

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра інженерії програмного забезпечення

ПОЯСНЮВАЛЬНА ЗАПИСКА

до бакалаврської роботи

на ступінь вищої освіти бакалавр

на тему: «**РОЗРОБКА ВЕБ-СЕРВІСУ "OWLSTUDY" ДЛЯ
ПРОХОДЖЕННЯ ОСВІТНІХ КУРСІВ НА БАЗІ JS, HTML, CSS**»

Виконав: студент 4 курсу, групи ПД - 42
спеціальності

121 Інженерія програмного забезпечення

(шифр і назва спеціальності)

Білодід Д.В.

(прізвище та ініціали)

Керівник

Бондарчук А.П.

(прізвище та ініціали)

Рецензент

(прізвище та ініціали)

Київ – 2023

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти - «Бакалавр»

Спеціальність підготовки – 121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри
Інженерії програмного
забезпечення

Негоденко О.В.

“ _____ ” _____ 2023
року

ЗАВДАННЯ
НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТА

БІЛОДІД ДАНІЕЛЬ ВАДИМОВИЧ

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка веб-сервісу "OWL study" для проходження освітніх курсів на базі JS, HTML, CSS»

Керівник роботи: Бондарчук Андрій Петрович, д.т.н., професор
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом вищого навчального закладу від «24» лютого 2023 року
№26.

2. Строк подання студентом роботи «1» червня 2023 року

3. Вхідні дані до роботи:

3.1 Науково-технічна література з питань, пов'язаних з програмним забезпеченням щодо розробки веб-сервісу.

3.2 Офіційна документація Figma.

3.3 Офіційна документація Microsoft Visual Studio Code.

4. Зміст розрахунково-пояснювальної записки.

- 4.1 Аналіз актуальності веб-сервісу з проходження освітніх курсів.
- 4.2 Аналіз та вибір інструментів для розробки веб-сервісу.
- 4.3 Проектування веб-сервісу з проходження освітніх курсів.
- 4.4 Висновки.
5. Перелік демонстраційного матеріалу
 - 5.1 Титульний слайд.
 - 5.2 Мета, об'єкт, предмет, наукова новизна дослідження.
 - 5.3 Задачі дипломної роботи
 - 5.4 Аналіз аналогів.
 - 5.5 Вимоги до веб-сервісу.
 - 5.6 Програмні засоби реалізації.
 - 5.6 User Flow.
 - 5.7 Діаграма варіантів використання.
 - 5.8 Персона користувача.
 - 5.9 Архітектура веб-сервісу
 - 5.10 Екрані форми.
 - 5.11 Апробація результатів дослідження.
 - 5.12 Висновки.
6. Дата видачі завдання
«25» лютого 2023р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	25.02.2023	Виконано
2	Дослідження аналогів та актуальності веб-сервісу	14.04.2023	Виконано
3	Аналіз та вибір інструментів для розробки веб-сервісу	19.04.2023	Виконано
4	Проектування та реалізація	29.04.2023	Виконано
5	Вступ, висновки, реферат	08.04.2023	Виконано
6	Розробка обов'язкових демонстраційних матеріалів	10.05.2023	Виконано
7	Попередній захист роботи	19.05.2023	Виконано
8	Здача роботи	01.06.2023	Виконано

Студент

Білодід Д.В.

(підпис)

(прізвище та ініціали)

Керівник роботи

Бондарчук А.П.

(підпис)

(прізвище та ініціали)

РЕФЕРАТ

Текстова частина бакалаврської роботи с.?, 1 табл., ? рис., ? джерел. Ключові слова: середовище розробки, веб-сервіс, Visual Studio Code, Figma, Освітні курси.

Об'єкт дослідження – процес взаємодії з інформаційною системою при проходженні освітніх курсів.

Предмет дослідження – веб технології для розробки електронних сервісів.

Мета роботи – забезпечення якісного функціонування інформаційних освітніх платформ застосовуючи веб технології JS, HTML, CSS.

Наукова новизна – веб-сервіс може використовувати різноманітні інтерактивні технології, такі як відеоуроки, інтерактивні тести та інші, для забезпечення взаємодії користувача з матеріалами курсу та засвоєння матеріалів за допомогою тестів.

У дипломному проекті був проведений аналіз ринку веб-сервісів для проходження освітніх курсів. Аналізом виявлені переваги та недоліки середовищ розробки. Проаналізовано особливості розробки веб-сервісів в цій сфері.

Веб-сервіс було створено за допомоги мови розмітки гіпертексту HTML, об'єктно-орієнтованою прототипною мовою програмування JavaScript та мови стилю CSS. Веб-сервіс був написаний у середовищі розробки Microsoft Visual Studio Code. Для створення макету веб-сервісу було використано Figma

Веб-сервіс може допомогти користувачу навчатися на відстані та розвивати професійні навички, забезпечує можливість для самооцінки також відстеження прогресу виконання курсу.

ЗМІСТ

ВСТУП.....	11
1. АНАЛІЗ СТРУКТУРИ ІСНУЮЧИХ ВЕБ-СЕРВІСІВ ДЛЯ ПРОХОДЖЕННЯ ОСВІТНІХ КУРСІВ.....	12
1.1 Аналіз та загальна характеристика веб-сервісу для проходження освітніх курсів.....	12
1.2 Аналіз вже існуючих веб-сервісів з освітніми курсами	13
1.3 Актуальність проекту.....	17
1.4 Перспективи галузі та проблеми розвитку	18
Висновок до першого розділу.....	18
2. ДОСЛІДЖЕННЯ ІНСТРУМЕНТІВ ПРОЕКТУВАННЯ ВЕБ-СЕРВІСУ ДЛЯ ПРОХОДЖЕННЯ ОСВІТНІХ КУРСІВ	20
2.1 Аналіз та обґрунтування вибору середовища розробки проекту	20
2.1.1 Visual Studio Code.....	21
2.1.2 Atom.....	23
2.1.3 Figma	25
2.1.4 Sketch.....	27
2.2 Загальний алгоритм реалізації проекту	28
2.3 Функціонал продукту	29
2.4 Аналіз мов для розробки веб-сервісу	31
2.4.1 Мова розмітки гіпертексту HTML	31
2.4.2 Мова стилів CSS	33
2.4.3 Об'єктно-орієнтована прототипна мова програмування Javascript	34
2.5 Підготовка матеріалів	35
Висновок до другого розділу.....	42
3. ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБ-СЕРВІСУ ДЛЯ ПРОХОДЖЕННЯ ОСВІТНІХ КУРСІВ	44
3.1 Прототипування веб-сервісу. UI та UX дизайн.....	44
3.2 Створення логотипу веб-сервісу.....	47
3.3 Аналіз використаної типографії.....	48
3.4 Теорія кольору.....	50
3.5 Компоненти веб-сервісу та їх стан.....	53
Висновок до третього розділу	56
4. ОПИС ВИКОРИСТАННЯ ТА ТЕСТУВАННЯ ВЕБ-СЕРВІСУ	58
4.1 Сценарій роботи з веб-сервісом	58
Реалізований веб-сервіс відповідає таким вимогам:	68
1. Пошукова система	68
2. Інтерфейс, що відповідає сучасним тенденціям та забезпечує зручність	

використання	68
3. Наявність тестування.....	68
4. Персональний профіль користувача	68
5. Підтримка різних мов.....	68
Висновок.....	69
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	71
ДОДАТОК А	72

ВСТУП

Об'єкт дослідження – веб-сервіс для проходження освітніх курсів.

Предмет дослідження – метод розробки веб-сервісу для проходження освітніх курсів.

Мета роботи – розробка веб-сервісу для проходження освітніх курсів з використанням мови розмітки гіпертексту HTML та об'єктно-орієнтованою прототипною мовою програмування JavaScript.

Наукова новизна – веб-сервіс може використовувати різноманітні інтерактивні технології, такі як відеоуроки, інтерактивні тести та інші, для забезпечення взаємодії користувача з матеріалами курсу та засвоєння матеріалів за допомогою тестів.

У дипломному проекті був проведений аналіз ринку веб-сервісів для проходження освітніх курсів. Аналізом виявлені переваги та недоліки середовищ розробки. Проаналізовано особливості розробки веб-сервісів в цій сфері.

Веб-сервіс було створено за допомоги мови розмітки гіпертексту HTML, об'єктно-орієнтованою прототипною мовою програмування JavaScript та мови стилю CSS. Веб-сервіс був написаний у середовищі розробки Microsoft Visual Studio Code. Для створення макету веб-сервісу було використано Figma

Веб-сервіс може допомогти користувачу навчатися на відстані та розвивати професійні навички, забезпечує можливість для самооцінки також відстеження прогресу виконання курсу.

1. АНАЛІЗ СТРУКТУРИ ІСНУЮЧИХ ВЕБ-СЕРВІСІВ ДЛЯ ПРОХОДЖЕННЯ ОСВІТНІХ КУРСІВ.

1.1 Аналіз та загальна характеристика веб-сервісу для проходження освітніх курсів

Веб-сервіс - це програмна система, яка надає функціональність через мережу Інтернет. Веб-сервіси можуть бути різноманітними, такими як веб-додатки, веб-сайти, платформи електронного навчання, соціальні мережі, електронні магазини, API (інтерфейси програмування додатків) та багато інших.

Веб-сервіс може бути розгорнутий на веб-сервері та доступний користувачам через браузер або інші клієнтські програми. Користувачі можуть взаємодіяти з веб-сервісом, використовуючи його функціональність, таку як реєстрація, вхід, пошук, відправлення даних та отримання відповідей.

Веб-сервіси можуть бути використані для різних цілей, включаючи комунікацію, обмін даними, автоматизацію бізнес-процесів, надання послуг, навчання, розваги та багато інших. Вони можуть бути створені розробниками програмного забезпечення та використовуватися різними користувачами, включаючи бізнеси, урядові організації, освітні заклади, веб-мастери та кінцевих користувачів.

Веб-сервіс для проходження освітніх курсів - це онлайн-платформа або додаток, який надає користувачам можливість вивчати різні курси, навчальні матеріали та ресурси через Інтернет. Веб-сервіси для проходження освітніх курсів можуть бути розгорнуті на веб-сервері та бути доступні користувачам з різних пристроїв, таких як комп'ютери, планшети, смартфони тощо.

Функціональність веб-сервісів для проходження освітніх курсів може включати:

1. Реєстрацію користувачів: Можливість створення облікових записів користувачів, встановлення профілів, керування власними даними та налаштуваннями.
2. Навчальний контент: Доступ до навчальних матеріалів, таких як відеоуроки, текстові матеріали, тести, завдання, інтерактивні вправи та інші ресурси, пов'язані з курсами.
3. Організація курсів: Можливість створення, редагування та організації курсів, включаючи структуру курсу, лекції, модулі, теми, завдання, оцінювання та інші параметри.
4. Взаємодія з викладачами: Можливість спілкування з викладачами, отримання фідбеку, задавання питань та взаємодії з іншими учасниками курсу.
5. Оцінювання та моніторинг: Відстеження прогресу користувачів, видача оцінок, статистика виконання завдань та інші функції моніторингу.
6. Сертифікація: Видача сертифікатів користувачам після успішного завершення курсів.

1.2 Аналіз вже існуючих веб-сервісів з освітніми курсами

Coursera (<https://www.coursera.org/>): Coursera - це веб-платформа, яка співпрацює з університетами та організаціями по всьому світу, щоб надавати відкритий доступ до курсів в різних галузях з відомих університетів та провідних компаній.

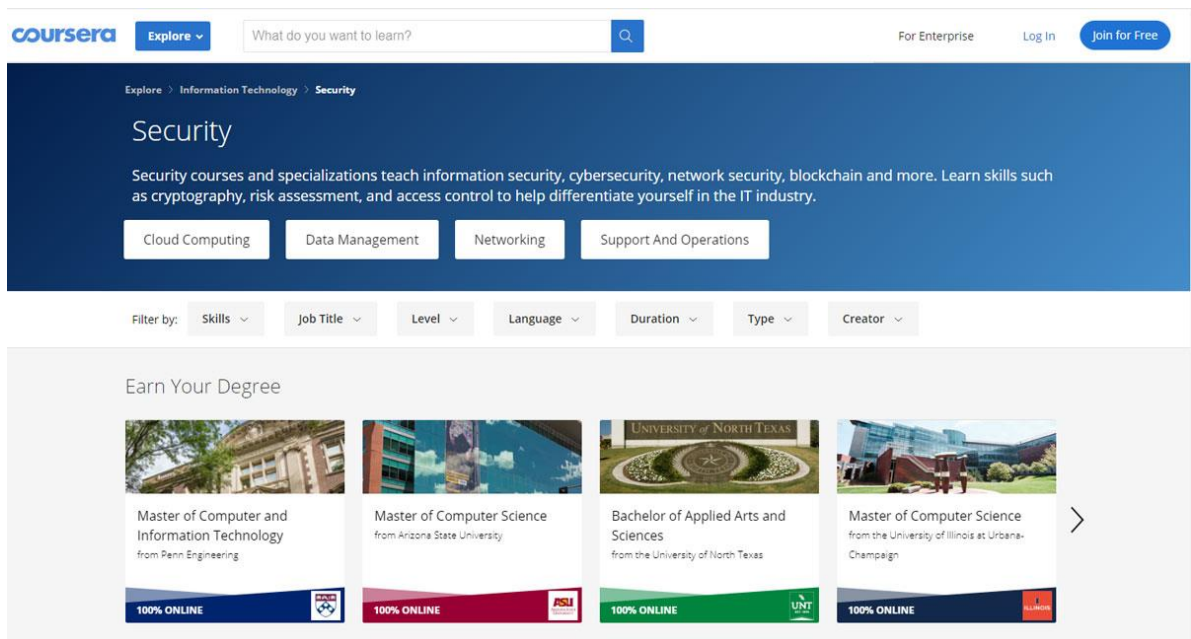


Рисунок 1.1 – Landing Page Coursera

Udemy (<https://www.udemy.com/>): Udemy - це платформа, яка дозволяє керівникам курсів публікувати свої власні курси в різних галузях, таких як програмування, музика, мистецтво, бізнес та інші.

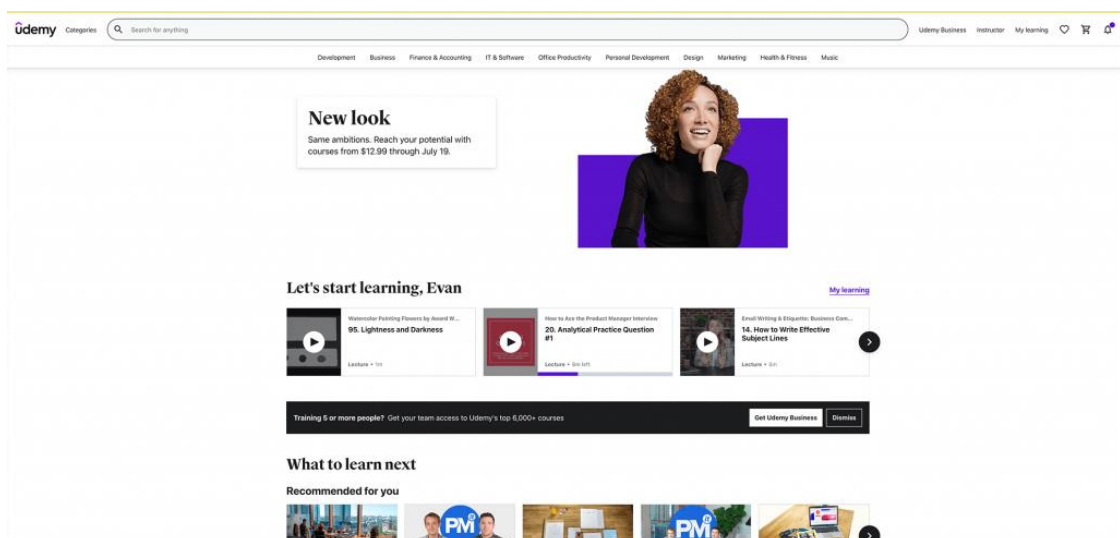


Рисунок 1.2 – Landing Page Udemy

LinkedIn Learning (<https://www.linkedin.com/learning/>): LinkedIn Learning - це платформа, яка надає доступ до більше ніж 16 000 курсів з різних галузей, таких як бізнес, техніка, маркетинг, дизайн та багато інших.

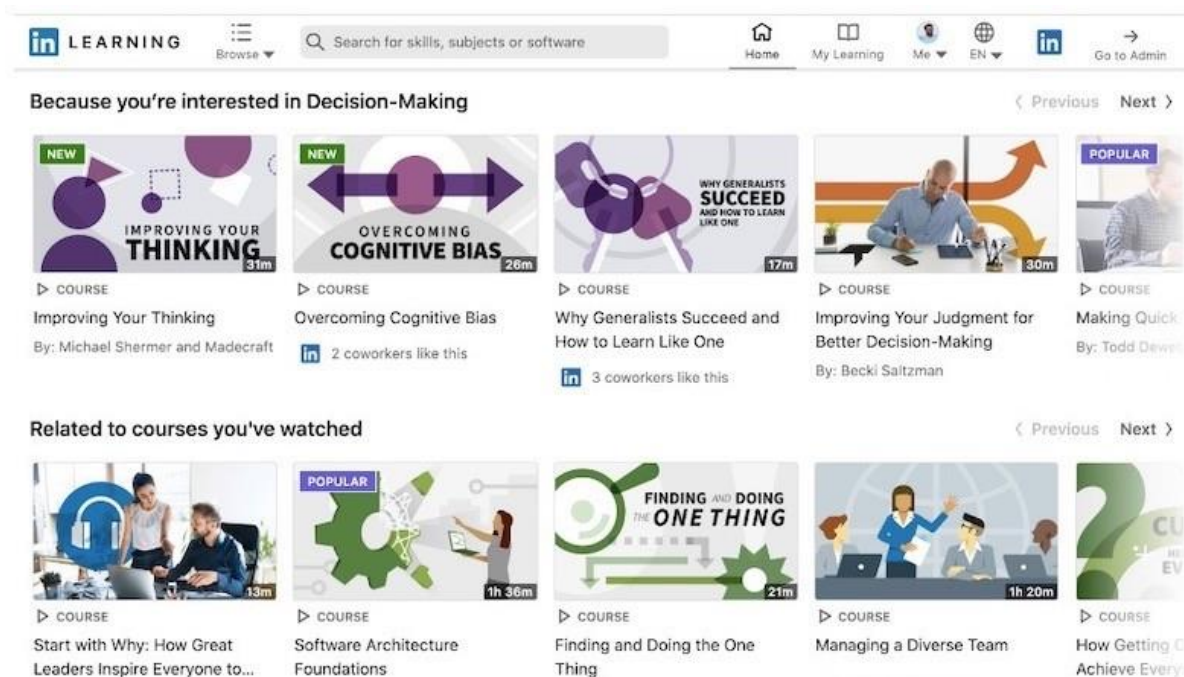


Рисунок 1.3 – Landing Page LinkedIn Learning

edX (<https://www.edx.org/>): edX - це веб-платформа, яка співпрацює з університетами, компаніями та організаціями, щоб надавати доступ до більше ніж 3000 курсів з різних галузей з відомих університетів та організацій.

