

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук та технологій
(повне найменування факультету)

Кафедра комп'ютерних систем та мереж
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

магістра

(ступінь вищої освіти)

на тему **ІНТЕРНЕТ-МАГАЗИН ПРОДАЖУ ТЕХНІКИ З ВИКОРИСТАННЯМ
СТРАТЕГІЇ ПЕРСОНАЛІЗАЦІЇ**
(назва теми)

Виконала: студентка 2 курсу, групи КНТ-513м

Спеціальності 123 Комп'ютерна інженерія
(код і найменування спеціальності)

Освітня програма (спеціалізація)

Комп'ютерні системи та мережі

КОНОВАЛОВА Є.А.

(ПРІЗВИЩЕ та ініціали)

Керівник КУДЕРМЕТОВ Р.К.

(ПРІЗВИЩЕ та ініціали)

Рецензент МАЛИЙ О.Ю.

(ПРІЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет Комп'ютерних наук і технологій
Кафедра «Комп'ютерні системи та мережі»
Ступінь вищої освіти магістерський
Спеціальність 123 Комп'ютерна інженерія
(код і найменування)
Освітня програма (спеціалізація) «Комп'ютерні системи та мережі»
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Зав. кафедри Кудерметов Р.К.

“ ” 2024 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТКИ

КОНОВАЛОВОЇ ЄВИ АНДРІЇВНИ

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Інтернет-магазин продажу техніки з використанням стратегії персоналізації

керівник проєкту (роботи) к. т. н., доцент, КУДЕРМЕТОВ Равіль Камілович,
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від « 18 » жовтня 2024 року № 149

2. Строк подання студентом проєкту (роботи) 16 грудня 2024 року

3. Вихідні дані до проєкту (роботи) база даних магазину з продажу техніки в форматі .xls у вигляді переліку техніки, комплектуючих, ціни на кожен товар та його кількості у наявності.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1) Аналіз предметної області;

2) Моделювання програмного забезпечення;

3) Проєктування системи;

4) Реалізація системи.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)

Презентація

6. Консультанти розділів проєкту (роботи)

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4	КУДЕРМЕТОВ Р.К., доцент		
нормоконтроль	ЩЕРБАК Н.В., ст. викл.		

7. Дата видачі завдання « 9 » _____ вересня 2024 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Аналіз технічного завдання: дослідження предметної області, збір даних для персоналізації	10.10.2024 р.	
2	Розробка моделі програмного забезпечення: створення UML-діаграм та сценаріїв використання	20.10.2024 р.	
3	Проектування бази даних: структура, моделювання зв'язків між таблицями	01.11.2024 р.	
4	Проектування інтерфейсу користувача: ескізи сторінок, адаптивний дизайн	10.11.2024 р.	
5	Реалізація вебресурсу: налаштування OpenCart, інтеграція модулів TensorFlow	20.11.2024 р.	
6	Оформлення пояснювальної записки, підготовка презентації	12.12.2024 р.	

Студентка

_____ (підпис)

Єва КОНОВАЛОВА

(Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)

_____ (підпис)

Равіль КУДЕРМЕТОВ

(Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

ПЗ: 77 стор., 13 табл., 49 рис., 18 джерел.

CSS, HTML, MYSQL, OPENCART, PHP, TENSORFLOW, OPENCART,
ВЕБРЕСУРС, ІНТЕРНЕТ-МАГАЗИН, СЕРВЕР, СУБД

Об'єкт розробки – інтернет-магазин техніки та комплектуючих.

Мета роботи – оптимізувати фінансові та часові ресурси компанії з продажу техніки та комплектуючих за рахунок розробки програмного забезпечення для інтернет-магазину. Основний акцент роботи зроблено на впровадженні персоналізованих рекомендаційних систем, адаптивного користувацького інтерфейсу та забезпеченні високого рівня функціональної зручності.

Проєкт складається з чотирьох частин. У першому розділі проведено аналіз предметної області, включаючи дослідження особливостей онлайн-торгівлі технікою. Розглянуто існуючі аналоги інтернет-магазинів, їх переваги та недоліки, а також сформовано технічні та функціональні вимоги до програмного забезпечення.

У другому розділі здійснено моделювання програмного забезпечення. Побудовано UML-діаграми прецедентів, які відображають сценарії використання системи. Також розроблено діаграми послідовності для ключових процесів.

У третьому розділі виконано проєктування бази даних та інтерфейсу користувача. Визначено структуру даних, зв'язки між таблицями, а також створено ескізи основних сторінок для забезпечення зручності навігації. Обґрунтовано вибір СУБД та інших технологій, необхідних для реалізації.

У четвертому розділі описано процес реалізації інтернет-магазину. Деталізовано етапи налаштування системи на основі платформи OpenCart, підключення Google Analytics і інтеграції модуля TensorFlow для персоналізації товарних рекомендацій.

ABSTRACT

Explanatory note to the master's work: 77 p., 13 tables, 49 figures, 18 sources.

PHP, MYSQL, TENSORFLOW, OPENCART, HTML, CSS, ONLINE STORE,
DATABASE MANAGEMENT SYSTEM (DBMS), WEB RESOURCE, SERVER

Development Object – online store for selling equipment and components.

Objective – optimize the financial and time resources of a company engaged in selling equipment and components by developing software for an online store. The primary focus is on implementing personalized recommendation systems, an adaptive user interface, and ensuring a high level of functional convenience.

The project consists of four parts. First Section analyzes the subject area, including the specifics of online equipment trading. Existing online store analogs are reviewed, highlighting their advantages and disadvantages, and technical and functional requirements for the software are defined.

Second Section focuses on software modeling. UML use-case diagrams illustrating system usage scenarios are developed, along with sequence diagrams for key processes.

Third Section covers database and user interface design. The structure of the data, relationships between tables, and sketches of the main pages for easy navigation are defined. The choice of DBMS and other technologies necessary for implementation is justified.

Fourth Section describes the process of implementing the online store. It details the stages of configuring the system based on the OpenCart platform, integrating Google Analytics, and embedding the TensorFlow module for personalized product recommendations.

ЗМІСТ

Перелік скорочень	
Вступ	9
1 Аналіз предметної області	11
1.1 Опис предметної області	11
1.2 Порівняння існуючих аналогів	12
1.3 Визначення вимог до програмного забезпечення	20
1.3.1 Існуючі і самостійні розроблені системи	20
1.3.2 Аналіз систем управління для інтернет-магазину	22
1.3.3 Опис технічних вимог та характеристик	25
1.3.4 Обґрунтування вибору інструментів для створення інтернет-магазину	26
2 Моделювання програмного забезпечення	28
2.1 Моделювання прецедентів та опис акторів	28
2.2 Моделювання діаграм послідовності	29
3 Проєктування системи	35
3.1 Архітектура програмного забезпечення	35
3.2 Проєктування бази даних	36
3.2.1 Обґрунтування і вибір СУБД	37
3.2.2 Концептуальне проєктування бази даних	39
3.2.3 Побудова логічної моделі бази даних	45
3.3 Проєктування користувацького інтерфейсу	48
4 Реалізація системи	53
4.1 Вибір інструментів реалізації	53
4.2 Опис основних етапів технологічного циклу	56
4.3 Встановлення та налаштування системи «OpenCart»	57
4.4 Розробка бази даних	59
4.5 Під'єднання Google Analytics до вебресурсу	63
4.5 Інтеграція TensorFlow для персоналізації інтернет-магазину	66

4.6 Реалізація користувацького інтерфейсу	68
4.7 Програмна реалізація вебресурсу	71
4.8 Ефекти персоналізації в інтернет магазині	74
Висновки	75
Перелік джерел посилання	76

ПЕРЕЛІК СКОРОЧЕНЬ

PHP – Personal Home Page Tools;

SQL – Structured Query Language;

CSS – Cascading Style Sheets;

HTML – Hyper Text Markup Language;

СУБД – Система управління базами даних;

UML – Unified Modeling Language;

MVC – Model-View-Controller;

ER – Entity-Relationship;

CMS – Content Management System;

SEO – Search Engine Optimization.

ВСТУП

Тема інтернет-магазинів є особливо актуальною в сучасному світі, оскільки мільйони людей щоденно здійснюють покупки онлайн, не виходячи з дому. Із швидким зростанням кількості користувачів Інтернету, відповідно збільшується і кількість потенційних покупців.

Електронна комерція, як форма продажу товарів через мережу, стала однією з найбільш перспективних сфер для підприємців, дозволяючи їм скоротити витрати на фізичні магазини і збільшити доступ до аудиторії. Електронна комерція – це така форма постачання продукції, при якій вибір і замовлення товарів здійснюється через комп'ютерні мережі, а розрахунки між покупцем і постачальником здійснюються з використанням електронних документів і/або засобів платежу [1].

Для України на сучасному етапі, онлайн-магазини мають значні переваги перед фізичними, зокрема через економічні, соціальні та технологічні фактори. Такий вид комерції дозволяє уникнути більшості витрат на оренду приміщень, оплату комунальних послуг, зарплати співробітникам та інше, знижуючи бар'єри для входу в бізнес, що особливо важливо в умовах економічної нестабільності та фінансових труднощів. Також однією з переваг можливість доступу покупцям з будь-якого регіону отримати доступ до продукції, навіть якщо вони живуть у віддалених або тимчасово окупованих територіях. В умовах військових загроз та бойових дій, це рішення безпосередньо забезпечує безпеку як для власників магазину, так і для покупців, зменшуючи необхідність фізичної присутності у магазинах.

В умовах глобальної конкуренції Інтернет-магазини повинні не лише забезпечувати якісний товар і зручний інтерфейс, але й використовувати маркетингові інструменти, один з яких є персоналізована реклама, що дає можливість використовувати дані про поведінку користувачів для формування індивідуальних пропозицій, що значно підвищує ефективність продажів.

91% клієнтів висловлюють думку, що вони надають перевагу купувати у тих брендів, які діляться рекомендаціями та пропонують релевантні для них пропозиції [2]. Використання технологій відстеження активності клієнтів і алгоритмів машинного навчання дозволяє створювати рекламні стратегії, які допомагають підвищити рівень залученості клієнтів. Це не тільки покращує досвід використання клієнтів, але й дозволяє підприємцям досягати більш високих показників конверсії та збільшувати свої прибутки.

Таким чином, інтернет-магазини є не лише економічно вигідними, але й більш безпечними, зручними та ефективними для бізнесу та покупців, а використання принципів персоналізованої реклами ще й допомагає підвищити продажі та залучити нових клієнтів. Тому дана тема дипломної роботи є актуальною.

Мета роботи – оптимізувати фінансові та часові ресурси компанії з продажу техніки та комплектуючих за рахунок розробки програмного забезпечення для інтернет-магазину.

Для досягнення поставленої мети необхідно виконати такі завдання:

- проаналізувати предметну область, визначивши технічні та функціональні вимоги до розроблюваної системи;
- провести моделювання програмного забезпечення;
- спроектувати базу даних та користувацький інтерфейс;
- розробити модуль персоналізації;
- реалізувати інтернет-магазин.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Опис предметної області

Сучасні інтернет-магазини продажу техніки все більше використовують стратегії персоналізації для підвищення рівня задоволеності клієнтів і збільшення продажів. Персоналізація в онлайн-торгівлі полягає в адаптації контенту, пропозицій і функціоналу магазину під потреби, уподобання та поведінку конкретного користувача. Впровадження такої стратегії дозволяє створювати індивідуальні пропозиції для кожного клієнта, що сприяє підвищенню лояльності та покращенню користувацького досвіду.

Інтернет-магазин техніки, використовуючи персоналізацію, може рекомендувати товари, що відповідають інтересам клієнта, на основі його попередніх покупок або переглядів. Це включає динамічне налаштування головної сторінки, рекомендації продуктів, персоналізовані рекламні повідомлення та спеціальні пропозиції для окремих груп користувачів. Крім того, персоналізація може охоплювати налаштування інтерфейсу магазину, фільтрів для пошуку товарів та індивідуальні варіанти оплати й доставки.

Використання стратегії персоналізації в інтернет-магазині техніки має низку переваг:

- дозволяє скоротити час на пошук потрібного товару та підвищити конверсію завдяки точнішим рекомендаціям;
- знижуються витрати на рекламу за рахунок більш точного алгоритму рекомендації товару;
- дозволяє онлайн-магазину збирати й аналізувати дані про поведінку клієнтів, їхні уподобання та запити, що в свою чергу допомагає удосконалювати бізнес-стратегію та підвищувати якість пропозицій;
- підвищується рівень довіри з боку клієнтів, що позитивно впливає на їх бажання здійснювати повторні покупки.

Проте впровадження персоналізації потребує використання сучасних технологій обробки даних, таких як машинне навчання, аналітика великих даних та автоматизовані системи рекомендацій. Такі рішення допомагають аналізувати поведінку користувачів, їхні інтереси, історію покупок та інші важливі фактори для створення максимально індивідуалізованого досвіду.

Однак, попри численні переваги, персоналізована реклама має й певні недоліки. Насамперед, це стосується питань конфіденційності даних, адже для персоналізації необхідно збирати та обробляти особисту інформацію користувачів, що може викликати занепокоєння і юридичні проблеми. Також складність реалізації персоналізованої реклами потребує значних фінансових і технічних ресурсів, оскільки використання сучасних технологій є витратним і вимагає постійної оптимізації. Окрім того, складні системи персоналізації можуть бути вразливими до технічних збоїв або кібератак. Втрата даних або порушення роботи алгоритмів може призвести до серйозних наслідків для бізнесу.

Отже, якщо компанія створює свій онлайн магазин с функцією персоналізованої реклами, це дає змогу зробити свій асортимент більш доступним для широкої аудиторії та значно підвищити ефективність продажів.

Під час реалізації проєкту потрібно розробити інтернет-магазин для продажу комп'ютерної техніки та комплектуючих до нього, у якому має бути представлений необхідний асортимент товарів, зручний інтерфейс і стратегія персоналізованої реклами.

