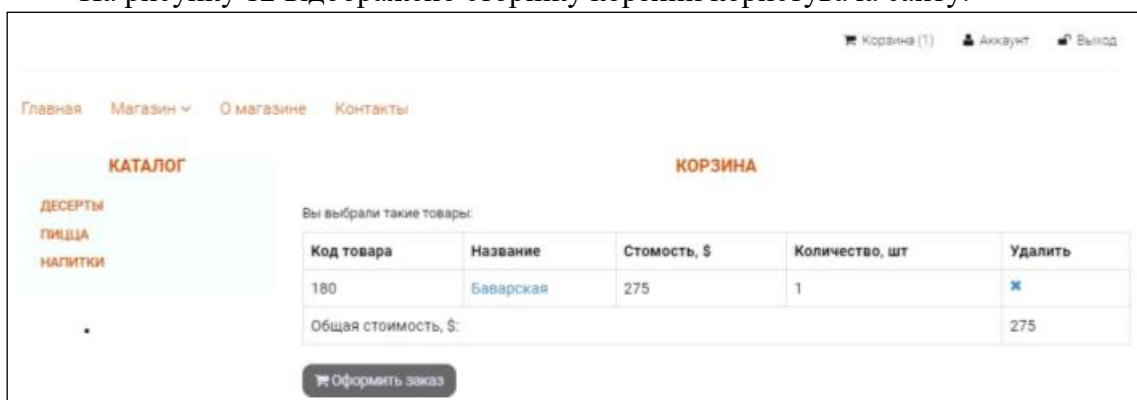


Рисунок 12. Корзина системи

На рисунку 13 відображено сторінку зворотного зв'язку сайту.

Рисунок 13. Сторінка «Зворотній зв'язок» сайту

На рисунку 14 відображено сторінку інформації про місцезнаходження магазину.

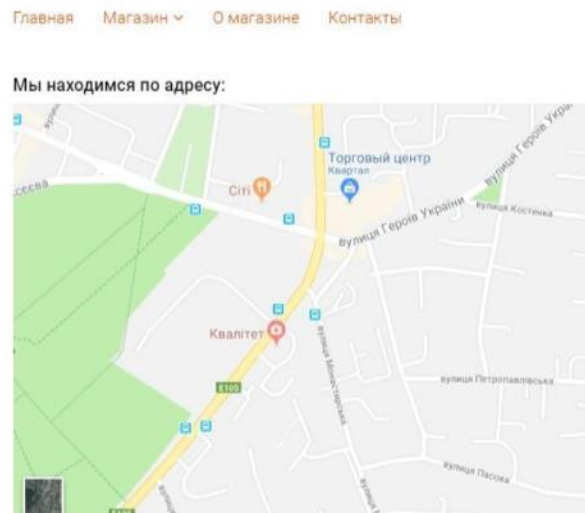


Рисунок 14. Інформація про місцезнаходження Інтернет-магазину

На рисунку 15 відображено розроблену головну сторінку адміністратора.

Рисунок 15. Сторінка адміністратора системи

На рисунку 16 відображено сторінку вибору пункту редагування.

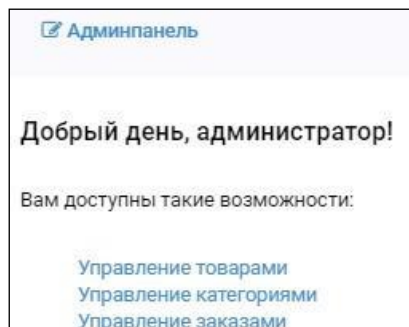


Рисунок 16. Елементи редагування Інтернет-магазину

На рисунку 17 наведено сторінку відображення наявних товарів піццерії.

+ Добавить товар					
Список товаров					
ID товара	Артикул	Название товара	Цена		
34	180	Баварская	275		
35	201	Дель Песто	150		
36	29	Диабло	200		
37	3	Coca-cola	65		
38	117	Sandora	45		
39	117	Наполеон	75		
40	117	Чизкейк	75		
41	117	Тирамису	75		

Рисунок 17. Список товарів Інтернет-магазину

Далі ми виконуємо програмування та створення серверної частини.

Серверна частина сайту була створена на мові програмування PHP. PHP - це широко використовувана мова сценаріїв загального призначення з відкритим вихідним кодом. Для цього ми створили файл Cart.php, який відповідає за управління товарами в корзині.

Далі ми виконуємо програмування та створення клієнтської частини.

За допомогою мов програмування PHP, HTML, CSS та фреймворку Bootstrap ми створили файл create.php, який відповідає за форму створення категорії.

Останній етап це розміщення магазину в Інтернеті.

Для розміщення Інтернет-магазину ми використали хостинг zzz.com.ua. Посилання на Інтернет-магазин за цією адресою, але термін розміщення на ньому на безоплатній основі обмежений коротким терміном.

Висновок. Під час роботи над темою було виконано створення інформаційної веб-системи „Інтернет-магазин Піццерії”. В процесі виконання роботи були ралізовані наступні завдання: виконано аналіз і опис предметної області; виконана розробка моделі варіантів використання системи; проведена розробка бази даних системи, а саме виконано опис моделі даних та типів даних; виконано проектування структури інтернет-магазину; виконано проектування макетів сторінок системи; виконано програмування серверної та клієнтської частини сайту; виконана авторизація користувача в системі; описано розміщення сайту-магазину в Інтернеті.

Програмна реалізація веб-системи та її опис проектування дозволить збільшити продаж продукції та підвищити прибутки підприємств, які займаються даним напрямком реалізації бізнесу.

Список використаних джерел:

1. <https://uk.wikipedia.org/wiki/Інтернет-магазин>
2. Флэнаган, Д. JavaScript. Подробное руководство [Электронный ресурс Флэнаган. -6-е изд. -Электрон. текстовые дан. Символ-плюс, 2012. -1080 с.
3. Фрэйн, Б. HTML5 и CSS3. Разработка сайтов для любых браузеров устройств [Электронный ресурс] Responsive Web Design with HTML5 and CSS3 Фрэйн. - Электрон. текстовые дан., 2014. - 304 с.
4. Роббинс, Дж. HTML5, CSS3 и JavaScript. Исчерпывающее руководство [Электронный ресурс] / Дж. Роббинс ; пер. с англ. М. А. Райтман. - 4-е изд. Электрон. текстовые дан. ЭКСМО , 2014. 528 с.
5. Martin Fowler. UML Distilled: A Brief Guide to the Standard Object Modeling Language, 3rd Edition. Addison-Wesley Professional. 2016. - p. 208.
6. Пасічник В. В. Організація баз даних та знань. – К.: Видавнича група BHV, 2006. – 384 с.
7. Бикнер К. Экономичный Web-дизайн / Кэрри Бикнер. – Киев: Пресс 2005. – 238 с.
8. Купер А., Рейман Р., Кронин Д. Алан Купер об интерфейсе. Основы проектирования взаимодействия. – Пер.с англ. Символ-Плюс, 2009. – 688 с.

УДК 004.89

ВИЗНАЧЕННЯ НЕСПРАВНОСТЕЙ КОМП'ЮТЕРА ЗА ДОПОМОГОЮ ПРОЕКТУВАННЯ ІНФОРМАЦІЙНО-ДОВІДКОВОЇ СИСТЕМИ

Лубко Д.В., к.т.н.

e-mail: dmytro.lubko@tsatu.edu.ua

Зінов'єва О.Г., ст. викладач

e-mail: olha.zinovieva@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. У кожного сучасного чоловіка сьогодні є ноутбук або стаціонарний персональний комп'ютер (ПК) за яким вона працює дома або на роботі. Так як наявність комп'ютера збільшує ефективність праці, то для покращання його роботи та продовження строку його служби слід регулярно проводити його огляд та діагностику. Цим займаються в спеціалізованих сервісних центрах. Але як тільки гарантійний термін, при покупці вашого комп'ютера збігає – займатися ремонтом вашої техніки вже не будуть безкоштовно. Ремонт будь-якої техніки зараз наразі коштує дуже дорого. Саме тому тема даної роботи є дуже актуальною.

У роботі ми будемо виконувати розробку інформаційно-довідкової системи для визначення несправностей комп'ютера (ноутбука, тощо).

При проектуванні системи важливо розуміти сутність поставленого завдання, а для цього розглянемо базові поняття та визначення.

Інформаційна система (англ. Information system) – це сукупність організаційних і технічних засобів для збереження та обробки інформації з метою забезпечення інформаційних потреб користувачів. За ДСТУ 2392-94: Інформаційна система - комунікаційна система, що забезпечує збирання, пошук, оброблення та пересилання інформації [1].

Також існують ще декілька визначень даного поняття. Розглянемо їх.

Інформаційна система - автоматизована система, комп'ютерна мережа або система зв'язку [2].

Інформаційна система - організаційно-технічна система обробки інформації за допомогою технічних і програмних засобів [3].

Інформаційна система - система, призначена для одержання, обробки, зберігання, відображення та/або реєстрації даних про технічний стан конструкцій, систем, елементів, їх властивості та/або функціонування [4].

Крім цього для проектування системи потрібно визначитися ще з наступними термінами.

Інформаційно-довідкова система - це організаційно впорядкована сукупність документів (масивів документів) та інформаційних технологій по збору, обробці, зберіганню та передачі інформації, яка використовується партнером при обробці запиту абонента [5].

Технічне діагностування (контроль технічного стану) - визначення технічного стану об'єкта з означеною (заданою) точністю [6].

Об'єкт технічного діагностування (контролю технічного стану) - виріб і (або) його складові частини, що підлягають (піддаються) діагностуванню (контролю) [6, 7].

Засіб технічного діагностування (контролю технічного стану) - апаратура та програми, за допомогою яких здійснюється діагностування (контроль) [6, 7].

Система технічного діагностування (контролю технічного стану) - це сукупність засобів, об'єкта та виконавців, необхідна для проведення діагностування (контролю) за правилами, встановленими технічною документацією [6, 7].

Завдання технічного діагностування - визначення (розпізнання, оцінка) технічного стану, у якому перебуває об'єкт на момент діагностування; оцінка відповідності чи невідповідності технічного стану об'єкта діагностування та у разі невідповідності - визначення причини невідповідності: несправності, дефекту, граничного ступеню зношеності, відхилення від регулювань тощо; виявлення ушкоджень чи дефектів на початковій стадії їх розвитку, виявлення конкретних дефектних вузлів чи деталей, визначення і усунення причин, що викликали дефект; оцінка допустимості та доцільності подальшої експлуатації обладнання з урахуванням прогнозування його технічного стану при виявленні дефекту, оптимізація режимів експлуатації, що дозволяє безпечно експлуатувати агрегат з виявленими дефектами до моменту його виводу у плановий ремонт; організація обслуговування та ремонту обладнання за технічним станом (замість регламентного обслуговування і ремонту), забезпечення підготовки та виконання якісних ремонтів.

Для здійснення технічного діагностування виробу необхідно [6]: встановити показники і характеристики діагностування; забезпечити пристосованість виробу до технічного діагностування; розробити діагностичне забезпечення виробу.

Розглянемо види технічного діагностування за організацією в часі.

Періодичне технічне діагностування - діагностування (контроль), за якого надходження інформації про технічний стан об'єкта відбувається через встановлені інтервали часу.

Безперервне технічне діагностування - діагностування (контроль), за якого надходження інформації про технічний стан об'єкта відбувається безперервно.

Оперативне технічне діагностування - діагностування, за якого надходження інформації про технічний стан об'єкта відбувається із завідома заданою стратегією в процесі функціонування об'єкта.

Види технічного діагностування за методом впливу на об'єкт.

Тестове технічне діагностування - діагностування, під час якого на об'єкт подаються тестові впливи.

Робоче технічне діагностування - діагностування, під час якого на об'єкт подаються робочі впливи.

Метою роботи є проектування такої системи для виявлення несправностей ПК та рекомендацій по їх усуненню, за допомогою якої можна буде попереджати виникненню несправностей [8]. Якщо вони все ж трапилися – усувати їх, та проводити періодичне діагностування різних окремих комплектуючих вашої комп'ютерної техніки. Цільова аудиторія даного програмного забезпечення – це по-перше домашні користувачі та фахівці-початківці сервісних центрів з початковою та середньою комп'ютерною освітою.

Дана система дозволить попереджати виникненню несправностей ПК (ноутбука), якщо вони все ж трапилися – засобам їх усунення, та проведенню періодичного діагностування різних окремих комплектуючих комп'ютерної техніки по певним ознакам.

Основні матеріали дослідження.

Перед початком проектування потрібно визначитися до якого класу буде відноситися наша система, яка розробляється. Згідно з класифікації (рис. 1) – за

рівнем автоматизації процесів управління вона буде відноситися до інформаційно-довідкових систем. А за рівнем централізації обробки інформації – до систем колективного використання.

Дана інформаційно-довідкова система – це фактично інструмент для швидкого діагностування одної з частин ПК, а саме - апаратної частини, зовнішніх пристроїв та операційної системи, або всіх частин одночасно. Головне завдання даної системи (програмного забезпечення) – надати особам, які не володіють в досконалому знаннями влаштування персональних комп'ютерів, чітку та детальну інформацію щодо існуючих дефектів системи, причин їх виникнення та варіантів вирішення проблеми (їх усунення).

Даний проект розробляється для користувача незалежно від його привілей та задач, які виконує система. Всю взаємодію користувача з системою можна показати за допомогою діаграми варіантів використання, яка зображена на рисунку 2.

Під час розробки програми була використана мова програмування Python, та графічний метод за допомогою редактора графічних форм - Qt Designer.

Python – це інтерпретована об'єктно-орієнтована мова програмування високого рівня з строгою динамічною типізацією [9; 10].

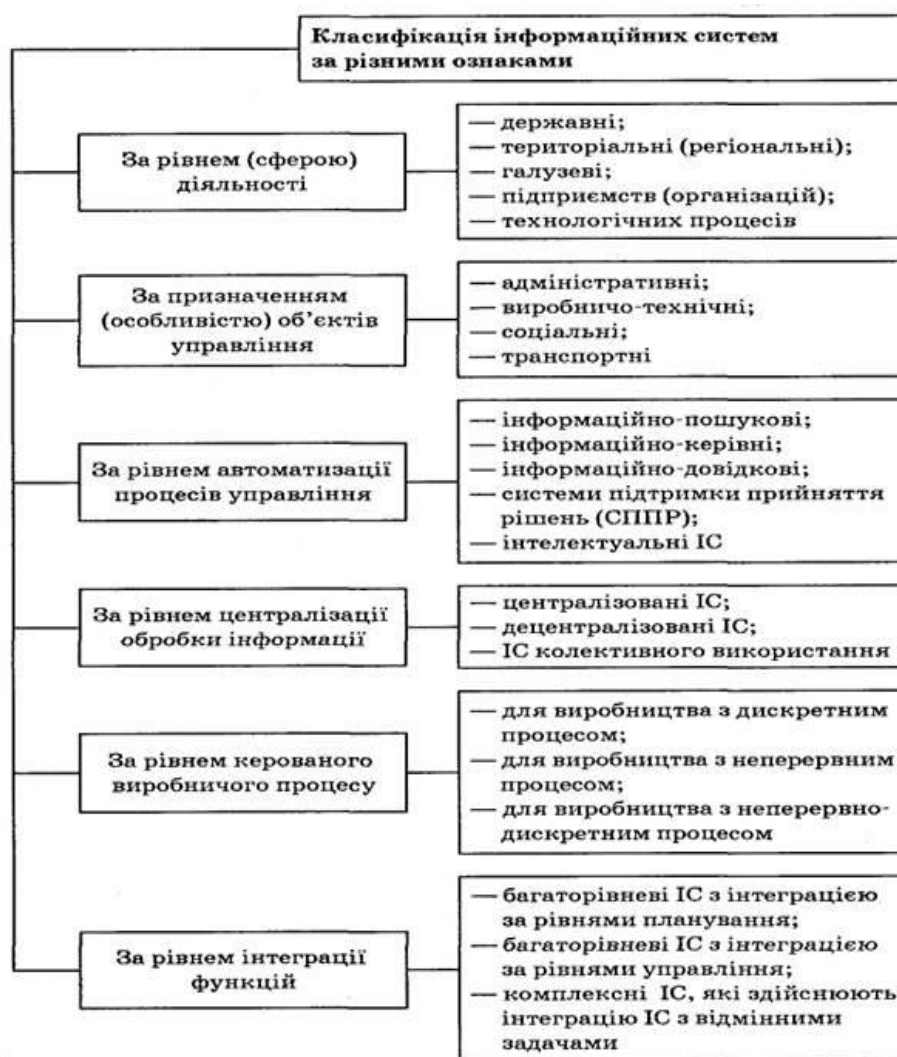


Рисунок 1. Класифікація інформаційних систем за різними ознаками



Рисунок 2. Діаграма варіантів використання

Структури даних високого рівня разом із динамічною семантикою та динамічним зв'язуванням роблять її привабливою для швидкої розробки програм, а також як засіб поєднання існуючих компонентів.

Python має ефективні структури даних високого рівня та простий, але ефективний підхід до об'єктно-орієнтованого програмування [13].

Ефективний синтаксис Python, динамічна обробка типів, а також те, що це інтерпретована мова, роблять її ідеальною для написання скриптів та швидкої розробки прикладних програм у багатьох галузях на більшості платформ.

Qt Creator – це інтегроване середовище розробки, призначене для створення крос-платформових додатків з використанням бібліотеки Qt [11]. Має вбудований редактор форм (Qt Designer) і довідкову систему (Qt Assistant). Результатом створення форм є файли з розширенням “.ui”, які за допомогою стандартних бібліотек PyQt5 можна з легкістю транслювати в об'єктно-орієнтований код мови Python з розширенням “.py”.

Інформаційно-довідкова система (програмний комплекс) має графічний інтерфейс [12], а саме головну форму та 4 закладки (окремні вікна). Завдяки закладкам можна розділити різні типи інформації.

На головній формі відображаються поля для введення вхідної інформації (моделі ноутбука, або материнської плати, тип процесора, модель відео карти, ємність HDD, тощо).

Далі на цій формі потрібно вказати (за допомогою перемикачів) – яка проблема трапилася з вашим ПК (ноутбуком).

Після цього у закладці „Проблеми” ми можемо побачити у вигляді списку – ті проблеми які ми вказали раніше. У закладці „Причини» вказується, що саме є причиною цієї проблеми. У закладці „Вирішення” – пропонується спосіб вирішення проблеми з ПК яка виникла. І у останній закладці – „Архів”, наведена дата діагностики, то коротко проблеми які виникли в цю дату з обраним пристроєм. Деякі вікна (закладки) розробленої системи діагностування наведені на рисунку 3.

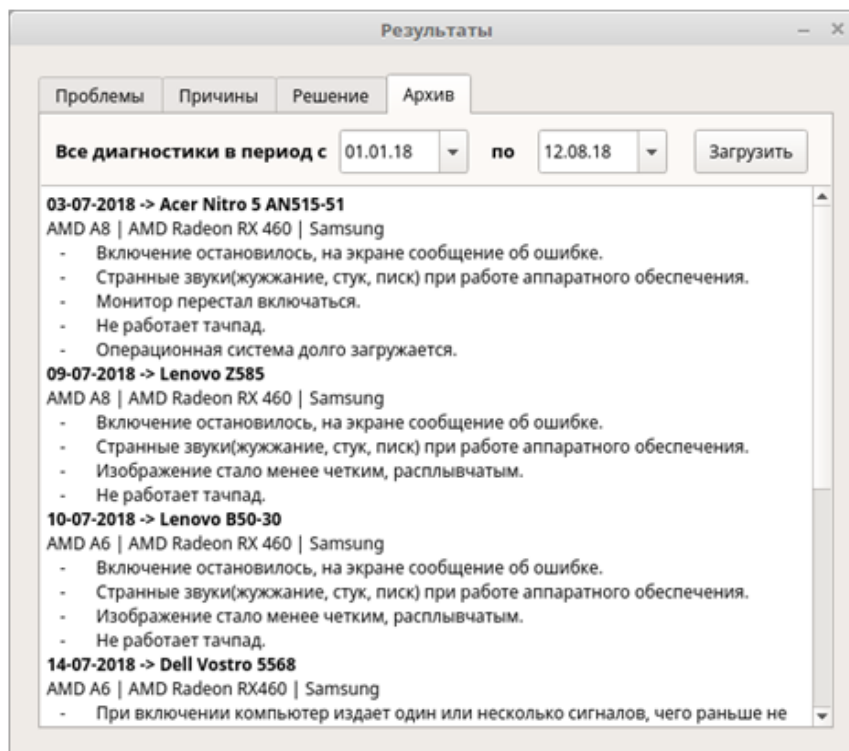
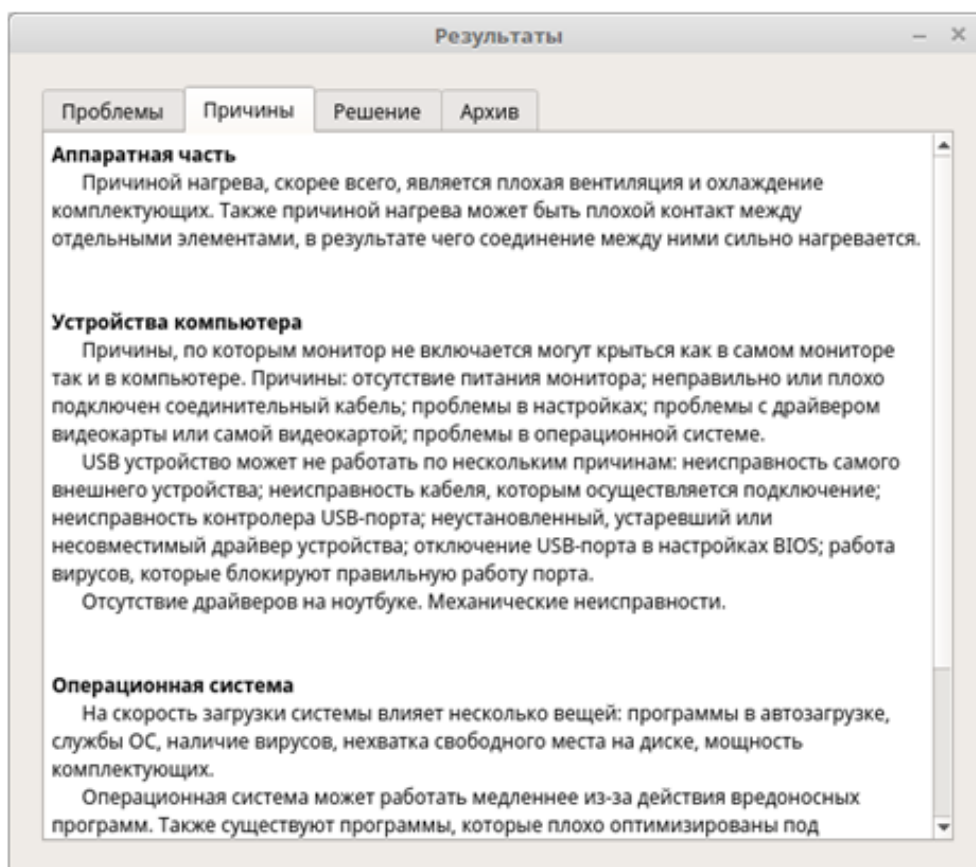


Рисунок 3 – Приклады вікон роботи розробленої системи

Для того щоб скористатися допомогою системи, користувач повинен створити новий запис. Для цього, на початковій сторінці, достатньо заповнити чотири текстових поля, які вказують на характеристики комп'ютера, що

діагностується. Дата діагностики автоматично вказує сьогоднішню дату, але, за потреби, її можна змінити. Після цього слідує три блоки питань, які сформовані таким чином, щоб охопити якомога більше деталей діагностування.

Потрібно відмітити, що для створення нового запису обов'язковими є тільки початкові характеристики. Якщо користувач не хоче створювати новий запис, а просто має намір передивитись архів, то ця можливість доступна з самого початку.

Способів зберігання та обробки даних (записів діагностики) є два.

Перший спосіб – це створення бази даних для записів та виведення під кожний окремий випадок. Такий спосіб - більш зручний у розробці, завдяки мові структурованих запитів (SQL), яка полегшує роботу з базою даних.

Але головний недолік цього способу – це залежність від серверу MySQL на тому комп'ютері, де запускається програма. Також таблиці будуть мати текстові записи, які становлять більший об'єм ніж стандартні типи даних в SQL.

Другий спосіб – це створення окремих файлів поряд з програмою, які будуть зберігати тимчасову інформацію. Такий спосіб є більш мобільним та незалежним від операційних систем та серверів інших програм. Завдяки формату даних JSON створюються записи. Цей формат є дуже зручним для обробки даних. Другий спосіб менш залежний від формату кодування операційної системи та від інших програм.

Завдяки створеним кнопкам у програмі відбувається перехід між вкладками. В залежності від поточної вкладки змінюється основний зміст сторінки.

Розроблена інформаційна система реалізує поставлені задачі, а саме: відповідає вимогам кросплатформеності; має простий, зручний та логічний інтерфейс; розроблена база знань (в кодї), на основі якої на основі вхідних даних надаються відповідні рекомендації.

Розроблена система має логічну структуру та оптимальне розташування елементів керування.

Тестування розробленого програмного забезпечення показало, що створена система працює швидко та якісно. Її завантаження проходить приблизно за 5 секунд, в залежності від потужності комп'ютера.

Проведена верифікація системи показала повну відповідність результатів всіх поточних етапів розробки системи умовам, сформованим на початку кожного етапу. А саме виконання цілям, термінам та завданню з розробки проекту, які були визначені на початку роботи.

Проведена валідація системи показала те що, між очікуваннями і потребами користувача є повна відповідність. А саме те, що розроблене програмне забезпечення повністю відповідає вимогам системи та технічному завданню. Проведений аналіз показав, що виведення рекомендацій системи відбувається без помилок, у відповідності до стандартів.

Висновок. Розроблена інформаційно-довідкова система реалізує поставлені задачі, а саме: відповідає вимогам кросплатформеності; має простий, зручний та логічний інтерфейс; розроблена база знань (в кодї), на основі якої на основі вхідних даних надаються відповідні рекомендації.

Використання даної системи дозволить будь-якому користувачу швидко та якісно визначати несправності персонального комп'ютера та отримувати шляхи їх вирішення без залучення сторонніх осіб та заощадити свої кошти.

Створену систему можна розширювати, як функціонально (інтерфейсно), так і програмно (наприклад, збільшити об'єм бази знань).

Як перспектива для розвитку, є можливість написання додаткових модулів у дану систему, а саме використовувати її як основу для створення інших систем, але вже для діагностування іншої електронної техніки.

Тому, розроблена система є досить широко-профільною, що дозволить її використання для всіх суміжних електронних галузей різного напрямку у майбутньому.

Список використаних джерел:

1. [https://uk.wikipedia.org/wiki/Інформаційна система](https://uk.wikipedia.org/wiki/Інформаційна_система)
2. Положення про технічний захист інформації в Україні, затверджене Указом Президента України від 27.09.1999 № 1229/99.
3. Порядок взаємодії органів виконавчої влади з питань захисту державних інформаційних ресурсів в інформаційних та телекомунікаційних системах, затверджений постановою Кабінету Міністрів України від 16.11.2002 р. № 1772.
4. Вимоги з ядерної та радіаційної безпеки до інформаційних та керуючих систем, важливих для безпеки атомних станцій, затверджені Наказом Державної інспекції ядерного регулювання України від 22.07.2015 № 140.
5. [https://uk.wiktionary.org/wiki/Інформаційно-довідкова система](https://uk.wiktionary.org/wiki/Інформаційно-довідкова_система)
6. ДСТУ 2389-94. Технічне діагностування та контроль технічного стану.
7. [https://uk.wikipedia.org/wiki/Технічне діагностування](https://uk.wikipedia.org/wiki/Технічне_діагностування)
8. Лубко Д.В. Зінов'єва О.Г. Шаров С.В. Проектування та розробка експертної системи діагностування несправностей транспортних засобів. Системи обробки інформації. Випуск 1(156). 2019. С. 15-21.
9. Python [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Python/>.
10. Майк МакГрат. Python. Программирование для начинающих / Майк МакГрат. – Издательство: Эксмо, 2015. – 194 с.
11. Qt Creator [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Qt_Creator/.
12. Графічний інтерфейс користувача [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Графічний_інтерфейс_користувача/.
13. Пол Бэрри. Изучаем программирование на Python / Пол Бэрри. – Москва: Эксмо, 2016. – 611 с.

УДК 004.415

МЕТОДИКА РОЗРОБКИ ТА ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ ВЕБ-СИСТЕМИ «БІБЛІОТЕКА»

Назаров Є.М.

e-mail: jevgenij.matvijovich.nazarov@gmail.com

Науковий керівник: к.т.н., доцент Лубко Д.В.

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Інтернет став невід'ємною частиною нашого життя. Подорожуючи по мережі Інтернет, ми зустрічаємо різноманітні веб-сайти. На одних продають речі, на інших запрошують знайти друзів для спілкування. Є широко затребувані інформаційні портали, а є сайти, присвячені окремим людям або компаніям. Тенденція така, що для успіху необхідно заявити про себе в Інтернеті, а значить, власна персональна сторінка повинна бути у будь-якої фірми і навіть у особистості. WWW - сторінки (веб-сторінки) представляють собою гіпертекстові документи. Веб-сторінки, об'єднані однією темою, що мають однаковий дизайн і знаходяться на одному веб-сервері, являють собою веб-сайт (веб-систему). Переглядати веб-сторінки можна за допомогою спеціальної програми – браузера.

З розвитком технологій гіпертекстової розмітки в Інтернеті стало з'являтися все більше сайтів, тематика яких є абсолютно різною - від сайтів крупних компаній, які показують успіхи компанії, до сайтів маленьких фірм, що пропонують відвідати їх офіси в межах одного міста.

Користувачам приємно відвідувати ті веб-сторінки, які мають стильне оформлення, не обтяжені надмірно графікою і анімацією, швидко завантажуються і правильно та коректно відображаються у вікні веб-браузера.

Завдання веб-дизайну будь-якого сайту – це викликати довіру відвідувача, до якості пропонованих нею послуг, зберігаючи при цьому фірмовий стиль, корпоративну культуру і позиціонування на ринку. Дизайн сайту і саме його побудова поєднує в собі різні нюанси, які необхідно враховувати при розробці. Перш за все, це наявність зручної навігації та прозорості структури сайту. При цьому потрібно враховувати цінність і витонченість графічного дизайну.

Створення і розробка сайтів включає: створення первісного технічного завдання на розробку системи (сайту); визначення структурної схеми сайту - розташування розділів, контенту і навігації; веб-дизайн - створення графічних елементів макету сайту, стилів і елементів навігації; розробка програмного коду, модулів, бази даних та інших елементів сайту необхідних для проектування; тестування і розміщення сайту в мережі Інтернет. Існує безліч засобів для створення веб-сайтів, але лише деякі з них здатні надати розробникам інструменти для вирішення переважної більшості завдань. При розробці веб-сайту з усіх сучасних веб-технологій, що дозволяють створювати інтерактивні веб-сторінки, необхідно вибрати найбільш підходящі для виконання поставлених на первинному етапі завдань.

Тому актуальність проектування тематичної веб-системи з докладним описом методології проектування є дуже актуальною.

Основні матеріали дослідження. Наведемо методологію розробки та проектування інформаційної веб-системи на прикладі такої предметної області як бібліотека з книжками.

Назвемо веб-систему - „Бібліотека” та опишемо кроки її побудови.

Початкове проектування починається з розробки макету інформаційної веб-системи (в подальшому просто сайту), тобто з визначення місця розташування блоків з різною інформацією.

Макетування сайту. Структурно дизайн сайту був поділений на три частини: меню, інформаційні блоки, робоча область, тобто вся інформація та додатковий контент, відображені у блоках (тег <div>), які з'являються у робочій області після натискання відповідного їм пункту меню [1-3].

Меню призначається для навігації по сайту, в ньому відображаються посилання на всі блоки сайту. Основний зміст сторінки змінюється в залежності від дій користувача.

Наведемо коротку інформацію про вміст меню розроблюваного сайту. Меню складається з: головної сторінки (ця сторінка відображається при першому вході на сайт, її оформлення і зміст дає користувачеві перше уявлення про сайт); вкладка «Усі книжки»; вкладка «За жанром»; вкладка «Фільми»; вкладка «Галерея»; вкладка «Цікаве»; вкладка «Цитати». Також на сайті звичайні та анімовані блоки

Директорія (каталог) сайту виглядає наступним чином: файл index.html (це головна html-сторінка); css (каталог css-файлів); images (каталог зображень);

На сайті знаходиться: текстова інформація (наприклад, історія створення книжки, зміст книги, анотації до кожної з них, основна інформація про сайт, тощо); мультимедійний контент; відео-файли.

Верстка сайту. На сайті була використана блокова верстка, тобто за допомогою блоків (тег <div>) та таблиць стилів (CSS3), анімація блоків була зроблена за допомогою JavaScript з використанням методу SetTimeout. Меню було розроблене завдяки маркованому списку з використанням таблиць стилів (CSS3) [4-6]. Динамічний фон сайту був створений завдяки псевдо-елементу after, властивостям animation-delay, opacity, function та іншим [2].

Наповнення сайту вмістом. В якості вміста була використана не тільки текстова інформація, а і мультимедійний контент: анімовані зображення, відео, аудіо. Для анімації зображень була використана таблиця стилів (CSS3).

Для того, щоб динамічний фон не відволікав відвідувача і не впадав в очі, був використаний графічний пакет Adobe Photoshop.

Розглянемо докладніше та наведемо інструкцію користувача сайтом.



Рисунок 1. Головна сторінка сайту

Виконаємо опис та проектування інструкції користувача сайтом.

Інструкція користувача сайтом наступна:

1) Після завантаження сайту можна побачити головну сторінку (рис. 1).

Головна сторінка сайту має шість посилань. Кожне з яких індивідуальне. Для перегляду інформації, тобто виклику необхідної сторінки з контентом, необхідно навести курсор на потрібний розділ меню та зробити клік лівою кнопкою миші.

2) Натиснув посилання «Все книги» ви перейдете на сторінку з усім об'ємом книжок, розміщених на сайті.

3) Натиснув посилання «По жанрам» перед вами з'явиться вибіркове меню. Вибравши жанр, ви переходите до вибору книжок, заданого жанру.



Рисунок 2. Вкладка «Все книги»

4) Натиснув посилання «Фильмы» ви перейдете на сторінку з усіма розміщеними екранізованими фільмами на сайті.

5) Натиснув посилання «Галерея» з'являться зображення, котрі поміщені на сайті.

6) Натиснув посилання «Интересное» з'явиться сторінка з цікавими моментами із життя книжок.

7) Натиснув посилання «Цитаты» ви будете перенаправлені на цікаві цитати з книжок, які обрані за думкою адміністратора.

8) Вкладка «Все книги» має в собі стрічку з усіма наданими книжками у асортименті. При натисканні на ту або іншу книжку, з'являється вибрана вами вкладка. На вкладці розположені: анотація, обкладинка книжки та можливі дії, такі як посилання для читання, скачування та перегляду онлайн. Для повернення на головну сторінку є кнопка «На главную».

9) Вкладка «По жанрам» має в собі 4 додатки, в яких розміщуються книги, відсортовані за книжними жанрами. Це дуже зручно. Так як, не потрібно прокручувати всі книжки для пошуку потрібної. Варто, лише, вибрати потрібний вам жанр. Перейти на нову сторінку (перехід відбувається автоматично, при натисненні лівої кнопки миші на потрібному жанрі) та в тому списку знайти

потрібну вам книжку. Для зручності пошуку, книжки розташовані таблицею. Книжку можна побачити за обкладинкою, або назвою.



Рисунок 3. Вкладка «По жанрам»

10) Натиснувши на той або інший жанр, додаток відкриється у новому вікні. Вибравши потрібну книжку, натисніть на ній лівою кнопкою миші. Відбудеться перехід на вкладку розміщеної книжки. Перехід буде здійснений на нову сторінку.

11) На вкладці з книжкою є анотація до, вибраної вами, книжки; обкладинка; три можливі дії: посилання для читання онлайн, скачування на ПК та перегляд екранізації книжки онлайн. Перехід здійснюється за кліком лівої кнопки миші. Для повернення на головну сторінку є кнопка «На главную».



Рисунок 4. Вкладка «Фильмы»

12) Вкладка «Фильмы» містить посилання на перегляд в режимі онлайн. Для відкриття схованого контенту, потрібно навести курсор миші на слово

«Выбор». Після цього з'являться посилання у вигляді круглих зображень. Зображення вибрані таким чином, щоб користувач сайту міг знайти потрібний йому фільм, за кадром з цього фільму та назвою.

При виборі потрібного фільму, треба натиснути на зображення лівою кнопкою миші. Відбудеться автоматичний перехід на нову сторінку. На цій сторінці буде відбуватися перегляд фільму в режимі онлайн. Для повернення на головну сторінку треба натиснути кнопку «На главную».

Після всіх проведених в подальшому тестувань сайту планується розмістити його на платному хостінгу (зараз це локальна веб-система).

Висновок. Під час роботи над темою, була розроблена методологія та описані етапи проектування інформаційної веб-системи „Бібліотека”. Наведена докладна структура тематичної інформаційної веб-системи.

Розроблена інформаційна веб-система „Бібліотека” буде корисною для використання у першу чергу у бібліотечній сфері, а також як демонстрація етапів проектування для схожих веб-систем – для усіх зацікавлених даною проблематикою.

Список використаних джерел:

1. Стив Круг. Как сделать сайт удобным. Юзабилити по методу Стива Круга. - Киев: Пресс. 2010. - 208 с.
2. Бикнер К. Экономичный Web-дизайн / Кэрри Бикнер. – Киев: Пресс 2005. – 238 с.
3. Молчанов В.П. Основы проектирования WEB-видань. Конспект лекцій. – Харків: Вид. ХНЕУ, 2008. – 168 с.
4. Роббинс, Дж. HTML5, CSS3 и JavaScript. Исчерпывающее руководство [Электронный ресурс] / Дж. Роббинс ; пер. с англ. М. А. Райтман. - 4-е изд. Электрон. текстовые дан. ЭКСМО , 2014. - 6 файл ; 528 с. - (Мировой компьютерный бестселлер)
5. Флэнаган, Д. JavaScript. Подробное руководство [Электронный ресурс Флэнаган. -6-е изд. -Электрон. текстовые дан. Символ-плюс, 2012. -1080 с.
6. Фрэйш, Б. HTML5 и CSS3. Разработка сайтов для любых браузеров устройств [Электронный ресурс] Responsive Web Design with HTML5 and CSS3 Фрэйш. - Электрон. текстовые дан., 2014. - 304 с.

УДК 004.415.2

ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ ГРОМАДСЬКОЇ ВОЛОНТЕРСЬКОЇ ОРГАНІЗАЦІЇ

Нарватов О.П.

email: oleksandr.narvatov@nure.ua

Полозов Д.М.

email: denys.polofov@nure.ua

Широкопетлева М.С. ст. викл.

email: marija.shirokopetleva@nure.ua

Харківський національний університет радіоелектроніки

Актуальність та постановка проблеми. Благодійність – безкорисливе надання допомоги тим особам, які її потребують найбільше, але не в змозі самостійно забезпечити потреби через певні обставини у буденному житті, під час карантину чи в умовах військового стану.

Благодійна діяльність часто реалізується через організації такого спрямування як фонди чи за допомогою волонтерів. Значна кількість волонтерів займається збором коштів, закупкою необхідних товарів, ліків, продуктів харчування, допомагають з перевезенням та евакуацією, допомагають отримати фахову медичну допомогу, допомагають переселенцям тощо.

За результатами опитування, 57% українців з центру та заходу країни долучалися до волонтерства чи благодійності [1]. Таким чином зросла потреба в інформаційній системі, що буде координувати чи об'єднувати між собою волонтерів, а також осіб що потребують допомоги.

Саме тому необхідно спроектувати систему, що надає можливість користувачам надавати нематеріальну допомогу іншим або просити її. До того ж система повинна враховувати всі недоліки вище приведених систем.

Основні матеріали дослідження. Збільшити об'єм волонтерської допомоги можна декількома способами. Перший – збільшення кількості волонтерів, що приймають участь в наданні допомоги, оскільки більше немає необхідності вступати в волонтерську організацію. Другий – зняття певної частки навантаження з вже існуючих центрів надання допомоги. Інформаційна система збирає всі запити на допомогу(покупка ліків, побутовий ремонт), а також пропозицій по її наданню (безкоштовну доставку товарів містом). Така система зменшить відповідне навантаження з волонтерів, а також долучить нових.

Взаємодію між учасниками волонтерського руху можна розподілити за двома основними напрямками - взаємодія між конкретним отримувачем допомоги та тим, хто може надати таку допомогу, а також сумісна робота групи учасників для вирішення задач протягом певного часу (організація громадської події).

Проблематика. На даний момент існує значна кількість фондів чи сервісів, що збирає кошти чи надає загальну допомогу наприклад СпівДія — волонтерська P2P-платформа [2] чи платформа Dobro.ua [3].

Можна виділити ряд проблем цих платформ.

- жодна з них не надає можливість надавати адресну допомогу до кінцевих споживачів;
- можливість реєстрації надання вузько спеціалізованої допомоги від малого бізнесу (наприклад вставка вікон);
- неможливість створити запит на адресну матеріальну, не грошову допомогу;
- неможливість напряду зв'язатись з виконавцем/споживачем допомоги.

Таким чином, жодна з цих платформ не надає можливість раціонально взаємодіяти будь-яким користувачам, які можуть в різних ситуаціях виступати і як волонтери, так і ті, хто потребує допомоги певного виду.

Розробка. В першу чергу необхідно визначити вимоги до інформаційної системи. Коротко перелічимо деякі з них. Інформаційна система повинна мати Web і Mobile частини, для роботи у браузері та смартфоні, з операційною системою Android. Система повинна підтримувати розподілення користувачів за ролями: адміністратор, організатор подій, учасник та гість.

«Адміністратор» виконує адміністрування системи, обробку скарг «Користувачів», для вирішення конфліктних ситуацій. За рішенням адміністратора контент може бути видаленим, а користувач може бути заблокованим.

«Організатор івентів» створює громадські події з можливістю заповнити детальну інформацію щодо формату проведення, часу, тип роботи. Для івенту організатор має створити ролі для користувачів та завдання, кожне з яких може мати різні набори ролей. Після завершення івенту, організатор повинен прикріпити звіт, щодо виконаної роботи.

«Учасник» створює запити на допомогу, на які можуть відгукнутись інші учасники та може самостійно надати свою допомогу на інші запити. Кожен учасник також може подати заяву на участь у громадському івенті та орати бажану для себе роль.

«Гість» має можливість перегляду громадських івентів та запитів на допомогу.

Для розробки серверної частини пропонується обрати хмарний провайдер AWS, щоб мати можливість зручно проводити розгортання програми. AWS було обрано за велику кількість сервісів та їх простоту використання, можливість створювати Serverless застосунки та підтримки користувачів платформи. Серверну частину буде створено за допомогою Cloud-Native дизайну, що надасть змогу розгорнути її у Elastic Kubernetes Service з рантаймом для контейнерів Docker. Також, за допомогою Kubernetes отримаємо відмовостійкість та гнучкість налаштувань. Основною мовою програмування було обрано Go, через її високий потенціал у створенні програм у Cloud-Native стилі. Основною СУБД було обрано PostgreSQL - за її надійність та принцип SQL бази даних ACID [4].

Висновок. Для збільшення кількості охочих займатись волонтерським рухом, слід створити відповідні умови для входу у цей простір, що зробить процеси більш зрозумілими та доступними. Для цього слід реалізувати як браузерну версію благодійної інформаційної системи, так і мобільну, щоб волонтер мав можливість для доступу як вдома, так і на вулиці. Інтерфейс повинен бути зрозумілим, простим та інформативним. Для уникнення шахрайських схем, слід реалізувати адмін панель, для виконання адміністрування системи.

Список використаних джерел:

1. Більш як половина українців волонтерили чи долучалися до благодійності під час війни. URL: <https://www.ukrinform.ua/rubric-uarazom/3501782-bils-ak-polovina-ukrainciv-volonterili-ci-dolucalisa-do-blagodijnosti-pid-cas-vijni.html>.
2. СпівДія. URL: <https://spivdiia.org.ua/>
3. Dobro.ua. URL: <https://dobro.ua/>
4. Hans-Jürgen Schönig. Mastering PostgreSQL 12: Advanced techniques to build and administer scalable and reliable PostgreSQL database applications, 3rd Edition. 2019. P. 20-34.

ПЕРЕВАГИ ТА НЕДОЛІКИ ГНУЧКОЇ МЕТОДОЛОГІЇ SCRUM ДЛЯ УПРАВЛІННЯ ІТ ПРОЕКТАМИ

Савчук Ю.В., студент

email:uitpm221_syuv@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Від правильності управління ІТ проектом залежить його успішність. Велика конкуренція на ринку серед ІТ компаній змушує покращувати внутрішні процеси всередині компаній. Проекти вимагають організації гнучкої взаємодії з замовником, компетентних методів управління виконавцями проекту. До того ж в процесі роботи можуть виникнути непередбачувані обставини, тож управління проектом повинно бути максимально підготовлене до майбутніх складнощів. Менеджмент прагне скоротити строки виконання проекту так як від цього залежить прибуток компанії, покращити результат щоб задовольнити замовника, а також оптимально використовувати ресурси. Це необхідно щоб компанія була успішною та конкурентно спроможною на ринку надання ІТ послуг. Так як всі ІТ компанії стикаються з подібними проблемами то з часом з'явилися різні методології управління ІТ проектами. Перевагами використання існуючих методологій є те, що вони мають зрозумілі правила та принципи. Кожна із методологій має окремі області застосування та передумови, аналізуючи які можна вибрати яка із методологій буде оптимальною у кожному окремому випадку.

Основні матеріали дослідження. Найпопулярнішою адаптивною методологією управління проектами зараз є Scrum. Ця методологія має зрозуміло сформульовані правила та процес роботи. Фреймворк передбачає управління проектом шляхом розбиття його на кілька частин та проходження циклу планування, оцінювання та виконання на кожному етапі. Цикли Scrum-у більш відомі як спринти, і вони зазвичай охоплюють від двох до чотирьох тижнів. В кінці кожного спринта результат роботи перевіряється, і робляться висновки, що потрібно покращити. Це є вагомою перевагою цієї методології перед іншими, адже команда безперервно вчиться на свої помилках та стає більш сильною і підготовленою до кожного наступного спринта або проекту. Методологія є досить гнучкою, починати розробку можна навіть тоді коли ще не зрозуміло який буде результат в самому кінці виконання проекту.

Учасники команди відіграють три різні ролі у процесі розробки: власник продукту, який відповідає за кінцевий результат, Scrum мастер, який організатором робочого процесу в команді, а також команда спеціалістів, яка розробляє проект і складається з одного або більше людей.

Scrum особливо корисний в індустрії розробки програмного забезпечення оскільки він допомагає команді концентруватися на досягненні менших цілей задля подальшої інтеграції окремих частин в цілісний продукт і досягненні головної мети. При цьому результатом виконання кожної ітерації має бути об'єм робіт, які потенційно можна передати замовнику, як частину виконаної роботи. Однак інструменти фреймворку можна використовувати для широкого спектру проектів у багатьох галузях, особливо тих, які вимагають швидкої відповіді на термінові запити та гнучкої команди розробки проектів.

Підхід Scrum найкраще застосовується у відносно невизначеному середовищі, де нелегко визначити вимоги та рішення перед тим, як точно розпочати проект. Команда scrum може внести зміни в будь-яку частину проекту, не впливаючи на результат проекту. Гнучкість і адаптивність мають важливе

значення для визначення вимог проекту, коли він триває. А також методологія забезпечує можливість виходу на ринок на ранній стадії, оскільки для запуску потрібно короткий час. Подальше доповнення проекту новим функціоналом також призводить до більш швидкого виходу на ринок без обов'язкового завершення всього проекту. Бізнес-середовище стає все більш конкурентним, і кожен прагне досягти кращого. Scrum наголошує на творчості та інноваціях; таким чином, його загальний результат завжди вражаючий і відмінної якості. Підхід Scrum знижує витрати за рахунок скорочення документації та засобів контролю, які не є значними, таким чином знижуючи накладні витрати. Це також сприяє підвищенню продуктивності команди в цілому. Ця перевага є надзвичайно важливою для замовників, тому що вони не хочуть платити гроші за документацію, вони хочуть платити за продукт який буде корисним для них. А також методологія гарантує що співробітники дотримуються вимог до виконання їхньої роботи, і створює прозорість процесів перед замовниками. Найменші зміни в будь якій частині проекту є видимими для кожного учасника. Прозорість призводить до довіри, а компанія отримує більше клієнтів. Методологія має встановлені дедлайни та зрозумілі очікування для кожного із учасників робочого процесу. Спеціалісти завжди старанно працюють як команда для досягнення їхньої спільної цілі. Їх також мотивує винагорода, яку вони отримують, виконавши завдання вчасно та переконавшись, що воно виконано правильно. Таким чином, методологія створює комплексний набір знань, який команда може продемонструвати клієнту.

Методологія також має ряд недоліків. Перший з них це те що кожен член команди прагне досягти його персональних дедлайнів виконання певної кількості роботи, але при цьому дедлайн виконання проекту в цілому не береться до уваги, що може затримати розробку. Наступний недолік це те, що методологія не є досить зручною для великих команд, які складаються з десяти або більше спеціалістів, так як така велика кількість людей буде гальмувати прийняття рішень. Методологія більше підходить для менших команд, які складаються з трьох та більше людей. Наступним недоліком є те що всі учасники робочого процесу повинні мати певний досвід або теоретичні знання для роботи із Scrum процесами. Недолік досвіду хоча б одного із спеціалістів може гальмувати увесь процес, не кажучи вже про те що Scrum мастер повинен мати необхідні знання для побудови всіх процесів.

Висновок. Отже Scrum є сучасною та надзвичайно зручною методологією для більшості ІТ проектів, при цьому її використання є зручним для всіх зацікавлених сторін, як для замовника так для команди розробників та менеджменту. Вона використовується великою кількістю ІТ компаній для розробки проектів будь якої складності. Методологія має велику кількість переваг перед іншими схожими методологіями. Проте потрібно пам'ятати, що існують і недоліки, тож потрібно з розумом віднестись до вибору методології, і використовувати її там де вона є корисною, і не використовувати там де недоліки переважають переваги.

Список використаної літератури:

1. Роб коул, едвард скотчер. Блискучий agile. Практичний посібник для проект-менеджерів із використання agile, scrum, kanban. 2020.
2. Scrum project management: advantages and disadvantages. 2022 url: <https://www.simplilearn.com/scrum-project-management-article> (дата звернення: 20.11.2022)

УДК-004.8

ВИКОРИСТАННЯ ТЕХНОЛОГІЇ DOCKER В РОЗРОБЦІ ВЕБ-ОРІЄНТОВАНОГО ДОДАТКУ «ПЛАТФОРМА СТВОРЕННЯ ОНЛАЙН ОГОЛОШЕНЬ»

Вакалюк Т.А., д. пед.н., проф.

e-mail : kik_vta@ztu.edu.ua

Кияшенко А.С.,

e-mail: ipzm221_kas@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Веб-додатки – це найбільш розповсюджений тип програмного забезпечення. Кожен персональний комп'ютер чи смартфон має вбудований браузер, що дозволяє користуватися веб-додатками без необхідності встановлювати для цього спеціальне програмне забезпечення. Веб-додатки бувають різні. Це може бути односторінковий сайт-портфоліо, в якому відсутня внутрішня логіка та алгоритми. Подібні проекти можуть бути розроблені однією людиною, що спеціалізується на верстці веб-сайтів. Але це також може бути велика система, в розробці якої приймає участь ціла команда спеціалістів. При такій розробці дуже важливо, щоб у кожного була однакова система, з однаковими версіями веб-серверу, бази даних, мови програмування тощо. В іншому випадку розробник може зіштовхнутися з конфліктами чи помилками, що не будуть відтворюватися на локальних проектах інших розробників. Також можуть бути проблеми при розгортанні, адже на виділеному під розгортання сервері версії компонентів системи теж мають бути однакові з локальними проектами.

Основні матеріали дослідження. Для вирішення цієї проблеми було розроблено програмне забезпечення під назвою Docker. Docker – це контейнеризатор додатків. Він дозволяє відокремити кожен компонент системи в окремий контейнер, автоматизувати їх запуск та розгортання та керувати їх життєвим процесом. Контейнеризація – це спосіб упакування додатку та його залежностей в окремий образ, що запускається в ізольованому віртуальному середовищі, що не впливає на основну систему. На відміну від віртуалізації це не емулює окрему операційну систему та віртуальне обладнання. Контейнеризація використовує ядро основної операційної системи і використовує реальне обладнання, за рахунок чого не потребує додаткових ресурсів. Це дозволяє знехтувати реальним середовищем розгортання і буде неважливо, які саме налаштування та залежності в ньому будуть. Головна умова, це наявність Docker та підключення до мережі Інтернет.

Образи можна створювати самостійно та можна встановити із відкритого репозиторію DockerHub. DockerHub має більше ста тисяч готових образів, які можуть виставляти розробники зі всього світу. Багато з них завантажені офіційними розробниками (Nginx, MySQL, Apache, Python, PHP та інші). Підтримка цього репозиторію вже включена в Docker, тому потрібно лише написати команду для установки потрібного образу чи внести цей образ в конфігурацію і він буде встановлений самостійно при першому розгортанні проекту.

Docker вирішує наступні задачі:

- Розгортання середовища чи додатку. Docker дозволяє перенести додаток з усіма залежностями та налаштуваннями на іншу систему. Налаштування вручну займає набагато більше часу, а у випадку з

Docker потрібно лише один раз створити файл конфігурації, що буде переноситися між робочими машинами.

- Ізольований запуск. Docker запускається окремо від основної системи і не викликає конфліктів між програмами.
- Підвищення безпеки. Якщо контейнеризований додаток має вразливість, це не вплине на основну систему і не вплине на роботу інших додатків.
- Прискорення циклу розробки. Немає необхідності витратити час на налаштування середовищ.

Висновок. Docker вирішує велику проблему веб-додатків – їх розгортання. Саме тому на даний момент це одна з найбільш затребуваних технологій. Docker вже використовують такі великі компанії, як PayPal, Uber, eBay, Shopify, Spotify, Quora тощо.

Список використаних джерел:

1. <https://www.section.io/engineering-education/why-is-docker-so-popular/>

УДК 004.77

ТЕОРЕТИЧНІ АСПЕКТИ ОРГАНІЗАЦІЇ АУТЕНТИФІКАЦІЇ, АВТОРИЗАЦІЇ ТА АУДИТУ В СУЧАСНИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ

Білявський Н.А., магістрант

e-mail: kim221_bna@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Комп'ютерна мережа має бути спроектована так, щоб була можливість контролювати, кому і коли дозволено до неї підключатися і що їм дозволено робити. Ці специфікації дизайну мережі визначені у політиці безпеки мережі. Політика визначає доступ до мережевих ресурсів мережевих адміністраторів, корпоративних користувачів, віддалених користувачів, ділових партнерів та клієнтів. Політика мережної безпеки також може визначати систему звітності, де реєструється, хто і коли входив до мережі та що вони у ній робили. Система керування доступом до мережі з використанням лише режиму користувача або тільки команд паролів привілейованого режиму обмежена і не дуже добре масштабується. Використання протоколу аутентифікації, авторизації та обліку (AAA) створює необхідну структуру для масштабованого захисту доступу.

Основні матеріали дослідження. Аутентифікація, авторизація та аудит, які часто називають AAA або Triple-A, - це набори служб і протоколів, які дозволяють детально контролювати доступ до комп'ютерних мереж. До популярності основних протоколів аутентифікації на основі HTTP, таких як OAuth і SAML, протоколи AAA були основним способом аутентифікації користувачів або машин для мережевих служб. Протоколи AAA як і раніше часто зустрічаються в корпоративних мережах LAN і WAN, провайдерах, стільникових мережах і мережевих пристроях, таких як брандмауери, маршрутизатори і комутатори.

Аутентифікація відноситься до процесу, в якому аутентифікується особа суб'єкта, як правило, шляхом надання доказів того, що він володіє певною цифровою ідентифікацією, наприклад ідентифікатором і відповідними обліковими даними. Прикладами типів облікових даних є паролі, одноразові маркери, цифрові сертифікати та цифрові підписи.

Функція авторизації визначає, чи уповноважений певний об'єкт виконувати певну дію, яка зазвичай успадковується від аутентифікації під час входу в програму або службу. Авторизація може бути визначена на основі ряду обмежень, наприклад: обмеження часу, обмеження фізичного розташування або обмеження для множинного доступу одного і того ж об'єкта або користувача. Типовою авторизацією в повсякденному комп'ютерному житті може бути, наприклад, надання доступу для читання до певного файлу для аутентифікованого користувача. Приклади типів послуг включають, але не обмежуються ними: фільтрацію IP-адреси, призначення адреси, призначення маршруту, якість обслуговування/диференціальних послуг, контроль пропускну здатності/керування трафіком, обов'язкове тунелювання до певної кінцевої точки та шифрування. Процес аудиту відноситься до відстеження споживання мережевих ресурсів користувачами з метою аналізу потужності та тенденцій, розподілу витрат та виставлення рахунків. Крім того, він може записувати такі події, як помилки аутентифікації та авторизації, а також функції аудиту, перевіряючи, чи були виконані правильні процедури на основі даних обліку. Облік у режимі реального часу відноситься до облікової інформації, яка надається одночасно зі споживанням ресурсів.

Частина аутентифікації AAA відповідає за реалізацію методу ідентифікації користувачів. Це може включати доступ до входу та інші типи доступу, такі як доступ до мережі PPP. За допомогою аутентифікації AAA визначається один або кілька методів аутентифікації, які маршрутизатор повинен використовувати під час аутентифікації користувача.

Коли аутентифікація користувача успішно завершена, авторизація AAA використовується для перешкоджання тому, які дії користувачі можуть виконувати або до яких послуг користувачі можуть отримати доступ. Наприклад, може з'явитися необхідність, щоб адміністратор мережі мав секретний доступ EХЕС, але щоб він використовував лише команду debug. З авторизацією AAA ви можете встановити це обмеження.

Компонент AAA аудит відповідає за збереження списку подій аутентифікації та авторизації. Це може бути так само просто, як відстеження того, хто отримує доступ до маршрутизатора та будь-яких змін статусу на маршрутизаторі, або щось таке ж інтенсивне, як відстеження кожного елемента керування, який користувач виконує на маршрутизаторі. Аудит AAA веде перелік цих подій. Однією з обмежень облікового компонента є те, що для запису фактичних облікових записів йому потрібен зовнішній сервер безпеки AAA.

Протоколи AAA в основному використовуються для контролю доступу до мережі (ресурси LAN, WAN) та адміністрування мережевих пристроїв (брандмауер, комутатори, маршрутизатори). Протоколи AAA були розроблені як централізований спосіб реалізації контролю доступу, що охоплює можливості аутентифікації, авторизації та аудиту. Нижче наведені способи надання можливостей для адміністраторів, користувачів та постачальників.

Для постачальників мережевих послуг: багатофункціональні методи контролю доступу можуть бути поза межами постачальників мережевих послуг або виробників мережевих пристроїв. Підтримка стандартизованих протоколів передає функціональність і безпеку добре визначеним і відомим протоколам.

Для адміністраторів: може бути складно розробити відповідні рішення для контролю доступу в мережі, яка містить пристрої різних виробників. Впровадження контролю доступу на основі стандартних протоколів допомагає інтегрувати найкращі в своєму роді продукти від різних постачальників.

Для користувачів: користувачам може швидко стати недоцільним мати індивідуальний обліковий запис для кожної окремої служби мережі та пристрою в побудованій мережі.

Висновок. У централізованій мережі з реалізацією AAA адміністратори зазвичай впроваджують єдиний вхід (SSO), що дозволяє користувачам отримувати доступ до мережевих ресурсів за допомогою єдиної особистості та облікових даних, тому саме завдяки використанню моделі AAA можна виконати налаштування безпеки розробленої мережі, оскільки протокол AAA дозволяє детально контролювати доступ до комп'ютерних мереж

Список використаних джерел:

1. Буров Є.В. Комп'ютерні мережі. Підручник. Том 2. / Є.В. Буров, М.М. Митник. – Львів: «Магнолія 2006», 2021. – 334 с.
2. Навчальний курс CCNA Routing and Switching: Connecting Networks [Електронний ресурс] – Режим доступу: www.netacad.com.
3. Навчальний курс CCNA Routing and Switching: Introduction to Networks [Електронний ресурс] – Режим доступу: www.netacad.com.
4. Authentication, Authorization, Accounting. [Електронний ресурс]. – Режим доступу: <https://contenthub.netacad.com/legacy/CCNAS/en/3.0.1.1>.

УДК 004.77

АНАЛІЗ МОЖЛИВОСТЕЙ СЕРВЕРНИХ РІШЕНЬ АУТЕНТИФІКАЦІЇ НА БАЗІ ПРОТОКОЛУ RADIUS

Білявський Н.А., магістрант

e-mail: kim221_bna@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Протокол RADIUS особливо цінний для тих, хто працює в організаціях, які мають справу з багатьма різними мережевими та інфраструктурними пристроями або не мають центрального механізму аутентифікації для забезпечення доступу до мережі. Він підключає та аутентифікує ідентифікатори користувачів у мережі або VPN. Ці ідентифікатори можуть зберігатися в Microsoft Active Directory (AD), OpenLDAP, хмарному каталозі або на сервері RADIUS. Це не тільки покращує безпеку мережі, але й заощаджує багато ручної праці IT. Для використання даного протоколу варто виконати аналіз можливостей серверних рішень аутентифікації на базі протоколу RADIUS.

Основні матеріали дослідження. Протокол RADIUS, розроблений компанією Livingston Enterprises, заснований на відкритому стандарті IETF. Протокол RADIUS розроблений для забезпечення доступу до мережі або мобільної IP-адресації. RADIUS працює в локальних мережах та роумінгу і зазвичай використовується для цілей обліку.

Протокол RADIUS приховує паролі під час передачі навіть із протоколом аутентифікації з паролем (PAP). Для цього використовується досить складна операція, що включає хешування дайджесту 5 повідомлень (MD5) і загальний секрет. Проте решта пакета передається у простому текстовому форматі.

RADIUS поєднує аутентифікацію та авторизацію в один процес. При аутентифікації користувача проводиться його авторизація. RADIUS використовує порт UDP 1645 або 1812 для аутентифікації та порт UDP 1646 або 1813 для обліку.

RADIUS широко використовується постачальниками послуг VoIP. Він передає облікові дані кінцевих пристроїв SIP, наприклад широкосмугових телефонів, до реєстру SIP за допомогою аутентифікації дайджесту, а потім на сервер RADIUS за допомогою RADIUS. Протокол аутентифікації RADIUS зазвичай використовується у стандарті безпеки 802.1X.

Далі представлений список доступних серверних рішень протоколу RADIUS, таких як FreeRADIUS, Jumpcloud, TekRADIUS, RouterOS та AuthenticateMyWiFi:

1) Сервіс FreeRADIUS;

Це безкоштовне програмне забезпечення з відкритим вихідним кодом є одним із найпопулярніших серверів RADIUS. FreeRADIUS можна налаштувати на достатньо невимогливій машині для обслуговування від десятка до кількох сотень користувачів. Крім того, його можна встановити на відповідних серверах для підтримки мільйонів користувачів і запитів. FreeRADIUS розроблено для роботи на Unix, Linux та інших Unix-подібних операційних системах.

FreeRADIUS можна знайти у сховищах більшості дистрибутивів Linux, які легко встановлюються або скопійовані вручну на більшості інших. За замовчуванням FreeRADIUS має інтерфейс командного рядка, а зміни налаштувань вносяться за допомогою редагування файлів конфігурації. Він

найкраще підходить для IT-фахівців із досвідом Unix/Linux. Оскільки це відкритий вихідний код, ви навіть можете вносити зміни в код програмного забезпечення.

Ключові відмінності:

- підтримка більшої кількості типів аутентифікації, ніж будь-який інший сервер із відкритим кодом;
- обслуговує організації з кількістю користувачів від десяти до понад мільйона користувачів;
- використовується щодня понад 50 тисячами сайтів;
- повідомляється, що він відповідає за аутентифікацію більше однієї третини користувачів в Інтернеті;
- інші продукти, пов'язані з RADIUS, включають клієнтську бібліотеку, модуль для Apache і модуль аутентифікації (PAM) для аутентифікації та обліку.

2) Сервіс Jumpcloud;

Cloud RADIUS від Jumpcloud — це спосіб розгортання хмарних серверів RADIUS для надання та скасування доступу користувачів до мереж VPN та WiFi із браузера. Він використовує хмарний підхід до впровадження протоколу RADIUS без необхідності створення, підтримки або моніторингу фізичних серверів. Це дає змогу IT швидко розгорнути керований RADIUS в організації та безпечно аутентифікувати користувачів Wi-Fi, VPN, комутаторів та мережевих пристроях.

Ключові відмінності:

- безпечний завдяки багатофакторної аутентифікації та шифрування;
- можна налаштувати на використання EAP-TTLS, PAP або PEAP, а також підтримку режимів шифрування WPA2 Enterprise і RADIUS;
- підтримує доступ до VPN, включаючи Meraki, Palo Alto і OpenVPN;
- сегментування доступу користувачів за допомогою динамічних тегів VLAN і використання VLAN для ізоляції мережевих пристроїв;
- дозволяє користувачам отримувати доступ до мереж з тими ж основними ідентифікаторами, які вони використовують для доступу до інших ресурсів;
- автоматично генерує складні паролі для аутентифікації між WAP і RADIUS-серверами.

3) Сервіс TekRADIUS;

TekRADIUS від KaplanSoft працює на Windows. Основні функції пропонуються безкоштовно, але можна придбати додаткові функції. Версія TekRADIUS Enterprise додає підтримку EAP-TLS, динамічне створення самопідписаних сертифікатів для сеансів PEAP, аутентифікацію NTLM для методів аутентифікації MS-CHAP і відповідність атрибутів на основі регулярних виразів.

Ключові відмінності:

- вбудований DHCP-сервер для призначення IP-адрес дротовим або бездротовим пристроям;
- перевірено на серверах Microsoft Windows Vista, Windows 7-10 і Windows 2008-2019;
- відповідає RFC 2865 і RFC 2866;
- підтримує транспорти TCP і TLS;
- TekRADIUS підтримує Microsoft SQL Server, а TekRADIUS LT підтримує SQLite;
- працює як служба Windows і має інтерфейс керування Windows;

- реєструє системні повідомлення, помилки та інформацію про сеанс у щоденний ротаційний файл журналу та журнал подій Windows;
- створення бази даних та таблиць SQL за допомогою TekRADIUS Manager;
- може проксувати запити RADIUS на інші сервери RADIUS;
- підтримка атрибутів IPv6;
- аутентифікація користувачів у домені Windows або Active Directory.

4) Сервіс RouterOS;

RouterOS — це операційна система, яку MikroTik використовує для своїх продуктів RouterBOARD. Він включає в себе вбудований сервер RADIUS.

Оскільки він пропонує всі основні функції маршрутизатора, його можна навіть використовувати як основний мережевий маршрутизатор. Його можна встановити на ПК, перетворивши його в роутер.

Ключові відмінності:

- такі функції, як маршрутизація, керування пропускнуою здатністю, бездротова точка доступу, зворотний зв'язок;
- Cloud Hosted Router — це підхід, створений для віртуальних машин, доступний у вигляді спеціального інсталяційного образу безкоштовно;
- надає такі функції, як брандмауер NAT, сервер VPN і шлюз точки доступу;
- ОС можна завантажити як образ ISO, який можна записати на компакт-диск для встановлення;
- зміни конфігурації можна внести за допомогою командного рядка, веб-браузера та утиліти Windows WinBox;
- утиліта Windows також пропонується для запису RouterOS на додатковий диск, який був підключений, і диск можна перемістити на виділений ПК або сервер.

5) Сервіс AuthenticateMyWiFi.

AuthenticateMyWiFi від NoWiresSecurity — це розміщена або хмарна послуга, яка пропонує доступ до спеціально розміщеного сервера для аутентифікації.

Це дозволяє малим і середнім організаціям легко використовувати корпоративний режим безпеки WPA або WPA2 для своєї мережі WiFi. Оскільки сервера для налаштування немає, це чудово підходить для організацій без IT-персоналу. Оскільки AuthenticateMyWiFi працює в хмарі, це також полегшує захист мереж WiFi у кількох офісах.

Ключові відмінності:

- використовується режим Enterprise для захисту захищеного доступу Wi-Fi (WPA або WPA2) для приватної мережі Wi-Fi;
- надає доступ до сервера RADIUS, який виконує необхідну аутентифікацію 802.1X;
- аутентифікація на основі пароля за допомогою протоколу PEAP;
- працює для дротових з'єднань, коли використовується з комутаторами бізнес або корпоративного рівня;
- використовує хмарну службу на маршрутизаторах і точках доступу в кількох місцях;
- обмеження доступу користувачів у встановлений час або дні тижня;
- підтримуються Windows, Mac OS X і Linux.

Висновок. За результатами дослідження різних RADIUS-серверів був обраний сервер FreeRADIUS за такими критеріями: обслуговування організацій з кількістю користувачів від десяти до понад мільйона користувачів,

розповсюдженість у більшості дистрибутивів Linux, відкритий вихідний код, безкоштовне використання.

Список використаних джерел:

1. Authentication Server: Setting up FreeRADIUS. [Електронний ресурс] – Режим доступу: <https://tldp.org/HOWTO/8021X-HOWTO/freeradius.html>.
2. Cisco IOS – Privilege Levels. [Електронний ресурс]. – Режим доступу: <https://learningnetwork.cisco.com/s/blogs/a0D3i000002eeWTEAY/cisco-ios-privilege-levels>.

УДК 004.77

МОДЕЛЬ AAA, ЯК ОСНОВА СУЧАСНИХ СИСТЕМ АУТЕНТИФІКАЦІЇ НА БАЗІ ПРИСТРОЇВ КОМПАНІЇ CISCO

Русятинська А.О., магістрантка
Державний університет «Житомирська політехніка»

e-mail: kim221_rao@student.ztu.edu.ua

Актуальність та постановка проблеми. Налаштування безпеки мережі повинне забезпечувати можливість контролю доступу, а також розподілення різних прав в залежності від ролі користувача. Ці параметри складають політику мережевої безпеки. Для забезпечення використовують різні методи, протоколи з метою налаштування мережі. Незважаючи на розмір мережі варто завжди пам'ятати про необхідність налаштування параметрів безпеки для неї.

Для більш великих корпоративних організацій вимагається більш масштабоване рішення аутентифікації. Маршрутизатори та комутатори Cisco IOS можна налаштувати таким чином, щоб вони використовували AAA для аутентифікації в систему Cisco Secure Access Control System. Cisco ACS гарно масштабуються, тому що всі інфраструктурні пристрої мають доступ до центрального сервера. Дана система відрізняється відмовостійкістю, тому що дозволяє налаштовувати декілька серверів.

Основні матеріали дослідження. За замовчуванням пристрій Cisco IOS виконує автентифікацію на основі пароля лінії та авторизацію на основі пароля ввімкнення рівня 15. Це проблема для будь-якої організації, яка бажає деталізації або можливості відстежувати діяльність одного з кількох користувачів. Рішенням цього є AAA, аббревіатура від аутентифікації, авторизації та обліку. Це дозволяє адміністратору налаштувати детальний доступ і можливість аудиту до пристрою IOS.

Служби безпеки мережі AAA забезпечують базову інфраструктуру налаштування контролю доступу на мережевому пристрої. AAA дозволяє контролювати, кому дозволено доступ до мережі (authentication) і що дозволено робити (authorization), а також перевіряти дії при доступі до мережі (accounting).

Система мережевої та адміністративної безпеки AAA в середовищі Cisco має три функціональних компонента:

Аутентифікація – користувачі та адміністратори повинні довести, що вони – саме ті, за кого самі себе видають. Аутентифікація здійснюється за допомогою комбінації імені користувача та пароля, питань та відповідей, токен-карт та інше.

Авторизація – після аутентифікації користувача, служби авторизації визначають, яким користувачам ресурсу можна отримати доступ і які операції вони мають право виконувати.

Облік і аудит – система записує дії користувача, включаючи ресурси, якими від отримувач доступу, тривалість доступу до певного ресурсу та будь-які внесені зміни. Система обліку відстежує спосіб використання мережевих ресурсів.

В системі управління доступом Cisco Secure Access Control System (ACS) підтримуються два основних протокола, які використовуються в пристроях безпеки, маршрутизаторах і комутаторах для реалізації AAA: Tacacs+ та Radius.

Централізована безпека доступу до мережі AAA від Cisco надає багато переваг для організації та мережевого адміністратора, зокрема:

- підвищена гнучкість;
- підвищена безпека;

- масштабованість;
- стандартні методи аутентифікації;
- кілька систем резервного копіювання.

Підвищена гнучкість моделі AAA полягає в підтримці AAA авторизації, на додаток до аутентифікації, означає, що доступ можна підтримувати на основі «необхідності», тобто підтримувати кілька паролів. Підтримка обліку означає, що аудит користувачів і політика розподілу витрат можуть бути реалізовані та бути корисним для усунення проблем мережі.

Підвищена безпека моделі реалізується в тому що декілька пристроїв з однаковим іменем користувача/паролем, що керується локально, забезпечують низький рівень безпеки. Кожен, хто має доступ до всього, незалежно від потреби, також невиправдано збільшує ризик. Здається, що кілька локально адміністрованих паролів підвищують безпеку, але можуть призвести до того, що співробітники будуть записувати паролі, якщо їх занадто багато, щоб їх запам'ятати. Ситуація посилилася б, якщо б були реалізовані складні паролі.

AAA з його централізованою базою даних безпеки та функціями авторизації дозволяє використовувати єдину безпечну комбінацію імені користувача та пароля для кожного співробітника, але при цьому дозволяє обмежувати доступ за принципом «потреба». У той же час AAA дозволяє швидко вирішити зламаний пароль або видалення записів для звільнених співробітників.

Важливим моментом є масштабованість AAA — це шаблонний підхід до управління безпекою, який залишається надійним і гнучким, оскільки мережа стає більшою і складнішою. Завдяки централізації баз даних безпеки та підтримці авторизації AAA уникає керування багатьма комбінаціями імені користувача/пароля у зростаючому середовищі або альтернативної «слабкої» безпеки використання невеликої кількості комбінацій. Локально збережена автентифікація означає, що в будь-який момент, коли є ймовірність скомпрометованого імені користувача/паролем або звільнення співробітника, кожен пристрій має бути переналаштованим. Чим більше пристроїв існує, тим більше зусиль.

Стандартна модель AAA підтримує протоколи безпеки RADIUS, TACACS+ і Kerberos для захисту сеансів підключення. Ці протоколи забезпечують безпечну аутентифікацію, включаючи зашифровані комунікації та взаємодію з системами безпеки мережевого сервера.

З метою забезпечення безперервної роботи AAA підтримує кілька серверів безпеки в одній мережі (TACACS+), для забезпечення резервування в разі відмови пристрою або перевантаження каналів. Крім того, AAA дозволяє вказати кілька методів аутентифікації, тому, якщо перший з них недоступний, можна використовувати другий або третій варіант. Наприклад, якщо вказаний сервер TACACS+ перебуває в автономному режимі, може використовуватися локально збережена база даних імен користувачів/паролів або навіть пароль увімкнення. Ці альтернативи мають бути визначені заздалегідь, інакше доступ може бути заблокований, доки не буде відновлено вказану службу.

Висновок. TACACS+ забезпечує окремі сервіси AAA. Відділення сервісів AAA забезпечує гнучкість реалізації, дозволяючи використовувати протокол TACACS+ для авторизації та урахування при цьому використання іншого методу аутентифікації. Розширення протоколу TACACS+ дають додаткові типи запитів аутентифікації та коди відповідей відповідно до вихідної специфікації TACACS. TACACS+ забезпечує підтримку різноманітних протоколів, включаючи IP та застарілий протокол AppleTalk. В звичайних операціях TACACS+ для

найбільшій безпеці зв'язку шифрується все тіло пакету і використовується порт TCP 49. Саме тому для реалізації моделі AAA, буде використано сервер TACACS+.

Список використаних джерел:

1. Enterprise Networking, Security, and Automation Course. [Електронний ресурс]. – Режим доступу: <https://contenthub.netacad.com/ensa>.
2. Odom Wendell. CCNA 200-301 Official Cert Guide. Volume 2. / Wendell Odom. Cisco Press, 2020 – 1444 p.

УДК 004.77

АНАЛІЗ ФУНКЦІЙ ТА БЕЗПЕКИ ПРОТОКОЛІВ МОДЕЛІ AAA

Русятинська А.О., магістрантка e-mail: kim221_rao@student.ztu.edu.ua
Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Розробка підсистеми аутентифікації користувачів корпоративної мережі на базі моделі AAA та протоколу TACACS+ дозволяє побудувати структуровану, безпечну та адміністровану мереж, яка надасть змогу керувати правами користувачів, виконувати їх розмежування та контроль обліку процесів автентифікації та авторизації. З метою побудови захищеної корпоративної мережі варто виконати аналіз функцій та можливостей безпеки протоколів моделі AAA.

Основні матеріали дослідження. Модель AAA підтримує три протоколи безпеки для контролю комутованого доступу до мереж:

- Kerberos;
- RADIUS;
- TACACS+.

Cisco підтримує Kerberos як застарілий протокол безпеки для тих мереж, які вже підключені до нього. Сервер безпечного контролю доступу Cisco (ACS), реалізує лише TACACS+ і RADIUS.

На найбільш очевидному рівні кожен з цих трьох протоколів виконує те саме. Кожен із них забезпечує безпечний процес аутентифікації, який дозволяє віддаленим користувачам отримати доступ до мережевих ресурсів організації.

Розглянемо кожний з протоколів більш детально:

Kerberos отримав свою назву від триголового собаки, що охороняє ворота Аїда в грецькій міфології. Kerberos (протокол безпеки) — це система аутентифікації, розроблена в Массачусетському технологічному інституті (MIT), яка використовує криптографічний алгоритм стандарту шифрування даних (DES) для шифрування та аутентифікації. Kerberos ґрунтується на припущенні, що внутрішні користувачі за своєю природою не є більш надійними, ніж зовнішні користувачі, і застосовує безпеку та зашифровані комунікації для таких служб, як Telnet, для всіх користувачів.

