

Міністерство освіти і науки України

Відокремлений структурний підрозділ «Тернопільський фаховий коледж
Тернопільського національного технічного університету імені Івана Пулюя»

(повне найменування вищого навчального закладу)

Відділення інформаційних технологій, менеджменту, туризму
та підготовки іноземних громадян

(назва відділення)

Циклова комісія комп'ютерної інженерії

(повна назва циклової комісії)

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи

фахового молодшого бакалавра

(освітньо-професійного ступеня)

на тему: Розробка інтернет-магазину з продажу кави “Initim”

Виконав: студент IV курсу, групи KI-412

Спеціальності 123 Комп'ютерна інженерія
(шифр і назва спеціальності)

Назар ПАРТИКА

(ім'я та прізвище)

Керівник

Леся ШТОКАЛО

(ім'я та прізвище)

Рецензент

(ім'я та прізвище)

**ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ
«ТЕРНОПІЛЬСЬКИЙ ФАХОВИЙ КОЛЕДЖ
ТЕРНОПІЛЬСЬКОГО НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ
імені ІВАНА ПУЛЮЯ»**

Відділення **інформаційних технологій, менеджменту, туризму
та підготовки іноземних громадян**

Циклова комісія **комп'ютерної інженерії**

Освітньо-професійний ступінь **фаховий молодший бакалавр**

Освітньо-професійна програма: **Обслуговування комп'ютерних систем і мереж**

Спеціальність: **123 Комп'ютерна інженерія**

Галузь знань: **12 Інформаційні технології**

ЗАТВЕРДЖУЮ

Голова циклової комісії
комп'ютерної інженерії

_____ Андрій ЮЗЬКІВ

“31” березня 2025 року

**З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**

Партиці Назару Віталійовичу

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи: **Розробка інтернет-магазину з продажу кави
“Initim”**

керівник роботи **Штокало Леся Ярославівна**

(прізвище, ім'я, по батькові)

Затверджені наказом Відокремленого структурного підрозділу «Тернопільський фаховий коледж Тернопільського національного технічного університету імені Івана Пулюя» від 28.03.2025р № 4/9-166а.

2. Строк подання студентом роботи: **13 червня 2025 року.**

3. Вихідні дані до роботи: **Visual Studio Code, Postman, Google Ghrome, React, Redux,
Django, Django REST Framework, MariaDB, Python, JavaScript.**

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити):
**Загальний розділ. Розробка технічного та робочого проєкту. Спеціальний розділ.
Економічний розділ. Охорона праці та безпека життєдіяльності.**

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

- структурна схема сайту;
- скрипт;
- блок схема до скрипта;
- таблиця техніко-економічних показників.

6. Консультанти розділів роботи

Розділ	Ім'я, прізвище та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний розділ	Богдана МАРТИНЮК викладач		
Охорона праці та безпека життєдіяльності	Володимир ШТОКАЛО викладач		

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Отримання і аналіз технічного завдання	01.04	
2	Збір і узагальнення інформації	05.05	
3	Написання першого розділу	16.05	
4	Розробка технічного та робочого проекту	23.05	
5	Написання спеціального розділу	30.05	
6	Розрахунок економічної частини	2.06	
7	Написання розділу охорони праці	4.06	
8	Виконання графічної частини	9.06	
9	Оформлення проєкту	11.06	
10	Погодження нормоконтролю	12.06	
11	Попередній захист роботи	13.06	
12	Захист кваліфікаційної роботи		

7. Дата видачі завдання: 01 квітня 2025 року

Студент

(підпис)

Назар ПАРТИКА

(ім'я та прізвище)

Керівник роботи

(підпис)

Леся ШТОКАЛО

(ім'я та прізвище)

АНОТАЦІЯ

Пояснювальна записка до кваліфікаційної роботи на тему «Розробка інтернет-магазину з продажу кави "Initim"» присвячена створенню функціонального веб-застосунку для продажу кавової продукції. Обсяг пояснювальної записки становить 78 сторінок.

Кваліфікаційна робота складається з п'яти розділів: в першому загальному розділі зроблено аналітичний огляд існуючих рішень, вказано на доцільність роботи та розроблено технічне завдання.

Другий – основний розділ, містить матеріал по вибору стеку на якому буде розроблено інтернет-магазин, де буде тестуватися цей веб-сайт, а також огляд структури та вмісту сайту.

Третій, спеціальний розділ містить інструкції з встановлення та налаштування веб-сайту, локального сервера, та програмного забезпечення які потрібні для його роботи.

Наступний, четвертий, економічний розділ містить розрахунок собівартості розробки веб-сайту.

В п'ятому розділі розглянуті питання охорони праці, техніки безпеки, та екологічних вимог.

Графічна частина проекту складає чотири плакати формату А3.

Кваліфікаційна робота носить практично-орієнтований характер і як частину містить повноцінно-функціонуючий сайт, який розміщений локально.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		

ANNOTATION

The explanatory note to the qualification work on the topic "Development of an online coffee store "Initim" is devoted to the creation of a functional web application for the sale of coffee products. The volume of the explanatory note is 78 pages.

The qualification work consists of five sections: the first general section provides an analytical review of existing solutions, indicates the feasibility of the work and develops the terms of reference.

The second, the main section, contains material on the choice of the stack on which the online store will be developed, where this website will be tested, as well as an overview of the structure and content of the site.

The third, special section contains instructions on how to install and configure the website, local server, and software required for its operation.

The next, fourth, economic section contains the calculation of the cost of developing a website.

The fifth section deals with health, safety, and environmental issues.

The graphic part of the project consists of four A3 posters.

The qualification work is practice-oriented and contains a fully functioning website hosted locally as part of it.

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		

ЗМІСТ

1 ЗАГАЛЬНИЙ РОЗДІЛ.....	7
1.1 Аналітичний огляд існуючих рішень.....	7
1.2 Технічне завдання	13
1.2.1 Найменування та область застосування	13
1.2.2 Призначення розробки	14
1.2.3 Вимоги до програмного забезпечення	15
1.2.4 Вимоги до програмної документації.....	15
1.2.5 Технічно-економічні показники	17
1.2.6 Стадії та етапи розробки	18
1.2.7 Порядок прийому та контролю сайту	20
2 РОЗРОБКА ТЕХНІЧНОГО ТА РОБОЧОГО ПРОЕКТУ	23
2.1 Постановка задачі на розробку програмного забезпечення	23
2.2 Опис та обґрунтування вибору методу організації вхідних та вихідних даних	26
2.3 Розробка алгоритму	29
2.3.1 Зовнішнє проектування програми.....	29
2.3.2 Проектування логіки програми	37
2.4 Написання скрипта.....	40
2.5 Тестування сайту.....	43
3 СПЕЦІАЛЬНИЙ РОЗДІЛ.....	48
3.1 Інструкція з інсталяції та налаштування веб серверів та супутнього програмного забезпечення	48
3.2 Інструкція по встановленню та налаштуванню розробленого проекту	50

					2025.КВР.123.412.13.00.00 ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Партика Н.В.			Розробка інтернет-магазину з продажу кави “Initim” Пояснювальна записка		Літ.	Арк.
Перевір.		Штокало Л.Я						Аркушів
Реценз.								
Н. Контр.		Юзьків А.В.					ВСП «ТФК ТНТУ ім. Івана Пулюя» КІ-412 м. Тернопіль	
Затверд.								

3.3 Інструкція з використання тестових наборів	51
3.4 Інструкція з експлуатації програмного комплексу	54
4 ЕКОНОМІЧНИЙ РОЗДІЛ.....	57
4.1 Визначення стадій технологічного процесу та загальної тривалості проведення НДР	57
4.2 Визначення витрат на оплату праці та відрахувань на соціальні заходи	58
4.3 Розрахунок витрат на електроенергію	60
4.4 Розрахунок суми амортизаційних відрахувань.....	60
4.5 Обчислення накладних витрат.....	61
4.6 Складання кошторису витрат та визначення собівартості НДР	62
4.7 Розрахунок ціни НДР	62
4.8 Визначення економічної ефективності і терміну окупності капітальних вкладень	63
5 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ.....	65
5.1 Пожежонебезпечні властивості речовин і матеріалів	66
5.2 Заходи щодо оптимізації виробничого освітлення	67
5.3 Моделювання та прогнозування небезпечних ситуацій	68
ВИСНОВКИ.....	71
ПЕРЕЛІК ПОСИЛАНЬ	72
ДОДАТОК А. КОД СТОРІНКИ СКРИПТА РЕЄСТРАЦІЇ	77
ДОДАТОК Б. КОД FRONTEND СТОРІНКИ З ВІДГУКАМИ	81
ДОДАТОК В. КОД BACKEND СТОРІНКИ З ШЛЯХАМИ ВІДГУКІВ	86
ДОДАТОК Г. КОД BACKEND СТОРІНКИ GET ЗАПИТІВ З ВІДГУКІВ	87

ВСТУП

Сучасний розвиток електронної комерції відкриває широкі можливості для бізнесу, зокрема у сфері роздрібної торгівлі спеціалізованими товарами, такими як кава. Зростання попиту на якісну кавову продукцію, а також потреба у зручних та швидких способах її придбання через інтернет зумовлюють актуальність створення спеціалізованих онлайн-платформ. Однак багато існуючих рішень для інтернет-магазинів не повною мірою відповідають потребам малого та середнього бізнесу, зокрема через складність у налаштуванні, обмежену гнучкість або недостатню адаптацію до специфічних вимог цільової аудиторії.

У рамках кваліфікаційної роботи розроблено інтернет-магазин «Initim» для продажу кави, який покликаний вирішити ці проблеми. Основною задачею роботи є створення функціонального веб-застосунку, що забезпечує зручний інтерфейс для користувачів, ефективне управління асортиментом, обробку замовлень та інтеграцію з платіжними системами. Робота спрямована на проектування модульної архітектури системи та впровадження сучасних технологій для забезпечення стабільності й масштабованості платформи.

Метою роботи є розробка інтернет-магазину, який поєднує інтуїтивно зрозумілий інтерфейс, адаптивний дизайн та гнучку систему управління контентом, що дозволяє легко адаптувати платформу до потреб бізнесу. Розробка здійснювалась із використанням сучасних технологій: React та Redux для фронтенду, Django та Django REST Framework для бекенду, а також MariaDB як базу даних. Ключові функції, як каталог товарів, кошик, пошук за назвою, обробку платежів та адміністративну панель для управління контентом.

Розроблений інтернет-магазин може бути використаний у сфері електронної комерції для продажу кавової продукції, а завдяки модульній структурі адаптований для реалізації інших товарів.

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		6

1 ЗАГАЛЬНИЙ РОЗДІЛ

1.1 Аналітичний огляд існуючих рішень

Веб-ресурси, орієнтовані на електронну комерцію, активно розвиваються протягом останніх років, що обумовлено зростанням попиту на зручні платформи для онлайн-покупок. Особливої популярності набули спеціалізовані інтернет-магазини, зокрема ті, що реалізують продукцію вузької спеціалізації — наприклад, каву. Для розробки таких систем необхідно враховувати не лише загальні принципи побудови веб-додатків, а й досвід створення подібних рішень, що вже існують на ринку.

Постановка проблеми

Розробка інтернет-магазину передбачає створення ефективного інструменту для управління товарами, обробки замовлень, взаємодії з клієнтом і забезпечення безпечних платіжних операцій. У зв'язку з цим виникає потреба у виборі таких програмних засобів, які забезпечать гнучкість, масштабованість і відповідність функціональним вимогам, викладеним у технічному завданні.

Виробниче проєктування веб-додатка розпочинається з вивчення вже існуючих реалізацій — прототипів, платформ, CMS та фреймворків. Частину інформації було отримано в результаті тестування демо-версій популярних систем, іншу — з технічної документації та спеціалізованої літератури.

Аналіз сучасних інструментів і підходів

CMS-системи для електронної комерції

Одним з традиційних рішень для створення інтернет-магазину є використання систем управління контентом. Серед них варто виділити:

- WordPress + WooCommerce - зручне рішення для невеликих магазинів. Має широкий вибір тем і плагінів, однак обмежене з точки зору

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7

масштабування та кастомізації логіки.

- OpenCart - орієнтований саме на e-commerce, пропонує готові модулі та шаблони. Потребує глибшого втручання при зміні бізнес-логіки.

- Magento - потужна платформа для великих інтернет-магазинів, але потребує серйозних ресурсів і технічних знань для розгортання та підтримки.

Фреймворки для гнучкої розробки

Для створення унікальних, адаптованих під конкретний бізнес веб-додатків частіше використовують фреймворки. Вони забезпечують гнучкість у розробці і краще контролю над функціональністю:

- Django (Python) - один з найпотужніших веб-фреймворків. Його структура «все включено» дозволяє швидко створити повнофункціональний веб-застосунок. У поєднанні з Django REST Framework (DRF) стає ефективним засобом розробки API для frontend-частини.[11]

- Flask (Python) - мінімалістичний мікрофреймворк, який підходить для невеликих проєктів або прототипів. Проте потребує більше ручної роботи для побудови повної архітектури сайту.[17]

- Express.js (Node.js) - популярний JavaScript-фреймворк для backend-розробки. Добре інтегрується з React, але не надає стільки інструментів із коробки, як Django.[18]

Інструменти для фронтенду

- React - JavaScript-бібліотека, яка забезпечує побудову швидкого, інтерактивного інтерфейсу. Широко використовується в сучасних проєктах. Має велику спільноту та підтримує JSX, компонентну архітектуру та підключення до API.[13]

- Redux - менеджер стану для React-додатків, який дозволяє централізовано керувати даними між компонентами, що особливо корисно в складних додатках з великою кількістю взаємодій.[14]

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

Бази дани

- MySQL / MariaDB - надійні реляційні СУБД із відкритим кодом, що добре підходять для структурованих проєктів, які потребують транзакційної цілісності.
- PostgreSQL - ще одна потужна реляційна СУБД, яка має розширені можливості в обробці складних запитів.
- MongoDB - нереляційна база даних, зручна для зберігання документно-орієнтованих даних, але менш підходить для класичних інтернет-магазинів зі складною структурою замовлень.[12]

Візуалізація технічного рішення

Для кращого розуміння обраного підходу до розробки веб-додатка, доцільним є використання візуальних засобів подання архітектури:

- Схема клієнт-серверної взаємодії - для відображення потоків даних між frontend і backend.
- Логічна структура сайту - навігаційна ієрархія сторінок (товари, корзина, оформлення замовлення тощо).
- Схема бази даних - з описом таблиць, полів і зв'язків (ER-діаграма).

1. CoffeeOK - великий мультибрендовий інтернет-магазин кави.

- Посилання: <https://coffeeok.com.ua/ua/>
- Платформа: CMS OpenCart
- Опис: CoffeeOK є одним із найбільших українських онлайн-магазинів кави та супутніх товарів. Асортимент включає зернову, мелену, капсульну каву, чай, кавомашини, аксесуари та засоби для догляду. Сайт має класичну структуру з фільтрами, кошиком, особистим кабінетом і блогом.
- Особливості:
- Широкий вибір брендів (Lavazza, Totti, Kimbo тощо)
- Акції, знижки, подарунки

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

- В наявності є платіжна система та доставка
- Наявність мобільної версії та адаптивного дизайну

На рисунку 1.1 наведено головну сторінку інтернет-магазину CoffeeOK.

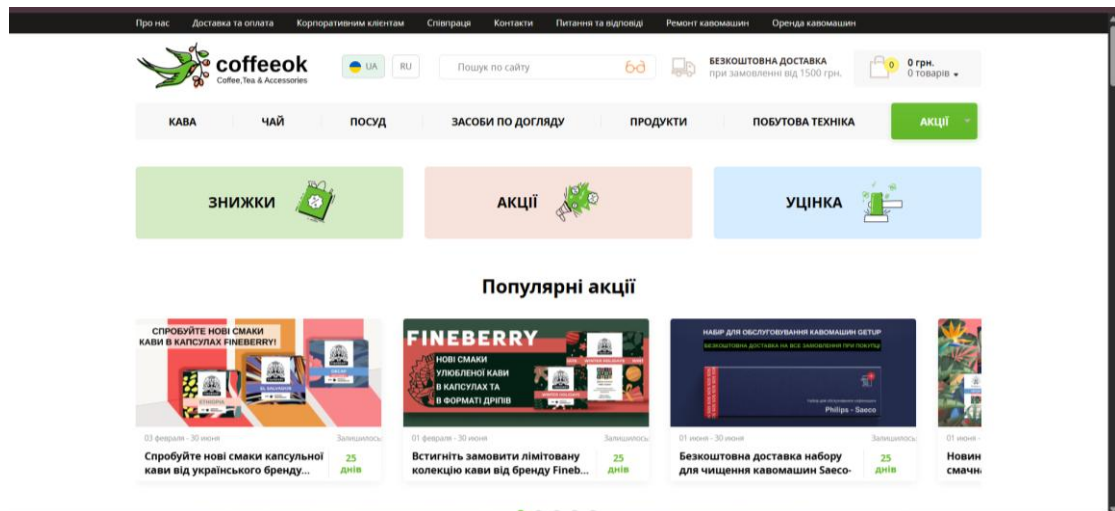


Рисунок 1.1 – Головна сторінка інтернет-магазину CoffeeOK

2. Екава- спеціалізований магазин кави та обладнання.

