

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Таврійський державний агротехнологічний університет**  
**імені Дмитра Моторного**  
**Факультет енергетики і комп'ютерних технологій**

ЗАТВЕРДЖУЮ

Зав. каф. "Комп'ютерні науки"

доц. \_\_\_\_\_ Сергій ШАРОВ

"17" \_\_\_\_\_ лютого \_\_\_\_\_ 2025 р.

**Пояснювальна записка**  
**до кваліфікаційної роботи здобувача СВО Магістр**  
(ступінь вищої освіти)

на тему: «Штучний інтелект в UI/UX-дизайні: автоматизація створення  
користувацьких інтерфейсів»

**52/4КНД.11260547.000000ПЗ**

Виконала: здобувач вищої освіти 2 курсу,  
групи 21МБКН

спеціальності 122 Комп'ютерні науки

за ОПП Комп'ютерні науки

(шифр і назва спеціальності та ОПП)

\_\_\_\_\_  
(підпис) Соф'я ВЕЛИЧКО  
прізвище й ініціали

Керівник, проф. \_\_\_\_\_  
(підпис) Сергій ТЕРЕЩУК  
прізвище й ініціали

Нормоконтроль, ст. викл. \_\_\_\_\_  
(підпис) Ольга ЗІНОВ'ЄВА  
прізвище й ініціали

Рецензент, проф. \_\_\_\_\_  
(підпис) Галина ТКАЧУК  
прізвище й ініціали

Запоріжжя – 2025 рік

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Таврійський державний агротехнологічний університет імені Дмитра Моторного

Факультет: Енергетики і комп'ютерних технологій

Кафедра: Комп'ютерні науки

Ступінь вищої освіти: Магістр

Спеціальність: 122 Комп'ютерні науки

ОПП: Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри КН

к.пед.н., доц. Сергій ШАРОВ

“04” жовтня 2024 року

**З А В Д А Н Н Я**

**НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ**

Величко Соф'я Дмитрівна

(прізвище, ім'я, по-батькові)

1. Тема кваліфікаційної роботи Штучний інтелект в UI/UX-дизайні:  
автоматизація створення користувацьких інтерфейсів

керівник проєкту Терещук Сергій Іванович, д.пед.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом університету від «23» вересня 2024 року № 442-С

2. Строк подання здобувачем вищої освіти роботи 14 лютого 2025

3. Вихідні дані до роботи: дані обстеження об'єкту, статистичні дані, технічне  
завдання на дипломне проєктування, матеріали виробничих практик, нормативні  
документи, науково-технічна література, електронні ресурси та ін.

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

1. Аналіз предметної області.

2. Специфікація вимог до комп'ютерної системи.

3. Опис прийнятих проєктних рішень.

4. Аналіз дослідної експлуатації та можливих застосувань.

Висновки.

Список літератури.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслеників)

Діаграма використання, діаграма класів, діаграма IDEF0, діаграма станів, діаграма послідовностей, інтерфейс головної сторінки

Презентація – 10 слайдів

1. Інформаційний

2. Діаграма використання

3. Діаграма класів

4. Методологія IDEF0 системи та її декомпозиція

5. Діаграма станів комп'ютерної системи

- 6. Діаграма послідовностей комп'ютерної системи
- 7. Діаграма компонентів
- 8. Інтерфейс головної сторінки

6. Консультанти розділів кваліфікаційної роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата   |                  |
|--------|-------------------------------------------|----------------|------------------|
|        |                                           | завдання видав | завдання прийняв |
|        |                                           |                |                  |

7. Дата видачі завдання 01 жовтня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів кваліфікаційної роботи           | Строк виконання етапів кваліфікаційної роботи | Примітка |
|-------|-----------------------------------------------|-----------------------------------------------|----------|
| 1     | Аналіз предметної області                     | 12.09.2024                                    | виконано |
| 2     | Специфікація вимог до комп'ютерної системи    | 06.10.2024                                    | виконано |
| 3     | Проектування інтелектуальної системи          | 16.10.2024                                    | виконано |
| 4     | Обґрунтування інструментальних засобів        | 05.11.2024                                    | виконано |
| 5     | Розробка інтелектуальної системи              | 10.12.2024                                    | виконано |
| 6     | Тестування та налагодження програмного засобу | 20.01.2025                                    | виконано |
| 7     | Підпис керівником роботи                      | 14.02.2025                                    | виконано |
| 8     | Підпис завідувачем кафедри                    | 17.02.2025                                    | виконано |

Здобувач вищої освіти

( підпис )

Соф'я ВЕЛИЧКО

(власне ім'я та прізвище)

Керівник кваліфікаційної роботи

( підпис )

Сергій ТЕРЕЩУК

(власне ім'я та прізвище)

## РЕФЕРАТ

Кваліфікаційна робота: 68 с., 10 рис., 4 табл., 34 посилань.

Актуальність: Штучний інтелект все активніше інтегрується у сферу UI/UX-дизайну, змінюючи підходи до створення інтерфейсів. Автоматизація дизайну дозволяє значно скоротити час розробки, оптимізувати процеси та підвищити ефективність взаємодії користувачів із цифровими продуктами. Використання ШІ в UI/UX відкриває можливості для автоматичного створення макетів, аналізу UX-поведінки та персоналізації дизайну відповідно до індивідуальних потреб користувачів.

У сучасних умовах, коли швидкість та гнучкість розробки є критичними факторами, впровадження ШІ в дизайн дозволяє дизайнерам фокусуватися на творчих аспектах, делегуючи рутинні завдання алгоритмам. Важливим є дослідження ефективності таких технологій, їхніх переваг та обмежень, а також створення інструментів, які зроблять автоматизацію UI/UX доступною для широкого кола розробників.

Об'єкт дослідження: процес автоматизації UI/UX-дизайну.

Предмет дослідження: використання штучного інтелекту для автоматизації процесу створення користувацьких інтерфейсів.

Мета роботи: розробити систему, що використовує штучний інтелект для автоматизації UI/UX-дизайну, з можливістю інтеграції в Figma та подальшого експорту в Visual Studio Code.

Основні задачі:

Проаналізувати сучасні підходи до автоматизації UI/UX-дизайну.

Дослідити можливості використання штучного інтелекту у сфері UI/UX.

Розглянути існуючі інструменти та сервіси, що застосовують ШІ у дизайні.

Розробити концепцію інтеграції ШІ у процес автоматизованого UI/UX-дизайну.

Скласти технічне завдання на створення системи автоматизації UI/UX-дизайну.

Реалізувати прототип системи, що автоматично генерує UI-макети у Figma.

Виконати тестування та оцінити ефективність використання ШІ у дизайні.

Практичне значення дослідження: розроблена система дозволить автоматизувати процеси UI/UX-дизайну, що дасть змогу значно скоротити час на створення макетів, покращити UX-аналіз та спростити інтеграцію UI-елементів у робочий процес. Це стане корисним як для дизайнерів, так і для розробників, які прагнуть оптимізувати процес створення інтерфейсів за допомогою штучного інтелекту.

Ключові слова: ШТУЧНИЙ ІНТЕЛЕКТ, UI/UX-ДИЗАЙН, АВТОМАТИЗАЦІЯ, ШІ-ГЕНЕРАЦІЯ ІНТЕРФЕЙСІВ, ПРОТОТИПУВАННЯ, Figma, Visual Studio Code, UX-АНАЛІЗ, КОРИСТУВАЦЬКИЙ ДОСВІД.

## SUMMARY

Qualification work: 68 p., 10 figures, 4 tables, 34 references.

Relevance: Artificial intelligence is increasingly being integrated into UI/UX design, transforming the approaches to interface creation. Design automation significantly reduces development time, optimizes processes, and enhances user interaction with digital products. The use of IIII in UI/UX enables automated mockup generation, UX behavior analysis, and design personalization according to individual user needs.

In modern conditions, where speed and flexibility of development are critical factors, IIII integration into design allows designers to focus on creative aspects while delegating routine tasks to algorithms. It is important to study the effectiveness of such technologies, their advantages and limitations, and to develop tools that make UI/UX automation accessible to a wide range of developers.

The object of research: UI/UX design automation process.  
The subject of research: the use of artificial intelligence to automate the creation of user interfaces.

The purpose of the work: development of a system that utilizes artificial intelligence for UI/UX design automation, with integration into Figma and further export to Visual Studio Code.

IIIIIn tasks:

To analyze modern approaches to UI/UX design automation.

To examine the possibilities of artificial intelligence in UI/UX design.

To analyze existing tools and services that apply IIII in design.

To develop a concept for integrating IIII into the automated UI/UX design process.

To create a technical task for the development of a UI/UX design automation system.

To implement a prototype system that automatically generates UI mockups in Figma.

To test the developed system and evaluate the effectiveness of IIII in UI/UX design.

Practical significance of the study:

The developed system will automate UI/UX design processes, significantly reducing the time required for mockup creation, improving UX analysis, and simplifying UI element integration into the workflow. This will be useful for both designers and developers looking to optimize the interface creation process with artificial intelligence.

Keywords: ARTIFICIAL INTELLIGENCE, UI/UX DESIGN, AUTOMATION, IIII-GENERATED INTERFACES, PROTOTYPING, Figma, Visual Studio Code, UX ANALYSIS, USER EXPERIENCE.

## ЗМІСТ

|                                                                                                                 |    |
|-----------------------------------------------------------------------------------------------------------------|----|
| ВСТУП.....                                                                                                      | 9  |
| РОЗДІЛ I. ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ ТА АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ                                                     | 11 |
| 1.1. Поняття штучного інтелекту та його роль у сучасному дизайні .....                                          | 11 |
| 1.2. Вплив ШІ на користувацький досвід та ресурси .....                                                         | 12 |
| 1.3. Технічне завдання на розробку системи автоматизації UI/UX-дизайну<br>за допомогою штучного інтелекту ..... | 15 |
| РОЗДІЛ II. АВТОМАТИЗАЦІЯ ПРОЦЕСУ ПРОЕКТУВАННЯ ІНТЕРФЕЙСІВ ЗА<br>ДОПОМОГОЮ ШІ.....                               | 17 |
| 2.1. Основні поняття UI та UX-дизайну.....                                                                      | 17 |
| 2.2 Концептуальна модель використання системи автоматизованого<br>UI/UX-дизайну.....                            | 20 |
| 2.3. Специфікація функціональних та нефункціональних вимог .....                                                | 25 |
| РОЗДІЛ 3. ВИБІР ТА ОБҐРУНТУВАННЯ ТЕХНОЛОГІЙ РОЗРОБКИ.....                                                       | 27 |
| 3.1. Вплив технологій на створення інтерфейсів та використання<br>машинного навчання у дизайні .....            | 27 |
| 3.2. Алгоритми та методи машинного навчання для генерації дизайну .....                                         | 32 |
| 3.3. Інструменти на основі ШІ для створення макетів, прототипів та аналізу<br>користувацької поведінки .....    | 34 |
| 3.4 Загальний огляд технологічного стеку .....                                                                  | 37 |
| РОЗДІЛ IV. ДОСЛІДНА ЕКСПЛУАТАЦІЯ ТА ТЕСТУВАННЯ .....                                                            | 46 |
| 4.1 Опис програмної реалізації .....                                                                            | 46 |
| 4.2 Тестування.....                                                                                             | 53 |
| 4.3 Інтеграція з іншими системами .....                                                                         | 56 |
| 4.4 Розгортання ІС.....                                                                                         | 57 |
| ВИСНОВКИ.....                                                                                                   | 58 |
| СПИСОК ЛІТЕРАТУРИ.....                                                                                          | 60 |
| Додаток А.....                                                                                                  | 64 |
| Додаток Б .....                                                                                                 | 68 |



## ВСТУП

Актуальність. Штучний інтелект (ШІ) стрімко розвивається та займає ключову роль у сучасному цифровому середовищі. Його інтеграція у сферу UI/UX-дизайну значно змінює підходи до створення користувацьких інтерфейсів, забезпечуючи автоматизацію рутинних процесів та підвищення ефективності розробки. Впровадження ШІ дозволяє дизайнерам зосередитися на творчих аспектах проектування, зменшуючи витрати часу та ресурсів. Разом із цим актуальним є аналіз можливостей та обмежень використання ШІ у UI/UX-дизайні, що робить цю тему надзвичайно важливою для сучасної індустрії.

ШІ в UI/UX-дизайні відкриває нові горизонти автоматизації завдяки застосуванню алгоритмів машинного навчання, аналізу поведінки користувачів та генерації адаптивних інтерфейсів. Завдяки використанню інноваційних технологій стає можливим швидке створення якісних дизайн-рішень, оптимізація взаємодії користувачів із цифровими продуктами та персоналізація інтерфейсів відповідно до їхніх потреб. Водночас важливо дослідити виклики, пов'язані з інтеграцією ШІ у дизайн-процеси, включаючи питання етики, алгоритмічної упередженості та залежності від даних.

Об'єкт дослідження: процеси розробки користувацьких інтерфейсів у сфері UI/UX-дизайну.

Предмет дослідження: використання технологій штучного інтелекту для автоматизації створення користувацьких інтерфейсів.

Мета дослідження: аналіз можливостей застосування штучного інтелекту в UI/UX-дизайні та оцінка його впливу на автоматизацію створення користувацьких інтерфейсів.

Основні завдання дослідження:

1. Дослідити сучасні технології штучного інтелекту, що використовуються у сфері UI/UX-дизайну.

2. Проаналізувати практичні кейси застосування ШІ для автоматизації інтерфейсів.
3. Оцінити переваги та виклики інтеграції ШІ у дизайн-процеси.
4. Розробити концепцію автоматизованого створення користувацьких інтерфейсів із використанням ШІ.
5. Провести тестування інтеграції ШІ у UI/UX-дизайн та оцінити його ефективність.

Методи дослідження:

- аналітичний метод для вивчення існуючих технологій та практик у сфері UI/UX-дизайну;
- порівняльний аналіз для оцінки ефективності використання ШІ в автоматизації інтерфейсів;
- метод моделювання для вивчення можливостей інтеграції ШІ в дизайн-процеси;
- експериментальний метод для оцінки ефективності автоматизації створення інтерфейсів.

Практичне значення дослідження полягає у розробці рекомендацій щодо використання штучного інтелекту у UI/UX-дизайні, що може бути корисним для дизайнерів, розробників та компаній, які прагнуть оптимізувати процеси створення користувацьких інтерфейсів.

Структура і обсяг роботи. Дипломна робота складається з переліку умовних скорочень, вступу, трьох розділів, висновків, списку використаних джерел та додатків. Перший розділ містить теоретичний огляд сучасних тенденцій і можливостей застосування штучного інтелекту в UI/UX-дизайні. У другому розділі розглядаються практичні кейси використання ШІ у створенні інтерфейсів, включаючи аналіз реальних проєктів та інструментів. Третій розділ присвячений методиці інтеграції ШІ у процеси UI/UX-дизайну, а також оцінці ефективності його застосування. У висновках підбиваються підсумки дослідження, окреслюються перспективи розвитку та можливі напрямки подальших досліджень.

Загальний обсяг роботи складає 68 сторінок.

## РОЗДІЛ I. ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ ТА АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ

### 1.1. Поняття штучного інтелекту та його роль у сучасному дизайні

Штучний інтелект (ШІ) – це галузь інформатики, яка займається розробкою систем, здатних виконувати завдання, що зазвичай вимагають людського інтелекту, зокрема розпізнавання образів, аналіз великих обсягів даних, генерування текстів і зображень. В останнє десятиліття ШІ став невід’ємною частиною сучасного цифрового світу, включаючи сфери освіти, науки, бізнесу та дизайну. Одним із найбільш перспективних напрямів розвитку є використання генеративних моделей штучного інтелекту, які здатні створювати візуальний контент, а також автоматизувати процеси розробки користувацьких інтерфейсів.

