

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»
ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра інформаційних технологій

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«ОНЛАЙН СИСТЕМА ЕЛЕКТРОННОЇ КОМЕРЦІЇ»

Рівень «бакалавр»

Галузь знань 12 «Інформаційні технології»,

спеціальність 122 «Комп'ютерні науки»

Виконав здобувач 4 курсу

Сімонян Левон Арменович

Керівник: к.т.н., доцент

Щербина Юрій Володимирович

Рецензент: к.т.н., доцент

Чикунів Павло Олександрович

Одеса – 2024

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»
 Факультет кібербезпеки та інформаційних технологій
 Кафедра інформаційних технологій
 Галузь знань: 12 Інформаційні технології
 Спеціальність: 122 Комп'ютерні науки
 Назва освітньої програми: Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри інформаційних технологій
 доцент Н.І. Логінова

_____ « ____ » _____ 20__ року

З А В Д А Н Н Я
на кваліфікаційну (бакалаврську) роботу
здобувачу Сімоняну Левону Арменовичу

1. Тема роботи: «Онлайн система електронної комерції»
 Керівник роботи: к.т.н, доцент Щербина Юрій Володимирович
 затверджені наказом НУ ОЮА від «02» квітня 2024 року № 809-06
2. Строк подання здобувачем роботи «20» травня 2024 року
3. Дата видачі завдання «15» вересня 2023 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Аналіз предметної області	02.10.2023	
2	Проектування веб-застосунку	04.12.2023	
3	Вибір програмних засобів для реалізації поставлених задач	28.01.2024	
4	Розробка клієнтської частини	14.02.2024	
5	Розробка серверної частини	15.03.2024	
6	Тестування функціональності системи	26.04.2024	
7	Оформлення пояснювальної записки	17.05.2024	

Здобувач _____
 (підпис) (прізвище та ініціали)

Керівник роботи _____
 (підпис) (прізвище та ініціали)

АНОТАЦІЯ

Сімонян Л. А. «Онлайн система електронної комерції». – Кваліфікаційна робота бакалавра за спеціальністю 122 «Комп'ютерні науки», освітньою програмою «Комп'ютерні науки».

Кваліфікаційна робота містить 3 розділи, 22 рисунка, 1 таблицю, 15 джерел за переліком посилань. Робота виконана на 42 сторінках загального тексту.

Мета кваліфікаційної роботи – розробка вебзастосунку для віртуальної торговельно-клірингової платформи.

Об'єкт дослідження – онлайн система електронної комерції.

Предмет дослідження – процес розробки вебзастосунку електронної комерції.

Практична значущість результатів дослідження: створення та впровадження ефективної та конкурентоспроможної онлайн система електронної комерції.

В першому розділі роботи проведений аналіз предметної області, визначені особливості електронної комерції. Порівнянні програмні продукти-аналоги, які є на ІТ-ринку. Визначені функціональні вимоги до онлайн системи електронної комерції. Другий розділ присвячений проектуванню онлайн системи. Побудовані інформаційна модель та база даних такої системи. Розглянута архітектура онлайн системи. В третьому розділі кваліфікаційної роботи розроблена клієнт-серверна частина. Описано шляхи реалізації та тестування онлайн системи електронної комерції.

Ключові слова: онлайн система, електронна комерція, вебзастосунок, клієнт-сервер, база даних

ABSTRACT

Simonyan L. A. "Online system of electronic commerce." – Bachelor's qualification work in specialty 122 "Computer Science", educational program "Computer Science".

The qualification work contains 3 chapters, 22 figures, 1 table, 15 sources according to the list of references. The work is completed on 42 pages of the general text.

The purpose of the qualification work is to develop a web application for a virtual trading and clearing platform.

The object of the study is an online e-commerce system.

The subject of research is the process of developing an e-commerce web application.

Practical significance of research results: creation and implementation of an effective and competitive online e-commerce system.

In the first section of the work, the analysis of the subject area was carried out, the features of electronic commerce were determined. Comparable analogue software products available on the IT market. The functional requirements for the online e-commerce system are defined. The second section is devoted to the design of the online system. The information model and database of such a system have been built. The considered architecture of the online system. In the third section of the qualification work, the client-server part is developed. Ways of implementing and testing an online e-commerce system are described.

Keywords: online system, e-commerce, web application, client-server, database

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ	6
ВСТУП	7
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	8
1.1 Особливості електронної комерції.....	8
1.2 Аналіз програмних продуктів-аналогів.....	9
1.3 Визначення функціональних вимог онлайн системи електронної комерції.....	16
1.4 Вибір засобів розробки онлайн системи електронної комерції.....	18
2 ПРОЄКТУВАННЯ ОНЛАЙН СИСТЕМИ ЕЛЕКТРОННОЇ КОМЕРЦІЇ.....	23
2.1 Інформаційна модель та база даних онлайн системи електронної комерції.....	23
2.2 Архітектура онлайн системи електронної комерції.....	26
3 РОЗРОБКА ТА ТЕСТУВАННЯ ОНЛАЙН СИСТЕМИ ЕЛЕКТРОННОЇ КОМЕРЦІЇ	29
3.1 Розробка клієнт-серверної частини	29
3.2 Реалізація онлайн системи.....	30
3.3 Тестування онлайн системи електронної комерції	37
ВИСНОВКИ	40
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	41

ПЕРЕЛІК СКОРОЧЕНЬ

API – Application Programming Interface

DOM – Document Object Model

ER-діаграма – Entity-relationship diagram

MVC – Model-View-Controller

URL – Uniform Resource Locator

СКБД – система керування базами даних

ВСТУП

Внаслідок стрімкого розвитку й впровадження технологій в наше повсякдення, багато видів діяльності неминуче переходять в онлайн-середовище, роблячи інтернет невід'ємною частиною життя. До цього сприяла велика частина факторів, серед яких збільшення потужності комп'ютерів, швидший інтернет-зв'язок, менші формфактор електронних пристроїв та більша енергоефективність.

Цей зростаючий вплив технологій сприяв переходу бізнесу та послуг в Інтернет, що стало важливим кроком для розвитку та успіху ділового світу.

Електронна комерція – це процес купівлі та продажу товарів, а також надання послуг через інтернет, який допомагає компаніям пропонувати нові продукти та послуги, вдосконалювати існуючі технології та методи продажу, розширювати свою базу користувачів та завдяки цьому збільшувати свій прибуток.

Метою кваліфікаційної роботи є розробка вебзастосунку для віртуальної торговельно-клірингової платформи.

Відповідно до мети *завданнями* кваліфікаційної роботи є:

- провести аналіз існуючих рішень для е-комерції в різних сферах бізнесу та визначити їх характеристики;
- формування основних вимог;
- проєктування вебзастосунку;
- розробка бази даних, клієнтської та серверної частин.

Об'єкт дослідження – онлайн система електронної комерції.

Предмет дослідження – процес розробки вебзастосунку електронної комерції.

Практична значущість результатів дослідження: створення та впровадження ефективної та конкурентоспроможної онлайн система електронної комерції.

Структура кваліфікаційної роботи складається зі вступу, основної частини з трьох розділів, висновків, списку використаних джерел. Робота викладена на 42 сторінках, містить 22 рисунка, 1 таблицю. Перелік посилань включає 15 джерел.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Особливості електронної комерції

Електронний бізнес активно вивчається науковцями, про що свідчить велика кількість публікацій та інформаційних джерел, присвячених цій темі [1–2].

Українські дослідники та автори чітко окреслили базові концепції та принципи електронного бізнесу та інтернет-торгівлі. Під електронним бізнесом можна розуміти ведення підприємницької діяльності через мережу інтернет.

Сьогодні відбувається перехід від вебсторінок до повноцінних інтернет-магазинів. Це дозволяє компаніям збільшити свою присутність на ринку, швидше продавати товари та послуги, а також отримати нові канали для маркетингу та реклами.

Електронна комерція – це спосіб ведення електронного бізнесу і являє собою широкий спектр інтерактивних методів, що використовуються для доставки товарів і послуг споживачам. Під електронною комерцією розуміється будь-яка комерційна операція, в якій сторони взаємодіють за допомогою електронних технологій.

Останні тенденції та перспективи розвитку розглядаються в [2]. Світова економіка переживає процес глобальної трансформації, в основі якого лежить діджиталізація.

Електронна комерція стала дуже важливою для багатьох міжнародних організацій і є єдиним способом зберегти та підвищити свою конкурентоспроможність на ринку. Модернізація бізнес-процесів з використанням новітніх інтернет-технологій не тільки створює можливості для розширення бізнесу. Вона стимулює створення нових типів інфраструктури, а також нових видів діяльності та комерційних організацій. Як видно з рис. 1.1, очікується, що роздрібні продажі в цьому секторі зростуть до 2025 року порівняно з 2017 роком. Це свідчить про помітне зростання популярності та визнання електронної комерції як важливого каналу збуту для багатьох підприємств та споживачів.

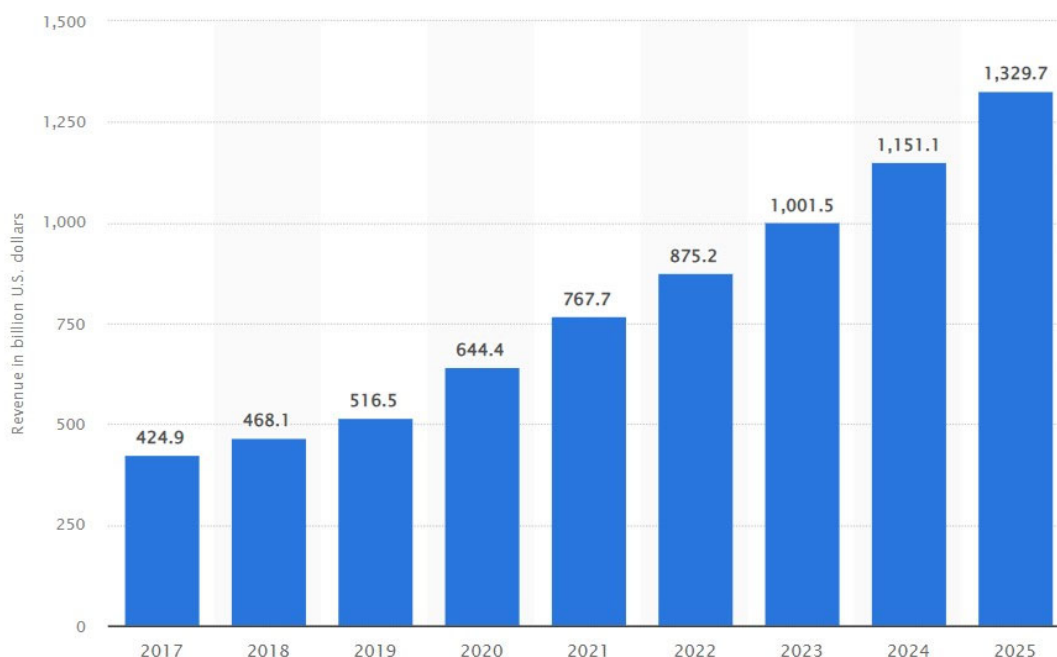


Рисунок 1.1 – Роздрібні продажі засобами е-комерції в глобальному вимірі, млн дол. США

Цифровізація торгівлі призвела до поширення електронної комерції в міжнародних двосторонніх та багатосторонніх угодах. Впровадження цифрових рішень та інтернет-технологій дає нові можливості для поширення ринку клієнтів, дозволяючи компаніям ефективно просувати свої товари та послуги на міжнародному рівні.

