Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Медіасистем та технологій
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Проєктування та розробка web-сайту
«Використання сучасного штучного інтелекту у житті»
(тема)

Виконав:
студент 4 курсу, групи ВПВПС-19-3

Сергєєв Д.С.
(прізвище, ініціали)

Спеціальність 186 Видавництво та поліграфія
(код і повна назва спеціальності)
Тип програми освітньо-професійна

Освітня програма
Видавничо-поліграфічна справа
(повна назва освітньої програми)

Керівник доц. Табакова І.С.
(посада, прізвище, ініціали)

Допускається до захисту
Зав. кафедри МСТ _____ Дейнеко Ж.В.
(підпис) (прізвище, ініціали)

2023 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
Кафедра Медіасистем та технологій
Рівень вищої освіти перший (бакалаврський)
Спеціальність 186 Видавництво та поліграфія
Тип програми Освітньо-професійна
Освітня програма Видавничо-поліграфічна справа
(шифр і назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри МСТ_____
(підпис)
« 22 » травня 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Сергєєву Дмитру Сергійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Проєктування та розробка web-сайту «Використання сучасного штучного інтелекту у житті»

Затверджена наказом по університету від 22 травня 2023р. № 506 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 21 червня 2023 р.

3. Вихідні дані до роботи
Національні та міжнародні стандарти оцінки якості web-сайтів; ДСТУ Інженерія систем і програмних засобів. Розроблення та керування WEB-сайтами для систем, програмних засобів та інформаційних послуг

4. Перелік питань, що потрібно опрацювати в роботі
Вступ; Аналіз технічного завдання; Аналітичний огляд літератури за темою; Аналіз аналогів; Процес проєктування web-сайту; Процес розробки web-сайту; Економічна частина; Висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п. 5 включається до завдання за рішенням випускової кафедри)
Титульний слайд презентації; Актуальність та мета роботи; Задачі роботи; Економічна частина; Висновки; Публікації.

6. Консультанти розділів роботи (п. 6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п. 1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	доц. Табакова І.С.		19.06.2023
Економічна частина	проф. Помогалова Н.В.		19.06.2023

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз технічного завдання	17.05.2023	виконано
2	Аналітичний огляд літератури за темою	18.05.2023	виконано
3	Аналіз аналогів	19.05.2023	виконано
4	Процес проєктування web-сайту	22.05.2023	виконано
5	Процес розробки web-сайту	26.05.2023	виконано
6	Економічна частина	09.06.2023	виконано
7	Оформлення пояснювальної записки	10.06.2023	виконано
8	Оформлення графічної частини	16.06.2023	виконано

Дата видачі завдання 22 травня 2023 р.

Студент _____ Сергєєв Д.С.
(підпис)

Керівник роботи _____ доц. Табакова І.С.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи містить 66 стор., 10 рис., 2 табл., 1 дод., 10 джерел.

ШТУЧНИЙ ІНТЕЛЕКТ, ВЕБ-ДОДАТОК, REACT, EXPRESSJS, MONGODB, CLOUDINARY, OPENAI, AWS, NGINX, GIT, ОПТИМІЗАЦІЯ, ДИЗАЙН, ФІГМА, HTML, CSS, JAVASCRIPT, АДАПТИВНІСТЬ, ТЕСТУВАННЯ, ПУБЛІКАЦІЯ, ВЗАЄМОДІЯ, ДОКУМЕНТАЦІЯ, ІНТЕРФЕЙС, ПРОТОТИПУВАННЯ, ВЕРСТКА, РОЗРОБКА, СЕРВЕР, КЛІЄНТ.

Мета роботи полягає в проєктуванні дизайну та розробці web-сайту «Використання сучасного штучного інтелекту у житті», який буде надавати можливість взаємодії користувачів з штучним інтелектом.

Було проаналізовано технічне завдання. Кваліфікаційна робота складається з трьох частин. У першій частині визначається мета завдання та предметна область. У другій частині здійснюється аналіз, вибір інструментального та програмного забезпечення, для вирішення завдання з розробки web-додатку. Третій етап включає проєктування та розробку клієнтської та серверної частини web-додатку.

Web-додаток «AI & YOU» представляє собою унікальне середовище для спілкування з AI. Він дозволяє користувачам обмінюватися думками з ChatGPT, створювати арт з DALL·E 2 і ділитися ним зі світом. Web-додаток було розроблено з урахуванням найкращих практик в області web-розробки, використовуючи сучасні технології та інструменти. Він був протестований, оптимізований і публікувався на віддаленому web-сервері, що забезпечує його високу продуктивність та доступність для користувачів по всьому світу.

ABSTRACT

The explanatory note of the qualification work contains 66 p., 10 pic., 2 tabl., 1 app., 10 sources.

ARTIFICIAL INTELLIGENCE, WEB APPLICATION, REACT, EXPRESSJS, MONGODB, CLOUDINARY, OPENAI, AWS, NGINX, GIT, OPTIMIZATION, DESIGN, FIGMA, HTML, CSS, JAVASCRIPT, ADAPTABILITY, TESTING, PUBLISHING, INTERACTION, DOCUMENTATION, INTERFACE, PROTOTYPING, LAYOUT, DEVELOPMENT, SERVER, CLIENT.

The purpose of the work is to design and develop the «The use of modern artificial intelligence in life» website, which will provide an opportunity for users to interact with artificial intelligence.

The terms of reference were analyzed. The qualification work consists of three parts. The first part defines the purpose of the task and the subject area. The second part analyzes, selects tools and software to solve the task of developing a web application. The third stage includes the design and development of the client and server side of the web application.

The «AI & YOU» web application is a unique environment for communicating with AI. It allows users to exchange opinions with ChatGPT, create art with DALL-E 2, and share it with the world. The web application has been developed following the best practices in web development, using modern technologies and tools. It was tested, optimized and published on a remote web server, which ensures its high performance and availability to users around the world.

ЗМІСТ

С.

ВСТУП 7
1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ 10
2 АНАЛІТИЧНИЙ ОГЛЯД ЛІТЕРАТУРИ ЗА ТЕМОЮ РОБОТИ 12
2.1 Введення у Web-технології 12
2.2 Сучасні тенденції у Web-технологіях 13
2.3 Сучасні тенденції у Web-дизайні 15
2.4 Створення сучасних web-додатків 17
3 АНАЛІЗ АНАЛОГІВ 20
4 ПРОЦЕС ПРОЄКТУВАННЯ ВЕБ-САЙТУ 25
4.1 Визначення основних ідей та цілей проєкту 25
4.2 Проєктування майбутнього web-сайту 25
4.3 Інструментальні засоби розробки 27
4.4 Програмні засоби розробки 30
4.5 Підготовка інформаційного вмісту 33
5 ПРОЦЕС РОЗРОБКИ ВЕБ-САЙТУ 35
5.1 Створення дизайну web-сайту та прототипування 36
5.2 Розробка web-сайту 43
5.3 Оптимізація web-сайту 45
5.4 Попереднє тестування web-сайту 48
5.5 Публікація на віддаленому web-сервері 50
5.6 Тестування серед користувачів 51
6 ЕКОНОМІЧНА ЧАСТИНА 53
ВИСНОВКИ 64
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ 65
ДОДАТОК А Результат розробки сайту 67

ВСТУП

В сучасному світі штучний інтелект все більше проникає у наше повсякденне життя, проте розуміння та використання штучного інтелекту залишаються складними завданнями для багатьох людей, які мають обмежені технічні знання. Брак основних знань про те, що таке штучний інтелект, як він функціонує, та як його можна практично використати, створює складнощі у розумінні цієї технології та зумовлює потребу в повчальних ресурсах.

Багато людей не розуміють основних концепцій, принципів роботи та можливостей штучного інтелекту. Вони можуть плутати штучний інтелект з автоматизацією, роботами або навіть фантастичними сценаріями, які часто зустрічаються у кіно. Існує потреба в доступному та зрозумілому поясненні концепцій штучного інтелекту та у навчанні людей щодо того, як практично використовувати цю технологію в реальному житті.

Зростаюча роль штучного інтелекту у різних галузях життя підкреслює актуальність створення повчального web-додатка, який допомагатиме людям з обмеженими технічними знаннями зрозуміти сутність штучного інтелекту, його принципи роботи та надаватиме можливість практично застосовувати цю технологію. Такий додаток зможе покращити освіченість людей у сфері штучного інтелекту і допомогти їм використовувати ці знання в повсякденному житті.

Web-додаток – це програмне забезпечення, яке можна використовувати у web-браузері і дозволяє користувачам виконувати певні функції або отримувати певні послуги через Інтернет. Вони можуть бути створені для різних цілей, таких як комунікація, торгівля, освіта та розваги.

Web-додатки працюють на основі взаємодії між клієнтом (користувачем) та сервером. Користувач взаємодіє з додатком через свій web-браузер, вводить дані, отримує результати і взаємодіє з різними функціями,

які надає додаток. Сервер обробляє запити користувача, виконує потрібні операції та повертає результати користувачеві.

Web-додатки використовуються для різних завдань, таких як спілкування в соціальних мережах, покупки товарів онлайн, отримання освітніх матеріалів тощо. У контексті штучного інтелекту, web-додатки можуть використовуватись для навчання та популяризації концепцій штучного інтелекту, розповсюдження повчальних матеріалів, надання доступу до інструментів та сервісів, що використовують штучний інтелект, а також для демонстрації реальних застосувань штучного інтелекту у різних сферах життя. Web-додатки стають зручним інструментом для доступу до інформації та взаємодії з штучним інтелектом для тих, хто має обмежені технічні знання.

Для створення необхідного web-додатка необхідно виконати наступні кроки [1]:

– визначити цілі та об’єкти web-додатка;

– провести аналіз аудиторії майбутніх користувачів web-додатком;

– розробити детальний план створюваного проєкту;

– розробити дизайн та прототип web-додатка;

– розробити інтерфейс користувача та основний функціонал web-додатка;

– провести тестування та перевірити web-додаток на наявність помилок та проблем з різних пристроїв та браузерів;

– розмістити web-додаток на web-сервері, щоб він був доступний для користувачів.

Метою кваліфікаційної роботи є проєктування та розробка web-додатка «Використання сучасного штучного інтелекту у житті», ціллю якого є допомогти людині, яка жодного разу не працювала зі штучним інтелектом, зрозуміти що це таке і як із цим працювати. Додаток ділиться на дві частини – інформаційна та практична. Інформаційна частина представляє собою розділ додатка, який є документацією. У цьому розділі користувачі можуть

ознайомитись із поняттям штучного інтелекту та з інструкціями до інтерфейсів взаємодії, які є практичною частиною додатка. Завдяки цим інтерфейсам відвідувачі сайту можуть поспілкуватись із сучасним штучним інтелектом чи отримати зображення за необхідним запитом. Web-додаток відповідає багатьом сучасним тенденціям web-дизайну та web-розробки, має зручну навігацію. Відповідно до сучасних тенденцій, сайт буде крос-браузерним, крос-платформовим та адаптивним, тобто буде однаково працювати та добре виглядати на будь-якому пристрої та у будь-якому браузері.

Кваліфікаційна робота складається із трьох частин. Перша частина включає постановку мети, завдань і опис предметної області. Друга частина складається з аналізу та вибору програм для вирішення поставленого завдання з розробки web-додатка. Третя частина складається з проєктування та розробки клієнтської частини web-додатка.

1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

Створення повчального web-додатка, який допомагає людям з обмеженими технічними знаннями зрозуміти штучний інтелект і навчитись практичному його застосуванню, включає кілька основних етапів.

Спочатку необхідно чітко визначити мету додатка, таку як пояснення основних концепцій штучного інтелекту, демонстрація реальних застосувань або навчання користувачів практичному використанню штучного інтелекту.

Далі слід визначити цільову аудиторію додатка, враховуючи їх потреби та рівень технічного розуміння. Це можуть бути початківці без попередніх знань, студенти, підприємці або будь-яка інша група користувачів.

Після цього слід розробити структуру і дизайн додатка, забезпечити для цього зручну навігацію та логічну організацію інформації. Важливо врахувати естетику та простоту використання для кращого користувальницького досвіду. Потім необхідно розробити зміст додатка, який буде зрозумілим та доступним для цільової аудиторії. Це можуть бути, наприклад, пояснювальні статті та ілюстрації, що надають прості та зрозумілі пояснення концепцій штучного інтелекту.

Розробка функціональності включає створення інтерактивних інтерфейсів взаємодії, які допоможуть користувачам застосовувати сучасний штучний інтелект на практиці прямо у даному web-додатку.

Після розробки додатка необхідно провести тестування та «відладку», щоб виявити та виправити помилки. Залучення реальних користувачів для тестування та отримання їх відгуків може бути корисним для подальшого вдосконалення додатка.

Нарешті, після успішного тестування, необхідно розгорнути додаток на web-сервері та забезпечити його доступність для користувачів. Регулярне оновлення та підтримка додатка є важливими аспектами, які допоможуть забезпечити його краще функціонування та задоволення потреб користувачів.

Загалом, створення повчального web-додатка про штучний інтелект для людей з обмеженими технічними знаннями вимагає цілеспрямованої підготовки, врахування потреб цільової аудиторії та створення змісту та функціональності, які допоможуть користувачам зрозуміти і практично використовувати штучний інтелект.

Для даного проєкту, цільова аудиторія може включати в себе різноманітних користувачів, які мають різний рівень знань і досвіду у роботі зі штучним інтелектом. Серед цільової аудиторії можуть бути студенти, які вивчають технології інформаційних систем, а також фахівці з різних галузей, які бажають поглибити свої знання в цій області. Крім того, серед цільової аудиторії можуть бути підприємці, які бажають вивчити нові можливості застосування штучного інтелекту у своїй роботі.

Даний web-додаток надає можливість обирати мову інтерфейсу – українська чи англійська. Web-додатки, які підтримують як українську, так і англійську мову, мають кілька переваг. По-перше, це дає можливість залучити більше користувачів з різних країн та культур, що розширює аудиторію додатка. По-друге, для багатомовних користувачів це забезпечує зручне користування, оскільки вони можуть обирати мову, яка їм більш зрозуміла. Крім того, підтримка англійської мови дозволяє web-додатку залучати користувачів з міжнародних ринків та покращувати конкурентоспроможність.

Перед виконанням практичної частини необхідно створити дизайн та спроєктувати майбутній web-додаток. Дизайн сайту має відповідати сучасним трендам та вимогам, а також відповідати тематиці сайту. Далі, за прототипом додатка необхідно виконати практичну частину. Практична частина має включати у себе вміст сайту: ілюстрації, статті, інтерфейси взаємодії тощо.