Kerberos розроблено для того, щоб дозволити двом сторонам обмінюватися особистою інформацією через відкриту мережу, як-от Інтернет, призначаючи унікальний ключ, кожному користувачеві, який входить у мережу. Потім ключ вбудовується в повідомлення, щоб ідентифікувати відправника повідомлення. Kerberos заснований на концепції надійної третьої сторони, яка називається центром розподілу ключів (KDC), сервером аутентифікації Kerberos (AS), що виконує безпечну перевірку всіх користувачів і служб у мережі. Ця перевірка передбачає обмін зашифрованими повідомленнями аутентифікації без передачі пароля користувача.

Протокол Kerberos в своїй роботі для забезпечення доступу користувача до мережі передбачає виконання наступних кроків

- 1) Віддалений користувач відкриває з'єднання PPP з маршрутизатором NAS організації.
- 2) Маршрутизатор запитує у користувача ім'я користувача та пароль.

3) Тоді маршрутизатор використовує лише ім'я користувача для запиту ключа або облікових даних, які підтверджують ідентичність користувача (TGT) від центру розподілу ключів (KDC).

4) Якщо припустити, що ім'я користувача відоме KDC, зашифрований TGT надсилається назад на маршрутизатор, який містить (серед іншого) ідентичність користувача. Термін дії ключа становить вісім годин, щоб зменшити ризик несанкціонованого захоплення та повторного відтворення.

5) Маршрутизатор використовує пароль з кроку 2, для розшифрування TGT. Після цього, якщо розшифровка успішна, віддалений користувач аутентифікується на маршрутизаторі та для нього надається доступ до мережі.

6) Коли віддалений користувач успішно аутентифікується на граничному маршрутизаторі, користувач і машина користувача стають частиною мережі. Інший TGT від KDC необхідний для доступу до мережевих служб, оскільки оригінальний TGT зберігається на маршрутизаторі і не використовується для додаткової аутентифікації, якщо користувач фізично не ввійде на цей маршрутизатор.

7) Захист мережевої служби передбачає подвійне шифрування. Коли користувач запитує доступ до мережевої служби, наприклад, доступ до принтера або Telnet до хоста, KDC створює облікові дані служби. Ці облікові дані служби містять особу клієнта, ідентичність бажаної мережевої служби та обмеження часу. Облікові дані служби спочатку шифруються за допомогою пароля, наданого KDC і службою. Результат знову шифрується з використанням TGT користувача як ключа.

8) Цільова служба використовує TGT, наданий користувачем, для виконання першого дешифрування. Якщо результат можна успішно розшифрувати за допомогою пароля, наданого KDC і службою, значить, служба доступна.

Починаючи з Cisco IOS 11.2, організації, які вже використовують безпеку Kerberos 5, можуть використовувати свої існуючі сервери аутентифікації для автентифікації своїх маршрутизаторів і комутаторів. Можливості аутентифікації Kerberos програмного забезпечення IOS підтримують такі мережеві служби:

- Telnet;
- rlogin;
- rsh;
- rcp.

Вважається, що попередні програми були Kerberized, програми та служби, які були модифіковані для підтримки інфраструктури облікових даних Kerberos та зашифрованого зв'язку.

Служба віддаленої автентифікації Dial-In User Service (RADIUS) — це протокол автентифікації, авторизації та обліку сервера доступу, розроблений Livingston Enterprises, Inc., нині підрозділом Lucent Technologies. RADIUS — це система розподіленої безпеки, яка захищає віддалений доступ до мереж і мережевих служб від несанкціонованого доступу.

RADIUS — це повністю відкритий протокол, розповсюджений у форматі вихідного коду, який можна змінити для роботи з будь-якою системою безпеки, доступною на ринку. Численні реалізації коду сервера RADIUS є комерційно та вільно доступними. Сервери Cisco включають CiscoSecure ACS для Windows, CiscoSecure UNIX і Cisco Access Registrar. Кілька RFC IETF визначають протокол RADIUS, але наразі це все ще є запропонованим стандартом.

Реалізація RADIUS складається з трьох наступних компонентів:

- 1) Протокол з використанням UDP/IP зв'язку.

2) RADIUS-сервер — служба, що працює на центральному сервері Windows або UNIX, як правило, на сайті клієнта.

3) Клієнт RADIUS — служба, що знаходиться на серверах комутованого доступу, яка може бути поширена по всій мережі. Cisco додала підтримку клієнта RADIUS, починаючи з IOS версії 11.1.

Протокол RADIUS у своєму функціонуванні використовує модель сервер клієнт. Маршрутизатор NAS, на якому працює клієнтська служба RADIUS, передає інформацію про користувача визначеному серверу(ам) RADIUS, а потім діє на основі отриманої відповіді. Сервер RADIUS відповідає на всі запити на підключення користувача, аутентифікуючи користувача, а потім пересилаючи всю інформацію про конфігурацію, необхідну клієнту для надання послуги користувачеві. Сервери RADIUS можуть діяти як проксі-клієнти для інших типів серверів аутентифікації.

Після перевірки облікових даних сервер RADIUS відповідає клієнту однією з таких відповідей:

- прийняти – користувач пройшов автентифікацію;
- відхилити – користувач не автентифікований, і доступ заборонено. Користувачеві буде запропоновано повторно ввести ім'я користувача/пароль;
- виклик – запит на отримання додаткової інформації від користувача для підтвердження особи;
- змінити пароль – користувач повинен змінити свій пароль.

Технологія безпеки мережі полягає в тому, що клієнти та сервери RADIUS використовують спільну секретну технологію для всіх транзакцій аутентифікації. Спільний секретний ключ ніколи не надсилається по мережі. Щоб виключити можливість того, що хтось успішно перевірить пароль користувача в незахищеній мережі, будь-які передані паролі користувача надсилаються зашифрованими між клієнтом і RADIUS-сервером.

Одне з переваг є гнучкість використання, оскільки сервер RADIUS підтримує різноманітні методи автентифікації користувача, включаючи PPP PAP або CHAP, MS-CHAP, логін UNIX та інші. Інша сторона будь-якого відкритого стандарту полягає в тому, що він не завжди гарантує взаємодію з реалізаціями інших виробників. Реалізації RADIUS у різних постачальників можуть відрізнятися, оскільки стандарт RADIUS спеціально дозволяє постачальникам пропонувати власні функції або атрибути в так званих парах AV. Якщо пристрій постачальника не розпізнає AV-пару, він ігнорує її. Щоб продемонструвати варіації, реалізація IETF підтримує атрибути, тоді як Ascend підтримує 254. Зараз Cisco підтримує 58 атрибутів у Cisco Secure ACS, серверах доступу, комутаторах Ethernet, брандмауерах PIX та концентраторах VPN 3000.

Контролер доступу до терміналів Access Control System Plus (TACACS+) — це протокол аутентифікації, який дозволяє NAS спілкуватися з сервером аутентифікації, щоб визначити, чи має користувач доступ до мережі.

TACACS+, новий протокол, розроблений Cisco, замінює два попередні галузеві стандарти: TACACS і XTACACS (система контролю доступу до розширеного термінального контролера доступу). TACACS+ несумісний із двома старішими протоколами. Cisco подала специфікацію протоколу TACACS+ у проєкті RFC до IETF для розробки стандарту та для тих клієнтів, які зацікавлені в розробці власного програмного забезпечення TACACS+.

Серверні служби TACACS+ зберігаються в базі даних на демоні TACACS+, що працює на хості Windows 2000/NT або UNIX. Сервери Cisco, які підтримують TACACS+, включають CiscoSecure ACS для Windows, CiscoSecure UNIX і Cisco

Access Registrar. Сервери доступу Cisco (Cisco Secure ACS) можуть реалізувати як TACACS+, так і RADIUS. Основна архітектура протоколу TACACS+ доповнює архітектуру AAA. TACACS+ забезпечує окремі сервіси AAA. Відділення сервісів AAA забезпечує гнучкість реалізації, дозволяючи використовувати протокол TACACS+ для авторизації та урахування при цьому використання іншого методу аутентифікації.

Висновок. Отже, в результаті порівняння вдалося виокремити переваги та недоліки кожного з протоколів. В результаті аналізу можна зробити висновки, що протокол TACACS+ більш надійний, оскільки використовує TCP, окрім цього TACACS+ забезпечує більший контроль команд авторизації та всі пакети AAA зашифровані на відміну від Radius, тому для створення підсистеми аутентифікації користувачів на підприємстві буде використано протокол TACACS+.

Список використаних джерел:

1. TACACS+. [Електронний ресурс]. – Режим доступу: https://www.pro-bono-publico.de/projects/tac_plus.html.
2. Tacacs+ Authentication. [Електронний ресурс]. – Режим доступу: <https://contenthub.netacad.com/legacy/CCNAS/2.0/en/index.html#3.3.2.2>.

ТЕХНІКИ ОЦІНЮВАННЯ ПРОЕКТІВ

Савчук Ю.В., студент

e-mail: uitpm221_syuv@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Точна оцінка проекту є одним із найскладніших процесів в управлінні проектами. Оцінка проекту стає складнішою по мірі зростання складності програмного продукту. Важливість оцінювати роботу точно важко переоцінити, так як від неї залежить бюджет та терміни здачі проекту. Точність оцінки залежить від досвіду спеціалістів які її надають, а також точність оцінки залежить від того наскільки якісно описано об'єм робіт, які потрібно виконати. Оцінка проекту може бути точною.

Основні матеріали дослідження. Розмір, складність і стадія проекту суттєво впливатимуть на необхідний рівень точності, кількість витрат і часу, які компанія може витратити на оцінку проекту, а також на рівень розуміння та ясності обсягу проекту.

Точність оцінки проекту. При оцінці проекту важливо розуміти, на якій стадії знаходиться проект, оскільки це визначатиме рівень необхідної точності оцінки проекту. Наприклад, якщо проект знаходиться на стадії ініціації, оцінка проекту може мати точність $\pm 50\%$. Це означає, що якщо кошторис проекту становить 2 000 000 доларів з точністю оцінки $\pm 50\%$, компанія повинна врахувати 1 000 000 доларів в нижній частині оцінки та 3 000 000 доларів як верхню межу оцінки проекту. Оцінка проекту це процес, і цілком імовірно, що оцінка буде змінюватись по мірі покращення розуміння та уточнення обсягу проекту, при цьому оцінка буде ставати більш точною. Кошторис проекту на етапі планування може зменшитись від $\pm 50\%$ до $\pm 30\%$. Перед виконанням проекту та забезпеченням належного планування точність оцінки може скласти $\pm 10\%$.

Оцінити проект може бути надзвичайно складно, оскільки проект за визначенням є унікальним за своєю природою, часто це новий продукт, послуга або бізнес-зіміна. Проте є декілька доступних методів, які можуть значно покращити процес оцінки проекту та точність, а саме:

Експертна оцінка. Оцінка проекту вимагає деякого рівня експертної оцінки, не тільки самого проекту, а також експертної оцінки бізнес середовища, в якому проект виконується. Зазвичай роль керівника проекту полягає в тому, щоб надати це експертне судження, щоб вирішити, якими бізнес-ризиками потрібно керувати та яка методика оцінки проекту, дозволить досягти найкращий результатів.

Оцінка за аналогами. Оцінка за аналогами є чудовою технікою для оцінки проекту на ранніх етапах. Але вона потребує наявності вже розробленого аналогічного проекту. Проекти повинні мати схожу область використання, а також вони мають бути схожими по складності та об'єму робіт. Наприклад, проект може полягати у розробці мобільного додатку для середнього за популярністю банку, додаток має містити функціональність керування картою, можливість проведення платежів та видачу кредитів. В якості аналога для такого проекту може виступати тільки інший проект для схожого банку, при цьому він містить в півтора рази більше функціоналу. Маючи подібну інформацію можна зробити висновок, що складність є подібною, але масштаб проекту в півтора рази більший. Тоді потрібно оцінку наявного проекту розділити на 1.5 і тоді результатом буде необхідна нам оцінка.

Параметрична оцінка. Параметрична оцінка проекту подібна до техніки оцінки за аналогами, але забезпечує подвійний рівень точності. Ця методика оцінки часто базується на середніх відомих одиницях виміру, таких як кілограми в реальному житті або метрах в будівництві. Наприклад, якщо типова квартира коштує в середньому 1000 доларів за квадратний метр, то квартира площею 50 квадратних метрів буде коштувати 50 000 доларів. Тоді оцінюючи черговий проект можна оцінити не час на розробку, а кількість строк коду який він буде містити, після цього необхідно визначити скільки в середньому коштує одна строка коду. Наприклад, якщо припустити, що проект буде складатись із 50 000 строк коду, а одна строка коду коштує 2 долари, тоді вартість розробки проекту буде становити 100 000 доларів. Якщо потрібно визначити час, тоді потрібно визначити скільки в середньому займає написання однієї строки коду, і на основі цього порахувати необхідний час на розробку всього проекту. Цей метод також потребує наявності попередніх проектів, проте проекти можуть бути різнотипними.

Оцінювання знизу вгору. Оцінка проекту знизу вгору розбиває та оцінює кожен компонент проекту. Кожна окрема задача повинна бути декомпонована так, щоб один компонент можна було оцінити з високою точністю. Потім окремі підзадачі підсумовуються щоб отримати загальну оцінку проекту. Такий спосіб оцінювання займає найбільше часу, і при цьому він є найбільш точним. Зазвичай даний спосіб оцінювання застосовується вже безпосередньо перед процесом розробки.

Оцінювання зверху вниз. Оцінювання зверху вниз відбувається, коли керівникам проекту скласти кошторис проекту без будь-якої детальної інформації. Ця техніка також потребує наявності досвіду розробки подібних проектів. Ця методика полягає в тому, що оцінювання роблять досвідчені співробітники, а іноді і сторонні експерти. Точність оцінки залежить від того, наскільки новий проект відображає попередній досвід спеціалістів, залучений до оцінювання. Оцінювання проводиться великими частинами, без декомпозиції великих задач на дрібніші. В результаті цього точність не може бути високою. Проте очікується, що по мірі розробки проекту, буде проведена декомпозиція великих задач. В результаті чого оцінка груба оцінка буде замінена сумою оцінки підзадач.

Висновок. Проаналізувавши найпопулярніші методи оцінювання проектів, можна зробити висновок, що точність оцінювання проектів може бути різною. Ми можемо обирати необхідну точність в залежності від наступних факторів:

- на якому етапі життєвого циклу проекту ми знаходимось, якщо це ранній етап, тоді точність може бути низькою;
- чи маємо ми результат роботи над попередніми проектами, якщо так, то тоді ми можемо застосувати оцінку за аналогами або параметричну оцінку;

А також для покращення результату оцінки може бути корисним залучення експертів із бізнес середовища.

Список використаної літератури:

1. Project estimation: how accurate is your project estimation. 2022 url: <https://www.simplilearn.com/how-accurate-is-your-project-estimation-rar227-article?tag=estimation> (дата звернення: 28.11.2022)
2. 6 successful project estimation techniques in 2023. 2022 url: <https://www.simplilearn.com/project-estimation-techniques-article?tag=estimation> (дата звернення: 28.11.2022)

УДК 004.43

РОЗРОБКА СУЧАСНОГО ІНТЕРНЕТ-МАГАЗИНУ МЕБЛІВ

Фельдшерев Е.О.

e-mail: raizar804@gmail.com

Науковий керівник: к.т.н., доцент Лубко Д.В.

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Згідно визначення інтернет-магазин (англ. Internet shop, англ. Online shop) - це місце в інтернеті, де відбувається прямий продаж товарів споживачеві (юридичній або фізичній особі), враховуючи доставку. При цьому розміщення споживацької інформації, замовлення товару і угода відбуваються там само, всередині мережі (на сайті інтернет-магазину) [1].

Інтернет-магазин - це не корпоративний сайт, який можна рекомендувати практично всім компаніям. Інтернет-магазин насамперед корисний власникам стандартизованих товарів та послуг. Наприклад, продавати з віртуальних полиць стільникові телефони та автозапчастини значно простіше, ніж, скажімо, одяг чи взуття, які потрібно приміряти. Друга група потенційних власників Інтернет-магазинів – компанії, що займаються традиційною роздрібною комерцією. У цьому випадку онлайн-бізнес тісно взаємопов'язаний із звичайними продажами і виступає як додатковий до основного — роздрібна торгівля. Така схема, мабуть, найбільш вдала. Адже діяльність Інтернет-магазинів регламентує практично та ж законодавча база, як і роботу звичайних (офф-лайн-ових) торгових точок.

Глобальна мережа Internet сьогодні використовується не тільки для обміну інформаційними повідомленнями і для доступу до різноманітних інформаційних ресурсів, все більше застосування знаходять Internet-технології для здійснення конкретних комерційних операцій. У даному контексті використання Internet і сучасних технологій інформаційного обміну дозволить отримати істотні переваги завдяки скороченню витрат, прискоренню різноманітних бізнес-процесів, внаслідок чого зросте інформаційна привабливість, загальна прибутковість діяльності компаній або банків.

Метою даної роботи є створення інтернет-магазину меблів з використанням різноманітних веб-технологій, що наразі є дуже актуальною темою для того щоб у складні часи збільшити продажі товарів та підвищити прибутки даних фірм.

Основні матеріали дослідження.

Проведемо аналіз структури та розробку моделі варіантів використання веб-системи. USE CASE-діаграм мови UML - це важливий і цінний метод аналізу вимог, який широко використовується в сучасній розробці програмного забезпечення [2]. Дана діаграма визначає взаємодію між зовнішніми учасниками і розглянутою системою для досягнення мети. Актор може бути людиною, компанією або організацією, комп'ютерною програмою, системної апаратурою або програмним забезпеченням. Учасник може грати як активну, так і пасивну роль: наприклад, споживач є одночасно покупцем (не взаємодіючим з системою) і користувачем («актором», активно взаємодіє з придбаним продуктом). У свою чергу, користувач є звичайним оператором («актором», що використовують систему за прямим призначенням) і функціональним бенефіціаром (зацікавленою стороною, що користується системою).

Зробимо діаграму варіантів використання системи, щоб дати опис функціональних вимог веб-сайту (рис. 1).

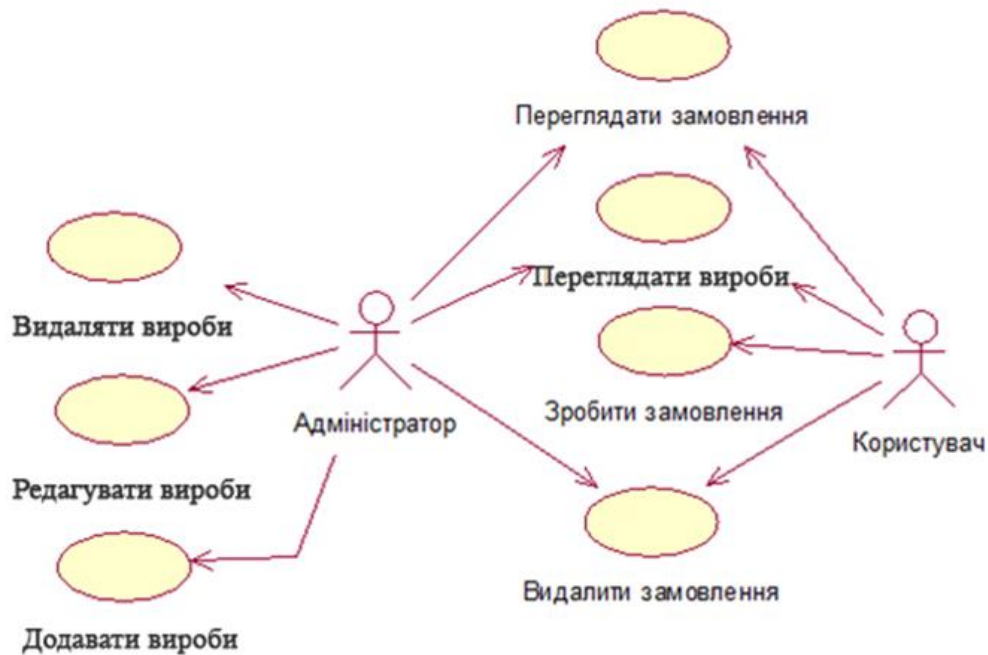


Рисунок 1. Діаграма варіантів використання

Виконаємо розробку моделі даних веб-системи.

Базовим поняттям ER-моделі даних (ER – Entity-Relationship) є сутність, атрибут і зв'язок. Розглянемо основні поняття ER-діаграм [2]:

1. Сутність – це клас однотипних об'єктів, інформація про які повинна бути врахована в моделі. Сутність має назву, виражену іменником в єдиному числі, і позначається прямокутником з найменуванням. Прикладами сутностей можуть бути такі класи об'єктів, як «Студент», «Співробітник», «Товар».

2. Екземпляр сутності – це конкретний представник даної сутності. Наприклад, конкретний представник сутності «Студент» – «Мексимов». Причому сутності повинні мати деякі властивості, унікальні для кожного екземпляра цієї сутності, для того щоб розрізняти екземпляри.

3. Атрибут сутності – це іменована характеристика, яка є деякою властивістю сутності. Найменування атрибута повинно бути виражено іменником в єдиному числі (можливо, з описовими оборотами чи прийменниками). Прикладами атрибутів сутності «Студент» можуть бути такі атрибути, як «Номер залікової книжки», «Прізвище», «Ім'я», «Стать», «Вік», «Середній бал» і т.д. Атрибути зображуються в прямокутнику, який означає сутність.

4. Ключ сутності – це набір атрибутів, значення яких у сукупності є унікальними для кожного екземпляра сутності. При видаленні любого атрибута з ключа порушується його унікальність. Ключів у сутності може бути декілька. На діаграмі ключові атрибути підкреслені.

5. Зв'язок – це відношення однієї сутності до іншої чи до самої себе. Можливо по одній сутності знаходити інші, зв'язані з нею. Наприклад, зв'язки між сутностями можуть виражатись наступними фразами – «СПІВРОБІТНИК може мати декілька дітей», «СПІВРОБІТНИК повинен числитись тільки в одному відділі». Графічно зв'язок зображується лінією, яка з'єднує дві сутності.

Кожний зв'язок має одне чи декілька найменувань. Найменування зазвичай виражається невизначеною формою дієслова: «Продавати», «Бути проданим».

Кожне з найменувань відноситься до свого кінця зв'язку. Іноді найменування не пишуться у зв'язку їх очевидності.

На рисунку 2 наведена я ER-діаграма розробляємої системи.

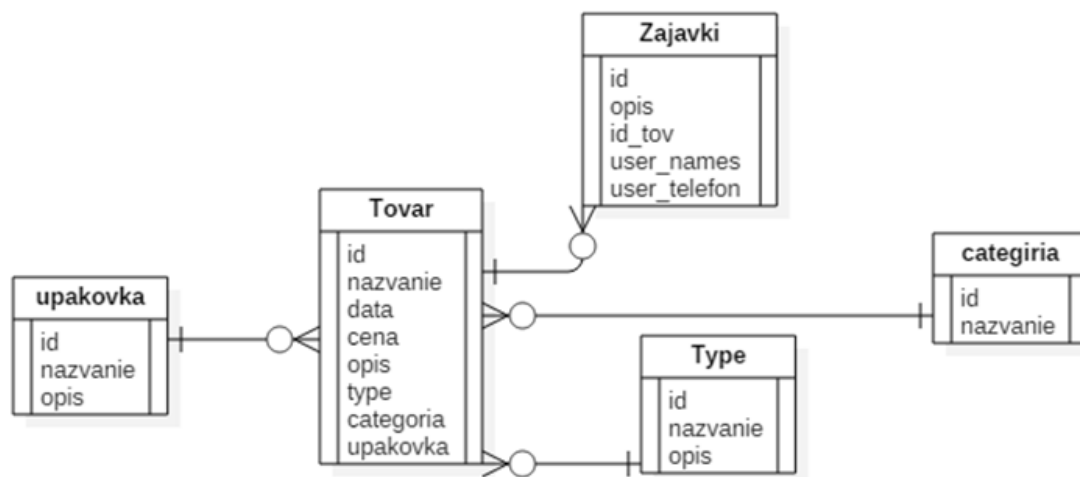


Рисунок 2. ER-діаграма системи, яка розробляється

Виконаємо проектування структури веб-системи [3-5].

Завдяки веб-системі (сайту), люди зможуть робити замовлення послуг, підтримувати зв'язок з продавцями онлайн, отримуючи відповіді на поставлені питання. Важливим є те, що ці дані будуть доступні потенційному клієнту цілодобово.

Розроблений інтернет-магазин (веб-система) містить такі веб-сторінки: головна сторінка; каталог товарів; реєстрація/авторизація; особистий кабінет користувача; адміністративна панель; корзина; про магазин; контакти.

Далі проведемо проектування макетів сторінок системи.

Сторінки веб-системи (сайту) складаються з шапки (хедер), навігації, контенту та футера. У шапці сайту знаходиться логотип сайту, меню з корзиною заявок, авторизацією та реєстрацією, головне меню. Виходячи з попереднього пункту навігація сайту містить такі пункти: головна сторінка, каталог товарів, який містить перелік видів телефонів, про нас та контакти.

На головній сторінці розміщено перелік останніх товарів.

На сторінці каталог товарів буде виведено список категорій телефонів.

Сторінка «Контакти» надає можливість відправити адміністратору повідомлення. Для замовлення будь-якого товару слід буде ввести ім'я, номер телефону, та за бажанням залишити коментар до замовлення.

На вкладці «Вхід» можна пройти авторизацію на сайті, за допомогою кнопки «Зареєструватись» можна буде зареєструватися на сайті.

Для того щоб зареєструватися користувач повинен ввести логін, електронну адресу та пароль.

Далі проведемо програмування серверної частини системи.

Коли користувач дає запит на яку-небудь сторінку (переходить на неї за посиланням, або вводять адресу в адресному рядку свого браузера), то викликана сторінка спочатку обробляється на сервері, тобто виконуються всі скрипти, пов'язані зі сторінкою, і тільки потім повертається до відвідувача у вигляді простого HTML-документа (тобто відвідувач вже ніяк не зможе побачити код вашого скрипта).

Але робота ваших скриптів вже повністю залежна від сервера, на якому розташований ваш сайт, і від того, яка версія тієї чи іншої мови підтримується хостингом.

Серверні мови програмування [6-8] відкривають перед програмістом великі простори в діяльності, проте, скільки б не писали люди, які просувають мову, що їх мова дуже легка. Серед серверних мов програмування виділяються PHP (PHP: Hypertext Preprocessor), Perl, SSI (Server Side Include). Для тестування цих скриптів знадобиться: сервер (www.apache.org), інтерпретатор мови (можна взяти на сайті виробника), ну і СУБД.

Наша веб-система (сайт) написана на мові PHP, тому всі запити, які створюються на сайті обробляються сервером.

Сайт взаємодіє з БД MySQL [9], тому для з'єднання було створено файл `db_params`, у якому прописано скрипт з конфігурацією для підключення до БД:

```
<?php return array(
    'host' => mysql.zzz.com.ua',
    'datb_name' => 'mebel',
    'user' => mebel_admin,
    'pass' => 'Sf4!dK98E',
); ?>
```

Далі проведемо програмування клієнтської частини системи.

Клієнтські мови обробляються на стороні клієнта користувача, а якщо простіше — програми на клієнтському мовою обробляє браузер. Звідси випливає і недолік — це те, що обробка скрипта залежить від браузера користувача, і користувач має повноваження налаштувати браузер так, щоб він взагалі ігнорував написані вами скрипти. При цьому, якщо браузер старий, він може не підтримувати ту чи іншу мову або версію мови, на яку ви спираєтесь. З сучасними браузерами таких проблем виникати не повинно, до того ж мови програмування не так вже часто кардинально оновлюються (раз в декілька років) і кращі з них давно відомі. Також код клієнтського скрипта може подивитися кожен, вибравши в меню «Вид» свого браузера вкладку «Вихідний код» (або щось в цьому роді).

Перевага ж клієнтського мови полягає в тому, що обробка скриптів мовою може виконуватися без відправки документа на сервер. Це легше пояснити на прикладі: припустимо, вам треба перевірити чи правильно користувач ввів e-mail (тобто наприклад, перевірити у нього наявність «@»); щоб це зробити користувачеві, треба було б відправити форму з заповненими даними, потім дочекатися, поки вона буде опрацьовано, і лише після цього отримати повідомлення про помилку (якщо вона, звичайно, присутня). Процес дуже довгий. З клієнтською ж мовою програма відразу перевірить правильне заповнення форми перед відправленням, і, якщо необхідно, виведе помилку. Найпоширенішою з клієнтських мов є JavaScript тому ми її і використовуємо [6-8].

Дизайн сайту було зроблено простим та зрозумілим для звичайних людей. При розробці дизайну враховувались ергономічні вимоги.

Нижче наведені рисунки розробленої веб-системи (рис. 4 та 5).

Інтерфейс сайту було розроблено зручним та інтуїтивно зрозумілим. При розробці дизайну велика увага приділялась кольорам та шрифтам. Таким чином виконано верстку основних сторінок сайту.

Вся клієнтська частина сайту так як і серверна, написана на мові програмування PHP.

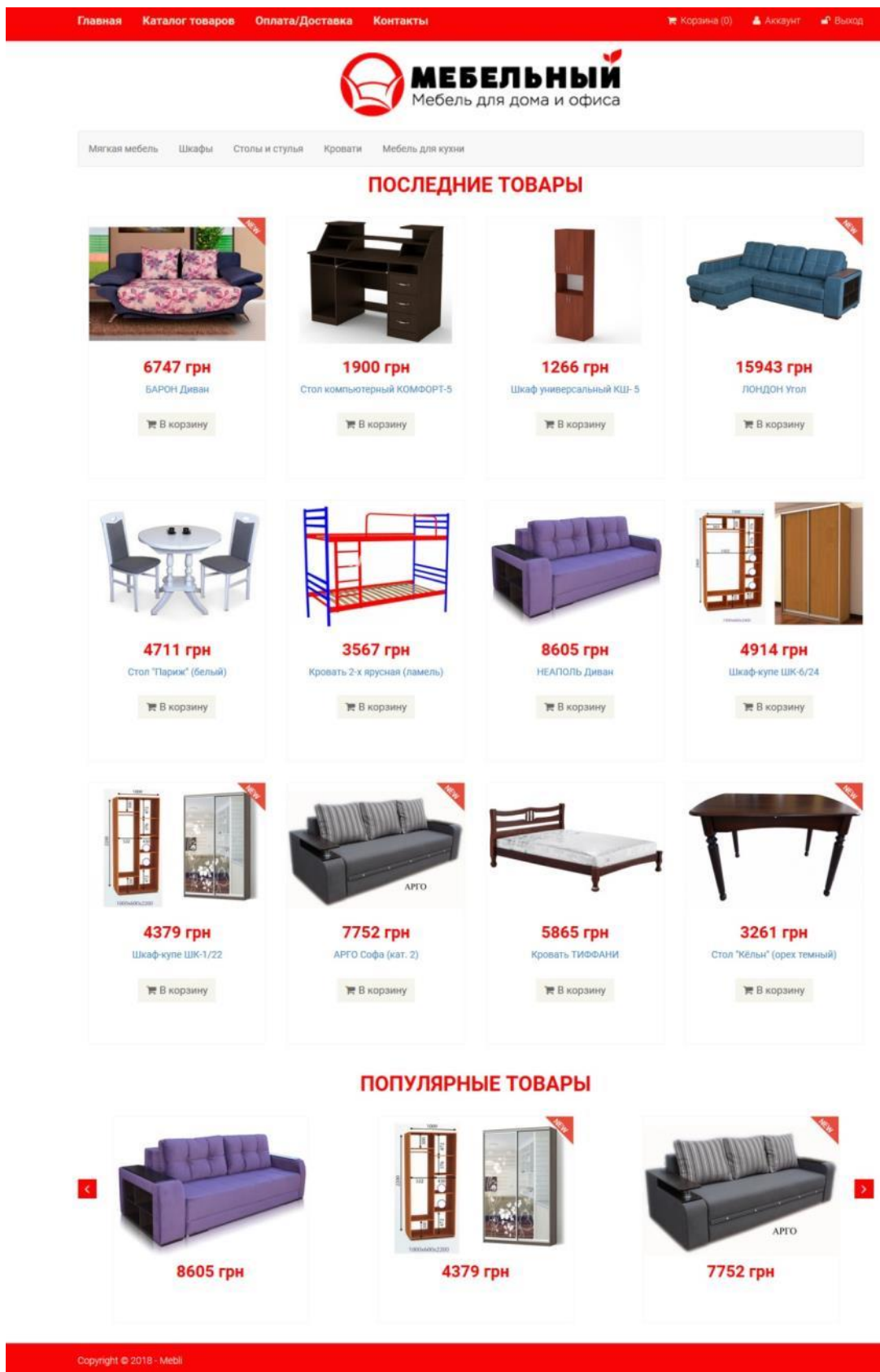


Рисунок 3. Головна сторінка розробленої системи

The screenshot shows a web interface for an administrative panel. At the top, there is a red header with the text 'Админпанель'. Below the header, there are three tabs: 'Админпанель', 'Управление заказами', and 'Просмотр заказа'. The 'Просмотр заказа' tab is selected. The main content area is titled 'Просмотр заказа #46'. Below this title, there is a section 'Информация о заказе' containing a table with the following data:

Номер заказа	46
Имя клиента	Іван
Телефон клиента	0684875465
Комментарий клиента	Доставка новой почтой, отделение №4
ID клиента	4
Статус заказа	Новый заказ
Дата заказа	2022-05-18 18:34:42

Below the table, there is a section 'Товары в заказе' containing a table with the following headers:

ID товара	Артикул товара	Название	Цена	Количество

At the bottom left of the interface, there is a button labeled '← Назад'.

Рисунок 4. Перегляд замовлення в магазині

Далі виконаємо автентифікацію користувачів системи.

Головна перевага парольної аутентифікації – простота і звичність. Паролі давно вбудовані в операційні системи й інші сервіси. При правильному використанні паролі можуть забезпечити прийнятний для багатьох організацій рівень безпеки. Проте, по сукупності характеристик їх варто визнати найслабшим засобом перевірки дійсності.

Щоб пароль запам'ятався, його найчастіше роблять простим (ім'я подруги, назва спортивної команди й т.п.). Однак простий пароль неважко вгадати, особливо якщо знати пристрасті даного користувача.

Іноді паролі із самого початку не зберігаються в таємниці, тому що мають стандартні значення, зазначені в документації, і далеко не завжди після установки системи виробляється їхня зміна. Уведення пароля можна підглянути. Іноді для підглядання використовуються навіть оптичні прилади.

Паролі нерідко повідомляють колегам, щоб ті могли, наприклад, підмінити на якийсь час власника пароля. Теоретично в подібних випадках більш правильно залучити засоби керувань доступом, але на практиці так ніхто не робить: а таємниця, яку знають двоє, це вже не таємниця.

Пароль можна вгадати "методом грубої сили", використовуючи, скажемо, словник. Якщо файл паролів зашифрований, але доступний для читання, його можна скачати до себе на комп'ютер і спробувати підібрати пароль, запрограмувавши повний перебір (передбачається, що алгоритм шифрування відомий).

Проте, важливі заходи дозволяють значно підвищити надійність парольного захисту: накладення технічних обмежень (пароль повинен бути не занадто коротким, він повинен містити букви, цифри, знаки пунктуації й т.п.); керування терміном дії паролів: їхня періодична зміна; обмеження доступу до файлу паролів; обмеження числа невдалих спроб входу в систему, це затруднить

застосування «методу грубої сили»); навчання користувачів; використання програмних генераторів паролів (така програма, ґрунтуючись на нескладних правилах, може породжувати тільки благозвучні й, отже, запам'ятовувані паролі).

Перераховані заходи доцільно застосовувати завжди, навіть якщо поряд з паролями використовуються інші методи автентифікації.

У цьому проекті авторизація користувачів відбувається за допомогою сесій. При використанні сесій дані зберігаються у тимчасових файлах на сервері. Веб-клієнт (звичай веб-браузер) щоразу при спробі відкрити сторінку відповідного сайту пересилає цей фрагмент даних веб-сервера в складі HTTP-запиту.

Для аутентифікації користувачів розроблено такий код:

```
<div class="signup-form">
  <h2>Вход на сайт</h2>
  <form action="#" method="post">
    <input type="email" name="email" placeholder="E-mail" value="<?php echo $email; ?>"/>
    <input type="password" name="password" placeholder="Пароль" value="<?php echo $pass; ?>"/>
    <input type="submit" name="submit" class="btn btn-default" value="Вход" />
  </form>
</div>
```

Розміщення Інтернет-магазину в Інтернеті. Розміщення Інтернет-магазину в Інтернеті планується на платному хостінгу, так як він більш надійний, швидше працює, має багато вільного місця, на відміну від обмеженого безкоштовного.

Висновок.

Результатом виконання роботи є розроблений інтернет-магазин меблів, який складається з декількох сторінок, кожна з яких посилається на розділи сайту. Інформаційна система має два інтерфейси: користувача та адміністратора.

Під час виконання роботи придбані наступні навички: верстки веб-сторінок з використанням HTML і CSS, створення структури бази даних інформаційної системи, програмування інтерфейсу користувача, керування доступом до ресурсу зареєстрованих користувачів, розміщення ресурсу в мережі Інтернет, реєстрації в пошукових системах просування сайту.

Також в процесі виконання роботи були виконані наступні завдання: виконано аналіз і опис предметної області; виконана розробка моделі варіантів використання веб-системи; проведена розробка бази даних системи, а саме виконано: розробка моделі даних веб-системи, нормалізація бази даних веб-системи, визначення типів даних системи, обмеження цілісності даних системи, визначення індексів та тригерів БД системи; виконано проектування структури Інтернет-магазину; виконано проектування макетів сторінок системи; виконано програмування серверної частини; виконано програмування клієнтської частини; виконана автентифікація користувача в системі; описано план розміщення Інтернет-магазину в Інтернеті.

Програмна реалізація веб-системи та її опис проектування дозволить збільшити продаж продукції та підвищити прибутки підприємств, які займаються даним напрямком реалізації бізнесу.

Список використаних джерел:

1. <https://uk.wikipedia.org/wiki/Інтернет-магазин>
2. Martin Fowler. UML Distilled: A Brief Guide to the Standard Object Modeling Language, 3rd Edition. Addison-Wesley Professional. 2016. - p. 208.
3. Стив Круг. Как сделать сайт удобным. Юзабилити по методу Стива Круга. - Киев: Пресс. 2010. - 208 с.
4. Бикнер К. Экономичный Web-дизайн / Кэрри Бикнер. – Киев: Пресс 2005. – 238 с.
5. Купер А., Рейман Р., Кронин Д. Алан Купер об интерфейсе. Основы проектирования взаимодействия. – Пер.с англ. Символ-Плюс, 2009. – 688 с/
6. Роббинс, Дж. HTML5, CSS3 и JavaScript. Исчерпывающее руководство [Электронный ресурс] / Дж. Роббинс ; пер. с англ. М. А. Райтман. - 4-е изд. Электрон. текстовые дан. ЭКСМО , 2014. 528 с.
7. Фрэйи, Б. HTML5 и CSS3. Разработка сайтов для любых браузеров устройств [Электронный ресурс] Responsive Web Design with HTML5 and CSS3 Фрэйи. - Электрон. текстовые дан., 2014. - 304 с.
8. Флэнаган, Д. JavaScript. Подробное руководство [Электронный ресурс Флэнаган. -6-е изд. -Электрон. текстовые дан. Символ-плюс, 2012. -1080 с.
9. Пасічник В. В. Організація баз даних та знань. – К.: Видавнича група BHV, 2006. – 384 с.

УДК 519.832.2

SKEW-SYMMETRIC MATRIX METHOD FOR BALANCING INTRANSITIVE GAMES

Yevhenii Krupchak
Yurii Novikov, senior lecturer
Kharkiv national university of radio electronics

*e-mail: yevhenii.krupchak@nure.ua
e-mail: yuriy.novikov@nure.ua*

Relevance and problem formulation. Intransitive games are games in which there is no single dominant strategy because everything is beaten by literally everything. We see intransitive mechanics in games all the time. In fighting games, a typical pattern is that normal attacks are defeated by blocks, blocks are defeated by throws, and throws are defeated by attacks. In real-time strategy games, a typical pattern is that you have fliers that can destroy infantry, infantry that works well against archers, and archers are great at bringing down fliers. Turn-based strategy games often have some units that work well against others, an example pattern being that heavy tanks lose to anti-tank infantry which loses to normal infantry which lose to heavy tanks. First-person shooters sometimes have an intransitive relationship between different weapons or vehicles, like rocket launchers being good against tanks (since they're slow and easy to hit) which are good against light vehicles (which are destroyed by the tank's fast rate of fire once they get in range) which in turn are good against rocket launchers (since they can dodge and weave around the slow incoming rockets). MMOs and tabletop RPGs often have some character classes that are particularly good at fighting against other classes, as well. So, you can see that intransitive mechanics are in all kinds of places.

Basic research materials. To design an intransitive game, we can create an oriented graph on n vertices whose indegree is equal to the outdegree:

$$d^+(v_i) = d^-(v_i) = \left\lfloor \frac{n-1}{2} \right\rfloor$$

The following method can be used to construct such a graph:

- 1) The main diagonal of the incidence matrix is filled with 0. If n is even, the antidiagonal is filled with 0 too.
- 2) Empty spaces above the main diagonal (and antidiagonal, if n is even) are filled with +1 and -1 in staggered order.
- 3) The remaining cells are filled based on the definition of the skew-symmetric matrix. And in case n is even, we apply the same rule to the antidiagonal.

For the example, take $n = 5$:

$$\begin{bmatrix} 0 & & & & \\ & 0 & & & \\ & & 0 & & \\ & & & 0 & \\ & & & & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & +1 & -1 & +1 & -1 \\ & 0 & +1 & -1 & +1 \\ & & 0 & +1 & -1 \\ & & & 0 & +1 \\ & & & & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & +1 & -1 & +1 & -1 \\ -1 & 0 & +1 & -1 & +1 \\ +1 & -1 & 0 & +1 & -1 \\ -1 & +1 & -1 & 0 & +1 \\ +1 & -1 & +1 & -1 & 0 \end{bmatrix}$$

You can see the resulting graph in Figure 1.

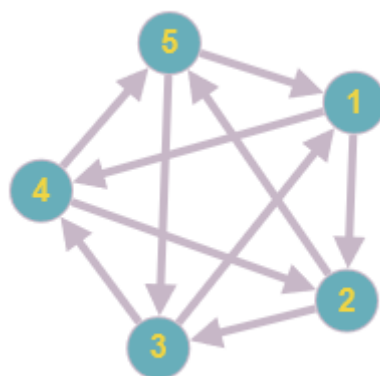


Figure 1. Graph for 5 vertices

For another example, take $n = 6$:

$$\begin{bmatrix} 0 & & & & & \\ & 0 & & & & \\ & & 0 & 0 & & \\ & & 0 & 0 & & \\ & 0 & & 0 & & \\ 0 & & & & & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & +1 & -1 & +1 & -1 & 0 \\ & 0 & +1 & -1 & 0 & \\ & & 0 & 0 & & \\ & & 0 & 0 & & \\ & 0 & & 0 & & 0 \\ 0 & & & & & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & +1 & -1 & +1 & -1 & 0 \\ & 0 & +1 & -1 & 0 & +1 \\ & & 0 & 0 & +1 & -1 \\ & & 0 & 0 & -1 & +1 \\ & 0 & & 0 & -1 & \\ 0 & & & & 0 & \end{bmatrix} \rightarrow \begin{bmatrix} 0 & +1 & -1 & +1 & -1 & 0 \\ -1 & 0 & +1 & -1 & 0 & +1 \\ +1 & -1 & 0 & 0 & +1 & -1 \\ -1 & +1 & 0 & 0 & -1 & +1 \\ +1 & 0 & -1 & +1 & 0 & -1 \\ 0 & -1 & +1 & -1 & +1 & 0 \end{bmatrix}$$

You can see the resulting graph in Figure 2.

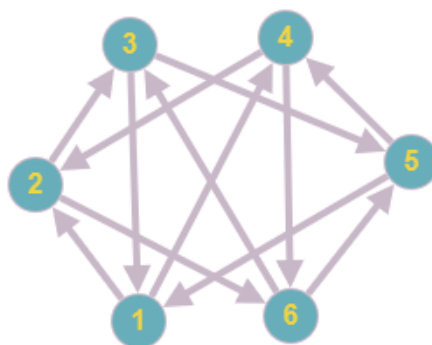


Figure 2. Graph for 6 vertices

But this method has a disadvantage: with large n the graph becomes quite complex, and the number of interactions is so large that it will be difficult for the user to deal with them. Therefore, if the graph is too complex, it should be divided into a

“graph of graphs”. For example, let $n = 15$. Then it would be convenient to divide it into 3 graphs of 5 vertices each, and combine these 3 graphs into 1 “graph of graphs”.

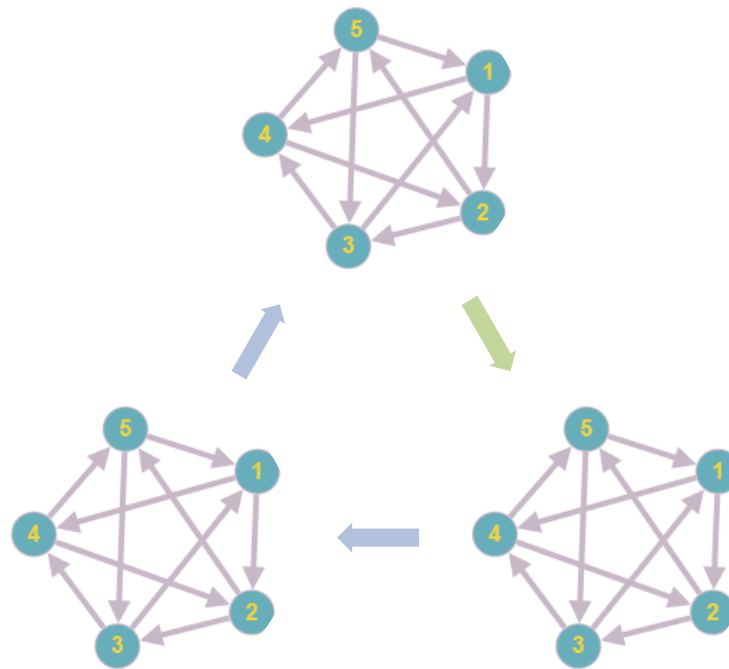


Figure 3. Graph of graphs for 15 vertices

Unfortunately, this method does not work well for large prime n (13, 17, 23), or n , with a large prime divisor (22, 26).

Conclusions. A method has been developed that can be used to balance intransitive games with a small number of types, as well as with a large number of types that can be divided into smaller ones. Theoretically, the method can work for any number of types, but it is also worth remembering that it is necessary to find the optimal balance between complexity and quantity.

References:

1. Skew-symmetric matrix – Wikipedia. URL: https://en.wikipedia.org/wiki/Skew-symmetric_matrix (date of access: 28.11.2022).
2. Schreiber I. Level 9: intransitive mechanics. *Game Balance Concepts*. URL: <https://gamebalanceconcepts.wordpress.com/2010/09/01/level-9-intransitive-mechanics/> (date of access: 28.11.2022).

УДК 004.891.2

АНАЛІЗ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ В УПРАВЛІННІ РЕСТОРАННИМ БІЗНЕСОМ

Верещага Ю.В.

e-mail: codefergal@gmail.com

Науковий керівник к.т.н. Мірошніченко М.Ю.

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. З відкриттям кожного нового ресторану стрімко посилюється і конкуренція між ними, що надалі призводить до необхідності ефективно та раціонально використовувати наявні ресурси.

При таких умовах для успішного ведення бізнесу необхідно інвестувати у засоби та інструменти його підтримки та розвитку. Один із основних інструментів розвитку ресторанного бізнесу - це сучасна система автоматизації ресторанів.

Нині ресторану потрібна автоматизація в той момент, коли власник перестає бути менеджером закладу. С того моменту, як він найняв співробітників і передав їм відповідальність за бізнес-процеси, кожна дія має бути оцифрованою.

Системи автоматизації ресторану допомагають розкласти процеси в ресторані на “ланцюжки” і знизити роль людського фактору: чіткість дій, що повторюються, буде визначати алгоритм, а не людина. Автоматизація підприємства в індустрії харчування має такі основні переваги:

- підвищення оперативності обробки замовлень, точності роботи персоналу. Програмне забезпечення дозволяє стежити за столиками;
- можливості обліку. Менеджер завдяки системі автоматизації ресторану отримує низку ефективних інструментів для аналізу. Він може складати калькуляційні карти, управляти складськими залишками, зводити звіти;
- виключення фінансових махінацій із боку персоналу. Керівник може постійно відстежувати всі дії офіціантів та барменів. Менеджер проводить точкові перевірки. Будь-якої миті він має доступ до кожного замовлення;
- можливості проведення спеціальних акцій. Автоматизація ресторану дозволяє вносити спеціальні пропозиції до меню, давати знижки постійним та новим клієнтам. Програмне забезпечення дає змогу оцінки всіх акцій, їх ефективності. Ви завжди знатимете, чи працюють впроваджені програми лояльності.

Метою роботи є аналіз рішень представників ринку сервісу програмної автоматизації ресторанного бізнесу, їх порівняння та виявлення переваг і недоліків. На основі чого, буде виявлено самий ефективний, зручний та багатофункціональний сервіс.

Основні матеріали дослідження. Автоматизація роботи закладів громадського харчування неможлива без програми планування ресурсів підприємства. Інтегрування такого рішення, приведе до того, що всі облікові процеси у ресторану, кафе, їдальні або бару стануть прозорими, чіткими та зрозумілими, а обслуговування швидким і безпроблемним.

Кращим варіантом такого програмного додатку буде застосування його у вигляді хмарного сервісу, яке буде зручним у використанні та мультидівайсним.

Хмарні сервіси - це онлайн-програми та інші інструменти, які допомагають дистанційно керувати бізнес-процесами. За рахунок «хмар» компанії можуть автоматизувати роботу, заощадити та краще захистити свої дані.

Переваги хмарного сервісу:

- складання та налаштування під завдання клієнта;
- постійні поліпшення та розширення функцій;
- віддалений доступ;
- економічність.

Задачі які вирішують програмні системи автоматизації ресторанного бізнесу:

- приймають замовлення;
- розраховують;
- видають чек про отриману послугу.

Якщо бар також автоматизовано, процес створення напою буде розміщений у програмі.

За допомогою програмного забезпечення можна керувати:

- маркетингом - робити знижку на блюдо чи чек, проводити рекламні акції;
- логістикою - замовляти продукти, відстежувати їхнє місце;
- бухгалтерією - вести весь необхідний облік зручно, коли персонал постійно пробиває чеки та зазначає, коли певна продукція закінчилася.

Хмарний сервіс для автоматизації ресторану, кафе, їдальні або бару на мобільному девайсі дозволяє створити віртуальну карту зали, приймати та контролювати замовлення (столи, за якими є відкриті замовлення), вести касу, друкувати чеки, вести складський облік, базу клієнтів та список співробітників, контроль надходженням та списанням товару, розрахунки з клієнтами та постачальниками, фіксація обороту коштів.

Серед таких додатків, які застосовують технологію хмарного сервісу для автоматизації ресторанного бізнесу є: SkyService, Poster, Presto.

SkyService POS – це мобільна та гнучка система обліку продажів для foodtruck. Вона дозволяє працювати як онлайн, так і без доступу до мережі Інтернет. Всі продажі будуть синхронізовані з сервером в режимі онлайн при підключенні до інтернету, проте доти програма буде здатна працювати, і ваші продажі не пропадуть. Інтерфейс системи SkyService представлено на рис. 1. Система дозволяє вести складський облік, контролювати співробітників, аналізувати статистику продажу. Всі дані зберігаються у хмарі та доступні з будь-якого пристрою по всьому світу.

Серед переваг сервісу SkyService POS можна виділити:

- зручний інтерфейс та простота використання;
- на мобільному пристрої відображається цілодобово статистика та аналітика доставки;
- доступна ціна пристрою;
- надійність обладнання;
- програмне забезпечення має можливість суміщення з усіма сучасними пристроями.

Серед недоліків вказаної системи можна виділити

- відсутність складського обліку та техкарти;
- неможливість підтримки терміналу та аналітики закладу;
- відсутність можливості управління меню та заказами.

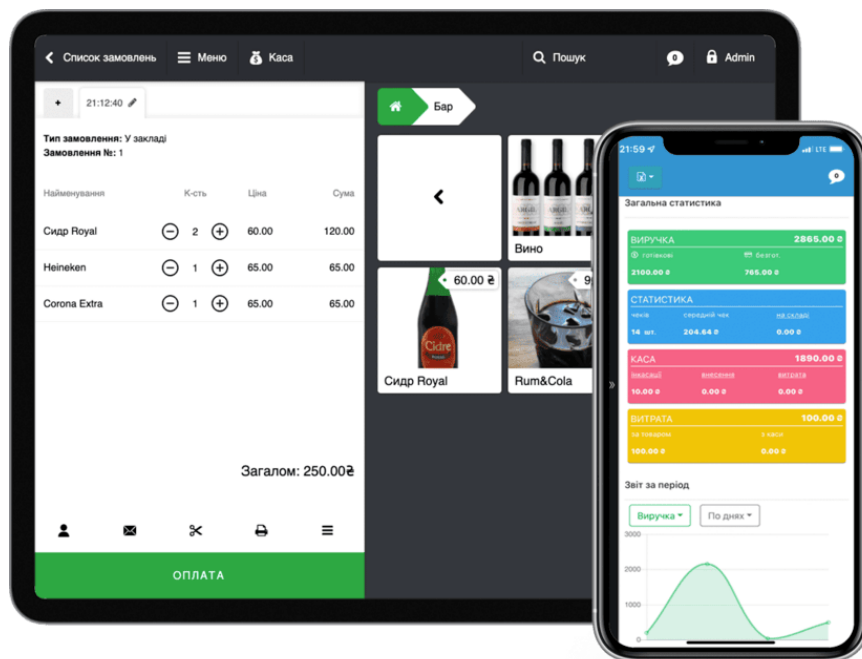


Рисунок 1. Інтерфейс системи SkyService

Система **Poster POS** – це інформаційна система, яка також автоматизує роботу кафе, бару, ресторану чи магазину. Система автоматизації складається з фронт-офісу — додатку для працівників залу (рис. 2) та бек-офісу — адміністративної панелі для менеджерів та власників. Також для власників є спеціальний додаток - Poster Boss, де в реальному часі відображаються основні звіти: виручка, прибуток, кількість чеків, середній час обслуговування тощо.

Перевагами сервісу Poster POS є:

- Проста установка та навчання;
- Робота без інтернету. Якщо у закладі зник інтернет, робота не зупиниться: буде здатність приймати замовлення, друкувати чеки. Дані автоматично потраплять у хмару, коли з'єднання мережі відновиться;
- Доступна ціна. Сервіс буде доступний при оформленні щомісячної передплати;
- Програмне забезпечення має можливість суміщення з усіма сучасними пристроями;
- Багатофункціональність: онлайн-каса, склад, фінанси, аналітика, CRM (Система управління взаємовідносинами із клієнтами);
- Активна технічна підтримка.

До недоліків цієї системи можна віднести технічні складнощі з синхронізацією обліку та відсутність управління правами користувача, управління білінгом і рахунками.

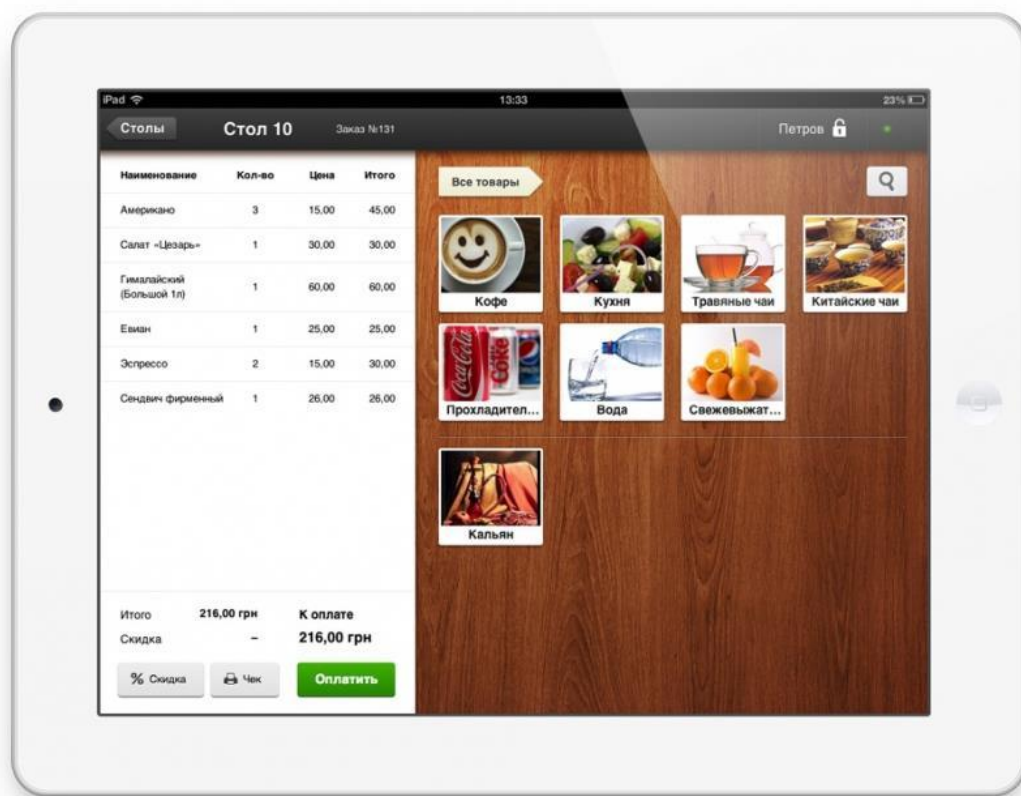


Рисунок 2. Интерфейс системы Poster



Рисунок 3. Интерфейс системы POS Sector

POS Sector – це програма, яка в Україні з'явилася відносно недавно, але вже встигла здобути довіру багатьох українських власників закладів громадського харчування. Система дає можливість створювати оперативне обслуговування клієнтів у своєму закладі. Вдається навести повний порядок обліку. Власник ресторану зможе отримати додатковий прибуток, ретельний аналіз результатів роботи. POS Sector на відміну від аналогічних програм для автоматизації кафе, барів або ресторанів, встановлюється на будь-який комп'ютер або планшет з операційною системою Windows.

Серед переваг POS Sector можна виділити: доступну ціну (сервіс буде доступний при оформленні щомісячної передплати) та програмне забезпечення, яке має можливість суміщення з усіма сучасними пристроями. Також є можливість замовити індивідуальну конфігурацію програми у розробника.

Із проблем сервісу можна виділити: невелику кількість можливостей у зрівнянні з аналогами (немає виставлення рахунків, менеджмент кухонь, управління бронюванням, управління списком очікування і менеджменту таблиць) та відсутність управління правами користувача та обліку робочого часу персоналу.

Таблиця 1 – Порівняння сервісів за певними можливостями

Назва	Skyservice POS 	Poster POS 	POS Sector 
Аналітика та звіти	+ – Тільки аналітика доставки	+	+
Складський облік	–	+	–
CRM	–	+	–
Програма лояльності	+	–	+
Конструктор страв	–	+	+
Відстеження залишків та запасів	–	+	+

На основі отриманих результатів дослідження та проблематики системи пропонується розібратися який сервіс є кращим для автоматизації ресторанного бізнесу.

Просте програмне забезпечення значно збільшує продуктивність праці та якість, з якою працівники обслуговують відвідувачів. А також робить доступним контроль за всім, що відбувається на території закладу.

Заявлені сервіси мають такий технічний та програмний комплекс, складаючи із трьох рівнів.

1. Касовий рівень:

- система касира;
- система офіціанта;

- система бармена;
- 2. Фронт офіс:
 - система формування даних
 - система звітів
 - система складського обліку
- 3. Бек офіс:
 - система формування даних
 - система звітів.

Проаналізувавши та зрівнявши між собою системи можна дійти до такого висновку, що найкращим варіантом, за своєю функціональністю, зручністю та візуалізації буде хмарний сервіс - Poster. Незважаючи на нечисленні та несуттєві недоліки, Poster буде найкращою реалізацією хмарного сервісу для автоматизації ресторанного бізнесу в Україні.

Висновки. Автоматизація ресторану дуже важлива щодо покращення якості управління та здійснення суворого контролю усіх сфер роботи закладу. Насамперед - це безпека фінансових операцій та точне визначення суми виручки за певний час. Підприємці, яким не байдужі справжні суми доходів від їхньої справи, давно перейшли на систему програмно-апаратного комплексу. Автоматизація ресторану багато в чому визначає успішне ведення справ і до певної міри виступає гарантом благополуччя та процвітання. У галузі дослідження та моніторингу ресторанного бізнесу зазначено, що успіх підприємства багато в чому визначається оснащеністю та кваліфікацією персоналу. Автоматизація ресторану необхідна для згладжування якихось недоліків у процесі роботи та усунення будь-яких можливих неполадок, пов'язаних із роботою персоналу.

Список використаних джерел:

1. Александрова В.О., Якименко-Терещенко Н.В., Строков Є.М. Дослідження та оптимізація економічних процесів в готельно-ресторанному бізнесі: колективна монографія. Харків: НТУ «ХПІ», 2019. 100 с.
2. Мальська М. П., Пандяк І. Г. Готельний бізнес: теорія та практика навчальний посібник. К.: Центр учбової літератури, 2009. 472 с.
3. Інформаційні технології в управлінні готельно-ресторанним бізнесом. [Електронний ресурс]. URL: https://tourlib.net/statti_ukr/borzenko.htm (дата звернення: 08.10.2022)
4. Михайлицька М., Фактор Ч. Усе про облік та організацію ресторанного господарства, 2012. 432 с.
5. Гужва В.М. Інформаційні системи і технології на підприємствах. К., 2001. 400 с.
6. Пророчук Ж.А. Роль програмного забезпечення в управлінні підприємством. *Современные информационные технологии*. 2010. № 1. С. 18 – 20.
7. Лутай А. П. Інформаційні технології і системи управління в готельно-ресторанному бізнесі України. *Торгівля і ринок України*. Донецьк: ДонНУЕТ, 2011. Вип. 31. Т. 2. С. 80–90.
8. Єсіпова К. А. Єдність і відмінність понять електронний бізнес та електронна комерція. [Електронний ресурс]. Режим доступу: <http://intkonf.org/esipova-k-a-ednist-i-vidminnosti-ponyat-elektronniybiznes-ta-elektronna-komertsiya> (дата звернення 14.11.2022)

УДК 004.891

ІНТЕЛЕКТУАЛЬНА ІНФОРМАЦІЙНА СИСТЕМА «ПІДБОРУ МОНОБЛОКУ»

Супрун М.В.

e-mail: mrsolomka2447@gmail.com

Холодняк Ю.В., к.т.н., доц.

e-mail: yuliya.kholodnyak@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. В останні роки в усьому світі спостерігається все зростаючий інтерес до систем, які в автоматизованому виді видають товар клієнту за набором ознак та характеристик. Власноруч проводити обробку такої кількості інформації неможливо. Тому виникає необхідність в створенні комп'ютерних засобів автоматизації обробки, пошуку, збереження, систематизації даних, при цьому здатних аналізувати дані, пристосовуватися до змін інформації, робити власні висновки на основі аналізу.

Метою статті є розробка інтелектуальної інформаційної системи, яка дасть можливість враховувати побажання користувача по вибору моноблоку.

Основні матеріали дослідження. Інтелектуальна система (ІС) – це технічна або програмна система, яка забезпечує розв'язування неформалізованих задач користувача в деякій предметній галузі та організовує його взаємодію з комп'ютером у звичних поняттях, термінах, образах, що вважалися творчими, які традиційно належать конкретним предметним областям, інформація зберігаються в пам'яті інтелектуальної системи. Структура інтелектуальної системи включає три основні блоки - базу знань, механізм виведення рішень і інтелектуальний інтерфейс. Властивість ІС виконувати творчі завдання називається штучним інтелектом.

Комунікаційні здібності ІС характеризують спосіб взаємодії кінцевого користувача з системою – тобто можливість формулювання довільного запиту в діалозі системи. Складні, погано формалізовані завдання – завдання, що погано формалізуються та вимагають побудови оригінального, нестандартного алгоритму рішення, залежно від конкретної ситуації, для якої можуть бути характерні невизначеність і динамічність.

Оскільки в основі інтелектуальних систем, що створюються як додаток теорії несилової взаємодії, лежить обробка різних дій з виробленням адекватною цим діям реакції, то такі системи отримали назву рефлексорних (по аналогії з поведінкою об'єктів живої природи). За критерієм ступеня інтелектуалізації, який у першому наближенні може характеризуватися обсягом інформації, що обробляється, інтелектуальні системи можна поділити на:

- системи перебору варіантів рішень згідно встановленої пріоритетності для наперед змодельованих ситуацій;
- ІС, які приймають рішення за детермінованими вирішальними правилами без навчання;
- інтелектуальні системи, що реалізують алгоритми компараторного розпізнавання за еталонами;
- експертні системи, що з метою прийняття ефективних рішень маніпулюють спеціальними знаннями
- накопиченими фахівцями експертами у конкретній предметній сфері знань;

- інтелектуальні системи, що здатні самонавчатися;
- знання-орієнтовані ІС, що здатні утворювати та використовувати базу знань.

Інтелектуальні системи, що здатні самонавчатися, можна поділити на такі основні класи:

1. ІС, що розв'язують задачу розпізнавання образів за апіорно класифікованою навчальною матрицею (навчання з “учителем”).
2. ІС, що реалізують алгоритми факторного кластер-аналізу.
3. ІС, що реалізують алгоритми кластер-аналізу при незмінному словнику ознак і за апіорно некласифікованими навчальними матрицями, тобто за умови неповної апіорної інформації про функціональний стан системи (навчання без “учителя”).
4. ІС, які реалізують алгоритми автоматичної класифікації за апіорно некласифікованими навчальними матрицями та здатні оптимізувати параметри словника ознак розпізнавання.
5. Відмово стійкі ІС, що здатні самостійно діагностувати свій функціональний стан і відновлювати свою функціональну спроможність при виникненні відмов.
6. Адаптивні ІС, що здійснюють класифікаційне самонастроювання та самоорганізацію.
7. ІС, що вирішують проблему шалювання, яка полягає у побудові для шкал з різними мірами виміру зведеної шкали, координати якої можуть бути обернено відображені на відповідні вихідні шкали.
8. Сенсорні ІС, що моделюють чуттєві функції людини та базуються на “образному” комп'ютері, наділеному властивостями “технічного зору”, усномовного розпізнавання, розпізнавання пахощів тощо.
9. Гібридні ІС, які поєднують різні алгоритми та методи автоматичної класифікації.

До знання-орієнтованих інтелектуальних систем відносять:

1. Системи, що ґрунтуються на інструктивних знаннях (rulebased reasoning).
2. Системи, що ґрунтуються на автоматичному доведенні теорем (automatic theorem-proving techniques).
3. Системи, що ґрунтуються на автоматичному породженні гіпотез (automatic hypothesizing).
4. Системи, що ґрунтуються на доведенні за аналогією (analogical reasoning).
5. Об'єктно-орієнтовані інтелектуальні системи (object-oriented intelligent systems).
6. Об'єктно-логічні інтелектуальні системи, що поєднують окремі переваги об'єктно-орієнтованих систем з системами автоматичного доведення теорем і використовують об'єктно-логічні мови, фреймові логіки (F-logics), логіки транзакції (transaction logics) та інше. Зрозуміло, що наведена класифікація не є досить повною, оскільки відбувається неперервне розширення номенклатури ІС як за призначенням, так і за принципами функціонування.

Одна з головних переваг інтелектуальної інформаційної системи, у тому що вона не потребує великих потужностей для її запуску та користування, також займає не велику кількість місця та може бути портативною, також може бути відкритою з будь-якого ПК, ноутбуку або моноблоку.

Дана інтелектуальна система має наступні переваги:

- простий інтерфейс;
- малі мінімальні характеристики для запуску.

- великий товарний асортимент;
- велика кількість параметрів для підбору;

Користувач має можливість пошуку товару за назвою через поле пошуку, або через параметри моделей, такі як:

- ціна;
- об'єм вбудованої пам'яті;
- об'єм оперативної пам'яті;
- розширення дисплею;
- наявність дискретної відеокарти;
- наявність веб камери.

PyQt — це бібліотека Python для створення програм із графічним інтерфейсом за допомогою інструментарію Qt. PyQt є вільним програмним забезпеченням (за ліцензією GPL) і розробляється з 1999 року. Остання версія PyQt6 – на основі Qt 6 – випущена у 2021 році, і бібліотека продовжує оновлюватись. На основі знань бібліотеки PyQt також можна використовувати PySide2, PySide6, PyQt6 і PyQt5.

Сьогодні використовуються дві основні версії: PyQt5 на основі Qt5 та PyQt6 на основі Qt6. Обидві майже повністю сумісні, за винятком імпорту та відсутності підтримки деяких сучасних модулів з Qt6. У PyQt6 вносяться зміни у роботу просторів імен та прапорів, але ними легко управляти.

Після підключення потрібних бібліотек в нашому випадку будемо використовувати PySide2.

Створюється вікно, потім додаються елементи з назвами блоків та чекбокси, розташовуються на площині вікна щоб не накладались один на одного та щоб не були в одній кучі, так само додаються і елементи з виводом інформації результатів.

Розширення дисплею

☐ 1920x1080

☐ 2560x1080

Об'єм оперативної пам'яті

☐ 8 GB

☐ 16 GB

☐ 32+ GB

Об'єм накопичувача (SSD)

☐ 120 - 256 GB

☐ 480 - 512 GB

☐ 1 TB

Ціна у гривнях

Від

До

☐ Наявність відеокарти

☐ Наявність веб камери

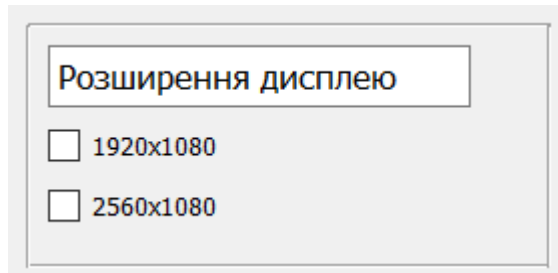
Виконати рекомендацію

Моделі моноблоків які підходять вам

Рисунок 1. Тіло інтерфейсу

Як ми бачило на рис. 1 у вигляді блоків оформлено вибір чекбоксів з характеристиками моноблоків, також у середині блока підписано характеристики та що саме у ньому знаходяться, по інший бік вікна розташований вивід інформації.

На рис. 2 ми бачимо блок програми з розширенням дисплею та наявністю у середині нього чекбоксу з вибором розширення дисплею, також поле яке вказує назву блоку та лінії які візуально відділяють блок від інших елементів.



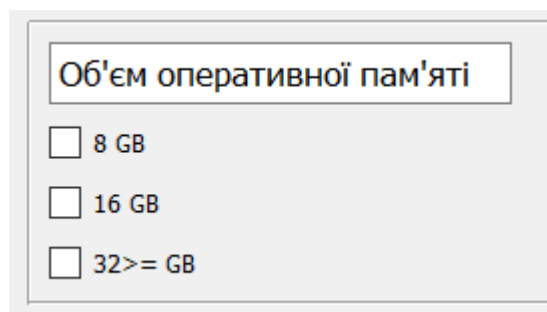
Розширення дисплею

☐ 1920x1080

☐ 2560x1080

Рисунок 2. Блок “Розширення дисплею”

На рисунку 3 показано блок з об'ємом оперативної пам'яті, у ньому містяться чекбокси з вибором кількості оперативної пам'яті, а саме 8, 16, 32 та більше гігабайти оперативної пам'яті, також розміщене поле з назвою блоку та лінії які виділяють поле тим самим візуально виділяють поле та показують межі блоку.



Об'єм оперативної пам'яті

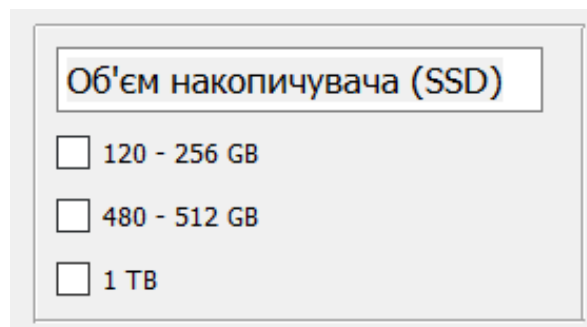
☐ 8 GB

☐ 16 GB

☐ 32 >= GB

Рисунок 3. Блок “Об'єм оперативної пам'яті”

На рисунку 4 показано блок “Об'єм накопичувача (SSD)”, у ньому містяться три чекбокси з вибору об'єму накопичувача який буде встановлений в моноблок, 120-256 Гб, 480-512 Гб, 1Тб, також у блоці є поле яке вказує назву блоку та що в нбому містиця, лінії які візуально виділяють блок від інших елементів.



Об'єм накопичувача (SSD)

☐ 120 - 256 GB

☐ 480 - 512 GB

☐ 1 TB

Рисунок 4. Блок “Об'єм накопичувача (SSD)”

На рисунку 5 показано блок “Ціна у гривнях”, у ньому міститься, два поля для вводу інформації у вигляді максимальної ціни та мінімальної ціни на яку розраховує користувач, розміщені слова “Від”, “До”, відповідно до поля у яке повинно вноситися число, також назва блоку та лінії які візуально розділяють блок від інших елементів.

Рисунок 5. Блок “Ціна у гривнях”

На рисунку 6 наведено елементи які не входять до блоку, це чекбокси які відповідають за характеристики наявності відеокарти та вебкамери.

Рисунок 6. Елементи поза блоками

На рисунку 7 наведено блок з полем виведення інформації та полем яке виводить інформацію про поле яке знаходиться під ним.

Рисунок 7. Елементи поза блоками

Для розробки клієнтської частини була задіяна мова програмування Python при використанні вбудованих бібліотек та встановленої бібліотеки PySide2 для створення графічних інтерфейсів.

Для отримання інформації від користувача було прийнято рішення використовувати чекбокси та рядок введення для визначення цінового діапазону.

Приклад коду з додавання елементу чекбокс:

```
self.checkBox = QtWidgets.QCheckBox(self.centralwidget)
self.checkBox.setGeometry(QtCore.QRect(310, 180, 151, 20))
self.checkBox.setObjectName("checkBox")
self.checkBox.setText("Наявність відеокарти")
self.checkBox.stateChanged.connect(self.clickBox)
```

По цьому прикладу ми додали елемент, потім задали місце розташування та розміри елемента у пікселях, задали назву, останній рядок перевіряє статус чекбокса.

Приклад функції перевірки статусу чекбоксу та зберігання його:

```
def clickBox(self, x):
    if x == QtCore.Qt.Checked:
        self.x_1 = 1
    else:
        self.x_1 = 0
```

Функція “clickBox” перевіряє стан чекбоксу та якщо чекбокс ввімкнутий то значення змінної “x_1” присвоюється одиниця, у іншому випадку нуль.

Опис кожного блока та блока з виводом результату рекомендації для користувача.

Приклад додавання тексту з описом блоків:

```
self.textBrowser = QtWidgets.QTextBrowser(self.centralwidget)
self.textBrowser.setGeometry(QtCore.QRect(30, 40, 211, 31))
self.textBrowser.setObjectName("textBrowser")
self.textBrowser.setHtml("<!DOCTYPE HTML PUBLIC \"/>
\"http://www.w3.org/TR/REC-html40/strict.dtd\">
<html>
<head>
<meta name=
\"qrichtext\" content=
\"1\" />
<style type=
\"text/css\">
<p, li { white-space: pre-wrap; }
</style>
</head>
<body style=
\" font-family:
'MS Shell Dlg 2'; font-size:
7.8pt; font-weight:
400;font-style:
normal;\">
<p style=
\" margin-top:
0px; margin-bottom:
0px; margin-left:
0px; margin-right:
0px; -qt-block-indent:
0; text-indent:
0px;\">
<span style=
\" font-size:
10pt;\">
Розширення дисплею
</span>
</p>
</body>
</html>
\"")
```

Тільки для цього елемента застосовується Html, ця мова розмітки дає змогу втілити майже все на що здатна Html та Css, нам вона дала можливість додати опис до блоку саме таким методом.

Приклад додавання звичайного тексту та розміщення його на вікні:

```
self.labelFrom = QtWidgets.QLabel(self.centralwidget)
self.labelFrom.setGeometry(QtCore.QRect(450, 90, 80, 20))
self.labelFrom.setText("Від")
```

У тій самій функції у якій знаходяться наші елементи які ми додали до інтерфейсу знаходиться й додавання слів “Від” та “До” у блоку з вибором ціни, у прикладі ми бачимо додавання елемента “лейбл” та задання йому розташування та розміри, потім вказуємо те що ми хочемо бачити на місці розташування елемента, аналогічним чином створюється й другий “лейбл” з словом “До”.

Приклад створення кнопки та розташування її у інтерфейсі:

```
self.pushButton = QtWidgets.QPushButton(self.centralwidget)
self.pushButton.setGeometry(QtCore.QRect(340, 420, 190, 40))
self.pushButton.setObjectName("pushButton")
self.pushButton.clicked.connect(self.magic)
self.pushButton.setText("Виконати рекомендацію")
```

На прикладі ми можемо бачити дуже схожі команди на ті що приведені у прикладах вище, тут також задіяна віддача у вигляді ініціації виклику функції “magic”, вона починає обробку узятих даних та у кінцевому результаті виводить на поле справа рекомендовані моноблоки.

Системне тестування тестує інтегровану систему для перевірки відповідності всім вимогам. Перевірка повноти та правильності документації користувача є важливою частиною системного тестування. Всі тестові комбінації

повинні розроблятися тільки з використанням документації користувача. Однією з головних цілей тестування є перевірка відповідності працездатності інформаційної системи в цілому або її окремих модулів очікуванням користувача.

Тестування включає в себе наступні етапи:

- функціонального тестування – виявлення невідповідностей між реальною поведінкою реалізованих функцій і очікуваною поведінкою відповідно до специфікації і вимог.