Сьогодні штучний інтелект не лише змінює підхід до навчання та роботи дизайнерів, а й створює нові можливості для швидшої та ефективнішої розробки UI/UX-дизайну. Використання алгоритмів машинного навчання дозволяє значно спростити процес створення візуального контенту, автоматизувати рутинні задачі та оптимізувати взаємодію між користувачами та цифровими системами. Одним із таких інструментів є Galileo AI, який використовує глибокі нейронні мережі для аналізу текстових запитів і генерування відповідних інтерфейсів [11, с. 37].

Значною перевагою застосування ШІ у дизайні є можливість створення персоналізованих інтерфейсів, які адаптуються до потреб користувача. Galileo AI аналізує поведінку користувачів і відповідно налаштовує елементи інтерфейсу, забезпечуючи зручний досвід роботи з додатком. Завдяки цьому дизайнер може зосередитися на творчій складовій, залишаючи технічну реалізацію штучному інтелекту. Це значно спрощує процес створення UI та дозволяє досягати високої якості в коротші терміни.

Ще одним важливим аспектом використання ШІ у дизайні є інтеграція із популярними інструментами розробки. Наприклад, Galileo AI підтримує експорт макетів до Figma, що дозволяє дизайнерам легко змінювати згенеровані інтерфейси

та доопрацьовувати їх за потреби. Важливим фактором є також можливість генерування дизайну на основі текстового опису, що допомагає прискорити створення прототипів і впровадження нових ідей у реальні проєкти.

Штучний інтелект також активно використовується у сфері освіти, зокрема для навчання майбутніх дизайнерів. Генеративні моделі, такі як DALL-E 2, Stable Diffusion та Midjourney, широко застосовуються у навчальних закладах для вивчення основ візуального дизайну. Ці системи дозволяють студентам експериментувати з різними стилями, отримувати натхнення та працювати з новітніми технологіями, які вже зараз активно застосовуються у комерційних проєктах.

Попри значні переваги, використання штучного інтелекту у дизайні також несе певні виклики. Наприклад, алгоритми можуть сприймати візуальні патерни по-різному, що може призводити до нестандартних або помилкових рішень. Водночас розвиток нейронних мереж і глибоке навчання дають змогу постійно вдосконалювати такі системи та адаптувати їх до нових потреб. Тому в сучасному дизайні ШІ не є просто інструментом, а перетворюється на потужного партнера, що допомагає розширити межі креативності та автоматизувати складні процеси [11, с. 38]. Таким чином, роль штучного інтелекту у дизайні продовжує зростати, відкриваючи перед спеціалістами нові можливості для створення ефективних та адаптивних інтерфейсів. Його застосування дозволяє не лише пришвидшити розробку дизайну, а й покращити якість взаємодії між користувачами та цифровими продуктами. Завдяки платформам, таким як Galileo AI, майбутнє UI/UX-дизайну виглядає ще більш технологічним, гнучким та інноваційним.

## 1.2. Вплив ШІ на користувацький досвід та ресурси

Застосування штучного інтелекту у UI/UX-дизайні суттєво змінює сприйняття користувачів, оскільки алгоритми можуть адаптувати контент та

інтерфейс відповідно до індивідуальних потреб кожного відвідувача. Завдяки персоналізації, що базується на аналізі великих обсягів даних, веб-сайти та додатки стають більш зручними, інтуїтивними та ефективними у взаємодії з користувачем. Наприклад, алгоритми рекомендацій Netflix чи Amazon аналізують поведінку користувачів, історію переглядів, уподобання та навіть час активності, що дозволяє платформам пропонувати контент, який максимально відповідає інтересам конкретної особи. Це не лише підвищує рівень задоволення користувачів, а й сприяє утриманню аудиторії та збільшенню тривалості взаємодії з платформою. Інструменти на основі ШІ також здатні враховувати емоційний стан користувачів, що відкриває нові горизонти для підвищення якості обслуговування. Наприклад, чат-боти, що аналізують тон повідомлень, здатні адаптувати відповіді, підлаштовуючи їх під емоційний контекст користувача. Це створює відчуття емпатії та залученості, що особливо важливо у сфері обслуговування клієнтів, підтримки користувачів та навіть онлайн-освіти. Такі чат-боти можуть не лише відповідати на запитання, а й навчати, допомагати приймати рішення та навіть змінювати тональність комунікації залежно від настрою користувача.

ШІ значно скорочує час, необхідний для виконання рутинних завдань, таких як розробка макетів чи тестування, що дозволяє дизайнерам зосередитися на більш креативних аспектах роботи. Наприклад, інструменти на зразок Adobe Sensei можуть автоматично створювати декілька варіантів дизайну за лічені хвилини, що раніше потребувало кількох годин роботи дизайнерів. Подібні технології не лише економлять час, а й допомагають уникнути людських помилок, оскільки алгоритми аналізують великі масиви даних і пропонують найбільш ефективні рішення. Крім того, зменшуються витрати на дослідження та тестування. Алгоритми машинного навчання дозволяють швидко аналізувати поведінку користувачів, визначати слабкі місця в інтерфейсі та пропонувати їхнє вирішення. Це особливо важливо для стартапів та малих компаній, де ресурси обмежені, і можливість швидкого вдосконалення продукту без значних фінансових витрат є ключовою перевагою. ШІ також покращує процес тестування інтерфейсів, автоматизуючи аналіз

користувацьких сценаріїв та прогнозуючи можливі проблеми у взаємодії з продуктом. Тестування, яке раніше потребувало значної кількості часу та людських ресурсів, тепер може здійснюватися автоматично, що значно прискорює процес випуску продукту на ринок.

Штучний інтелект вже активно використовується у багатьох компаніях, що дозволяє значно покращити досвід користувачів. Наприклад, Spotify застосовує алгоритми ШІ для створення унікальних плейлистів, ґрунтуючись на активності користувачів, їхніх музичних вподобаннях та частоті прослуховування певних треків. Це підвищує лояльність клієнтів, оскільки вони отримують персоналізований досвід, який відповідає їхнім музичним вподобанням. Подібним чином Airbnb використовує алгоритми машинного навчання для оптимізації пошуку житла, що дозволяє користувачам швидше знаходити найбільш відповідні варіанти. Алгоритми аналізують запити користувачів, геолокаційні дані, історію бронювань та навіть відгуки, щоб пропонувати найкращі варіанти відповідно до їхніх вподобань. Це не лише економить час для користувачів, а й підвищує ефективність платформи. У сфері електронної комерції великі компанії, такі як Shopify, впроваджують ШІ-рішення для автоматизації маркетингових стратегій, персоналізованого підбору товарів та навіть автоматичного створення контенту для рекламних кампаній. Завдяки цьому магазини можуть швидше адаптуватися до поведінки споживачів і покращувати свої продажі.

Штучний інтелект суттєво покращує користувацький досвід, водночас оптимізуючи витрати часу та ресурсів. Він дозволяє автоматизувати персоналізацію контенту, покращувати взаємодію користувачів із продуктами та прогнозувати їхні потреби на основі аналізу даних. Однак його впровадження потребує ретельного аналізу отриманих даних, адаптації під конкретні бізнес-завдання та врахування етичних аспектів, пов'язаних із конфіденційністю користувацьких даних та алгоритмічною упередженістю. Поєднання традиційного підходу з використанням ШІ забезпечує найкращі результати у дизайні, дозволяючи поєднувати автоматизовані процеси з людським творчим підходом. Це

створює оптимальний баланс між ефективністю, інноваційністю та персоналізацією користувацького досвіду, що є ключовим фактором успіху цифрових продуктів у сучасному світі.

### 1.3. Технічне завдання на розробку системи автоматизації UI/UX-дизайну за допомогою штучного інтелекту

Будь-яка розробка програмного продукту на початковому етапі повинна передбачати створення технічного завдання (ТЗ). Цей документ регламентує функціональність системи та визначає вимоги до процесу розробки. Виконавець робіт має чітко зрозуміти сутність завдання, представити замовнику приблизний інтерфейс майбутньої системи, а замовник, у свою чергу, повинен підтвердити та засвідчити його відповідність очікуванням.

При написанні ТЗ ми орієнтувалися на загальні принципи розробки програмного забезпечення, враховуючи специфіку автоматизації UI/UX-дизайну.

Загальні визначення:

Назва проєкту: система автоматизації створення користувацьких інтерфейсів.

Розробник: студент спеціальності 122 «Комп'ютерні науки» Велічко Софія.

Інструменти розробки: Figma (для створення макетів), штучний інтелект (для аналізу і генерації дизайну), Visual Studio Code (для розробки інтерактивних елементів).

Призначення: автоматизоване створення UI/UX-дизайну з використанням ШІ для прискорення процесу розробки.

Основні етапи автоматизації UI/UX-дизайну:

Аналіз існуючих UI/UX-рішень та користувацьких патернів.

Використання ШІ для створення шаблонних елементів інтерфейсу.

Інтеграція генеративних моделей у Figma для швидкого прототипування.

Експорт готових UI-макетів у Visual Studio Code для подальшого використання.

Таким чином, система автоматизації UI/UX-дизайну спрямована на спрощення та прискорення процесу розробки інтерфейсів, використовуючи можливості штучного інтелекту для створення зручних, сучасних та адаптивних рішень. У цьому документі наведено повний перелік вимог до розробки та впровадження системи. Виконавець підготував і розробив цей документ, що іменується Технічне Завдання, який містить перелік вимог до виконуваних робіт.



## РОЗДІЛ II. АВТОМАТИЗАЦІЯ ПРОЦЕСУ ПРОЕКТУВАННЯ ІНТЕРФЕЙСІВ ЗА ДОПОМОГОЮ ШІ

### 2.1. Основні поняття UI та UX-дизайну

UI (User Interface) та UX (User Experience) є ключовими аспектами дизайну цифрових продуктів, які формують враження користувача від взаємодії з інтерфейсом і функціональністю продукту.

UX (User Experience) – це досвід користувача, отриманий під час взаємодії з продуктом. UX охоплює зручність використання, швидкість виконання завдань, а також емоційні аспекти, які формуються у користувача. Основна мета UX – створення продуктів, які задовольняють потреби користувачів та приносять позитивний досвід.

Ключові компоненти UI:

Кольорова схема: визначення основних і додаткових кольорів для інтерфейсу, які допомагають у навігації та створюють візуальну ієрархію.

Шрифти: підбір типографії, яка буде легко читатися і відповідатиме стилю продукту.

Іконки та кнопки: забезпечують зрозумілість і доступність інтерфейсу, що дозволяє користувачам швидко знайти потрібну інформацію або виконати дію.

Відступи та розміщення: визначення відстаней між елементами для створення естетично привабливого, структурованого інтерфейсу.

Дослідження користувача: аналіз потреб і поведінки користувачів для визначення їхніх цілей та викликів, які допоможуть сформуванню структури продукту.

Інформаційна архітектура: планування структури продукту, щоб забезпечити легкий доступ до інформації та інтуїтивну навігацію.

Проектування прототипів: створення чернеток і прототипів для тестування та покращення інтерфейсу ще на ранніх стадіях розробки.

Тестування з користувачами: перевірка інтерфейсу на зручність використання та виявлення можливих проблем у взаємодії користувачів із продуктом.

UI та UX працюють у тісному зв'язку. Візуальна складова інтерфейсу (UI) забезпечує зовнішню привабливість, тоді як UX зосереджується на функціональності та задоволенні потреб користувачів. Наприклад, гарний дизайн кнопок (UI) повинен супроводжуватися інтуїтивно зрозумілим розташуванням та зручною навігацією (UX). До ключових принципів UI/UX-дизайну відноситься простота та мінімалізм (уникнення зайвих елементів, які можуть заплутати користувача), послідовність (усі елементи інтерфейсу повинні відповідати єдиному стилю та логіці), доступність (продукти мають бути зручними для використання людьми з різними можливостями).

UI/UX-дизайн охоплює широкий спектр завдань, які включають дослідження користувачів, тестування інтерфейсів та аналіз даних. Наприклад, процес проектування починається зі збору вимог користувачів, створення концептуальних моделей і завершується тестуванням прототипів. Завдяки цьому процесу забезпечується висока якість продукту та зростає рівень задоволеності клієнтів.

Основні етапи розробки UI/UX:

- Збір даних про потреби, звички та вподобання користувачів.
- Візуалізація інтерфейсу для попереднього тестування.
- Перевірка прототипів на зручність та відповідність потребам користувачів.

Інструменти для UI/UX-дизайну:

Серед сучасних інструментів варто виділити Figma, Adobe XD, Sketch та Axure. Ці платформи дозволяють дизайнерам працювати над спільними проектами, швидко створювати прототипи та інтегрувати результати з іншими інструментами, такими як платформи аналітики або системи управління проектами. UI/UX-дизайн є важливою частиною процесу створення цифрових продуктів. Завдяки правильно організованій роботі над інтерфейсами та досвідом користувача, компанії можуть

досягати кращих результатів, забезпечуючи своїх клієнтів якісними та зручними рішеннями.

Штучний інтелект (ШІ) – це галузь комп'ютерної науки, що займається створенням систем, здатних виконувати завдання, які зазвичай потребують людського інтелекту. До таких завдань належать розпізнавання мови, зображень, аналіз даних, прогнозування, а також прийняття рішень. У сучасному дизайні ШІ стає важливим інструментом, що дозволяє автоматизувати рутинні процеси, підвищити ефективність та якість розробки продуктів.

ШІ знаходить застосування у дизайні через: генеративний дизайн, аналіз користувацької поведінки та прогнозування трендів. Генеративний дизайн дозволяє створювати нові варіанти дизайну на основі заданих параметрів. Аналіз користувацької поведінки визначає патерни використання, які допомагають оптимізувати інтерфейси. Прогнозування трендів здійснюється через аналіз великих обсягів даних для визначення майбутніх потреб користувачів.

Основні напрямки досліджень ШІ охоплюють:

Машинне навчання (Machine Learning), яке дозволяє алгоритмам "навчатися" з даних і самостійно вдосконалювати свої прогнози або рішення. Підходи машинного навчання включають підкріплювальне навчання, навчання з учителем та без учителя.

Нейронні мережі та глибинне навчання (Neural Networks and Deep Learning) – це підвиди машинного навчання, які моделюють роботу людського мозку для вирішення складних завдань. Глибинні нейронні мережі складаються з великої кількості шарів, що дозволяє їм розпізнавати складні шаблони та залежності.

Обробка природної мови (Natural Language Processing, NLP) забезпечує взаємодію між комп'ютерами та людьми за допомогою природної мови, що використовується у чат-ботах, системах перекладу тощо.

Комп'ютерний зір (Computer Vision) – здатність комп'ютерів розпізнавати та обробляти візуальну інформацію, що використовується у розпізнаванні обличчя, аналізі зображень, автономних транспортних засобах.

Роль ШІ у UI/UX-дизайні можна звести до декількох напрямків. По-перше, Штучний інтелект дозволяє автоматизувати створення макетів, оптимізувати кольорові схеми та шрифти. Це значно економить час дизайнера. По-друге, завдяки аналізу великих даних, системи ШІ можуть адаптувати інтерфейси до індивідуальних потреб користувачів. Далі, Інструменти ШІ аналізують користувацьку поведінку в режимі реального часу, визначаючи слабкі місця інтерфейсу та пропонуючи способи їх вирішення. Водночас, використання ШІ також має свої обмеження, зокрема залежність від якості вихідних даних та складність розробки алгоритмів. Однак потенціал цієї технології для розвитку дизайну є величезним. Завдання ШІ систем з часом стають все складнішими, а саме поняття інтелекту розширюється, охоплюючи не тільки здатність до обчислень, а й вміння вирішувати креативні завдання та адаптуватися до нових умов.

## 2.2 Концептуальна модель використання системи автоматизованого UI/UX-дизайну

Проектування системи UI/UX-дизайну з використанням штучного інтелекту (ШІ) передбачає автоматизацію створення інтерфейсів, що допомагає скоротити час розробки та підвищити ефективність дизайнерських рішень.

Перелік основних термінів та їх визначень, що використовуються в рамках цього проекту, подано у таблиці 2.1.