Електронна комерція в міжнародній торгівлі стала невід'ємною, динамічно зростаючою сферою, що впливає на стратегії компаній.

Компанії змушені адаптуватися до нових реалій, впроваджуючи інноваційні технології, створюючи електронні платформи, модернізуючи маркетингові підходи та гарантуючи зручність та безпеку для своїх клієнтів.

1.2 Аналіз програмних продуктів-аналогів

Завдяки появі інтернету, а з ним й інтернет-магазинів та електронних платіжних систем у 1990-х роках, виникла та почала розвиватися електронна комерція. З того часу популярність цієї сфери значно зросла, з'явилися численні сервіси електронної комерції. Проаналізуємо їх, відмітимо їх переваги та недоліки.

ROZETKA – заснований у 2004 році як інтернет-магазин парфумерії та косметики, «Розетка» вийшов в онлайн у серпні того ж року. З червня 2005 року він розширив свій асортимент, додавши побутову техніку. У 2019 році «Розетка» трансформувалася в повноцінний маркетплейс, а у 2020 році запустив франшизу. Сьогодні компанія має понад 400 відділень у 122 містах і селищах України, а у 2020 році «Розетка» посіла сьоме місце за відвідуваністю серед українських сайтів.

На сайті дуже зручно зроблені категорії товарів, що дозволяє шукати необхідні товари. Завдяки зручним фільтрам, можна звузити коло пошуку та швидко знайти потрібний товар, та відсортувати його за ціною. Функція відгуків дозволяє користувачам, які вже придбали товар, ділитися своїми враженнями про товар та про продавця. Це дозволяє іншим користувачам краще розуміти характеристики товару та надійність продавця. Система оцінок та відгуків сайту допомагає оцінювати репутацію продавців. Інтерфейс головної сторінки сайту зображено на рис. 1.2.

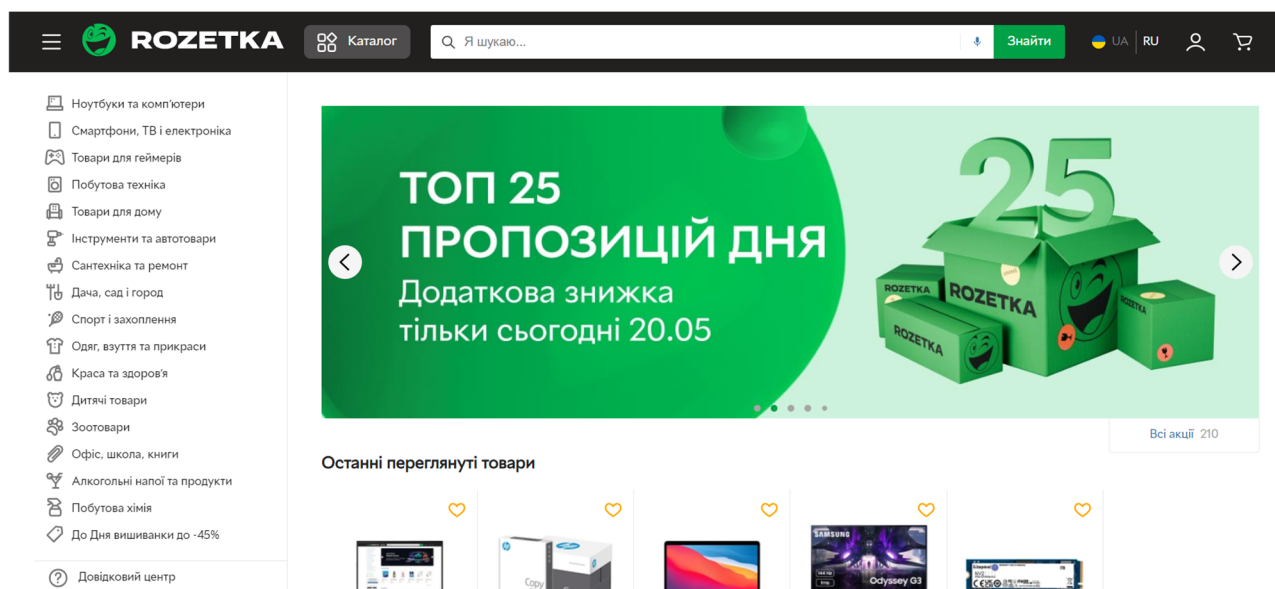


Рисунок 1.2 - Інтерфейс головної сторінки сайту ROZETKA

Переваги сайту ROZETKA:

1. Зручний інтерфейс: дозволяє без проблем знаходити потрібний товар завдяки чіткій структурі сайту та інтуїтивному дизайну.
2. Ефективний пошук та фільтрація дозволяє швидко знаходити потрібні

товари за ціною, характеристиками, брендом та іншими параметрами.

3. Система відгуків та рейтингів - користувачі можуть ділитися своїми враженнями про продавців, залишаючи відгуки, що допомагає іншим користувачам приймати обґрунтовані рішення про покупку того чи іншого товару.

4. Широкий асортимент товарів: ROZETKA пропонує широкий спектр товарів з різних категорій.

5. Адаптивний дизайн: сайт чудово виглядає на будь-якому пристрої, будь то ПК, планшет чи смартфон, забезпечуючи комфортний користувацький досвід.

eBay – це один з найвідоміших онлайн-маркетплейсів, що надає користувачам можливість брати участь в онлайн-аукціонах та створювати власні інтернет-магазини. Платформа має величезну аудиторію - близько 135 мільйонів покупців та 1,7 мільярда активних оголошень по всьому світу. Цей масштаб робить eBay зручним інструментом для ведення електронної комерції та здійснення онлайн-транзакцій для людей з різних країн [3].

eBay пропонує користувачам широкий спектр товарів, охоплюючи безліч категорій, які відповідають різним смакам та потребам. Користувачі можуть легко шукати товари за категоріями, регіонами або за допомогою зручних фільтрів, що робить процес пошуку швидким та ефективним.

eBay пропонує різні варіанти доставки товарів, включаючи як безкоштовні, так і платні. Користувачі мають можливість вибрати найзручніший для них варіант, залежно від своїх потреб та бюджету.

Платформа має систему обміну повідомленнями для покупців і продавців, де вони можуть легко спілкуватися, узгоджувати деталі оплати та доставки, а також вирішувати будь-які питання, що виникають.

eBay надає продавцям доступ до низки корисних інструментів, що допомагають їм успішно розвивати свої магазини. Ці інструменти включають аналітику ринку, можливості просування товарів та інші функції, що сприяють зростанню продажів та розвитку бізнесу.

eBay – це чудовий вибір для тих, хто шукає широкий асортимент товарів, зручний процес покупки та безліч можливостей для продавців.

Інтерфейс домашньої сторінки платформи показаний на рис. 1.3.

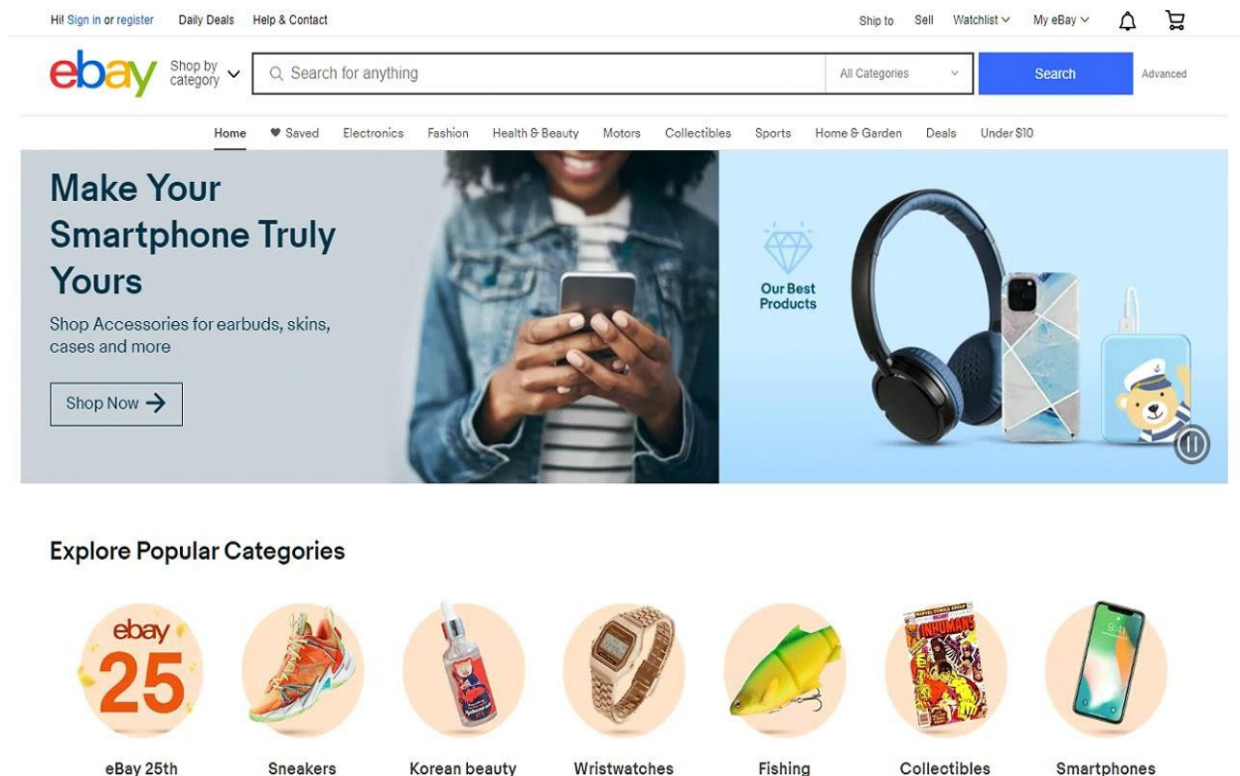


Рисунок 1.3 - Головна сторінка сервісу Ebay

Платформа також пропонує цікавий функціонал аукціону.