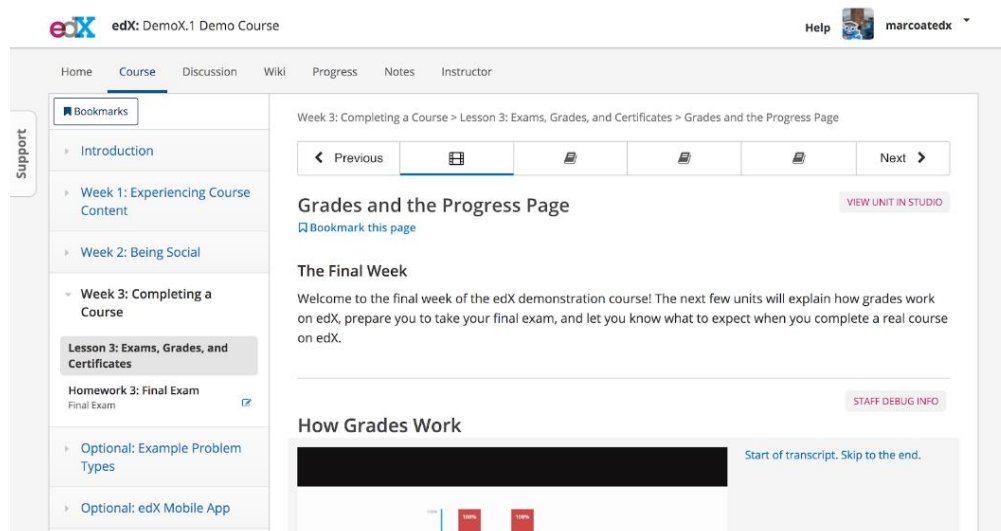


Рисунок 1.4 – Landing Page edX

Таблиця 1.1 – Зведені результати аналізу характеристик веб-сервісів для проходження освітніх курсів

Розглянуті веб-сервіси	Coursera	Udemy	LinkedIn Learning	edX
Мобільна версія	+	+	+	+
Наявність тестів	+	+	+	+
Доступність курсів	безкоштовні	від 449€	Від 559€	безкоштовні
Зручність інтерфейсу	-	+	+	-
Відстеження прогресу	+	+	+	-
Локалізація на українську мову	+	-	-	-
Можливість завантажувати власні курси	-	+	-	+

1.3 Актуальність проекту

Глобальна криза COVID-19: Пандемія COVID-19 змінила підхід до навчання, змушуючи багатьох людей шукати онлайн-рішень для здобуття освіти. Веб-сервіси для проходження освітніх курсів стали важливим інструментом для навчання з використанням віддалених технологій.

Нові технології: З ростом технологій та доступу до Інтернету, веб-сервіси стали більш доступними та зручними для навчання. Вони дозволяють студентам навчатися власним темпом, в будь-який зручний час і місце, використовуючи різноманітні мультимедійні ресурси.

Постійний розвиток ринку праці: Зміни на ринку праці вимагають від працівників постійно оновлювати свої навички та знання. Веб-сервіси для проходження освітніх курсів дозволяють студентам оновлювати свої навички та отримувати нові знання, щоб відповідати вимогам сучасного ринку праці.

Глобалізація та міжнародна освіта: Онлайн-навчання дозволяє студентам з різних країн та регіонів отримувати доступ до високоякісної освіти від провідних університетів та освітніх установ світу. Веб-сервіси для проходження освітніх курсів можуть надавати можливості міжнародного співпраці та обміну знаннями.

Персоналізоване навчання: Веб-сервіси для проходження освітніх курсів можуть надавати персоналізоване навчання, адаптоване до індивідуальних потреб та рівня знань студента. Це дозволяє краще враховувати особисті можливості, ритм та стиль навчання кожного студента, що робить процес більш ефективним.

Розширення доступності освіти: Веб-сервіси для проходження освітніх курсів можуть бути використані для розширення доступності освіти, зокрема для тих, хто має обмежені можливості або не має доступу до традиційної освіти. Це може включати людей з віддалених регіонів, з обмеженими руховими можливостями, або тих, хто має інші обмеження, які ускладнюють фізичний

доступ до навчальних установ.

1.4 Перспективи галузі та проблеми розвитку

Об'єкт проекту веб-сервісу для проходження освітніх курсів - це користувачі, які бажають отримати додаткові знання та навички в різних областях. Це можуть бути студенти, професіонали, працівники компаній, навчальні заклади та інші особи, які зацікавлені в підвищенні своєї освіти.

Предмет проекту - це навчання та розвиток користувачів в різних сферах знань, таких як бізнес, техніка, наука, мистецтво, мови, технології та інші. Веб-сервіс може пропонувати різні типи курсів, такі як відеолекції, інтерактивні вправи, тести, домашні завдання, форуми для спілкування з іншими учасниками тощо.

Мета проекту - це надати можливість користувачам отримати якісну освіту та розвиватися в своїх обраних областях. Метою може бути також забезпечення доступу до освіти без обмежень місця та часу, створення спільноти навчальних співтовариств, сприяння саморозвитку та самоосвіті, підвищення професійних навичок, підтримка навчання на відстані та інші.

1.5 Висновок до першого розділу.

В цьому розділі було акцентовано увагу на те, що веб-сервіси відіграють значну роль у нашому суспільстві, вони є важливою складовою частиною сучасного технологічного прогресу, веб-сервіси використовуються в різних сферах таких як освіта, магазини та інше. Ми розглянули веб-сервіс для проходження освітніх курсів.

Основні критерії веб-сервісу для проходження освітніх курсів є:

- Навчальний контент
- Організація курсів

- Взаємодія з викладачами
- Оцінювання та моніторинг
- Сертифікація

Наступним кроком було проаналізовані вже існуючі веб-сервіси для проходження освітніх курсів, такі як:

- Coursera
- Udemy
- LinkedIn Learning
- edX

За допомогою таблиці було проведено аналіз позитивних та негативних аспектів технологій, використаних під час створення цих веб-сервісів.

Додатково була оцінена актуальність створення веб-сервісу, який надає можливість проходження освітніх курсів.

Сформована мета проекту - надати можливість користувачам отримати якісну освіту та розвиватися в своїх обраних областях.

2. ДОСЛІДЖЕННЯ ІНСТРУМЕНТІВ ПРОЕКТУВАННЯ ВЕБ-СЕРВІСУ ДЛЯ ПРОХОДЖЕННЯ ОСВІТНІХ КУРСІВ

2.1 Аналіз та обґрунтування вибору середовища розробки проекту

Для виконання проекту важливо мати два різні середовища. Одне призначене для розробки початкового макету веб-сайту, включаючи дизайн та інтерфейс, а інше - для розробки власне коду веб-сайту відповідно до розробленого макету.

Середовище розробки (або IDE, Integrated Development Environment) - це комплексне програмне забезпечення, яке надає розробникам зручний і ефективний інтерфейс для розробки програмного забезпечення. Це включає ряд інструментів, таких як редактори коду, компілятори/інтерпретатори, налагоджувачі (debuggers), системи контролю версій, засоби автоматичної побудови (build tools) та інші допоміжні функції, призначені для забезпечення швидкого і ефективного процесу розробки програмного забезпечення.

Середовище розробки допомагає розробникам виконувати такі дії, як створення, редагування, компіляція, налагодження та тестування коду, а також керування версіями і роботою з різними інструментами та бібліотеками. Використання середовища розробки може допомогти підвищити продуктивність розробника, спростити розробку програмного забезпечення та полегшити процес розробки.

Найбільш використовуваними середовищами для розробки вважаються Visual Studio Code та Atom.

Макет проекту - це візуальна модель або прототип проекту, яка відображає його загальний вигляд та функціональність. Макет може бути відтворенням веб-сторінки, додатку, програми або іншого проекту, зробленого за допомогою спеціальних програм або ручної роботи. Макет допомагає команді розробників

та клієнтам краще зрозуміти остаточний вигляд та функціональність проекту перед його реалізацією. Макет може включати елементи дизайну, колірну палітру, компоненти і інші деталі, які допомагають відобразити остаточний вигляд проекту.

Більш використовуваними програмами для розробки макетів вважають Figma та Sketch.

2.1.1 Visual Studio Code

Visual Studio Code (VS Code) це безкоштовний редактор коду, розроблений компанією Microsoft, який використовується для розробки програмного забезпечення. Він є одним з найпопулярніших редакторів коду серед програмістів та розробників з відкритим вихідним кодом.

VS Code має високу швидкість роботи, має багатий функціонал та багато розширень, що дозволяє налаштувати робоче середовище під потреби розробника. Він підтримує ряд мов програмування, включаючи JavaScript, Python, C++, C#, Java, Go, Ruby, Swift та багато інших.

Деякі основні особливості Visual Studio Code включають розширену підтримку автодоповнення, можливість редагування коду на віддалених серверах, вбудовану систему керування версіями, зручний інтерфейс користувача з можливістю налаштування тем та кольорових схем, відладчик, можливість розширення функціональності за допомогою розширень, підтримку Docker та багато інших корисних інструментів для розробки програмного забезпечення.

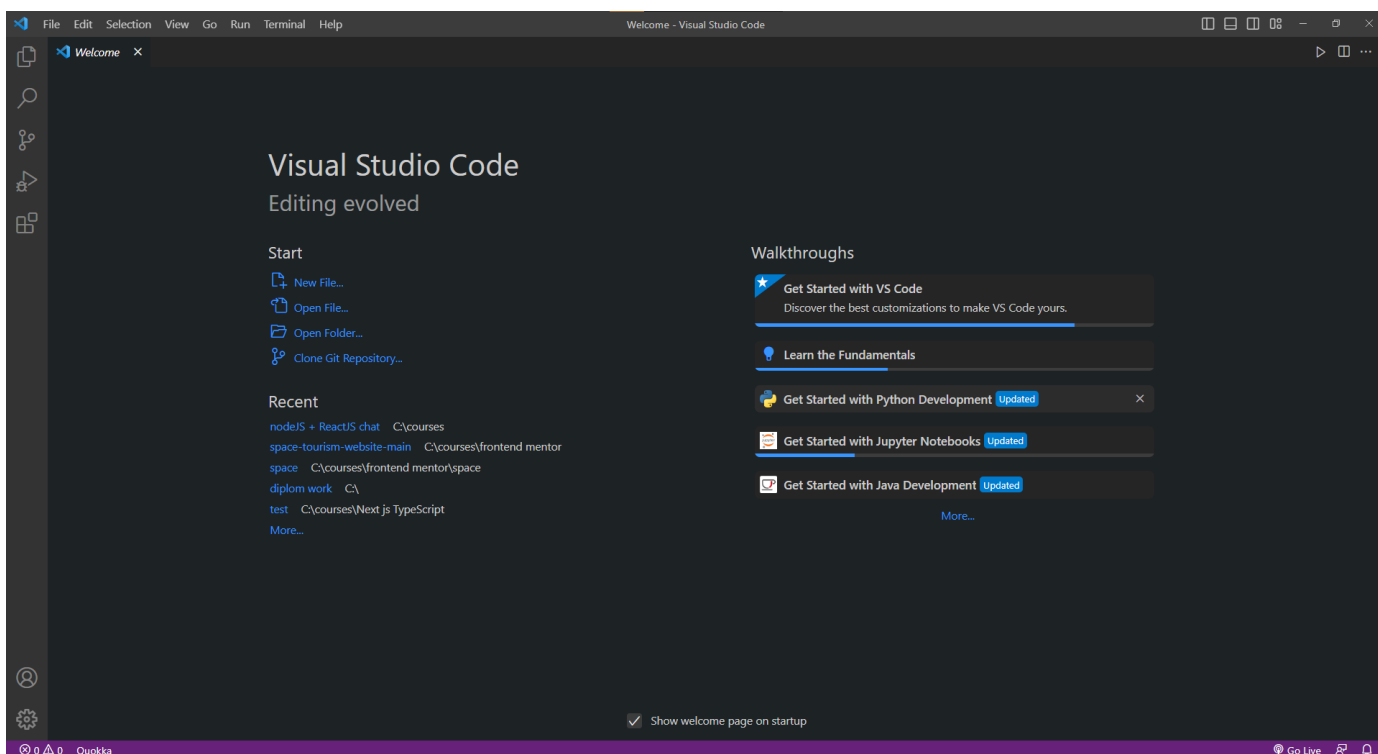


Рисунок 2.1 – Visual Studio Code

Функціональність: VS Code має більший функціонал порівняно з Atom. Він має вбудований відладчик, підтримку редагування коду на віддалених серверах, вбудований систему керування версіями, велику кількість розширень та багато інших корисних інструментів для розробки. Atom також має ряд розширень, але їхній функціонал може бути менший порівняно з VS Code.

Швидкість: VS Code відомий своєю високою швидкістю роботи, особливо при роботі з великими проектами. Atom також є швидким, але може бути менш продуктивним при роботі з великими обсягами коду порівняно з VS Code.

Розширюваність: Обидва редактори мають велику кількість розширень, які дозволяють розширити їхню функціональність. Однак, екосистема розширень VS Code вважається більш розгалуженою та ширшою, з великою кількістю розширень, розроблених спільнотою розробників.

Підтримка та оновлення: VS Code має активну спільноту розробників та володіє швидким циклом оновлень, що дозволяє отримувати регулярні оновлення з новими функціями та виправленнями помилок. Atom також має активну спільноту, але оновлення можуть бути менш регулярними порівняно з VS Code.

2.1.2 Atom

Atom - це відкритий текстовий редактор, розроблений компанією GitHub. Він призначений для редагування, розробки та налагодження коду. "Середа розробки" (Integrated Development Environment або IDE) - це програмний продукт, який надає розробникам інструменти та можливості для зручного написання, редагування, компіляції, налагодження та відлагодження коду.

Atom може бути використаний як основна середа розробки, оскільки він має ряд функцій, таких як кольорове підсвічування синтаксису, автодоповнення, керування версіями та інші, що роблять його зручним для розробки програмного забезпечення. Однак, на відміну від типових IDE, таких як Visual Studio або IntelliJ IDEA, Atom не має всіх функцій, що зазвичай пов'язуються з повноцінними середами розробки. Atom більше фокусується на простоті, розширюваності та співпраці зі спільнотою розробників.

Atom має велику спільноту розробників, які створюють розширення (плагіни) для покращення функціональності редактора. Це дозволяє розширювати можливості Atom та налаштовувати його під власні потреби. Atom також інтегрується з рядом інших інструментів розробки, таких як Git, що полегшує роботу зі сховищами коду та сприяє ефективнішому управлінню проектами.

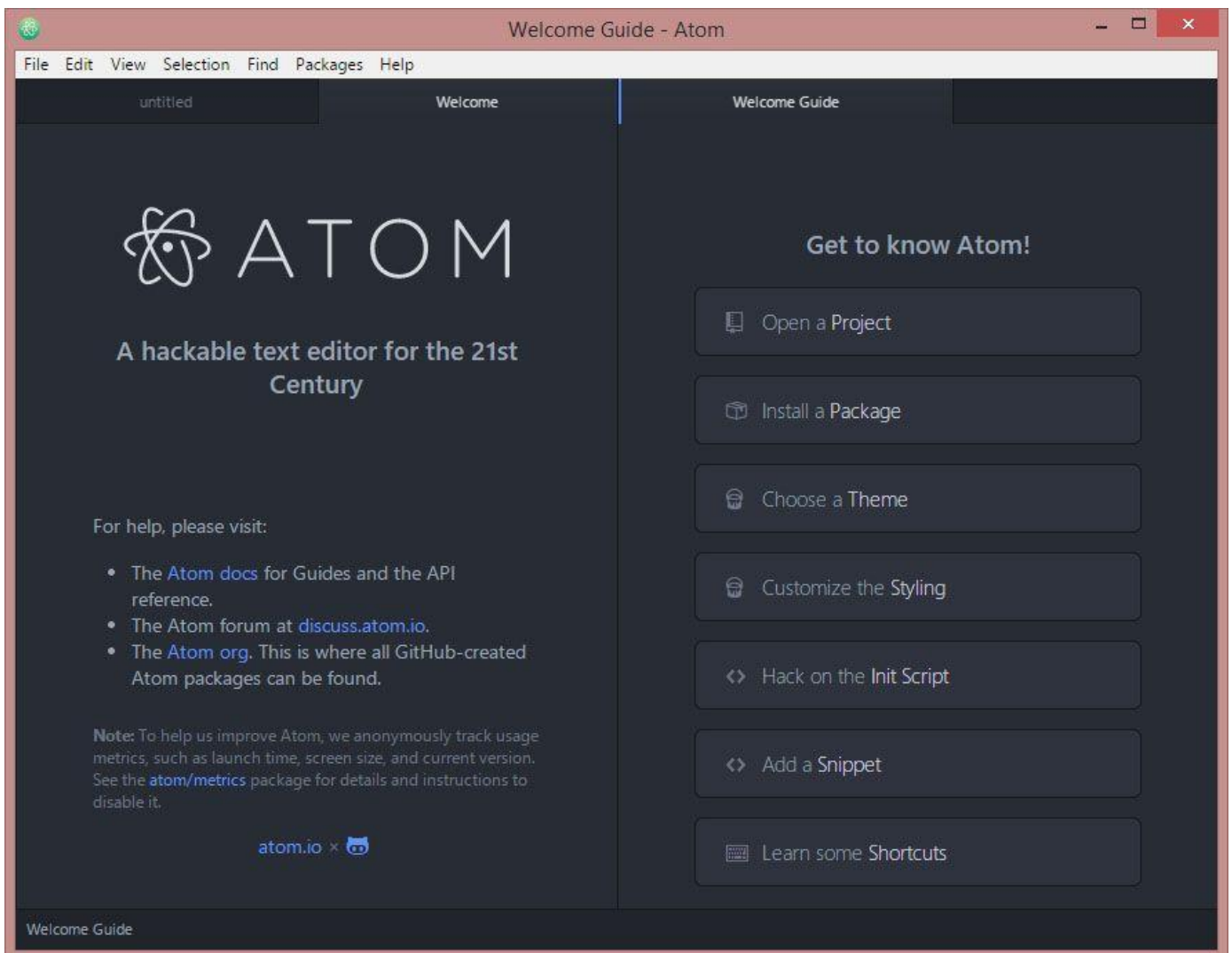


Рисунок 2.2 – Atom

Відтворюваність: Atom відомий своєю відтворюваністю та можливістю налаштувати його за власними потребами. Він використовує мову програмування JavaScript та має вбудований рушій Electron, що дозволяє розширювати його функціонал за допомогою різних плагінів та пакетів. Це дозволяє розробникам створювати власні функції, налаштувати інтерфейс та робочий процес редактора на свій смак.

Простота: Atom має простий та легкий використовуваний інтерфейс. Він може бути відмінним варіантом для початківців, оскільки має менш складний інтерфейс порівняно з VS Code.

Загалом, обидва редактори мають свої переваги та недоліки, і вибір між ними залежить від ваших власних потреб та вподобань. VS Code відомий своєю широкою функціональністю, швидкістю та великою спільнотою розробників, тоді як Atom відомий своєю відтворюваністю та налаштуваннями. Обидва редактори можуть бути ефективними інструментами для розробки програмного забезпечення, залежно від вашого вибору.

2.1.3 Figma

Figma - це веб-програма для дизайну і прототипування інтерфейсів користувача (UI) та веб-дизайну. Вона розроблена компанією Figma Inc. та була випущена в 2016 році. Figma використовується професіоналами в галузі дизайну, веб-розробки, продуктового дизайну, мобільного дизайну та інших візуальних проєктів.