Таблиця 2.1 Глосарій

| Термін              | Опис                                                                                                     |
|---------------------|----------------------------------------------------------------------------------------------------------|
| UI (User Interface) | Користувацький інтерфейс – графічне або текстове середовище, через яке користувач взаємодіє із системою. |

|                           |                                                                                                                        |
|---------------------------|------------------------------------------------------------------------------------------------------------------------|
| UX (User Experience)      | Досвід користувача при взаємодії з цифровими продуктами, що включає зручність, доступність та ефективність роботи.     |
| Штучний інтелект (ШІ)     | Система, що використовує алгоритми машинного навчання для аналізу даних та прийняття рішень без прямого програмування. |
| Figma                     | Онлайн-інструмент для створення UI/UX-дизайну, що підтримує інтеграцію із ШІ для автоматизації процесів.               |
| Візуальне програмування   | Процес створення цифрових інтерфейсів без використання коду або з мінімальним втручанням розробників.                  |
| Адаптивний дизайн         | Підхід до розробки UI, що дозволяє інтерфейсу коригуватися під різні екрани та пристрої.                               |
| Прототипування            | Створення інтерактивних макетів інтерфейсу перед його повною реалізацією.                                              |
| Автоматизоване тестування | Використання алгоритмів для перевірки зручності та функціональності UI перед його впровадженням.                       |

Етап проектування інформаційних систем є необхідною складовою життєвого циклу програмного забезпечення. Його виконання впливає в кінцевому

рахунку на успішність програмного засобу серед кінцевих користувачів. Для проектування програмного забезпечення використовують структурну та об'єктно-орієнтовану методологією проектування [6, с. 226].

Концептуальна модель – це модель предметної області, яка складається з переліку взаємопов'язаних понять, що використовуються для опису цієї області, разом з властивостями й характеристиками, класифікацією цих понять, за типами, ситуаціями, ознаками в даній області і законами протікання процесів у ній. Концептуальна модель визначає структуру модельованої системи, властивості її елементів і причинно-наслідкові зв'язки, що властиві системі і суттєві для досягнення мети моделювання.

Основні елементи концептуальної моделі:

- умови функціонування об'єкта: визначають характер взаємодії між об'єктом і його оточенням, а також між елементами об'єкта;
- мета дослідження об'єкта: описує напрями покращення функціонування об'єкта;
- можливості керування об'єктом: включає визначення складу керованих змінних об'єкта;

Універсальним засобом об'єктно-орієнтованого моделювання є уніфікована мова UML (Unified Modeling Language), розробка якої розпочалася з середини 90-х років минулого століття на базі декількох об'єктноорієнтованих методів і нотацій опису інформаційних систем. Зараз мова UML є загальноприйнятим стандартом документування процесу створення інформаційних систем і програмного забезпечення. На сьогодні існує версія UML 2.0. До пакетів, які працюють з UML, можна віднести IBM Rational Rose, Telelogic TAU G2, Microsoft Visio, Enterprise Architect, Modelio та ін. [14, с. 12]. Мова UML активно застосовується для проектування реляційних БД за допомогою діаграм класів. У межах термінології UML діаграмою класів називається діаграма, на якій показаний набір класів, зв'язків між ними, а також може містити коментарі та обмеження [13, с. 7].

Діаграма варіантів використання є ключовим інструментом візуалізації функціональних вимог до системи (Рис. 2.1). Вона ілюструє, як різні типи користувачів (актори) взаємодіють із системою автоматизованого UI/UX-дизайну для виконання певних завдань. Для даної системи діаграма демонструє всі основні функції, такі як генерація інтерфейсу за допомогою ШІ, редагування UI-елементів у Figma, автоматизований аналіз UX-поведінки користувачів, а також експорт готових макетів у Visual Studio Code. Ця модель допомагає визначити ключові сценарії використання системи та забезпечує розуміння її функціональних можливостей і взаємодії між користувачем та ШІ-інструментами.

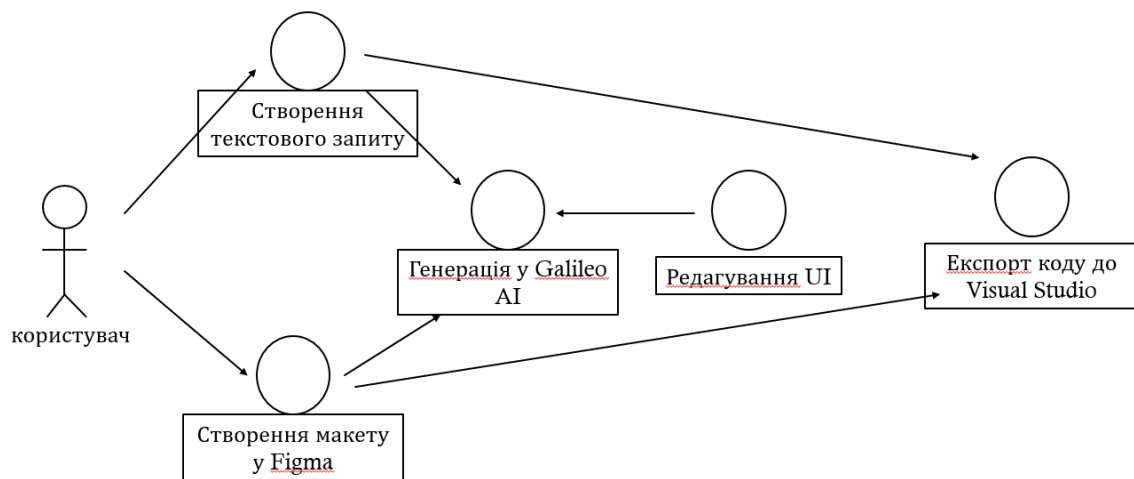


Рисунок 2.1 – Діаграма варіантів використання

Специфікація варіантів використання є текстовим описом сценаріїв взаємодії користувачів із системою, що доповнює діаграму варіантів використання. Вона деталізує кожен сценарій, включаючи його мету, основний потік подій, альтернативні потоки та виняткові ситуації. Цей документ слугує основою для проектування інтерфейсу користувача та логіки системи, а також допомагає команді розробників і зацікавленим сторонам краще зрозуміти функціональні вимоги системи.

Специфікація варіантів використання представлена в таблиці 2.4.

Таблиця 2.2. Розкадровка варіантів використання

| Варіант використання          | Контекст використання                               | Дійові особи                 | Передумова                            | Тригер                           |
|-------------------------------|-----------------------------------------------------|------------------------------|---------------------------------------|----------------------------------|
| Створення текстового запиту   | Користувач вводить текстовий запит для генерації UI | Основна: Користувач          | Користувач авторизований              | Введення текстового запиту       |
| Генерація UI у Galileo III    | III обробляє запит і генерує UI                     | Основна: Galileo III         | Запит створено, III доступний         | Передача запиту III              |
| Редагування UI                | Користувач змінює згенерований інтерфейс            | Основна: Користувач          | Згенерований макет доступний у Figma  | Натискання кнопки «Редагувати»   |
| Створення макету у Figma      | Користувач працює над UI у Figma                    | Основна: Користувач          | Доступ до Figma, макет згенеровано    | Відкриття Figma                  |
| Експорт коду до Visual Studio | Готовий UI експортується у Visual Studio Code       | Основна: Користувач, Система | Макет остаточно відредаговано у Figma | Натискання кнопки «Експортувати» |



### 2.3. Специфікація функціональних та нефункціональних вимог

Розділ функціональних та нефункціональних вимог є ключовим етапом у розробці системи автоматизації UI/UX-дизайну за допомогою штучного інтелекту. Функціональні вимоги визначають, які завдання має виконувати система для задоволення потреб користувачів, зокрема дизайнерів та розробників. Вони охоплюють основні функції автоматизації UI/UX-процесів, аналізу користувацьких патернів і генерації інтерфейсів на основі ШІ. Ці вимоги визначають взаємодію користувачів із системою та описують, як вона повинна обробляти запити, надавати інформацію та підтримувати різні сценарії використання (табл. 2.3).

Нефункціональні вимоги зосереджуються на характеристиках системи, які забезпечують її ефективну та стабільну роботу. Вони охоплюють аспекти, пов'язані з продуктивністю, доступністю, безпекою, гнучкістю та іншими критеріями, що визначають якість функціонування системи. Хоча ці вимоги не стосуються конкретних функцій, вони є критично важливими для забезпечення належної продуктивності та надійності системи в умовах реального використання (табл. 2.4).

Визначення чітких функціональних і нефункціональних вимог є необхідним для формування правильної архітектури системи, ефективного управління процесом розробки та тестування, а також для забезпечення якості й стабільності роботи після впровадження. У цьому розділі детально описані всі основні функції системи та ключові нефункціональні характеристики, які впливають на загальний користувацький досвід та ефективність роботи автоматизованої системи UI/UX-дизайну.

Таблиця 2.3 - Специфікація функціональних вимог.

|   | Вимога                                                | Пріоритет   | Складність | Користувач |
|---|-------------------------------------------------------|-------------|------------|------------|
| 1 | Генерація UI-елементів: використання ШІ для створення | Обов'язкове | Високе     | Дизайнер   |

|   |                                                                                    |             |         |           |
|---|------------------------------------------------------------------------------------|-------------|---------|-----------|
|   | базових макетів інтерфейсів у Figma                                                |             |         |           |
| 2 | Аналіз UX: автоматичне визначення оптимального розташування елементів              | Обов'язкове | Високе  | Дизайнер  |
| 3 | Автоматизоване стилізування: підбір кольорових схем, шрифтів та розмірів елементів | Обов'язкове | Середнє | Дизайнер  |
| 4 | Експорт дизайну в код: інтеграція макетів з Visual Studio Code для розробки        | Обов'язкове | Високе  | Розробник |
| 5 | Тестування макетів: перевірка відповідності UI принципам доступності               | Обов'язкове | Високе  | Дизайнер  |

Таблиця 2.4 - Специфікація нефункціональних вимог.

|   | Назва вимоги          | Характеристика                                                                    |
|---|-----------------------|-----------------------------------------------------------------------------------|
| 1 | Простота використання | Інтерфейс має бути зрозумілим для користувачів без глибоких знань у програмуванні |
| 2 | Гнучкість             | Можливість редагування ШІ-згенерованих макетів вручну у Figma                     |
| 3 | Швидкодія             | Генерація дизайну не повинна перевищувати 5 секунд                                |
| 4 | Сумісність            | Система повинна працювати у браузері та бути доступною для редагування в Figma    |
| 5 | Надійність            | Система повинна забезпечувати стабільну роботу без збоїв 99,9% часу               |

## РОЗДІЛ 3. ВИБІР ТА ОБҐРУНТУВАННЯ ТЕХНОЛОГІЙ РОЗРОБКИ

### 3.1. Вплив технологій на створення інтерфейсів та використання машинного навчання у дизайні

Розвиток сучасних технологій суттєво змінює підходи до створення користувацьких інтерфейсів (UI) та досвіду користувача (UX). Сьогодні дизайн перестав бути виключно творчим процесом, адже технології надають нові інструменти, які автоматизують рутинні задачі, оптимізують процес розробки та роблять його більш точним.

З появою нових технологій дизайн інтерфейсів зазнав значних змін. Традиційні підходи, які раніше базувалися на ручному проектуванні, тепер все частіше замінюються автоматизованими процесами. Наприклад, інструменти на основі штучного інтелекту (ШІ) дозволяють швидше розробляти макети та вдосконалювати існуючі інтерфейси.

Технології також впливають на адаптивність інтерфейсів. Використання гнучких макетів забезпечує зручність роботи користувачів на різних пристроях. Алгоритми машинного навчання аналізують дані про поведінку користувачів і дозволяють системам адаптувати дизайн у режимі реального часу, роблячи його більш релевантним.

Однією з ключових змін стала інтеграція штучного інтелекту (ШІ) та машинного навчання (МН) у процес створення інтерфейсів. Машинне навчання дозволяє аналізувати великі обсяги даних про поведінку користувачів, прогнозувати їхні дії та адаптувати дизайн до конкретних потреб. Наприклад, аналіз кліків, часу перебування на сторінці та взаємодій із елементами інтерфейсу дозволяє створювати персоналізовані інтерфейси, що покращують зручність та ефективність роботи. Дослідження показують, що ШІ трансформує поведінку користувачів, особливо в контексті їхньої взаємодії з цифровими продуктами та послугами. Машинне навчання (ML) є однією з основних технологій, що

використовуються у процесі автоматизації UI/UX-дизайну. Основні напрями його застосування включають:

1. Розпізнавання патернів. Алгоритми ML аналізують великі обсяги даних, виявляючи ключові тенденції та переваги користувачів. Це дозволяє дизайнерам створювати інтерфейси, що відповідають реальним потребам.
2. Генеративний дизайн. Використання ШІ для створення унікальних варіантів макетів на основі заданих параметрів, таких як стиль, кольори або розміри елементів.
3. Ітеративне вдосконалення. ШІ-інструменти пропонують варіанти оптимізації існуючих інтерфейсів на основі відгуків користувачів і тестування.

Технології також значно вплинули на процес прототипування та створення макетів. Інструменти на основі ШІ, такі як Adobe XD з елементами автоматизації або Figma з інтегрованими плагінами, дозволяють дизайнерам швидше генерувати макети, проводити А/В тестування та отримувати зворотний зв'язок. Використання алгоритмів оптимізації спрощує процес створення складних анімацій, адаптивних елементів та інтерактивних компонентів. Зокрема, алгоритми машинного навчання дають змогу оптимізувати призначений для користувача інтерфейс, пропонуючи більш інтуїтивні шляхи навігації сайтом.

Крім того, автоматизація за допомогою ШІ допомагає оптимізувати взаємодію з користувачами. Нейронні мережі можуть створювати рекомендаційні системи, які адаптують контент для кожного користувача, або ж формувати інтерфейси, які відповідають його потребам у реальному часі. Це відкриває нові можливості для динамічних і складних проєктів, таких як мобільні додатки, онлайн-платформи чи веб-сервіси. ШІ дає вебдизайнерам можливість надавати користувачам максимально персоналізований досвід, використовуючи алгоритми машинного навчання та дані.

Таким чином, вплив технологій на створення інтерфейсів полягає у переході від інтуїтивного до обґрунтованого підходу. Завдяки машинному навчанню дизайнери можуть не лише створювати естетично привабливі продукти, а й

забезпечувати їх функціональність і відповідність потребам користувачів, що значно підвищує якість кінцевого продукту. Як зазначено в дослідженні, інтеграція ШІ у дизайн інтерфейсів сприяє створенню адаптивних та персоналізованих інтерфейсів.

Розширення можливостей дизайнерів. Інтеграція ML у процес проектування не лише скорочує час на виконання рутинних завдань, але й розширює можливості дизайнерів. Наприклад, інструменти типу Figma, що інтегрують ML-моделі, дозволяють швидко створювати інтерактивні прототипи та тестувати їх на основі реальних даних. Використання технологій також має свої виклики, зокрема етичні питання, пов'язані з автоматизацією. Чи є ризик втрати творчого підходу у дизайні через надмірну залежність від алгоритмів? Це питання залишається відкритим для дискусій серед фахівців.

Застосування штучного інтелекту (ШІ) у дизайні відкриває значні можливості для оптимізації процесів, підвищення ефективності та покращення користувацького досвіду. Проте, поряд із перевагами, існують також певні виклики та обмеження, які необхідно враховувати при впровадженні автоматизованих рішень. Переваги використання штучного інтелекту у дизайні є численними, і вони суттєво впливають на ефективність роботи дизайнерів та якість кінцевого продукту. Однією з головних переваг є швидкість розробки, оскільки автоматизація процесів дозволяє значно скоротити час на створення прототипів, тестування та впровадження нових рішень. Алгоритми машинного навчання можуть швидко генерувати дизайн-концепти, використовуючи задані параметри.

Ще одним важливим аспектом є оптимізація UX/UI, що досягається завдяки аналізу користувацьких даних. Штучний інтелект здатний адаптувати інтерфейси відповідно до потреб різних сегментів аудиторії, покращуючи досвід взаємодії користувачів із продуктом. Також ШІ сприяє персоналізації контенту, що дозволяє створювати динамічні інтерфейси, які змінюються відповідно до вподобань користувачів. Наприклад, алгоритми можуть автоматично підбирати кольорові схеми чи змінювати розташування елементів на основі попередніх дій користувача.