На сторінці аукціону користувачі можуть переглянути наступну інформацію:

1. Детальний опис лоту, включаючи технічні характеристики, стан та інші деталі, що дозволяє користувачеві дізнатися всю інформацію про лот до того, як він зробить ставку.
2. Найвищу ставку, яку користувач коли-небудь робив, що дозволяє учасникам торгів відстежувати поточну ціну лота і робити відповідні ставки.
3. Кількість ставок, зроблених на товар - показник зацікавленості користувача в товарі та популярності товару серед учасників торгів.
4. Інформація про час, що показує час, який залишився до кінця аукціону - дозволяє учасникам планувати свої ставки і встигнути купити товар до закінчення аукціону.

На таких аукціонах продавці вказують стартову ціну, а користувачі змагаються між собою, пропонуючи найвищу ціну, яку вони готові заплатити за товар. Коли закінчується встановлений продавцем часовий ліміт, товар отримує

той, хто запропонував найвищу ціну.

Платформа eBaу має низку переваг, завдяки яким вона є популярним вибором для онлайн-покупок:

1. Широкий асортимент товарів: eBaу пропонує широкий спектр товарів, від електроніки та одягу до іграшок, автомобілів, меблів та ювелірних виробів.
2. Різноманітні канали зв'язку: покупці та продавці можуть легко спілкуватися за допомогою різних способів, доступних на платформі.
3. Рейтинги продавців: система рейтингів допомагає користувачам оцінювати надійність продавців і приймати обґрунтовані рішення при покупці.
4. Зручні аукціони: завдяки системі аукціонів на eBaу можна продавати та купувати широкий асортимент товарів, включаючи рідкісні, за доступними цінами.
5. Інструменти для продавців: eBaу надає продавцям корисні інструменти для зручного продажу товарів та аналізу ринку.
6. Технічна підтримка: eBaу пропонує кваліфіковану технічну підтримку як для продавців, так і для покупців.

Платформа eBaу також має певні недоліки:

Складний інтерфейс: інтерфейс здається громіздким і складним для розуміння, що може відлякувати як покупців, так і продавців.

Незручна адаптація до мобільних пристроїв: сайт не дуже добре адаптується до маленьких екранів мобільних пристроїв, що може ускладнити навігацію та використання платформи.

3. Висока комісія: eBaу стягує значну комісію з продажів, що може призвести до значних фінансових витрат для продавців.

4. Недостатня інформація про товари: іноді на платформі може не вистачати детальної інформації про товари, що може призвести до того, що покупці отримають не те, що очікують.

Але, не зважаючи на недоліки, eBaу - один із найбільших і найвідоміших міжнародних онлайн-маркетплейсів, де продавці та покупці з усього світу можуть купувати та продавати різноманітні товари.

Prom (рис. 1.4) – один із провідних українських онлайн-маркетплейсів, що

працює як у B2B (бізнес для бізнесу), так і B2C (бізнес для споживача) сегментах. Платформа пропонує широкий спектр товарів та послуг як для підприємців, так і для приватних осіб. Бізнесмени можуть створювати власні інтернет-магазини на Prom та додавати свої товари до загального каталогу Prom пропонує широкий асортимент категорій і привертає велику кількість відвідувачів (близько 4,8 млн). На сайті діє система рейтингів та відгуків, що допомагає покупцям швидко та легко обирати надійних продавців. Для продавців - можливість швидко відкрити власний інтернет-магазин, залучити нових клієнтів, розширити аудиторію та збільшити продажі. Платформа пропонує набір інструментів для управління замовленнями, організації доставки та оплати, а також аналітичні інструменти для відстеження продажів і поведінки покупців [4].

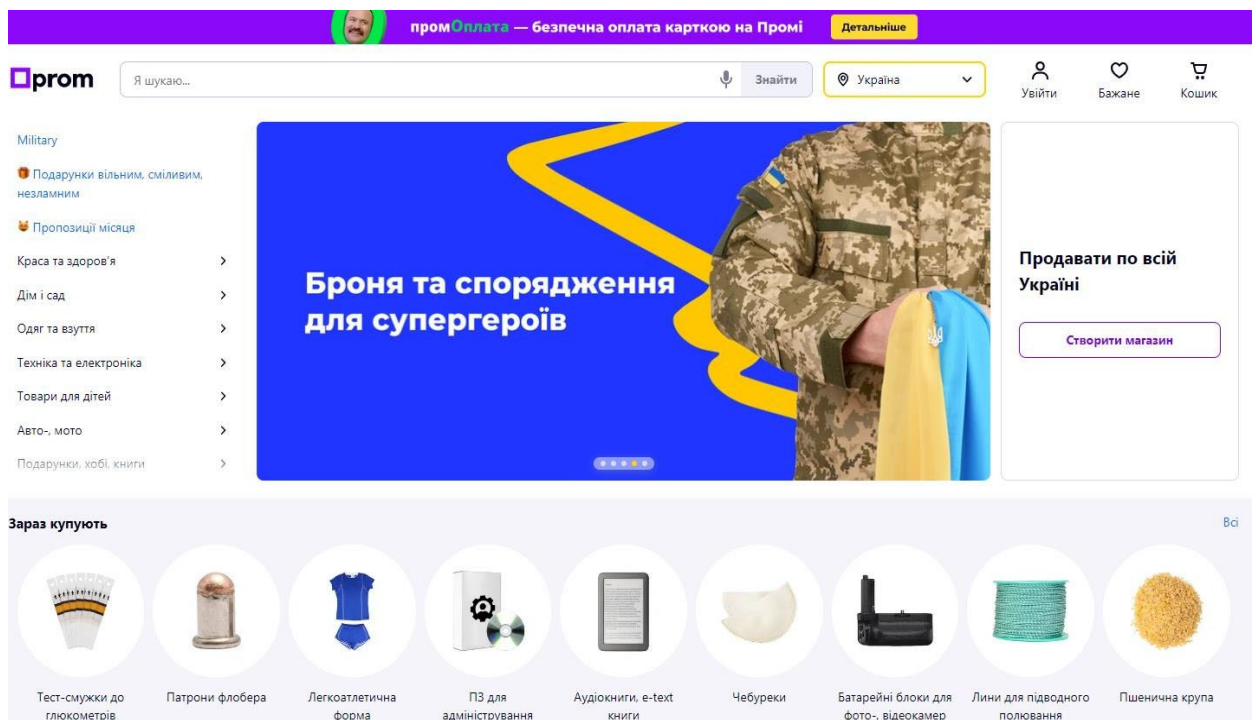


Рисунок 1.4 - Інтерфейс маркетплейсу Prom

Платформа надає детальну інформацію про кожного постачальника, включаючи:

- відсоток позитивних відгуків та рейтинг задоволеності клієнтів - дані допомагають оцінити надійність та відповідальність постачальника;
- інформація про досвід роботи продавця на ринку;
- загальна кількість оцінок - цей показник показує, скільки людей оцінили

роботу постачальника;

- контактні дані продавця: інформація необхідна для зв'язку з продавцем;
- час роботи продавця: ці дані допомагають спланувати спілкування з продавцем;
- інформація про послуги доставки та оплати;
- умови повернення та гарантії: ця інформація важлива для захисту прав покупців;
- короткий опис про продавця, який може допомогти краще зрозуміти характер та спеціалізацію постачальника;
- перелік усіх пропозицій продавця.

Переваги маркетплейсу Prom.ua:

1. Prom.ua має інтуїтивно зрозумілий інтерфейс, що дозволяє легко знайти та купити потрібний товар.
2. Система рейтингів та відгуків допомагає покупцям обирати надійних продавців та уникати шахраїв.
3. Технічна підтримка - продавці отримують своєчасну допомогу та консультації з питань продажів і ведення бізнесу.
4. Зручні інструменти для швидкого налаштування та управління інтернет-магазином.

Недоліки маркетплейсу Prom.ua:

1. Складність платежів: у деяких випадках здійснення платежів на платформі Prom.ua може бути незручним.
2. Високі комісії: Prom.ua стягує з продавців високі комісії за користування платформою, що може негативно вплинути на їх прибуток.
3. Відсутність гарантії: Prom.ua як маркетплейс не надає постійної гарантії на товари, що продаються продавцями, що може створювати ризик невизначеності для покупців.

Для більш зручного аналізу розглянутих інтернет-платформ електронної комерції виконаємо SWOT-аналіз (таблиця 1.1).

Таблиця 1.1 – SWOT-аналіз

Аспекти	Rozetka	eBay	Prom.ua
Сильні сторони	Велика кількість клієнтів, широкий асортимент товарів	Глобальна охопленість ринку. Велика кількість користувачів. Можливість продажу великої кількості товарів	Популярність серед українських споживачів. Широкий вибір товарів. Зручний інтерфейс сайту.
Слабкі сторони	Висока конкуренція на ринку. Проблеми з доставкою в окремих регіонах	Висока комісія для продавців. Ризик шахрайства	Обмежений асортимент товарів. Проблеми з доставкою в окремих регіонах
Можливості	Розширення асортименту товарів. Використання інноваційних технологій	Розширення в нові регіони. Розвиток мобільного додатку	Розширення співпраці з постачальниками. Використання маркетингових акцій.
Загрози	Конкуренція від інших платформ	Конкуренція від інших платформ	Конкуренція від інших платформ

Даний SWOT-аналіз показує, що потрібно розробити онлайн систему електронної комерції, яка дозволить скласти конкуренцію розглянутим платформам. У контексті розробки сайту електронної комерції потрібно створити добре продуманий інтерфейс користувача, зручну навігацію, дружній дизайн тощо.

1.3 Визначення функціональних вимог онлайн системи електронної комерції

Кваліфікаційна робота має на меті розробку вебзастосунку, який буде функціонувати як онлайн сервіс електронної комерції. Для успішного виконання завдань потрібно визначити функціональні вимоги, які покажуть функціональність, яку повинен мати програмний продукт для того, щоб задовольнити потреби користувачів.

Функціональні вимоги онлайн системи електронної комерції повинні включати:

1. Реєстрацію користувачів і управління їх облікових записами.
2. Можливість перегляду товарів, додавання їх до кошика та оформлення замовлення.
3. Опції пошуку товарів за категоріями, ціною, виробником тощо.
4. Інтеграцію різних способів оплати, таких як кредитні картки, електронні

гаманці тощо.

5. Обробку замовлень та надання підтвердження відправлення.
6. Керування запасами товарів та їх оновлення в реальному часі.
7. Надання зворотного зв'язку для користувачів та можливість залишення відгуків про товари і обслуговування.
8. Систему рекомендацій для користувачів на основі їх історії покупок та переглядів.
9. Захист персональних даних користувачів і безпеку транзакцій.
10. Адміністративний доступ для керування системою та звітності.

На основі даних функціональних вимог можна відокремити користувачів онлайн системи – продавці, покупці та адміністратор.

Для зручності представимо функціональні вимоги для кожного користувача системи у вигляді діаграми варіантів використання (рис. 1.5).

