

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «бакалавр»
за спеціальністю 121 «Інженерія програмного забезпечення»
(освітня програма «Інженерія програмного забезпечення»)

на тему:

«Маркетплейс комп'ютерної техніки»

Виконав студент групи ПЗ-21-3
ПАНЧУК Петро Сергійович

Керівник роботи:
ЄФРЕМОВ Юрій Миколайович

Рецензент:
ГОРДЛІВСЬКИЙ Юрій Олександрович

Житомир – 2025

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

«ЗАТВЕРДЖУЮ»

Зав. кафедри інженерії
програмного забезпечення

Тетяна Вакалюк

«26» лютого 2025 р.

ЗАВДАННЯ
на кваліфікаційну роботу

Здобувач вищої освіти: **ПАНЧУК Петро Сергійович**

Керівник роботи: **ЄФРЕМОВ Юрій Миколайович**

Тема роботи: **«Маркетплейс комп'ютерної техніки»,**

затверджена Наказом закладу вищої освіти від **«14» лютого 2025 р., №61/с**

Вихідні дані для роботи: **зазначається предмет і об'єкт дослідження**

Консультанти з бакалаврської кваліфікаційної роботи із зазначенням розділів,
що їх стосуються:

Розділ	Консультант	Завдання видав	Завдання прийняв
1	Єфремов Ю.М.	27.02.2025р.	27.02.2025р.
2	Єфремов Ю.М.	27.02.2025р.	27.02.2025р.
3	Єфремов Ю.М.	27.02.2025р.	27.02.2025р.

РЕФЕРАТ

Випускна кваліфікаційна робота бакалавра складається з вебдодатку маркетплейсу комп'ютерної техніки та пояснювальної записки. Пояснювальна записка до випускної роботи містить 58 сторінок, 36 ілюстрацій та 14 таблиць

Метою випускної роботи є розробка та реалізація програмного комплексу для онлайн маркетплейсу комп'ютерної техніки

Архітектура платформи побудована за принципами сучасної багаторівневої вебсистеми. Вона забезпечує збереження, оновлення товарів, їх пошук та фільтрацію. Дозволяє користувачам створювати та керувати акаунтами. Базується на фреймворку Symfony

Додаток має зручний інтерфейс, який дозволяє користувачам зручно купувати та продавати комп'ютерні товари.

КЛЮЧОВІ СЛОВА: МАРКЕТПЛЕЙС, ЕЛЕКТРОННА КОМЕРЦІЯ, КОМП'ЮТЕРНА ТЕХНІКА, ЗАМОВЛЕННЯ, ВЕБДОДАТОК

					ІПЗ.КР.Б – 121 – 25 – ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Панчук П.С.			Маркетплейс комп'ютерної техніки	Літ.	Арк.	Аркушів
Керівник		Єфремов Ю.М.					3	58
						ФІКТ Гр. ІПЗ-21-3		
Н. контр.		Тичина О.П.						
Зав. каф.		Вакалюк Т.А						

ABSTRACT

The bachelor's thesis consists of a web application of a computer equipment marketplace and an explanatory note. The explanatory note to the thesis contains 56 pages, 35 illustrations and 15 tables

The purpose of the graduation work is to develop and implement a software package for an online marketplace of computer equipment

The platform architecture is built on the principles of a modern multi-level web system. It provides storage, updating of goods, their search and filtering. Allows users to create and manage accounts. Based on the Symfony framework

The application has a user-friendly interface that allows users to conveniently buy and sell computer goods.

KEYWORDS: MARKETPLACE, ELECTRONIC COMMERCE,
COMPUTER EQUIPMENT, ORDER, WEB APPLICATION

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		4

ЗМІСТ

РЕФЕРАТ.....	3
ABSTRACT	4
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	7
ВСТУП.....	8
РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ У СТВОРЕННІ СПЕЦІАЛІЗОВАНИХ ТОРГОВИХ ПЛАТФОРМ.....	10
1.1 Постановка задачі.....	10
1.2 Аналіз аналогів програмного продукту	12
1.3 Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення	18
Висновки до першого розділу.....	21
РОЗДІЛ 2 ПРОЕКТУВАННЯ МАРКЕТПЛЕЙСУ КОМП'ЮТЕРНОЇ ТЕХНІКИ	22
2.1 Визначення вимог та варіантів використання. Проектування структури системи.....	22
2.2 Розробка фізичної моделі бази даних системи.	28
2.3 Проектування та реалізація алгоритмів роботи системи.	32
2.4 Реалізація маркетплейсу комп'ютерної техніки.	35
Висновки до другого розділу	39
РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З МАТРКЕТПЛЕЙСОМ КОМП'ЮТЕРНОЇ ТЕХНІКИ.....	40
3.1 Порядок встановлення та налаштування параметрів платформи	40
3.2 Структура інтерфейсу	41
3.3. Тестування платформи.	50
Висновки до третього розділу.....	54
ВИСНОВКИ	55
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	56

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		5

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		6

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БД – База даних.

СУБД – Система управління базами даних.

CRUD – Create, Read, Update, Delete – основні операції управління даними.

ORM – Object-Relational Mapping.

API – Інтерфейс прикладного програмування.

MVC – Model View Controller – схема розділення керуючої логіки і даних додатків.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		7

ВСТУП

Атуальність теми полягає у тому, що у сучасному суспільстві, сфера електронної комерції набуває все більшої ваги. Це зумовлено стрімким розвитком інформаційних технологій та зміною споживчої поведінки. Маркетплейси займають особливу нішу в нашому суспільстві, надаючи можливість легко знайти як продавців певного товару, так і аудиторію, яка буде його купувати. Вони забезпечують швидку комунікацію та можливість зручно проводити угоди між сторонами. Натомість, сегмент комп'ютерної техніки є одним з найбільш динамічних та найбільш конкурентних на ринку. Дрібні продавці та магазини потребують інструментів для зручного продажу товарів в інтернеті, без необхідності створювати власний вебсайт, а покупців привабить великий вибір комплектуючих та можливість знайти найвигідніші пропозиції від різних постачальників. Загальні маркетплейси наявні на ринку, вони дуже розвинені та популярні, але їм бракує спеціалізації саме на комп'ютерній техніці, тому вони не завжди є ідеальним вибором. Вони часто не враховують деталі технічних характеристик та не можуть надати особливого функціоналу, який буде корисним окремо вибраному сегменту. Отже розробка вузько-спеціалізованого маркетплейсу саме для комп'ютерної техніки є актуальним завданням, яке може задовольнити потреби як кінцевих споживачів, так і продавців.

Метою випускної роботи є розробка та реалізація програмного комплексу для онлайн маркетплейсу комп'ютерної техніки.

Для реалізації поставленої мети, необхідно виконати наступні завдання:

1. Проаналізувати предметну область. Виявити основні вимоги до функціональності маркетплейсу.
2. Сформулювати структуру системи, визначити її архітектуру та об'єктно-орієнтовану модель.
3. Вибір та обґрунтування технологічного стеку для реалізації системи.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		8

4. Проектування структурних складових системи; Реалізація програмного комплексу для веб-орієнтованої системи маркетплейсу комп'ютерної техніки.

Об'єктом дослідження є особливості електронної комерції в сучасному світі

Предметом дослідження є методи та засоби розробки веб-орієнтованої системи маркетплейсу для продажу комп'ютерної техніки, з урахуванням специфіки товару та функціональних вимог.

Практичне значення отриманих результатів: результати виконаної роботи мають практичну цінність у сфері електронної комерції. Систему можна використати як окрему платформу для надання послуг маркетплейсу, з подальшою можливістю розвитку та масштабування. Рішення доцільне для малих магазинів та незалежних продавців, а також для використання споживачами. Розробка не є системою, яку можна впровадити на підприємствах, а є незалежним продуктом, послугами якого можна скористатись. Вона може бути використана в господарській діяльності компанії-замовника самої розробки.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		9

РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ У СТВОРЕННІ СПЕЦІАЛІЗОВАНИХ ТОРГОВИХ ПЛАТФОРМ

1.1 Постановка задачі

Метою кваліфікаційної роботи бакалавра є створення онлайн-маркетплейсу для продажу та купівлі комп'ютерної техніки, комплектуючих та периферії з можливістю подальшого масштабування в майбутньому.

Система має забезпечити легку взаємодію між зацікавленими сторонами в одному онлайн-ресурсі, надати максимальну безпеку особистим даним відвідувачів. Також вона має бути вузько-спеціалізованою для управління товарами зв'язаними з комп'ютерами, надавати унікальні, в порівнянні з конкурентами, можливості. Окрім безпеки даних, необхідно передбачити фінансову безпеку, методи запобігання шахрайству та обманам. Це дозволить завоювати довіру клієнтів та не програвати усталеним гравцям на ринку в цьому плані.

Онлайн платформа має представляти собою зв'язані функціональні модулі, які надаватимуть необхідні та унікальні, в рамках даної системи, можливості. Вони повинні працювати злагоджено та продумано, як окремо, так і між собою, щоб надавати клієнтам максимально рівномірний досвід користування:

- модуль автентифікації та авторизації – забезпечує безпечну реєстрацію даних, можливість увійти в свій акаунт та призначення ролей для користувачів. Ролі контролюватимуть доступ до ресурсів веб-сайту відповідно до потреби;
- модуль управління товарами. Модуль дозволяє створювати товари (розміщати їх на сайті для продажу), назначати їм категорію та атрибути, додавати фото до товару. Всю інформацію можна буде відредагувати, а сам товар видалити, якщо буде така необхідність;

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		10

- модуль фільтрації та пошуку дозволить переглядати каталог товарів, шукати потрібний за іменем та фільтрувати отримані дані за категоріями, окремими атрибутами та ціною;
- модуль кошика та оформлення замовлення. Модуль дозволяє покупцям зберегти знайдені товари до кошика, змінити їх кількість, переглянути загальну суму та перейти до оформлення замовлення;
- модуль управління замовленнями. Модуль надає можливість продавцю переглянути його замовлені товари, змінити статус замовлення. Покупці також можуть переглянути як активні замовлення, що ще не були виконані, так їх історію;
- модуль відгуків – дозволить покупцям залишати відгуки на товари: оцінювати та писати коментар. Інші покупці будуть бачити відгук та можуть вирішити чи варто купляти товар саме цей товар, саме у цього продавця. Продавці також матимуть рейтинг;
- модуль адміністрування. Модуль дозволить людям мати роль адміністратору сайту. Вони мають можливість карати порушників та допомагати клієнтам у критичних ситуаціях. Також вони відіграватимуть ключову роль у модерації контенту, такого як створення нового товару. Відвідувачі також матимуть можливість залишати скарги на товар, продавця чи покупця.

В результаті, веб-сайт повинен надавати змогу бажаному купити товар, або його продати, максимально зручно. Важливо щоб розробка була сфокусована на своїй вузькій області і не перевищувала задуманий обсяг.

Основні етапи розробки платформи:

1. Розробка архітектури продукту
 - а. моделювання та побудова схеми бази даних.
 - б. визначення архітектури клієнтської та серверної частини
2. Проектування інтерфейсу користувача
3. Реалізація бекенду та фронтенду

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		11

- a. розробка системи реєстрації, авторизації та імплементація логіки ролей
- b. створення механізмів для роботи з товарами
- c. реалізація кошика, замовлення та відгуків
- d. глобальна та уніформна стилізація елементів інтерфейсу

4. Забезпечення безпеки

- a. валідація форм та їх CSRF-захист
- b. обмеження маршрутів за ролями та перевіркою особи
- c. обробка помилок та виключень

5. Тестування додатку

- a. функціональне тестування модулів
- b. тестування графічного інтерфейсу

За результатом роботи, буде створена повністю функціональна платформа, на якій користувачі зможуть шукати та купувати комп'ютерні товари, виставлені на продаж після проходження модерації адміністраторами. Клієнти зможуть взаємодіяти один з одним: залишати відгуки та писати повідомлення. Вони матимуть можливість переглядати та управляти інформацією про товари та замовлення.

1.2 Аналіз аналогів програмного продукту

Ринок електронної комерції забитий різними магазинами та маркетплейсами. В таких конкурентних обставинах, варто провести аналіз популярних платформ, щоб зрозуміти на чому треба сфокусувати увагу та на чому можна в них виграти.

Одним з найвідоміших маркетплейсів України є Rozetka (рис. 1.1.).

Ця платформа була від початку просто інтернет магазином, але згодом розрослася і додала також можливість продавати товари користувачам.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		12

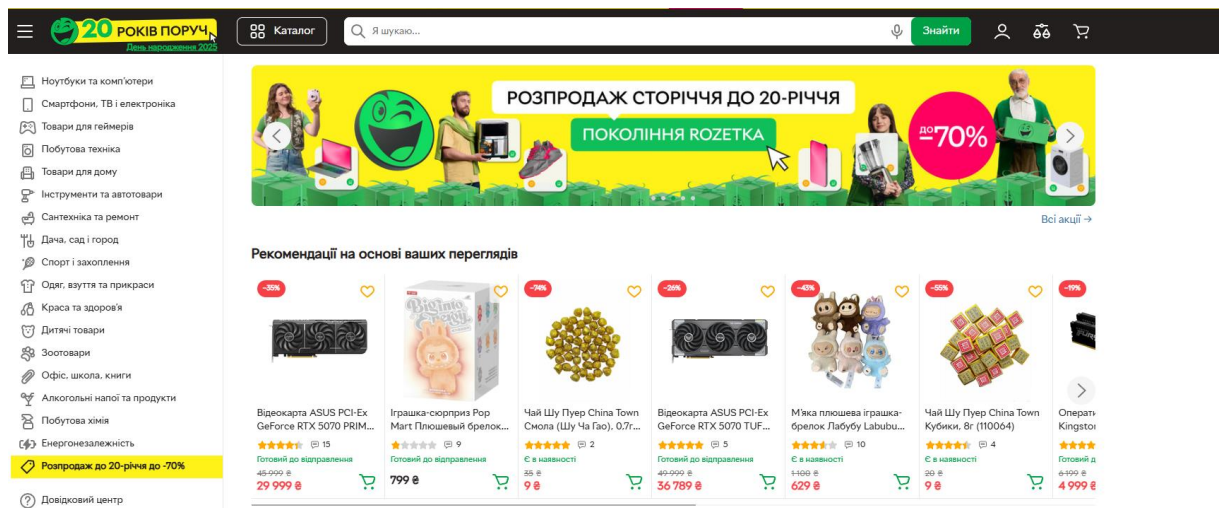


Рис. 1.1. Онлайн-маркетплейс Rozetka [1]

Основними перевагами Rozetka є:

- Широкий асортимент товарів
- Підтримка сторонніх продавців
- Зручна система фільтрації та категорій
- Зручна навігація

Недоліки Rozetka:

- приватні продавці обмежені: повинні бути юридичними особами
- відсутність вузької спеціалізації на ПК-комплектуючих

AliExpress – міжнародний маркетплейс, який є одним з найбільших на ринку. Він працює з виробниками з КНР та інших східних країн, товари можна купити та доставити в будь яку країну світу. Доставка та оплата контролюється самим сервісом. Є функція, яка підбирає схожі товари до того, який був переглянутий. Добре розвинуті соціальні функції, можна відстежувати окремі магазини (як на соц. мережах), писати їм в приватні повідомлення ставити лайки, ділитись продуктом. Головна сторінка зображена на рис. 1.2.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		13

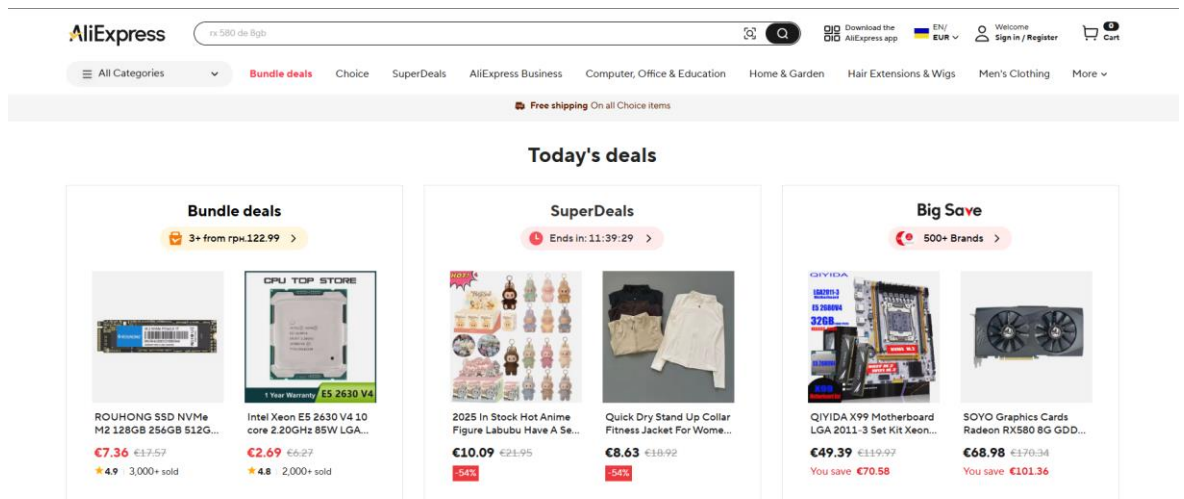


Рис. 1.2. AliExpress [2]