Окрім цього, використання штучного інтелекту веде до зниження витрат, оскільки автоматизація рутинних процесів скорочує потребу у великій команді дизайнерів, що дозволяє компаніям ефективніше розподіляти ресурси.

ШІ також забезпечує масштабованість, що дає змогу швидко адаптувати дизайн до різних платформ і пристроїв без необхідності вручну змінювати кожен елемент. Це особливо важливо для компаній, які працюють на кількох ринках та пристосовують свої продукти для різних цифрових середовищ.

Важливим аспектом є ітеративне вдосконалення, оскільки штучний інтелект може аналізувати ефективність дизайну, проводити А/В тестування та вносити зміни на основі отриманих результатів, що сприяє постійному покращенню продукту.

Нарешті, використання ШІ сприяє доступності, оскільки технології штучного інтелекту можуть автоматично підлаштовувати інтерфейси для людей з обмеженими можливостями. Це підвищує загальний рівень інклюзивності цифрових продуктів і робить їх доступними для ширшої аудиторії.

Хоча штучний інтелект відкриває широкі можливості у сфері дизайну, його використання має певні обмеження та виклики, які слід враховувати при впровадженні технології. Однією з головних проблем є обмежена креативність. Попри здатність ШІ генерувати нові рішення, він все ж базується на аналізі наявних даних, що може обмежувати його здатність створювати дійсно оригінальні та інноваційні концепції. Внаслідок цього дизайнери можуть отримувати схожі шаблонні варіанти, які не завжди відповідають унікальним вимогам проєкту.

Ще одним важливим фактором є залежність від якісних даних. Ефективність штучного інтелекту напряму залежить від обсягу та точності вхідної інформації, на якій він навчається. Якщо дані є некоректними або неповними, ШІ може пропонувати нерелевантні рішення, що впливає на якість кінцевого продукту.

Додатково варто враховувати можливість алгоритмічної упередженості, оскільки ШІ може підсилювати певні стереотипи або допускати систематичні помилки. Це відбувається через особливості вихідних даних, які можуть містити

несвідомі упередження, що відображаються у згенерованих дизайнах та UX-рішеннях.

Висока вартість впровадження також є одним із викликів. Хоча штучний інтелект допомагає економити ресурси у довгостроковій перспективі, початкові інвестиції у його інтеграцію можуть бути значними. Це особливо стосується малих компаній та стартапів, для яких витрати на впровадження ШІ-рішень можуть бути критичними.

Ще одним обмеженням є низька гнучкість штучного інтелекту у творчих завданнях. ШІ добре працює у структурованих процесах, однак у ситуаціях, що потребують нестандартного підходу та художнього мислення, він поступається людині. Це може бути проблемою у проєктах, які вимагають нестандартного та креативного дизайну.

Крім того, складність інтеграції ШІ у вже існуючі дизайнерські процеси може стати додатковим бар'єром. Імплементация нових технологій вимагає певного рівня технічної підготовки та адаптації робочого процесу. Дизайнери та розробники повинні навчитися ефективно взаємодіяти з ШІ-інструментами, що потребує часу та ресурсів.

Останнім, але не менш важливим фактором, є ризики безпеки та конфіденційності. Автоматизовані рішення можуть обробляти великі обсяги персональних даних, що створює ризик витоку або неправомірного використання інформації. Тому важливо забезпечувати дотримання стандартів кібербезпеки при роботі з ШІ.

Таким чином, автоматизовані рішення на основі штучного інтелекту значно прискорюють процеси розробки, оптимізують UX/UI та дозволяють персоналізувати користувацький досвід. Водночас вони мають свої обмеження, пов'язані з креативністю, безпекою та вартістю впровадження. Щоб досягти найкращих результатів, використання ШІ у дизайні має бути збалансованим, а його впровадження має поєднувати технологічні можливості з людським творчим

підходом. Це дозволить забезпечити ефективність процесу розробки та зберегти гнучкість у створенні унікальних дизайнерських рішень.

### 3.2. Алгоритми та методи машинного навчання для генерації дизайну

Особливості алгоритмізації та машинного навчання (МН) у царині дизайну полягають у поєднанні обчислювальних підходів із творчими процесами, характерними для UI/UX-розробки. МН дозволяє системам виявляти закономірності у великих обсягах даних та пропонувати оптимальні рішення для проєктування інтерфейсів. Створення дизайну інтерфейсу вимагає врахування не лише візуальної складової, а й зручності, доступності та емоційного залучення користувачів.

Основні етапи та підходи у проєктуванні інтерфейсу:

1. Дослідження користувацьких потреб: Збір і аналіз даних про цільову аудиторію, її поведінкові патерни та цілі взаємодії з продуктом.
2. Формування концепції дизайну: Визначення стилю, кольорової гами, типографіки, ключових елементів і взаємодій.
3. Створення прототипів: Швидка візуалізація та перевірка гіпотез щодо зручності та привабливості інтерфейсу.
4. Тестування: Оцінка прототипів на реальних або тестових користувачах.
5. Ітеративне вдосконалення: Оптимізація дизайну на основі зворотного зв'язку.

Машинне навчання може інтегруватися на кожному з цих етапів, аналізуючи дані, пропонуючи варіанти дизайну та передбачаючи реакції користувачів.

Алгоритми генерації дизайну можуть мати наступні аспекти:

1. Генеративні змагальні мережі (GANs) Генеративні змагальні мережі складаються з двох моделей – генератора та дискримінатора, які навчаються



одночасно. Генератор створює нові зразки (наприклад, елементи дизайну), а дискримінатор оцінює їхню якість. Такий підхід дозволяє отримувати унікальні варіації інтерфейсів, не обмежені традиційними шаблонами.

2. Рекомендаційні системи. Застосовуються для аналізу поведінки користувачів та їхніх вподобань. Наприклад, ML-алгоритми можуть пропонувати кольори, шрифти чи компоненти, які будуть більш привабливими для цільової аудиторії. Рекомендаційні системи також використовують колаборативну фільтрацію, щоб передбачити, які елементи дизайну можуть бути ефективнішими.

3. Алгоритми пошуку та оптимізації. До цієї групи належать еволюційні алгоритми, жадібні алгоритми пошуку та метаевристики. Вони дають змогу автоматично перебирати можливі комбінації елементів інтерфейсу, визначаючи оптимальний варіант за критеріями зручності, естетики та швидкості взаємодії.

#### Методи навчання

1. Контрольоване (Supervised) навчання Використовується, коли є попередньо розмічені дані про дизайн чи поведінку користувачів. Алгоритми будуються на основі прикладів, що дозволяє передбачувати, які елементи дизайну найімовірніше задовольнять потреби користувачів.

2. Без учителя (Unsupervised) навчання Дозволяє системі самостійно виявляти патерни без попередньої розмітки. У дизайні це може бути кластеризація схожих за стилем чи структурою варіантів макетів. Завдяки цьому методу можна побачити приховані зв'язки між різними елементами, оптимізуючи процес розробки.

3. Підкріплювальне навчання (Reinforcement Learning) Застосовується у випадках, коли алгоритм повинен самостійно визначати кращі рішення шляхом проб і помилок. У контексті генерації дизайну це може стосуватися адаптації елементів інтерфейсу в реальному часі, реагуючи на взаємодію користувачів.

Виклики та перспективи, які виникають при використанні ШІ для створення дизайну, можуть мати наступні характерні риси. Хоча алгоритми та методи машинного навчання відкривають нові можливості для генерації дизайну, існують

і певні перешкоди. По-перше, необхідність великих обсягів даних, які не завжди доступні або можуть бути неповними. По-друге, ризик зниження креативності, якщо дизайнери надмірно покладаються на алгоритми. Однак постійний розвиток обчислювальної потужності та поява нових архітектур нейронних мереж дозволяють сподіватися на поглиблення синергії між людиною-дизайнером та машиною.

Таким чином, використання машинного навчання в автоматизації дизайну інтерфейсів стає дедалі більш актуальним завданням. Правильне поєднання алгоритмів та людського досвіду допомагає створювати не лише зручні, але й естетично привабливі та персоналізовані рішення.

#### Використання штучного інтелекту в процесі створення дизайну сайту

Штучний інтелект (ШІ) відкриває широкі можливості для автоматизації та оптимізації процесів створення веб-дизайну. Він значно прискорює розробку інтерфейсів, аналізує поведінку користувачів, автоматизує рутинні завдання та допомагає у генерації контенту. Використовуючи сучасні ШІ-інструменти, можна швидко створити наприклад сайт для заходу «Створення гри-платформера за 1,5 години».

### 3.3. Інструменти на основі ШІ для створення макетів, прототипів та аналізу користувацької поведінки

Розглянемо інструменти для створення прототипів та покажемо перевагу таких інструментів на основі штучного інтелекту.

Функціональні підсистеми відіграють ключову роль у забезпеченні автоматизації та інтеграції процесів створення UI/UX-дизайну. Вони дозволяють організовувати роботу з великими обсягами даних, керувати користувацькими взаємодіями та підтримувати гнучкість і адаптивність інтерфейсів. У межах UI/UX-

дизайну такі підсистеми об'єднують дизайнерські, аналітичні та технічні компоненти для забезпечення оптимальної продуктивності.

Інтеграція функціональних підсистем.

Функціональні підсистеми об'єднують кілька рівнів управління, дозволяючи автоматизувати процеси проектування, тестування та аналізу взаємодії користувачів. Основними компонентами таких підсистем є:

- аналітичний модуль – збирає та обробляє дані про користувацьку активність, допомагаючи у прийнятті рішень щодо покращення інтерфейсу;
- генеративний модуль – створює адаптивні макети на основі машинного навчання та аналізу користувацьких патернів;
- тестувальний модуль – виконує автоматизовані а/б тестування та збирає фідбек користувачів для ітеративного покращення дизайну.

Принципи побудови функціональних підсистем.

Функціональні підсистеми UI/UX-дизайну можуть будуватися за такими принципами:

1. Предметний принцип – побудова підсистем з урахуванням специфіки проєкту та потреб користувачів.
2. Функціональний принцип – модульний підхід до структурування системи, розділення функцій аналізу, створення та тестування дизайну.
3. Проблемний принцип – вирішення конкретних завдань, наприклад, покращення доступності чи персоналізації UI.

Інструменти для створення прототипів. Для реалізації функціональних підсистем у UI/UX-дизайні активно використовуються прототипувальні інструменти, які базуються на ШІ. Вони дозволяють автоматизувати частину творчого процесу, пропонуючи адаптивні рішення на основі зібраних даних.

Серед найпопулярніших інструментів:

- figma ШІ – забезпечує швидке створення інтерактивних прототипів, підтримує автоматизований аналіз их;

- adobe xd sensei – використовує алгоритми машинного навчання для адаптивного налаштування дизайну;
- uizard – платформа, що перетворює ескізи на цифрові прототипи за допомогою ші;

Використання штучного інтелекту у UI/UX-дизайні. Сучасні технології ШІ значно змінюють підхід до створення UI/UX-дизайну. Інструменти на базі ШІ можуть аналізувати поведінку користувачів, прогнозувати їхні потреби та автоматично адаптувати дизайн інтерфейсів під індивідуальні вимоги.

Деякі популярні ШІ-рішення для UI/UX-дизайну:

- chatgpt (openAI) – використовується для генерації текстових підказок, допомагає з оптимізацією контенту в інтерфейсах;
- runway ml – створює динамічні та адаптивні ui-елементи за допомогою ШІ;
- deep dream generator – використовується для створення унікальних візуальних стилів та графічного оформлення інтерфейсів;
- khroma – ШІ, який допомагає підібрати колірні палітри на основі трендових рішень;
- fontjoy – автоматизований генератор поєднання шрифтів, який використовує нейронні мережі;
- visualeyex – аналізує їх та прогнозує поведінку користувачів на основі heatmaps та фокус-групових даних.

Переваги використання ШІ в UI/UX-дизайні наступні:

- автоматизація рутинних завдань (зменшує потребу у ручному проектуванні, дозволяючи дизайнерам зосередитися на стратегічних аспектах);
- персоналізація інтерфейсів – ші аналізує поведінку користувачів і пропонує найкращі рішення для ui;
- прискорення розробки – швидке генерування варіантів інтерфейсу, що спрощує процес ітераційного тестування;
- оптимізація їх – ШІ може передбачати поведінку користувачів та пропонувати рішення для покращення юзабіліті;

- гнучкість і адаптивність – завдяки ШІ можна автоматично створювати різні версії дизайну під різні пристрої та користувачів;
- масштабованість – ШІ може бути легко інтегрований у великі проекти та дозволяє швидко адаптуватися до нових вимог;

#### Недоліки

1. Залежність від даних – для ефективної роботи ШІ потрібні великі обсяги якісних даних.
2. Обмеження креативності – ШІ може обмежувати варіативність дизайну, пропонуючи рішення, що базуються на шаблонах.
3. Вартість впровадження – інтеграція ШІ у робочі процеси потребує значних ресурсів та технічної підготовки.
4. Можливі алгоритмічні упередження – ШІ може підсилювати стереотипи або неправильно аналізувати деякі користувацькі групи.
5. Потреба у навчанні персоналу – необхідно володіти технічними знаннями для ефективного використання ШІ-інструментів.
6. Залежність від алгоритмів – автоматизовані рішення можуть не завжди відповідати специфічним вимогам користувачів або бізнесу.

Функціональні підсистеми UI/UX-дизайну у поєднанні з інструментами на базі ШІ дозволяють значно покращити якість користувацького досвіду, підвищити ефективність командної роботи та забезпечити швидку адаптацію інтерфейсів до нових умов ринку.

### 3.4 Загальний огляд технологічного стеку

ChatGPT (спочатку версія 3.0, потім 3,5 і відтепер платна версія 4.0) спеціально налаштований для взаємодії в режимі чату, тобто відповіді на користувацькі запитання та діалоговий контекст. Ця модель використовується для створення текстових відповідей на широкий спектр запитань, завдань та

комунікації з користувачами через інтерфейс обробки природної мови. ChatGPT належить до класу «великих мовних моделей» (Large Language Model), а тому більшою мірою орієнтований на створення текстів - есе, звітів, оповідань тощо. Є багато прикладів, коли ChatGPT дуже швидко та якісно впорався зі складними завданнями з написання текстів. Тому можна стверджувати *a priori*, що завдання з розв'язування задач з фізики та математики або розкриття фізичного змісту наукових понять, що вивчаються у шкільному курсі фізики, йому не під силу, навіть попри наявний успішний досвід в окремих випадках.

Однією з основних переваг використання ChatGPT є можливість швидкого створення текстових елементів, які можуть бути використані у різних частинах сайту. Це особливо корисно для:

- написання заголовків та описів сторінок – chatgpt допомагає створити привабливі заголовки, що відповідають принципам seo та підвищують залученість користувачів;
- створення тексту для кнопок, банерів та інших ui-елементів – ШІ може автоматично підбирати фрази, які спонукають користувача до дії;
- оптимізації seo-контенту – chatgpt здатен аналізувати ключові слова та генерувати структуровані тексти, які покращують індексацію сайту у пошукових системах.

Хоч ChatGPT-3,5 (безкоштовна версія) часто допускає помилки й неточності, не посилається на джерела та допускає інші помилки, серед освітянської спільноти почали превалювати панічні настрої щодо руйнування окремих методик і технологій навчання, які засновані на навчанні через читання та письмо.

Слід вказати на головну відмінність чат-ботів від інших моделей штучного інтелекту. Чат-боти, які використовують підходи машинного навчання замість зіставлення шаблонів витягують вміст із введених користувачем даних за допомогою обробки природної мови і позбавляються можливості навчатися з (PLN) розмов. Натомість вони враховують весь контекст діалогу, а не лише поточний хід і не вимагають попередньо визначеної відповіді для кожного

можливого введення користувача. Як правило їм потрібен великий навчальний набір слів для пошуку. Це може становити вирішальну складність, оскільки доступні набори даних можуть бути неадекватними. Саме тому для реалізації чат-ботів використовують штучні нейронні мережі (ANN).