2 АНАЛІТИЧНИЙ ОГЛЯД ЛІТЕРАТУРИ ЗА ТЕМОЮ РОБОТИ

2.1 Введення у Web-технології

Web-технології – це набір інструментів та методів, які використовуються для створення web-сайтів та web-додатків. Вони включають у себе різні ас-пекти, такі як web-дизайн, web-розробка, web-сервери, мережеві протоколи, безпека в Інтернеті, а також стандарти та рекомендації, які встановлені Всесвітнім web-консорціумом (W3C).

При web-розробці використовуються HTML, CSS та JavaScript, які призначені для того, щоб web-сайти могли відображатися в браузері. HTML до-помагає створити структуру web-сторінки, CSS використовується для того, щоб задати розмітці стилі, а JavaScript додає інтерактивність. Також важливою частиною web-технологій є сервери, які зберігають дані web-сайтів та відправляють ці дані до браузера користувача, коли він відкриває web-сайт. Все це разом створює те, що ми бачимо і використовуємо в Інтернеті кожного дня.

Розглянемо деякі важливі поняття, що описують web-технології.

Web-сторінка – це документ, доступний в Інтернеті. Вона створюється за допомогою HTML – мови розмітки, яка визначає структуру web-сторінки. CSS використовується для оформлення web-сторінки, включаючи колір, шрифт, розмір тексту та інші аспекти дизайну. JavaScript може бути використаний для додавання інтерактивності та функціональності web-сторінки.

Web-сайт – це збірка взаємопов'язаних web-сторінок, які розміщені під одним доменним ім'ям. Він може включати текст, зображення, відео, аудіо та інтерактивні елементи. Web-сайти можуть бути статичними, що означає, що вони не змінюються після їх створення, або динамічними, що означає, що вони можуть змінюватися відповідно до взаємодії користувача або змін на сер-вері.

Web-додаток – це програма, що використовує web-браузер як клієнтське програмне забезпечення. Він може виконувати різноманітні завдання, від простого відображення вмісту web-сторінки до складних операцій з даними. Web-додатки часто використовуються для таких задач, як онлайн-покупки, банківські операції, соціальні мережі, онлайн-ігри, інтерактивна візуалізація даних та багато іншого.

Коли користувач вводить URL web-сайту в браузер, браузер відправляє запит до web-сервера, на якому зберігається web-сайт. Web-сервер відповідає, надсилаючи відповідні файли (зазвичай HTML, CSS та JavaScript) назад до браузера, який потім інтерпретує ці файли та відображає web-сторінку. Якщо сторінка вимагає додаткової інформації, наприклад, якщо вона включає в себе форму пошуку, браузер може відправити додаткові запити до сервера.

Web-додатки працюють за допомогою цього ж процесу, але вони також можуть включати більш складну логіку на стороні клієнта, що дозволяє їм взаємодіяти з користувачем, використовувати дані в реальному часі та забезпечувати динамічний та багатофункціональний досвід.

2.2 Сучасні тенденції у Web-технологіях

Web-технології демонструють вражаючу здатність до адаптації і розвит-ку. З першого web-сайту, створеного в 1991 році, web-технології перетворили-ся в середовище, що забезпечує глобальну комунікацію, комерцію, розваги та навчання. Така глибока інтеграція web-технологій в наше повсякденне життя стала можливою завдяки безперервним інноваціям і постійному розвитку.

Останнім часом з'явилося багато нових тенденцій, що впливають на web-технології. Від динамічних односторінкових додатків (SPA), які роблять взаємодію з web-сайтами більш швидкою та зручною, до масштабованої архітектури мікросервісів, яка дозволяє великим web-сайтам працювати більш

ефективно. Не можна забувати про широке застосування штучного інтелекту та машинного навчання, що сприяє розвитку персоналізованих і інтерактивних web-сервісів. А також про новітні прогресивні web-додатки, які пропонують наближений до нативних досвід користування на мобільних пристроях.

Розглянемо деякі сучасні тенденції в області web-технологій:

– бібліотеки та фреймворки JavaScript. JavaScript продовжує домінувати в індустрії web-розробки. Це пояснюється широким використанням мови і доступністю сучасних фреймворків, таких як React, Vue і Angular. Ці фреймворки полегшують розробку складних web-додатків, надаючи інструме-нти для управління станом, взаємодією з API, маршрутизацією та іншими ти-повими задачами. Розуміння та використання цих фреймворків є важливим для сучасного web-розробника;

– односторінкові додатки (Single Page Applications, SPA). Односторінкові додатки (SPA) стають все більш популярними через їхню здатність до простої інтеракції з користувачами. У SPA весь контент завантажується лише один раз, після чого всі подальші дії відбуваються без перезавантаження сторінки. Це робить SPA подібними до настільних програм, але з доступністю з будь-якого web-браузера. Фреймворки та бібліотеки JavaScript, такі як React.js, Vue.js та Angular, використовуються для створення SPA і стали стандартами в індустрії;

– API та Мікросервіси. Використання підходів API та мікросервісів дозволяє підвищити гнучкість і масштабованість web-додатків. Підходи API стимулюють використання web-сервісів через стандартизовані інтерфейси, що спрощує інтеграцію з різними системами. З іншого боку, мікросервіси дозво-ляють розбити великі і складні системи на менші, незалежні компоненти, що працюють разом, що полегшує розробку, розгортання і обслуговування web-додатків;

– Progressive Web Apps (PWA). Progressive Web Apps (PWA) поєд-нують переваги web-сайтів та мобільних додатків. Вони можуть працювати офлайн, надсилати сповіщення і бути встановлені на головний екран при-

строю, подібно до звичайних мобільних додатків. PWA використовують сучасні web-стандарти, такі як Service Workers для офлайн-роботи і Push API для сповіщень. Це дозволяє їм надавати більш швидкі і зручні web-досвіди для користувачів;

– штучний інтелект та машинне навчання. Штучний інтелект (AI) і машинне навчання (ML) стають все більш актуальними в контексті web-технологій. Вони використовуються для розробки інтелектуальних систем, що можуть адаптуватися і вивчати патерни користувачів для надання більш персоналізованого контенту. Це може включати рекомендаційні системи, автоматичний пошук, чат-боти, системи аналізу даних та багато іншого.

Підсумовуючи, можна зробити висновок. Web-технології продовжують розвиватися і стають все більш витонченими, зумовлюючи спосіб, яким ми створюємо та використовуємо інтернет. Набирають обертів тенденції, які спрямовані на підвищення зручності та доступності web-сайтів. Безперечно, розвиток web-технологій продовжуватиметься, призводячи до появи нових тенденцій і можливостей для розробників і користувачів.

2.3 Сучасні тенденції у Web-дизайні

Web-дизайн – це сфера, яка перебуває у постійній динаміці і розвитку, відіграючи ключову роль в тому, як ми сприймаємо інформацію в інтернеті. Він не тільки визначає естетичні аспекти web-сторінки, а й впливає на функціональність сайту, його використання та загальне враження користувача. Сучасні тенденції web-дизайну продовжують розвиватися, орієнтуючись на розширення можливостей інтерактивності, удосконалення користувацького досвіду та використання новітніх технологічних досягнень.

Нижче представлено деякі з сучасних тенденцій у web-дизайні:

– мінімалістичний дизайн (Minimalist Design). Мінімалістичний дизайн, котрий характеризується своєю простотою та чистотою, продовжує підкорювати світ web-дизайну. Він став не лише стилем, а філософією, що

наголошує на меншій кількості елементів, зосереджуючись на функціональності. Ключ до мінімалістичного дизайну полягає у вмінні використовувати мінімум елементів для максимального ефекту. Це може означати обмеження кольорової палітри, використання простих або геометричних форм та мінімізація використання тексту та графіки. Мета полягає у тому, щоб уникнути зайвого шуму та спрямувати увагу користувача на основний контент;

– темний режим (Dark Mode). «Dark Mode», який спочатку з'явився в операційних системах та мобільних додатках, тепер став важливою функцією у web-дизайні. Темний режим зменшує втому очей в низькому освітленні, що робить його ідеальним для користувачів, які використовують пристрої пізно ввечері або в темних приміщеннях. Він також може допомогти з економією енергії на пристроях з OLED-дисплеями, оскільки чорний колір на цих дисплеях вимикає пікселі;

– мікровзаємодії (Microinteractions). «Microinteractions» – це маленькі, майже непомітні деталі, які роблять інтерфейс більш приємним і зрозумілим для користувачів. Це можуть бути дрібні анімації, які реагують на дії користувача, зміни кольору або форми елементів при наведенні курсору, звукові ефекти або вібрація при виконанні певних дій. Ці мікровзаємодії можуть зробити досвід користування більш задовільним та «відгуковим»;

– 3D-графіка та доповнена реальність. Завдяки прогресу технологій, 3D-графіка та доповнена реальність стають все доступнішими для web-дизайнерів. Ці технології можуть бути використані для створення більш вражаючих візуальних ефектів та інтерактивного досвіду, які можуть поліпшити залученість та взаємодію користувачів;

– скломорфізм (Microinteractions). «Glassmorphism» – це сучасний тренд у web-дизайні, який характеризується використанням «скляних» або прозорих елементів, що створюють відчуття глибини та багатомірності. Цей стиль дизайну включає в себе засоби, такі як прозорість, кольоровий градієнт, а також розмиття і відблиск на фоні, створюючи ілюзію скляного ефекту.

2.4 Створення сучасних web-додатків

Створення web-додатків – це велика та складна область, яка включає в себе багато різних технологій, методів та навичок. Однак основні етапи створення web-додатків можна згрупувати в наступні категорії:

– планування. Перед початком будь-якого web-проєкту необхідно зробити певну попередню роботу. Планування – це час, коли визначаються основні ідеї та цілі проєкту. Наприклад, обирається тип web-додатку: інтернет-магазин, блог, соціальна мережа, інформаційний портал тощо. Потім слід розробити структуру сайту, визначити основні сторінки і їх зміст. Важливо визначити й основний функціонал сайту: в який спосіб буде реалізовано взаємодію з користувачами, які функції будуть доступні для відвідувачів. Також на етапі планування обираються технології, які будуть використані для реалізації проєкту: мови програмування, фреймворки, бібліотеки, система керування базами даних. Розробники мають враховувати, що кожна технологія має свої особливості, переваги та недоліки, тому важливо вміти правильно обрати найбільш оптимальні інструменти для конкретного проєкту. Крім того, враховуються ресурси (час, кошти, людські ресурси), які будуть потрібні для реалізації проєкту. Розробляється графік робіт з урахуванням термінів виконання кожного етапу проєкту;

– дизайн. Незважаючи на те, що дизайн часто розглядається як частина розробки, насправді це окремий важливий етап створення web-додатку. Дизайн включає в себе створення візуального зображення сайту, його кольорової схеми, вибір шрифтів, розташування блоків, розробка логотипу, іконок та інших графічних елементів. Дизайнери повинні враховувати не тільки естетичні, але й функціональні аспекти. Вони повинні забезпечити зручність використання сайту, інтуїтивність інтерфейсу, доступність для людей з обмеженнями. Дизайн повинен відповідати цілям і цільовій аудиторії сайту. Сучасний web-дизайн також включає адаптивність: сайт

повинен належним чином відображатися на різних пристроях – настільних комп'ютерах, ноутбуках, планшетах, смартфонах;

– розробка. Цей етап включає безпосередньо написання коду web-додатку. Це починається з розробки “скелета” сайту – HTML-розмітки. За допомогою HTML розробник описує структуру web-сторінки: де будуть розташовані заголовки, абзаци, списки, форми, кнопки та інші елементи інтерфейсу. Далі йде стилізація сторінки за допомогою CSS. За допомогою CSS можна контролювати вигляд кожного HTML-елемента: його колір, розмір, шрифт, відступи, рамки, фон та багато іншого. Щоб зробити web-сторінку динамічною і взаємодіючою, використовується JavaScript. За його допомогою можна обробляти дії користувачів (наприклад, натискання кнопок), змінювати вміст сторінки без перезавантаження, взаємодіяти з сервером та ін. Якщо web-додаток передбачає взаємодію з сервером (наприклад, для збереження даних користувачів), то потрібно також розробити серверну частину. Тут можуть використовуватися такі мови програмування, як Python, Ruby, PHP або JavaScript (Node.js). Ця частина роботи також включає роботу з базами даних. Розробка – це не просто написання коду, а й його тестування. Розробник повинен перевірити, що весь функціонал працює правильно, що сайт відображається «коректно» в різних браузерах, на різних екранах, що не виникає помилок і проблем з безпекою;

– розгортання. Після того, як web-додаток розроблено і протестовано, його можна розгортати – перенести з локального комп’ютера розробника на сервер, де він буде доступний для користувачів Інтернету. При розгортанні важливо врахувати багато факторів: безпеку даних, швидкість завантаження сторінки, можливість масштабування (якщо кількість користувачів зросте). Можливо, буде потрібно встановити SSL-сертифікат для захисту передачі даних, налаштувати резервне копіювання, систему моніторингу стану сервера та ін. Після розгортання web-додатку його роботу треба постійно контролювати, регулярно оновлювати, виправляти помилки, додавати новий

функціонал. Також важливо займатися просуванням сайту, щоб привернути до нього увагу більшої кількості відвідувачів.

Щоб створити web-додаток, необхідно знати різні мови програмування, такі як HTML, CSS та JavaScript, а також можливо, серверні мови програмування, такі як Python, Ruby, PHP або JavaScript (Node.js). Також можуть знадобитися знання з баз даних, браузерів, серверів, протоколів Інтернету, безпеки в Інтернеті та багатьох інших аспектів web-технологій.

Крім того, при розробці web-додатків можна використовувати різні інструменти та фреймворки для полегшення процесу розробки. Фреймворки та бібліотеки, такі як React, Angular або Vue.js, можуть полегшити розробку інтерфейсу користувача, тоді як Node.js, Django або Ruby on Rails можуть полегшити розробку серверної частини. Інструменти, такі як Git, дозволяють контролювати версії та співпрацювати з іншими розробниками, тоді як редактори коду, такі як Visual Studio Code або Sublime Text, надають зручне середовище для написання коду.