- Посилання: <https://ekava.com.ua/>
- Платформа: CMS WordPress + WooCommerce
- Опис: Екава орієнтований на продаж кави, чаю, кавового обладнання та аксесуарів. Асортимент включає зернову та мелену каву, кавомашини, вендінгові автомати, аксесуари для бариста та одноразовий посуд.

- Особливості:
- Підтримка корпоративних клієнтів та HoReCa-сегменту
- Оренда кавового обладнання
- Консультації з підбору продукції
- Можливість замовлення через сайт та соціальні мережі

На рисунку 1.2 наведено головну сторінку інтернет-магазину Екава.

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

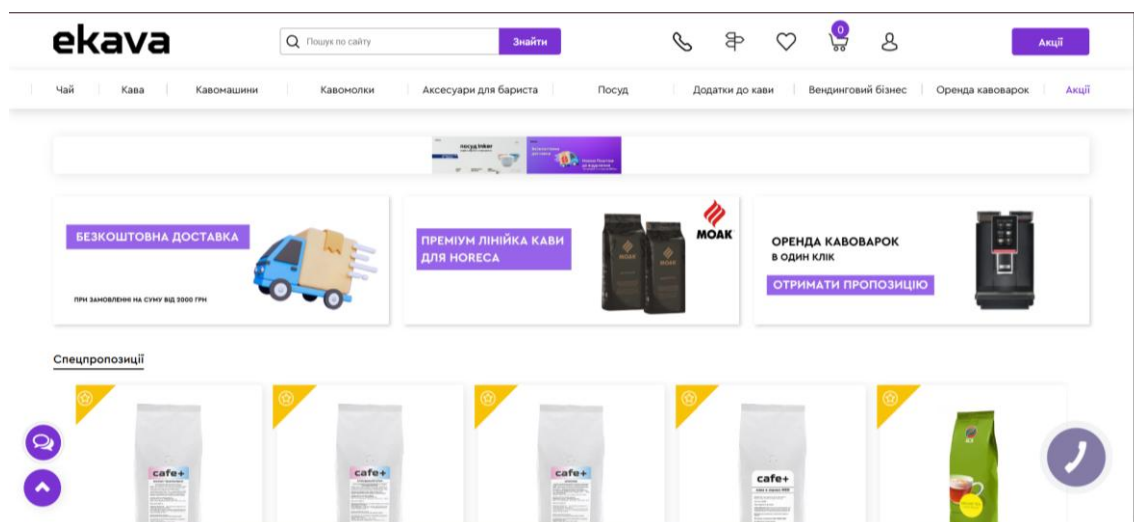


Рисунок 1.2 – Головна сторінка інтернет-магазину Ekava

3. Coffee Town - магазин свіжообсмаженої кави.

- Посилання: <https://coffee.town/>
- Платформа: Фреймворк Django (Python)
- Опис: Coffee Town спеціалізується на продажу свіжообсмаженої кави власного виробництва. Компанія ретельно відбирає сорти, обсмажує зерна повітряним методом кожні 3–4 дні та пропонує продукцію як для дому, так і для офісів.
- Особливості:
 - Власне обсмаження зерен
 - Можливість вибору ступеня обсмаження
 - Інформація про походження кожного сорту
 - Сучасний адаптивний дизайн з акцентом на користувацький досвід

На рисунку 1.3 наведено головну сторінку інтернет-магазину Coffee Town.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

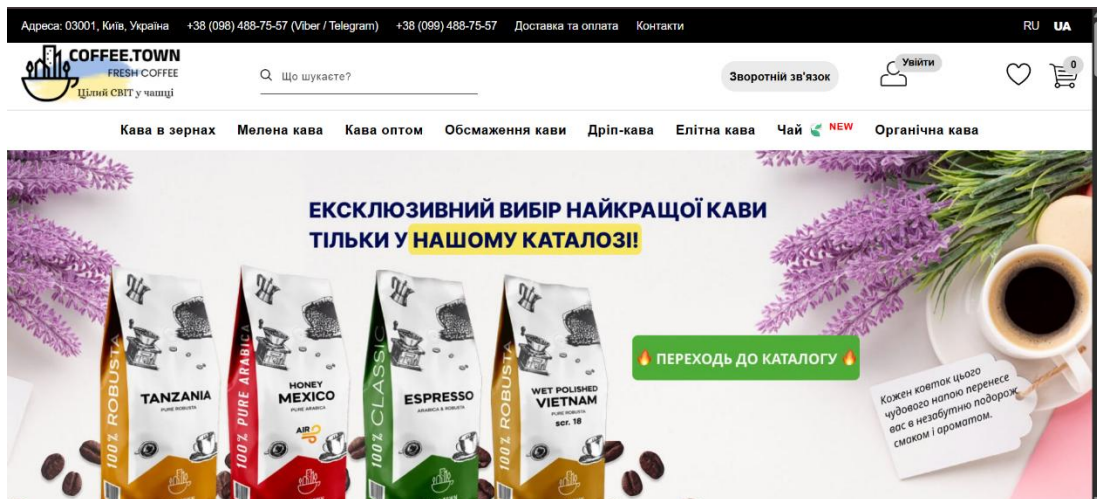


Рисунок 1.3 – Головна сторінка інтернет-магазину Coffee Town

Таблиця 1.1 – Порівняльна таблиця інтернет-магазинів

Критерій	CoffeeOK	Ekava	Coffee Town
Тип продукції	Кава, чай, обладнання	Кава, обладнання	Свіжо-обсмажена кава
Цільова аудиторія	Широке коло споживачів	Домашні та корпоративні клієнти	Поціновувачі свіжої кави
Асортимент	Широкий	Середній	Власне виробництво
Унікальні особливості	Акції, подарунки	Оренда обладнання	Власне обсмаження
Інтерфейс	Класичний	Мінімалістичний	Сучасний, адаптивний
Платформа	Open Cart	WordPress + WooCommerce	Django (Python)

Таким чином, аналіз наявних рішень у сфері розробки веб-додатків для електронної комерції показав, що готові CMS-системи обмежені у можливості

створення унікальної логіки, інтерфейсу та інтеграцій. Тому для реалізації функціонального та масштабованого інтернет-магазину з продажу кави "Initim" було обрано сучасний стек технологій:

- Frontend: React + Redux - для створення взаємодії та пристосування для інтерфейсу користувача; [14]
- Backend: Django + Django REST Framework - для побудови надійного серверного API; [11]
- Database: MariaDB - як ефективне рішення для зберігання структурованих даних про товари, замовлення та користувачів. [12]

Обрана архітектура дозволяє забезпечити гнучкість, швидкодію, зручність адміністрування і підтримку у майбутньому. Додаткове візуальне представлення - у вигляді схем - дає змогу краще представити взаємозв'язки всіх компонентів системи.

1.2 Технічне завдання

1.2.1 Найменування та область застосування

Ключовим завданням кваліфікаційної роботи є розробка інтернет-магазину з продажу кави під назвою "Initim". Сайт призначений для використання як на платформах для продажу кавових продуктів, включаючи зернову каву, мелену каву, кавові аксесуари та супутні товари.

Область застосування сайту охоплює:

- Інтернет-магазини, які спеціалізуються на продажі кави та супутніх товарів.
- Кав'ярні, які бажають розширити свою діяльність в онлайн-просторі.
- Користувачів, які шукають зручний спосіб замовлення кави з

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

доставкою додому.

Сайт забезпечує зручний інтерфейс для користувачів, можливість перегляду каталогів товару, оформлення замовлення, а також інтеграцію з платіжними системами для здійснення онлайн-оплат.

1.2.2 Призначення розробки

Метою даної кваліфікаційної роботи є створення інтернет-магазину з продажу кави “Initim”, який дозволяє користувачам переглядати каталог товарів, оформлювати замовлення та здійснювати оплату онлайн. Розробка здійснена з використанням таких середовищ та технологій:

- Frontend: React + Redux для створення інтерактивного інтерфейсу користувача.[14]
- Backend: Django + Django REST Framework для реалізації серверної частини та API.[11]
- База даних: MariaDB для зберігання даних про товари, замовлення та користувачів.[12]
- VSCode: для написання коду.[26]
- Google Chrome: для відображення інтерактивного інтерфейсу користувача.[27]
- Postman: для тестування API запитів таких як GET, POST, PUT.[25]
- HeidiSQL: графічний інтерфейс MariaDB, для доступу до бази даних.[12]

Сайт розроблений для розміщення в мережі інтернет та подальшого використання як інформаційного та комерційного ресурсу. Він забезпечує зручний доступ до товарів для клієнтів, а також можливості адміністрування для власників магазину.

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

1.2.3 Вимоги до програмного забезпечення

Метою даної кваліфікаційної роботи є: написання сайту для інтернет-магазину з продажу кави “Initim”.

Вимоги до сайту:

1. Сайт мусить чудово відображатися і функціонувати в найпоширеніших браузерях:
 - Google Chrome (версія 90 і вище)
 - Mozilla Firefox (версія 88 і вище)
 - Microsoft Edge (версія 90 і вище)
 - Opera (версія 75 і вище)
2. Сайт повинен бути адаптованим під різні розширення екрану, включаючи мобільні пристрої, планшети та десктопи.

1.2.4 Вимоги до програмної документації

Програмна документація для інтернет-магазину “Initim” повинна відповідати наступним вимогам:

1. Повнота:
 - Документація повинна містити всі необхідні розділи, включаючи опис архітектури системи, інструкції з розгортання, використання та адміністрування сайту.
 - Опис API, включаючи всі доступні ендпоінти, методи запитів, формати даних та приклади використання.
2. Зрозумілість:
 - Документація повинна бути написана зрозумілою мовою, з використанням термінів, які відповідають галузевим стандартам.
 - Використання схем, діаграм та прикладів для пояснення складних

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

концепцій.

3. Актуальність:

- Документація повинна бути актуальною та відповідати поточній версії програмного забезпечення.

- Усі зміни в коді повинні супроводжуватися відповідними оновленнями в документації.

4. Структурованість:

- Документація повинна бути структурована за розділами, такими як:

1. Вступ.

2. Опис архітектури.

3. Інструкція з розгортання.

4. Інструкція з використання.

5. Інструкція з адміністрування.

6. Опис API.

7. Часті запитання (FAQ).

5. Формат:

- Документація повинна бути доступною в електронному вигляді PDF.

- Для API може бути використаний формат OpenAPI (Swagger) для автоматичної генерації документації.

6. Інструкції з розгортання:

- Документація повинна містити покрокові інструкції з налаштування середовища розробки, встановлення залежностей, запуску серверу та розгортання сайту.

7. Інструкція з використання:

- Опис функціоналу сайту для кінцевих користувачів, включаючи реєстрацію, авторизацію, перегляд каталогу, оформлення замовлень та оплату.

8. Інструкція з адміністрування:

- Опис функцій адміністратора, таких як управління товарами, перегляд

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

замовлень, управління користувачами.

9. Тестова документація:

- Опис тестових сценаріїв, які використовувалися для перевірки функціональності сайту.

10. Контактна інформація:

- Вказівка контактів розробника для зворотного зв'язку або отримання технічної підтримки.

1.2.5 Технічно-економічні показники

У процесі створення сайту для інтернет-магазину з продажу кави “Initim” використовувалися наступні технічні та програмні засоби:

Технічні засоби:

1. Ноутбук:

- Процесор: AMD Ryzen 7 7435HS (8 ядер, 16 потоків, 3.1-4.7 ГГц).
- Оперативна пам'ять: 16 ГБ DDR5 (5600 Mhz).
- Жорсткий диск: SSD 512 ГБ.
- Дисплей: 15.6-дюймовий екран з роздільною здатністю 1920x1080.
- Операційна система: Linux Ubuntu 24.04.2 LTS.

2. Локальні сервери: які розміщені у папці backend, backend сервер <http://127.0.0.1:8000/>, frontend сервер <http://localhost:3000/>, розміщені на ноутбуці для тестування та розробки.[11]

Програмні засоби:

1. Frontend:

- React (версія 18.2.0).
- Redux (версія 4.2.1).
- Bootstrap.min.css (версія 6.7.2).

2. Backend:

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		17

- Django (версія 5.2).
- Django REST Framework (версія 3.16.0).

3. База даних:

- MariaDB (версія 11.7.2)

4. Інші інструменти:

- Node.js (версія 22.15.0) для управління залежностями та побудови фронтенду.

- Python (версія 3.13.3) для розробки серверної частини.

- Postman (версія 11.48.5) для тестування API

- Google Chrome (версія 137.0.7151.56) для доступу до локальних серверів.

- VS Code (версія 1.99.3) для написання коду.

Ці технічні та програмні засоби забезпечили ефективну розробку, тестування та запуску сайту.

1.2.6 Стадії та етапи розробки

Весь обсяг робіт по створенню веб-сайту поділяється на наступні етапи:

1. Аналіз вимог:

- Визначення цільової аудиторії.
- Формування технічного завдання.
- Аналіз конкурентів та визначення ключових функцій сайту.

2. Проектування:

- Розробка структури сайту.
- Створення макетів інтерфейсу користувача.
- Планування бази даних.

3. Розробка:

- Frontend:

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

1. Створення інтерфейсу користувача за допомогою React.
2. Реалізація управління станом додатку через Redux.
 - Backend:
 1. Розробка API за допомогою Django REST Framework.
 2. Налаштування бази даних MariaDB.
 - Інтеграція frontend і backend частин.
4. Тестування:
 - Перевірка функціональності сайту.
 - Тестування адаптивності на різних пристроях та браузерах.
 - Виправлення помилок.
5. Розгортання:
 - Налаштування серверного середовища.
 - Розгортання сайту локально.
 - Перевірка роботи сайту на сервері.
6. Підтримка та оновлення:
 - Моніторинг роботи сайту.
 - Додавання оновлень та змін згідно до потреб користувачів.
7. Оптимізація:
 - Оптимізація коду для підвищення продуктивності.
 - Зменшення розміру статичних файлів за допомогою мініфікації.
 - Налаштування кешування для швидкого завантаження сторінок.
8. Інтеграція з зовнішніми сервісами:
 - Підключення платіжних систем для обробки онлайн-оплат.
 - Інтеграція з сервісами доставки для автоматизації обробки замовлень.
 - Використання сторонніх API для додаткових функцій.
9. Документування:
 - Створення технічної документації для розробників.
 - Написання інструкцій для користувачів та адміністраторів сайту.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

- Генерація документації API за допомогою OpenAPI (Swagger).

10. Розгортання та підтримка:

- Розгортання сайту на хостингу або сервері.
- Налаштування моніторингу для відстеження продуктивності та помилок.
- Регулярне оновлення для забезпечення безпеки та додавання нових функцій.

1.2.7 Порядок прийому та контролю сайту

Прийом готового веб-сайту здійснюється відповідно до домовленості між замовником і розробником. Порядок контролю та прийому включає наступні етапи:

1. Перевірка функціональності:

- Тестування всіх основних функцій сайту, таких як перегляд каталогу товарів, оформлення замовлень, реєстрація та авторизація користувачів.
- Перевірка роботи платіжної системи.

2. Перевірка адаптивності:

- Тестування сайту на різних пристроях (мобільні телефони, планшети, десктопи).
- Перевірка коректного відображення сайту на різних розширеннях екрану.

3. Перевірка сумісності з браузером:

- Тестування роботи сайту в браузерах Google Chrome, Mozilla Firefox, Microsoft Edge, Opera відповідно до зазначених версій.

4. Перевірка продуктивності:

- Оцінка швидкості завантаження сторінок.
- Перевірка роботи сайту під навантаженням.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

5. Прийом замовником:

- Замовник перевіряє відповідність сайт технічному завданню.
- Підписання акту прийому-передачі після успішного завершення перевірок.

6. Розгортання в продакшин:

- Розміщення сайту на хостингу або сервері.
- Перевірка роботи сайту в реальному середовищі.

Перевірка працездатності інтернет-магазину “Initim” здійснюється шляхом тестування функціональності сайту на основі контрольного прикладу та набору тестів.

Обсяг контрольного прикладу:

1. Тестові дані:

- Створення тестового каталогу товарів (10 позицій кави з різними характеристиками).
- Реєстрація тестових користувачів (3 користувачі з різними ролями: адміністратор, зареєстрований користувач, гість).
- Імітація замовлення товарів з різними параметрами(усі поля які потрібно ввести при оформленні замовлення).

2. Тестові сценарії:

- Перегляд каталогу товарів.
- Реєстрація та авторизація користувачів.
- Оформлення замовлення.
- Перевірка роботи платіжної системи.
- Адміністрування товарів (додавання, редагування, видалення).

Набір тестів:

1. Функціональні тести:

- Тестування основних функцій сайту (каталог, кошик, оформлення замовлення).

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

- перевірка роботи API (отримання даних про товари, користувачів, створення замовлень).

2. Тести адаптивності:

- Перевірка коректного відображення сайт на різних пристроях (мобільні телефони, планшети, десктопи).

- Тестування роботи сайту на різних розширеннях екрану.