1.2 Порівняння існуючих аналогів

З метою визначення необхідних функціональних можливостей для розробки інтернет-магазину проведемо аналіз двох існуючих онлайн-платформ, що спеціалізуються на продажі техніки:

- ТТТ.ua – інтернет-магазин гаджетів та аксесуарів до них [3];

- МТА.ua – інтернет-магазин цифрової техніки [4].

ТТТ.ua (Територія Твоєї Техніки) – це компанія, яка займається реалізацією мобільних телефонів та електроніки в Україні. Він пропонує широкий вибір товарів, включаючи смартфони, ноутбуки, планшети, телевізори, фотоапарати, і багато іншого.

На цьому сайті користувачі можуть:

- переглянути великий вибір електроніки та мобільних пристроїв;
- прочитати відгуки та оцінки продуктів;
- порівняти ціни на схожі продукти;
- переглянути технічні характеристики та особливості продуктів;
- зареєструватися на сайті;
- оформити замовлення онлайн або в магазині;
- відстежувати свої замовлення;
- зареєструватися для створення облікового запису клієнта;
- переглянути історію своїх замовлень;
- зв'язатися з службою підтримки клієнтів для отримання допомоги.

На головній сторінці сайту ttt.ua знаходяться такі елементи:

- верхнє меню з вибором мов (українська, російська), контактна інформація, кнопки для входу в особистий кабінет та реєстрації;
- каталог продукції в якому представлено різні категорії товарів, зокрема смартфони, планшети, телевізори, ноутбуки, аксесуари, побутова техніка, електротранспорт тощо;
- банери, що пропонують акції, знижки та нові надходження;
- зручний пошук по сайту та кошик для перегляду товарів;
- функція збереження товару, що сподобався, для подальшого остаточного рішення щодо покупки;
- фонові реклами, які зображені на банерах, які можна персоналізувати під інтереси користувача, базуючись на його поведінці на сайті;
- стисла інформація про устрій, принципи та політику інтернет-магазину.

Загальний вигляд головної сторінки сайту ТТТ.ua зображено на рисунку 1.1.

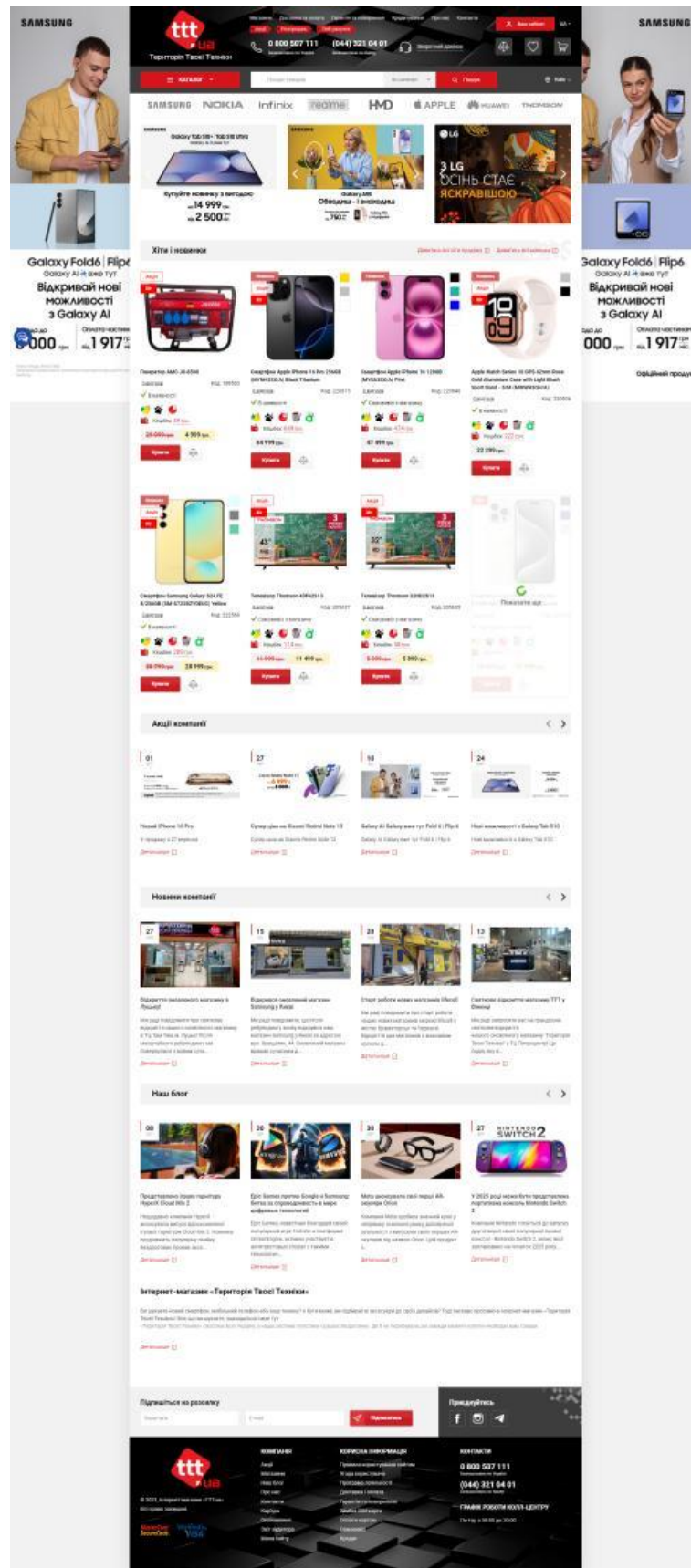


Рисунок 1.1 – Головна сторінка інтернет-магазину ТТТ.ua

Якщо натиснути на банер з рекламою на головній сторінці сайту ttu.ua, відбудеться перенаправлення на сторінку, пов'язану з рекламованим продуктом або акцією.

На час виконання аналізу, на сторінці рекламується нова серія смартфонів Samsung Galaxy Fold 6 і Galaxy Flip 6. Банери на сайті акцентують увагу на інноваційних функціях складних екранів цих моделей, їх новий дизайн та можливість попереднього замовлення. Також виділені спеціальні пропозиції для покупців, включаючи знижки та бонуси при купівлі.

Загальний вигляд сторінки, на яку перенаправляють під час натискання на банер, зображена на рисунку 1.2.

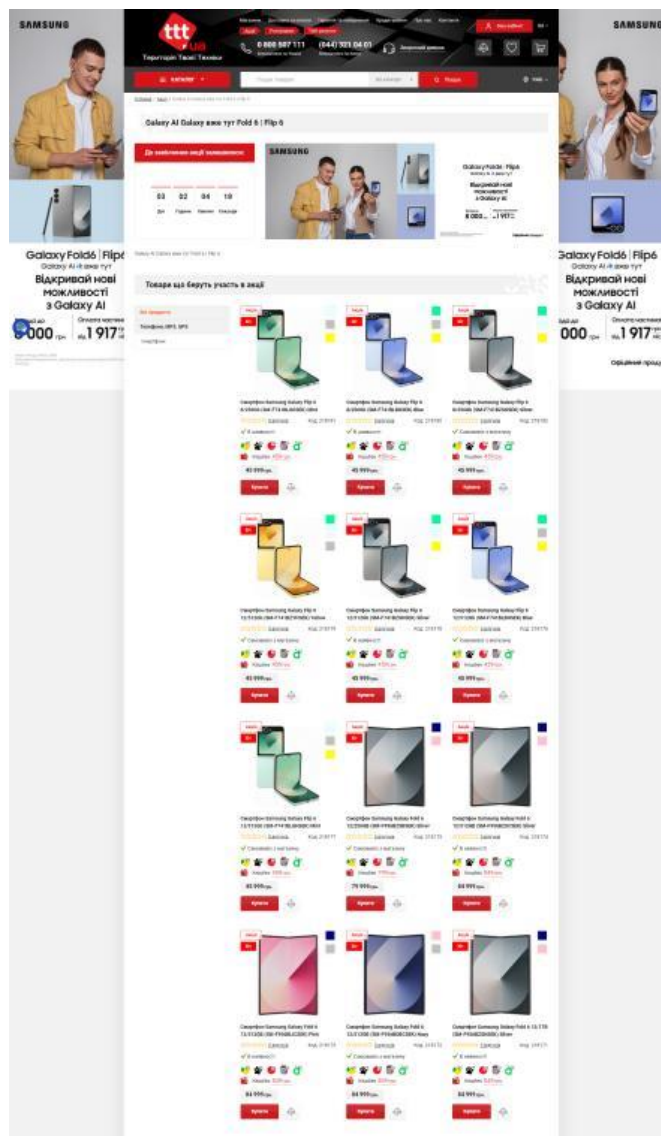


Рисунок 1.2 – Загальний вигляд сторінки рекламної пропозиції на сайті ТТТ.ua

Переходимо до розгляду наступного сайту – МТА.ua.

МТА.ua – це національний, український інтернет-магазин персональної цифрової техніки, яка має мережу магазинів і сервісних центрів. Істотно від попередньо розглянутого сайту, цей не має багато відмінностей, але далі приведено основні з них:

- мінімалістичний дизайн з акцентом на великих зображеннях товарів, рекламні банери вбудовані в головну сторінку, як всі інші товари;
- акцент на ексклюзивних розпродажах і кешбеках, таких як обмежені у часі розпродажі, кешбек-програми та інші вигідні акції, це, в свою чергу, дозволяє залучати більш широке коло клієнтів;
- товари на головній сторінці вже відсортовані за рекламними пропозиціями та акціями, що допомагає швидше знаходити товари зі знижками чи іншими вигідними умовами;
- інтеграція відгуків покупців прямо на сторінці товарів та сервісу, бо наявність відгуків додає прозорості процесу покупок та допомагає потенційним клієнтам прийняти рішення на основі реального досвіду інших користувачів, що в свою чергу підвищує рівень довіри до сервісу;
- перелік фізичних магазинів з їх графіками роботи та адресами у містах де вони розташовані фізичні філіали цього сервісу для тих, кому важливо заздалегідь ознайомитися з товаром;
- чат-бот, який може присилати рекламні пропозиції сайту, щоб потенційні покупці не пропустили важливі оновлення та вигідні пропозиції від магазину.

Залежно від рекламної пропозиції, яку користувач обирає на головній сторінці або в бічній панелі, зміст сторінки зміниться відповідно до обраної акції. Це може включати спеціальні знижки на певні категорії товарів, ексклюзивні пропозиції для обмеженого часу, або навіть персоналізовані рекомендації, засновані на попередніх покупках користувача.

Наприклад, реклама благодійної акції, де можна сдати стару, непотрібну техніку, яку відремонтують та подарують дітям (рисунки 1.3), або розпродаж товарів минулих сезонів (рисунки 1.4).

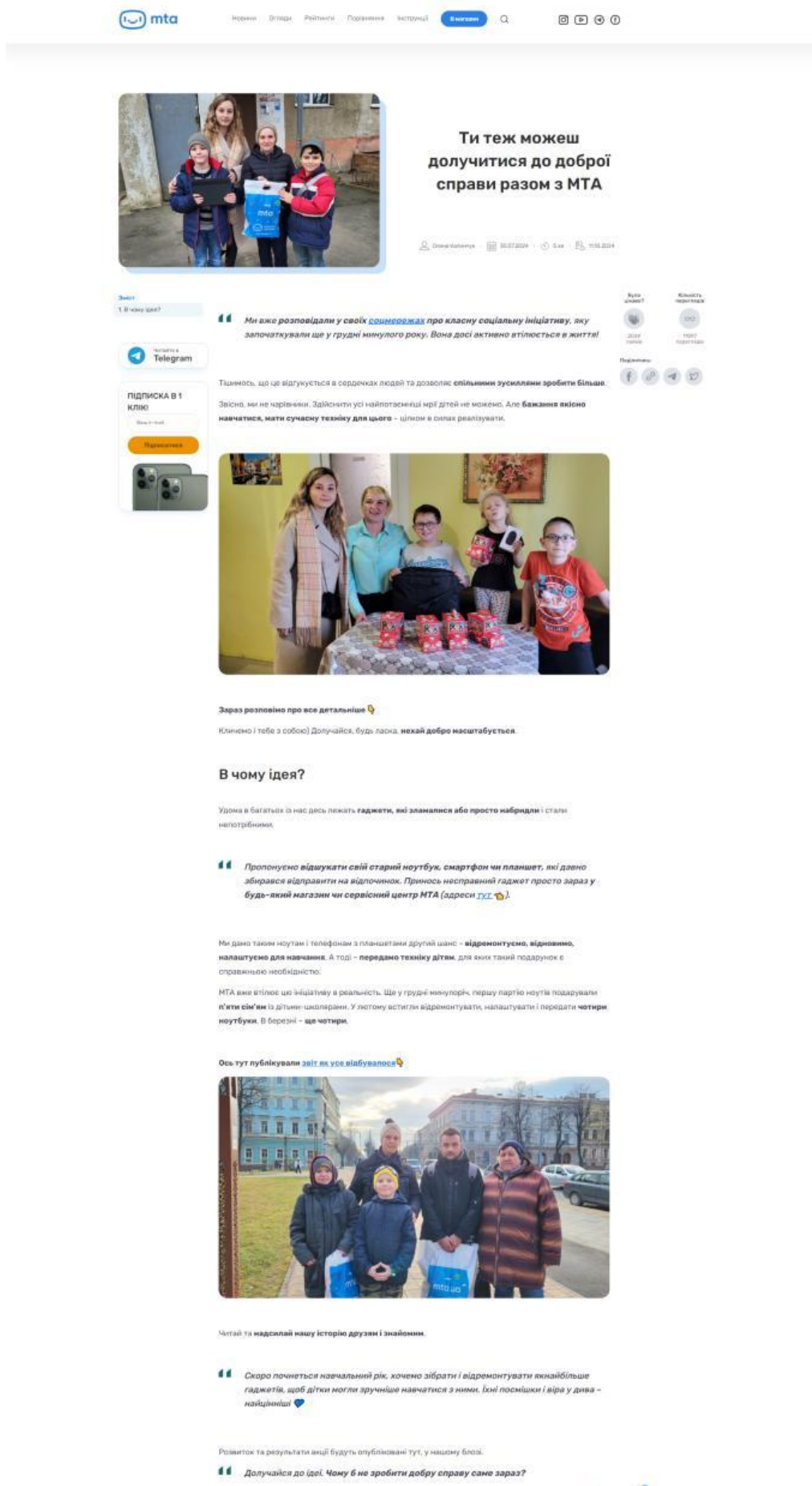


Рисунок 1.3 – Благодійна акція на сайті магазину МТА.ua

mta дружи́мо техні́ко +38 098 405 15 15 -
з 09:00 до 21:00 щодня

Блог Магазин МТА Доставка та оплата

Київ Увійти

Каталог Пошук

ІНТЕРНЕТ-МАГАЗИН МТА > ГАРАЖНИЙ РОЗПРОДАЖ

ГАРАЖНИЙ РОЗПРОДАЖ

▼ СПОЧАТКУ ДЕШЕВІ ▲ СПОЧАТКУ ДОРОГІ ✕ ПО РОЗМІРУ ЗНИЖКИ 🔥 СПОЧАТКУ ПОПУЛЯРНІ

ЦІНА
29 49599

РОЗМІР ЗНИЖКИ

- ☐ від 51% до 70% 52
- ☐ від 41% до 50% 84
- ☐ від 31% до 40% 66
- ☐ від 21% до 30% 51
- ☐ від 11% до 20% 57
- ☐ від 6% до 10% 13
- ☐ до 5% 10
- ☐ Подарунок 4

КАТЕГОРІЇ

Пошук по ключовому слову...