3 АНАЛІЗ АНАЛОГІВ

Аналіз аналогів – це важливий етап при створенні web-додатку, який дає можливість розуміти, як ваш продукт може вирізнятися на ринку, визначити його конкурентних переваг та слабкі місця. Цей процес включає кілька ключових етапів:

– визначення потенційних аналогів. Першим кроком в аналізі аналогів є визначення потенційних конкурентів або подібних продуктів. При цьому важливо зосередитися не лише на тих, хто прямо конкурує з вами, але й на тих, хто може вирішити ті ж проблеми, що й ваш продукт;

– аналіз функціоналу. Далі варто провести детальний аналіз функціоналу кожного з вибраних аналогів. Це допоможе вам зрозуміти, які особливості, інструменти та сервіси вони пропонують своїм користувачам, які відмінності і спільні риси між ними;

– аналіз дизайну (UI) і користувацького досвіду (UX). Дизайн та користувацький досвід також важливі компоненти при аналізі аналогів. Зверніть увагу на то, як вони впливають на взаємодію користувачів з продуктом, чи є дизайн зручним і привабливим, чи легко користувачам знайти потрібну інформацію або виконати потрібну дію;

– аналіз відгуків користувачів. Відгуки користувачів можуть бути надзвичайно корисними при аналізі аналогів. Вони дають вам змогу зрозуміти, що користувачам подобається і що не подобається в інших продуктах, які проблеми вони вирішують, і які можуть виникнути у вас;

– порівняння цін та стратегій монетизації. Останнім кроком є аналіз цін і стратегій монетизації аналогів. Це може допомогти вам визначити, чи є ваш продукт конкурентоспроможним за ціною, і які стратегії монетизації можуть бути найбільш ефективними для вашого web-додатку.

Розглянемо схожі ресурси в Інтернеті.

«OpenAI API» (https://platform.openai.com/) – це платформа, на якій компанія OpenAI надає доступ до своїх технологій, досліджень та інших ресурсів. Наприклад, на цьому сайті можна знайти доступ до моделей машинного навчання OpenAI, API, інструкцій та документації. Платформа надає доступ до великої кількості документації, що дозволяє користувачу різними способами скористатися штучним інтелектом, а саме моделями штучного інтелекту, які створила компанія OpenAI (рис. 3.1).

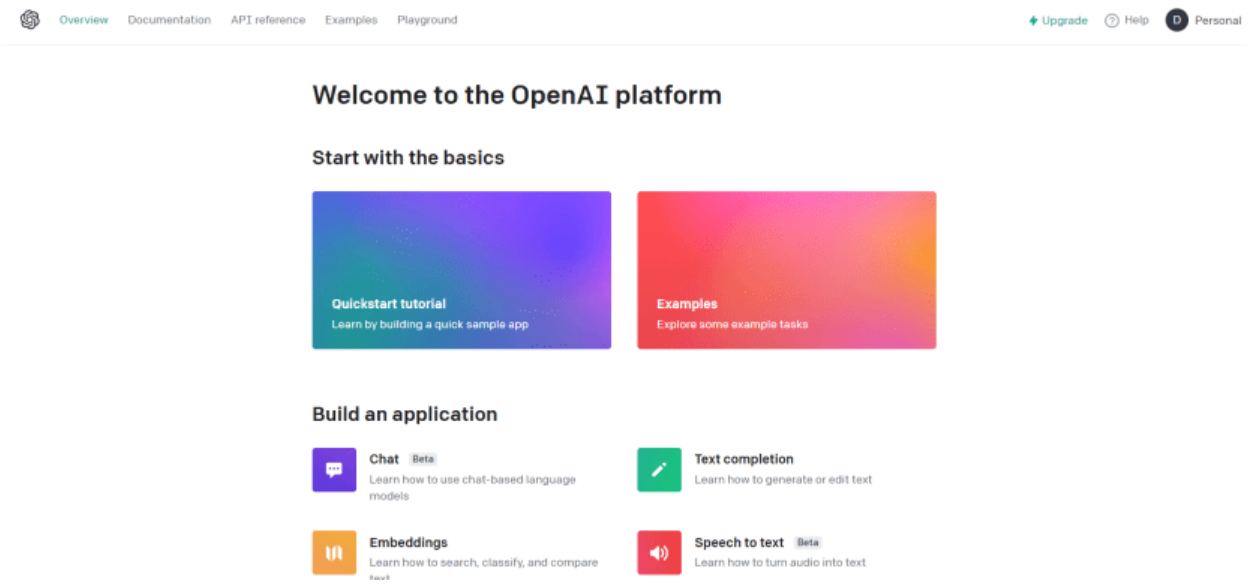

Рисунок 3.1 – Перший екран сайту OpenAI API

Розглянемо переваги та недоліки цього web-ресурсу.

Переваги:

– дизайн сайту. Web-сайт OpenAI API має мінімалістичний та сучасний дизайн. Він використовує чудове поєднання сучасних шрифтів без засічок та кольорову схему, завдяки чому навіть велика кількість тексту легко читається;

– наявність необхідного вмісту. Сайт містить величезну кількість інформації щодо взаємодії із моделями штучного інтелекту, які безпосередньо розробила компанія OpenAI;

– адаптивність. Сайт загалом має просту структуру, тому адаптив присутній, він є простим і зручним у використанні;

– необхідна функціональність. На сайті можна знайти всю необхідну документацію, яка розташована в одному розділі, і сторінку, де можна скористатися різними моделями штучного інтелекту.

Недоліки:

– доступність інформації. Даний ресурс є не навчальним посібником, а документацією і написана вона для розробників. Тому для звичайного користувача вона може здатися складною і з купою непотрібною інформації;

– умовна безкошновність. Для використання моделей штучного інтелекту на web-сайті OpenAI API необхідно мати певну кількість грошей на балансі профілю. Після реєстрації на цьому ресурсі OpenAI надає безкоштовно $5 кожному користувачу для випробовування можливостей штучного інтелекту, але після їх використання необхідно буде внести кошти, щоб продовжити користування.

Сайт «DeepLearning.AI» (рис. 3.2) – надає курси зі штучного інтелекту. Безпосередньо курси розміщені на іншій платформі, а на сайті розміщена інформація щодо курсів, інформація про компанію та інше .

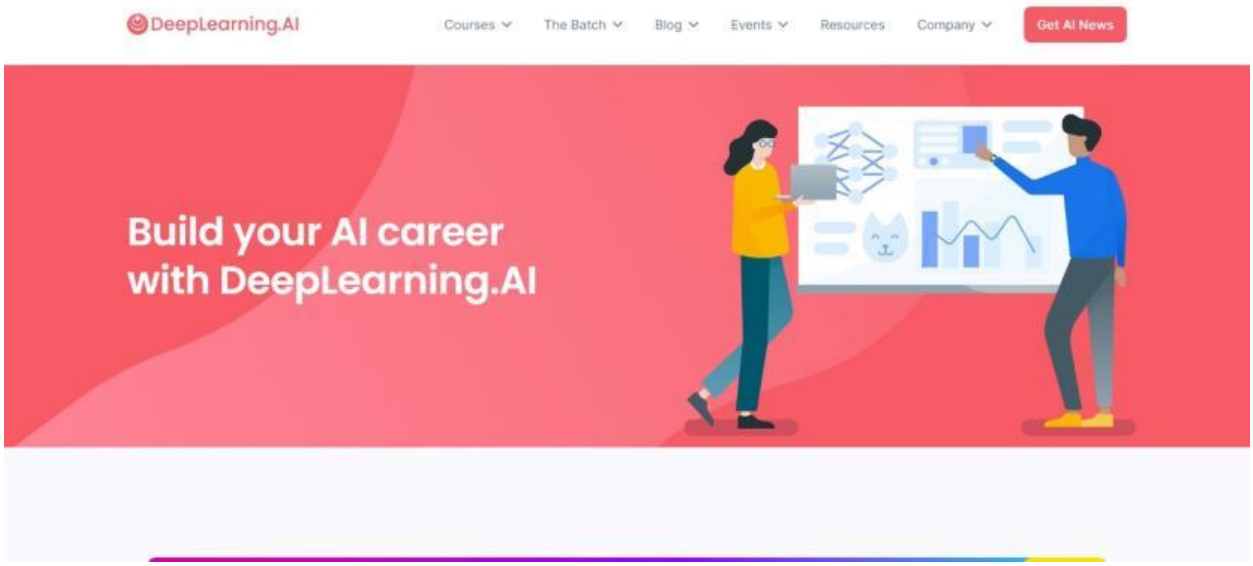

Рисунок 3.2 – Перший екран сайту DeepLearning.AI

Переваги:

– дизайн сайту. Сайт має чистий та професійний дизайн. Він використовує простий шрифт та кольорову схему, що сприяє читабельності. Вміст сайту організовано так, що користувачам легко знайти потрібну інформацію;

– наявність необхідного вмісту. Сайт містить велику кількість вмісту, що відповідає потребам користувачів, які шукають інформацію про штучний інтелект. Він пропонує різноманітні курси, новини та інформацію про події;

– адаптивність. Сайт також чудово адаптований, тому добре виглядає на усіх пристроях;

– необхідна функціональність. Ресурс містить всі необхідні функції для навчального ресурсу, включаючи каталог курсів, інформацію про курси та можливість реєстрації.

Недоліки:

– доступність інформації. Варто зауважити, курси (платні та безкоштовні) з вивчення штучного інтелекту, які надає ця платформа, також розрахована на розробників, тому звичайний користувач тут нічого не знайде;

– заплутана система навігації. Система навігації має багато посилань та розгалуджень, тому може заплутати користувача.

Останнім розглянутим ресурсом є «Elements Of AI» (https://www.elementsofai.com/) – це web-сайт, створений університетом в Фінляндії і компанією «MinnaLearn». Сайт пропонує два основні курси: «Вступ до AI» та «Створення AI». Курси комбінують теорію з практичними вправами і можуть бути завершені власним темпом користувача. Ця платформа навчання має на меті заохотити якомога більше людей дізнатися, що таке AI, що можна (та що не можна) робити з AI, і як почати створювати методи AI (рис. 3.3).

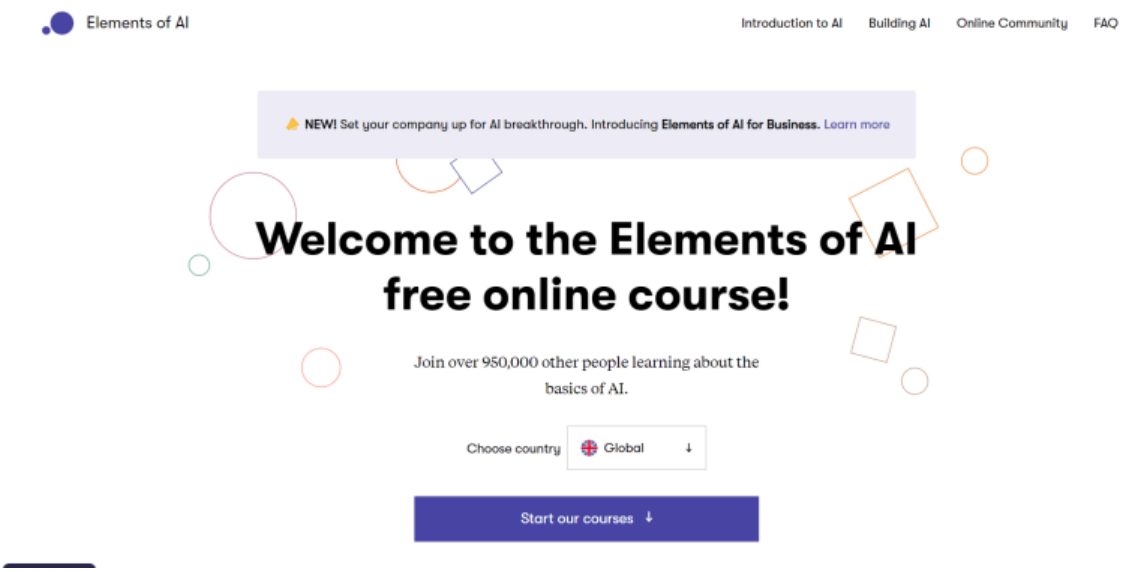

Рисунок 3.3 – Перший екран сайту DeepLearning.AI

Переваги:

– дизайн сайту. Щодо дизайну – він привабливий та мінімалістичний. У ньому дуже проста навігація, лише кілька посилань. Немає нічого зайвого, тільки найголовніше;

– наявність необхідного вмісту. На сайті пропонується два основні курси: «Вступ до AI» та «Створення AI», а також інформацію про спільноту;

– необхідна функціональність. Сайт містить всі необхідні функції для навчального ресурсу, включаючи каталог курсів, інформацію про курси та можливість реєстрації.

Недоліки:

– адаптивність. Хоча сайт має адаптив, він зроблений неякісно – елементи «стрибають», міняють своє положення і у цілому сайт виглядає погано на мобільних пристроях;

– доступність інформації. Хоча усі курси, що представлені на сайті є безкоштовними, вони також створені для розробників, тому інформація для простих користувачів буде дуже складною.

4 ПРОЦЕС ПРОЄКТУВАННЯ ВЕБ-САЙТУ

4.1 Визначення основних ідей та цілей проєкту

Визначення основних ідей та цілей проєкту є важливим кроком у процесі проєктування web-додатку. Основна ідея проєкту «AI & YOU» полягає в створенні web-платформи, яка не просто описує штучний інтелект, але й надає їм можливість безпосередньо взаємодіяти з AI технологіями.

Ціль проєкту полягає в тому, щоб зробити штучний інтелект доступним для всіх, незалежно від їхнього технічного досвіду або знань. Проєкт прагне демонструвати, як AI може бути використаний в різних областях життя, від навчання й роботи до творчості та розваг.

«AI & YOU» включає в себе ряд розділів, які допомагають користувачам краще зрозуміти штучний інтелект, його можливості та як його використовувати. Web-додаток надає інструменти для взаємодії з AI, такі як ChatGPT та DALL·E 2, що дозволяє користувачам спілкуватися з AI або генерувати зображення за допомогою AI.

Все це робиться з метою показати потенціал штучного інтелекту та його можливості для зміни світу. Проєкт «AI & YOU» прагне відкрити нові горизонти в областях навчання, роботи, творчості та розваг, демонструючи можливості штучного інтелекту.

4.2 Проєктування майбутнього web-сайту

Проєктування майбутнього проєкту є важливим етапом в процесі розробки web-додатку. Проєктування майбутнього проєкту вимагає глибокого розуміння потреб користувачів, технологій розробки та кращих практик у сфері дизайну та розробки web-додатків. Це включає в себе не

тільки технічні аспекти, але і стратегічне планування, творчість та уважність до деталей.