3. Тести сумісності:

- Перевірка роботи сайту в браузерях Google Chrome, Mozilla Firefox, Microsoft Edge, Opera.

4. Тести продуктивності:

- Оцінка швидкості завантаження сторінок.

- Перевірка роботи сайту під навантаженням (одночасний доступ кількох користувачів)

5. Тести Безпеки:

- Перевірка захисту даних користувачів.

Умови прийому:

- Сайт вважається прийнятим, якщо всі тести успішно пройдені.

- Замовник підтверджує відповідність сайту технічному завданню.

- Підписується акт прийому-передачі після завершення перевірок.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

2 РОЗРОБКА ТЕХНІЧНОГО ТА РОБОЧОГО ПРОЕКТУ

2.1 Постановка задачі на розробку програмного забезпечення

Загальний опис задачі

Розробка інтернет-магазину "Initim" спрямована на створення зручної платформи для продажу кави та супутніх товарів. Основна мета — забезпечити користувачам можливість переглядати каталог товарів, оформлювати замовлення, здійснювати оплату онлайн, адміністрування товарів, користувачів і замовлень а також отримувати інформацію про статус замовлення.

Система повинна бути адаптивною, працювати в різних браузерх і підтримувати різні розширення екранів. Для реалізації проекту використовується сучасний стек технологій, що включає React + Redux для фронтенду, Django + Django REST Framework для бекенду, а також MariaDB для зберігання даних. У проекті реалізовано функціонал на обробки замовлень, включаючи розрахунок загальної вартості замовлень.

Специфічні особливості реалізації задачі

1. Реалізація API:

- Для взаємодії між фронтендом і бекендом використовується REST API, розроблене за допомогою Django REST Framework.
- API забезпечує доступ до даних про товари, замовлення та користувачів, а також підтримує CRUD-операції.[11]

2. Адаптивність:

- Інтерфейс користувача розроблений з урахуванням адаптивного дизайну, що дозволяє коректно відображати сайт на мобільних пристроях, планшетах і десктопах.

3. Інтеграція з платіжними системами:

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		23

- Для обробки онлайн-оплат через API передбачена інтеграція з популярним платіжним сервісом PayPal, що забезпечує безпечність і зручність транзакцій.[28]

4. Адміністрування:

- Адміністративна панель дозволяє управляти товарами, замовленнями та користувачами.

5. Безпека:

- Реалізовано захист даних користувачів, включаючи шифрування паролів, захист від SQL-ін'єкцій та інших поширених атак.[15]

Математична модель

Для розрахунку загальної вартості замовлення використовується наступна формула:

$$S_{total} = \sum_{i=1}^n (P_i * Q_i) + S_{shipping} + S_{tax}$$

де:

- (S total) — загальна вартість замовлення;
- (S tax) — податок;
- (P i) — ціна одиниці товару (i);
- (i) — одиниця товару;
- (Q i) — кількість одиниць товару (i);
- (S shipping) — фіксована вартість доставки;
- (n) — кількість товарів у замовленні.

Ця формула дозволяє автоматизувати розрахунок вартості замовлення на основі даних, що вводяться користувачем.[7]

Опис вхідної та вихідної інформації

1. Вхідна інформація:

- Дані про товари:

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						24
Змн.	Арк.	№ докум.	Підпис	Дата		

- Назва товару.
 - Відгук про товар.
 - Ціна.
 - Кількість.
 - Зображення.
 - Категорія.
 - Опис.
 - Дані про користувачів:
 - Ім'я.
 - Місто.
 - Поштовий індекс.
 - Адреса.
 - Електронна пошта.
 - Дані про замовлення:
 - Список товарів.
 - Кількість кожного товару.
 - Спосіб оплати.
 - Сума оплати.
 - Статус оплати.
 - Статус доставки.
 - Адреса доставки.
 - 2. Вихідна інформація:
 - Загальна вартість замовлення.
 - Список товарів у замовленні.
 - Статус замовлення (оформлено, оплачено, доставлено).
 - Інформація про доставку.
- Використані методи
1. Метод REST:

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

- REST API забезпечує стандартизовану взаємодію між фронтендом і бекендом, дозволяючи виконувати операції створення, читання, оновлення та видалення даних.[11]

2. Метод адаптивного дизайну:

- Використання медіа-запитів у CSS для забезпечення коректного відображення сайту на різних пристроях.

3. Методи роботи з базою даних:

- CRUD-операції для управління товарами, замовленнями та користувачами.[16]

- Оптимізація запитів до бази даних для підвищення продуктивності.

4. Методи безпеки:

- Використання бібліотек для шифрування паролів (Django's make_password).[15]

- Захист від SQL-ін'єкцій через використання ORM Django.[15]

5. Методи тестування:

- Тести для перевірки функціональності окремих компонентів.
- Інтеграційні тести для перевірки взаємодії між фронтендом і бекендом.

2.2 Опис та обґрунтування вибору методу організації вхідних та вихідних даних

Структура та метод організації даних

Для інтернет-магазину "Initim" організація вхідних та вихідних даних базується на використанні реляційної бази даних MariaDB. Цей вибір обґрунтований її високою продуктивністю, підтримкою транзакцій, масштабованістю та сумісністю з Django ORM, що спрощує роботу з даними,

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

та захист від SQL-ін'єкцій.[15]

1. Вхідні дані:

- Дані про товари:

- Назва товару.
- Опис товару.
- Ціна.
- Зображення.
- Категорія.
- Кількість.

- Дані про користувачів:

- Ім'я.
- Електронна пошта.
- Пароль (зашифрований).
- Адреса доставки.

- Дані про замовлення:

- Список товарів.
- Кількість кожного товару.
- Спосіб оплати.
- Статус замовлення.

2. Вихідні дані:

- Для користувачів:

- Каталог товарів.
- Інформація про замовлення (загальна вартість, статус, деталі

доставки).

- Для адміністратора:

- Статистика продажів.
- Список замовлень із деталями.
- Інформація про користувачів.

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

Ієрархічна структура даних

1. Товари:

- Категорія → Товар → Опис, Ціна, Зображення.

2. Користувачі:

- Користувач → Замовлення → Товари в замовленні.

3. Замовлення:

- Замовлення → Товари → Кількість, Ціна.

Упорядкування вхідних даних

Для забезпечення ефективної роботи з даними прийнято наступні рішення:

1. Ключі упорядкування:

- Для таблиці товарів: первинний ключ id, індексація за полями category та price для швидкого пошуку.

- Для таблиці користувачів: первинний ключ id, унікальний ключ email.

- Для таблиці замовлень: первинний ключ id, індексація за полем user_id для швидкого доступу до замовлень конкретного користувача.

2. Упорядкування даних:

- Дані про товари сортуються за категоріями.

- Замовлення сортуються за датою створення.

Обґрунтування вибору

1. Реляційна модель:

- Використання реляційної моделі даних дозволяє ефективно організувати зв'язки між таблицями (наприклад, зв'язок між користувачами та їх замовленнями).[12]

2. Django ORM:

- Django ORM автоматично генерує SQL-запити, що спрощує роботу з базою даних і забезпечує захист від SQL-ін'єкцій.[15]

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

3. Індексція:

- Індексція ключових полів значно підвищує швидкість пошуку та фільтрації даних.

2.3 Розробка алгоритму

2.3.1 Зовнішнє проектування програми

Структурна схема сайту інтернет-магазину “Initim”:

На рисунку 2.1 наведено структурну схему сайту.

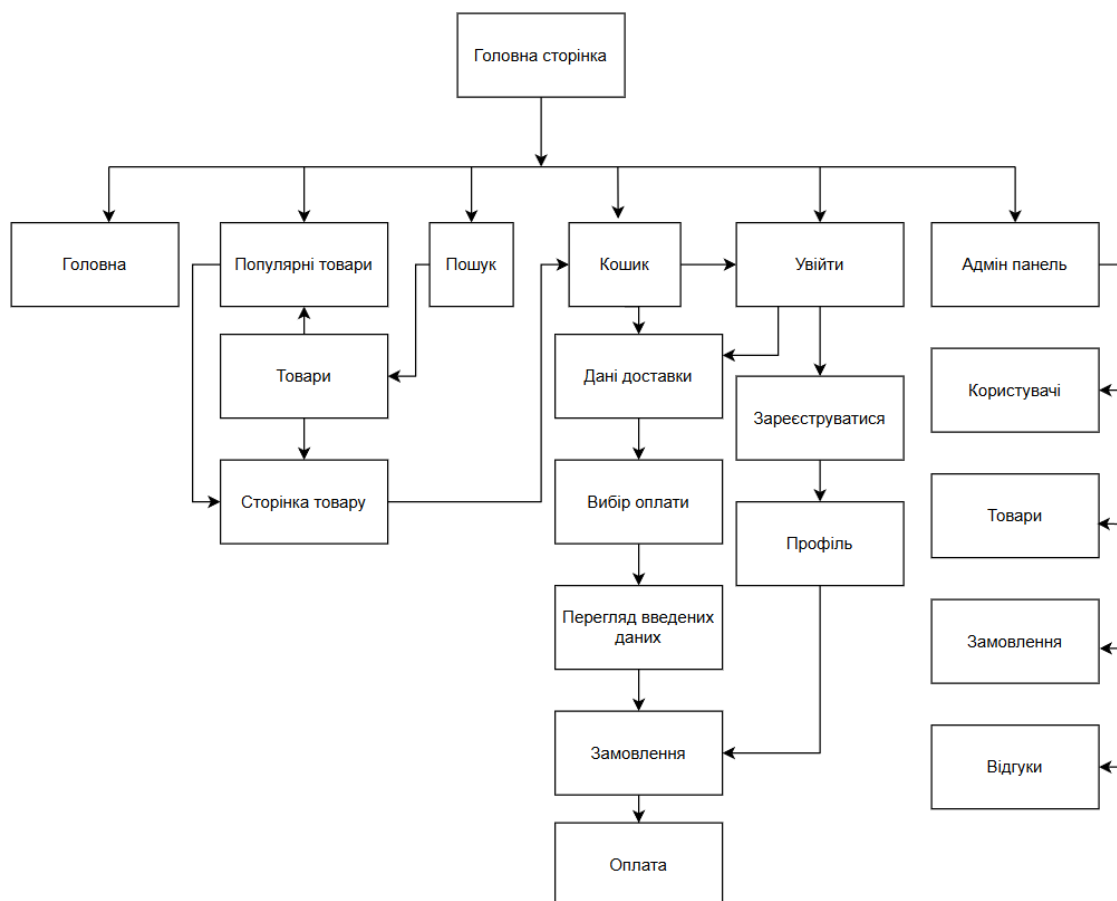
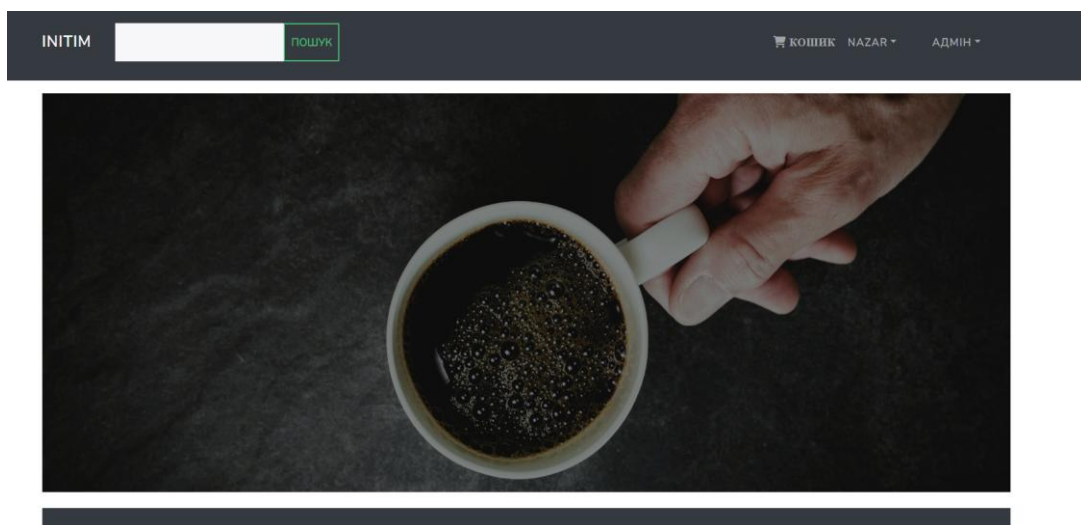


Рисунок 2.1 – Структурна схема сайту

На рисунку 2.2 наведено головну сторінку сайту.



На рисунку 2.2 – Головна сторінка сайту

На рисунку 2.3 наведено продовження головної сторінки сайту.

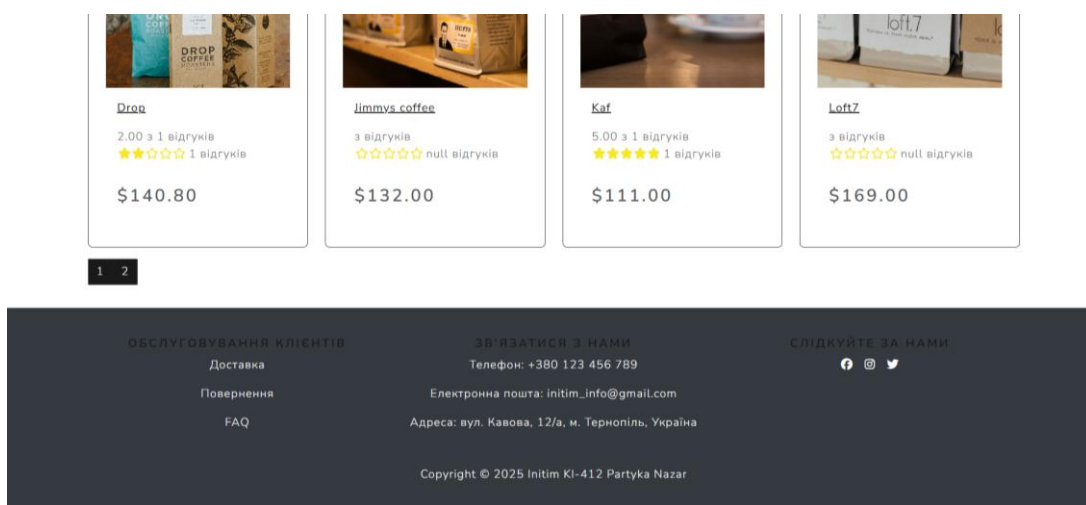


Рисунок 2.3 – Продовження головної сторінки сайту

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

На рисунку 2.4 наведено сторінку кошика.

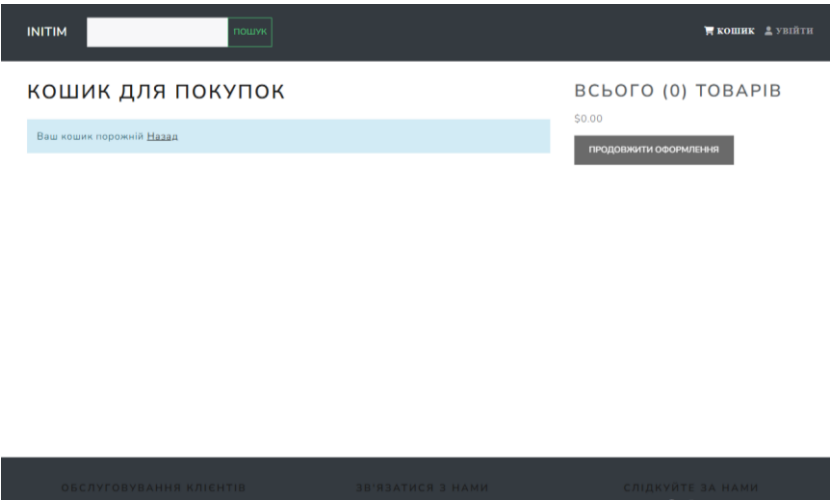


Рисунок 2.4 – Сторінка кошика

На рисунку 2.5 наведено сторінку продукту.