- ☐ Акумулятори для телефонів 17
- ☐ Батареї 15
- ☐ Бездротові навушники 9
- ☐ Плитарники 17
- ☐ Парохідники 10
- ☐ Ремінь для смарт-годинника 23
- ☐ Смартфони 7
- ☐ Стилузи 10
- ☐ Тримачі для телефонів Popholder 11
- ☐ USB лампи 6

ПОКАЗАТИ ЩЕ 192

ВИРОБНИКИ

Пошук по ключовому слову...

- ☐ Bambi 7
- ☐ Baseus 9
- ☐ BeCover 7
- ☐ Hoco 7
- ☐ Infinity 89
- ☐ Noveen 9
- ☐ Panasonic 8
- ☐ PowerPlant 15
- ☐ Volttronic 19
- ☐ Xiaomi 11

ПОКАЗАТИ ЩЕ 171

НАВІСНІСТЬ

Пошук по ключовому слову...

- ☐ Відправимо сьогодні 374
- ☐ В наявності на складі 421
- ☐ м. Івано-Франківськ, вул. Незалежності, 37 8
- ☐ м. Київ, пр. Бажанів, 3д, метро "Жарківська" 8
- ☐ м. Київ, пр. Бондарів, 23, ТЦ "ОБОРОДОК" (напроти кас FZZV) 32
- ☐ м. Коломия, пр. Михайла Грушевського, 7 7
- ☐ м. Львів, вул. Городища, 90 20
- ☐ м. Львів, вул. Наукова, 35а, ТЦ "ВАН" (1-й поверх) 47
- ☐ м. Львів, пл. Князя Ярослава Осмомисла, 90 23
- ☐ м. Львів, пл. Соборна, 11 5

ПОКАЗАТИ ЩЕ 22

Обігрівач Noveen OH13 Gray White 3000 W
25999 з 3999 з
KEURICK 26.8

USB-хаб Baseus AcmeJoy 4-Port Type-C Dark Gray (KCJ2010015)
899 з 999 з
KEURICK 9.8

Планшет Lenovo Tab P11 (2nd Gen) 6/128GB Wi-Fi Storm Gray + Pen...
12999 з
KEURICK 97.8

Стилуз Baseus Golden Cudgel Black (ACPL-01) з тонким пером
199 з 299 з
KEURICK 2.8

Смартфон Xiaomi Poco M6 Pro 12/512GB Purple (Global)
10999 з 11499 з
KEURICK 42.8

Бездротові навушники Huawei Freebuds 2 Pro Black
5599 з 6499 з
KEURICK 42.8

Випрямач для волосся Sensor SH1 S3000D
1299 з 1799 з
KEURICK 15.8

Маятниковий TP-Link Deco M4 (2-pack) White
3799 з 3999 з
KEURICK 38.8

Смартфон Samsung Galaxy S24 Ultra 5928 12/256GB Titanium Violet
49599 з 55099 з
KEURICK 2.8

Пальник газовий Volttronic K-202 Orange
199 з 399 з
KEURICK 2.8

Зовнішній портативний акумулятор Joyroom JR-T015 10000mAh Black AFC...
899 з 1299 з
KEURICK 9.8

Робот-пилосос Dreame Bot D10S Pro Black (RL56A)
11999 з 16999 з
KEURICK 90.8

Акустика портативна JBL Xtreme 3 Blue
12999 з
KEURICK 97.8

Зовнішній портативний акумулятор Gellus GP-PB10-015 10000mAh Black 10W Pt...
499 з 999 з
KEURICK 5.8

Зарядна станція Volttronic S-500W-30A
7899 з 9999 з
KEURICK 84.8

Камера ниттєвого друку Polaroid Go Black
3699 з 4499 з
KEURICK 37.8

Фітнес-браслет Xiaomi Mi Band 7 Pro Ivory (CN) (англ. мова)
1599 з
KEURICK 16.8

Накладні навушники JBL T500 Pink Sand
799 з
KEURICK 8.8

Кавомашинка Siemens EQ.9 Plus S100 Black TI921309RW
34999 з 43099 з
KEURICK 175.8

Адаптер живлення для телефона Infinity Iphone mini White (adapter (500 mAh))
199 з 299 з
KEURICK 5.8

Зовнішній портативний акумулятор Tracer TRABAT47073 10000mAh...
699 з 999 з
KEURICK 7.8

Пилосос Dyson V10 Animal Silver
24499 з 24999 з
KEURICK 121.8

Мережевий зарядний пристрій для телефона Baseus GaN5 Pro Desktop...
1899 з 2999 з
KEURICK 19.8

Планшет Samsung Galaxy T735 Tab S7 FE Black LTE 12.4
15899 з 16099 з
KEURICK 1.8

ЗАВАНТАЖИТИ ЩЕ 24

< 1 2 3 4 ... 20 >

Рисунок 1.4 – Розпродаж на сайті інтернет магазину МТА.ua

Отже, після проведення аналізу, можемо зробити висновок, що функціонал обох сайтів МТА.ua та ТТТ.ua включає каталог товарів з акціями, знижками та зручний пошук. Важлива роль відведена рекламним банерам, які виділяють спеціальні пропозиції. На МТА.ua більше акцент на кешбек, розпродажі та вигідні покупки дня, тоді як на ТТТ.ua акцент робиться на новинки техніки та рекламні кампанії. Відгуки користувачів на обох сайтах допомагають оцінити товар перед покупкою, що додає прозорості й довіри.

Основні вимоги до інтернет-магазину:

- інтуїтивно зрозумілий та легкий у використанні інтерфейс;
- модульна структура інтерфейсу для гнучкого управління;
- показ рекомендованих товарів для користувачів;
- можливість сортувати товари за різними категоріями;
- доступність контактної інформації на сайті;
- наявність функціоналу пошуку для швидкого знаходження товарів;
- адміністративна панель з розширеними функціями для управління магазином;
- система реєстрації та авторизації для користувачів;
- проста та швидка процедура оформлення замовлень;
- можливість збереження адреси користувача для зручності подальших замовлень;
- функція перегляду історії замовлень для зареєстрованих користувачів;
- опція залишати відгуки під товарами для інших покупців;
- можливість редагування особистих даних зареєстрованими користувачами;
- система для відновлення забутого пароля;
- гарантія захисту та конфіденційності персональних даних користувачів.

1.3 Визначення вимог до програмного забезпечення

Для створення інтернет-магазину з продажу техніки рекомендується використовувати вільне програмне забезпечення з відкритим вихідним кодом (open-source), що являє собою систему управління контентом (CMS). Це рішення допоможе значно знизити фінансові та часові витрати, водночас надаючи широкі можливості для налаштування та розширення функціоналу магазину. Обрана CMS повинна відповідати наступним вимогам:

- підтримка широкого функціоналу для інтернет-магазинів, включаючи управління каталогом товарів, можливість додавання детальної інформації про кожен продукт (описи, характеристики, фотографії);
- можливість інтеграції додаткових модулів, таких як система обробки платежів, інструменти аналітики, засоби seo-оптимізації та модулі для управління користувачами;
- інтуїтивно зрозумілий інтерфейс для зручності як кінцевого користувача, так і адміністратора, який забезпечить легке керування контентом і товарами;
- гнучкі можливості налаштування зовнішнього вигляду і структури сайту для відповідності вимогам компанії;
- сумісність із сучасними веб-стандартами, що гарантує стабільну роботу сайту на різних пристроях та браузерах.

Це забезпечить швидкий запуск інтернет-магазину з можливістю його подальшого розширення і розвитку за потреби.

1.3.1 Існуючі і самостійні розроблені системи

На етапі обговорення процесу розробки вебсайту першим постає питання: використовувати вже існуючу систему управління контентом чи створювати власну, спеціально під конкретний проєкт. Для прийняття рішення були проаналізовані переваги та недоліки кожного підходу.

Розглянемо переваги власноруч розробленого сайту:

- можливість налаштовувати і вдосконалювати окремі ключові модулі відповідно до потреб проєкту, що дозволяє оптимізувати його роботу;
- незалежність у питаннях оновлень та підтримки сайту;
- відсутність витрат на ліцензії або придбання CMS;
- можливість подальшого продажу або здачі стабільно працюючої системи в оренду.

Недоліки власної розробки:

- відсутність готових рішень і модулів, що призводить до необхідності їх розробки з нуля;
- брак документації, що ускладнює процес розробки та підтримки;
- розробка потребує значних часових та ресурсних витрат.

Тепер розглянемо негативні та позитивні моменти використання готової системи управління контентом (CMS).

Переваги:

- у відкритому доступі можна знайти численні відповіді на питання та рекомендації на форумах розробників, що значно спрощує вирішення технічних проблем;
- встановлення та налаштування готової займає набагато менше часу, ніж створення системи з нуля;
- популярні CMS мають широкий вибір шаблонів та додаткових модулів, що дозволяють розширювати функціонал без значних зусиль;
- панелі адміністрування мають зручний й інтуїтивний інтерфейс, що допомагає знизити навантаження при роботі над проєктом, навіть для користувачів без глибоких технічних знань;
- сучасні CMS постійно оновлюються і розвиваються, забезпечуючи актуальність функціоналу;
- високий рівень якості коду, що забезпечує стабільність роботи системи.

Недоліки:

- існує ризик уразливості: якщо не встановлювати вчасно оновлення безпеки, сайт може стати мішенню для хакерів;
- багато сайтів на базі однієї CMS можуть виглядати схоже через використання тих самих шаблонів;
- не всі додаткові модулі безкоштовні, що може збільшити витрати;
- можливі конфлікти між встановленими модулями, плагінами та розширеннями, що призводить до технічних проблем;
- велика кількість тем і доповнень низької якості, які можуть погіршити роботу сайту;
- при використанні CMS може збільшитися навантаження на хостинг, що потребує більшої продуктивності серверу;
- з відкритим кодом виникає додатковий ризик безпеки, оскільки до нього мають доступ всі користувачі.

З огляду на всі перелічені переваги та недоліки, можна зробити висновок, що для створення інтернет-магазину з мінімальними витратами часу та коштів краще використовувати готову систему управління контентом (CMS).

1.3.2 Аналіз систем управління для інтернет-магазину

Системи управління контентом (CMS) були розроблені для того, щоб спростити створення і адміністрування веб-сайтів, зокрема інтернет-магазинів. Сьогодні для створення власного сайту вже не потрібно володіти глибокими знаннями в програмуванні. Достатньо мати базове розуміння принципів роботи CMS та вміти користуватися інтерфейсом панелі адміністратора. Також важливо знати основи налаштування стилів, що дозволить зробити сайт естетично привабливим і зручним для користувачів [356, 5].

Кожна система управління контентом (CMS) надає унікальний набір інструментів та функціональних можливостей, які дозволяють гнучко адаптувати платформу під конкретні потреби бізнесу. Ці інструменти включають засоби для управління контентом, налаштування зовнішнього вигляду за допомогою шаблонів, управління товарами, замовленнями, а також інтеграцію з платіжними

системами та службами доставки. Завдяки цьому бізнес може створювати як прості, так і складні інтернет-магазини з великим асортиментом продукції.

CMS можна умовно поділити на три основні категорії: комерційні, відкриті (Open Source) та індивідуальні CMS.

У таблицях 1.1 та 1.2 представлено огляд кількох найбільш популярних безкоштовних CMS для інтернет-магазинів. У таблицях порівнюються ключові вимоги до систем, їхні переваги та недоліки, що допоможе визначити найбільш підходящу платформу для конкретних завдань інтернет-магазину.

Серед ключових критеріїв оцінки систем управління можна виділити гнучкість у налаштуваннях, рівень безпеки, доступність додаткових модулів і можливість масштабування. Для більш точного вибору платформи необхідно враховувати специфіку бізнесу, його цілі, а також технічні й функціональні вимоги.