У процесі розробки web-додатку «AI & YOU» були виконані наступні кроки проєктування:

– тип web-додатку. Web-додаток «AI & YOU» було розроблено як web-додаток однієї сторінки (SPA), що забезпечує плавний досвід користувача та ефективне використання ресурсів;

– розробка структури сайту. Структура сайту була розроблена з урахуванням потреб користувачів та забезпечення легкого доступу до всіх функцій web-додатку;

– визначення основних сторінок та їх вмісту. Основні сторінки були розроблені з урахуванням цілей web-додатку та потреб користувачів. Вони включають сторінки документації, сторінки взаємодії з AI та інші;

– визначення дизайну сайту. Дизайн сайту було розроблено з урахуванням принципів UX та UI дизайну, що забезпечують зручність використання та естетичний вигляд;

– визначення основного функціоналу сайту. Основний функціонал web-додатку включає в себе взаємодію з AI, можливість перемикання мови та теми, відображення статистики та інші функції;

– вибір технологій розробки. Використовувались сучасні технології, такі як React, Express.js, MongoDB, Cloudinary, AWS, Nginx та інші, для розробки високопродуктивного та масштабованого web-додатку;

– створення дизайну сайту та прототипування. Процес прототипування допоміг визначити кінцевий вигляд web-додатку та виявити можливі проблеми на ранніх стадіях розробки;

– створення web-додатку. Процес розробки включав в себе написання коду, інтеграцію з базою даних та іншими сервісами, тестування та виправлення помилок;

– оптимізація web-додатку. Web-додаток було оптимізовано для підвищення продуктивності, швидкості завантаження та загальної ефективності;

– попереднє тестування. Перед публікацією web-додатку було проведено ряд тестів для виявлення та виправлення помилок та проблем з продуктивністю;

– публікація на віддаленому web-сервері. Web-додаток було опубліковано на віддаленому web-сервері за допомогою AWS, що забезпечило його доступність для користувачів по всьому світу;

– тестування серед користувачів. Було проведено тестування серед користувачів для отримання зворотнього зв'язку та виявлення проблем, які можуть бути непомітними для розробників;

– виправлення помилок, додавання необхідних функцій на основі відгуків користувачів та фінальна оптимізація. На основі отриманого зворотнього зв'язку було внесено ряд змін та вдосконалень до web-додатку, включаючи виправлення помилок, покращення інтерфейсу користувача, оптимізацію продуктивності та додавання нових функцій.

Даний план проєктування допоміг створити високоякісний, надійний та масштабований web-додаток, який відповідає потребам користувачів та надає їм позитивний досвід використання.

4.3 Інструментальні засоби розробки

Вибір правильних інструментальних засобів є важливим етапом у процесі розробки будь-якого web-додатку. Це включає вибір програмного забезпечення для редагування зображень, створення дизайну, написання коду, тестування та інших задач, що виникають під час розробки. Вибір правильних інструментів може значно полегшити процес розробки, підвищити продуктивність та якість кінцевого продукту.

У процесі розробки web-додатку «AI & YOU» було використано ряд програм, кожна з яких виконувала свою специфічну роль. Вони були вибрані на основі їх функціональності, надійності, зручності використання та сумісності з іншими інструментами розробки.

Ці програми включають:

– Adobe Illustrator для редагування векторних зображень;

– Adobe Photoshop для редагування растрових зображень;

– Figma для створення дизайну та прототипування web-додатку;

– Visual Studio Code для написання коду та розробки;

– web-браузери Google Chrome, Mozilla Firefox, Microsoft Edge та Opera для тестування web-додатку.

Розглянемо кожен з цих інструментів, їх особливості та значимість у контексті розробки web-додатку «AI & YOU».

Adobe Illustrator є інструментом для роботи з векторною графікою, який був використаний під час розробки web-додатку «AI & YOU». Ця програма дозволяє створювати та редагувати векторні зображення, які є ідеальними для використання в web-дизайні, оскільки вони можуть бути масштабовані без втрати якості. У контексті проєкту «AI & YOU», Adobe Illustrator було використано для створення та редагування логотипів, іконок, ілюстрацій та інших графічних елементів, які використовуються в дизайні web-додатку. Це дозволило створити високоякісні, професійно виглядаючі графічні елементи, які покращують візуальний досвід користувачів та допомагають відобразити бренд та ідентичність web-додатку.

Adobe Photoshop є потужним інструментом для редагування растрових зображень. Ця програма дозволяє редагувати растрові зображення на детальному рівні, включаючи корекцію кольору, контрасту, яскравості, редагування текстур, додавання ефектів та багато іншого.

При створенні проєкту, Adobe Photoshop було використано для обробки та оптимізації зображень, які використовуються на web-сайті. Це включає в себе зображення, що використовуються в дизайні інтерфейсу, а також зображення, що використовуються для ілюстрації контенту та документації.

Використання Adobe Photoshop дозволило створити високоякісні, оптимізовані зображення, які покращують візуальний досвід користувачів та підтримують загальний вигляд та «відчуття» web-додатку.

Figma – це інноваційний інструмент для дизайну інтерфейсів, який використовується для створення високоякісних прототипів web- та мобільних додатків. Цей інструмент надає дизайнерам можливість створювати інтерактивні прототипи, які можуть відтворювати реальний досвід користувача.

У процесі створення проєкту «AI & YOU», Figma було використано для створення детальних макетів та прототипів для кожної сторінки web-додатку. Це дозволило візуалізувати та тестувати різні елементи дизайну та інтерфейсу користувача перед тим, як почати процес розробки.

Visual Studio Code (VS Code) – це відкритий редактор коду, розроблений Microsoft, який став незамінним інструментом під час розробки web-сайту «AI & YOU». Цей інструмент надає розробникам широкий спектр можливостей, включаючи підсвічування синтаксису, автодоповнення коду, вбудований термінал, підтримку різних мов програмування та фреймворків, а також можливість розширення функціоналу за допомогою плагінів.

У процесі розробки web-додатку, Visual Studio Code був використаний як основний інструмент для написання коду [3]. Він сприяв ефективній та продуктивній роботі з такими технологіями, як React, Express.js, MongoDB, OpenAI API тощо. Особливо важливою є можливість VS Code підтримувати розширення, які дозволяють налаштувати редактор коду під конкретні потреби проєкту. Наприклад, розширення для підтримки React дозволило більш ефективно працювати з цією JavaScript бібліотекою.

Все це робить Visual Studio Code незамінним інструментом для розробки web-додатків, особливо для проєкту «AI & YOU», де він був використаний для написання більшості коду проєкту.

Також було використано декілька розширень Visual Studio Code, які значно полегшили процес розробки web-додатку. Ось декілька з них:

– Auto Complete Tag. Цей плагін автоматично закриває HTML та XML теги, що значно прискорює процес написання коду та зменшує ймовірність помилок;

– ES7+ React/Redux/React-Native snippets, JavaScript (ES6) code snippets, Reactjs code snippets. Ці плагіни надають набір шаблонів коду (snippets) для React, Redux, React Native та ES6, що дозволяє швидко генерувати часто використовуваний код. Вони значно прискорюють процес розробки та зменшують ймовірність помилок, дозволяючи розробникам зосередитися на більш складних аспектах розробки;

– ESLint: Цей плагін допомагає виявляти та виправляти проблеми з кодом на ранніх етапах, що підвищує якість коду та зменшує ймовірність помилок та прискорює їх виправлення;

– Import Cost. Цей плагін показує розмір імпортованих пакетів (сторонніх бібліотек) прямо в редакторі коду, що допомагає контролювати розмір кінцевого бандла (файлу JavaScript, що містить весь код) та оптимізувати продуктивність web-додатку;

– RapidAPI. Цей плагін дозволяє тестувати API запити прямо з редактора коду, що значно полегшує процес розробки та тестування API. Він дозволяє відправляти запити до API, переглядати відповіді в реальному часі, а також відстежувати статуси відповідей. Це було незамінним для тестування та налагодження API, яке використовується в «AI & YOU» для швидкого виявлення та вирішування проблем.

4.4 Програмні засоби розробки

У процесі розробки web-додатку «AI & YOU» було використано ряд програмних засобів та бібліотек, що дозволили створити ефективний, надійний та високопродуктивний продукт. Ці інструменти включають платформи, бібліотеки та сервіси, кожен з яких відіграє свою роль у створенні web-додатку. Вони сприяють створенню якісного коду, забезпечують гнучкість розробки та підтримують високу продуктивність web-додатку.

Розглянемо основні засоби, що були використані.

Основою для розробки web-додатку «AI & YOU» стала платформа Node.js. Node.js – це відкрита крос-платформна середовище виконання JavaScript, що дозволяє виконувати код JavaScript на сервері. Вона використовується для розробки серверних додатків та інструментів, а також для розробки повноцінних web-додатків. Node.js став основою для розробки web-додатку «AI & YOU», надаючи можливість використовувати одну мову програмування (JavaScript) на всіх етапах розробки, від клієнтської частини до серверної.

React.js є відкритою JavaScript бібліотекою для побудови користувацьких інтерфейсів, яку розробила команда Facebook. Вона використовується для створення багатосторінкових та односторінкових web-додатків, а також мобільних додатків. React дозволяє розробникам створювати великі web-додатки, які можуть змінювати дані, не перезавантажуючи сторінку.

В основі React лежить концепція компонентів. Компоненти – це повторно використовувані частини коду, які можуть мати власний стан і методи життєвого циклу. Це дозволяє розробникам створювати складні інтерфейси з простих частин.

Web-додаток «AI & YOU» повністю побудований на React. Він використовує React для створення інтерактивного користувацького інтерфейсу, який включає в себе різні сторінки, модальні вікна та інші елементи інтерфейсу. React дозволяє додатку «AI & YOU» бути швидким, відповідати на дії користувача в реальному часі та забезпечувати гладкий користувацький досвід.

Все це робить React вибором номер один для розробки web-додатку «AI & YOU». Він надає потужні інструменти для створення інтерактивних, відповідних та високопродуктивних web-додатків, які можуть забезпечити відмінний користувацький досвід.

Express.js є мінімалістичним та гнучким web-фреймворком для Node.js, який надає потужний набір функцій для web- та мобільних додатків. Він

включає безліч методів HTTP-утиліт та проміжного програмного забезпечення, завдяки чому створення надійного API стає швидким та простим. Express.js надає тонкий шар основних функцій web-додатків, не затуляючи при цьому функції Node.js, які ви знаєте та любите.

У контексті проєкту «AI & YOU», Express.js використовується для створення двох серверів: сервера для передачі файлів build клієнту та окремого API сервера для обробки запитів від клієнта. Це дозволяє розділити логіку обробки запитів та відправки статичних файлів, що підвищує гнучкість та масштабованість додатку.

Create-React-App є офіційно підтримуваним способом створення односторінкових додатків React. Він надає сучасне середовище збірки з нульовою конфігурацією, що дозволяє розробникам зосередитися на написанні коду, а не на налаштуванні інструментів.

У контексті проєкту «AI & YOU», Create-React-App був використаний для ініціалізації проєкту. Він надає стартовий шаблон, що включає в себе все необхідне для розробки додатку на React: встановлення необхідних залежностей, налаштування Webpack для збірки та Babel для транспіляції коду, налаштування тестування та багато іншого.

Webpack, який входить до складу Create-React-App, використовується для збірки додатку, дозволяючи об'єднати та мінімізувати JavaScript-код, CSS, зображення та інші ресурси в оптимізовані статичні файли, які потім можуть бути відправлені клієнту.

Babel, також включений до Create-React-App, використовується для транспіляції сучасного JavaScript (ES6, ES7 та вище) в код ES5, який сумісний з більшістю браузерів. Це дозволяє розробникам використовувати новітні функції JavaScript, не турбуючись про сумісність з браузерами.

Відмінність web-додатку «AI & YOU» від інших додатків полягає в його унікальному наборі використовуваних технологій та бібліотек. Ці інструменти були вибрані з метою оптимізації роботи додатку, поліпшення його функціональності та забезпечення найкращого користувацького досвіду:

– dotenv – це модуль, який дозволяє завантажувати змінні середовища з файлу .env в process.env. В проєкті «AI & YOU» він використовується для зберігання та використання конфіденційних даних, таких як ключі API, без їхнього включення в код проєкту;

– i18next – це міжнародний фреймворк для перекладу JavaScript. В «AI & YOU» він використовується для створення багатомовного інтерфейсу, що дозволяє користувачам вибирати мову інтерфейсу відповідно до їхніх вподобань;

– cloudinary – це хмарна платформа, яка надає рішення для управління зображеннями та відео. В «AI & YOU» вона використовується для завантаження, зберігання та відтворення зображень, що використовуються в додатку;

– mongoose – це бібліотека MongoDB об'єктного моделювання для Node.js. Вона дозволяє працювати з MongoDB на більш високому рівні абстракції, надаючи зручний API для створення, читання, оновлення та видалення даних. В «AI & YOU» вона використовується для взаємодії з базою даних MongoDB, що зберігає інформацію про користувачів та їхні діалоги з AI;

– openai – це модуль, наданий OpenAI, який дозволяє використовувати API OpenAI для використання їхніх моделей машинного навчання для генерації тексту. В «AI & YOU» він використовується для створення діалогів з AI, де AI генерує відповіді на запити користувачів, використовуючи модель машинного навчання GPT-3 та DALL·E 2.

4.5 Підготовка інформаційного вмісту

Підготовка інформаційного вмісту для web-додатку «AI & YOU» була виконана з великою увагою до деталей, щоб забезпечити зручність користувача та зрозумілість інформації.

Текстовий вміст був ретельно створений з урахуванням потреб користувачів та специфіки web-додатку. Частина тексту в розділі

документації була написана власноруч, використовуючи відкриті джерела в Інтернеті, а частина була згенерована з використанням штучного інтелекту ChatGPT. Весь текстовий вміст був розміщений у відповідні розділи для зручності користувачів.

Растрові зображення, які використовуються на головній сторінці, були створені за допомогою штучного інтелекту Midjourney і додатково відредаговані за допомогою Adobe Photoshop для досягнення оптимальної якості та естетики. Зображення у розділі документації були підібрані на Dribbble з безкоштовною ліцензією і також були оброблені за допомогою Adobe Photoshop.

Графіки, що використовуються в документації, були побудовані за допомогою ChatGPT, що дозволило відобразити необхідні дані в найбільш зрозумілій та візуально привабливій формі.

Весь текстовий вміст було написано на українській мові, а потім перекладено на англійську за допомогою онлайн-перекладача DeepL Translator, що використовує штучний інтелект.

Іконки та інші векторні зображення, що використовуються у web-додатку, були підготовлені та відредаговані у Adobe Illustrator.