Основними перевагами AliExpress є:

- Широкий асортимент товарів
- Дуже детальна категоризація товарів

Недоліки AliExpress:

- Відсутність фільтрації товарів
- Обмежена співпраця з малим бізнесом та окремими підприємцями
- Незручна навігація, загромаджений дизайн

Алло (рис. 1.3.) – українська площадка, яка активно розвивається. Вона працює і як інтернет магазин, на якому торгує сама компанія, і як маркетплейс, на якому інші магазини чи виробники можуть розміщати свої товари. Веб-сайт фокусується саме на електроніку, або наближені до неї категорії. Імплементовано особистий профіль користувача, в якому можна стежити за замовленням та зберігати товари, що сподобались.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		14

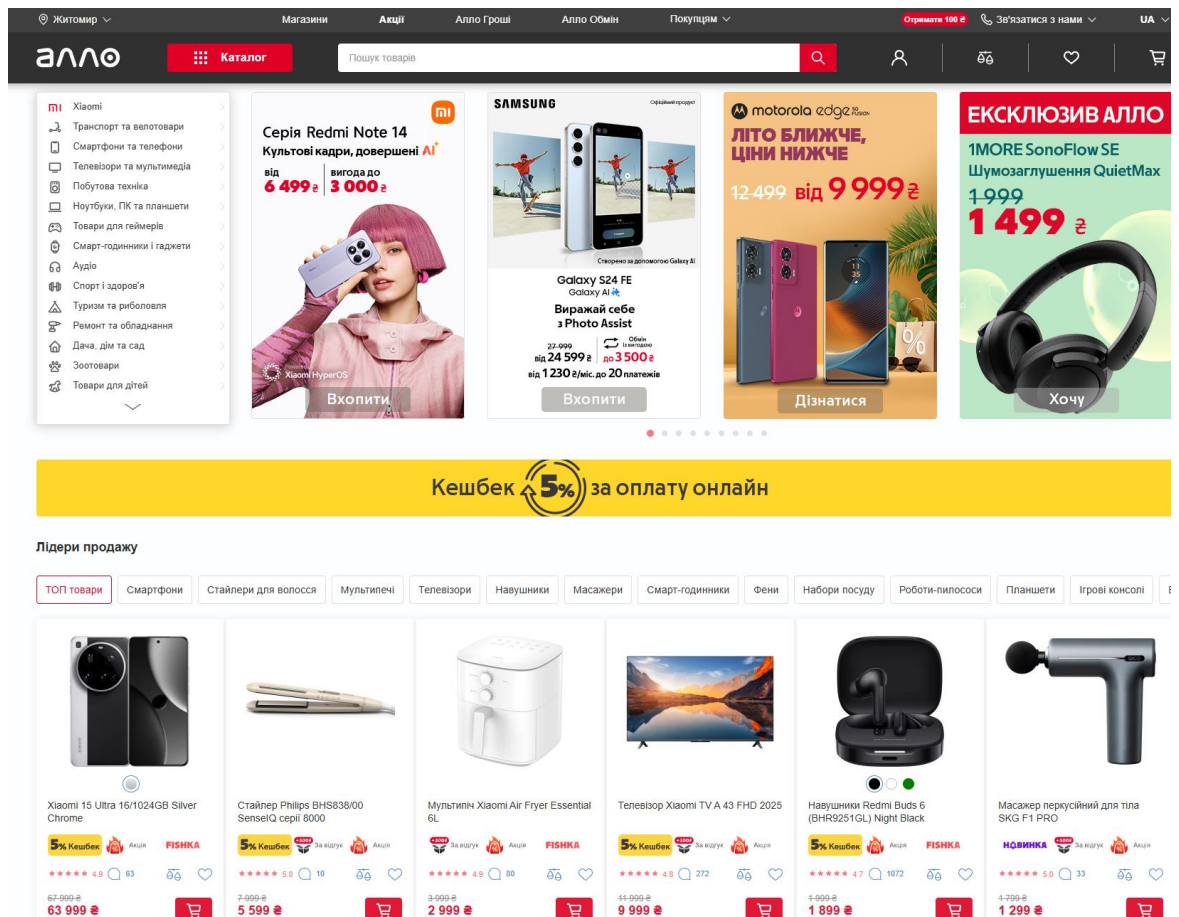


Рис. 1.3. Алло [3]

Переваги Алло:

- Акцент на перевірену електроніку та наявність авторизованих магазинів.
- Зручні та детальні фільтри по атрибутам.
- Вузька категоризація товару

Недоліки Алло:

- Не всі мають можливість продавати на платформі

OLX – онлайн-платформа для розміщення оголошень. Нею легко користуватись, поріг входження низький. Але натомість сайт не такий структурований, він не керує оплатою та доставкою сам, тільки дає платформу для спілкування (рис. 1.4).

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				15
Змн.	Арк.	№ докум.	Підпис	Дата		

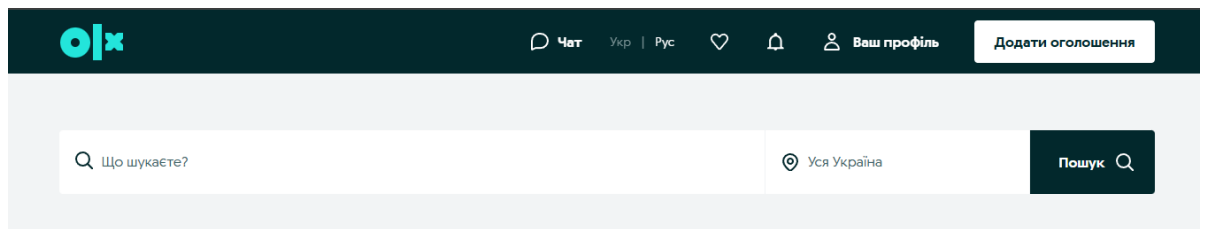


Рис. 1.4. Дошка оголошень OLX [4]

Переваги OLX:

- Простота доступу
- Масовість
- Можливість прямого контакту між продавцем та покупцем

Недоліки OLX:

- Нема функціоналу магазину, тільки оголошення
- Відсутність структурованого каталогу
- Немає відгуків та рейтингу користувачів.

Prom – один з найбільших українських маркетплейсів (рис. 1.5). Він надає можливість власникам бізнесів приєднатись до порталу та викладати свої товари на сайті. Prom надає продавцям багато інструментів та багато можливостей кастомізації сторінки свого магазину. Комп'ютерні товари також в продажу, хоча вони не особливо структуровані, але за допомогою дуже великої кількості фільтрів, можна знайти все що завгодно.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		16

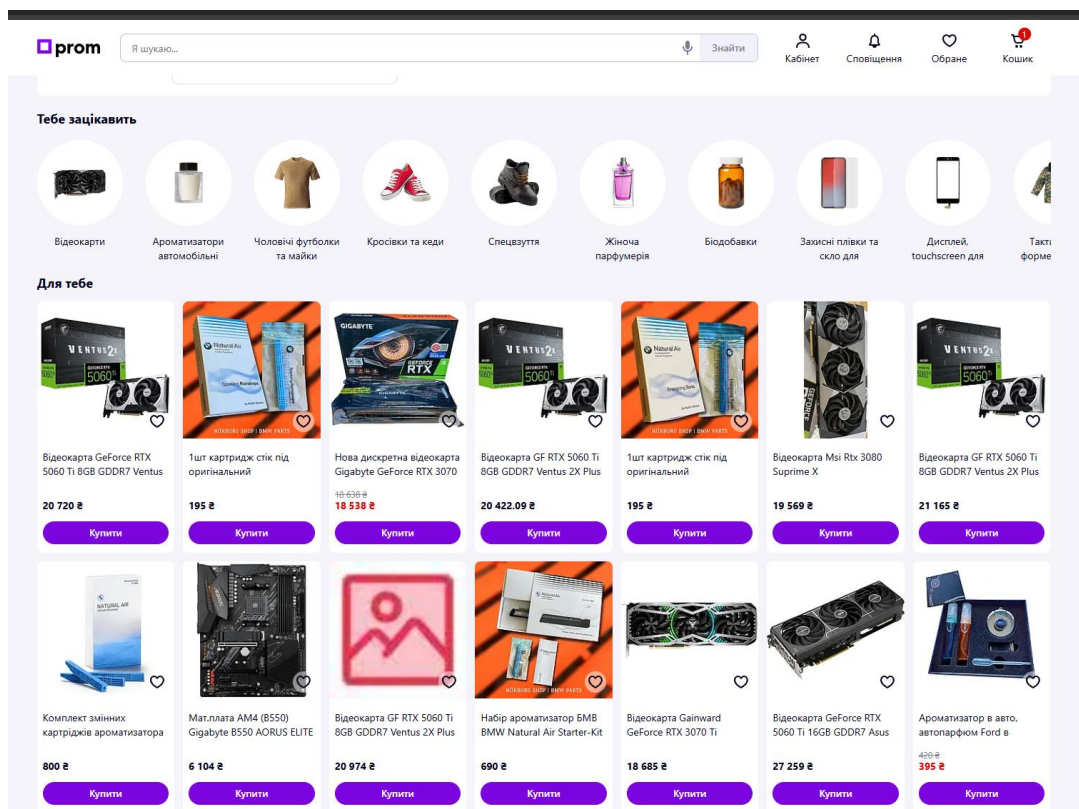


Рис. 1.5. Маркетплейс Prom.ua [5]

Переваги Prom:

- Легка інтеграція для бізнесів
- Направленість на підтримку продавців
- Динамічна система фільтрації, що облегшує масштабування

Недоліки Prom:

- Низька організованість, неінтуїтивна навігація для покупця
- Люди без ФОП або юрособи не можуть повноцінно торгувати
- Обмежена модерація

Після огляду існуючих платформ електронної комерції, можна дійти висновку: хоча ці ресурси мають свої, окремі, сильні сторони, вони не можуть об'єднати їх повною мірою, щоб запропонувати користувачам досвід спеціалізованого маркетплейсу. Також вони мають певні недостатки, які можуть заважати, особливо щодо доступності для малих продавців, чи глибини технічних характеристик. Якщо об'єднати сильні сторони перелічених ресурсів та мітигувати їх недоліки, можна заповнити нишу між універсальними

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		17

платформами та магазинами окремого виробника. Створити по істині конкурентно-спроможний продукт з можливістю домінувати в своєму сегменті.

Таким чином, було прийняте рішення створити платформу вузької спеціалізації на сегменті комп'ютерної техніки. Вона надаватиме можливість клієнтам, як купувати речі, так і продавати їх іншим. Система дозволить малим продавцям чи магазинам знайти аудиторію без потреби у власному сайті. Для продажу не потрібна буде реєстрація як ФОП чи юридичної особи. Тому в плані продажу вона буде орієнтована на звичайних людей. Такий ресурс має на меті поєднати соціальні аспекти та зручність універсальних маркетплейсів з глибиною та точністю нішевих магазинів.

1.3 Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення

Програмний продукт буде побудований на основі клієнт-серверної архітектури. Розробляти він буде на мові PHP [6]. Фронтенд та бекенд частини будуть реалізовані Symfony.

Symfony [7] – це сучасний фреймворк для розробки веб-застосунків. Він підтримує архітектурну модель MVC [8], що дозволить вести проект структуровано та полегшує масштабування розробки. Ідея Symfony заключається в зменшенні повторень типових рішень, які є в кожному веб-проекті, щоб розробники могли зфокусуватись на унікальності проекту, а не на самому базовому функціоналі. Окрім фреймворку, Symfony також є незалежною збіркою бібліотек, кожна з яких можна використовувати окремо, щоб налаштувати проект під себе. Для їх завантаження використовується Composer [9]. Окрім структурності розробки, Symfony також має багато інтегрованих інструментів та рішень для різних проблем.

Одним з таких інструментів є Doctrine [10]. Це одне з найпопулярніших рішень для роботи з базами даних, яке ґрунтується на реалізації ORM патерну, та побудоване на абстрактному рівні взаємодії з базою даних. Основним плюсом Doctrine є можливість працювати з сутностями, як просто з об'єктами проекту.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				18
Змн.	Арк.	№ докум.	Підпис	Дата		

Одразу декілька функцій дозволяє замінити повсюдне написання сирого коду SQL. Використання діалекту Doctrine Query Language, або класу QueryBuilder, що дозволяє більш інтуїтивно та зрозуміло будувати запити до бази даних.

Symfony, з коробки, містить зручну та потужну систему маршрутизації. Вона дозволяє встановити відповідності між адресою та контролером, функціями в ньому. Маршрутизація має механізм контролю доступу, який дозволяє давати чи не давати доступ до певної сторінки користувачу, зважаючи на його роль. Все це можна налаштувати під потреби окремого проекту. Завдяки чіткому розділу логіки між контролерами, представленнями та маршрутизацією, забезпечується модульність проекту, що сприяє легкому масштабуванню.

Фреймворк Symfony добре дбає про безпеку як користувачів, так і самої системи. Має вбудовані інструменти захисту від різних, поширених видів атак. Cross-Site Request Forgery (CSRF) [11], до форм автоматично додаються захисні токени, які перевіряються при відправці. SQL-ін'єкції [12]. Doctrine використовує параметризовані запити, що забезпечує захист від подібних атак. Cross-Site Scripting (XSS) [13]. шаблонізатор Twig, який використовується в фреймворку за замовчуванням, автоматично екранує зміни в HTML, завдяки чому, символи (наприклад < , > , &) перетворюються на безпечні текстові аналоги та інтерпретуються як текст, а не код.

Twig [14] – шаблонізатор, який відповідає за Фронтендну частину проекту. Він дозволяє динамічно виводити дані в виді HTML. Через пряму інтеграцію, шаблони легко підключити до контролерів, передати в них дані та збудувати згідно них сторінку. Twig дозволяє одному шаблону наслідувати інший. За допомогою цього, їх можна розбивати на окремі частини, які відповідатимуть за щось своє, а потім динамічно зібрати сторінку для користувача. Через це, зміна дизайну стає більш простою задачею, а код менше повторюється в проекті. В шаблонах Twig також можна виконувати JavaScript, або JQuery [15] код, щоб зробити інтерфейс динамічнішим. Підтримуються AJAX [16] запити, за допомогою яких, веб-сайт буде відчуватись безшовним, що покращить user experience. Для візуальної частини інтерфейсу використовується фреймворк

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		19

Bootstrap [17], який забезпечує адаптивну, зручну та сучасну верстку компонентів сторінки.

Symfony є потужним фреймворком, який орієнтований на розробника, допомагає зосередитись на логіці проекту, а не на технічних дрібницях базового функціоналу та рутинних справ. Фреймворк сприяє легкому масштабуванню проекту та має чітку структуру. Він є базою для багатьох інших фреймворків та систем, таких як Laravel та Dupal. Symfony активно підтримується спільнотою розробників, регулярно оновлюється, добре документований та використовується в багатьох успішних онлайн-продуктах. Це робить Symfony дуже привабливим вибором для даного проекту.

Microsoft SQL Server [18] – реляційна система управління базами даних, створена корпорацією Microsoft. Це одна з найбільш широко використовуваних систем, через велику кількість функцій, високий рівень надійності та можливість інтеграції її з іншими продуктами компанії-виробника. Основною причиною вибору MSSQL є висока стабільність і надійність при роботі з великими обсягами даних. Також вона підтримує використання складних запитів, створення представлень, процедур та тригерів. MSSQL має зручне графічне середовище, яке дозволяє проводити будь які дії з базою даних: SQL Server Management Studio (SSMS). Його можливості перевершують будь який подібний додаток для інших СУБД, які були розглянуті. За його допомогою можна зручно спроектувати базу даних, розставити відносини між таблицями, описати логіку роботи та автоматизувати процеси в базі даних. Завдяки цьому, створення та підтримка проекту спрощується. Ще одною перевагою SQL Server є безпека. СУБД підтримує гнучку систему прав доступу, шифрування даних та керування користувачами. Це дозволить захистити інформацію в базі даних від доступу до неї зловмисників, що є особливо важливо при роботі з особистими даними користувачів, як в даному випадку.