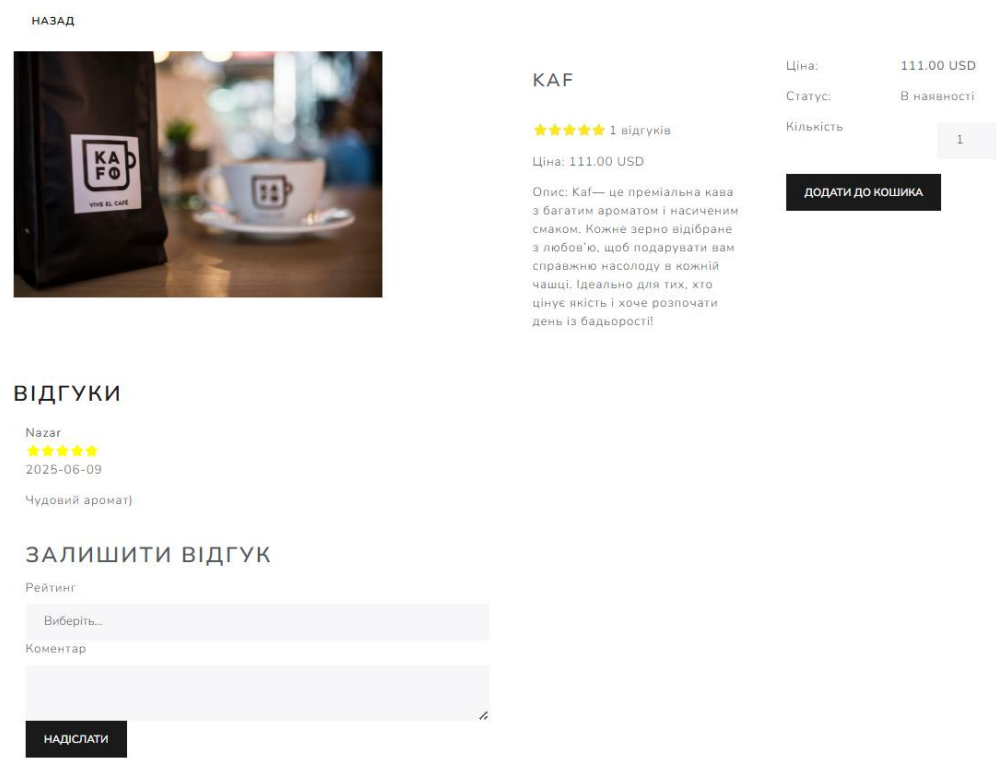


Рисунок 2.5 – Сторінка продукту

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

На рисунку 2.6 наведено профіль користувача.

INITIM

ПОШУК

КОШИК NAZAR АДМІН

ПРОФІЛЬ КОРИСТУВАЧА

Ім'я

Nazar

Електронна пошта

hohojkag@gmail.com

Пароль

Введіть пароль

Підтвердження пароля

Підтвердіть пароль

ОНОВИТИ

МОЇ ЗАМОВЛЕННЯ

ID	ДАТА	СУМА	ОПЛАЧЕНО	ДОСТАВКА	
1	2025-05-31	\$288.07	2025-05-31	2025-05-31	ДЕТАЛІ
2	2025-06-07	\$214.35	2025-06-08	2025-06-08	ДЕТАЛІ
3	2025-06-09	\$145.35	×	×	ДЕТАЛІ

ОБСЛУГОВУВАННЯ КЛІЄНТІВ

ЗВ'ЯЗАТИСЯ З НАМИ

СЛІДКУЙТЕ ЗА НАМИ

Рисунок 2.6 – Профіль користувача

На рисунку 2.7 наведено сторінку з товарами.

INITIM

ПОШУК

КОШИК NAZAR АДМІН

КАВА

Arts

з відгуків

☆☆☆☆☆ null відгуків

\$41.00

Ba cafe

5.00 з 1 відгуків

☆☆☆☆☆ 1 відгуків

\$100.00

Base blend

3.00 з 1 відгуків

☆☆☆☆☆ 1 відгуків

\$109.00

Beans

4.00 з 1 відгуків

☆☆☆☆☆ 1 відгуків

\$150.50

Drop

з відгуків

☆☆☆☆☆ null відгуків

\$140.80

jimmys coffee

з відгуків

☆☆☆☆☆ null відгуків

\$132.00

Kaf

5.00 з 1 відгуків

☆☆☆☆☆ 1 відгуків

\$111.00

Loft7

з відгуків

☆☆☆☆☆ null відгуків

\$169.00

1

2

Рисунок 2.7 – Сторінка з товарами

Таблиця 2.1 Ієрархічна структура зовнішніх функцій.

Розділ	Функціонал
Головна сторінка	Відображення популярних товарів, товарів, пошук товарів, навігація до товарів.
Сторінка товару	Відображення детальної інформації про товар (назва, опис, ціна, рейтинг, зображення), можливість додати товар до кошика.
Кошик	Відображення списку товарів, додавання до кошика, можливість змінювати кількість товарів або видаляти їх, розрахунок загальної вартості замовлення.
Оформлення замовлення	Введення даних для доставки (адреса, місто, поштовий індекс), вибір способу оплати, підтвердження замовлення.
Особистий кабінет	Перегляд історії замовлень, оновлення особистих даних.
Адміністративна панель	Управління товарами (додавання, редагування, видалення), перегляд замовлень, управління користувачами.

Функціональні специфікації

1. Головна сторінка:
 - Відображає список популярних товарів, отриманих через API.
 - Реалізовано пошук товарів за ключовими словами.
2. Сторінка товару:
 - Використовує API для отримання детальної інформації про товар.
 - Додає товар до кошика через Redux.
3. Кошик:
 - Зберігає дані про товари у локальному сховищі браузера

(localStorage).

- Використовує Redux для управління станом кошика.

4. Оформлення замовлення:

- Валідація введених даних для доставки.
- Надсилання даних замовлення на сервер через API.

5. Особистий кабінет:

- Отримує дані про замовлення користувача через API.
- Дозволяє оновлювати особисту інформацію.

6. Адміністративна панель:

- Використовує захищені API для управління товарами, замовленнями та користувачами.

Засоби та методи реалізації

1. Тип верстки:

- Використовується адаптивна верстка для забезпечення коректного відображення на різних пристроях (мобільні телефони, планшети, десктопи).

2. Фреймворки:

- React:
- Реалізація компонентів інтерфейсу користувача.
- Управління станом через Redux.[13]

Bootstrap:

- Використовується для створення адаптивного дизайну.[29]
- Django REST Framework:
- Реалізація API для взаємодії фронтенду з бекендом.[11]

3. Методи реалізації:

- React:
- Компоненти, такі як ProductCard, CartScreen, OrderScreen, реалізують функціонал відображення товарів, кошика та замовлень.
- Приклад коду для додавання товару до кошика:

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						34
Змн.	Арк.	№ докум.	Підпис	Дата		

На рисунку 2.8 наведено код для додавання товару до кошика.

```
const addToCartHandler = (id) => {  
  dispatch(addToCart(id, qty));  
};
```

Рисунок 2.8 – Код для додавання товару до кошика

- Redux:

— Управління станом кошика:

На рисунку 2.9 наведено код управління станом кошика.

```
export const cartReducer = (state = { cartItems: [], shippingAddress: {} }, action) => {  
  switch (action.type) {  
    case CART_ADD_ITEM:  
      const item = action.payload;  
      const existItem = state.cartItems.find(x => x.product === item.product);  
      if (existItem) {  
        return {  
          ...state,  
          cartItems: state.cartItems.map(x => x.product === existItem.product ? item : x),  
        };  
      } else {  
        return {  
          ...state,  
          cartItems: [...state.cartItems, item],  
        };  
      }  
    }  
  }  
};
```

На рисунку 2.9 – Код управління станом кошика

- Django REST Framework:

— Реалізація API для отримання даних про товари:

На рисунку 2.10 наведено код для отримання даних про товар.

```
@api_view(['GET'])  
def getProducts(request):  
    products = Product.objects.all()  
    serializer = ProductSerializer(products, many=True)  
    return Response(serializer.data)
```

Рисунок 2.10 – Код для отримання даних про товар

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

Приклади з програми

1. Головна сторінка:

— Відображення популярних товарів:

На рисунку 2.11 наведено код відображення популярних товарів.

```
useEffect(() => {  
    dispatch(listProducts());  
}, [dispatch]);
```

Рисунок 2.11 – Код відображення популярних товарів.

2. Сторінка товару:

— Отримання даних про товар:

На рисунку 2.12 наведено код отримання даних про товар.

```
const { id } = useParams();  
useEffect(() => {  
    dispatch(getProductDetails(id));  
}, [dispatch, id]);
```

Рисунок 2.12 – Код отримання даних про товар.

3. Оформлення замовлення:

— Надсилення даних замовлення:

На рисунку 2.13 наведено код надсилення даних замовлення.

```
const placeOrderHandler = () => {  
    dispatch(createOrder({  
        orderItems: cart.cartItems,  
        shippingAddress: cart.shippingAddress,  
        paymentMethod: cart.paymentMethod,  
        itemsPrice: cart.itemsPrice,  
        shippingPrice: cart.shippingPrice,  
        taxPrice: cart.taxPrice,  
        totalPrice: cart.totalPrice,  
    }));  
};
```

Рисунок 2.13 – Код надсилення даних замовлення

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						36
Змн.	Арк.	№ докум.	Підпис	Дата		

2.3.2 Проектування логіки програми

Логіка програми інтернет-магазину "Initim" базується на взаємодії між фронтендом, бекендом і базою даних. Основні функції програми включають управління товарами, кошиком, замовленнями та користувачами. Логіка реалізована за допомогою сучасних підходів, таких як REST API, Redux для управління станом, а також ORM Django для роботи з базою даних.

Структурна модель програмного комплексу

На верхньому рівні програма поділяється на такі групи функцій:

1. Функції користувача:
 - Перегляд каталогу товарів.
 - Додавання товарів до кошика.
 - Оформлення замовлення.
 - Перегляд історії замовлень.
2. Функції адміністратора:
 - Управління товарами (додавання, редагування, видалення).
 - Перегляд замовлень.
 - Управління користувачами.
3. Системні функції:
 - Аутентифікація та авторизація.
 - Обробка запитів до бази даних.
 - Розрахунок вартості замовлення.

Функціональні специфікації

1. Введення даних:
 - Користувач вводить дані через форми (наприклад, форма реєстрації, форма оформлення замовлення).
 - Адміністратор вводить дані через адміністративну панель (наприклад, додавання нового товару).

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						37
Змн.	Арк.	№ докум.	Підпис	Дата		

2. Виведення даних:
 - Відображення списку товарів, деталей замовлення, статусу доставки.
 - Виведення повідомлень про помилки (наприклад, "Неправильний логін або пароль").
3. Перетворення даних:
 - Зміна стану системи при додаванні товару до кошика.
 - Розрахунок загальної вартості замовлення (включаючи податки та доставку).
 - Оновлення статусу замовлення після оплати.

Алгоритм розв'язання задачі

Алгоритм роботи програми, який відображає основні етапи:

1. Користувач заходить на сайт.
2. Вибирає товар і додає його до кошика.
3. Переходить до оформлення замовлення.
4. Вводить дані для доставки та вибирає спосіб оплати.
5. Підтверджує замовлення.
6. Система обробляє замовлення:
 - Розраховує загальну вартість.
 - Зберігає дані в базі.
 - Відправляє підтвердження користувачу.

Для відображення структури програми використано класи, які включають основні класи:

1. Product:
 - Поля: name, price, category, description, countInStock.
 - Методи: __str__.
2. Order:
 - Поля: user, totalPrice, isPaid, isDelivered.

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

- Методи: __str__.
- 3. OrderItem:
 - Поля: product, order, qty, price.
 - Методи: __str__.
- 4. ShippingAddress:
 - Поля: address, city, postalCode, country.
 - Методи: __str__.

Схема для оформлення замовлення:

1. Ввід:
 - Дані про товари в кошику.
 - Дані для доставки.
 - Спосіб оплати.
2. Обробка:
 - Розрахунок загальної вартості.
 - Збереження замовлення в базі даних.
3. Вивід:
 - Підтвердження замовлення.
 - Відображення статусу замовлення.

Реакція на неочікувані дані

1. Неправильний логін або пароль:
 - Виведення повідомлення "Неправильний логін або пароль".
2. Порожній кошик:
 - Заборона оформлення замовлення з повідомленням "Кошик порожній".
3. Неправильний формат даних:
 - Валідація введених даних із виведенням відповідних повідомлень.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

2.4 Написання скрипта

У проєкті інтернет-магазину "Initim" реалізовано функціонал реєстрації нового користувача, який дозволяє створювати облікові записи для доступу до персоналізованих функцій сайту, таких як оформлення замовлень та перегляд історії покупок.

Скрипт реєстрації нового користувача, забезпечує введення даних користувача, їх валідацію та передачу на сервер для створення нового облікового запису.

Опис функціоналу

1. Введення даних:

- Користувач вводить ім'я, електронну пошту, пароль та підтвердження пароля через форму.

- Поля форми мають валідацію, яка забезпечує обов'язковість введення даних.

2. Валідація:

- Перевірка відповідності пароля та підтвердження пароля.
- У разі невідповідності паролів виводиться повідомлення про помилку.

3. Передача даних:

- Дані користувача передаються на сервер через функцію register, яка викликається через Redux.

4. Обробка відповіді:

- У разі успішної реєстрації користувач перенаправляється на головну сторінку або сторінку, зазначену в параметрі redirect.

- У разі помилки виводиться відповідне повідомлення.

Реакція на помилки

1. Невідповідність паролів:

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

- Виводиться повідомлення: "Паролі не співпадають".
2. Помилка сервера:
- Виводиться повідомлення про помилку, отримане від сервера.

Взаємодія з сервером

Дані реєстрації передаються на сервер через функцію register, яка викликається через Redux. Сервер обробляє запит, створює нового користувача та повертає відповідь.

На рисунку 2.14 наведено код скрипта реєстрації нового користувача.

```
<Form onSubmit={handleSubmit}>
  <Form.Group controlId='name'>
    <Form.Label>Ім'я</Form.Label>
    <Form.Control
      required
      type='name'
      placeholder='Введіть ім'я'
      value={name}
      onChange={e => setName(e.target.value)} />
  </Form.Group>
  <Form.Group controlId='email'>
    <Form.Label>Електронна пошта</Form.Label>
    <Form.Control
      required
      type='email'
      placeholder='Введіть електронну пошту'
      value={email}
      onChange={e => setEmail(e.target.value)} />
  </Form.Group>
  <Form.Group controlId='password'>
    <Form.Label>Пароль</Form.Label>
    <Form.Control
      required
      type='password'
      placeholder='Введіть пароль'
      value={password}
      onChange={e => setPassword(e.target.value)} />
  </Form.Group>
  <Form.Group controlId='passwordConfirm'>
    <Form.Label>Підтвердження пароля</Form.Label>
    <Form.Control
      required
      type='password'
      placeholder='Підтвердіть пароль'
      value={confirmPassword}
      onChange={e => setConfirmPassword(e.target.value)} />
  </Form.Group>
  <Button type='submit' variant='primary'>
    Зареєструватися
  </Button>
</Form>
```

Рисунок 2.14 – Код скрипта реєстрації нового користувача

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

На рисунку 2.15 наведено блок-схему коду скрипта реєстрації нового користувача.

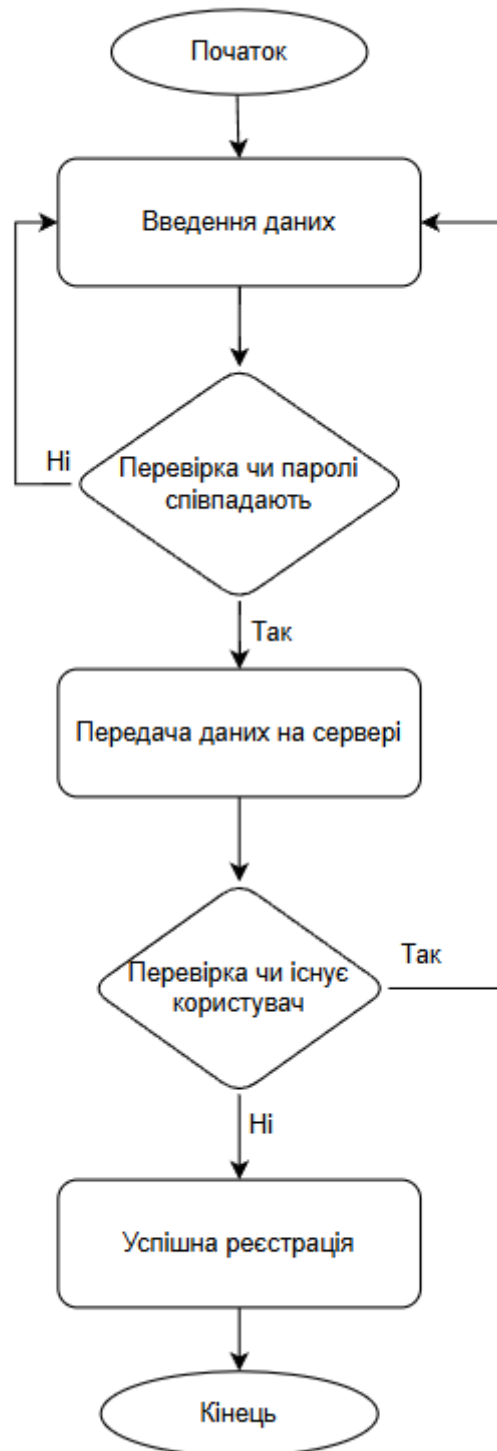


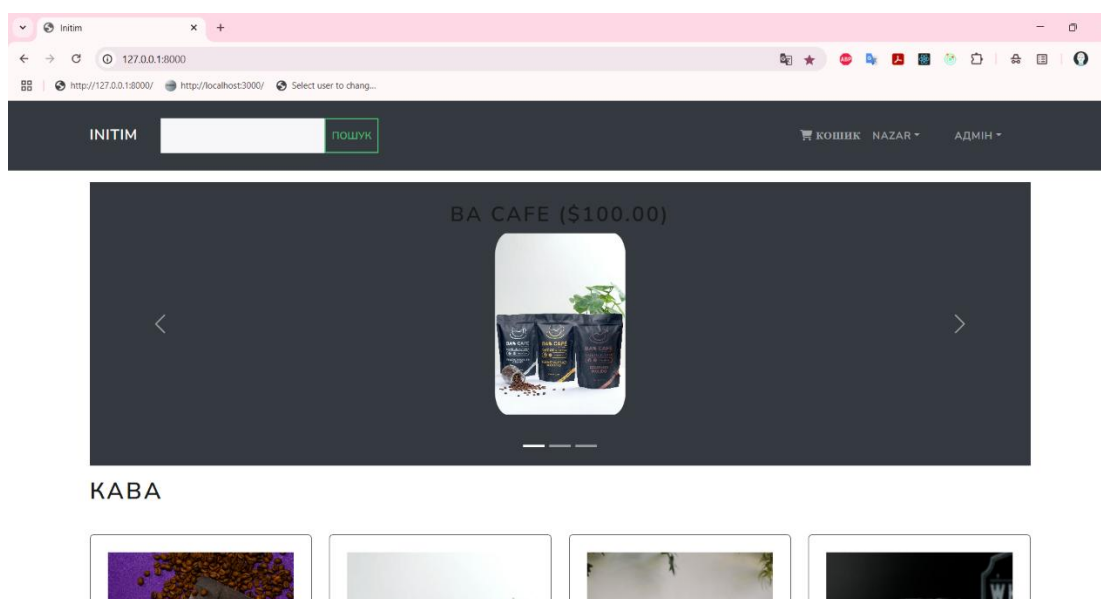
Рисунок 2.15 - Блок-схема коду скрипта реєстрації нового користувача

2.5 Тестування сайту

Тестування є важливим етапом розробки програмного забезпечення, який дозволяє перевірити коректність роботи сайту, його функціональність, продуктивність та сумісність із різними браузерами. У цьому розділі буде описано процес тестування інтернет-магазину "Initim", який включає ручне тестування.