Залежно від мети, функціональне тестування може проводитися:

- на основі функціональних вимог, зазначених у специфікації вимог. При цьому для тестування створюються тестові випадки (test cases), створення яких враховує пріоритетність функцій ПЗ, які необхідно покрити тестами. Таким чином ми можемо переконатися в тому, що всі функції продукту, що розробляється, працюють коректно при різних типах вхідних даних, їх комбінацій, кількості і т.д.

- на основі бізнес-процесів, які має забезпечити додаток. У цьому випадку, нас цікавить не так працездатність окремих функцій ПЗ, як коректність виконуваних операцій, з точки зору сценаріїв використання системи. Таким чином, тестування в даному випадку буде ґрунтуватися на варіантах використання системи (use cases).

- тестування продуктивності, це комплекс типів тестування, метою якого є визначення працездатності, стабільності, споживання ресурсів та інших атрибутів якості додатка в умовах різних сценаріїв використання та навантажень. Тестування продуктивності дозволяє знаходити можливі вразливості та недоліки в системі з метою запобігання їх негативному впливу на роботу програми в умовах використання. Необхідні параметри роботи системи в певному середовищі можна тестувати за допомогою:

- визначення робочої кількості користувачів додатку.
- вимірювання часу виконання різних операцій системи.
- визначення продуктивності додатка при різних рівнях навантаження.
- визначення допустимих меж продуктивності програми при різних рівнях навантаження.

Тестування програмного забезпечення — процес перевірки відповідності заявлених до продукту вимог і реально реалізованої функціональності, який здійснюють шляхом спостереження за його роботою в штучно створених ситуаціях і на обмеженому наборі тестів, обраних певним чином.

Можуть оцінювати:

- відповідність вимогам, якими керувалися проектувальники та розробники,
- правильність відповіді для всіх можливих вхідних даних,
- виконання функцій за прийнятний час,
- практичність,
- сумісність із програмним забезпеченням та операційними системами,
- відповідність задачам замовника.

- компонентне тестування перевіряє функціональність і шукає дефекти в частинах програми, які доступні і можуть бути протестовані окремо (модулі програми, об'єкти, функції і т.д.). Зазвичай компонентне (модульне) тестування проводиться викликаючи код, який необхідно перевірити чи за підтримки середовищ розробки, таких як фреймворки (каркаси) для модульного тестування або інструменти для дебагу. Всі знайдені дефекти, як правило виправляються в коді без формального їх опису в системі багів (Bug Tracking System). Один з найбільш ефективних підходів до компонентного (модульного) тестування – це

підготовка автоматизованих тестів до початку основного кодування ПЗ. Це називається розробка від тестування (test-driven development) або підхід тестування спочатку (test first approach). При цьому підході створюються і інтегруються невеликі шматки коду, навпроти яких запускаються тести, написані до початку кодування. Розробка ведеться до тих пір поки всі тести не будуть успішними.

- приймальне тестування проводиться з метою: визначення чи задовольняє система приймальні критерії там винесення рішення замовником або іншою уповноваженою особою приймається програма чи ні. Приймальне тестування виконується відповідно до Плану приймальних Робіт. Рішення про проведення приймального тестування приймається, коли: продукт досяг необхідного рівня якості та замовник ознайомлений з Планом приймальних Робіт (Product Acceptance Plan) або іншим документом, де описаний набір дій, пов'язаних з проведенням приймального тестування, дата проведення, відповідальні і т.д.

В процесі тестування системи були виявлені наступні результати:

- сторінки системи відповідають макету на всіх пристроях;
- контент відображається коректно;
- обробники подій працюють коректно;
- оформлення системи працює без помилок;
- система працює з БД без помилок;
- процес підбору працює коректно.

Після проведення усіх тестувань ПЗ пройшла всі етапи верифікації та відповідає усім умовам успішного проходження верифікації.

Висновок. В результаті проведеної роботи було створено інтелектуальну інформаційну систему «Підбір моноблоку» відповідно до бажань клієнта та було виконано розробку технічного завдання, в якому наведена мета, параметри системи її призначення, вимоги до створення цієї системи. Був описаний користувацький інтерфейс системи, а саме те яким чином створювались об'єкти у вікні, за допомогою яких команд та якого коду вони виконують свою роботу, також описана взаємодія та взяття інформації від користувача та вивід цієї інформації у вікні, та дублюванні її у вигляді текстового документу який створюється у директорії в якій і розташовується інтелектуальна інформаційна система. Було проведено тестування яке складалося з декількох етапів: функціонального тестування, тестування продуктивності, тестування програмного забезпечення, компонентне тестування, приймальне тестування. Після проведення усіх тестувань не було виявлено багів, верифікація пройшла на всіх етапах тестування.

Список використаних джерел:

1. Drabick R. Growth of maturity in the testing process. International Software Testing Institute 1999. <http://www.softtest.org/articles/rdrabick3.htm>.
2. Gelperin D., Hetzel B. The Growth of Software Testing. Commun. ACM. 1988. V. 31, N 6, P. 687 – 695..
3. IOM. To Err is Human: Building a Safer Health System: Institute of Medicine (IOM); 1999.
4. Wang, S., et al. A Cost-Benefit Analysis of Electronic Medical Records in Primary Care. The American Journal of Medicine. 2003. Vol. 114. P. 397–403.

УДК004.514

РОЗРОБКА ЗАСТОСУНКУ ДЛЯ УПРАВЛІННЯ СИСТЕМАМИ РОЗУМНОГО БУДИНКУ

Назаров Є.М.

email:jevgenij.matvijovich.nazarov@gmail.com

Науковий керівник к.т.н. Мірошниченко М.Ю.

*Таврійський державний агротехнологічний університет імені Дмитра
Моторного*

Актуальність та постановка проблеми. Світ технологій розвивається безперервно і динамічно. З кожним роком з'являються нові концепції та їх реалізації, а існуючі розширюють свої функціональні характеристики і можливості. На сьогодні однією з найпоширеніших ІТ-технологій є технологія «Інтернет речей». Нажаль ціна готових систем надто дорога для пересічного громадянина.

Метою даної статті є опис створення мобільного додатку для операційної системи Android, що дозволяє управляти пристроями розумного будинку, з мінімальним рівнем навичок програмування. Для досягнення поставленої мети, треба вирішити наступні задачі:

- проаналізувати застосунки, поширені на ринку;
- дослідити оптимальний спосіб розробки інтерфейсу;
- розробити масштабований інтерфейс програми для контролю системи.

Основні матеріали дослідження. Інтелектуальні будинки вже давно розглядається як повсякденна реальність. Про це свідчить високий рівень вигоди здійснення інвестування аналогічних проєктів, окупність яких є досить швидкою. Їх призначення міститься у забезпеченні комфорту, зручності, підвищення якості життя та безпеки для людини. Розумний будинок - не дослівний переклад англійського терміна "intelligent building". Під інтелектом в цьому випадку розуміється вміння розпізнавати певні ситуації і реагувати на них. Це означає, що будівля повинна бути спроектована так, що всі сервіси могли б інтегруватися один з одним з мінімальними витратами (з точки зору фінансів, часу і трудомісткості), а їх обслуговування було б організовано оптимальним чином.

Сучасна людина пред'являє дуже високі вимоги до комфортності свого середовища проживання:

1. естетичність приміщення, його дизайн і стиль, інтер'єр, ландшафт;
2. забезпечення оптимального мікроклімату у кімнатах будинку/квартири (температура, вологість, загазованість повітря тощо);
3. управління побутовими процесами: вода, газ, електрика, інтернет, телефонний зв'язок, кухонні прилади, система гігієни;
4. забезпечення безпеки та її контроль: сигналізація, пожежна безпека тощо.

В результаті інженерне оснащення будинків, квартир, котеджів ускладнюється, і зростає кількість пристроїв, що беруть участь у формуванні цього середовища. Покладати на господаря житла управління всіма системами стає незручно, невигідно і небезпечно. Комплексна система управління житлом «Розумний будинок» бере на себе вирішення цього завдання, залишаючи людині лише прийняття головних, «базових» рішень.

Такий «розумний дім» представляє собою саморегулюючу систему, що містить ряд підсистем: підсистеми мікроклімату, підсистеми освітлення та

регулювання роботи електронних пристроїв. В рамках «розумного будинку» може працювати «розумний холодильник», оснащений пристроями зі штучним інтелектом, які можуть аналізувати по вимогах тих чи інших продуктів, їх запас та динаміку витрат. Можна включити сюди систему безпеки. Потрібно сказати, що концепція «розумного будинку» передбачає пристрої, які можуть працювати автономно. На рис. 1 зображено складові системи «Розумний будинок».

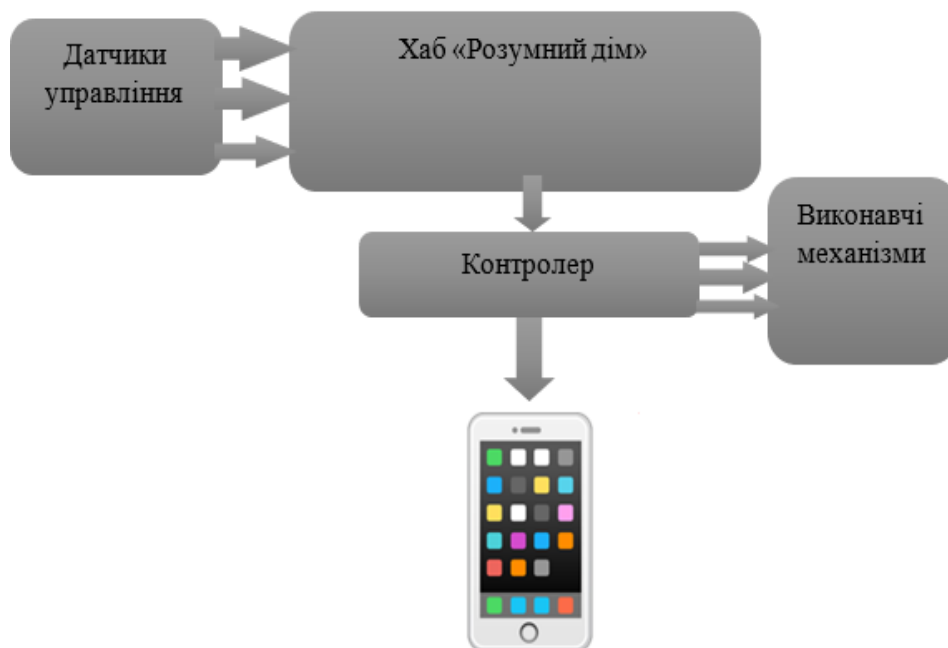


Рисунок 1 - Структура системи «Розумний будинок»

В наш час існує чимало відомих та нових брендів, але, зазвичай, в усіх є переваги та недоліки.

Наприклад система Ажах забезпечує комфорт і зручність в управлінні життєзабезпеченням приміщення; гарантує безпеку житла в повній мірі, контролюючи межі об'єкту на предмет злому, а також електричну, пожежну, газову та інші можливі загрози для дому.

Обладнання «Розумний будинок» Ажах працює на надійно зашифрованою і захищеною двосторонньою технологією радіозв'язку – Jeweller власної розробки, має повну автономність від електромережі завдяки резервному джерелу живлення. Перевагами цього апаратного та програмного застосування є:

- простий монтаж;
- бездротовий канал зв'язку між системними елементами;
- велика зона дії сигналу (до 2000 м);
- наявність захисту від зняття будь-якого з датчиків;
- автономна робота від акумулятора (до 16 годин);
- Wi-Fi і GSM-зв'язок.

До недоліків можна віднести:

- функціонування тільки з роботою центрального контролера, тобто відсутність автономності датчиків;
- відсутність власної камери відео спостереження;
- управління тільки через телефон.

Система «Розумний будинок» BroadLink являє собою комплект сучасних цифрових пристроїв, створених для раціонального управління побутовою

технікою, а також освітлювальної, енергетичної, охоронної та іншими системами в будинку.

Кожен елемент такого комплексу може працювати як самостійно, так і взаємодіяти один з одним.

До переваг цієї системи можна віднести:

- легке встановлення, підключення і налаштування;
- широкий асортимент датчиків (вологості, температури, освітлення, шуму, забруднення повітря);

- автономна робота датчиків; бездротова взаємодія пристроїв між собою;
- наявність власної камери відеоспостереження;
- можливість контролю по Wi-Fi через Інтернет.

Недоліки системи «Розумний будинок» BroadLink:

- відсутність українського інтерфейсу;
- невелика дальність дії сигналу (до 50 м);
- відсутність резервного живлення; пульт працює тільки на прийом сигналів.

Розумний будинок Fibaro відноситься до професійного обладнання по забезпеченню автоматизації та безпеки будинку з найширшим функціоналом. Однак, на відміну від багатьох подібних систем, потребує встановлення та налаштування своєї апаратури досвідченими фахівцями.

Перевагами цієї системи є:

- різноманітність датчиків і пристроїв;
- наявність камери відеоспостереження;
- великий вибір сценаріїв для користувача;
- розсилка повідомлень відразу на кілька телефонів.

Недоліками системи Fibaro є:

- висока вартість обладнання;
- потреба в професійному монтажі і налагодженні;
- обов'язкове підключення центрального контролера до інтернету через LAN-кабель;
- неможливість функціонування без центрального хаба;
- відсутність резервного живлення хаба;
- обмежена дальність сигналу (до 50 м без перешкод, хоча ця проблема вирішувана);
- затримка Push-повідомлень;
- необхідність в обов'язковій установці програмного забезпечення на ПК, а також урізаний мобільний застосунок.

В таблиці 1 наведено основні характеристики систем «Розумний дім» і управління та проведено співставлення даних вимог до можливостей цих систем. Позначення «+» означає, що дана функція є в системі, а знак «-» показує, що дана функція відсутня у системі, яка розглядається.

Враховуючи недоліки розглянутих систем і програмного забезпечення до них, робимо висновок про необхідність створення нової інтелектуальної системи «Розумний будинок», яка б мала змогу не тільки контролювати і управляти температурним, вологісним режимом, рівнем освітлення, рівнем загазованості у приміщенні, але й зберігати поточні дані з можливістю створення запиту до системи та власноруч розробленим інтерфейсом створеного під необхідні умови експлуатації. Для розробки необхідного програмного забезпечення буде використано новітній сервіс Blynk.

Таблиця 1 – Порівняльний аналіз можливостей систем «Розумний дім»

	Ajax Systems	BroadLink	Fibaro
Назва			
Простота підключення	+	+	-
Зона дії	до 2000 м	до 50 м	до 50 м
Автономність	+	+	-
Вартість	200\$	200\$	600\$
Достатня кількість підключених пристроїв	+	+	+
Підключення до мережі	Wi-Fi та GSM-зв'язок	Wi-Fi	LAN-кабель

Blynk – це сервіс Інтернету речей (IoT), розроблений для того, щоб зробити дистанційне керування та зчитування даних датчиків. Цей сервіс позбавляє необхідності інтенсивного кодування і полегшує доступ до пристроїв на смартфоні. Його можна безкоштовно використовувати, як і тим хто не володіє навичками розробки та програмування, так і розробникам з досвідом.

Додаток Blynk є безкоштовним для Android та iOS. Робочий процес швидкий: під час запуску нового проекту пропонується вибрати свою дошку для розробки з великого списку, а також спосіб підключення. Потім програма відправляє токен авторизації електронною поштою для підключення до пристрою через сервер Blynk.

Елементи управління називаються відметами. Це різні типи методів введення та виведення інформації, включаючи кнопки, повзунки, джойстик, графіки та текстовий зворотний зв'язок. Існують також віджети, специфічні для компонентів, зі стилізованими елементами управління світлодіодами, РК-дисплеями і навіть потоковим відео. Також примітні віджети, які додають функції, такі як автоматична публікація в Twitter і сповіщення користувача.

На рисунку 2 зображено функціональну схему. Технічне супроводження системи управління «Розумний будинок» для вирішування поставної задачі включає в себе вимірювальні пристрої (датчик вимірювання температури, гігrometer, датчик диму, датчик руху) і визначає стан технологічного середовища у приміщенні.

Виміряні дані поступають в базу даних, де зберігаються протягом тривалого часу, і на вхід блоку моделювання.

В блоці моделювання виміряні дані порівнюються з нормованими даними з бази нормованих значень.

Далі відповідно до заданого алгоритму роботи системи і результатів порівняння виміряних і нормованих значень блок прийняття рішень видає інформацію, яка у вигляді управляючих впливів передається на виконавчі механізми.

Одночасно інформація подається на блок візуалізації даних (мобільний пристрій), тобто інтерфейс користувача.

За допомогою інтерфейсу користувача відбувається інформування користувачів не тільки про стан технологічного середовища, але й про стан виконавчих механізмів.

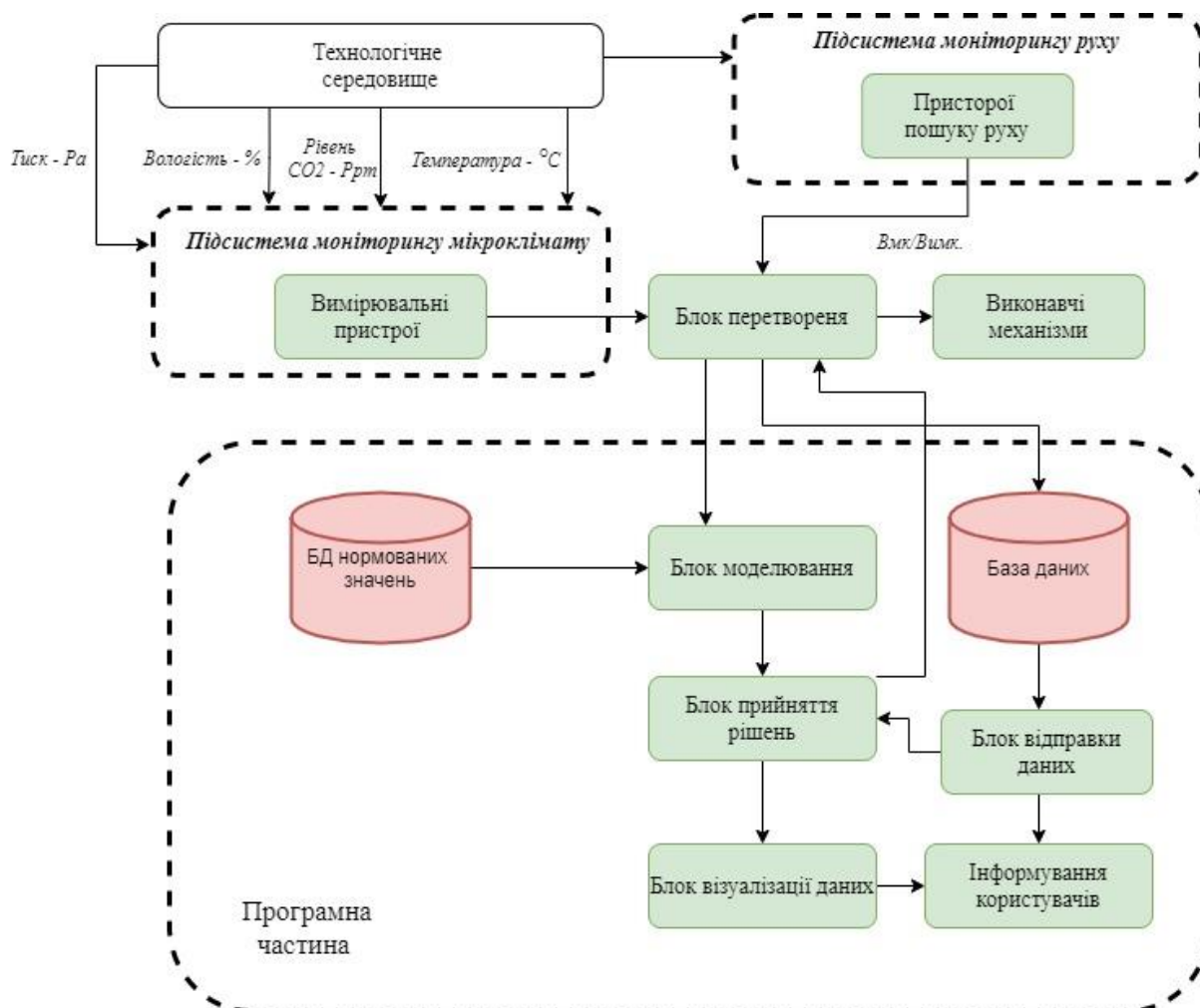


Рисунок 2. Функціональна схема роботи системи для управління пристроями «Розумний будинок»

Головними користувачами системи управління пристроями «Розумний будинок» є :

- Адміністратор;
- Клієнт;
- Веб-сервіс;
- Мікроконтролер.

Адміністратор відповідає за супровід програмного забезпечення та програмної реалізації моніторингу системи і управління мікрокліматом.

Клієнт (користувач) слідкує за роботою програмного забезпечення, формує статистичні звіти станів мікроклімату приміщення та датчиків у приміщенні, з'єднує систему з веб-сервісом.

Веб-сервіс проводить авторизацію клієнтів, забезпечує обмін даними між клієнтом та мікроконтролером.

На рисунку 3 наведена діаграма варіантів використання системи для управління пристроями «Розумний будинок» відповідно людям та технічним системам.

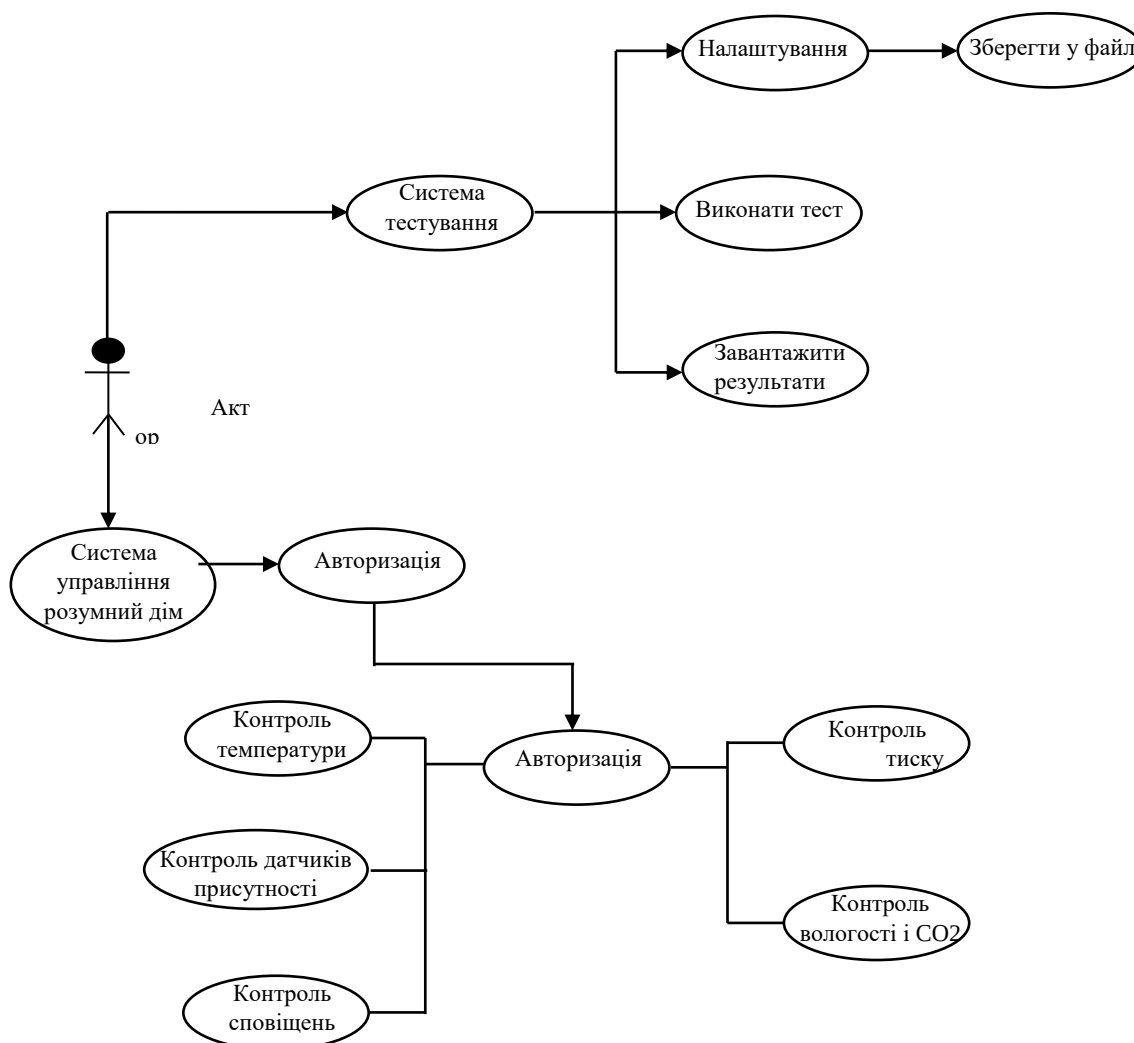


Рисунок 3. Діаграма варіантів використання

Мікроконтролер виконує команди, отримані від веб-сервісу, здійснює відправку поточного стану на веб-сервіс, якщо надійшов запит та забезпечує вимірювання та регулювання температури та вологості.

Для роботи по створенню інтерфейсу бажано мати вже готову апаратну частину. На рисунках 4 та 5 зображено апаратну частину проекту.

Програмна реалізація системи освітлювання виконується на мові програмування C/C++ у середовищі розробки Arduino IDE. При розробці використовуються бібліотеки «ArduinoOTA.h» та «BlynkSimpleEsp8266.h».

Програмна реалізація системи моніторингу макроклімату в приміщенні виконано також використовуючи мову програмування C/C++ у середовищі розробки Arduino IDE, при роботі були використані бібліотеки <ESP8266WiFi.h>, <ESP8266mDNS.h>, <WiFiUdp.h> та <Adafruit_BMP280.h>.

Arduino IDE - це середовище програмування, яке використовує C++ і призначене для програмування всіх плат серії Arduino. Аббревіатура IDE розшифровується як Integrated Development Environment та перекладається як «інтегроване середовище розробки». Arduino IDE дозволяє створювати програми

в зручному текстовому редакторі, компілювати їх в рідний код, і завантажувати на всі версії плати Arduino.

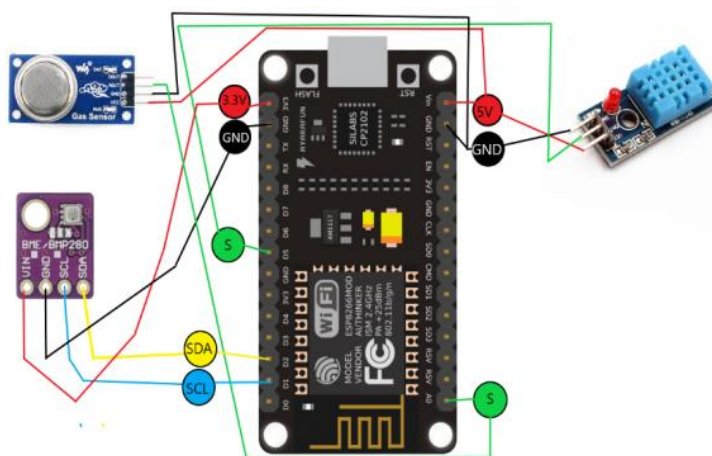


Рисунок 4. Апаратна реалізація системи моніторингу мікроклімату в приміщенні

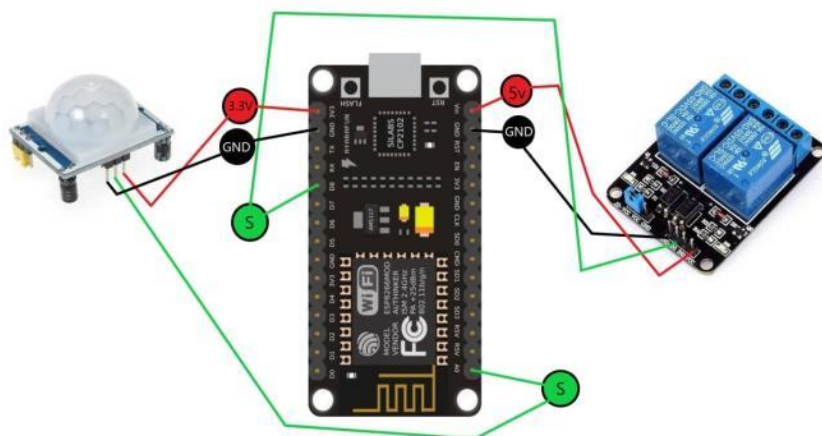


Рисунок 5. Апаратна реалізація системи автоматизації освітлення

Бібліотека ArduinoOTA в комплекті з пакетами ESP8266 і ESP32 Arduino працює тільки з рідними бібліотеками WiFi. Ця бібліотека дозволяє завантажувати ескіз на esp8266 або esp32 через Ethernet з бібліотекою Ethernet або EthernetENC. Завантаження через рідну бібліотеку Wi-Fi також працює.

Бібліотека Arduino для датчиків тиску і висоти BMP280 сумісна з усіма архітектурами та використовується на всіх платах Arduino.

Середовище «Blynk» має свою бібліотеку де вже згенеровані всі можливі команди. Це полегшує процес розробки, а велика кількість інформації в відкритому доступі згладжує поріг входження. Розроблений інтерфейс мобільного додатку показано на рисунку 6.

Інтерфейс складається з анімованого графіку. Зміна кривої відбувається в реальному часі віджетів-індикаторів, зміна кольору яких відбувається в реальному часі. Відповідно змінюється градієнт перемикача. Також є можливість відкрити графік в окремому вікні, і скачати базу даних зі всіма показниками для

подальшого аналізу. Нижче приведено лістинг фрагменту коду використаного для автентифікації.

```
// Аутентифікація
char auth[] = "FtjQAgXNx5I4ly52lKZpKJmndYGsspTr";
char ssid[] = "BlynkKONTROL"; //ім'я точки доступу WIFI
char pass[] = "123456780"; //пароль точки доступу WIFI
```



Рисунок 6. Інтерфейс мобільного додатку

Цей код використовується для з'єднання плати з інтерфейсом. Це дозволить контролювати пристрій з будь-якої точки світу з виходом в інтернет використовуючи вже готові сервери середовища «Blynk», що полегшує розробку комплексної системи та дає свої переваги.

На рисунку 7 приведено функцію відкриття графіку моніторингу клімату в реальному часі.

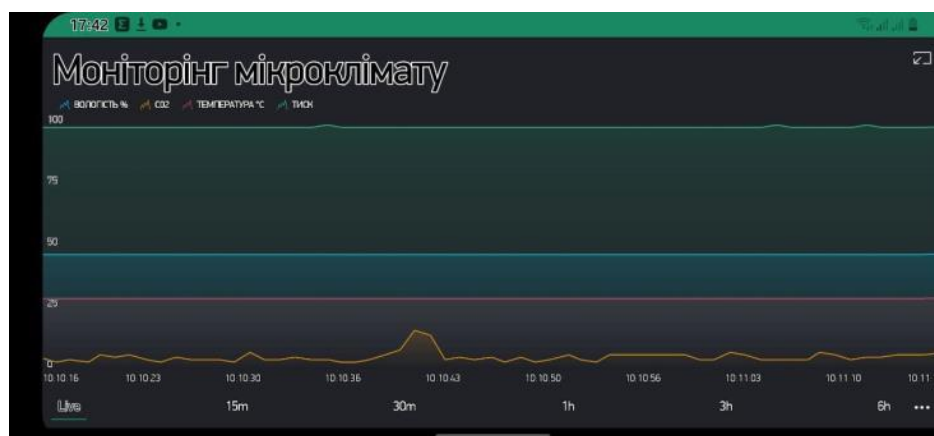


Рисунок 7 - Графік моніторингу клімату в реальному часі

Нижче наведено фрагмент лістингу коду для підключення датчиків до інтерфейсу та зчитування інформації з них в реальному часі на віртуальних пінах в середовищі «Blynk».

```
Blynk.virtualWrite(PIN_HUMIDITY, hh);  
Blynk.virtualWrite(PIN_TEMPERATURE, t);  
Blynk.virtualWrite(PIN_PRESSURE, p);  
Blynk.virtualWrite(PIN_ALTITUDE, a);  
int Gas_analog = A0; // used for ESP8266  
int Gas_digital = D1; // used for ESP8266
```

Висновки. В роботі показано спосіб створення мобільного додатку для операційної системи Android, що дозволяє управляти пристроями розумного будинку, з мінімальним рівнем навичок програмування. Перевагою описаного в роботі способу є відсутність необхідності створення власного сервера та інтегрування в нього логіки Arduino. Це дозволяє мінімізувати розмір програмного коду вивантаженого в апаратну частину та зменшити час розробки проекту.

Список використаних джерел:

1. Миронець І.В., Луценко Б.О. Функціональні можливості та оптимізація роботи автоматизованої системи управління тату-обладнанням. Вісник Черкаського державного технологічного університету, 2020. №2. С. 60–67.
2. Gerhart James Home Automation and Wiring. McGraw-Hill Professional, 1999. 336 pp.
3. Що таке інтернет речей? [Електронний ресурс] URL: <https://tokar.ua/read/26780>
4. Розумний будинок - з чого він складається та чи потрібен вам [Електронний ресурс] URL: <https://nachasi.com/tech/2018/06/25/smart-house-faq/>
5. Які переваги «розумного дому»? [Електронний ресурс] URL: <https://klyuch.com.ua/m/articles/economy/yaki-perevagy-rozumnogo-domu/>
6. Кращі системи "Розумний будинок" по виробниках 2021 року. [Електронний ресурс] URL: <https://vencon.ua/ua/articles/rejting-sistem-umnyy-dom-ro-proizvoditelyam>
7. Розробка об'єктної моделі конкретної системи збору даних [Електронний ресурс] URL: <https://works.doklad.ru/view/cG6yb8ulsNs.html>

УДК 004.7

ОПТИЧНІ-ВОЛОКНА

Гузьок В.В.

e-mail: goxo1512@gmail.com

Державного університету «Житомирська політехніка»

Актуальність та постановка проблеми. Потреби людей у швидкому та стабільному підключенні до глобальної мережі, вимагає від провайдерів використання сучасних технологій та покращення старих частин мережі. таких як використання оптично-волоконних кабелів та сучасного мережевого обладнання.

Глобальна мережа покриває величезну частину планети. Для забезпечення якісної і швидкої передачі інформації звичних металевих кабелів буде не достатньо через значне затухання сигналу на невеликій відстані, що створює необхідність встановлювати великої кількості ретрансляторів.

Основні матеріали дослідження. Для того щоб спростити створення глобальних чи місцевих магістралей було винайдено волоконно-оптичні для яких наші відстані не є проблемою.

"Всі різновиди оптоволокна володіють подібною структурою. Це двохшарова кварцова нитка циліндричної форми, яка складається з серцевини і оболонки. Серцевина забезпечує поширення світлових хвиль, які в процесі повністю відбиваються від внутрішньої поверхні оболонки. Серцевина модифікується германієм, що забезпечує більш високе значення заломлення, ніж у оболонки. Поверх оболонки знаходиться шар з акрилатного лаку, який захищає нитку від механічних впливів і вологи. Світло не може проникнути в оболонку до рівня лакового покриття (максимум на глибину, рівну довжині хвилі), а значить, не взаємодіє з ним." [1]

За технічними параметрами та зовні всі волоконно-оптичні кабелі схожі, що практично не розрізнити (рис. 1). Через що необхідно маркувати та вести облік обладнання.

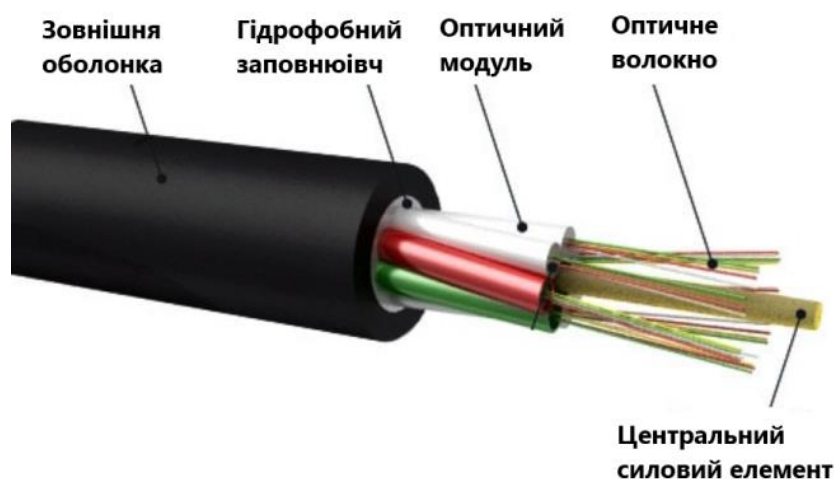


Рисунок 1. Конструкція волоконно-оптичного кабелю

Є два основних типів волокон: одномодові(SM) та багатомодові(MM) (рис. 2). В обох типів оболонка кабелю має діаметр 125+1мкм. Серцевини мають

різний діаметр. У одномодовій діаметр в межах 3-10 мкм, а у багатомодовій діаметр в межах 50-62,5 мкм.

Головними параметрами для одномодового волокна є діаметр модової плями та дисперсні характеристики. В залежності від виду волокна та функціональної довжини хвилі, розмір плями може бути від 8 до 10 мкм. Максимальне відхилення повинно складати 10% від середньо допустимого діаметру, згідно з міжнародними стандартами.

Через високі характеристики одномодовий кабель дорожчий за багатомодовий, а головна відмінність є в режимі проходження світлових променів.

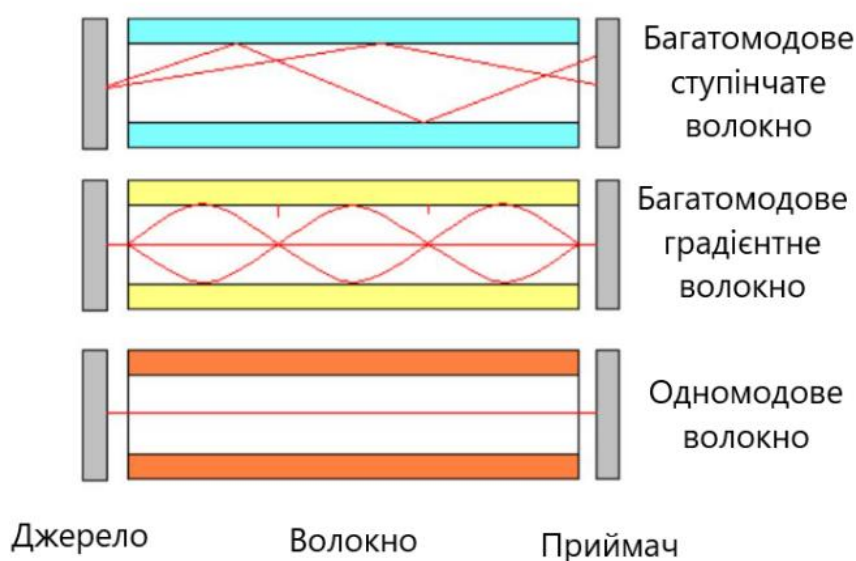


Рисунок 2. Типи оптичних-волокон

"Передача даних в одномодових здійснюється на довжинах хвиль 1300/1500 нм. Вони забезпечують мінімальну дисперсію і втрату сигналу, що дає можливість передавати дані на більшу відстань, в порівнянні з мультимодовим кабелем (до 80 км). "[1]

У багатомодових волокнах сигнал піддається спотворенням, через істотний розкид траєкторії променів. При такому спотворенні максимальна дальність передачі сигналу становить 2-5 км.

Є два типи багатомодових волокон, в залежності від заломлення по перетину: ступінчасті та градієнтні.

Ступінчасті волокна передають сигнал до 1 км. Градієнтні дозволяють передавати сигнал до 5 км.

В ступінчастих волокнах передається одночасно до 1000 мод з різним розподілом по волокну. Градієнтний забезпечує плавну зміну заломлення, що забезпечує стабільну передачу сигналу.

Одномодові волокна застосовують для побудови в трьох транспортних рівняннях: магістрального, розподільного та рівня доступу.

Багатомодові волокна використовують для локальних мереж та транспортних мереж на рівні доступу

Нині досі ведуться розробки у цій сфері. Вчені з фізико-технічного інституту (MIPR) розробили новий тип оптичного волокна, яке має великий

діаметр серцевини та зберігає когерентні властивості світла. Збереження властивостей світла має вирішальне значення, при використанні оптично-волоконних кабелів.

"«Дослідження в області оптичних волокон є однією з найбільш швидко що розвиваються областей оптики. За останнє десятиліття було запропоновано та реалізовано безліч технологічних рішень. Наприклад, вчені і інженери IRE RAS тепер можуть виробляти оптичне волокно практично будь-якого діаметру з довільною поперечною структурою», - каже Василь Устимчик, співавтор дослідження та професор MIPR. "[2]

Подібні дослідження є майбутнім для розвитку глобальної мережі адже всім хочеться мати стабільне та швидке з'єднання з мережею.

Висновок. Було розглянуто оптичні-волокна, їхні види, характеристики та місця застосування. Використання оптично-волоконних кабелів значно покращує якість зв'язку та спрощує розгортання мереж. Було представлено приклад дослідження які ведуться для покращення оптичних-волоконних кабелів.

Список використаних джерел:

1. Які бувають типи оптичних волокон? [Електронний ресурс]. – Режим доступу: <https://klaster.ua/ua/stati-i-obzory/kakie-byvayut-tipy-opticheskikh-volokon/>
2. Вчені розробили новий тип оптичного волокна, яке зберігає властивості світла [Електронний ресурс]. – Режим доступу: https://dimicom.com/uk/module/xipblog/module-xipblog-single?page_type=post&id=2&rewrite=vcheni-rozrobili-novij-tip-optichnogo-volokna-yake-zberigaye-vlastivosti-svitla

УДК 004.891.2

РОЗРОБКА ІГРОВОГО ЗАСТОСУНКУ ДЛЯ МОБІЛЬНОЇ ПЛАТФОРМИ У ЖАНРІ ПРИГОДНИЦЬКОЇ СТРАТЕГІЇ

Арієнєков О.М.

e-mail: oleksii.arinienkov@nure.ua

Новіков Ю.С. ст. викл.

e-mail: yuriy.novikov@nure.ua

Харківський національний університет радіоелектроніки

Актуальність та постановка проблеми. Розмір ринку ігор станом на 2021 рік складає приблизно 160 мільярдів доларів (за даними сайту statista.com). Цей показник є другим за розміром в усій галузі ринку розваг, уступаючи лише телебаченню (приблизно 400 мільярдів доларів). Окрім того, ігрова індустрія є першою по темпу розвитку. Навіть після «буму» у перше десятиріччя двадцять першого століття вона зберегла ріст у 8-9% кожного року та показала стійкість перед всесвітніми кризами.

45% доходів в індустрії приходить саме на мобільний геймінг. Це пов'язано з доступністю пристроїв на основі ОС Android та IOS.

Значна частина зусиль фахівців в ІТ-індустрії як правило направлена на створення та підтримку певного програмного продукту. Але на українському ринку більшість діяльності ІТ-компаній направлена саме на надання послуг, з стійким показником в межах 75-80%. Це пов'язано з низкою різноманітних факторів (складність отримання фінансування, ризику просування власного продукту і т. д.) та приводить не лише до негативних, але й до позитивних наслідків, таких як неймовірно високий експорт ІТ-послуг та доходу з них. Однак через це частка програмних продуктів, які належать саме українським компаніям, доволі низька, в особливості при врахуванні високої долі ІТ-фахівців серед громадян України.

Метою даного проекту є створення, подальше просування та підтримка оригінального мобільного ігрового додатку. Додаткова увага буде приділена розробці комплексних програмних алгоритмів.

Основні матеріали дослідження. Однією з ідей даного проекту є створення саме “оригінального” ігрового додатку. Цей вибір приведе до необхідності у: ретельному дослідженні ринку (з метою пошуку цільової аудиторії та аналізу їх потреб), створенні плану розробки та просування продукту, формуванні унікальних алгоритмів, за якими буде працювати ігрова логіка.

Ігровий додаток буде виконано у жанрі пригодницької стратегії. Ігрок буде контролювати корабель та за його допомогою мандрувати пригодницькою мапою, досліджуючи ігровий світ, торгуючи з містами та вступаючи у пошагові стратегічні битви у морі. Таким чином, якість ігрового продукту буде залежати від ступені реалізації наступних складових:

- Пригодницька мапа та алгоритм її генерації;
- Різномісність системи ігрової економіки;
- Арт-складова проекту;
- Комплексність системи ігрових боїв.

Одним з ключових аспектів ігрового світу є комплексна система внутрішньої економіки. За допомогою симуляції процесів з реального світу можливо досягти одночасно цікавих та правдоподібних ігрових ситуацій.

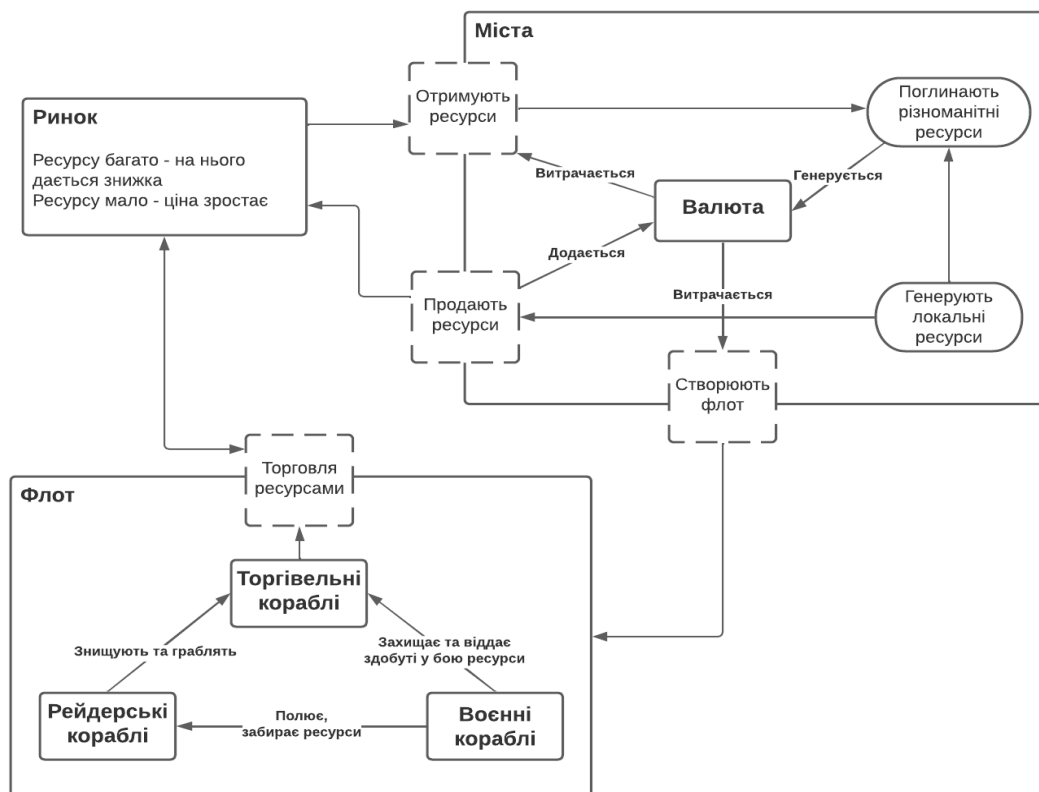


Рисунок 1. Діаграма економічних зв'язків в ігровому застосунку

Таким чином, система одночасно буде “живою” та динамічною, вона буде балансуватись самостійно. Гравець буде змотивований шукати можливості максимізувати свій прибуток, наприклад, користуючись війною між двома містами і відповідно виникшим дисбалансом ресурсів.

Для створення правдоподібної та цікавої з ігрової точки зору ігрового досвіду пригодницької мапи було прийнято рішення генерувати її на початку гри. Для цього використовується алгоритм wave function collapse. За допомогою такого підходу вирішується ряд проблем, таких як необхідність великого об'єму праці для власноручного створення ігрової мапи та створення передумов для повторного проходження гри, вже з новою мапою та економічними зв'язками.

У другій половині 2022 року технології на базі штучного інтелекту одночасно досягли відносно високого рівня якості та стали доступні звичайним користувачам. Існує дискусія з питання «чи замінить штучний інтелект розумову діяльність людини». У якості експерименту значна частина арт-складової ігрового додатку буде згенерована за допомогою технології stable diffusion. Дослідження ставить на меті такі питання, як:

- Чи можна досягти відносно високого рівня 2D-графіки (спрайтів та інтерфейсів), користуючись лише генератором на основі stable diffusion та базовою обробки у графічних редакторах;
- Наскільки раціональне виконання задач такого типу у порівнянні з делегуванням їх професіоналу (художнику);
- Чи можна витримати згенеровані ілюстрації у єдиній стилістиці для формування цілісності ігрового візуалу.

Зацікавленість гравця у ігровому процесі як правило напряму корелює з адекватністю поведінки “акторів” гри, так званих НПС. У даному проєкті їх роль віддається кораблям, як нейтральним, так і тим, що відносяться до певних городів.

Їх поведінка буде реалізована за допомогою технології behaviour tree. Завдяки цьому можна буде коригувати діяльність НПС в залежності від їх типів.

Вагомим викликом при створенні власного ІТ-продукту є необхідність його просування. Для подолання цієї перешкоди будуть прийняті такі міри, як аналіз ринку, пошук цільової аудиторії, аналіз потреб аудиторії, ведення соцмереж та створення таргетингової реклами.

Висновок. Було розроблено план створення та просування мобільного ігрового додатку. Запропоновано втілення комплексних алгоритмів для вирішення типових проблем для ігор подібних жанрів. Досліджено можливість використання технологій на базі штучного інтелекту для покращення якості програмного додатку. Розглянуто типові алгоритми, за допомогою яких реалізується логіка ігрових об'єктів.

При подальшій розробці та реалізації гри поставлено ряд цілей для дослідження. Серед них розглядаються особливості просування ІТ-продукту, доцільність використання технологій на базі штучного інтелекту та алгоритми подолання перешкод, що виникають по ходу створення комплексної програмної системи.

УДК 004.9

ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОПЕРАТИВНОГО РОЗРАХУНКУ НОРМ ЧАСУ ВИГОТОВЛЕННЯ ДЕТАЛЕЙ ТА МЕХАНІЗМІВ

Вершков О.О., к.т.н., доцент

e-mail: oleksandr.vershkov@tsatu.edu.ua

Бондаренко Л.Ю., к.т.н., доцент

e-mail: larysa.bondarenko@tsatu.edu.ua

Гавриленко Є.А., д.т.н., професор

e-mail: yevhen.havrylenko@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність дослідження та постановка проблеми. Ефективне планування та управління підприємством та виготовленням продукції сільськогосподарського машинобудування обумовлюється, в тому числі, оперативністю визначення параметрів технологічних операцій машинобудівного виробництва, які є основною розрахунків елементів технологічного процесу. У нашому випадку – це оперативний розрахунок норм часу на виготовлення деталей певного типу.

Час обробки заготовки і собівартість виконання операції є критерієм, що характеризує доцільність її побудови з урахуванням заданої виробничої програми і певних організаційно-технічних умов. Технічна норма часу, що визначає витрати часу на виконання операції, служить основою для оплати роботи верстатнику, калькуляції собівартості деталі та виробу.

На основі технічних норм часу розраховуються тривалість виробничого циклу, потрібну кількість верстатів, інструментів, робітників, визначається виробнича площа ділянок і цехів. Норма штучного часу є одним з основних факторів для оцінки досконалості технологічного процесу і вибору найбільш прогресивного варіанту обробки заготовки. При проектуванні технологічних процесів одним із важливих завдань є визначення економічно обумовлених норм часу на обробку деталей.

Застосування ЕОМ при проектно-конструкторських роботах у своєму розвитку пройшло кілька стадій і перетерпіло значні зміни. З появою обчислювальної техніки створюються програми автоматизованих розрахунків на основі методів обчислювальної математики. Із впровадженням спеціалізованих термінальних пристроїв з'являються універсальні програми для ЕОМ для рішення як розрахункових, так і деяких рутинних проектних завдань (виготовлення креслень, специфікацій, текстових документів і т.п.).

Автоматизація розрахунку норм праці проводиться в даний час багатьма організаціями: частіше створюються системи розрахунку операційних норм часу для умов індивідуальної організації праці. Подібні системи створюються іноді як автономні системи нормування на базі заданого трудового і технологічного процесу.

Метою даного дослідження є розробка спеціалізованого програмного модуля для визначення норм часу виготовлення деталі «Корпус А32.657.003А».

Об'єкт дослідження – технічна підготовка виробництва корпусних деталей на підприємстві.

Предмет дослідження – автоматизація процесу визначення норм часу виготовлення деталі «Корпус А32.657.003А».

Для досягнення поставлених цілей в роботі вирішуються наступні задачі:

- провести аналіз предметної області;
- розробити вимоги до програмного модуля;
- виконати опис прийнятих проектних рішень та програмної реалізації;
- розробити програмний модуль визначення норм часу;
- провести аналіз дослідної експлуатації;
- провести аналіз умов праці користувача пропонованого програмного забезпечення.

- виконати економічні розрахунки ефективності впровадження програмного модулю;

Основні матеріали дослідження. Одним з важливих техніко-економічних показників в машинобудуванні є показник трудомісткості виготовлення. На даний момент існує цілий ряд автоматизованих систем, тим чи іншим чином пов'язаних із завданням визначення витрат часу в машинобудівному виробництві.

Аналіз досвіду підприємств свідчить, що сучасні інформаційні технології використовуються, в основному, для вирішення наступних завдань у сфері організації та нормування:

- розробки нормативів з праці;
- розрахунку норм часу і виробітку;
- проектування оптимальних систем обслуговування виробництва і розрахунку оптимальних норм обслуговування і чисельності працівників.
- проектування оптимального розподілу праці в бригаді та розрахунку комплексних норм трудомісткості;

Автоматизація розробки нормативів з праці здійснюється як за результатами вивчення витрат робочого часу, так і на основі мікроелементних нормативів (найчастіше використовується вдосконалена базова система мікроелементних нормативів часу - БСМ-1).

У першому випадку вихідною інформацією є карти аналізу трудового процесу, розчленування технологічного процесу і результати хронометражних спостережень, що містять відомості про витрати робочого часу на елементи трудового процесу, значення кількісних і якісних факторів, що характеризують ці процеси. В результаті обробки даних формуються математичні моделі залежності часу виконання елементів операції від факторів, що впливають, а також створюються нормативні таблиці, що містять нормативні значення часу відповідно до значень впливають на них факторів. Більшість програмних систем передбачають автоматизацію лише частини робіт, пов'язаних з проектуванням нормативів за даними вивчення витрат робочого часу, а саме: оцінку суттєвості впливу кількісних факторів на час виконання елементів трудового процесу і побудова моделей, що характеризують залежності часу від значущих чинників.

Крім того, нормативи з праці можуть розроблятися на основі нормативів меншій мірі укрупнення (наприклад, мікроелементних нормативів). У цьому випадку обов'язковим етапом є проектування можливих варіантів виконання трудового процесу, опис їх за допомогою символіки мікроелементів і вибір на основі порівняння варіантів оптимального з них. При цьому реалізується системний підхід до проектування та нормування трудових процесів на основі використання базової системи мікроелементних нормативів і ЕОМ. Розроблено прикладні програмні системи та Методичні рекомендації з розрахунку на ЕОМ норм часу на базі мікроелементних нормативів.

Автоматизація розрахунку норм праці проводиться в даний час багатьма організаціями: частіше створюються системи розрахунку операційних норм часу для умов індивідуальної організації праці. Подібні системи створюються іноді як

автономні системи нормування на базі заданого трудового і технологічного процесу. В цілому автономна система моделює дії фахівця в галузі нормування праці, тобто забезпечує введення даних, необхідних для розрахунку, пошук відповідної нормативної формули, заздалегідь введеної в пам'ять ЕОМ, визначення величин основного, допоміжного і штучного часу за конкретним значенням факторів, що впливають на витрати часу, роздруківку операційних карт механічної обробки. Проте частіше створюються комплексні системи автоматизованого проектування виробничих (трудових і технологічних) процесів і нормування праці.

У таких системах розрахунок норм часу проводиться в єдиному циклі з автоматизованим проектуванням технологічних і трудових процесів. Це більш перспективний напрямок у порівнянні з вищевказаним, оскільки вихідна інформація, що задається для автоматизованого проектування технологічних і трудових процесів, використовується і для розрахунків норм часу, що відповідно зменшує трудомісткість її підготовки. Такі системи забезпечують більш високий рівень нормування праці.

Таким чином, в даний час накопичений певний досвід застосування сучасних інформаційних технологій для вирішення завдань в області організації та нормування праці. Однак охоплення робіт автоматизованим нормуванням недостатньо високий: в середньому нормується з використанням ЕОМ лише 5-7% технологічної трудомісткості.

Нормування праці встановлює витрати необхідного часу на виготовлення даного предмета праці на даному підприємстві, тобто норму праці - конкретне вираження міри праці.

Одною з складових, які визначають ефективність САПР є прикладне програмне забезпечення, під яким звичайно розуміють набір програм, що реалізують рішення на ЕОМ конкретних задач проектування. Важливим етапом втілення наукової ідеї у реальність є проектно-конструкторська та технологічна підготовка виробництва.

Автоматизація машинобудівного підприємства повинна починатися не з впровадження систем управління підприємства або електронного документообігу, а з впровадження CAD / CAM / CAE систем. На сучасному рівні розвитку вітчизняного виробництва та систем обміну інформації на ньому, може йти мова тільки про системи автоматизованого проектування та управління машинобудівним підприємством.

Інформаційна система підприємства покликана автоматизувати традиційно існуючу технологію проектування і виготовлення. В умовах ринкової економіки для виживання підприємств велике значення має зниження собівартості продукції і зменшення термінів розробки виробу. В процесі дослідження ринку програмного забезпечення нормування технологічних процесів, увагу до себе привернули СПРУТ-ТП та ВЕРТИКАЛЬ.

ВЕРТИКАЛЬ - система автоматизованого проектування технологічних процесів, здатна вирішувати більшість завдань автоматизації процесів технологічної підготовки виробництва. САПР ТП ВЕРТИКАЛЬ дозволяє: проектувати технологічні процеси в декількох автоматизованих режимах; розраховувати матеріальні і трудові витрати на виробництво; формувати всі необхідні комплекти технологічної документації, використовувані на підприємстві; вести паралельне проектування складних і наскрізних техпроцесів групою технологів в реальному режимі часу; підтримувати актуальність технологічної інформації за допомогою процесів управління змінами;

забезпечувати інженерний документообіг в частині заявок на проектування засобів технологічного оснащення. Надає користувачам всю необхідну довідкову інформацію, а також дозволяє організувати і розвивати бази даних підприємства – універсальний технологічний довідник, що входить в САПР ТП ВЕРТИКАЛЬ.

Інтегрована інструментальне середовище автоматизованого проектування СПРУТ являє собою набір інструментальних засобів для швидкої розробки програмного забезпечення прикладних систем конструкторсько-технологічної підготовки виробництва, що реалізують принципи RAD-технології при їх створенні.

Призначення системи:

- автоматизація розробки і нормування технологічних процесів;
- формування документації: від конструкторської специфікації до виробничих документів;
- управління процесом технологічного проектування;
- підготовка даних для систем управління ресурсами підприємства ERP і планування виробництва MES.

СПРУТ забезпечує швидке отримання прототипу прикладної системи з одночасною її адаптацією та інформаційної інтеграцією в єдину проектне програмне середовище. Інструментальна програмне середовище виступає в даному випадку як ефективний засіб для формування структури, зберігання і подальшої інтеграції знань, зберігаючи і об'єднуючи розробки, виконувані окремими розробниками проекту, виконуючи тим самим роль системного та інформаційного інтегратора.

Використання єдиної інструментального середовища дозволяє здійснити інформаційну, організаційну та системну інтеграцію, реалізувати в єдиному інформаційному просторі системи проектування виробів, технологічну та технічну підготовку виробництва, управління матеріальними ресурсами і необхідний документообіг.

Вдалий вибір концептуальної основи системи, широкий набір функціональних підсистем, наявність тексто-графічної СУБД і потужного мови розробки дозволили розробникам не обмежуватися якимось певним застосуванням системи СПРУТ.

Архітектура системи відмінно зарекомендувала себе в машинобудуванні - від локальних систем САПР різного призначення (проектування вузлів, виробів, оснащення, інструменту, УП для верстатів з ЧПУ і т.д.) до комплексних систем конструкторсько-технологічної підготовки виробництва з виходом на управління ресурсами підприємства; в муніципальній сфері - системи обліку та управління об'єктами житлово-комунального господарства, ГІС для працівників ДАІ, пожежних служб і т.д., а також при розробці систем проектування виставкових комплексів, інтер'єрів, будівельного проектування і багато в чому іншому.

Норми праці у виробництві виконують важливі функції вимірювача продуктивності праці, заходи витрат праці та винагороди за працю. Будучи вимірником рівня продуктивності праці, норма витрат праці виконує функцію засобу, інструменту управління виробництвом. За допомогою норми оцінюється рівень продуктивності праці. Як міра витрат праці норма є основою розрахунку й обліку багатьох показників виробничо-господарської діяльності підприємства.

На основі норм вибираються варіанти конструкції виробу, способи здійснення технологічних процесів, методи організації виробництва, праці та управління. Як міра винагороди за працю норма є основою розрахунку заробітної плати, її диференціації за кількістю та якістю праці. Для розрахунку норм часу та

технологічних процесів найбільш широко використовуються такі програмні продукти, як СПРУТ або ВЕРТИКАЛЬ, які надають можливість не тільки автоматизувати розробку і нормування технологічних процесів, а й ще сформувати документації: від конструкторської специфікації до виробничих документів тощо [19, 22].

У кожній з перерахованих систем є як позитивні, так і негативні сторони. Саме тому жодна із цих систем не має можливості одержати найкращі показники розрахунків норм часу та заробітної плати на підприємстві. Таким чином, для ефективного розрахунку норм часу та заробітної плати, з'являється необхідність створення програмного продукту, який би наступним вимогам:

- мати простий і зручний для користувача стандартний інтерфейс до програм, розроблених для Windows XP;
- забезпечувати можливість роботи з файлами вихідних даних;
- виконувати необхідні розрахунки;
- імпортувати та експортувати результати розрахунків до 1С системи.

Програмний модуль розрахунку норм часу повинен здійснювати швидкий збір та обробку інформації на підприємстві, зменшувати часові та грошові витрати і розумові витрати на реалізацію процесів розрахунку, а так само даний програмний модуль повинен інтегруватися з існуючими ПЗ на підприємстві.

У програмному модулі розрахунку норм часу пропонується виділити наступні функціональні підсистеми:

- підсистема вивантаження, обробки та завантаження даних, яка призначена для реалізації процесів збору даних із систем джерел, приведення зазначених даних до виду, необхідного для наповнення підсистеми зберігання даних;
- підсистема зберігання даних, яка призначена для зберігання даних в структурах, націлених на прийняття рішень;
- підсистема формування та візуалізації звітності, яка призначена для формування бізнес-орієнтованих вітрин даних та звітності.

Висновки. В умовах машинобудівного виробництва широко використовуються різні методи обробки на токарних, фрезерних, свердлильних, шліфувальних і інших верстатах.

Норми праці у виробництві виконують важливі функції вимірювача продуктивності праці, заходи витрат праці та винагороди за працю. Будучи вимірником рівня продуктивності праці, норма витрат праці виконує функцію засобу, інструменту управління виробництвом. За допомогою норми оцінюється рівень продуктивності праці. Як міра витрат праці норма є основою розрахунку й обліку багатьох показників виробничо-господарської діяльності підприємства.

Нормування праці встановлює витрати необхідного часу на виготовлення даного предмета праці на даному підприємстві, тобто норму праці - конкретне вираження міри праці.

В даний час накопичений певний досвід застосування сучасних інформаційних технологій для вирішення завдань в області організації та нормування праці. Однак охоплення робіт автоматизованим нормуванням недостатньо високий: в середньому нормується з використанням ЕОМ лише 5-7% технологічної трудомісткості.

Список використаних джерел:

1. Гавриленко Е.А., Холодняк Ю.В., Антонова Г.В., Чаплинский А.П. Разработка алгоритма программного обеспечения для формирования обводов по заданным геометрическим условиям. Праці Таврійського державного

агротехнологічного університету імені Дмитра Моторного. – Мелітополь: ТДАТУ ім. Д. Моторного, 2020. – Вип. 20, т. 3. (с.293-303).-DOI: 10.31388/2078-0877-2020-20-3-293-303.

2. Мацулевич О.Є., Щербина В.М., Бондаренко Л.Ю., Малюта С.І., Антонова Г.В. Програмне забезпечення для автоматизованого визначення параметрів різального інструменту фрезерної обробки корпусних деталей. *Праці Таврійського державного агротехнологічного університету*. Мелітополь: ТДАТУ, 2020. Вип. 20, т. 3. С. 275-281.

3. Інженерна та комп'ютерна графіка. Частина 1: навчальний посібник / В.М. Щербина, О.Є. Мацулевич, Є.А. Гавриленко, Ю.В. Холодняк, О.В. Івженко, І.В. Пихтєєва, О.О. Вершков, С.В. Галько, А.П. Чаплінський. Мелітополь: Люкс, 2020. 238 с.

4. Проектування керуючих програм для верстатів токарної групи з пристроєм числового програмного керування: навчальний посібник / Дмитрієв Ю.О., Мацулевич О.Є., Щербина В.М., Холодняк Ю.В. Мелітополь, Люкс, 2018. 132 с.

5. Теоретична механіка. Динаміка матеріальної точки: навчально – методичний посібник / О.М. Леженкін, Г.В. Антонова, О.О. Вершков, Л.Ю. Бондаренко, О.Є. Мацулевич, А.О. Смелов, О.Ю. Михайленко. Мелітополь: Люкс, 2021. 160с.

6. Інженерна механіка: практикум / Г.В. Антонова, О.Є. Мацулевич, О.Ю. Михайленко, І.В. Пихтєєва, О.В. Івженко, Ю.В. Холодняк, В.М. Щербина, Ю.О. Дмитрієв. Мелітополь: Люкс, 2021. 147 с.

7. Нарисна геометрія та креслення: навчально–методичний посібник для підготовки бакалаврів зі спеціальностей 131 “Прикладна механіка” та 133 “Галузеве машинобудування” / Івженко О.В., Пихтєєва І.В., Гавриленко Є.А., Мацулевич О.Є., Щербина В.М., Холодняк Ю.В., Бондаренко Л.Ю., Михайленко О.Ю. Мелітополь: Люкс, 2020. 217 с.

8. Програмування автоматизованих процесів обробки деталей: навчально–методичний посібник. Лабораторний практикум /Ю.О. Дмитрієв, О.Є. Мацулевич, Є.А. Гавриленко, Ю.В. Холодняк, Г.В. Антонова – Мелітополь (протокол №8 від 28.04.2022р.), 2022.–170с.

9. Мацулевич О.Є., Вершков О.О., Холодняк Ю.В., Дмитрієв Ю.О., Чаплінський А.П. Розробка мурашиного алгоритму для оптимізації оперативного планування робіт по збиранню врожаю кісточкових. *Плодовий сад – новітнє в теорії та практиці*: матеріали V Всеукр. наук.-практ. інтернет-конференції. Мелітополь: ТДАТУ, 2021. С. 106-110.

10. Холодняк Ю.В., Гавриленко Е.А., Івженко А.В., Найдыш А.В. Моделирование участка пространственной монотонной кривой линии /Сучасні проблеми моделювання: наукове фахове видання. – Мелітополь: МДПУ ім. Б. Хмельницького, 2020. Вып.17. С. 131-137.

11. Вершков О.О., Мацулевич Ю.О. Визначення шорсткості поверхонь із застосуванням програмного забезпечення SOLIDWORKS ф. DELCAM plc // Сучасні комп'ютерні та інформаційні системи і технології, Матеріали і всеукраїнської науково-практичної інтернет-конференції, Мелітополь 7-25 грудня 2020р. С. 17-23

12. Спірінцев В.В., Мацулевич О.Є., Холодняк Ю.В., Чаплінський А.П. Застосування графічного редактора ARCHICAD при вивченні дисципліни «Комп'ютерне проектування простору інженерних споруд». *Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації*: матеріали Міжнар. наук.-практ. інтернет-конф. Мелітополь: ТДАТУ, 2020. С. 262-266.

УДК 004.77

РОЗРОБКА УЗАГАЛЬНЕНОГО ПРОЕКТУ КОМП'ЮТЕРНОЇ МЕРЕЖІ

Білявський Н.А., магістрант

e-mail: kim221_bna@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Комп'ютерні мережі (також відомі під назвою «обчислювальні мережі») являють собою систему, яка складається як з машин, які розташовані в одній будівлі або приміщенні, так і з комп'ютерів, віддалених на достатній відстані. Між собою, всі комп'ютери можуть бути пов'язані бездротовим способом, або при використанні кабелів. На сьогоднішній день ефективність створення мережі компанії ні в кого не викликає сумніву, оскільки побудова локальної мережі супроводжується рядом переваг:

- спільний доступ до всіх корпоративних файлів;
- загальний доступ до використання дорогої оргтехніки (факси, сканери, принтери і т.п.);
- ефективне і швидке додавання / переміщення обладнання і нових робочих місць;
- гарантія додаткової безпеки корпоративних даних конфіденційного характеру і т.д.

Основні матеріали дослідження. Для виконання даного завдання була обрана трирівнева ієрархічна модель мереж, яка визначає підхід до проектування мереж і включає в себе три логічних рівня: рівень доступу (access layer); рівень розподілу / агрегації (distribution layer); рівень ядра (core layer).

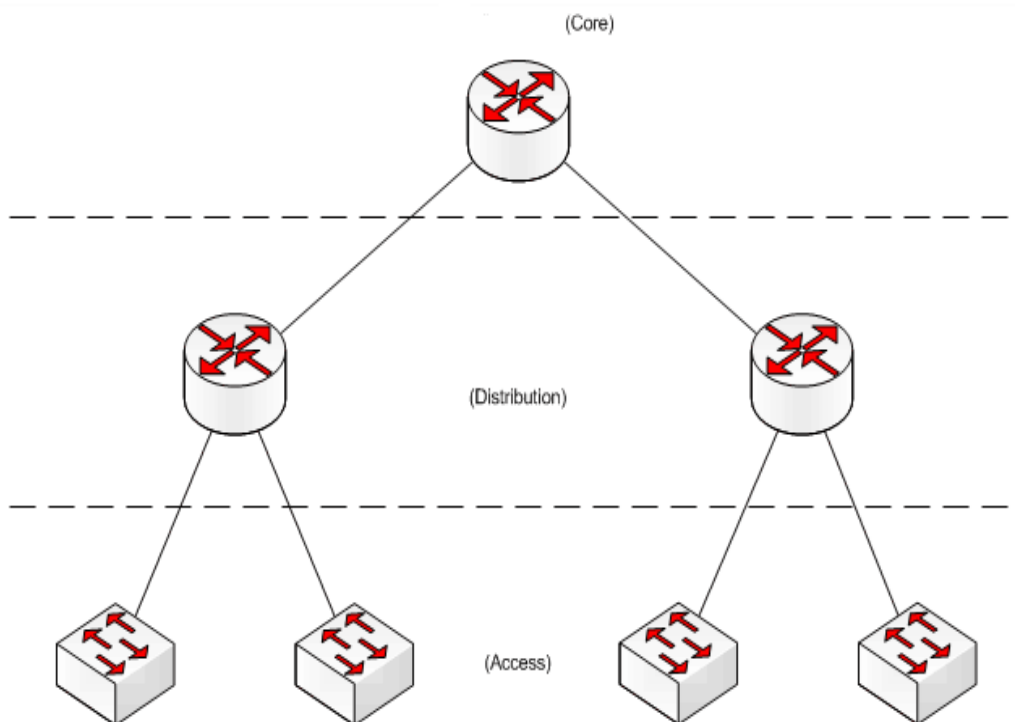


Рисунок 1. Трирівнева ієрархічна модель мережі

Проект мережі показано на рис. 2. Дана мережа являє собою об'єднання двох корпоративних мереж після об'єднання самих компаній. Для корелювання використання різних методів маршрутизації буде використана редистрибуція маршрутів.

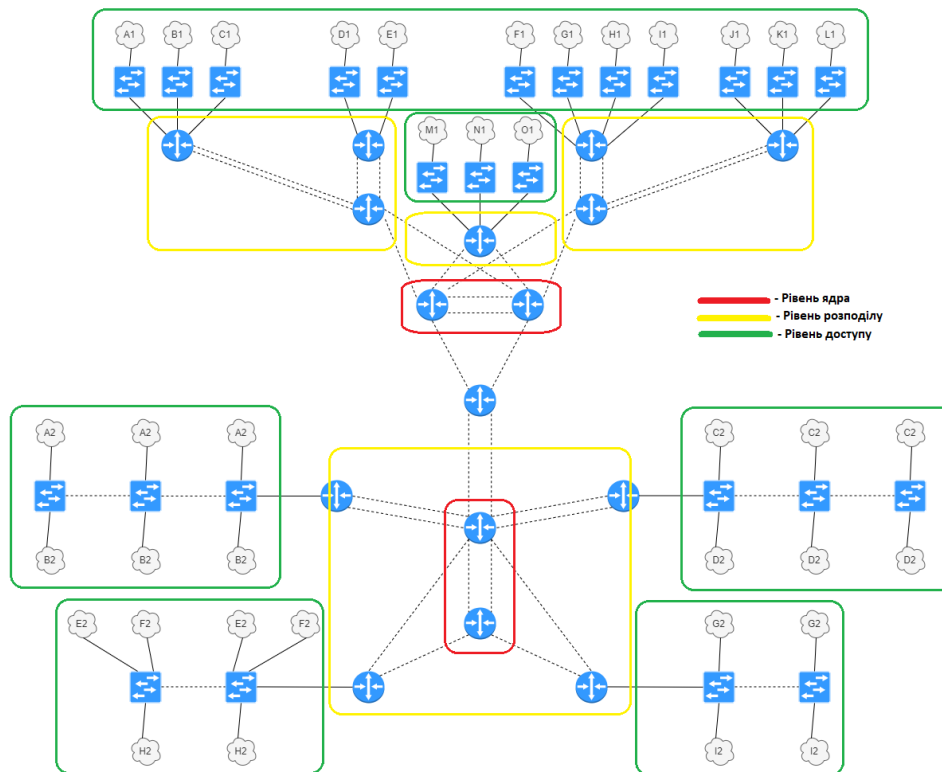


Рисунок 2. Проект корпоративної мережі

Розглянемо кожну з мереж окремо:

1) Мережа №1 (тип маршрутизації - плаваюча статична маршрутизація, тип адресації - статична):

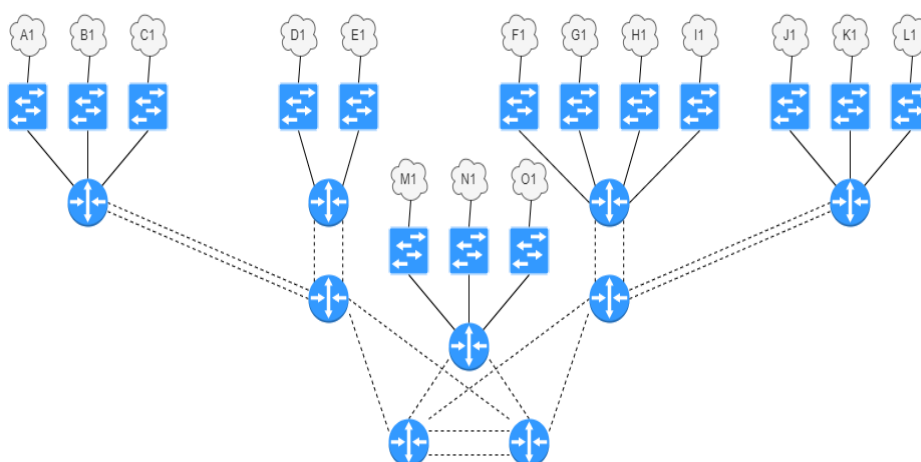


Рисунок 3. Проект мережі №1

Використання обладнання на різних рівнях:

- рівень ядра: 2 маршрутизатора з'єднані між собою за допомогою витой пари Gigabit Ethernet;
- рівень розподілу: 7 маршрутизаторів які з'єднані з ядром мережі та між собою за допомогою витой пари Gigabit Ethernet;
- рівень доступу: 15 комутаторів які з'єднані з рівнем розподілу та кінцевими пристроями за допомогою витой пари Ethernet.

Розглянемо окремо частини даної мережі:

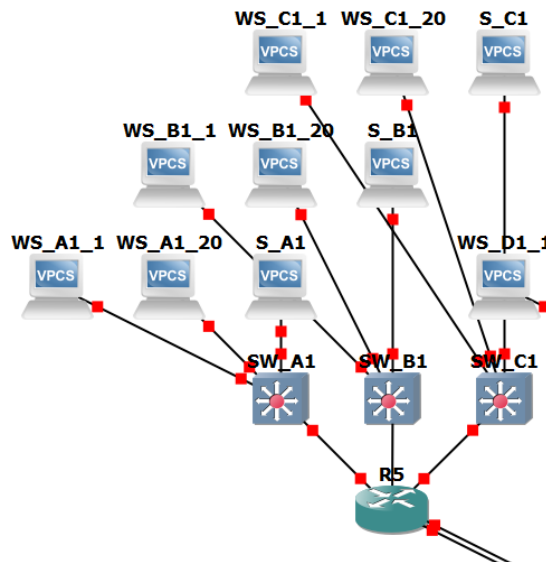


Рисунок 4. Підмережі A1, B1, та C1

Дана група підмереж складається з комутаторів з'єднаних з загальним маршрутизатором. До кожного комутатора також під'дана певна кількість робочих станцій та серверів, що виконують роль кінцевих пристроїв та здійснюють інформаційний обмін як з сусідніми пристроями так і з пристроями інших мереж.

Подібно з даним прикладом побудовані інші підмережі даної мережі.