5 ПРОЦЕС РОЗРОБКИ ВЕБ-САЙТУ

Існує безліч засобів розробки web-сайтів і вибір певного інструменту залежить від вимог проєкту, навичок розробника та інших факторів. Ось кілька популярних засобів розробки web-сайтів:

– HTML та CSS. HTML (мова гіпертекстової розмітки) та CSS (каскадні таблиці стилів) є основними мовами для створення структури та візуального оформлення web-сторінок. Вони є основою будь-якого web-сайту;

– JavaScript. JavaScript є мовою програмування, яка дозволяє створювати інтерактивні елементи на web-сторінках. Вона широко використовується для розробки динамічних функцій, валідації форм, створення анімації тощо;

– фреймворки. Фреймворки, такі як React, Angular та Vue.js, пропонують інструменти та бібліотеки, які полегшують створення web-додатків. Вони дозволяють обробляти стан програми та створювати складні інтерфейси;

– середовище розробки. Середовища розробки, такі як Visual Studio Code, Sublime Text, Atom та інші, пропонують зручне середовище для написання коду. Вони забезпечують підсвічування синтаксису, автодоповнення коду, налагодження та інші корисні функції;

– бази даних, такі як MySQL, PostgreSQL та MongoDB, використовуються для збору даних, таких як інформація про користувачів, контент та інші;

– системи керування версіями. Системи керування версіями, такі як Git, дозволяють відстежувати зміни в коді та співпрацювати з іншими розробниками. Вони також забезпечують контроль версій та відновлення попередніх версій коду;

– інструменти тестування. Для забезпечення якості коду та функціональності web-сайту використовуються інструменти тестування, такі

як Selenium, Jest, Mocha та інші. Вони дозволяють автоматизувати тестування та перевірку працездатності web-сайту;

– засоби розгортання. Розгортання web-сайту на сервері може відбуватися за допомогою багатьох інструментів, таких як Docker, Heroku, AWS і Netlify. Вони гарантують, що web-сайт буде доступним і працездатним, що полегшує процес розміщення web-сайту.

Крім перерахованих вище засобів, існують інші інструменти та технології, які можуть бути використані при розробці web-сайту в залежності від конкретних потреб проєкту. Важливо вибрати інструменти, які найкраще відповідають вимогам проєкту і якими розробник володіє або готовий освоїти.

Основні етапи розробки web-сайту включають проєктування, розробку інтерфейсу, програмування функціональності, тестування, оптимізацію та розгортання на сервері. Кожен із цих етапів може вимагати використання різних засобів розробки web-сайтів.

Розробка web-сайтів – це ітеративний процес, який потребує постійного розвитку та покращення. Розробники повинні бути готові вивчати нові інструменти та технології, щоб залишатися в курсі останніх тенденцій і гарантувати, що їхні проєкти будуть виконані з високою якістю.

Давайте детально розглянемо деякі із цих засобів.

5.1 Створення дизайну web-сайту та прототипування

Створення дизайну сайту та прототипування є одним з ключових етапів розробки web-додатку. Для цього процесу було вибрано застосунок Figma, який є потужним інструментом для дизайну та прототипування в режимі реального часу. Він дозволяє створювати деталізовані макети, проводити тестування прототипів та співпрацювати з командою, що значно спрощує процес розробки.

Дизайн web-додатку «AI & YOU» було створено з урахуванням сучасних тенденцій web-дизайну:

– мінімалізм – цей принцип відображається в простоті і чистоті дизайну, що сприяє зосередженню користувача на основному контенті сайту. Відсутність зайвих елементів дозволяє користувачу легко орієнтуватися та зосереджує його увагу на ключових функціях web-додатку;

– темний режим – ця функція дозволяє користувачам змінювати тему сайту відповідно до своїх переваг або умов освітлення. Темний режим забезпечує комфортне використання сайту в нічний час або в умовах слабкого освітлення;

– мікровзаємодії – це невеликі анімації або ефекти, які відбуваються відповідно до дій користувача. Вони додають сайту динаміки, підвищують його взаємодію з користувачем та покращують загальний досвід користування сайтом.

Кольорова палітра web-додатку була ретельно підібрана для створення приємного та комфортного візуального досвіду для користувачів (рис. 5.1).

Рисунок 5.1 – Кольорова палітра

Світла тема використовує градієнт з двох світло-блакитних, наближених до білих, кольорів: #F0F4F7 та #DCE4EE. Ці кольори створюють відчуття свіжості та простору, що сприяє комфортному сприйняттю інформації. Для тексту був обраний колір #1E1D1E, що створює достатній контраст з фоном для зручного читання, але при цьому не викликає дискомфорту.

Темна тема використовує градієнт з двох дуже темних, наближених до чорних, кольорів: #2D2F35та #101216. Ці кольори створюють відчуття глибини та загадковості, а також зменшують втому очей при використанні

web-додатку в темних умовах освітлення. Для тексту був обраний колір #DFE5EA, що забезпечує чіткість та читабельність тексту на темному фоні.

Усі інші елементи web-додатку, такі як кнопки, поля вводу та карточки, використовують тіні та напівпрозорий фон для того, щоб виділятись на фоні. У світлій темі використовується напівпрозорий фон темного кольору, а у темній темі – світлого кольору. Це дозволяє забезпечити контраст між елементами та фоном, але при цьому не відволікає від основного контенту.

Вибір шрифтів для web-додатку був зроблений з урахуванням їхньої читабельності, естетики та відповідності загальному стилю дизайну.

Mont Regular та Mont SemiBold були обрані як основні шрифти для тексту та заголовків відповідно. Mont – це сучасний геометричний гротеск, який відрізняється високою читабельністю та зручністю для використання в різних контекстах. Regular використовується для основного тексту, а SemiBold – для заголовків та посилань, що дозволяє зробити акцент на важливих елементах інтерфейсу.

Comfortaa Regular та Comfortaa Bold використовуються для великих заголовків. Comfortaa – це округлий геометричний шрифт, який додає відчуття дружелюбності та комфорту. Його використання для великих заголовків допомагає привернути увагу користувача та зробити інтерфейс більш зручним.

JetBrains Mono використовується для декоративного блоку на головній сторінці. Це моноширинний шрифт, який спеціально розроблений для відображення коду.

Для структуризації контенту web-додатку була використана стандартна 12-колонкова модульна сітка. Цей вибір був зроблений з урахуванням декількох ключових переваг цієї системи.

Перш за все, 12-колонкова сітка є надзвичайно гнучкою та універсальною. Вона дозволяє легко створювати різноманітні макети, оскільки 12 може бути поділено на 2, 3, 4 або 6 рівних частин. Це означає, що дизайнери можуть легко розміщувати елементи так, щоб вони займали будь-

яку кількість колонок, від однієї до дванадцяти, що дозволяє створювати багато різних композицій та макетів.

Другим важливим аспектом є те, що 12-колонкова сітка є стандартом в індустрії web-дизайну. Багато популярних фреймворків для розробки web-додатків, таких як Bootstrap або Foundation, використовують 12-колонкову сітку, що робить процес перетворення дизайну в код більш прямолінійним та ефективним.

Нарешті, використання модульної сітки допомагає забезпечити консистентність та організованість дизайну. Вона служить як основа для розміщення елементів на сторінці, що допомагає створити зрозумілий, організований та привабливий інтерфейс (рис. 5.2).

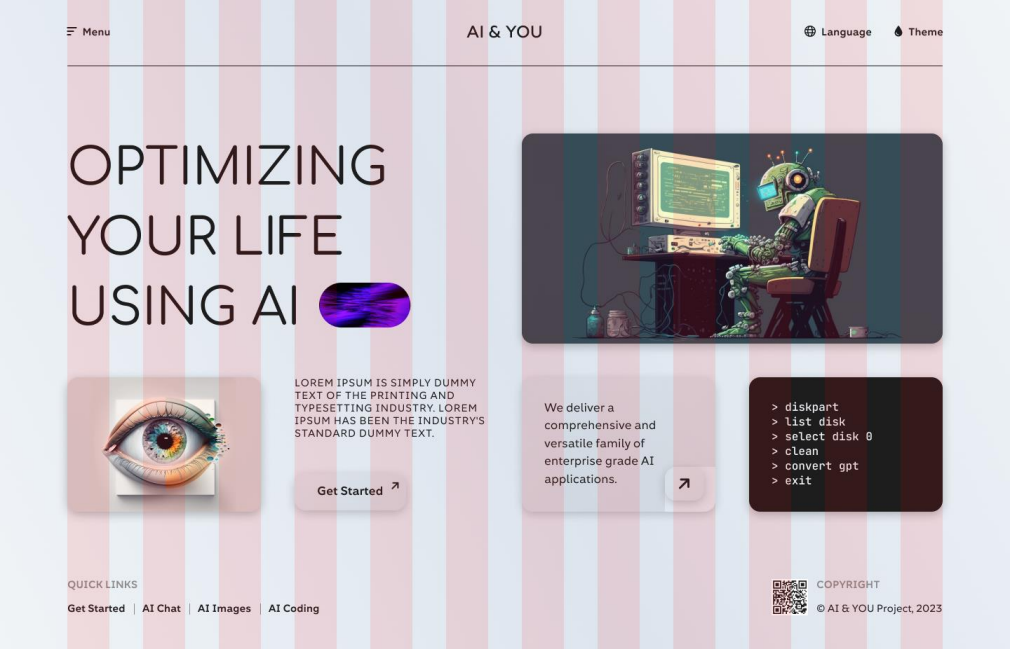

Рисунок 5.2 – Модульна сітка

Дизайн кожної сторінки web-додатку був ретельно продуманий, щоб забезпечити зручність та ефективність використання.

Хедер (шапка сайту) є важливим елементом навігації, який знаходиться на кожній сторінці web-додатку. Він містить:

– загальне випадаюче меню, яке містить всі головні посилання на сайті. Це надає користувачам легку навігацію між різними розділами сайту;

– текстовий логотип web-додатку, розташований посередині хедера. Цей логотип є посиланням на головну сторінку, що дозволяє користувачам швидко повернутися до початкової точки навігації;

– випадаюче меню для зміни мови додатку та випадаюче меню для зміни режиму web-додатку (світлий чи темний). Ці опції дають користувачам можливість налаштувати web-додаток відповідно до їхніх персональних вподобань.

Футер (підвал сайту) також є важливим елементом навігації, який знаходиться на кожній сторінці web-додатку. Він містить:

– список «швидких» посилань, який містить найголовніші посилання на розділи сайту. Це допомагає користувачам швидко знайти посилання на потрібний розділ сайту;

– напис копірайту, який вказує на авторські права web-додатку;

– QR код, що містить посилання на даний web-додаток. Це зручно для користувачів, які хочуть швидко відкрити web-додаток на мобільному пристрої.

Головна сторінка містить лише головний екран, що зустрічає кожного користувача при заході на сайт. Цей головний екран містить:

– великий заголовок, що відображає мету цього проєкту, а також невелика анімація біля заголовку для додавання інтерактивності. Це допомагає користувачам швидко зрозуміти, про що йде web-додаток, і залучає їх до подальшого використання;

– велике та маленьке зображення, згенеровані штучним інтелектом, демонструють можливості технології, що використовується в додатку;

– «провокуючий» напис, що підштовхує користувача перейти до розділу документації та почати ознайомлюватись із штучним інтелектом, а також кнопка, при кліці по якій відбувається перехід до розділу документації;

– блок з невеликим описом сайту з кнопкою-посиланням на розділ «Про Нас» у розділу документації;

– декоративний блок, що містить код. Це додає web-додатку технічний вигляд і підкреслює його технологічну спрямованість.

Будь-яка сторінка документації містить:

– зліва розташований блок навігації, що містить посилання на усі глави документації. Блок навігації завжди прикріплений до верху сайту, щоб користувач при гортанні довгої сторінки документації не загубив меню;

– справа розташована сама документація. Кожен розділ документації містить шапку, де є фон, який є зображенням, заголовок та короткий опис розділу. Після цього йде основний вміст розділу документації, розділений на глави. Кожна глава містить заголовок, при кліці по якому копіюється посилання на цю главу, та вміст документації. Це дозволяє користувачам легко навігуватися по документації та швидко знайти потрібну інформацію;

– також справа внизу сторінки міститься кнопка “нагору”, щоб користувач швидко міг повернутись до головного меню сайту. Це забезпечує зручність навігації, особливо при роботі з довгими сторінками.

Останній розділ web-сайту – це інтерфейс взаємодії з штучним інтелектом, який має свої унікальні особливості.

Зліва розміщена інформаційна частина інтерфейсу, яка включає заголовок (при кліці на якому копіюється посилання на цю сторінку), короткий опис даного штучного інтелекту, посилання на сторінку документації до цього інтерфейсу взаємодії та блок із статистикою використання цього інтерфейсу взаємодії, що автоматично оновлюється при використанні інтерфейсу користувачами.

Справа розміщений сам інтерфейс, за допомогою якого користувач може взаємодіяти із штучним інтелектом. У верхній його частині розміщуються повідомлення користувача та штучного інтелекту. Знизу – панель керування, у якій розміщено: поле вводу повідомлення користувача, кнопка “відправити”, що відправляє повідомлення до штучного інтелекту (недоступна доки користувач не введе повідомлення), кнопка “видалити” (недоступна доки користувач не введе повідомлення), що видаляє історію повідомлень.

У текстовому інтерфейсі взаємодії також є кнопка «завантажити», що завантажує історію повідомлень у форматі JSON (недоступна доки

користувач не введе повідомлення). У графічному інтерфейсі штучний інтелект надсилає зображення, на яких є кнопка «поділитись», при кліці на яку можна поділитись даним зображенням із спільнотою. Також у графічному інтерфейсі є кнопка «спільнота», при кліці на яку відкривається модальне вікно із галереєю зображень, якими поділились користувачі. При наведенні на кожне зображення відображається інформація про зображення – ким надіслано та за яким запитом згенеровано, а також є кнопка «завантажити», при кліці на яку завантажується дане зображення.

Адаптивність дизайну web-сайту є важливим аспектом, який забезпечує зручність та доступність для користувачів, незалежно від того, який пристрій вони використовують.

Дизайн нашого web-сайту було розроблено з урахуванням потреб користувачів різних пристроїв – від настільних комп'ютерів до планшетів та мобільних телефонів. Це означає, що елементи інтерфейсу, розмір тексту, розміщення кнопок та інші аспекти дизайну були оптимізовані для кожного типу пристрою, щоб забезпечити найкращий досвід користувача.

Для настільних комп'ютерів було створено повнофункціональний інтерфейс з великими, чіткими елементами та зручним меню навігації.

Для планшетів було зроблено акцент на збереженні функціональності, але з більш компактним розміщенням елементів, щоб вони були легко доступні на меншому екрані.

Для мобільних пристроїв було створено спрощений інтерфейс, який зосереджується на основних функціях сайту. Меню навігації, текст та інші елементи були оптимізовані для маленьких екранів, щоб забезпечити зручність користувача.

Таким чином, незалежно від того, який пристрій використовує користувач, він може легко та зручно взаємодіяти з нашим web-сайтом, отримуючи найкращий можливий досвід.

Прототипування є важливим етапом в процесі дизайну, який дозволяє перевірити та відточити ідеї перед їхньою реалізацією в кінцевому продукті.

У контексті цього проєкту, після створення дизайну сайту, було розроблено прототип за допомогою інструменту “Prototype” в Figma.

Цей інструмент дозволяє створювати інтерактивні прототипи web-сайтів, налаштовуючи зв'язки між різними сторінками, встановлюючи переходи та анімації, а також налаштовуючи роботу модальних вікон та інших інтерактивних елементів.