Ручне тестування проводиться у найпоширеніших браузерах, таких як Opera, Google Chrome, Microsoft Edge та Mozilla Firefox. Це дозволяє перевірити коректність відображення інтерфейсу, адаптивність дизайну, а також функціональність основних компонентів сайту.

На рисунку 2.16 наведено скріншот тестування відображення сайту у Google Chrome.



Рисунку 2.16 - Скріншот тестування відображення сайту у Google Chrome.

На рисунку 2.17 наведено скріншот тестування відображення сайту у Опера.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

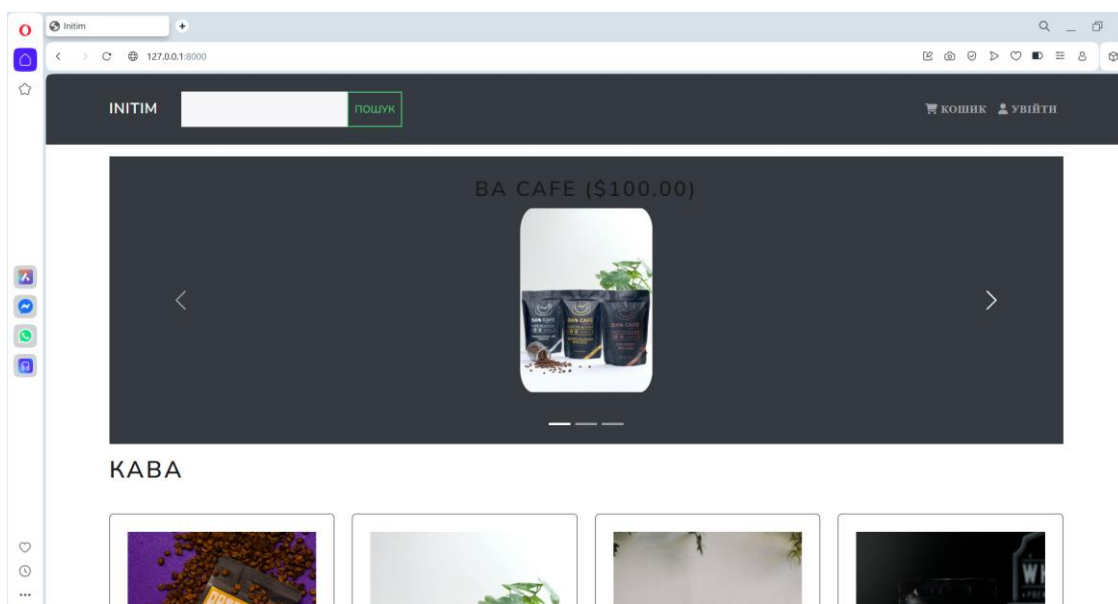


Рисунок 2.17 – Скріншот тестування відображення сайту у Орега

На рисунку 2.18 наведено скріншот тестування відображення сайту у Mozilla Firefox.

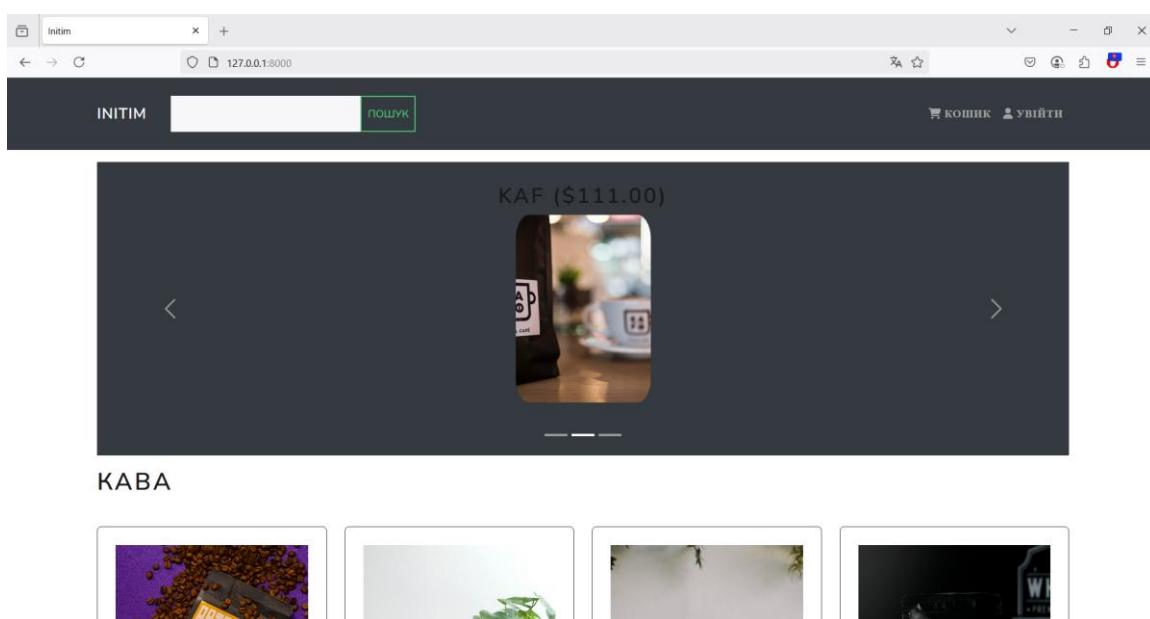


Рисунок 2.18 – Скріншот тестування відображення сайту у Mozilla Firefox

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		44

На рисунку 2.19 наведено скріншот тестування відображення сайту у Microsoft Edge.

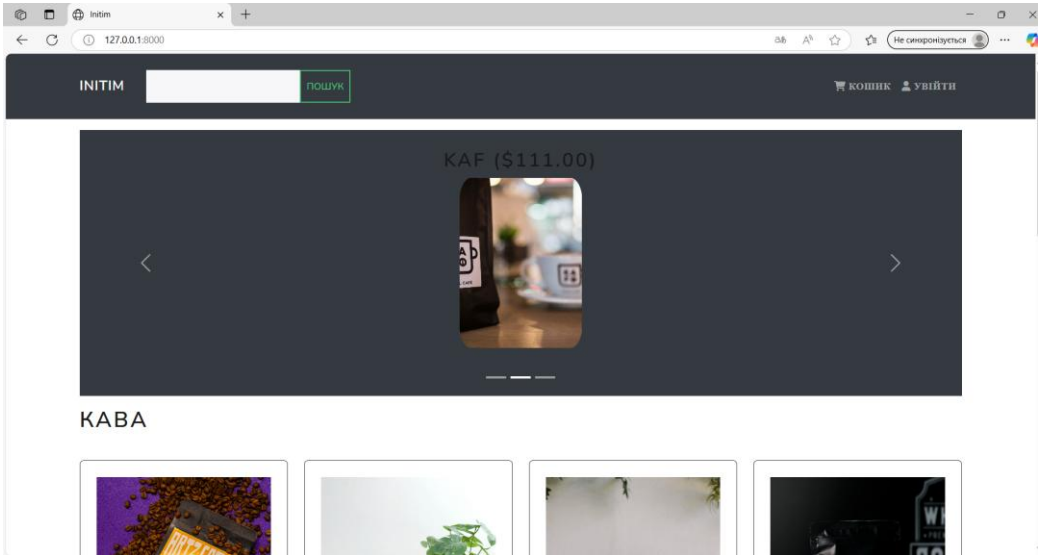


Рисунок 2.19 – Скріншот тестування відображення сайту у Microsoft Edge

На рисунку 2.20 наведено скріншот тестування сайту на ноутбучі розміром екрану 1920 на 1080.

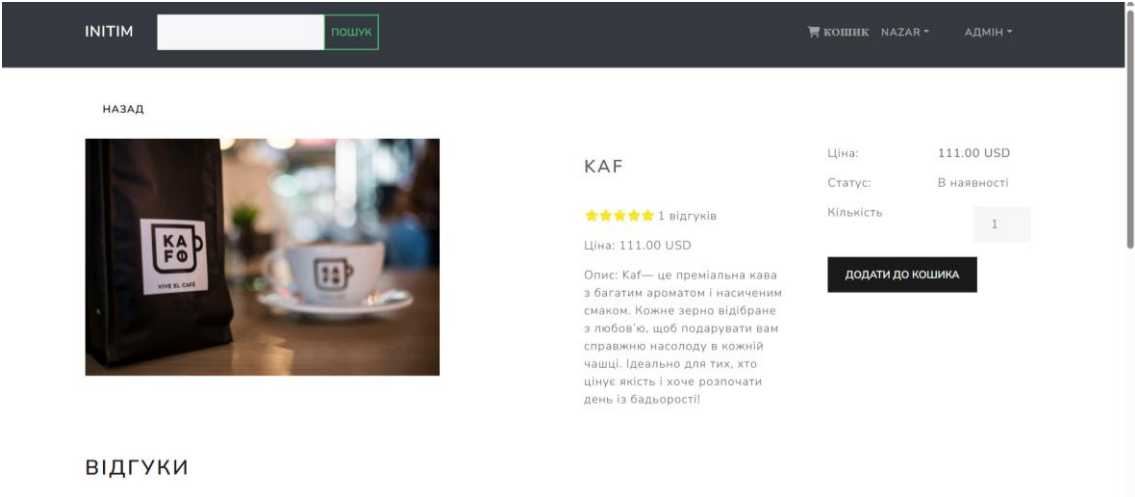


Рисунок 2.20 – Скріншот тестування сайту на ноутбучі розміром екрану 1920 на 1080

На рисунку 2.21 наведено скріншот тестування сайту на мобільному пристрої Samsug Galaxy A51/71 з розміром екрану 412 на 914.

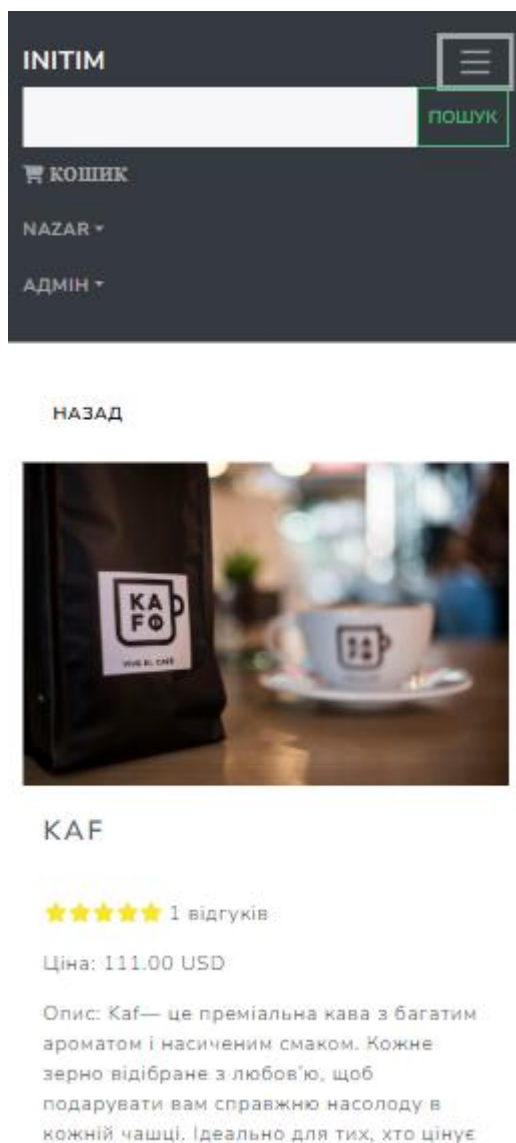
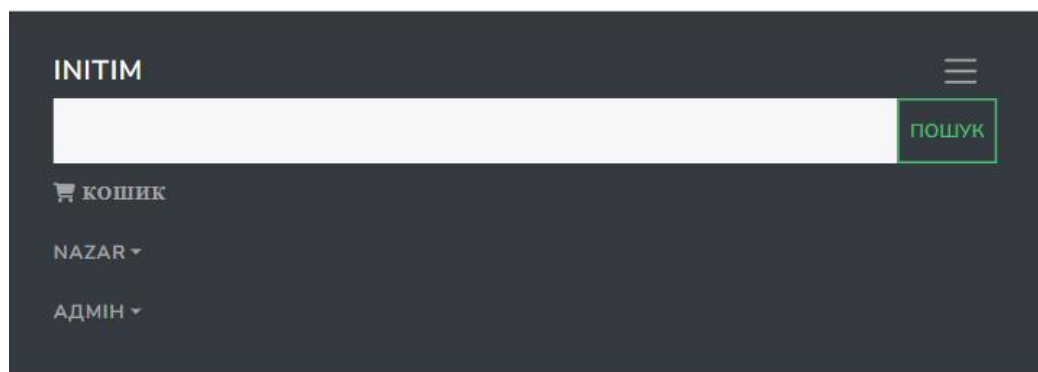


Рисунок 2.21 – Скріншот тестування сайту на мобільному пристрої Samsug Galaxy A51/71 з розміром екрану 412 на 914

На рисунку 2.22 наведено скріншот тестування сайту на планшеті розміром екрану 768 на 811.

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		



НАЗАД



KAF

Ціна:
111.00 USD

★★★★★ 1
відгуків

Статус:
В наявності

Ціна: 111.00
USD

Кількість
1

Опис: Kaf— це преміальна кава з багатим ароматом і насиченим смаком. Кожне зерно відібране з любов'ю, щоб

ДОДАТИ
ДО
КОШИКА

Рисунок 2.22 – Скріншот тестування сайту на планшеті розміром екрану 768 на 811

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

3 СПЕЦІАЛЬНИЙ РОЗДІЛ

3.1 Інструкція з інсталяції та налаштування веб серверів та супутнього програмного забезпечення

Для розгортання інтернет-магазину "Initim" необхідно встановити та налаштувати сервер, а також супутнє програмне забезпечення. У цьому проєкті використовується стек технологій, який включає Django для бекенду, React для фронтенду, MariaDB для бази даних, а також віртуальне середовище Python.[30]

Спочатку потрібно встановити програми які потрібні для роботи сайту локально: Node.js (додатково поставити галочку на встановлення усіх залежностей щоб встановити мову програмування Python), MariaDB.[30]

Щоб налаштувати MariaDB потрібно спочатку створити базу даних initim локально через командну стрічку або скачати графічний інтерфейс HeidiSQL, потім підключитися до неї. У файлі з налаштуваннями на бекенді потрібно прописати наступне:[12]

На рисунку 3.1 наведено підключення до бази даних.