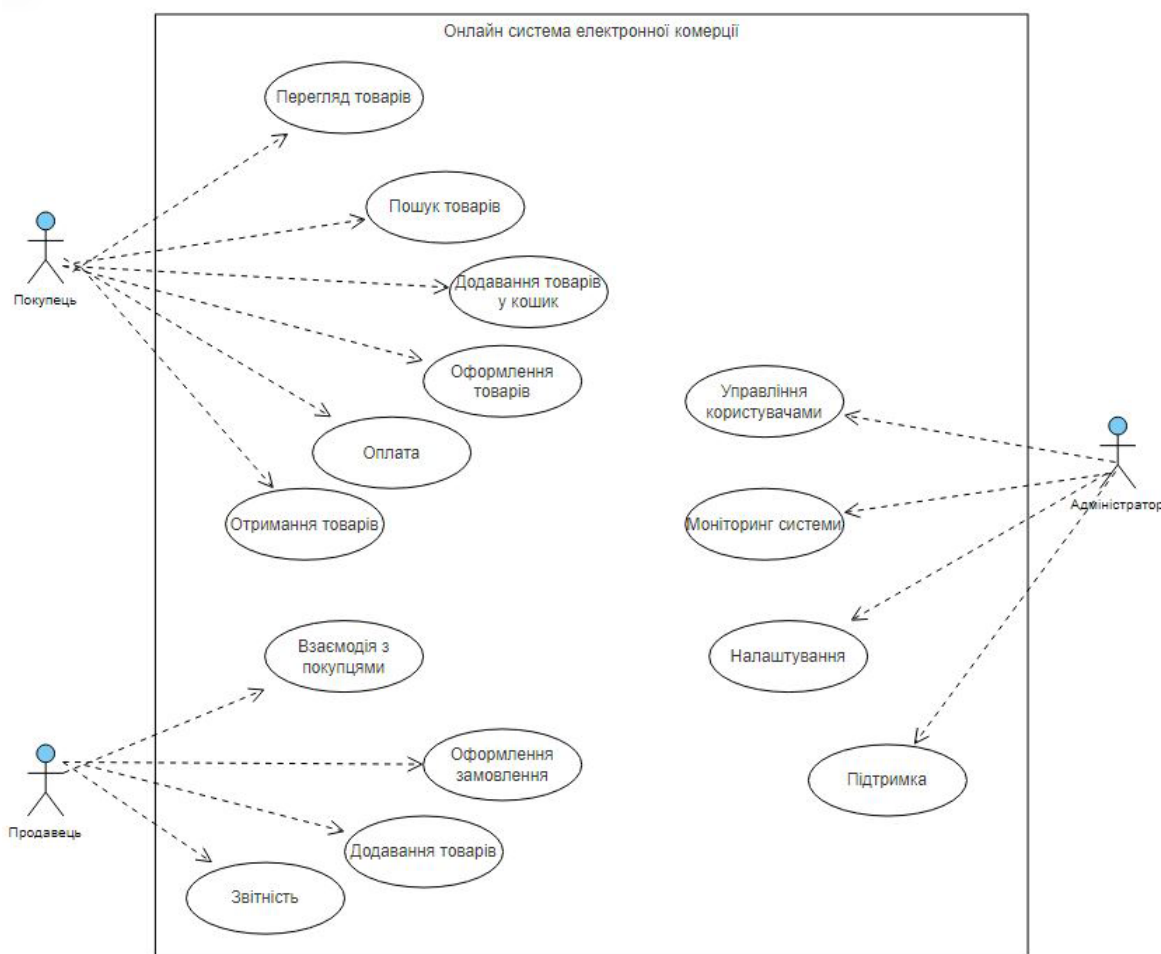


Рисунок 1.5 – Діаграма варіантів використання

Продавці використовують онлайн систему для продажу своїх товарів або послуг, тоді як покупці користуються онлайн системою для придбання товарів або послуг. Адміністратор відповідає за налаштування та керування системою, здійснює контроль за безпекою та відстеженням активності користувачів, встановлює правила та політики користування, розробляє стратегію маркетингу та просування ресурсу, а також забезпечує якість обслуговування користувачів.

1.4 Вибір засобів розробки онлайн системи електронної комерції

Онлайн система електронної комерції буде містити базу даних, яку створимо з використанням Prisma ORM та СКБД PostgreSQL, а також здійснимо міграцію та розгортання спроектованої бази даних.

PostgreSQL – це одна з найпопулярніших та широко використовуваних СКБД. Вона відрізняється:

- високою продуктивністю: PostgreSQL гарантує швидке та ефективне виконання операцій з даними;
- надійністю: PostgreSQL забезпечує збереження та цілісність даних навіть у разі збоїв або помилок;
- масштабованістю: PostgreSQL може підтримувати великі обсяги даних та високі навантаження;

PostgreSQL пропонує широкий спектр функцій, які включають:

- складні операції з даними: підтримує складні запити, агрегації та фільтрації;
- розширені можливості запитів – дозволяє використовувати різні типи запитів, такі як SQL, PL/pgSQL, та інші мови;
- підтримку географічних даних – може зберігати та опрацьовувати географічні координати та дані;
- роботу з JSON – підтримує формат JSON для зручного зберігання та обміну даними;
- підтримку відкритих розширень – має активну спільноту розробників, яка створює та підтримує численні розширення для додаткових функцій.

Відкритість PostgreSQL сприяє активному співробітництву в спільноті розробників та користувачів, що робить цю СКБД динамічною та постійно оновлюваною.

Для серверної частини системи обираємо технології Node.js, Nest.js, TypeScript та Prisma ORM [7-14].

Node.js – це безкоштовна платформа з відкритим кодом, яка використовує JavaScript для виконання серверних програм.

Node.js використовує асинхронну модель програмування, яка дозволяє обробляти запити без блокування сервера. Це робить його ідеальним для створення масштабованих та швидких веб-додатків.

Node.js використовує двигун V8 JavaScript від Chrome для виконання коду на сервері. Це робить його знайомим для веб-розробників, які вже знайомі з JavaScript.

Node.js має велику екосистему пакетів, доступних через npm (Node Package Manager), що дозволяє розробникам легко знаходити та використовувати готові рішення для різних завдань [7].

Зростаюча популярність Node.js робить її оптимальним вибором для розробки серверних застосунків, де важливі фактори продуктивності та масштабованості.

Для створення користувацького інтерфейсу будемо використовувати технології React та TypeScript [15].

Проблема розробки якісного та ергономічного інтерфейсу є важливою в сфері веброботи, оскільки від його якості залежить значна частина взаємодії користувача з онлайн-продуктом і, відповідно, досягнення цілей проєкту. Вебінтерфейс, як ключовий елемент вебзастосунку, визначає спосіб взаємодії користувача з ним та його сприйняття. Якість вебінтерфейсу безпосередньо впливає на задоволеність користувачів, їхню залученість, утримання та конверсію.

Для реалізації інтерфейсу користувача широко використовуються сучасні технології, серед яких виділяються такі:

1. HTML: Основна мова розмітки, призначена для створення структури веб-сторінок та визначення їхньої загальної організації.

2. CSS: Використовується для визначення зовнішнього вигляду веб-сторінок, дозволяючи розділити структуру сторінки та її візуальне оформлення.

3. JavaScript: Мова програмування, що забезпечує можливість створення динамічних елементів на веб-сторінці, включаючи анімацію, взаємодію з користувачем, валідацію форм та інші динамічні функції.

4. UI-бібліотеки: Такі бібліотеки, як Bootstrap, Foundation, Material-UI, забезпечують набір готових компонентів та стилів, що спрощують процес розробки веб-інтерфейсу та дозволяють швидко створювати привабливий і функціональний інтерфейс для користувачів.

Центральним питанням у веброзробці є вибір фреймворку для створення користувацького інтерфейсу, що є ключовим фактором у досягненні успіху проєкту. Ринок пропонує різноманітні фреймворки для розробки інтерфейсу веб-застосунку, серед яких можна виділити декілька основних кандидатів.

Одним із найпопулярніших виборів у цій сфері є фреймворк React. Розроблений компанією Facebook, React є відкритим програмним забезпеченням, яке широко використовується для створення ефективних та масштабованих веб-додатків. В основі його функціоналу лежить принцип віртуального DOM, який дозволяє оновлювати лише необхідні частини сторінки при зміні даних, що сприяє підвищенню продуктивності додатку.

Серед переваг React можна відзначити його ефективність, що базується на використанні віртуального DOM, JSX – розширення синтаксису JavaScript для опису компонентів, а також компонентну архітектуру, яка сприяє повторному використанню компонентів та полегшує супровід коду. Важливим аспектом є також використання одностороннього потоку даних, що спрощує управління станом додатку.

Однак, поряд з перевагами, існують деякі недоліки фреймворку. Висока крива навчання може стати перешкодою для початківців, а велика кількість абстракцій може ускладнити розуміння роботи додатку. Крім того, React активно розвивається, що може призводити до змін у його API та вимагати додаткових зусиль для збереження сумісності.

Angular – це фреймворк з відкритим кодом, розроблений Google, що використовується для створення динамічних односторінкових вебзастосунків (SPA). Він побудований на TypeScript, надмножині JavaScript, що робить код більш чітким та організованим.

Можна виділити основні характеристики Angular:

- компонентно-орієнтований – Angular розбиває інтерфейси користувача на модульні компоненти, що полегшує повторне використання та тестування коду;
- двостороння прив'язка даних – Angular використовує двосторонню прив'язку даних, що синхронізує дані між моделлю та представленням, роблячи код чітким та декларативним;
- директивні – Angular надає директиви – спеціальні атрибути HTML, що розширюють можливості HTML-елементів та спрощують створення UI;
- інструментарій CLI – це інструментарій командного рядка, який допомагає розробникам швидко створювати та керувати проектами Angular;
- TypeScript робить код Angular більш чітким, безпечним та зручним для підтримки завдяки статичній типізації та іншим функціям.

Angular використовується для створення різних вебзастосунків. Angular, попри свої особливості та обмеження, пропонує низку переваг, що роблять його привабливим вибором для розробки веб-застосунків для підприємств та масштабних проектів [11].

Vue.js – це прогресивний JavaScript-фреймворк, який використовується для створення веб-інтерфейсів та односторінкових програм (SPA). Він відомий своєю простотою використання, гнучкістю та елегантною архітектурою [12].

Особливостями Vue.js є:

- простота: Vue.js має простий та інтуїтивно зрозумілий API, що робить його доступним для розробників з різним рівнем досвіду;
- гнучкість: Vue.js не нав'язує жорсткої структури, дозволяючи вам обирати найкращі інструменти та бібліотеки для вашого проекту;
- компонентний підхід: Vue.js розбиває інтерфейс на багаторазові компоненти, що полегшує повторне використання коду та управління складною

логікою;

- реактивність: Vue.js використовує систему реактивних даних, що автоматично оновлює інтерфейс при зміні даних;
- віртуальний DOM: Vue.js використовує віртуальний DOM для оптимізації оновлення інтерфейсу, роблячи його більш продуктивним;
- маленький розмір: Vue.js має невеликий розмір, що робить його ідеальним для мобільних додатків та сайтів з обмеженою пропускнуою здатністю.