Таким чином, Microsoft SQL Server – це не просте сховище даних. Це потужна платформа для розробки складної логіки, яка гарантує надійність роботи та високий рівень безпеки. Вона має великий набір функцій та

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремюв Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		20

можливостей, за допомогою яких можна розробити найбільш підходяще рішення для будь якого проекту. Цій СУБД довіряють найвідоміші гравці на ринку інформаційних технологій, вона використовується як для менеджмента онлайн-порталів, так і для контролю внутрішньої логістики на великих підприємствах і навіть для роботи державних установ.

Висновки до першого розділу

Проведено аналіз предметної області, за результатами якого, були сформульовані основні вимоги для маркетплейсу комп'ютерної техніки та були виділені функціональні модулі, які повинні бути реалізовані. Основною задачею розроблюваного продукту є надання користувачам зручної платформи для продажу та купівлі комп'ютерної техніки.

Після аналізу аналогічних та схожих за концепцією продуктів, були порівняні різні платформи, що фокусуються на електронній комерції, визначені їх сильні та слабкі сторони. На основі цього, були сформовані основні вимоги до напрямку та функціоналу власної розробки.

На основі вимог та функціональних критеріїв, був сформований набір технологій та засобів реалізації для розробки веб-продукту.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		21

РОЗДІЛ 2 ПРОЕКТУВАННЯ МАРКЕТПЛЕЙСУ КОМП'ЮТЕРНОЇ ТЕХНІКИ

2.1 Визначення вимог та варіантів використання. Проектування структури системи

Визначення вимог користувачів.

Основні цілі. Система розробляється з метою створення зручного маркетплейсу для продажу та купівлі комп'ютерної техніки, що забезпечує ефективну взаємодію між покупцями, продавцями та адміністраторами. Представлення проекту. Проект буде реалізовано у вигляді вебсайту, який надаватиме можливості пошуку товарів, перегляду їх характеристик, оформлення замовлень та обробки оплати.

Вимоги користувачів. Можливість реєстрації та авторизації користувачів із розподілом на ролі: покупець, продавець, адміністратор. Можливість покупцям шукати, фільтрувати та переглядати товари з детальними характеристиками. Можливість додавання товарів у кошик, оформлення замовлень та здійснення оплат. Можливість продавцям додавати товари на платформу, редагувати їх інформацію та переглядати замовлення. Можливість залишати відгуки про товари, переглядати рейтинги продавців та товарів. Можливість адміністратору керувати категоріями товарів, модерацією контенту та обробкою скарг. Можливість покупцям відстежувати статус доставки замовлень.

Вимоги до функціональних характеристик.

Програмна система має забезпечувати: автентифікацію та авторизацію користувачів (реєстрація покупців та продавців, вхід у систему з використанням логіна та пароля); створення, редагування та видалення оголошень про продаж товарів; перегляд асортименту, сортування та фільтрацію товарів за різними параметрами (ціна, категорія, бренд); оформлення замовлень, обробку оплати та відстеження статусу доставки; можливість залишення відгуків про товари та продавців; підтримку спілкування між покупцями та продавцями через вбудований чат.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				22
Змн.	Арк.	№ докум.	Підпис	Дата		

Організація вхідних і вихідних даних.

Вхідними даними є: інформація про користувачів(реєстраційні дані та профілі), дані про товари(назва, опис, категорія, ціна...), повідомлення між користувачами у чатах, дані про замовлення. Вихідними даними є: Інформація про зареєстрованих користувачів(історія покупок чи продажів, активність), асортимент товарів, повідомлення та історія листування, звіти про замовлення та доставку включаючи фінансову інформацію.

Часові характеристики і розмір пам'яті, необхідної для роботи програми. Час реакції програми на дії користувача (натискання кнопок, запити до системи) не повинен перевищувати 1 секунди. Відображення великих масивів даних, наприклад, списків товарів за запитами користувача, має виконуватися не довше ніж 5 секунд. Час генерації звітів (аналітика продажів, рейтинги товарів тощо) не повинен перевищувати 2 хвилин. Доступність бази даних повинна забезпечуватися на рівні не менше ніж 99% часу цілодобово. Операції з'єднання із базою даних (включаючи запити на читання чи запис) мають виконуватися не довше ніж 30 секунд. Обсяг оперативної пам'яті, необхідний для роботи програми на клієнтському пристрої, повинен становити не менше 4 ГБ. Обсяг пам'яті на сервері для обробки запитів та зберігання бази даних залежить від кількості активних користувачів, але для базової конфігурації рекомендовано не менше 16 ГБ оперативної пам'яті.

Вимоги до надійності функціонування.

Програма повинна функціонувати стабільно за умови безперебійної роботи серверного обладнання та пристрою користувача. Доступність бази даних повинна бути не менше 99% при одночасному навантаженні в 5000 користувачів. У разі апаратних збоїв, повинно бути забезпечено відновлення Серверної частини: перезавантаження ОС сервера та автоматичний запуск сервера бази даних, використання резервного сервера за технологією failover [19], використання технологій RAID для збереження цілісності даних. Та відновлення Клієнтської частини - повторний запуск клієнтської програми. У разі програмних збоїв, система повинна: забезпечувати збереження та

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		23

відновлення даних користувачів [20], реалізовувати механізми фіксації транзакцій з можливістю «відкату» до попереднього стану у випадку помилок, підтримувати обробку виняткових ситуацій, зокрема некоректних введених даних[21] або перевантаження серверів, без припинення роботи всієї системи.

Контроль вхідної та вихідної інформації. Для забезпечення коректності вхідної інформації відбувається перевірка відповідності даних: формату цін, правильність введення емейлу чи телефону, та захист від помилок видаленням зайвих пробілів та обмеження введення некоректних символів. Для контролю вихідної інформації відбувається Форматування даних для коректності відображення та зручного перегляду надається можливість фільтрації та пошуку даних.

Умови експлуатації і збереження. Програма працює у багатокористувацькому середовищі з доступом до мережі Інтернет, де одночасно можуть бути зареєстровані покупці та продавці. Для надійного збереження даних адміністратори маркетплейсу регулярно створюють резервні копії (бекапи) бази даних.

Вимоги до інформаційної і програмної сумісності. Система має коректно функціонувати у сучасних браузерях (Opera, Chrome, Firefox, Edge тощо) та бути сумісною зі службами, які забезпечують підтримку веб-фреймворків і бібліотек.

Варіанти використання системи. Аналіз вимог користувачів дав підстави для визначення основних акторів та варіантів використання системи ними. Користувачі програми можуть бути розбиті на 3 окремі групи: покупці, продавці та адміністратори. Короткий опис акторів представлено в табл. 2.1. Варіанти використання наведено в табл. 2.2.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		24

Таблиця 2.1

Виявлення акторів

Актор	Короткий опис
Покупець	Користувач, який переглядає товари, оформлює замовлення та залишає відгуки про товари.
Продавець	Користувач, який додає товари на продаж, обробляє замовлення та спілкується з покупцями.
Адміністратор	Користувач, який керує системою, модерує контент, обробляє скарги та створює звіти.

Таблиця 2.2

Виявлення варіантів використання

Основний актор	Найменування	Формулювання
Покупець	Перегляд товарів	Дає можливість переглядати асортимент товарів у системі.
Покупець	Пошук та фільтрація товарів	Дає змогу здійснювати пошук товарів за назвою, категорією чи іншими параметрами та застосовувати фільтри.
Покупець	Перегляд історії замовлень	Дозволяє переглядати попередні замовлення користувача.
Покупець	Додавання товару до кошика	Дозволяє додати обраний товар до кошика для подальшого оформлення замовлення.
Покупець	Оформлення замовлення	Дає змогу здійснити замовлення товарів із вказанням даних доставки та оплати.
Покупець	Реєстрація та вхід у особистий кабінет	Дозволяє створити обліковий запис або авторизуватись у системі.
Покупець	Перегляд та редагування особистих даних	Дозволяє переглядати та змінювати дані профілю користувача.
Покупець	Залишити відгук про товар	Дозволяє залишити текстовий або рейтинговий відгук про придбаний товар.
Покупець	Зв'язок з продавцем	Дає змогу зв'язатися з продавцем через чат для уточнення деталей товару.
Продавець	Додавання товарів на продаж	Дозволяє створювати оголошення про продаж товарів.
Продавець	Редагування інформації про товар	Дає змогу змінювати деталі товару після його додавання.
Продавець	Видалення товару з продажу	Дозволяє зняти товар із продажу, якщо він більше недоступний.
Продавець	Перегляд замовлень	Дозволяє переглядати список замовлень на товари продавця.
Продавець	Обробка замовлень	Дає змогу підтверджувати замовлення та змінювати їх статуси.
Продавець	Управління особистими даними	Дозволяє переглядати та змінювати профіль продавця.

Продовження таблиці 2.2.

Продавець	Управління особистими даними	Дозволяє переглядати та змінювати профіль продавця.
Продавець	Управління статусом товарів	Дозволяє змінювати доступність товарів (в наявності/відсутній).
Адміністратор	Управління користувачами	Дозволяє додавати, редагувати або видаляти облікові записи користувачів.
Адміністратор	Перегляд та редагування даних користувачів	Дозволяє переглядати дані профілів користувачів і вносити зміни за потреби.
Адміністратор	Перегляд та обробка скарг	Дає змогу розглядати скарги від користувачів і приймати відповідні заходи.
Адміністратор	Модерація контенту	Дозволяє перевіряти товари, відгуки та інший контент на відповідність політикам платформи.
Адміністратор	Редагування та видалення товарів	Дозволяє змінювати або видаляти товари у разі порушення правил.
Адміністратор	Моніторинг роботи системи	Забезпечує відстеження технічного стану платформи та її компонентів.
Адміністратор	Створення та редагування категорій товарів	Дозволяє створювати або змінювати категорії для зручності роботи користувачів.
Адміністратор	Статистична звітність	Дає змогу формувати звіти про активність користувачів, продажі та інші показники.

Діаграма, яка наглядно показує варіанти використання показана на рис. 2.1

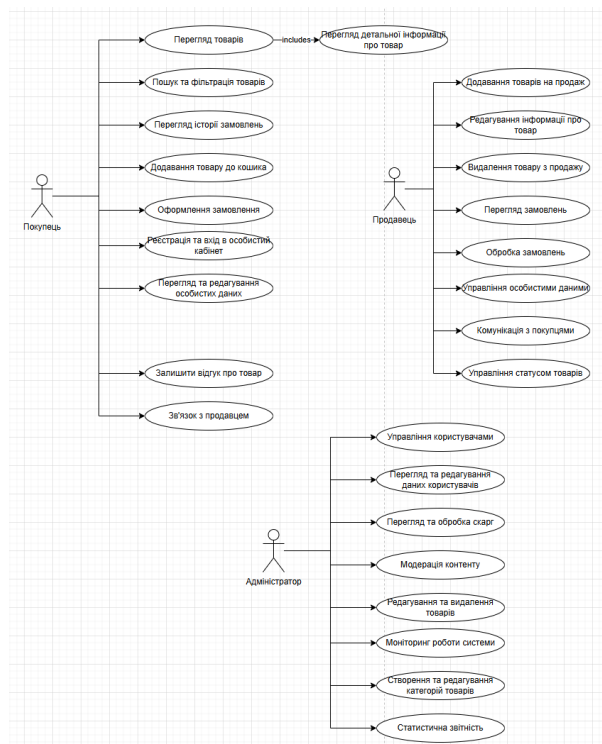


Рис. 2.1. Діаграма варіантів використання

Аналіз функціональних вимог дозволив виділити наступні сутності, що забезпечать програмну реалізацію додатку. На рис. 2.2 наведено Діаграму класів, яка використовуються для його роботи.

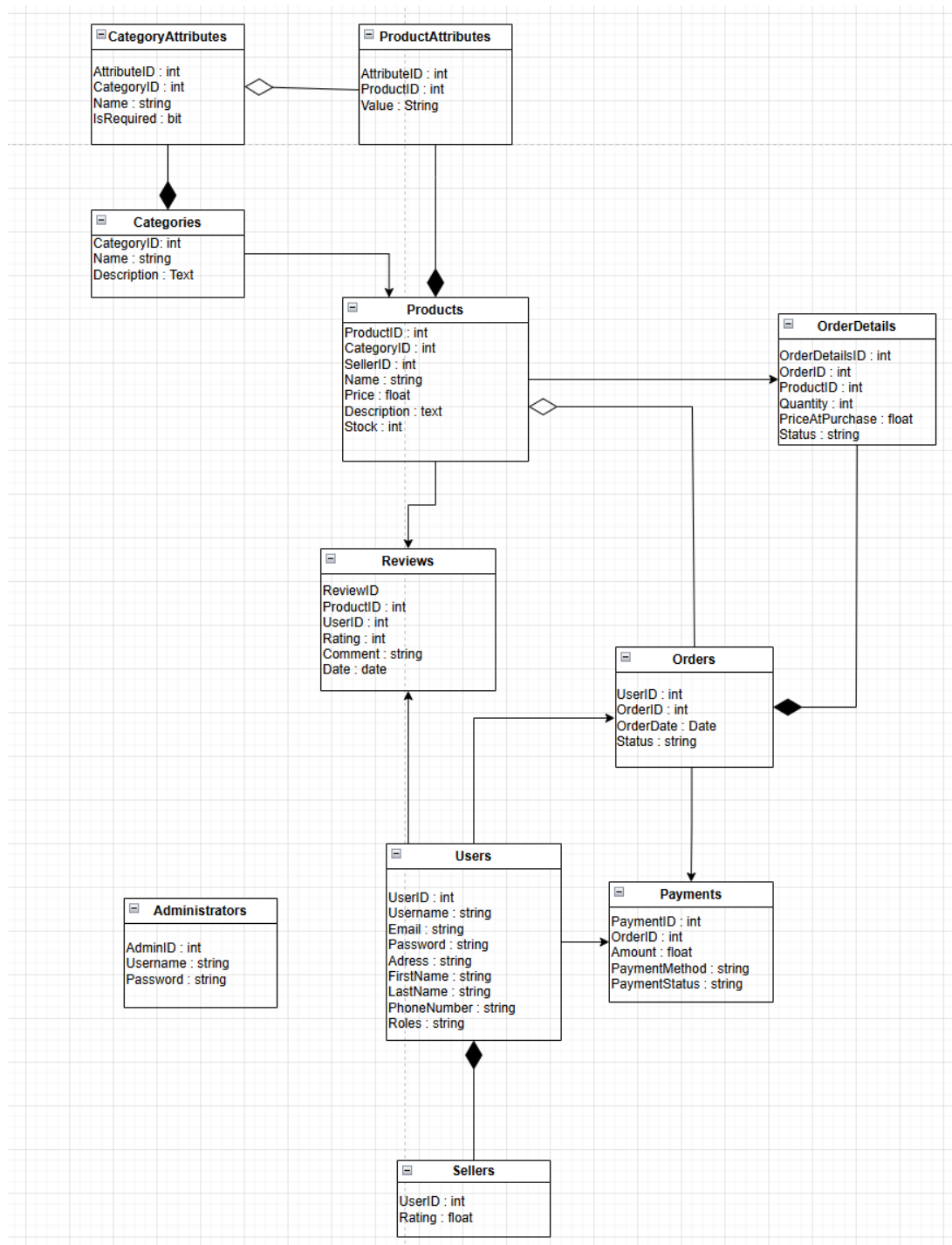


Рис. 2.2. Діаграма класів

Users – відповідає за збереження персональних даних користувачів.

Administrators – окрема сутність для адміністратора, зберігає мінімальну кількість даних.

Categories – Зберігає категорії товарів. З цією сутністю пов’язані атрибути, які характерні саме для цієї категорії.

CategoryAttributes – Містить набір атрибутів, характерних для певної категорії, з якою пов’язана сутність.

Products – Представляє окремі товари. Зберігає основну інформацію про них та зв’язана з Categories та Sellers.

ProductAttributes – Зберігає конкретні значення атрибутів для кожного товару. Пов’язаний з Products.

Orders – Зберігає інформацію про замовлення. Пов’язане з Users, які їх створюють. Не містить самих продуктів.

OrderDetails – Представляє окрему позицію в замовленні. Пов’язане з Orders та Users.

Payments – Містить інформацію про оплату замовлення. Пов’язане з Orders.

Reviews – Зберігає відгуки від користувачів на товари. Зв’язки з Products та Users.

2.2 Розробка фізичної моделі бази даних системи.

Для збереження файлів та інформації використовується система управління базами даних Microsoft SQL Server, або MSSQL. Ця система призначення для створення та управління саме реляційними базами даних.

База даних системи складається з дванадцяти таблиць:

1. Administrators - зберігає дані для автентифікації адміністраторів.
2. Categories – зберігає список категорій товарів.
3. CategoryAttributes – зберігає список характеристик, які повинні бути зазначені в продукті, відповідно до його категорії.
4. OrderDetails – зберігає інформацію про частини замовлення – окремі продукти в ньому.
5. Orders – зберігає загальну інформацію про замовлення.
6. Payments – зберігає інформацію про оплату замовлення.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		28

7. ProductAttributes – для кожного продукту зберігає характеристики, зазначені в таблиці CategoryAttributes.
8. Products – зберігає інформацію про товари.
9. Reviews – зберігає відгуки користувачів стосовно якогось товару.
10. Sellers – в цій таблиці зберігаються користувачі, які мають можливість продавати товар.
11. Shipping – зберігає інформацію про доставку товару.
12. Users – зберігає інформацію про користувачів.