Таблиця 1.1 – Необхідні вимоги CMS для Інтернет-магазину

	OpenCart	Magento	Woo Commerce
Кількість онлайн магазинів	180 000	216 000	396 000
Частка ринку	6	58	15
Кількість плагінів для системи	Велика	Велика	Велика
Вартість обслуговування	Низька	Низька	Середня
Доступність фахівців	Багато	Дуже багато	Багато
Масштабованість	Для малого і середнього бізнесу	Для великих магазинів	Для малого і середнього бізнесу
Безпека	Середня	Висока	Середня
Підтримка мобільних пристроїв	Добра	Відмінно	Відмінно

Таблиця 1.2 – Переваги та недоліки популярних CMS для інтернет-магазинів

CMS	Переваги	Недоліки
OpenCart	– підтримка великої кількості товарів і категорій; – SEO-оптимізація; – зручний інтерфейс для адміністрування; – швидкий старт; – понад 2200 шаблонів для оформлення.	– складнощі зі створенням користувацьких сторінок; – обмежена функціональність; – проблеми з оновленням системи; – мінімальна підтримка; – імпорт товару.
Magento	– мобільна інтеграція; – гнучкість у налаштуваннях товарів і категорій; – можливість зберігати кошик покупок тривалий час; – понад 1000 доступних тем; – база знань і форум підтримки.	– відсутня офіційна технічна підтримка; – складність управління великим обсягом товарів; – необхідні розширені навички веброзробника.
Woo Commerce	– легке підключення додаткових модулів; – партнерські програми і соціальні мережі; – SEO-оптимізація; – аналітика і звітність.	– складність налаштувань – виникнення помилок при використанні додаткових плагінів; – висока залежність від тем WordPress.

Підібрати ідеальну систему, яка б включала всі необхідні функції, практично неможливо. Саме тому був проведений детальний аналіз переваг та недоліків різних систем управління контентом (CMS). На основі цього аналізу було обрано платформу, яка найкраще відповідає ключовим вимогам для створення ефективного інтернет-магазину, а саме систему управління «OpenCart», оскільки вона максимально відповідає потребам інтернет-магазину та забезпечує необхідну гнучкість, функціональність і зручність для користувачів.

Додатково, «OpenCart» пропонує широкий вибір плагінів та тем, що дозволяє легко масштабувати магазин і адаптувати його до змінних бізнес-потреб.

1.3.3 Опис технічних вимог та характеристик

Система «OpenCart» є вільним програмним забезпеченням, яке розповсюджується за ліцензією GNU General Public License v3. Це означає, що її можна використовувати, змінювати та розповсюджувати вільно, дотримуючись умов ліцензії. Однією з ключових переваг цієї CMS є підтримка численних доповнень та модулів, що дозволяє розширювати функціональні можливості інтернет-магазину та адаптувати його під індивідуальні потреби бізнесу.

Система була розроблена і підтримується британським розробником Даніелем Керром. Перша версія «OpenCart» вийшла в серпні 2012 року, і вже за кілька років вона стала одним із лідерів серед платформ для інтернет-магазинів у країнах СНД. Завдяки простоті використання, відкритому коду та широкій підтримці спільноти, «OpenCart» залишається популярним вибором для малого та середнього бізнесу [6].

Програмне забезпечення «OpenCart» написано на мові PHP, а в його архітектурі використовується шаблон MVC (Model-View-Controller), що дозволяє розділяти логіку програми, інтерфейс і дані, спрощуючи розробку та обслуговування. Завантажити програмне забезпечення можна з офіційного сайту, який доступний усім користувачам Інтернету.

Технічні вимоги для встановлення та роботи «OpenCart»:

- вебсервер: Система працює на Apache (або іншому сумісному вебсервері), з попередньо встановленим необхідним ПЗ;
- база даних: рекомендовано використовувати MySQLi або інші підтримувані бази даних;
- PHP: мінімальна версія PHP – 5.2, але бажано використовувати більш нові версії для забезпечення стабільності та безпеки;
- PHP memory_limit: мінімальне значення – 64 МБ, проте для нормальної роботи системи з більшими обсягами трафіку або товарів рекомендується виділити не менше 128 МБ пам'яті.

Для великих інтернет-магазинів, де передбачається значний потік відвідувачів, рекомендується використовувати більш потужні сервери та збільшувати об'єм оперативної пам'яті до 256 МБ і більше, що дозволить забезпечити стабільну роботу магазину під навантаженням і прискорити обробку запитів.

Таким чином, «OpenCart» є досить гнучкою платформою, яка може працювати навіть на скромних серверах при мінімальних вимогах, але для повноцінної і стабільної роботи в умовах зростання бізнесу необхідно враховувати масштабованість і забезпечити відповідні технічні ресурси.

1.3.4 Обґрунтування вибору інструментів для створення інтернет-магазину

Основні принципи вибору інструментів для розробки включають наступні вимоги:

- програмне забезпечення повинно забезпечувати можливість налагодження без необхідності постійного внесення змін до проєкту безпосередньо на вебсервері;
- кросплатформенність;
- використання безкоштовного програмного забезпечення.

Виходячи з цих критеріїв, для реалізації інтернет-магазину були обрані наступні інструменти:

- HTML;
- CSS;
- JavaScript;
- PHP.

HTML (Hyper Text Markup Language) – це мова розмітки гіпертексту, який використовується для створення та структурування веб-сторінок. Вона визначає структуру веб-сторінок [7]. HTML складається з тегів, які визначають різноманітні компоненти сторінки та їх функції, наприклад заголовки, параграфи, таблиці тощо. HTML використовується для структурування вебсторінок магазину. Це основа фронтенду, що визначає розташування елементів, таких як меню,

каталоги товарів, форми для реєстрації та входу, кошик і сторінки товарів. У даному проєкті цей інструмент використовується для створення базової структури вебресурсу, зокрема, у процесі проєктування інтерфейсу користувача.

CSS (Cascading Style Sheets – каскадні таблиці стилів) – це мова, яка визначає зовнішній вигляд HTML-документа, тобто як повинні відображатися елементи HTML, надання необхідного стилю. CSS дає можливість задавати візуальні характеристики елементів вебсторінки, такі як шрифт, колір, розміри, відступи й відображення для різних пристроїв і розмірів екрана. CSS дозволяє контролювати дизайн декількох вебсторінок одночасно, використовуючи зовнішні таблиці стилів, які зберігаються в окремих файлах [8]. Завдяки CSS було реалізовано кольорові схеми, розташування блоків, шрифти та інші візуальні елементи, що сприяють створенню привабливого дизайну магазину.

PHP (Personal Home Page Tools) – це серверна мова з відкритим вихідним кодом, яка є потужним інструментом для створення динамічних і інтерактивних вебсторінок. PHP є популярною безкоштовною альтернативою конкурентам, таким як ASP від Microsoft. PHP-скрипти виконуються на сервері [9]. Для проєкту використаємо PHP для:

- обробки запитів користувачів, таких як реєстрація, вхід до облікового запису та оформлення замовлень;
- взаємодії з базою даних, наприклад, збереження інформації про користувачів, товари, замовлення та їхній статус;
- реалізації бізнес-логіки магазину, включаючи обробку кошика та управління замовленнями.

PHP також інтегрується з платформою OpenCart, яка використовує цю мову для розширення функціоналу магазину.

JavaScript – це мова програмування, яка дозволяє додати інтерактивність на вебсторінку, наприклад, реакції на натискання кнопок, введення даних у форму або створення анімацій [48, 10]. JavaScript потрібен для нашого вебресурсу для реалізацією функціоналу взаємодії користувачів із магазином без необхідності перезавантаження сторінок, а саме:

- обробка кліків на кнопках, таких як «Додати до кошика» або «Оформити замовлення»;
- динамічне оновлення контенту, як-от автоматичне оновлення кількості товарів у кошику;
- реалізація фільтрів для пошуку товарів за категоріями або ціною.

2 МОДЕЛЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Моделювання прецедентів та опис акторів

Метою діаграми прецедентів є документування дійових осіб, варіантів використання та взаємозв'язків між цими компонентами. Опис дійових осіб наведено в таблиці 2.1.

Таблиця 2.1 – Опис акторів та їх характеристики

Класи користувачів	Характеристика користувачів
Відвідувач	Новий користувач який уперше зайшов на сайт, може переглядати каталоги товарів та їх описи.
Клієнт	Зареєстрований відвідувач, який може оформляти замовлення.
Адміністратор	Працівник інтернет-магазину, який керує наявними товарами, додає нові або видаляє старі. Він також має доступ до адміністративних функцій.
Робітник на складу	Співробітник, який відповідає за облік товарів на складі та передає їх інформацію службі доставки.
Бухгалтер	Співробітник, який контролює платежі та веде бухгалтерський облік.

На рисунку 2.1 представлена діаграма прецедентів для інтернет-магазину, де візуалізовані основні актори та їх взаємодії із системою.



Рисунок 2.1 – UML-діаграма прецедентів системи

2.2 Моделювання діаграм послідовності

Для наочного відображення поведінки об'єктів у сценаріях використовується діаграма взаємодії. Ця нотація надає можливість відобразити ключові аспекти взаємодії, включаючи виклики методів, передачу повідомлень та зміну стану об'єктів. Лінії життя, що є невід'ємною частиною діаграми, символізують час існування об'єктів, і зазвичай відображаються у вигляді вертикальних пунктирних ліній. На вершині кожної лінії життя розташовується прямокутник із назвою об'єкта або актора, що дозволяє легко ідентифікувати його роль у сценарії.

У процесі взаємодії об'єкти обмінюються повідомленнями, які графічно представлені у вигляді стрілок між лініями життя. Ці повідомлення можуть бути синхронними або асинхронними, що дозволяє моделювати реальні процеси взаємодії в системі. Діаграма включає всі об'єкти, які беруть участь у виконанні сценарію, забезпечуючи повне розуміння процесу. Особливу роль у діаграмі взаємодії відіграє крайня права лінія, яка представляє зовнішнє середовище або користувача, з яким взаємодіє система. Це можуть бути як люди (актори), так і зовнішні системи, що виконують функції інтерфейсу. У складних сценаріях може бути кілька таких інтерфейсів, кожен з яких відповідає за певний аспект взаємодії.

Цей вид моделювання особливо корисний для документування сценаріїв, що вимагають чіткої послідовності дій, таких як процес авторизації, оформлення замовлення або обробка запиту. Таким чином, діаграма взаємодії забезпечує зручний інструмент для аналізу та оптимізації бізнес-процесів, дозволяючи виявити слабкі місця та знайти шляхи для їхнього вдосконалення.

На рисунку 2.2 представлена діаграма послідовності для прецеденту «Реєстрація».

Зацікавлені сторони:

- клієнт: прагне швидко зареєструватися для оформлення замовлення;
- підприємство: бажає, щоб клієнт почав купувати його товари.

Результат: клієнт успішно зареєстрований і може робити покупки.

Основний потік:

- 1) Клієнт відкриває головну сторінку сайту і натискає кнопку «Зареєструватися».
- 2) Модуль реєстрації надсилає форму для реєстрації.
- 3) На сторінці з'являється форма реєстрації.
- 4) Клієнт заповнює форму та натискає кнопку «Підтвердити».
- 5) Модуль реєстрації відправляє початкові дані клієнта для запису в базу даних.

6) На електронну пошту, вказану клієнтом, надсилається лист для підтвердження реєстрації, а на сайті з'являється повідомлення з проханням підтвердити реєстрацію через пошту.

7) Клієнт переходить за посиланням у листі для підтвердження реєстрації.

8) Після успішного підтвердження модуль реєстрації надсилає повні дані про нового користувача.

9) Клієнт отримує підтвердження про успішну реєстрацію.

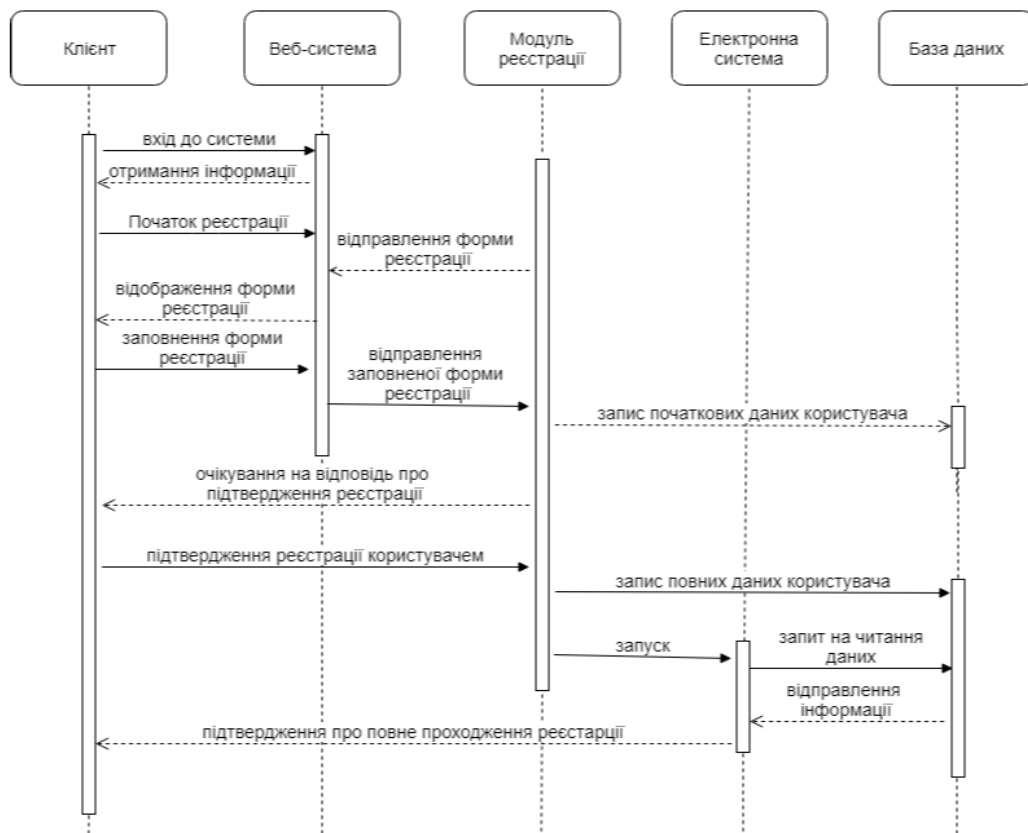


Рисунок 2.2 – Діаграма послідовності прецеденту «Реєстрація»

На рисунку 2.3 представлена діаграма послідовності для прецеденту «Придбання товару».