Vue.js використовується для створення динамічних та інтерактивних веб-інтерфейсів. Ідеально підходить для розробки SPA, які завантажують контент без перезавантаження сторінки. Vue.js можна використовувати для створення веб-інтерфейсів для мобільних гібридних додатків.

Після ретельного аналізу технологій для розробки клієнтської частини віртуального майданчика електронної комерції, було прийнято рішення вибрати ReactJS, який має наступні переваги:

- компонентна архітектура: React спрощує процес розробки та підтримки завдяки компонентному підходу. Компоненти – це повторювані блоки інтерфейсу, що прискорює розробку та робить проект легко масштабованим;
- React володіє великою кількістю доступних розширень та бібліотек, що дозволяє розширити функціональність та полегшити розробку;
- React має широку спільноту, яка надає доступ до різноманітних ресурсів та підтримки, включаючи документацію, навчальні матеріали та інструменти.

Ці переваги роблять React оптимальним вибором для розробки клієнтської частини віртуального майданчика.

2 ПРОЄКТУВАННЯ ОНЛАЙН СИСТЕМИ ЕЛЕКТРОННОЇ КОМЕРЦІЇ

2.1 Інформаційна модель та база даних онлайн системи електронної комерції

Розробка онлайн системи електронної комерції залежить від інформаційної моделі програмного продукту.

Інформаційна модель створюється для того, щоб описати дані та їх взаємозв'язки, що використовуються в рамках функціонування онлайн системи. Вона включає структуру інформації про товари, клієнтів, замовлення, оплату та інші аспекти роботи інтернет-магазину. Ця модель допомагає впорядкувати дані, забезпечує ефективну взаємодію між різними компонентами системи, а також покращує процеси керування та прийняття рішень.

Сутностями системи, що розробляється, є:

Категорії товарів: ID категорії, назва категорії товарів.

Товари: ID товару категорія, назва товару, ціна, опис.

Користувачі: ID користувача, персональні дані (ім'я, логін, пароль, e-mail, місто, країна), роль, права доступу, історія.

Замовлення: ID замовлення, статус замовлення (створення, обробка, підтвердження, відправлення, отримання), доставка.

Оплата: ID оплати, методи оплати, повернення коштів.

Повідомлення: ID повідомлення, від кого та кому, текст повідомлення, тип, дата.

На рис. 2.1 показано ER-діаграма бази даних, яка була створена для онлайн системи електронної комерції. На її основі в СКБД PostgreSQL за допомогою інструмента pgAdmin була побудована база даних для онлайн системи (рис. 2.2).

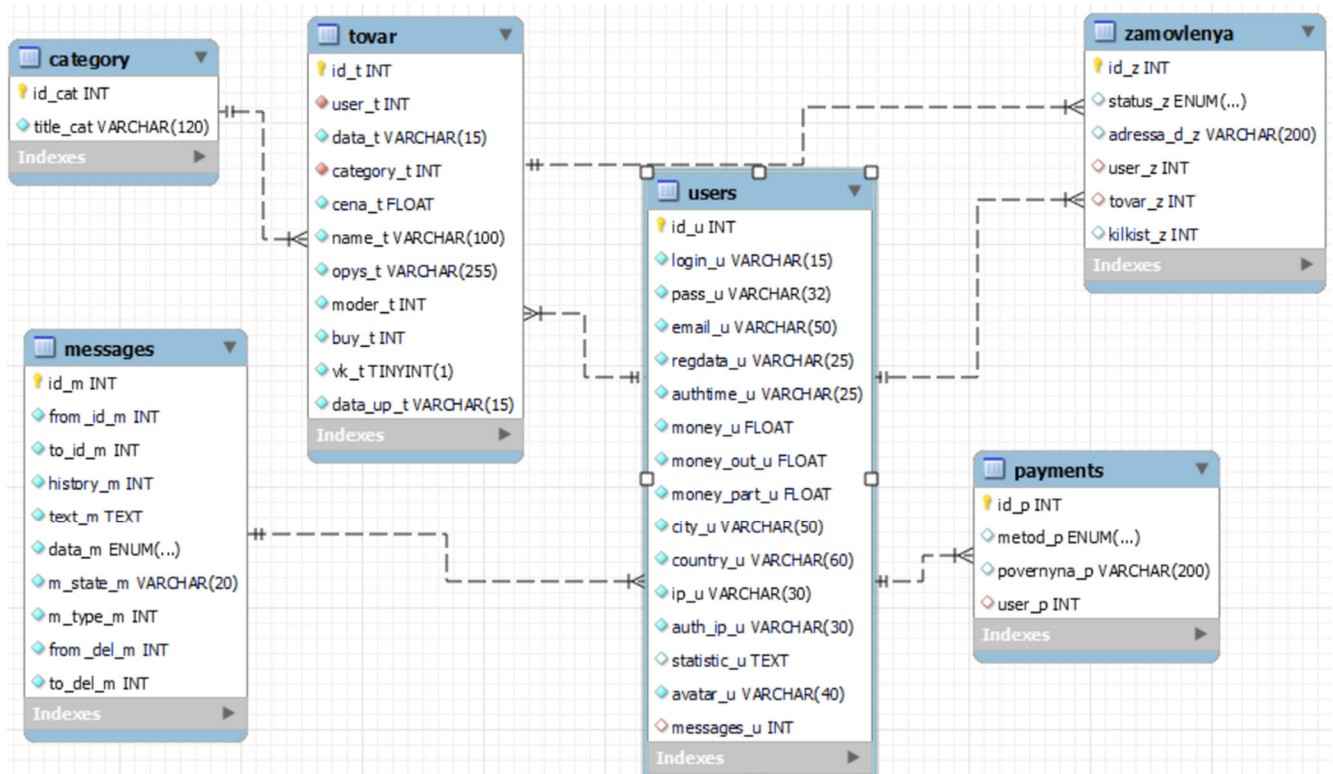


Рисунок 2.1 – ER-діаграма бази даних

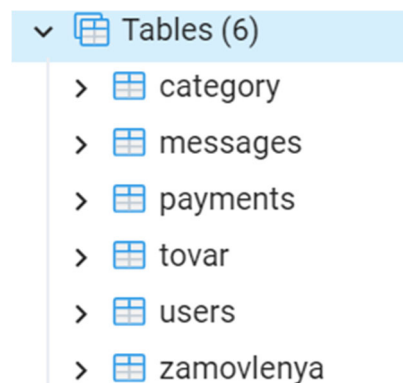


Рисунок 2.2 – Таблиці бази даних

Потоки управління та процеси у системі були представлені у формі інтелектуальної карти (Mind Map), яка відображає послідовність дій учасників. Ментальна карта дозволяє розглядати будь-яку систему у вигляді окремих взаємопов'язаних підсистем.

На рис. 2.3 представлені дії покупця онлайн системи. На рис. 2.4 – дії продавця.

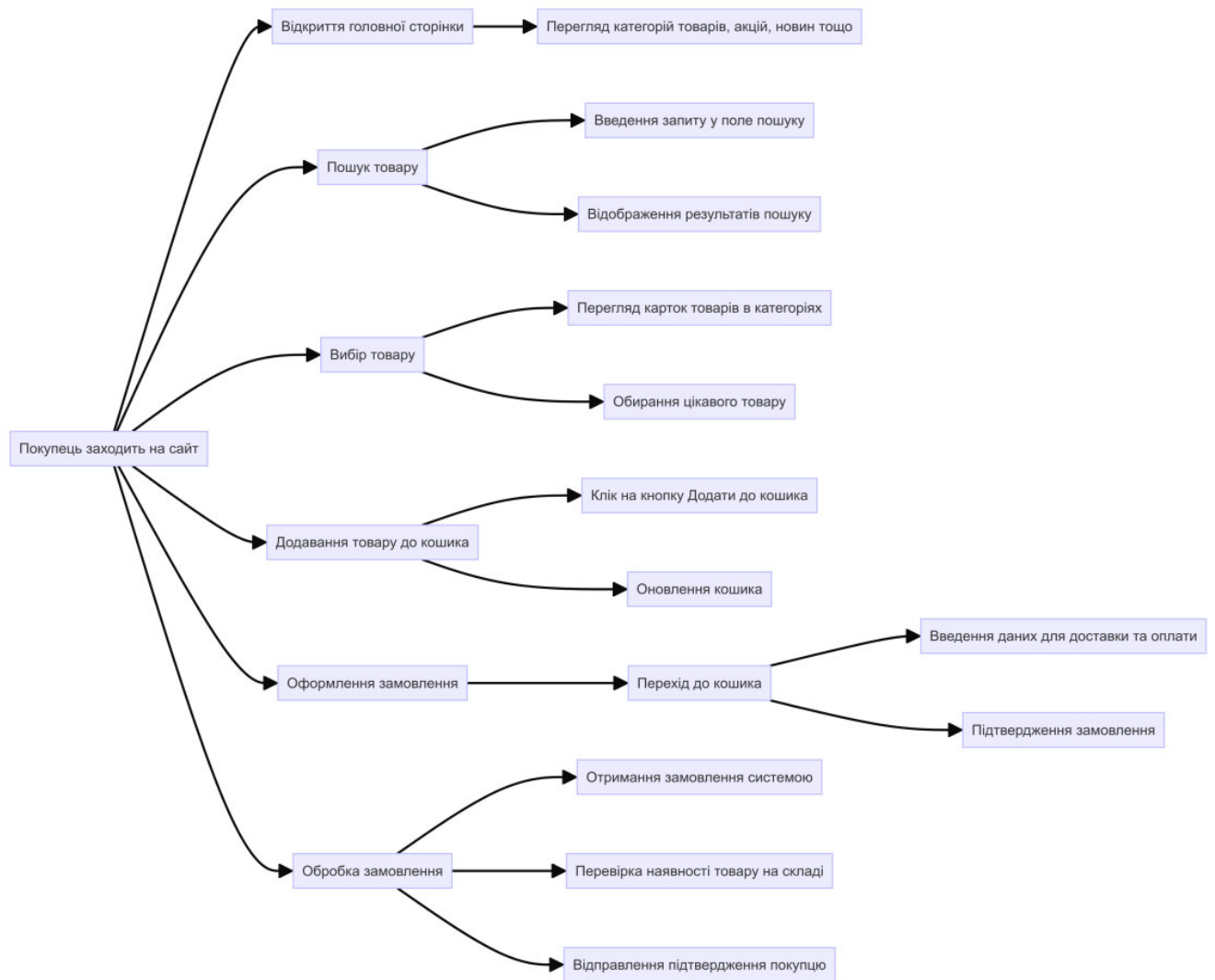


Рисунок 2.3 – Ментальна карта дій покупця онлайн системи

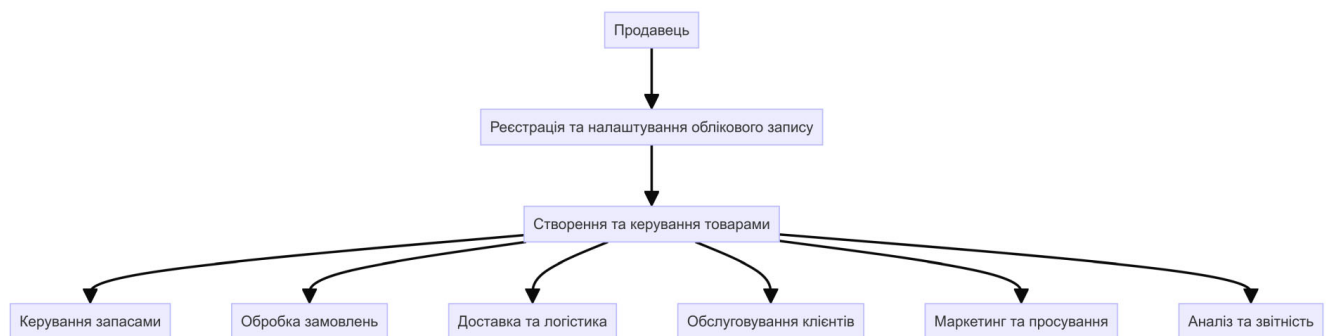


Рисунок 2.4 – Ментальна карта дій продавця онлайн системи

Процес дій продавця включає наступні кроки:

1. Продавці закупають продукти від постачальників або виробників для розміщення в системі.
2. Продавці створюють списки продуктів із детальним описом,

зображеннями та цінами на платформі системи.

3. Продавці керують рівнем своїх запасів, оновлюють запаси та поповнюють запаси за потреби.

4. Продавці отримують і обробляють замовлення від клієнтів, підтверджують платежі та готують замовлення до відправлення.

5. Продавці організовують доставку замовлень, включаючи вибір перевізників, пакування замовлень і відстеження доставки.

6. Продавці відповідають на запити клієнтів, розглядають скарги та надають підтримку, щоб забезпечити позитивний досвід покупки.

7. Продавці можуть займатися маркетинговою діяльністю для просування своїх товарів, наприклад проводити кампанії, пропонувати знижки або брати участь у рекламних акціях.

На рис. 2.5 зображена ментальна карта дій адміністратора онлайн системи.

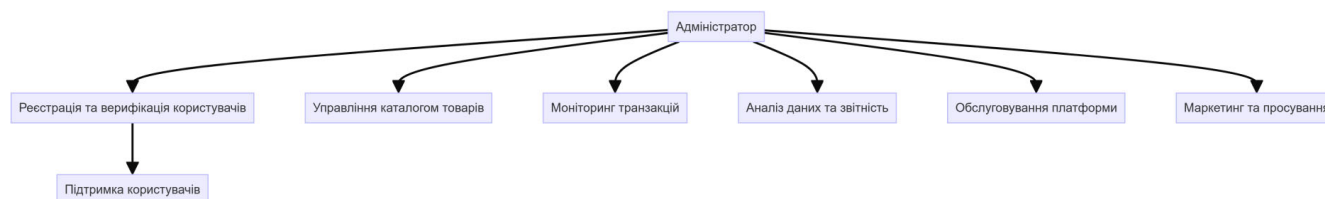


Рисунок 2.5 – Ментальна карта дій адміністратора онлайн системи

Діаграма активності адміністратора онлайн системи електронної комерції зазвичай включає завдання та процеси, пов'язані з керуванням та підтримкою платформи.

2.2 Архітектура онлайн системи електронної комерції

Онлайн система електронної комерції буде побудована на MVC архітектурі (рис. 2.6). Контролер моделі проектування забезпечує структурування програми, розподільчі на три основні компоненти: модель, представлення та контролер.

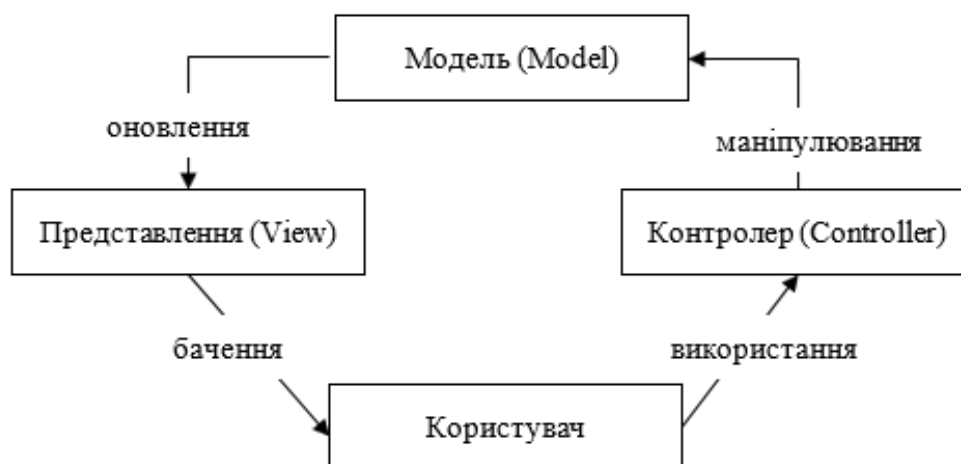


Рисунок 2.6 – MVC-архітектура

Використання архітектури MVC при створенні онлайн системи електронної комерції забезпечує низку переваг.

Архітектура MVC дозволяє забезпечити розділення завдань між компонентами програми. Завдяки цьому, модель буде ефективно керувати даними, представлення відображає інформацію для користувачів, а контролер обробляє запити та координує взаємодію між моделлю та представленням.

Також така архітектура надає можливість організувати багатокористувацький доступ до онлайн-системи. Відокремлюючи логіку бізнесу від інтерфейсу користувача та обробку запитів, кожен компонент може функціонувати незалежно, що забезпечує швидку роботу системи навіть із великою кількістю одночасних користувачів. Це значно підвищує продуктивність та задовольняє потреби користувачів у доступі до системи.

Ще одна перевага такої архітектури – підвищення рівня тестування програми. Шляхом розділення компонентів та їх функціональність тестування кожного з них стає простішим. Модель може бути перевірена на правильність логіки бізнесу, представлення проводиться щодо правильності відображення даних, а також контролера на обробку запитів користувачів. Це дозволяє швидко визначити та виправити помилки, а також забезпечити стабільне застосування програми.

Для створення онлайн системи електронної комерції потрібно відокремити компоненти онлайн системи. До них відносяться:

1. Інтерфейс:

Інтерфейс користувача, якій включає всі видимі елементи, такі як кнопки, меню та списки продуктів, з якими користувач взаємодіє на сайті.

Вебдизайн: загальний вигляд сайту, включаючи макет, кольори, шрифти та зображення.

Швидкість реагування: сайт має бути оптимізовано для різних пристроїв і розмірів екрану, щоб забезпечити бездоганну взаємодію з користувачем.

2. Сервер:

База даних: тут зберігається вся інформація про продукт, дані клієнтів і деталі замовлення.

Сервер: сервер обробляє запити від інтерфейсу, обробляє дані та надсилає відповіді назад користувачеві.

Платіжна система: це дозволяє клієнтам здійснювати безпечні онлайн-платежі за свої покупки.

3. Функції електронної комерції:

Каталог продукції: база даних усіх товарів, які можна придбати на веб-сайті.

Кошик для покупок: дозволяє клієнтам додавати товари до свого кошика для оформлення замовлення.

Обробка замовлень: забезпечує підтвердження замовлення, обробку платежів і відстеження замовлень.

4. Пошукова оптимізація.

Кожен компонент відіграє головну роль у забезпеченні безпечних покупок для користувачів.

3 РОЗРОБКА ТА ТЕСТУВАННЯ ОНЛАЙН СИСТЕМИ ЕЛЕКТРОННОЇ КОМЕРЦІЇ

3.1 Розробка клієнт-серверної частини

У процесі розробки електронно-комерційних сервісів виникають низка складних завдань, які вимагають високого рівня професійної експертизи у сфері програмного забезпечення. Ці сервіси зазвичай формуються у вигляді веб-додатків з використанням передових програмних технологій та методів розробки. Незважаючи на те, що основний набір функцій у таких додатках часто має схожість, він може змінюватися залежно від вимог та цілей. Всі ці сервіси мають досить інтуїтивний інтерфейс користувача, спрощуючи взаємодію з ними.

Електронні магазини пропонують користувачам зручний та багатофункціональний шлях до пошуку та купівлі товарів. Завдяки зручним фільтрам, користувачі можуть легко знайти те, що шукають, сортуючи результати за ціною, датою, категорією або іншими параметрами. Додаткові фільтри, такі як ціновий діапазон, дозволяють ще більш точно налаштувати пошук.

Авторизація в електронних магазинах відкриває доступ до розширених функцій, таких як:

Відгуки та оцінки товарів: користувачі можуть ділитися своїми враженнями про товари, допомагаючи іншим покупцям зробити правильний вибір.

Персоналізовані рекомендації: завдяки аналізу історії покупок, система може рекомендувати товари, які можуть зацікавити користувача.

Програми лояльності та знижки: активні користувачі можуть отримувати бонуси, знижки та інші переваги.

Для продавців електронні магазини надають зручну платформу для розміщення та продажу своїх товарів. Продавці можуть створювати детальні описи продуктів, додавати фотографії, вказувати категорії, адресу та інші характеристики, які допоможуть потенційним покупцям знайти потрібний товар.

Чітка та інформативна презентація продуктів, а також якісні візуальні матеріали, сприяють кращому розумінню пропонованого товару та підвищують

шанси на його успішну продаж.

Серверна сторона (Back-end) – це «мозок» віртуального майданчика. Вона відповідає за:

- обробку запитів від клієнтської сторони: користувачі взаємодіють з майданчиком через інтерфейс, а сервер приймає запити на пошук, фільтрацію, авторизацію, аутентифікацію, валідацію, опрацювання замовлень та інші дії;
- взаємодію з базою даних: сервер управляє базою даних, де зберігається вся інформація про майданчик: оголошення, дані користувачів, транзакції, і т.д. Він забезпечує надійне зберігання, доступ та обробку даних.

Клієнтська сторона (Front-end) – це інтерфейс, з яким взаємодіють користувачі. Клієнтська сторона:

- надсилає запити на сервер: Коли користувач шукає щось, вводить дані або натискає кнопки, клієнтська сторона надсилає запити на сервер;
- відображає інтерфейс: клієнтська сторона отримує відповіді від сервера, оновлює інтерфейс та відображає інформацію користувачу;
- забезпечує зручність: клієнтська сторона робить взаємодію з майданчиком максимально простою та зрозумілою.