2) Мережа №2 (тип маршрутизації – протокол динамічної маршрутизації OSPF, тип адресації - статична)

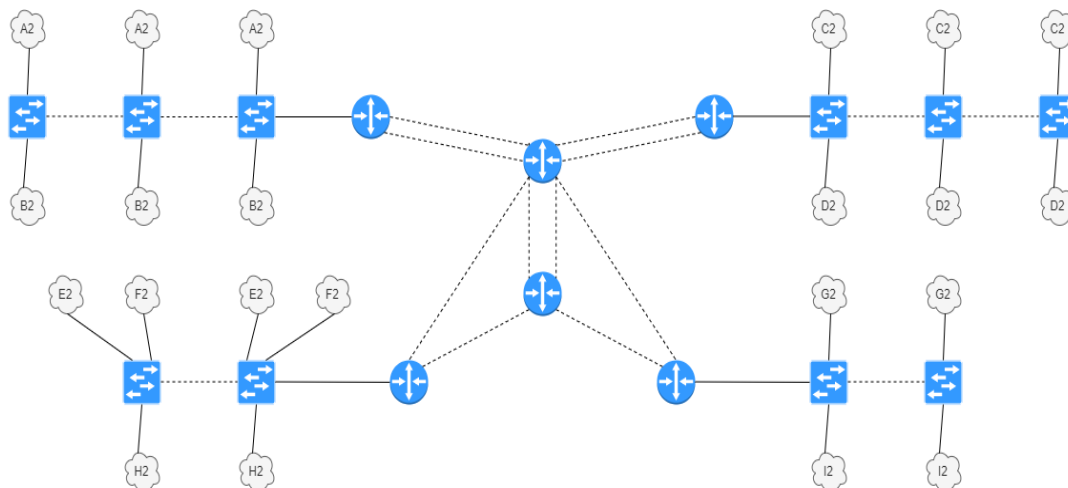


Рисунок 5. Проект мережі №2

Використання обладнання на різних рівнях:

- рівень ядра: 2 маршрутизатора з'єднані між собою за допомогою витої пари Gigabit Ethernet;
- рівень розподілу: 4 маршрутизаторів які з'єднані з ядром мережі за допомогою витої пари Gigabit Ethernet;
- рівень доступу: 10 комутаторів які з'єднані з рівнем розподілу та кінцевими пристроями за допомогою витої пари Ethernet.

Розглянемо окремо частини даної мережі:

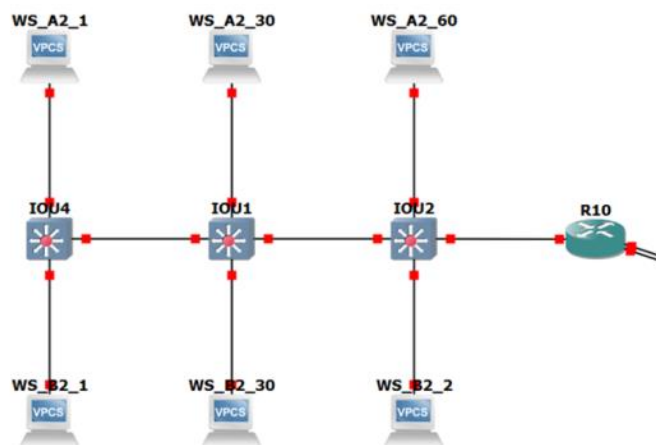


Рисунок 6. Підмережі A2 та B2

Дана група підмереж складається з комутаторів з'єднаних між собою та з загальним маршрутизатором. До кожного комутатора також під'єднана певна кількість робочих станцій та серверів, що виконують роль кінцевих пристроїв та здійснюють інформаційний обмін як з сусідніми пристроями так і з пристроями інших мереж. Дана підмережа побудована на принципі транкування каналів.

Висновок. Після виконання досліджень щодо побудови локальних мереж було вирішено, що при побудові корпоративної мережі використовуватиметься об'єднання двох локальних підмереж з трирівневими ієрархічними моделями. Для однієї частини мережі буде налаштовано статичну маршрутизацію, а для іншої частини динамічну. Дві підмережі будуть об'єднані в одну мережу за допомогою реалізації редистрибуції.

Список використаних джерел:

1. Налагодження та дослідження роботи протоколу маршрутизації ospf. [Електронний ресурс]. – Режим доступу: https://learn.ztu.edu.ua/pluginfile.php/156823/mod_resource/content/1/KM_KI_LR-23-2020-2021_KI-4.pdf
2. Налагодження та дослідження редистрибуції маршрутів у мережі. [Електронний ресурс]. – Режим доступу: https://learn.ztu.edu.ua/pluginfile.php/160433/mod_resource/content/1/KM_KI_LR-27-2020-2021_KI-4.pdf
3. Одом Уэнделл. Официальное руководство Cisco по подготовке к сертификационным экзаменам CCENT/CCNA ICNDI 100-101, акад. изд.: Пер. с англ. – М.: 000 "И.Д. Вильямс", 2015. – 912 с.

УДК 004.77

РОЗРОБКА ПРОЕКТУ КОМП'ЮТЕРНОЇ МЕРЕЖІ ТА ПІДСИСТЕМИ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ

Русятинська А.О., магістрантка
Державний університет «Житомирська політехніка»

e-mail: kim221_rao@student.ztu.edu.ua

Актуальність та постановка проблеми. Будь-яке проектування, як відомо являє собою моделювання з певним ступенем спрощення. Саме тому, щоб передбачити всі можливі фактори, врахувати всі потреби, які можуть виникнути в майбутньому, виконується попередній розрахунок необхідних параметрів та проектування відповідно них.

При проектування мережі необхідно враховувати:

- необхідний розмір мережі;
- необхідну структуру, ієрархію і основні частини;
- основні напрями та інтенсивність використання;
- вимоги до програмних засобів.

Окрім цього для кожної мережі важливо виконати коректний розподіл прав доступу, адже від цього залежить безпека всієї системи для цього варто розглянути аутентифікацію користувачів в мережі.

Основні матеріали дослідження. Під структурою мережі розуміється спосіб поділу мережі на частини, а також спосіб сполучення цих сегментів між собою. Мережа може містити робочі групи комп'ютерів, мережі підрозділів, опорні мережі, засоби зв'язку з іншими мережами. В ідеалі структура мережі має відповідати структурі будинку або комплексу будинків.

Під час проектування та вибору топології власної мережі було використано трирівнева ієрархічна модель мережі, що складається з рівня ядра, рівня розподілу та рівня доступу. Модель даної ієрархії відображена на рис. 1.

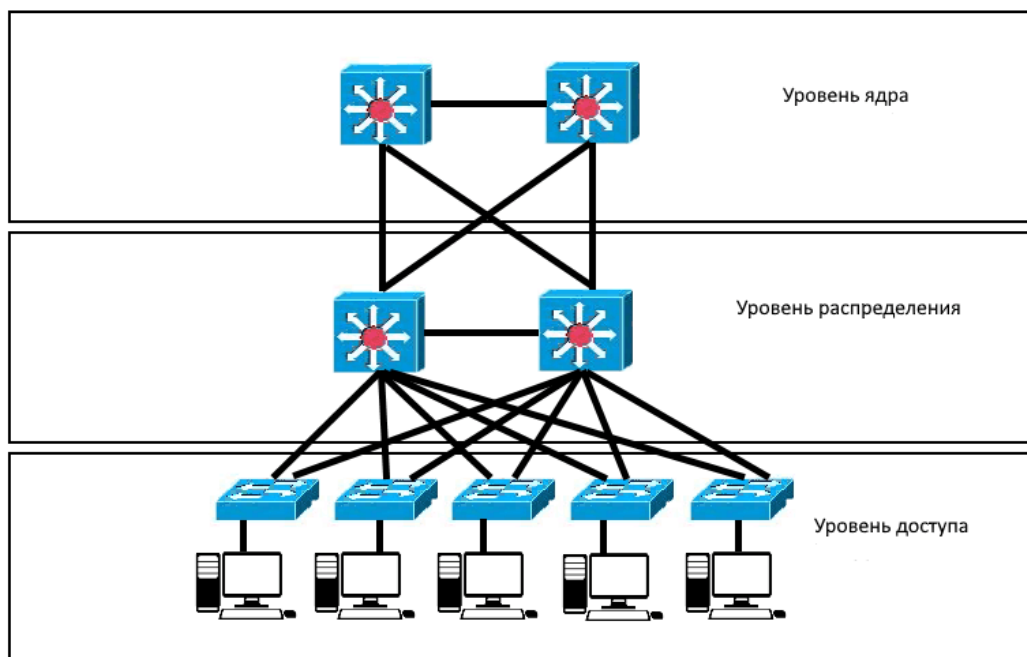


Рисунок 1. Трирівнева ієрархічна модель мережі

Відповідно до рис. 1 була створена власна модель мережі, яка відображена на рис. 2. Дана модель мережі містить в собі дві підмережі двох компаній, в першій використовується статична маршрутизація (плаваюча), в іншій динамічна (EIGRP), також схема має два ядра, які з'єднані між собою, та налаштовані за допомогою редистрибуції.

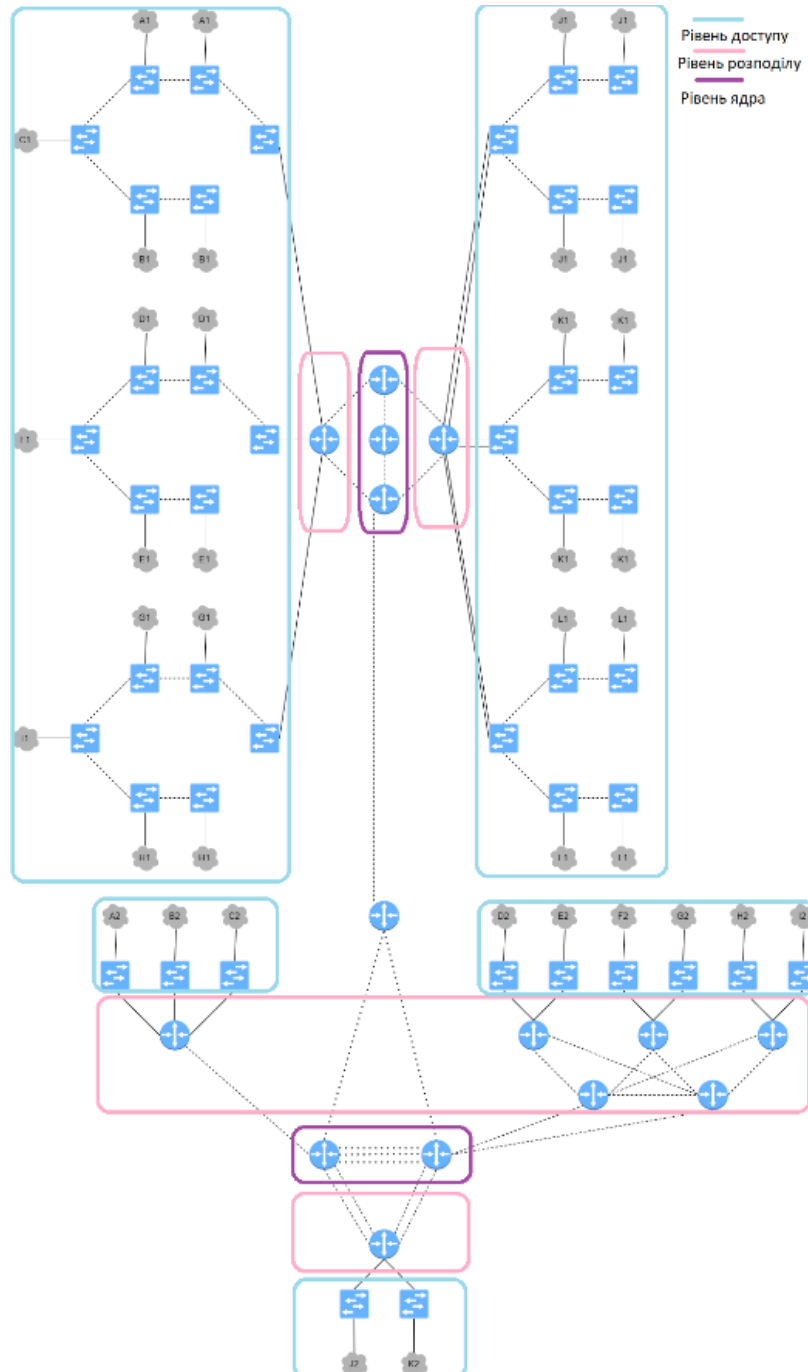


Рисунок 2. Проект корпоративної мережі

Розпочнемо розбір корпоративної мережі по частинам.

Перша частина мережі складається з 5 маршрутизаторів, 3 з яких складають рівень ядра, а 2 – рівень розподілу. Взагалом на рівні доступу використовується 36 комутаторів, а також 12 окремих мереж. Маршрутизатори з'єднані між собою

за допомогою витої пари Gigabit Ethernet. Всі решта елементів мережі з'єднані між собою за допомогою витої пари Ethernet.

Друга частина мережі складається з 9 маршрутизаторів, 2 з яких складаються рівень ядра, а 7 – рівень розподілу. Взагалом на рівні доступу використовується 11 комутаторів, а також 11 окремих мереж. Маршрутизатори з'єднані між собою за допомогою витої пари Gigabit Ethernet. Всі решта елементів мережі з'єднані між собою за допомогою витої пари Ethernet.

Отже, після побудови та розрахунку відповідної кількості елементів та моделей мережі, можна переходити до побудови системи аутентифікації мережі.

Саме для розмежування буде створено поділ всіх користувачів на основні групи, а також призначення відповідних прав для кожної групи. Розмежування буде відбуватися за таким принципом:

- Головний адміністратор
- Група адміністраторів
- Група менеджерів
- Група користувачів

Для кожної з наведених груп відповідно будуть розмежовані права доступу. При розподілі прав враховувалось необхідність доступу певним групам осіб до пристроїв мережі. Контролюватись ці дії будуть за допомогою виконання моделі AAA, а саме частини Authorization та Authentication.

Окрім цього, також буде проведено налаштування Accounting, тобто обліку входу та виходу користувача, а також виконання обліку всіх команд, що використовує кожний користувач. Дані налаштування можна виконати застосувавши певні команди.

1) Налаштування сервера TACACS.

```
aaa authentication-server tacacs <ім'я_сервера>  
хост <IP_адреса>  
ключ <спільний_секрет>
```

2) Налаштування групи для серверів TACACS.

```
aaa server-group <ім'я_групи_сервера>  
auth-server <name_of_tacacs_server>
```

3) Увімкнення обліку TACACS.

```
aaa accounting command <priv-lvl> start-end default group tacacs+
```

В результаті даних налаштувань буде виконуватися облік в log file.

Для забезпечення доступу до пристроїв буде налаштований ssh – мережевий протокол прикладного рівня, що дозволяє виконувати віддалене управління операційними системами, а також пристроями. На відміну від telnet шифрує весь трафік, включаючи і паролі, які передаються.

Висновок. В результаті проведеного дослідження вдалося обрати топологію мережі, розглянути основні вузли та побудувати мережу в цілому. Після цього було проведено розподіл підмереж на основі цих даних. Також розроблено групи користувачів та розподілені відповідні права між ними, наведені основні команди для налаштування обліку всіх дій користувачів на пристроях.

Список використаних джерел:

1. Basic Route Redistribution. Cisco Academy Conference. [Електронний ресурс]. – Режим доступу: https://www.cisco.com/c/dam/en_us/training-events/le21/le34/downloads/689/academy/2005/BRK-105.pdf

2. Комп'ютерні мережі : навчальний посібник / [Азаров О. Д., Захарченко С. М., Кадук О. В. та ін.] — Вінниця : ВНТУ, 2013. — 371 с

УДК-004.8

ВИКОРИСТАННЯ LAMP ДЛЯ РОЗРОБКИ ВЕБ-ЗАСТОСУНКІВ

Вакалюк Т.А., д. пед.н, професор

e-mail: kik_vta@ztu.edu.ua

Кузьмук В.О., магістрант

e-mail: ipzm221_kvo@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Кожний програміст, який стикається з проблемою створення бекенд частини проекту, або серверної частини розуміє, що йому доведеться використовувати певний стек технологій. Але не всі з них є безкоштовними, легкими для освоєння та мають гарну підтримку. Стеком, що має усі ці якості одночасно, являється LAMP (Linux, Apache, MySQL, PHP).

Основні матеріали дослідження. Стек LAMP – це набір із чотирьох різних програмних технологій, які розробники використовують для створення веб-сайтів та веб-додатків. LAMP – це абревіатура від операційної системи Linux, веб-сервера Apache, сервера баз даних MySQL та мови програмування PHP. Всі чотири з цих технологій мають відкритий вихідний код, що означає, що вони підтримуються спільнотою і доступні для використання всіма. Розробники використовують стеки LAMP для створення, розміщення та підтримки веб-контенту. Це популярне рішення, яке підтримує багато веб-сайтів, які зазвичай використовуються сьогодні.

Linux – це операційна система з відкритим вихідним кодом, яку можна встановити та налаштувати відповідно до різних вимог додатків. Linux знаходиться на першому рівні стека LAMP та підтримує інші компоненти на верхніх рівнях.

Apache – це веб-сервер з відкритим вихідним кодом, який утворює другий рівень стека LAMP. Модуль Apache зберігає файли веб-сайту та обмінюється інформацією з браузером за допомогою HTTP, інтернет-протоколу для передачі інформації про веб-сайт у вигляді звичайного тексту.

MySQL – це система управління реляційними базами даних з відкритим вихідним кодом та третій рівень стеку LAMP. Модель LAMP використовує MySQL для зберігання та запиту інформації в реляційних базах даних, а також керування нею.

PHP – файл, який розшифровується як препроцесор гіпертексту, є четвертим та останнім шаром стека LAMP. Це мова сценаріїв, яка дозволяє веб-сайтам запускати динамічні процеси. Динамічний процес включає інформацію у програмному забезпеченні, яке постійно змінюється. Веб-розробники вбудовують мову програмування PHP у HTML, щоб відображати на веб-сайтах інформацію в реальному часі або оновлену інформацію. Вони використовують PHP, щоб веб-сервер, база даних та операційна система могли узгоджено обробляти запити від браузерів.

Веб-розробники вибирають стек LAMP для створення веб-додатків з таких причин:

- Вартість. Усі технології LAMP мають відкритий вихідний код, що означає, що будь-який розробник чи компанія можуть використовувати їх без сплати ліцензійних зборів. Замість купівлі пропрієтарних стекових компонентів користувач може безкоштовно

завантажити операційну систему, веб-сервер, базу даних та мову сценаріїв. Це знижує вартість створення веб-застосунків.

- Ефективність. Налаштування нового стека веб-розробки потребує ретельного тестування різних платформ, модулів, бібліотек та інструментів. З іншого боку, стек LAMP – це перевірене рішення для веб-розробки. Веб-розробники можуть розставляти пріоритети та прискорювати розробку додатків, щоб зосередитись на тому, що вони створюють, а не на тому, як вони це створюють.
- Обслуговування. Експерти з програмного забезпечення з усього світу роблять свій внесок у розвиток стічних технологій LAMP, змінюючи, коментуючи та аналізуючи загальнодоступні вихідні коди. Вони регулярно підтримують та оновлюють технології, щоб вони залишалися актуальними та безпечними.
- Підтримка. Популярні технології з відкритим вихідним кодом, такі як стеки LAMP, мають підтримку великої світової ІТ-спільноти. Таким чином, користувачам стек LAMP легше знаходити інформацію на публічних ІТ-форумах. Веб-розробники можуть звернутися до прикладів коду або використовувати протестовані плагіни, створені спільнотою розробників відкритого вихідного коду.
- Гнучкість. Стек LAMP забезпечує веб-розробникам надійність та гнучкість. Хоча архітектура LAMP визначає програмні компоненти для кожного рівня, розробники можуть замінювати їх на власний розсуд. Наприклад, вони можуть використовувати іншу операційну систему крім Linux як основу стека.

Висновок. Отже, LAMP це стек технологій для розробки серверної частини проекту, що розповсюджується безкоштовно, регулярно підтримується та оновлюється, рішення багатьох проблем що з'являються під час розробки проекту можна знайти на спеціальних форумах, має високу надійність та гнучкість. Саме тому LAMP використовує велика кількість веб-розробників.

Список використаних джерел:

1. <https://aws.amazon.com/what-is/lamp-stack/>

УДК 005.95

ХАРАКТЕРИСТИКА ГНУЧКОГО (AGILE) ПІДХОДУ УПРАВЛІННЯ ІТ ПРОЕКТАМИ

Ковальчук О.А., магістр
Державний університет «Житомирська політехніка»

e-mail: uitpm221_koa@student.ztu.edu.ua

Актуальність та постановка проблеми. Сьогодні дедалі більше компаній переходять на проєктні методи роботи. По суті будь-яке велике робоче завдання є проєктом, оскільки має мету, строки, а на його проведення виділяються ресурси. Проєктом потрібно вміло керувати. Управління проєктами, або проєкт-менеджмент, активно застосовують у своїй роботі ІТ-компанії, креативні та вебагенції, міжнародні корпорації і навіть державні інституції.

Agile сьогодні надзвичайно популярний метод управління проєктами. Це досить гнучка система управління, характерними ознаками якої є надання кінцевого продукту на кожному етапі роботи та незрозумілий фінал проєкту.

Основні матеріали дослідження. Основний принцип роботи – поділ проєкту на короткі цикли (ітерації), наприкінці кожного з яких замовник отримує певний продукт.

У agile широка сфера застосування, однак найкраще він прижився в ІТ.

Розглянемо основні переваги Agile:

- Легка адаптація до змін.

Agile підходить для проєктів із важко визначеними параметрами або ймовірністю зміни вимог. Це знижує ризики проєкту та підвищує здатність вашої команди надавати найкращий продукт або послугу відповідно до потреб клієнта.

Зміни до вимог можуть бути внесені навіть на пізній стадії розробки. Гнучкі процеси використовують зміни для конкурентної переваги клієнта.

Повна видимість прогресу кожного проєкту в режимі реального часу.

Ще однією перевагою використання гнучкого підходу є прозорість кожного проєкту завдяки частому обміну інформацією з клієнтами. Це дозволяє їм відчувати себе більш залученими та вимагати змін у проєкті.

Крім того, залучені команди можуть показати клієнту свій прогрес разом із перешкодами, з якими вони зіткнулися.

Це встановлює відносини довіри та співпраці між командою та клієнтом і може призвести до покращення задоволеності клієнтів і підвищення цінності бізнесу.

Клієнти задоволені, оскільки після кожної ітерації їм надається робоча функція програмного забезпечення.

Крім того, підхід Agile може покращити організаційну синергію шляхом руйнування організаційних бар'єрів і розвитку духу довіри та партнерства навколо організаційних цілей.

- Залучення зацікавлених сторін.

Ключовою частиною використання гнучкого методу є залучення зацікавлених сторін до завершення проєктів.

Співпрацюючи з різними зацікавленими сторонами на кожному етапі проєкту, ми побудуємо динамічну систему, засновану на довірі та впевненості кожного члена команди, і налагодите міцніші стосунки у своїх командах.

Для ефективного використання цього методу рекомендується, щоб зацікавлені сторони брали активну участь у процесі просування проекту. Це дозволить їм переконатися, що завдання виконуються згідно з планом, і за необхідності внести зміни.

- Контроль витрат.

Гнучкий метод також можна використовувати для покращення контролю над витратами. Після кожного етапу команда переглядає бюджет для прийняття майбутніх рішень. Потім вони вирішують, чи будуть вони продовжувати, призупиняти чи скасовувати завдання чи навіть сам проект.

Це важлива частина управління проектом, оскільки дозволяє командам легко зрозуміти вартість кожної функції, що потім буде враховано під час прийняття стратегічних рішень.

- Покращена якість.

Використовуючи гнучку методологію, команди можуть розбивати проекти на ітерації та співпрацювати одна з одною, щоб отримати високоякісні результати.

Крім того, для кожного завдання існує етап тестування, який дозволяє командам швидко виявляти та вирішувати проблеми, щоб уникнути будь-яких довгострокових негативних наслідків. Також, якщо клієнти мають будь-які відгуки або будь-які зміни у функції, їх можна врахувати в поточному етапі продукту.

Через регулярні проміжки часу команда розмірковує над тим, як стати більш ефективною, а потім відповідно налаштовує та коригує свою поведінку. Команда викликає та надає зворотній зв'язок, сприймає відгуки та коригує, де це необхідно.

Незважаючи на те, що використання гнучкої методології має багато переваг, є також кілька **недоліків**, які слід враховувати перед використанням:

- У методології Agile вимоги не дуже чіткі, тому важко передбачити очікуваний результат. Також на початку проекту може бути важко передбачити такі складові, як вартість, час і ресурси;

- Важко виміряти прогрес, оскільки гнучкі методи забезпечують поступовий результат;

- Проект може бути важко реалізувати, оскільки люди природно опираються змінам;

- **Більше часу та зобов'язань;**

- Спілкування та співпраця - це чудово, але ця постійна взаємодія потребує більше часу та енергії для всіх учасників.

Підхід Agile також може потребувати певного рівня організаційної трансформації, щоб зробити його успішним. Він вимагає від бізнес-користувачів співпраці з командою розробників у дусі довіри та партнерства. Це може вимагати подолання деяких організаційних бар'єрів, які ускладнюють або роблять неможливим це зробити.

- **Підвищені вимоги до розробників і клієнтів.**

Щоб гнучка методологія була ефективною, необхідна відданість усіх учасників. Будь-хто, хто не бере участь, може негативно вплинути на якість проекту. Залежність результату

- **Відсутність необхідної документації.**

Оскільки завдання часто виконуються «саме вчасно» (Just in Time, JIT) для розробки за гнучким методом, документація, як правило, менш ретельна, що може призвести до непорозумінь і труднощів у майбутньому.

- Можливість регулярного спілкування з клієнтом можна розглядати не тільки як плюс, але і як мінус.

Таким чином, терміни здачі готового проекту можуть постійно переноситися, оскільки замовник, аналізуючи проміжні результати, може вимагати все більшого вдосконалення кінцевого проекту, сильно затягуючи робочий процес.

Висновок. Головна перевага Agile-управління – це гнучкість, гнучкість і ще раз гнучкість. Невеликі кроки і готовність до змін дозволяють швидко адаптуватися до динамічних умов. Ризики і витрати знижуються до мінімуму, так як більше не потрібно міняти стратегію підприємства. Як можна прослідкувати, що не дивний той факт, що зараз такий підхід набирає широких обертів та здебільшого популярний в ІТ-підприємствах.

Незважаючи на можливі недоліки, за умови правильного використання командою, яка бажає дотримуватися своїх принципів, гнучка методологія має численні переваги. Хоча методологія Agile може не ідеально підходити для кожного проекту та ситуації, її принципи можна вибірково адаптувати до будь-якого типу проекту, щоб допомогти команді залишатися на ногах і готовою до будь-яких змін.

Список використаних джерел:

1. <https://financesonline.com/efficient-project-management-approach/>
2. <https://www.professionaldevelopment.ie/what-is-agile-methodology>
3. <https://hbr.org/2021/11/agility-hacks>
4. <https://agilemanifesto.org/iso/uk/principles.html>
5. <https://ccaps.umn.edu/story/agile-methodology-advantages-and-disadvantages>
6. <https://managedagile.com/what-are-the-advantages-and-disadvantages-of-agile-scrum/>

Наукова публікація розроблена в рамках проекту Erasmus+ “Діджиталізація економіки як елемент сталого розвитку України та Таджикистану (DigEco) 618270-EPP-1-2020-1-LT-EPPKA2-CBHE-JP”/ The paper is developed in the framework of ERASMUS+ CBHE project “Digitalization of economic as an element of sustainable development of Ukraine and Tajikistan” / DigEco 618270-EPP-1-2020-1-LT-EPPKA2-CBHE-JP

Цей проект фінансується за підтримки Європейської Комісії. Цей документ відображає лише погляди автора, і Комісія не несе відповідальності за будь-яке використання інформації, що міститься в документі/This project has been funded with support from the European Commission. This document reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained there in.

УДК-004.8

ПЕРЕВАГИ ТА НЕДОЛІКИ ВИКОРИСТАННЯ МОВИ JAVASCRIPT

Вакалюк Т.А., д. пед.н., професор

e-mail: kik_vta@ztu.edu.ua

Кузьмук В.О., магістрант

e-mail: ipzm221_kvo@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. JavaScript, наразі, є однією з найпопулярніших мов програмування в інтернеті. В перші роки існування, більшість професійних програмістів скептично ставилися до мови, цільова аудиторія якої складалася з програмістів-аматорів. Поява AJAX змінила ситуацію та звернула увагу професійної спільноти до мови, а її подальші модифікації за стандартами ES6+ внесли багато корисних можливостей, яких не вистачало для ефективного програмування. В результаті, були розроблені та покращені багато практик використання JavaScript (зокрема, тестування та налагодження), створені бібліотеки та фреймворки, поширилося використання JavaScript поза браузером. Саме тому мова JavaScript є актуальною для вивчення.

Основні матеріали дослідження. JavaScript — це мова програмування, яка починалася просто як механізм додавання логіки та інтерактивності до статичного браузера Netscape. За роки, що минули після його появи, він не тільки витіснив низку інших конкуруючих мов і технологій, щоб стати стандартом для програмування на основі браузера, але й вийшов за межі клієнтського простору, щоб стати домінуючою мовою на стороні сервера.

Для чого можна використовувати JavaScript

Протокол транспортування гіпертексту, або HTTP, був розроблений, щоб виконувати саме те, що випливає з назви, тобто передавати HTML-текст через мережу для перегляду. Але проблема з HTML полягає в тому, що він повністю статичний, не надає можливості для реалізації логіки.

Основні функції, які ми наразі сприймаємо як належне, такі як перевірка введених даних, авто виправлення, індикатори прогресу або зміна графіки під час наведення курсора миші, усі вони були б неможливі для браузера без певного типу можливостей сценаріїв. У результаті, коли Інтернет вперше став популярним, кожен постачальник у цьому просторі надав певний механізм, щоб зробити веб-сторінки HTML більш інтерактивними.

Альтернативи JavaScript

Microsoft просувала конкурента JavaScript під назвою VBScript разом із технологіями плагінів, такими як ActiveX Data Object (ADO). Sun Microsystems просувала Java-аплети, які були кросплатформними і могли використовуватися в будь-якому браузері.

Багато інших технологій з'являлися та зникали разом із розвитком Інтернету. Зрештою, JavaScript виграв битву за популярність, і це єдина технологія сценаріїв на стороні клієнта, яку підтримують браузери Chrome, Firefox, Safari та Internet Explorer.

JavaScript є інтерпретованою мовою, на відміну від скомпільованих мов, таких як C++ і Java. Це означає, що код, написаний на JavaScript, не проходить через проміжну стадію компіляції, на якій вихідний код перетворюється на машинну мову, яку легко обробити ЦП.

Натомість JavaScript інтерпретується на льоту комп'ютером, який його обробляє. Оскільки код JavaScript не упакований у зручну для ЦП форму, як-от

байт-код Java або двійковий виконуваний файл, виконання програми може бути повільнішим, ніж порівнянна програма, написана на Scala або F#. Однак сучасні двигуни JavaScript дуже оптимізовані, і неефективність рідко помітна при використанні сучасного апаратного забезпечення.

Про фреймворки JavaScript

Велика галузева тенденція в розробці інтерфейсу користувача полягає в тому, щоб передати якомога більше логіки, обробки та управління станом клієнту та взаємодіяти з сервером через легкі RESTful API і веб-сервіси. Це призвело до розробки повномасштабних фреймворків JavaScript, які контролюють усі аспекти представлення на стороні клієнта, часто натискаючи на односторінковий інтерфейс (SPI).

Популярними фреймворками в цьому просторі є Angular, React і Ember, причому ці фреймворки часто використовують скомпільовані мови JavaScript, такі як TypeScript і бібліотеки JavaScript, такі як jQuery для логіки та Handlebars.js або Mustache.js для шаблонів веб-сторінок.

Висновок. Мова JavaScript це єдина технологія сценаріїв на стороні клієнта, яка призначена для перевірки даних які вводить користувач, авто-виправлення, індикаторів прогресу та зміни зображення під час наведення миші на об'єкт. Жодний з проектів не обходиться без використання цієї мови. Хоча ця мова є дуже старою, але разом з цим вона є найкращою для додавання інтерактивності сайту.

Список використаних джерел:

1. <https://www.theserverside.com/definition/JavaScript>

УДК 004.9

ПРОЕКТУВАННЯ І РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ

Величко С.Д., студентка

e-mail: lifesofi2016@gmail.com

Науковий керівник: ст. викладач Зінов'єва О.Г.

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. У світі, зокрема в Україні, великими темпами зростає кількість користувачів Internet, як наслідок, кількість "електронних" покупців і потенційних "електронних" покупців.

Вибрана тема вважається актуальною на сьогоднішній день, оскільки сьогодні мільйони людей щодня, не виходячи з дому, купують різні товари в електронних магазинах.

Ступінь технічного прогресу та інформатизації, а також грамотності населення в галузі мобільних та онлайн технологій збільшується постійно. В даний час присутність сайту для постачальника продуктів і пропозицій вважається практично обов'язковою умовою його фінансового становлення, тому що високоякісний веб-ресурс збільшує мобільність, доступність і впізнаваність постачальника на ринку продуктів і пропозицій.

Незважаючи на різноманітність онлайн-сервісів, одним із найактуальніших у сфері інтернет-торгівлі є інтернет-магазин. Інтернет-магазин дуже зручний тим, що він працює цілодобово і без вихідних, подивитися каталоги, придбати товар, може покупець з будь-якої точки світу де є доступ до інтернету, на відміну від звичайних магазинів, в яких площа приміщення обмежена, на сайтах можна розмістити скільки завгодно товару, реклами та інформації

Основні матеріали дослідження. Метою даної роботи є проектування та розробка сайту інтернет-магазину з продажу кавових напоїв

Інтернет-магазин – це магазин "вітрина", яка розташована в інтернеті, що дає можливість замовити товар через інтернет.

Вдалий Web-сайт - це надзвичайно ефективний інструмент торгівлі - він здатний захоплювати увагу аудиторії. Як і будь-який інший маркетинговий інструмент, заснований на принципі безпосереднього відгуку, перш за все він повинен заінтригувати відвідувача, а потім спонукати його на певні дії. Однак, багато хто ігнорує цю особливості головної сторінки, що часто призводить до того, що відвідувачі не затримуються на сайті надовго і залишають його, ледь зайшовши. Такі Web-сайти, нехай навіть містять іноді величезна кількість корисних порад та статей, практично ніколи не досягають передбачуваного рівня відвідуваності, не кажучи вже про продажі.

Головне завдання інтернет-магазину - здійснення продажу товарів клієнтам. Користувач, який бажає придбати той чи інший товар, додає його в кошик товарів, оплачує замовлення безготівковим способом за допомогою електронних терміналів. Важливо, щоб покупець міг швидко знайти та вивчити товар, порівняти та зробити замовлення.

Для того щоб більш точно зрозуміти, як повинна працювати система, скористаємося функціональною діаграмою IDEF0, яка відображає структуру і функції системи та потоки інформації, що зв'язують ці функції

Процес функціонування інтернет-магазину складається з контекстної діаграми і подальшої декомпозиції контекстної діаграми. Основна мета

контекстної діаграми – виявити головне завдання, єдину та неповторну функцію, яку вирішує виконання бізнес-процесу. На рисунку 1 представлена контекстна діаграма діяльності інтернет-магазину. Вхідними даними є дані про покупця. Результатом процесу є замовлений товар та статистика про продажі. Весь процес керується законом про захист прав споживачів та Податковим кодексом України.

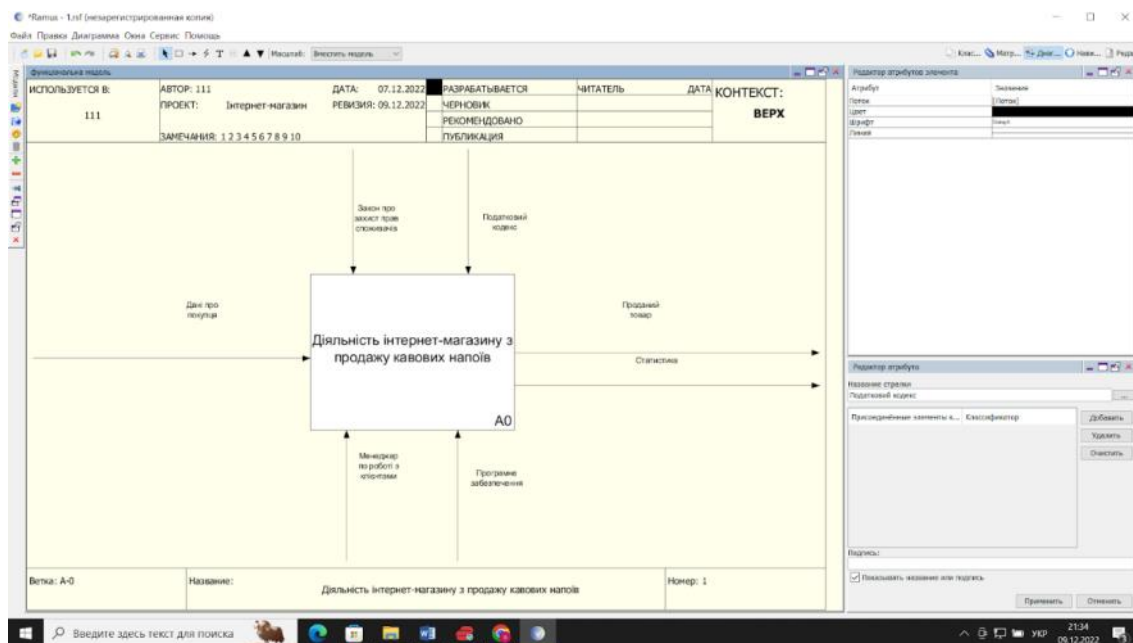


Рисунок 1. Контекстна діаграма «Діяльність інтернет-магазину з продажу кавових напоїв»

Далі процес декомпонується на 3 блоки: реєстрація та вхід на сайт, оформлення замовлення та оформлення продажу (рис. 2). В блок «Реєстрація та вхід на сайт» надходять дані про покупця, дані оброблюються, виконується реєстрація та вхід на сайт. Далі виконується оформлення замовлення. Після того, як покупець закінчив оформлення, йому надходить повідомлення, що підтверджує замовлення. Після цього виконується оформлення продажу, оплата і доставка товару.

Проектування сайту – це вдумливе моделювання ефективного онлайн-ресурсу. Це первинний етап у розробці сайту і саме він відповідає за адекватність подальшої розробки. Під час проектування сайту, розробляється ідея сайту, визначаються його цілі і завдання, а також способи їх досягнення. По завершенні етапу проектування, розробники мають чітке уявлення про те, як виглядатиме і працюватиме завершений проект.

В даний час перед web-розробником ставиться безліч різних завдань - від створення інтерактивних розважальних сайтів до серйозних бізнес-проектів, що вимагають підвищеної надійності та безпеки від несанкціонованого доступу. Для їх реалізації потрібні правильно підібрані інструменти – мови програмування, фреймворки або, що стають все більш актуальними, системи керування контентом (CMS – Content Management System).

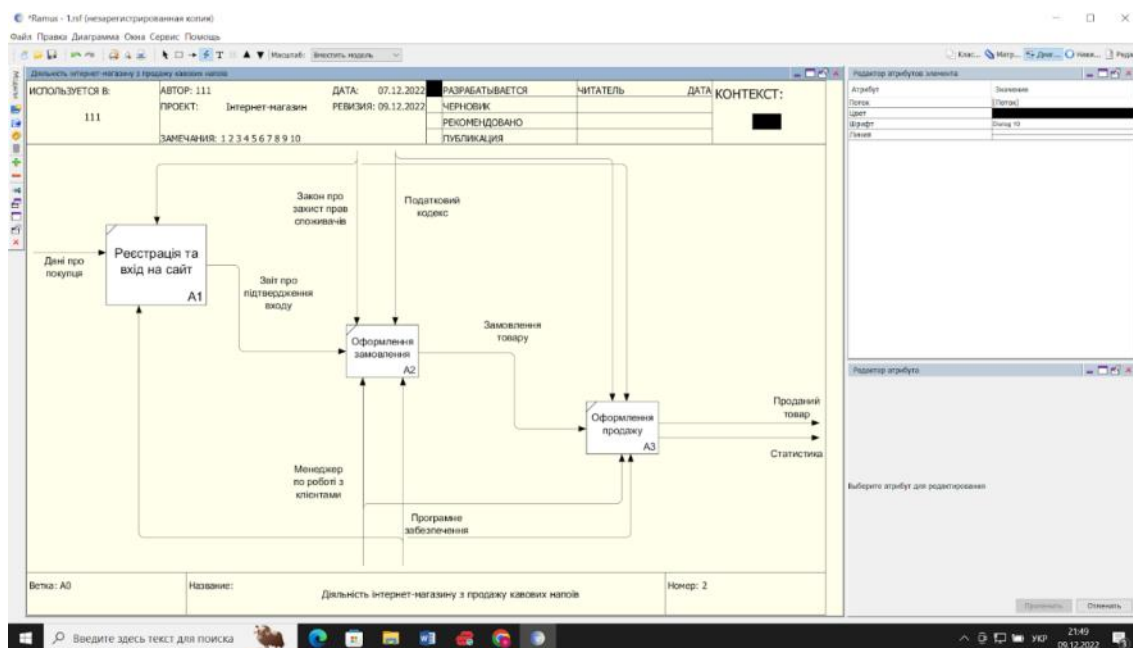


Рисунок 2. Декомпозиція контекстної діаграми

На даний момент існує багато мов програмування. Перевага кожного з них може виявлятися лише в контексті будь-якого завдання. Вибір мови або фреймворку визначається тим, які знання мають програмісти, готові реалізувати даний проект. Слід розмежовувати такі поняття, як мова програмування та фреймворк. Серед клієнтських мов web-програмування треба виділити JavaScript, який, як і HTML, лежить в основі багатьох web-технологій (наприклад, в основі популярної останнім часом технології AJAX) і вміння програмувати на ньому відноситься до базових знань web-розробника.

Для того, щоб створити гарний та функціональний сайт ми використали мову програмування html, css, php, js, sql

На етапі проектування сайту інтернет-магазину необхідно добре продумати структуру сайту. Ми створюємо сайти з особливими структурою. Зручна і зрозуміла структура сайту допомагає користувачеві легко знаходити потрібну йому інформацію. Структура інтернет-магазину складається з таких функціональних елементів:

- каталог товарів;
- користувальницький кошик;
- реєстраційна форма;
- форма відправлення замовлення.

На рисунку 3 наведена структура сайту інтернет-магазину

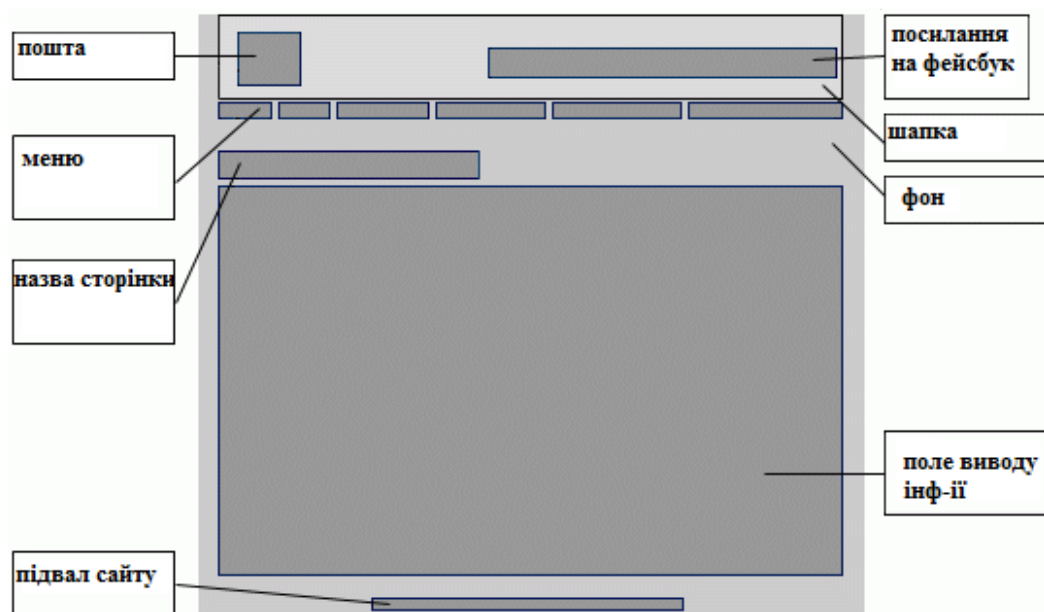


Рисунок 3. Структурні одиниці макету сторінок

У процесі розробки було створене адміністративне управління сайтом. Тобто, через ведення певного логіну та паролю, який призначений тільки для адміністратора, після авторизації адміністратор переходить одразу до адмін-панелі

Веб браузері взаємодіють з веб-серверами за допомогою гіпертекстового транспортного протоколу (HTTP). Коли ви натискаєте на посилання на веб-сторінці, заповнюєте форму або запускаєте пошук, HTTP запит відправляється з вашого браузера на цільовий сервер. Запит включає в себе URL, який визначає порушене ресурс, метод, який визначає необхідну дію (наприклад, отримати, видалити або опублікувати ресурс) і може включати додаткову інформацію, закодовану в параметрах URL (пари поле-значення, вставлені як рядок запиту), як POST запит (дані, відправлені в форматі HTTP POST), або в куки-файлах

Призначена для користувача (або клієнтська) частина (frontend) будується на html-розмітці, css-стилях і javascript. Html потрібен для відображення контенту сайту: тексти, заголовки, зображення, таблиці, текстові блоки, нумеровані і нелінійні списки. Css - це стильове оформлення контенту: колір і розмір шрифту, позиціонування елементів, відображення меж об'єктів, розміри блоків. JavaScript реалізує динамічне взаємодія з користувачем: перевірка введених даних, відображення діалогових вікон, додавання і приховування html-елементів. Серверна частина (backend) забезпечує формування html-коду, збереження призначених для користувача даних, взаємодія зі сторонніми web-сервісами.

На рисунку 4 наведена головна сторінка сайту.

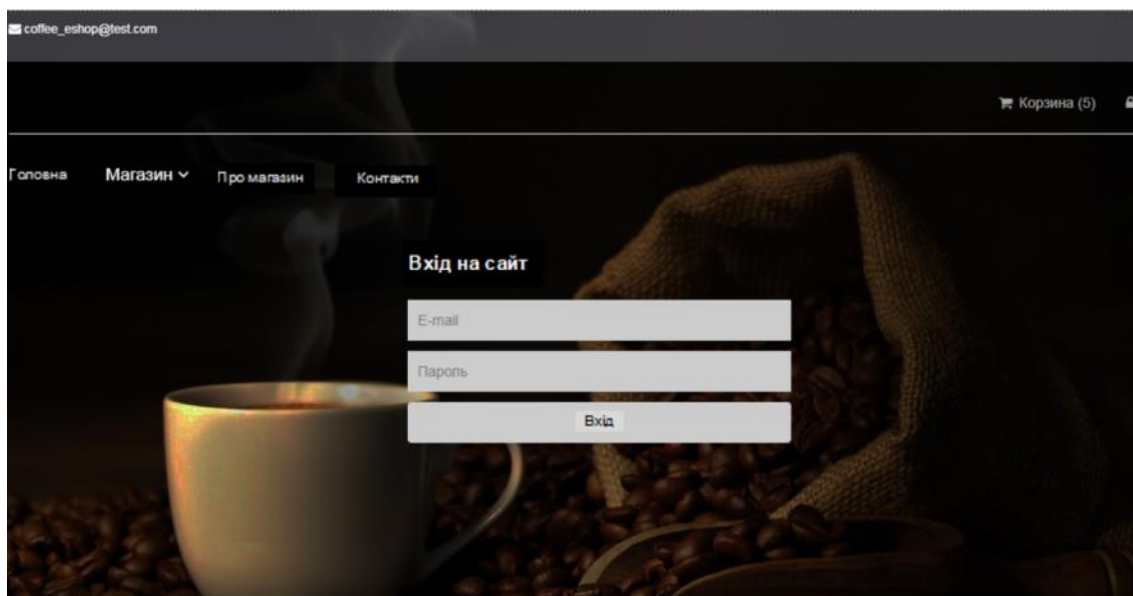


Рисунок 4. Сторінка авторизації користувача

Для розміщення сайту ми використали локальний сервер і програму XAMP localhost (так званий, «локальний хост», за змістом - цей комп'ютер) - в комп'ютерних мережах, стандартне, офіційно зарезервоване доменне ім'я для приватних IP-адрес (в діапазоні 127.0.0.1 - 127.255.255.255, RFC 2606)

Висновки. Інтернет-магазин з продажу кавових напоїв призначений для продажу товарів через Інтернет. Області застосування даного Web-додатка виступає інтернет-торгівля.

Створення даного інтернет-магазину дозволить просувати та продавати товари, зменшити витрати на рекламу та придбати додатковий ринок збуту свого товару, розширити клієнтську аудиторію та, як наслідок, збільшення прибутку

Розроблений інтернет-магазин містить в собі функціонал авторизації, реєстрації, адмін-панелі, виведення товарів та категорій. Також була розроблена база даних, в якій зберігається уся необхідна інформація для функціонування сайту.

Список використаних джерел:

1. Цеслів О.В. WEB-програмування: навч. посібник; М-во освіти і науки, молоді та спорту України, Нац. техн. ун-т України "Київ. політехн. ін-т". Київ : НТУУ "КПІ", 2011. 296, с. .
- 2.Пасічник В. В., Резніченко В.А. Організація баз даних та знань К.: Видавнича група BVH, 2006. – 384 с.
3. Бернерс-Лі Заснування павутини = Weaving the web. The original design and ultimate destiny of the world wide web: З чого починалася і до чого прийде Всесвітня мережа / Тім Бернерс-Лі разом з Марком Фічетті; пер. з англ. А. Іщенко. Київ : Києво-Могилянська академія, 2007. 208с.
4. Куленко М.Я. Основи графічного дизайну: підручник для студентів вищих навч. закладів / Михайло Куленко; МОНУ; Київський нац. ун-т будівництва і архітектури. – 2-ге вид., виправл. та доп. Київ : Кондор, 2007. 492с..
5. The Modern JavaScript Tutorial [Електронний ресурс] – Режим доступу: <https://javascript.info/>
6. Daniel Woolston. Pro Ajax and the .NET 2.0 Platform М.: Инфра-М, 2017. 143с.

УДК 004.42

РОЗРОБКА ВЕБЗАСТОСУНКУ МАПИ ВИЗНАЧНИХ МІСЦЬ З АУДІОВІДТВОРЕННЯМ ІНФОРМАЦІЇ

Перевалова А.Д., магістрантка

e-mail: zipzm221_pad@student.ztu.edu.ua

Чижмотря О.Г., ст. викл.

e-mail: ch-o-g@ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. У час стрімкого розвитку інтернету, інформаційних технологій та діджиталізації, люди прагнуть оптимізувати витрати своїх ресурсів. Також це стосується їхнього часу, одного з найважливіших ресурсів. Так більша перевага надається ресурсам, що задовольняють усі наші потреби та не вимагають багато часу. Які є зручними, містять достатню кількість інформації для нас.

Під час подорожі в нове місто або країну людина прагне дізнатися більше про історичну та культурну спадщину даної місцевості шляхом пошуку пам'яток та знакових місць. Перебуваючи в новому для себе місці, існує велика ймовірність натрапити на невідомі нам пам'ятки архітектури, історичні будинки або інші визначні об'єкти. Тому актуальною є потреба в ідентифікації цих місць, а також достатнього спектру інформації про такі місця.

Головою метою провадження вебзастосунку мапи визначних місць є: знаходження визначних об'єктів та інформації про них з як найменшою затратою ресурсів користувачем. Впроваджений вебзастосунок має бути корисним багатьом людям, що подорожують, відвідують нові місця, а також цікавляться історією та культурою рідної місцевості. Поєднання корисного функціоналу в одній системі матиме перевагу над аналогами.

Необхідність дослідження полягає в перспективі розробки вебзастосунку мапи визначних місць з аудіовідтворенням інформації, що може стати в нагоді людям з порушеннями зору. У ході дослідження було проаналізовано низку аналогів та вони найчастіше орієнтовані на іншу цільову аудиторію, а також пропонують відмінну кількість функціональних можливостей.

Основні матеріали дослідження. Переглянувши та проаналізувавши аналоги вебзастосунку з мапою визначних місць, можна стверджувати, що кожен програмний продукт відрізняється від своїх конкурентів, але в повному обсязі не покриває потреби користувача. Враховуючи дану інформацію виникає потреба в проектуванні та розробці вебзастосунку з мапою визначних місць, додавши можливість аудіовідтворення інформації. Також програмний продукт матиме більш вузьке призначення, що дасть змогу сфокусуватися на головних речах.

Отож, основними рисами даного вебзастосунку мають бути: наявність мапи, масштабування мапи, визначення місцезнаходження, можливість пошуку та фільтрації об'єктів, достатня кількість інформації про об'єкт, аудіовідтворення опису про визначне місце, система автентифікації користувачів, пропонування нового місця користувачем та робота з закладками. Результатом реалізації поставленого завдання буде вебзастосунок, що ним можна буде користуватися на комп'ютерах та менш габаритних пристроях.

Доцільно буде поділити користувачів на різні ролі. Відтак кожен користувач вебзастосунку може переглядати основну сторінку на якій розміщена мапа з об'єктами, шукати та фільтрувати їх, переглядати інформацію про визначні місця,

переглядати фото та відтворювати в аудіорежимі опис конкретного визначного місця. Також користувач вебзастосунку має змогу пройти реєстрацію в системі та підтвердити свою електронну адресу аби активувати обліковий запис. У такий спосіб цей користувач набуває роль авторизованого користувача після логування.

Наступний тип користувача – зареєстрований активний користувач, він має доступ до всіх функцій вебзастосунку. До них належать: додавання та видалення визначного місця зі списку закладок, робота зі списками закладок, редагування списку закладок, перегляд та редагування інформації власного профілю, перегляд сторінки налаштувань, пропонування визначного місця.

Натомість тип користувача «Адміністратор» отримує значно ширший доступ до вебзастосунку, що призначений лише для персоналу. Він може керувати вмістом сайту на сторінці адміністратора. У свою чергу можна також створити обмеження для певного адміністратора, а саме: до яких даних він матиме доступ та як зможе з ним взаємодіяти.

Також у цьому вебзастосунку продумана функція відтворення інформації про визначне місце. У такий спосіб користувач не повинен читати опис самостійно, а відбуватиметься його награвання, що покращить досвід користування цим продуктом.

Інтерфейс вебзастосунку з мапою визначних місць має бути зрозумілим та зручним. Мапа повинна займати найбільше простору на екрані для кращої орієнтації на ній. Приділяється також увага доступності вебзастосунку для людей з порушеннями зору.

Для розробки цього вебзастосунку використовувалася хмарна технологія Google Cloud Text-to-Speech. Google Cloud Text-to-Speech API є частиною інфраструктури Google Cloud. Він дозволяє синтезувати природне звучання мови з понад 100 голосами, доступними кількома мовами та варіантами. Він застосовує новаторські дослідження DeepMind у WaveNet і потужних нейронних мережах Google, щоб забезпечити найвищу точність [1].

Google Cloud Text-to-Speech API перетворює текст в аудіоформати, такі як: WAV, MP3 або Ogg Opus. Процес синтезу мовлення генерує необроблені аудіодані як рядок у кодуванні base64. Тому потрібно декодувати рядок у кодуванні base64 в аудіофайл, перш ніж програма зможе його відтворити. Він також підтримує вхідні дані Speech Synthesis Markup Language (SSML) для визначення пауз, чисел, форматування дати й часу та інших інструкцій щодо вимови. Окрім голосу, також можна налаштувати інші аспекти виведення аудіоданих, створених синтезом мовлення. Синтез мовлення підтримує налаштування швидкості розмови, висоти, гучності та частоти дискретизації в герцах.

Також варто зазначити, що Google Cloud Speech API один з небагатьох, хто пропонує перетворення тексту в мову саме українською мовою. Окрім того, надається можливість створити власну модель голосу, використовуючи власні аудіозаписи, щоб створити унікальний і більш природний голос для конкретної мети. Можна визначити та вибрати голосовий профіль, який найбільш підходить, і швидко адаптуватися до змін голосових потреб без необхідності записувати нові фрази.

Необхідно розглянути безпосереднє використання Google Cloud Text-to-Speech API у вебзастосунку. Насамперед користувач повинен знаходитися на головній сторінці та обрати об'єкт на мапі, інформацію про який би він хотів прослухати. Тоді відобразиться компонента, що являє собою бічну панель обраного визначного місця з інформацією про нього. У даній компоненті

знаходиться рядок з доступними діями, а першим елементом є кнопка з назвою «Прослухати». Після натискання на цю кнопку відбувається перевірка, чи наявний аудіофайл у даного визначного місця. Якщо він є, то відображається елемент audio та автоматично відтворюється аудіоінформація. Якщо ж такий файл відсутній, то відображається компонента text-to-speech, якій передається текст з описом місця. Після створення компоненти формуються дані та виконується запит до Google Cloud Text-to-Speech API, для проведення синтезації. Якщо дані перетворено успішно, то повертається аудіоконтент у текстовому форматі, який вставляється в HTML елемент audio, та також автоматично відтворюється. Якщо запит завершився помилкою, то користувачу відображається сповіщення про помилку.

Висновок. Отже, у ході роботи було спроектовано та розроблено вебзастосунок з пошуком визначних місць на мапі. Даний програмний продукт може використовуватися в повсякденному житті для пошуку визначних місць певного регіону. Вебзастосунок стане в нагоді людям, що часто подорожують та цікавляться культурною та історичною спадщиною відвідуваної місцевості.

Інструмент Text-to-Speech від платформи Google Cloud надає можливість синтезування тексту в мову, при використанні вебзастосунку з пошуком визначних місць на мапі, що має на меті зберегти час користувача та уможливити використання вебзастосунку людей з порушеннями зору.

Список використаних джерел:

1. Text-to-Speech [Електронний ресурс] – Режим доступу до ресурсу: <https://cloud.google.com/text-to-speech/>.

УДК 004.414.2; 004.415.25

АНАЛІЗ МЕТОДОЛОГІЙ ВИЗНАЧЕННЯ ЛОЯЛЬНОСТІ КЛІЄНТІВ ТА ОЦІНКИ РІВНЯ ЇХ ЗАДОВОЛЕНOSTІ

Лейба Я.А.

e-mail: yana.leiba.cpe@nure.ua

Широкопетлева М.С.

e-mail: marija.shirokopetleva@nure.ua

Харківський національний університет радіоелектроніки

Актуальність та постановка проблеми. В час стрімкого розвитку у сфері надання послуг, велике значення набуває клієнто-орієнтованість. Задоволеність клієнтів є одним із ключових елементів, що сприяють зростанню бізнесу. Задоволений клієнт вважається рушійною силою будь-якого бізнесу. Немає кращого маркетингу, ніж реферали та «сарафанне радіо» серед клієнтів.

Ціллю бізнесу стає не тільки факт надання послуг клієнтові, а також задоволеність самого клієнта (замовника) як якістю надання цих послуг, так і якістю обслуговування. Ці фактори напряду впливають на ймовірність повторного звернення клієнта до компанії та підвищують її конкурентоспроможність на ринку [1].

Задля відслідковування фактору задоволеності, використовуються багато методів, таких як опитування (як електронні, так і особисті), отримання зворотного зв'язку у вигляді фідбеків, надання знижок для залучення до опитувань. Всі ці методи діючі, але забирають багато часу у спеціалістів компанії, включаючи процес збору даних, їх обробку та аналіз, а клієнта такий процес зазвичай втомлює. Але, не зважаючи на дорогу вартість та великі затрати часу, компанії продовжують проводити дані заходи.

Таким чином, клієнто-орієнтованість для компаній все частіше виступає на рівні з якістю продукції, тому дана тема набуває актуальності для бізнесу.

Метою даної роботи є аналіз методів отримання даних з оцінювання клієнтами надання послуг компанією для подальшого впровадження при розробці програмної системи оцінки лояльності клієнтів.

Основні матеріали дослідження. Задоволеність клієнта визначається як рівень лояльності клієнта до компанії, який формується та визначається під час взаємодії його з командою компанії, її продуктами та послугами. Достатньо високий рівень лояльності обумовлюється високою якістю надання цих послуг та високим рівнем обслуговування. Таким чином, даний рівень впливає на те, чи скористається клієнт послугами компанії повторно [2].

Для підтримання рівня лояльності, необхідно постійно налагоджувати вже існуючі процеси компанії, удосконалювати їх та розглядати нові напрями роботи. Також, важливою частиною є відслідковування якості процесу роботи компанії, команди та продуктів. Не менш важливою складовою є відстеження тенденції та рівня задоволеності клієнтів, на фоні цих змін.

До поширених показників відносяться наступні:

- індекс CSAT;
- індекс CSI;
- індекс NPS.

Індекс CSAT – це метрика, яка допомагає визначити рівень задоволеності досвідом взаємодії клієнта з компанією, ґрунтуючись на оцінках проведення досліджень клієнтів на базі опитувальника.

Для визначення відповідей використовується шкала від 1 до 5, яка відображає рівень задоволеності клієнта від значення «дуже незадоволений» до «дуже задоволений» відповідно. Зібравши всі відповіді, розраховується індекс CSAT за формулою:

$$CSAT = \frac{A_5 + A_4}{A_a} * 100\%, \quad (1)$$

де A_a – загальна кількість оцінок,

A_5 – кількість оцінок «дуже задоволений»,

A_4 – кількість оцінок «задоволений».

За метрикою CSAT достатнім рівнем задоволеності клієнтів вважається значення індексу від 76,5%.

Індекс задоволеності клієнтів (CSI) – це метрика, яка використовується для загальної оцінки ступеня, в якій клієнт задоволений продуктом, послугою та досвідом.

За допомогою цієї метрики, можна дізнатися про ставлення клієнта до компанії, а вона також дає уявлення про різні аспекти бізнесу та ставлення клієнта до нього. Для розрахунку показника використовується наступна формула:

$$CSI = \frac{1}{n} \sum_{i=0}^n a_i \quad (2)$$

де a – вимірюваний атрибут,

n – кількість атрибутів.

Показник клієнтської лояльності (NPS) – це метрика, яка визначає рівень лояльності клієнта до компанії. Вона використовується на базі опитувальника. Зазвичай мірною шкалою вважаються значення від 1 до 10, де:

- від 0 до 6, це критики;
- від 7 до 8, нейтральні споживачі;
- від 9 до 10, прихильники.

Варіативність значення даної метрики в межах від -100 до 100, а розраховується значення за наступною формулою:

$$NPS = C_p - C_d \quad (3)$$

де C_p – відсоток прихильників,

C_d – відсоток критиків.

При використанні цих метрик, компанія більше розуміє свого кінцевого споживача та має змогу скорегувати свої дії для покращення роботи, якості продукції та ін.

Виходячи з того, що дані метрики мають різні шкали оцінювання, необхідно здійснити їх приведення до загального виду для подальшого використання та збільшення точності оцінювання.

Окрім опитувальників, частою формою зворотного зв'язку є фідбеки. Зазвичай, на їх обробку необхідно витрачати не тільки багато часу, але й багато спеціалістів. Зазвичай вони відображають загальне ставлення клієнта до компанії, яке відображається у вигляді емоційно-окрашених слів.

У зв'язці, фідбеки та опитувальники дають більш точну оцінку користувача. За допомогою опитувальників користувач відповідає на прямі питання компанії та оцінює її в рамках поставлених питань. На відміну від опитувальника, фідбек дає змогу користувачеві висловити думку та звернути увагу на відповідні аспекти, які знаходяться поза межами цих питань, та висловити своє ставлення до них. На відміну від опитувальників, даний вид інформації має більшу аналітичну

складність та потребує впровадження обробки природної мови та машинного навчання [3].

Так, як природна мова потребує попередньої обробки, необхідно провести очищення та векторизацію тексту. До етапів обробки фідбеків відноситься: токенизація речень, видалення «стоп-слів», нормалізація слів та їх перетворення в числове уявлення для класифікації [4].

До етапу токенизації речень відноситься розбиття тексту на більш атомарні одиниці. До таких можуть відноситися окремі речення або слова. На етапі видалення «стоп-слів» відбувається очищення даних від смислових та емоційно-невантажених слів, які не мають великого значення під час класифікації. На етапі нормалізації слів відбувається приведення всіх форм слів до одного виду методом легімізації. Цей метод використовує спеціальну структуру даних, яка пов'язує всі похідні форми слова з його найпростішою формою, що називається лемой. На етапі векторизації відбувається перетворення токенів в числовий масив, який собою представляє різноманітні характеристики токена та використовується для подальшої класифікації.

За допомогою класифікаторів машинного навчання формується резюме, яке включає інформацію про об'єкти висловлювання та належну їм тональну лексику.

Для правильної роботи класифікатора та більшої точності визначення тональності тексту необхідно навчити модель на заготовлених прикладах, які беруться з відкритих джерел.

Зазвичай, аналітики вимушені використовувати різні сервіси для визначення рівня лояльності клієнтів, що ускладнює загальний процес та вартість проведення таких заходів, що негативно впливає на точність результатів, збільшує час обробки даних та потребує додаткової обробки результатів для приведення їх до загального вигляду для подальшого використання.

Висновки. Таким чином, можна зробити висновки, що використання цих методів в комплексі мають дозволити проводити обширну аналітику відношення клієнтів, яка включатиме оцінку опитувальників за різними метриками, а також аналіз тональності фідбеків, що призведе до більшої точності аналітичних висновків та позитивно вплине на бізнес процеси в компанії за рахунок централізованої обробки даних та формування більш уніфікованої оцінки.

В даній роботі було проаналізовано методи збору даних від користувачів, обробки, аналізу та оцінки цих даних. В ході аналізу визначено основні метрики та технології, які можуть виступати у якості вимог до майбутньої програмної системи.

Список використаних джерел:

1. Боліла С. Ю., Кириченко Н. В., Осадчук І. В. Клієнторієнтовані технології формування конкурентних переваг підприємства на локальному ринку. Агросвіт. 2020. No 10.
2. Основні принципи клієнтоорієнтованості. Підвищення клієнтоорієнтованості. URL: <https://biznesua.com.ua/osnovni-printsipikliyantooriyentovanostipidvishhennya-kliyantooriyentovanosti>
3. Resnik P. Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language. Journal of Artificial Intelligence Research. 1999. No11. P. 95-130.
4. Hu M., Liu B. Mining and summarizing customer reviews. Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. 2004. P. 168-177.

PECULIARITIES OF LEGACY PROJECTS SUPPORT

Oleksii Kucherenko

e-mail: uitpm221_koi@student.ztu.edu.ua

Scientific consultation - doc. Yuliya Bogoyavlenska, Dr., Ph.D.

Zhytomyr Polytechnic State University

Relevance and statement of the problem. Sooner or later, every more or less large project enters the stage when it can be called Legacy. Legacy systems are often perceived as something negative. But it is not quite so. In order to deal with this, you need to understand what a legacy system is. Legacy systems have many definitions, but Bennett [1] defines it as “large software systems that we don’t know how to cope with but that are vital to our organizations”. As we can see, the main thing is that such systems are very important for business, but at the same time they are difficult to maintain due to their long life cycle.

The reason for the complexity of the systems is that the field of IT is developing very quickly. Certain solutions or technologies used today may be considered obsolete in a few years. Developers of durable products have two options: 1) keep the code up-to-date, update everything possible, replace the architecture with a more modern one every few years, switch from one programming language to another, etc.; 2) make changes only to vital parts of the code. In this work, we will be talking about methods and approaches of working with the second option.

The main features of the Legacy project include: the presence of several programming languages and technologies, often used haphazardly, the use of different architectural patterns in different parts of the application, confusing code, complex relationships, redundant classes, non-optimal or outdated architectural solutions. The reason does not necessarily have to be the lack of desire of developers to write good code. It's all about the constant evolution of the application's behavior and, as a result, the code. During constant changes, it is not always possible to follow the harmony of the code structure, sometimes it is simply impossible to add new functionality, following already existing approaches. All this can be complicated by the lack of documentation. In such cases, it can sometimes take hours to simply understand how a certain function should work. In addition, it is necessary to take into account the fact of the rotation of developers. There are known cases when a new manager starts implementing a new technology. As a rule, the previous technology remains in existence, because its complete replacement will require colossal efforts in the absence of additional value for the customer. As a result, many solutions, approaches and technologies exist simultaneously in one project. Obsolete technologies can carry risks of vulnerabilities. The search for developers for such projects is greatly complicated due to high knowledge requirements. Against the background of what has been announced, the appearance of the same bugs from version to version is not rare.

The main research materials. Let's look at the main methods of working with legacy projects. Conventionally, they can be divided into purposeful and ongoing. We can plan the first by conducting a thorough analysis of the technologies, programming languages, and code used. You can even make an action plan. This will include planned work on replacing outdated technologies, libraries, approaches, and architectures. As a rule, all these actions will require considerable effort and do not always make practical sense and are trivially likely to be completed in a reasonable time frame. For example, if the core of the system is built using a specific technology and programming language, the entire environment of the core, application modules are closely related to the core, then replacing the technology will be equivalent to creating a new system.

Recommendations for daily work with legacy code come to our aid in such a way that the state of the system does not deteriorate, and individual parts and modules are refactored, possibly written using modern programming languages. Such recommendations are described in Michael Feathers' book "Working Effectively with Legacy Code" [2]. They include:

- First, add tests, then do your changes - When code is not tested, you don't know you didn't break anything
- Adding tests: the Legacy Code dilemma - Before you change code, you should have tests in place. But to put tests in place, you have to change code. This is the paradox of Legacy Code! You should perform minimal, safe refactorings until you got tests in place.
- Identify Seams to break your code dependencies - Adding tests on the existing code can be challenging. That's because the code was not written to be testable in the first place. 99% of the time, this is a dependency problem. To test your code, you need to break these dependencies in the tests. Therefore, you need to identify Seams.
- Unit tests are fast and reliable – Michael Feathers gives a clear definition of unit tests: they should run fast (<100ms/test) and should not access the Infrastructure (DB, network, file system etc)
- Characterization tests - With most systems, what the code actually does is more important than what it should do. You can quickly cover Legacy Code with these tests, giving you a safety net to refactor.
- Use Sprout & Wrap techniques to add code when you don't have time to refactor
- Use scratch refactoring to get familiar with the code

Conclusion. Peculiarities of Legacy projects support are quite well studied. Time-tested recommendations are available for developers. One of the most influential sources of such recommendations is the book Working Effectively with Legacy Code by Michael Feathers. The main attention is paid to such recommendations as: extensive use of unit tests, refactoring, Sprout & Wrap techniques, Seams Identifying. It is also important to think about eliminating critical problems (updating vulnerable libraries, tools) and keeping the documentation up to date.

References:

1. K. Bennett, "Legacy Systems: Coping with Success", IEEE Software, page 19-23, 1995.
2. M. C. Feathers, Working Effectively with Legacy Code. Upper Saddle River: Prentice Hall PTR, 2013

The paper is developed in the framework of ERASMUS+ CBHE project "Digitalization of economic as an element of sustainable development of Ukraine and Tajikistan" / DigEco 618270-EPP-1-2020-1-LT-EPPKA2-CBHE-JP

This project has been funded with support from the European Commission. This document reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained there in.

УДК 004.4

РОЗРОБКА СОЦІАЛЬНОЇ МЕРЕЖІ ДЛЯ МЕЛОМАНІВ

Гордеєв Р.С., магістрант

e-mail: ipzm221_grs@student.ztu.edu.ua

Вакалюк Т.А., д. пед. наук, професор

e-mail: kik_vta@ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Кількість людей, що використовує інтернет дуже значна й продовжує рости. Насамперед ця технологія використовується як інструмент комунікації, який не має територіальних меж, та дозволяє обмінюватись різноманітними видами інформації. А для цих цілей одним з найпопулярнішим засобом є соціальна мережа.

Соціальні мережі, з часу їх появи, все більше проникають до нашого життя. Для багатьох вони вже зараз являються основним місцем проведення часу в інтернеті. Це трапляється через те, що ми не завжди можемо знайти співрозмовників зі спільними інтересами у реальному житті. Соціальні мережі дозволяють познайомитись з зовсім незнайомими людьми, яких об'єднує спільні зацікавленості.

В соціальних мережах люди збираються й обговорюють теми пов'язані зі спільними інтересами: ігри, книги, мистецтво і т.д. Я, як людина що цікавиться музикою, вирішив пошукати соціальну мережу, де люди можуть вести дискусії щодо музичних тем. Мені вдалось знайти всього два англомовних веб-сайти, а на українському ринку – ні одного. Тому я вирішив створити соціальну мережу для меломанів.

Об'єктом дослідження є автоматизація процесу обговорення та надання відгуків меломанами на музичні твори. Предметом дослідження є використання веборієнтованих технології та клієнт-серверної архітектури для автоматизації процесу обговорення та надання відгуків меломанами на музичні твори.

Основні матеріали дослідження. Основною метою впровадження соціальної мережі для меломанів є створення порталу для обговорення музики. Ця система має забезпечити можливість написання відгуків на музичні проекти, їх оцінювання; вести свою сторінку, де ділитися своїм музичним смаком; переглядати різноманітні музичні списки та створювати їх; брати участь в обговореннях, що будуть доступні на сторінці кожного альбому.

Для виконання мети необхідно виконати наступні завдання:

- здійснити аналіз функціональних потреб соціальної мережі для меломанів;
- обрати технології розробки веб-проекту;
- розробити архітектуру соціальної мережі для меломанів;
- визначити об'єктно-орієнтовану модель соціальної мережі для меломанів.

Для того, щоб розробити соціальну мережу якомога краще, потрібно переглянути існуючі розробки. Для цього було знайдено два веб-проекти: Metacritic та RateYourMusic. Проаналізувавши існуючі розробки, зробимо висновки, що нам потрібно використати з даних веб-сайтів: створити оцінки для альбомів від критиків та від користувачів; розрахування загальної оцінки музичного проекту; використання системи кольорів для оцінювання; створення

більших можливостей для оцінювання проєктів. Також потрібно зробити більш читабельний дизайн.

Для того, щоб реалізувати систему, потрібно обрати інструментальні засоби. Для цього для фронтенд частини оберемо: SASS – препроцесор, призначений для спрощення розробки на CSS[1]; React – фронтенд-бібліотека JavaScript для побудови користувацьких інтерфейсів[2]; Redux – це бібліотека JavaScript для управління та централізації стану додатків[3]. Для бекенд частини оберемо: Node.js – це середовище виконання JavaScript з відкритим вихідним кодом[4]; Express.js – це фреймворк, який знадобиться для розробки API[5]; node-postgres – SQL-драйвер, який потрібен для роботи з PostgreSQL[6]. База даних цього проєкту відповідно буде PostgreSQL.

Для опису архітектури використаємо діаграму розгортання (зображена на рисунку 1). Діаграма розгортання в UML моделює фізичне розгортання артефактів на вузлах. Для опису веб-сайту, наприклад, діаграма розгортання покаже, які апаратні компоненти ("вузли") існують (наприклад, веб-сервер, сервер додатків та сервер бази даних), які компоненти програмного забезпечення ("артефакти") запускаються на кожному вузлі (наприклад, веб-додаток, база даних) та як з'єднуються різні частини.

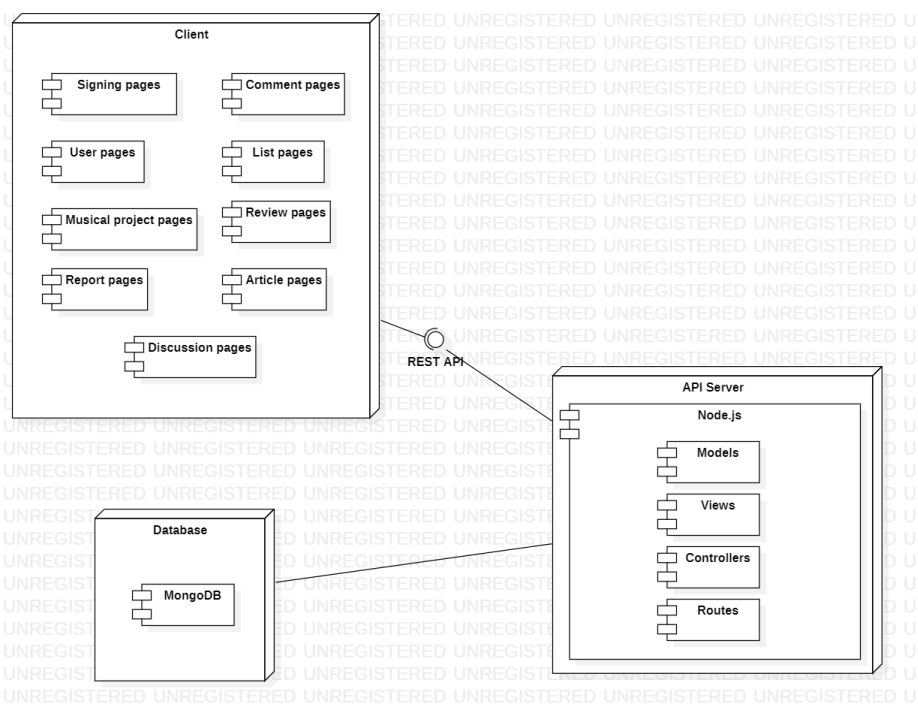


Рисунок 1. Схема архітектури соціальної мережі для меломанів

Для опису класів використаємо діаграму класів (зображена на рисунку 2). Діаграма класів є основним будівельним блоком об'єктно-орієнтованого моделювання. Використовується як для загального концептуального моделювання структури програми, так і для детального моделювання, перетворення моделей в програмний код. Діаграми класів також можна використовувати для моделювання даних. Класи на діаграмі класів представляють як основні елементи, взаємодії в додатку, так і класи, які потрібно запрограмувати.

Рисунок 2. Схема об'єктно-орієнтованої моделі соціальної мережі для меломанів

Список використаних джерел:

УДК 519.872

ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ПРОГРАМНОГО ПРОДУКТУ GPSS WORLD ПРИ МОДЕЛЮВАННІ ІНФОРМАЦІЙНИХ СИСТЕМИ

Мацулевич О.Є., к.т.н., доцент

e-mail: oleksandr.matsulevych@tsatu.edu.ua

Тетервак І.Р., асистент

e-mail: illia.tetervak@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність дослідження та постановка проблеми. Ефективність побудови і використання інформаційних систем стала надзвичайно актуальним завданням, особливо в умовах недостатнього фінансування інформаційних технологій на підприємствах. Основу інформаційної системи складає обчислювальна система, яка включає такі компоненти, як локальна мережа, комп'ютерне та периферійне обладнання, обладнання зберігання даних, системне програмне забезпечення.