Основний виконавець: клієнт.

Зацікавлені сторони:

- клієнт: має наміри купити товар;
- підприємство: прагне продати товар та задовольнити потреби клієнта.

Результат: клієнт отримає свій товар і залишиться задоволеним, а підприємство отримає оплату за замовлення.

Основний потік:

- 1) Клієнт відкриває сторінку сайту з каталогом товарів.
- 2) Вебсистема надсилає запит до бази даних для отримання каталогу товарів.
- 3) База даних перевіряє наявність товарів.
- 4) На сайті відображається каталог товарів.
- 5) Клієнт обирає товари та натискає «Купити».
- 6) Вебсистема надсилає клієнту форму для заповнення.
- 7) Клієнт заповнює форму та відправляє її.
- 8) Вебсистема перевіряє правильність введеної інформації.
- 9) Реквізити передаються до системи оплати.
- 10) Вебсистема перевіряє правильність реквізитів.
- 11) Система оплати підтверджує успішність платежу.
- 12) Вебсистема оновлює дані про наявність товару в базі даних.
- 13) База даних реалізує зміни.
- 14) Вебсистема надсилає інформацію про замовлення на електронну пошту клієнта.
- 15) Вебсистема передає інформацію про замовлення службі доставки.
- 16) Служба доставки передає товар клієнту.

Альтернативні потоки:

8а. Якщо інформація некоректна:

- 1) Вебсистема повідомляє клієнта про помилку у введених даних.
- 2) Клієнт виправляє помилку.
- 3) Вебсистема перевіряє дані повторно.

10а. Якщо платіжні реквізити некоректні:

- 1) Вебсистема повідомляє клієнта про помилку у введених платіжних даних.
- 2) Клієнт виправляє помилку.

3) Вебсистема перевіряє дані повторно.

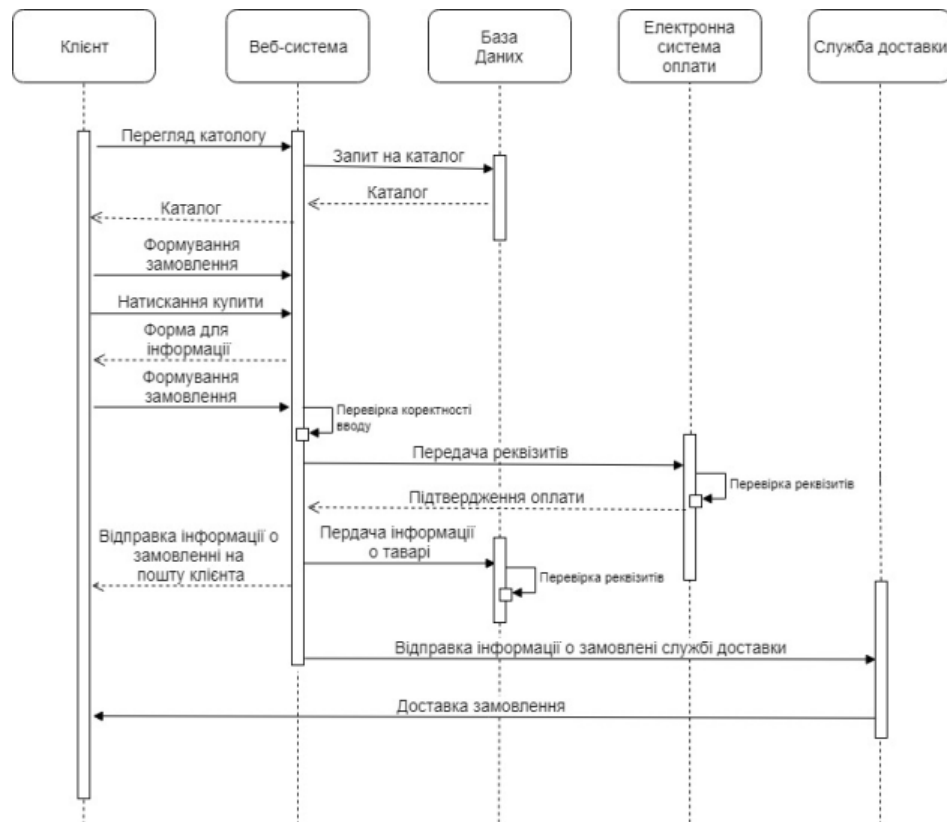


Рисунок 2.3 – Діаграма послідовності прецеденту «Покупка товару»

На рисунку 2.4 представлена діаграма послідовності для прецеденту «Додавання товару».

Основний виконавець: адміністратор.

Зацікавлені сторони:

- адміністратор: хоче додати новий товар;
- підприємство: прагне збільшити асортимент товарів.

Результат: Адміністратор успішно додає новий товар, підприємство розширює свій асортимент.

Основний потік:

- 1) Адміністратор заходить на сторінку товарів.
- 2) Вебсистема надсилає запит до бази даних для отримання списку товарів.
- 3) База даних повертає список наявних товарів.

- 4) На сторінці сайту відображається список усіх товарів.
- 5) Адміністратор натискає кнопку «Додати товар».
- 6) Вебсистема відкриває форму для додавання нового товару.
- 7) Адміністратор вводить необхідні дані про товар.
- 8) Вебсистема перевіряє введену інформацію.
- 9) Вебсистема надсилає інформацію до бази даних.
- 10) База даних додає новий товар.
- 11) База даних повертає дані про доданий товар.
- 12) На сайті відображається сторінка нового товару.

Альтернативні потоки:

8а. Якщо введена інформація некоректна:

- 1) Вебсистема вказує адміністратору на помилки у введених даних.
- 2) Адміністратор коригує неточності у введених даних.
- 3) Вебсистема повторно перевіряє коректність інформації.

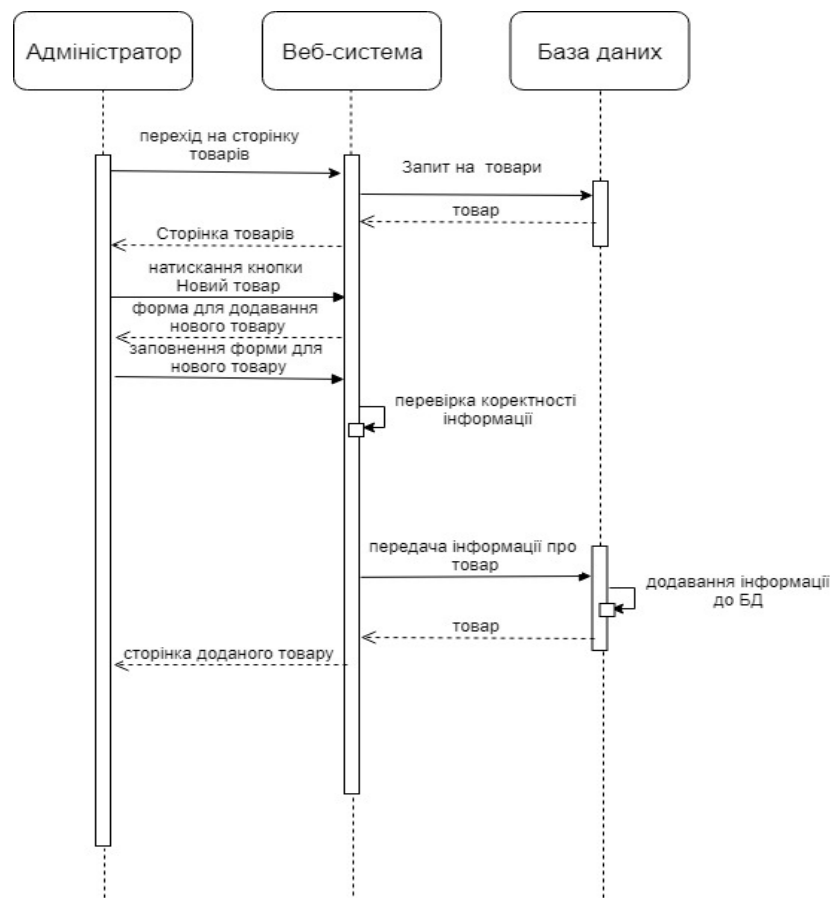


Рисунок 2.4 – Діаграма послідовності прецеденту «Додавання товару»

3 ПРОЄКТУВАННЯ СИСТЕМИ

3.1 Архітектура програмного забезпечення

Інтернет-магазин функціонує на основі клієнт-серверної архітектури. Цей тип архітектури передбачає, що обчислювальні ресурси і навантаження розподіляються між серверами (постачальниками послуг) та клієнтами (споживачами послуг). У даному контексті як сервер, так і клієнт є програмними компонентами [11]. Зазвичай, вони розміщуються на різних фізичних машинах, взаємодіючи між собою через мережу за допомогою стандартних мережевих протоколів. Однак, у деяких випадках обидва компоненти можуть знаходитися на одній і тій самій машині.

Ключова перевага клієнт-серверної архітектури полягає в її масштабованості: можна додавати нові сервери для обробки збільшених обсягів запитів клієнтів. Це робить її оптимальним вибором для інтернет-магазинів, де одночасно може працювати велика кількість користувачів.

Програми-сервери чекають запитів від клієнтських програм і надають їм свої ресурси у вигляді даних або сервісних функцій. Оскільки сервер може обслуговувати велику кількість клієнтів одночасно, його зазвичай розміщують на спеціалізованій обчислювальній машині, налаштованій для високої продуктивності. Ця машина часто працює в парі з іншими програмами-серверами. Через важливість такої машини для мережі, її не лише називають сервером з технічної точки зору, але й обладнують більш потужними ресурсами в порівнянні з клієнтськими машинами.

Основними перевагами такої архітектури є:

- централізоване управління кодом: програма-сервер виконує всі ключові обчислення, що дозволяє уникнути дублювання коду клієнтськими програмами;
- знижені вимоги до клієнтів: клієнтські машини можуть бути менш потужними, оскільки основне навантаження на обчислення та обробку даних бере на себе сервер;

- підвищений захист даних: оскільки всі дані зберігаються на сервері, вони краще захищені, ніж на клієнтських пристроях. це також дає можливість централізовано контролювати доступ до інформації та управляти правами доступу користувачів.

Однак існують і недоліки:

- залежність від працездатності сервера: якщо сервер виходить з ладу, це може паралізувати всю мережу;
- високі витрати на обладнання: потужні сервери вимагають значних фінансових інвестицій як у самі машини, так і в їх обслуговування.

Така архітектура часто є основою для корпоративних систем і великих інтернет-платформ завдяки своїй ефективності та можливостям централізованого управління.

3.2 Проєктування бази даних

База даних відображає інформацію, яка відноситься до певної предметної області. Під предметною областю розуміють конкретну частину реального світу, дані про яку необхідно відобразити в процесі розробки бази даних. Саме ці дані є основою для побудови інформаційної системи, оскільки вони використовуються для підтримки роботи прикладних задач.

Система управління базами даних (СУБД) – це програмна платформа, яка забезпечує збереження, управління і маніпуляцію даними, які важливі для користувачів. Іншими словами, СУБД виступає інтерфейсом між базою даних і прикладними програмами, дозволяючи здійснювати запити, оновлювати інформацію та забезпечувати її цілісність [12].

Основною особливістю СУБД є її здатність зберігати не лише самі дані, а й їхню структуру. Це дозволяє системам не тільки виконувати обчислення та зберігати інформацію, а й формувати повноцінні моделі предметної області. Така

інтеграція даних із їхнім описом призвела до виникнення концепції банків даних, які згодом отримали назву бази даних (БД).

При проєктуванні БД важливо враховувати структуру даних, їх взаємозв'язки, а також вимоги до доступу та безпеки. Це дозволяє побудувати ефективну і надійну систему зберігання даних, яка задовольняє потреби користувачів і підтримує продуктивність прикладних систем.

3.2.1 Обґрунтування і вибір СУБД

У цьому розділі були розглянуті та обрані дві системи управління базами даних (СУБД):

- SQLite [13];
- MySQL [14].

SQLite – це вбудована реляційна база даних з відкритим вихідним кодом. Вона відрізняється від класичних моделей СУБД, оскільки не використовує клієнт-серверну архітектуру і не є окремим процесом. По суті, SQLite інтегрується безпосередньо в програму, з якою працює, і функціонує як частина вебдодатку. Дані в SQLite зберігаються у вигляді звичайного текстового файлу, який можна розмістити у зручному місці на пристрої. Це значно спрощує роботу з базою даних, оскільки немає потреби налаштовувати окремі сервери або мережеві з'єднання.

Перевага SQLite полягає в тому, що вона працює безпосередньо з файлами, що робить її дуже швидкою. Її продуктивність досягається за рахунок ефективної обробки запитів і використання технологій підтримуючих бібліотек. Крім того, SQLite є досить потужною системою для додатків, що не потребують масштабованих розподілених систем, і пропонує великий набір інструментів для роботи з даними без складної серверної інфраструктури.

Цей підхід підходить для невеликих або середніх проєктів, де необхідна висока продуктивність і простота інтеграції, але немає потреби у потужних можливостях клієнт-серверних систем, таких як MySQL.

Основні типи даних у SQLite поділяються на кілька категорій:

- INTEGER – цілі числа будь-якої довжини. Значення зберігаються як 1, 2, 3, 4, 6 або 8 байтів залежно від розміру числа;

- REAL – числа з плаваючою комою, що використовують формат IEEE з 8 байтами для збереження;

- TEXT – текстові дані, що зберігаються в кодуванні UTF-8, UTF-16BE або UTF-16LE;

- BLOB – двійкові дані, які зберігаються "як є" без змін;

- NULL – позначає відсутність даних.

SQLite використовує динамічну типізацію, тому не обов'язково вказувати тип даних для кожного стовпця. СУБД дозволяє зберігати різні типи даних у одному стовпці, залежно від вмісту.