Figma пропонує можливості редагування в реальному часі, співпрацю в реальному часі та хмарне зберігання проєктів. Завдяки Figma можна створювати дизайн-системи, розробляти макети, створювати взаємодіє, експортувати графічні елементи та співпрацювати з командою на проєкті. Figma також має велику кількість розширень, що дозволяють розширити можливості програми та автоматизувати рутинні задачі.

Одним з відмінних переваг Figma є можливість співпрацювати в реальному часі, дозволяючи кільком користувачам одночасно працювати над одним проєктом та бачити зміни в режимі реального часу. Крім того, Figma підтримує створення прототипів з можливістю взаємодії, що дозволяє створювати живі демонстрації функціональності дизайну. Figma також забезпечує автоматичне зберігання змін, що дозволяє відновлювати попередні версії проєктів та відслідковувати зміни у часі.

Figma є популярним інструментом для роботи в команді над дизайн-проектами, особливо в режимі дистанційної роботи, завдяки можливостям співпраці та хмарного зберігання.

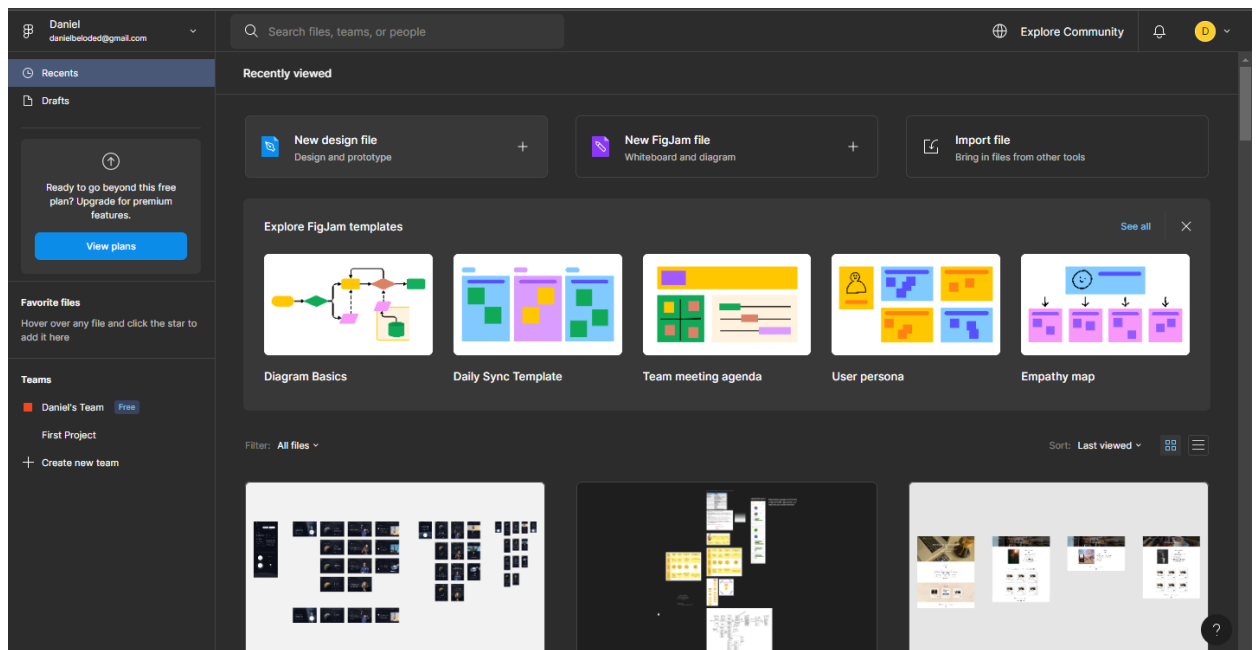


Рисунок 2.3 – Figma

Операційна система: Figma - це веб-додаток, що працює в браузері, і може бути використаний на будь-якій операційній системі, такі як Windows, MacOS, Linux. Sketch - це додаток, розроблений спеціально для MacOS.

Співпраця в команді: Figma має розширені можливості співпраці в команді, такі як одночасне редагування, можливість залишати коментарі та відстежувати зміни в реальному часі. Sketch також має деякі можливості співпраці, але вони обмежені в порівнянні з Figma.

Доступність: Figma має безкоштовний тарифний план з обмеженими можливостями, що робить його доступним для студентів, фрілансерів та невеликих команд. Sketch, з іншого боку, має платний ліцензійний план з високою ціною, що може бути більш складним для бюджетного використання

2.1.4 Sketch

Sketch - це програмне забезпечення для розробки графічного дизайну, яке широко використовується в індустрії дизайну інтерфейсів користувача (UI/UX). Воно створено спеціально для розробників веб-додатків, мобільних додатків та інших цифрових продуктів, які вимагають високоякісного графічного дизайну.

Sketch дозволяє створювати векторні графічні елементи, такі як кнопки, іконки, банери, фони та інші, що можуть бути масштабовані без втрати якості.

Sketch дозволяє створювати системи символів, які дозволяють повторно використовувати елементи дизайну в різних місцях проекту, що сприяє єдності та консистентності дизайну.

Sketch дозволяє експортувати графічні елементи в різних форматах, таких як PNG, SVG, JPG, забезпечуючи можливість використання їх в різних контекстах.

Sketch підтримує велику кількість розширень та інтеграцій з іншими популярними інструментами для дизайну, розробки та взаємодії.

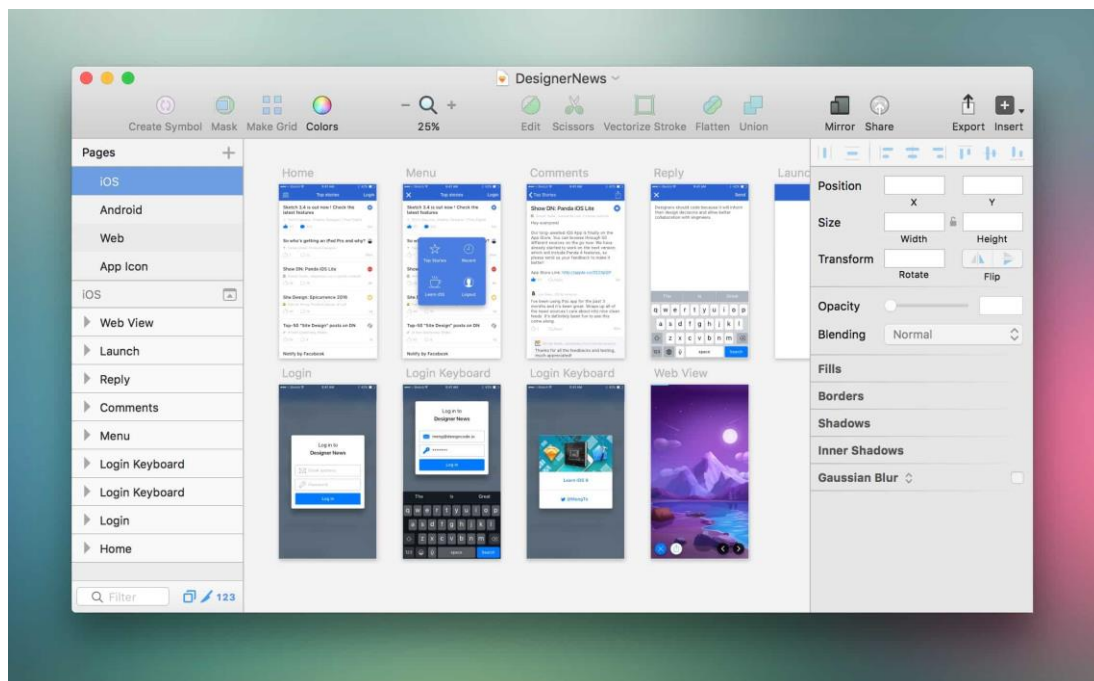


Рисунок 2.4 – Sketch

Sketch розроблений для Mac-платформи, що дає йому перевагу в тому, що може бути більш оптимізованим та більший мірі інтегрованим з екосистемою Apple. Це може бути корисно для дизайнерів, які працюють виключно на платформі Mac.

Sketch відомий своєю швидкістю та продуктивністю, особливо при роботі з великими макетами та векторними зображеннями. Він пропонує ряд оптимізованих інструментів, таких як векторний редактор, макетні групи, символи та багато інших, що роблять роботу швидшою та ефективнішою.

Один зі специфічних аспектів Sketch - це можливість зберігати файли проектів на локальному комп'ютері або на локальному сервері, що може бути зручним для деяких дизайнерів, особливо тих, хто працює з конфіденційними або приватними даними.

Обидві програми мають свої унікальні особливості та придатність в різних випадках використання. Вибір між Figma та Sketch залежить від ваших потреб, вимог та пріоритетів, таких як співпраця в команді, доступність, функціональність та інтеграції з іншими інструментами, а також ваші особисті вподобання та знайомство з інтерфейсами цих програм.

2.2 Загальний алгоритм реалізації проекту

Основний алгоритм реалізації проекту веб-сервісу для проходження освітніх курсів може включати наступні кроки:

1. **Визначення вимог проекту:** Взаємодія з клієнтом або замовником, визначення функціональності сервісу, встановлення технічних вимог, збір вимог до інтерфейсу користувача, встановлення обмежень та умов реалізації проекту.
2. **Розробка дизайну інтерфейсу користувача:** Створення дизайну веб-інтерфейсу, розробка макетів, визначення структури та навігації, розробка дизайну сторінок та елементів інтерфейсу.

3. **Розробка функціональності:** Реалізацію функцій авторизації та автентифікації користувачів, створення профілів користувачів, можливість вибору та проходження курсів, відстеження прогресу проходження курсів, збереження та аналіз даних користувачів.
4. **Розробка адміністративної панелі:** Реалізація функціоналу для адміністраторів системи, такого як керування користувачами, курсами, виставлення оцінок, створення та редагування вмісту курсів.
5. **Інтеграція з іншими сервісами:** Взаємодія з іншими веб-сервісами, такими як системи оплати, сервіси електронної пошти, системи відео конференцій, для забезпечення різноманітної функціональності, такої як оплата за курси, сповіщення користувачів, вебінари тощо.
6. **Тестування та відлагодження:** Проведення ряду тестів для впевненості в якості роботи веб-сервісу, виявлення та виправлення можливих помилок, оптимізація продукту для кращої продуктивності та відповідності вимогам.

Задачі та цілі проекту було визначено у першому розділі, спрямованому на розробку веб-сервісу для проходження освітніх курсів.

2.3 Функціонал продукту

Функціонал веб-сервісу для проходження освітніх курсів може включати різноманітні функції, які допоможуть користувачам ефективно вивчати матеріал, взаємодіяти з викладачами та іншими учасниками курсів, та відстежувати свій прогрес. Ось кілька можливих функцій:

Реєстрація та автентифікація: Користувачі можуть створювати акаунти веб-сервісу, реєструватися на курси та увійти в свій обліковий запис, використовуючи електронну пошту або соціальні мережі.

Каталог курсів: Користувачі можуть переглядати доступні курси в

каталозі, переглядати описи, деталі та вибирати курси, які їх цікавлять.

Онлайн-лекції: Курси можуть містити відео-або аудіо-лекції, які користувачі можуть переглядати або слухати в режимі реального часу або за запланованим графіком.

Матеріали курсу: Користувачі можуть мати доступ до різних матеріалів курсу, таких як тексти, презентації, вправи, завдання та інші ресурси, які допомагають вивчати матеріал більш детально.

Взаємодія з викладачами та іншими учасниками: Користувачі можуть взаємодіяти з викладачами та іншими учасниками курсу, наприклад, за допомогою форумів, чатів або коментарів, де вони можуть обговорювати матеріал курсу, задавати питання та обмінюватися ідеями.

Відстеження прогресу: Користувачі можуть відстежувати свій прогрес виконання курсу, такий як виконані завдання, оцінки, пройдені розділи, і відстежувати свої досягнення.

Оцінювання та звітність: Викладачі можуть оцінювати роботи, вправи та завдання, надавати повідомлення про успішне виконання та надавати звіти про прогрес користувачів.

Тестування та оцінювання: Система може включати тестування та оцінювання користувачів з метою перевірки рівня їхнього знання після вивчення матеріалу.

Сертифікати: Користувачі можуть отримувати сертифікати про успішне закінчення курсу, якщо вони успішно виконали всі вимоги та завдання курсу.

2.4 Аналіз мов для розробки веб-сервісу

2.4.1 Мова розмітки гіпертексту HTML

Для розробки веб-сервісу було обрано мову розмітки гіпертексту

HTML - є мовою розмітки, яка використовується для створення веб-сторінок та їх структурування. HTML використовує різні теги та атрибути для визначення зовнішнього вигляду вмісту веб-сторінки, такого як текст, зображення, посилання, таблиці, форми та інші елементи. Браузери інтерпретують HTML-код та відображають його на екрані відповідно до вказаних правил. HTML є основним стандартом для веб-розробки та є необхідною складовою веб-технологій, таких як CSS та JavaScript. з кількох причин.

По-перше, HTML є основною мовою розмітки для веб-сторінок, що означає, що вона підтримується більшістю веб-браузерів і має велику кількість документації та підтримки. Це робить її ідеальним вибором для розробки веб-сервісу, який можна легко доступати з будь-якого пристрою з доступом до Інтернету.

По-друге, HTML дозволяє легко включати графіку, відео, аудіо та інші медіа-елементи в веб-сервіс, що може зробити його більш привабливим та зрозумілим для користувачів.

Крім того, HTML дозволяє легко взаємодіяти з іншими мовами програмування, такими як JavaScript та CSS, що може покращити функціональність та вигляд веб-сервісу.

В загальному, HTML є потужним та універсальним інструментом для розробки веб-сервісів, і я вважаю, що вона є найкращим вибором для мого дипломного проекту.

Приклад HTML коду для створення заголовка першого рівня і абзацу

```

<!DOCTYPE html>

<html>
<head>
<title>Заголовок сторінки</title>
</head>
<body>
<h1>Заголовок першого рівня</h1>
<p>Це абзац тексту.</p>
</body>
</html>

```

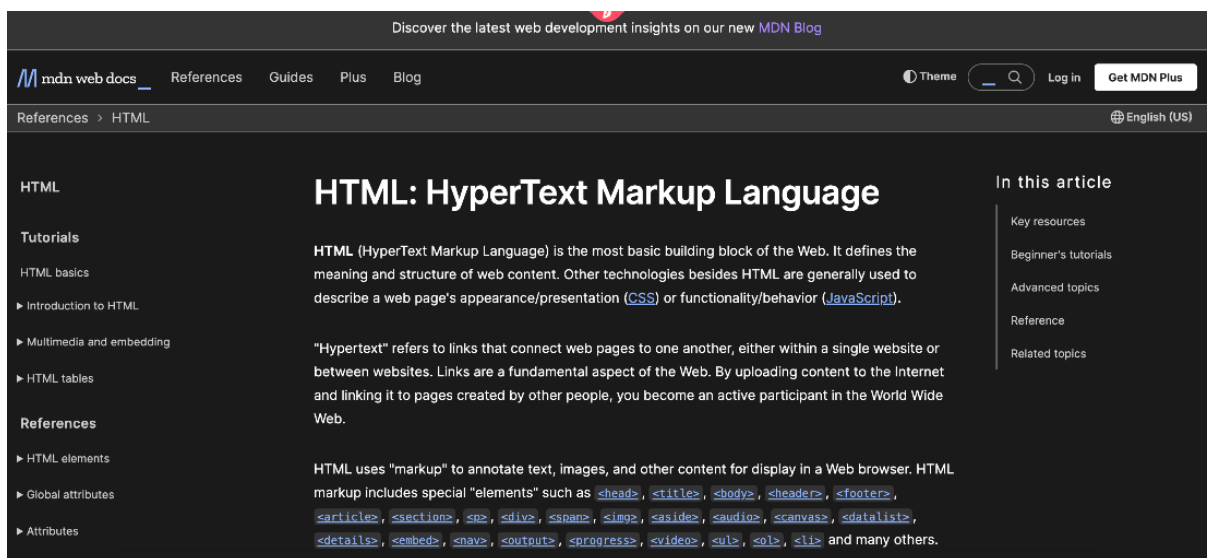


Рисунок 2.4 - Документація HTML на веб-сайті MDN

MDN Web Docs (раніше відомий як Mozilla Developer Network) - це веб-ресурс, який надає документацію, підручники, поради та іншу інформацію для розробників веб-додатків та веб-технологій.

MDN Web Docs є однією з найбільш авторитетних та найбільш повних джерел інформації про веб-технології, такі як HTML, CSS, JavaScript та інші.

Ресурс надає користувачам можливість швидко знайти відповіді на різноманітні запитання, пов'язані з веб-розробкою, включаючи відомості про синтаксис, стандарти, засоби розробки, фреймворки та бібліотеки.

MDN Web Docs є вільним та відкритим проектом, заснованим на співпраці та взаємодії спільноти розробників. Ресурс має багато користувачів та прихильників по всьому світу та став важливим джерелом інформації для веб-розробників.

2.4.2 Мова стилів CSS

CSS(Cascading Style Sheets) - це мова стилів, що використовується для оформлення веб-сторінок. Вона дозволяє відокремити стиль та дизайн веб-сторінки від її структури та контенту, що робить кодування більш ефективним та покращує зручність редагування веб-сторінки. причини вибору є.

По-перше, CSS дозволяє створювати багато різних стилів та оформлень для веб-сторінок, що дозволяє зробити веб-сервіс більш привабливим та привабливим для користувачів.

По-друге, використання CSS дозволяє розділити стиль та дизайн веб-сервісу від його кодування та функціональності. Це забезпечує більш ефективне управління веб-сервісом та дозволяє змінювати його вигляд без зміни його функціональності.

Крім того, CSS підтримується більшістю браузерів та використовується майже всіма веб-розробниками, що забезпечує його високу сумісність та надійність.

Приклад CSS коду

```
/* Задаємо стиль для заголовку першого рівня */
```

```
h1
```

```
{
```

```
font-size: 36px;

color: blue;

text-align: center;

}

/* Задаємо стиль для абзацу */

p

{

font-size: 18px;

color: black;

line-height: 1.5;

}
```

2.4.3 Об'єктно-орієнтована прототипна мова програмування Javascript

Була обрана мова JavaScript - це мова програмування, яка використовується для створення веб-сторінок та веб-додатків. Вона може бути використана для створення різноманітних ефектів, взаємодії з користувачем, анімації, перевірки форм, валідації даних та багатьох інших функцій на веб-сторінках. JavaScript є однією з найпоширеніших мов програмування, що використовуються у веб-розробці. Вона є складовою частиною технології веб-розробки, що включає HTML та CSS.

По-перше, JavaScript є мовою програмування, яка підтримується браузерами. Це означає, що я можу використовувати JavaScript для створення клієнтської частини веб-сервісу, яка буде працювати безпосередньо у браузері користувача. Це дозволяє створювати веб-сервіси, які працюють швидко та ефективно, забезпечуючи при цьому зручний та інтерактивний користувацький

інтерфейс.

По-друге, JavaScript має велику та активну спільноту розробників, що робить його дуже гнучким та масштабованим. Існує безліч бібліотек та фреймворків, які дозволяють використовувати JavaScript для розробки веб-сервісів на будь-якому рівні складності та масштабу.