Детальний опис всіх таблиць бази даних наведено в таблицях 2.3-2.15.

Таблиця 2.3

Опис таблиці Administrators

Обмеження	Поле	Тип даних	Призначення
PK	AdminID	int	Ідентифікатор
	Username	varchar	Ім'я користувача
	Password	varchar	Пароль користувача

Таблиця 2.4

Опис Таблиці Categories

Обмеження	Поле	Тип даних	Призначення
PK	CategoryID	int	Ідентифікатор
	Name	varchar	Назва категорії
	Description	text	Опис категорії

Таблиця 2.5

Опис таблиці CategoryAttributes

Обмеження	Поле	Тип даних	Призначення
PK	AttributeID	int	Ідентифікатор
FK	CategoryID	int	ID категорії, для якої зберігається атрибут
	Name	varchar	Назва атрибуту
	DataType	varchar	Тип даних
	IsRequired	bit	Чи необхідний атрибут для заповнення

Таблиця 2.6

Опис таблиці OrderDetails

Обмеження	Поле	Тип даних	Призначення
PK	OrderDetailsID	int	Ідентифікатор
FK	OrderID	int	ID замовлення, до якого належить ця частина
FK	ProductID	int	ID продукту, що входить в цю частину
	Quantity	int	Кількість замовленого продукту
	PriceAtPurchase	float	Ціна в момент замовлення
	Status	varchar	Статус виконання цієї частини

Таблиця 2.7

Опис таблиці Orders

Обмеження	Поле	Тип даних	Призначення
PK	OrderID	int	Ідентифікатор
FK	BuyerID	int	ID користувача, хто замовив
	OrderDate	int	Дата замовлення
	Status	varchar	Статус виконання всього замовлення

Таблиця 2.8

Опис таблиці Payments

Обмеження	Поле	Тип даних	Призначення
PK	PaymentID	int	Ідентифікатор
FK	OrderID	int	ID замовлення, якого стосується оплата
	Amount	float	Кількість грошей, яку треба сплатити
	PaymentMethod	varchar	Метод оплати
	PaymentStatus	varchar	Статус оплати

Таблиця 2.9

Опис таблиці ProductAttributes

Обмеження	Поле	Тип даних	Призначення
PK, FK	ProductID	int	Ідентифікатор, ID продукту, якому належить атрибут
PK, FK	AttributeID	int	Ідентифікатор, ID атрибуту, взятого з CategoryAttributes
	Value	varchar	Значення атрибуту

Таблиця 2.10

Опис таблиці Products

Обмеження	Поле	Тип даних	Призначення
PK	ProductID	int	Ідентифікатор
FK	CategoryID	int	ID категорії, до якої належить товар
FK	SellerID	int	ID продавця, який продає товар
	Name	varchar	Назва товару
	Price	float	Ціна товару
	Description	text	Опис товару
	Stock	int	Заявлена кількість товару на складі

Таблиця 2.11

Опис таблиці Reviews

Обмеження	Поле	Тип даних	Призначення
PK	ReviewID	int	Ідентифікатор
FK	ProductID	int	ID продукту, на який залишений відгук
FK	UserID	int	ID користувача, що залишив відгук
	Rating	int	Оцінка продукту
	Comment	text	Коментар до відгуку
	Date	date	Дата написання відгуку

Таблиця 2.12

Опис таблиці Sellers

Обмеження	Поле	Тип даних	Призначення
PK, FK	UserID	int	Ідентифікатор
	Rating	float	Рейтинг продавця

Таблиця 2.13

Опис таблиці Sipping

Обмеження	Поле	Тип даних	Призначення
PK	OrderID	int	Ідентифікатор
	TrackingNumber	varchar	Номер для відстеження посилки
	DeliveryDate	date	Дата доставки
	ShippingStatus	varchar	Статус виконання доставки
	DeliveryAddress	varchar	Адреса, на яку потрібно доставити

Таблиця 2.13

Опис таблиці Users

Обмеження	Поле	Тип даних	Призначення
PK	UserID	int	Ідентифікатор
	Username	varchar	Ім'я користувача
	Email	varchar	Електронна пошта користувача
	Password	varchar	Пароль користувача
	FirstName	varchar	Власне, справжнє ім'я користувача
	LastName	varchar	Фамілія користувача
	PhoneNumber	varchar	Телефонний номер користувача
	Address	text	Адреса користувача
	Roles	varchar	Ролі користувача для доступу до сайту

Окрім вище описаних таблиць, в базі даних також будуть створені деякі представлення, збережені процедури та тригери. Це необхідно як для часткової автоматизації системи, так і для зручності користування та масштабування. Представлення: SellerView буде зберігати в собі всіх користувачів, які є або продавцями(Sellers). ProductReviewsView – для продуктів та відгуків на них. Додамо процедури для зручного додавання даних в таблиці: Orders, OrderDetails, Reviews і т.д. CheckSellerBeforeInsertProduct тригер: Перевірятиме чи ID продавця, яке ми намагаємось передати в Product, є в представленні SellerView. Якщо нема - то користувач не є продавцем і не може продати товар. PreventDuplicateReviews тригер: Не дозволить залишити відгук на товар, який користувач вже оцінював. UpdateTotalAmount тригер: Після змін в таблиці OrderDetails, перераховує всю ціну замовлення. UpdateSellerRating тригер: Після змін в таблиці Reviews, перераховує середній рейтинг продавця. AutoCreatePayment тригер: після створення замовлення створює новий запис в таблиці Payments.

2.3 Проектування та реалізація алгоритмів роботи системи.

Діаграма активності, зображена на рис. 2.3, Ілюструє очікувану послідовність дій у системі маркетплейсу. Також вона показує взаємодію основних ролей системи – між покупцями та продавцями, а також самої системи.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		32

Взаємодія показана на різних етапах, від авторизації на сайті до виходу з нього. Також діаграма поділена на три частини, щоб наглядно показати хто за що відповідає.

Процес розпочинається одночасно для покупця та продавця з дії авторизації. Користувач вводить свої облікові дані, які перевіряються системою. Якщо дані коректні, користувач отримує доступ до свого облікового запису. Після авторизації покупець переходить до пошуку товару в базі даних за допомогою фільтрів або ключових слів. Система формує список знайдених товарів, і покупець вибирає конкретний товар для перегляду детальної інформації. Якщо вибір зроблено, товар додається до кошика. Далі покупець переходить до оформлення замовлення, заповнюючи форму із зазначенням контактних даних та адреси доставки. Система перевіряє правильність введених даних, і, якщо дані некоректні, покупець отримує можливість виправити їх. У разі успішного заповнення даних покупець переходить до етапу оплати. На етапі оплати система перенаправляє покупця до платіжного сервісу. Якщо транзакція завершується успішно, кошти зараховуються на рахунок компанії, і система повертає покупця на головну сторінку. Після авторизації продавець отримує доступ до свого облікового запису, де може додавати товари на продаж. Цей процес включає введення інформації про товар, яку система зберігає у базі даних. У разі отримання замовлення продавець отримує сповіщення та перевіряє деталі. Замовлення може бути підтверджене або відхилене залежно від наявності товару. У разі відмови система повертає кошти покупцю, а в разі підтвердження — розпочинається процес підготовки до відправлення замовлення. Система виступає посередником між покупцем і продавцем. Вона обробляє всі дії користувачів, зокрема пошук товарів, обробку замовлень, перевірку даних і оплати. У разі успішного завершення оплати система передає інформацію про замовлення продавцеві. Після підтвердження замовлення продавцем покупець отримує сповіщення про його стан.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		33

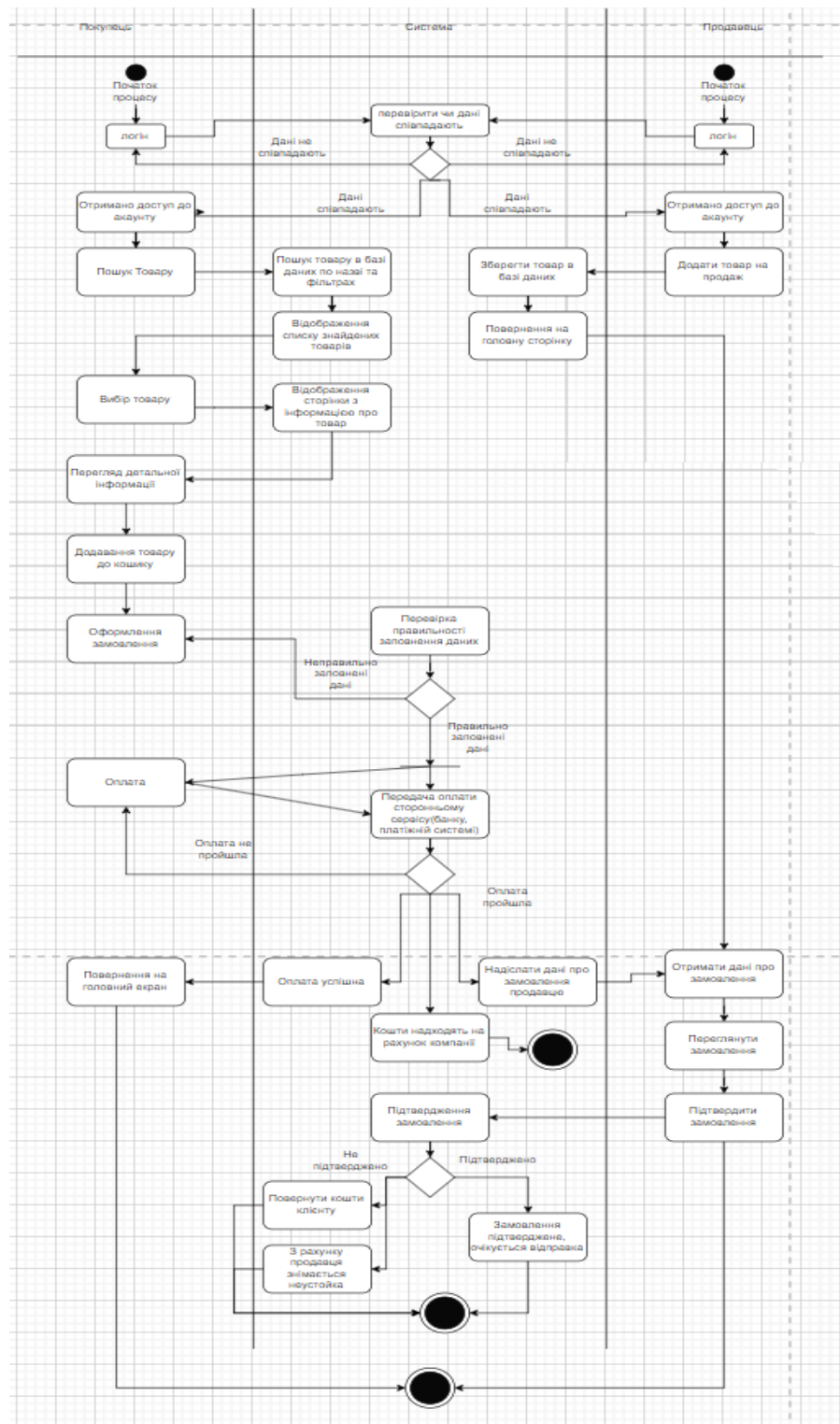


Рис. 2.3 Діаграма активності системи

Діаграма компонентів, рис. 2.4, ілюструє архітектуру системи маркетплейсу, демонструючи, як основні частини системи взаємодіють між собою. Вона забезпечує зрозуміле уявлення про розподіл функціональності між фронтендом, бекендом і зовнішніми сервісами. Основним елементом взаємодії користувачів із системою є Web Browser, який через HTTPS протокол передає

запити до Frontend. Фронтенд складається з двох основних компонентів: Website Layout, який відповідає за структуру та дизайн сайту, та User Interface, який забезпечує взаємодію користувача з системою. На рівні бекенду запити обробляються через контролери, зокрема: User Controller, який взаємодіє з User Service для перевірки автентифікації та управління користувачами. Product Controller, який забезпечує доступ до даних про товари, отримуючи інформацію з Product Service і Database. Order Controller, який керує створенням і обробкою замовлень через Order Service. Database є основним джерелом зберігання даних про користувачів, товари, замовлення та інші сутності. У свою чергу, External Services забезпечують інтеграцію із зовнішніми API, такими як платіжні системи або служби доставки. Бекенд структуровано таким чином, щоб кожен компонент виконував чітко визначені завдання: User Service обробляє автентифікацію та запити користувачів. Product Service відповідає за доступ до даних про товари та їхню актуалізацію. Order Service виконує обробку замовлень, включаючи перевірку статусу та зв'язок із зовнішніми системами.

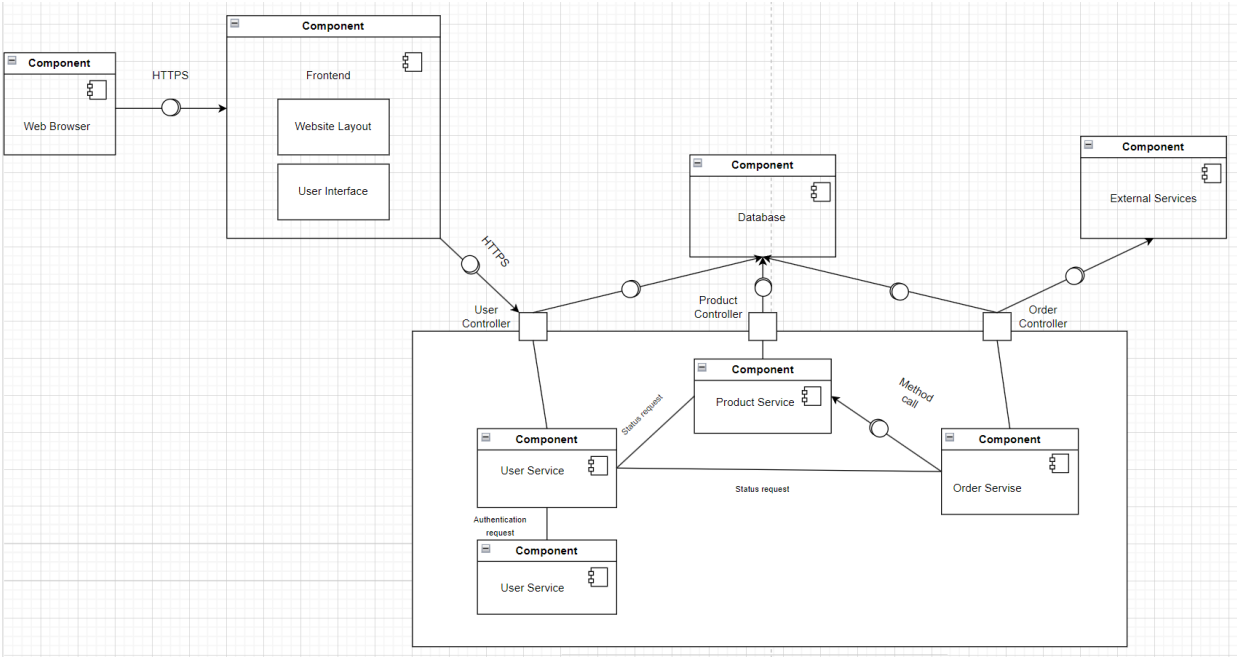


Рис. 2.4. Діаграма компонентів

2.4 Реалізація маркетплейсу комп’ютерної техніки.