Переваги:

- база даних зберігається в одному файлі, що полегшує управління;
- через відсутність мережових з'єднань і прямий доступ до файлу, SQLite забезпечує високу продуктивність для невеликих і середніх проєктів;
- використовує дуже малий обсяг пам'яті;
- працює на різних платформах і легко інтегрується в мобільні додатки, вебдодатки та вбудовані системи;
- відкритий вихідний код, що забезпечує безкоштовний доступ до програмного забезпечення та дозволяє модифікувати його під свої потреби.

Недоліки:

- обмежена масштабованість, при зростанні кількості користувачів або обсягів даних її продуктивність може знижуватися;
- не підтримує такі функції, як складні транзакції на рівні сервера або налаштування на рівні користувачів, що може обмежити її у великих корпоративних проєктах;
- при значних обсягах даних продуктивність знижується через роботу з великими файлами, оскільки вся база даних зберігається у вигляді одного файлу.

Наступна СУБД для розгляду – MySQL.

MySQL – це реляційна система управління базами даних (СУБД) з відкритим вихідним кодом, що використовує мову SQL (Structured Query Language) для управління та обробки даних. Вона побудована на клієнт-серверній архітектурі, де сервер відповідає за зберігання і управління базою даних, а клієнти взаємодіють із нею через запити.

MySQL широко використовується для вебдодатків, включаючи популярні платформи, як WordPress, Joomla, і є основою для багатьох онлайн-сервісів завдяки своїй продуктивності, гнучкості та відкритості.

Переваги:

- працює на сервері, що обробляє запити через мережу у великій кількості користувачів і додатків;
- може обслуговувати одночасно багато клієнтів, що робить її придатною для масштабованих вебсайтів і великих систем;
- забезпечує цілісність даних через підтримку транзакцій, гарантує ACID-принципи (атомарність, консистентність, ізоляція, довговічність);
- високий рівень захисту даних через різні методи аутентифікації, шифрування і управління правами доступу для користувачів;
- працює на різних операційних системах, таких як Windows, Linux і macOS, що робить її універсальною для різних середовищ.

3.2.2 Концептуальне проєктування бази даних

Концептуальне (інфологічне) проєктування полягає в аналізі предметної області та її формалізованому описі. На цьому етапі немає прив'язки до конкретних технічних або програмних засобів, що дозволяє сфокусуватися на суті даних та їх взаємозв'язках.

Для цього використовують ER-модель (Entity-Relationship модель) або ER-діаграму, яка служить інструментом для відображення структури предметної області. Це графічне представлення системи за допомогою стандартизованих позначень, що допомагає наочно описати об'єкти та зв'язки між ними. На основі цієї моделі пізніше створюється логічна схема для подальшої реалізації в конкретній СУБД.

Сутність – це об'єкт, реальний або уявний, інформація про який повинна зберігатися у базі даних. На ER-діаграмі сутність представляється прямокутником із зазначенням її назви, яка відображає тип сутності.

Зв'язок – це асоціація між сутностями, що вказує на взаємозв'язки між ними. Графічно зв'язок зображується у вигляді ліній, що з'єднують дві сутності. Він завжди бінарний, тобто встановлюється між двома сутностями, які можуть бути як різними, так і однаковими.

Концептуальне проектування допомагає структуровано зрозуміти предметну область і визначити, які дані мають бути збережені, а також як вони будуть взаємодіяти між собою. Це критично важливий етап для створення правильної логічної моделі бази даних, яка пізніше буде реалізована у вибраній СУБД.

Атрибут сутності – це будь-яка характеристика, яка описує сутність, допомагає її ідентифікувати, класифікувати або виражати її стан. Атрибути додають деталізацію до сутностей, і їхні назви зазвичай зображуються в овалах на ER-діаграмі, які з'єднуються лініями із сутністю.

Після проведеного аналізу предметної області було визначено такі сутності та їх атрибути:

- categories (category_id, name, description, image): відображає категорії товарів;
- comments_product (comment_id, date_com, comment): зберігає інформацію про коментарі до товарів;
- content_favorites (content_favorites_id, product_id): зберігає інформацію про вподобані товари;
- customers (customers_id, name, surname, password, email, phone_number): містить дані про клієнтів;
- delivery (delivery_id, delivery_name): інформація про типи доставки;
- favorites (favorites_id, customers_id): зберігає дані про списки улюблених товарів клієнтів;

- manufacturer (manufacturer_id, manufacturer_name): зберігає інформацію про виробників товарів;
- order (order_id, customers_id, delivery_id, product_id, status_id, date_order, address): зберігає інформацію про замовлення;
- order_status (status_id, status_name): відображає статуси замовлень;
- product (product_id, category_id, manufacturer_id, product_name, price, attributes, description, image): зберігає дані про товари.

Ці сутності та їх атрибути допомагають чітко структурувати інформацію та визначити основні компоненти бази даних для реалізації інтернет-магазину.

Сутності можуть перебувати в різноманітних взаємозв'язках, в яких зв'язок між ними визначаються типами і класу приналежності. Можна виділити наступні основні зв'язки:

- «один до одного» (1:1);
- «один до одного» (1:1);
- «багато до багатьох» ($\infty:\infty$).

Клас приналежності може бути обов'язковим, якщо всі об'єкти одного класу мають у зв'язок з об'єктами іншого класу, або необов'язковим, якщо деякі об'єкти можуть не мати зв'язків з іншими класами.

На основі цих сутностей можна визначати зв'язки між ними. Наприклад, між сутностями «categories» і «product» існує зв'язок типу 1: ∞ (рис. 3.1).

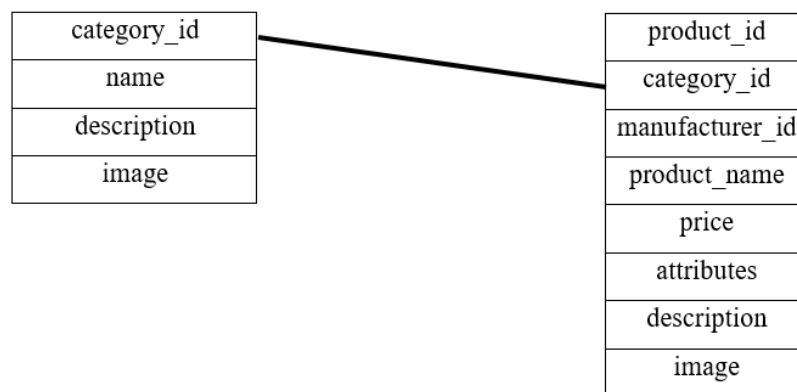


Рисунок 3.1 – Зв'язок типу 1: ∞ між сутностями «categories» та «product»

Між сутностями «favorites» та «customers» зв'язок 1:∞ (рис. 3.2).

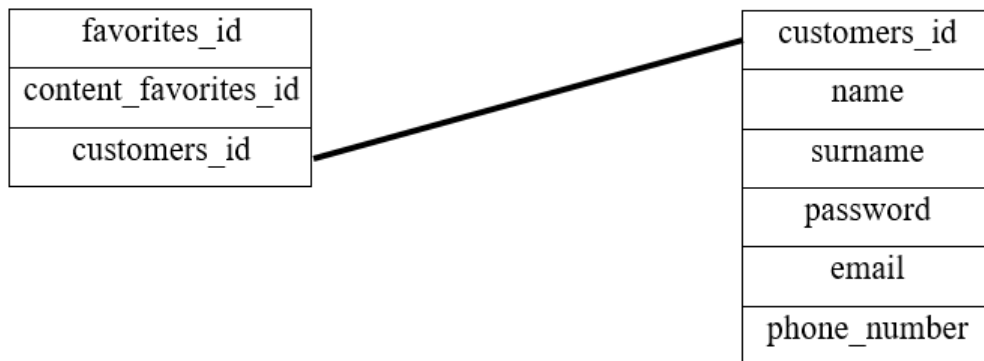


Рисунок 3.2 – Зв'язок між сутностями «favorites» та «customers»

Між сутностями «comment» та «customer» зв'язок ∞ : 1 (рис. 3.3).

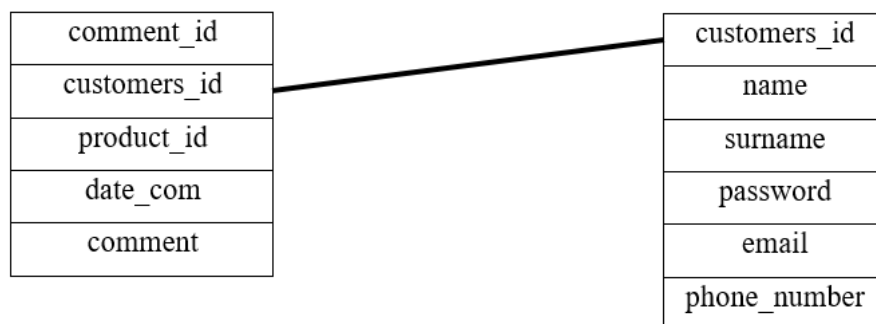


Рисунок 3.3 – Зв'язок між сутностями «comment» та «customer»

Між сутностями «manufacturer» та «product» зв'язок 1:∞ (рис. 3.4).

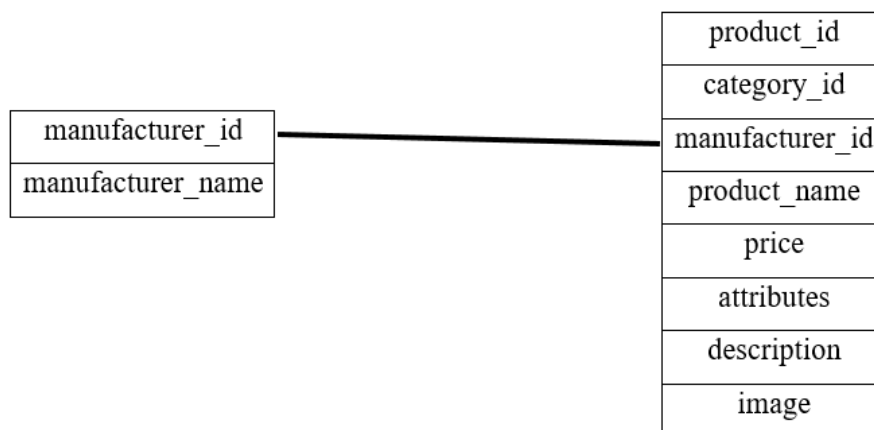


Рисунок 3.4 – Зв'язок між сутностями «manufacturer» та «product»

Між сутностями «order» та «delivery» зв'язок $\infty : 1$ (рис. 3.5).

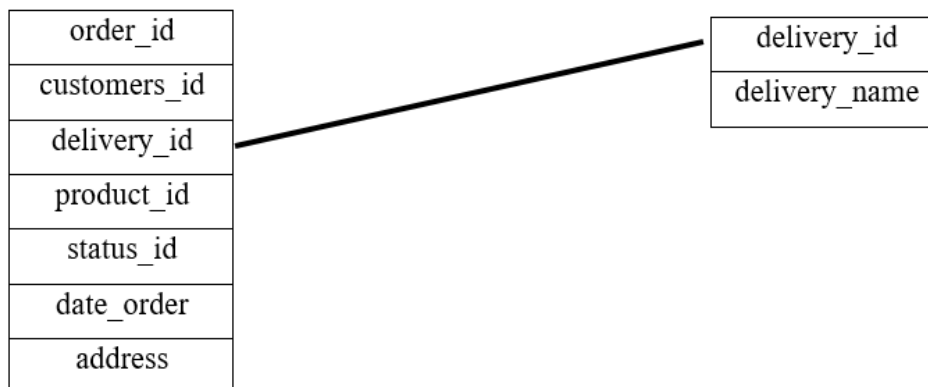


Рисунок 3.5 – Зв'язок між сутностями «order» та «delivery»

Між сутностями «order» та «customers» зв'язок $\infty : 1$ (рис. 3.6).

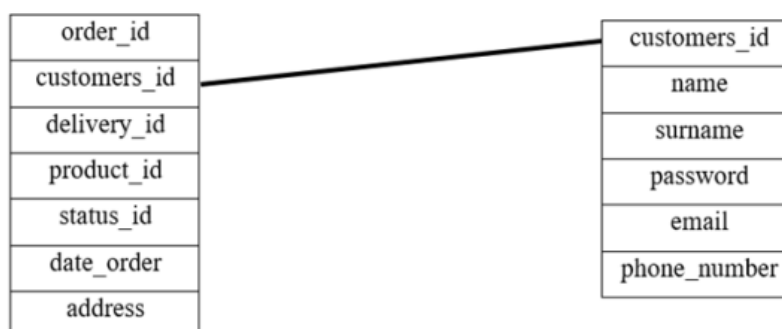


Рисунок 3.6 – Зв'язок між сутностями «order» та «customers»

Між сутностями «order» та «status» зв'язок $\infty : 1$ (рис. 3.7).

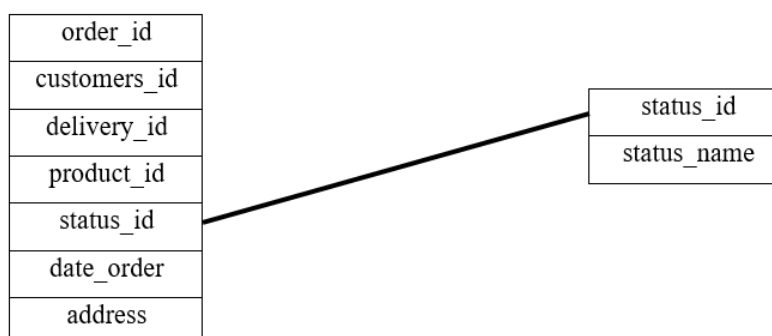


Рисунок 3.7 – Зв'язок між сутностями «order» та «status»

Між сутностями «comment» та «product» зв'язок 1:∞ (рис. 3.8).