По-третє, JavaScript є мовою програмування, яка може бути використана на більшості серверних платформ. Це означає, що я можу використовувати JavaScript для розробки серверної частини веб-сервісу, що дозволяє писати код з використанням однієї мови програмування на всіх рівнях стеку технологій.

Приклад JavaScript коду

```
<script>  
// Створюємо змінну з повідомленням  
var message = "Привіт, світ!";  
// Виводимо повідомлення у діалоговому вікні  
alert(message);  
</script>
```

2.5 Підготовка матеріалів

Розробка веб-сервісу починається з створення структури проекту, треба визначити секції веб-сервісу такі як `<header>`, `<main>`, `<aside>`, `<footer>` для того щоб розуміти як розмістити контент

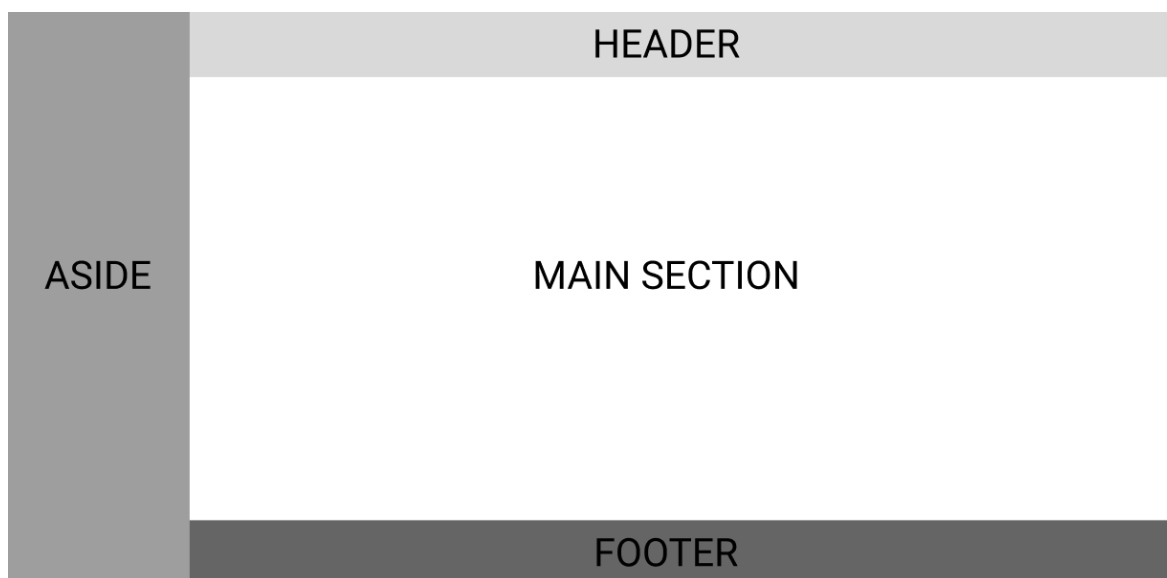


Рисунок 2.5 – Структура сайту

Header - є одним з елементів розмітки сторінки, який використовується для визначення вступної або початкової частини веб-сторінки. Заголовок може містити логотип, назву сайту, навігаційні посилання, контактну інформацію, пошукову форму та інші елементи, які мають бути видимими на початку сторінки.

Основна мета заголовка - це надати користувачеві важливу інформацію про вміст сторінки, яка допомагає зрозуміти контекст сторінки та швидко орієнтуватися в її змісті. Заголовок може бути розміщений в верхній частині сторінки (над головним вмістом) або в боковій панелі.

Aside - є елементом розмітки, який використовується для виділення бокового вмісту або контенту, який не є основним вмістом сторінки, але все ж має певний зв'язок з основним вмістом. Елемент `<aside>` може містити додаткові інформаційні блоки, банери, рекламу, посилання на подібний контент, або інші елементи, які можуть бути використані для доповнення основного вмісту сторінки.

Елемент `<aside>` може розміщуватися в боковій панелі, на боці сторінки, або в блоці, розташованому поруч з основним вмістом. Він може бути використаний в різних веб-сторінках, таких як блоги, статті, новини, або веб-

додатки, для відображення додаткової інформації, яка може бути корисною, але не є необхідною для розуміння основного контенту.

Main - є елементом розмітки, який використовується для визначення основного контенту сторінки, такого як головний вміст сторінки, який є унікальним для кожної сторінки. Елемент `<main>` зазвичай містить заголовки, параграфи, зображення, відео, форми та інші елементи, які становлять основний контент сторінки.

Використання елемента `<main>` в HTML рекомендується для поліпшення доступності веб-сторінок, допомагаючи веб-читачам легше розпізнавати основний контент сторінки, особливо в тих випадках, коли сторінка містить багато різноманітного контенту, такого як бокові панелі, навігацію, заголовки, підвал та інше.

Footer - є елементом розмітки, який використовується для визначення підвалу (нижньої частини) сторінки. Елемент `<footer>` зазвичай містить інформацію, таку як авторські права, контактні дані, посилання на соціальні мережі, додаткові посилання та інші додаткові елементи, які можуть бути корисні в нижній частині сторінки.

Використання елемента `<footer>` в HTML допомагає в покращенні доступності веб-сторінок, надає користувачам зрозумілу інформацію про підвал сторінки, і допомагає зберегти відповідну структуру та розмітку сторінки.

Після укладення структури проекту наступним кроком є створення user flow та user persona - це два різні поняття, що використовуються в маркетингу та веб-дизайні для розуміння та оптимізації користувацького досвіду взаємодії з продуктом чи послугою.

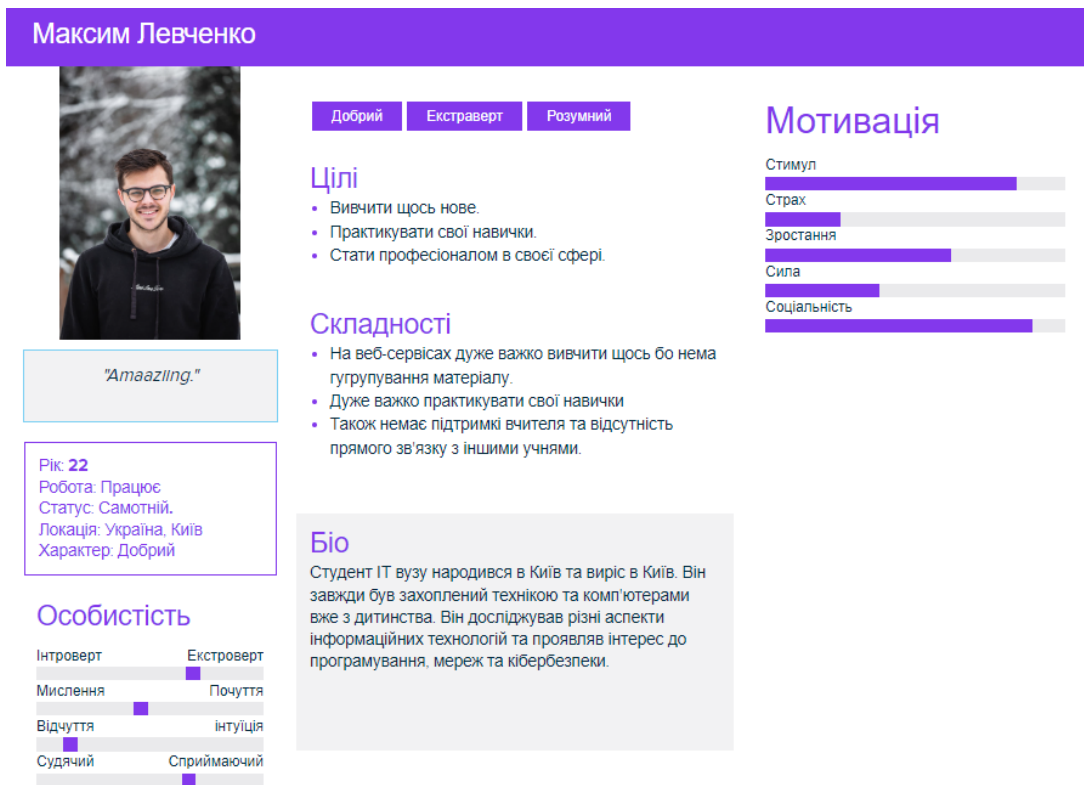


Рисунок 2.6 – User Persona

User persona створюється на основі дослідження та аналізу даних про реальних користувачів та включає такі характеристики, як вік, стать, освіта, заняття, інтереси, поведінка тощо. User persona допомагає розуміти потреби, мотивації, вимоги та очікування цільової аудиторії, що допомагає розробникам продукту створити більш ефективний та сприятливий користувацький досвід.

User persona може бути використана в процесі розробки продукту, маркетингових кампаній, тестування та оптимізації продукту для задоволення потреб користувачів.

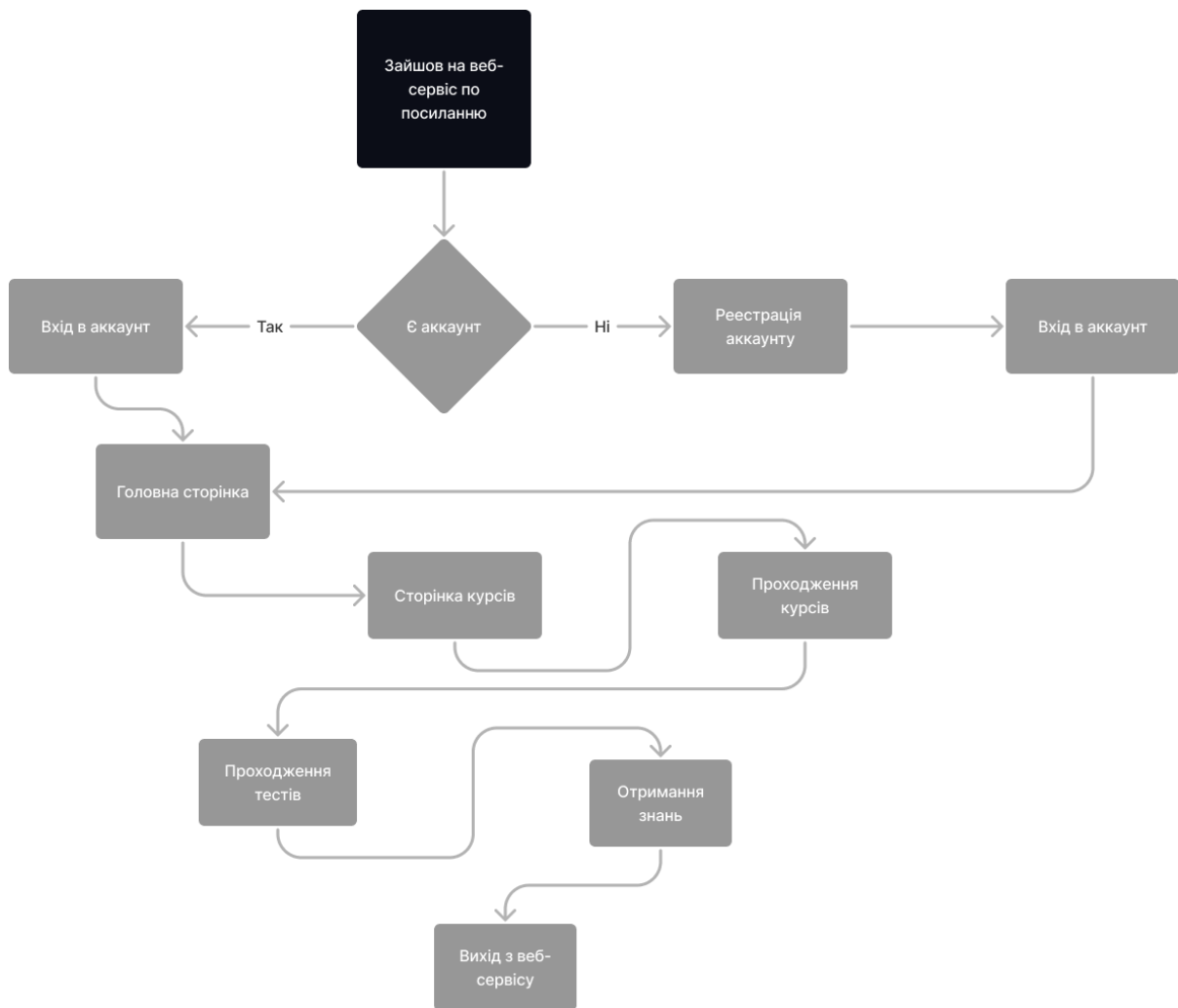


Рисунок 2.7 – UserFlow

User flow (потік користувача) - це шлях, який користувач проходить при взаємодії з продуктом чи послугою від початку до кінця. Він відображає послідовність кроків, які користувач здійснює, включаючи дії, які він виконує та взаємодії з інтерфейсом продукту. User flow допомагає розуміти, як користувачі взаємодіють з продуктом, яким чином вони переходять від одного кроку до іншого, та де можуть виникати проблеми або складнощі в процесі взаємодії.

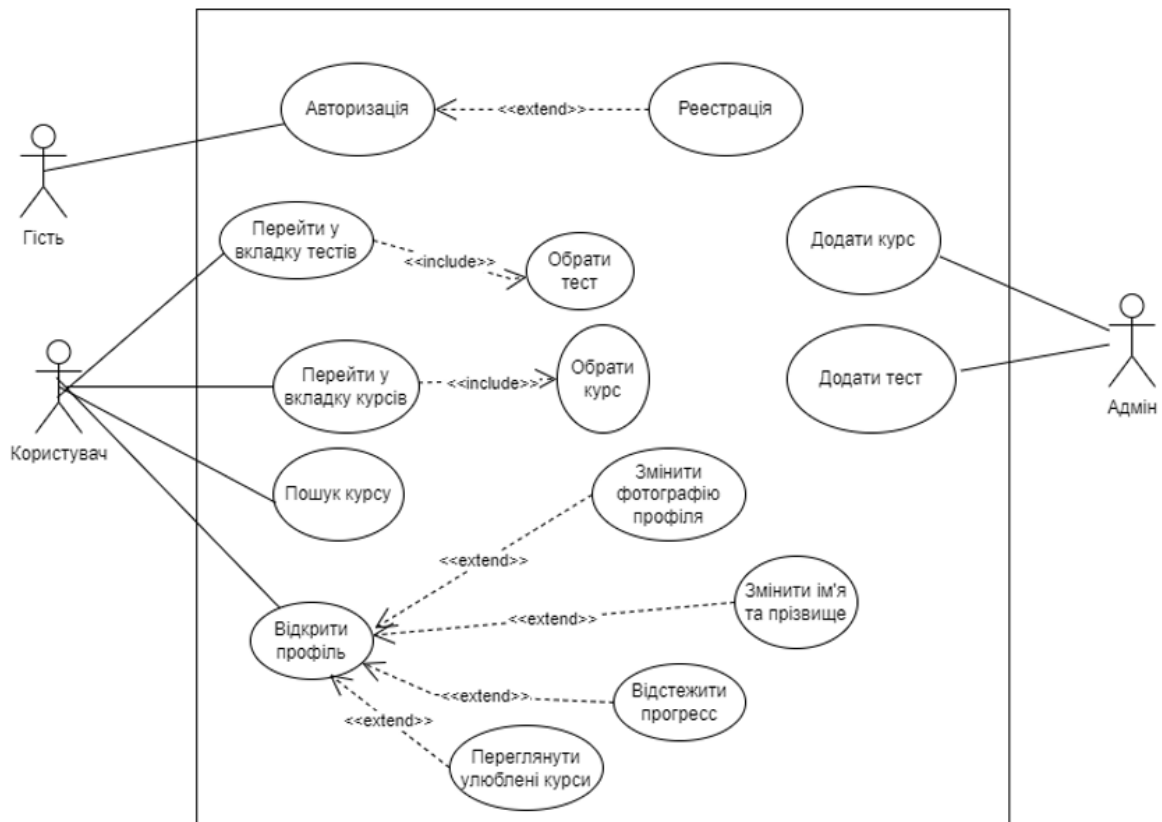


Рисунок 2.8 - діаграма Use Case

Use case визначає, як система повинна функціонувати в реальних умовах та які дії виконує користувач, щоб досягти певних результатів. Він детально описує кроки, які користувач виконує, а також взаємодію з системою, включаючи введення даних, обробку, виведення результатів тощо.

Use case може бути використаний для розробки програмного забезпечення, проектування інтерфейсу користувача, визначення вимог до системи та відповідного тестування. Він є важливим інструментом для аналізу та документування функціональності системи.

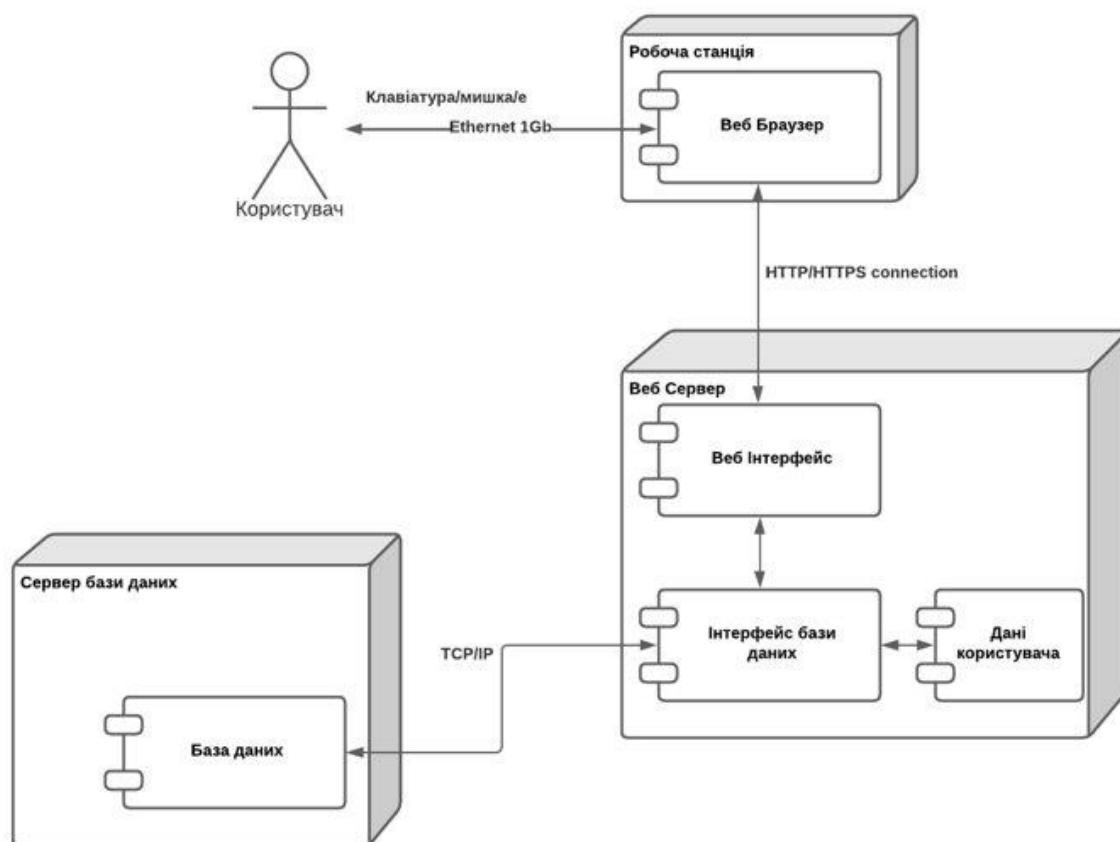


Рисунок 2.9 - Архітектура веб-сервісу

Зв'язок між веб-сервісом та базою даних відбувається за допомогою HTTP-запитів та HTTP-відповідей. HTTP (Hypertext Transfer Protocol) є протоколом передачі даних у мережі Інтернет і використовується для комунікації між клієнтом (наприклад, веб-браузером) та сервером.