Приклади створеного дизайну представлені у додатку А.

5.2 Розробка web-сайту

Створення web-додатку – це багатоетапний процес, який включає в себе різні аспекти. Цей процес вимагає глибокого розуміння використовуваних технологій, а також ефективної координації зусиль. У цьому розділі детально розглядається процес створення web-додатку, включаючи ключові етапи розробки й використані технології, розглядається процес створення як клієнтської, так і серверної частини web-додатку.

Першим кроком у процесі розробки web-додатку було створення нового проєкту за допомогою інструменту Create React App [7]. Цей інструмент надає комфортне середовище для вивчення React, і з його допомогою можна створити новий односторінковий додаток. Він надає сучасне налаштування збірки та тестування з нуля.

Після ініціалізації проєкту, було проведено налаштування середовища розробки. Це включало в себе налаштування редактора коду [3], встановлення необхідних пакетів та бібліотек тощо.

Після ініціалізації та налаштування проєкту, почалась основна стадія розробки. Всі необхідні компоненти були створені [6] відповідно до створеного раніше дизайну. Весь web-додаток був поділений на окремі компоненти, що сприяє вірній структуризації й архітектурі проєкту.

Кожний компонент було розроблено та стилізовано, за допомогою написання файлу стилів відповідно до кожного компоненту. Для стилізації

компонентів [9] було використано препроцесор SASS, який потім Webpack компілює його у CSS, а для динамічної поведінки – JavaScript.

В процесі розробки компонентів було використано різні бібліотеки та інструменти, включаючи React Router для маршрутизації, React Helmet Async для налаштування мета тегів web-сторінок, React i18next для інтернаціоналізації додатку та інші.

Після розробки окремих компонентів, вони були об'єднані у більші компоненти. Цей процес включав створення «батьківських» компонентів, які включають в себе «дочірні» компоненти, створюючи таким чином ієрархічну структуру. Наприклад, може бути створений компонент «Header», який включає в себе компоненти «Logo» та «Navigation». Така структура дозволяє легко керувати компонентами та робить код більш зрозумілим та легким для підтримки.

Далі, для кожної сторінки сайту був створений окремий React компонент, у якому використовувались раніше створені компоненти. Наприклад, сторінка «Home» має у собі компоненти «Header», «FirstScreen» та «Footer». Цей етап включав в себе не тільки об'єднання компонентів, але й налаштування їх взаємодії [6]. Наприклад, було налаштовано передачу даних між компонентами, обробку подій (наприклад, кліків по кнопкам), а також створення умовного рендерінгу (тобто відображення різних компонентів в залежності від певних умов).

Після розробки клієнтської частини web-додатку, наступним етапом було створення серверної частини. Ця частина була написана на Node.js з використанням фреймворку Express.js [8], який дозволяє створювати масштабовані серверні додатки.

Серверна частина web-додатку включає в себе API сервер, який відповідає за обробку запитів від клієнтської частини, виконання бізнес-логіки та взаємодію з базою даних MongoDB [7]. Для роботи з MongoDB була використана бібліотека Mongoose, яка дозволяє моделювати дані та працювати з ними в уніфікований спосіб.

На сервері було налаштовано маршрутизацію, яка визначає, як сервер відповідає на певні HTTP запити від клієнтської частини. Наприклад, є маршрут, який обробляє GET запити по адресу «/api/v1/ai-chat-stats» та повертає список статистики використання AI Чату.

Сервер також використовує різні зовнішні сервіси та API. Наприклад, для зберігання та отримання зображень використовується сервіс Cloudinary, а для взаємодії з моделями штучного інтелекту – OpenAI API.

Після створення серверної частини, наступним важливим етапом було інтегрування клієнтської та серверної частин web-додатку [7]. Цей процес включає в себе налаштування взаємодії між клієнтським та серверним кодом, що дозволяє обмінюватися даними між ними.

На клієнтській частині було створено сервіс, який відповідає за виконання HTTP запитів до серверу та обробку відповідей від нього. Сервіс використовує різні методи для виконання GET, POST, PUT та DELETE запитів до серверу [7]. Кожен метод приймає URL та дані (якщо потрібно), виконує запит до серверу та повертає відповідь. Це дозволяє легко виконувати запити до серверу з різних частин клієнтського коду.

Після того, як сервіс було створено та налаштовано, його можна було використовувати в компонентах React для виконання запитів до серверу та отримання даних.

5.3 Оптимізація web-сайту

Оптимізація web-додатку є важливим етапом після його розробки. Це процес, який включає в себе ряд дій [5], спрямованих на підвищення продуктивності, швидкості завантаження та загалом покращення користувацького досвіду. Оптимізація може включати в себе різні аспекти, від оптимізації зображень до оптимізації коду та структури web-додатку.

Растрові зображення були сконвертовані у формат webp, який надає високу якість зображення при значно меншому розмірі файлу. Це дозволяє

зменшити час завантаження сторінки, особливо для користувачів з повільним інтернет-з'єднанням.

Оптимізація коду. Весь код web-додатку було ретельно переглянуто, з метою видалення непотрібного та оптимізації існуючого коду. Це включає в себе видалення невикористовуваного коду, оптимізацію циклів та умовних операторів, а також використання ефективних алгоритмів та структур даних.

Для оптимізації часу завантаження, великих бібліотек та компонентів було налаштоване на «ліниве» завантаження за допомогою React Lazy [5]. Це означає, що вони завантажуються лише тоді, коли вони дійсно потрібні, що дозволяє зменшити час завантаження сторінки.

Було використано бібліотеку express-static-gzip, яка стискає вихідні статичні файли проєкту з використанням алгоритмів стискання gzip та brotli [4], передає їх користувачу стиснутими, завдяки чому користувачу необхідно завантажити файли набагато меншого розміру. Завдяки стисканню розмір статичних файлів проєкту було зменшено приблизно на 75% від їх початкового розміру, що значно впливає на оптимізацію (рис. 5.3.).

```
Build compressed with br
      The build size was reduced to 23% of its initial size, 484.14 KB was removed.
      148.30 KB instead of 632.43 KB

Build compressed with gz
      The build size was reduced to 28% of its initial size, 454.50 KB was removed.
      177.94 KB instead of 632.43 KB
```

Рисунок 5.3 – Величина стискання статичних файлів

Для подальшого покращення продуктивності та швидкості завантаження, було налаштовано кешування статичних файлів. Кешування дозволяє браузеру зберігати копію статичних файлів (наприклад, CSS, JavaScript тощо), що використовуються на web-сайті, на пристрої користувача. Це означає, що при повторному відвідуванні сайту, браузеру не потрібно буде заново завантажувати ці файли, що значно прискорює час завантаження сторінки.

Google PageSpeed є інструментом, який дозволяє аналізувати продуктивність web-сторінки та виявляти можливі проблеми з оптимізацією. Цей інструмент було використано для виявлення та вирішення проблем, які могли впливати на швидкість завантаження та продуктивність web-додатку.

На рис. 5.4-5.7 наведено результати оптимізації на різних пристроях.

Ці оптимізації, виконані на різних рівнях web-додатку, сприяють створенню швидкого, ефективного та користувацького web-додатку, що забезпечує високу продуктивність та відмінний користувацький досвід.

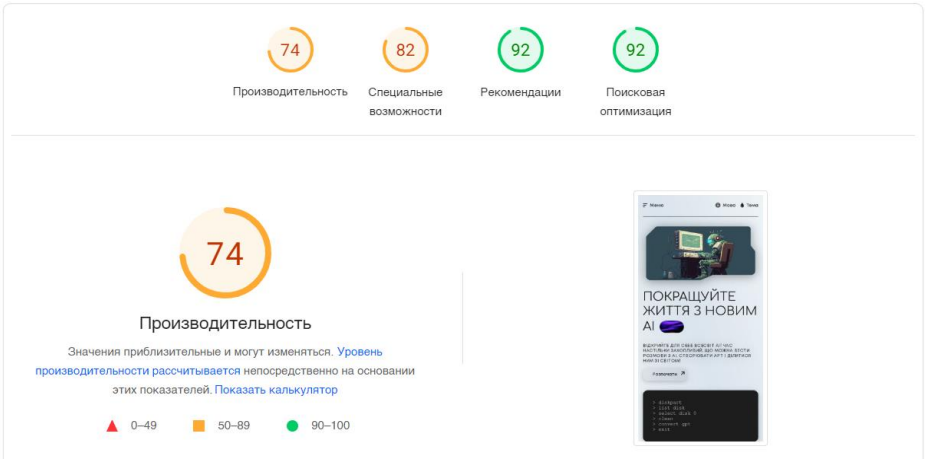

Рисунок 5.4 – Результат аналізу мобільної версії сайту до оптимізації

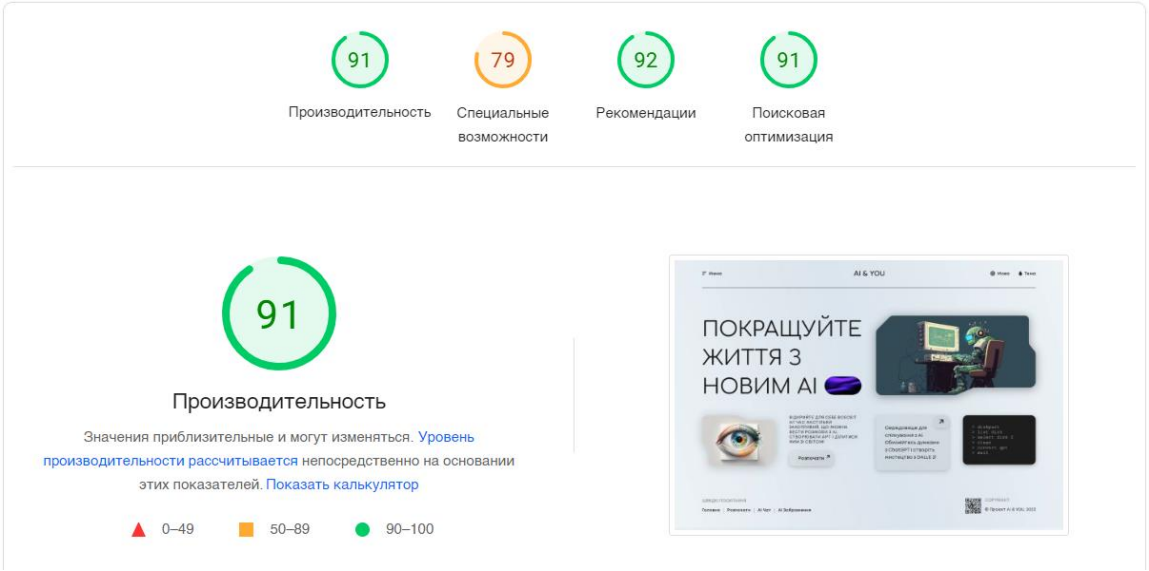

Рисунок 5.5 – Результат аналізу комп’ютерної версії сайту до оптимізації

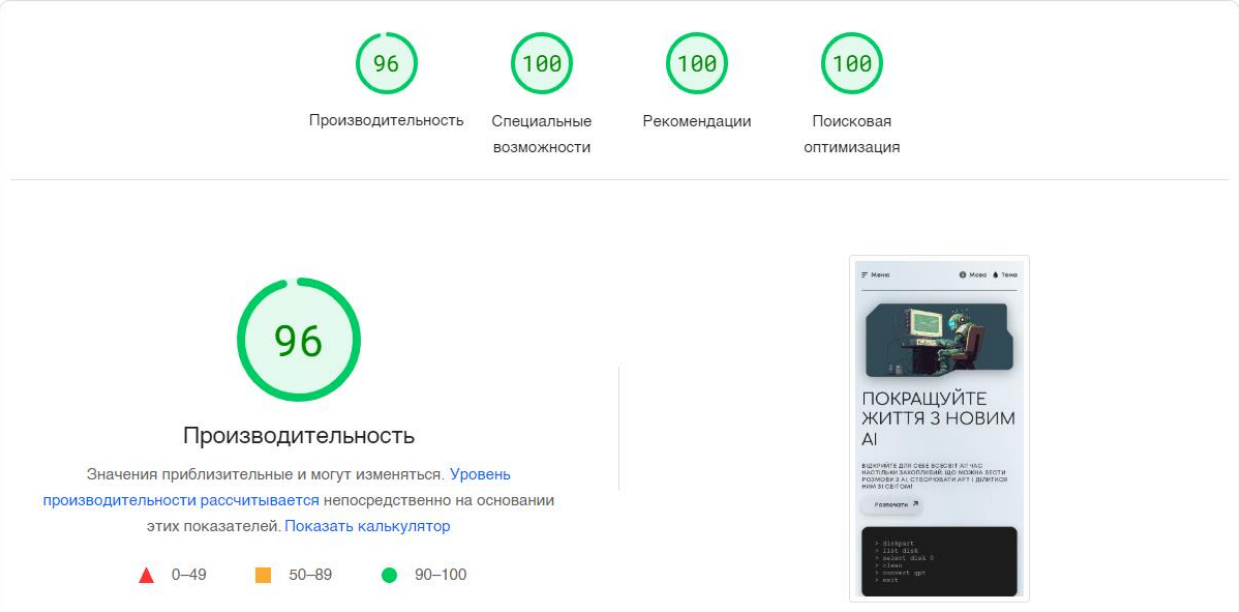

Рисунок 5.6 – Результат аналізу мобільної версії сайту після оптимізації

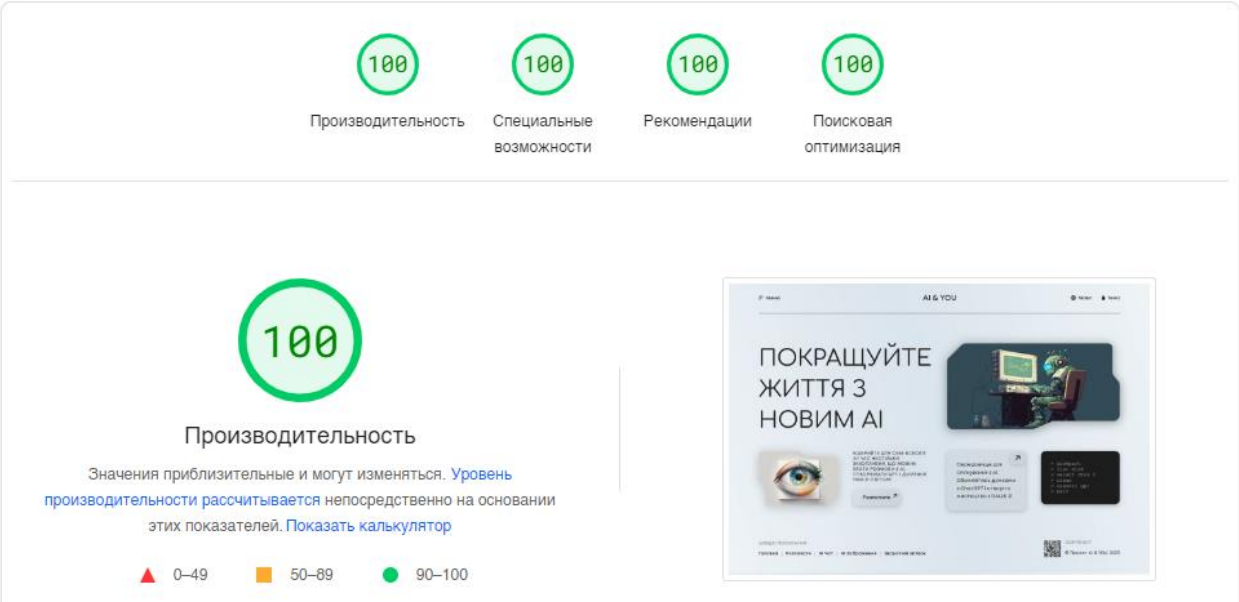

Рисунок 5.7 – Результат аналізу комп'ютерної версії сайту після оптимізації

5.4 Попереднє тестування web-сайту

Тестування web-додатку є критично важливим етапом у процесі його розробки. Це дозволяє переконатися, що web-додаток працює належним чином, відповідає всім вимогам та специфікаціям, а також забезпечує надійність та стабільність його роботи.