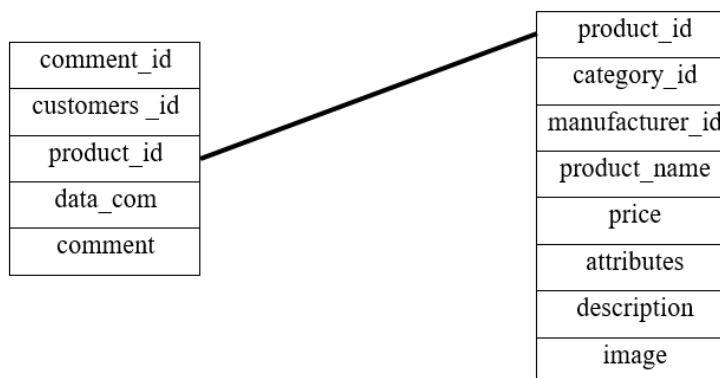


Рисунок 3.8 – Зв'язок між сутностями «comment» та «product»

Між сутностями «order» та «product» зв'язок 1:∞ (рис. 3.9).

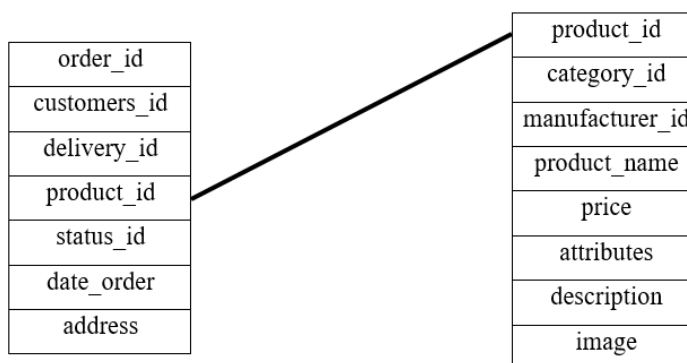


Рисунок 3.9 –Зв'язок між сутностями «order» та «product»

Між сутностями «content_favorites» та «product» зв'язок ∞ : 1 (рис. 3.10).

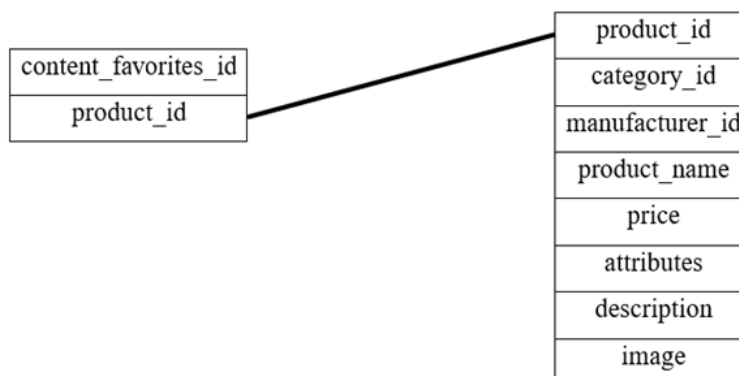


Рисунок 3.10 –Зв'язок між сутностями «content_favorites» та «product»

Після створення концептуальних моделей бази даних, можна побудувати ER-модель, яка представлена на рисунку 3.11.

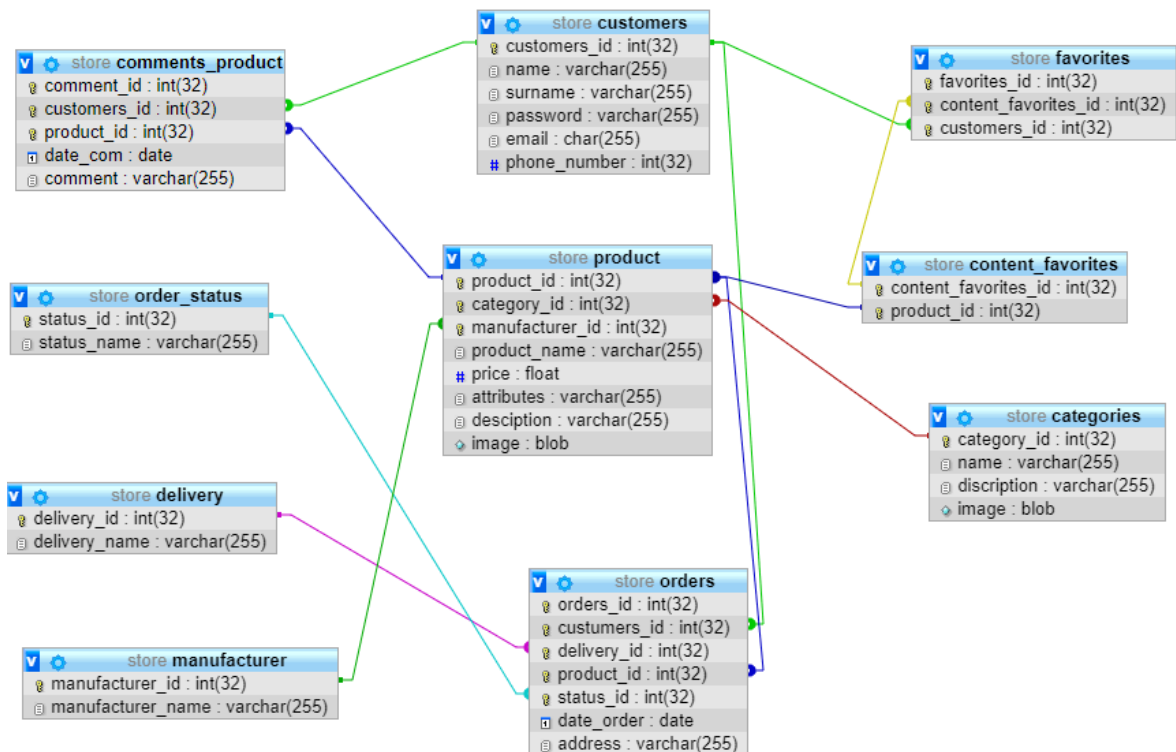


Рисунок 3.11 –ER-модель бази даних інтернет-магазину

3.2.3 Побудова логічної моделі бази даних

Логічне проєктування полягає в створенні схеми бази даних на основі певної моделі даних, наприклад, реляційної. У випадку реляційної моделі логічна модель є набором схем відносин, що включають визначення первинних ключів і зв'язків між відносинами через зовнішні ключі.

Процес перетворення концептуальної моделі в логічну здійснюється за допомогою формальних правил. Більшість етапів цього процесу можуть бути значною мірою автоматизовані.

На етапі логічного проєктування враховуються особливості обраної моделі даних, але специфіка конкретної СУБД може не братися до уваги. Під час проєктування логічної структури реляційної бази даних визначається оптимальний набір таблиць для зберігання необхідної інформації. Для кожної таблиці встановлюються її назва, перелік полів і первинний ключ.

У межах логічного проєктування бази даних також можуть бути визначені обмеження цілісності, прийняті рішення щодо створення індексів тощо. Структури сутностей наведено у вигляді таблиць 3.1-3.10.

Таблиця 3.1 – Структура сутності «customers»

Поле	Тип даних	Обов'язковість	Примітки
customers_id	int(32)	Так	key
name	varchar	Так	
surname	varchar	Так	
password	varchar	Так	
email	varchar	Так	
phone	int(32)	Так	

Таблиця 3.2 – Структура сутності «product»

Поле	Тип даних	Обов'язковість	Примітки
product_id	int(32)	Так	key
category_id	int(32)		
manufacturer_id	int(32)		
product_name	varchar	Так	
price	float	Так	
attributes	varchar	Так	
description	varchar	Так	
image	blob	Так	

Таблиця 3.3 – Структура сутності «delivery»

Поле	Тип даних	Обов'язковість	Примітки
delivery_id	int(32)	Так	key
delivery_name	varchar	Так	

Таблиця 3.4 – Структура сутності «orders»

Поле	Тип даних	Обов'язковість	Примітки
orders_id	int(32)	Так	key
customers_id	int(32)	Так	
delivery_id	int(32)	Так	
product_id	int(32)	Так	

Продовження таблиці 3.4

stats_id	int(32)	Так	
date_order	varchar	Так	

Таблиця 3.5 – Структура сутності «categories»

Поле	Тип даних	Обов'язковість	Примітки
categories_id	int(32)	Так	key
name	varchar	Так	
description	varchar	Так	
image	blob	Так	

Таблиця 3.6 – Структура сутності «content_favorites»

Поле	Тип даних	Обов'язковість	Примітки
content_favorites_id	int(32)	Так	key
product_id	int(32)	Так	

Таблиця 3.7 – Структура сутності «favorites»

Поле	Тип даних	Обов'язковість	Примітки
favorites_id	int(32)	Так	key
content_favorites_id	int(32)	Так	
customers_id	int(32)	Так	

Таблиця 3.8 – Структура сутності «coments_product»

Поле	Тип даних	Обов'язковість	Примітки
comment_id	int(32)	Так	key

Продовження таблиці 3.8.

customers_id	int(32)	Так	
product_id	int(32)	Так	
date_com	date	Так	
comment	varchar	Так	

Таблиця 3.9 – Структура сутності «manufacturer»

Поле	Тип даних	Обов'язковість	Примітки
manufacturer_id	int(32)	Так	key
manufacturer_name	varchar	Так	

Таблиця 3.10 – Структура сутності «order_status»

Поле	Тип даних	Обов'язковість	Примітки
status_id	int(32)	Так	key
status_name	varchar	Так	

3.3 Проєктування користувацького інтерфейсу

Розробка інтернет-магазину, як і будь-якого іншого сайту, починається з дизайну. Першим етапом стало створення схем сторінок (wireframes), які деталізують розташування інформації та елементів управління на кожній сторінці. Ці схематичні макети показують, яка інформація повинна бути відображена, і дозволяють визначити пріоритетність елементів – що на сторінці є ключовим, а що другорядним [15]. Wireframes мають виконувати кілька основних завдань:

- відображати основні блоки контенту;
- демонструвати інформаційну структуру;
- показувати, як користувач взаємодітиме з інтерфейсом, надаючи приблизну візуалізацію цього процесу.

Wireframes слід розглядати як каркас майбутнього дизайну, де представлені всі важливі елементи кінцевого продукту. Важливо розуміти, що це не фінальний варіант дизайну, а умовна схема, де розміри та пропорції можуть бути змінені на подальших етапах розробки.

Для даного проекту були створені такі прототипи:

- прототип головної сторінки, де користувачі отримують загальну інформацію та доступ до ключових розділів магазину;
- прототип сторінки реєстрації, що включає поля для створення облікового запису;
- прототип сторінки категорії товарів, яка допомагає користувачам легко знаходити необхідні товари;
- прототип сторінки товару, що містить детальну інформацію про конкретний продукт і можливість його придбання.

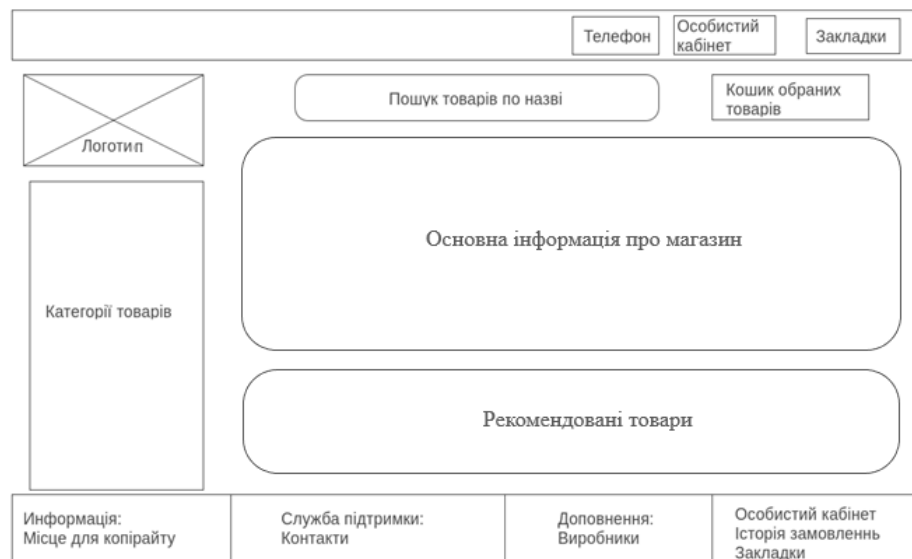


Рисунок 3.12 – Прототип головної сторінки

На цьому wireframe головної сторінки інтернет-магазину представлені основні елементи:

а) Заголовок (header):

1) телефон – контактний номер для зворотного зв'язку, щоб клієнти могли звернутися до магазину;

2) особистий кабінет – форма реєстрації або входу до свого облікового запису;

3) закладки – форма, що дозволяє користувачам додавати обрані товари до списку бажаних.

б) Лівий верхній кут:

1) логотип – розташований у верхньому лівому куті, який асоціює сайт з брендом магазину.

в) Центральна верхня частина:

1) форма пошуку товарів по назві – дозволяє користувачам здійснювати пошук товарів за їх назвою.

г) Правий верхній кут:

1) кошик обраних товарів – розділ, де відображаються товари, які клієнт планує купити.

д) Ліва частина сторінки:

1) категорії товарів – колонка, що містить перелік категорій.

е) Центральна частина:

1) основна інформація про магазин – у цьому блоці можна розмістити опис магазину, актуальні пропозиції або акції;

2) рекомендовані товари – секція з персоналізованими рекомендаціями.

ж) Нижня частина сторінки (footer):

1) інформація: місце для копірайту – блок для інформації про авторські права;

2) служба підтримки: контакти – інформація про службу підтримки та контакти магазину;

3) доповнення: виробники – додаткова інформація, що включає список виробників, товари яких представлені на сайті;

4) особистий кабінет, історія замовлень, закладки – швидкий доступ до персональних функцій для зручної навігації користувачів.