Коли веб-сервіс потребує доступу до бази даних, він може виконати HTTP-запит до сервера бази даних. Цей запит може мати різні методи, такі як GET, POST, PUT або DELETE, що визначають тип операції, яку необхідно виконати над базою даних.

HTTP-запит містить різні елементи, зокрема:

1. URL (Uniform Resource Locator): Вказує адресу ресурсу, з яким потрібно взаємодіяти. Це може бути адреса таблиці бази даних або конкретного запиту до неї.

2. Метод: Визначає тип операції, яку необхідно виконати. Наприклад, GET-запит використовується для отримання даних, POST-запит - для створення нових записів, PUT-запит - для оновлення існуючих даних, DELETE-запит - для видалення даних.
3. Заголовки: Містять додаткову інформацію про запит, наприклад, тип даних, які приймаються або повертаються, авторизаційні дані, кешування тощо.
4. Тіло запиту: Містить додаткові дані, які передаються разом із запитом. Наприклад, в POST-запитах тіло може містити дані, які необхідно додати до бази даних.

Коли сервер бази даних отримує HTTP-запит, він обробляє його відповідно до методу та параметрів, виконує необхідну операцію в базі даних і формує HTTP-відповідь. HTTP-відповідь містить статус-код, що вказує на результат виконання.

2.6 Висновок до другого розділу

В цьому розділі була розглянута підготовка до створення веб-сервісу для проходження освітніх курсів. Та покроковий план створення проекту для розуміння того як проект має бути написаний.

Наш перший крок був аналізом інструментів для створення програмних забезпечень та в нашому випадку саме веб-сервісу. Також був розглянутий термін “Середовище розробки”

Проаналізовані середовища розробки Visual Studio Code та Atom.

Були проаналізовані особливості такі як

- Швидкість
- Функціональність
- Розширюваність
- Підтримка та оновлення

Розглянули недоліки та переваги середовища розробки Visual Studio Code.

Середовищем для цього проекту було обрано Visual Studio Code, для макету проекту було обрано Figma, та мова програмування була обрана JavaScript

Далі було розроблено детальний алгоритм для створення програмного продукту в цілому. Виділені етапи розробки включають наступні пункти:

- Визначення вимог проекту.
- Розробка дизайну інтерфейсу користувача.
- Розробка функціональності.
- Розробка адміністративної панелі.
- Інтеграція з іншими сервісами.
- Тестування та відлагодження

Проаналізовано та визначено головні функції проекту:

- Реєстрація та автентифікація.
- Каталог курсів.
- Онлайн-лекції.
- Матеріали курсу.
- Взаємодія з викладачами та іншими учасниками.
- Відстеження прогресу.
- Оцінювання та звітність.
- Тестування та оцінювання.
- Сертифікати.

Першим кроком будуть реалізовані саме ці задачі для майбутнього веб-сервіс.

Також провели аналіз та підготували матеріал та на його основі буде розроблено веб-сервіс.

Була розроблена User Persona для аналізу вимог та User Flow який допомагає розібратися з послідовністю кроків, які користувач повинен здійснити для досягнення певної мети, такої як реєстрація або оформлення замовлення

3. ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБ-СЕРВІСУ ДЛЯ ПРОХОДЖЕННЯ ОСВІТНІХ КУРСІВ

3.1 Прототипування веб-сервісу. UI та UX дизайн

Прототип або макет сайту - це візуальне представлення вигляду та функціональності майбутнього веб-сайту, що допомагає визначити його структуру, дизайн та навігацію. Він дозволяє проводити тестування взаємодії з користувачами та виявляти можливі проблеми на ранніх етапах розробки, економлячи час та ресурси. Прототип або макет є важливим інструментом у веб-розробці, допомагаючи створити ефективний та зручний веб-сайт.

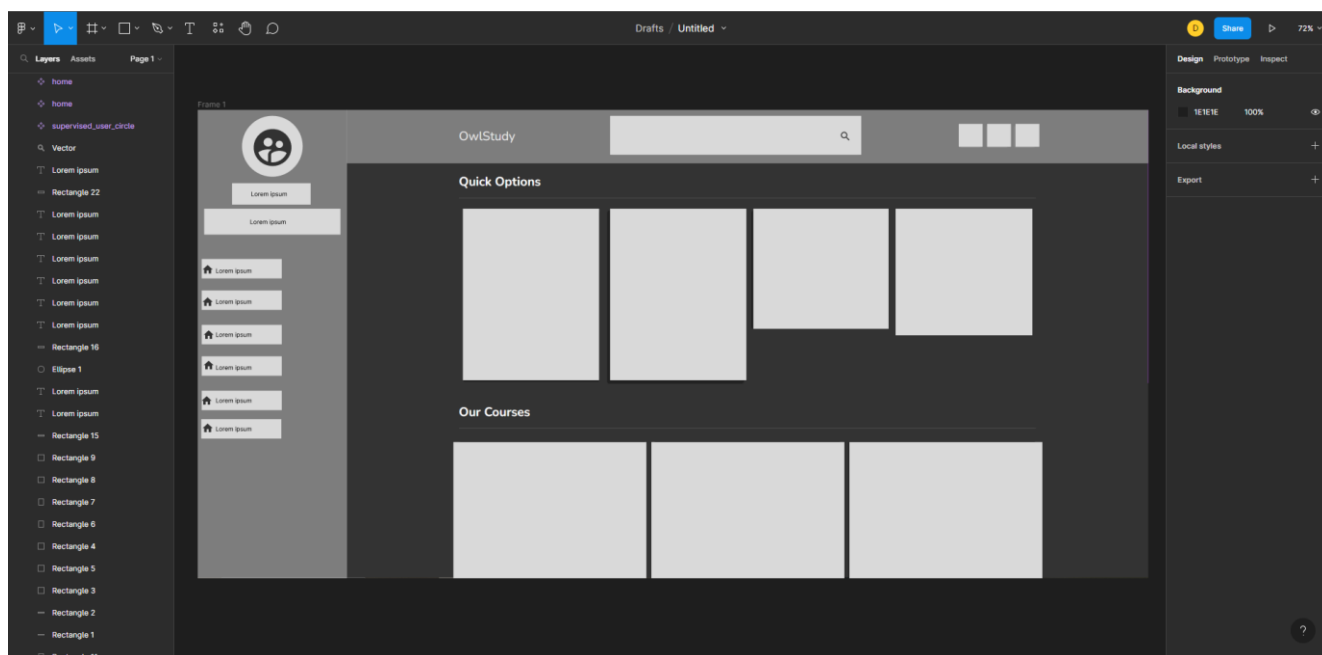


Рисунок 3.1 – Макет веб-сервісу

Макет був розроблений за допомогою Figma. Figma - це інтерактивний веб-інструмент для дизайну, прототипування та співпраці в реальному часі. Він дозволяє дизайнерам створювати векторні графічні елементи, розробляти макети

веб-сайтів та мобільних додатків, створювати анімацію, налаштовувати взаємодію елементів, а також співпрацювати з командою в реальному часі. Figma є популярним інструментом серед дизайнерів, який дозволяє ефективно працювати над проектами веб-дизайну, спрощуючи процес створення дизайну та співпраці між командами.

Шаблон веб-сервісу повинен містити:

- Header веб-сервісу, який містить логотип, search bar та кнопки авторизації
- Aside веб-сервісу, який містить горизонтальне меню в якому знаходяться посилання на розділи веб-сервісу
- Контент веб-сервісу

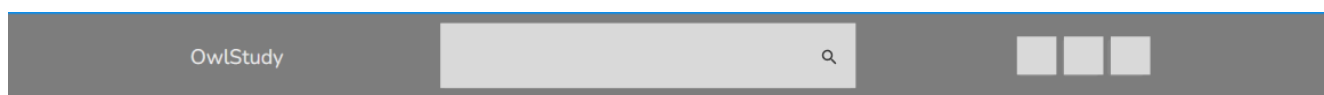


Рисунок 3.2 – Search Bar

Search bar на веб-сайті - це інструмент, який дозволяє відвідувачам здійснювати пошук по вмісту сайту, швидко знаходити необхідну інформацію та полегшує навігацію. Він забезпечує зручний спосіб пошуку інформації, дозволяє користувачам знаходити потрібний контент швидко та ефективно, покращує користувацький досвід та забезпечує зручність взаємодії з сайтом. Search bar є важливим елементом веб-дизайну, допомагає забезпечити відвідувачам зручний пошуковий функціонал та полегшує навігацію на сайті.



Рисунок 3.3 – Aside меню

Aside на веб-сайті - це розділ з додатковим вмістом, таким як реклама, навігація, соціальні медіа, додатковий контент або контактна інформація, який забезпечує зручність користувачам та можливості монетизації для веб-сайту. Aside може містити контактну інформацію, таку як адреса, номер телефону, або форма зворотного зв'язку, допомагаючи відвідувачам зв'язатися з веб-сайтом.



Рисунок 3.4 – Контент Веб-сервісу

Контент на веб-сервісі потрібен для залучення аудиторії та забезпечення цінного вмісту, який цікавить користувачів, покращує їх досвід на сайті та взаємодію з веб-сервісом.

3.2 Створення логотипу веб-сервісу

Логотип сайту є графічним символом або знаком, який використовується для ідентифікації сайту, компанії або бренду. Він зазвичай розміщений в верхній частині веб-сторінки та може містити назву сайту, логотип компанії, графічні елементи, кольори та інші дизайнерські елементи, що відрізняють його від інших сайтів і роблять його легко впізнаваним для відвідувачів.

Логотип сайту має відображати візуальний стиль, цінності та характер сайту або компанії, яку він представляє. Він може бути простим або складним, використовувати текст, графічні елементи, або комбінацію обох. Логотип має бути відповідним до бренду та повідомлення, яке сайт хоче передати своїм відвідувачам.



Рисунок 3.5 – Логотип сайту

Для веб-сервісу був обраний логотип з совою тому що:

1. Візуальна асоціація: Сова є символом мудрості, інтелекту та розуму в багатьох культурах. Було використано сову в логотипі, щоб відобразити ці якості або встановити асоціації з веб-сервісом.
2. Асоціація з освітнім середовищем: Сиви є нічними птахами, і їх пов'язують з нічним навчанням, дослідженням та розумовою діяльністю. Це може бути відповідним символом для освітнього сайту, школи, університету або іншої освітньої організації.
3. Креативність та оригінальність: Використання сови в логотипі може допомогти зробити його унікальним та запам'ятовуваним, оскільки це неординарний елемент дизайну, який може виділити веб-сервіс.

3.3 Аналіз використаної типографії

Типографія на веб-сайті - це застосування принципів типографії для дизайну та вигляду текстового вмісту на веб-сторінках. Вона включає в себе вибір шрифтів, розмірів, колірної схеми, міжлітерних і міжсловових інтервалів, вирівнювання, інтерфейсних елементів та інших аспектів, що впливають на

зовнішній вигляд тексту на веб-сторінках.

Вибір правильних шрифтів для веб-сайту є важливим аспектом типографії на веб-сторінках. Варто враховувати читабельність шрифтів на різних пристроях та екранах, таких як комп'ютери, смартфони, планшети та інші. Розмір шрифту також має бути належним, забезпечуючи зручність читання для користувачів.

```
*{  
  font-family: 'Nunito', sans-serif;  
  margin:0; padding:0;  
  box-sizing: border-box;  
  outline: none; border:none;  
  text-decoration: none;  
}
```

Рисунок 3.6 – Додавання веб-сервісу шрифт

В розробці був використаний шрифт Nunito з типом шрифту sans-serif

Sans-serif - це тип шрифту, який не має засічок (малих вертикальних виступів на кінцях літер) на кінцях літер. Він відрізняється від serif-шрифту, де на кінцях літер присутні засічки.

Sans-serif шрифти вважаються більш сучасними, простими та чистими в порівнянні з serif-шрифтами, які можуть викликати відчуття традиційності та класики. Вони часто використовуються в дизайні веб-сайтів, друкованих матеріалах, логотипах, рекламних банерах та інших графічних проектах.

3.4 Теорія кольору

Теорія кольору – це наука, яка вивчає властивості, взаємодію та сприйняття кольору. Вона розглядає різні аспекти кольорового явища, такі як його фізичні властивості, взаємодію зі світлом, сприйняття кольору людським зором, взаємозв'язок між різними кольорами та їх використання в різних контекстах.

```
:root{
  --main-color: #8e44ad;
  --red: #e74c3c;
  --orange: #f39c12;
  --light-color: #888;
  --light-bg: #eee;
  --black: #2c3e50;
  --white: #fff;
  --border: .1rem solid rgba(0,0,0,.2);
}
```

Рисунок 3.4.1 – CSS Variables з кольорами веб-сервісу

CSS змінні (CSS Variables) - це новий функціонал в CSS, який дозволяє визначати змінні, що містять значення, які можуть використовуватися в різних місцях в CSS коді. Змінні в CSS починаються з префікса "--" і можуть містити різноманітні типи значень, такі як кольори, розміри, рядки тощо.

Один з основних принципів змінних CSS - це можливість використовувати одне значення в багатьох місцях в CSS коді, і зміна значення всього однієї змінної може призвести до автоматичного оновлення значень у всіх використовуваних місцях. Це забезпечує більшу гнучкість і зручність в управлінні стилями на сторінці веб-сайту або додатку.



Рисунок 3.4.2 – Основні кольори які були використані в веб-сервісі

Акцентований кольором веб-сервісу є фіолетовий тому що він часто асоціюється зі знаннями, розумом і технологіями. Він може викликати відчуття інтелектуальності і дослідництва, що може бути відповідним для веб-сервісу, який пропонує освітні курси. Фіолетовий колір також може викликати асоціації з креативністю і інноваціями. Він може стимулювати творчий мислення і виражати оригінальність. Це може бути важливим аспектом веб-сервісу, пов'язаному з освітніми курсами, де розвиток нових ідей та рішень може бути важливим елементом навчання.

Для веб-сервісу було створено дві теми, світла та темна для світлої теми був використаний основний колір #EEEEEE та для темної теми #333333.

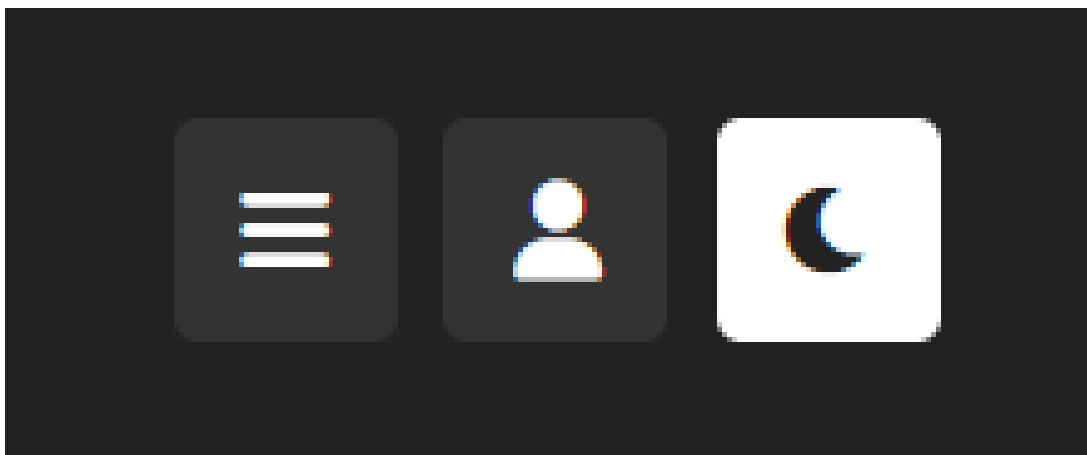


Рисунок 3.4.3 – Перемикач теми веб-сервісу

У веб-сервіс була додана темна тема для комфортного користування вночі або в умовах обмеженого освітлення. Темна тема забезпечує менш яскраве освітлення екрану, що може бути корисним в нічний час або в умовах, коли користувач перебуває в темному середовищі, такому як кімната зі зниженим освітленням. Це може допомогти зменшити вплив яскравого світла на очі користувача, забезпечуючи комфортніші умови для використання веб-сервісу.

```

let toggleBtn = document.getElementById('toggle-btn');
let body = document.body;
let darkMode = localStorage.getItem('dark-mode');

const enableDarkMode = () =>{
  toggleBtn.classList.replace('fa-sun', 'fa-moon');
  body.classList.add('dark');
  localStorage.setItem('dark-mode', 'enabled');
}

const disableDarkMode = () =>{
  toggleBtn.classList.replace('fa-moon', 'fa-sun');
  body.classList.remove('dark');
  localStorage.setItem('dark-mode', 'disabled');
}

if(darkMode === 'enabled'){
  enableDarkMode();
}

toggleBtn.onclick = (e) =>{
  darkMode = localStorage.getItem('dark-mode');
  if(darkMode === 'disabled'){
    enableDarkMode();
  }else{
    disableDarkMode();
  }
}

```

Рисунок – 3.4.4 JavaScript реалізація перемикача тем

3.5 Компоненти веб-сервісу та їх стан

Графічно зображені зміни стану елемента фотографії при наведенні, які можна побачити на наданому рисунку рис 3.5.1. Це надає можливість користувачам чітко сприймати, які можливі дії викликають різні ефекти візуальної взаємодії з елементом. Зміна стану може бути обумовлена різними факторами, такими як позиція курсору миші, натискання на елемент, або будь-яка інша подія, яка ініціює зміну його стану. Важливою складовою веб-дизайну є правильна реалізація змін стану елементів, що може покращити взаємодію користувача з веб-сайтом і зробити його візит більш продуктивним та задовільним.