Спочатку треба підготуватись до роботи з базою даних. Для цього в Symfony інтегровано Doctrine ORM. Через сутності, які ми створимо, вона

синхронізується з існуючою базою даних. Це дозволить легко з нею взаємодіяти. Щоб добитись правильної синхронізації, треба розписати поля кожну таблицю БД, з всіма її полями в окремі сутності (Entity). Doctrine має свій синтаксис, якщо можна так виразитись, яким специфікуються: назва поля, тип даних, відношення до других сутностей тощо. Далі, на рис. 2.5, буде наданий приклад опису сутності Categories в ORM Doctrine (гетери та сетери наведені не будуть. Вони потрібні для правильної роботи деяких методів Symfony та не впливають на синхронізацію з БД).

```
#[ORM\Entity]
#[ORM\Table(name: 'Categories')]
class Categories
{
    #[ORM\Id]
    #[ORM\GeneratedValue]
    #[Column(name: "CategoryID", type: Types::INTEGER)]
    private int $CategoryID;

    #[ORM\Column(type: 'string', length: 100, nullable: true)]
    private ?string $Name = null;

    #[ORM\Column(type: 'text', nullable: true)]
    private ?string $Description = null;
}
```

Рис. 2.5. Приклад створення сутності в Doctrine

Коли всі сутності будуть прописані, можна легко виконувати CRUD операції. Для початку, користувачі повинні зареєструватись та увійти на веб-сайт, щоб отримати доступ до його функціоналу. Тому модуль реєстрації та автентифікації буде першим. Зробимо контролер SecurityController, який буде відповідати за логіку. В метод реєстрації підготуємо форму та передасть її на новостворений шаблон – сторінку security/register.html.twig. При спробі надсилання форми на сервер, відбудеться автоматична перевірка правильності заповнення її полів. Якщо перевірка буде пройдена, інформація з форми перенесеться до бази даних. Новий користувач буде створений. Це все досягається за допомоги EntityManager[22], який взаємодіє з базою даних через

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		36

сутності в ORM. З логіном все простіше. Після недовгого налаштування security.yaml файлу, Symfony сам перевіряє введені користувачем дані на відповідність БД, при успіху сам створює сесію для користувача [23]. Вихід з акаунту також перехоплюється Symfony автоматично. Залишається лише створити форму логіну.

Модуль управління профілем користувача буде логічним продовженням. Створена сама сторінка профілю, яка відображає дані користувача. Контролер перенаправляє на неї. Далі була створена можливість змінити дані користувача при натисканні на кнопку. Зміни можна зберегти в бд при натисканні на кнопку. Звичайно, до редагування профілю може бути допущений тільки користувач, чий цей профіль і є.

Модуль управління товарами. Спочатку треба реалізувати створення нових товарів. З контролері також передається форма, як в попереднім модулях. В цій формі будуть також характеристики продукту, які необхідно буде заповнити (при виборі категорії, з'являться відповідні поля). Та окрім текстових даних, можна зберігати також картинки – фотографії товарів [24]. Вони зберігаються не в базі даних, а в файлах самого веб-сайту. В директорії public/uploads/products створюються папки для кожного товару, називаються за ID ідентифікатором, тому не повторяться і їх легко достати. Буде два види картинок – main, яка буде відображатись на картках товару, та інші фото. Вони будуть нумеруватись починаючи з 2, та збільшуватись при кожній доданій картинці. Коли товар було створено, його можна відредагувати. Картинки теж можна буде відредагувати: замінити головну картинку, додати нові додаткові, або видалити ті, що непотрібні. Для створеного товару треба сторінка. Контролер буде передавати інформацію про певний товар на сторінку в залежності від ID шуканого товару.

Каталог товарів. Створено сторінку, яка показує всі товари. Імена, картинки та ціна товарів підтягується, при натисканні на карточку товару, перекидує на його сторінку. Також реалізована фільтрація за ціною та категоріями і пошук за назвою. Це зроблено за допомогою queryBuilder, який

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		37

бере дані з полів форми та динамічно створює SQL запит, що бере до уваги ці поля [25].

На товари можна залишати відгуки, тому далі зробимо модуль відгуків. На сторінці товару та на його картці відображається поточна середня оцінка. При натисканні на неї, можна перейти на сторінку «залишити відгук». Користувач обирає оцінку від 1 до 5 (тільки цілі числа) та може написати коментар відгуку. Це зберігається в базі даних. Також створена сторінка для відображення відгуків від різних користувачів. Продавці не можуть залишати відгуки на свій товар за зрозумілою причиною.

Модуль кошику. Зі сторінки товару, його можна додати до кошику при натисканні на велику зелену кнопку. Товари в кошику зберігаються в сесії користувача, а не в базі даних. Продавець не може додати до кошика свій власний товар. Буде створена сторінка кошика, через яку можна буде міняти кількість певного товару в ньому, бачити підрахунок вартості всіх товарів у кошику.

Для покращення зручності сайту, було імплементовано модуль рекомендованих товарів. Він підбирає підходящі комп'ютерні компоненти на основі змісту кошику, дивлячись на товари з підходящими характеристиками (наприклад якщо в процесора сокет АМ-4, шукає материнські плати з таким же сокетом). Результати також показуються на сторінці кошику. Приклад програмного коду наведено в додатку А.

Модуль розрахунку та керування замовленнями. З кошика можна перейти на сторінку розрахунку. При введенні платіжної інформації створюється замовлення Orders: кожне найменування в корзині записується в новий OrderDetails, з яким взаємодіятиме продавець. Частини замовлення прив'язуються до новоствореного Order. Замовлення буде очікувати зміни статусу, а поки, за ним можна слідкувати через сторінку myOrders, яка показує всі замовлення. При натискання на окреме замовлення можна побачити всі його складові та їх статус. Продавці також мають подібну сторінку – змовлення, які були зроблені у них. Але продавці будуть бачити тільки пов'язані з ними частини

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				38
Змн.	Арк.	№ докум.	Підпис	Дата		

великого замовлення, а не його саме. Продавці мають можливість вручну міняти статус частини замовлення через ту саму сторінку.

Основний функціонал було імплементовано. Додамо також обробку помилок, щоб можна було показувати користувачу кастомні сторінки 404, 403 і так далі. Створимо EventListener для моніторингу статус коду [26]. При отриманні 404 чи 403 помилки, перенаправляємо на відповідну сторінку сайту. Але щоб воно працювало, потрібно зареєструвати їх як служби в services.yaml.

Висновки до другого розділу

В процесі проектування маркетплейсу комп'ютерної техніки, були проаналізовані всі варіанти використання продукту. На основі отриманого результату, була спроектована структура проекту та взаємодії між його частинами. Було спроектовано базу даних для зберігання всієї необхідної для роботи системи інформації: даних про користувачів, товари, замовлення, тощо. Було описано алгоритм роботи з додатком, описані послідовні пункти варіантів його використання. Також було наведено опис основного функціоналу веб-сайту – авторизація користувачів, управління товарами; перегляд, пошук та фільтрація продуктів; кошик та замовлення.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		39

РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З МАТРКЕТПЛЕЙСОМ КОМП'ЮТЕРНОЇ ТЕХНІКИ

3.1 Порядок встановлення та налаштування параметрів платформи

Для розгортання системи треба підготувати серверне середовище, розгонути базу даних та налаштувати середовище Symfony.

Для опису архітектури платформи, на рис. 3.1 показано діаграму розгортання.

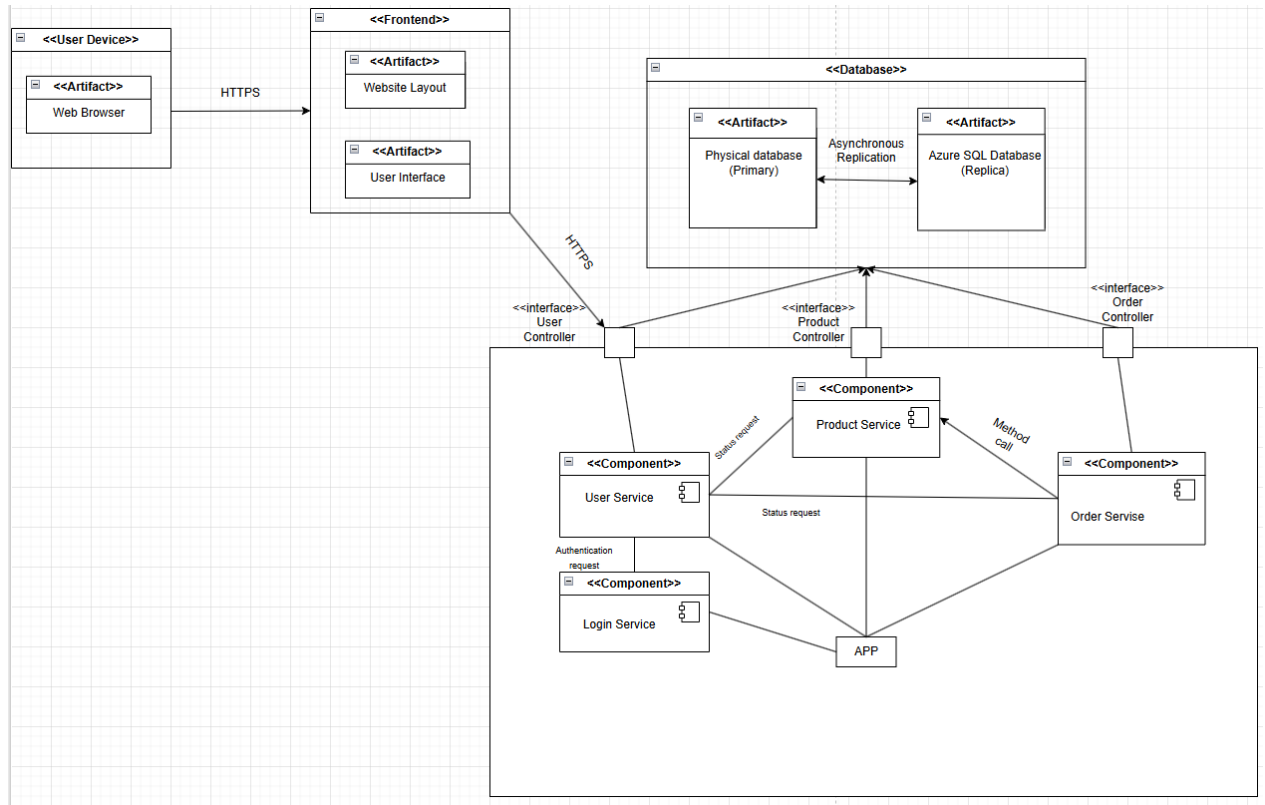


Рис. 3.1 Діаграма розгортання маркетплейсу

UserDevice – містить веб-браузер на пристрої користувача, через який передаються запити до серверу. Frontend містить макети сайту, інтерфейс користувача. Отримує дані від бекенду та відображає їх.

Серверна частина застосунку APP відповідає за основний функціонал. Вона складається з компонентів: User Service – обробка користувацьких запитів, Login Service – реєстрація та автентифікація, Product Service – операції з товарами, Order Service – створення та перегляд замовлень. База даних Database зберігає структуровані дані.

3.2 Структура інтерфейсу

Взаємодія користувача з додатком відбувається через веб-сайт. На головній сторінці, користувач бачить 10 рекомендованих товарів. Збоку список категорій, які можуть його зацікавити.

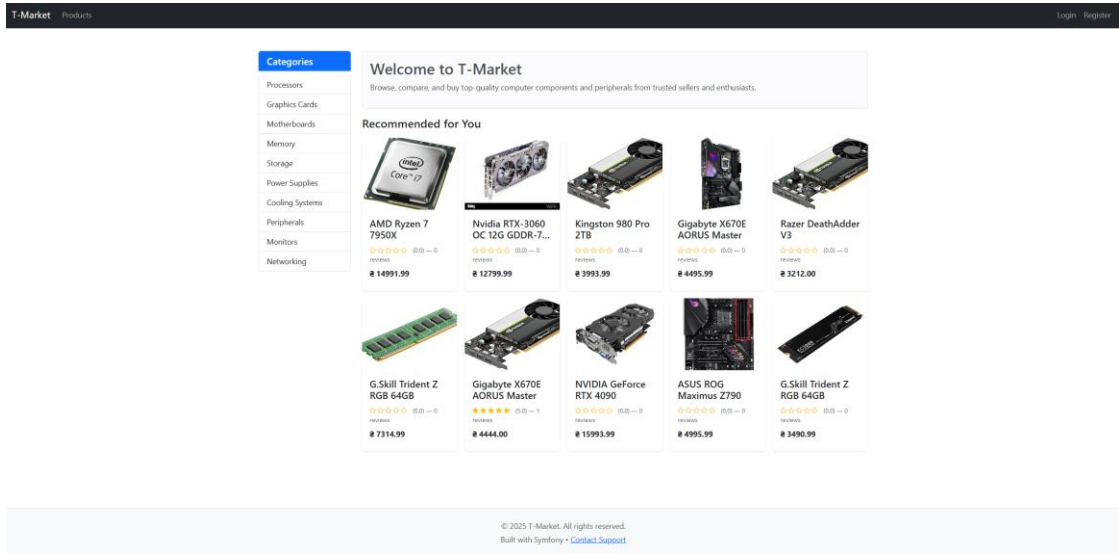


Рис. 3.2 Головна сторінка

Незареєстрований користувач може тільки переглядати товари. Для доступу до всіх функцій маркетплейсу, потрібно зареєструватись. На рис. 3.3 показана форма реєстрації, рис. 3.4 – форму авторизації.

Register

Username

Email

Password

First name

Last name

Phone number

Address

Register

Рис. 3.3. Форма реєстрації

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		41

Please sign in

Username

Password

☐ Remember me

Sign in

Рис. 3.4 Форма авторизації

Після реєстрації, користувач може перейти на сторінку каталогу: рис. 3.5.

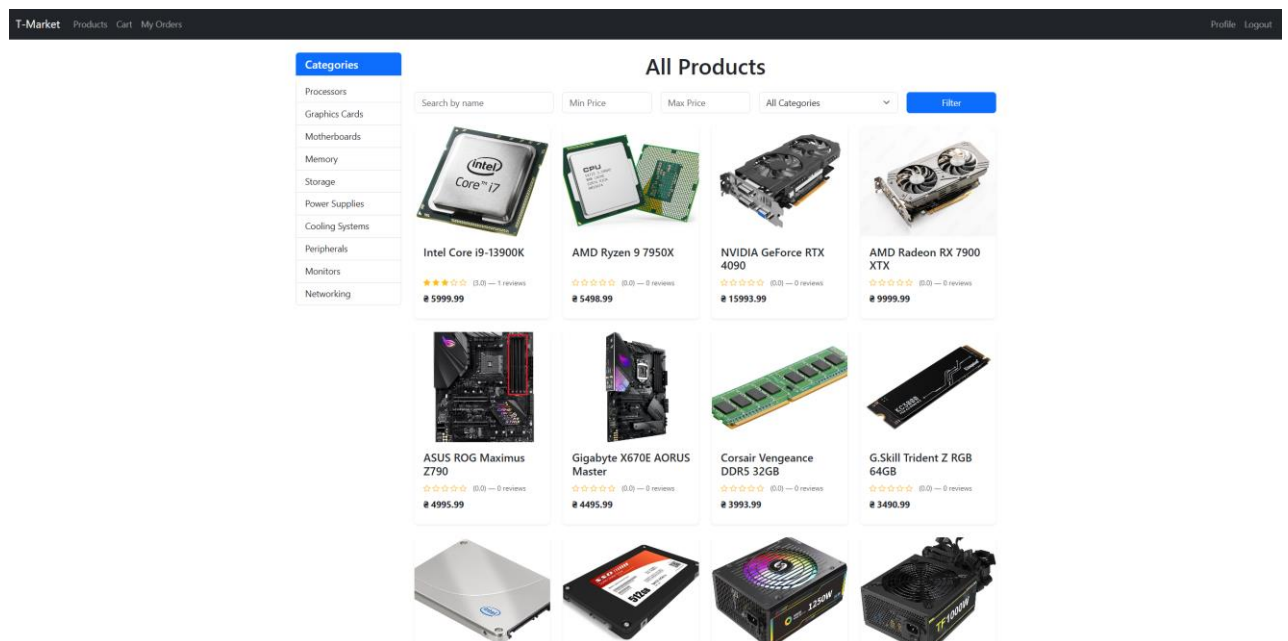


Рис. 3.5 Каталог товарів

На цій сторінці, користувач може також фільтрувати товари за ціною, категорією (рис. 3.6) та шукати їх за назвою (рис. 3.7).