Найбільш поширеним підходом до проектування інформаційних систем в даний час є використання експертних оцінок. Відповідно до цього підходу фахівці в області комп'ютерних мереж на підставі досвіду і експертних оцінок здійснюють проектування обчислювальної системи. Цей підхід дозволяє мінімізувати витрати на етапі проектування, швидко оцінити вартість реалізації інформаційної системи. Проте рішення, отримані з використанням експертних оцінок, носять суб'єктивний характер, як і оцінка гарантій працездатності пропонованого проекту системи.

Як альтернативний може бути використаний підхід, що передбачає розробку моделі і моделювання поведінки обчислювальної системи. Моделювання обчислювальної системи дозволяє провести точніший, в порівнянні з експертними оцінками, розрахунок необхідної продуктивності окремих компонентів і всієї системи в цілому. При моделюванні на комп'ютері відтворюються реальні процеси в обстежуваному об'єкті, досліджуються особливі випадки, відтворюються реальні і гіпотетичні критичні ситуації.

Таким чином, актуальною є проблема застосування моделювання для проектування та дослідження роботи інформаційних систем.

Існують спеціальні, орієнтовані на моделювання обчислювальних мереж програмні системи [1], в яких процес створення моделі спрощений. Такі програмні системи самі генерують модель мережі на основі вихідних даних про її топологію. Проте жодну з програмних систем не можна розглядати як повністю готовий до вживання засіб, здатний в точності змодельовати роботу мережі. Необхідно витратити значні кошти на навчання персоналу, перш ніж стануть можливими побудова коректних моделей і інтерпретація отриманих результатів. Крім того ці програмні продукти коштують дуже дорого і не кожне підприємство здатне їх купити. Існують спеціальні мови імітаційного моделювання, які полегшують процес створення програмної моделі і коштують на порядок нижче в порівнянні із спеціалізованими системами імітаційного моделювання обчислювальних мереж. Прикладом мови імітаційного моделювання може служити така мова, як GPSS World.

Основні матеріали дослідження. Імітаційне моделювання - один з видів комп'ютерного моделювання, що використовує методологію системного аналізу, центральною процедурою якого є побудова узагальненої моделі, що відображає

всі фактори реальної системи, як методологію дослідження виступає обчислювальний експеримент. Імітаційна модель будується строго цілеспрямовано, тому для неї характерно адекватне відображення об'єкта, що досліджується, логіко-математична модель системи являє собою програмно реалізований алгоритм функціонування системи. При імітаційному моделюванні структура системи, що моделюється, адекватно відображається в моделі, а процес її функціонування імітується на побудованій моделі. Під імітацією розуміють проведення на комп'ютерах різних серій експериментів з моделями, які представлені деяким набором (комплексом) комп'ютерних програм. Порівняння характеристик (конструкцій, управлінь) об'єкта, що моделюється, здійснюється шляхом варіантних прорахунків. Особливу роль має можливість багаторазового відтворення моделюваних процесів з подальшою їхньою статистичною обробкою, що дозволяє враховувати випадкові зовнішні впливи на об'єкт, що вивчається. На основі статистики, що набирається в ході комп'ютерних експериментів, робляться висновки на користь того чи іншого варіанту функціонування або конструкції реального об'єкта або сутності явища. У ряді випадків формувати рішення за допомогою формальних методів не вдається - експерт має бути включений до процесу ухвалення рішення. Він стає активним компонентом інформаційної системи; деталізує проблему та модель, здійснює постановку спрямованого обчислювального експерименту на моделі, генерацію та ранжування альтернатив, вибір критеріїв для прийняття рішень, а також формує раціональний варіант управління за допомогою бази знань. Прийняття рішень в умовах ризику, наприклад, вимагає ведення діалогових процедур формування статистично достовірних результатів та поетапного зіставлення їх із функцією ціни ризику. Необхідно здійснювати пряму участь експерта у формуванні оптимальної множини варіантів рішень та у процедурах варіантного синтезу. Таким чином, імітаційне моделювання значно розширює можливості та ефективність роботи осіб, які приймають рішення (ЛПР), надаючи їм зручний інструмент та засоби для досягнення поставлених цілей. Імітаційне моделювання реалізує ітераційний характер розробки моделі системи, поетапний характер деталізації підсистем, що моделюються, що дозволяє поступово збільшувати повноту оцінки прийнятих рішень у міру виявлення нових проблем і отримання нової інформації. Імітаційна модель не дає оптимального рішення подібно до класичного вирішення задач оптимізації, але вона є зручним для системного аналітика допоміжним засобом для пошуку вирішення певної проблеми.

Область застосування імітаційних моделей практично не обмежена, це можуть бути завдання: дослідження структур складних систем та їх динаміки, аналізу вузьких місць, прогнозування та планування тощо. Головною перевагою імітаційного моделювання і те, що може відповісти питанням: «Що буде, якщо ...», тобто. з допомогою експерименту моделі виробляти стратегію розвитку. Останнім часом ведуться роботи з розробки систем, здатних надати допомогу експерту у відповідь на зворотне запитання «Що треба, щоб...». Це можна назвати «цільовим моделюванням», при якому на вхід системи подаються показники цільового стану, а також перелік можливих регуляторів із зазначенням діапазону та кроку їх зміни. Система в автоматичному або напівавтоматичному режимі знаходить поєднання значень цих регуляторів досягнення заданого цільового стану. Отже, переваги системно-динамічного моделювання полягають у наступному: системно-динамічний підхід починається зі спроби зрозуміти ту систему причин, яка породила проблему та продовжує її підтримувати. Для цього збираються необхідні дані з різних джерел, включаючи літературу,

поінформованих людей (менеджерів, споживачів, конкурентів, експертів) та проводяться спеціальні кількісні дослідження. Після того, як елементарний аналіз причин проблеми зроблено, формальна модель вважається побудованою. Спочатку вона представляється у вигляді логічних діаграм, що відображають причинно-наслідкові зв'язки, які потім перетворюються на мережеву модель.

Це гнучких засіб, що посилює можливості людини, використовує її глибшого розуміння проблеми. Таким чином, у сфері сучасних інформаційних технологій імітаційне моделювання набуває у світових наукових дослідженнях та практичній діяльності вкрай вагомego значення. З допомогою імітаційного моделювання ефективно вирішуються завдання найширшої проблематики, - у сфері стратегічного планування, бізнес-моделювання, менеджменту (моделювання різноманітних фінансових проєктів, управління виробництвом), реінжинірингу, проектування, актуальне застосування імітаційного моделювання у сфері інвестиційно-технологічного проектування.

Процеси функціонування локальних обчислювальних мереж можуть бути представлені сукупністю систем масового обслуговування (СМО) – дискретно-безперервних математичних моделей. Дослідження характеристик таких моделей може проводитися шляхом імітаційного моделювання. Імітаційна модель відображує стохастичний процес зміни дискретних подій СМО в безперервному часі у формі моделюючого алгоритму. При його реалізації на комп'ютері проводиться накопичення статистичних даних по тих атрибутах моделі, характеристики яких є предметом досліджень. Після закінчення моделювання накопичена статистика обробляється, і результати моделювання виходять у вигляді вибірових розподілів досліджуваних величин.

Складні функції моделюючого алгоритму можуть бути реалізовані засобами універсальних мов програмування (Паскаль, C++), що надає необмежені можливості в розробці, відладці і використанні моделі. Проте подібна гнучкість отримується ціною великих зусиль, що витрачаються на розробку і програмування складних моделюючих алгоритмів, що оперують із списковими структурами даних. Альтернативою цьому є використання спеціалізованих мов імітаційного моделювання. Програми імітаційних моделей на спеціалізованих мовах моделювання близькі до описів модельованих систем на природній мові, що дозволяє конструювати складні імітаційні моделі користувачам, що не є професійними програмістами.

Одним з найбільш ефективних і поширених пакетів моделювання складних дискретних і безперервних систем є в даний час пакет GPSS World, розроблений компанією Minuteman (США). Він може бути з найбільшим успіхом використаний для моделювання систем, що формалізуються у вигляді систем масового обслуговування. Як об'єкти мови використовуються аналогі таких стандартних компонентів СМО, як заявки, обслуговуючі прилади, черги. На персональних комп'ютерах GPSS World реалізований в рамках пакету прикладних програм. Основний модуль пакету включає окрім транслятора з вхідної мови, засіб введення і редагування тексту моделі, її відладки і спостереження за процесом моделювання, графічні засоби відображення атрибутів моделі, а також засоби накопичення результатів моделювання в базі даних і їх статистичної обробки. Окрім основного модуля до складу пакету входить модуль створення стандартного звіту GPSS World, а також ряд додаткових модулів і файлів.

Постановка задачі. Розглянемо локальну обчислювальну мережу (ЛОМ). Нехай ЛОМ має чотири автоматизованих робочих місця (АРМ) і сервер, сполучених спільною шиною. Запити поступають від АРМ, які мають свої бази

даних (БД). Запити, що поступають, первинно обробляються на АРМ. З вірогідністю P_i ($i=1..4$, порядкові номери АРМ) потрібна інформація виявляється в БД АРМ, після чого продовжується подальша обробка запиту. Інакше необхідна посилка запиту на сервер. Після посилки запиту на сервер АРМ обробляє інші запити, що поступають на нього. Отримавши відповідь з сервера, АРМ завершує обробку запиту. Інтервали часу отримання запитів АРМ розподілені по показовому закону. Канал передачі даних (КПД), що сполучає АРМ і сервер, не має накопичувачів і передача по ньому можлива лише тоді, коли він вільний. Коли він зайнятий, АРМ знаходиться в режимі очікування і лише після звільнення каналу передає запит. При передачі даних на сервер по запиту АРМ цим даним привласнюється вищий пріоритет в порівнянні із запитами, що поступають на АРМ. На сервері є два накопичувачі. Накопичувач 1 призначений для запитів, що поступають, а другий – для передачі відповідей на запити. Ємкість накопичувача 1 обмежена 1000 запитами, тобто запити, що поступають з АРМ, можуть втрачатися в разі повного заповнення накопичувача 1. Ємкість накопичувача 2 практично необмежена, тобто відповіді на запити не втрачаються. Час первинної обробки запиту, його передачі при потребі на сервер, обробки на сервері і зворотної передачі на потрібне АРМ підпорядковані експоненціальному закону.

Необхідно промодельовувати функціонування ЛОМ протягом 12 годин і визначити:

- 1) раціональну ємкість накопичувача 1 на сервері, при якій не відбувається втрат запитів з АРМ;
- 2) вірогідність відмови відповіді на запит з АРМ унаслідок повного заповнення накопичувача 1 на сервері;

В результаті 12 прогонів моделі було отримано наступний результат:

- 1) раціональна ємкість накопичувача 1 на сервері, при якій не відбувається втрат запитів з АРМ дорівнює 1741;
- 2) вірогідності відмови відповіді на запит з АРМ унаслідок повного заповнення накопичувача 1 на сервері дорівнюють:

ЛВС – 0.202

АРМ1 – 0.062

АРМ2 – 0.088

АРМ3 – 0.119

АРМ4 – 0.147

Висновки. Запропоновано нові можливості моделювання інформаційних систем засобами програми GPSS World. Застосування програми GPSS World дасть можливість проведення будь-яких експериментів з інформаційною системою, не удаючись до фізичної реалізації, що дозволить запобігти великої кількості неочікуваних ситуацій.

Список використаних джерел:

1. Олифер Н.А., Олифер В.Г. Использование моделирования для оптимизации производительности сети. Центр информационных технологий. http://www.citforum.ru/nets/optimize/locnop_09.shtml
2. Щербина В.М., Мацулевич О.Є. Спосіб побудови дотичних у вузлах спиралеподібних дискретно представлених кривих із використанням спеціальної функції // Праці Таврійського державного агротехнологічного університету імені Дмитра Моторного. Мелітополь: ТДАТУ імені Дмитра Моторного, 2019. Вип. 19, т. 2. С. 278-293

3. Мацулевич О.Є., Щербина В.М., Коломієць С.М. Геометричне моделювання складних тривимірних поверхонь із застосуванням матричного рівняння еліптичного повороту. //Праці Таврійського державного агротехнологічного університету. Мелітополь, 2019. Вип. 19, т. 2. С. 294-300

4. Мацулевич О.Є., Щербина В.М., Бондаренко Л.Ю., Малюта С.І., Антонова Г.В. Програмне забезпечення для автоматизованого визначення параметрів різального інструменту фрезерної обробки корпусних деталей // Праці Таврійського державного агротехнологічного університету. – Мелітополь: ТДАТУ, 2020. – Вип. 20, т. 3. С 275-281.

5. Холодняк Ю.В., Гавриленко Е.А., Ивженко А.В., Найдыш А.В. Моделирование участка пространственной монотонной кривой линии /Сучасні проблеми моделювання: наукове фахове видання. – Мелітополь: МДПУ ім. Б. Хмельницького, 2020. Вип.17. С. 131-137.

6. Гавриленко Е.А., Холодняк Ю.В., Антонова Г.В., Чаплинский А.П. Разработка алгоритма программного обеспечения для формирования обводов по заданным геометрическим условиям. Праці Таврійського державного агротехнологічного університету імені Дмитра Моторного. – Мелітополь: ТДАТУ ім. Д. Моторного, 2020. – Вип. 20, т. 3. (с.293-303).-DOI: 10.31388/2078-0877-2020-20-3-293-303.

7. Мацулевич О.Є., Вершков О.О., Холодняк Ю.В., Дмитрієв Ю.О., Чаплінський А.П. Розробка мурашиного алгоритму для оптимізації оперативного планування робіт по збиранню врожаю кісточкових. *Плодовий сад – новітнє в теорії та практиці*: матеріали VВсеукр. наук.-практ. інтернет-конференції. Мелітополь: ТДАТУ, 2021. С. 106-110.

8. Вершков О.О., Мацулевич Ю.О. Визначення шорсткості поверхонь із застосуванням програмного забезпечення СОРУСАD ф. DELCAM plc // Сучасні комп'ютерні та інформаційні системи і технології, Матеріали і всеукраїнської науково-практичної інтернет-конференції, Мелітополь 7-25 грудня 2020р. С. 17-23

УДК 004.9

ОПИС РОБОТИ КОРИСТУВАЧА З ПРОГРАМНИМ МОДУЛЕМ «РОЗРАХУНОК РЕЖИМІВ РІЗАННЯ ПРИ ВИКОНАННІ СВЕРДЛИЛЬНИХ ОПЕРАЦІЙ»

Дережа О.О., к.т.н., доцент

e-mail: olena.dereza@tsatu.edu.ua

Бондаренко Л.Ю., к.т.н., доцент

e-mail: larysa.bondarenko@tsatu.edu.ua

Антонова Г.В., ст. Викладач

e-mail: galina.antonova@tsatu.edu.ua

Тетервак І.Р., асистент

e-mail: illia.tetervak@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність дослідження та постановка проблеми. Основними технологічними способами обробки отворів різного ступеня точності і з різною шорсткістю обробленої поверхні є свердління, зенкування і розгортання.

В наявних пакетах програм для автоматизованого проектування відсутній модуль для розрахунку режимів різання свердел.

У зв'язку з цим на підприємствах стає задача проектування програмного модуля для розрахунку режимів різання свердел, що можна вважати *актуальним*.

Основні матеріали дослідження. Для початку роботи з програмою «Розрахунок режимів різання при виконанні свердлильних операцій» необхідно запустити файл RRSver.exe, що знаходиться в робочій папці програми. Відкриється вікно авторизації де необхідно буде ввести логін і пароль користувача.

Якщо програма не знайде файл з користувачами по вказаному шляху на локальному диску «C:\Розрахунок режимів різання свердел\usr.txt», то необхідно буде створити нового користувача, методами, які зображені на рисунку 1. В наслідок чого, в корні каталогу буде створено новий файл з користувачами.

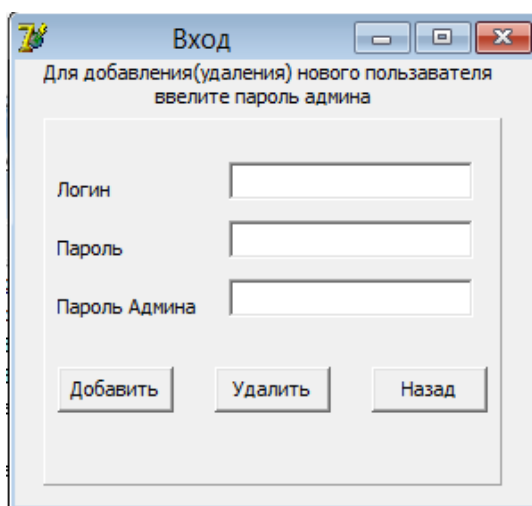


Рисунок 1. Створення нового користувача

Головна форма (рис. 2) складається з полів вхідних даних, які необхідно обрати та заповнювати.

Расчет режимов резания при сверлении

Файл Расчет Справка

Здесь необходимо выбрать вид требуемой операции :

Обработанные сверлением отверстия имеют параметр шероховатости $Ra=12.5$ мкм и точность соответствующую 12-14 качеству

Обработанные зенкерованием отверстия имеют параметр шероховатости $Ra=6.3$ мкм и точность соответствующую 10-14 качеству

Обработанные развертыванием отверстия имеют параметр шероховатости $Ra=0.32$ мкм и точность соответствующую 6-9 качеству

Вид обработки : Сверление

Вид отверстия : Сквозное

Обрабатываемый материал : Сталь углеродистая

Инструментальный материал

☒ Быстрорежущая сталь P6M5

☐ Твердый сплав

Модель станка 2N118

Инструмент Сверло

Длина обрабатываемой поверхности : 12 мм

Твердость обрабатываемого материала

☒ HВ 260

☐ Сигма

Диаметр отверстия

Диаметр получаемого отверстия D= 6

Расчет >>

Рисунок 2. Головна форма програми.

Головна форма програми складається з наступних властивостей:

- Вид обробки;
- Вид отвору;
- Оброблюваний матеріал;
- Інструментальний матеріал;
- Модель верстата;
- Інструмент;
- Твердість оброблюваного матеріалу;
- Діаметр отвору;
- А також допоміжної форми з підказками.

На формі «Результати розрахунків» (рис. 3), відображаються наступні елементи:

- Глибина різання;
- Подача;
- Період стійкості інструменту;
- Дійсна швидкість різання;
- Частота обертання шпинделя;
- Обертальний момент;
- Осьова сила;
- Потужність двигуна;
- Кнопка для збереження результатів розрахунку, для подальшого друку.

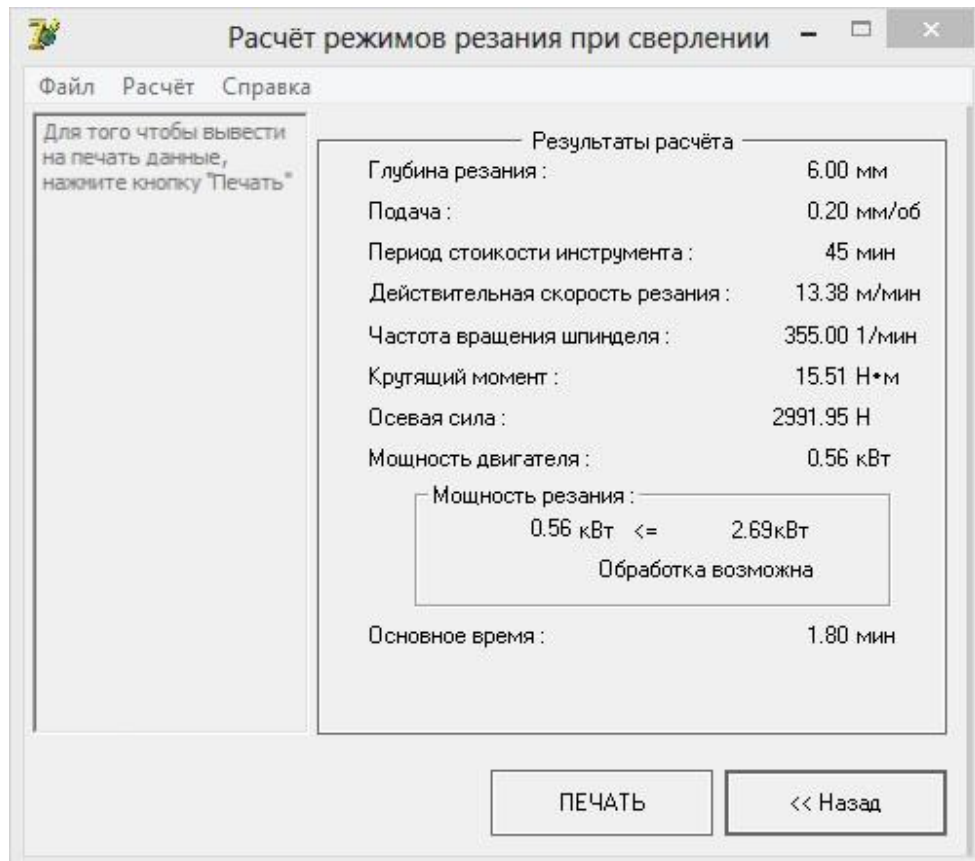


Рисунок 3. Результат роботи програми

Для створення моделі «Плита 2П20-401» була використана програма КОМПАС 3D V14. У вікні обираємо «Файл» → «Створити новий» → «Деталь».

Після того, як вікно нової деталі відкриється, ми обираємо початкову площину, та створюємо наш базовий ескіз (рис. 4).

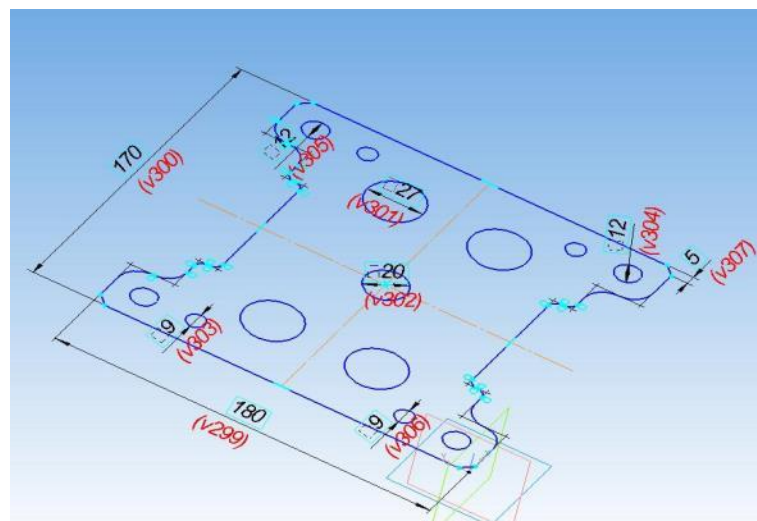


Рисунок 4. Ескіз деталі

Після того, як ми закінчили створення ескізу, обираємо операцію «Операція видавлення» видавлюємо на відстань 20 мм. В ескізі за допомогою функції «зкруглення» зкруглюємо гострі кути радіусом 0,5 мм.

Наступним кроком буде створення отворів (рис. 5) вирізаємо методом видавлювання отвори в кількості 4 отворів діаметром 15 мм.

Далі створюємо канавку в нижній частині плити. Використовуючи функцію «Вирізати видавлюванням».

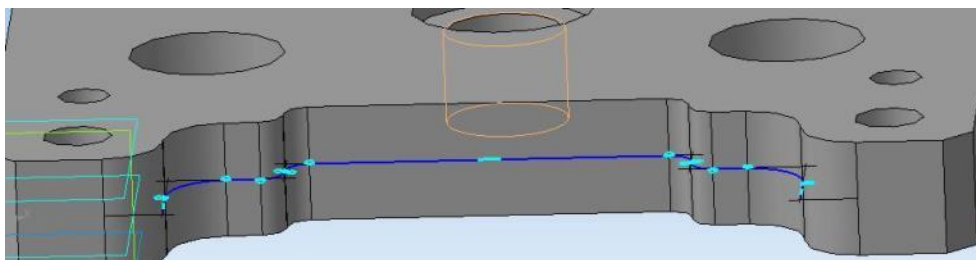


Рисунок 5. Створення канавки

Для створення креслення деталі необхідно натиснути «Файл» → «Створити новий» → «Креслення». Редагуємо формат нашого креслення, обираючи формат A2 та горизонтальну форму листа. Далі натискаємо кнопку «Додати з файлу» (рис. 6).

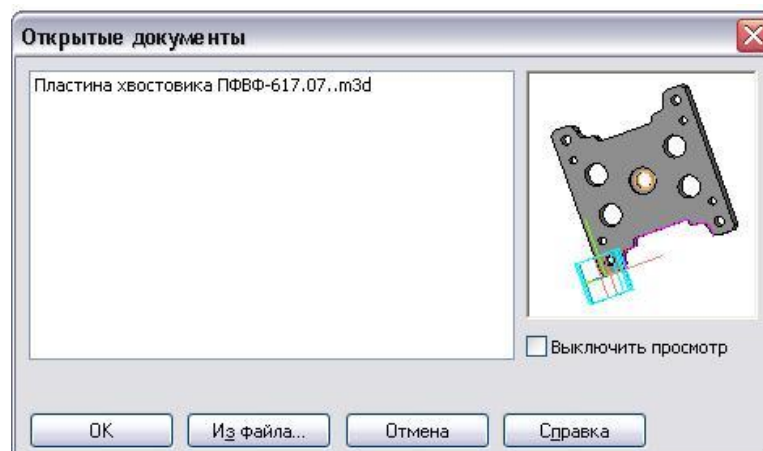


Рисунок 6. Перехід від 3d моделі до ескізу

Обираємо два головних види, та розміщуємо їх на нашому кресленні та пропонуємо головні розміри. Отримуємо готове креслення нашої моделі

Найважливішою задачею інженерних розрахунків є пошук коефіцієнта запасу міцності деталі.

Для обґрунтування вибору автоматизованої системи інженерних розрахунків був використаний пакет COSMOSWorks, який дозволяє розраховувати різноманітні деталі і зборки на міцність, теплове й аеродинамічні навантаження, що дає можливість обґрунтовано створювати надійні конструкції.

В результаті необхідно знайти таку силу при якій деталь може не витримати прикладеної до неї сили, тоді коефіцієнт запасу міцності буде дорівнювати приблизно одиниці.

Ітераційним методом були знайдені сили, при яких коефіцієнт запасу

міцності існуючої та модифікованої деталі дорівнює 1,1.

Наступний випробування було проведено з існуючою та модифікованою деталлю «Плита» з силою, що діє на модель в процесі роботи та приблизно дорівнює 52 000 Н (рис. 7).

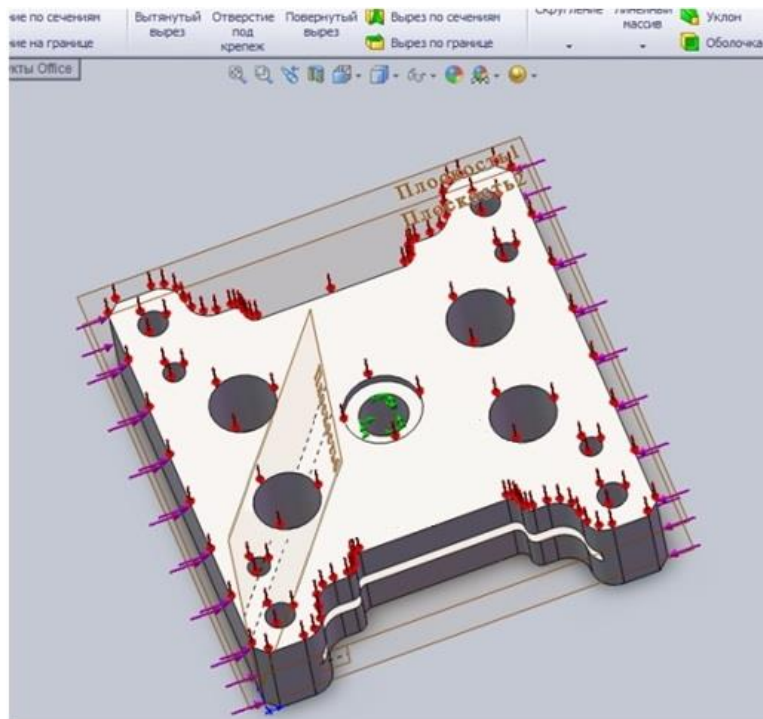


Рисунок 7. Накладання обмежень

Одним з основних та складних етапів у машинобудуванні є процес розрахунку деталей і вузлів на міцність.

Аналіз міцності за допомогою COSMOSWorks 2007, дозволяє розв'язати поставлені задачі.

Для запуску процесу аналізу міцності необхідно визначити матеріал деталі (механічні властивості), навантаження на деталь та обмеження.

Для початку роботи в COSMOSWorks завантажується 3D модель. Розрахунки в COSMOSWorks виконуються у вигляді Вправ. В Менеджері COSMOSWorks за допомогою контекстного меню вибирається команда Вправа, вказуються наступні параметри: Тип аналізу → Статичний, Тип сітки → Сітка на твердому тілі.

Наступним етапом задаються вихідні параметри розрахунку на міцність.

У контекстному меню твердотільного елемента «Плита» в Менеджері COSMOSWorks вибирається команда «Застосувати/редагувати матеріал». У діалоговому вікні «Матеріал» обираються матеріали, наявні в базах Solid Works. Для «Плити» обрано матеріал лита вуглецева сталь (рис.8).

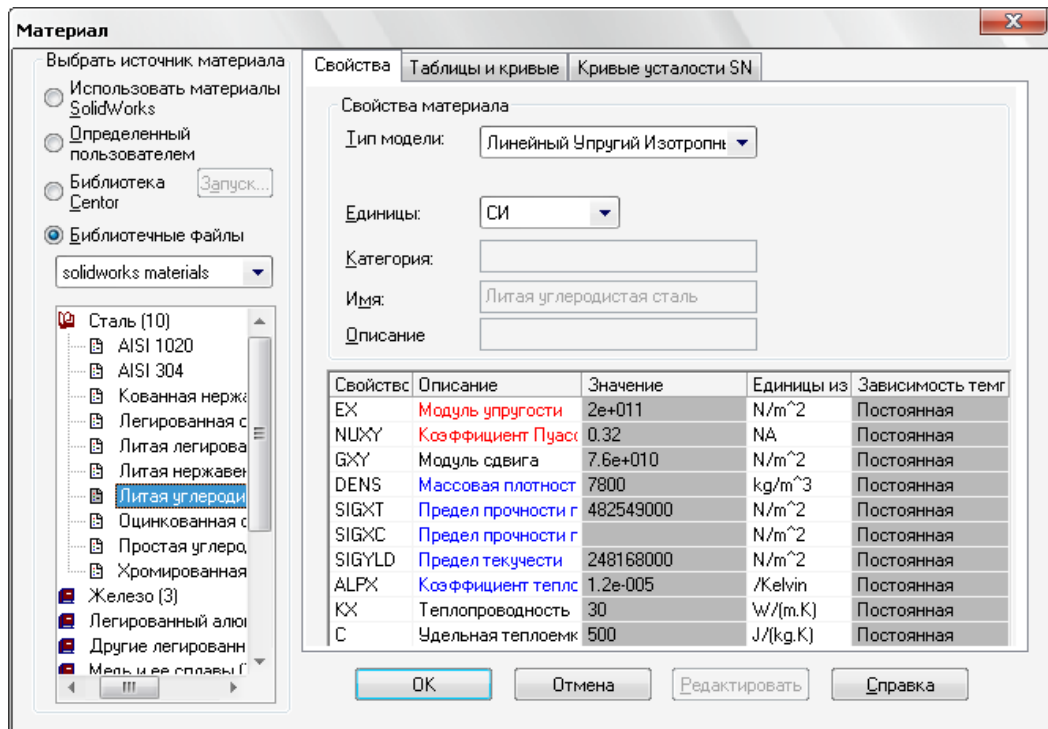


Рисунок 8. Діалогове вікно Матеріалу

Зробивши аналіз результату на міцність видно, що при незначних конструктивних змінах та матеріалу деталі «Пластина» коефіцієнт міцності був збільшений майже у 1,5 рази (рис. 9).

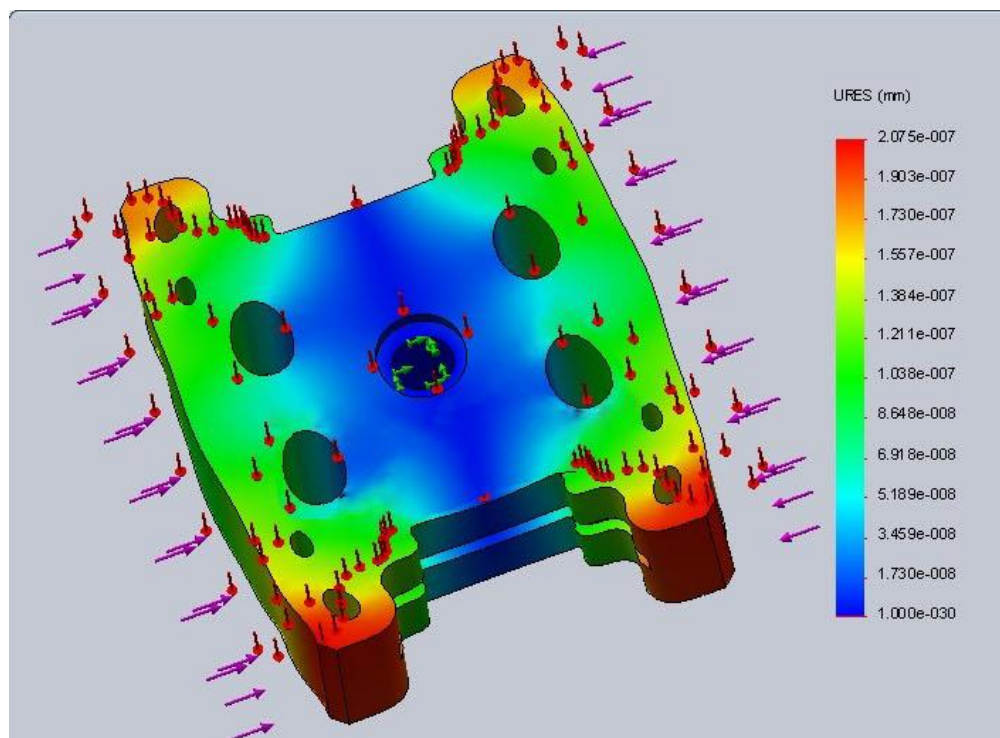


Рисунок 9. Аналіз результату на міцність

Висновки. В даній роботі було представлено опис спроектованого програмного забезпечення, проведено верифікацію програми а саме:

- перевірено роботу програмного модулю розрахунку режимів різання свердел;
- перевірено роботу модуля розрахунку міцнісних показників спроектованої деталі;
- виявлено, що створений програмний модуль повністю відповідає поставленому завданню і не має логічних помилок.

Список використаних джерел:

1. Аверченко В.И. «САПР технологічних процесів, пристроїв та ріжучих інструментів» – К. Виш. шк., 1993 – 288 с.
2. Проектування керуючих програм для верстатів токарної групи з пристроєм числового програмного керування: навчальний посібник / Дмитрієв Ю.О., Мацулевич О.Є., Щербина В.М., Холодняк Ю.В. Мелітополь, Люкс, 2018. 132 с.
3. Теоретична механіка. Динаміка матеріальної точки: навчально – методичний посібник / О.М. Леженкін, Г.В. Антонова, О.О. Вершков, Л.Ю. Бондаренко, О.Є. Мацулевич, А.О. Смєлов, О.Ю. Михайленко. Мелітополь: Люкс, 2021. 160с.
4. Програмування автоматизованих процесів обробки деталей: навчально–методичний посібник. Лабораторний практикум /Ю.О. Дмитрієв, О.Є. Мацулевич, Є.А. Гавриленко, Ю.В. Холодняк, Г.В. Антонова – Мелітополь (протокол №8 від 28.04.2022р.), 2022.–170с.

УДК 515.8 + 004.9

**СПЕЦІАЛІЗОВАНИЙ ПРОГРАМНИЙ МОДУЛЬ РОЗРАХУНКУ
ОПЕРАЦІЙНИХ НОРМ ЧАСУ ОБРОБКИ ДЕТАЛЕЙ
СІЛЬСЬКОГОСПОДАРСЬКОЇ ТЕХНІКИ ЗА УМОВ ІНДИВІДУАЛЬНОЇ
ОРГАНІЗАЦІЇ ПРАЦІ**

Івженко О.В., к.т.н., доцент

e-mail: oleksandr.ivzhenko@tsatu.edu.ua

Антонова Г.В., ст. викладач

e-mail: galina.antonova@tsatu.edu.ua

Чаплінський А.П., ст. викладач

e-mail: andrii.chaplinskyi@tsatu.edu.ua

Михайленко О.Ю., ст. викладач

e-mail: olena.mykhailenko@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність дослідження та постановка проблеми. Застосування ЕОМ при проектно-конструкторських роботах у своєму розвитку пройшло кілька стадій і перетерпіло значні зміни. З появою обчислювальної техніки створюються програми автоматизованих розрахунків на основі методів обчислювальної математики. Із впровадженням спеціалізованих термінальних пристроїв з'являються універсальні програми для ЕОМ для рішення як розрахункових, так і деяких рутинних проектних завдань (виготовлення креслень, специфікацій, текстових документів і т.п.).

Автоматизація розрахунку норм праці проводиться в даний час багатьма організаціями: частіше створюються системи розрахунку операційних норм часу для умов індивідуальної організації праці. Подібні системи створюються іноді як автономні системи нормування на базі заданого трудового і технологічного процесу.

Основні матеріали дослідження. Під вхідними документи слід розуміти документи, що містять інформацію, необхідну для ініціалізації роботи відповідного бізнес-процесу. Будь-який бізнес-процес перед початком роботи повинен отримати необхідну для його правильної роботи інформацію і після чого приступити до виконання, притаманних йому, бізнес-функцій. В результаті виконання бізнес процесу формується вихідна інформація.

Виходячи з того, що бізнес-процес включає декілька етапів, вихідні документи одного етапу є вхідними для наступного. Всі вхідні та вихідні документи можна звести у таблицю 1.

Таблиця 1 – Характеристика бізнес процесу обліку торгових представників

Назва характеристики	Значення характеристики
Вхідні документи	3D модель деталі
Вхідні документи	Дані з 1С про норми часу виготовлення аналогічних деталей
Вхідні документи	Попередній розгляд, розмітка
Вихідні документи	Погодження проекту документа

Назва характеристики	Значення характеристики
Вхідні документи	Реєстрація
Вихідні документи	Перевірка правильності оформлення проекту документа секретарем
Вхідні документи	Розгляд документів керівництвом
Вихідні документи	Підписання документа керівником (у необхідних випадках - затвердження)
Вхідні документи	Направлення на виконання
Вихідні документи	Реєстрація документа
Вхідні документи	Контроль виконання
Вхідні документи	Виконання документів
Вихідні документи	Підшивка другого примірника (копії) документа до справи

При плануванні програмного проекту є величезний вибір мов програмування, в лабіринтах яких легко заблукати. Вибір мови залежить від багатьох факторів. Якщо це особистий проект або хобі, можна вибрати знайомий вам мову. Якщо вибір залежить від наявних ресурсів, результат може бути досить неочевидним. Можна також витратити багато часу на розробку повторно використовуваних компонентів, так що документація перетвориться на кошмар.

При виборі мови програмування потрібно враховувати безліч чинників. Наприклад, якщо при розробці динамічної Web-сторінки ви в якості найкращого варіанту виберіть JavaServer Pages (JSP) / сервлети, інші можуть віддати перевагу PHP або аналогічний мова сценаріїв. Не існує якогось одного мови, яка є найкращим вибором. Можна віддати перевагу певним факторам, таким як продуктивність і безпеку корпоративних додатків, в порівнянні з іншими факторами, такими як кількість рядків коду. Будь-яке рішення пов'язане з якимись компромісами.

Після отримання проекту або завдання потрібно виконати підготовчу роботу до вирішення поставленого завдання. Часто вибір мови не розглядається як частина цієї підготовчої роботи.

При виборі мови для персонального проекту можна покластися на свої особисті переваги. Тут може виявитися важливим кількість рядків коду; очевидним вибором буде мова, що дозволяє виконати завдання за допомогою 10 рядків коду замість 20. Спочатку хочеться отримати рішення, а потім подбати про зручності читання або продуктивності.

У проектах для великих організацій застосовується інший сценарій. Для вирішення конкретної проблеми групи розробників створюють компоненти, які взаємодіють і взаємопов'язані між собою. На вибір мови можуть вплинути такі фактори, як переносимість програми на іншу платформу або доступність ресурсів.

Правильний вибір мови програмування допоможе створити компактне, просте в налагодженні, розширенні, документуванні та виправленні помилок рішення. При виборі мови програмування враховуються такі чинники:

- цільова платформа;
- гнучкість мови;
- час виконання проекту;
- продуктивність;
- підтримка і спільнота.

Існує дуже багато різноманітних мов програмування, але для реалізації поставленої задачі нами був обраний саме язык C#.

C# (Сі-шарп) - об'єктно-орієнтована мова програмування для платформи .NET. Розроблено в 2000 Андерсом Хейлсбергом, Скоттом Вілтанутом і Пітером Гольде під наглядом Microsoft Research. Основним постулатом C# є вислів: "всяка сутність є об'єкт". мова заснований на строгій компонентній архітектурі і реалізує передові механізми забезпечення безпеки коду.

C# був створений спеціально для технології ASP.NET. У той же час на C# повністю написана і сама ASP.NET.

C# - це повнофункціональний об'єктно-орієнтована мова, яка підтримує всі три «стовпи» об'єктно-орієнтованого програмування: інкапсуляцію, успадкування і поліморфізм. Він має прекрасну підтримку компонентів, надійний і стійкий завдяки використанню «збірки сміття», обробки виключень, безпеки типів [4].

Мова C# розроблявся "з нуля" і увібрав в себе багато корисних властивостей таких мов, як C++, Java, Visual Basic, а також Pascal, Delphi та ін. При цьому необхідність забезпечення сумісності з попередніми версіями відсутня, що дозволило мові C# уникнути багатьох негативних сторін своїх попередників.

Як і Java, C# розроблявся для Інтернет і приблизно 75% його синтаксичних можливостей аналогічні мові програмування Java, його також називають «очищеною версією Java. 10% подібні мови програмування C++, а 5% - запозичені з мови програмування Visual Basic. Обсяг нових концептуальних ідей в мові C# близько 10%.

Виділення і об'єднання кращих ідей сучасних мов програмування робить мову C# не просто сумою їх достоїнств, а мовою програмування нового покоління.

Для розробки програмного модуля визначення норм часу та заробітної плати була обрана мова програмування C# та середовище розробки Visual Studio. Перш за все треба розробити структурну схему програмного модуля, опираючись на яку, ми будемо крок за кроком створювати нашу програму.

Структурна схема — схема, яка визначає основні функціональні частини виробу, їх взаємозв'язки та призначення. Під функціональною частиною розуміють складову частину схеми: елемент, пристрій, функціональну групу, функціональну ланку.

Структурна схема призначена для відображення загальної структури пристрою, тобто його основних блоків, вузлів, частин та головних зв'язків між ними. Із структурної схеми повинно бути зрозуміло, навіщо потрібний даний пристрій і як він працює в основних режимах роботи, як взаємодіють його частини. Позначення елементів структурної схеми можуть обиратись довільно, хоча загальноприйнятих правил виконання схем слід дотримуватись.

Після того, як ми визначились з структурною схемою програмного модуля, можна починати процес програмування. Для початку створимо новий проект у

Visual Studio 2012. При створенні програми в Visual Studio спочатку створюється проект і рішення, для цього необхідно виконати наступні команди: Файл → Створити → Проект, крім того, можна ввести "New Project" у вікні Швидкий запуск, щоб виконати те ж саме діяння.

Виберіть шаблон додатки WPF Visual Basic або Visual C #, скориставшись, наприклад, шляхом по меню в лівій області Встановлено, Шаблони, Visual C #, Windows, а потім вибравши «Додаток WPF» у середній області. У нижній частині діалогового вікна нового проекту назвіть проект HelloWPFApp. На рисунку 1 наведено діалогове вікно створення нового проекту у Visual Studio.

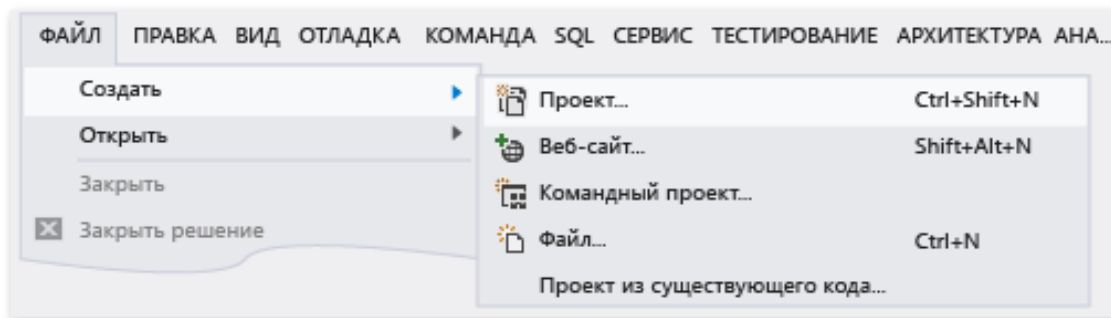


Рисунок 1. Вікно створення нового проекту у Visual Studio

Visual Studio створює рішення і проект HelloWPFApp, і Оглядач рішень показує різні файли. Конструктор WPF відображає уявлення конструювання та подання XAML файлу MainWindow.xaml в одному розділеному поданні. Зрушуючи роздільник, можна робити будь-яке з уявлень більше або менше. Можна вибрати для перегляду тільки візуальне уявлення або тільки уявлення XAML.

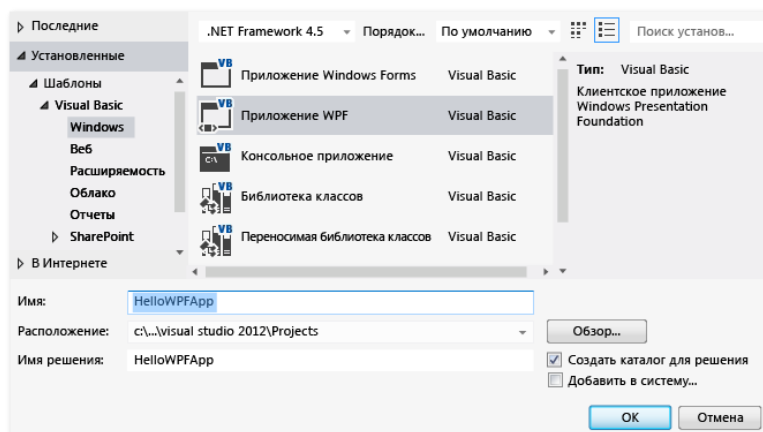


Рисунок 2. Вибір шаблону додатка

Після створення проекту його можна налаштовувати. За допомогою вікна Властивості (в меню Вид) можна відображати і змінювати параметри елементів проекту, елементів управління та інших елементів у додатку. За допомогою властивостей проекту і сторінок властивостей можна відображати і змінювати параметри проектів і рішень.

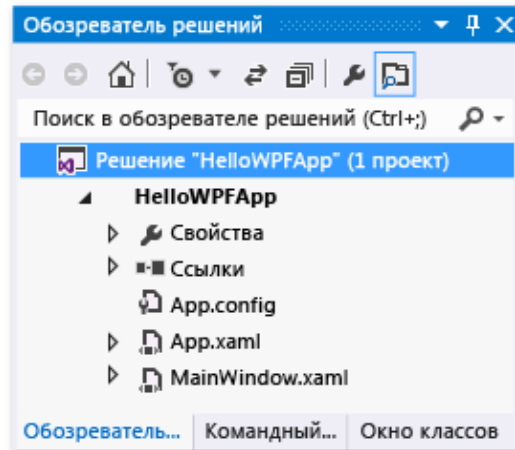


Рисунок 3. Элементы проекта

Після створення нового проекту можна переходити до додавання на форму різноманітних елементів управління з головної панелі програми, після чого отримуємо початкову форму для програмного модуля.

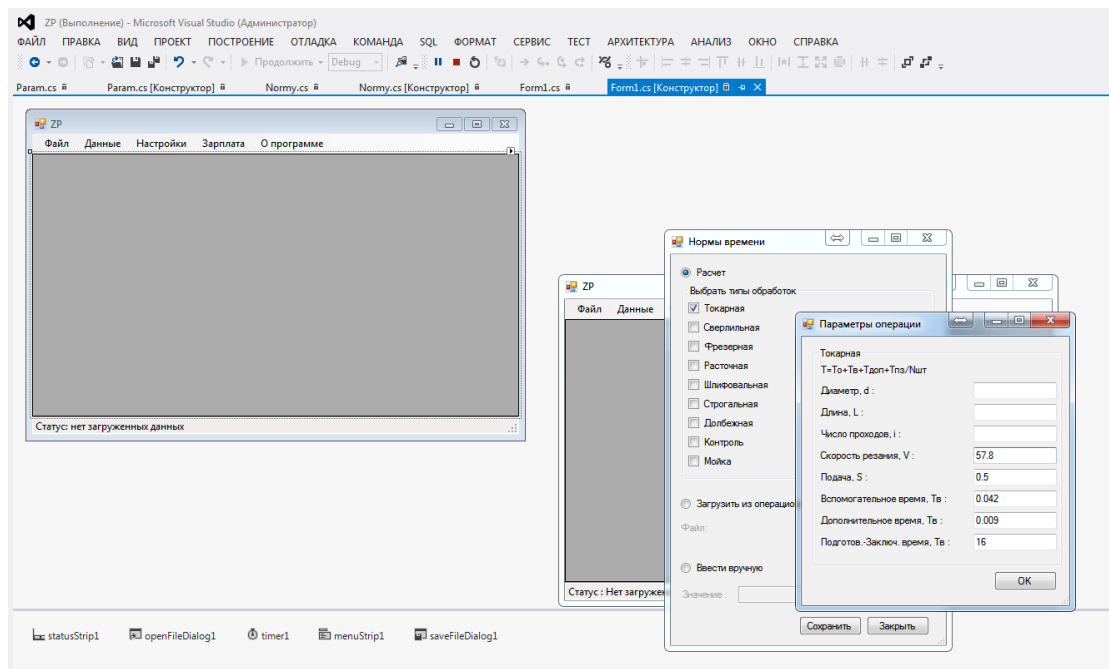


Рисунок 4. Створення головного вікна програми

Було створено декілька функціональних вікон, де будуть реалізовані розрахунки норм часу та заробітної плати, а також використані поля для введення усіх необхідних параметрів. Для того, щоб зв'язати між собою усі вищезазначені вікна, необхідно прописати це у файлі програмного кода.

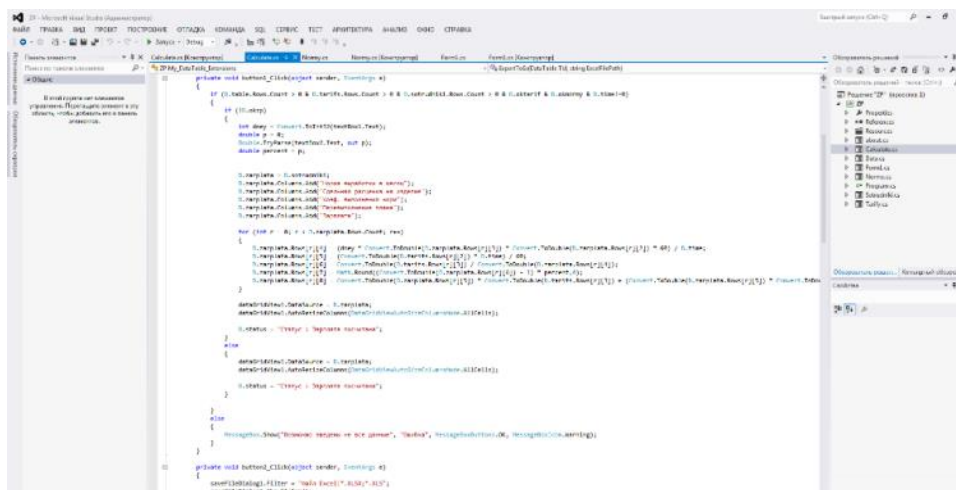


Рисунок 5. Файл з програмним кодом

На початку файлу йдуть директиви using після яких йдуть назви підключаються просторів імен. Простори імен являють собою організацію класів в загальні блоки. Наприклад, на першому рядку using System; підключається простір імен System, яке містить фундаментальні та базові класи платформи .NET. Фізично простору імен знаходяться в підключаються бібліотеках dll, які можна побачити у вікні Solution Explorer, відкривши вузол References.

Далі починається вже власне наш простір імен, яке буде створювати окрему збірку або виконувану програму: спочатку йде ключове слово namespace, після якого назва простору імен. За замовчуванням Visual Studio дає йому назву проекту. Далі всередині фігурних дужок йде блок простору імен. І так як C# має Сі-подібний синтаксис, кожен рядок завершується крапкою з комою, а кожен блок коду поміщається у фігурні дужки.

Тепер, після написання коду, йдемо в меню «Побудова» і вибираємо там пункт «Побудувати рішення ...» (або просто натискаємо F7). Якщо побудова пройде успішно (тобто немає помилок у вашій програмі), то у вікні виводу ви побачите відповідне повідомлення.

Після цього ми можемо запустити на виконання за допомогою клавіші F5 або з панелі інструментів, натиснувши на зелену стрілку. На рисунку 6 зображено підсумкове вікно програми, яке ми отримаємо після запуску проекту.

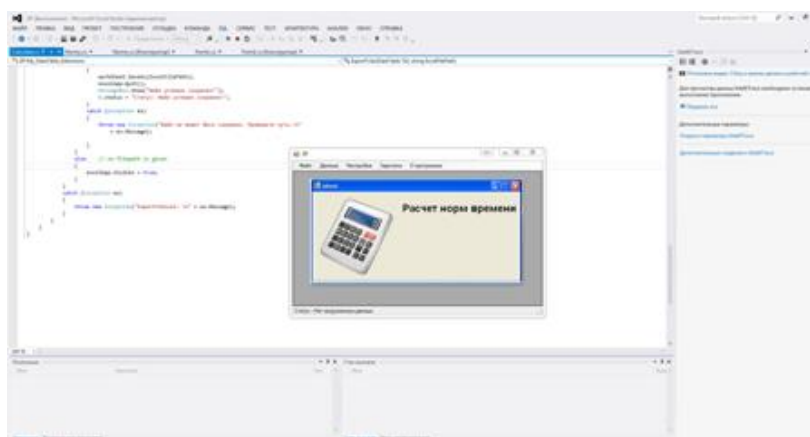


Рисунок 6. Підсумкове вікно програми

Висновки. В даний час існує великий вибір засобів розробки на платформі операційної системи Windows. Для розробки програмного модуля визначення норм часу та заробітної плати була обрана мова програмування C # та середовище розробки Visual Studio.

Microsoft Visual Studio надає кілька способів розробки додатків Windows, що працюють локально на комп'ютерах користувачів. Visual Studio дозволяє створювати додатки Windows і інтерфейси за допомогою Windows Forms. Можна створювати додатки служб Windows або програми Windows Win32. Починаючи з Visual Studio 2008, можна створювати додатки Windows за допомогою Windows Presentation Foundation (WPF).

Структурна схема призначена для відображення загальної структури пристрою, тобто його основних блоків, вузлів, частин та головних зв'язків між ними. Із структурної схеми повинно бути зрозуміло, навіщо потрібний даний пристрій і як він працює в основних режимах роботи, як взаємодіють його частини. Позначення елементів структурної схеми можуть обиратись довільно, хоча загальноприйнятих правил виконання схем слід дотримуватись.

Список використаних джерел:

1. Гавриленко Е.А., Холодняк Ю.В., Антонова Г.В., Чаплинский А.П. Разработка алгоритма программного обеспечения для формирования обводов по заданным геометрическим условиям. Праці Таврійського державного агротехнологічного університету імені Дмитра Моторного. – Мелітополь: ТДАТУ ім. Д. Моторного, 2020. – Вип. 20, т. 3. (с.293-303).-DOI: 10.31388/2078-0877-2020-20-3-293-303.

2. Мацулевич О.Є., Щербина В.М., Бондаренко Л.Ю., Малюта С.І., Антонова Г.В. Програмне забезпечення для автоматизованого визначення параметрів різального інструменту фрезерної обробки корпусних деталей. *Праці Таврійського державного агротехнологічного університету*. Мелітополь: ТДАТУ, 2020. Вип. 20, т. 3. С. 275-281.

3. Інженерна та комп'ютерна графіка. Частина 1: навчальний посібник / В.М. Щербина, О.Є. Мацулевич, Є.А. Гавриленко, Ю.В. Холодняк, О.В. Івженко, І.В. Пихтєєва, О.О. Вершков, С.В. Галько, А.П. Чаплінський. Мелітополь: Люкс, 2020. 238 с.

4. Проектування керуючих програм для верстатів токарної групи з пристроєм числового програмного керування: навчальний посібник / Дмитрієв Ю.О., Мацулевич О.Є., Щербина В.М., Холодняк Ю.В. Мелітополь, Люкс, 2018. 132 с.

5. Теоретична механіка. Динаміка матеріальної точки: навчально – методичний посібник / О.М. Леженкін, Г.В. Антонова, О.О. Вершков, Л.Ю. Бондаренко, О.Є. Мацулевич, А.О. Смілов, О.Ю. Михайленко. Мелітополь: Люкс, 2021. 160с.

6. Інженерна механіка: практикум / Г.В. Антонова, О.Є. Мацулевич, О.Ю. Михайленко, І.В. Пихтєєва, О.В. Івженко, Ю.В. Холодняк, В.М. Щербина, Ю.О. Дмитрієв. Мелітополь: Люкс, 2021. 147 с.

7. Нарисна геометрія та креслення: навчально–методичний посібник для підготовки бакалаврів зі спеціальностей 131 “Прикладна механіка” та 133 “Галузеве машинобудування” / Івженко О.В., Пихтєєва І.В., Гавриленко Є.А., Мацулевич О.Є., Щербина В.М., Холодняк Ю.В., Бондаренко Л.Ю., Михайленко О.Ю. Мелітополь: Люкс, 2020. 217 с.

8. Програмування автоматизованих процесів обробки деталей: навчально-методичний посібник. Лабораторний практикум /Ю.О. Дмитрієв, О.Є. Мацулевич, Є.А. Гавриленко, Ю.В. Холодняк, Г.В. Антонова – Мелітополь (протокол №8 від 28.04.2022р.), 2022.–170с.

9. Мацулевич О.Є., Вершков О.О., Холодняк Ю.В., Дмитрієв Ю.О., Чаплінський А.П. Розробка мурашиного алгоритму для оптимізації оперативного планування робіт по збиранню врожаю кісточкових. *Плодовий сад – новітнє в теорії та практиці*: матеріали V Всеукр. наук.-практ. інтернет-конференції. Мелітополь: ТДАТУ, 2021. С. 106-110.

10. Холодняк Ю.В., Гавриленко Е.А., Ивженко А.В., Найдыш А.В. Моделирование участка пространственной монотонной кривой линии /Сучасні проблеми моделювання: наукове фахове видання. – Мелітополь: МДПУ ім. Б. Хмельницького, 2020. Вып.17. С. 131-137.

11. Вершков О.О., Мацулевич Ю.О. Визначення шорсткості поверхонь із застосуванням програмного забезпечення COPYCAD ф. DELCAM plc // Сучасні комп'ютерні та інформаційні системи і технології, Матеріали і всеукраїнської науково-практичної інтернет-конференції, Мелітополь 7-25 грудня 2020р. С. 17-23

12. Спірінцев В.В., Мацулевич О.Є., Холодник Ю.В, Чаплінський А.П. Застосування графічного редактора ARCHICAD при вивченні дисципліни «Комп'ютерне проектування простору інженерних споруд» *Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації*: матеріали Міжнар. наук.-практ. інтернет-конф. Мелітополь: ТДАТУ, 2020. С. 262-266.

13. Вершков О.О., Мацулевич Ю.О. Визначення шорсткості поверхонь із застосуванням програмного забезпечення COPYCAD ф. DELCAM plc // Сучасні комп'ютерні та інформаційні системи і технології, Матеріали і всеукраїнської науково-практичної інтернет-конференції, Мелітополь 7-25 грудня 2020р. С. 17-23

14. Спірінцев В.В., Мацулевич О.Є., Холодник Ю.В, Чаплінський А.П. Застосування графічного редактора ARCHICAD при вивченні дисципліни «Комп'ютерне проектування простору інженерних споруд» *Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації*: матеріали Міжнар. наук.-практ. інтернет-конф. Мелітополь: ТДАТУ, 2020. С. 262-266.

УДК 004.052.32

РОЗРОБКА ІНФОРМАЦІЙНО-ТЕСТУВАЛЬНОЇ СИСТЕМИ ДЛЯ ОЦІНЮВАННЯ СТУДЕНТІВ У ЗВО

Лубко Д.В., к.т.н.

e-mail: dmytro.lubko@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Інтеграція до Європейського простору вищої освіти та запровадження Європейської кредитно-трансферної та акумулюючої системи (ECTS) передбачає введення кредитно-модульної системи формування навчальних програм; посилення ролі самостійної роботи студентів, використання новітніх педагогічних методик та сучасних інформаційних технологій навчання [1]. Під час впровадження кредитно-модульної системи навчання, яка передбачає зменшення частки аудиторного навантаження та відповідне збільшення обсягу самостійної роботи, особливого значення набуває проблема організації та контролю самостійної роботи студентів.

Існуюча система контролю і оцінки якості навчальних досягнень студентів відрізняється високою мірою консерватизму оскільки все ще зберігає такі риси як [2]:

- суб'єктивізм оцінок викладачів, пов'язаний з прийнятими у ВНЗ традиційними формами, методами і засобами визначення навчальних досягнень студентів;
- відсутність загальних критеріїв оцінювання, стандартизованих засобів і єдиних шкал призводить до результатів, які не можна порівнювати, навіть при рівних інших умовах;
- низьку ефективність контрольно-оцінних систем розроблених внутрішньо ВНЗ, без урахування досягнень теорії педагогічних вимірів і сучасних інформаційних технологій;
- слабе використання методів самоконтролю, самокорекції і самооцінки результатів навчальної діяльності.

Також можуть виникати труднощі, пов'язані з особливостями викладацької роботи [2]:

- досить часто проявляються неспівпадання вимог різних викладачів, відмінності в їх рівні суворості при оцінці однієї і тієї ж відповіді;
- відмінність в професійній кваліфікації;
- при організації поточних перевірок знань великої кількості студентів, коли оцінювання проводиться, головним чином, лише по формальним критеріям, спостерігається завантаженість викладача рутинною мало творчою роботою, пов'язаною з великим об'ємом інформації, яку вимагається підготувати, обробити і проаналізувати за відносно короткий проміжок часу;
- можлива упередженість викладача (з психологічних чи інших причин) при оцінці відповідей деяких студентів;
- іноді оцінки, що виставляються студентам, виявляються неадекватними за причин побоювання викладача, що вони будуть використані для оцінювання роботи самого викладача.

Отже, традиційний сесійно-екзаменаційний контроль навчальних досягнень студентів, якщо і стимулює академічну активність студентів і дозволяє здійснювати відбір найбільш здатних до навчання, але не вирішує завдань

забезпечення якості підготовки фахівців відповідно до критеріїв професіоналізму і конкурентоспроможності, які висуває ділове середовище сучасного суспільства знань.

На сучасному етапі при оцінці знань студентів перераховані проблеми більшою мірою вирішуються використанням такої форми контролю, як тестування. Також однією з необхідних умов для запровадження кредитно-модульної системи організації навчального процесу є створення системи тестування та рейтингового оцінювання знань студентів.

Саме все вищезгадане і визначає актуальність теми даної статті.

Основні матеріали дослідження. Безумовно, контроль, перевірка й оцінка результатів навчання – невід'ємні елементи навчально-виховного процесу, без яких неможливо уявити певний цикл педагогічної взаємодії між студентом і викладачем. Найбільш економною формою контролю, а також об'єктивним показником засвоєння навчального матеріалу є тести.

До загальновизнаних переваг системи тестових завдань належать [3]: оперативність контролю і економія часу, що витрачається на перевірку рівня знань студентів; можливість одночасного тестування великої кількості студентів; контроль широкого діапазону набутих знань; перевірки результативності самостійної роботи студентів з теми або розділу навчального курсу; забезпечення індивідуального підходу; більш об'єктивна оцінка набутих знань, умінь, навичок і уникнення при цьому можливого суб'єктивізму з боку викладача; можливість швидкого інформування студентів про результати тестування; використання ПК, що істотно підвищує ефективність тестового контролю.

Для забезпечення якісного та глибокого процесу навчання студентів, нами було вирішено розробити та спроектувати програмну систему для вивчення певної дисципліни, при викладанні для студентів ВНЗ спеціальності „Інформатика”, яка також би мала окремий тестувальний блок.

При проектуванні цієї системи були проаналізовані схожі аналоги. Було виявлено, що вони всі мають дещо схожу будову. А саме, мають однотипні меню, як то: лекції, практичні або лабораторні роботи, література і майже все.

Ми, при проектуванні системи, вирішили розширити комплекс матеріалів для вивчення, так як це дасть змогу значно покращити якість та глибину засвоєння навчального матеріалу студентами. Було запропоновано включити в систему такий перелік матеріалів (пункти меню системи): матеріали для лекцій, матеріали для лабораторних, матеріали для практичних робіт, розподіл балів (для зрозуміння кількості балів по видам робіт), шкала оцінювання знань (у ECTS), система оцінювання (загальна інформація), матеріали для самостійної роботи, питання до екзамену, питання до підсумкового модульного контролю, кафедра та її становлення (для ознайомлення), тестувальний блок, навчальна література, місцезнаходження (на карті ВНЗ).

При проектуванні системи вирішено було додати в нього додатковий тестувальний модуль (блоку), який дозволяє проводити тестування з даної дисципліни та оцінювати їх результати. Це є одною з найважливіших переваг розробленої системи.

До важливих переваг тестового контролю слід віднести [3]:

- більшу об'єктивність і, як наслідок, більше позитивний стимулюючий вплив на пізнавальну діяльність студента;
- виключається вплив негативного впливу на результати тестування таких факторів як настрої, рівень кваліфікації та ін. характеристики викладача;

- орієнтованість на сучасні технічні засоби на використання в середовищі комп'ютерних (автоматизованих) навчальних систем;
- універсальність, охоплення всіх стадій процесу навчання;
- підвищення швидкості перевірки якості засвоєння знань і вмінь учнями;
- можливість математико-статистичної обробки результатів контролю, і як наслідок, підвищення ефективності педагогічного контролю;
- можливість збільшення частоти та регулярності контролю за рахунок зменшення часу виконання завдань і автоматизації перевірки.

Після проведеного попереднього теоретичного аналізу проблеми, було виконано проектування інформаційно-тестувальної оцінювальної системи. При реалізації системи - у якості дисципліни, яка викладається для студентів у нашому ВНЗ, було обрана дисципліна - „Проектування розподілених та паралельних систем” (ПРПС), яка трансформувалась надалі у «Технології розподілених систем та паралельних обчислень». Це базова та важлива дисципліна у курсі програмування для майбутніх фахівців спеціальності 122 «Комп'ютерні науки».

Проектування системи проводилося з використанням мови програмування Delphi [4, 5]. При програмній реалізації тестувального модулю інформаційно-тестувальної оцінювальної системи, використовувались продукційні правила у вигляді зв'язків (умовних операторів) „IF-THEN”. Саме тому, враховуючи наявність у коді ядра системи даних, які реалізуються за допомогою продукційних правил, було вирішено назвати систему як «експертна система», що і відобразилося у її назві на скріншотах інтерфейсу нижче [6, 7, 8].

Розглянемо загальний інтерфейс системи, яка була спроектована (рис. 1).

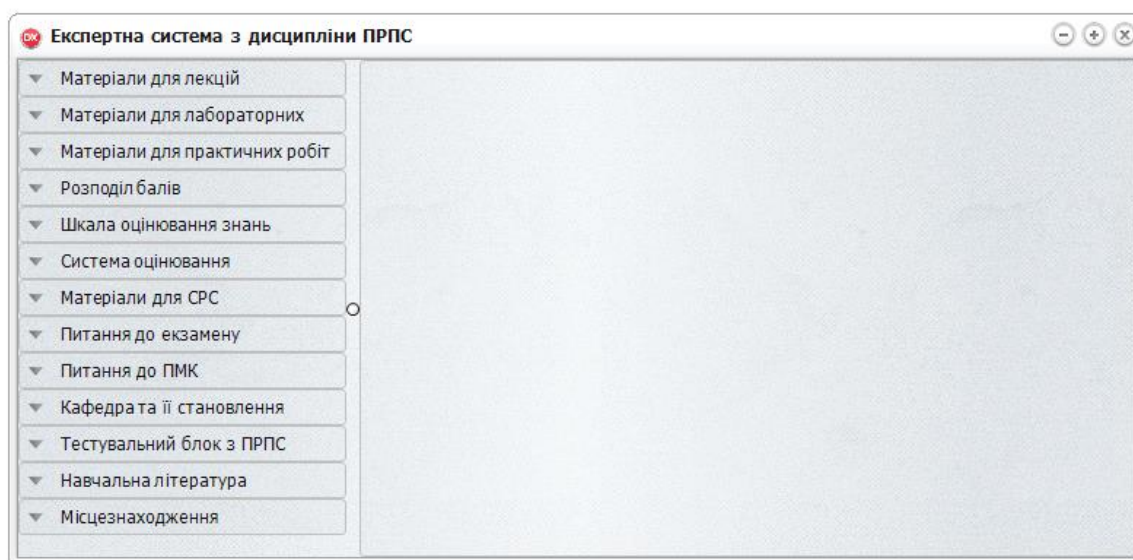


Рисунок 1. Головний інтерфейс розробленої системи

Головний інтерфейс системи складається з однієї загальної форми, на якій зліва розташовані меню (підрозділи) системи (рис. 1). З правої частини розташоване поле для виведення поточних даних (інформації). Рядок заголовку містить назву системи – „Експертна система з дисципліни ПРПС” (рис. 1).

Наведемо перелік меню системи з поясненнями (рис. 1). Воно містить:

1. Меню „Матеріали для лекцій”. Міститься лекційна база (рис. 2).
2. Меню „Матеріали для лабораторних”. Міститься база лабораторних робіт з дисципліни (рис. 2).

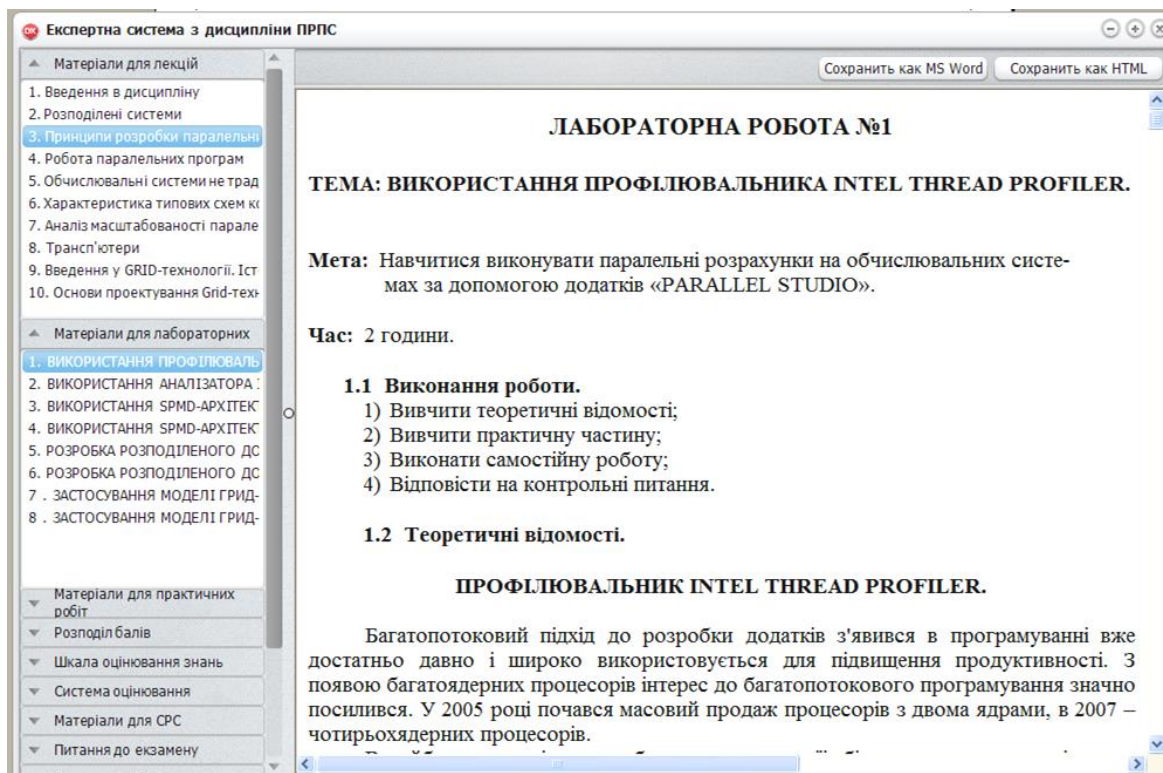


Рисунок 2. Розкриті меню „Матеріали для лекцій” та „Матеріали для лабораторних” (з відкритим правим діалоговим вікном)

3. Меню „Матеріали для практичних робіт”. Міститься база практичних робіт з дисципліни.

4. Меню „Розподіл балів”. Міститься розподіл балів, які отримують студенти при вивченні дисципліни.

5. Меню „Шкала оцінювання знань”. Міститься інформація про структуру оцінювання знань студентів у системі ECTS.

6. Меню „Система оцінювання”. Міститься загальна інформація про прийняту шкалу оцінювання та зарахування, перезарахування кредитів у системі ECTS, а також перелік посилань та частини нормативних документів для зрозуміння оцінювання у системі ECTS.

7. Меню „Матеріали для СРС”. Містяться методичні вказівки для виконання самостійної роботи студентів.

8. Меню „Питання до екзамену”. Міститься перелік питань, які виносяться на іспит для студентів при вивченні даної дисципліни (рис. 3).

9. Меню „Питання до ПМК”. Міститься перелік питань, які виносяться до ПМК-1 та ПМК-2 при вивченні даної дисципліни.

10. Меню „Кафедра та її становлення”. Міститься інформація про становлення кафедри в університеті, її наукову та методичну роботи за останні роки та штатний перелік осіб кафедри – професорсько-викладацький та лаборантсько-інженерний склад на сьогодні.

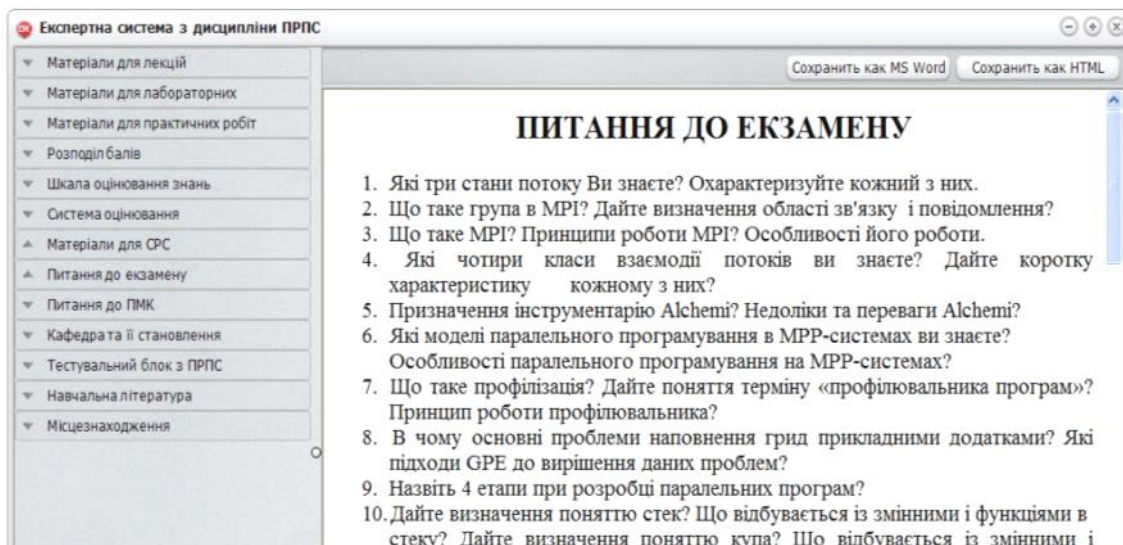


Рисунок 3. Розкрите меню „Питання до екзамену”
(з діалоговим вікном)

11. Меню „Тестувальний блок з ПРПС”. Містяться тести по 1 блоку (ПМК-1) та 2 блоку (ПМК-2) для оцінювання знань студентів (рис. 4 та 5).

12. Меню „Навчальна література”. Міститься перелік основної та додаткової навчальної літератури для викладання дисципліни ПРПС.

13. Меню „Місцезнаходження”. Міститься карта розташування та контактні дані університету.