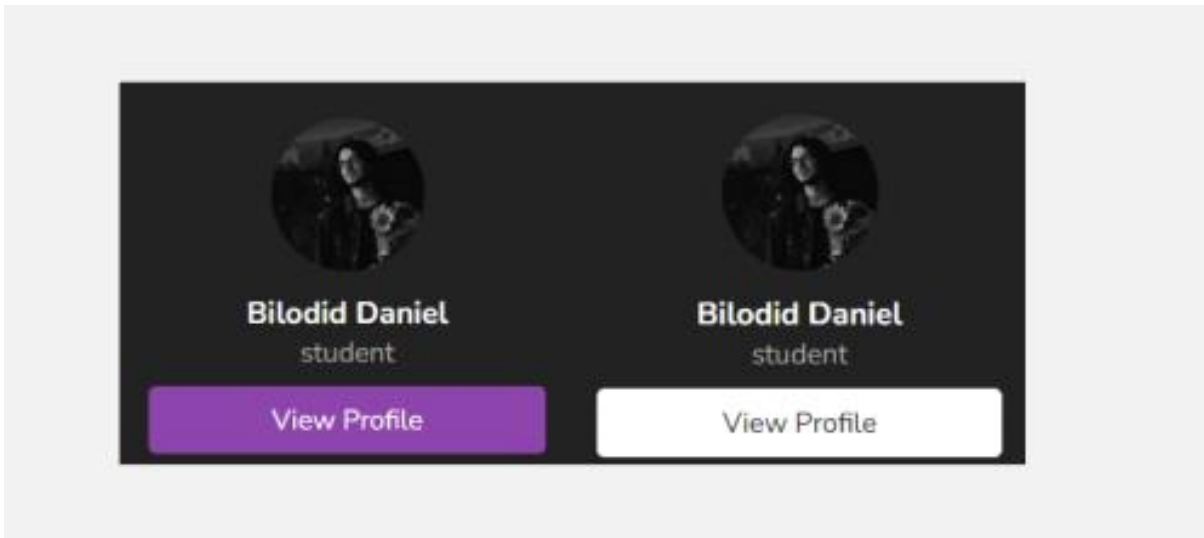


Рисунок – 3.5.1 Зміна стану кнопки при наведенні

Зміна стану була розроблена за допомогою псевдокласу `hover` - це псевдоклас, який використовується для створення ефектів, що відбуваються, коли користувач наводить курсор миші на елемент веб-сторінки, але ще не відпускає кнопку миші. Зазвичай це використовується для надання візуального змінного стану елементу при наведенні на нього мишою, що дозволяє створювати взаємодію із користувачем.

```

123  .inline-btn:hover,
124  .inline-option-btn:hover,
125  .inline-delete-btn:hover,
126  .btn:hover,
127  .delete-btn:hover,
128  .option-btn:hover {
129      background-color: var(--black);
130      color: var(--white);
131  }
  
```

Рисунок - 3.5.2 Код псевдокласу Hover

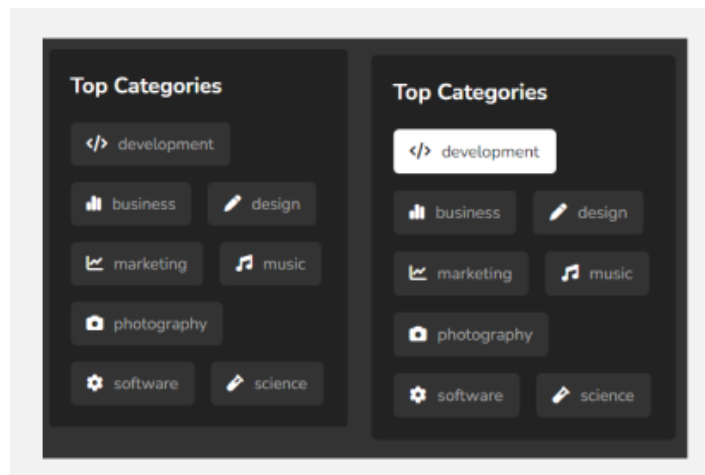


Рисунок – 3.5.3 Зміна стану категорії при наведенні

Popup window - це вікно, яке з'являється перед користувачем на екрані, зазвичай, поверх веб-сторінки або програми, яку він використовує. Popup windows можуть відображати різну інформацію, таку як повідомлення про помилки, запити на введення даних або рекламні матеріали.

Popup windows можуть з'являтися автоматично, коли користувач зайшов на сторінку або виконав певну дію, наприклад, натиснув на кнопку. Це може бути корисним для отримання додаткової інформації або для отримання дозволу від користувача на виконання певної дії.

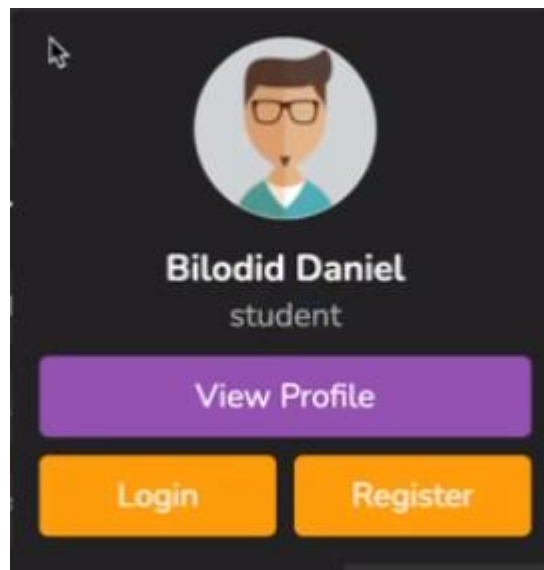


Рисунок - 3.5.4 Popup login

Popup login - це форма автентифікації, яка з'являється у вікні, коли користувач намагається отримати доступ до певної сторінки або ресурсу, який вимагає авторизації. Замість того, щоб переходити на іншу сторінку для введення даних автентифікації, користувач може ввести свої дані прямо у вікні, яке з'являється на екрані.

3.6 Висновок до третього розділу

Підсумуючи третій розділ, можна зазначити, що прототипування є важливим етапом у створенні веб-сервісу. Для цього було використано середовище Figma, де було розроблено кожен сторінку веб-сервісу з унікальним дизайном для подальшого поліпшення.

Було створено оригінальний логотип для веб-сервісу, який включає в собі зображення сови та використання фіолетового кольору. Логотип був розроблений унікальним чином. Сова є твариною, яка асоціюється зі знанням та мудрістю в багатьох культурах світу. Вона має гострий зір та відома своєю здатністю бачити в темряві. Також, її спокійна та розсудлива натура символізує розуміння та здатність до обдуманих рішень. У зв'язку з цим, сова відображає ідеали, які важливі для

навчання та освіти, такі як знання, мудрість, розуміння та навички критичного мислення. Фіолетовий колір символізує знання, мудрість, духовність, саморозвиток та інтелектуальний потенціал.

Після цього було проведено аналіз типографії з метою вибору найбільш підходящих шрифту, розміру та кольору тексту. Для веб-сервісу було вибрано два типи шрифтів: один для основного тексту, який має стиль без засічок (Sans-serif), та інший для заголовків, що має стиль Nunito.

За результатами аналізу кольорографії і психології кольорів було обрано 3 основні кольори веб-сервісу. Було вибрано фіолетовий колір, білий колір та чорний колір. Фіолетовий колір символізує знання, мудрість, духовність, саморозвиток та інтелектуальний потенціал. Білий колір символізує інтелектуальну чистоту, гармонію, чесність, інновації та нові початки. У багатьох країнах білий колір використовується як символ освіти. Чорний колір був використано задля темної теми.

Далі було розроблено декілька станів елементів з метою поліпшення враження від користування. При наведенні на кнопки та картки, користувач буде спостерігати анімації переходів, що дозволить отримати більш зручний та привабливий користувацький досвід.

4. ОПИС ВИКОРИСТАННЯ ТА ТЕСТУВАННЯ ВЕБ-СЕРВІСУ

4.1 Сценарій роботи з веб-сервісом

Коли користувач зайшов на веб-сервіс з освітніми курсами, першим, що йому зустрічається, є головна сторінка (home page). Вона має на меті привернути увагу користувача та заохотити його ознайомитись з курсами.

Окрім головної сторінки, користувач також бачить меню панель (aside) з вибором курсів та категорій, які допомагають зорієнтуватись на веб-сервісі. Це меню з розділами курсів за темами, рівнем складності.

Також на веб-сервісі присутня панель з пошуком, яка допомагає знайти необхідний курс. Користувач може шукати курси за ключовими словами, авторами, темами та іншими параметрами.

Крім того, на веб-сервісі присутня панель з профілем користувача, де користувач може знайти інформацію про користувача та його курси. Тут користувач може переглянути свій прогрес в курсах, додати курс до списку обраних та відредагувати свій профіль.

Нарешті, користувач може перемикати теми веб-сервісу для кращого комфорту використання. Наявність різних тем може допомогти користувачам знаходити курси та навчальний матеріал у зручному та приємному для них вигляді.

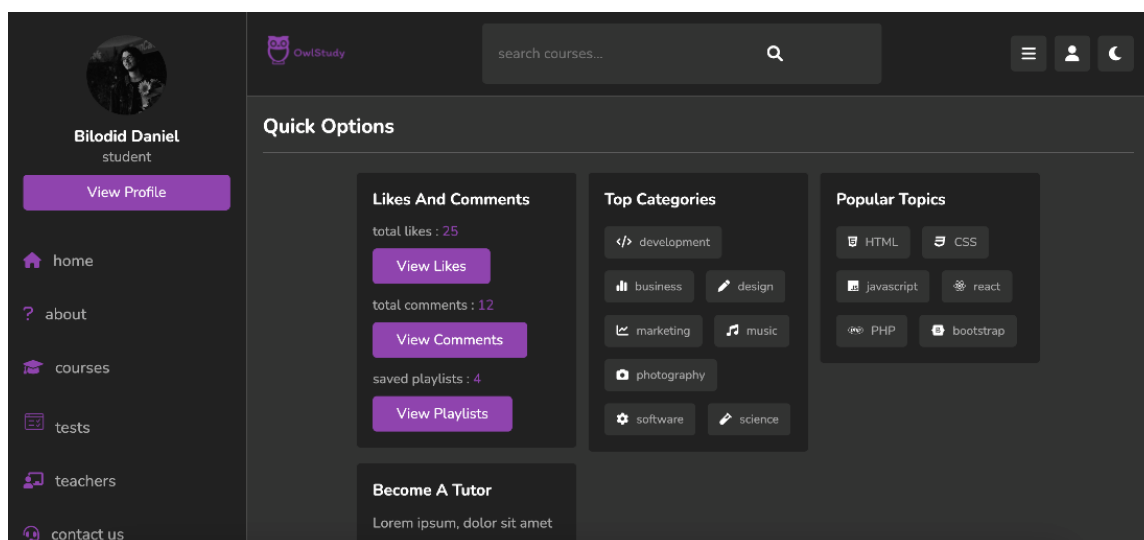


Рисунок 4.1.1 - Home page веб-сервісу

Однією з головних інформацій, яку користувач може знайти на сторінці "Про нас", є кількість онлайн-курсів, які доступні на веб-сервісі. Це дає користувачу уявлення про розмаїття курсів, які можна пройти на веб-сервісі, та допомагає знайти курс, який відповідає його потребам та інтересам.

Також на сторінці "Про нас" може бути вказано, скільки учнів вже успішно закінчили курси на веб-сервісі. Це свідчить про популярність та ефективність курсів, які надає веб-сервіс, та дає користувачу додаткову мотивацію для проходження курсу.

Нарешті, на сторінці "Про нас" користувач може знайти відгуки від інших користувачів, які вже пройшли курси на веб-сервісі. Відгуки можуть містити інформацію про якість курсів, процес навчання, викладачів та інші важливі аспекти, які можуть допомогти користувачеві зробити правильний вибір курсу.

Загалом, на сторінці "Про нас" користувач може дізнатися більше про веб-сервіс та його курси, що допоможе йому зробити більш обґрунтований вибір та отримати те за чим він прийшов.

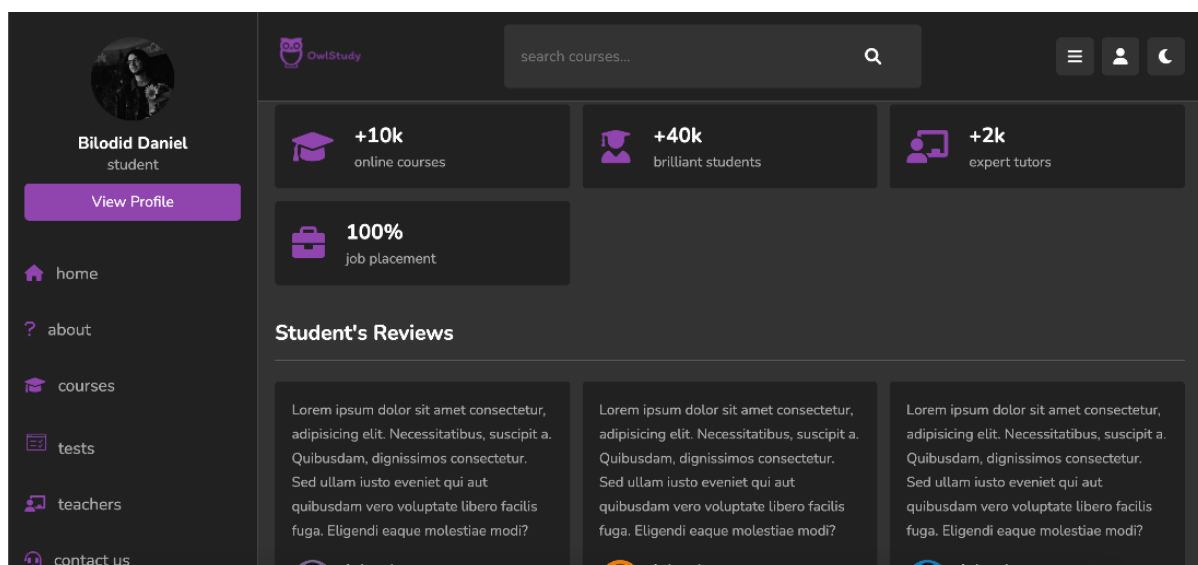


Рисунок 4.1.2 About page веб-сервісу

Коли користувач заходить на веб-сервіс з освітніми курсами, він може перейти на сторінку Courses, де йому буде доступна інформація про всі курси, які доступні на цьому веб-сервісі. Кожен курс буде представлений в окремому блоку, де буде вказана назва курсу.

Крім того, користувач може бачити кнопку "Перейти до курсу", за якою може перейти на сторінку конкретного курсу. На цій сторінці користувач отримає більш детальну інформацію про курс, таку як опис, план навчання, викладачі, матеріали курсу та інші параметри.

Це дозволяє користувачу зручно вибрати курс, який він хоче вивчати, залежно від своїх потреб та інтересів.

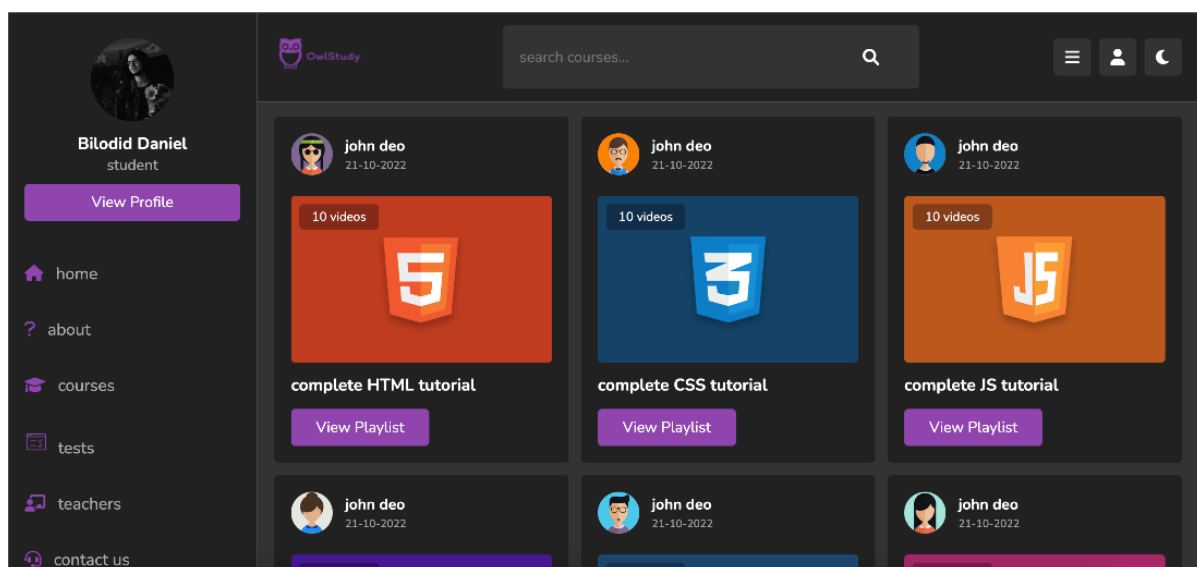


Рисунок 4.1.3 Courses page веб-сервісу

Коли користувач вирішив перейти на сторінку конкретного курсу, він зможе бачити плейліст курсу, який складається з різних уроків, тем, завдань та інших матеріалів. Кожен елемент плейлісту буде представлений в окремому блоку з коротким описом.

Крім того, на сторінці курсу користувач може бачити кнопку "Додати в обране", щоб зберегти курс у своїх обраних, а також може перейти до профілю викладача, який розробив курс, щоб отримати додаткову інформацію про нього.

Також на сторінці курсу користувач може бачити детальну інформацію про курс, яка містить опис курсу, мету та цілі, тривалість, складність, матеріали, які використовуються в курсі, та інші важливі параметри. Всю цю інформацію можна знайти на відповідних вкладках на сторінці курсу.

Також на сторінці курсу користувач може зайти до профілю викладача, який створив цей курс. В профілі викладача може бути додаткова інформація про його освіту та досвід роботи, а також перелік інших курсів, які він пропонує на цьому веб-сервісі. Користувач може взяти участь у інших курсах цього викладача, які можуть бути цікавими для нього.

Такі можливості надають користувачам можливість знайти не тільки курси,

але й професійних викладачів з великим досвідом у своїй галузі. Це допомагає забезпечити високу якість навчання та зробити процес навчання більш персоналізованим та ефективним.

Такі функції дозволяють користувачам зручно відслідковувати свій прогрес у навчанні та зберігати курси, які їм дуже сподобалися, щоб повернутися до них пізніше.

Такі функції роблять процес навчання більш ефективним та цікавим для користувача. Користувач може вчитися в своєму власному темпі та в будь-який час, який йому зручний. Більш того, він може ділитися своїм прогресом з іншими студентами, що допоможе йому виявити недоліки та покращити свої знання.

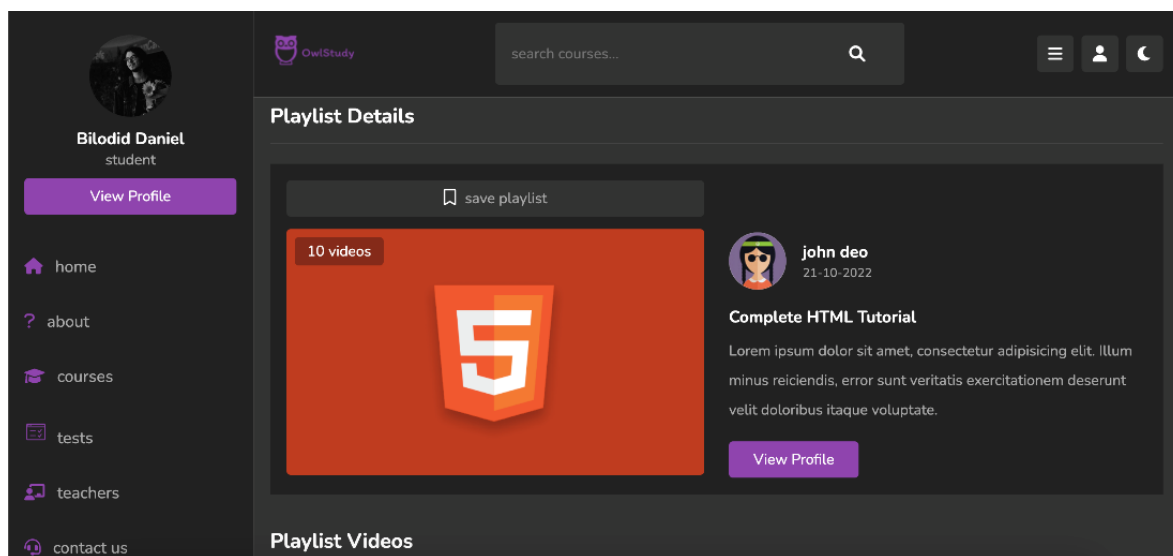


Рисунок 4.1.4 Playlist page веб-сервісу

Коли користувач обирає тест на веб-сервісі, він переходить на сторінку з питаннями. На цій сторінці буде набір питань з можливими варіантами відповідей. Користувач повинен обрати відповіді на кожне питання. Якщо тест містить більше одного питання, то після того, як користувач відповість на останнє питання, він зможе натиснути кнопку "Submit".