Взаємодія між серверною та клієнтською сторонами

Ці дві сторони віртуальної онлайн системи електронної комерції працюють разом, щоб забезпечити користувачам зручний та функціональний досвід. Серверна сторона обробляє запити та дані, а клієнтська сторона відображає інтерфейс та робить спілкування з майданчиком легким та інтуїтивно зрозумілим..

3.2 Реалізація онлайн системи

Для побудови інтерфейс додатка було використано JavaScript для типізації коду. Структура користувацької частини була розбита на дрібні та розділені компоненти.

Структура проекту складається з файлів, що показанні на рис. 3.1:

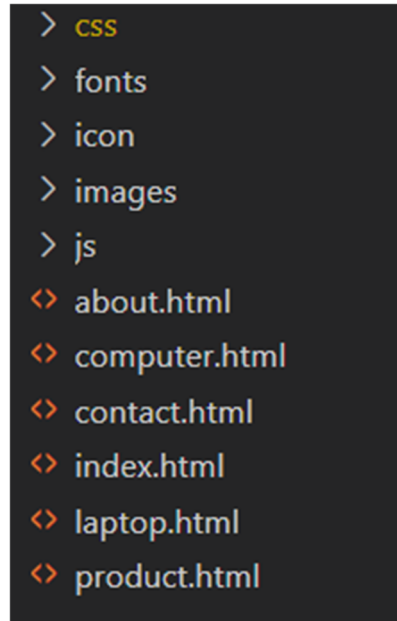


Рисунок 3.1 – Загальна файлова структура

Взаємодія з API сервера є ключовою складовою користувацького інтерфейсу, тому для її реалізації було обрано бібліотеку Axios. Цей інструмент надає зручні методи для виконання HTTP-запитів, дозволяючи легко налаштовувати заголовки, передавати параметри, обробляти відповіді сервера та керувати помилками.

На рис. 3.2 зображено метод з сервісу AdvertService, який виконує запит до API для отримання списку оголошень. Метод використовує передані параметри для визначення відповідного маршруту, що дозволяє фільтрувати та отримувати різні списки оголошень

```

    async getAdverts(dto: GetAdvertsDto):
    Promise<AxiosPromise<IGetAdvertsResponse>> {
        if (dto.categoryId) {
            return
            axiosInstance.get(`/advert/allByCategory/${dto.categoryId}`
        , {
            params: {
                ...dto
            }
        });
        }
        if (dto.subCategoryId) {
            return
            axiosInstance.get(`/advert/allBySubCategory/${dto.subCategoryId}`, {
            params: {
                ...dto
            }
        });
        }
        return axiosInstance.get('/advert/all', {
            params: {
                ...dto
            }
        });
    }
}

```

Рисунок 3.2 – Метод getAdverts

На рис. 3.3 показана реалізація звернення до сервера з запитом на оновлення балансу кредитної картки.

```

    async topUpBalance(token: string | null, id:
    string, body: TopUpMyBalanceDto) {
        return axiosInstance.patch(`/credit-
        card/updateCard/${id}`, body, { headers: { Authorization
        `Bearer ${token}` } });
    },

```

Рисунок 3.3 – Метод topUpBalance

Під час розробки клієнтського застосунку важливо мати централізоване сховище даних, яке полегшує управління станом програми. Для цього я використовую бібліотеку Redux.

Redux пропонує зручний спосіб зберігати стан додатка в одному місці, а також надає набір правил для його оновлення. Це робить Redux потужним інструментом для управління складними станами додатків.

На рис. 3.4 зображено набір правил які в залежності від дії оновлюють стан моделі оголошення.


```

export const advertReducer = (state = initialState, action: AdvertAction): IAdvertState => {
  switch (action.type) {
    case AdvertActionTypes.SET_ADVERTS:
      return {
        ...state,
        currentAdverts: action.payload,
      };
    case AdvertActionTypes.SET_CURRENT_ADVERT: return {
      ...state, currentAdvert:
action.payload.fullAdvert,
      avgReviewRating: action.payload.avgReviewRating
    };
    case AdvertActionTypes.UPDATE_ADVERT: return {
      ...state,
      currentAdvert: action.payload
    };
    case AdvertActionTypes.DELETE_ADVERT: return {
      ...state,
      currentAdverts: {
        adverts: [...state.currentAdverts.adverts.filter(advert => advert.id
!== action.payload)],
        total: state.currentAdverts.total
      }
    };
    default:
      return state;
  }
}

```

Рисунок 3.4 – Набір правил для зміну стану Advert

Цей код описує функцію `advertReducer`, яка приймає два параметри: `state` (початковий стан) та `action` (дія), і повертає оновлений стан об'єкта. За допомогою оператора `switch` функція визначає різні типи дій (`actions`) та відповідні змінні стану.

Кожен блок `case` описує реакцію на певний тип дії, де відбувається оновлення стану моделі оголошень. Оновлення відбувається шляхом заміни поточних оголошень або оголошення на значення, передані через `action.payload`.

Оновлення стану за допомогою різних типів дій

AdvertActionTypes.SET_ADVERTS: У цьому випадку стан оновлюється шляхом заміни поточних оголошень на ті, що передаються через `action.payload`.

AdvertActionTypes.SET_CURRENT_ADVERT: Тут стан оновлюється шляхом заміни поточного оголошення та середнього рейтингу на значення, що передаються через `action.payload`.

AdvertActionTypes.UPDATE_ADVERT: Цей тип дії оновлює стан шляхом заміни поточного оголошення на те, що передається через `action.payload`.

`AdvertActionTypes.DELETE_ADVERT`: У цьому випадку стан оновлюється шляхом видалення оголошення зі списку поточних оголошень. Для цього використовується `advert.id`, яке передається через `action.payload`.

Якщо жоден з визначених типів дій не збігається з поточним, повертається незмінений початковий стан `state`. Такий підхід дозволяє відокремити логіку оновлення стану від інших частин додатка, роблячи код більш модульним та керованим.

Головною перевагою React є створення функціональних компонентів, які є основною одиницею будівництва користувацького інтерфейсу. Розглянемо приклад одного з таких компонентів що наведено нижче (рис. 3.5).

```
export const UserAdvertsPage = () => {
  const { adverts, total } = useTypedSelector(state
=> state.advertReducer.currentAdverts);
  const [searchParams, setSearchParams] = useSearchParams();
  const [currentPage, setCurrentPage] = useState<number>(1);
  const { isReady, getUserAdverts } = useAdvert(); const dispatch = useDispatch();
  const navigate = useNavigate(); useEffect(() => {
    const pageNumber = searchParams.get('page'); setCurrentPage(pageNumber && +pageNumber > 0 ?
+pageNumber : 1);
    dispatch(getUserAdverts({ limit: 7, offset: pageNumber ? (+pageNumber - 1) * 7 : 0 }));
  }, [dispatch, getUserAdverts, searchParams]); const handleAdvertClick = (advertId: string) => {
    navigate(`/advert/${advertId}`);
  }
  > {
  }
  const handlePaginationClick = (pageNumber: number)
  const params = new URLSearchParams(); params.append('page', pageNumber.toString()); setSearchParams(params);
  }
  if (!isReady) { return <Loader />
  }
  return (
    <div className={styles.user_adverts}>
      <div className={styles.user_adverts_body}>
        <h1>Мої Оголошення</h1>
        {adverts?.map((advert) => {
          return (<div key={advert.id} className={styles.user_adverts_items} onClick={() => handleAdvertClick(advert.id)}>
            {advert.images &&
              advert.images[0] &&
                <div
                  className={styles.user_adverts_img_container}>
                    <img src={`${process.env.REACT_APP_BACKEND_URL}/${advert.images[ 0].path}` } alt={`Not load`} /></img>
                  </div>
                    <div className={styles.user_adverts_item_info}>
                      <div className={styles.user_adverts_item_title}>
                        <h2>{advert.title}</h2>
                      </div>
                      <div className={styles.user_adverts_item_location}>
                        <span>{`м. ${advert.location.name}`}</span>
                        <span>{advertPrice(advert)}</span>
                      </div>
                    </div>
                  </div>
                </div>
              </div>
            </div>
          </div>
        )}
        {total > 7 ? <PaginationListComponent currentPage={currentPage} limit={7} total={total} paginate={handlePaginationClick} />
        : <></>}
      </div>
    </div>
  )
}
```

Рисунок 3.5 – Компонент *ImageInput*

Цей код описує компонент `ImageInput`, який використовується для перегляду та завантаження нових зображень.

Компонент має кілька станів, які управляються за допомогою хука `useState`:

`imageUrls`: Масив URL-адрес вже завантажених або обраних користувачем зображень.

`images`: Масив завантажених файлів зображень.

При зміні `imageUrls` або `images` спрацьовує хук `useEffect`, який викликає функцію `onImagesChange` з масивом `images` як аргументом.

Ще один хук `useEffect` запускається при зміні `defaultImageUrls` і використовується для перетворення URL-адрес зображень в файли. Завантажені файли зберігаються в масив `images`, а URL-адреси - в `imageUrls`.

Функція `handleChange` викликається при зміні вмісту поля вибору файлу. Вона перевіряє обраний файл на відповідність обмеженням щодо розміру. Якщо файл відповідає цим обмеженням, він додається до масивів `newUrls` та `newImages`, після чого значення `imageUrls` та `images` оновлюються.

Функція `deleteImage` видаляє зображення з масивів `imageUrls` та `images` за допомогою індексу.

В компоненті відбувається мапінг `imageUrls` для створення блоків зображень. Кожен блок містить зображення з відповідним URL-адресом та кнопку видалення. Нижче блоків зображень розташовується поле вибору файлу, якщо не досягнуто максимальної кількості зображень.

Компонент `UserAdvertsPage`

На рис. 3.6 зображено код компонента `UserAdvertsPage`, який відповідає за відображення сторінки зі списком інформації про всі оголошення користувача.

поточну сторінку оголошень.

Хук `useAdvert` надає методи для отримання оголошень користувача.

Хук `useDispatch` використовується для відправлення дій до `Redux store`.

Хук `useEffect` виконується після монтажу компонента і завантажує оголошення користувача з використанням методу `getUserAdverts`.

Поточна сторінка встановлюється з параметрів `URL`.

Функція `handleAdvertClick` викликається при кліку на оголошення і переадресовує користувача на сторінку конкретного оголошення.

Функція `handlePaginationClick` викликається при кліку на пагінацію та оновлює параметри `URL` для відображення правильної сторінки оголошень.

Якщо оголошення ще не завантажено (`isReady === false`), компонент повертає компонент `Loader` для відображення загрузки.

У протилежному випадку компонент відображає список оголошень користувача. Компонент пагінації (`PaginationListComponent`), якщо загальна кількість оголошень більше ніж 7.