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		42

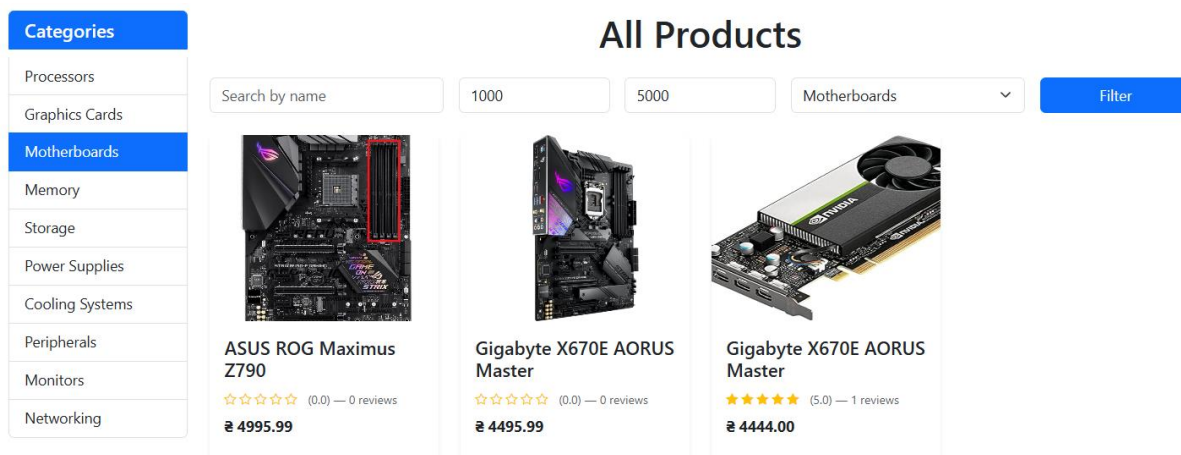


Рис. 3.6. Фільтри

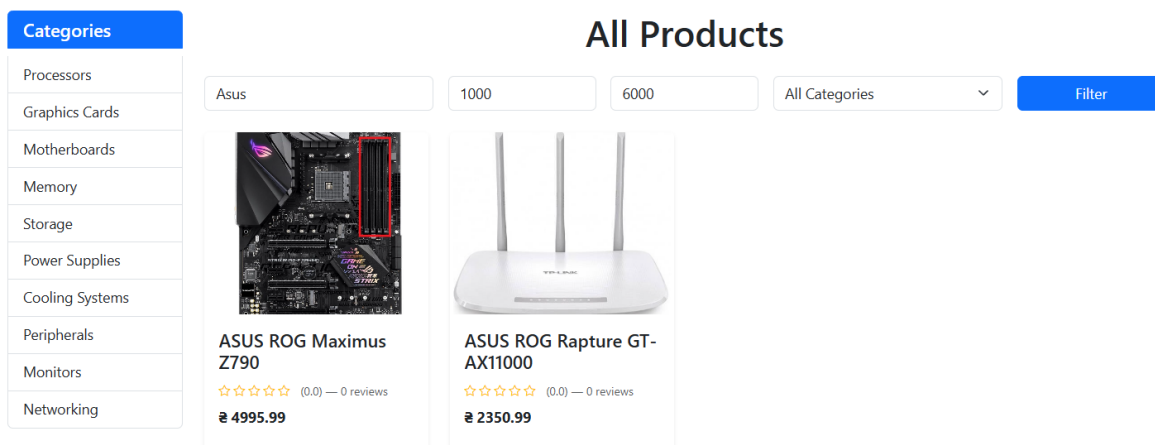


Рис. 3.7 Пошук по імені.

При натисканні на картинку товару, користувач переходить на його сторінку (рис. 3.8).

Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card

☆☆☆☆☆ (0) — 0 reviews

[View Reviews](#)

Seller: Sellerman39

Price: 12799.99 UAH

Description: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Specifications

Core Clock	3.7 GHz
GPU Model	Nvidia RTX-3060
Interface Type	Pci-e 4.0
Memory Type	GDDR-7
Video Memory (VRAM)	12 Gb
Wattage	180

Stock:



Add to Cart

Рис. 3.8 Сторінка товару

При натисканні на зелену кнопку, товар додається до корзини, рис. 3.9, а користувача перенаправляє на неї. Кількість товарів можна зменшити, корзину очистити, або перейти до оформлення замовлення. На основі товарів в кошику, знизу сторінки буде показано рекомендовані товари.

Your Shopping Cart

Product	Quantity	Actions
Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card	1	- +

Proceed to Checkout

Clear Cart

Рис. 3.9 Кошик.

Якщо додати до кошика ще і процесор, програма зможе рекомендувати блок живлення, який задовольняє вимоги потужності відеокарти та процесора, а також материнську плату, яка підійде до обраного процесора (рис. 3.10).

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		44

Your Shopping Cart

Product	Quantity	Actions
Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card	1	- +
Ryzen 5 5600x	1	- +

Proceed to Checkout

Clear Cart

Motherboards



Gigabyte X670E AORUS Master
4444.00

View Details



MSI Rogue A-320
5000.00

View Details

Power Supplies



PSU Product X32
3212.00

View Details

Рис. 3.10. Кошик з рекомендаціями

Сторінка оформлення замовлення виглядає наступним чином (рис. 3.11).

Review Your Order

Product	Quantity	Price	Subtotal
Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card	1	12799.99	12799.99
Ryzen 5 5600x	1	6500.00	6500

Total: 19299.99

Payment Method

Choose a method:

Cash on Delivery

Confirm and Place Order

Рис. 3.11 Оформлення замовлення

Після замовлення товару, можна залишити відгук про нього (натиснувши на рейтинг на сторінці товару). рис. 3.12.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		45

Leave a Review for "Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card"

Rating

☒

☐

☐

☐

☐

☐

★

★

★

★

★

★

Your Review

>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Submit Review

Рис. 3.12. Залишити рейтинг

Можна також переглянути всі відгуки з коментарями. Сторінка відгуків на рис. 3.13.

Reviews for "Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card"

User Rating: 4/5

Based on 3 reviews

5

★

2

4

★

0

3

★

0

2

★

1

1

★

0

Leave a Review

regularshopper17

2025-06-12

★★★★★

>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

testinguser

2025-06-12

★★★★☆

Bad product Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore

fatherof2

2025-06-12

★★★★★

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

Рис. 3.13. Сторінка відгуків

Подивимось на можливості продавця. Для початку, можна додати новий продукт. В залежності від категорії, необхідні характеристики будуть мінятись. Можна також вибрати фото. Форма продемонстрована на рис. 3.14.

Create New Product

Name

The new product RAM 9292

Description

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

Price

3212.00

Category

Power Supplies

Wattage

800

Efficiency Rating

Gold

Modular Design

No

Form Factor

Full-Size

Number of SATA Connectors

6

Main Photo

Выберите файл 713KdNRoKaL_AC_UF350,350_QL80_.jpg

Additional Photos

Выбрать файлы Число файлов: 3

Create Product

Рис. 3.14. Форма створення товару.

Продавець може редагувати дані товару та управляти фотографіями до нього. Під кожним фото є можливість обрати іншу фотографію та видалити її. Зміни зберігаються при натисканні на кнопку Replace. Можна також додати нові картинки. Ця сторінка показана на рис. 3.15.

Manage Photos for "The new product RAM 9292"

Выберите файл Файл не выбран Delete

Выберите файл Файл не выбран Delete

Выберите файл Файл не выбран Delete

Выберите файл Файл не выбран Main photo

Replace Selected Images

Add New Additional Images

Выбрать файлы Файл не выбран Upload New Images

Рис. 3.15. Сторінка управління фотографіями

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		47

У продавця також є доступ до сторінки, де показані товари, що у нього замовлені (рис. 3.16). Він може міняти статус замовлення, динамічно фільтрувати вивід таблиці по статусу. На рис. 3.17 показана зміна статусу, на рис. 3.18 фільтрація.

Orders for Your Products

☐ Show only active (Pending) orders

Product	Buyer	Quantity	Price	Status	Order Date
Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card	regularshopper17	1	12799.99	Pending	2025-06-12
Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card	testinguser	2	12799.99	Pending	2025-06-12
Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card	fatherof2	1	12799.99	Pending	2025-06-12

Рис. 3.16. Сторінка управління замовленнями

Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card	fatherof2	1	12799.99	Pending	2025-06-12
--	-----------	---	----------	---------	------------

Pending

Pending

Completed

Canceled

Рис. 3.17. Зміна статусу замовлення

☒ Show only active (Pending) orders

Product	Buyer	Quantity	Price	Status	Order Date
Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card	regularshopper17	1	12799.99	Pending	2025-06-12
Nvidia RTX-3060 OC 12G GDDR-7 Gaming Graphics Card	testinguser	2	12799.99	Pending	2025-06-12

Рис. 3.18. Фільтрація

Всі користувачі можуть моніторити свої замовлення, та статус їх виконання (рис. 3.19).

My Orders

Order #	Date	Status	Details
11	2025-06-12	Pending	View
2	2025-06-05	Completed	View
3	2025-06-05	Pending	View

Рис. 3.19. Замовлення користувача

Можна також подивитись на деталі замовлення, тобто на кожний окремий товар. (Рис. 3.20).

Order #3

Status: Pending

Order Date: 2025-06-05

Product	Quantity	Price at Purchase	Subtotal
Samsung Odyssey G9	3	444.00 UAH	1,332.00 UAH
Razer DeathAdder V3	2	3,212.00 UAH	6,424.00 UAH
Total:			7,756.00 UAH

Рис. 3.20. Окремі товари з замовлення.

Користувачі мають можливість переглядати свій профіль. В ньому є їх інформація, посилання на замовлення: від них та до них. Також у продавців показано список товарів, які вони додали. Сторінка показана на рис. 3.21.

Profile

Edit Profile

Account Details

Username: testinguser

Email: afdskfsd@gjnsq

First Name: dfghfgdh

Last Name: fdghdfgh

Phone: fdghfgdh

Address: fdghfdgh

My Orders

Seller Orders

Products



Gigabyte X670E AORUS Master

€ 4444.00

View Details



Samsung Odyssey G9

€ 7314.99

View Details



Razer DeathAdder V3

€ 3212.00

View Details



Kingston 980 Pro 2TB

€ 3993.99

View Details



Intel Core i7-13900K

€ 4000.99

View Details

Рис. 3.21. Профіль користувача.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				49
Змн.	Арк.	№ докум.	Підпис	Дата		

3.3. Тестування платформи.

Для тестування метрик платформи, її було розміщено на тимчасовому хостингу localhost.run [27]. Для тестування було використано GTMetrix [28] та Pagespeed [29].

GTMetrix – інструмент для тестування швидкості завантаження сторінок, продуктивності, кількості запитів. Він створює звіти на основі таких метрик: розмір сторінки, час її завантаження, кількість запитів до серверу і тому подібне.

Після завершення тестування, було отримано звіт, показаний на рис. 3.22.

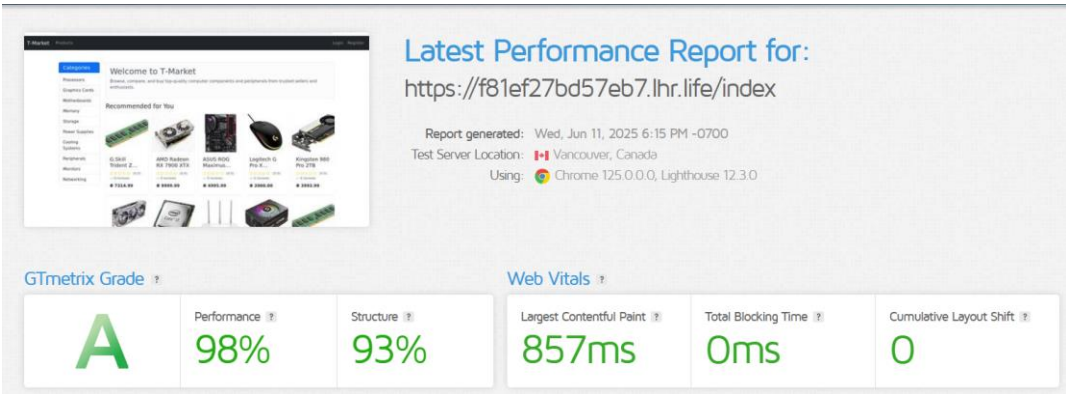


Рис. 3.22. Звіт після тестування веб-сайту на GTMetrix

GTMetrix формує інформативні звіти, які містять дані про швидкість завантаження сторінки, час завантаження вмісту, швидкість завантаження на різних типах з'єднання. Ця інформація дозволяє оцінити продуктивність веб-сайту та чи буде користувачу приємно на ньому знаходитись. GTMetrix також дає свої оцінки кожного критерію, та надає поради щодо того що потрібно покращити, та як саме це зробити. Показники показані на рис. 3.23.

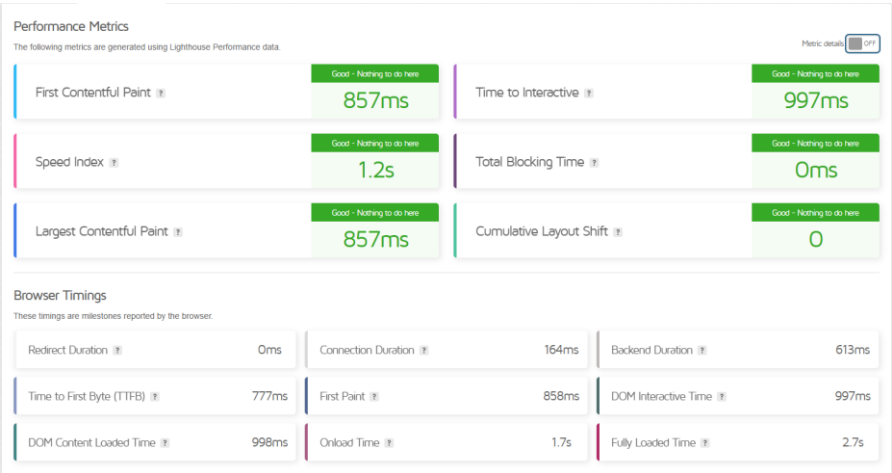


Рис. 3.23. Результати тестування швидкості

Pagespeed – веб-сервіс, створений командою Google, який надає інструменти для оцінки швидкості завантаження сторінок, на основі дає детальні рекомендації щодо покращення сайту. Можна провести оцінку свого веб-сайту і отримати детальний аналіз швидкості його завантаження. Pagespeed оцінює різні аспекти, такі як час завантаження, розмір сторінки, кількість запитів та швидкість їх обробки, тощо. На рис. 3.24 показаний результат тестування веб-додатку.

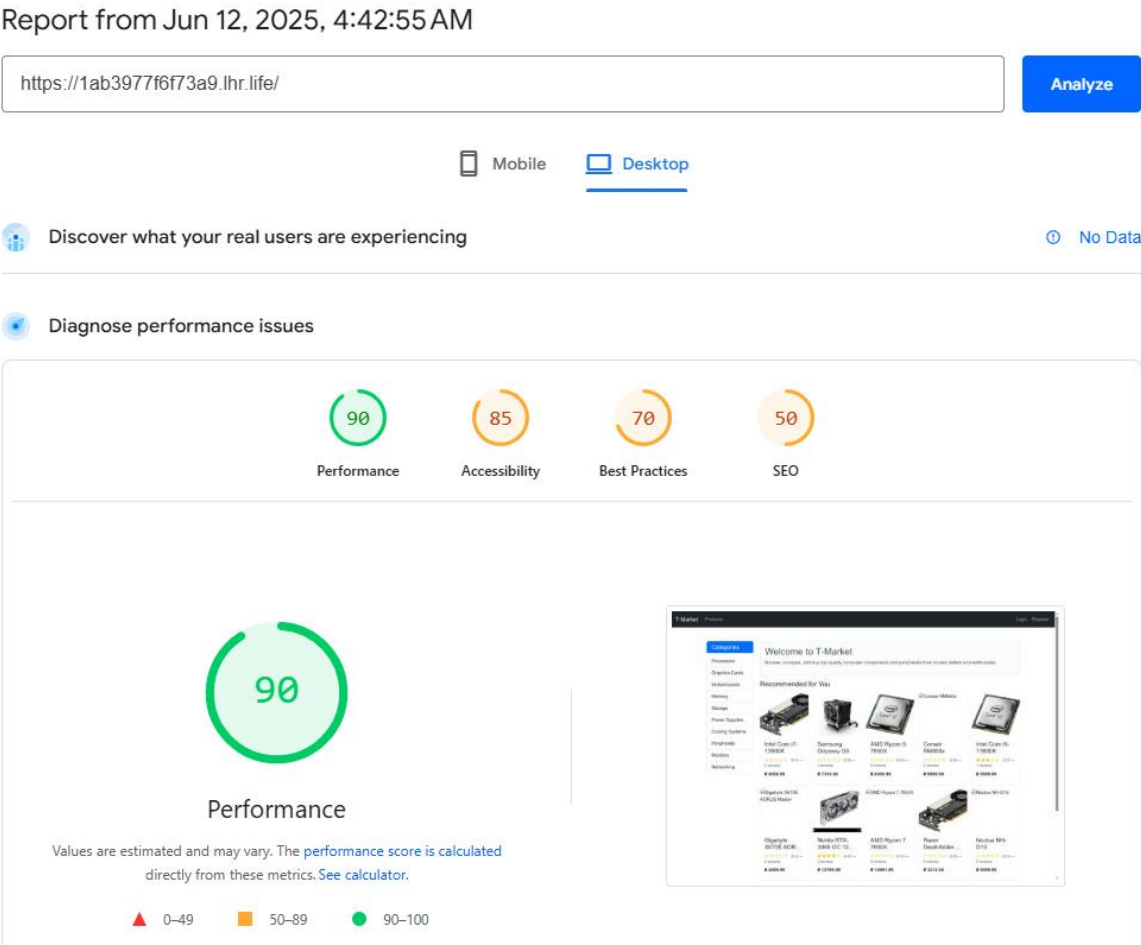


Рис. 3.24. Результат тестування сторінки на Pagespeed

Веб-сайт надає детальний список порад та рекомендацій, щодо практичних дій, які допоможуть покращити швидкість завантаження сторінок. Ці рекомендації ґрунтуються на кращих практиках Google для оптимізації веб-сайтів. На рис. 3.25 показаний приклад рекомендацій, який показує що можна зробити, та як саме це вплине на продуктивність сторінки.