Після завершення тесту користувачу буде показано, скільки балів він

отримав за відповіді на питання. Будуть показані правильні та неправильні відповіді на кожне питання, щоб користувач міг перевірити свої знання та дізнатися, що він зробив не так. Також зберігаються результати тестування та відстежувати прогрес користувача в кожному тесті, можна надати можливість реєстрації на веб-сервісі.

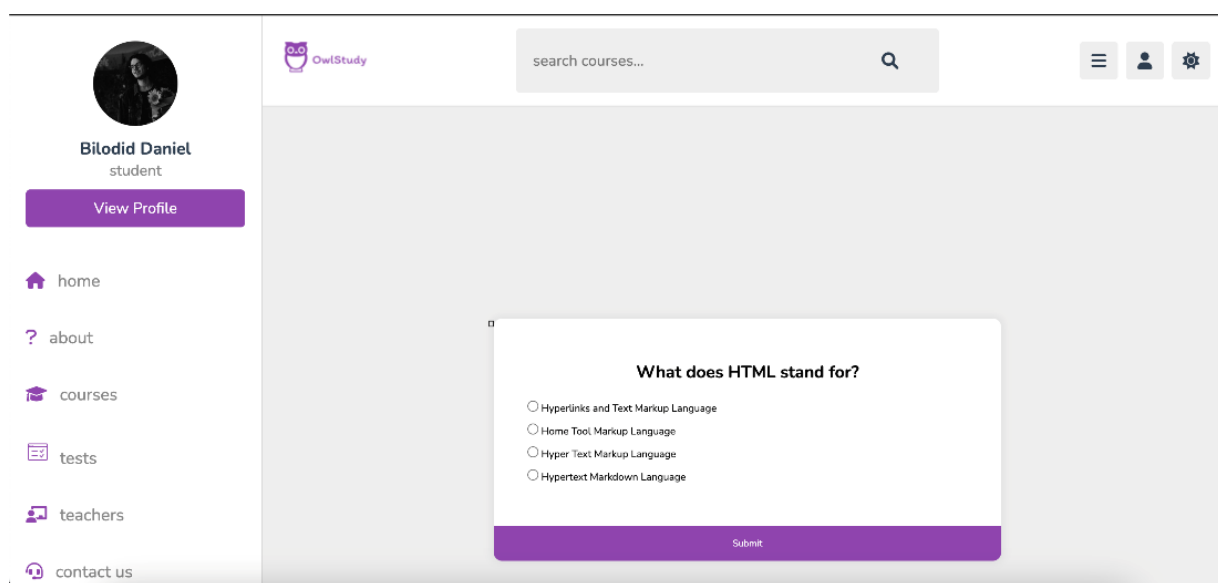


Рисунок 4.1.5 Test page веб-сервісу

Коли користувач обирає в меню опцію "Вчителі", він потрапляє на сторінку, де може знайти інформацію про доступних вчителів. Кожен вчитель має свій профіль зі списком курсів, які він надає. Крім того, на сторінці можна побачити загальну кількість курсів, які записав вчитель, а також іншу корисну інформацію, таку як освіта, досвід роботи тощо.

Якщо користувач зацікавлений в одному з курсів, які пропонують вчителі, він може натиснути на кнопку "Перейти до курсу" та перейти на сторінку курсу, де можна отримати докладну інформацію про нього та почати навчання. Крім того, на сторінці профілю вчителя можна побачити кнопку "Стати вчителем", яка дозволяє користувачу подати заявку на викладання курсів на платформі.

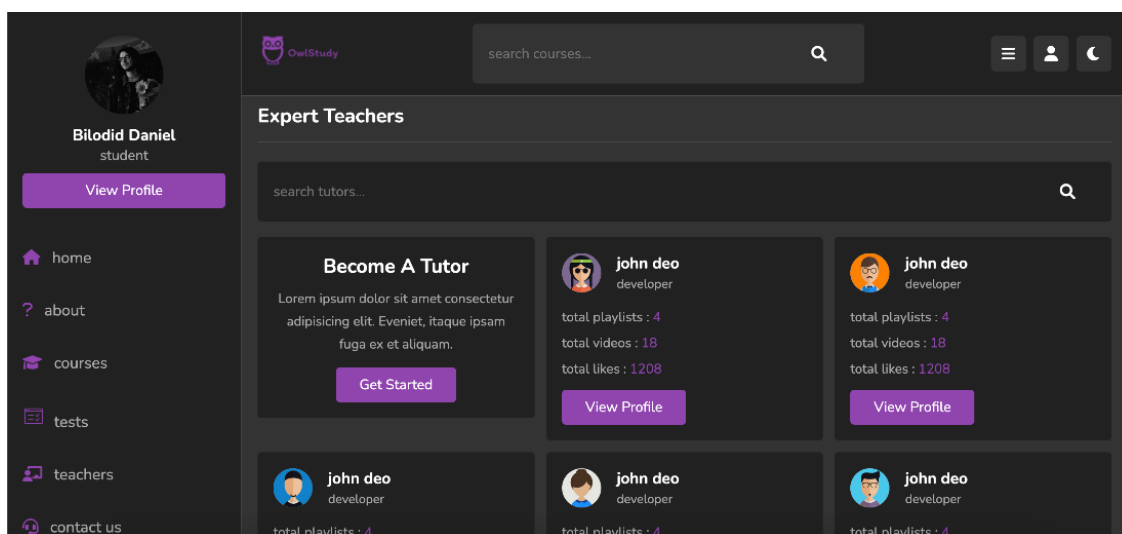


Рисунок 4.1.6 Teachers page веб-сервісу

Коли користувач хоче зареєструватися на веб-сервісі або стати вчителем, він повинен пройти процес реєстрації, заповнивши відповідну форму. Форма в HTML є одним з найважливіших елементів веб-сторінок, який дозволяє збирати інформацію від користувачів та передавати її на сервер.

Форма зазвичай складається з різних полів, таких як ім'я, прізвище, електронна пошта, пароль та інші, які користувач повинен заповнити. Крім того, форма може містити кнопки для відправки заповнених даних на сервер, очищення форми або скасування введення.

Під час заповнення форми користувач повинен бути уважним і перевірити правильність введених даних, оскільки ці дані будуть використовуватися для створення його облікового запису або профілю вчителя.

Зазвичай, на веб-сервісах форма реєстрації містить поля для введення основної інформації про користувача, такі як ім'я, прізвище, електронна пошта, пароль та підтвердження пароля. Також можуть бути додаткові поля для введення додаткової інформації, наприклад, місце проживання, номер телефону, дата народження тощо.

Отже, форма реєстрації є важливим елементом веб-сторінок, який дозволяє збирати інформацію від користувачів та передавати її на сервер для створення

облікового запису або профілю вчителя.

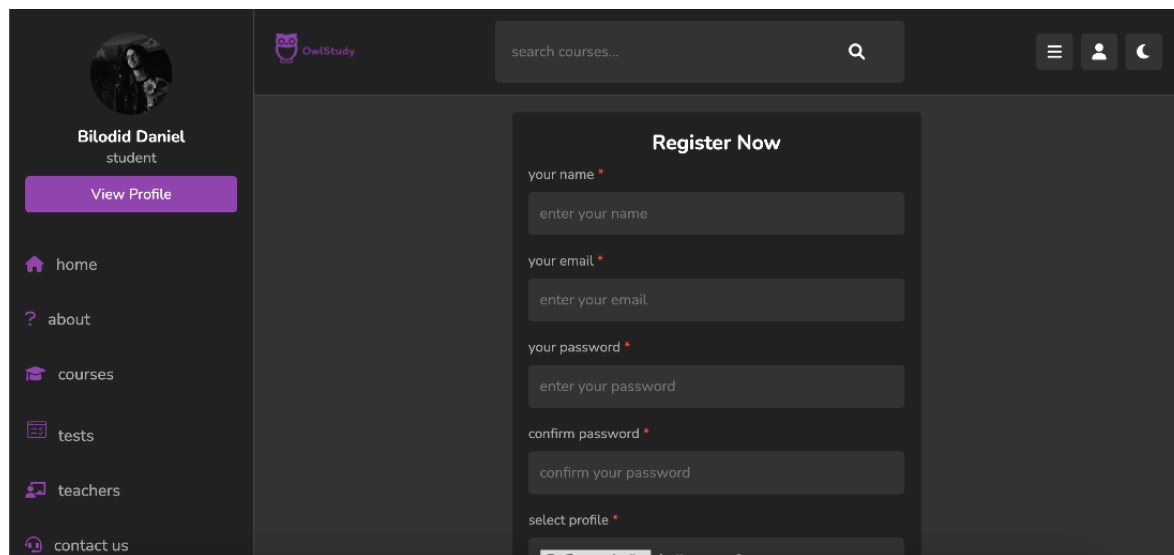


Рисунок 4.1.7 Register page веб-сервісу

Коли користувач хоче зв'язатися з адміністрацією веб-сервісу, він може натиснути на кнопку "Contact" в меню. Після цього він буде переправлений на сторінку з формою зворотного зв'язку. Форма містить поля для введення імені, електронної пошти, теми повідомлення та його тексту.

Крім того, на сторінці можуть бути надані контактні дані адміністрації, такі як електронна адреса та номер телефону. Ці дані можуть бути корисні користувачам, які хочуть зв'язатися з адміністрацією безпосередньо.

Форма зворотного зв'язку - це стандартний елемент HTML, який використовується для збору даних від користувачів. Вона може бути створена за допомогою різних тегів HTML, таких як `<form>`, `<input>`, `<textarea>` та інших. Форма може бути зв'язана з сервером, який обробляє дані, введені користувачем, та відправляє відповідь на веб-сторінку. Форма зворотного зв'язку дозволяє користувачам легко зв'язатися з адміністрацією веб-сервісу та надати їй потрібну інформацію для вирішення їхніх питань і проблем.

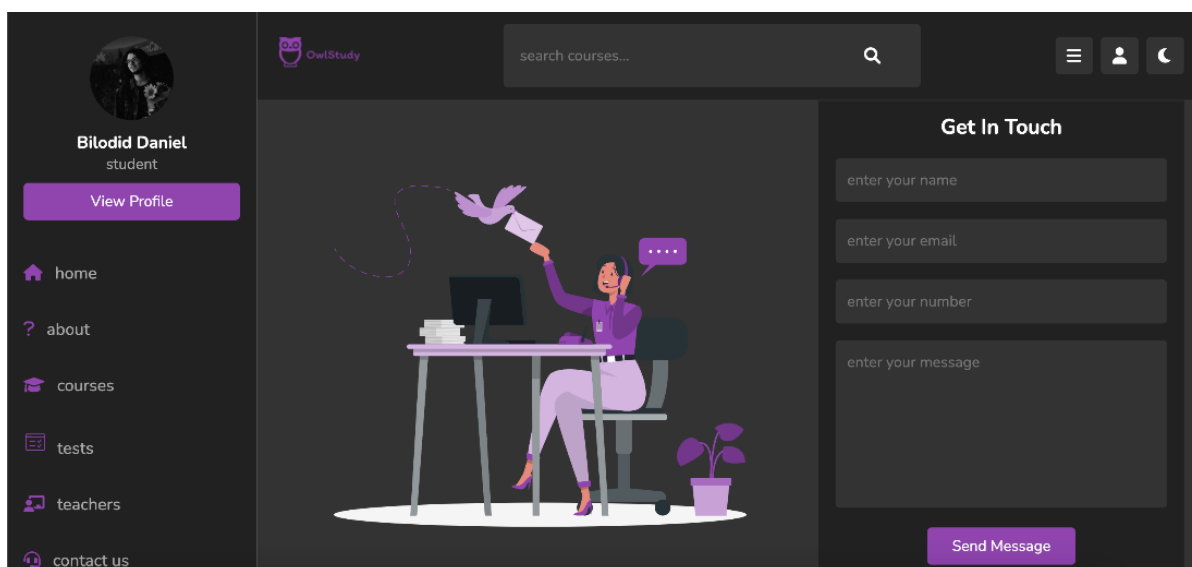


Рисунок 4.1.8 Contact page веб-сервісу

Профіль користувача на веб-сервісі з освітніми курсами - це особистий кабінет, де зберігається вся інформація про користувача та його активності на платформі. На сторінці профілю, користувач може переглянути свої дані, такі як ім'я, прізвище, електронну адресу, інформацію про платіжні реквізити та інші особисті дані, які він вказав під час реєстрації на сайті.

Крім того, на сторінці профілю зазвичай є можливість перегляду списку курсів, які користувач додав до обраного та курсів, на які він підписався. Також користувач може бачити коментарі та лайки, які він залишив на курсах, а також коментарі та лайки, які отримали його власні курси.

Залежно від функціональності веб-сервісу, на сторінці профілю користувач може змінювати свої особисті дані, додавати нові курси до обраного або видаляти їх, переглядати статистику своєї активності на сайті та інші налаштування. Профіль користувача на веб-сервісі з освітніми курсами є важливим елементом для зберігання та управління всіма активностями на платформі та робить користування веб-сервісом більш зручним та персоналізованим.

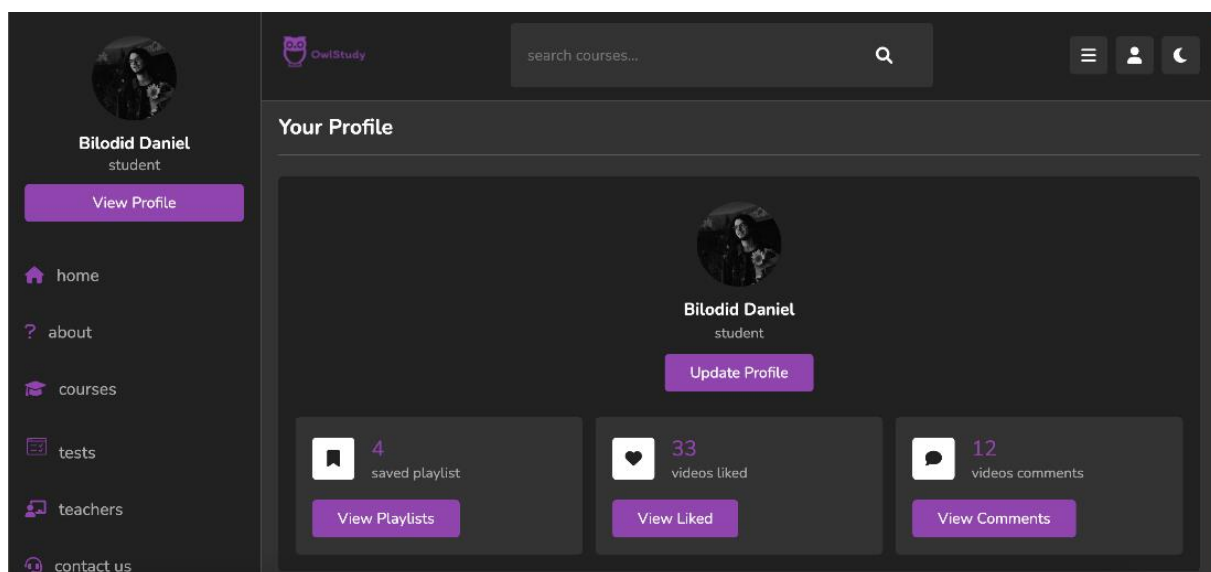


Рисунок 4.1.9 Профіль користувача

Профіль викладача - це сторінка на веб-сервісі, де користувач може дізнатися більше про конкретного викладача. На цій сторінці можна знайти інформацію про курси, які викладач пропонує, його досвід роботи та рейтинг серед учнів.

Перш за все, на сторінці профілю викладача можна побачити основну інформацію про нього, таку як його ім'я, фото, досвід роботи, освіту та спеціалізацію. Користувач може дізнатися, скільки років викладач має досвіду в певній галузі та які курси він пропонує.

Крім того, на сторінці профілю викладача можна знайти відгуки та лайки від учнів, які вже відвідували курси викладача. Це допомагає користувачам зрозуміти, наскільки ефективним та цікавим є курс, який викладає викладач.

Користувач може переглянути список курсів, які викладач пропонує, і дізнатися про їхній зміст та ціну. Кожен курс має свою сторінку з детальною інформацією, яку можна переглянути, натиснувши на назву курсу.

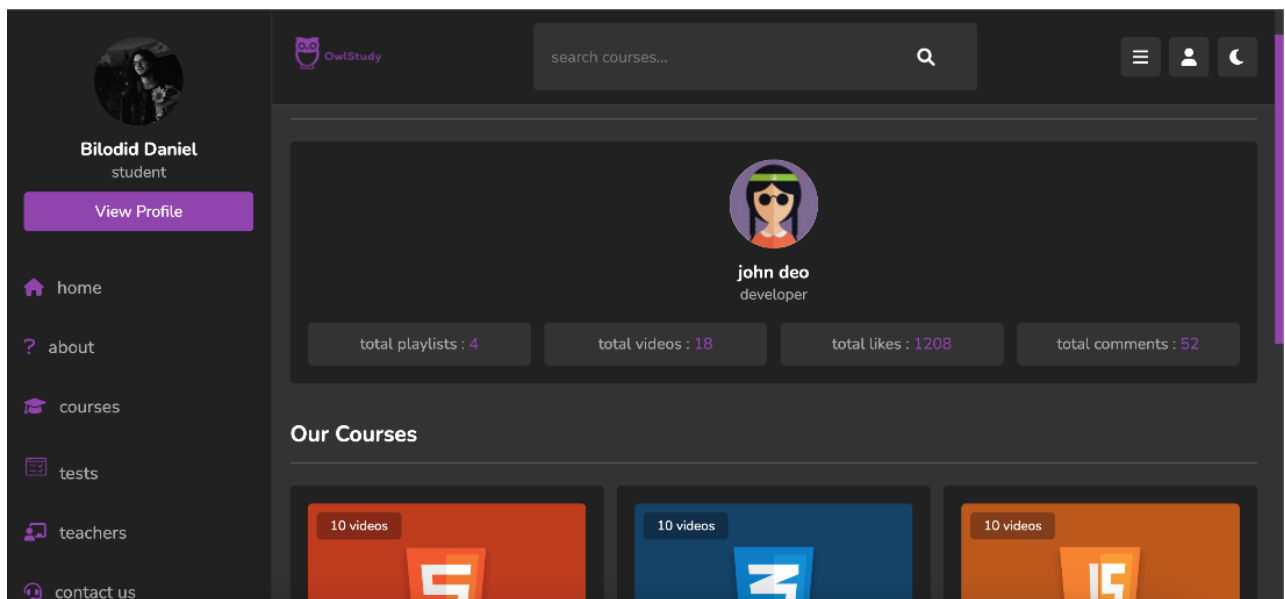


Рисунок 4.1.10 Профіль викладача

4.2 Тестування веб-сервісу

Реалізований веб-сервіс відповідає таким вимогам:

1. Пошукова система
2. Інтерфейс, що відповідає сучасним тенденціям та забезпечує зручність використання
3. Наявність тестування
4. Персональний профіль користувача
5. Підтримка різних мов

ВИСНОВКИ

У цій роботі було ставлено за мету створити веб-сервіс, який дозволяє користувачам проходити освітні курси. Для реалізації цього сервісу використовувалися мова розмітки гіпертексту HTML та об'єктивно-орієнтована прототипна мова програмування JavaScript.