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'initim',
        'USER': 'root',
        'PASSWORD': 'password',
        'HOST': '127.0.0.1',
        'PORT': '3306',
    }
}
```

Рисунок 3.1 – Підключення до бази даних

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		48

1. Створення віртуального середовища:

- встановлення середовища `pip install virtualenv`;
- створення середовища `virtualenv myenv`;
- активація середовища `myenv\Scripts\activate`. [15]

2. Створення бекенд проекту:

- створення проекту Django `django-admin startproject backend`;
- перехід у папку `cd backend`;
- створення додатку `python manage.py startapp base`. [15]

Створення залежностей для бекенду:

- `pip install django`;
- `pip install pillow`;
- `pip install django-cors-headers==3.6.0`;
- `pip install django-rest-framework-simplejwt`;
- `pip install mysqlclient`; [15]

Міграція бази даних:

- `python manage.py makemigrations`;
- `python manage.py migrate`; [12]

Створення користувача для адмін-панелі:

- `python manage.py createsuperuser`; [15]

Запуск серверу у папці backend:

- `python manage.py runserver` на хості <http://127.0.0.1:8000/>. [15]

3. Створення фронтенд проекту:

- `npm create-react-app frontend`; [13]

Перехід до папки фронтенд і встановлення залежності:

- `npm install react-bootstrap`;
- `npm install react-router-dom react-router-bootstrap`;
- `npm install axios`;
- `npm install redux@4.2.1 react-redux@8.0.5 redux-thunk@2.4.2 redux-`

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

devtools-extension@2.13.9;

- npm і @paypal/react-paypal-js.[13]

Запуск сервера npm start на хості <http://localhost:3000/>. [13]

Щоб зібрати проект потрібно виконати команду:

- npm run build.[13]

4. Взаємодія між фронтендом і бекендом відбувається через порт:

<http://127.0.0.1:8000/>. [15]

3.2 Інструкція по встановленню та налаштуванню розробленого проекту

Для встановлення проекту інтернет-магазину "Initim" з носія на який було записано, необхідно виконати наступні кроки:

1. Завантажити Node.js (поставити галочку на встановлення усіх залежностей щоб завантажити мову програмування Python).[30]

2. Завантажити MariaDB і створити базу даних під назвою initim.[12]

3. Підключити носій до ноутбука

4. Скопіювати папку з проектом на локальний диск

5. У папці проекту запустити віртуальне середовище:

- myenv\scripts\activate.[11]

6. Зробити міграцію бази даних у папці бекенд:

- python manage.py makemigrations;

- python manage.py migrate.[11]

7. Для доступу до адмін-панелі створити суперкористувача:

- python manage.py createsuperuser.[11]

8. Зайти у папку бекенд і запустити сервер:

- python manage.py runserver.[11]

9. Зайти у папку фронтенд і запустити сервер:

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

- npm start.[13]

3.3 Інструкція з використання тестових наборів

Тестування проекту інтернет-магазину "Initim" спрямоване на перевірку функціональності, продуктивності та сумісності системи. Використання тестових наборів дозволяє автоматизувати процес перевірки, зменшити кількість помилок та забезпечити стабільну роботу проекту.

Послідовність виконання ручного тестування:

1. У папці проекту активувати віртуальне середовище:

- myenv\scripts\activate.[11]

2. У папці бекенд виконати команди, щоб перевірити що база даних налаштована:

- python manage.py makemigrations;

- python manage.py migrate.[11]

3. У папці бекенд перевірити чи сервер запускається:

- python manage.py runserver.[15]

4. У папці фронтенд перевірити чи сервер запускається:

- npm start.[13]

Тестові сценарії

1. Тестування функціональності:

- Реєстрація нового користувача.

- Авторизація користувача.

- Додавання товарів до кошика.

- Оформлення замовлення.

- Перегляд історії замовлень.

На рисунку 3.2 наведено скріншот сторінки реєстрації нового користувача.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						51
Змн.	Арк.	№ докум.	Підпис	Дата		

INITIM

ПОШУК

КОШИК

УВІЙТИ

ЗАРЕЄСТРУВАТИСЯ

Ім'я

Введіть ім'я

Електронна пошта

Введіть електронну пошту

Пароль

Введіть пароль

Підтвердження пароля

Підтвердіть пароль

ЗАРЕЄСТРУВАТИСЯ

Маєте акаунт? [Увійти](#)

На рисунку 3.2 – Скріншот реєстрації нового користувача

На рисунку 3.3 наведено скріншот сторінки аторизації користувача.

INITIM

ПОШУК

КОШИК

УВІЙТИ

УВІЙТИ

Електронна пошта

Введіть електронну пошту

Пароль

Введіть пароль

УВІЙТИ

Новий користувач? [Зареєструватися](#)

Рисунок 3.3 – Скріншот авторизації користувача

На рисунку 3.4 наведено скріншот кошика з доданим товаром.

INITIM

ПОШУК

КОШИК

УВІЙТИ

КОШИК ДЛЯ ПОКУПОК



Drop

\$140.80

1

ВСЬОГО (1) ТОВАРІВ

\$140.80

ПРОДОВЖИТИ ОФОРМЛЕННЯ

Рисунок 3.4 – Скріншот кошика з доданим товаром

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

На рисунку 3.5 наведено скріншот оформлення замовлення.

INITIM

🛒 КОШИК NAZAR АДМІН

Логін Доставка Оплата Місце замовлення

ДОСТАВКА

Адреса

Пряма 10

Місто

Київ

Поштовий адрес

28400

Країна

Україна

Рисунок 3.5 – Скріншот оформлення замовлення

На рисунку 3.6 наведено скріншот оформлених замовлень.

INITIM

🛒 КОШИК NAZAR АДМІН

ПРОФІЛЬ КОРИСТУВАЧА

Ім'я

Nazar

Електронна пошта

hohojkag@gmail.com

Пароль

Введіть пароль

Підтвердження пароля

Підтвердіть пароль

МОЇ ЗАМОВЛЕННЯ

ID	ДАТА	СУМА	ОПЛАЧЕНО	ДОСТАВКА	
1	2025-05-31	\$288.07	2025-05-31	2025-05-31	ДЕТАЛІ
2	2025-06-07	\$214.35	2025-06-08	2025-06-08	ДЕТАЛІ
3	2025-06-09	\$145.35	×	×	ДЕТАЛІ

Рисунок 3.6 – Скріншот оформлених замовлень

2. Тестування API:
 - Перевірка отримання списку товарів.
 - Перевірка створення замовлення.
 - Перевірка оновлення статусу замовлення.
3. Тестування сумісності:

- Перевірка роботи проекту у різних браузерах.
- Перевірка адаптивності дизайну на різних пристроях.

Аналіз результатів тестування:

- Занотуйте та перевірте всі виявлені помилки та недоліки.
- Перевірте, чи всі функції працюють відповідно до технічного завдання.

3.4 Інструкція з експлуатації програмного комплексу

Програмний комплекс інтернет-магазину "Initim" складається з бекенд частини, реалізованої на Django, фронтенд-частини, побудованої на React, та бази даних MariaDB. Цей комплекс забезпечує функціональність для управління товарами, замовленнями, користувачами, а також інтеграцію з платіжними системами.[15]

Запуск програмного комплексу

1. Запуск бекенду:
 - Перехід до папки backend:
 - `cd backend`. [15]
 - Активація віртуального середовища:
 - `myenv\Scripts\activate`. [15]
 - Запуск сервера Django:
 - `python manage.py runserver`. [15]
 - Сервер буде доступний за адресою:
 - <http://127.0.0.1:8000/>. [15]
2. Запуск фронтенду:
 - `cd frontend`. [13]
 - Запуск сервера React:
 - `npm start`. [13]

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		54

- Сервер досупний за адресою:
- <http://localhost:3000/>. [13]

Основні функції програмного комплексу

1. Функції для користувачів:
 - Реєстрація та авторизація.
 - Перегляд каталогу товарів.
 - Додавання товарів до кошика.
 - Оформлення замовлення.
 - Перегляд історії замовлень.
2. Функції для адміністратора:
 - Управління товарами (додавання, редагування, видалення).
 - Перегляд замовлень.
 - Управління користувачами.

Інструкція з використання

1. Реєстрація нового користувача:
 - Перейдіть на сторінку реєстрації.
 - Введіть ім'я, електронну пошту, пароль та підтвердження пароля.
 - Натисніть кнопку "Зареєструватися".
2. Авторизація:
 - Перейдіть на сторінку входу.
 - Введіть електронну пошту та пароль.
 - Натисніть кнопку "Увійти".
3. Додавання товарів до кошика:
 - Перейдіть на сторінку товару.
 - Виберіть кількість товару.
 - Натисніть кнопку "Додати до кошика".
4. Оформлення замовлення:
 - Перейдіть до кошика.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

- Натисніть кнопку "Оформити замовлення".
 - Заповніть форму доставки та оплати.
 - Підтвердіть замовлення.
5. Управління товарами (для адміністратора):
- Увійдіть до адміністративної панелі.
 - Додайте новий товар, заповнивши форму.
 - Редагуйте або видаляйте існуючі товари.

Завершення роботи

1. Зупинка серверів:

- Для бекенду та фронтенду:
- Ctrl + C.[15]

Деактивація віртуального середовища:

- Deactivate.[15]

Технічна підтримка

У разі виникнення проблем з експлуатацією програмного комплексу звертайтеся до розробника або технічної підтримки. Усі помилки та недоліки повинні бути задокументовані для подальшого аналізу та усунення.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		

4 ЕКОНОМІЧНИЙ РОЗДІЛ

Метою економічної частини кваліфікаційної роботи є здійснення економічних розрахунків, спрямованих на визначення економічної ефективності розробки інтернет-магазину “Initim”, повної ціни і прийняття рішення про її подальший розвиток і впровадження.

4.1 Визначення стадій технологічного процесу та загальної тривалості проведення НДР

Загальну тривалість проведення кваліфікаційної роботи, дані витрат часу по окремих операціях технологічного процесу наведено у таблиці 4.1.

Таблиця 4.1 - Середній час виконання НДР та стадії технологічного процесу

№ п/п	Назва операції (стадії)	Виконавець	Середній час виконання операції, год.
1	видача завдання та постановка задачі	керівник роботи	4 години
2	аналіз задачі, складання технічного завдання	розробник	10 годин
3	планування	розробник	15 годин
4	проектування	розробник	11 годин
5	розробка (програмування)	розробник	180 годин
6	розгортання	розробник	16 годин
7	тестування	розробник	7 годин
8	затвердження здачі проєкту	керівник роботи	3 години
Разом		—	246 годин

Розробка роботи та виконання завдання для інтернет-магазину “Initim” складає 246 годин.

4.2 Визначення витрат на оплату праці та відрахувань на соціальні заходи

Відповідно до Закону України «Про оплату праці» заробітна плата - це «винагорода, обчислена, як правило, у грошовому виразі, яку власник або уповноважений ним орган виплачує працівникові за виконану ним роботу».

Розмір заробітної плати залежить від складності та умов виконуваної роботи, професійно-ділових якостей працівника, результатів його праці та господарської діяльності підприємства. Заробітна плата складається з основної та додаткової оплати праці.

Основна заробітна плата нараховується на виконану роботу за тарифними ставками, відрядними розцінками чи посадовими окладами і не залежить від результатів господарської діяльності підприємства.

Додаткова заробітна плата — це складова заробітної плати працівників, до якої включають витрати на оплату праці, не пов’язані з виплатами за фактично відпрацьований час. Нараховують додаткову заробітну плату залежно від досягнутих і запланованих показників, умов виробництва, кваліфікації виконавців. Джерелом додаткової оплати праці є фонд матеріального стимулювання, який створюється за рахунок прибутку.

Місячний оклад кожного працівника слід враховувати згідно існуючих на даний час тарифних окладів.

Основна заробітна плата розраховується за формулою:

$$З_{\text{осн.}} = T_c \cdot K_r$$

де T_c - тарифна ставка, грн;

K_r - кількість відпрацьованих годин.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		58

Звідси, для керівника проєкту $Z_{\text{осн.}} = 400 \cdot 7 = 2800$ гривень, для розробника $Z_{\text{осн.}} = 60 \cdot 350 = 21000$ гривень.

Додаткова заробітна плата становить 10–15 % від суми основної заробітної плати.

$$Z_{\text{дод.}} = Z_{\text{осн.}} \cdot K_{\text{допл.}}$$

де, $K_{\text{допл.}}$ - коефіцієнт додаткових виплат працівникам 0,1 - 0,15.

Для керівника проєкту $Z_{\text{дод.}} = 2800 \cdot 0,11 = 308$ гривні, для розробника $Z_{\text{дод.}} = 21000 \cdot 0,8 = 16800$ гривень.

Звідси загальні витрати на оплату праці ($V_{\text{о.п.}}$) визначаються за формулою:

$$V_{\text{о.п.}} = Z_{\text{осн.}} + Z_{\text{дод.}}$$

Для керівника - $2800 + 308 = 3108$ гривні, для розробника - $21000 + 16800 = 37800$ гривень.

Крім того, слід визначити суму нарахування на заробітну плату: єдиний соціальний внесок - 22%.

Отже, сума нарахувань на заробітну плату буде становити:

$$V_{\text{с.з.}} = \text{ФОП} \cdot 0,22 \text{ де, ФОП - фонд оплати праці, грн.}$$

$$V_{\text{с.з.}} = 40908 \cdot 0,22 = 8999,76 \text{ гривень.}$$

Проведені розрахунки витрат на оплату праці зведено у таблиці 4.2.

Таблиця 4.2 - Зведені розрахунки витрат на оплату праці

№ п/п	Категорія працівників	Основна заробітна плата, грн.			Додатк зароб. плата, грн.	Нарах. на ФОП, грн.	Всього витрати на оплату праці, грн.
		Тариф. ставка, грн.	К-сть відпрацьов. год.	Фактично нарах. з/пл., грн.			
1	2	3	4	5	6	7	8
1	керівник	400	7	2800	308	-	-

Продовження таблиці 4.2

1	2	3	4	5	6	7	8
2	розробник	60	350	21000	16800	-	-
Разом		450	357	23800	17108	8999,76	49907,76

4.3 Розрахунок витрат на електроенергію

Затрати на електроенергію 1-ці обладнання визначаються за формулою:

$$Z_e = W \cdot T \cdot S$$

де, W - необхідна потужність, кВт;

T - кількість годин роботи обладнання;

S - вартість кіловат-години електроенергії.

Для наявного ноутбука у розробника ASUS TUF GAMING A15
 $W = 0,2$ кВт, $T = 203$ год.

Вартість кіловат-години електроенергії слід приймати згідно існуючих на даний час тарифів (7 грн).

Звідси:

$$Z_e = 0,2 \cdot 203 \cdot 7 = 284,2 \text{ грн}$$

4.4 Розрахунок суми амортизаційних відрахувань

Характерною особливістю застосування основних фондів у процесі виробництва є їх відновлення. Для відновлення засобів праці у натуральному виразі необхідне їх відшкодування у вартісній формі, яке здійснюється шляхом амортизації.

Амортизація – це процес перенесення вартості основних фондів на вартість новоствореної продукції з метою їх повного відновлення.

Комп'ютери та оргтехніка належать до четвертої групи основних

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

фондів. Для ноутбука ASUS TUF GAMING A15 коштує 33000 грн. Амортизація на нього нараховується тому що, вартість перевищує 2500 грн., а мінімально допустимі строки їх корисного використання 2 роки, звідси норма амортизації становить 20%.

Для визначення амортизаційних відрахувань застосовуємо формулу:

$$A = \frac{B_v \cdot H_a}{100\%}$$

де А - амортизаційні відрахування за звітний період, грн.;

B_v - балансова вартість групи основних фондів на початок звітного періоду, грн.;

H_a - норма амортизації, %.

$$A = \frac{33000 \cdot 0,04}{150} \cdot 203 = 1786,4$$

4.5 Обчислення накладних витрат

Накладні витрати пов'язані з обслуговуванням виробництва, утриманням апарату управління підприємства (фірми) та створення необхідних умов праці.

В залежності від організаційно-правової форми діяльності господарюючого суб'єкта, накладні витрати можуть становити 20-60 % від суми основної та додаткової заробітної плати працівників.

$$H_v = B_{o.п} \cdot 0,2 \dots 0,6$$

де, H_v - накладні витрати.

Звідси, $H_v = 40908 \cdot 0,5 = 20454$ гривень.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						61
Змн.	Арк.	№ докум.	Підпис	Дата		

4.6 Складання кошторису витрат та визначення собівартості НДР

Результати проведених вище розрахунків зведено у таблиці 4.4.

Таблиця 4.3 - Кошторис витрат на НДР

Зміст витрат	Сума, грн.	В % до загальної суми
Витрати на оплату праці (основну і додаткову заробітну плату)	40908	55,34
Відрахування на соціальні заходи	8999,76	12,17
Витрати на електроенергію	284,2	0,67
Амортизаційні відрахування	1786,4	4,15
Накладні витрати	20454	27,67
Собівартість	72432,36	100

Собівартість(C_v) НДР розрахуємо за формулою:

$$C_v = B_{o.п.} + B_{c.з.} + Z_{м.в.} + Z_e + A + H_v$$

Звідси, $C_v = 72432,36$ грн.

4.7 Розрахунок ціни НДР

Ціну НДР можна визначити за формулою:

$$Ц = (C_v \cdot (1 + P_{рен.}) + B_{н.і.}) \cdot (1 + ПДВ)$$

де, $P_{рен.}$ - рівень рентабельності;

$B_{н.і.}$ - вартість носія інформації, грн.;

ПДВ - ставка податку на додану вартість, (20 %).

Звідси, ціна НДР становитиме:

$$Ц = 72432,36 \cdot (1 + 0,3) \cdot (1 + 0,2) = 112994,48 \text{ гривні.}$$

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						62
Змн.	Арк.	№ докум.	Підпис	Дата		

4.8 Визначення економічної ефективності і терміну окупності капітальних вкладень

Ефективність виробництва - це узагальнене і повне відображення кінцевих результатів використання робочої сили, засобів та предметів праці на підприємстві за певний проміжок часу.

Для визначення ефективності продукту розраховують чисту теперішню вартість (ЧТВ) і термін окупності ($T_{ок}$).

$$ЧТВ = -K_B + \sum_{i=1}^t \frac{\Gamma_{\pi}}{(1+i)^t}$$

де K_B - затрати на проєкт;

Γ_{π} - грошовий потік за t-ий рік;

t - відповідний рік проєкту;

i - величина дисконтної ставки (10...15%).

Якщо $ЧТВ \geq 0$, то проєкт може бути рекомендований до впровадження.

$$ЧТВ = -72432,36 + \frac{40562,12}{(1+0,1)} + \frac{40562,12}{(1+0,1)^2} + \frac{40562,12}{(1+0,1)^3} = 28439,63$$

Термін окупності визначається за формулою:

$$T_{ок} = T_{пв} + \frac{H_B}{\Gamma_{пр}}$$

де $T_{пв}$ - період до повного відшкодування витрат, років;

H_B - невідшкодовані витрати на початок року, грн;

$\Gamma_{пр}$ - грошовий потік на початок року, грн.