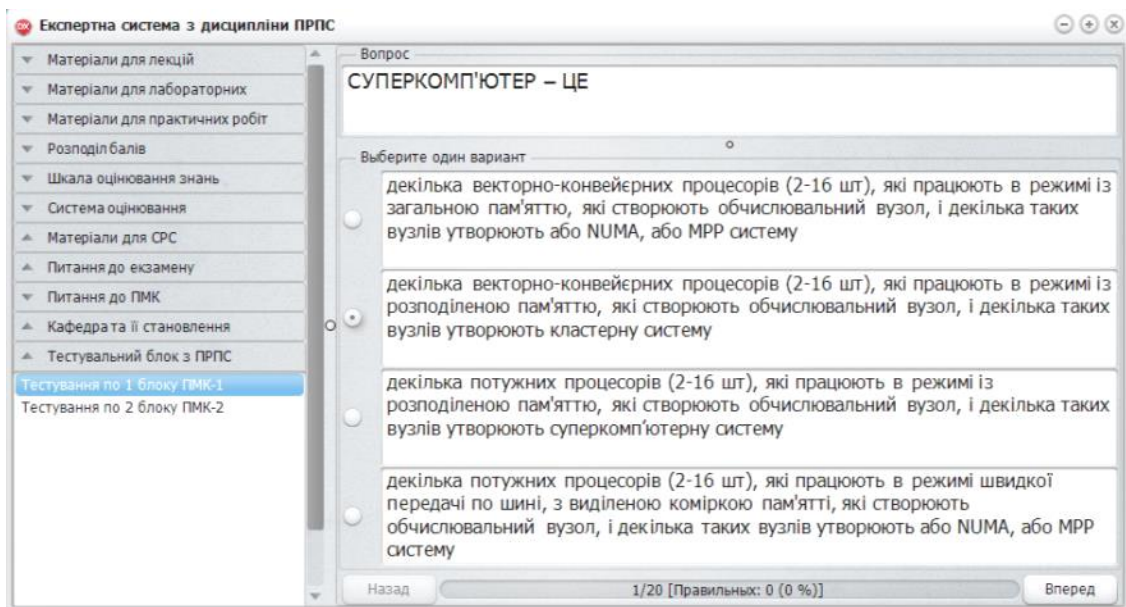


Рисунок 4. Розкрите меню „Тестувальний блок з ПРПС”
(з відкритим вікном тестів)

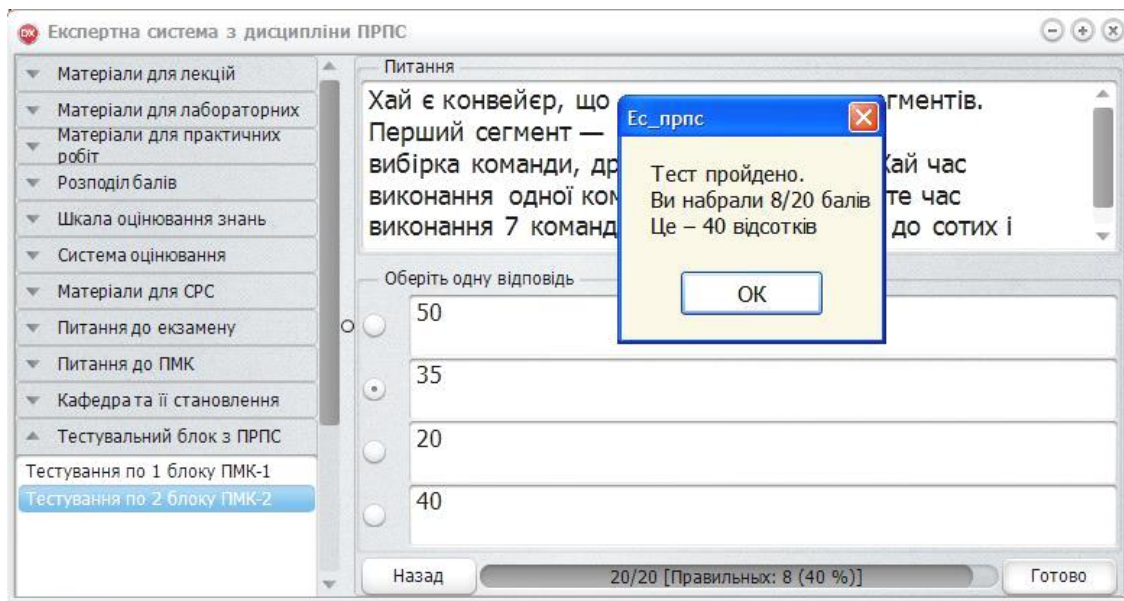


Рисунок 5. Розкрите меню „Тестувальний блок з ПРПС”
(з вікном повідомлення про результати тестування студентом)

Висновок. Процес контролю це одна з найбільш трудомістких і відповідальних операцій, пов'язана з гострими психологічними ситуаціями як для студентів, так і для викладача. З іншого боку його правильна постановка сприяє поліпшенню якості підготовки фахівців. Систематичне проведення контрольних заходів за допомогою складених на вищому рівні інструментів контролю дозволяє ВНЗ формувати висококласних фахівців в різних галузях знань готових застосовувати накопичений багаж знань у будь-який час. Саме у таких фахівцях і нині, і в майбутньому має потребу наша країна

Розроблена інформаційно-тестувальна оцінювальна система має деякі переваги перед іншими системами такого типу.

А саме: вона безкоштовна, має простий та зручний інтерфейс; дозволяє якісно та поглиблено вивчити дисципліну - „Проектування розподілених та паралельних систем”. Дана система використовує майже всі види навчання та контролю знань студентів, та відрізняється від інших схожих систем наявністю тестового модуля для перевірки знань. Створену систему можна розширювати, як функціонально (інтерфейсно), так і програмно (наприклад, збільшити об'єм та кількість усіх видів робіт, у тому числі і тесту вального блоку). Як перспектива для розвитку, є можливість написання перетворення та переорієнтація системи для будь якою іншої дисципліни ВНЗ.

Використання при навчанні певної дисципліни саме тестування для студентів - дуже допомагає об'єктивно проводити наприклад атестацію ВНЗ, яка проводиться з метою встановлення відповідності змісту, рівня і якості підготовки випускників вимогам державних освітніх стандартів. А планове та систематичне проведення контрольних заходів за допомогою складених інструментів контролю (наприклад, як у розробленій системі) дозволяє ВНЗ формувати висококласних фахівців у різних галузях знань, готових застосовувати накопичений багаж знань у будь-яку хвилину. Саме такі фахівці і потрібні країні сьогодні.

Список використаних джерел:

1. Шимкова І.В. Використання автоматизованого тестового контролю знань для організації самостійної навчально-пізнавальної діяльності студентів. Збірник наукових праць «Педагогічні науки», №46, 2007. С. 408-411.
2. Швець Д.Є. Тестування як ефективна форма контролю та підвищення якості знань. Гуманітарний вісник Запорізької державної інженерної академії, №41, 2010. С. 169-177.
3. Танько Т.П. Тестовий контроль знань студентів як умова модернізації освіти в Україні. Витоки педагогічної майстерності. Серія: Педагогічні науки, №7, 2010. С. 14-18
4. Осипов Д. Delphi. Профессиональное программирование. Издательство: Символ-Плюс. 2006. - 1056 с.
5. Іванків К.С., Мацьонг Я. Є., Мандзак Т. І. Delphi для початківців. Видавництво: Львів, ЛНУ. 2006. - 188 с.
6. Lubko D.V., Zolotukhin Y.O. Розробка експертної системи для вивчення дисципліни «Проектування інформаційних систем». Ukrainian Journal of Educational Studies and Information Technology, Volume 5(1), 2017, p. 182-186.
7. Лубко Д.В. Проектування довідкової інтелектуальної експертної системи для вівчарства у приватних господарствах країни. Ukrainian Journal of Educational Studies and Information Technology. 2017. Vol.5, №3. – pp. 1 – 18.
8. Shevchuk, O. (2016). Педагогічні принципи проектування та розробки експертних систем навчання. Наукові записки Тернопільського національного педагогічного університету імені Володимира Гнатюка. Серія: педагогіка, Том 1, С. 38-43.

УДК 004.891.2

РОЗПІЗНАВАННЯ ФІГУР РУКОПИСНИХ ДІАГРАМ ТА СХЕМ ЗА ДОПОМОГОЮ ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ

Українець М.О.,

Вакалюк Т.А., д. пед. н., професор

Державний університет «Житомирська політехніка»

email: ukraine45@gmail.com

email: kik_vta@ztu.edu.ua

Актуальність та постановка проблеми. Будь-які діаграми чи схеми складаються з двох основних компонентів: геометричних фігур та написів. Геометричні фігури використовуються для зображення різних сутностей, процесів, активностей. Наприклад, прямокутник на блок-схемі позначає процес, ромб – перевірку істинності певної умови, а паралелограм – введення інформації. В залежності від типу діаграми чи схеми, одні і ті самі геометричні фігури мають різні значення, проте зображуються однаково.

Для пояснення взаємодій між сутностями та порядку взаємодії між ними використовуються різного роду текстові написи, зроблені користувачем або визначені специфікою діаграми.

Для реалізації розпізнавання рукописних фігур пропонується використати згорткову нейронну мережу (Convolutional Neural Network або CNN). Кожен її шар використовує власне перетворення. Якщо на перших шарах мережа оперує такими поняттями як «ребра», «грані», то далі використовується поняття «текстура», «частини об'єктів». В результаті роботи цих шарів можливо правильно класифікувати зображення або виділити на ньому потрібний об'єкт.

Основні матеріали дослідження. В основі роботи згорткових нейронних мереж лежить різновид багатошарових перцептронів, розроблений так, щоб вимагати мінімум попередньої обробки від вхідного зображення. Це означає, що згорткова мережа навчається за допомогою фільтрів, які для інших мереж треба задавати вручну. Тому, на відміну від нейронних мереж, які базуються на методі зворотного поширення помилки, згорткові нейронні мережі потребують меншого втручання людини при її навчанні. Завдяки відсутності втручання людини згорткова мережа може приймати зображення на вхід і автоматично визначати особливості зображення для подальшого розпізнавання.

Для згорткових нейронних мереж притаманні такі особливості. Перша з них це те, що зв'язок між двома шарами нейронів здійснюється через ядро згортки, а не через повний зв'язок. Таким чином, шар згортки, підключений до вхідного зображення, є локальним зв'язком, створеним для блоків пікселів, а не традиційним повним з'єднанням на основі пікселів. Друга особливість полягає в тому, що вагові параметри ядра згортки є спільними на одному рівні. Ці дві особливості зменшують кількість параметрів глибинної мережі, зменшують складність моделі та прискорюють швидкість навчання. [1]

Існуючі методи розпізнавання діаграм можна поділити на дві групи: за контуром та за об'єктами. Методи, які базуються на контурах, припускають, що контури фігур можуть бути встановлені на етапі попередньої обробки. Для цього часто проводять процедуру бінаризації, яка полягає в конвертації багатотонального зображення в двотональне. Після цього проводиться видалення шумів та заповнення отриманих контурів кольором. Наступним кроком є робота алгоритму визначення контурів, результат роботи якого, це визначене розташування зображених блоків. [2]

Для методів, які базуються на об'єктах застосовують згорткові нейронні мережі. Існують детектори об'єктів побудовані на прив'язках. Вони

встановлюють блоки прив'язки в кожній позиції карти об'єктів, а мережа передбачає ймовірність наявності об'єктів у кожному блоку прив'язки та регулює їх розмір відповідно до розмірів об'єкта. Зазвичай, потрібно ретельно розраховувати розміри блоків прив'язки, для того щоб вони відповідали різним об'єктам. Проте існують детектори, які не потребують блоків прив'язки, їх називають детектори без прив'язки. Замість блоків прив'язки, вони передбачають ключові точки на зображенні та групують їх для створення обмежувальних рамок. [3]. Наразі застосовують три методи для тренування нейронних мереж: навчання з вчителем (supervised learning), навчання без вчителя (unsupervised), навчання з підкріпленням (reinforcement learning).

Під час навчання зі вчителем, нейронній мережі на вхід подається навчальна пара, яка складається з вхідного вектора та вектора вихідних значень. Під час навчання нейронній мережі по черзі подаються навчальні пари, після чого вираховується похибка між дійсним та бажаним значенням і відбувається корегування параметрів для зменшення цієї похибки.

В свою чергу, під час навчання без вчителя, нейронна мережа намагається навчитися виконувати поставлене завдання без зовнішнього втручання. Зазвичай, такий спосіб навчання підходить для визначення внутрішніх взаємозв'язків та залежностей в певній множині об'єктів. Для реалізації навчання з підкріпленням замість «вчителя» вводиться блок «критика», завдання якого, відслідковувати реакцію мережі на вхідні дані і опираючись на цю реакцію, визначає евристичну похибку, яка покладена в процес навчання. [4]

Висновок. Розглянуто наявні методи розпізнавання фігур рукописних діаграм та схем. Для навчання згорткової нейронної мережі можна застосувати навчання з учителем. Таким чином, під час навчання мережа вивчатиме та виділятиме характеристики вхідного зображення і надаватиме класифікацію зображеної фігури на виході. Під час навчання з учителем згорткова нейронна мережа рахуватиме різницю між прогнозованими і фактичними значеннями для постійного коригування параметрів мережі. Це дозволить правильно класифікувати всі види вхідних зображень. Проте для коректного навчання слід обрати правильний набір даних, який дозволить мережі правильно класифікувати зображені рукописні геометричні фігури.

Список використаних джерел:

1. Zhang, L. (2021) Hand-drawn sketch recognition with a double-channel Convolutional Neural Network - EURASIP Journal on advances in Signal Processing, SpringerOpen. Springer International Publishing. Available at: <https://asp-eurasipjournals.springeropen.com/articles/10.1186/s13634-021-00752-4> (Accessed: November 25, 2022).
2. Rastogi, S. (2021) Enclosed shape extraction from hand-drawn images, Analytics Vidhya. Available at: <https://www.analyticsvidhya.com/blog/2021/05/enclosed-shape-extraction-from-hand-drawn-images/> (Accessed: November 25, 2022).
3. Fang, J., Feng, Z. and Cai, B. (2022) Drawnnet: Offline hand-drawn diagram recognition based on keypoint prediction of aggregating geometric characteristics, Entropy (Basel, Switzerland). U.S. National Library of Medicine. Available at: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8947756/> (Accessed: November 25, 2022).
4. Artificial Neural Network - Quick Guide (no date) Tutorials Point. Available at: https://www.tutorialspoint.com/artificial_neural_network/artificial_neural_network_quick_guide.htm (Accessed: November 25, 2022).

UDC 004.891.2

HANDWRITTEN DIAGRAMS AND SCHEMES OF TEXT RECOGNITION USING CONVOLUTIONAL NEURAL NETWORKS

Ukrainets M.O.
Vakaliuk T.A., Dr.Sc, professor
Zhytomyr Polytechnic State University

email: ukrainec45@gmail.com
email: kik_vta@ztu.edu.ua

Relevance and statement of the problem. Diagrams and schemes usually consist of two main components that transform a simple drawing into a valuable explanation of processes or interactions between different entities. These components are shapes and texts. Shapes usually represent different entities, arrows show the way of interaction between entities, and texts are used for describing details that help to understand processes that take place between entities more clearly. Texts play a very important role in different kinds of visualizations. Texts are used for naming entities, description of interactions, and giving explanations and details.

Handwriting recognition is a complex task for the computer system. Many factors make handwriting recognition complicated. Firstly, there is huge variability and ambiguity of strokes from person to person. Every person has unique handwriting, it is hard to find two people with identical handwriting. However, even when we have one person and there is a need to recognize their handwriting it will not be easy because handwriting tends to vary and depends on different factors like mood, situation, and even the tool used for writing. Another difficulty of handwriting recognition is that people tend to write letters from a different angle. So cursive writing makes the separation and recognition of characters challenging.

To create an efficient handwriting system that will recognize text from hand-drawn visualizations, the convolutional neural network (CNN) can be used.

The main research materials. CNN is a deep learning technique that is used to classify visual input. Most deep learning methods use a neural network architecture, often called deep neural networks. Unlike traditional neural networks, which typically contain 2-3 hidden layers, deep neural networks can contain up to 150 hidden layers.

CNN can directly take the image as input and automatically extract image features for recognition. The weight sharing and local sensing characteristics of CNN allow reducing the number of parameters in the network. This also makes CNN work similarly to the animal visual nerve cells. Because of that, the recognition accuracy and efficiency of the network are greatly improved. [1]

A convolutional neural network consists of different types of layers. The first one is the input layer. It is used for giving the network the input image, which contains the handwritten text. Another layer that takes place in CNN architecture is the convolution layer, where the most of computational work tends to happen. This layer consists of a set of learnable filters known as parameters of the convolution layer. Between the convolution layers of CNN, the pooling layers can be found. Their main purpose is to reduce the spatial size of computational work in the network. It is achieved by reducing the dimension of the input data by combining the outputs of the neuron clusters. They operate independently. That way, the system can achieve the intended outputs. In addition, CNN contains a fully connected layer. All the neurons in this layer are fully connected to all activations in the prevision layer.

To make it possible for CNN to recognize the handwriting on the image preprocessing is required. This step is needed for almost all kinds of recognition and

helps to improve the quality of recognition. The preprocessing of the image with handwriting implements the following steps:

- noise removal – the noise of different types can contaminate the image and reduce its quality;
- binarization – used to convert an image to black and white to significantly reduce the amount of information contained within the original image.
- normalization – the main purpose is to convert into a range of pixel values that are more familiar to the senses. Normalization involves converting images into a standard size.
- skew correction and thinning – help to get a single-pixel width to allow easy character recognition.

After the input image is preprocessed, the segmentation process begins. Segmentation involves separating individual characters from the image. The result of segmentation is a set of all letters and symbols that are present in the input image. The next step after segmentation is feature extraction. During this stage, special attributes are extracted from the segmented characters. These attributes are height, number of horizontal lines, widths of the character, number of circles, and pixels. Then the neural network can recognize characters from the input. [2]

To make CNN recognize handwritten texts the appropriate dataset should be used for the model training. The most suitable dataset for handwritten text recognition would be MNIST (Modified National Institute of Standards and Technology database) or EMNIST (Extended MNIST database). It consists training set of 60000 examples of handwritten characters and a testing set of 10000 examples. [3]

Conclusion. The process of handwriting text recognition with using of CNN was considered. The CNN features and the role of different kinds of layers that are used in CNN architecture were described. To recognize handwritten characters an appropriate preprocessing should be implemented. It will allow optimizing input data and improving the quality of recognition significantly. In addition, the main stages of handwriting text recognition were described. To train CNN to recognize characters from input images MNIST or EMNIST datasets can be used.

References:

1. Zhang, L. (2021) Hand-drawn sketch recognition with a double-channel Convolutional Neural Network - EURASIP Journal on advances in Signal Processing, SpringerOpen. Springer International Publishing. Available at: <https://asp-earasipjournals.springeropen.com/articles/10.1186/s13634-021-00752-4> (Accessed: December 6, 2022).
2. Handwriting recognition using artificial intelligence (no date). Available at: https://thesai.org/Downloads/Volume11No7/Paper_19-Handwriting_Recognition_using_Artificial_Intelligence.pdf (Accessed: December 6, 2022).
3. Papers with code - mnist dataset (no date) Dataset | Papers With Code. Available at: <https://paperswithcode.com/dataset/mnist> (Accessed: December 7, 2022).

УДК 004.9

МЕДИЧНІ ІНФОРМАЦІЙНІ СИСТЕМИ. ОГЛЯД ТА ПОРІВНЯННЯ

Коломоець Д.А., магістрант

e-mail: dany1a20011@gmail.com

Науковий керівник: к.т.н., доцент Лубко Д.В.

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Інформаційні технології в світовій системі охорони здоров'я. Застосування інформаційних технологій (ІТ) в системі охорони здоров'я набуває щораз більшого значення. Медична інформатика у поєднанні з організаційними змінами в закладах охорони здоров'я може сприяти наданню якіснішої медичної допомоги, одночасно зменшуючи фінансові видатки. Особливого значення набуває впровадження Медичних інформаційних систем (МІС), які допомагають автоматизувати всі аспекти діяльності лікувальної установи. Вони не лише забезпечують зберігання інформації в Електронній карті пацієнтів (ЕКП), але й допомагають здійснювати всю організаційну роботу установи [1].

В процесі впровадження інформаційних систем особливої уваги набувають задачі надійності збереження інформації про пацієнтів, швидкого доступу до даних, можливості взаємо обміну інформацією між різними лікувальними установами та проведення статистичного аналізу зведених даних. В країнах Європи електронна історія хвороби вже на 50-90% замінила паперовий варіант документації, а в США – на 70%. Економія часу медичного персоналу на ведення документації в електронному вигляді становить 63,4% [6].

Наукові дослідження, спрямовані на оцінку економічної ефективності від впровадження МІС, свідчать, що час, витрачений лікарем на кожну виписку з паперової історії хвороби, займає приблизно 4 хвилини, що в середньому за рік складає 384 години робочого часу. Суттєво знижуються затрати на пошук даних попередніх обстежень пацієнта. Згідно досліджень, 15% ресурсів лікувальних закладів витрачається на знаходження та збір попередніх даних. Лікарі і медсестри витрачають 25% свого часу на пошук необхідної їм інформації.

Важливою перевагою застосування МІС є можливість зменшення кількості медичних помилок. Згідно звіту Американського інституту медицини, в США щорічно близько 44000 смертей є причиною медичних помилок [5]. За його даними, електронна історія хвороби, яка забезпечує збір, обробку та ефективне використання медичної інформації, дозволяє не лише підняти ефективність надання медичної допомоги і знизити вартість лікування, але й зменшити кількість медичних помилок.

В Україні протягом останніх п'яти років спостерігається незначна позитивна динаміка розвитку ринку медичних інформаційних систем. Слід відзначити, що цей ринок є ризикованим, затрати на розробку сучасних систем значно переважають економічну ефективність від присутності на ньому, тому нові розробники появляються рідко.

Значними перешкодами на шляху до інформатизації вітчизняної системи охорони здоров'я є недостатнє фінансування медичних закладів, відсутність у керівництва розуміння можливостей, які дають високі технології для підвищення якості роботи медичних закладів, та недостатня стандартизація даних та способів її обробки.

Разом з цим система охорони здоров'я стикається з такими проблемами, як: підвищення вимог населення до систем охорони здоров'я; обмежений бюджет; часта зміна місця проживання пацієнтів та ін.

Більшість медичних інформаційних систем, які функціонують у лікувальних закладах в даний час, є морально і фізично застарілими [1]. Переважно вони розроблені ще 10-15 років тому, їх ніхто вже давно не підтримує і не удосконалює. Ці системи дозволяють автоматизувати тільки підготовку звітних форм.

Саме тому за вищевказаним тема даної роботи є дуже актуальною.

Основні матеріали дослідження. Під медичною інформаційною системою (МІС) розуміється програмний комплекс, призначений для автоматизації основних робочих процесів медичних установ. Автоматизовані медичні інформаційні системи дозволяють швидко й ефективно налагодити електронний документообіг, гнучко вибудовувати роботу з пацієнтами, вести оперативний облік роботи адміністративного персоналу, контролювати всі організаційні і фінансові питання [1].

Інформаційні системи (ІС), які використовують в охороні здоров'я (по суті – МІС), можна класифікувати за структурно-технологічним принципом, тобто ознаці їх використання в медичних установах [2]. Це: автоматизовані робочі місця; ІС структурних підрозділів медичних установ (лікувального відділення, клінічної лабораторії, аптеки, тощо); ІС установ системи охорони здоров'я (поліклініки, стаціонару, діагностичного центру, станції переливання крові, тощо); ІС територіального рівня (управління охорони здоров'я, територіального фонду обов'язкового медичного страхування, тощо); ІС федерального рівня (Міністерства охорони здоров'я України, регіональних центрів, тощо). Дана класифікація підкреслює особливості побудови медичних інформаційних систем різних рівнів і полегшує розуміння питань їх інформаційної взаємодії і інтеграції між собою.



Рисунок 1. Функціональна структура МІС

Поширеність основних типів МІС [2]: довідкові – 3%; системи страховальної медицини – 2%; лікувально-діагностичні – 37%; керування

охороною здоров'я – 13%; організаційно-економічні – 11%; фармакологічні – 9%; комплексні МІС – 8%; засоби навчання – 6%; обробка зображення – 6%; телемедичні – 5%.

На рисунку 1 наведена функціональна структура МІС [2].

У нових системах управління додаткова функція – це вироблення керуючих впливів.

Велике поширення у медичних закладах отримали інформаційно-пошукові системи, які, залежно від характеру інформації, що зберігається, діляться на системи документальні та фактографічні.

На етапі, коли МІС стає пакетом численних програм, гостро постає питання їх підтримки. Актуальність її зростає разом зі строком експлуатації і зростанням кількості користувачів. Поряд з початковими капітальними витратами, адміністрування МІС становить значну цифру в кошторисі витрат. Застосування складних комплексних інформаційних систем вимагає висококваліфікованого штату програмістів і адміністраторів. Із зростанням кількості підключених до бази даних системи користувачів зростає і складність її обслуговування.

Математичне забезпечення МІС складають комплекси спеціальних алгоритмів і програм. Програмне забезпечення МІС являє собою комплекс спеціальних програм і підпрограм, баз даних і баз знань, що реалізують рішення поставлених перед МІС завдань відповідно до заданого комплексу алгоритмів [4].

Основною особливістю медичної інформації є різноманітність даних, які можуть бути представлені в стандартизованому електронному форматі, як, наприклад, дані комп'ютерної томографії, рентгенографії, ультразвукових методів обстеження, паперові записи лікаря, тощо. Організація системи зберігання медичних даних починається із затвердження концепції зберігання, яка визначає вибір програмно-апаратного комплексу. Інфраструктура системи зберігання даних включає різні апаратні засоби, які можуть об'єднуватися в мережі зберігання даних, організацію доступу серверів до масивів даних, а також програмне забезпечення управління зберіганням даних.

Для крупних медичних установ, а також медичних систем з великою кількістю даних оптимальним є створення мереж зберігання даних SAN, невеликі медичні організації, як правило, використовують або NAS, підключені до локальної мережі, або локальні пристрої зберігання SAS, підключені безпосередньо до сервера або DAS, підключені до окремих комп'ютерів.

Організація системи архівації медичних даних передбачає створення єдиного сховища даних, з можливістю доступу до нього всіх суб'єктів інформаційної системи медустанови.

Апаратний комплекс зберігання даних може бути представлений декількома альтернативними варіантами, вибір яких визначається концепцією зберігання даних. До можливих варіантів слід віднести RAID-масиви, JBOD, стримери і стрічкові бібліотеки, оптичні носії, роботизовано DVD-бібліотеки. У масштабних інформаційних системах використовують RAID-масиви, стрічкові і DVD-бібліотеки.

Програмне забезпечення систем зберігання, інтегроване в загальну інформаційну систему, дозволяє створити комп'ютерний медичний атлас – інтелектуальний інтерфейс, побудований за принципом графічного гіпертексту, що полегшує роботу з базою медичних даних. Роботизована бібліотека є масивом DVD дисків, розміщених в окремому корпусі, сумарна місткість бібліотеки вимірюється десятками терабайт. Використання малопотужних роботизованих

DVD-бібліотек зарекомендувало себе в інформаційних системах невеликих медичних підприємств, і навіть в локальних системах.

На сьогоднішній день існує багато медичних інформаційних систем (МІС), які відрізняються ціною та функціональними можливостями. Будь-яка МІС дає можливість зберігати основну медичну інформацію, отримувати деякі підсумкові результати, включає засоби захисту, перевірки прав доступу до інформації. Проте існують деякі принципи переваги та недоліки, особливості, які необхідно враховувати при виборі МІС.

Найбільш відомі на сьогодні МІС це [3]: медична інформаційна система “TherDep”; інформаційно-аналітична система “Електронна лікарня”; підсистема “Стаціонар”; підсистема “Поліклініка”; програмний комплекс “С-госпіталь”; конфігурація STANDART системи “Media-log”; програмний комплекс “Інтелектуальна поліклініка”, тощо.

Можна та доцільно буде провести порівняння МІС за основними характеристиками, для визначення найбільш ефективної системи.

Такими оцінювальними характеристиками можуть бути: наявність механізмів захисту інформації; використання баз даних; наявність web-інтерфейсу; підключення додаткових модулів до системи МІС; рівень вимог до апаратних засобів; здатність співпрацювати з медичним обладнанням; рівень вимог до апаратних засобів; наявність клієнт-серверної архітектури, тощо.

Тепер проведемо порівняння МІС за основними характеристиками, результати занесемо в таблицю 1 [3].

Таблиця 1 - Порівняння основних характеристик МІС

Характеристика	МІС					
	Електронна лікарня	Інтелектуальна поліклініка	Стаціонар	Поліклініка	TherDep	Medialog
Клієнт-серверна архітектура	+	+	+	+	+	+
Рівень вимог до апаратних засобів	Високий	Низький	Низький	Низький	Низький	Середній
Здатність співпрацювати з медичним обладнанням	+	–	ДВ	ДВ	–	+
Наявність механізмів захисту інформації	+	+	ДВ	ДВ	–	+
Наявність web-інтерфейсу	+	–	–	–	–	ДВ
Підключення додаткових модулів	ДВ	+	+	+	ДВ	+
Механізми статистичної обробки даних	+	+	+	+	+	+
Використання баз даних	+	+	+	+	+	+
Можливість конфігурації системи	ДВ	+	+	+	+	+

Примітка: ДВ – дані відсутні

З результатів проведеного порівняння МІС видно, що програмний комплекс “Інтелектуальна поліклініка” та система “Medialog” мають найкращі характеристики.

“Ідеальна” МІС має володіти наступними якостями: мати низький рівень вимог до апаратних засобів, механізми захисту інформації (шифрування, електронний підпис), механізми статистичної обробки даних, бути здатною до співпраці з медичним обладнанням, з БД, надавати можливість конфігурації системи та підключення додаткових модулів.

Висновок. Впровадження інформаційних технологій в медицині заслуговує на безпосередню увагу керівників галузі і зацікавлених відомств. Одним з пріоритетних напрямів розвитку системи охорони здоров'я є створення єдиного медичного інформаційного простору, який забезпечить прийняття ефективних

управлінських рішень на всіх рівнях. Це дасть можливість медичним закладам налагодити ефективний облік діяльності організації здійснювати на сучасному рівні менеджмент, своєчасно отримувати інформацію про передові досягнення в галузі медичної науки, використовувати всю медичну інформацію про пацієнта (за весь період його життя), накопичену зі всіх рівнів надання медичної допомоги для досягнення кращого лікувального ефекту.

Використання медичних інформаційних систем є доволі перспективним напрямком діяльності медичних установ. Це дозволить підвищити якість обслуговування, запровадити електронний документооборот, знизити вплив людського фактору. Завдяки цій системі пацієнти зможуть отримати доступ до своїх даних про результати аналізів, тримати оперативний зв'язок з лікарем. Водночас, знижується ризик підробки лікарських даних, адже все відбувається в автоматичному режимі.

Список використаних джерел:

1. Качмар В. О. Медичні інформаційні системи – стан розвитку в Україні. Український журнал телемедицини та медичної телематики, Т.8, №1. 2010. С. 12-17.
2. Медичні інформаційні системи: огляд можливостей і приклади використання. URL: <https://evergreens.com.ua/ua/articles/medical-information-systems.html>.
3. Злепко С.М., Овчарук Т.І., Овчарук А.А. Огляд медичних інформаційних систем. Системи обробки інформації. № 3. 2011. С. 189-192.
4. Методи та системи штучного інтелекту: навч. посіб. / укл. Д.В. Лубко, С.В. Шаров. Мелітополь: ФОП Однорог Т.В., 2019. 264 с. ISBN 978-617-7566-68-6
5. IOM. To Err is Human: Building a Safer Health System: Institute of Medicine (IOM); 1999.
6. Wang, S., et al. A Cost-Benefit Analysis of Electronic Medical Records in Primary Care // The American Journal of Medicine, Vol. 114, 2003, pp. 397–403.

УДК 004.891

АНАЛІЗ МЕДИЧНИХ ЕКСПЕРТНИХ СИСТЕМ

Лубко Д.В., к.т.н.

e-mail: dmytro.lubko@tsatu.edu.ua

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Сьогодні ми спостерігаємо повсюдне впровадження інформаційно-комунікаційних технологій в різні сфери діяльності людини, зокрема економіку, науку, освіту. Крім того, багато сучасних програмних засобів використовують елементи штучного елементу, різні моделі подання знань тощо. Не минули ці процеси і медицину, яка ще з часів розробки експертної системи MYCIN у 1970 році активно використовує інтелектуальні, зокрема експертні, системи по теперішній час. Слід зазначити, що експертні системи призначені для виконання окремих завдань в окремій наочній області, тому необхідність в медичних експертних систем є актуальною [1].

Експертні системи є одним з найпоширеніших типів систем штучного інтелекту. Вони розроблялися як науково-дослідні інструментальні засоби з 1960-х років і розглядалися як штучний інтелект спеціального типу, призначеного для успішного вирішення складних задач у вузькій предметній галузі, такий як медична діагностика захворювань. Найбільш широке поширення експертних систем в різних областях людської діяльності почалося на початку 80-х років ХХ століття. Оскільки одними з перших експертних систем, які мають попит і зараз, є медичні, то основною метою даної статті є їх огляд та аналіз [2].

Експертна система (ЕС) – напрямок досліджень в області штучного інтелекту по створенню обчислювальних систем, здатних приймати рішення, схожі з рішеннями експертів в заданій предметній області [3]. Експертною системою є спеціальна програма для ЕОМ, основана на алгоритмах штучного інтелекту, що передбачає використання відповідної інформації, отриманої раніше від предметних експертів. Широкого розповсюдження використання експертних систем набуло в медичній діагностиці. За допомогою заміни лікаря-експерта експертною системою (різновидом системи штучного інтелекту) досягається суттєве підвищення ефективності діагностики та лікування [4].

На сьогоднішній день все більше зростає значення інформаційного забезпечення різноманітних медичних напрямків. Розвиток експертних систем є актуальною темою, адже це дає можливість вийти на новий рівень представлення перебігу захворювань. Експертні комп'ютерні медичні системи дозволяють лікарю не тільки перевірити власні діагностичні припущення, а і звернутися до машини за консультацією у важких діагностичних випадках.

В наш час побудова ефективних експертних систем різного призначення, у тому числі, медичних систем, залишається завданням актуальним і перспективним. Пройдений лише перший відрізок довгого шляху, труднощі подолання якого обумовлені, з одного боку, надзвичайною складністю об'єкту, що моделюється з використанням експертних систем – мозку людини, а з іншого боку високою складністю завдань, вирішення яких покладається на медичні експертні системи [5].

Основні матеріали дослідження. Комп'ютерні інформаційні технології давно і успішно допомагають медицині. Сучасний лікар використовує в своїй практиці велику кількість електронних приладів. Практично немає таких областей

в медичній діяльності, в яких ПК не могли б принести користі. Перші використання ПК у медицині пов'язані з автоматизацією найпростіших рутинних операцій, які виконує медичний персонал. З часом питання автоматизації роботи медичних закладів зростало, з'являлися електронні картотеки, діагностичні системи, бази даних, що зберігають інформацію про донорів, кров, медикаменти та ін. Основними перевагами запровадження ПК в медичному закладі [6] є:

- збільшується ефективний час перебування пацієнта в госпіталі) і як наслідок, зростання задоволеності пацієнтів;
- використання електронних медичних протоколів дає можливість підрахунку ефективності лікування, зменшення вірогідності медичних помилок, підвищує медичну якість послуг.
- можливість швидкого формування будь-яких звітів для прийняття обґрунтованих управлінських рішень (кількість наданих послуг, завантаженість лікарів та кабінетів, статистика пацієнтів за діагнозами, віком, статтю, і т.д.).
- ефективне управління складськими запасами медикаментів та витратних матеріалів за рахунок автоматизації процесу замовлення-списання та підтримки оптимального залишку;
- підвищення рівня безпеки та конфіденційності інформації завдяки запровадженню політики прав доступу до різних даних та для входу в систему за паролем або за відбитком пальця;
- повноцінне охоплення предметної області – це єдина база даних, яка дає можливість вести оперативний обіг всіх ресурсів (матеріальних, людських, фінансових);
- збільшення пропускної спроможності медичного закладу за незмінних ресурсів, за рахунок оптимізації процесів введення, пошуку, зведення та аналізу даних; швидшої взаємодії між підрозділами;
- планування завантаженості лікарів, кабінетів, обладнання;
- підвищення якості обслуговування пацієнтів (зменшується час очікування за рахунок планування).

Експертні системи - це діагностичні програми високого рівня, які належать до інтелектуальних інформаційних систем, що базуються на знаннях. Розділ інформатики, що вивчає інтелектуальні системи, називається штучним інтелектом. До складу експертної системи окрім бази даних входить база знань і розвинена система програм її обробки. Що розуміють під базою знань? База знань — це особливого роду база даних, розроблена для управління знаннями (мета-даними), тобто збиранням, зберіганням, пошуком і видачею знань. Найважливіший параметр бази знань — якість знань, що містяться в ній. Кращі бази знань включають останні наукові знання, досконалі системи пошуку інформації, мають ретельно продумані структуру і формат знань. Під медичними знаннями розуміють інформацію, потенційно необхідну лікареві. Зокрема це може бути інформація про стан здоров'я пацієнта, певну предметну медичну галузь, ситуацію, проблеми, правила їхнього вирішення, а також про конкретні умови, що визначають вибір цих правил. Медичні знання відрізняються від медичних даних більшою складністю, абстрактністю, повнотою й багатобічністю опису деякої предметної галузі медицини. Нагадаємо, що медицина з погляду інформатики — це неконкретна наука, тому створення баз знань у медицині — процес досить трудомісткий. У реальному житті кількість ситуацій і "діагностичних правил" іноді виходить за межі вкладених у систему знань.

Використання експертних систем відзначено безумовною їх ефективністю в порівнянні з людським фактором при їх застосуванні. Таким чином, експертна

система має безумовні переваги, а саме [5]: надає коректну роботу з інформацією, що містить помилки, за рахунок використання імовірнісних методів досліджень; надає висновок після всіх етапів обстежень; краще ніж людина при вирішенні надзвичайно громіздких питань; не має хибної думки, в той момент людина може бути під впливом численних зовнішніх хибних факторів; забезпечує діалоговий режим роботи; проводить та надає одночасну обробку альтернативних версій; дає можливість аргументувати рішення та відтворити шлях його впровадження.

Експертні системи відрізняються від інших систем штучного інтелекту наступними основними ознаками [2]:

- 1) орієнтованість на вузькоспеціалізовану предметну галузь;
- 2) вони розробляються для вирішення задач, які підходять для людини-експерта;
- 3) здатність пояснювати свої рішення зрозумілим користувачеві образом.

Ці ознаки в тій чи іншій мірі простежуються в медичних експертних системах, перші суттєві розробки яких відносяться до початку 1970-х років. Застосування експертних систем в медицині найефективніше при вирішенні задач діагностики, інтерпретації даних, прогнозуванні перебігу захворювань і ускладнень, моніторингу перебігу захворювання і планування лікувально-діагностичного процесу. В наш час побудова ефективних експертних систем різного призначення, у тому числі, медичних систем, залишається завданням актуальним і перспективним. Пройдений лише перший відрізок довгого шляху, труднощі подолання якого обумовлені, з одного боку, надзвичайною складністю об'єкту, що моделюється з використанням експертних систем – мозку людини, а з іншого боку високою складністю завдань, вирішення яких покладається на медичні експертні системи [7, 8].

На відміну від експериментальних даних медичні знання є структурованими, внутрішньо інтерпретованими, зв'язними, активними, конвертованими, семантично метризованими. Медичні діагностичні ЕС (МЕС) працюють в двох режимах: режим набуття медичних знань та режим медичної консультації.

В такій медичній ЕС виділяють наступні типи користувачів: користувачі-пацієнти; експерти; інженери зі знань. Структурна особливість будь-якої такої системи – наявність бази знань (БЗ), що є сукупністю фактів (наявність певних симптомів, ступінь їх вираженості, значимість для твердження про наявність захворювання, про прогноз протікання захворювання і т.д.) і правил логічного висновку. Процес набуття знань реалізується експертом (лікарем), процес формування БЗ – інженером знань.

Основною задачею медичної експертної системи є визначення можливого фахівця, до якого пацієнтові слід звернутися, на основі отриманих діагностичних гіпотез. Така система не може нести відповідальність за отримане з її допомогою діагностичне рішення, а виступає в ролі своєрідного консультанта.

На рисунку 1 зображена узагальнена схема експертної системи [4].

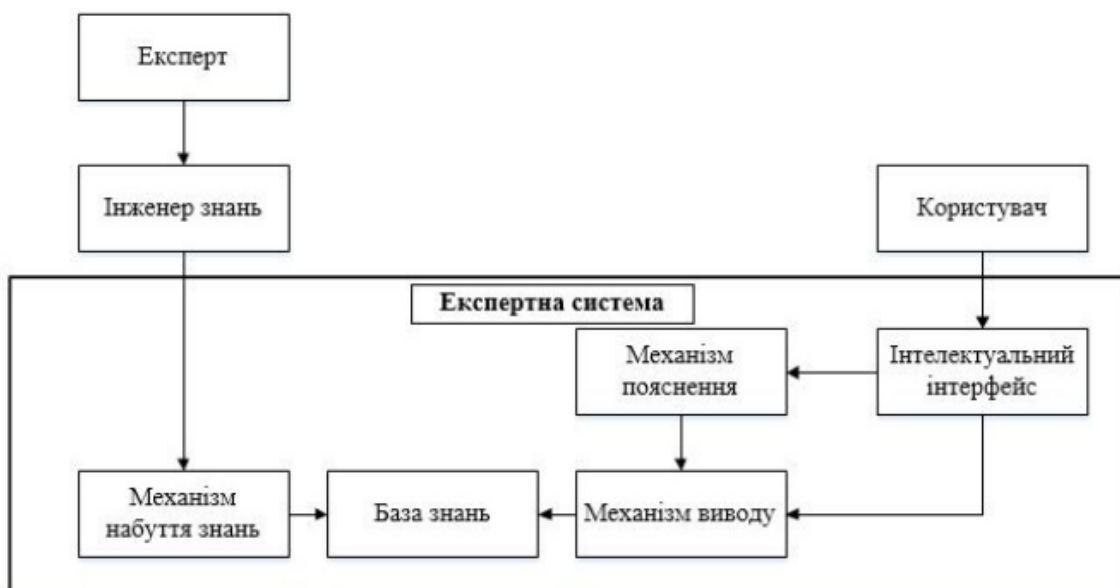


Рисунок 1. Структурна схема експертної системи

Враховуючи складність використання такого програмного забезпечення як МЕС, виникає гостра необхідність в розробці дружнього інтерфейсу, для кращого розуміння використання цього складного програмного забезпечення, коли проявляються нові знання під час сеансу експертизи «навички» системи не завжди покращуються.

Слід зазначити, що експертні системи під час використання не мають можливості оновлювати якісно нові знання, які не були передбачені під час проектування програмного забезпечення, і не можуть володіти таким логічним апаратом як людина-експерт. На відміну від експертної системи, людина-експерт може використовувати такі елементи як інтуїція та здоровий глузд, у разі відсутності певних формальних ознак вирішення або відсутні аналогічних рішень для схожих задач [5].

Прийняття рішень у різних складних ситуаціях лікарем - експертом – це відповідальний процес, що найчастіше відбувається при наявності великої кількості різноманітних медичних даних. Для проведення процедури діагностування й наступного призначення лікування лікареві необхідно враховувати не тільки зовнішні ознаки дії хвороби, аналізи пацієнта, але ще й анамнез пацієнта. Дуже часто випускаються з виду різні фактори зовнішнього середовища, що діють на хворого [9]: наявність промислового сектора в районі, генетичні схильності, нездоровий спосіб життя. Тому в даній ситуації рекомендується створення бази даних, використання якої дозволить адекватно оцінити стан хворого, враховуючи всі аспекти життя пацієнта. Обробка інформації з бази даних ускладнюється різноманітною природою таких даних: дати, строкові дані: ім'я, адреса, стать (бінарні дані), анамнез (строкові, шкальні), поточний діагноз (строкові) та стадія лікування (шкальні дані).

Одна з класифікацій [9] медичних даних пацієнта представлена на рисунку 2.

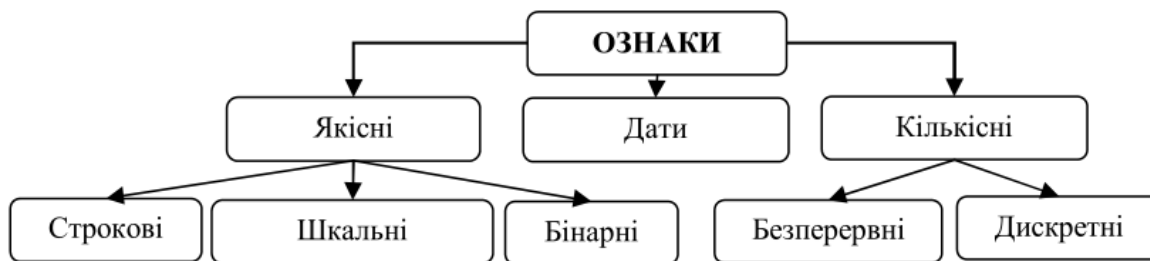


Рисунок 2. Класифікація медичних даних

Для ефективної обробки бази даних необхідно використовувати медичну експертну систему, що спеціалізується на процесі діагностування й розробці та підтримці процесу лікування. В основу такої експертної системи покладається база знань, що була накопичена експертами (лікарями) за деякий проміжок часу, та математичні методи, що дозволяють обробити вхідну інформацію з мінімальними погрішностями. Така комп'ютерна система, дозволить скоротити час та засоби, відведені на прийняття рішень щодо лікування та призначення процедур. Наприклад, скорочення строків госпіталізації дозволило б заощадити значні засоби за рахунок зменшення відсотку прийняття помилкових рішень щодо діагностування та лікування з за більш ефективної обробки вхідної інформації.

Треба зауважити, що складність процесу розробки експертної системи суттєвим чином залежить від інструментальних засобів, які для цього використовуються [1]. І оскільки на даний момент існує безліч мов та середовищ програмування, призначених для розробки різноманітного програмного забезпечення, при виборі інструментального засобу слід враховувати декілька чинників, а саме: специфіки програмного засобу, який буде розроблюватися. Це може бути мобільний додаток, веб-ресурс, віконний додаток та ін.; функціональних можливостей, які повинні бути реалізовані; досвіду програміста при роботі з конкретним інструментальним засобом; часу, який відводиться на розробку.

Більшість програмних засобів [1], які можуть використовуватися для розробки інтелектуальних систем, зокрема медичної експертної системи, можна поділити на декілька груп:

1. Мови програмування високого рівня позбавляють розробників від необхідності заглиблюватися в механізми реалізації програмного засобу на машинно-залежному рівні. У цю групу входить дві велика підгрупи: традиційні процедурні мови типу Pascal, C++ та традиційні середовища програмування, які дозволяють створювати додатки з використанням об'єктно-орієнтованого підходу та технології візуального програмування, такі як C# та Delphi.

2. Оболонки експертних систем, які дозволяють створити експертну систему на основі введених правил, параметрів, вбудованого програмного коду. Оболонку можна представити у вигляді експертної системи, яка містить інструменти та компоненти, які мають загальну спеціалізацію.

3. Спеціалізовані мови логічного програмування, такі як LISP, Prolog, які застосовуються для створення баз знань та обробки об'єктів та відносин між ними. Для мови логічного програмування Prolog типовими є завдання щодо розробки різноманітних програмних комплексів з елементами штучного інтелекту (експертні системи, комп'ютерні перекладачі, інтелектуальні ігри тощо). Prolog

виконує пошук рішення серед опису об'єктів та правил у базі знань за допомогою механізму уніфікації та механізму повторення.

Основні етапи процесу розробки медичних експертних систем [2] представлені на рисунку 3.

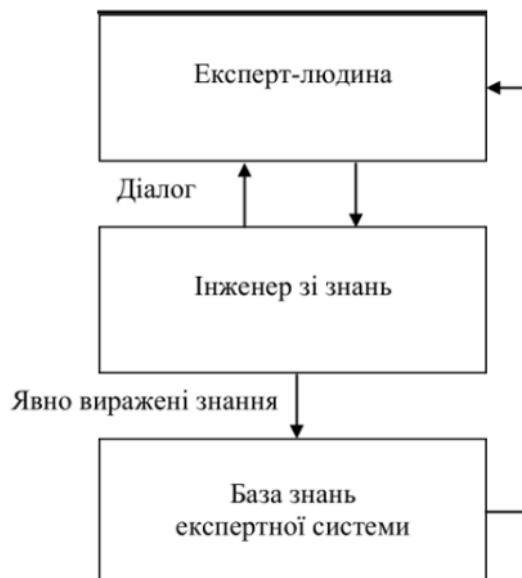


Рисунок 3. Процес розробки медичної експертної системи

На рисунку 4 наведена загальна схема уніфікованої медичної експертної системи.



Рисунок 4. Загальна схема уніфікованої медичної експертної системи

Висновок. Впровадження та використання медичних експертних систем дуже корисно для сектору охорони здоров'я. Особливо часто їх використовують для діагностування, моніторингу, прогнозування, підтримки прийняття рішень, тобто тих самих завдань, які становлять природу медицини. Якість діагностики оцінюють на рівні кваліфікованого лікаря, а це дуже високий показник. Експертні системи високо-кваліфікаційні. Поки вони забезпечують вирішення ізольованих завдань (наприклад, медичної діагностики). Також їх часто використовують в медичних приладо-комп'ютерних системах. Найважливіші галузі застосування такої експертної системи — це невідкладні та загрозливі стани, що характеризуються дефіцитом часу, обмеженими можливостями обстеження та консультацій і

нерідко бідною клінічною симптоматикою. Наразі медичні експертні системи — найперспективніший напрямок. Безсумнівно, майбутнє за медичними експертними системами.

Використання таких (медичних) експертних систем покращить роботу лікарів в разі та дасть змогу бути точними в своїх рішеннях. Користь від використання медичних експертних систем є вагомою, а саме це: сталість в рішеннях, легкість передачі або відтворення інформації, стійкість і відтворюваність отриманих результатів, вартість в експлуатації.

Список використаних джерел:

1. Шаров С.В., Печерський Р.В. Аналіз інструментальних засобів для розробки експертної системи медичної лабораторії. Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення: зб. тез доп. Міжн. наук. Інтернет-конф.(м. Тернопіль, 17 травня 2017 р.). Тернопіль. 2017. - С.87-90.
2. Любченко К.М. Експертні системи в практичній медицині. 2008.
3. Методи та системи штучного інтелекту: навч. посіб. / укл. Д.В. Лубко, С.В. Шаров. Мелітополь: ФОП Однорог Т.В., 2019. 264 с.
4. Герасімова Д.С. Медичні нейромережеві експертні системи в діагностиці (Докторська дисертація, Тернопіль, ТНЕУ). 2016.
5. Білоус Д.В. Використання експертних систем в медицині. 2021.
6. Муляр М.М.. Медична Інформаційна Система — зручний інструмент для управління медичним закладом [Електронний ресурс] / Муляр М.М. // Наукове видання «Управління закладом охорони здоров'я». — 2008. - №8 — Режим доступу: http://medexpert.com.ua/nashi_statti_medichna_informacijna_sistema.html.
7. Продеус А. Н., Захрабова Е. Н. Экспертные системы в медицине / А. Н. Продеус, Е. Н. Захрабова. — К.: БЕК+, 1998. — 320 с.
8. Филунова Е. Г. АТМ Express: Биорезонансная диагностика и терапия (ПБРТ): методические рекомендации / Филунова Е. Г., Сиряковская Е. И., Демьянцева И. В. — К.: Алтимед, 2007. — 27 с.
9. Мельник К.В. Особливості обробки даних для медичної експертної системи (Докторська дисертація, Національний технічний університет" Харківський політехнічний інститут"). 2010.

УДК 004.85

ДОСЛІДЖЕННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ ДЛЯ РІШЕННЯ ЗАДАЧ МЕДИЧНОГО ПРОФІЛЮ

Островська К.Ю., к.т.н.

e-mail: kuostrovskaya@gmail.com

Мінаєнко А.С., магістр

e-mail: anitamin16@gmail.com

Український державний університет науки і технологій

Актуальність та постановка проблеми. Методи машинного навчання застосовуються в різноманітних областях та допомагають вирішувати безліч завдань: від виявлення спаму та актів шахрайства до розпізнавання та генерації зображень та музичних композицій. Важливим та перспективним додатком методів машинного навчання є медична діагностика. Зокрема, прогнозування стану пацієнта, диференціальна діагностика захворювань, перевірка ефективності препаратів тощо [2, 3].

На сьогоднішній день в медицині накопичуються величезні масиви різноманітних даних, а з удосконаленням та впровадженням нових медичних технологій, у тому числі комп'ютерних, швидкість їхнього накопичення постійно зростає. Велику роль відіграють текстові та числові медичні дані, перебувають у виписках пацієнтів. У зв'язку з цим існує потреба в їх обробці та аналізі, а також застосування методів штучного інтелекту для отримання нових знань [4].

Найбільш поширеними захворюваннями сечостатевої системи у дітей є пієлонефрит, гломерулонефрит та тубулоінтерстиціальний нефрит. Їх особливістю є складна діагностика і часом безсимптомна клінічна картина. Наприклад, при пієлонефриті зазвичай переважають загальні симптоми в клінічній картині, що призводить до досить тяжкого виявлення даного захворювання [5]. Своєчасна діагностика перерахованих захворювань допоможе оперативно підібрати індивідуальне лікування та уникнути ускладнень у майбутньому.

Мета роботи – аналіз методів машинного навчання для підвищення точності та скорочення часу діагностики захворювань сечостатевої системи у дітей.

Об'єкт дослідження – методи машинного навчання.

Предмет дослідження – класифікатор захворювань сечостатевої системи пацієнтів Дніпропетровської обласної дитячої клінічної лікарні "Дніпропетровської обласної ради".

Актуальність цієї роботи зумовлюється необхідністю підвищення точності та скорочення часу діагностики захворювань сечостатевої системи у дітей.

Практична значущість дослідження полягає у підборі та описі таких методів машинного навчання, які допоможуть лікарям проводити діагностику захворювань сечостатевої системи у дітей.

Основні матеріали дослідження. У цьому дослідженні використовувалася база даних анамнезів (виписок) пацієнтів дитячого віку з різними захворюваннями сечостатевої системи Дніпропетровської обласної дитячої клінічної лікарні у період з 2014 по 2021 роки. У виписках міститься важлива інформація про перебіг захворювань, що проводяться аналізах та лікуванні. Розмір бази даних – 3773 файли. Кожен файл формату *.docx був частково або повністю заповнений шаблон анамнезу пацієнта. Необхідно було написати програму для переведення всієї бази даних у стандартний табличний формат за допомогою парсингу.

Використані дані з медичних виписок пацієнтів мають складну багатовимірну структуру, значний дисбаланс у захворюваннях, велика кількість пропущених значень, що потрібно враховувати при застосування методів машинного навчання. Також багато методів вимагають від ознак певних властивостей, тому важливим етапом є передобробка даних, що запобігатиме уповільненню збіжності алгоритмів та зниження якості класифікації.

Після первинного аналізу та передобробки даних слідує етап застосування методів машинного навчання. У рамках цієї роботи використовувалися такі методи класифікації, як: дерево рішень (DT), випадковий ліс (RF), градієнтний бустинг (GB), логістична регресія (LR), метод K найближчих сусідів (KNN) та багатосаровий перцептрон (MLP) [6].

Розроблена в даній роботі програма дозволить аналітику витягувати в напівавтоматичному режимі необхідну інформацію відповідно до рекомендаціями лікаря та проводити її аналіз, що надалі може допомогти прискорити процес постановки діагнозу та збільшити його точність.

Також у рамках даного дослідження стосовно діагностики захворювань сечостатевої системи у дітей отримано оцінку практичної значимості наступних методів машинного навчання: дерево рішень, випадковий ліс, градієнтний бустинг, логістична регресія, K найближчих сусідів, багатосаровий перцептрон, а також таких підходів до підвищення їх якості, як рекурсивний відбір ознак та заповнення пропущених значень методом K найближчих сусідів.

Згідно з цією оцінкою градієнтний бустинг виявився кращим серед інших методів з погляду застосування на практиці і показав більше високу передбачувальну здатність. Відбір найбільш значущих ознак і заповнення пропущених значень не покращили якість методу на тестовому набір даних.

По суті, градієнтний бустинг є непараметричним методом класифікації і дозволяє будувати ансамблі слабких класифікаторів так, щоб кожен наступний компенсував помилки вже збудованої множини попередніх. З точки зору медицини градієнтний бустинг моделює ситуацію, коли кожного наступного лікаря тренують фокусувати увагу на помилково поставлених діагнозах попередніх лікарів і тим самим підвищувати якість діагностики.

Висновок. У рамках цієї роботи були вирішені різні проблеми роботи з виписками пацієнтів і була написана програма, яка одержує необхідну інформацію з них у напівавтоматичному режимі.

Також були проаналізовані різні методи машинного навчання та підходи до підвищення їх якості стосовно вилучених даних. В результаті градієнтний бустинг показав велику в порівнянні з рештою методами адекватність та передбачувану здатність за всіма підходами. Метод рекурсивного відбору ознак загалом значно не покращив та не погіршив якість класифікаторів, а метод заповнення пропущених значень у більшості випадків сильно погіршив його.

Результати роботи показали необхідність розвитку цієї проблематики. Подальші дослідження в галузі застосування на практиці методів машинного навчання до діагностики захворювань сечостатевої системи у дітей можуть допомогти лікарю не тільки у витягуванні інформації з масиву виписок та його аналіз, а й у скороченні часу постановки діагнозу та збільшенні його точності.

Список використаних джерел:

1. Мінцер О.П. Оброблення клінічних і експериментальних даних у медицині: навч. посібник / О.П. Мінцер, Ю.В. Вороненко, В.В. Власов - К.: Вища шк., 2003. - 350 с.

2. Голованова І. А. Основи медичної статистики : навч. посіб. для аспірантів та клінічних ординаторів / І.А. Голованова, І.В. Белікова, Н.О. Ляхова. – Полтава, 2017. – 113 с.
3. United States Patent № US 7,657,521 B2, 02.02.2010. System and method for parsing medical data [text] / Fred E. Masarie, Stuart Lopez, Michael I. Lieberman // United States Patent № US 7657521 B2. 2010.
4. Kazemi Y. A novel method for predicting kidney stone type using ensemble learning / Y. Kazemi, S.A. Mirroshandel // Artificial Intelligence in Medicine. – 2017. – Vol. 79, No. 3. – P. 1696–1707.
5. Фадеев П.А. Болезни почек. Пилонефрит. – М.: Мир и Образование, 2011. – 180 с.
6. Флаx П. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных / пер. с англ. А.А. Слинкина. – М.: ДМК Пресс, 2015. – 400 с.

УДК 004.85

ДОСЛІДЖЕННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗРОБКИ СИСТЕМИ АВТОМАТИЧНОЇ КЛАСИФІКАЦІЇ ЗОБРАЖЕНЬ ФОТОБАНКУ

Островська К.Ю., к.т.н.

e-mail: kuostrovskaya@gmail.com

Рогбак К.С., магістр

e-mail: kklirill9882@gmail.com

Український державний університет науки і технологій

Актуальність та постановка проблеми. Фотобанк (фотосток) – це банк зображень, що виступає посередником між авторами зображень та їх покупцями. Сукупний обсяг світового ринку комерційних цифрових зображень у 2011р. становив \$11 млрд. і до 2016 р. досяг рівня \$13 млрд [1]. Основними покупцями комерційних цифрових зображень є ілюстратори та дизайнери. Мікрофотостоки – різновид фотобанків, спеціалізуються на продажу зображень на неексклюзивній основі з використанням ліцензій «Royalty Free» та подібних за меншу вартість, ніж при купівлі ексклюзивних матеріалів. Опитування читачів професійного журналу дизайнерів GDUSA [2] за останні роки показали, що понад 90 % дизайнерів набувають готових зображень саме в мікрофотостоківих агентствах.

Фотостоки сьогодні містять десятки та сотні мільйонів зображень. Наприклад, компанія Shutterstock (shutterstock.com) має фотобанк, який налічує близько 136,5 млн. зображень, IstockPhoto (istockphoto.com) – близько 31 млн. зображень.

На даний момент визначенням категорії зображення та зі складання списку ключових слів займаються самі автори зображень.

Найчастіше, перелік ключових слів і категорій зображення неточний, оскільки автори, намагаючись збільшити кількість видачі свого зображення в пошуковій системі, додають популярні ключові слова, що не відносяться до зображення. Через це пошукові видачі не дозволяють відібрати потрібні зображення, і користувачеві доводиться докладати зусиль для того, щоб знайти потрібний матеріал.

Таким чином, метою є дослідження методів машинного навчання для розробки системи автоматичної категоризації зображень.

Основні матеріали дослідження. Наповнення різних фотобанків різномірне, внаслідок чого для адекватного налаштування моделі необхідний збір даних із кількох джерел. Як такі були взяті три фотобанки: Fotolia (fotolia.com), Istock (istockphoto.com), Pressfoto (pressfoto.com). Збір даних повинен здійснюватися за допомогою аналізу веб-сторінок фотобанків (web-crawling) або звернення до їх API, якщо така є. На рисунку 1 представлена загальна схема циклу збору даних.

Система збирання даних була реалізована мовою Python версії 3.5. Мова Python має такі бібліотеки для роботи з HTTP:

- Urllib - стандартний Python-модуль для роботи з веб-сторінками;
- PyCurl - Python-обгортка, яка підтримує FTP, HTTP та HTTPS протоколи, дозволяє здійснювати POST-запити, підтримує cookies.
- Requests - бібліотека Python, що надає простий та мінімалістичний інтерфейс для здійснення запитів.

Збір даних було здійснено у два етапи: збір інформації про зображення та посилань на завантаження шляхом розбору HTML-сторінок та завантаження зображень.

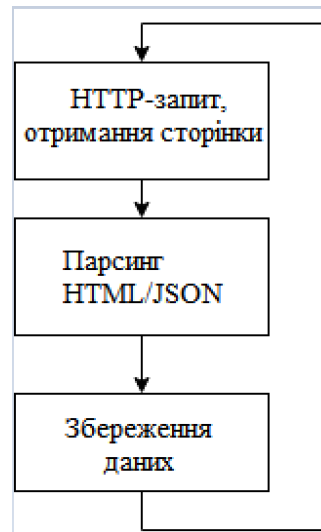


Рисунок 1. Схема циклу збору даних

Для очищення даних було проведено кластеризацію зображень за допомогою алгоритму поділу суміші нормальних розподілів [3], реалізованого в бібліотеці Sci-kit learn [4]. Як ознаковий опис об'єктів було взято вихідні сигнали мережі VGG16 [5], навченої на наборі даних ImageNet [6]. Схема архітектури мережі VGG16 представлена на рисунку 2. Для отримання ознакового опису зображень були відсічені останні повнозв'язні шари мережі, а як ознаковий описи об'єкта виступили усереднені значення вихідних сигналів останнього згорткового шару.

Вага навченої мережі знаходиться у відкритому доступі на веб-сторінці проекту [7], їх отримання та операції над нейронною мережею були проведені засобами бібліотеки Keras [8].

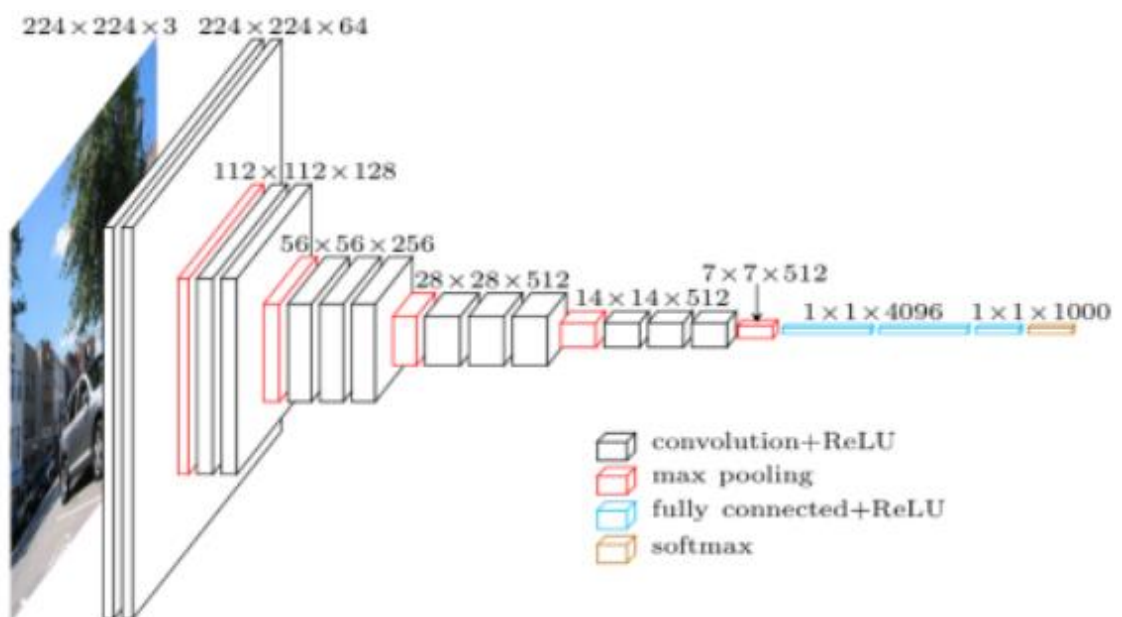


Рисунок 2. Схема архітектури мережі VGG16

Оцінка ефективності моделей проводиться за допомогою ковзного контролю (крос-валідації) [3] – методики тестування та порівняння алгоритмів класифікації, регресії та прогнозування. Процес крос-валідації полягає в навчанні та оцінюванні моделі на різних підвиборках навчальної вибірки. Для цього проводиться деяка безліч розбиття вихідної вибірки на дві підвиборки: навчальну та контрольну. Оцінкою ковзного контролю є середня за всіма розбиттям величина помилки на контрольних підвиборках.

Загальний алгоритм проведення експериментів виглядає так:

- 1) вибір моделі машинного навчання;
- 2) навчання моделі при деяких початкових параметрах та архітектурі, отримання початкового результату;
- 3) зміна архітектури моделі та її параметрів з метою отримання кращого результату;
- 4) порівняння моделей.

Висновок. В результаті було реалізовано систему, що дозволяє класифікувати зображення фотобанку на основі методів машинного навчання. Проведені експерименти показали, що набір моделей random forest на кшталт "один проти всіх" дає найкращу точність класифікації, тому його застосування ефективніше. Подальшим напрямком розвитку буде покращення якості класифікації для досягнення значення aver age accuracy 0,9 та введення системи в експлуатацію

Список використаних джерел:

1. Sokolova M., Lapalme G. A systematic analysis of performance measures for classification tasks. // Information Processing and Management, 2009. – No. 4 (45). – P.427–437.
2. Graphic Design USA. [Електронний ресурс] URL: <http://gdusa.com> (дата звернення: 20.09.2022).
3. Методи машинного навчання у задачах системного аналізу і прийняття рішень: монографія / В.Є. Стрілець, С.І. Шматков, М.Л. Угрюмов та ін. – Харків: Харківський національний університет імені В. Н. Каразіна, 2020. – 160 с.
4. Sci-kit learn. Machine Learning in Python. [Електронний ресурс] URL: <http://scikit-learn.org/stable/> (дата звернення: 20.09.2022).
5. VGG in TensorFlow. Model and pre-trained parameters for VGG16 in TensorFlow. [Електронний ресурс] URL: <https://www.cs.toronto.edu/~frossard/post/vgg16/> (дата звернення: 22.10.2022).
6. ImageNet. [Електронний ресурс] URL: <http://www.image-net.org> (дата звернення: 22.10.2022).
7. Very Deep Convolutional Networks for Large-Scale Visual Recognition. [Електронний ресурс] // Visual Geometry Group. Department of Engineering Science, University of Oxford. URL: http://www.robots.ox.ac.uk/~vgg/research/very_deep/ (дата звернення: 20.11.2022).
8. Keras Documentation. [Електронний ресурс] URL: <http://keras.io/> (дата звернення: 20.11.2022).

УДК-004.8

ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В МЕДИЦИНІ

Вакалюк Т.А., д. пед.н, професор

e-mail : kik_vta@ztu.edu.ua

Андрусенко О.М., магістрант

e-mail: kim221_aom@student.ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Штучний інтелект одна з найбільш прогресивних галузей в сучасному світі. Він може бути використаний в багатьох напрямках досліджень, у тому числі медицині, де штучний інтелект використовує моделі машинного навчання для пошуку необхідних медичних даних та встановлення правильного діагнозу задля призначення необхідного лікування та зменшення хвилювань пацієнта. Завдяки досягненням в галузях комп'ютерних наук та інформатики, штучний інтелект швидко та впевнено стає невід'ємною частиною сучасної системи охорони здоров'я. Також він є швидким та зручним інструментом для аналізу величезних обсягів медичних даних, таких як магнітно-резонансна томографія, мамографія, геноміка, комп'ютерна томографія сканування та багато інших, задля встановлення правильного діагнозу. Крім того, штучний інтелект в першу чергу розширює лікарський досвід та дозволяє прискорити процес підготовки пацієнтів до продовження реабілітації вдома.

Основні матеріали дослідження. Хоча машинне навчання все ще перебуває на ранніх стадіях впровадження, воно має потенціал для забезпечення більшої точності діагностичних результатів, а також економії часу та грошей і, що найголовніше, рятування життів.

Наприклад, машинне навчання здатне раніше виявляти хвороби. 6 із 10 американців хоча б раз страждають від хронічних захворювань, таких як рак або хвороба серця. У такому випадку штучний інтелект може допомогти в діагностиці раку, використовуючи дані медичних досліджень для виявлення, вимірювання та аналізу пухлин. Застосовуючи свою перевагу в обчислювальній потужності для проведення аналізу даних і зображень швидше, ніж це можуть робити медичні працівники власноруч, машинне навчання може завершувати процес скринінгу за менший час, що дозволить скоротити час очікування направлення для пацієнтів з високим ризиком і зменшити навантаження на клініки, які стикаються з нестачею персоналу або іншими проблемами.

Технології машинного навчання можуть також покращити послідовність і точність діагностики, усунувши помилки, викликані людським фактором.

Ця відносно нова технологія також здатна автоматизувати певні завдання, що дозволить зменшити клінічне навантаження та дати можливість спеціалістам виконувати складні завдання, такі як візуалізація та аналіз досліджень серця, простіше. Це дозволить медичним працівникам охопити більшу кількість населення за допомогою догляду вдома або в невеликих клінічних установах та забезпечити більшу кількість пацієнтів доступом до медичної допомоги.

Висновки.

Хоча дослідники продовжують розширювати можливості штучного інтелекту та машинного навчання в медичній діагностиці, ці технології, як правило, ще не отримали широкого застосування та стикаються з низкою проблем, які обмежують більш широке використання.

Наприклад, деякі медичні працівники не бажають використовувати засоби штучного інтелекту в своїх клініках, доки його ефективність не буде більш широко доведена в різних клінічних умовах. Недостатня обізнаність серед деяких медичних працівників щодо того, як машинне навчання інтегрується в їхній робочий процес і покращить його, а також прогалини в нормативних вказівках і вимогах, а також вартість впровадження та обслуговування можуть суттєво обмежити його розвиток та використання. Проте лідери в індустрії штучного інтелекту прогнозують, що в найближчому майбутньому не лише будуть вдосконаленні існуючі рішення технології, а й буде створено нові шляхи його конвергенції з медициною.

Список використаних джерел:

1. Disease screening & diagnosis – AI solution by provectus. Provectus. URL: <https://provectus.com/disease-screening-healthcare/>.
2. Health care: how AI can impact medical diagnosis. Swarajyamag. URL: <https://swarajyamag.com/tech/healthcare-how-ai-can-impact-medical-diagnosis>.
3. How AI technologies accelerate progress in medical diagnosis. Robotics & Automation News. URL: <https://roboticsandautomationnews.com/2020/03/09/how-ai-technologies-accelerate-progress-in-medical-diagnosis/31184/>.

УДК 004.4

ДОСЛІДЖЕННЯ АЛГОРИТМІВ РЕКОМЕНДАЦІЙНИХ СИСТЕМ

Гордєєв Р.С., магістрант

e-mail: ipzm221_grs@student.ztu.edu.ua

Вакалюк Т.А., д. пед. н, професор

e-mail: kik_vta@ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Ми живемо в час, коли людина просто переповнена інформацією. І, коли людина заходить на сайт пошуку музики, якогось товару чи відео, найменше, що вона хоче – це поглибитись у ці вебресурси на довгий час, щоб знайти те, що їй потрібно. В таких ситуаціях на допомогу приходять рекомендаційні системи. Рекомендаційна система надає персоналізований список тих елементів, що повинні сподобатись користувачу найбільше. Такі рекомендаційні системи використовуються розповсюджено по всьому інтернету і дозволяє зекономити нам багато часу та нервів при пошуку необхідного матеріалу.

Основні матеріали дослідження. Зі зростанням таких сервісів, як: Spotify, YouTube, Netflix і так далі – рекомендаційні системи почали мати все більшу і більшу роль у нашому житті [1]. Через це є доцільним створення високоякісної рекомендаційної системи, яка б могла надавати користувачеві персоналізовані рекомендації у різних додатках. Незважаючи на велику розповсюдженість рекомендаційних систем та їхнє вдосконалення, все ж є багато аспектів які б можна було покращити в них. Для цього потрібно подальше дослідження існуючих алгоритмів функціонування рекомендаційних систем, які використовуються у додатках.

Різні алгоритми фільтрації рекомендаційних систем можна категоризувати в 3 типи:

- **Колаборативна фільтрація.** В системах, побудованих на цьому алгоритму, користувачеві рекомендуються предмети, які основані на оцінюванні всіх користувачів разом.

- **Фільтрація за вмістом.** Цей підхід визначається тим, що користувачеві рекомендуються елементи, що за вмістом схожі на предмети, що користувачеві колись подобались.

- **Гібридна фільтрація.** Цей алгоритм, як можна здогадатись, об'єднує в собі колаборативну фільтрацію і фільтрацію за вмістом.

Колаборативна фільтрація. Алгоритми колаборативної фільтрації працюють за збором відгуків користувача у вигляді рейтингу для елементів і використовує їх при визначенні спільної поведінки оцінювань між кількома користувачами для того, щоб визначити який елемент порекомендувати. Колаборативну фільтрацію можна далі поділити на підвиди, як зазначено на рисунку 1: *фільтрація за сусідством* і *фільтрація за моделлю*. Розглянемо ці алгоритми.

Колаборативна фільтрація за сусідством. В підходах, що базуються на сусідстві, підмножина користувачів обирається за їхньою спільністю відносно поточного користувача і зважена комбінація їх оцінок використовується для створення передбачень для заданого користувача. Більшість кроків алгоритму можна записати таким чином:

1. Присвоїти вагу всім користувачам в залежності від їхнього рівня спільності із поточним користувачем.

2. Вибрати k користувачів, які мають найбільше спільного із поточним користувачем (часто називаються *сусідами*).

3. Розрахувати передбачення з вагової комбінації з обраних оцінок сусідів.

У кроці 1 вага $w_{a,u}$ являється мірою спільності між користувачем u та поточним користувачем a . Найбільш поширеною мірою спільності є коефіцієнт кореляції Пірсона. Він записується такою формулою:

$$w_{a,u} = \frac{\sum_{i \in I} (r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I} (r_{a,i} - \bar{r}_a)^2 \sum_{i \in I} (r_{u,i} - \bar{r}_u)^2}} \quad (1)$$

де I – множина елементів оцінена обома користувачами, $r_{u,i}$ – оцінка надана елементу i користувачем u і $\bar{r}_a$ – середня оцінка надана користувачем u [2].

У кроці 3 передбачення розраховуються як зважене середнє значення відхилень від середнього значення сусіда:

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u \in K} (r_{u,i} - \bar{r}_u) \times w_{a,u}}{\sum_{u \in K} w_{a,u}}, \quad (2)$$

де $p_{a,i}$ – передбачення для поточного користувача a для елемента i , $w_{a,u}$ – подібність між користувачами a та u і K – це сусіди або множина найбільш подібних користувачів [2].

Колаборативна фільтрація за моделлю. Розробка та розвиток моделей (машинне навчання, алгоритми добування даних) дозволяють системам розпізнавати складні шаблони, ґрунтуючись на тренувальних даних, і, після цього, робити розумні передбачення для тестових чи реальних даних. Алгоритми колаборативної фільтрації за моделлю, такі як: Баєсові моделі, кластерні моделі – були досліджені для усунення недоліків попередніх алгоритмів.

Баєсовий алгоритм колаборативної фільтрації за моделлю. Баєсова мережа – це орієнтований, ациклічний граф з триплетом $\langle N, A, \Theta \rangle$, де кожне ребро $n \in N$ представляє випадкову змінну, а кожна направлена дуга $a \in A$ між ребрами це імовірнісний зв'язок між змінними і Θ – це таблиця умовної ймовірності, що розраховує наскільки ребро залежить від його батьків. Баєсові мережі часто використовуються для завдань класифікації.

Алгоритм колаборативної фільтрації просто Байєса використовує стратегію найного Байєса для здійснення передбачень. Припускаючи, що ознаки є незалежними для кожного класу, ймовірність, що певний клас, якому надані всі ознаки, може бути розрахований, і тоді клас з найвищою ймовірністю буде класифікований як спрогнозований клас. Для неповних даних розрахування ймовірності й виконання класифікації визначаються по спостереженням даним:

$$\text{class} = \arg \max p(\text{class}_j) \prod_0 P(X_0 = x_0 | \text{class}_j), \quad (3)$$

Оцінювач Лапласа використовується для розгладження ймовірності розрахунків і уникнення умовної ймовірності 0:

$$P(X_i = x_i | Y = y) = \frac{\#(X_i = x_i, Y = y) + 1}{\#(Y = y) + |X_i|}, \quad (4)$$

де $|X_i|$ – це розмір множини класу $\{X_i\}$ [3].

Кластерний алгоритм колаборативної фільтрації за моделлю. Кластер – це колекція об'єктів даних, які подібні один до одного в одному кластері і неподібні до об'єктів у інших кластерах. Міра подібності об'єктів визначається використовуючи такі метрики: метрика Мінковського та кореляція Пірсона.

Для двох об'єктів даних, $X = (x_1, x_2, \dots, x_n)$ та $Y = (y_1, y_2, \dots, y_n)$, метрика Мінковського визначається так:

$$d(X, Y) = \sqrt[q]{\sum_{i=1}^n |x_i - y_i|^q}, \quad (5)$$

де n – це розмірність об'єкта, x_i та y_i – значення i -того простору об'єкта X та Y відповідно і q – додатне ціле число (коли $q = 1$ – це *відстань Манхеттена*, коли $q = 2$ – це *відстань Евкліда*) [3].

Фільтрація за вмістом. Можна покращити індивідуальну рекомендацію знаючи більше про користувача, наприклад, демографічну інформацію; або про елемент, наприклад, інформація про продюсера та жанр музичного альбому. Рекомендаційні системи за вмістом надають рекомендації порівнюючи контент, що описує елемент, з контентом, що цікавить користувача.