В рамках даної роботи було проведено дослідження, яке мало на меті перевірити актуальність і наукову новизну даного проекту. Основним об'єктом дослідження були аспекти, пов'язані зі зручністю проходження курсів та інтересом сучасного суспільства до них.

У першому розділі, під назвою "АНАЛІЗ СТРУКТУРИ ІСНУЮЧИХ ВЕБ-СЕРВІСІВ ДЛЯ ПРОХОДЖЕННЯ ОСВІТНІХ КУРСІВ", було розглянуто загальне поняття веб-сервісу та визначено основні складові, які присутні в веб-сервісах для проходження освітніх курсів. Серед цих складових можна виділити: реєстрацію користувачів, наявність навчального контенту, організацію курсів, взаємодію з викладачами, оцінювання та моніторинг, а також сертифікацію.

Далі було проведено аналіз веб-сервісів, що пропонують схожі курси, і були взяті на увагу такі платформи: Coursera, Udemy, LinkedIn Learning, edX. Для цих платформ було проведено дослідження їх функціональності, наявності сучасного дизайну, а також були визначені загальні позитивні та негативні аспекти.

У другому розділі, під назвою "ДОСЛІДЖЕННЯ ІНСТРУМЕНТІВ ПРОЕКТУВАННЯ ВЕБ-СЕРВІСУ ДЛЯ ПРОХОДЖЕННЯ ОСВІТНІХ КУРСІВ", було проведено аналіз необхідних етапів та інструментів для розробки майбутнього веб-сервісу, зокрема, веб-сервісу для проходження освітніх курсів. В рамках дослідження були розглянуті середовища та інструменти для створення макетів та програмування веб-сервісів. Здійснений детальний аналіз цих середовищ дав змогу виявити їх переваги та недоліки для вибору найбільш зручного варіанту.

Також були визначені першочергові функції, які має мати веб-сервіс, та їх реалізація в першій черзі. Було проведено аналіз і встановлення необхідних

матеріалів, які будуть використовуватись під час подальшої розробки веб-сервісу. Також була створена логічна структура веб-сервісу та розроблені User Persona та User Flow, що сприятиме більш ефективному проектуванню та розробці веб-сервісу.

У третьому розділі, під назвою "ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБ-СЕРВІСУ ДЛЯ ПРОХОДЖЕННЯ ОСВІТНІХ КУРСІВ", було відзначено важливість прототипування як ключового етапу у створенні веб-сервісу. З використанням середовища для прототипування Figma було розроблено кожен унікальний сторінку веб-сервісу з метою подальшого поліпшення та розвитку самого сервісу. Крім того, було створено унікальний логотип для веб-сервісу.

Наступним кроком було проведено аналіз типографії для вибору найкращого шрифту, його розміру та кольору тексту. З урахуванням результатів аналізу кольорографії та психології кольорів були вибрані основні кольори для веб-сервісу.

У четвертому розділі, під назвою "ОПИС ВИКОРИСТАННЯ ТА ТЕСТУВАННЯ ВЕБ-СЕРВІСУ", був наданий детальний сценарій використання веб-сервісу користувачем. Описано інтерфейс та необхідні дії для досягнення користувачем цілей, пов'язаних з проходженням освітніх курсів.

Крім того, було успішно протестовано розроблений веб-сервіс, і він повністю відповідає всім вимогам, поставленим до нього.

В результаті виконання дипломної роботи було успішно розроблено веб-сервіс для проходження освітніх курсів, який повністю задовольняє всім вимогам, меті та цілям проекту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. MDN Web Docs (Mozilla Developer Network): [Електронний ресурс] Mozilla. – 2023. – Режим доступу до ресурсу: <https://developer.mozilla.org/>
2. HTML and CSS: Design and Build Websites // Jon Duckett
3. W3C (World Wide Web Consortium) [Електронний ресурс]. – Режим доступу до ресурсу: <https://www.w3.org/>
4. Codecademy [Електронний ресурс].– Режим доступу до ресурсу: <https://www.codecademy.com/>.
5. Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics" - Jennifer Niederst Robbins (<https://www.amazon.com/Learning-Web-Design-Beginners-JavaScript/dp/1449319270>)
6. JavaScript Documentation [Електронний ресурс] // Oracle. – 2023. – Режим доступу до ресурсу: <https://devdocs.io/javascript/>.
7. HTML Documentation [Електронний ресурс] // WHATWG. – Режим доступу до ресурсу: <https://devdocs.io/html/>.
8. CSS Documentation [Електронний ресурс] // World Wide Web Consortium. – Режим доступу до ресурсу: <https://devdocs.io/css/>.
9. Coursera [Електронний ресурс] – Режим доступу до ресурсу: <https://www.coursera.org> .
10. Udemy [Електронний ресурс] – Режим доступу до ресурсу: <https://www.udemy.com> .
11. Linkedin [Електронний ресурс] – Режим доступу до ресурсу: <https://www.linkedin.com> .
12. edX [Електронний ресурс] – Режим доступу до ресурсу: <https://www.edx.com>
13. "CSS: The Missing Manual" - David Sawyer McFarland – David Sawyer McFarland

ДОДАТОК А ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



РОЗРОБКА ВЕБ-СЕРВІСУ "OWLSTUDY" ДЛЯ ПРОХОДЖЕННЯ ОСВІТНІХ КУРСІВ НА БАЗІ JS, HTML, CSS

Виконав студент 4 курсу
Групи ПД-42
Білодід Даніель Вадимович
Керівник роботи
д.т.н., професор Бондарчук Андрій Петрович

1

МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

- **Мета роботи** – забезпечення якісного функціонування інформаційних освітніх платформ застосовуючи веб технології JS, HTML, CSS.
- **Об'єкт дослідження** – процес взаємодії з інформаційною системою при проходженні освітніх курсів.
- **Предмет дослідження** – веб технології для розробки електронних сервісів.

2

ЗАДАЧІ ДИПЛОМНОЇ РОБОТИ

1. Аналіз структури існуючих веб-сервісів для проходження освітніх курсів.
2. Дослідження інструментів проектування веб-сервісу для проходження освітніх курсів.
3. Програмна реалізація веб-сервісу для проходження освітніх курсів.
4. Опис використання веб-сервісу

3

АНАЛІЗ АНАЛОГІВ

Розглянуті веб-сервіси	Coursera 	Udemy 	LinkedIn Learning 	edX 	OWLstudy 
Мобільна версія	+	+	+	+	+
Наявність тестів	+	+	+	+	+
Доступність курсів	безкоштовні	від 449€	Від 559€	безкоштовні	безкоштовні
Зручність інтерфейсу	+	-	+	-	+
Відстеження прогресу	+	+	+	-	+
Локалізація на українську мову	+	-	-	-	+
Можливість завантажувати власні курси	-	+	-	+	-

4

ВИМОГИ ДО ВЕБ-СЕРВІСУ

1. Реєстрація та аутентифікація.
2. Каталог курсів.
3. Онлайн-лекції.
4. Матеріали курсу.
5. Взаємодія з викладачами та іншими учасниками.
6. Відстеження прогресу.
7. Оцінювання та звітність.
8. Тестування та оцінювання.
9. Сертифікати.

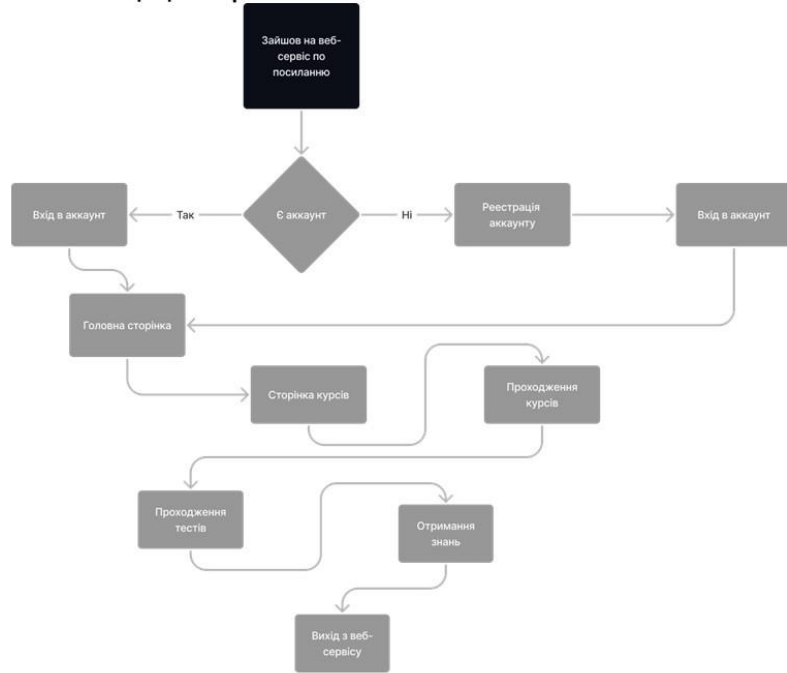
5

ПРОГРАМНІ ЗАСОБИ РЕАЛІЗАЦІЇ



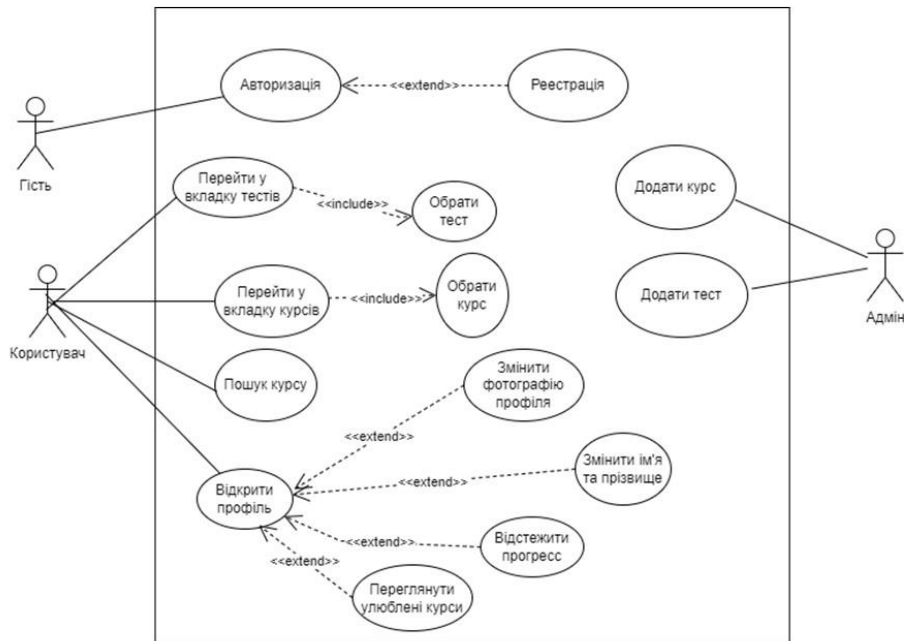
6

Діаграма userFlow



7


Діаграма варіантів використання



8

User Persona

Максим Левченко



"Amazing."

Рік: 22
 Робота: Працює
 Статус: Самотній,
 Локація: Україна, Київ
 Характер: Добрий

Особистість

Інтроверт	Екстроверт
Мислення	Почуття
Відчуття	Інтуїція
Судячий	Сприймаючий

Добрий
Екстраверт
Розумний

Цілі

- Вивчити щось нове.
- Практикувати свої навички.
- Стати професіоналом в своїй сфері.

Складності

- На веб-сервісах дуже важко вивчити щось бо нема гурпування матеріалу.
- Дуже важко практикувати свої навички
- Також немає підтримки вчителя та відсутність прямого зв'язку з іншими учнями.

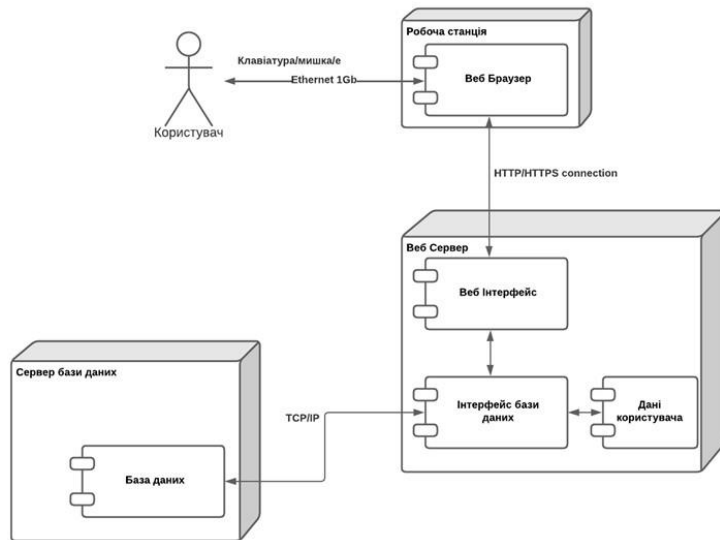
Біо

Студент ІТ вузу народився в Київ та виріс в Київ. Він завжди був захоплений технікою та комп'ютерами вже з дитинства. Він досліджував різні аспекти інформаційних технологій та проявляв інтерес до програмування, мереж та кібербезпеки.

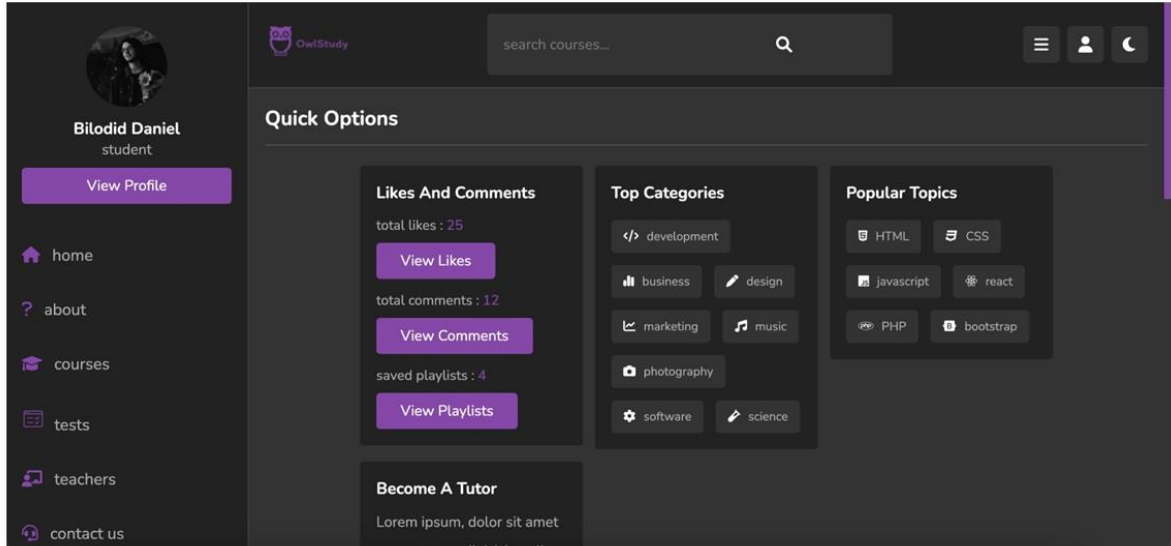
Мотивація

Стимул	
Страх	
Зростання	
Сила	
Соціальність	

АРХІТЕКТУРА ВЕБ-СЕРВІСУ

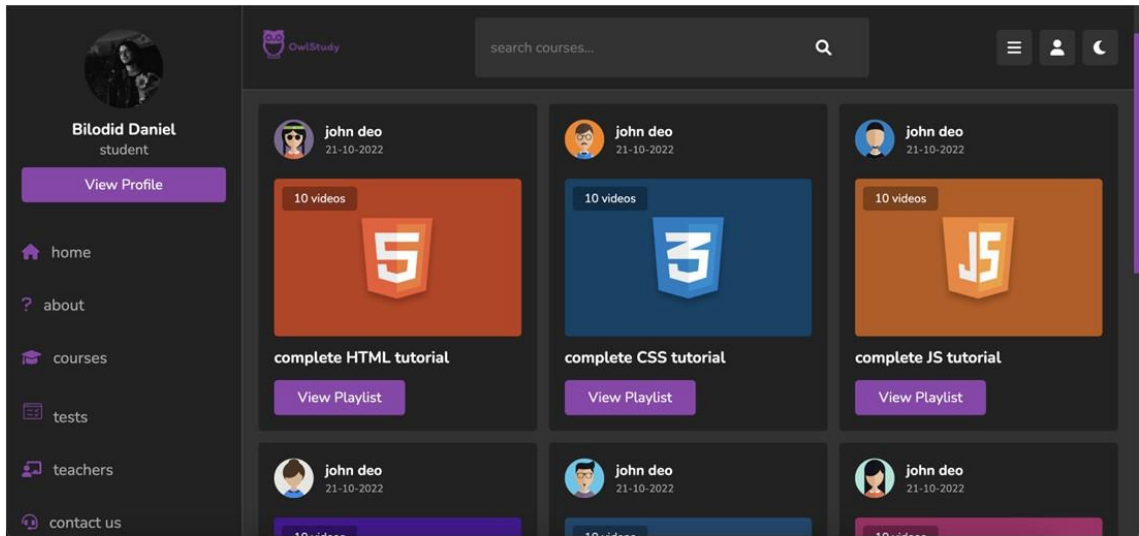


ГОЛОВНА СТОРІНКА ВЕБ-СЕРВІСУ



11

СТОРІНКА З КУРСАМИ ВЕБ-СЕРВІСУ



12

АПРОБАЦІЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

1. Білодід Д.В. Роль сучасного ПЗ в ІКТ /Білодід Д.В., Бондарчук А.П.// Сучасний стан та перспективи розвитку ІоТ: Матеріали всеукраїнської науково-технічної конференції. Збірник тез. 7.04. 2023, ДУТ К.: ДУТ, 2023. с. 237-239.
2. Білодід Д.В Аналіз використання ІоТ у навчальних курсах /Білодід Д.В., Бондарчук А.П.// Сучасний стан та перспективи розвитку ІоТ: Матеріали всеукраїнської науково-технічної конференції. Збірник тез. 7.04. 2023, ДУТ К.: ДУТ, 2023. с. 240-241.

13

ВИСНОВКИ

При виконанні бакалаврської кваліфікаційної роботи мною було:

1. Проаналізовані вже існуючі веб-сервіси для проходження освітніх курсів;
2. Розглянута підготовка до створення веб-сервісу для проходження освітніх курсів;
3. Проаналізовані інструменти для створення програмних забезпечень та в нашому випадку саме веб-сервісу;
4. Спроектований прототип веб-сервісу у Figma;
5. Розроблений веб-сервіс за допомогою HTML/CSS та JavaScript.

14

ДЯКУЮ ЗА УВАГУ!