

Секція №3. Інформаційні технології загального призначення суспільством. Зазначимо, що без дизайну багато інновацій не знайшли б такого широкого визнання. Наприклад, дизайн техніки від Apple визначив її привабливість та популярність, надаючи їй конкурентну перевагу на ринку. Отже, дизайн визнається як невід’ємна та важлива частина сучасного суспільства, впливаючи на розвиток технологій та їхню інтеграцію в повсякденне життя. У майбутньому роль дизайну в інтеграції нових технологій буде ще більш зростати. Оскільки нові технології стають все більш складними, дизайн буде відігравати важливу роль у їхньому розумінні та використанні.

Список літератури

1. 10 принципів гарного дизайну від Дітера Рамса. URL: <https://www.komarov.design/10-printsipiv-vdalogho-dizainu-ditiera/> (дата звернення: 29.11.2023).
2. Майбутнє дизайну інтер’єру: експерти про те, як може виглядати дизайн інтер’єру. URL: <https://www.homedesign.in.ua/maybutnie-dyzaynu-inter-ieru-eksperty-pro-te-iak-mozhe-vyhliadaty-dyzayn-inter-ieru/> (дата звернення: 29.11.2023).

Валієва Карина Мадатівна
студентка

Гешева Ганна Валеріївна
*асистентка кафедри комп’ютерних наук,
Таврійський державний агротехнологічний університет
імені Дмитра Моторного,
м. Запоріжжя, Україна*

ТРЕНДИ В МЕТОДОЛОГІЇ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Звідки беруться тренди в розробці програмного забезпечення? Під час останніх десятиліть індустрія програмування і розробки програмного забезпечення пережила багато трансформацій. Вона стала суттєвою складовою нашого цифрового світу, і разом з цим виникають нові підходи, методології та техніки розробки, які визначають її розвиток. Ці підходи і структури розвиваються відповідно до змін в технологіях, вимог користувачів і ринкових тенденцій.

Одним з провідних трендів є певна методика для розв’язання проблем у взаємодії між командами експлуатації та розробки. DevOps поєднує розробку (Development) і операції (Operations) з метою автоматизації та оптимізації процесів. Інтеграція неперервної розробки (CI) та неперервної доставки (CD) дозволяє автоматизувати тестування і розгортання програмного забезпечення, зменшуючи час від ідеї до виробництва.

Свою популярність в методології розробки програмного забезпечення набирає мікросервісна архітектура. Мікросервіси стають популярним через свій підхід до структуризації програмного забезпечення при якому система складається з набору слабко пов’язаних сервісів і де кожен мікросервіс виконує одну бізнес-функцію. Розбиваючи додаток на невеликі, незалежні компоненти дозволяє підвищити гнучкість і масштабованість системи.

Секція №3. Інформаційні технології загального призначення

Ще одним основним трендом в методології розробки програмного забезпечення є контейнеризація і технологія Docker. Контейнеризація – це підхід до розробки програмного забезпечення, при якому додаток і його конфігурація (абстрактні файли маніфесту розгортання) упаковуються разом в образ контейнера. Тобто контейнер – це стандартна одиниця програмного забезпечення, в якій упакований додаток з усіма необхідними для його роботи залежностями – кодом програми, середовищем запуску, системними інструментами, бібліотеками і налаштуваннями. Одна з переваг технології Docker — це незалежність платформи. Тобто ви можете описати запуск програми, працюючи на Windows, а потім без проблем запустити на MacOS. Це дає можливість дуже швидко переносити і налаштовувати програму на різних серверах.

Штучний інтелект, безумовно, не оминув і індустрію розробки програмного забезпечення: нещодавнє опитування GitLab показало, що майже чверть організацій вже використовують штучний інтелект (ШІ) для покращення процесів розробки програмного забезпечення. Більше того, понад дві третини опитаних організацій планують впроваджувати ШІ в майбутньому. Ці результати чітко вказують на те, що штучний інтелект і надалі залишатиметься невід’ємною частиною розробки програмного забезпечення. Найпоширеніші випадки використання ШІ в розробці програмного забезпечення – це природномовні чат-боти в документації, автоматизована генерація тестів, узагальнення змін у коді, моделі машинного навчання для відстеження експериментів, підказки при перегляді коду та узагальнення коментарів до випуску.

Підбиваючи підсумки, у сучасному світі розробка програмного забезпечення постійно змінюється, реагуючи на нові технології та потреби користувачів. Деякі з основних трендів в методології розробки програмного забезпечення включають DevOps та CI/CD для автоматизації процесів, мікросервісну архітектуру для підвищення гнучкості, контейнеризацію для забезпечення переносимості коду, агільні методології для швидкості та відкритості до змін, і використання штучного інтелекту та машинного навчання для створення інтелектуальних додатків. Загалом, розробка програмного забезпечення стає більш інноваційною та орієнтованою на вимоги ринку, а нові технології та методології дозволяють розробникам створювати високоякісне та конкурентоздатне програмне забезпечення. Важливо залишатися відкритим до змін і навчатися новому, щоб успішно працювати в цьому швидкозмінному сегменті індустрії.

Список літератури

1. Все, що потрібно знати про методології розробки ПЗ: тренди, принципи та пастки для новачків. *JavaRush*. URL: <https://javarush.com/ua/groups/posts/uk.2443.vse-jsho-potrbno-znati-pro-metodolog-rozrobki-pz-trendi-principi-ta-pastki-dlja-novachkv> (дата звернення: 13.10.2023).
2. Штучний інтелект у розробці програмного забезпечення: опитування відзначає зростання прийняття та стурбованість. *TS2 SPACE*. URL: <https://ts2.space/uk/штучний-інтелект-у-розробці-програмн/> (дата звернення: 13.10.2023).
3. Хто такі DevOps фахівці? | nt.ua. *IT-курси, бізнес-тренінги, сертифікація* | nt.ua. URL: <https://nt.ua/blog/devops-engineers> (дата звернення: 13.10.2023).
4. DSpace KNU: Автоматизація розгортання WEB-додатків з використанням контейнерів Docker. *DSpace KNU: Головна сторінка*. URL: <http://ds.knu.edu.ua/jspui/handle/123456789/4462> (дата звернення: 13.10.2023).