Багато досліджень в цій сфері сфокусовані на рекомендації предметів з асоційованим текстовим вмістом, такий як: вебсторінки, альбоми, книги, фільми; де вебсторінки самі чи асоційований контент у вигляді опису чи відгуку користувача доступні. Декілька підходів вирішили цю проблему як завдання Отримання Інформації, де вміст асоціюється з уподобаннями користувача і розглядається як запит та неоцінені документи розраховуються з актуальністю/подібністю до цього запиту.

Гібридна фільтрація. З ціллю отримати по максимуму переваг з рекомендаційних систем, що використовують фільтрацію за вмістом та колаборативну фільтрацію, були запропоновані декілька гібридних підходів, що об'єднують найкраще з двох алгоритмів. Один простий підхід – це дозволити колаборативній фільтрації і фільтрації за вмістом створювати окремі рейтингові списки рекомендацій і потім об'єднувати їхні результати в кінцевий список.

Був запропонований загальний фреймворк для колаборативної фільтрації посиленої контентом, де передбачення основані на вмісті застосовані для конвертації розрідженої матриці оцінок користувача в повністю заповнену оцінками матрицю і потім метод колаборативної фільтрації використовується для здійснення рекомендації. Цей алгоритм працює краще за чисту колаборативну фільтрацію, фільтрацію за вмістом та лінійну комбінацію цих двох.

Висновок. Отже, існує багато алгоритмів рекомендаційних систем, які можна поділити на 3 великі великі групи: колаборативна фільтрація, фільтрація за вмістом та гібридна фільтрація. Колаборативну фільтрацію можна поділити на підмножини: фільтрація за сусідством та фільтрація за моделлю. Фільтрація за сусідством визначає подібність між користувачем та елементом, а потім розраховує зважену суму оцінок або зважене середнє значення для створення передбачення на значенні подібності. Фільтрація за моделлю використовує моделі, які потрібно тренувати, наприклад, Баєсові мережі, кластерні методики для створення передбачень. Також можливо використовувати фільтрацію за вмістом. Цей алгоритм використовує ознаки предметів для того, щоб порекомендувати користувачеві елементи, які подібні до тих, що подобаються користувачеві. Гібридна фільтрація, в свою чергу, поєднує в собі найкраще з

колаборативної фільтрації і фільтрації за вмістом, а також уникає їх основних недоліків.

Список використаних джерел:

1. A systematic review and research perspective on recommender systems // 3 травня 2022 [Електронне джерело]. – Режим доступу: <https://journalofbigdata.springeropen.com/articles/10.1186/s40537-022-00592-5>
2. Recommender Systems / Прем Мелвіль, Вікас Сіндгвані // 22 квітня 2010 [Електронне джерело]. – Режим доступу: <https://www.ime.usp.br/~jstern/miscellanea/seminario/Melville1.pdf>.
3. Survey of Collaborative Filtering Techniques / Шяююан Су, Тагі М., Хошгофтаар // 3 серпня 2009 [Електронне джерело]. – Режим доступу: <https://downloads.hindawi.com/archive/2009/421425.pdf>

УДК 004.4

ВИКОРИСТАННЯ RANDOM FOREST REGRESSOR ДЛЯ ПРОГНОЗУВАННЯ ЦІНИ АВТОМОБІЛЯ

Загацький В.В., магістрант

e-mail: ipzm222_zvv@student.ztu.edu.ua

Вакалюк Т.А., д. пед. н., професор

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Нині авто не є предметом розкоші, як це було деякий час тому, вирішуючи багато питань, таких як пересування з точки А в точку Б, перевезення вантажів та людей. Тому важко уявити людину, яка б не мала, чи не хотіла мати у володінні власний засіб пересування. Для інших, це одна з перших необхідностей, це джерело прибутків, завдяки яким вони можуть жити.

Автомобілісти хочуть купити, замінити автомобіль на більш новий або іншої моделі, деяким потрібен засіб пересування, іншим комерційне авто, яке зможе забезпечувати їхні потреби або запити компаній.

Як і для будь-якого товару, вони оцінюються за допомогою багатьох параметрів: рік випуску, типу двигуна, його потужності, коробки передач, комплектації. Тому однозначно вирішити, чи вартий цієї ціни автомобіль чи ні майже неможливо без досвіду у цій сфері. Це стосується як і покупців, так і продавців.

Основні матеріали дослідження. Для того, щоб оцінювати автомобілі правильно, потрібні розуміння та підготовка даних для побудови моделі. Дані аналізуються за допомогою алгоритмів для виявлення закономірностей, які використовуються у майбутньому для створення прогнозних моделей. Відповідно, штучний інтелект стає більш точним при обробці більшої кількості даних.

Random Forest Regressor – регресійна модель на основі ансамблевого навчання, або ж процесу використання кількох моделей, навчених на однакових даних, усереднення результатів моделей та знаходження більш точних результатів прогнозування або класифікації. Використовується “дерево рішень”, тобто декілька дерев рішень для створення ансамблевої моделі, яка і генерує прогноз. В цілому це працює за наступними кроками:

1. Вибираються випадкові вибірки з завантаженого набору даних
2. Для кожного зразка надається дерево вибору
3. Голосування по кожному очікуваному результату
4. Вибирається результат прогнозу з найбільшим числом голосів

Перевагами випадкових лісів можна назвати використання як для класифікації та задач регресії, обробка як лінійних так і нелінійних відношень, висока точність та збалансовані зміщення з дисперсією.

Однак одним із речей, які потрібно враховувати – досить великий час очікування проходження навчання при збільшенні кількості дерев, відповідно до того потрібні більші обчислювальні потужності для прискорення усіх процесів. Також недоліком даного методу є потреба у великих масивах початкових даних для високої точності результатів.

Для того щоб обробляти дані, їх необхідно перетворити з необроблених даних в зрозумілий формат. Частіше за все, в реальні дані мають багато неточних значень, що може призвести до побудови неточних моделей на основі них. Дана

проблема вирішується шляхом попередньої обробки – процесу зміни або перекодування інформації, у зрозумілому форматі для машини. Наприклад об'єм двигуна з 1598см^3 доцільніше перетворити на дробове значення 1,6.

Відповідно, перетворивши всі значення та проаналізувавши їх отримаємо результати найбільшого впливу на ціну від значень пробігу автомобіля, року випуску та типу коробки передач.

Після тренування моделі отримаємо близько 95% точності для оцінки автомобіля по заданих параметрах. Даний метод отримує найкращі результати, так як має найвищу точність відносно Bagging regressor та Linear regression

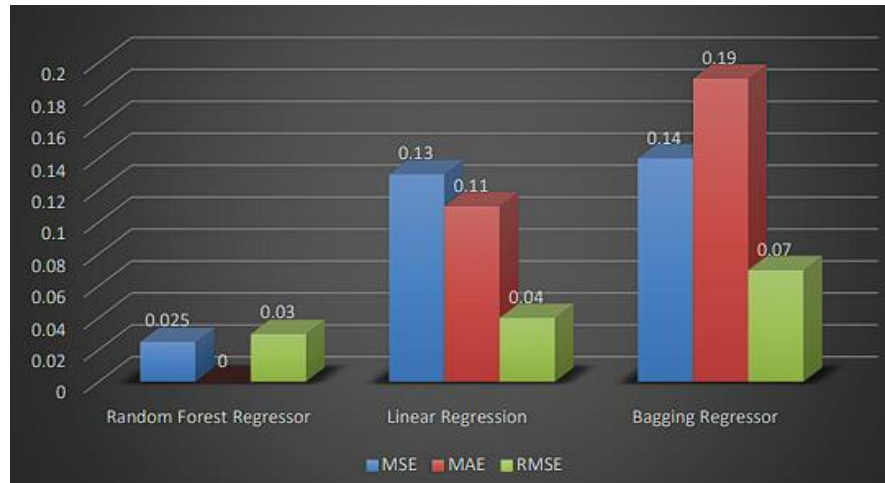


Рисунок 1. Графічна ілюстрація помилок моделей

Висновок. Впровадження технології машинного навчання для прогнозування цін автомобіля може значно підвищити довіру клієнтів до системи, в якій вона впроваджена. Завдяки високій точності можна проводити оцінку оголошень на відповідність ринковій ціні, що також допоможе знизити ризики махінацій над автомобілями. Це підвищить конкурентоспроможність інформаційної системи, конверсії і відповідно збільшить прибутки компанії.

Список використаних джерел

1. Used Cars Price Prediction and Valuation using Data Mining Techniques.URL: <https://scholarworks.rit.edu/cgi/viewcontent.cgi?article=12220&context=theses>.
2. Car Price Prediction – Machine Learning vs Deep Learning. URL: <https://www.analyticsvidhya.com/blog/2021/07/car-price-prediction-machine-learning-vs-deep-learning/>

УДК 004.891.2

ІНФОРМАЦІЙНО-ПОРАДНА СИСТЕМА ВИБОРУ МОВИ ПРОГРАМУВАННЯ

Засипко В.П., магістрант

e-mail: zasypkovlad209@gmail.com

Науковий керівник к.т.н., доц. Холодняк Ю.В.

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Одним із головних завдань кафедри "Комп'ютерні науки" є надання здобувачам вищої освіти нових знань та навичок у галузі інформаційних технологій. Для освоєння всіх сучасних рішень та реалізації різних подальших наукових проєктів, студентам необхідно правильно вибирати інструменти для реалізації своїх проєктів, наприклад, мову програмування. Вибір конкретної мови програмування в залежності від задачі, яку повинен вирішити студент, є досить важливим етапом, який зможе заощадити багато часу та сил. Виходячи з цього виникає потреба в розробці системи опитування студента, результати якої допоможуть останньому визначитися з необхідною йому мовою програмування.

Незважаючи на значну поширеність експертних систем у світі, систем які би виконували рекомендацію щодо мови програмування залежно від задачі, досить мало. Особливо серед пропозицій, що є на ринку експертних систем сьогодні. Так, наприклад, в роботі [1] була розроблена система, яка за допомогою серії питань, зменшує кількість варіантів запропонованих мов відповідно до відповідей користувача на ці питання. Але недоліками такої системи є застарілість та незручність для кінцевого користувача.

Метою статті є розробка на основі бази знань та бази даних експертної системи у вигляді програмного інтерфейсу, за допомогою якого користувач зможе взаємодіяти з базою знань.

Основні матеріали дослідження. Система, яка проводить опитування користувача і видає на основі опитування рекомендацію, буде будуватися на базі експертних систем.

Експертна система (ЕС) – предикативна система, що включає в себе знання про певну слабо структуровану предметну область і здатна пропонувати і пояснювати користувачеві розумні рішення [2]. За типом розв'язуваного завдання розроблена експертна система є інформаційно-дорадчою системою (ІДС). Така система забезпечує особу, яка приймає рішення, необхідною інформацією та рекомендаціями, що полегшують процес прийняття рішення [3].

Експертна система, розроблена в даній роботі, працює по принципам продукційної моделі представлення даних. Продукційна модель - це принцип представлення даних в експертній системі, який базується на так званих продукціях (правилах) «якщо (умова), то (дія)». Під умовою (антецедентом) розуміється деякий пропозиція-зразок, за якою здійснюється пошук в базі знань (БЗ). Під дією (консеквентом) розуміються операції, що виконуються при успішному результаті пошуку. Дії можуть бути проміжними, виступаючими далі в якості умов, і термінальними (або цільовими), такими, які завершають роботу системи [3]. У загальному випадку продукційну модель можна представити в наступному вигляді:

$$i = \langle S; L; A \rightarrow B; Q \rangle$$

де S – опис класу ситуацій;
 L – умова, при якій активізується продукція;
 $A \rightarrow B$ – ядро продукції;
 Q – постумова продукційного правила.

Принцип роботи продукційної моделі в даній ІДС наступний: програма складається з однієї обов'язкової (головної) умови та групи необов'язкових (уточнюючих) умов. Кожна з цих умов, у разі її виконання, надає вагові коефіцієнти відповідним у консеквенті мовам програмування. На рис. 1 наведено блок-схему продукційної моделі в ІДС.

Ваговий коефіцієнт – числовий коефіцієнт, що відображає значимість, відносну важливість даного фактора в порівнянні з іншими факторами [4].

Виконання або невиконання тих чи інших анцендентів залежить від дій особистості (вибору відповідних його інтересам опцій). Головна умова необхідна, щоб вибрати список (підходящих під обрану тему) мов програмування, а додаткові умови звужують діапазон варіантів.

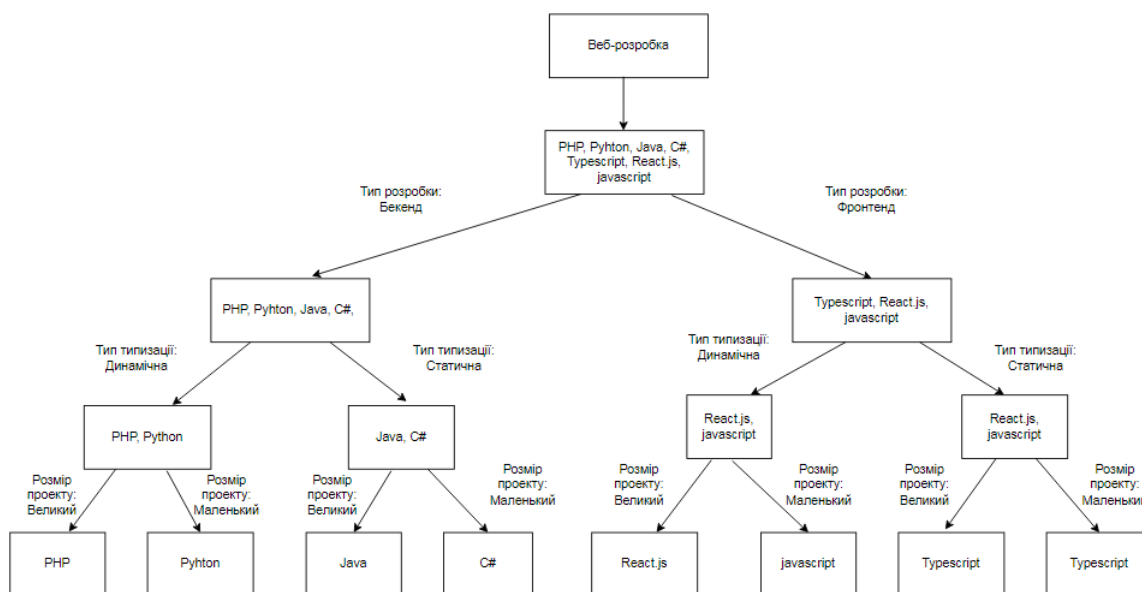


Рисунок 1. Приклад блок-схеми продукційної моделі в ІДС

В якості бази знань для зберігання та використання правил предметної області використовувався окремий файл з розширенням ".py" мови програмування Python. Самі правила зберігалися як змінні рядкового типу.

Для формування та зберігання фактів у структурованій формі використовувалася база даних реляційного типу. Реляційні бази даних - це бази даних, які використовуються для зберігання та надання доступу до взаємопов'язаних елементів інформації. У реляційних базах даних вся інформація зведена в таблиці, рядки і стовпці, якої називаються записами і полями відповідно [5]. В якості системи управління бази даних використовувалася СУБД sqlite, а саме вбудована бібліотека для роботи з реляційними СУБД в мові Python – sqlite3.

Програмний модуль розроблений за допомогою об'єктно-орієнтованої мови програмування Python в інтегрованому середовищі розробки Visual Studio Code на

основі відкритої бібліотеки для розробки комп'ютерних програм tkinter. Сама програма концептуально поділена на дві частини - головне вікно (рис. 2), в якому проходить основна взаємодія з програмою, та додаткове, в якому користувач проходить опитування (рис. 3).

Головне вікно візуально поділене на три функціональні блоки:

- блок варіантів проекту;
- блок керування програмою;
- блок виводу рекомендацій.

Перша частина складається з кнопок, які відображають теми розробки, що потенційно можуть зацікавити користувача. Друга частина це функціональні кнопки, за допомогою яких користувач може керувати програмою (почати опитування, очистити екран, зберегти результати роботи, тощо). Третя частина - це поля виводу рекомендацій, в яких виводиться результат опитування користувача. Цей блок складається з 9 полів, в кожне з яких виводиться своя частина рекомендації (рекомендована мова програмування, час на вивчення мови, додаткові бібліотеки для реалізації бажаного проекту, тощо).

Рисунок 2. Інтерфейс головного вікна

Додаткове вікно розділене на два функціональні блоки. Перший блок - це питання, які уточнюють вимоги користувача щодо обраної теми. Кожне питання відокремлене від інших рамкою і має кілька варіантів відповіді. Користувач може вибрати один з варіантів відповідей, тим самим зменшивши кількість варіантів пропонує мов програмування, додавши вагові коефіцієнти мов програмування, які відповідають зазначеному варіанту. Другий функціональний

блок - кнопка "Розрахувати", розташована внизу додаткового вікна. При натисканні на неї відбуватиметься розрахунок.

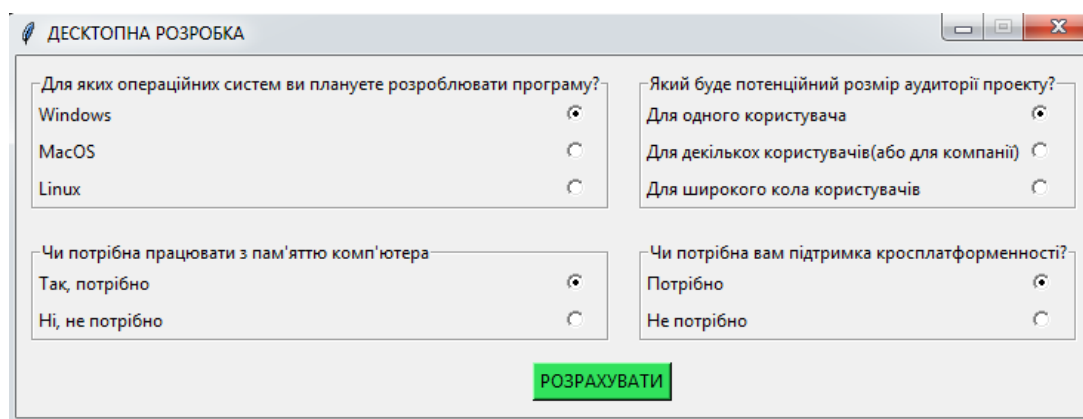


Рисунок 3. Інтерфейс додаткового вікна програми

Процес визначення найбільш відповідної мови програмування залежно від поставленого завдання складається з трьох етапів та проводиться за наступним алгоритмом.

Перший етап – визначення теми проекту в головному вікні. На цьому етапі користувачеві необхідно у головному вікні вибрати бажаний тип розробки. Для цього йому треба натиснути на одну з п'яти кнопок, що знаходяться у верхній частині головного вікна, відразу під написом "ОБЕРІТЬ ТИП ПРОЕКТУ" (рис. 4).

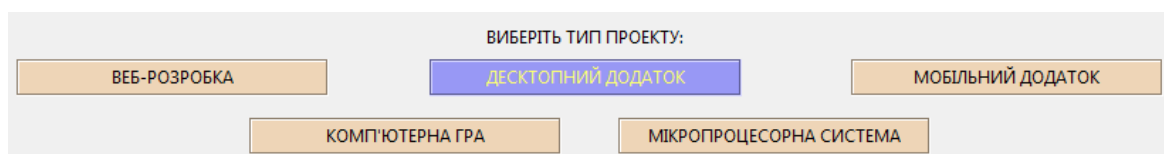


Рисунок 4. Блок вибору варіантів проекту

Кожна з 5 кнопок означає конкретну інформаційну область, в якій користувач теоретично хотів би працювати. (веб-додаток, десктопний додаток, мобільний додаток, комп'ютерна гра, мікропроцесорна система). Після вибору теми розробки користувачеві необхідно натиснути на кнопку "ДАТИ ПОРАДУ", яка знаходиться в центральній частині головного вікна.

Другий етап - уточнення вимог та розрахунок кінцевого результату. Після виконання першого етапу відкриється додаткове вікно програми, в якому користувачу потрібно або відповісти на питання по обраній темі, або одразу розрахувати результат, натиснувши відповідну кнопку "РОЗРАХУВАТИ"(рис. 3).

Після розрахунку порада у фрагментарному вигляді виводиться у відповідних полях на головному вікні (рис. 5).

ПОРАДИ ЩОДО ВИБОРУ МОВИ ПРОГРАМУВАННЯ		
ОПТИМАЛЬНА МОВА ПРОГРАМУВАННЯ:	ПРИБЛИЗНИЙ ЧАС НА РОЗРОБКУ:	ДОДАТКОВІ БІБЛІОТЕКИ(ЯКЩО НЕОБХІДНО):
№ 1 C#;	№ 1 2 місяця;	№ 1 Windows Forms;
ПРИБЛИЗНИЙ ЧАС НА ВИВЧЕННЯ:	ТИП ОПЕРАЦІЙНОЇ СИСТЕМИ:	КОРИСНА ДОДАТКОВА ДОКУМЕНТАЦІЯ:
№ 1 8 місяців;	№ 1 Windows MacOS;	№ 1 Windows Forms Programming in C#;
РЕКОМЕНДОВАНА СЕРЕДА РОЗРОБКИ:	МІНІМАЛЬНА КОМП'ЮТЕРНА КОНФІГУРАЦІЯ:	ОСОБЛИВОСТІ МОВИ ПРОГРАМУВАННЯ:
№ 1 Micosoft Visual Studio;	№ 1 CPU: AMD Ryzen 5 5500 RAM: 8 ГБ VideoCard: GeForce GTX 1050;	№ 1 Строга типізація ООП-орієнтований Підтримка .NET Framework Підтримка більшості продуктів в Microsoft Автоматична «скла

Рисунок 5 - Виведення результатів на екран

Третій етап-збереження результатів. Для збереження рекомендації в окремий текстовий файл для використання у подальшому, користувачу необхідно натиснути на одну з двох кнопок: "ЗБЕРЕГТИ РЕЗУЛЬТАТИ" або "АВТОМАТИЧНО ЗБЕРЕГТИ У ФАЙЛ". Різниця між цими двома кнопками в тому, що перший варіант дозволяє вибрати директорію для збереження та назву файлу, а другий автоматично зберігає рекомендацію в тій же директорії, де знаходиться програма під назвою "Порада.txt" (рис. 6).

Локальный диск (D:) ► Новая папка ► Информационно-поядча система ►				
Библиотеку ▼	Общий доступ ▼	Записать на оптический диск	Новая папка	
Имя	Дата изменения	Тип	Размер	
build	26.11.2022 10:52	Папка с файлами		
config.py	13.11.2022 17:59	Python File	1 КБ	
database.py	16.11.2022 23:15	Python File	2 КБ	
expert_system.db	16.11.2022 0:07	Data Base File	28 КБ	
IDS.py	25.11.2022 14:50	Python File	32 КБ	
IDS.spec	26.11.2022 10:48	Файл "SPEC"	1 КБ	
setup.py	18.11.2022 15:56	Python File	5 КБ	
weight_coefficient_vault.py	15.11.2022 23:58	Python File	5 КБ	
Информационно-доядча система.exe	26.11.2022 10:48	Приложение	9 470 КБ	
Пояда.txt	26.11.2022 10:53	Текстовый докум...	1 КБ	

Рисунок 6. Результат автоматичного збереження рекомендації у файл

В збереженому текстовому файлі зберігається порада щодо вибору мови програмування в структурованому вигляді (рис. 7).

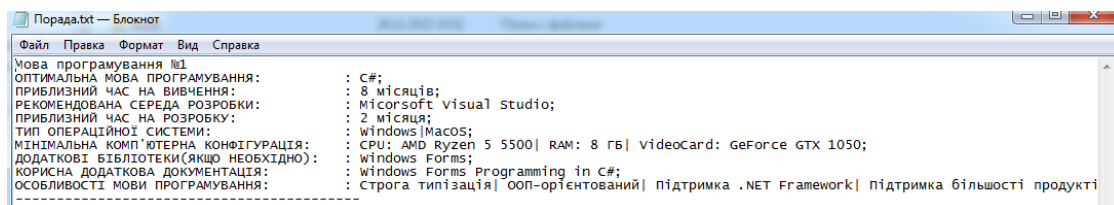


Рисунок 7. Вигляд поради в збереженому текстовому файлі

Висновок. Розроблена інформаційно-дорадча система дозволяє визначати оптимальну для користувача мову програмування, ґрунтуючись на його відповідях. Експертна система працює в двох режимах: режимі придбання знань та в режимі вирішення задачі (режим надання поради). В режимі придбання знань експерт, використовуючи компонент отримання знань, наповнює систему знань. В режимі опитування програма збирає дані про завдання користувача використовуючи діалоговий компонент, після чого ці дані надходять до робочої пам'яті. Система на основі даних з робочої пам'яті та правил з бази знань формує рекомендацію та виводить її на екран користувачу. Також система дозволяє зберегти результати опитування в окремий файл декількома способами.

Реалізація інформаційно-дорадчої системи вибору мови програмування в залежності від поставленого завдання дозволяє скоротити грошові витрати на кваліфікованих експертів у разі необхідності консультації та отримувати стабільні і точні результати у будь-який час та у будь-яких умовах.

Список використаних джерел:

1. Програма по выбору языка программирования. URL: <https://www.bibliofond.ru/view.aspx?id=551425#text> (дата звернення: 4.10.2022)
2. [https://www.tadviser.ru/index.php/Статья: Экспертные системы](https://www.tadviser.ru/index.php/Статья:_Экспертные_системы). URL: <https://www.tadviser.ru/index.php/> (дата звернення: 10.10.2022)
3. Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем: СПб: Питер, 2000. 384 с.
4. Продукционная модель представления знаний URL: [https://ru.wikipedia.org/wiki/Продукционная модель представления знаний](https://ru.wikipedia.org/wiki/Продукционная_модель_представления_знаний) (дата звернення: 22.10.2022)
5. Способ расчета весовых коэффициентов вершин семантической сети научного текста URL: <https://fundamental-research.ru/ru/article/view?id=30087#знаний> (дата звернення: 30.10.2022)
6. Базы данных: что это такое, и какие они бывают URL: <https://fundamental-research.ru/ru/article/view?id=30087#> (дата звернення: 9.11.2022)

УДК: 004.4

СТВОРЕННЯ ЗАСОБУ РОЗУМІННЯ МОВИ ЖЕСТІВ ЗА ДОПОМОГОЮ НЕЙРОННИХ МЕРЕЖ.

Ващенко К.Я., студент

e-mail: kostya.vaschenko@gmail.com

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Відеозв'язок набирає обертів, стає доступним в більшості країн світу, даючи змогу розмовляти і бачити співбесідника, який знаходиться на іншому куточку планети. Проте великому відсотку людей з обмеженими можливостями слуху та мови дана технологія не є доступною. Розпізнавання жестів та особливо жестової мови в системах відеокommunікацій може подолати даний бар'єр і вивести відеозв'язок на новий рівень.

Розпізнавання жестової мови важливе для автоматизації процесів спілкування між глухою і чуучою спільнотами. Наразі більша частина таких комунікацій в значній мірі залежить від послуг перекладу, який виконується людьми, що не завжди зручно. Автоматичне розпізнавання мови жестів має на меті розуміння значення знаків без допомоги експертів. Жести можна перетворити в звук або текст в залежності від потреб кінцевих користувачів. Розпізнавання мови жестів важливе для надання рівних можливостей кожній людині і поліпшення суспільного добробуту. Проте, незважаючи на численні дослідження, проведені протягом останніх кількох десятиліть, розпізнавання жестів, як і раніше, є складною проблемою [1].

Основні матеріали дослідження. Перед розпізнаванням жесту необхідно виділити ознаки зображення. Після цього слід застосувати будь-який метод класифікації. Для класифікації та розпізнавання потрібна велика кількість ознак. Звичайні моделі розпізнавання образів обмежені в своїй здатності обробляти природні дані в їх початковому вигляді. Протягом десятиліть створення системи розпізнавання образів або машинного навчання вимагало ретельного проєктування і значного досвіду в предметній області, щоб розробити засіб екстракції ознак, що перетворював би необроблені дані (наприклад, значення пікселів зображення) в підходяще внутрішнє подання або вектор ознак, з якого підсистема навчання, часто класифікатор, може виявляти або класифікувати образи вхідних даних [3].

Процес розпізнавання жестів з веб-камери зображений на Рисунок 1.1 та складається з трьох кроків:



Рисунок 1. Процес розпізнавання жестів з веб-камери

1) Захоплення зображення. Користувач запускає програму та показує жести. Кожну секунду веб-камера захоплює по 30 кольорових зображень, кожне з яких передається на попередню обробку.

2) Попередня обробка. Отримане кольорове зображення перетворюється у сіре монохромне, також змінюється його розмір. Нормалізується значення пікселів до діапазону від 0 до

3) Розпізнавання. Оброблене зображення передається на нейромережу. Вона визначає жест та повертає його переклад [2].

Кількість літер в українському дактильному алфавіті відповідає кількості у звичайному українському алфавіті. Алфавіт містить як статичні жести, так і динамічні. Деякі динамічні жести у статичному вигляді відповідають іншим статичним жестам, а оскільки датасет складається зі статичних зображень, то такі жести будуть виключені. В набір увійде 29 жестів, літери «г», «і», «й», «щ» за своєю схожістю можуть бути замінені на «г», «і», «и», «ш».

На сьогодні зазначена тема є досить актуальною, оскільки незважаючи на наявність наукових робіт та статей по цій темі, систем розпізнавання української мови жестів, зокрема дактильного алфавіту (відображення слів побуквенно), у відкритому доступі немає. Такі програми необхідні для полегшення комунікації між людьми з вадами слуху та іншими та можуть застосовуватися як для простого перекладу, так і для навчання.

Висновок. Після дослідження існуючих методів машинного навчання та їх застосування у вирішенні проблеми розпізнавання жестів, а також ретельного аналізу існуючих рішень було обрано декілька моделей нейронних мереж, що відповідають поставленим задачам. Було підібрано датасет розмірністю в 46 жестових слів для тренування нейронної мережі.

На базі обраних моделей нейронних мереж було розроблено програмний комплекс, що складається з модулю для навчання та адаптації нейронної мережі під мобільні пристрої та власне мобільного додатку для операційної системи Android у вигляді відео-месенджера з можливістю розпізнавання жестів. Для роботи логіки розпізнавання було підключено бібліотеку TensorFlow, а для реалізації графічного інтерфейсу й передачі відеоданих було застосовано Android SDK та фреймворк Mesibo API відповідно.

Список використаних джерел та літератури:

1. T. Starner, J. Weaver, A. Pentland. Real-time American sign language recognition using desk and wearable computerbased video. Pattern Analysis and Machine Intelligence [Electronic resource]. – 1998. – Access mode: <https://ieeexplore.ieee.org/document/735811/> (lastaccess: 04.06.2021). Title from the screen.
2. Давидов М.В., Нікольський Ю.В. Методи та засоби опрацювання зображень реального часу для ідентифікації елементів жестової мови / М.В. Давидов, Ю.В. Нікольський // *Штучний інтелект, науково-технічний журнал, Державний університет інформатики і штучного інтелекту*. – 2008. – № 1. – С. 131- 138.
3. Сіряк Р.В. Модель обробки поточкових даних для розпізнавання окремих одиниць жестової мови / Р.В. Сіряк, І.С. Скарга-Бандурова // *Вісник Національного технічного університету «ХПІ». Серія: Інформатика та моделювання*. – 2018. – № 42. – С. 73-81

УДК 681.3

ЗАБЕЗПЕЧЕННЯ ПРИНЦИПІВ FAIR У СЕМАНТИЧНИХ WIKI-РЕСУРСАХ ДЛЯ ПІДТРИМКИ ВІДКРИТОЇ НАУКИ В УКРАЇНІ

Рогушина Ю.В., к.ф.-м.н.

e-mail: ladamandraka2010@gmail.com

Інститут програмних систем НАНУ, м.Київ

Актуальність. Сучасні наукові дослідження потребують використання достовірної та актуальної інформації, і відкритий доступ в електронній формі до результатів раніше проведених робіт дозволяє зробити такі дослідження більш ефективними. Відкрита наука (Open Science), за визначенням Організації економічного співробітництва та розвитку (OECD), – це в першу чергу доступність (у цифровому форматі, без обмежень або з мінімальними обмеженнями) результатів наукових досліджень [1]. Така доступність має забезпечувати відкритий доступ до наукових публікацій, відкриті дані експериментів та відкрите рецензування наукових досліджень та відкрите програмне забезпечення, відкриті електронні освітні ресурси тощо. Відкрита наука є відображенням принципових зміни у наукового-дослідного процесі, що пов'язані переходом від традиційних методів публікації результатів досліджень до розповсюдження результатів досліджень на основі сучасних інформаційних технологій [2].

Відповідно до розпорядження Кабінету Міністрів України від 8 жовтня 2022 р. затверджено Національний план щодо розвитку відкритої науки (Open Science), в якому передбачаються різні організаційні заходи, що спрямовані на забезпечення відкритого доступу до результатів наукових досліджень та інтеграцію з Європейською хмарою відкритої науки (European Open Science Cloud – EOSC). Це віртуальне середовище з вільним доступом для збереження, керування, аналізу і передачі даних із всіх областей знань. Створення EOSC спрямоване на розробку інструментів та сервісів, що дозволяють дослідникам знаходити, отримувати доступ, повторно використовувати та комбінувати результати наукової діяльності.

Парадигма Відкритої Науки є спробою світової наукової спільноти розв'язати проблему наукової невідтворюваності (scientific irreproducibility) [3]. «Наукова невідтворюваність – неспроможність повторити чужі експерименти та дійти того ж висновку – [4]. Для цього запропоновано базові принципи, на яких повинні ґрунтуватися наукові дослідження:

- *Відкритий доступ.* Тобто результати досліджень, наукові публікації, які поширюються онлайн і без затрат або інших перешкод, повинні мати вільний доступ.
- *Відкрита наука.* Дослідники діляться своїми методами, програмним кодом та даними досліджень через централізовані спеціалізовані репозиторії.
- *Відкриті дані.* Дані повинні бути вільно доступні кожному для використання і повторного аналізу і публікації на свій розсуд, без обмежень з боку авторського права, патентів або інших механізмів контролю.

Портал EOSC був створений в 2016 році з метою подолання різноманітності й об'єднання наукової інфраструктури. Зараз EOSC – це сукупність репозиторіїв даних і пов'язаної з ними інфраструктури, що потрібна для підтримки наукових

досліджень [5]. Одним з напрямків розвитку EOSC є забезпечення пошуку, доступності, інтероперабельності та повторного використання цифрових об'єктів, які можуть містити як результати експериментів, так і тексти наукових публікацій. Набір таких вимог щодо даних позначають акронімом FAIR (Findable, Accessible, Interoperable, Reusable).

В інформаційних ресурсах (IP), що відповідають принципам FAIR, функції пошуку, здобуття і представлення даних реалізують не користувачі, а сама інформаційна система, але до керування науковими даними залучаються як науково-дослідні організації та навчальні заклади, так і окремі дослідники. Крім того, ці принципи враховують специфіку інформації, що стосується процесу та результатів наукових досліджень. Керівні принципи FAIR не потребують якій-небудь стандартизації чи конкретної технології їхньої підтримки. Принципи виступають як керівництво для створення даних для озер даних з урахуванням функціональності їх пошуку, приступності, сумісності і повторного використання.

Основні принципи FAIR відображають різні аспекти вимог до даних:

1. Пошук (*Findable*). Щоб використовувати дані, необхідно не тільки мати доступ до них, але їх спочатку потрібно знайти серед інших інформаційних об'єктів (IO). Тому виникає низка вимог щодо метаданих, які описують кожен такий IO, та засобів їх автоматизованого аналізу. Саме виразність метаданих та їх структура є базовими факторами знаходження пертинентних відповідей на запити користувачів.
2. Доступ (*Accessible*). Дані та метадані можна отримати за їх ідентифікатором за допомогою стандартизованого відкритого протоколу зв'язку, який припускає процедуру аутентифікації та авторизації, якщо це необхідно
3. Інтероперабельність (*Interoperable*): дані, що використовуються, потребують інтеграції з іншою інформацією та взаємодії із застосунками або робочими процесами для аналізу, зберігання та обробки, використовують загальноприйняті стандарти подання та посилання на інші дані та метадані.
4. Повторне використання (*Reusable*). Метадані та дані повинні бути детально описані набором точних і відповідних атрибутів з чіткою та доступною ліцензією на використання та відповідати стандартам спільноти, що стосуються ПрО.

На рішення подібних проблем націлена ініціатива Go FAIR, що містить базові принципи поліпшення можливостей пошуку, забезпечення доступу до даних, їхньої сумісності і, що особливо важливо, повторного використання.

Згідно FAIR, функції пошуку, здобуття і представлення даних реалізують не користувачі, а інформаційна система. При цьому мова йде не тільки про самі дані і метадані, але і про алгоритми й інструменти керування ними. Крім того, до розробки підходів до керування науковими даними залучаються всі зацікавлені сторони: науково-дослідні організації й окремі вчені; оператори баз даними і видання, що публікують наукові статті і результати експериментів; організації, що фінансують ці наукові дослідження; виробники програмного забезпечення й інструментів обробки даних; компанії, що надають послуги з аналізу й інтерпретації даних. Важливо, що в коло зацікавлених сторін також включаються самі обчислювальні системи (алгоритми обробки даних) як самостійний об'єкт — у залежності від їхнього рейтингу приймається рішення про включення обчислювального методу до конфігурації.

Для оптимізації доступу до даних потрібно керуватися наступними засадами: дані (метадані) можуть бути отримані за їх ідентифікатором за

стандартизованими протоколами зв'язку; протокол доступу до даних – відкритий і припускає використання уніфікованого протоколу доступу — за необхідності для доступу до даних використовується процедура аутентифікації й авторизації, а метадані можуть бути доступні навіть за відсутності доступу до самим даним.

Має бути забезпечена сумісність даних не тільки з іншими даними, але і з застосунками й інструментами для їх аналізу, збереження й обробки: дані (метадані) використовують формальну, доступну і поширену мову опису даних; дані (метадані) використовують словники, що реалізовані відповідно до керівного принципами FAIR; дані (метадані) містять у собі повні посилання на інші дані (метадані).

Кінцева мета FAIR — оптимізація повторного використання даних та їх об'єднання в різних задачах: дані (метадані) докладно описують із застосуванням набору однозначних і релевантних атрибутів; дані (метадані) супроводжуються чіткою і доступною ліцензією на їхнє використання; дані (метадані) мають детальну історію їхнього походження; дані (метадані) подаються у відповідності зі стандартами тематичного наукового співтовариства.

Представлені елементи даних і метаданих взаємопов'язані, але при цьому незалежні і віддільні. Кожний з них визначає сукупність метрик (характеристик) – вимог, які передаються ресурсам, інструментам, словникам обробки даними з метою забезпечення їхнього повторного використання третіми сторонами, у тому числі не мають прямого відношення до науки. При цьому існує можливість керувати рівнем входження в озера даними FAIR тих чи інших користувачів за рахунок градації при визначенні характеристик наданих ресурсів. Варіюючи і комбінуючи метрики опису об'єктів, можна досягти високого ступеня адаптивності представлення даних і метаданих в інформаційній системі. .

Зараз багато європейських дослідницьких інфраструктур використовують концепцію FAIR для надання доступу до своїх наукових даних, але більшість з них орієнтуються на загальні набори метаданих, які не призначені для підтримки проблемно-орієнтованого пошуку з використанням знань ПрО. Інфраструктура відкритої науки ЄС інтегрує велику кількість спеціалізованих відкритих репозиторіїв та сховищ даних, які надають інструменти і механізми для керування контентом, але використовують різні схеми метаданих. Для цього EOSC застосовує спільну розробку стандартів та впровадження практики Open Science шляхом взаємодії з дослідницькими спільнотами, а також адаптація для використання науковими співтовариствами таких технічних компонентів екосистеми FAIR, структури метаданих для цифрових об'єктів FAIR.

Одним з напрямів розбудови відкритої науки є забезпечення відкритого доступу до наукових результатів та науково-технічної інформації шляхом використання хмарних платформ для проведення фундаментальних та прикладних досліджень широкими колами науковців. Крім широкого спектру спеціалізованих сервісів аналізу даних, які потрібні для досліджень, однією з базових функцій такої системи є *пошук інформації* на змістовному рівні як одна з базових складових інфраструктури відкритої науки. Це має забезпечити повторне використання результатів раніше виконаних досліджень в різних галузях, результати експериментів та розрахунків тощо.

Однією з передумов для цього є створення портальних версій науково-технічних ресурсів з відкритим доступом – електронних бібліотек, галузевих довідників та енциклопедій, архівів та наукометричних баз даних. Щоб забезпечити їх ефективне використання, виникає необхідність у засобах навігації та пошуку в цих ресурсах на змістовному рівні, з використанням специфічних

знань тих предметних областей (ПрО), до яких відносяться дослідження. Це потребує використання додаткових джерел знань для семантичної розмітки інформації, що відрізняються від універсальних стандартів подання метаданих, таких як Dublin Core [6].

Великий потенціал для такого доступу до інформації забезпечують семантично розмічені Wiki-ресурси, які можуть використовувати для побудови семантичної розмітки онтології тих ПрО, що пов'язані з розв'язанням прикладних задач та відповідають персоніфікованим інформаційним потребам окремих користувачів. Такі ІР не є альтернативою електронним бібліотекам з відкритим доступом до публікацій та порталам наукових установ та спільнот, а також сервісам, що забезпечують функціонування інфраструктури відкритої науки, але вони можуть стати більш зручним засобом для пошуку інформації, яка представлена в цій інфраструктурі.

Постановка проблеми. З метою підвищення якості інформаційного забезпечення відкритої науки пропонуємо провести дослідження й експериментальні розробки в області інтелектуалізації інформаційних систем та ресурсів на основі семантичного розширення Wiki-технологій та застосування методів онтологічного аналізу. Обґрунтування такого вибору потребує аналізу відповідності функціоналу Wiki-ресурсів вимогам FAIR та дослідження можливостей його масштабування.

Семантичний пошук у відкритому середовищі. У найбільш широкому розумінні наукова проблема, що стосується даного дослідження, полягає у тому, що пошук інформації у відкритому інформаційному середовищі потребує використання формалізованих знань про ту ПрО, що цікавить користувача. Для складних ПрО, що стосуються сучасних наукових досліджень, такі знання мають великий обсяг, складну структуру та значною мірою залежать від специфіки наукових напрямків та суб'єктивних уподобань окремих дослідників. Тому наявність відкритих наукових ресурсів та принципова досяжність даних, що у них зберігаються, є необхідною, але не достатньою умовою ефективного повторного використання результатів досліджень.

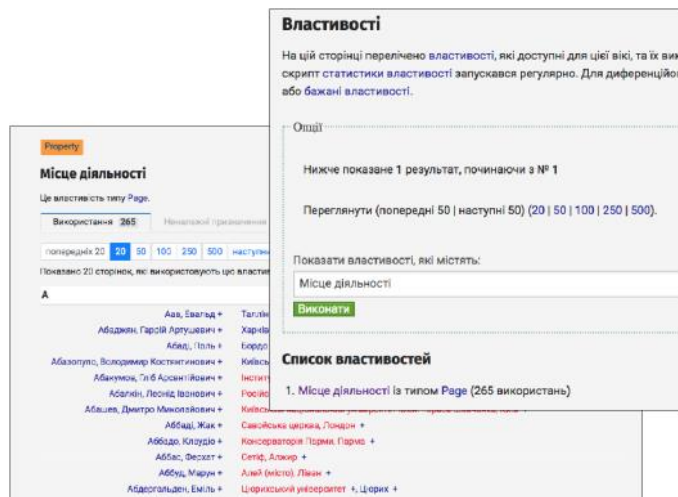
Семантичні Wiki-ресурси та FAIR.

Семантичні розширення технології Wiki (наприклад плагіну Semantic MediaWiki (SMW) для MediaWiki) [7] дозволяють створювати ІР, що відповідають вимогам FAIR. Але для цього слід використовувати ці технології разом із засобами керування знаннями та онтологічними моделями відповідних ПрО. Такі моделі базуються на Wiki-онтологіях – спеціальних випадках онтологій ПрО із набором обмежень щодо аксіом та характеристик відношень між поняттями [8]. Більш детально це проаналізовано в [9].

Використання для цього семантичного розширення Wiki-технології [10], яке історично себе показало досить потужним інструментом, має широку спільноту використання, відкритий код та відкриті принципи розробки і підтримки, постійно розвивається, – є найбільш оправданим вибором.

Розглянемо, як виконуються вимоги FAIR у семантичних Wiki-ресурсах:

1. *Пошук:* SMW надає можливості семантичного пошуку на основі семантичних властивостей та категорій окремих Wiki-сторінок, які можуть розглядатися як гнучкий набір метаданих. Забезпечується можливість машинної обробки таких метаданих. Дані та метадані пов'язані з окремими Wiki-сторінками, що мають глобальний унікальний і постійний ідентифікатор.



2. *Доступ*: Дані та метадані можна отримати за їх ідентифікатором за допомогою стандартизованих відкритих протоколів http та https; метадані доступні, навіть якщо дані більше не доступні.
3. *Інтероперабельність*: дані та метадані в SMW використовують стандарти Semantic Web, експорт даних в XML, RDF, а за можливості встановлення інших додаткових плагінів – і в OWL або інші спеціалізовані стандарти; експорт результатів семантичного пошуку в формат RDF входить до функціоналу SMW; дані та метадані в SMW включають посилання на інші метадані.
4. *Повторне використання* даних в SMW забезпечується механізмами семантичних властивостей та категорій, які можна відтворювати та комбінувати в різних налаштуваннях, а можливість додавати необмежену кількість атрибутів та категорій кожній сторінці забезпечує вимогу FAIR щодо того, що метадані повинні бути детально описані множиною точних і відповідних атрибутів.

Масштабованість Wiki-технології. Через те, що інформаційні ресурси відкритої науки можуть мати великий обсяг та складну структуру, виникає потреба у тому, щоб такі ІР були масштабованими та могли ефективно виконувати свої функції у випадку збільшення об'єктів обробки та розширення їх структури. При цьому багато питань потребують не тільки теоретичного дослідження, але й практичних випробувань.

Крім тих аспектів масштабування, що є спільними для всіх розподілених інформаційних систем, у створенні систем, що орієнтовані на аналіз інформації на семантичному рівні, необхідно особливу увагу звертати на форми подання знань предметної області та засоби їх обробки, які, з одного боку, дозволяють відображати специфіку цієї області, а з іншої – придатні для виконання семантичного пошуку за прийнятний час за умов збільшення елементів бази знань інформаційного ресурсу та ускладнення її структури. Для цього необхідно виокремити ті фактори, які впливають на масштабованість семантичних ІР, та визначити умови, що забезпечують успішний розвиток такого ресурсу.

Практичний досвід розробки e-BYE (vue.gov.ua) – семантизованого Wiki-ресурсу зі складною структурою, який реалізовано на технологічній платформі MediaWiki та її семантичного розширення Semantic MediaWiki, дозволив визначити набір рекомендацій, дотримання яких забезпечує ефективне масштабування Wiki-ресурсу. Основними факторами, що впливають на це, є:

- Схема бази знань, тобто система семантичних властивостей, що використовуються для структурування контенту Wiki-сторінок, яка визначає потенційну виразність таких запитів;

- Вчасна індексація змін у контенті (як у наборах семантичних відношень, так і у самих сторінках), яка, з одного боку, не повинна знижувати продуктивність роботи IP, а з другого – забезпечувати актуальність бази даних;

- Вчасне видалення сторінок та елементів інфраструктури, що не використовуються;

- Якість виконання семантичної розмітки (відсутність помилок у назвах властивостей, семантизація посилань на інші сторінки, коректне введення значень властивостей);

- Наявність вбудованих запитів, що відповідають типовим (часто повторюваним) інформаційним потребам користувачів;

- Кількість вбудованих запитів та кількість звертань до сторінок з такими запитами;

- Зручне представлення результатів запитів;

- Розташування семантичних запитів, що вбудовані у Wiki-сторінки, таким чином, щоб результати запитів надавалися саме в тому місці IP, де користувачам потрібна така інформація.

В процесі розробки Wiki-ресурсу доцільно дотримуватися наступних вимог:

- Контролювати розмір мультимедійних IO, що використовуються в IP;

- Передбачати засоби масового імпорту інформації із зовнішніх джерел у форматі IP;

- Контролювати загальну кількість Wiki-сторінок та видаляти непотрібні, помилково створені сторінки та сторінки-дублі;

- Уніфікувати метаописи мультимедійних IO, щоб уникнути дублювання збережених файлів (наприклад, ті самі зображення можуть використовуватися на різних сторінках IP);

- Розробляти шаблони ТІО, щоб уникнути збільшення кількості подібних імен семантичних властивостей та помилок в цих іменах та спростити сприйняття інформації користувачами;

- Створювати шаблони, які інтегрують за допомогою запитів контент різних сторінок IP;

- Формалізувати структуру бази знань IP та інтероперабельно визначити семантику відношень між сторінками, яка використовується у семантичній розмітці ресурсу (вбудованих можливостей Semantic MediaWiki для цього недостатньо, і тому для цього доцільно застосовувати різноманітні зовнішні системи організації знань на основі онтологій);

- Визначити семантику гіперпосилань між сторінками IP та створити відповідні семантичні властивості, явно описуючи їх область значення, область визначення та зміст;

- Розробляти шаблони для введення та подання значень семантичних властивостей ТІО, явно описати категорії сторінок, для яких вони мають використовуватися;

- Для вбудованих семантичних запитів, що викликаються на декількох різних сторінках, розробляти відповідні шаблони;

- Визначити доцільність підключення розширень (плагінів), які розширюють функціонал обробки, і не встановлювати ті з них, в яких немає реальної потреби;

- Розробити адекватну політику індексування контенту, який враховує частоту оновлення інформації та кількість відвідувань користувачами;
- Створювати запити без надлишкових умов;
- Мінімізувати інтегровані до ресурсу зовнішні програмні засоби (такі як лічильники відвідувань сторінок);
- Зменшувати кількість семантичних запитів та ІО на тих сторінках, які користувачі відвідують найчастіше.

Типові інформаційні об'єкти (ТІО) відкритої науки. Пошукові сервіси мають не тільки забезпечувати доступ до різних типів інформації (природномовних документів і мультимедійних даних) і підтримувати здобуття знань зі знайденої інформації, але й надавати можливість знаходження складних інформаційних об'єктів, що відповідають вимогам користувачів. Прикладами таких об'єктів є:

- різноманітні обчислювальні засоби та ресурси, що можуть бути отримані в тимчасове користування,
- наукові колективи територіально розподілених дослідників, які об'єднуються для виконання спільного проекту;
- створення колективних монографій або тематичних видань, публікація статей у найбільш релевантних виданнях.

Тому важливою часткою досліджень є створення системи метаданих для опису не тільки інформаційних ресурсів відкритої науки, але й усіх її об'єктів та суб'єктів. Така модель метаданих, що використовує знання із зовнішніх онтологій предметних областей, організаційних онтологій, онтологій професій та компетенцій (таких, як ESCO), а також різноманітні національні та міжнародні стандарти, має стати основою для семантичної розмітки та пошуку всіх типів інформації, що генерується, зберігається та передається інфраструктурою відкритої науки.

Крім загальноновживаних онтологічних описів для створення метаданих доцільно генерувати персоніфіковані онтології, що відображають уявлення про звіт різних наукових спільнот та окремих дослідників. Щоб інтегрувати ці персоніфіковані уявлення, доцільно розробляти методи та програмні засоби співставлення та вирівнювання таких онтологій зі спільною онтологією наукового співтовариства. Це дозволить, з одного боку, явно представляти відмінності поглядів різних наукових груп та напрямків, а з іншого – забезпечити ефективний пошук потрібної інформації незалежно від того, яку саме терміносистему обирають для метаданих розробники окремих ресурсів.

Приклади використання такого підходу – інтеграція національних та міжнародних рамок кваліфікацій, формування багатонаціональних територіально розподілених дослідницьких колективів, пошук навчальних курсів дистанційної освіти та інтеграція ринку освітніх послуг із ринком праці.

Важливою особливістю запропонованого підходу є те, що, хоча розробка моделі метаданих виконується спеціалістами з онтологічного інженірингу у співпраці з експертами ПрО, але використання технології Wiki дозволяє вільно використовувати цю модель широким колом користувачів (як для розмітки, так і для пошуку інформації) без спеціальних знань та без потреби у встановленні додаткового програмного забезпечення.

Механізм Wiki-шаблонів дозволяє фіксувати структуру ТІО, що характерні для ІР та формалізовані у Wiki-онтології. У розробці Wiki-довідників відкритої науки можна вирізняти дві групи ТІО, що базуються на різних онтологіях:

1. загальні ТІО на основі організаційної онтології наукових досліджень (такі, як Персоналія, Публікація, Проект, Організація та їх підтипи);

2. специфічні для ПрО, що використовують онтології відповідної сфери досліджень і характеризують тематику та тип виконаної роботи або сферу інтересів дослідника. Важливо, що створення ІР для іншої ПрО не потребує внесення додаткових змін до налаштувань інформаційної системи, а тільки перехід до іншої структури понять та відношень між ними, що отримуються із довільної зовнішньої онтології.

Висновок. Проведений аналіз показав, що семантичні Wiki-технології, що запропоновані для підтримки відкритої науки в Україні, відповідають таким основним вимогам:

- використання відкритих стандартів та відкритого програмного забезпечення;
- забезпечення принципів FAIR щодо подання даних та метаданих;
- масштабованість;
- використання знань предметної області із зовнішніх баз знань.

Крім того, досвід використання такої технологічної платформи для створення е-ВУЕ підтвердив її ефективність та надійність.

Створення семантичної розмітки на основі використання відкритих форматів подання знань Semantic Web є основою для інтеграції наукових інформаційних ресурсів України у світове/європейський науковий інформаційний простір. Практичні результати цього, наприклад, мають забезпечити інтеграцію з науковими виданнями України та з порталом е-ВУЕ.

Список використаних джерел

1. OECD Science, Technology and Industry Policy Papers № 25 Making Open Science a Reality. URL: <https://cutt.ly/JcuwQtg>.
2. Open innovation, open science, open to the world - a vision for Europe. URL: <https://ec.europa.eu/digital-single-market/en/news/open-innovation-open-science-open-world-vision-europe>.
3. Challenges in irreproducible research, Nature, 18-10-2018, URL: <https://www.nature.com/collections/prbfkwmwvz/>.
4. Baker, 1,500 scientists lift the lid on reproducibility. Nature, 533(7604): 452-454. (2016) doi:10.1038/533452a, URL: <https://www.nature.com/articles/533452a>.
5. EOSC Multi-Annual Roadmap 2023-2024 - https://eosc.eu/sites/default/files/2022-05/20220523_MAR_02_GL.pdf.
6. Dublin Core Metadata Initiative. Dublin Core Metadata Element Set, version 1.1. (2014) URL: <http://dublincore.org/documents/dces/>.
7. MediaWiki. Available from: <https://www.mediawiki.org/wiki/MediaWiki>.
8. *Рогущина Ю.В.* Використання систем організації знань на основі онтологій у WIKI-ресурсах. *Проблеми програмування*, 2021, №4. – С.23-33. DOI: <https://doi.org/10.15407/pp2022.01>.
9. *Рогущина Ю.В., Гришанова І.Ю.* Дослідження принципів, моделей та методів парадигми менеджменту наукових даних FAIR для аналізу метаданих big data. *Проблеми програмування*, 2021, №4. – С.26-35. DOI: <http://doi.org/10.15407/pp2021.04.026>.
10. Manual:What is MediaWiki? Available from: https://www.mediawiki.org/wiki/Manual:What_is_MediaWiki%3F [Accessed 18/11/2021]

УДК 681.3

ВИКОРИСТАННЯ ПАРАДИГМИ МУЛЬТИАГЕНТНИХ СИСТЕМИ ДЛЯ ПЛАНУВАННЯ ПОВЕДІНКИ КОМАНДИ РОЮ ДРОНІВ

Рогушина Ю.В., к.ф.-м.н.

e-mail: ladamandraka2010@gmail.com

Інститут програмних систем НАНУ, м.Київ

Гладун А.Я., к.т.н.

e-mail: glanat@yahoo.com

*Міжнародний науково-навчальний центр інформаційних технологій та систем
НАНУ та МОНУ, Київ, Україна,*

Актуальність. Безпілотні літальні апарати (БПЛА) викликають дедалі більший інтерес як у військових, промислових так і в дослідницьких областях через велику кількість сценаріїв і програм, які вони можуть підтримувати і виконувати в динамічному/невизначеному середовищі. Одним із великих викликів сьогодення є застосування роїв БПЛА для використання переваг, які можуть надати скоординовані дії рою таких пристроїв. Кожен БПЛА може розглядатися як окремий інтелектуальний агент, який має певні цілі, ресурси та стратегії. Планування дій та оцінка навколишнього середовища дроном може використовувати елементи штучного інтелекту (ШІ).

Значно підвищити ефективність використання БПЛА дозволяє концепція “рою дронів”, яку американський Інститут сучасної війни пов’язує із технологією “роїв масового знищення”, що спочатку розроблялася для балістичних ракет. Спільна діяльність групи БПЛА доцільно аналізувати як МАС, в якій окремі раціональні агенти можуть обмінюватися інформацією та виконувати спільну задачу, оптимально розподіляючи цілі. Наприклад, внаслідок такого обміну два дрони можуть атакувати різні об’єкти для нанесення максимальної шкоди або навпаки спільно атакувати один об’єкт для його гарантованого ураження. При цьому вибір об’єктів здійснюється динамічно на основі вбудованих правил та аналізу поточного стану.

Для полегшення співпраці дронів у рої їм необхідно обмінюватися інформацією з її однозначним розумінням, однак неоднорідність представлення інформації, що застосовується різними БПЛА, є основною перешкодою. Тому ми пропонуємо модель онтології, для вирішення неоднорідності інформації та дозволяє БПЛА однаково використовувати інформацію в процесі взаємодії.

Сьогодні БПЛА, або дрони, привертають велику увагу різних сферах застосування – як військових (спостереження та атаки), так і цивільних (наприклад, у сільському господарстві, для моніторингу навколишнього середовища, пошуково-рятувальних робіт), оскільки вони придатні для розв’язання великої кількості задач, а зменшення їх вартості, різноманітні технічні рішення та розширення функціоналу керування робить доцільним їх використання. Використання БПЛА виявилось особливо ефективним у динамічно-невизначених середовищах з важкодоступними зонами, проте у таких випадках вони зазвичай потребують оснащення спеціальними датчиками для полегшення виконання завдання. Керування спільною роботою дронів є складною науковою задачею, яка потребує об’єднання результатів з різних галузей досліджень [1].

У деяких випадках використання одного БПЛА може стати обмежуючим фактором, наприклад, для пошуково-рятувальних чи розвідувальних операцій, якщо є можливість швидко розгорнути декілька дронів, ймовірність успішного

виконання поставленої задачі може значно зрости. Використання роїв БПЛА, груп невеликих безпілотних літальних апаратів, які можуть рухатися та діяти у групі з обмеженим втручанням людини, представляє інтерес оскільки завдання, яке раніше було покладено на один дрон, можна розділити та розпаралелити.

Використання груп БПЛА забезпечує низку переваг: 1) загальна вартість придбання та обслуговування кількох невеликих комерційних БПЛА нижча, ніж загальна вартість одного великого; 2) масштабованість, яка є ключовою особливістю роїв БПЛА, відсутня в задачах одиничних БПЛА; 3) підвищена відмовостійкість, оскільки несправність одного дрону має обмежений вплив на рій; 4) більш швидкі операції завдяки розподіленню роботи. Але організація взаємодії груп БПЛА потребує створення відповідних теоретичних моделей їх спілкування та взаємодії та програмної реалізації алгоритмів, які забезпечують прийняття рішень окремими БПЛА на основі наявної інформації та з урахуванням спільної цілі.

Особливої важливості ця проблематика (створення команд дронів під конкретне завдання) набуває в задачах критичного застосування роїв БПЛА при моніторингу аварійних станів промислових об'єктів та в рятувальних операціях над територією певної площі, захисту критичних інфраструктур, виконання бойових операцій.

Існують переваги використання централізованого управління всією групою пристроїв при невеликій кількості БПЛА. Але при збільшенні кількості БПЛА у рої зростає навантаження на канал зв'язку між командним центром (КЦ) та кожним БПЛА. Наприклад, у Китаї в червні 2017 р. заявили про створення рою з 119 БПЛА, який функціонує на основі штучного інтелекту. У квітні 2018 р. китайський виробник БПЛА «Ehang Egret» запустив зграю з 1374 дронів (4 ряди на 1200 м), які керувалися з одного комп'ютера (наземної станції), але через перешкоди дрони порушили побудову, а декілька з них впали на землю.

Тому подальшим розвитком є створення методів організації мережі (команди або рою) дронів з наборами параметрів, що гарантують виконання поставленого завдання, а також побудова ієрархії підгруп рою (розподіл рою на рої меншого розміру) за функціональними ознаками або задачами, що виконуються.

В таких умовах актуальним є вирішення проблеми автономної поведінки БПЛА з дотриманням вимог щодо конфіденційності інформації про їх місцезнаходження. Швидкий розвиток конкурентоспроможних рішень на ринку комп'ютерних компонентів для БПЛА зумовив відсутність актуальних досліджень в напрямку співвідношення зростаючого їх енергоспоживання та часу працездатності батарей для живлення безпілотних апаратів. Розмір завдань рою БПЛА визначається також обмеженими обчислювальними ресурсами на борту.

Слід враховувати, що комп'ютерні компоненти об'єктів рою, що використовуються у якості хостів комп'ютерної мережі, а також модулі БПЛА можуть надходити від різних виробників, що призводить до гетерогенності елементів рою і потребує вирішення питання інтероперабельності. Зростаюча неоднорідність призводить до значного підвищення складності управління пристроями та сервісами у складі рою.

Розподіл завдань і застосування машинного навчання є досить важливим для рою БПЛА і забезпечує кращу пристосованість рою до різних середовищ та виконання різних завдань.

Зараз для організації взаємодії груп БПЛА застосовую елементи загальної теорії групового керування роботами в умовах динамічних, недетермінованих

середовищ [2]. Існують різні методи моделювання рою дронів для того, щоб визначити сильні та слабкі сторони його поведінки за різних умов. Найбільш часто використовують *метод рою часток* (МРЧ) - алгоритм на основі ШІ, який імітує процес скупчення птахів. Це метод числової оптимізації, для використання якого не потрібно знати точного градієнта оптимізованої функції. МРЧ, запропонований Кеннеді, Еберхартом та Ші, спочатку призначався для імітації соціальної поведінки [3], але для роїв дронів застосовують його спрощені модифікації.

Основу поведінки членів рою включає в себе функції, забезпечені модулем управління (автопілот). Тоді об'єкти рою можуть менше покладатися на прямий зв'язок один з одним. Рій може поліпшити продуктивність з меншим обміном інформацією і високою масштабованістю. БПЛА мережі рою можуть мати додаткові функції, включаючи, задачі декомпозиції, розподіл завдань, адаптивне навчання тощо. БПЛА за допомогою цих функцій в апаратній частині можуть спростити структуру алгоритму руху для створення більш складної фізичної конструкції дронів. БПЛА можуть також виконувати аналогічні функції з ретельно сконструйованими кооперативними алгоритмами. Реалізація таких функцій в апаратному або програмному забезпеченні залежить від фізичної конструкції БПЛА, контролерів та датчиків, найкращого використання їх комп'ютерних компонентів.

Сучасний етап розвитку інформаційних технологій (ІТ) характеризується впровадженням систем, що базуються на знаннях (СБЗ). Це новий якісний етап в еволюції систем обробки інформації, який забезпечує обробку даних, що мають складну структуру, використовуючи потенціал практичного досвіду людей. Інший важливий аспект сучасних ІТ – це орієнтація на відкритий світ та широке застосування інформаційних ресурсів Web. Агентні технології (АТ) дозволяють розв'язувати слабо формалізовані задачі відповідно до змін в оточуючому середовищі [4]. Через складність задач досить часто виникає потреба не в одному агенті, а в мультиагентних системах (МАС) – групі агентів, що працюють разом для спільних цілей. Використання інтелектуальних ІТ спільно з агентно-орієнтованим програмуванням (АОП) дозволяє підтримувати роботу складних систем у відкритому середовищі, координувати дії їх компонентів та прогнозувати їх поведінку [5].

АОП набуло останнім часом великої популярності. АОП сприяє підвищенню ефективності праці та дозволяє користувачам доручити ІС виконання досить складних завдань. Інтелектуальні програмні агенти (ІПА) – це новий клас абстракції програмного забезпечення, що дозволяє створювати розширювані, масштабовані, інтероперабельні системи. Програмному агенту делегуються певні функції людини, часом досить складні, для надання користувачеві потрібних йому послуг. Моделювання поведінки окремих ІПА і МАС у цілому потребує глибокого теоретичного обґрунтування. Необхідні інтелектуальні, інтерактивні та автономні програмні системи, які здатні до співпраці з користувачем для вирішення його задач. Такі складні задачі не можуть бути вирішені засобами одного наукового напрямку та потребують використання міждисциплінарного підходу, який містить технології з інформатики, штучного інтелекту, когнітивної науки та математики. Розподілене керування даними та обчислювальними процесами в умовах глобальних та корпоративних мереж спонукало до створення нової концепції середовища функціонування ІЗ як середовища взаємодії МАС, кооперації і конкуренції інтелектуальних ІПА.

Постановка проблеми. Пропонується адаптувати методи моделювання децентралізованих МАС та засоби обміну інформацією між інтелектуальними програмними агентами (ПА), що входять до складу цих систем, до специфіки та проблем керування поведінкою рою дронів. Щоб інтегрувати гетерогенні пристрої та однозначно описувати їх властивості, доцільно розробити онтологічну модель БПЛА, яка створює термінологічне середовище для ефективного обміну інформацією. Крім того, для більш ефективного функціонування такої МАС доцільно розробити методи ієрархічного поділу рою на рої меншого розміру та визначити ознаки, за якими доцільно групувати ці пристрої.

Основний матеріал. Теоретичні розробки з МАС надають підґрунтя для підтримки розумового інтелекту групи дронів. Важливо розуміти, що ефективність вдало організованої МАС більша за сумарну ефективність ПА, що входять до її складу, але невдалий вибір архітектури або засобів взаємодії між агентами можуть критично зменшити її – іноді навіть до ефективності одного ПА.

ПА можна розглядати як застосунки, для яких здатність до комунікації з іншими застосунками і до пошуку знань має першочергову важливість. Компоненти ПА можна поділити на такі: подання, комунікації і ті, що не відносяться безпосередньо до розподіленої взаємодії.

Для рішення проблеми комунікацій між інтелектуальними ПА необхідна спільна мова, тобто спільні синтаксис, семантика й прагматика. Зараз не існує загальноприйнятої мови подання інформації і запитів. Такі мови, як KIF і SQL, досить поширені, але цього недостатньо, щоб прийняти їх як стандарт.

Агенти мають взаємодіяти на рівні знань і тому не можуть задовільно обслуговуватися звичайними мовами та протоколами, що призначені для розподілених обчислень і звичайно фокусуються на процесах, а не на програмах або коаліціях ПА. Мова комунікації агентів має бути досить потужною для підтримки взаємодії між програмами високого рівня. Мова комунікації – не тільки протокол.

Протокол взаємодії – це високорівнева стратегія, що здійснюється агентом і взаємодіє з іншими агентами. Такий протокол може знаходитися в діапазоні від схеми переговорів і протоколів теорії ігор до простих протоколів типу "*щораз, коли я щось не знаю, я знаходжу кого-небудь, хто знає, і я запитую його*". Вибір транспортного протоколу залежить від специфіки його застосування.

Протокол, що використовується в контексті мови комунікацій, може мати одне з таких значень:

- транспортний протокол (HTTP, SMTP, FTP);
- високорівнева структура взаємодій (переговори, протоколи теорії ігор, планування);
- обмеження (constraints) на можливі зміни в комунікаційних примітивах.

Мова комунікацій може використовувати протоколи першого типу як транспортний механізм, другого – як спосіб їх використання, і третього – як частину визначення.

Можна сформулювати такі вимоги до мови взаємодії агентів:

- *форма*: декларативна, синтаксично проста, зрозуміла;
- *зміст*: відрізняє мови опису комунікаційних актів від мов передачі вмісту повідомлень;
- *семантика*: подає бажані властивості;
- *реалізація*: проста, сумісна з існуючим ПЗ;
- *робота в мережі*: враховує основні аспекти сучасних ІТ і має незалежний транспортний механізм ;

- *середовище*: підтримує гетерогенність і динамізм;
- *надійність*: забезпечує надійність і безпеку взаємодії агентів.

Важливим елементом взаємодії між ПА на рівні знань є наявність спільної онтології [6], яка забезпечує однозначне розуміння понять предметної області та сфери їх використання у переговорах.

Дрони групуються як за призначенням (та наявними компетенціями), та і за низькорівневим підпорядкуванням у територіально близькі групи, що орієнтовані на виконання окремих підзадач.

При цьому персональні агенти пристроїв утворюють набір ієрархій. Наприклад, можуть групуватися дрони з однаковим функціоналом, одного виробника або з подібним значенням певного параметру (дальності польоту, ваги, наявності певних пристроїв або озброєння). Іншим критерієм побудови підгруп є спільна робота, коли до підгрупи включають, наприклад, один дрон із розвинутими засобами спостереження та кілька простіше укомплектованих БПЛА з більшим навантаженням.

Для більш ефективного управління доцільно підтримувати кілька рівнів ієрархії, на якому (за аналогією з еталонною моделлю відкритих систем) пристрої взаємодіють або з пристроями свого рівня ієрархії (членами своєї мікрогрупи), або з пристроями на один рівень вище – безпосередніми командирами чи координаторами, або з пристроями на один рівень нижче – безпосередніми підлеглими. Для вибору найбільш вдалої структури МАС доцільно застосовувати методи аналізу ієрархій (наприклад, метод Сааті та його розширення).

Щоб визначити функціонал окремих пристроїв (як самих дронів, так і їх обладнання та озброєння), доцільно розробити онтологічну модель їх компетенцій, де різні потенційно припустимі елементи функціоналу характеризуються як якісно (наявність, тип сприйняття зображення тощо), так і кількісно (швидкість, вага, дальність тощо). Крім того, ця онтологія має відображати зміст взаємодії між окремими типами дронів, операторами та зовнішнім обладнанням. Апарат атомарних компетенцій, що створений для оцінювання результатів навчання та моделювання подальшої освіти, може бути легко адоптований для опису наявних технічних характеристик незалежно працюючих дронів та напрямків їх вдосконалення. При цьому виникає можливість оцінювання та співставлення різних наборів пристроїв та їх відповідності конкретним завданням та параметрам середовища. При цьому окремі дрони розглядаються як потенційні працівники, їх обладнання – як наявні компетенції, можливості доукомплектування іншим обладнанням – як здатність до навчання додатковим компетенціям, а та задача, яку має виконати група дронів – як набір вакансій для виконання проекту.

Висновки. Доцільно розглянути можливість накопичення досвіду як мультиагентної системи в цілому, так і різних підмножин її агентів для створення навчальних вибірок: застосування методів машинного навчання дозволить побудувати евристичні найбільш вдалі поведінки та формально відобразити їх в онтологічній моделі. Крім того, актуальним напрямком інтелектуалізації поведінки рою дронів є використання елементів Internet of Things та стандартів подання знань Semantic Web.

Список використаних джерел:

1. Mueller, M. W., Lee, S. J., & D'Andrea, R. Design and control of drones. *Annual Review of Control, Robotics, and Autonomous Systems*, 5, 2022, P.161-177.

2. Кравчук, С. О., Кравчук, І. М. Підходи та стратегії щодо колективного керування роєм дронів. Збірник матеріалів Міжнародної науково-технічної конференції «ПЕРСПЕКТИВИ ТЕЛЕКОМУНІКАЦІЙ», 2019. URL: <http://conferenc.its.kpi.ua/proc/article/view/167832/167528>
3. Shi, Y. Particle swarm optimization. *IEEE connections*, 2004, 2(1), P.8-13.
4. Плескач В.Л., Рогушина Ю.В., Агентні технології. К.: Київ. нац. торг.-екон. ун-т, 2005. – 338 с.
5. Shoham Y. Agent-oriented programming. *Artificial Intelligence*. N 60. 1993. – P.51-92.
6. Рогушина Ю.В. Використання онтологічних знань інтелектуальними агентами. *Проблеми програмування*, 2017, №2. – С.82-95.

УДК 004.8

ОСОБЛИВОСТІ ПОБУДОВИ СИСТЕМ НАДАННЯ ВІДПОВІДЕЙ

Пироженко М.Ю., аспірант

e-mail: mykhailo.pyrozhenko@nure.ua

Харківський національний університет радіоелектроніки

Актуальність та постановка проблеми.

На сьогоднішній день значна частина неструктурованою інформації представлена у вигляді текстів на природних мовах. Письмовий текст, з яким ми щодня взаємодіємо – книги, новини, інформаційні сторінки веб-сайтів, листи, наукові статті та художні твори – є продуктом глибоко соціального явища, в основі якого стоять люди. За деякими винятками, весь текст, з яким ми щодня працюємо, написаний людьми. Інформація в цих текстах містить багато корисних даних та знань, необхідних для вирішення практичних завдань.

Процес обробки текстів природних мов має широкий спектр сфер для застосування та вивчення, в яких проводяться величезні дослідження. Прикладами практичних завдань є видалення спаму, виправлення помилок, машинний переклад, розпізнавання мовлення, побудова діалогу (чат-боти), інформаційний пошук та включає, у тому числі, систем надання відповідей. Для вирішення завдань вказаного спектру необхідно застосовувати системи глибокого аналізу текстів та штучного інтелекту.

Метою цього дослідження становитиме подальший розвиток методологічної бази розробки програмного забезпечення з обробки текстів природної мови, та систем здатних відповідати на питання користувачів на базі знань отриманих з колекцій текстових даних.

Основні матеріали дослідження. Системи надання відповідей – це класичні системи з обробки текстів природної мови[1]. На відміну від традиційних систем інформаційного пошуку, при роботі з системою користувач ставить певне запитання, замість запиту з набору ключових слів, та отримує коротку сфокусовану відповідь, замість переліку документів. Відмінність також присутня і на стороні системи у тому, як ці дані та знання формуються.

Системи використовують різні типи джерел, таких як текстові колекції, так і структуровані бази знань[2]. Тобто, це не лише текстові документи, як у випадку традиційного інформаційного пошуку, але це також включає дані, що представлені у якості баз знань, інформація у яких міститься у структурованому вигляді та може бути надана користувачу.

При розробці систем надання відповідей використовуються як дуже прості методи, так і методи, які припускають глибоку лінгвістичну обробку тексту. Надійні методи аналізу тексту дають нам спосіб зрозуміти або синтезувати великі обсяги тексту, не читаючи його. Комерційні успіхи таких продуктів, як Alphabet Google, IBM Watson та Apple Siri, значною мірою підтверджують залежність успіху систем від використання надійних обчислювальних моделей.

Для систем надання відповідей критичним є завдання інформаційного пошуку, видалення сутностей та відносин, ідентифікація кандидатів на відповідь. Вирішення цих завдань займає більшість часу розробника.

Для того, щоб ефективно здійснювати пошук за великою колекцією документів, потрібно індексувати цю колекцію. Підсистема інформаційного пошуку намагається знайти невеликі обсяги тексту у великих неструктурованих текстових колекціях даних, та повертає документи, в яких ймовірно може

міститися відповідь, а потім, більш глибокими та продуктивними методами, система шукає фрагменти документів. Система може містити як локальні колекції текстових даних, так і звертатися до глобальної мережі Інтернет за пошуком відповіді.

Після вилучення фрагментів неструктурованої інформації у вигляді текстів, системі потрібно також витягти з неї певну частину інформації в структурованому вигляді. Тип питання дає можливість зрозуміти тип сутності, які необхідно вилучити для подальшого аналізу. Використовуючи регулярні вирази, словники, класифікатори, статистичні методи та методи штучного інтелекту, система знаходить потрібні за типом сутності.

Після того, як система розбила документи на уривки та знайшла сутності, що можуть бути відповіддю на питання, необхідно зрозуміти, які з них найкраще підходять як кандидат на відповідь та ранжувати їх за цим критерієм. Питання та уривок вводяться в модель вилучення відповіді, а модель виводить відповіді зміщені за балом.

Система ранжує отриманих кандидатів на основі балів і представляє відповідь за найвищими балами користувачеві як остаточну відповідь. Надання відповідей відбувається шляхом витягування відповіді з окремих уривків, але відповідь також може бути генерована.

Впровадження систем надання відповідей призводить до збільшення швидкості пошуку необхідної інформації[3]. Таким чином, користувач, в дійсності, працює з інтелектуальною системою, які не просто повертають документи, пов'язані з запитанням, але надають релевантну інформацію з документів і пропонують остаточну відповідь.

Висновок. Розглянуто особливості побудови систем надання відповідей. Системи надання відповідей дають остаточну відповідь на питання безпосередньо, замість повернення лише списку релевантних фрагментів або гіперпосилань, та, таким чином, забезпечують кращу зручність та ефективність, ніж звичайні пошукові системи.

Розглянуті завдання відносяться до дисциплін пошуку інформації, вилучення інформації та обробки природної мови. Зважаючи на складність систем текстової аналітики, проектування систем обробки текстів природної мови є одним із важливих завдань в побудові інформаційних систем. Вирішення цих проблем дозволить економити зусилля користувачів, оскільки дозволить приймати більш швидкі рішення.

Оскільки кількість відцифрованих текстів постійно зростає можна прогнозувати збільшення потреби у системах текстової аналітики. Системи надання відповідей дозволяють зменшити час на пошук потрібної інформації та зручно інтегруються із сучасними пристроями.

Список використаних джерел:

1. Jurafsky, D. (2020). Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition.
2. Diefenbach, D., Lopez, V., Singh, K., & Maret, P. (2018). Core techniques of question answering systems over knowledge bases: a survey. Knowledge and Information systems, 55(3), 529-569.
3. Voorhees, E. M. (2008). Evaluating question answering system performance. Advances in Open Domain Question Answering, 409-430.