Після появи ChatGPT кількість сервісів на основі штучного інтелекту починає швидко зростати, а також набирають популярності ті, що створені раніше (наприклад, Deep Dream Generator). У січні 2022 року, на ринку з'являються численні моделі штучного інтелекту, розроблені різними компаніями та дослідницькими групами. Ось лише деякі з них.

BERT (Bidirectional Encoder Representations from Transformers) - алгоритм глибокого навчання, що базується на технології обробки природньої мови (NLP) на основі нейронної мережі. Розроблений Google, BERT є моделлю, яка використовує контекстне уявлення слів в обидва напрямки. Основна ідея BERT полягає в тому, щоб навчити модель передбачати пропущені слова в реченні, використовуючи контекст навколо них. Це дозволяє BERT зрозуміти значення слів у контексті та враховувати взаємодію слів у реченні, що поліпшує якість розуміння природньої мови.

ELMo (Embeddings from Language Models) розроблений компанією Allen Institute for Artificial Intelligence, ELMo є моделлю, яка створює представлення слів, враховуючи їх контекст в реченні.

Transformer-XL - вдосконалена версія трансформаторної архітектури, яка розроблена для роботи з довгими залежностями в тексті. Transformer-XL розроблений компанією Google.

RoBERTa (Robustly optimized BERT approach) - оптимізована версія BERT, розроблена Facebook. Вона включає вдосконалені методи тренування та параметри для покращення продуктивності.

XLM-R (Cross-lingual Language Model - RoBERTa) ще одна велика мовна модель штучного інтелекту, розроблена Facebook. XLM-R спрямована на роботу з багатомовними завданнями та забезпечує ефективну підтримку кількох мов.

DALL-E штучний інтелект, який генерує зображення. Розроблений OpenAI, DALL-E генерує зображення з текстового опису за допомогою генеративної моделі.

У березні 2023 року Google випускає Bard - генеративний чат-бот штучного інтелекту, який є конкурентом ChatGPT. Bard призначений для участі в розмовах, відповідей на запитання та допомоги з різними завданнями. З липня 2023 року Bard запрацював в Україні.

Midjourney – незалежна дослідницька лабораторія та назва її власної програми штучного інтелекту, яка створює цифрові зображення з описів природною мовою (з 12 липня 2022 перебуває у відкритому бета-тестуванні).

Deep Dream Generator - генератор зображень на основі штучного інтелекту (створений Google ще у 2009 році). Створює сюрреалістичні зображення, тому має таку назву. Має три основні стилі: «Глибокий сон», «Глибокий стиль» та «Тонкий стиль». Генерує зображення різних стилів, різних місць, художніх стилів та епох.

Математичний пакет Wolfram Mathematica, який використовується в науці, технологіях та інженерії для вирішення складних обчислювальних завдань. Крім цього, ресурс Wolfram надає доступ до великої кількості знань та даних через свої вебресурси, такі як Wolfram Alpha та Wolfram Demonstrations Project, які працюють на основі штучного інтелекту.

Можливості Figma у сфері UI/UX-дизайну.

Figma є одним із найпопулярніших інструментів для UI/UX-дизайну, що забезпечує комплексний підхід до створення користувацьких інтерфейсів. Її основна перевага – це хмарна основа, яка дозволяє працювати з будь-якого пристрою без потреби встановлення додаткового програмного забезпечення. Завдяки цьому Figma стала універсальним інструментом для командної роботи над дизайном інтерфейсів, прототипуванням та розробкою дизайн-систем.

Однією з ключових функцій Figma є варіанти та властивості компонентів (Variants and Properties), які забезпечують створення гнучких та масштабованих дизайн-систем. Ця функція дозволяє дизайнерам створювати єдиний компонент з різними станами, що значно полегшує роботу над складними проектами [22, с. 47].



Figma також забезпечує інтерактивне прототипування, що дозволяє дизайнерам створювати динамічні макети, які можна протестувати без написання коду. Це значно спрощує перевірку UX-рішень та полегшує спілкування між дизайнерами та розробниками [23, с. 112].

Ще однією важливою функцією є автоматизація за допомогою плагінів та інтеграцій. Figma підтримує широкий вибір плагінів, що розширюють її можливості, наприклад, автоматичне генерування макетів, переклад текстів та інтеграція з розробницькими середовищами [8, с. 25]. Крім того, Figma використовується в освітньому процесі для навчання UI/UX-дизайну. Завдяки доступності та можливості командної роботи вона є ефективним інструментом для викладання основ візуального дизайну та розробки інтерфейсів [12, с. 321]. Ще однією ключовою перевагою Figma є адаптивність і підтримка командної роботи. Інструмент дозволяє одночасно редагувати файли кільком дизайнерам та відстежувати зміни в режимі реального часу, що робить його зручним для розподілених команд [7, с. 71].

Також важливо відзначити можливості анімації та інтерактивних ефектів. Figma дозволяє створювати прості анімаційні переходи між екранами, що допомагає покращити розуміння взаємодії користувача з інтерфейсом ще на етапі дизайну [2, с. 14]. Завдяки всім цим можливостям, Figma залишається одним із найбільш затребуваних інструментів у сфері UI/UX-дизайну, забезпечуючи ефективність, гнучкість та простоту роботи для дизайнерів, розробників і менеджерів проєктів.

Основна перевага Figma – це хмарна платформа, яка дозволяє спільно працювати над проєктом, редагувати макети та надавати коментарі у реальному часі. Завдяки інтуїтивному інтерфейсу, Figma підходить як для досвідчених дизайнерів, так і для початківців.

Основні можливості Figma:

- швидке створення макетів та прототипів – дизайнери можуть розробляти статичні макети або створювати інтерактивні прототипи з переходами між сторінками;
- автоматизація елементів дизайну за допомогою плагінів ШІ – figma підтримує численні плагіни на основі штучного інтелекту, які допомагають автоматизувати процес створення ui-елементів, наприклад, автоматично підбирати кольорові схеми або генерувати іконки;
- робота в команді та зручний обмін файлами – можливість одночасної роботи декількох користувачів, що робить figma ідеальним інструментом для колаборативної розробки;
- гнучка система компонентів – створення власних бібліотек ui-елементів для багаторазового використання у проєктах;

Galileo ШІ є передовою платформою, що використовує штучний інтелект для автоматичної генерації користувацьких інтерфейсів (UI) на основі текстових описів. Це дозволяє дизайнерам та розробникам швидко створювати прототипи інтерфейсів, значно скорочуючи час від ідеї до реалізації. Основна функціональність Galileo ШІ полягає в перетворенні текстових інструкцій у візуальні компоненти інтерфейсу, що забезпечує більш інтуїтивний та ефективний процес розробки.

Однією з ключових особливостей Galileo ШІ є його здатність інтегруватися з популярними інструментами дизайну, такими як Figma. Це дозволяє дизайнерам безперешкодно експортувати згенеровані інтерфейси для подальшого редагування та налаштування. Така інтеграція спрощує робочий процес та забезпечує узгодженість між різними етапами розробки продукту.

Крім того, платформа пропонує функціонал для оптимізації продуктивності та моніторингу ШІ-додатків у масштабах підприємства. Це включає можливості оцінки, ітерації, спостереження та захисту ШІ-рішень, що дозволяє командам швидко виявляти та виправляти помилки, такі як галюцинації моделей, забезпечуючи високу якість кінцевого продукту.[5, с. 15]

Використання Galileo III сприяє прискоренню процесу розробки, дозволяючи командам зосередитися на творчих аспектах дизайну, тоді як рутинні завдання автоматизуються. Це не тільки підвищує ефективність, але й забезпечує більш високий рівень персоналізації та адаптивності інтерфейсів, що відповідають потребам кінцевих користувачів. [21, с. 14]

У сучасному світі, де швидкість виведення продукту на ринок є критичною, інструменти на кшталт Galileo III стають незамінними для компаній, що прагнуть залишатися конкурентоспроможними та впроваджувати інновації у свої продукти та послуги.

Основні можливості Galileo III:

- генерація дизайну з нуля або на основі базового макета – інструмент аналізує введені дані та створює візуально привабливі компоненти інтерфейсу;
- налаштування стилів, кольорових схем і ui-компонентів – можливість автоматично підібрати стиль, палітру кольорів та візуальні елементи відповідно до концепції проєкту;
- автоматичне підлаштування дизайну під різні пристрої – адаптація створеного макета для мобільних, планшетних та десктопних версій без необхідності ручного налаштування;
- швидка ітерація – можливість швидко змінювати та тестувати варіанти дизайну, зберігаючи найкращі рішення для фінальної версії;
- експорт коду – отримання чистого html, css або навіть react-коду для подальшої інтеграції у проєкт;

Visual Studio Code (VS Code) є одним із найпопулярніших текстових редакторів для програмування, що поєднує в собі легкість, швидкодію та гнучкість. Його основними перевагами є підтримка багатьох мов програмування, інтеграція з Git, можливість встановлення розширень і зручний інтерфейс. Microsoft створила цей редактор з урахуванням потреб розробників, які працюють як над невеликими проєктами, так і над складними програмними системами. Відкритий вихідний код

і підтримка спільноти дозволяють покращувати функціональність редактора та адаптувати його під конкретні потреби розробки.

Одна з ключових особливостей VS Code – це потужна система автодоповнення IntelliSense, яка використовує аналіз коду для прогнозування та пропонування можливих варіантів завершення. Завдяки цьому програмісти можуть швидше писати код, зменшуючи кількість синтаксичних помилок. Крім того, редактор підтримує роботу з нативним дебагером, що дає змогу тестувати код без необхідності виходу з середовища розробки. Наприклад, при розробці програм на Python можна запускати сценарії та переглядати помилки безпосередньо у VS Code, що значно прискорює процес налагодження [3, с. 5].

Ще однією важливою функцією є підтримка вбудованого терміналу, що дозволяє працювати з командним рядком без переходу до окремого вікна. Це особливо зручно для користувачів, які активно працюють з командними утилітами, такими як `npm`, `pip` або `Docker`. Крім того, в VS Code реалізовано можливість віддаленої розробки, що дозволяє підключатися до серверів або контейнерів без необхідності встановлення додаткового ПЗ. Така функція є корисною для DevOps-інженерів та розробників, які працюють з хмарними середовищами.

Завдяки системі розширень у маркетплейсі VS Code можна значно розширювати функціональність редактора. Наприклад, встановлення розширень для роботи з `Docker`, `Kubernetes` або спеціальних мов програмування (наприклад, `Rust` або `Go`) додає необхідні інструменти безпосередньо в середовище розробки. Також можна додавати плагіни для покращення зручності роботи, такі як автозбереження файлів, підсвічування синтаксису для специфічних мов або інтеграція зі штучним інтелектом для генерації коду.

Крім можливостей для бекенд-розробки, VS Code широко використовується і в веб-розробці. Його потужні інструменти для роботи з `HTML`, `CSS`, `JavaScript` та `TypeScript` роблять цей редактор одним із найкращих виборів для фронтенд-розробників. Завдяки розширенням можна налаштувати автоматичне

форматування коду, автоперезавантаження сторінок у браузері та інтеграцію з різними фреймворками, такими як React, Angular або Vue.js [15, с. 37].

У сучасних умовах, коли ефективність розробки має велике значення, VS Code стає незамінним інструментом для програмістів різних напрямків. Він надає широкий набір функцій для покращення продуктивності та забезпечує зручний інтерфейс для роботи. Його відкритий код, активна підтримка з боку Microsoft та величезна спільнота користувачів гарантують стабільний розвиток цього редактора в майбутньому.

Основні переваги VS Code:

- зручне редагування html, css, javascript – платформа підтримує автодоповнення коду, перевірку синтаксису та інтеграцію з сучасними фреймворками;
- інтеграція з git та іншими інструментами розробки – це дозволяє ефективно керувати версіями файлів, здійснювати командну розробку та швидко вносити зміни;
- вбудовані засоби для дебагінгу та тестування – розробники можуть налагоджувати код безпосередньо у vs code, що значно прискорює виправлення помилок;
- підтримка розширень та плагінів – доступно безліч розширень для роботи з різними мовами програмування, ui-бібліотеками та фреймворками;
- інтерактивний термінал – дозволяє виконувати команди безпосередньо у середовищі редактора, що покращує швидкість розробки;

## Розділ IV. ДОСЛІДНА ЕКСПЛУАТАЦІЯ ТА ТЕСТУВАННЯ

### 4.1 Опис програмної реалізації

Інформаційна система 'Створення користувацького інтерфейсу за допомогою ІІ' реалізована як веб-застосунок, що працює у середовищі Figma та використовує штучний інтелект для автоматизації створення UI-дизайнів. Архітектура системи включає клієнтську частину, яка надає користувацький інтерфейс, і серверну частину, що обробляє запити ІІІ. Взаємодія між клієнтом та Figma API дозволяє автоматично створювати макети, редагувати їх та зберігати результати у Figma.

Клієнтська частина реалізована у вигляді веб-застосунку на основі React.js. Вона взаємодіє з Figma API, дозволяючи користувачам створювати та змінювати UI-макети безпосередньо у веб-інтерфейсі. Функціонал включає автоматичну генерацію UI-елементів на основі введених параметрів, редагування дизайнів та експорт готових макетів.

Система використовує алгоритми штучного інтелекту для автоматизації створення UI-елементів, враховуючи сучасні дизайн-тренди та потреби користувачів. ІІІ-модель аналізує введені параметри та пропонує оптимальні рішення на основі попередньо навченої вибірки. Також реалізовано інтеграцію з Figma API, що дозволяє користувачам імпортувати та експортувати макети між системами без втрати даних.

#### Процес створення сайту

##### 1. Використання ChatGPT для написання текстового контенту.

ChatGPT – це потужний інструмент генерації текстів, що допомагає створювати описові блоки, інформаційні повідомлення та навіть маркетингові тексти для сайту. Його застосування дозволяє значно скоротити час на написання контенту та підвищити його якість. Завдяки можливості аналізу трендів і наявності великої мовної бази, ІІІ може пропонувати релевантні та структуровані тексти для будь-якої тематики.

Крім того, ChatGPT (Рис. 2) може адаптувати стиль написання тексту відповідно до цільової аудиторії. Це означає, що, наприклад, для молодіжного заходу в університеті ШІ може генерувати більш неформальний стиль подачі інформації, а для корпоративного сайту – професійний та офіційний тон.

На цьому етапі ми або самостійно прописуємо зміст сайту, або використовуємо ChatGPT для його генерації. Додатково можна налаштувати ШІ так, щоб він пропонував кілька варіантів контенту, після чого обрати найкращий або комбінувати кілька варіантів для досягнення оптимального результату.