3.3 Тестування онлайн системи електронної комерції

Для перевірки працездатності створеної системи електронної комерції були проведено тестування реєстрації користувачів та роботи з системою: пошуку товарів, додавання товарів у кошик тощо.

Головна сторінка онлайн системи електронної комерції містить головне меню, яке надає всі функції інтернет-магазину (рис. 3.7).

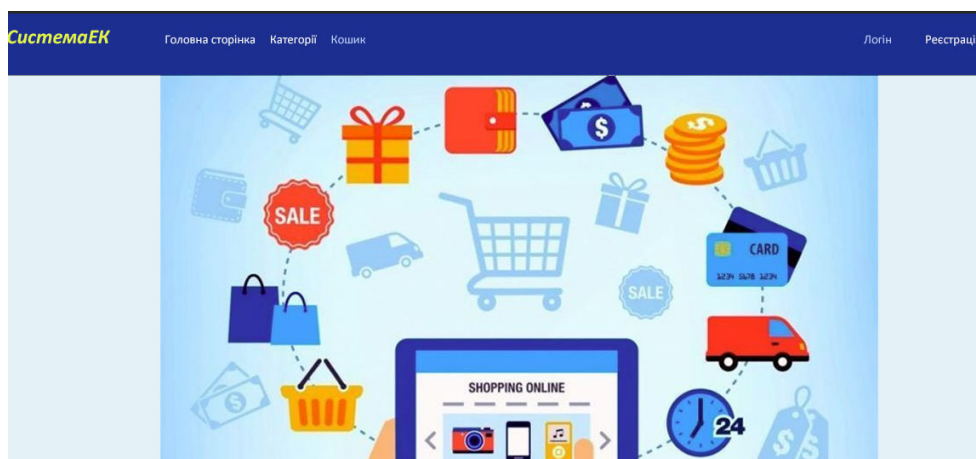


Рисунок 3.7 – Головна сторінка онлайн системи електронної комерції

Якщо вибрати пункт меню *Категорії*, то відкривається список категорій товарів (рис. 3.8).

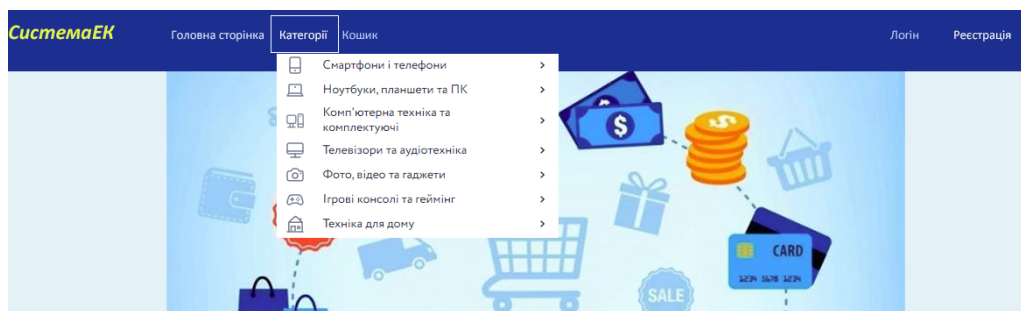


Рисунок 3.8 – Категорії товарів

При виборі певної категорії, відкривається список товарів, що до неї відноситься (рис. 3.9).

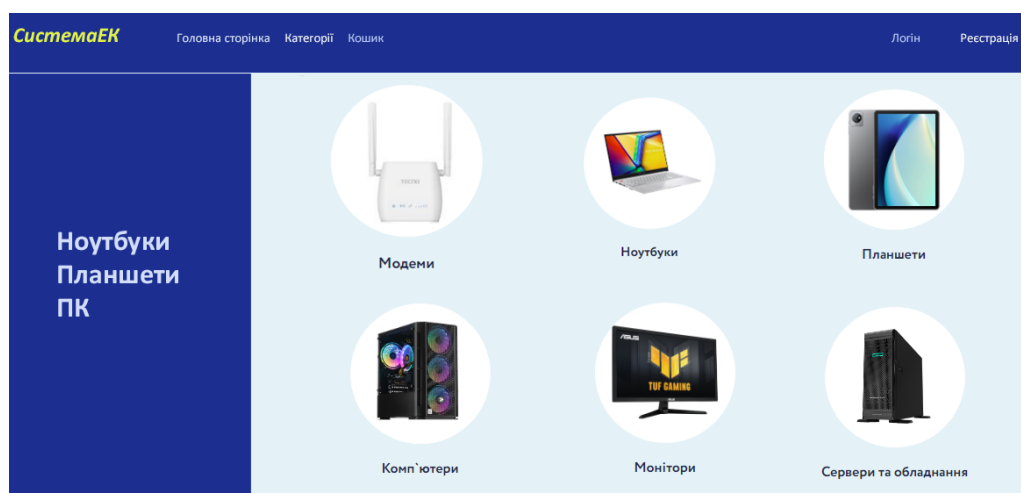


Рисунок 3.9 – Певна категорія

Для реєстрації нового користувача передбачена сторінка реєстрації (рис. 3.10).

Рисунок 3.10 – Реєстрація нового користувача

При виборі товару, він додається у кошик. Переглянути кошик можна використовуючи пункт меню (рис. 3.11).

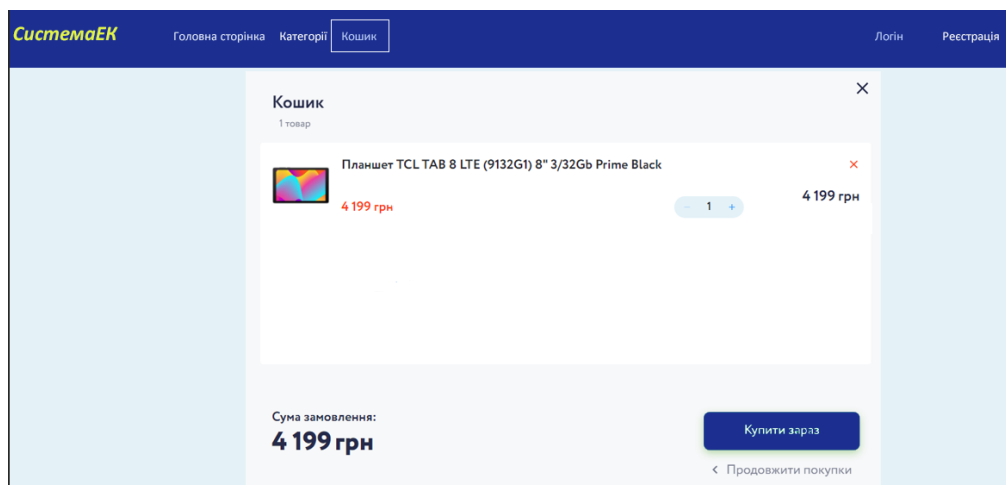


Рисунок 3.11 – Кошик

Товар можна видалити з кошику або продовжити покупки та додавати інші товари.

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було проведено аналіз наукової літератури за темою створення онлайн системи електронної комерції та проаналізовано аналоги, які є на IT-ринку. Це надало глибоке розуміння теоретичних аспектів та практичних реалій створення онлайн системи електронної комерції.

Аналіз доступних інформаційних матеріалів та результатів наукових досліджень підтвердив, що розробники та дослідники активно приділяють увагу IT-технологіям, спрямованим на поліпшення процесів електронної комерції, швидку обробку та передачу інформації, ефективну комунікацію з клієнтами та оптимізацію взаємодії з базами даних. Саме тому метою розробки став вебзастосунок віртуального майданчика торгівлі.

Після проведення аналізу поточного стану проблеми, було складено технічне завдання та розроблено план робіт. Перед початком реалізації було проведено проектування архітектури системи, бази даних та визначено варіанти використання. Розроблена система повністю задовольняє всі встановлені функціональні вимоги.

Реалізована серверна частина, яка забезпечує надійну та ефективну роботу системи. Вона включає в себе розроблені модулі, сервіси та компоненти, що взаємодіють між собою з метою швидкої обробки даних та виконання запитів.

Розроблено інтерфейс користувача, що дозволяє зручно та інтуїтивно взаємодіяти з онлайн системою електронної комерції.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Краус К. М., Краус Н. М., Манжура О. В. Електронна комерція та Інтернет-торгівля : навч.-метод. посібник. Київ : Аграр Медіа Груп, 2021. 454 с.
2. Смолій, Л., & Костюк, В. (2021). НОВІТНІ ТРЕНДИ ТА ПЕРСПЕКТИВИ РОЗВИТКУ ЕЛЕКТРОННОЇ КОМЕРЦІЇ В МІЖНАРОДНОМУ БІЗНЕСІ. Економіка та суспільство, (29). URL: <https://doi.org/10.32782/2524-0072/2021-29-43>.
3. Головна сторінка Ebay : веб-сайт. URL:<https://www.ebay.com/>
4. Головна сторінка Prom : веб-сайт. URL: <https://zp.prom.ua/ua/>
5. Andrew Lock. ASP.NET Core in Action, Second Edition : Manning Publications, 2021, 832 p.
6. Iuliana Cosmina, Rob Harrop, Chris Schaefer, Clarence Ho. Pro Spring 5: An In-Depth Guide to the Spring Framework and Its Tools Fifth Edition: Apress, 2017. 878 p.
7. Mike Cantelon, Marc Harter, T.J. Holowaychuk, and Nathan Rajlich. Node.js in Action. Second Edition : Manning Publications, 2017. 392 p.
8. Miguel Grinberg. Flask Web Development: Developing Web Applications with Python 2nd Edition. O'Reilly Media, 2018. 312p.
9. Regina Obe (Author), Leo Hsu. PostgreSQL: Up and Running: A Practical Guide to the Advanced Open Source Database 3rd Edition. O'Reilly Media, 2017. 312p.
10. Alex Banks and Eve Porcello. Learning React: Functional Web Development with React and Redux 1st edition. O'Reilly, 2017. 350p.
11. Nathan Murray , Felipe Coury , Ari Lerner , Carlos Taborda ng-book: The Complete Guide to Angular 5th Edition. CreateSpace Independent Publishing Platform, 2018. 626p.
12. Andrea Passaglia. Vue.js 2 Cookbook: Build modern, interactive web applications with Vue.js. Packt Publishing, 2017, 454p.
13. Документація до Prisma ORM: веб сайт. URL: <https://www.prisma.io/docs>.
14. Документація до NestJS : веб сайт. URL: <https://docs.nestjs.com>.

- 15.Булигін В.В., студент 4 курсу ІННІ ім. Ю.М. Потебні ЗНУ. Наук. кер.: к.т.н., доц. Міхайлуца О.М. «Використання React та Node.js для створення віртуального майданчика торгівлі та бартерного обміну». Збірник наукових праць студентів, аспірантів і молодих вчених «Молода наука- 2023». Запоріжжя : ЗНУ, 2023. Т.5. С. 78-80.