DIAGNOSTICS

▲ Eliminate render-blocking resources — Est savings of 1,330 ms

Resources are blocking the first paint of your page. Consider delivering critical JS/CSS inline and deferring all non-critical JS/styles. [Learn how to eliminate render-blocking resources.](#) LCP FCP

URL	Transfer Size	Est Savings
JSDelivr CDN Cdn	27.5 KiB	900 ms
...css/bootstrap.min.css (cdn.jsdelivr.net)	27.5 KiB	900 ms

▲ Enable text compression — Est savings of 60 KiB

Text-based resources should be served with compression (gzip, deflate or brotli) to minimize total network bytes. [Learn more about text compression.](#) LCP FCP

URL	Transfer Size	Est Savings
lhr.life 1st Party	69.1 KiB	59.8 KiB
https://1ab3977f6f73a9.lhr.life	69.1 KiB	59.8 KiB

▲ Reduce initial server response time — Root document took 610 ms

▲ Largest Contentful Paint element — 2,940 ms

▲ Reduce unused CSS — Est savings of 37 KiB

▲ Does not have a <meta name="viewport"> tag with width or initial-scale No `<meta name="viewport">` tag found

■ Minify JavaScript — Est savings of 13 KiB

■ Serve images in next-gen formats — Est savings of 1,985 KiB

■ Serve static assets with an efficient cache policy — 10 resources found

■ Properly size images — Est savings of 2,748 KiB

■ Efficiently encode images — Est savings of 1,280 KiB

■ Reduce unused JavaScript — Est savings of 21 KiB

■ Avoid enormous network payloads — Total size was 3,085 KiB

○ Avoid large layout shifts — 2 layout shifts found

Рис. 3.25. Рекомендації щодо продуктивності.

Ресурс також перевіряє на готовність веб-сайту до роботи на мобільних пристроях. Надає рекомендації щодо адаптивного дизайну, розміщення контенту та інших аспектах, які можуть вплинути на досвід користувача при перегляду сторінки на телефоні [30]. (рис. 3.26).

Mobile

Desktop

SHOW DETAILS RELEVANT TO: LCP FCP TBT

INSIGHTS

▲ Render blocking requests

▲ Document request latency — Est savings of 46 KiB

▲ Optimize viewport for mobile

▲ Network dependency tree

■ Use efficient cache lifetimes — Est savings of 2,833 KiB

A long cache lifetime can speed up repeat visits to your page. [Learn more.](#) LCP FCP

Request	Cache TTL	Transfer Size
lhr.life 1st Party		2,833 KiB
...9/main-271897as4.jpg (1ab3977f6f73a9.lhr.life)	None	1,658 KiB
...17/main-271897as4.jpg (1ab3977f6f73a9.lhr.life)	None	661 KiB
...4/main-271897as4.jpg (1ab3977f6f73a9.lhr.life)	None	162 KiB
...14/main-271897as4.jpg (1ab3977f6f73a9.lhr.life)	None	134 KiB
...1007/main-6845e67f8361e.jpg (1ab3977f6f73a9.lhr.life)	None	61 KiB
...1005/main-6845e275b2e7f.jpg (1ab3977f6f73a9.lhr.life)	None	44 KiB
/images/Placeholder.jpg (1ab3977f6f73a9.lhr.life)	None	41 KiB
...3/main-271897as4.jpg (1ab3977f6f73a9.lhr.life)	None	30 KiB
...8/main-271897as4.jpg (1ab3977f6f73a9.lhr.life)	None	28 KiB
...15/main-271897as4.jpg (1ab3977f6f73a9.lhr.life)	None	14 KiB

■ Improve image delivery — Est savings of 2,781 KiB

Reducing the download time of images can improve the perceived load time of the page and LCP. [Learn more about optimizing image size](#) LCP

FCP

Рис. 3.26. Тестування роботи веб-сайту на мобільних пристроях

Pagespeed пропонує поради для різних платформ і технологій, що дозволяє розробникам отримати детальні поради та обрати відповідну стратегію оптимізації, не зважаючи чи вони працюють з React чи створюють сайт на Wordpress. Також ресурс є добре документованим та постійно підтримується. Документація містить багато корисних статей, керівництв та ресурсів стосовно досягнення максимальної оптимізації власної розробки.

За допомогою інструментів GTMetrix та Pagespeed було протестовано вебдодаток. За результатами отримано метрики: час завантаження сторінки, Largest Contentfull Page, Cumulative Layout Shift, Total Blocking Time, та отримано поради щодо покращення цих метрик.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		53

Висновки до третього розділу

В даному розділі була описана детальна схема розгортання платформи, було описано структурні елементи та продемонстровано їх взаємодію між собою на діаграмі.

Було проаналізовано інтерфейс платформи з точки зору користувача. Описана його взаємодія з веб-сайтом, та з іншими користувачами через нього. Були показані всі функціональні можливості продукту. Користувачі були розділені на ролі та мають різний функціонал.

Платформа була протестована та оцінена з боку продуктивності та швидкості роботи на ресурсах GTMetrix та Pagespeed. Було отримано звіти та поради щодо покращення ефективності роботи.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		54

ВИСНОВКИ

Під час виконання кваліфікаційної роботи бакалавра, було спроектовано та реалізовано онлайн маркетплейс комп'ютерної техніки. Нами було проведено аналіз предметної області та порівняно існуючі рішення для ведення електронної комерції. На основі цього аналізу були визначені основні вимоги до маркетплейсу комп'ютерної техніки. Була розроблена архітектура та обрані технології для реалізації програмного продукту.

Під час проектування платформи, було проаналізовано варіанти використання програмного продукту, змодельовано взаємодію складових частин між собою. Також було спроектовано базу даних для збереження інформації про користувачів, товари, замовлення. Були створені тригери, представлення та процедури.

Описано алгоритм роботи з додатком та послідовність варіантів його використання, для чого була представлена діаграма активності. Розглянуто основний функціонал: авторизація в акаунті, пошук та створення товарів, замовлення товарів.

Була надана детальна схема розгортання маркетплейсу. Описано взаємодію елементів системи та наведено опис інтерфейсу розробленого продукту для різних користувачів, описані основні можливості та типічний порядок дій на веб-сайті.

Архітектура платформи побудована щоб забезпечити зручність використання та масштабованість. Вона базується на модульному підході, що забезпечує чітке розділення відповідальності між її складовими частинами

Інтерфейс веб сайту інтуїтивний і дозволяє користувачеві зручно переглядати товари, шукати та фільтрувати їх, залишати відгуки тощо. Продавці можуть виставляти на продаж нові товари та керувати інформацією в існуючих.

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ПЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		55

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Rozetka [Електронний ресурс] – Режим доступу до ресурсу: <https://rozetka.com.ua/>
2. Aliexpress [Електронний ресурс] – Режим доступу до ресурсу: <https://www.aliexpress.com/>
3. Allo [Електронний ресурс] – Режим доступу до ресурсу: <https://allo.ua/>
4. OLX [Електронний ресурс] – Режим доступу до ресурсу: <https://www.olx.ua/uk/>
5. Prom [Електронний ресурс] – Режим доступу до ресурсу: <https://prom.ua/>
6. PHP Manual [Електронний ресурс] – Режим доступу: <https://www.php.net/manual/en/index.php>
7. Symfony [Електронний ресурс] – Режим доступу до ресурсу: <https://symfony.com/>
8. MVC Pattern – MDN Glossary [Електронний ресурс] – Режим доступу: <https://developer.mozilla.org/en-US/docs/Glossary/MVC>
9. Composer [Електронний ресурс] – Режим доступу: <https://getcomposer.org/doc/>
10. Doctrine ORM [Електронний ресурс] – Режим доступу до ресурсу: <https://www.doctrine-project.org/>
11. OWASP Foundation. CSRF Overview [Електронний ресурс] – Режим доступу до ресурсу: <https://owasp.org/www-community/attacks/csrf>
12. OWASP Foundation. SQL Injection [Електронний ресурс] – Режим доступу до ресурсу: https://owasp.org/www-community/attacks/SQL_Injection
13. OWASP Foundation. Cross-site Scripting (XSS) [Електронний ресурс] – Режим доступу до ресурсу: <https://owasp.org/www-community/attacks/xss/>
14. Twig [Електронний ресурс] – Режим доступу до ресурсу: <https://twig.symfony.com/>
15. JQuery [Електронний ресурс] – Режим доступу до ресурсу: <https://jquery.com/>

- 16.AJAX Documentation[Электронний ресурс] – Режим доступа до ресурсу:
<https://api.jquery.com/category/ajax/>
- 17.Bootstrap [Электронний ресурс] – Режим доступа до ресурсу:
<https://getbootstrap.com/>
- 18.SQL Server Documentation [Электронний ресурс] – Режим доступа до ресурсу:
<https://learn.microsoft.com/en-us/sql/sql-server/?view=sql-server-ver17>
- 19.Failover clustering [Электронний ресурс] – Режим доступа до ресурсу:
<https://learn.microsoft.com/en-us/windows-server/failover-clustering/failover-clustering-overview>
- 20.OWASP – Data Backup and Recovery [Электронний ресурс] – Режим доступа до ресурсу:
https://owasp.org/www-project-top-ten/2017/A6_2017-Security_Misconfiguration.html
- 21.Symfony Exception Handling [Электронний ресурс] – Режим доступа до ресурсу:
https://symfony.com/doc/current/controller/error_pages.html
- 22.EntityManager [Электронний ресурс] – Режим доступа до ресурсу:
<https://www.doctrine-project.org/projects/doctrine-orm/en/current/tutorials/getting-started.html>
- 23.Session management [Электронний ресурс] – Режим доступа до ресурсу:
<https://symfony.com/doc/current/session.html>
- 24.File Upload [Электронний ресурс] – Режим доступа до ресурсу:
https://symfony.com/doc/current/controller/upload_file.html
25. Doctrine Query Builder [Электронний ресурс] – Режим доступа до ресурсу:
<https://www.doctrine-project.org/projects/doctrine-orm/en/current/reference/query-builder.html>
- 26.Symfony Events and event listeners [Электронний ресурс] – Режим доступа до ресурсу:
https://symfony.com/doc/current/event_dispatcher.html
- 27.Localhost.run [Электронний ресурс] – Режим доступа до ресурсу:
<https://localhost.run/docs/>
- 28.GTMetrix [Электронний ресурс] – Режим доступа до ресурсу:
<https://gtmetrix.com/>

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				57
Змн.	Арк.	№ докум.	Підпис	Дата		

29.Pagespeed [Електронний ресурс] – Режим доступу:

<https://pagespeed.web.dev/>

30.NNgroup Articles on mobile and tablet design [Електронний ресурс] – Режим доступу до ресурсу:

<https://www.nngroup.com/topic/mobile-and-tablet-design/>

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		58

ДОДАТКИ

Контролер, відповідальний за реєстрацію та автентифікацію

```

final class SecurityController extends AbstractController
{
    #[\Symfony\Component\Routing\Annotation\Route('/register', name: 'app_register')]
    public function register(
        Request $request,
        EntityManagerInterface $em,
        UserPasswordHasherInterface $passwordHasher
    ): Response {
        //Create a new empty user object
        $user = new Users();

        //Build a form for registration using Symfony FormBuilder
        $form = $this->createFormBuilder($user)
            ->add('Username', TextType::class)
            ->add('Email', EmailType::class)
            ->add('Password', PasswordType::class)
            ->add('FirstName', TextType::class)
            ->add('LastName', TextType::class)
            ->add('PhoneNumber', TextType::class)
            ->add('Address', TextType::class)
            ->add('submit', SubmitType::class, ['label' => 'Register'])
            ->getForm();

        //Handle form submission
        $form->handleRequest($request);

        if ($form->isSubmitted() && $form->isValid()) {
            //Hash the password using the security component
            $hashedPassword = $passwordHasher->hashPassword($user, $user->getPassword());
            $user->setPassword($hashedPassword);
            $user->setRoles(['ROLE_USER']);
            //Save the user to the database
            $em->persist($user);
        }
    }
}

```

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				60
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        $sem->flush();

        //Redirect to homepage
        return $this->redirectToRoute('app_index');
    }

    //Render the registration page with the form
    return $this->render('security/register.html.twig', [
        'registrationForm' => $form->createView(),
    ]);
}

#[Route(path: '/login', name: 'app_login')]
public function login(AuthenticationUtils $authenticationUtils): Response
{
    $error = $authenticationUtils->getLastAuthenticationError();
    //save last username entered by the user
    $lastUsername = $authenticationUtils->getLastUsername();

    return $this->render('security/login.html.twig', ['last_username' => $lastUsername, 'error' =>
$error]);
}

#[Route(path: '/logout', name: 'app_logout')]
public function logout(): void
{
    throw new \LogicException('This method can be blank - it will be intercepted by the logout
key on the firewall.');
```

Метод для створення нового товару:

```

#[Route('/product/new', name: 'product_new')]
#[IsGranted('IS_AUTHENTICATED_FULLY')]
public function create(Request $request, EntityManagerInterface $em): Response
```

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		61

```

{
    $product = new Products();

    $form = $this->createFormBuilder($product)
        ->add('Name', TextType::class, [
            'constraints' => [
                new Length(['max' => 100]) // match DB column size
            ]
        ])
        ->add('Description', TextareaType::class)
        ->add('Price', MoneyType::class, [
            'currency' => false,
            'scale' => 2,
        ])
        ->add('CategoryID', EntityType::class, [
            'class' => Categories::class,
            'choice_label' => 'Name',
            'placeholder' => 'Choose a category',
            'label' => 'Category',
        ])
        ->add('mainPhoto', FileType::class, [
            'mapped' => false,
            'required' => false,
            'label' => 'Main Photo',
        ])
        ->add('additionalPhotos', FileType::class, [
            'mapped' => false,
            'required' => false,
            'multiple' => true,
            'label' => 'Additional Photos',
        ])
        ->add('submit', SubmitType::class, ['label' => 'Create Product'])
        ->getForm();

    $form->handleRequest($request);

```

```

if ($form->isSubmitted() && $form->isValid()) {
    $product->setSellerID($this->getUser());

    // Fetch selected category
    $category = $product->getCategoryID();
    $categoryAttributes = $sem->getRepository(CategoryAttributes::class)-
>findBy(['CategoryID' => $category]);

    // Get posted attribute data
    $attributeData = $request->request->all('attributes');

    // Validate all required attributes are filled
    foreach ($categoryAttributes as $catAttr) {
        $attrId = $catAttr->getAttributeID();
        if (!isset($attributeData[$attrId]) || trim($attributeData[$attrId]) === '') {
            $this->addFlash('error', 'Please fill in all required attribute fields.');
```

```

            return $this->render('product/create.html.twig', [
                'form' => $form->createView(),
                'categoryAttributes' => $categoryAttributes,
                'attributeValues' => $attributeData,
            ]);
        }
    }

    $sem->persist($product);
    $sem->flush();

    // Save product attributes
    foreach ($attributeData as $attributeId => $value) {
        $categoryAttribute = $sem->getRepository(CategoryAttributes::class)-
>find($attributeId);

        if ($categoryAttribute) {
            $productAttr = new ProductAttributes();
            $productAttr->setProductID($product);

```

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		63

```

        $productAttr->setAttributeID($categoryAttribute);
        $productAttr->setValue($value);
        $sem->persist($productAttr);
    }
}

$sem->flush();

// Create upload directory
$uploadDir = $this->getParameter('kernel.project_dir') . '/public/uploads/products/' .
$product->getProductID();
if (!is_dir($uploadDir)) {
    mkdir($uploadDir, 0777, true);
}

// Save main photo
$mainPhoto = $form->get('mainPhoto')->getData();
if ($mainPhoto) {
    try {
        $uniqueId = uniqid();
        $mainFilename = "main-$uniqueId." . $mainPhoto->guessExtension();
        $mainPhoto->move($uploadDir, $mainFilename);
    } catch (FileException $e) {
        $this->addFlash('error', 'Could not upload main photo.');
```

```

    }

    // Save additional photos
```

```

    $additionalPhotos = $form->get('additionalPhotos')->getData();
    $counter = 2;
    foreach ($additionalPhotos as $photo) {
        if ($photo) {
            try {
                $uniqueId = uniqid();
                $filename = "$counter-$uniqueId." . $photo->guessExtension();
```

		Панчук П.С.			ІІЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		64

```

        $photo->move($uploadDir, $filename);
        $counter++;
    } catch (FileNotFoundException $e) {
        $this->addFlash('error', 'Could not upload additional photo.');
```

```

    }
}

return $this->redirectToRoute('app_index');
}

$category = $form->get('CategoryID')->getData();
$categoryAttributes = [];

if ($category) {
    $categoryAttributes = $sem->getRepository(CategoryAttributes::class)-
>findBy(['CategoryID' => $category]);
}

return $this->render('product/create.html.twig', [
    'form' => $form->createView(),
    'categoryAttributes' => $categoryAttributes,
]);
}

```

Метод для керування картинками товару.