$$T_{ок} = 2 + \frac{2035,29}{40562,12} = 2,1$$

Всі дані розрахунків внесено у зведену таблицю 4.5 техніко-економічних показників.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						63
Змн.	Арк.	№ докум.	Підпис	Дата		

Таблиця 4.4 - Техніко-економічні показники НДР

№ п/п	Показник	Значення
1.	Собівартість, грн.	72432,36
2.	Плановий прибуток, грн.	40562,12
3.	Ціна, грн.	112994,48
4.	Чиста теперішня вартість, грн.	28439,63
5.	Термін окупності, рік	2,1

Сайт є доцільним для розробки оскільки його чиста теперішня вартість дорівнює

28439,63, термін окупності 2,1, собівартість 72432,36а ціна становить 112994,48 .

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						64
Змн.	Арк.	№ докум.	Підпис	Дата		

5 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ

Охорона праці є невід'ємною складовою будь-якої діяльності, спрямованої на забезпечення безпеки працівників, збереження їх здоров'я та створення комфортних умов праці. У процесі розробки інтернет-магазину "Initim" особливу увагу було приділено питанням безпеки, які стосуються пожежонебезпечних властивостей матеріалів, оптимізації виробничого освітлення та моделювання небезпечних ситуацій.[1]

Розділи з охорони праці мають на меті висвітлити ключові аспекти, пов'язані з безпекою праці, а також запропонувати ефективні заходи для мінімізації ризиків. У них розглядаються теоретичні основи, практичні рекомендації та сучасні методи, які можуть бути застосовані для забезпечення безпеки на виробництві та в офісних умовах.[1]

Особливу увагу приділено моделюванню та прогнозуванню небезпечних ситуацій, що дозволяє не лише оцінити потенційні ризики, але й розробити заходи для їх запобігання. Також розглядаються питання оптимізації освітлення, яке впливає на продуктивність праці та зниження ризику травматизму.[4]

Ці розділи є важливими для забезпечення безпеки праці в умовах сучасного виробництва та розробки програмного забезпечення, що відповідає високим стандартам якості та безпеки.

У цьому розділі детально розглядаються такі аспекти безпеки та охорони праці:

1. Пожежонебезпечні властивості речовин і матеріалів
2. Заходи щодо оптимізації виробничого освітлення
3. Моделювання та прогнозування небезпечних ситуацій

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						65
Змн.	Арк.	№ докум.	Підпис	Дата		

5.1 Пожежонебезпечні властивості речовин і матеріалів

Пожежонебезпечні властивості речовин і матеріалів є важливим аспектом забезпечення безпеки на виробництві та в побуті. Вони визначаються здатністю речовин до займання, горіння та виділення тепла. Основними характеристиками, які впливають на пожежонебезпеку, є температура займання, швидкість горіння, теплотворна здатність, а також склад продуктів горіння.[5]

Основні характеристики пожежонебезпечних речовин:

1. Температура займання:

- Мінімальна температура, при якій речовина починає горіти в присутності джерела займання.
- Наприклад, температура займання бензину становить близько 300°C.[2]

2. Швидкість горіння:

- Визначає, як швидко речовина згорає, виділяючи тепло.
- Тверді матеріали, такі як деревина, мають нижчу швидкість горіння порівняно з рідинами.[2]

3. Теплотворна здатність:

- Кількість тепла, що виділяється під час горіння речовини.
- Вимірюється в джоулях або калоріях.[2]

4. Склад продуктів горіння:

- Важливо враховувати токсичність продуктів горіння, таких як чадний газ (CO) або діоксид вуглецю (CO₂).[2]

Класифікація пожежонебезпечних речовин:

- Горючі гази: метан, пропан, бутан.
- Рідини: бензин, спирти, масла.
- Тверді матеріали: деревина, папір, пластмаси.[2]

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						66
Змн.	Арк.	№ докум.	Підпис	Дата		

Методи оцінки пожежонебезпечності:

1. Лабораторний аналіз:

- Визначення температури займання та швидкості горіння.

2. Моделювання процесів горіння:

- Використання програмного забезпечення для прогнозування поведінки речовин під час пожежі.

3. Експериментальні дослідження:

- Проведення тестів у контрольованих умовах.[2]

Практичне значення:

- Врахування пожежонебезпечних властивостей речовин при їх зберіганні та транспортуванні.

- Розробка заходів щодо запобігання пожежам.[3]

5.2 Заходи щодо оптимізації виробничого освітлення

Оптимізація виробничого освітлення є важливим фактором для забезпечення безпеки праці, зниження ризику травматизму та підвищення продуктивності. Неправильне освітлення може призводити до втоми очей, зниження концентрації уваги та збільшення кількості помилок.[3]

Основні принципи оптимізації освітлення:

1. Тип освітлення:

- Природне освітлення: використання вікон, світлових ліхтарів.

- Штучне освітлення: лампи розжарювання, люмінесцентні, світлодіодні.[2]

2. Рівень освітленості:

- Відповідність нормам освітлення для різних типів робіт.

- Наприклад, для точних робіт рівень освітленості повинен бути не менше 500 люкс.[3]

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		67

3. Розташування джерел світла:

- Рівномірний розподіл світла на робочих місцях.
- Уникнення утворення тіней та блиску.[2]

4. Контроль якості освітлення:

- Регулярна перевірка рівня освітленості.
- Заміна ламп, які втратили яскравість.[5]

Методи оптимізації:

1. Використання сучасних технологій:

- Світлодіодні лампи, які мають високу енергоефективність.[5]

2. Моделювання освітлення:

- Використання програмного забезпечення для розрахунку рівня освітленості.[5]

3. Автоматизація:

- Встановлення датчиків руху для автоматичного включення/вимкнення світла.

Практичне значення:

- Зниження енергоспоживання.
- Покращення умов праці.
- Підвищення ефективності виробничих процесів.[5]

5.3 Моделювання та прогнозування небезпечних ситуацій

Моделювання та прогнозування небезпечних ситуацій є ключовим елементом управління ризиками на виробництві. Воно дозволяє оцінити потенційні загрози, розробити заходи щодо їх запобігання та мінімізувати наслідки.[3]

Основні етапи моделювання:

1. Ідентифікація небезпечних факторів:

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						68
Змн.	Арк.	№ докум.	Підпис	Дата		

- Аналіз технологічних процесів.

- Виявлення джерел небезпеки (механічні, хімічні, електричні).[3]

2. Створення математичних моделей:

- Використання методів теорії ймовірностей для оцінки ризиків.

- Застосування програмного забезпечення для моделювання аварійних ситуацій.[3]

3. Прогнозування наслідків:

- Оцінка впливу небезпечних ситуацій на працівників, обладнання та навколишнє середовище.

- Розрахунок зон ураження та часу реагування.[3]

4. Розробка заходів реагування:

- Створення планів евакуації.

- Впровадження систем раннього виявлення небезпеки.[3]

Методи моделювання:

1. Комп'ютерне моделювання:

- Використання спеціалізованих програм для аналізу ризиків.[3]

2. Експериментальне моделювання:

- Проведення тестів у контрольованих умовах.[3]

Практичне значення:

- Підвищення рівня безпеки на виробництві.

- Зменшення кількості аварій.

- Забезпечення ефективного управління ризиками.[3]

У процесі розробки інтернет-магазину "Initim" було враховано ключові аспекти охорони праці, які спрямовані на забезпечення безпеки працівників, оптимізацію умов праці та мінімізацію ризиків. Розглянуті розділи охоплюють пожежонебезпечні властивості речовин і матеріалів, заходи щодо оптимізації виробничого освітлення та моделювання небезпечних ситуацій.[1]

Аналіз пожежонебезпечних властивостей речовин дозволив визначити

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						69
Змн.	Арк.	№ докум.	Підпис	Дата		

основні характеристики, які впливають на ризик займання та горіння. Це сприяло розробці рекомендацій щодо безпечного зберігання та транспортування матеріалів, а також запобігання пожежам.[5]

Оптимізація виробничого освітлення забезпечила комфортні умови праці, знизила ризик травматизму та підвищила продуктивність. Використання сучасних технологій, таких як світлодіодні лампи та автоматизовані системи освітлення, дозволило досягти високої енергоефективності.[2]

Моделювання та прогнозування небезпечних ситуацій стало важливим інструментом для управління ризиками. Використання математичних моделей та комп'ютерного моделювання дозволило оцінити потенційні загрози, розробити заходи реагування та мінімізувати наслідки аварій.

Таким чином, реалізовані заходи охорони праці сприяли створенню безпечного середовища для роботи, що відповідає сучасним стандартам безпеки та ефективності.[3]

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						70
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У кваліфікаційній роботі було розроблено інтернет-магазин для продажу кави “Initim”. Метою проекту було створення сучасного, зручного та функціонального веб-ресурсу, який би задовольняв потреби клієнтів та сприяв розвитку бізнесу.

В ході дослідження було:

- Проаналізовано ринок інтернет-магазинів кави, визначено основні вимоги до функціоналу та дизайну.
- Обрано оптимальний стек технологій для реалізації React + Redux, Django + DRF, MariaDB.
- Розроблено архітектуру сайту, включаючи систему управління товарами, кошиком, оплатою.
- Реалізовано адаптивний дизайн для коректного відображення на різних пристроях.
- Забезпечено інтуїтивно зрозумілий інтерфейс для користувачів та зручну адмін-панель для керування магазином.
- Протестовано роботу сайту, виправлено можливі помилки та оптимізовано продуктивність.

Подальший розвиток проекту може включати вдосконалення маркетингових інструментів (наприклад, система лояльності, email-розсилки), інтеграцію зі сторонніми сервісами логістики, а також масштабування для виходу на нові ринки.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						71
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ПОСИЛАНЬ

1. В. В. Березуцький. Основи професійної безпеки та здоров'я людини: підручник. Харків: НТУ "ХПІ", 2018. 553 с.
2. В.І. Кошель, Г.П. Сав'юк, Б.С. Дзундза. Основи охорони праці. навчально-методичний посібник для студентів вищих навчальних закладів педагогічного напрямку. Івано-Франківськ: НАІР, 2020. 182 с.
3. Грибан В. Г., Фоменко А. Є., Казначеев Д. Г. Безпека життєдіяльності та охорона праці: підруч. Дніпро: Дніпроп. держ. ун-т внутр. справ, 2022. 388 с.
4. Дупай М.М., Денчук П.Н., Питель С.В. Організація, облік, звітність та оподаткування малого бізнесу – Т.: Астон, 2002.
5. Дупай М.М., Денчук П.Н., Питель С.В. Організація, облік, звітність та оподаткування малого бізнесу – Т.: Астон, 2002.
6. Дячун О.В. Організація, нормування та оплата праці. Навчальний посібник –Л., 2001.
7. Дячун О.В. Організація, нормування та оплата праці. Навчальний посібник –Л., 2001.
8. Масікевич, В. Ф. Райко, О. В. Шестопапов, О. Г. Янчик Основи професійної безпеки та здоров'я. Чернівці: «Місто», 2023. 288 с.
9. Мелех Л. В. Безпека життєдіяльності та охорона праці. Навчальний посібник. Львів: Львівський державний університет внутрішніх справ, 2022. 219с.
10. Основи програмування мовою Python: навчальний посібник. Ч.: ФОП Баликіна С.М., 2020. 180 с.
11. Розробка веб сайтів для початківців html – css - javascript версія 2022.12.11 Сергій Курінний.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						72
Змн.	Арк.	№ докум.	Підпис	Дата		

12. Bootstrap [Електронний ресурс] – Режим доступу до ресурсу: <https://bootswatch.com/> – Дата доступу: 15.05.2025. – Css файл для відображення адаптивного дизайну сайту.

13. Bootstrap [Електронний ресурс] – Режим доступу до ресурсу: <https://getbootstrap.com/> – Дата доступу: 17.05.2025. – Встановлення бібліотек bootstrap.

14. Cdnj [Електронний ресурс] – Режим доступу до ресурсу: <https://cdnjs.com/api> – Дата доступу: 18.05.2025. – Документація API.

15. Course [Електронний ресурс] – Режим доступу до ресурсу: <https://ua.udemy.com/> – Дата доступу: 07.05.2025. – Django, DRF, react, redux.

16. Django [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.djangoproject.com/en/5.2/howto/deployment/checklist/#switch-away-from-manage-py-runserver> – Дата доступу: 19.05.2025. – Веб фреймворк.

17. Django cors headers [Електронний ресурс] – Режим доступу до ресурсу: <https://pypi.org/project/django-cors-headers/3.6.0/> – Дата доступу: 21.05.2025. – Встановлення залежностей для Django.

18. Django is a Python web framework [Електронний ресурс] – Режим доступу до ресурсу: <https://www.django-rest-framework.org/> – Дата доступу: 05.05.2025. – Бекенд частина.

19. Django REST Framework [Електронний ресурс] – Режим доступу до ресурсу: <https://www.django-rest-framework.org/> – Дата доступу: 20.05.2025. – Основи фреймворку.

20. Django web framework (Python) [Електронний ресурс] – Режим доступу до ресурсу: https://developer.mozilla.org/en-US/docs/Learn_web_development/Extensions/Server-side/Django – Дата доступу: 13.05.2025. – Тutorials.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		73

21. Django REST Framework [Електронний ресурс] – Режим доступу до ресурсу:

https://developer.mozilla.org/enUS/docs/Learn_web_development/Extensions/Server-side/Django – Дата доступу: 14.05.2025. – Основи фреймворку.

22. Google Chrome [Електронний ресурс] – Режим доступу до ресурсу: https://www.google.com/intl/uk_ua/chrome/ – Дата доступу: 15.05.2025. – Програма для відображення сайту.

23. JavaScript підручник [Електронний ресурс] – Режим доступу до ресурсу: <https://w3schoolsua.github.io/js/index.html#gsc.tab=0> – Дата доступу: 12.05.2025. – Основи веб програмування.

24. Node.js [Електронний ресурс] – Режим доступу до ресурсу: <https://nodejs.org/en/download> – Дата доступу: 15.05.2025. – Програма для встановлення Python, та залежностей для фронтенду.

25. PayPal [Електронний ресурс] – Режим доступу до ресурсу: <https://developer.paypal.com/home/> – Дата доступу: 15.05.2025. – Створення акаунту для тестування оплати.

26. Postman [Електронний ресурс] – Режим доступу до ресурсу: <https://www.postman.com/> – Дата доступу: 15.05.2025. – Програма для тестування API.

27. React bootstrap [Електронний ресурс] – Режим доступу до ресурсу: <https://react-bootstrap.netlify.app/> – Дата доступу: 17.05.2025. – Фронтенд фреймворк.

28. React-paypal-js [Електронний ресурс] – Режим доступу до ресурсу: <https://www.npmjs.com/package/@paypal/react-paypal-js> – Дата доступу: 18.05.2025. – Встановлення реакт пейпал бібліотек.

29. Simple JWT [Електронний ресурс] – Режим доступу до ресурсу: <https://django-rest-framework-simplejwt.readthedocs.io/en/latest/> – Дата доступу: 22.05.2025. – Використання jwt токенів.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		74

30. Tailwind CSS Classes [Електронний ресурс] – Режим доступу до ресурсу: <https://shuffle.dev/tailwind/classes/paddings/py-3> – Дата доступу: 16.05.2025. – Класи на мові програмування Python.

31. Visual Studio Code [Електронний ресурс] – Режим доступу до ресурсу: <https://code.visualstudio.com/> – Дата доступу: 15.05.2025. – Програма для написання коду.

32. База даних [Електронний ресурс] – Режим доступу до ресурсу: <https://mariadb.org/uk/> – Дата доступу: 02.05.2025. – MariaDB.

33. Збірник задач з програмування [Електронний ресурс] – Режим доступу до ресурсу: <https://pythonexercises.rozh2sch.org.ua/> – Дата доступу: 06.05.2025. – Основи програмування.

34. Основи JavaScript [Електронний ресурс] – Режим доступу до ресурсу: <https://developer.mozilla.org/en-US/> – Дата доступу: 08.05.2025. – Розробка фронтенд частини, JSON.

35. Основи програмування JavaScript [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.javascript.info/> – Дата доступу: 11.05.2025. – основи JavaScript.

36. Основи програмування. Python. Частина 1 [Електронний ресурс] – режим доступу до ресурсу: <https://ela.kpi.ua/server/api/core/bitstreams/dbbe8ff5-11d7-4a92-918c-d1445c3d20a7/content> - Дата доступу: 01.05.2025 – Python.

37. Підручник з Python [Електронний ресурс] – Режим доступу до ресурсу: <https://www.python.org/> – Дата доступу: 10.05.2025. – Тьюторіал по Python.

38. Принципи проектування в Python [Електронний ресурс] – Режим доступу до ресурсу: <https://medium.com/django-unleashed/design-principles-in-python-django-3011d6cee547> – Дата доступу: 09.05.2025. – Принципи написання коду.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		75

39. Фреймворк Flask [Електронний ресурс] – Режим доступу до ресурсу: <https://flask.palletsprojects.com/en/stable/> – Дата доступу: 07.05.2025. –

Що таке фреймворк і як з ним взаємодіяти.

40. Фреймворк [Електронний ресурс] – Режим доступу до ресурсу: <https://www.django-rest-framework.org/> – Дата доступу: 01.05.2025. – Бекенд частина.

41. Фронтенд частина [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.legacy.reactjs.org/> – Дата доступу: 03.05.2025. – React.js.

42. Фронтенд [Електронний ресурс] – Режим доступу до ресурсу: <https://react-redux.js.org/> – Дата доступу: 04.05.2025. – React.js, Redux.

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						76
Змн.	Арк.	№ докум.	Підпис	Дата		

ДОДАТОК А. КОД СТОРІНКИ СКРИПТА РЕЄСТРАЦІЇ