УДК 004.896

МУЛЬТИАГЕНТНА СИСТЕМА МЕРЕЖІ ДРОНІВ ДЛЯ ЗАХИСТУ КРИТИЧНОЇ ІНФРАСТРУКТУРИ З ОНТОЛОГІЧНИМ ПОДАННЯМ ЗНАНЬ

Гладун А.Я., к.т.н.,

e-mail: glanat@yahoo.com

Хала К.О.

e-mail: cecerongreat@ukr.net

*Міжнародний науково-навчальний центр інформаційних технологій та систем
НАНУ та МОНУ, Київ, пр. Глушкова, 40 Україна*

Актуальність. Сьогодні БПЛА, або дрони, привертають велику увагу різних сфер застосування від військових до цивільних, оскільки вони підходять для великої кількості застосунків і мають чимало переваг. Наприклад, зниження собівартості дронів робить їх цікавими у кількох цивільних сферах: наприклад, для точного рільництва у сільському господарстві [1,2], для спостереження або моніторингу навколишнього середовища. [3], або у пошуково-рятувальних програмах [4]. Використання БПЛА виявилось особливо ефективним у динамічно-невизначених середовищах з важкодоступними зонами, проте у таких випадках вони зазвичай потребують оснащення спеціальними датчиками для полегшення виконання завдання.

Прогрес у таких технологічних галузях, як блокчейн (розподілена база даних), штучний інтелект і машинне навчання, дозволив розробити системи безпілотних літальних апаратів з оновленими та вдосконаленими системами, які забезпечують більшу безпеку, надійність та ефективність, щоб виконувати задачі БПЛА більш успішно.

У деяких випадках використання одного БПЛА може стати обмежувальним фактором, наприклад, для пошуково-рятувальних чи розвідувальних операцій, якщо є можливість швидко розгорнути декілька дронів, ймовірність успішного виконання поставленої задачі може значно зрости. Так використання невеликих БПЛА, які можуть рухатися та діяти у групі з обмеженим втручанням людини, представляє інтерес оскільки завдання, яке раніше було покладено на один дрон, можна розділити та розпаралелити. Як зазначено в [5], використання груп БПЛА забезпечує низку переваг: 1) загальна вартість придбання та обслуговування кількох невеликих комерційних БПЛА нижча, ніж загальна вартість одного великого; 2) масштабованість, яка є ключовою особливістю груп БПЛА, відсутня в задачах одиничних БПЛА; 3) підвищена відмовостійкість, оскільки несправність одного дрону має обмежений вплив на групу; 4) швидші операції завдяки розподіленню роботи.

Тому підвищення ефективності координації роботи кількох БПЛА має велике значення для успішного кооперативного виконання ними завдань з максимізацією результатів та мінімізацією витрат. Однак, порівняно з прийняттям рішень для одного БПЛА, проблема координації кількох БПЛА охоплює низку проблем, таких як інтелектуальне прийняття рішень [6], розподілена співпраця [7] та керування формуванням строю [8].

Постановка проблеми. Для розгортання кількох БПЛА, їх координування, керування та обміном даними, пропонується модель мультиагентної системи (МАС), яка базується на онтологічному поданні знань та дозволяє ефективно приймати рішення в різних ситуаціях при виконанні поставлених завдань. При цьому МАС виконує тільки частину бойових завдань, а саме координація сумісної

роботи з розподілом задачі на окремі підзадачі для кожного БПЛА, обмін повідомленнями між БПЛА при сумісній роботі та зміна ролей БПЛА у випадку втрати БПЛА або ж його відмові (ролі при цьому динамічно переназначаються іншим БПЛА). Для ефективної роботи мережі БПЛА необхідно моделювання роботи окремих БПЛА, як агентів і формування планів роботи для вирішення завдання з урахування розподілених ролей агентів у мережі БПЛА (лідер, координатор, виконавець, тощо). Для планування роботи мережі БПЛА, як МАС, необхідно п'ять компонентів: знання про зовнішній світ, інформацію про обмежену територію для виконання завдання, знання про розподілення ролей (лідер, координатор, виконавець, тощо), набір виконуваних завдань (патрулювання, перегруповування, відступ, тощо) та розподіл (або перерозподіл) завдань.

Ключовим моментом також є вирішення задачі декомпозиції задачі на підзадачі та організацію взаємодії між програмними агентами дронів та центрами управління дронами, які знаходять на землі. Важливим завданням є інформаційна безпека, передавання інформації, ідентифікація між дронами та підвищення точності обчислень при виконанні поставлених завдань.

Основні матеріали дослідження. В основі мультиагентного підходу лежить поняття мобільного програмного агента, який реалізовано і функціонує як самостійна спеціалізована комп'ютерна програма або елемент штучного інтелекту. Агенти можуть самостійно визначати, як найкраще досягти своїх цілей і виконувати свої завдання, та володіють властивостями, як автономія, активність, проактивність і соціальна поведінка [9].

Агент має отримувати, зберігати і обробляти інформацію щодо поточного стану предметної області. Знання агента про навколишнє середовище, інших агентів і самого себе специфіковано в формі онтології. Онтологія є ефективним підходом, який можна використовувати для вирішення гетерогенних проблем, пов'язаних із узагальненням гетерогенних даних низького рівня до концепцій відносно високого рівня для цілей обміну даними, взаємодії системи та повторного використання програмного забезпечення. Якщо агенти будуть оперувати одними і тими ж поняттями, то це допоможе у вирішенні багатьох проблем, як то методів комунікації між агентами та способів адаптації до нових умов.

Онтології розширюють можливості виконання семантичних аргументів [10], які забезпечують такі функціональні можливості, як перевірка узгодженості, виконуваність концепції, класифікація та реалізація. Онтології також дають змогу обмінюватися загальним розумінням структури інформації між людьми та програмними агентами та повторним використанням основних знань [11]. Інтеграція таких семантичних технологій у МАС покращує особливості представлення знань і можливості міркувань застосунків, розроблених відповідно до цих парадигм [12]. Застосування онтологій у МАС дає можливості для створення логічних правил, які можуть бути застосованими для семантичного міркування та виведення нових знань. Проте використання онтологій ускладнює інваріантність МАС щодо предметної області, і тому онтологія (принаймні предметної області) буде змінюватися.

Розробка поведінки та взаємодії агента в МАС в основному стосується моделей обміну даними агента, як між собою так і з наземним центром керування. Структура обміну даними агента зазвичай охоплює повідомлення (контент повідомлення) і обмінювання повідомленнями (наприклад, протокол KQML). Зміст повідомлення складається з двох частин: мови змісту (забезпечення

синтаксису або граматики вмісту) та онтології (складає семантику або лексику повідомлення) [13]. Модель обміну даними агента на основі онтології показано на рисунку 1.

Оскільки агенти спілкуються завдяки обміну повідомленнями, то для розроблення MAC зазвичай використовують стандарт Foundation for Intelligent Physical Agents (FIPA - Фонд інтелектуальних фізичних агентів - www.fipa.org). Семантична мова FIPA широко використовується як стандартна мова контенту [15]. У FIPA онтологія містить перелік понять, предикатів і дій, специфічних для області комунікації. У сервісах FIPA онтологічному агенту надається низка пов'язаних з онтологіями сервісів для вирішення проблеми застосування кількох онтологій [13].

Під час проектування архітектури MAC вводиться синтаксис і семантика для визначення загальної онтології верхнього рівня, яка представляє загальні концепти, що використовують в системі. Тоді як синтаксис і семантика для онтології домену (ПрО) та онтології конкретних агентів, буде описувати цілі та функції онтологій. [16, 17].

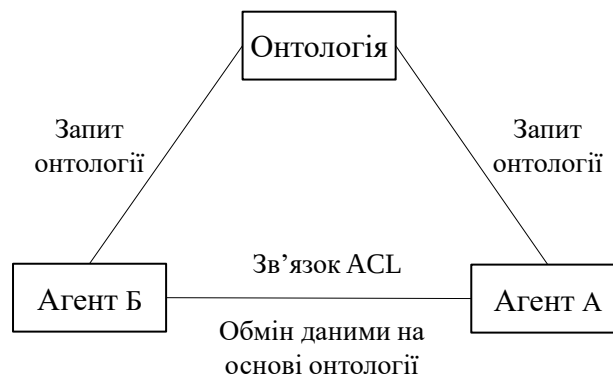


Рисунок 1. Модель обміну даними між агентами на основі онтології [14].

При розробці архітектури MAC, варто виділити такі базові компоненти:

- сервіс доступу для забезпечення доступ до атрибутів агентів;
- сервіс повідомлень, що відповідає за передачу повідомлень між агентами або додатковими системами;
- бібліотека агентів з інформацією про класифікацію агентів в MAC;
- взаємодія агентів, що відповідає за життєдіяльність агентів, забезпечує функції завантаження/запису агентів і їхніх властивостей, та відповідає за оптимізацію роботи агентів з ресурсами;
- онтологія - база знань, яка містить знання про світ (середовища функціонування), виконувані дії та знання для самопізнання агенту (оновлювані дані).

Отже, кожен БПЛА моделюється як агент, який має повинен рухатися, сприймати свій стан і середовище, слідувати плану, досягати мети, взаємодіяти та адаптуватися до поведінки. Для врахування складних взаємодій між агентами і охоплення необхідних гетерогенних властивостей, агент БПЛА моделюється як архітектура Belief-Desire-Intention (BDI - Переконавання-Прагнення-Наміри) [18].

Архітектура BDI для агента - це набір планів, що визначають засоби, за допомогою яких агент досягає кінцевої мети функціонування. План складається з

міток голови, тіла та хвоста робочого алгоритму агента. Мітки тіла плану - це послідовність дій, яка визначає цілі, яких агент має досягти, та умови, які агент має перевірити. Мітки голови та хвосту плану символізують *наміри*, де агент має наперед визначений перелік намірів. У кожний момент, коли агент вибирає план виконання, намір може бути активним чи неактивним. Коли агент вибирає деякий план виконання, він виконуватиметься лише тоді, коли всі наміри з його голови будуть активні. Після виконання плану агент переводить ці наміри в неактивні і робить активними всі хвостові наміри плану. Передбачається, що поточний набір активних намірів не порожній. Тіло плану та набір хвостових намірів можуть бути порожніми.

У цих термінах можна вважати, що агент складається з безлічі переконань B , планів P , ситуацій S , дій A і намірів I . Коли агент помічає зміни в навколишньому середовищі, він вважає, що відбулася подія E . Ці події є деякими ситуаціями зовнішнього середовища з S . Реєстрація події агентом полягає у зміні стану його міркувань, а саме вибору деякого переконання з B . Відповідно до нього і бажання (що визначається деяким планом з P) агент вибирає для виконання деяких намірів з I , які є послідовністю дій з A . Ці дії становлять план досягнення поставленої мети. Таким чином, запланована дія визначається вибраним планом з P . Потім ці дії виконуються, змінюючи поточну ситуацію у навколишньому середовищі.

На рисунку 2 наведено агента БПЛА реалізовано для руху, сприйняття, виконання плану, взаємодії та адаптації поведінки, чий контроль навичок базовано на онтології. Переконання пов'язані зі схильностями під час виконання завдання. Вони можуть бути оновлені відповідно до знань щодо навколишнього середовища і самопізнання агента. Агент БПЛА має перевірити кожне переконання відповідно до логічних правил із самопізнання та призначеного завдання.

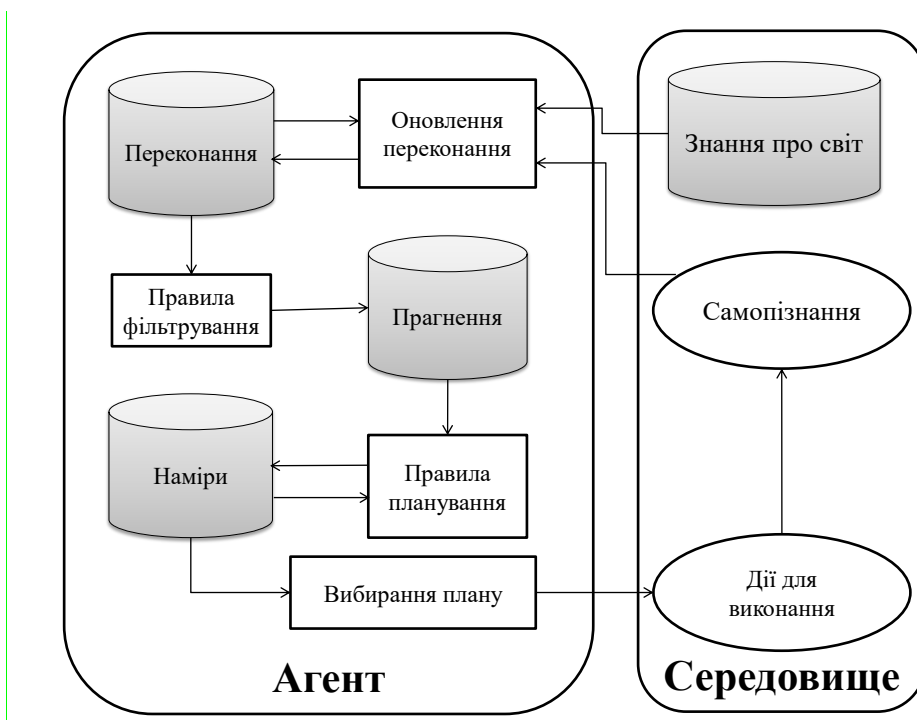


Рисунок 2. Модель BDI для агентів мережі БПЛА

Отже, агент БПЛА мотивований досягти якоїсь однієї мети, однак агент не може досягти мети без плану. Онтологія планів враховує *наміри* та *прагнення* для визначення дій, які будуть впливати на навколишнє середовище.

Запропонований метод передбачає створення ролі для визначення поведінки моделі агента, що створює ієрархічне координування. Модель передбачає часткове підключення до кількох найближчих агентів у обмежений час. Кожен агент БПЛА може взаємодіяти лише тоді, коли найближчий сусіді знаходяться у визначеному діапазоні. Взаємодіють агенти БПЛА для прийняти рішення щодо виконання місії потребує взаємодії не лише між агентами, а й з онтологією навколишнього середовища, що є ресурсовитратним процесом.

Ієрархічна модель охоплює ролі для керування неоднорідністю та зв'язністю групи відповідно до показників навичок, можливого часу польоту, обміну даними та прийняття рішень. До прикладу, *Лідер* повинен мати високий показник у прийнятті рішень та обміну даними, а також проміжне значення щодо часу польоту. Його роль полягає в прийнятті принципових рішень для завершення місії, тож агент *Лідера* повинен мати короткі шляхи для передавання та отримання повідомлень від інших агентів.

Актуальність онтології в МАС полягає у можливості вирішення проблеми інформаційного переповнення мережі [19]. Основні задачі, успішно вирішувані на основі онтологій, включають:

- подання знань для логічного виведення, яке повинне бути релевантним запиту (користувача чи агента);
- фільтрація і класифікація інформації;
- індексування зібраної інформації;
- організація загальної термінології, якою можуть користуватися для обміну даними програмні агенти і користувачі.

Висновок. Запропонована МАС базована на онтології спрямована на ефективну взаємодію мережі групи різнорідних БПЛА. Запропонований підхід формує ієрархію БПЛА за ролями, що мають відповідні маршрути пересування, визначений час польоту, навички обміну даними та здатні приймати рішення для досягнення спільної мети.

Онтологія розширює дані в базі знань за допомогою узгодженої структури відношень і чітко визначених термінів для формування логічних висновки з даних, позначених термінами з цієї онтології.

Необхідно вирішити завдання безпеки в МАС, зокрема наявність системи захисту від несанкціонованого доступу та “поганих кодів”. Це важливо з багатьох причин. Мобільні агенти, які приходять ззовні, можуть приховувати безліч небезпек для приймаючої сторони, оскільки вони виконуються в її адресному просторі. Для забезпечення безпеки перед передачею керування кожному агенту необхідно виконати процес авторизації агента, тобто перевірити, чи він зареєстрований і чи має відповідні повноваження (привілеї), щоб виконати ту чи іншу дію або звернутися до деяких ресурсів. Система безпеки повинна запобігати будь-яким несанкціонованим діям агента.

Список використаних джерел:

1. M. Bacco, E. Ferro, and A. Gotta, Radio propagation models for uavs: what is missing?" in Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2014, pp. 391-392.

2. F. M. Bacco, E. Ferro, and A. Gotta, UAVs in WSNs for agricultural applications: an analysis of the two-ray radio propagation model," in IEEE SENSORS 2014 Proceedings. IEEE, 2014, pp. 130-133.
3. G. Chmaj and H. Selvaraj, Distributed processing applications for UAV/drones: a survey," in Progress in Systems Engineering. Springer, 2015.
4. S. D'Auria, M. Luglio, C. Roseti, R. Strollo, and F. Zampognaro, "Real Time Transmission of Cultural Heritage 3D Survey in Case of Emergency," in 3rd International Conference on Information and Communication Technologies for Disaster Management, Vienna, Austria, December 2016.
5. I. Bekmezci, O. K. Sahingoz, and S. Temel, Flying ad-hoc networks (FANETs): A survey," Ad Hoc Networks, vol. 11, no. 3, pp. 1254-1270, 2013.
6. Han, Y.; Wang, H.; Zhang, Z.; Wang, W. Boundary-aware vehicle tracking upon uav. Electron. Lett. 2020, 56, 873–876.
7. Jiang, H.; Shi, D.; Xue, C.; Wang, Y.; Wang, G.; Zhang, Y. Multi-agent deep reinforcement learning with type-based hierarchical group communication. Appl. Intell. 2021, 51, 5793–5808.
8. Zhan, G.; Gong, Z.; Lv, Q.; Zhou, Z.; Wang, Z.; Yang, Z.; Zhou, D. Flight test of autonomous formation management for multiple fixed-wing uavs based on missile parallel method. Drones 2022, 6, 99.
9. Гладун А.Я. Интеллектуальные агенты в корпоративных системах// Корпоративные системы, №3, 2006. С. 48-64. https://comizdat.com/index_.php?in=ksks_allarticles&CATEGORY=38
10. E. Sirin, B. Parsia, B.C. Grau, A. Kalyanpur and Y. Katz, Pel-let: A practical OWL-DL reasoner, Web Semantics: Science, Services and Agents on the World Wide Web 5(2) (2007), 51– 53. doi:10.1016/j.websem.2007.03.004
11. T.R. Gruber, A translation approach to portable ontology specifications, Knowledge Acquisition 5(2) (1993), 199–220. doi:10.1006/knac.1993.1008.
12. M. Hadzic, P. Wongthongtham, T. Dillon and E. Chang, Ontology-Based Multi-Agent Systems, Studies in Computational Intelligence, Springer, 2009. doi:10.1007/978-3-642-01904-3.
13. Catterson, V.M.; Davidson, E.M.; McArthur, S.D.J. Issues in Integrating Existing Multi-Agent Systems for Power Engineering Applications. In Proceedings of the 13th International Conference on, Intelligent Systems Application to Power Systems, Arlington, VA, USA, 6–10 November 2005; p. 6.
14. FIPA. FIPA Ontology Service Specification; FIPA: Geneva, Switzerland, 2000.
15. Borgida, A. Description logics in data management. IEEE Trans. Knowl. Data Eng. 1995, 7, 671–682. [CrossRef]
16. Wooldridge, M.; Jennings, N.R.; Kinny, D. The Gaia Methodology for Agent-Oriented Analysis and Design. Auton. Agents Multi Agent Syst. 2000, 3, 285–312.
17. Uslar, M.; Specht, M.; Rohjans, S.; Trefke, J.; González, J.M. The Common Information Model. CIM: IEC 61968/61970 and 62325—A Practical Introduction to the CIM; Springer: Berlin, Germany, 2012.
18. Гриценко, В.И., Гладун, А.Я., Журавлев, Ю.Д., Несен, М.В. Модель мультиагентной системы для е-бизнеса и технология ее программной реализации// Проблемы програмування. — 2004. — N 2,3. — С. 510-519.
19. Dominguez, M.H.; Hernández-Vega, J.-I.; Palomares-Gorham, D.-G.; Hernández-Santos, C.; Cuevas, J.S. A BDI Agent System for the Collaboration of the Unmanned Aerial Vehicle. Res. Comput. Sci. 2016, 121, 113–124.

УДК 004.9

ОГЛЯД СУЧАСНИХ ГЕОІНФОРМАЦІЙНИХ СИСТЕМ

Мелешко О.Д., студент

e-mail: ushio159@gmail.com

Науковий керівник: Зінов'єва О.Г., ст. викладач

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Сучасні інформаційні технології змінюють образ життя людини. Для організованого зберігання, пошуку потрібної інформації, її обробки та аналізу потрібні сучасні, що базуються на комп'ютерних технологіях, засоби. З кожним роком інформаційні потреби людини торкаються нових сфер його діяльності. Практично у всіх сучасних галузях знань накопичено багатий досвід використання інформації, що отримується з численних джерел

За останній час дуже поширилося застосування карт у різних галузях науки. Це пов'язано з розвитком та застосуванням геоінформаційних систем, які мають принципово новий підхід до роботи з просторовими даними.

Геоінформаційна система – це цифрове середовище для збору, керування та аналізу даних з урахуванням їх територіального розподілу. На сьогоднішній день ГІС завойовують поступово всі галузі народного господарства нашої країни. Наприклад, в сільському господарстві моніторинг посіву та збирання врожаю сільськогосподарських культур здійснюється на основі ГІС. Роботи, які здійснюються із застосуванням ГІС-технологій, мають високу точність і достовірність. Сьогодні ГІС - одна з найсучасніших перспективних технологій, яку багато організацій впроваджують у свою виробничу діяльність як інструмент, що удосконалює бізнес-процеси підприємств

В даній роботі ми проведемо огляд та аналіз сучасних геоінформаційних систем.

Основні матеріали дослідження. Географічні інформаційні системи – це одна з інформаційних технологій, які інтенсивно розвиваються в сучасному світі. Суть цієї технології складається в поєднанні електронних карт з системами управління базами даних. Сучасні програмні засоби геоінформаційних систем дозволяють вирішувати широкий клас задач просторового аналізу (швидкий пошук об'єктів, створення тематичних карт, обчислення відстаней, площі, пошук найкоротший шляхів, оверлейний аналіз та ін.). Геоінформаційні системи знаходять широке застосування в тих предметних областях, де основною складовою є просторова інформація.

На даний час геоінформаційні системи – одна з частин ринку високих комп'ютерних технологій, в якій працює велика кількість відомих фірм, таких як Google, NASA, Microsoft, Intergraph, ESRI, Autodesk та інші.

Географічна інформаційна система представляє собою сукупність просторової та атрибутивної інформації, програмного забезпечення, технічного забезпечення, засобів аналізу інформації та її візуального представлення.

Географічні інформаційні системи з'явилися в 1960-х роках в США, Швеції та Канаді як засіб відображення в пам'яті комп'ютера об'єктів, розташованих на поверхні Землі. Більш широке розповсюдження геоінформаційних систем почалося в той час, коли стали активно розвиватися корпоративні та розподілені бази геоданих на основі інтеграції ГІС та СУБД, а також розробки мережових додатків.

В наш час на ринку ГІС існує не менш 100 комерційних систем. Найпоширенішою комерційною ГІС-програмою є ArcGIS, продукт від ESRI, який включає кілька програм для аналізу ГІС. Іншими прикладами поширених комерційних ГІС-програм є MapInfo, GeoMedia, GeoBas, DP/Map і Global Mapper.

Інший тип – програмне забезпечення ГІС з відкритим вихідним кодом – це безкоштовне програмне забезпечення, до якого користувач може отримати доступ, використовувати або змінювати його. Це означає, що програму можна завантажити безкоштовно, а потім користувачі можуть використовувати її так, як їм заманеться. Одним із найстаріших ГІС-програм із відкритим кодом є GRASS (Geographical Resources Analysis Support System), яка була розроблена з 1982 по 1995 рік і використовується досі. Зараз на ринку є декілька ГІС-програмних продуктів з відкритим кодом. Прикладами таких програм є QGIS, OpenJump і uDig.

Розглянемо і порівняємо можливість і та функціональність найбільш відомих геоінформаційних систем – ArcGIS, MapInfo та QGIS.

ArcGIS (розробник фірма ESRI, США) - провідний програмний продукт ESRI, який був розроблений у 1999 році. Це одна з геоінформаційних систем, яка найбільш часто використовується як великими компаніями, так і в освіті. Цей програмний продукт є високорівневою ГІС-системою з повним набором засобів геообробки, включаючи збір даних (растровий та векторний формат), їх інтеграцію, зберігання, автоматичну обробку, редагування, створення та підтримку топології, просторовий аналіз, роботу з регулярною та нерегулярною моделями, зв'язок із SQL DBMS, візуалізацію та створення твердих копій будь-якої картографічної інформації.

MapInfo Professional - це настільний програмний продукт, наданий компанією Pitney Bowes. Це географічна інформаційна система, призначена для збирання, зберігання, відображення, редагування та аналізу просторових даних. Перша версія ГІС MapInfo Professional була розроблена в 1987 році компанією MapInfo Corp. і швидко стала однією з найпопулярніших ГІС у світі. Зараз MapInfo Professional використовується в 130 країнах світу, перекладена 20 мовами і встановлена в десятках тисяч організацій. У MapInfo є безліч способів створення тематичних карт: картограми, кругові та стовпчасті гістограми, градуировані символи, щільність точок, окремі значення, безперервна поверхня, карта-призму, карта ізоліній тощо.

Quantum GIS (QGIS) — це ГІС з відкритим кодом, заснована Гарі Шерманом у 2002 році. Сьогодні QGIS використовується тисячами людей по всьому світу і дедалі більше стає життєздатною альтернативою комерційному настільному програмному забезпеченню ГІС. QGIS може бути завантажена безкоштовно та працює на більшості операційних систем.

QGIS пропонує широкий спектр функцій і можливостей, схожих на функціональність ArcGIS. При редагуванні карти QGIS надає користувачеві декілька функцій. Це включає, наприклад, оцифрування вектора функції, можливість додавати та видаляти точки, лінії та багатокутники з існуючих даних та різноманітні вибір символів.

Всі основні функції геоінформаційних систем виконуються всіма цими програмами, але при виконанні деяких функцій можливі відмінності.

Наприклад прив'язка даних в QGIS потребує наявності підключених модулів і плагінів. Спосіб прив'язки заснований на створенні контрольних точок з отриманими даними. Особливостями прив'язки даних у QGIS – це можливість перевірити точність прив'язки.

В MapInfo додаткових додаткових установок не потрібно, спосіб прив'язки простіший, система координат та проекція вказується один раз на початку роботи.

В ArcGIS географічна прив'язка здійснюється в декілька етапів, які залежать від типу даних.

Однією з важливих функцій геоінформаційних систем є створення тематичних карт. Тематичні карти - це карти, зміст яких розкриває певну тему, тобто відображають окремі елементи території та пов'язані з ними характеристики природних або суспільних явищ, наприклад, щільність населення в певній області. В QGIS при створенні карт можна у властивостях шару конструювати, створювати лінії вбудованими інструментами, що займає немало часу. Набори вбудованої бібліотеки обмежені.

В MapInfo є можливість встановлення додаткових утіліт, наборів шрифтів і типів ліній. Копіювання і вставка векторних об'єктів в обох цих програмах не відрізняється.

В ArcGIS міститься багато інструментів і операцій, які можуть застосовуватися в моделюванні і при створенні тематичних карт.

В таблиці 1 наведені критерії порівняння ГІС.

Таблиця 1 – Порівняльна характеристика ГІС

Критерій оцінювання	ArcGIS	MapInfo	QGIS
Розробник	ESRL, Inc (Калифорнія, США)	Pitney Bowes Software Inc. (США)	Міжнародна група розробників QGIS
Мова програмування	Visual C; Python; C++	Python	C++; Python
Робота в Windows	+	+	+
Робота в Linux	-	-	+
«Відкритість» архітектури	-	+	+
Повнофункціональність ГІС	+	+	+
Наявність розвиненої системи прав санкціонованого доступу до картографічних та семантичних даних	-	-	+
Суворі організація структури цифрових картографічних шарів	+	+	+
Зберігання картографічної та семантичної інформації на SQL-сервері	+	+	+
Зручність роботи з інтерфейсом прив'язки растру	+	-	+
Зручність роботи з шарами	+	-	+

Висновки. Дослідження показує, що функціональні можливості трьох програм ArcGIS, QGIS та MapInfo більш-менш однакові. Кожна з геоінформаційних систем має свої особливості.

Геоінформаційна система ArcGIS є багатофункціональною системою, має багатоплатформеність, мобільні додатки, але ж недоліком її є насамперед висока вартість та складність в експлуатації.

До переваг MapInfo Professional можна віднести можливість вирішувати основні завдання просторового аналізу, простоту використання.

Геоінформаційна система QGIS виграє серед розглянутих систем перш за все тим, що вона є системою з відкритим кодом, працює в Linux, Unix, Mac OSX та Windows,.

Можна зробити такі висновки: використання геоінформаційних систем дозволяють найефективніше використовувати людські ресурси за рахунок автоматизації процесу; кожна з геоінформаційних систем має свої особливості та нюанси, в залежності від задачі, яка стоїть перед спеціалістом.

Список використаних джерел:

1. Світличний О.О., Плотницький С.В. Основи геоінформатики. Суми: «Університетська книга», 2006. 295 с..
2. Геоінформаційні системи. Вступний курс : навч. посіб. / А. Д. Тевяшев, В. П. Ткаченко, М. І. Губа та ін. – Х. : ХНУРЕ, 2017. – 392 с.
3. Лабенко Д.П., Тімонін В.О. Геоінформаційні системи. Підручник. Харків: ХНАДУ, 2012. 260 с.
4. Что такое ArcGIS?/ESRI/ - Киев: ECOMM Co, 2003. - 45 с
5. Österman, A. (2014). Map visualization in ArcGIS, QGIS and MapInfo

УДК 004

ІНТЕЛЕКТУАЛЬНІ ІНФОРМАЦІЙНІ СИСТЕМИ ТА СИСТЕМИ ШТУЧНОГО ІНТЕЛЕКТУ.

Пранов Л.І., магістр

e-mail: ipzm221_pli@student.ztu.edu.ua

Вакалюк Т.А. д.пед.н., професор

e-mail: kik_vta@ztu.edu.ua

Державний університет «Житомирська Політехніка»

Актуальність та постановка проблеми. Поняття штучного інтелекту, як, втім, і просто інтелект, дуже розпливчате. Якщо узагальнити все сказане за останні тридцять років, то виявляється, що людина просто хоче створити собі подібного в тій чи іншій формі, хоче, щоб якісь дії виконували раціональніше, з меншими витратами часу та енергії. З кінця 40-х років вчені все більшої кількості університетських і промислових дослідницьких лабораторій прямували до зухвалої мети: побудова комп'ютерів, що діють таким чином, що за результатами роботи їх неможливо було б відрізнити від людського розуму. Останнім часом спостерігається зростання інтересу до штучного інтелекту, спричинене підвищенням вимог до інформаційних систем. Розумніє програмне забезпечення, розумніє побутова техніка. Ми неухильно рухаємось до нової інформаційної революції,

Штучний інтелект зараз є «гарячою точкою» наукових досліджень. У цій точці, як у фокусі, сконцентровані найбільші зусилля кібернетиків, лінгвістів, психологів, філософів, математиків та інженерів. Саме тут вирішуються багато корінних питань, пов'язаних із шляхами розвитку наукової думки, з впливом досягнень у галузі обчислювальної техніки та робототехнік і на життя майбутніх поколінь людей. Тут виникають і набувають права громадянства нові методи наукових міждисциплінарних досліджень. Тут формується новий погляд на роль тих чи інших наукових результатів і виникає те, що можна назвати філософським осмисленням цих результатів. Тому я вважав актуальним розкрити цю тему у рефераті.

Основні матеріали дослідження. При сучасному рівні розвитку науки і техніки все більшого поширення набувають інформаційні системи з елементами штучного інтелекту на виробництві, побуті, військовій техніці, а також там, де присутність людини неможлива. Їх особливістю є наявність у самій системі підсистем аналізу та контролю стану, як самої системи управління, так і стану об'єкта управління з метою своєчасного прийняття рішення та реагування на зовнішні впливи та зміни у самій системі. Системи автоматичного контролю та управління повинні забезпечити необхідну точність регулювання та стійкість роботи у широкому діапазоні зміни параметрів. Цілями інтелектуальних інформаційних технологій є, по-перше, розширення кола завдань, які вирішуються за допомогою комп'ютерів, особливо в слабоструктурованих предметних галузях, і по-друге, підвищення рівня інтелектуальної інформаційної підтримки сучасного спеціаліста. Інтелектуальна інформаційна система (ШІС, англ. intelligent system) - різновид інтелектуальної системи, один із видів інформаційних систем, іноді ШІС називають системою, заснованою на знаннях. ШІС є комплексом програмних, лінгвістичних та логіко-математичних засобів для реалізації основного завдання: здійснення підтримки діяльності людини, наприклад, можливість пошуку інформації в режимі просунутого діалогу природною мовою.

ШІ широко використовується в охороні здоров'я для покращення догляду за пацієнтами та покращення результатів. Наприклад, IBM Watson може аналізувати величезні обсяги даних, щоб допомогти лікарям рекомендувати найефективніші методи лікування своїх пацієнтів.

ШІ також використовується в фінансах [2], щоб краще інвестувати рішення. Такі компанії, як Wealthfront, використовують алгоритми штучного інтелекту для автоматизації управління портфелем та надання персоналізованих рекомендацій щодо інвестицій.

Нарешті, ШІ дуже впливає на те, як ми спілкуємося. Такі послуги, як Google Перекладач [3], використовувати алгоритми машинного навчання для перекладу тексту між мовами в режимі реального часу.

В останні роки багато говорять про штучний інтелект (ШІ) та його потенціал змінити світ. Хоча деякі досі не впевнені, що таке ШІ насправді, важко заперечувати, що ця нова технологія вже дуже впливає на різні галузі.

Від фінансів до охорони здоров'я та виробництва підприємства починають використовувати ШІ потужними та інноваційними способами. Ось лише кілька прикладів того, як ШІ використовується сьогодні.

Штучний інтелект більше не є просто плодом нашої уяви. Це тут, це реально, і це змінює те, як ми живемо та працюємо. Від розпізнавання голосу і чат-ботів до прогнозової аналітики і безпілотних автомобілів штучний інтелект впливає на всі аспекти нашого життя.

Де сьогодні використовують штучний інтелект?

Технології штучного інтелекту [4] широко використовуються сьогодні у різних галузях промисловості. Можливо, найвідоміше використання ШІ пов'язані з робототехнікою, де роботи програмуються до виконання певних завдань. Однак ШІ також використовується в низці інших галузей, у тому числі:

-Виробництво: ШІ використовується для створення моделей продуктів з метою оптимізації виробничих процесів.

-Транспорт: ШІ використовується для створення систем, які можуть автономно керувати транспортними засобами, такими як безпілотні автомобілі.

-Охорона здоров'я: ШІ використовується для створення планів діагностики та лікування пацієнтів

-Фінанси: ШІ використовується для прийняття інвестиційних рішень та виявлення шахрайства

-Кібербезпека: ШІ використовується для створення систем, які можуть виявляти загрози безпеці та реагувати на них.

Оскільки технологія продовжує розвиватися, цілком імовірно, що ШІ використовуватиметься ще більшій кількості галузей і додатків. В даний час використовується ШІ, який стежить за речами/полями

Ось лише кілька прикладів того, де ШІ використовується сьогодні:

1. Автоматизоване обслуговування клієнтів: Багато компаній використовують ШІ для автоматизованого обслуговування клієнтів. Це можуть бути чат-боти, які відповідають на запитання клієнтів, або віртуальні помічники, які допомагають клієнтам з такими завданнями, як купівля або бронювання.

2. Виявлення шахрайства: Виявлення шахрайства - ще одна область, в якій штучний інтелект використовується з великим успіхом.

Банки та компанії, що випускають кредитні картки, використовують системи штучного інтелекту для виявлення шахрайських дій, таких як відмивання грошей та шахрайство з кредитними картками.

Ці системи здатні аналізувати великі набори даних, щоб знаходити закономірності, які можуть свідчити про шахрайство. Це дозволяє банкам ловити шахраїв до того, як вони зможуть завдати значних фінансових збитків.

3. Персональні рекомендації. Інтернет-магазини використовують штучний інтелект для надання покупцям персоналізованих рекомендацій щодо товарів. Аналізуючи минулу поведінку покупця, ШІ може давати рекомендації щодо продуктів, які, ймовірно, зацікавлять покупця. Сюди можна віднести робо-консультантів — це системи штучного інтелекту, які надають фінансові консультації інвесторам.

Вони використовують алгоритми для аналізу інвестиційних цілей користувача та терпимості до ризику, щоб дати рекомендації про те, куди вкласти свої гроші.

Робо-консультанти часто можуть дати більш точні та об'єктивні поради, ніж фінансові консультанти-люди, тому що вони не схильні до тих же когнітивних упереджень.

4. Прогнозна аналітика: ШІ використовується в різних галузях для прогнозування майбутніх тенденцій. Це можна використовувати для прийняття рішень щодо таких речей, як рівень запасів, маркетингові кампанії та розробка продукту.

5. Робототехніка. ШІ використовується при розробці роботів, які можуть виконувати такі завдання, як виробництво, охорона здоров'я та логістика.

6. Онлайн-рекомендації: наступного разу, коли ви робитимете покупки в Інтернеті, погляньте на рекомендації, які вам показують. Вони, мабуть, харчуються від ШІ. Компанії використовують нашу історію покупок та поведінку в Інтернеті, щоб показувати нам продукти, які можуть нас зацікавити.

7. Безпілотні автомобілі: Одне з найвідоміших застосувань ШІ-технологій все ще знаходяться на ранній стадії, вони можуть змінити те, як ми живемо та працюємо. Безпілотні автомобілі можуть зменшити кількість пробок на дорогах та аварійність, а також зробити транспорт більш ефективним та доступним для всіх.

Безпілотні автомобілі оснащені датчиками та програмним забезпеченням, які дозволяють їм переміщатися міськими вулицями та автомагістралями без участі людини-водія.

Одне з найбільш захоплюючих застосувань штучного інтелекту знаходиться в розробці безпілотних автомобілів [5].

Наведемо кілька прикладів ШІ, які сильно зробили крок вперед у 2022 році.

1. AlphaGo від Google [6]. У березні 2016 року весь світ з подивом спостерігав, як система штучного інтелекту Google AlphaGo перемогла 18-кратного чемпіона світу Лі Седоля у грі в го — подвиг, який багато експертів визнали «неможливим».

AlphaGo використовує методи, що застосовуються для розпізнавання образів, оцінки позиції та вибору найбільш вигідних ходів для даної позиції - глибинне навчання за допомогою згорткових нейронів позиції та оцінної мережі (англ. value network), вона допомагає оцінити позицію, не переглядаючи гру до кінця.

Для оцінки сил програми, творці організували турнір між AlphaGo та кращими вільними та комерційними програмами для гри в го (Crazy Stone, Zen, Pachi, Fuego), які використовували метод Монте-Карло та GnuGo, яка була найкращою вільною програмою для використання методу Монте. Карло. AlphaGo виграла 494 матчі із 495.

2. IBM Watson. IBM Watson - це система штучного інтелекту [7], який був розроблений IBM для участі в ігровому шоу Jeopardy!

Насамперед, Watson загалом – це великий набір пакетів програмного забезпечення, що використовують найрізноманітніші алгоритми. Частина цих пакетів доступна у хмарі, а частина – призначена для місцевого розгортання. Компанія IBM збрала різноманітні аналітичні модулі та побудувала систему, яка може справлятися з воістину величезною кількістю даних. Ця система працює як з цифровою, так і з текстовою інформацією на різних мовах, у тому числі російською – поки що на базовому, але все ж таки досить глибокому рівні.

При обробці інформації встановлюються зв'язки та кореляції між різними даними, подіями, фактами та явищами. Одне з головних завдань системи - виявлення зв'язків, які непомітні простому оку і які не можуть бути виявлені звичайним способом або зробити це стандартними методами складно. Наприклад, якщо якесь підприємство має систему моніторингу, яка видає тисячі змінних параметрів за хвилину, то жоден аналітик не в змозі проаналізувати таку інформацію оперативно. Інструментарій Watson це дозволяє зробити. Це якщо говорити про IBM Watson загалом.

3. Амазонка Alexa. Amazon Echo (або в народі Алекса) – це пристрій у формі колонки, яке підключається через Wi-Fi або Bluetooth, спілкується тільки по-англійськи (можливо, з часом освоїть і інші мови), здатна управляти будинком як супер-жінка, яка все пам'ятає, спритно оперує будь-якими приладами: телевізором, електричним чайником, мікрохвильовкою, термостатом, електричними воротами, світлом, температурою води в душі, жартує, читає аудіокниги, програв музику і миттєво налаштовується на радіо канали, нагадує, коли витягнути пиріг з духовки, виконує функцію в інтелектуальні ігри. Їй можна задати будь-яке питання. Вона досить обізнана, особливо в частині інформації з Вікіпедії, знає про погоду будь-якій частині планети в цельсіях і фаренгейтах, про обмінний курс валют і ціни акцій на біржі.

4. Apple Siri. Віртуальний помічник, розроблений Apple, який вбудований у комп'ютерні продукти iPhone, iPad та Mac.

Siri це програма для iPhone, що імітує робота-помічника, який розмовляє жіночим голосом. Практично Siri є голосовою пошуковою системою з елементами штучного інтелекту. Починаючи з версії iOS 8.3, що вийшла 8 квітня 2015 р., у програмі запроваджено підтримку російської мови. Програма Siri вбудована в операційну систему iOS і не вимагає окремої установки, оскільки компанія, яка розробила Siri, була в 2011 куплена фірмою Apple з потрухами.

Можливості та функціонал Siri воістину величезні: на прохання власника вона вміє дзвонити мамі, запускати програми на вашому Айфоні, ставити будильник на 7 ранку, відправляти повідомлення в Twitter та Фейсбук, робити покупки в інтернет-магазинах, шукати вам найближчі ресторани, повідомляти результати матчів та ще багато-багато подібних марних речей. А якщо ви спробуєте обдурити Siri, вона жартівливо постарается на вас образитися.

5. Coranta. Cortana — цифровий помічник Microsoft. Кортана, вперше запущена в 2014 році, спочатку була розроблена для Windows Phone 8.1 і включена в інші пристрої та операційні системи (ОС) Windows, включаючи Windows 10 і 11.

Cortana може виконувати такі практичні завдання:

- попереджати користувачів про майбутні зустрічі;
- відповідати на запитання користувачів;
- здійснювати пошук на ПК користувача та в Інтернеті;

- встановити нагадування;
- показувати користувачам погоду;
- і відкриті програми.

Cortana також може надавати контекстно-залежну інформацію на основі поточного розташування та активності користувача.

6. Прогнозна аналітика. Ці системи використовуються в різних галузях, включаючи охорону здоров'я, роздрібну торгівлю і виробництво.

Вони часто використовуються для прогнозування таких речей, як споживчий попит, потреби в обслуговуванні та фінансові ризики. Прогнозна аналітика може допомогти компаніям приймати більш виважені рішення та бути більш ефективними та результативними.

7. Почуття нюху. Одним із найяскравіших прикладів інтенсивного розвитку технологій штучного інтелекту став створений фахівцями Intel Labs та Корнельського університету AI-комплекс, здатний розрізняти запахи та імітувати роботу нюхової нервової системи людини. В основу розробки лягли нейроморфні процесори Intel Loihi [8], що поєднують процеси навчання, тренування та прийняття рішень в одному чіпі і дозволяють системі бути автономною та «кмітливою» без підключення до хмари (до бази даних). У ході експериментів спроектований та оснащений хімічними датчиками комплекс продемонстрував високу ефективність при розпізнаванні повітря запахів небезпечних речовин навіть за умов сильних перешкод. Такі рішення, впевнені в Intel, допоможуть у розвитку робототехніки, коли роботи зможуть самі сортувати продукти, орієнтуючись на запах,

8. Медицина [10]. Значного прогресу в 2020 році розробники AI-систем досягли в галузі медицини. Так, компанією DeepMind, що належить холдингу Alphabet (Google), було оголошено про суттєвий прорив у передбаченні фолдингу (згортання) білків. Проблема передбачення згортання білка вважається однією зі 125 найважливіших для вирішення завдань сучасності, а також однією з найбільших проблем біології за останні 50 років. Справа в тому, що білки збираються з лінійних послідовностей амінокислот, які після синтезу набувають унікальної просторової форми, і таких форм безліч. На даний момент із сотень мільйонів білків (комбінацій амінокислот) вивчено лише 0,1 % сполук, чия просторова структура також добре відома. Невідомі білки, а також сполуки, властивості яких ще не були підтверджені експериментальним шляхом, вчені намагаються передбачити за допомогою комп'ютерів. Але досі ніхто не міг з достатньою мірою точності обчислити, яку 3D-форму прийме білок із заданих набору та послідовностей амінокислот. У DeepMind стверджують, що знайшли ключ до вирішення цього завдання. Якщо це дійсно так, то на нас може чекати прорив у відкритті нових ліків та вакцин, а також у розумінні виникнення та перебігу багатьох хвороб.

На думку експертів, повсюдне впровадження технологій штучного інтелекту в медицину обіцяє величезні перспективи, що дозволяє значно покращити якість лікування та принципово змінити підхід до ранньої діагностики небезпечних захворювань. Аналогічної позиції дотримується відомий Білл Гейтс (Bill Gates), який вважає, що використання нейронних мереж у складних біологічних системах здатне кардинально змінити життя людей у майбутньому. Потенціал AI тільки зараз починає розкриватися, вважає засновник Microsoft, - обчислювальні потужності у цій галузі подвоюються кожні 3,5 місяці. Пан Гейтс зазначив, що, поряд з поліпшеннями в обробці даних, це дає можливість синтезувати,

аналізувати, виявляти закономірності та робити прогнози щодо значно більшої кількості напрямків, ніж може усвідомити людина.

9. Перетворення розумових операцій [11]. Читання думок поки що залишається долею фантастичних фільмів та книг. Однак наука і техніка не стоять на місці, і є всі підстави вважати, що в майбутньому такі технології стануть реальністю. На крок уперед у цьому напрямку вдалося просунутися групі вчених з Каліфорнійського університету в Сан-Франциско, які експериментально довели можливість розпізнавання нервових сигналів у головному мозку людини та їх трансляції у зрозумілі слова за допомогою рекурентної нейромережі та вживлених у мозок електродів. В експерименті взяли участь пацієнти з епілепсією, електроди яких були вживлені для боротьби з неврологічним захворюванням та відстеження нападів. Так вийшло, що частина електродів опинилися в зонах мозку, в яких відбувається підбір слів, складання виразів і здійснюється зворотний зв'язок із ділянками мозку, які сприймають власну мову людини. Випробуванням було запропоновано подумки, а потім вголос вимовити кілька речень з обмеженим набором слів. Одночасно знімалися сигнали з імплантованих у мозок датчиків. Отримані дані були передані в нейронну мережу для навчання, а проміжний результат був відданий для аналізу іншої мережі AI. Імовірність помилкового визначення слів становила лише 3 відсотки. Вражаючий показник!

10. Музика. Цікаве застосування штучному інтелекту знайшли програмісти компанії OpenAI, які розробили Jukebox [12] - штучний інтелект, що складає музику з осмисленими текстами та вокалом. Для навчання нейронної мережі системи було використано безліч уривків із пісень найрізноманітніших жанрів, від року, джазу та блюзу до хіп-хопу з кантрі та класичними творами. Такий підхід дозволив команді OpenAI розширити можливості проекту та досягти ефекту імітації музичних композицій виконавців, на яких він навчався. Наприклад, Jukebox може написати музику в стилі кантрі-співака Джонні Кеша, репера Drake. На створення однієї хвилини музичного треку з вокальними партіями штучному інтелекту потрібно близько 9 годин та величезний обсяг обчислювальних ресурсів. Саме тому компанія не може надати відкритий доступ до своєї AI-системи. Проте розробники опублікували результати роботи Jukebox. Послухати їх можна на сайті jukebox.openai.com.

11. Живопис. Електронний розум знайшов застосування в образотворчому мистецтві. Якраз у середині 2020 року стало відомо про створення фахівцями Лабораторії штучного інтелекту Масачусетського технологічного інституту системи машинного навчання Timescraft [13], що дозволяє відтворити процес написання картин та нанесення мазків для творів живопису відомих художників, чи то Моне, Вінсент Ван Гог чи Сальвадор. Повідомляється, що нейромережа спочатку була навчена на двох сотнях відеороликів із прискореною зйомкою техніки написання реальних цифрових та акварельних картин. Після цього дослідники створили згорткову нейронну мережу, яка має «деконструювати» художні роботи на основі своїх знань про процес створення картин. В результаті система Timescraft зуміла показати більш високу ефективність, ніж існуючі подібні проекти. більш ніж у 90% випадків. Непоганий результат. Крім уроків віртуальної історії, AI-система Timescraft може бути корисною для ілюстрації загальних прийомів та техніки малювання новачкам.

12. Ігри. Чимало застосування штучний інтелект знайшов і в інших сферах людської діяльності. Наприклад, NVIDIA задіяла AI для відтворення ігрового процесу знаменитої аркадної відеоігри Pac-Man за допомогою нейромережі GameGAN. Для вирішення цього завдання штучному інтелекту знадобилося лише

4 дні. Компанія навчила нейромережу за допомогою 50 тисяч ігрових сесій у Рас-Ман. Потім перед нею поставили завдання - відтворити побачену гру повністю, починаючи від статичних стін і крапок і закінчуючи привидами, що рухаються, і самим Пакманом. Навчання та відтворення гри проводилося з використанням квартету графічних прискорювачів NVIDIA Quadro GP100 [14]. Найцікавіше полягає в тому, що GameGAN не надали доступу до оригінального коду гри або її движка. Все навчання зводилося до того, що одна нейромережа спостерігала за тим, як у Рас-Ман грала інша нейромережа. «Для створення гри, подібної до Рас-Ман, програмісту потрібно придумати і прописати правила поведінки та взаємодії всіх наявних агентів усередині гри. Це дуже копітка робота. GameGAN може спростити це завдання. Нейросеть здатна навчатися нових правил через спостереження. В ідеалі алгоритми, подібні до GameGAN, можна буде навчити процедурної генерації правил для тієї гри, яку ви хочете створити», — пояснюють дослідники NVIDIA, які підкреслюють, що в перспективі їх розробка може бути задіяна не тільки в ігровій індустрії, а й в інших областях. Нейросеть здатна навчатися нових правил через спостереження. В ідеалі алгоритми, подібні до GameGAN, можна буде навчити процедурної генерації правил для тієї гри, яку ви хочете створити», — пояснюють дослідники NVIDIA, які підкреслюють, що в перспективі їх розробка може бути задіяна не тільки в ігровій індустрії, а й в інших областях.

Висновок. Це лише кілька прикладів того, як сьогодні використовується штучний інтелект. Оскільки технології продовжують розвиватись, ми можемо очікувати ще більше дивовижних застосувань ШІ в майбутньому.

Список використаних джерел:

1. <https://www.daijiworld.com/news/newsDisplay?newsID=948538>
2. <https://www.bloggersideas.com/ru/coinzilla-review-ad-network/>
3. <https://www.bloggersideas.com/ru/google-translate-alternatives/>
4. <https://builtin.com/companies>
5. <https://www.investopedia.com/articles/investing/052014/how-googles-selfdriving-car-will-change-everything.asp>
6. <https://www.science.org/content/article/computer-program-can-beat-humans-go-no-human-instruction>
7. <https://www.bloggersideas.com/ru/where-is-artificial-intelligence-used-today/>
8. <https://www.intel.com>
9. <https://www.sciencedirect.com/science/article/pii/S0959438817300612>
10. <https://habr.com/ua/post/431948/>
11. <https://www.nature.com/articles/s41593-020-0608-8>
12. <https://en.wikipedia.org/wiki/Jukebox>
13. <https://xamyzhao.github.io/timecraft/>
14. <https://www.nvidia.com/content/dam/en-zz/Solutions/design-visualization/productspace/quadro/quadro-desktop/quadro-pascal-gp100-data-sheet-us-nv-704562-r1.pdf>

УДК: 378.14(72)

ОСОБЛИВОСТІ ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В СИСТЕМІ ВИЩОЇ ОСВІТИ США

Кулешов С.О.

e-mail: sergius.kul7@gmail.com

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Актуальність та постановка проблеми. Сьогодні системи штучного інтелекту відкривають нові можливості для викладання та навчання у вищій освіті США, але ще не готові повністю замінити викладачів. З іншого боку можна стверджувати, що використання штучного інтелекту значно посилює продуктивність освітньої діяльності. Масове використання штучного інтелекту несе з собою як надзвичайні можливості, так і ризики. Частка використання програмованих систем і машин замість людини в світовій практиці, зокрема в освіті, промисловості, економіці постійно збільшується через економічну вигоду, пандемії, військові конфлікти, природні катаклізми тощо. Така тенденція завдяки швидкому розвитку інформаційно-комунікативних технологій та штучного інтелекту згодом може призвести до повної заміни людини як викладача. З іншого боку великий потенціал таких технологій у вищій освіті за умови правильного використання розширює людські здібності та можливості дослідження, викладання та навчання. Тому це питання вимагає ретельного аналізу та досліджень з академічної точки зору.

Основні матеріали дослідження. Адміністрації американських університетів вже активно використовують системи штучного інтелекту для надання адміністративних послуг. Завдяки ним автоматизується велика кількість роботи з керування, обробки та розповсюдження інформації. Наприклад, система Watson від американської компанії IBM, яка була створена для розуміння питань та пошуку відповіді на них, може надавати поради студентам в будь який час [2]. Слід додати, що автоматизовані машини можуть стати вигідним способом економії бюджету університетів. Завдяки автоматизації роботи з інформацією штучний інтелект вже здатний замінити велику кількість адміністративного персоналу та асистентів викладачів у вищих навчальних закладах. Але поки такі системи не можливо використовувати для більш складних завдань вищої освіти. Для штучного інтелекту залишається важким розпізнавання людських емоцій, гумору або іронії тощо. Це стає величезним бар'єром для широкого використання рішень штучного інтелекту в навчальному процесі у вищій освіті.

Роль технологій в навчанні полягає в тому, щоб покращити людське мислення та розширити освітній процес, бо освіта є виключно людиноцентричною діяльністю, а не рішенням, орієнтованим на технології. Якщо не брати це до уваги, з часом освіта стане сукупністю алгоритмів для доставки контенту, контролю та оцінювання. Завдання закладів вищої освіти навчити майбутніх фахівців працювати з системами штучного інтелекту як інструментом з великою кількістю можливостей, що може не тільки використовувати джерела інформації, але й керувати ними. Останнє є актуальним питанням для сучасного світу, бо мережа Інтернет, яка поєднує величезну кількість інформаційних сховищ, має великий вплив на суспільство та безліч медіа інструментів для формування суспільної думки. Ці інструменти працюють завдяки алгоритмам, які написані програмістами, тому є небезпека того, що той, хто контролює системи

штучного інтелекту, може контролювати й суспільну думку. Саме тому необхідність досліджень питання штучного інтелекту в освіті є дуже актуальним на сьогодні.

На думку Вільяма Годдарда штучний інтелект може виконувати наступні функції у вищій освіті:

1. автоматизація базових або повторюваних дій – автоматизоване оцінювання майже для всіх видів тестування з вибором відповідей і заповнення пропусків;
2. надання персоналізованих навчальних платформ – зростаюча кількість адаптивних навчальних програм, ігор і програмного забезпечення використовує штучний інтелект для створення навчального середовища з урахуванням індивідуальних потреб здобувача освітніх послуг;
3. виявлення прогалин і недоліків у навчальній програмі – такі системи, як платформа електронного навчання Coursera, можуть сповіщати, якщо велика кількість учнів надає неправильну відповідь або має проблеми з певною темою;
4. віртуальні репетитори – системи репетиторства на базі штучного інтелекту;
5. як канал зворотного зв'язку для студентів і викладачів – системи штучного інтелекту можуть відстежувати прогрес студентів і сповіщати викладачів про можливі проблеми.
6. надання нових способів взаємодії з інформацією – з розвитком та впровадженням нових технологій студенти здобувають нові можливості для досліджень;
7. створення нової динаміки з вчителями – оскільки штучний інтелект стає все більш інтегрованим у систему освіти, вчителі можуть доповнювати уроки різноманітними завданнями з варіативним вирішенням;
8. зменшення академічного та соціального тиску – створюється середовище, де студенти не відчують тиску, порівнюючи себе зі своїми колегами.
9. надання нових шляхів для набору, навчання та підтримки – завдяки інтелектуальному збору даних система може надавати інформацію та рекомендації для покращення життя студентів та роботи навчальних закладів;
10. доступність навчання – системи штучного інтелекту, програмне забезпечення та підтримка в поєднанні з підключенням до Інтернету дозволяють учням навчатися з будь-якої точки світу в будь-який час [3].

Завдяки штучному інтелекту зараз вдосконалюються інструменти, які щодня використовуються в містах і кампусах в усьому світі. Від пошукових систем Інтернету, функцій і додатків смартфонів до громадського транспорту та побутової техніки. Зараз студенти мають величезну кількість можливостей і проблем для навчання та дослідження у вищій освіті. Рішення для взаємодії та співпраці людини зі штучним інтелектом вже доступні для викладачів, які можуть застосувати їх у навчанні, щоб підвищити рівень залучення учнів та варіативність завдань для більш захоплюючого процесу та для підвищення стандартів професійної підготовки.

Складні обчислювальні системи, що використовують алгоритми машинного навчання, можуть обслуговувати людей з усіма типами здібностей і певною мірою брати участь у людських процесах і складних задачах обробки, які можна використовувати під час викладання та навчання. Тому це стимулює таких гігантів IT-індустрії, як Google та Apple, робити великі інвестиції в проекти зі створення нових форм штучного інтелекту. Такий інтерес та інвестиції у штучний інтелект незабаром вплинуть на університети, бо велика частка досліджень припадає на базу американських закладів вищої освіти. Швидше за все,

фінансовий тиск, пов'язаний із великою кількістю студентів, які зараз здобувають вищу освіту, керуючись метою демократизації вищої освіти, і міжнародний студентський ринок будуть вагомою силою для розвитку систем штучного інтелекту. Окрім цього має місце й політичний заклик скоротити державне фінансування університетів, й підвищена популярність масових відкритих он-лайн курсів.

Концепція людства та людських можливостей наразі переосмислюються з безпрецедентною швидкістю: технологія швидко розширює потенціал використання функцій штучного інтелекту для вдосконалення наших навичок і здібностей. Прикладом є курс, запропонований професором зі штучного інтелекту Ашоком Гоелом, в онлайн-програмі магістра комп'ютерних наук у Технологічному інституті Джорджії у США. Асистент викладача, Джил Уотсон, надавала якісну підтримку студентам через мережу Інтернет, за що студенти пропонували номінувати її на нагороду. Наприкінці курсу студентам було повідомлено, що Джил Уотсон була не реальною людиною, а роботом-вчителем, віртуальним помічником вчителя, заснованим на платформі Watson від IBM [4;5].

Висновок. Зараз вища освіта США та усього світу робить активні кроки у впровадженні можливостей, які відкриває штучний інтелект у викладанні, навчанні, організації та управлінні вищою освітою. Персоналізоване навчання за допомогою вчителя-бота або «хмарного лектора» можна використовувати для змішаних курсів або повністю онлайн-курсів. Технології для створення ботів надають багато можливостей для адміністративної частини викладання, пов'язаної, в основному, з доставкою контенту, адміністративним і зворотнім зв'язком і наглядом. Розвиток штучного інтелекту настільки поширений, що комп'ютер може служити персоналізованим репетитором, здатним керувати навчанням і залученістю студентів. Це відкриває тривожну можливість побачити поверхневий, але прибутковий підхід, коли навчання замінюється автоматизованими рішеннями штучного інтелекту. Тому потрібно бути обережними у впровадженні штучного інтелекту в систему вищої освіти, бо освіта не повинна повністю контролюватися автоматизованими системами.

Список використаних джерел:

1. Popenici S. A. D., Kerr S. Exploring the impact of artificial intelligence on teaching and learning in higher education. *Research and Practice in Technology Enhanced Learning*. – 2017. – Т. 12. – №. 1. – С. 1-13.
2. IBM's Watson supercomputer signs on Aussie customers. URL: <https://www.afr.com/technology/ibm-s-watson-supercomputer-signs-on-aussie-customers-20141008-11bkwq>.
3. William Goddard. AI in Education. URL: <https://itchronicles.com/artificial-intelligence/ai-in-education>.
4. Meet Jill Watson: Georgia Tech's first AI teaching assistant. URL: <https://pe.gatech.edu/blog/meet-jill-watson-georgia-techs-first-ai-teaching-assistant>.
5. Ashok Goel. A teaching assistant named Jill Watson. URL: <https://www.youtube.com/watch?v=WbCguICyfTA&t=1042s>.

УДК 004.891.2

ВАЖЛИВІСТЬ НАПИСАННЯ ЮНІТ ТЕСТІВ ДЛЯ ПРОГРАМНОГО КОДУ

Сікайло В.О.,

e-mail: dren7777@gmail.com

Кравченко С.М., ст. викл.

e-mail: svetlanakravchenko2012@gmail.com

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. Юніт тестування (Модульне тестування) - це метод тестування програмного забезпечення, який полягає в окремому тестуванні кожного модуля (юніту) коду програми. Модулем називають найменшу частину програми, яка може бути протестованою. В об'єктно-орієнтованому програмуванні модулем вважається метод.

При розробці корпоративних додатків практика написання юніт тестів стає обов'язковою у великій кількості сучасних компаній. Цей процес пізнання важливості юніт тестів триває і досі.

Вважається що юніт-тестування покращує якість програмного коду. І це дійсно так: необхідність написання тестів призводить до покращення якості та чистоти коду. Але це не основна ціль юніт тестів, а лише сторонній ефект. Насправді, реальна ціль покриття коду тестами - забезпечення стабільного росту програмної системи.

Основні матеріали дослідження. З початку створення програми її розвиток відбувається швидкими темпами, тому що немає факторів які могли б гальмувати. Ще немає невдалих архітектурних рішень; ще немає кодової бази яку потрібно підтримувати.

Проблематика. З плином часу система стає дедалі складнішою, її розмір та складність виростають. Це призводить до того, що для внесення певних змін у систему необхідно докладати дедалі більше зусиль. Відповідно і швидкість розробки стрімко зменшується - іноді навіть до стану, в якому проект взагалі перестає рухатись вперед.

Як результат, основна ціль покриття коду юніт тестами є: забезпечення стабільності росту проекту, легкості внесення змін в існуючий код, спрощене впровадження та тестування інтеграцій, легкість рефакторингу та інше.

На наступному рисунку візуально відображена залежність легкості розширення проекту від використання юніт тестування.

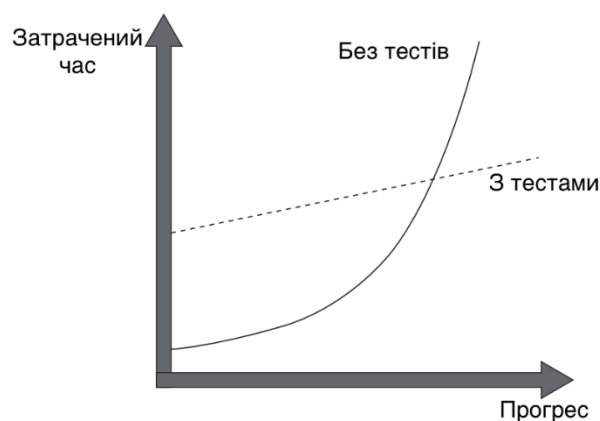


Рисунок 1. Залежність затраченого часу для прогресу від тестування

Сама можливість покриття коду юніт тестами є хорошою ознакою якості цього коду. Згідно з цим критерієм, неможливість юніт тестування говорить про те, що код потребує покращення. Найпоширенішою проблемою яку виявляє неможливість написання тестів, зазвичай, є tight coupling (з Англ. - тісна зв'язність). Це означає що частини коду тісно зв'язані між собою, тобто недостатньо ізольовані один від одного. Це, в свою чергу, призводить до неможливості тестування окремих юнітів.

При виявленні проблеми занадто тісно зв'язаних компонентів слід вдатися до SOLID принципів щоб побороти цю проблему. Найперший принцип, принцип єдиної відповідальності, ідеально підходить для вирішення цієї проблеми. Слідування цьому принципу призведе до розбиття компонентів системи на менші частини, кожна з яких буде мати одну і тільки одну область відповідальності. Саме завдяки цієї властивості такий компонент буде легко покрити юніт тестами. Тестування буде чітким та прозорим, таким як і сам компонент.

Дослідження ефективності юніт тестування призвело до появи цілого підходу до розробки, при якому написання тестів відбувається до фактичного написання програмного коду. Такий підхід називається Test Driven Development (з Англ. Розробка через тестування). Розробка через тестування - це технологія розробки програмного забезпечення, яка використовує короткі ітерації розробки, що починаються з попереднього написання тестів, які визначають необхідні покращення або нові функції. Кожна ітерація має на меті розробити код, який пройде ці тести.

Філософія розробки через тестування проголошує три правила, які призведуть до написання якісного програмного коду:

1. Не можна писати жодного вихідного коду, доки спершу не написано падаючого юніт-тесту.
2. Не можна писати більше юніт-тесту ніж необхідно для падіння (непроходження тесту). Помилка компіляції - це також падіння.
3. Не можна писати більше вихідного коду, ніж необхідно для проходження впалого юніт-тесту.

Таким чином написаний програмний код завжди буде відповідати бізнес потребам, керуючись якими були написані юніт тести. Це призводить до великого відсотку покриття коду тестами, що в свою чергу забезпечує стабільність росту проекту.

Висновок. Як результат, саме використання юніт тестів попереджує про погану архітектуру та дизайн системи, привносить стабільність росту проекту, забезпечує надійність при рефакторингу, та відділення абстракції від реалізації. Тести забезпечують легкість внесення змін в проект, не переймаючись тим, що якась інша частина може вийти з ладу. При змінах будь-якої частини програмного продукту завжди є можливість дуже легко запустити юніт тести та переконатись що все працює як очікувалось. Юніт тестування це одна з найважливіших аспектів сучасної розробки програмного забезпечення.

Список використаних джерел:

1. Хариков В. Принципи Юніт Тестування / Володимир Хариков., 2021. – 320 с.
2. Мартін Р. Чиста архітектура / Роберт Мартін., 2019. – 512 с. – (Фабула).

УДК 004.891.2

ПЕРСПЕКТИВИ ВИКОРИСТАННЯ ДИНАМІЧНИХ ПРОТОКОЛІВ КЕРУВАННЯ VLAN - МЕРЕЖАМИ У ХМАРНИХ СЕРВІСАХ

Сідлецька Д.Р., магістр.

e-mail: kbm221_sdr@student.ztu.edu.ua

Єфіменко А.А., к. т. н.

e-mail: yefimenko.andrii@ztu.edu.ua

Кручинський Я.Т., ст. викл.

e-mail: kkik_kyat@ztu.edu.ua

Вакалюк Т.А., д. пед. н.

e-mail: kik_vta@ztu.edu.ua

Державний університет «Житомирська політехніка»

Актуальність та постановка проблеми. З розвитком хмарних технологій підіймається все більше питань про захист та оптимізацію хмарних сервісів. Кожен потенційний користувач хмари обирає тип хмарного сервісу, який буде задовольняти їх потреби. Існують компанії, яким не підходить публічна хмара тому, що вони не можуть відмовитись від використання фізичного обладнання зовсім. Тому компанії обирають собі у користування приватні хмари, або як альтернативу гібридні, виходячи зі своїх потреб.

Безпека хмари тісно пов'язана з використанням віртуальних машин та віртуально локальних мереж. Якщо користувач зупиняється на приватній хмарі, то необхідно фізично відокремити мережу, яка буде утворювати дану хмару, оскільки це є важливим з міркувань експлуатації та безпеки. Хмари використовують одну із форм віртуальних мереж для абстрагування фізичної мережі та створення пулу мережевих ресурсів. Одна з форм віртуальних мереж це VLAN. Мережі VLAN не призначені для хмарної віртуалізації чи безпеки, і їх не слід розглядати самі по собі як ефективний засіб контролю безпеки для ізоляції мереж [1]. Але у приватній хмарі для захисту віртуальних машин, що розташовані на одному або кількох фізичних серверах, використання мережі VLAN буде ефективним.

Основні матеріали дослідження. Віртуальні мережі мають суттєві відмінності від фізичних мереж. Дані мережі працюють у фізичних мережах, але абстракція дає можливість глибоко модифікувати поведінку мережі так, щоб мати вплив на безпеку багатьох технологій та процесів [2].

Віртуальна локальна мережа визначає широкомовний домен, у якому лише ті вузли, що згруповані в кластері, можуть обмінюватися повідомленнями за бажанням. Магістральне з'єднання, яке використовується між інфраструктурою VLAN, може забезпечити дуже високий рівень безпеки з великою гнучкістю. З вище зазначеного можна зробити висновок, що VLAN підтримує багато функцій хмарних сервісів, зокрема масштабованість, безпека і використання віртуальних машин. Фактично комутатори підтримують віртуалізацію пристроїв як на рівні 2 (канал даних) моделі OSI, так і на рівні 3 (мережевий рівень) [3].

Для того, щоб коректно працювала віртуально локальна мережа, необхідно, щоб в базі даних фільтрації (Filtering Database) знаходилась інформація про членство в VLAN. Дана інформація потрібна для прийняття правильного рішення (переслати або відкинути) при передачі кадрів між портами комутатора [4].

Динамічна конфігурація VLAN дає можливість для визначення членства в мережі спираючись на характеристики використовуваних пристроїв, а не місця розташування їх порту комутатора. На основі протоколу GVRP (GARP VLAN Registration Protocol) встановлюється членство в динамічних VLAN на магістральних інтерфейсах комутаторів. Зазначений протокол GARP (Generic Attribute Registration Protocol) застосовується для реєстрації та скасування реєстрації атрибутів, таких як VID. Даний протокол GVRP створює та

розповсюджує динамічно записи про нову реєстрацію в VLAN (Dynamic VLAN Registration Entries). Записи, що динамічно розповсюджуються містять в собі інформацію про зареєстровані на порту VLAN та зміну їх переліку при зміні топології мережі. Ці записи створюються, оновлюються і видаляються в процесі роботи протоколу GVRP. Для активізації роботи протоколу GVRP на портах комутатора використовуються його керуючі елементи, а також для зазначення того, чи ця VLAN може бути зареєстрована на порту.

Висновок. Захист віртуальних машин від будь-якого витоку даних або несанкціонованого доступу є одним із головних обов'язків хмарних провайдерів. При використанні приватної хмари відповідальність лягає на користувача, за правильно налаштовану інфраструктуру, мережу, доступ до інфраструктури тощо.

Для великих підприємств, яким підходить тільки приватна хмара і які своєю чергою досить швидко масштабуються, буде досить зручним використання саме динамічних протоколів VLAN - мереж. Адже вони дають можливість для збільшення ефективності, зменшення часу на оновлення конфігурацій вручну та налаштувань, уникнення помилок які зумовлюються людським фактором та багато іншого.

Хоча на даний час динамічні VLAN не застосовуються широко, в цьому є перспектива та розвиток. Особливо з розвитком хмарних технологій та перенесенням більшості ресурсів компаній, державних установ та маленьких підприємств у хмару. З кожним днем цифра тих, хто переходить у хмару росте і досить з великою прогресією. Незважаючи на це, велика кількість компаній не може ще відмовитись від використання фізичного обладнання, і хоча темпи переходу у хмару є досить значні, відмова від фізичного обладнання станеться не скоро.

Список використаних джерел:

1. «Віртуалізація хмарної мережі: переваги SDN над VLAN» [Cloud Network Virtualization: Benefits of SDN over VLAN] Офіційний пресреліз CSA [Cloud Security Alliance(CSA) Official Press Release]/CSA. - 2021. - Режим доступу: <https://cloudsecurityalliance.org/blog/2021/06/25/cloud-network-virtualization-benefits-of-sdn-over-vlan/>
2. Cloud Security Alliance's Security Guidance for Critical Areas of Focus in Cloud Computing v4.0/CSA. - 2021.- Режим доступу до ресурсу: <https://cloudsecurityalliance.org/download/securityguidance-v4/>.
3. Сірацці Ф., Краснов А. «Хмарна безпека: реалізація віртуалізованої VLAN (V2LAN)» [Sirazzi F., Krasnov A. Cloud Security: A Virtualized VLAN (V2LAN) Implementation] / Міжнародна конференція з питань взаємодії людини і комп'ютера[International Conference on Human-Computer Interaction] - 2016. - Режим доступу до ресурсу: https://www.researchgate.net/publication/304107988_Cloud_Security_A_Virtualized_VLAN_V2LAN_Implementation
4. Воропаєва К. А. Статичні та динамічні VLAN-мережі. Взаємодія однорангових VLAN-мереж. "Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення (випуск 53)" : Міжнародна наукова інтернет-конф., 6–16 листопада 2020 р. – Тернопіль, 2020. – С. 20–21. - Режим доступу: https://openarchive.nure.ua/bitstream/document/13956/1/Voropaeva_20_21.pdf
5. Гунько М.А. Особливості побудови хмарних брандмауер-систем захисту веб-ресурсів / М.А. Гунько, науковий керівник – к.т.н. Ткачов В.М. // РАДІОЕЛЕКТРОНІКА І МОЛОДЬ У ХХІ СТОЛІТТІ : Тези доповіді / Харківський національний університет радіоелектроніки. — Харків, 2019. — С.145-146.

ПОКАЖЧИК АВТОРІВ

Martin Lesage	10	Лейба Я.А.	340
Oleksii Kucherenko	343	Лубко Д.В.	74, 189, 239, 368, 384
Ukrainets M.O.	377	Ляшенко А.І.	210
Vakaliuk T.A.	377	Мартиць Д.С.	83
Yevhenii Krupchak	280	Мацулевич О.Є.	28, 87, 139, 348
Yurii Novikov	280	Мелешко О.Д.	436
Zaitseva A.M.	126	Михайленко О.Ю.	360
Андрусенко О.М.	225, 397	Мінаєнко А.С.	391
Антонова Г.В.	35, 93, 203, 353, 360	Мірошниченко М.Ю.	147, 154, 162
Арієнєнєков О.М.	309	Назаров Є.М.	246, 297
Білявський Н.А.	257, 259, 318	Нарватов О.П.	251
Бондар А.М.	58	Новіков Ю.С.	309
Бондаренко Л.Ю.	93, 175, 312, 353	Новохацький В.С.	104
Борейченко Г.О.	67	Островська К.Ю.	106, 391, 394
Вакалюк Т.А.	102, 104, 118, 128, 136, 225, 227, 255, 325, 330, 345, 375, 397, 399, 403, 440, 452	Перевалова А.Д.	337
Ващенко К.Я.	411	Пироженко М.Ю.	428
Величко С.Д.	332	Поліщук О.Д.	63
Верещага Ю.В.	283	Полозов Д.М.	251
Вершков О.О.	93, 312	Полотай О.І.	195
Гавриленко Є.А.	147, 312	Поплавський В.С.	100, 109
Гешева Г.В.	46	Поспелов М.А.	139
Гладун А.Я.	422, 430	Постол Ю.О.	207
Гольцов В.В.	111	Пранов Л.І.	440
Гордєєв Р.С.	345, 399	Рогбак К.С.	394
Гузюк В.В.	306	Рогущина Ю.В.	414, 422
Дашивець Г.І.	58	Русятинська А.О.	263, 266, 322
Дереза О.О.	35, 87, 177, 353	Савчук Ю.В.	253, 270
Дереза С.В.	177	Семенов Є.О.	216
Дмитренко І.А.	220	Сідлецька Д.Р.	452
Дуб А.С.	131	Сікайло В.О.	450
Дяденчук А.Ф.	171	Сікора Я.Б.	210
Єфіменко А.А.	128, 452	Сіциліцин Ю.О.	216, 222
Загацький В.В.	403	Скріпченко Д.Г.	227
Засипко В.П.	405	Стеценко К.О.	169
Зінов'єва О.Г.	69, 74, 239	Стовпченко І.В.	106
Івженко О.В.	360	Супрун М.В.	289
Ковальчук О.А.	327	Тетервак І.Р.	35, 87, 93, 139, 348, 353
Каліберда Ю.О.	229	Українець М.О.	375
Катане О.Г.	231	Ускова С.О.	175
Кашкарьов А.О.	181	Фельдшерев Е.О.	272
Кияшенко А.С.	102, 118, 255	Хала К.О.	430
Козирєва Т.А.	220	Хімічук І.С.	120, 123
Коломоєць Д.А.	51, 379	Холодняк Ю.В.	147, 162, 289
Кравченко К.Р.	187	Хохлов М.О.	128
Кравченко С.М.	450	Чаплінський А.П.	28
Кручинський Я.Т.	452	Чаплінський А.П.	360
Кузьмук В.О.	325, 330	Чижмотря О.В.	67
Кулєшов С.О.	447	Чижмотря О.Г.	337
Курачинська А.Р.	136	Читулян В.О.	45
Кучерков А.О.	80	Шерстяних М.О.	106
		Широкопетлева М.С.	251, 340
		Яджак М.С.	63

НАУКОВЕ ВИДАННЯ

МАТЕРІАЛИ

**III Всеукраїнської науково-практичної інтернет-конференції
«Сучасні комп'ютерні та інформаційні системи і
технології»**

12 - 19 грудня 2022 р.

Відповідальний за випуск: Холодняк Ю.В., в. о. завідувача кафедри комп'ютерних наук Таврійського державного агротехнологічного університету імені Дмитра Моторного

Редактор: Ю.В. Холодняк, Г.В. Гешева

Дизайн і верстка: Максимчук С.М.

Адреса оргкомітету конференції:

Таврійський державний агротехнологічний університет імені Дмитра Моторного, факультет енергетики і комп'ютерних технологій,
кафедра комп'ютерних наук
69600, Україна, Запорізька обл., м. Запоріжжя, вул. Жуковського, 66
e-mail: cs.conference@tsatu.edu.ua

Сайт конференції: <https://sites.google.com/tsatu.edu.ua/csconference2021>

**Редакційна колегія не несе відповідальності за зміст
представлених матеріалів**