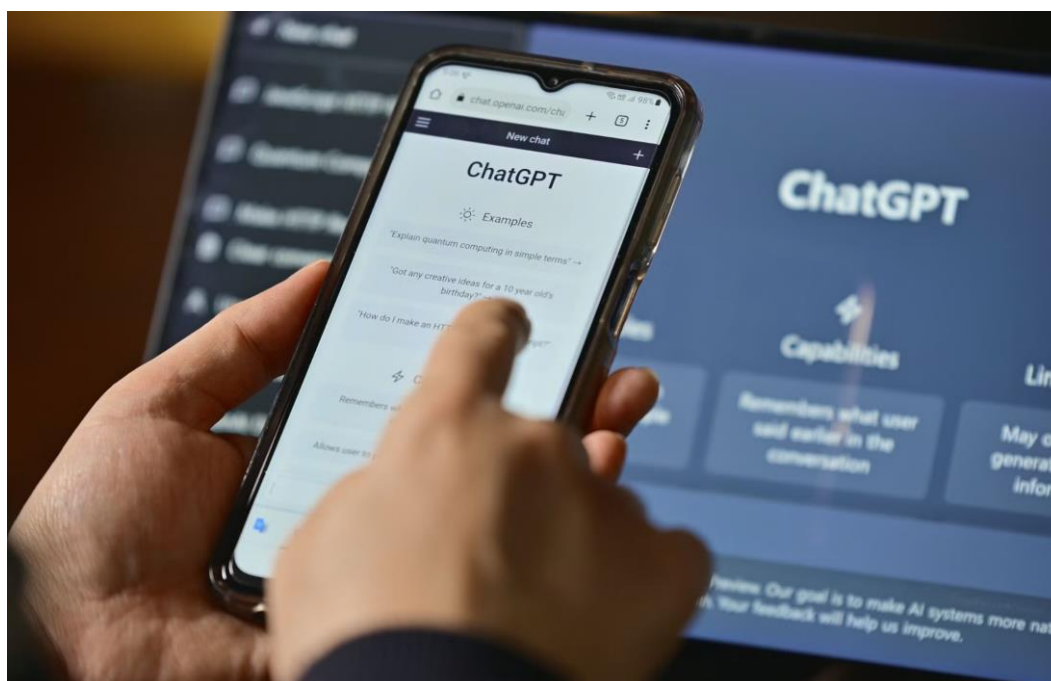


Рис. 4.1 ChatGPT

## 2. Ознайомлення з Figma та створення макета.

На цьому етапі ми створюємо приблизний макет сайту (Рис. 3.) за допомогою Figma, зокрема визначаємо розташування блоків, навігацію, стилістику (чорно-білий або кольоровий макет).

Макет може включати:

- головний банер з назвою заходу "створення гри-платформера за 1,5 години";
- інформаційний блок з коротким описом події;
- кнопки реєстрації або запису на майстер-клас;

- блок із розкладом заходу або списком доповідей;
- футер із контактною інформацією;

Для швидкого створення макета можна використовувати ШІ-плагіни, які автоматично розташують основні елементи сторінки, зроблять адаптацію для мобільних пристроїв та навіть запропонують оптимальні варіанти дизайну (Рис. 4.).

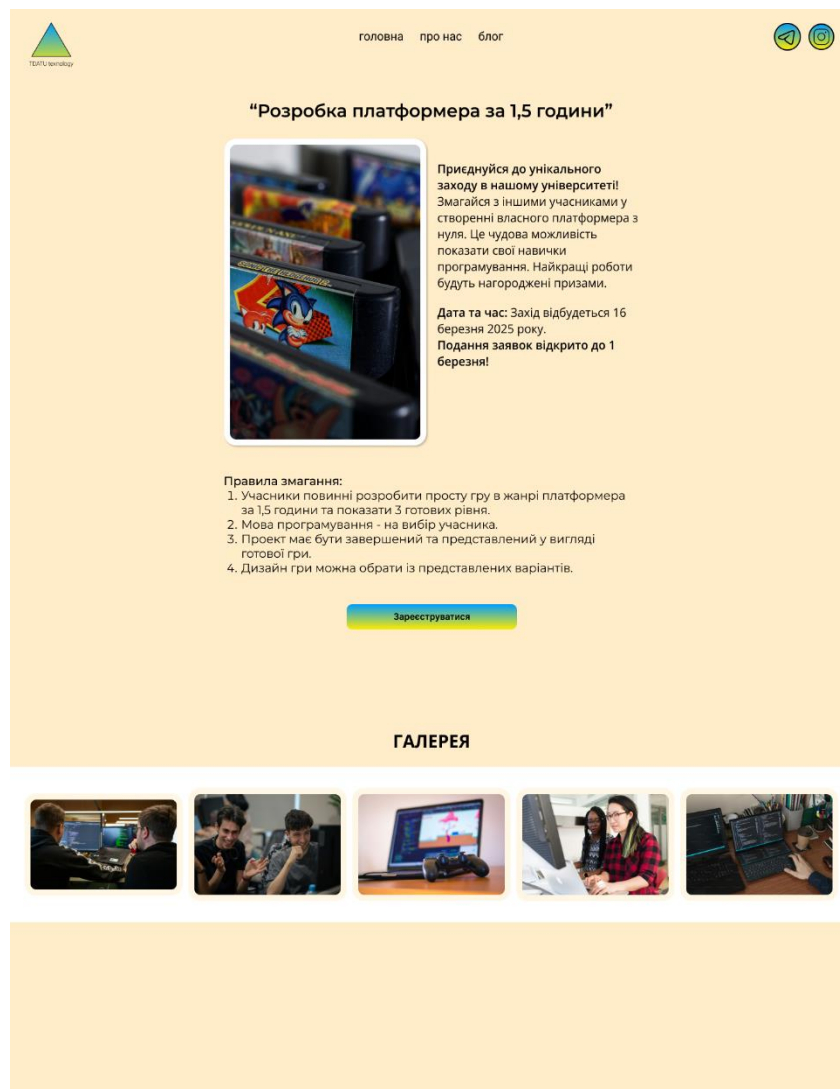


Рис. 4.2. Приблизний макет сайту



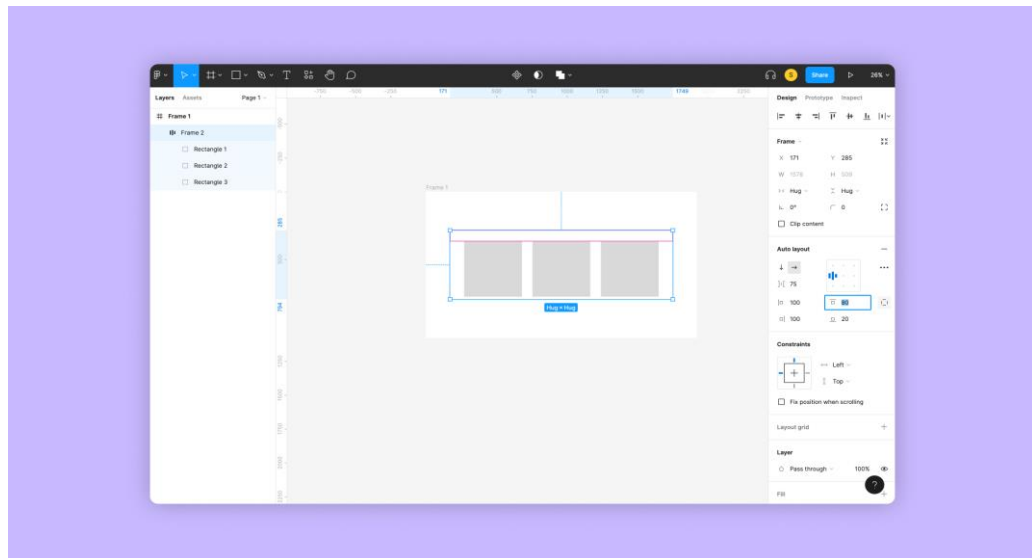


Рис. 4.3. Створення макету за допомогою ШІ-плагіну

### 3. Використання Galileo ШІ для автоматичної генерації дизайну

Galileo ШІ – це інструмент, що використовує штучний інтелект для створення готових дизайнів на основі опису та вхідних даних. Він дозволяє автоматизувати значну частину процесу UI/UX-дизайну, генеруючи професійні макети відповідно до заданих параметрів. Це значно спрощує роботу як для початківців, так і для досвідчених дизайнерів, адже дозволяє уникнути ручної розробки базових елементів інтерфейсу (Рис. 5.).

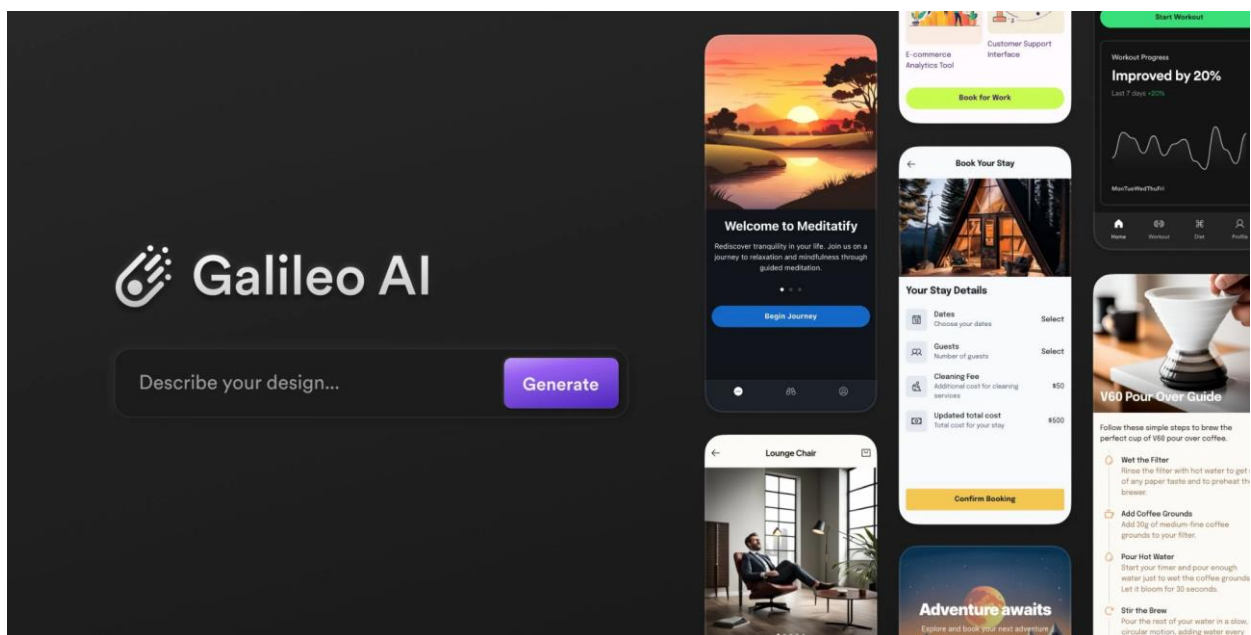


Рис. 4.4. Galileo ШІ

Процес роботи з Galileo III полягав у наступному:

1. Реєстрація та налаштування параметрів. Ми реєструємося на сайті Galileo III та визначаємо основні вимоги до дизайну, такі як структура макета, кольорова гама, типографія та основні інтерактивні елементи.

2. Завантаження макета або текстового опису Galileo III дозволяє працювати двома шляхами:

- завантаження створеного у figma або іншому редакторі макета як орієнтира;
- описування макета у текстовому форматі, де ші самостійно підбере оптимальні параметри;

3. Налаштування стилю та генерування дизайну. Ми обираємо стиль оформлення, встановлюємо бажані параметри (наприклад, мінімалізм, футуризм, корпоративний стиль тощо) та запускаємо генерацію (Рис. 6.).

4. Оцінка та коригування результатів. Після отримання першої версії макета можна внести правки: змінити шрифти, розташування елементів, підправити кольорові акценти. III дозволяє швидко тестувати різні варіанти та обирати найкращий.

5. Експорт у Figma та отримання коду. Після затвердження фінальної версії ми експортуємо макет у Figma для подальших правок або інтеграції. Крім того, Galileo III надає можливість отримати чистий код HTML, CSS чи навіть готові React-компоненти для використання у розробці(Рис. 7.).

Таким чином, використання Galileo III значно прискорює процес дизайну, дозволяє отримати якісний результат без необхідності вручну створювати всі елементи UI, а також забезпечує можливість експериментувати з різними варіантами макета. Крім того, інструмент дозволяє створювати як традиційні веб-сторінки, так і макети інтерфейсів для мобільних пристроїв, що забезпечує гнучкість у розробці та адаптацію дизайну до різних платформ.

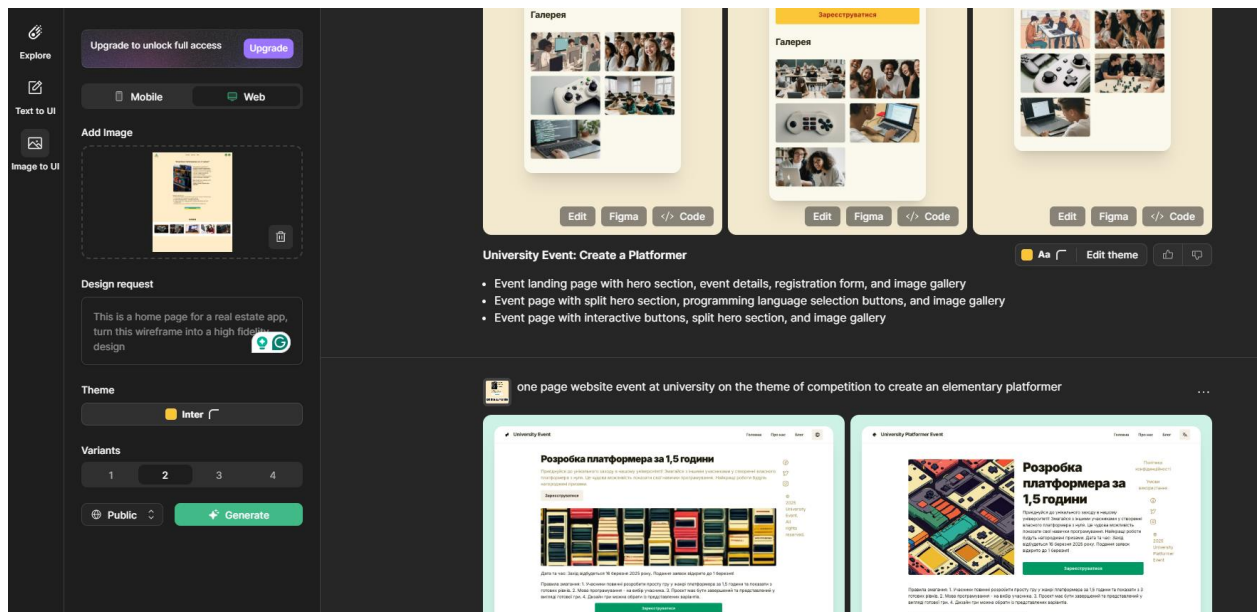


Рис. 4.5. Налаштування стилю та генерування дизайну

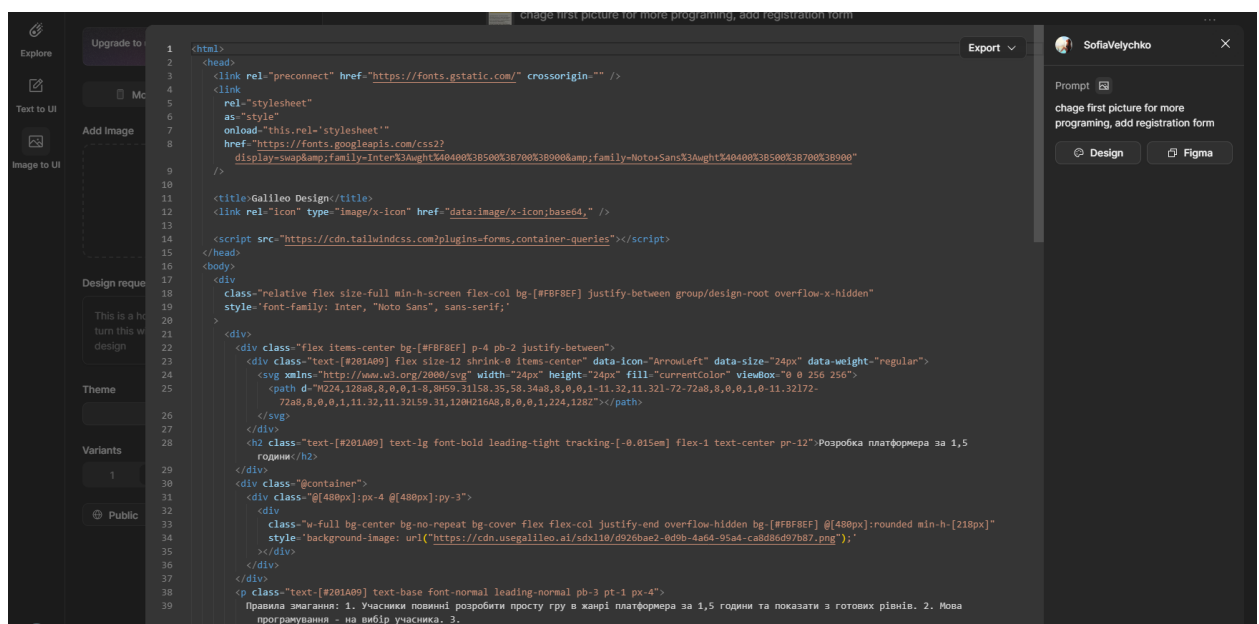


Рис. 4.6. Galileo III надає можливість отримати чистий код HTML

#### 4. Використання Visual Studio Code для розробки.

Visual Studio Code (VS Code) – це один із найпопулярніших текстових редакторів для веб-розробки, який використовується як початківцями, так і професійними розробниками (Рис. 8.).