Під час розробки web-додатку кожен компонент тестується окремо. Це включає перевірку функціональності, відповідності дизайну, взаємодії з іншими компонентами та відповідності вимогам до продуктивності. Це дозволяє виявити та виправити будь-які проблеми на ранніх стадіях розробки.

Після того, як окремі компоненти були розроблені та протестовані, вони об'єднуються в більші компоненти або модулі [6]. Ці об'єднані компоненти також тестуються, щоб переконатися, що вони правильно взаємодіють один з одним та працюють належним чином як єдиний модуль.

Функціональне тестування web-додатку зосереджується на перевірці його функціональності. Це включає перевірку того, чи виконує web-додаток всі очікувані функції, чи він відповідає всім вимогам до функціональності, та чи він працює належним чином у різних сценаріях використання.

Тестування з точки зору користувача (User Acceptance Testing, UAT) – це тип тестування, що виконується з метою переконатися, що web-додаток є інтуїтивно зрозумілим, зручним для користувача та відповідає стандартам дизайну. Під час UAT, кінцеві користувачі або клієнти перевіряють систему перед тим, як вона буде перенесена в продуктивне середовище. Головна мета UAT – це перевірка бізнес-процесів від початку до кінця. Вона не фокусується на косметичних помилках, орфографічних або системних помилках.

Тестування продуктивності – це етап тестування, що фокусується на визначенні продуктивності web-додатку. Він включає перевірку швидкості завантаження web-додатку, часу відклику, а також інших показників продуктивності. Це важливо, оскільки продуктивність web-додатку може впливати на задоволеність користувачів.

Web-додаток також був протестований на різних браузерах та пристроях, щоб переконатися, що він відображається та працює правильно в різних умовах. Це включає перевірку сумісності з різними версіями браузерів, роботу на різних операційних системах та адаптивність до різних розмірів екранів. Web-додаток було перевірено у таких браузерах, як Google

Chrome, Mozilla Firefox, Microsoft Edge та Opera. Щодо пристроїв, web-додаток був перевірений на таких пристроях:

– комп'ютер із роздільною здатністю 1920x1080 px;

– ноутбук із роздільною здатністю 1440x900 px;

– планшет iPad Air 2017;

– смартфон iPhone 13.

У всіх браузерах та на всіх протестованих пристроях web-додаток виглядає добре та функціонує так, як повинен функціонувати.

5.5 Публікація на віддаленому web-сервері

Процес публікації web-додатку на віддаленому web-сервері [2] включає в себе декілька ключових етапів:

– вибір хостингу. Для розміщення web-додатку було обрано Amazon Web Services (AWS). AWS є одним з найпопулярніших хмарних сервісів, який надає широкий спектр послуг, включаючи обчислювальну потужність, зберігання даних, доставку контенту та інші функції, які допомагають компаніям масштабувати та рости;

– встановлення та налаштування web-сервера. Для обслуговування web-запитів було встановлено та налаштовано Nginx. Nginx є високопродуктивним web-сервером, який також може використовуватися як зворотний проксі-сервер, завантажувач балансу та HTTP-кеш. Він був налаштований для обробки маршрутизації відповідно до web-додатку, а також для переадресації, налаштування стискання gzip та brotli, та налаштування захищеного з'єднання;

– перенесення коду web-додатку. Для перенесення коду web-додатку на віддалений сервер було використано Git;

– публікація статичних файлів клієнтської частини web-додатку. Для публікації статичних файлів клієнтської частини web-додатку було використано сервер Express;

– придбання домену. Для web-додатку було придбано домен aiandyou.tech. Домен – це унікальне ім'я, яке вказує на місце розташування web-сайту в інтернеті. Він використовується для ідентифікації та знаходження web-сайтів.

5.6 Тестування серед користувачів

Після розробки, тестування та публікації web-додатку, посилання на нього було надіслано до групи користувачів. Ці користувачі використовували web-додаток в режимі реального часу, виконуючи різні задачі та взаємодіючи з його функціоналом. Вони також надавали зворотний зв'язок щодо виявлених помилок, проблем з користувацьким інтерфейсом або відсутності необхідного функціоналу.

В ході цього тестування було виявлено декілька недоліків web-додатку:

– відсутність зворотнього зв'язку. За необхідності звернутись до адміністраторів проєкту, користувачі не могли цього зробити, так як на сайті були відсутні будь які контакти;

– відсутність кнопки «нагору» у документації при гортанні довгих сторінок. Користувачі відмітили, що при гортанні довгих сторінок документації вони втрачають орієнтири. Тому було вирішено додати кнопку «нагору», що дозволяє швидко повернутися до початку сторінки;

– посилання у документації були малопомітні. Користувачі відмітили, що посилання в документації були недостатньо виділені, що ускладнювало навігацію;

– адаптив на планшетах некоректно відображався. Було виявлено, що дизайн web-додатку не відображався коректно на деяких планшетах;

– наявність зайвої кнопки у інтерфейсі взаємодії «AI Зображення». Було виявлено, що в інтерфейсі взаємодії «AI Зображення» була зайва кнопка, яка не виконувала жодної функції. Тому ця кнопка була видалена;

– було виявлено, що кнопки «завантажити» та «очистити» у інтерфейсах взаємодії були активними навіть тоді, коли чат був порожнім, що могло викликати незрозумілість у користувачів.

Тестування серед користувачів допомогло отримати відгук від реальних користувачів, виявити недоліки, які не були помічені під час попереднього тестування, та внести необхідні корективи до web-додатку.

Після виправлення остаточних помилок, додавання необхідних функцій та проведення фінальної оптимізації, web-додаток «AI & YOU» став готовим до повноцінного використання користувачами.

6 ЕКОНОМІЧНА ЧАСТИНА

Як результат виконання кваліфікаційної роботи маємо розгорнутий на web-сервері, протестований та готовий до використання сучасний web-додаток «AI & YOU». Створення даного ресурсу є закономірним та ефективним кроком для залучення нових користувачів та представлення інформації про можливості використання штучного інтелекту в повсякденному житті.

Перед виконанням практичної частини було детально спроєктовано майбутній web-сайт. Але перед початком проєктування є ще один дуже важливий крок – розрахунок економічної ефективності.

Розрахунок економічної ефективності дозволяє заздалегідь визначити потенційну вигоду від розробки та впровадження web-додатку, а також допомагає прийняти обґрунтоване рішення про доцільність такого кроку. Оцінка собівартості розробки допомагає розуміти, скільки коштуватиме створення додатку, враховуючи всі витрати, від зарплати працівників до вартості програмного забезпечення. Для розрахунку потенційного прибутку буде врахована собівартість розробки. Обчислення собівартості враховує оцінку витрат, що потрібні для створення та розробки web-додатку, включаючи заробітну плату працівників, витрати на ліцензійне програмне забезпечення та інше/ Наступним кроком є розрахунок ціни продукту. Для цього необхідно провести дослідження ринку та можливих конкурентів й визначитись із ціновою стратегією. Правильно встановлена ціна – це ключ до конкурентоспроможності продукту та можливості зайняти свою нішу на ринку. Таким чином, доцільне планування та аналіз перед початком розробки web-додатку є важливими для його успіху.

Розглянемо переваги створеного web-додатку:

– ефективність розробки. Використання JavaScript з бібліотекою React для клієнтської частини та Node.js з Express.js для серверної частини

дозволяє використовувати одну мову програмування на всіх етапах розробки. Це сприяє більш швидкій розробці, оскільки розробники можуть легко перемикатися між клієнтською та серверною частинами проєкту. Це також зменшує витрати на навчання та розвиток персоналу, оскільки потрібно вивчити лише одну мову програмування;

– скорочення витрат на інфраструктуру. Використання хмарних сервісів, таких як AWS для розгортання та хостингу web-додатку, а також Cloudinary для зберігання та доставки медіа-файлів, дозволяє компанії зменшити витрати на інфраструктуру. Замість покупки та обслуговування власних серверів, компанія може використовувати хмарні ресурси, які можна легко масштабувати відповідно до потреб;

– оптимізація продуктивності. Використання стратегій оптимізації, таких як мінімізація та об'єднання файлів JavaScript й CSS, а також використання кешування, дозволяє покращити продуктивність web-додатку. Це не тільки покращує досвід користувача, але і зменшує навантаження на сервер, що може зменшити витрати на хостинг;

– створення цінності з даних. Збір та аналіз даних користувачів, які можна використовувати для вдосконалення продуктів та послуг, а також для розробки ефективних стратегій маркетингу та продажу. Це може привести до збільшення прибутку від продажу продуктів або послуг, а також до збільшення лояльності користувачів;

– монетизація. Хоча основні функції web-додатку безкоштовні для користувачів, є можливості для монетизації. Наприклад, компанія може впровадити преміум-функції або платні плани, які надають користувачам додаткові переваги. Це може створити новий потік доходів для компанії;

– позиціонування на ринку. Використання передових технологій може підсилити репутацію компанії як інноваційного лідера в області штучного інтелекту. Це може привести до збільшення довіри та визнання бренду, що може сприяти збільшенню ділових можливостей та партнерств.

Ці економічні переваги можуть допомогти компанії збільшити доходи, оптимізувати операції та підвищити конкурентоспроможність на ринку.

Тепер перейдемо до аналізу конкурентного середовище в контексті створеного web-додатку.

Аналіз конкурентного середовища є важливим етапом під час планування та проєктування web-сайту з кількох причин:

– розуміння ринку. Аналіз конкурентів допомагає зрозуміти, які продукти або послуги вже присутні на ринку. Це дозволяє визначити, які ніші ще не заповнені, і де можуть бути можливості для нового web-сайту;

– визначення унікальної пропозиції. Знання про конкурентів допомагає визначити, як ви можете відрізнитися. Це може включати унікальні функції, контент або користувацький досвід, який ви можете запропонувати;

– навчання від успіху та невдач конкурентів. Дослідження того, що працює або не працює для конкурентів, може надати цінні уроки. Це може допомогти уникнути помилок та використати перевірені стратегії;

– встановлення цінової стратегії. Розуміння того, як конкуренти цінують свої продукти або послуги, може допомогти визначити, як встановити власні ціни;

– планування маркетингових стратегій. Знання про конкурентів може допомогти в плануванні ефективних маркетингових кампаній, включаючи вибір каналів просування, визначення цільової аудиторії та розробку повідомлень.

Щодо аналогічних продуктів, їх сильні сторони можуть включати велику кількість корисного контенту, широку аудиторію та високу репутацію. Однак, їх слабкі сторони можуть включати складність використання, відсутність інтерактивних елементів, а також можливість надмірного навантаження інформацією, що може змусити користувачів відчувати себе переобтяженими або заплутаними.

В порівнянні з конкурентами, web-додаток «AI & YOU» виглядає більш привабливо та сучасно, пропонує більш глибокий та інтерактивний

досвід навчання, і, що найважливіше, він зосереджений на конкретній темі штучного інтелекту, що робить його більш релевантним для цільової аудиторії. Серед основних переваг створеного додатку можна виокремити:

– спеціалізований фокус. Проєкт зосереджений на штучному інтелекті і його впливі на суспільство. Це дуже актуальна тема, яка привертає увагу широкої аудиторії;

– освітній контент. Web-ресурс надає якісний освітній контент, який допомагає користувачам навчитися новому і розширити свої знання;

– інтерактивність. Web-сайт включає інтерактивні елементи, які допомагають користувачам краще зрозуміти тему штучного інтелекту;

– простота використання. Створений web-додаток має просту та інтуїтивно зрозумілу навігацію, що робить його легким для користувачів з різним рівнем знань та досвіду.

Розглянемо процес створення web-сайту на прикладі «AI & YOU». Загалом процес розробки інформаційного web-сайту має наступні етапи:

а) етап перед-проєктування:

1) розрахунок економічної ефективності проєкту «AI & YOU»;

2) визначення основних ідей та цілей проєкту;

б) етап проєктування web-сайту:

1) визначення типу web-сайту;

2) розробка структури web-сайту;

3) визначення основних сторінок та їх вмісту;

4) визначення особливостей дизайну сайту;

5) визначення основного функціоналу сайту;

6) вибір технологій розробки;

7) визначення інструментальних та програмних засобів;

в) етап створення дизайну сайту та прототипування;

г) етап розробки web-сайту:

1) розробка основного функціоналу сайту;

2) оптимізація web-сайту;

3) попереднє тестування web-сайту;

д) етап тестування та публікації web-сайту:

1) тестування web-сайту;

2) публікація на віддаленому web-сервері;

3) відкрите тестування web-сайту;

4) виправлення помилок, додавання необхідних функцій на основі відгуків користувачів та фінальна оптимізація.

Процес проєктування та дотримання встановлених етапів є критично важливим для успішного створення web-сайту. Він допомагає забезпечити систематичність, послідовність та контроль якості на всіх стадіях розробки, від початкового планування до фінального запуску та підтримки. Це не тільки забезпечує ефективне використання ресурсів, але й допомагає уникнути потенційних проблем, що можуть виникнути внаслідок недостатнього планування або неконтрольованого процесу розробки.