```
import React, { useEffect, useState } from "react";
import { useDispatch, useSelector } from "react-redux";
import { Link, useNavigate, useLocation } from "react-router-dom";
import { Form, Button, Row, Col } from "react-bootstrap";
import Loader from "../components/Loader";
import Message from "../components/Message";
import { register } from "../actions/userActions";
import FormContainer from "../components/FormContainer";
function RegisterScreen() {
  const [name, setName] = useState("");
  const [email, setEmail] = useState("");
  const [password, setPassword] = useState("");
  const [confirmPassword, setConfirmPassword] = useState("");
  const [message, setMessage] = useState("");
  const dispatch = useDispatch();
  const location = useLocation();
  const navigate = useNavigate();
  const redirect = location.search ? location.search.split("=")[1] : "/";
  const userRegister = useSelector((state) => state.userRegister);
  const { loading, error, userInfo } = userRegister;
  useEffect(() => {
    if (userInfo) {
      navigate(redirect);
    }, [navigate, userInfo, redirect]);
  }, [navigate, userInfo, redirect]);
  const submitHandler = (e) => {
    e.preventDefault();
```

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						77
Змн.	Арк.	№ докум.	Підпис	Дата		

```

if (password !== confirmPassword) {
  setMessage("Паролі не співпадають");
} else {
  dispatch(register(name, email, password));
}
return (
  <FormContainer>
    <h1>Зареєструватися</h1>
    {message && <Message variant="danger">{message}</Message>}
    {error && <Message variant="danger">{error}</Message>}
    {loading && <Loader />}
    <Form onSubmit={submitHandler}>
      <Form.Group controlId="name">
        <Form.Label>Ім'я</Form.Label>
        <Form.Control
          required
          type="name"
          placeholder="Введіть ім'я"
          value={name}
          onChange={(e) => setName(e.target.value)}
        ></Form.Control>
      </Form.Group>
      <Form.Group controlId="email">
        <Form.Label>Електронна пошта</Form.Label>
        <Form.Control
          required
          type="email"
          placeholder="Введіть електронну пошту"
          value={email}

```

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						78
Змн.	Арк.	№ докум.	Підпис	Дата		


```

onChange={(e) => setEmail(e.target.value)}
></Form.Control>
</Form.Group>
<Form.Group controlId="password">
<Form.Label>Пароль</Form.Label>
<Form.Control
required
type="password"
placeholder="Введіть пароль"
value={password}
onChange={(e) => setPassword(e.target.value)}
></Form.Control>
</Form.Group>
<Form.Group controlId="passwordConfirm">
<Form.Label>Підтвердження пароля</Form.Label>
<Form.Control
required
type="password"
placeholder="Підтвердіть пароль"
value={confirmPassword}
onChange={(e) => setConfirmPassword(e.target.value)}
></Form.Control>
</Form.Group>
<Button type="submit" variant="primary">
Зареєструватися
</Button>
</Form>
<Row className="py-3">

```

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						79
Змн.	Арк.	№ докум.	Підпис	Дата		

<Col>

Маєте акаунт?{" "}

<Link to={redirect ? `/login?redirect=\${redirect}` : "/login"}>

Увійти

</Link>

</Col>

</Row>

</FormContainer>);}

export default RegisterScreen;

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						80
Змн.	Арк.	№ докум.	Підпис	Дата		

ДОДАТОК Б. КОД FRONTEND СТОРІНКИ З ВІДГУКАМИ

```
import React, { useState, useEffect } from "react";
import { Row, Col, ListGroup, Button } from "react-bootstrap";
import axios from "axios";
import Rating from "../components/Rating";
import Loader from "../components/Loader";
import Message from "../components/Message";
import { Bar } from "react-chartjs-2";
import { CategoryScale } from "chart.js";
import Chart from "chart.js/auto";
Chart.register(CategoryScale);
function AnalyticsListScreen() {
  const [reviews, setReviews] = useState([]);
  const [averageRating, setAverageRating] = useState(null);
  const [averageRatings, setAverageRatings] = useState([]);
  const [loading, setLoading] = useState(true);
  const [error, setError] = useState(null);
  const [showAllReviews, setShowAllReviews] = useState(false);
  useEffect(() => {
    const fetchReviews = async () => {
      try {
        const { data } = await axios.get("/api/reviews/");
        setReviews(data);
        setLoading(false);
      } catch (err) {
        setError(err.message);
        setLoading(false);
      }
    };
  });
```

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						81
Змн.	Арк.	№ докум.	Підпис	Дата		

```

const fetchAverageRating = async () => {
  try {
    const { data } = await axios.get("/api/reviews/user/1/average/");
    setAverageRating(data.average_rating);
    setLoading(false);
  } catch (err) {
    setError(err.message);
    setLoading(false);}}};

const fetchAverageRatings = async () => {
  try {
    const { data } = await axios.get("/api/reviews/users/average/");
    setAverageRatings(data);
    setLoading(false);
  } catch (err) {
    setError(err.message);
    setLoading(false);}}};

fetchReviews();
fetchAverageRating();
fetchAverageRatings();
}, []);

const toggleReviews = () => {
  setShowAllReviews(!showAllReviews);
};

return (
  <div>
    <h1>Аналітика відгуків</h1>
    {loading ? (
      <Loader />

```

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						82
Змн.	Арк.	№ докум.	Підпис	Дата		

```

) : error ? (
<Message variant="danger">{error}</Message>
) : (
<Row>
<Col md={6}>
<h2>Відгуки</h2>
{reviews.length === 0 && (
<Message variant="info">Відгуків ще немає</Message>
)}
<ListGroup variant="flush">
{showAllReviews
? reviews.map((review) => (
<ListGroup.Item key={review._id}>
<strong>{review.name}</strong>
<Rating value={review.rating} color="yellow" />
<p>{review.createdAt.substring(0, 10)}</p>
<p>{review.comment}</p>
</ListGroup.Item>
))
reviews.slice(0, 1).map((review) => (
<ListGroup.Item key={review._id}>
<strong>{review.name}</strong>
<Rating value={review.rating} color="yellow" />
<p>{review.createdAt.substring(0, 10)}</p>
<p>{review.comment}</p>
</ListGroup.Item>
)}}
</ListGroup>

```

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						83
Змн.	Арк.	№ докум.	Підпис	Дата		

```

{reviews.length > 1 && (
<Button
variant="primary"
onClick={toggleReviews}
className="mt-3"
>
{showAllReviews ? "Сховати відгуки" : "Більше відгуків"}
</Button>
)}
</Col>
<Col md={6}>
<Bar
data={{
labels: reviews.map((review) => review.name),
datasets: [
{
label: "Рейтинг",
data: reviews.map((review) => review.rating),
backgroundColor: "rgba(21, 21, 196, 0.6)",},],}}
options={{
responsive: true,
plugins: {
legend: {
position: "top",},},}}/>
<Col md={6}>
<div style={{ width: "400px", height: "300px" }}>
<h2>Середній рейтинг</h2>
<Bar

```

					2025.КВР.123.412.13.00.00 ПЗ	Арк.
						84
Змн.	Арк.	№ докум.	Підпис	Дата		

```

data={{
labels: averageRatings.map((rating) => {
const userReview = reviews.find(
(review) => review.user === rating.user_id
);
return userReview
? userReview.name
: `Користувач ${rating.user_id}`;
}),
datasets: [{
label: "Середній рейтинг",
data: averageRatings.map(
(rating) => rating.average_rating),
backgroundColor: "rgba(75, 192, 192, 0.6)",},],}}
options={{
responsive: true,
plugins: {
legend: {
position: "top",},},
scales: {
y: {
beginAtZero: true,},},}}/>
</div>
</Col>
</Col>
</Row>}}
</div>);}

export default AnalyticsListScreen;

```

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						85
Змн.	Арк.	№ докум.	Підпис	Дата		

ДОДАТОК В. КОД BACKEND СТОРІНКИ З ШЛЯХАМИ
ВІДГУКІВ

```
from base.views.review_views import get_all_reviews,  
get_users_average_ratings, get_user_average_rating  
path('api/reviews/users/average/', get_users_average_ratings, name='get-  
users-average-ratings'),  
path('api/reviews/user/<int:user_id>/average/', get_user_average_rating,  
name='get-user-average-rating'),
```


ДОДАТОК Г. КОД BACKEND СТОРІНКИ GET ЗАПИТІВ З ВІДГУКІВ

```

from django.db.models import Avg
from rest_framework.decorators import api_view
from rest_framework.response import Response
from base.models import Review
from base.serializers import ReviewSerializer

@api_view(['GET'])
def get_all_reviews(request):
    reviews = Review.objects.all()
    serializer = ReviewSerializer(reviews, many=True)
    return Response(serializer.data)

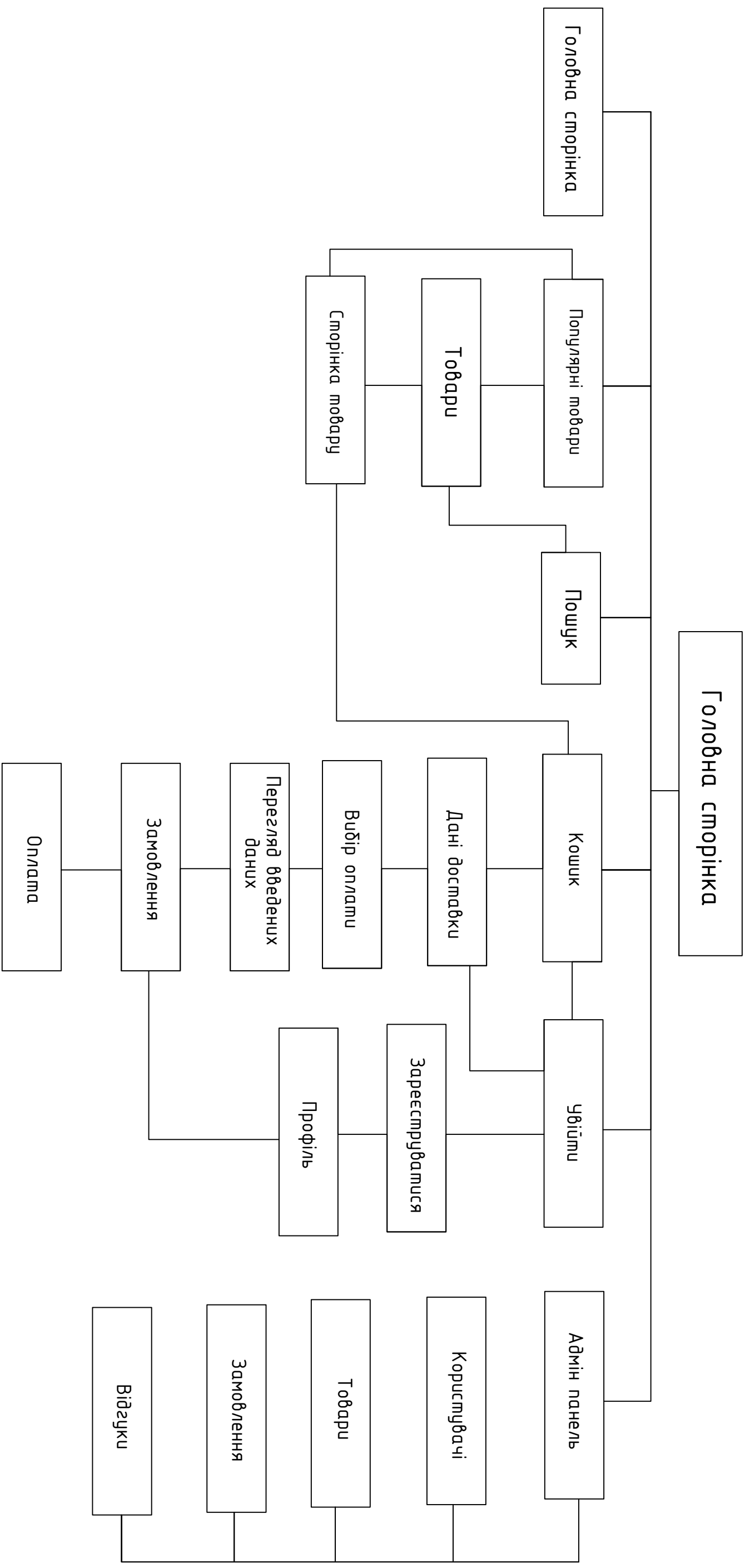
@api_view(['GET'])
def get_users_average_ratings(request):
    average_ratings=Review.objects.values('user_id').annotate(average_rating=
Avg('rating'))
    return Response(average_ratings)

@api_view(['GET'])
def get_user_average_rating(request, user_id):
    average_rating=Review.objects.filter(user_id=user_id).aggregate(Avg('rating'))['rating__avg']
    return Response({'user_id': user_id, 'average_rating': average_rating})

```

					2025.KBP.123.412.13.00.00 ПЗ	Арк.
						87
Змн.	Арк.	№ докум.	Підпис	Дата		

Структурна схема саїтту

[illegible]

Скрипт реєстрації

```
<FormContainer>
  <h1>Зареєструватися</h1>
  {message && <Message variant="danger">{message}</Message>}
  {error && <Message variant="danger">{error}</Message>}
  {loading && <Loader />}
  <Form onSubmit={submitHandler}>
    <Form.Group controlId="name">
      <Form.Label>Ім'я</Form.Label>
      <Form.Control
        required
        type="name"
        placeholder="Введіть ім'я"
        value={name}
        onChange={(e) => setName(e.target.value)}
      ></Form.Control>
    </Form.Group>
    <Form.Group controlId="email">
      <Form.Label>Електронна пошта</Form.Label>
      <Form.Control
        required
        type="email"
        placeholder="Введіть електронну пошту"
        value={email}
        onChange={(e) => setEmail(e.target.value)}
      ></Form.Control>
    </Form.Group>
    <Form.Group controlId="password">
      <Form.Label>Пароль</Form.Label>
      <Form.Control
        required
        type="password"
        placeholder="Введіть пароль"
        value={password}
        onChange={(e) => setPassword(e.target.value)}
      ></Form.Control>
    </Form.Group>
    <Form.Group controlId="passwordConfirm">
      <Form.Label>Підтвердження пароля</Form.Label>
      <Form.Control
        required
        type="password"
        placeholder="Підтвердіть пароль"
        value={confirmPassword}
        onChange={(e) => setConfirmPassword(e.target.value)}
      ></Form.Control>
    </Form.Group>
    <Button type="submit" variant="primary">
      Зареєструватися
    </Button>
  </Form>
  <Row className="py-3">
    <Col>
      Маєте акаунт?{" "}
      <Link to={redirect ? `/${login?redirect=${redirect}}` : `/${login}`}>
        Увійти
      </Link>
    </Col>
  </Row>
</FormContainer>
```

					2025.KBP.123.4 12.13.00.00 ТП				
Зм.	Арк.	№ документа	Підпис	Дата	Розробка інтернет-магазину з продажу кави "Initim"	Літ.	Маса	Масшт.	
Розробив		Партика Н.В.						1:1	
Перевірив		Штокало Л.Я.							
Консульт.									
Реценз.									
Н.Контр.	Юзьків А.В.					ВСП «ТФК ТНТУ ім. Івана Пулюя» КІ-412 м. Тернопіль			
Зав. каф.						Аркуш	Аркушів 1		