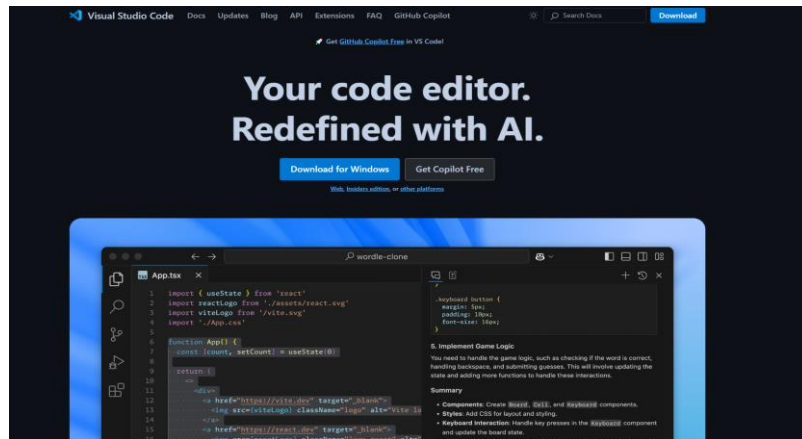


Рис. 4.7. Visual Studio Code (VS Code)

Його популярність пояснюється високою гнучкістю, продуктивністю та широким набором інструментів, що полегшують написання та тестування коду.

Процес роботи з Visual Studio Code.

1. Встановлення та налаштування VS Code. Спершу необхідно завантажити та встановити VS Code з офіційного сайту. Додатково можна налаштувати середовище під свої потреби, встановивши необхідні розширення, такі як Live Server, Prettier, ESLint та інші.

2. Створення нового проєкту. У VS Code відкриваємо нову папку, яка слугуватиме кореневим каталогом для нашого веб-сайту. Додаємо основні файли:

- index.html – головна HTML-сторінка сайту;
- style.css – файл для оформлення стилів;
- script.js – файл для інтерактивних функцій сайту;

3. Вставка коду, згенерованого Galileo III. Отриманий із Galileo III код можна безпосередньо вставити у відповідні файли та перевірити його коректність. Наприклад, HTML-код буде вставлений у index.html, стилі – у style.css, а логіка обробки взаємодій – у script.js. (Рис. 9.)

4. Редагування коду та налаштування адаптивності. Якщо згенерований III код потребує коригувань, у VS Code можна:

- Додати медіазапити (@media) для кращої адаптивності сторінки;
- Оптимізувати структуру HTML-коду для кращої SEO-індексації;



На завершальному етапі ми запускаємо перегляд сайту у браузері, перевіряємо коректність верстки, взаємодію елементів та адаптивність (Рис. 10.). Це важливий етап, оскільки саме тут відбувається остаточна перевірка працездатності проєкту перед його публікацією.

Основні аспекти тестування:

- перевірка коректного відображення сайту – тестування на різних браузерах (chrome, firefox, safari, edge) для впевненості у сумісності;
- адаптивність та коригування верстки – використання інструментів розробника у браузері (devtools) для перевірки роботи сайту на різних екранах і пристроях;
- функціональність інтерфейсу – перевірка коректної роботи кнопок, форм, анімацій та інтерактивних елементів;
- дебагінг – виявлення та виправлення можливих помилок у html, css або javascript-кодів;
- швидкість завантаження – аналіз продуктивності за допомогою google lighthouse або gtmetrix для виявлення можливих вузьких місць у швидкодії сайту;
- оптимізація коду – зменшення зайвих css-стилів, мінімізація javascript та використання сучасних технологій для покращення продуктивності;

Якщо під час тестування було виявлено проблеми, необхідно внести коригування в код у Visual Studio Code та повторно перевірити зміни. Після виправлення всіх помилок сайт готовий до розгортання та публікації (Рис. 11.).

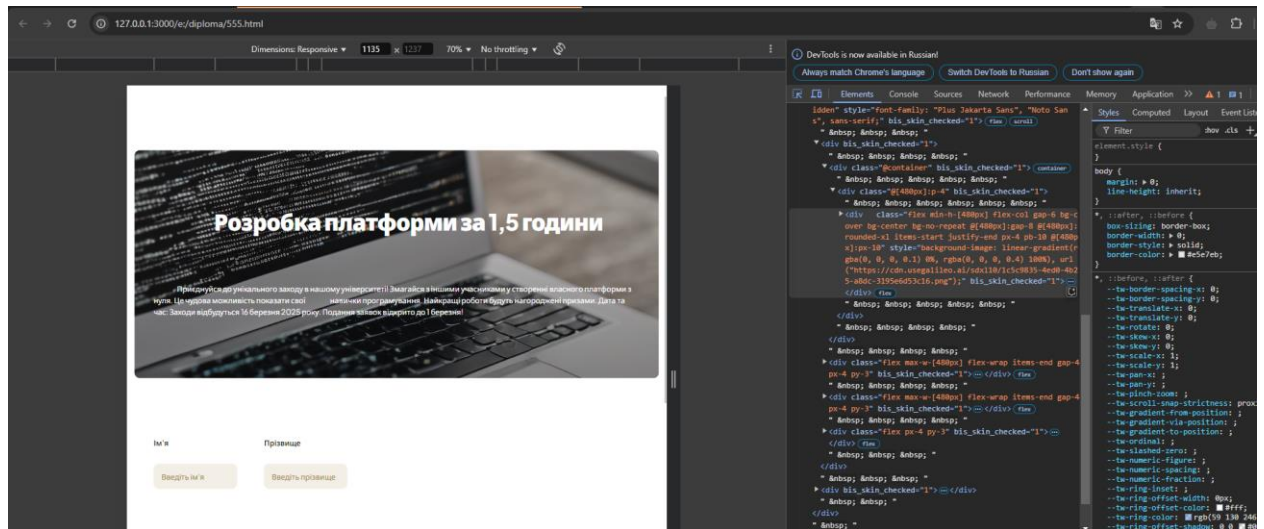


Рис. 4.9. Прегляд сайту у браузері

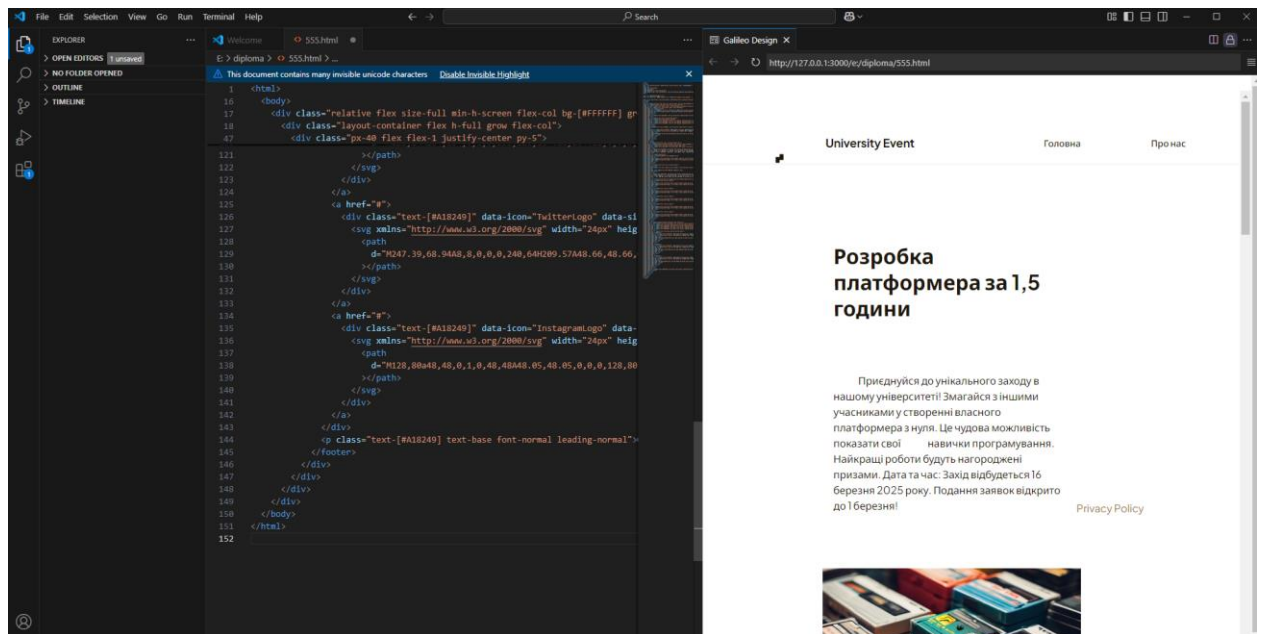


Рис. 4.10. Внесення коригування в код у Visual Studio Code

Підсумовуючи, відзначимо наступне. Штучний інтелект значно спрощує процес створення сайтів, дозволяючи автоматизувати дизайн, генерацію контенту та адаптацію інтерфейсу під потреби користувачів. Використання ШІ-інструментів у веб-розробці відкриває нові можливості для ефективного створення UI/UX-дизайну, оптимізації робочих процесів та економії часу.

Ключові переваги використання ШІ у веб-дизайні:



- автоматизація процесів – значне скорочення часу на створення макетів та генерацію контенту;
- покращена персоналізація – ШІ-інструменти дозволяють адаптувати дизайн під потреби користувачів;
- зручність для початківців – можливість швидко отримати якісний дизайн без глибоких знань кодування;
- швидке тестування та ітерація – можливість оперативно вносити зміни та покращувати сайт у реальному часі;

Інструменти, такі як ChatGPT, Figma, Galileo ШІ та Visual Studio Code, роблять процес розробки веб-сайтів доступнішим і ефективнішим. Завдяки цим технологіям навіть користувачі без глибоких технічних знань можуть створювати професійні та адаптивні інтерфейси, що відповідають сучасним стандартам веб-дизайну.

Перевірено стабільність основних функцій, включаючи генерацію UI, редагування у Figma, імпорт та експорт макетів. Тестування API проводилося через Postman, а для перевірки функціональності інтерфейсу використовувалися тестові сценарії у браузері.

На серверній частині протестовано основні модулі, зокрема механізм генерації UI та обробку запитів, за допомогою Jest. Клієнтську частину перевірено через React Testing Library, щоб оцінити коректність рендеру компонентів та взаємодію користувача з UI.

#### 4.3 Інтеграція з іншими системами

Система інтегрується з Figma API, що дозволяє користувачам імпортувати та експортувати макети у Figma безпосередньо з веб-застосунку. Також передбачена можливість експорту дизайнів у HTML/CSS, що робить систему зручною для подальшого використання у розробці веб-застосунків.



#### 4.4 Розгортання ІС

Система була протестована у локальному середовищі та працює у веб-браузері без необхідності складного серверного розгортання. Вихідний код зберігається у GitHub, що дозволяє зручно керувати версіями та оновленнями. Для демонстрації роботи користувач може експортувати UI-макети у Figma та переглядати результати безпосередньо в середовищі дизайну. Для підтвердження працездатності додаються скріншоти готових макетів та результати тестування.

Розроблена система дозволяє автоматизувати процес створення UI/UX-дизайну, спрощуючи роботу дизайнерів та оптимізуючи процес розробки користувацьких інтерфейсів.

## ВИСНОВКИ

В результаті проведеного дослідження було отримано важливі висновки щодо застосування штучного інтелекту в UI/UX-дизайні. Аналіз літературних джерел свідчить про те, що використання технологій ШІ є перспективним напрямом, який дозволяє суттєво оптимізувати процеси розробки інтерфейсів. Зокрема, автоматизація рутинних завдань за допомогою інтелектуальних алгоритмів сприяє скороченню часу на створення макетів, підвищенню якості розробки та мінімізації ймовірності помилок у дизайні. Крім того, інтеграція таких технологій забезпечує кінцевим користувачам більш інтуїтивно зрозумілий, адаптивний і зручний досвід взаємодії.

Практична частина дослідження продемонструвала реальні переваги використання ШІ у процесі розробки веб-дизайну. У ході реалізації проєкту було створено односторінковий сайт з розробки елементарного платформи, де активно застосовувалися сучасні інструменти штучного інтелекту. Використання плагінів та сервісів, таких як Figma ШІ та Uizard, дозволило значно спростити процес проєктування, автоматизувати частину рутинних процесів і водночас забезпечити високий рівень персоналізації контенту відповідно до потреб користувачів сайту. Впровадження таких технологій допомогло скоротити час розробки та покращити якість фінального продукту, що підтвердило ефективність інтеграції ШІ у дизайнерські процеси.

Отримані результати підтверджують, що впровадження штучного інтелекту у UI/UX-дизайн є важливим кроком до підвищення ефективності роботи дизайнерів та розробників. ШІ не лише автоматизує процеси, а й зберігає креативність та контроль над кінцевим продуктом. Це дозволяє спеціалістам зосередитись на стратегічних аспектах проєктування, таких як функціональність, унікальність і відповідність сучасним трендам. Використання інтелектуальних алгоритмів у розробці інтерфейсів відкриває нові можливості для створення

інноваційних, адаптивних та ефективних цифрових продуктів, що покращують взаємодію користувачів із технологіями.

Таким чином, мета дипломної роботи, яка полягала у розробці інтелектуальної системи для автоматизації UI/UX-дизайну з використанням штучного інтелекту, була досягнута, а всі поставлені у ході виконання дипломної роботи завдання виконані.

## СПИСОК ЛІТЕРАТУРИ

1. Айотунде О.О., Джаміль Д.І., Кавус Н. Вплив штучного інтелекту на вивчення іноземних мов за використанням систем управління навчанням: систематичний огляд літератури. 2023. *ITLT*. Т. 95, Вип. 3. С. 215–228.
2. Болотіна В. В., Вакалюк Т. А. Використання відкритого сервісу для проектування ui/ux figma у підготовці бакалаврів галузі інформаційних технологій. *Цифрова трансформація науково-освітніх середовищ в умовах воєнного стану* : збірник матеріалів. Звітна наукова конференція Інституту цифровізації освіти НАПН України (23 лютого 2024 р., м. Київ). С. 13–16.
3. Вастилкенко В.М., Іваншин В.В., Карпенко М.І. Роль та актуальні можливості застосування штучного інтелекту при написанні коду в середовищі Visual Studio Code. *Міжнародна науково-практична конференція «Дизайн у просторі новітніх технологій»*. С. 5–6.
4. Величко С.Д. Штучний інтелект в UI/UX-дизайні: впровадження та користь використання нейронних мереж. 2024. С. 12–15.
5. Дмитрієв Д. А. Мобільний додаток з динамічно генерованим інтерфейсом на основі нейронних мереж: магістерська дисертація. Київ, 2024. 94 с.
6. Зінов'єва О. Г. Використання CASE-засобів для проектування інформаційних систем. *Українські студії в європейському контексті*. 2023. № 7. С. 220–227.
7. Колесникова Т.А., Дзугань К.О. Дизайн-система як елемент оптимізації робочого процесу у Figma. URL: <https://openarchive.nure.ua/server/api/core/bitstreams/aa627d2a-8a9f-42d9-b702-bac77104b983/content>.
8. Ларіна Т., Шипова М. Figma у сфері дизайну UI/UX: аспекти використання. URL: <https://openarchive.nure.ua/server/api/core/bitstreams/ea1c8c25-4943-4ad4-8d95-06abdc076e0a/content>.
9. Подласов С. О. та Матвійчук О. В. Застосування chatgpt у навчанні фізики студентів бакалавра в технічному ВНЗ. 2023. *ITLT*. Т. 97, Вип. 5. С. 149–166.