```

#[Route('/product/{id}/photos', name: 'product_photos')]
#[IsGranted('IS_AUTHENTICATED_FULLY')]
public function editPhotos(int $id, Request $request, EntityManagerInterface $em): Response
{
    $product = $em->getRepository(Products::class)->find($id);

    if (!$product) {
        throw $this->createNotFoundException('Product not found');
    }
}

```

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		65

```

if (!$this->isGranted('ROLE_ADMIN') &&
    $product->getSellerID() !== $this->getUser()) {
    throw $this->createAccessDeniedException('You can only edit your own products.');
```

```

}

$uploadDir = $this->getParameter('kernel.project_dir') . '/public/uploads/products/' .
$product->getProductID();
```

```

// Handle submitted image replacements
```

```

if ($request->isMethod('POST')) {
```

```

    // Delete request
```

```

    $fileToDelete = $request->request->get('delete_file');
```

```

    if ($fileToDelete) {
```

```

        $safeFileToDelete = basename($fileToDelete);
```

```

        $fullPath = $uploadDir . '/' . $safeFileToDelete;
```

```

        if (file_exists($fullPath)) {
```

```

            @unlink($fullPath);
```

```

            $this->addFlash('success', 'Image deleted.');
```

```

        } else {
```

```

            $this->addFlash('error', 'Image not found.');
```

```

        }
```

```

        return $this->redirectToRoute('product_photos', ['id' => $id]);
```

```

    }
```

```

foreach ($request->files->all() as $key => $file) {
```

```

    if ($file && $file instanceof UploadedFile) {
```

```

        if (preg_match("/^(w+)-?/", str_replace('_jpg', '', $key), $matches)) {
```

```

            $prefix = $matches[1] . '-'; // main- or 2-
```

```

            // Delete old files with the same prefix to make room for new
```

```

            $filesToDelete = glob($uploadDir . '/' . $prefix . '*');
```

		Панчук П.С.			ІІЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		66

```

        foreach ($filesToDelete as $fileToDelete) {
            @unlink($fileToDelete);
        }

        $uniqueId = uniqid();
        $newFilename = $prefix . $uniqueId . '.jpg';
        $file->move($uploadDir, $newFilename);
    }
}

// Handle new additional images
/** @var UploadedFile[] $newPhotos */
$newPhotos = $request->files->get('new_additional_photos', []);

if (!empty($newPhotos)) {
    // Find the highest numeric prefix
    $existingFiles = scandir($uploadDir);
    $maxPrefix = 1;
    foreach ($existingFiles as $existingFile) {
        if (preg_match('/^(\\d+)-/', $existingFile, $matches)) {
            $prefix = (int)$matches[1];
            if ($prefix > $maxPrefix) {
                $maxPrefix = $prefix;
            }
        }
    }

    $counter = $maxPrefix + 1;
    foreach ($newPhotos as $photo) {
        if ($photo instanceof UploadedFile) {
            $uniqueId = uniqid();
            $filename = $counter . '-' . $uniqueId . '.' . $photo->guessExtension();
            $photo->move($uploadDir, $filename);
        }
    }
}

```

```

        $counter++;
    }
}

$this->addFlash('success', 'New images added.');
```

```

    return $this->redirectToRoute('product_photos', ['id' => $id]);
}

$this->addFlash('success', 'Old image(s) deleted.');
```

```

    return $this->redirectToRoute('product_photos', ['id' => $id]);
}

// Get all image files
$imageFiles = [];
if (file_exists($uploadDir)) {
    $allFiles = scandir($uploadDir);
    foreach ($allFiles as $filename) {
        if (preg_match('/\.(jpg|jpeg|png)$/i', $filename)) {
            $imageFiles[] = $filename;
        }
    }
    sort($imageFiles, SORT_NATURAL); // main.jpg, 2.jpg, 3.jpg...
}

return $this->render('product/photos.html.twig', [
    'product' => $product,
    'imageFiles' => $imageFiles,
]);
}

```

Контролер каталогу товарів:

```

final class ProductListController extends AbstractController
{
    #[Route('/products', name: 'product_list')]

```

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		68

```

public function index(Request $request, EntityManagerInterface $em): Response
{
    $name = $request->query->get('name');
    $minPrice = $request->query->get('min_price');
    $maxPrice = $request->query->get('max_price');
    $categoryId = $request->query->get('category');

    $qb = $em->getRepository(Products::class)->createQueryBuilder('p');

    if ($name) {
        $qb->andWhere('p.Name LIKE :name')
            ->setParameter('name', '%' . $name . '%');
    }

    if ($minPrice) {
        $qb->andWhere('p.Price >= :min')
            ->setParameter('min', $minPrice);
    }

    if ($maxPrice) {
        $qb->andWhere('p.Price <= :max')
            ->setParameter('max', $maxPrice);
    }

    if ($categoryId) {
        $qb->andWhere('p.CategoryID = :cat')
            ->setParameter('cat', $categoryId);
    }

    $products = $qb->getQuery()->getResult();
    $baseDir = $this->getParameter('kernel.project_dir') . '/public/uploads/products';

    $productsWithImages = [];

    foreach ($products as $product) {

```

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		69

```

$imagePath = 'images/Placeholder.jpg';
$imageDir = $baseDir . '/' . $product->getProductID();

if (is_dir($imageDir)) {
    foreach (scandir($imageDir) as $file) {
        if (preg_match('/^main-[a-zA-Z0-9]+\.(jpg|jpeg|png)$/', $file)) {
            $imagePath = 'uploads/products/' . $product->getProductID() . '/' . $file;
            break;
        }
    }
}

// Get rating stats
$reviews = $em->getRepository(Reviews::class)->findBy(['ProductID' => $product]);
$reviewCount = count($reviews);
$averageRating = $reviewCount > 0 ? array_sum(array_map(fn($r) => $r->getRating(),
$reviews)) / $reviewCount : 0;

$productsWithImages[] = [
    'product' => $product,
    'image' => $imagePath,
    'averageRating' => $averageRating,
    'reviewCount' => $reviewCount,
];
}

// Get all categories for filter dropdown
$categories = $em->getRepository(Categories::class)->findAll();

return $this->render('product/list.html.twig', [
    'products' => $productsWithImages,
    'categories' => $categories,
    'filters' => [
        'name' => $name,
        'min_price' => $minPrice,
        'max_price' => $maxPrice,

```

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				70
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        'category' => $categoryId,
    ],
    });
}
}

```

Леяут сторінки каталогу товарів:

```

{% extends 'base.html.twig' %}

{% block title %}Product Catalog{% endblock %}

{% block body %}
    <div class="container">
        <div class="row">

            {#Sidebar#}
            <div class="col-md-2 mb-4">
                <div class="list-group">
                    <h5 class="list-group-item active">Categories</h5>
                    {% for category in categories %}
                        <a href="{{ path('product_list', { category: category.CategoryID }) }}"
                            class="list-group-item list-group-item-action {{ filters.category ==
category.CategoryID ? 'active' }}">
                            {{ category.Name }}
                        </a>
                    {% endfor %}
                </div>
            </div>

            {#Main Content#}
            <div class="col-md-10">
                <h1 class="mb-4 text-center">All Products</h1>

                <form method="get" action="{{ path('product_list') }}" class="row g-3 mb-4">
                    <div class="col-md-3">

```

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		71

```

        <input type="text" name="name" class="form-control" placeholder="Search by
name" value="{{ filters.name ?? " }}">
    </div>
    <div class="col-md-2">
        <input type="number" name="min_price" class="form-control"
placeholder="Min Price" value="{{ filters.min_price ?? " }}">
    </div>
    <div class="col-md-2">
        <input type="number" name="max_price" class="form-control"
placeholder="Max Price" value="{{ filters.max_price ?? " }}">
    </div>
    <div class="col-md-3">
        <select name="category" class="form-select">
            <option value="">All Categories</option>
            {% for cat in categories %}
                <option value="{{ cat.CategoryID }}" {% if filters.category ==
cat.CategoryID %}>selected{% endif %}>
                    {{ cat.Name }}
                </option>
            {% endfor %}
        </select>
    </div>
    <div class="col-md-2">
        <button type="submit" class="btn btn-primary w-100">Filter</button>
    </div>
</form>

<div class="row row-cols-1 row-cols-sm-2 row-cols-md-4 g-4">
    {% for item in products %}
        {% set product = item.product %}
        {% set imagePath = item.image %}
        {% set averageRating = item.averageRating %}
        {% set reviewCount = item.reviewCount %}

        <div class="col">

```

```

<a href="{{ path('product_show', { id: product.ProductID }) }}" style='text-
decoration: none; color: inherit;'>

<div class="card h-100 shadow-sm border-0">
    
    <div class="card-body d-flex flex-column">
        <h5 class="card-title"

            style="display: -webkit-box; -webkit-line-clamp: 2; -webkit-box-
orient: vertical; overflow: hidden; text-overflow: ellipsis; height: 2.5em;">
            {{ product.Name }}
        </h5>

        {#Rating of a product#}
        <div class="mb-2" style="font-size: 0.8rem; color: #ffc107;">
            {% set fullStars = averageRating|round(0, 'floor') %}
            {% set halfStar = (averageRating - fullStars) >= 0.5 %}
            {% set emptyStars = 5 - fullStars - (halfStar ? 1 : 0) %}
            {% for i in 1..fullStars %}
                {% if loop.index < fullStars+1 %}
                    <i class="bi bi-star-fill text-warning"></i>
                {% endif %}
            {% endfor %}
            {% if halfStar %}
                <i class="bi bi-star-half text-warning"></i>
            {% endif %}
            {% for i in 1..5 %}
                {% if loop.index < emptyStars+1 %}
                    <i class="bi bi-star text-warning"></i>
                {% endif %}
            {% endfor %}
            <span

                class="text-muted

                ms-2">({{
averageRating|number_format(1) }}) — {{ reviewCount }} reviews</span>
        </div>

        <p class="card-text fw-bold mb-2">€ {{ product.Price }}</p>

```

		Панчук П.С.			ІІЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		73

```

        </div>
    </div>
</a>
</div>

    {% endfor %}
</div>
</div>

</div>
</div>
{% endblock %}

```

Метод, для рекомендації процесора на основі кошика.

Аналогічно цьому методу, працюють SuggestGPU, SuggestMotherboard, SuggestPSU. Вони шукають інші дані, але принцип такий самий.

```

public function suggestCPU(Request $request, EntityManagerInterface $em): array
{
    $cart = $request->getSession()->get('cart', []);
    $productRepo = $em->getRepository(Products::class);
    $attrRepo = $em->getRepository(ProductAttributes::class);

    $cartProducts = [];
    foreach ($cart as $id => $qty) {
        $product = $productRepo->find($id);
        if ($product) {
            $cartProducts[] = $product;
        }
    }

    $attributesByProduct = [];
    foreach ($cartProducts as $product) {
        $attrs = $attrRepo->findBy(['ProductID' => $product]);
        foreach ($attrs as $attr) {

```

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		74

```

        $attributesByProduct[$product->getProductID()][ $attr->getAttributeID()-
>getName()] = $attr->getValue();
    }
}

```

```

$socketType = null;
$psuWattage = null;
$gpuWattage = null;

```

```

foreach ($cartProducts as $product) {
    $productId = $product->getProductID();
    $categoryId = $product->getCategoryID()->getCategoryID();

```

```

    $attrs = $attributesByProduct[$productId] ?? [];

```

```

    if (isset($attrs['Socket Type'])) {
        $socketType = $attrs['Socket Type'];
    }

```

```

    if (isset($attrs['Wattage'])) {
        if ($categoryId === 6) { // PSU category
            $psuWattage = $attrs['Wattage'];
        } elseif ($categoryId === 2) { // GPU category
            $gpuWattage = $attrs['Wattage'];
        }
    }
}

```

```

$availableWattage = $psuWattage !== null && $gpuWattage !== null ? $psuWattage -
$gpuWattage : null;

```

```

// Build query to suggest CPUs
$qb = $em->createQueryBuilder();
$qb->select('p')

```

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				75
Змн.	Арк.	№ докум.	Підпис	Дата		

```

->from(Products::class, 'p')
->join('App\Entity\ProductAttributes', 'pa', 'WITH', 'pa.ProductID = p.ProductID')
->join('pa.AttributeID', 'a')
->where('p.CategoryID = :cpuCategory')
->setParameter('cpuCategory', 1);

```

```

// Filter by matching attributes
if ($socketType !== null) {
    $qb->andWhere("EXISTS (
        SELECT 1 FROM App\Entity\ProductAttributes pa2
        JOIN pa2.AttributeID a2
        WHERE pa2.ProductID = p.ProductID AND a2.Name = 'Socket Type' AND pa2.Value =
:socket
    )")
    ->setParameter('socket', $socketType);
}

if ($availableWattage !== null) {
    $qb->andWhere("EXISTS (
        SELECT 1 FROM App\Entity\ProductAttributes pa4
        JOIN pa4.AttributeID a4
        WHERE pa4.ProductID = p.ProductID AND a4.Name = 'Wattage' AND pa4.Value <=
:availableWatt
    )")
    ->setParameter('availableWatt', $availableWattage);
}

return $suggestedCPUs = $qb->setMaxResults(5)->getQuery()->getResult();
}

```

index метод кошика

```
#[Route('/', name: 'cart_index')]
```

		Панчук П.С.			ІПЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				
Змн.	Арк.	№ докум.	Підпис	Дата		76

```

public function index(Request $request, SessionInterface $session,
EntityManagerInterface $em): Response
{
    $cart = $session->get('cart', []);
    $productRepo = $em->getRepository(Products::class);

    // Handle quantity updates
    if ($request->isMethod('POST')) {
        $productId = $request->request->get('product_id');
        $action = $request->request->get('action');

        if ($productId && $action) {
            if (!isset($cart[$productId])) {
                $cart[$productId] = 0;
            }

            if ($action === 'increase') {
                $cart[$productId]++;
            } elseif ($action === 'decrease') {
                $cart[$productId]--;
                if ($cart[$productId] <= 0) {
                    unset($cart[$productId]);
                }
            }
        }

        $session->set('cart', $cart);
        return $this->redirectToRoute('cart_index');
    }
}

```

```

// Load cart product details
$cartItems = [];
foreach ($cart as $id => $quantity) {
    $product = $productRepo->find($id);
    if ($product) {
        $cartItems[] = [
            'product' => $product,
            'quantity' => $quantity,
        ];
    }
}

//SUGGESTIONS SECTION
$hasCPU = $hasGPU = $hasMotherboard = $hasPSU = false;

foreach ($cart as $id => $qty) {
    $product = $productRepo->find($id);
    if (!$product) continue;

    $categoryId = $product->getCategoryID()?->getCategoryID();

    if ($categoryId === 1) $hasCPU = true;
    if ($categoryId === 2) $hasGPU = true;
    if ($categoryId === 3) $hasMotherboard = true;
    if ($categoryId === 6) $hasPSU = true;
}

$suggestedCPUs = [];
if (!$hasCPU && $hasMotherboard) {

```

```

        $suggestedCPUs    =    $this->attachImages($this->suggestCPU($request,
$sem));
    }

    $suggestedGPUs = [];
    if (!$hasGPU && $hasPSU) {
        $suggestedGPUs    =    $this->attachImages($this->suggestGPU($request,
$sem));
    }

    $suggestedMotherboards = [];
    if (!$hasMotherboard && $hasCPU) {
        $suggestedMotherboards    =    $this->attachImages($this-
>suggestMotherboard($request, $sem));
    }

    $suggestedPSUs = [];
    if (!$hasPSU && $hasCPU && $hasGPU) {
        $suggestedPSUs    =    $this->attachImages($this->suggestPSU($request,
$sem));
    }

    return $this->render('cart/index.html.twig', [
        'cartItems' => $cartItems,
        'cpus' => $suggestedCPUs,
        'gpus' => $suggestedGPUs,
        'motherboards' => $suggestedMotherboards,
        'psus' => $suggestedPSUs,
    ]);

```

		Панчук П.С.			ІІЗ.КР.Б – 121 – 25 – ІІЗ	Арк.
		Сфремов Ю.М.				79
Змн.	Арк.	№ докум.	Підпис	Дата		