Тепер виконаємо розрахунок вартості виготовлення та ціни розробки web-сайту. Для розрахунку вартості врахуємо наступні складові витрат:

– основна заробітна плата – це безпосередньо оплата праці розробників, дизайнерів, тестувальників та інших учасників команди, які працюють над проєктом;

– додаткова заробітна плата – це додаткові виплати, які можуть включати премії, винагороди за надурочні роботи, бонуси, премії та інші виплати, які виходять за рамки основної заробітної плати;

– єдиний соціальний внесок – це внески до державних соціальних фондів, які робляться з заробітної плати працівників. Вони включають внески на пенсійне забезпечення, медичне страхування та інші соціальні виплати;

– інші витрати – це додаткові витрати, які можуть включати витрати на хмарні сервіси для розміщення сайту в мережі, ліцензійне програмне забезпечення, витрати на обслуговування, підтримку обладнання та інші витрати, які пов’язані з розробкою та підтримкою веб-сайту.

Розрахунок собівартості розробки та встановлення ціни за розробку web-додатку дозволить оцінити фінансові витрати на проєкт та встановити оптимальну ціну за надані послуги. Створення web-додатку «AI & YOU» – це складний процес, який вимагає співпраці цілої команди спеціалістів. Кожен учасник команди відіграє важливу роль у створенні продукту, використовуючи свої навички та досвід для вирішення конкретних завдань. Вони працюють разом, щоб перетворити початкову ідею в конкретний продукт, який відповідає вимогам ринку та високим стандартам якості.

Team Lead (керівник команди) відіграє вирішальну роль у процесі розробки web-додатку. Він координує всі аспекти проєкту, включаючи планування, розподіл завдань, контроль якості та комунікацію зі стейкхолдерами. Team Lead відповідає за встановлення та дотримання термінів виконання проєкту, а також за забезпечення високої продуктивності та ефективності роботи команди.

UI/UX дизайнер відповідає за створення візуального інтерфейсу web-додатку, забезпечуючи його зручність та привабливість для користувачів. Він розробляє дизайн сторінок, визначає структуру інформації, вибирає кольорову схему та шрифти. UI/UX дизайнер також відповідає за створення прототипів, проведення тестування користувачів та внесення відповідних змін на основі отриманих відгуків.

Frontend розробник – одна з ключових посад в команді, яка відповідає за створення візуальної частини web-додатку, що включає в себе інтерфейс користувача, анімацію, взаємодію з користувачем та інші візуальні аспекти. Frontend розробники втілюють дизайн, створений UI/UX дизайнером, в код, використовуючи технології, такі як HTML, CSS та JavaScript. В даному випадку, основною технологією є React. Frontend розробник також відповідає за забезпечення сумісності web-додатку з різними браузерами та пристроями, оптимізацію швидкості завантаження сторінок та впровадження доступності для користувачів з особливими потребами. В залежності від розміру та складності проєкту, команда може включати більше ніж одного Frontend

розробника. Вони можуть спеціалізуватися на різних аспектах розробки, таких як візуальний дизайн, анімація, або взаємодія з користувачем. Великі проєкти можуть вимагати більшої команди Frontend розробників для ефективного виконання всіх завдань та досягнення цілей проєкту вчасно.

QA інженер (інженер забезпечення якості) відповідає за перевірку web-додатку на наявність помилок та несправностей. Проводить різноманітні тести, включаючи функціональне тестування, тестування користувацького інтерфейсу, тестування продуктивності та інше, щоб переконатися, що web-додаток працює належним чином та відповідає всім вимогам.

DepOps інженер (інженер з розгортання та експлуатації) відповідає за розгортання web-додатку на сервері та його подальшу підтримку. Забезпечує стабільну роботу web-додатку, вирішує технічні проблеми, що виникають під час його експлуатації, та оптимізує процеси розгортання та оновлення.

Контент менеджер відповідає за наповнення web-додатку контентом. Він створює, редагує та оновлює текстовий та візуальний контент, забезпечуючи його актуальність та відповідність цілям web-додатку. Контент менеджер також може взаємодіяти з користувачами, відповідаючи на їхні запитання та отримуючи відгуки.

Кожен член команди відповідає за свою частину роботи. Team Lead керує процесом розробки, визначає стратегію та координує дії команди. UI/UX дизайнер створює інтуїтивно зрозумілий та привабливий інтерфейс, що забезпечує зручність користування додатком. Frontend розробник втілює дизайн та функціональність в код. QA інженер забезпечує якість продукту, проводячи тестування та виявляючи помилки. DevOps інженер забезпечує надійність та ефективність інфраструктури, що підтримує додаток. Контент-менеджер створює та оновлює інформаційний контент додатку, забезпечуючи його актуальність та корисність для користувачів. Ця взаємодія та співпраця є ключем до успішного створення web-додатку.

Заробітна плата Team Lead спеціаліста – 880 грн/год, UI/UX дизайнера – 320 грн/год, Frontend розробника – 430 грн/год, QA інженера –

320 грн/год, DevOps інженера – 400 грн/год, контент менеджера – 160 грн/год. Тривалість робочого дня всієї команди становить 8 годин. Розробка веб-додатку триває 23 дні (табл. 6.1).

Таблиця 6.1 – Розрахунок витрат на заробітну плату

Етап	Вид робіт	Виконавець		Годинна ставка, грн	Тривалість виконання, дні	Заробітна плата, грн
		Кіль-кість	Посада			
1. Перед-проєктування	Розрахунок економічної ефективності	1	Team Lead	880,00	1	7040,00
	Визначення основних ідей та цілей					
2. Проєктування	Визначення типу web-сайт	1	Team Lead	880,00	2	14080,00
	Розробка структури web-сайту					
	Визначення особливостей дизайну сайту					
	Визначення основного функціоналу сайту					
	Вибір технологій розробки					
	Визначення інструментальних та програмних засобів					
	Визначення основних сторінок та їх вмісту	1	Контент менеджер	160,00	1	1280,00
3. Створення дизайну та прототипу	Розробка дизайн макету та прототипу web-сайту	1	UI/UX дизайнер	320,00	3	7680,00
4. Розробка	Розробка основного функціоналу сайту	1	Frontend розробник	430,00	14	48160,00
	Оптимізація web-сайту					
	Попереднє тестування web-сайту					
5. Тестування та публікація	Тестування web-сайту	1	QA інженер	320,00	1	2560,00
	Публікація на віддаленому web-сервері	1	DevOps інженер	400,00	1	3200,00
Разом					23	84000,00
Додаткова заробітна плата (20 %)						16800,00
Усього						100800,00

Додаткова заробітна плата – це частина заробітної плати, яка виплачується працівникам додатково до основної заробітної плати. Вона може включати різні види виплат, які залежать від конкретних умов праці, досягнень працівника або специфіки роботи: премії, надбавки, компенсації, бонуси.

Додаткова заробітна плата є важливим інструментом для мотивації та винагороди працівників [10]. Вона стимулює працівників досягати вищих результатів, виконувати завдання ефективніше та підвищувати свою продуктивність. Це також спосіб визнати та винагородити працівників за виняткову роботу, досягнення певних цілей або виконання певних завдань. Додаткова заробітна плата може служити як форма визнання та вдячності за вклад працівника в успіх компанії.

Додаткова заробітна плата становить 20 % від основної:

$$84000{,}00 * 0{,}20 = 16800{,}00 \text{ грн.}$$

Розмір ЄСВ визначається відсотком від заробітної плати працівника або доходу підприємця. Станом на 2023 рік, стандартна ставка ЄСВ становить 22% від фонду заробітної плати працівника:

$$100800{,}00 * 0{,}22 = 22176{,}00 \text{ грн.}$$

Після розрахунку витрат на заробітну плату команди, наступним важливим елементом, який необхідно врахувати, є витрати на електроенергію. Ці витрати розраховуються на основі споживаної потужності комп’ютерів, іншої периферійної техніки та тарифу на електроенергію.

Оцінки витрат проведемо із використанням комп’ютера та іншої периферійної техніки з енергоспоживанням 0,65 кВт/год. Вартість електроенергії, а саме однієї кіловат-години (кВт/год), становить 1,68 грн. Тобто за кожну годину роботи споживається 0,65 кВт/год електроенергії вартістю 1,68 грн.

Тривалість використання електроенергії впродовж розробки:

8 * 23 = 184 год.

Витрати на електроенергію визначаються за формулою:

$$К_е = Р_п \times Ц_е \times Т,$$

0,65 * 1,68 * 184 = 200,93 грн.

де $Р_п$ – потужність споживана, кВт/год.;

$Ц_е$ – вартість 1 кВт/год.;

Т – кількість годин роботи обладнання.

Вартість обслуговування комп’ютерного обладнання визначається на основі його вартості та періоду використання. Зазвичай, термін служби такого обладнання не перевищує 3 роки, після чого його необхідно замінити. Упродовж року комп'ютерне обладнання використовується 257 робочих днів. Вартість комп’ютерного обладнання в середньому складає 20000,00 грн.

(20000,00 / (3 * 8 * 257)) * 184 = 596,63 грн.

Враховуючи, що проєкт розробляється лише для однієї компанії, загальні витрати на його розробку становлять 123773,56 грн.:

(100800,00 + 22176,00 + 200,93 + 596,63)/1=123773,56 грн.

Проведемо розрахунок прибутку, який ми отримаємо від впровадження проєкту, враховуючи рівень рентабельності в 30 %:

123773,56 * 0,3 = 37132,07 грн.

Визначимо вартість створення сайту без урахування ПДВ:

$$123773,56 + 37132,07 = 160905,62 \text{ грн.}$$

Виконаємо розрахунок суми податку на додану вартість (ПДВ), який складає 20% від вартості без врахування ПДВ:

$$160905,62 * 0,2 = 32181,12 \text{ грн.}$$

Проведемо аналіз вартості web-сайту, враховуючи податок на додану вартість (ПДВ), та визначимо фінальну ціну проєкту:

$$160905,62 + 32181,12 = 193086,75 \text{ грн.}$$

Результати розрахунків наведено у таблиці 6.2.

Таблиця 6.2 – Розрахунок витрат на розробку та ціни web-сайту

Стаття витрат	Сума, грн
Основна заробітна плата	84000,00
Додаткова заробітна плата	16800,00
Єдиний соціальний внесок	18480,00
Витрати на обслуговування ЕОМ	596,63
Витрати на електроенергію	200,93
Собівартість розробки сайту	120077,56
Прибуток	36023,27
Ціна без ПДВ	156100,83
Податок на додану вартість (ПДВ)	31220,17
Ціна з урахуванням ПДВ	187321

Загальна вартість створення web-сайту складає 193086,75 грн. Процес реалізації всіх етапів проєкту займає 23 дні і включає участь команди фахівців, яка складається з Team Lead, UI/UX дизайнера, Frontend розробника, QA інженера, DepOps інженера та контент менеджера. Очікуваний прибуток від реалізації проєкту оцінюється на рівні 37132,07 грн.

ВИСНОВКИ

Метою даної кваліфікаційної роботи є створення web-додатку «Використання сучасного штучного інтелекту у житті», головною ціллю ж було надання доступних та зрозумілих освітніх ресурсів про штучний інтелект. В результаті виконаної роботи web-додаток було успішно створено.

В процесі створення web-додатку «AI & YOU», було виконано ряд важливих кроків. Були чітко визначені цілі та об'єкти web-додатка, що полягали в створенні інтерактивного та освітнього середовища для вивчення та використання штучного інтелекту. Далі було проведено аналіз аудиторії, щоб краще зрозуміти потреби та інтереси майбутніх користувачів. На основі розробленого детального плану проєкту, було розроблено дизайн й основний функціонал та інтерфейс користувача, було проведено тестування, щоб виявити та виправити помилки та проблеми. Нарешті, web-додаток було розміщено на web-сервері, щоб зробити його доступним для користувачів.

Під час створення проєкту було використано ряд технологій та інструментів. Для розробки клієнтської частини використовувався JavaScript з бібліотекою React, а для розробки серверної частини – Node.js з фреймворком Express.js. Статистика використання інтерфейсів взаємодії зі штучним інтелектом та дані галереї спільноти зберігаються MongoDB. Для зберігання та доставки медіа-файлів використовувався Cloudinary, а для розгортання та хостингу web-додатку – AWS.

Результатом цієї роботи є високоякісний web-додаток, який дозволяє користувачам вивчати та використовувати штучний інтелект. Він не лише допомагає людям краще розуміти AI, але і дозволяє їм використовувати ці технології в своїх професійних та особистих сферах життя. Web-додаток також може служити важливим ресурсом для підготовки спеціалістів в області AI, що може сприяти розвитку цієї галузі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Методичні вказівки з виконання кваліфікаційної роботи для студентів денної та заочної форми навчання першого (бакалаврського) рівня вищої освіти спеціальності 186 «Видавництво та поліграфія» за освітньою програмою «Видавничо-поліграфічна справа» / В.П. Ткаченко, А.В. Бізюк, О.В. Вовк, І.М. Єгорова, В.Ф. Челомбітько. Харків: ХНУРЕ, 2020. 68 с.

2. Dikianidis A. Deploying your production ready React-Express App to AWS EC2. URL: https://medium.com/@adcanis/deploying-your-production-ready-react-express-app-to-aws-ec2-7094c7172b8a (дата звернення: 21.05.2023).

3. FreeCodeCamp. How to Set Up VSCode for Your React Projects. URL: https://www.freecodecamp.org/news/vscode-react-setup/ (дата звернення: 13.05.2023).

4. Au-Yeung J. Compress Express.js Responses with the Compression Middleware. JavaScript in Plain English. URL: https://javascript.plainenglish.io/compress-express-responses-with-the-compression-middleware-e9d784065065 (дата звернення: 19.05.2023).

5. Gupta M. 22 React Performance Optimization Techniques. URL: https://javascript.plainenglish.io/https-medium-com-mayank-gupta-6-88-21-performance-optimizations-techniques-for-react-d15fa52c2349 (дата звернення: 19.05.2023).

6. React. Quick Start – React. URL: https://react.dev/learn (дата звернення: 14.05.2023).

7. Barros S. How to create your first MERN (MongoDB, Express JS, React JS and Node JS) Stack. URL: https://medium.com/swlh/how-to-create-your-first-mern-mongodb-express-js-react-js-and-node-js-stack-7e8b20463e66 (дата звернення: 14.05.2023).

8. Section.io. How to Set up a Node.js Express Server for React. URL: https://www.section.io/engineering-education/how-to-setup-nodejs-express-for-react/ (дата звернення: 15.05.2023).

9. Smashing Magazine. Styling Components In React. URL: https://www.smashingmagazine.com/2020/05/styling-components-react/ (дата звернення: 13.05.2023).

10. Полозова Т.В. Методичні вказівки до виконання економічної частини кваліфікаційної роботи для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 186 Видавництво та поліграфія усіх форм навчання. Харків: ХНУРЕ, 2022. 48 с.