- 10.Співаковський С.А., Омельчук В.В., Кобець Н.В., Валько Д.С., Мальчикова. Інституційна політика щодо штучного інтелекту в університетському навчанні, викладанні та дослідженнях. 2023. *ITLT*. Т. 97, Вип. 5. С. 181–202.
- 11.Терещук С., Слободянюк О. Штучний інтелект як драйвер інновацій в освіті. Природничо-наукова освіта: розробка та впровадження інноваційних проєктів, програм, методик та технологій. 2023. С. 36–40.
- 12.Хачкарджі І.О., Ратушняк Т.В. Figma: ключ до інтерактивного дизайну програмних продуктів. Цифрова трансформація фінансової системи України та країн v-4 в умовах євроінтеграції. збірник тез IV міжнародної науково-практичної інтернет-конференції (Дубляни, 16 травня 2024 р.). Частина II. Дубляни: ЛНУП, 2024. С. 321–322.
- 13.Шаров С. В., Скрипка С. О. Використання мови UML для інфологічного моделювання реляційної бази даних. *Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення*. 2014. № 7. С. 5–9.
- 14.Шаров С. В., Шамардак О. А. Концептуальне моделювання інформаційної системи за допомогою мови UML. *Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення*. 2014. № 7. С. 10–14.
- 15.Шевченко, В. Застосування Visual Studio Code у програмуванні: бакалаврська робота. Дніпро, 2024. 65 с.
- 16.Щавінський Ю.В., Мужанова Т.М., Якименко Ю.М., Запорожченко М.М. Застосування штучного інтелекту для вдосконалення ситуаційної підготовки фахівців з кібербезпеки. *ITLT*. 2023. Т. 97. Вип. 5. С. 215–226.
- 17.Цідило І. М., Сендра Е. Штучний інтелект як методична інновація у підготовці майбутніх дизайнерів: інструменти проміжного». 2023. *ITLT*. Т. 97, Вип. 5, С. 203–214
- 18.Цідило І.М. Нечіткість та невизначеність: опис, вимірювання і способи вирішення в моделюванні педагогічних явищ. *ITLT*. 2023. Т. 31, Вип. 5. С. 20–25.

19. Adamopoulou E. Chatbots: History, technology, and applications. *Machine Learning with Applications*. 2020. Vol. 2. 100006
20. Brown P. III-Enhanced Prototyping Tools. *Autodesk*. URL: <https://www.autodesk.com/solutions/generative-design-III-software>.
21. Eccher A., L'Imperio V., Pantanowitz L. Galileo III: Revolutionizing User Interface Design with Neural Networks. Springer, 2024, p. 112.
22. Fabio S. Figma: Variants and Properties. In *Designing and Prototyping Interfaces with Figma - Second Edition: Elevate your design craft with UX/UI principles and create interactive prototypes*.
23. Figma for UI/UX Designers. Springer. URL: <https://link.springer.com/book/10.1007/979-8-8688-0324-6>.
24. Garcia P. Optimizing UX with III. *Adobe Help Center*. URL: <https://helpx.adobe.com/indesign/using/generative-III-faq-indesign.html>.
25. Johnson M. Manual vs Automated Design Processes. *Formlabs*. URL: <https://formlabs.com/blog/generative-design/>.
26. Johnson M. The Need for Interdisciplinary Skills in Design. *Interaction Design Foundation*. URL: <https://www.interaction-design.org/literature/topics/generative-III>.
27. Lee J., Kim S. III-Driven Prototyping Tools. *Autodesk*. URL: <https://www.autodesk.com/solutions/generative-design-III-software>.
28. Microsoft. Visual Studio Code: Code Editing. *Redefined*. 2024. URL: <https://code.visualstudio.com/>.
29. Smith A. Responsibility and Accountability in III Decisions. *OECD*. URL: <https://www.oecd.org/en/topics/sub-issues/artificial-intelligence-and-education-and-skills.html>.
30. Smith T.J. Using UX Design Principles When Designing Context-Rich Learning Experiences. *Designing Context-Rich Learning by Extending Reality*. IGI Global, 2023. 75-88.

31. Swanson J. Perspectives on Artificial Intelligence in Education. KnowledgeWorks. URL: <https://knowledgeworks.org/resources/artificial-intelligence-education-conversation-steve-nordmark/>.
32. Taylor S. Generative Design with III. *Adobe Blog*. URL: <https://blog.adobe.com/en/publish/2024/04/23/the-future-pro-design-generative-III-taking-off-in-design>.
33. Tlili I. et al. What if the devil is my guardian angel: ChatGPT as a case study of using chatbots in education”. *Smart Learning Environments*. 2023. Vol. 10(15).
34. Visual Studio Code – Open Source Code Editor by Microsoft. *Wikipedia*. 2024. URL: [https://uk.wikipedia.org/wiki/Visual\\_Studio\\_Code](https://uk.wikipedia.org/wiki/Visual_Studio_Code).

## Тези у збірнику наукових праць

*Матеріали IV Всеукраїнської науково-практичної Інтернет-конференції  
«Сучасні комп'ютерні та інформаційні системи і технології»*

9. Яджак М.С. Паралельні алгоритми цифрової фільтрації даних // Кібернетика та системний аналіз. – 2023. – 59, № 1. – С. 46–56.
10. The list Top500 [Electronic resource]. – Available: <http://www.top500.org>.
11. Яджак М.С. Модифікація методу пірамід для розпаралелювання циклів: організація обмінів між гілками // Математичні методи та фізико-механічні поля. – 2000. – 43, № 4. – С. 68–72.

УДК 004.8:[004.032.26-049.7-021.475](045)

### ШТУЧНИЙ ІНТЕЛЕКТ В UI/UX-ДИЗАЙНІ: ВПРОВАДЖЕННЯ ТА КОРИСТЬ ВИКОРИСТАННЯ НЕЙРОННИХ МЕРЕЖ

Величко С.Д., здобувач вищої освіти *email: meleshko.sofia.work@gmail.com*  
Таврійський Державний Агротехнологічний Університет імені Дмитра  
Моторного

**Актуальність та постановка проблеми.** Стаття пропонує всебічний аналіз впливу штучного інтелекту на дизайн інтерфейсів і користувацький досвід, розкриваючи переваги, виклики та етичні аспекти його застосування. Результати дослідження акцентують увагу на ключовій ролі ШІ в сучасному дизайні інтерфейсів і наголошують на важливості подальших досліджень у цій галузі. Практичні рекомендації, представлені у статті, роблять її корисною для дизайнерів та програмістів, а також підкреслюють її значущість для розвитку інновацій у сфері UI/UX-дизайну. Метою роботи є аналіз впливу штучного інтелекту на процеси веб-дизайну та дизайну користувацьких інтерфейсів. У дослідженні розглядаються ключові аспекти функціонування ШІ, його можливості та роль у вдосконаленні дизайн-процесів. Завданням є визначення способів оптимального використання технологій ШІ для підвищення ефективності розробки інтерфейсів та поліпшення користувацького досвіду. Для створення наукової публікації було використано такі методи для дослідження: літературний огляд, порівняльний метод, аналіз існуючих продуктів та сервісів, аналіз трендів, синтез результатів, сумарний аналіз.

**Основні матеріали дослідження.** Сучасний світ вже неможливо уявити без технологій. Кожен день з'являється все більше можливостей котрі активно використовують у різних сферах життя людини. Штучний інтелект (ШІ) – одна з таких технологій. Наприклад, у будівництві ШІМ використовуються для енергоефективного планування, прогнозування витрат і моделювання з використанням 3D-друку та цифрових двійників Технології ШІ використовують для покращення цифрового світу і того, як ми з ним взаємодіємо.. У сфері графічного дизайну ШІМ можуть допомагати у створенні генеративних дизайнів та автоматизації творчих процесів, що значно підвищує ефективність роботи дизайнерів. Технології надзвичайно важливі у сучасному веб-дизайні, пропонуючи різноманітні інструменти та методи, які веб-дизайнери можуть використовувати для створення цікавих та зручних для користувача веб-сайтів.

Системи дизайну на базі штучного інтелекту аналізують демографічні дані та інші змінні, щоб надавати персоналізовані рекомендації з урахуванням контексту використання. Такі рішення допомагають створювати інтерфейси, які



не лише привабливі з візуальної точки зору, а й інтуїтивно зрозумілі, орієнтовані на потреби користувачів.

Практичне застосування таких систем включає:

1. *Чат-боти та віртуальні асистенти*: Покращують взаємодію користувачів із продуктом, відповідаючи на запитання, надаючи рекомендації та вирішуючи проблеми.
2. *Персоналізація*: Алгоритми аналізують поведінку користувачів, пропонуючи індивідуальні рішення щодо контенту чи функціоналу, що покращує досвід взаємодії.
3. *Тестування та аналіз поведінки*: ШІ допомагає виявити патерни поведінки та недоліки інтерфейсу, сприяючи розробці інтуїтивних дизайнів.
4. *Автоматизація дизайну*: Інструменти ШІ створюють графіку, шрифти та композиції швидше, зменшуючи час розробки.
5. *Аналіз візуальної ідентичності*: Алгоритми оцінюють відповідність дизайну стандартам і брендовій айдентичності.
6. *Оцінка емоційної реакції*: Аналіз обличчя та емоцій дозволяє вдосконалювати UX, враховуючи реакції користувачів.
7. *Генерація контенту*: ШІ створює тексти, зображення та відео для наповнення інтерфейсів.

Однак використання таких технологій викликає дискусії. Фахівці застерігають від ризику уніфікації дизайну, коли інтерфейси стають занадто схожими, що може знизити унікальність продуктів і їхню конкурентоспроможність (Дерман, Ткач, 2023: 162).

З появою нових технологій змінюються і тактики, які використовують веб-дизайнери для створення нового рівня дизайну та функціональності веб-сайтів, а також для покращення користувацького досвіду відвідувачів.

За останні роки було проведено багато дослідницьких робіт щодо зручності та ефективності використання інструментів штучного інтелекту в роботі веб-дизайнерів. Адже ШІ не може замінити людину, але може допомогти в оптимізації роботи, тим самим прискорюючи швидкість створення дизайнів, адаптивів та написання коду даючи змогу людині більше зануритись в творчий процес не витрачаючи багато часу на планування, дослідження, тестування тощо. У дослідженні 2021 року, проведеному серед UX/UI-дизайнерів у Бразилії, було виявлено, що нейронні мережі здатні значно автоматизувати тестування, аналіз даних і навіть планування дизайну. Водночас дизайнери зазначають, що поточне використання ШІ в цій сфері залишається обмеженим, з основним акцентом на підтримці, а не генерації рішень (Bertão, Renato Antonio ; Joo, Jaewoo, 2021).

Автоматизація процесів, які можна об'єднати із розробкою та тестуванням дизайну інтерфейсів, не є тривіальною задачею через непередбачуваність поведінки і можливий вплив змін в інтерфейсі на кодову базу додатка. Тому науковці та промисловість вивчають різні стратегії автоматизації, зокрема використання машинного навчання.

Одна з яскравих прикладів ШІ де використовується машинне навчання є нейронна мережа Stable Diffusion. Ця нейронна мережа, яка генерує зображення з текстового опису. Важлива відмінність Stable Diffusion від інших схожих нейронних мереж полягає в тому, що це нейронка із відкритим кодом.

Є багато способів, як користуватися Stable Diffusion. Найскладніше – завантажити вихідний код з GitHub і запустити його на комп'ютері згідно з інструкцією. Її особливостями є: генерація текст-у-зображення, генерація зображення-у-зображення, безкоштовна у використанні та немає ніяких

обмежень, не захищена авторським правом, безкоштовна для комерційного та особистого використання. Одним з її основних плюсів є те, що вона має відкритий вихідний код, що означає, що її можна модифікувати та адаптувати відповідно до конкретних потреб користувача (Рис. 1).



Рис. 1. Приклад створення зображення, використовуючи Stable Diffusion

Розкрито значний потенціал застосування штучного інтелекту, а саме нейронних мереж, у дизайні, зокрема для створення шрифтів, типографічних елементів, плакатів, банерів, розробки графіки й ілюстрацій. Шляхом порівняння можливостей нейронних мереж Midjourney, Stable Diffusion та DALL-E виявлено, що кожна з них володіє специфічним призначенням та архітектурою, що ефективно для виконання різних завдань в сфері дизайну. Результати дослідження свідчать про потенціал нейронних мереж для покращення навчання студентів спеціальностей, пов'язаних з дизайном. Обґрунтовано, що впровадження відповідних методик і технік може сприяти розширенню творчого спектру, забезпечити стабільність та контроль у процесі генерації образів та зумовити більш ефективне втілення ідей у візуальні реалії (Наталія Володимирівна Дерев'янка, Олена Юріївна Залевська 2023).

ChatGPT, розроблений компанією OpenAI, активно використовується в різних аспектах створення вебсайтів, включаючи генерацію контенту, оптимізацію для пошукових систем (SEO) та автоматизацію процесів розробки (Рис. 2).



Рис. 2. Інтерфейс ChatGPT

Застосування ChatGPT у веб-дизайні демонструє значний потенціал для автоматизації створення сайтів. Завдяки можливостям генерації HTML, CSS, JavaScript-коду та розробки шаблонів, цей інструмент дозволяє дизайнерам швидше реалізовувати свої ідеї. Наприклад, платформи на кшталт Divi AI інтегрують ChatGPT для автоматизації створення адаптивних сторінок і персоналізованих елементів дизайну, що спрощує процес розробки навіть для користувачів із базовими технічними знаннями. Разом із тим, використання ШІ в дизайні потребує додаткової перевірки з боку фахівців, щоб забезпечити високий рівень якості результатів і врахувати нюанси людського досвіду користування (UX) (By Aayush 2024).

Додатково, кейси з Carnegie Mellon University демонструють, як ChatGPT може підтримувати розробників у вирішенні задач оптимізації, написанні коду й створенні прототипів інтерфейсів, що підвищує продуктивність команд і скорочує час розробки (Matthew Walsh, Dominic Andrew Ross, Clarence Worrell, and Alejandro Gomez 2023).

**Висновки.** Штучний інтелект докорінно змінює підхід до UI/UX-дизайну, відкриваючи нові горизонти автоматизації та інновацій. Використання таких інструментів, як ChatGPT, MidJourney, Stable Diffusion та інші нейронні мережі, дозволяє дизайнерам та розробникам створювати інтуїтивно зрозумілі інтерфейси, персоналізувати взаємодію з користувачем та підвищувати ефективність роботи. Генеративні можливості ШІ, такі як створення текстів, зображень, коду або рекомендацій, оптимізують процес дизайну, скорочуючи час на виконання рутинних задач і даючи змогу спеціалістам більше зосереджуватись на творчості.

Проте, із впровадженням ШІ з'являються нові виклики, включаючи питання етики, безпеки даних та ризик втрати унікальності дизайну. Незважаючи на це, потенціал ШІ для автоматизації, покращення UX та адаптації веб-дизайну є надзвичайно перспективним, і його подальше дослідження відкриває нові можливості для інтеграції технологій у цифровий світ.

**Список використаних джерел:**

1. Дерман, Л.М., Ткач Г.Л. Штучний інтелект у дизайні XXI століття: етичні, філософські аспекти. 2023.
2. Наталія Володимирівна Дерев'янка, Олена Юріївна Залевська Порівняльний аналіз нейронних мереж Midjourney, Stable Diffusion та DALL-E та способи їх упровадження в навчальний процес студентів дизайнерських спеціальностей. 2023
3. By Aayush How to Build a Website With ChatGPT (2024 Guide). 2024
4. Renato Antonio Bertao, Jaewoo Joo, Artificial intelligence in UX/UI design: a survey on current adoption and [future] practices. 2021.
5. Matthew Walsh, Dominic Andrew Ross, Clarence Worrell, and Alejandro Gomez Demonstrating the practical utility and limitations of chatgpt through case studies. 2023



Сертифікат учасника конференції