На рисунку 3.13 зображений прототип сторінки з формою реєстрації.

Основна інформація

Ім'я

Прізвище

E-Mail

Телефон

Ваш пароль

Пароль

Підтвердження пароля

Розсилка новин

Підписатися на новини ☐ Так ☐ Ні

Рисунок 3.13 – Прототип схема форми реєстрації

Реєстрація надає користувачам переваги, такі як швидке оформлення замовлення і можливість залишати відгуки на товари. Дані, введені у формі реєстрації, надсилаються до бази даних, а пароль зберігається у зашифрованому вигляді для безпеки.

На рисунку 3.14 представлений прототип сторінки з категоріями товарів, яка включає наступні елементи:

а) Заголовок (header):

- 1) логотип: розташований у лівій частині;
- 2) пошук товарів по назві: центр верхньої частини сторінки;
- 3) телефон, особистий кабінет та закладки: у правій верхній частині;
- 4) кошик обраних товарів: розташований праворуч від пошукового поля;

б) Центральна частина:

1) слайдер з товарами: великий блок у центрі, де відображаються товари, можливо, промо-акції або популярні продукти;

2) сортування: блок для сортування товарів за різними критеріями (наприклад, за ціною, популярністю тощо).

в) Товари під слайдером зображені товари у вигляді карток:

- 1) фото товару: зображення товару;
- 2) ціна товару: інформація про вартість;
- 3) кнопка "купити": для додавання товару до кошика.

г) Підвал (footer):

- 1) інформація про магазин: ліва частина — місце для копірайту;

2) служба підтримки та контакти: центр підвалу для контактних даних магазину;

3) додаткова інформація (виробники): інформація про бренди або постачальників товарів;

4) особистий кабінет, історія замовлень, закладки: доступ до історії замовлень і закладок користувача.

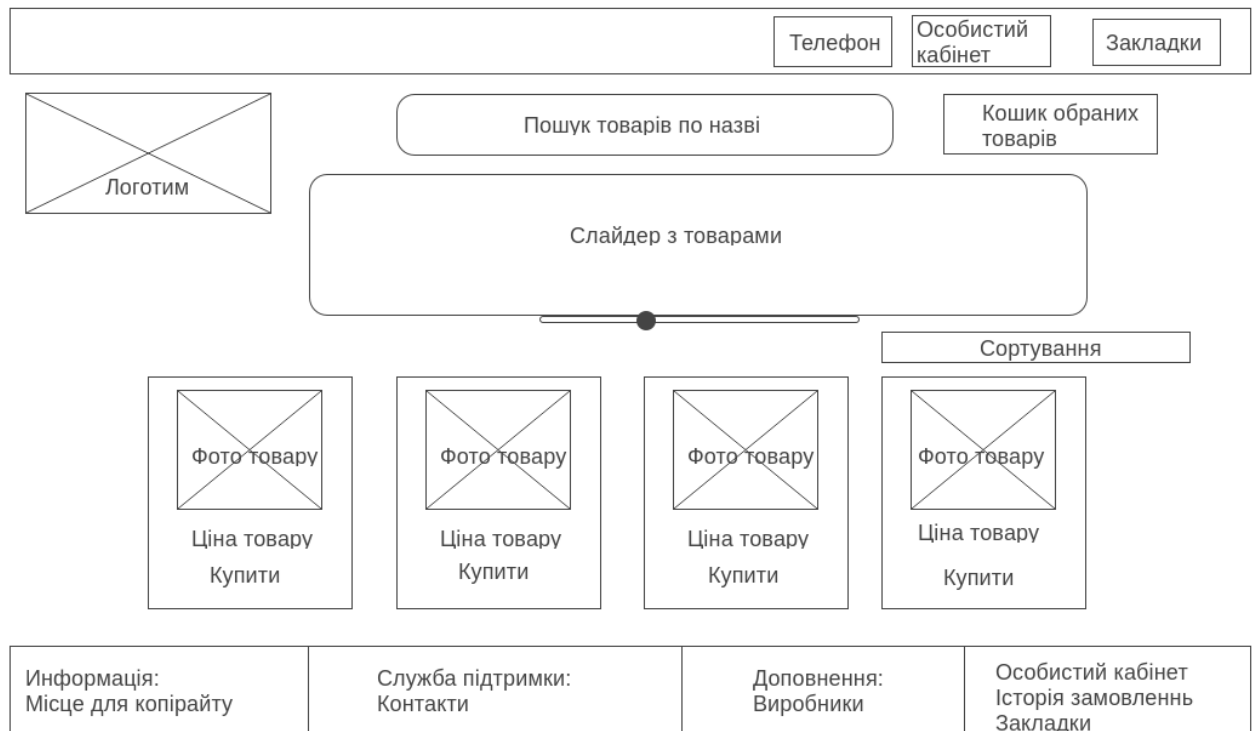


Рисунок 3.14 – Прототип сторінки з категоріями

На рисунку 3.15 представлено прототип сторінки товару, яка є ключовою в реалізації товару. На ній присутні такі основні елементи:

- логотип – у верхньому лівому куті, позначений великим порожнім квадратом;
- телефон, особистий кабінет, закладки – праворуч у верхній частині сторінки, є контактні дані та доступ до особистого кабінету покупця;
- пошук товарів по назві – розміщений у центрі у вигляді пошукового поля;
- кошик обраних товарів – праворуч від пошуку, для відображення обраних покупок;

- назва товару, виробник, модель, наявність, ціна – ключова інформація про товар, що міститься в основній частині сторінки;
- опис, характеристики, відгуки – три вкладки під інформацією про товар, які дозволяють переглянути його деталі;
- рекомендовані товари – розташовані внизу сторінки;
- додаткові елементи – інформація для копірайту, контакти служби підтримки, виробники та доступ до особистого кабінету.

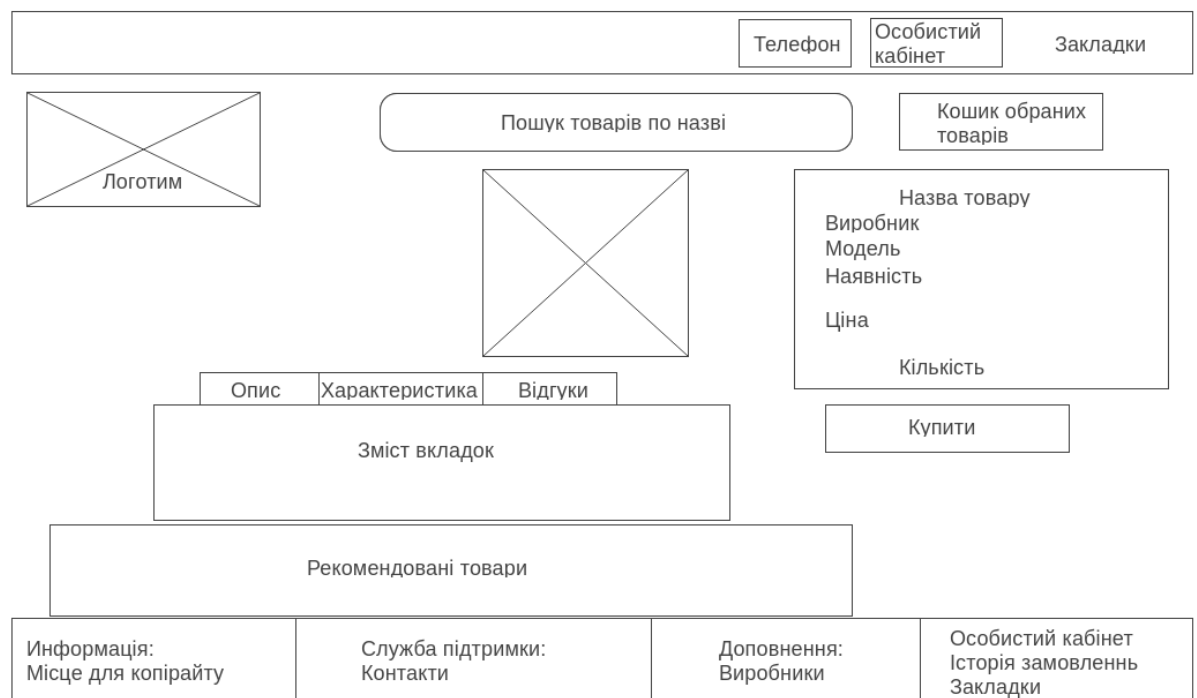


Рисунок 3.16 – Макет сторінки товару

4 РЕАЛІЗАЦІЯ СИСТЕМИ

4.1 Вибір інструментів реалізації

Інтернет-магазин розміщено на локальному сервері, що робить цей варіант не лише економічним, але й оптимально підходящим для нашого проєкту.

Використання локального сервера дозволяє скоротити витрати на хостинг та отримати більше контролю над безпекою та доступністю ресурсу.

Для автоматизації процесів обрано CMS OpenCart, яка пропонує великий набір інструментів для побудови та управління інтернет-магазинами. Однією з ключових переваг OpenCart є її мінімалістичний підхід на етапі початкового встановлення: система містить тільки базові елементи, які можна розширювати в разі потреби. Це дозволяє уникнути перевантаження адміністративної панелі зайвими елементами, знижуючи навантаження на сервер і оптимізуючи використання простору на хостингу.

OpenCart із його багатомовною підтримкою значно спрощує локалізацію інтернет-магазину, що дозволяє швидко адаптувати сайт під нові ринки або залучати клієнтів з різних країн. Модульна структура OpenCart дає змогу легко інтегрувати додаткові функції, такі як різноманітні платіжні системи або способи доставки, що розширює можливості магазину без зайвого навантаження на сервер.

Open Server 5.2.2, обраний для локальної розробки, дозволяє швидко та безпечно тестувати нові модулі чи оновлення, проводити відлагодження системи, а також забезпечує повну автономність. Наявність таких інструментів, як Apache, PHP і MySQL, створює майже ідентичне до реального серверне середовище, де можна перевірити всі аспекти роботи сайту до виходу у продакшн. Це знижує ризик технічних збоїв та оптимізує час на розгортання нових функцій або оновлень. Такий підхід дає змогу гнучко масштабувати інтернет-магазин, залишаючи запас на майбутній розвиток і інтеграцію нових функцій.

Apache Server – це вебсервер, який може працювати на локальному комп'ютері та підтримує одночасну роботу багатьох сайтів. Це забезпечує зручність при розробці й тестуванні різних проєктів, особливо корисно для перевірки сценаріїв на PHP. Локальне середовище дозволяє швидко вносити зміни та тестувати їх у реальному часі без необхідності завантаження файлів на віддалений сервер, що значно скорочує цикл розробки та полегшує процес налагодження.

Крім того, Apache підтримує налаштування віртуальних хостів, що дозволяє розробникам імітувати середовище, близьке до реального, для кожного проєкту. Такий підхід підвищує ефективність тестування, зменшує ймовірність помилок при переході на продакшн-сервер і дозволяє зосередитись на розвитку функціоналу без зайвих витрат часу на завантаження та розгортання змін.

phpMyAdmin – це вебдодаток з відкритим кодом, створений на мові PHP, що надає зручний інтерфейс для управління базами даних MySQL. Через phpMyAdmin адміністратори та розробники можуть виконувати різні операції з базами даних, такі як створення та редагування таблиць, запуск SQL-запитів, імпорт і експорт даних. Для роботи з цим інструментом потрібен лише браузер, який передає команди на сервер, що дозволяє легко адмініструвати бази даних навіть без глибоких знань SQL.

PHP 5 – це версія популярної скриптової мови програмування PHP, яка широко застосовується для розробки динамічних вебсайтів та вебдодатків. Завдяки своїй універсальності та підтримці більшістю хостинг-провайдерів, PHP залишається однією з найбільш популярних мов для створення інтерактивних вебсайтів. PHP 5, зокрема, додав підтримку об'єктно-орієнтованого програмування, покращену продуктивність і безліч нових функцій, які полегшують розробку великих і складних вебдодатків.

Такі інструменти, як phpMyAdmin і PHP, у поєднанні з MySQL, створюють потужне середовище для розробки та управління динамічними проєктами, забезпечуючи зручність адміністрування і підтримку великої кількості функцій, необхідних для сучасних вебресурсів.

Google Analytics – це вебсервіс для аналізу статистики та поведінки користувачів на вебсайтах, розроблений компанією Google. Він дозволяє власникам сайтів та розробникам отримувати детальну інформацію про відвідувачів, включаючи дані про кількість візитів, джерела трафіку, час перебування на сторінці, географічне розташування, використані пристрої та інші показники. Google Analytics допомагає аналізувати ефективність маркетингових кампаній і вивчати поведінкові шаблони, що дозволяє приймати обґрунтовані

рішення для поліпшення контенту, структури та залучення аудиторії. Встановлення Google Analytics здійснюється через додавання спеціального коду на сторінки сайту, який збирає дані та передає їх у вигляді звітів, доступних через вебінтерфейс сервісу.

TensorFlow – це бібліотека з відкритим кодом, розроблена компанією Google для побудови й навчання моделей машинного навчання та штучного інтелекту. TensorFlow підтримує численні алгоритми глибокого навчання та дозволяє створювати складні нейронні мережі для розв'язання різних завдань, таких як обробка зображень, розпізнавання мови, аналіз тексту та багато інших. Бібліотека написана на мові Python і оптимізована для швидкої обробки великих обсягів даних, що робить її популярним вибором серед дослідників і розробників у сфері штучного інтелекту. TensorFlow може працювати як на локальних машинах, так і на хмарних сервісах, підтримуючи багатоплатформову розробку і розширене налаштування, що дозволяє розробникам створювати рішення, які масштабуються відповідно до потреб бізнесу або проєкту.

4.2 Опис основних етапів технологічного циклу

Основний технологічний цикл інтернет-магазину з персоналізованою рекламою передбачає кілька етапів. Спершу система збирає і обробляє дані про дії користувачів на сайті, включаючи перегляди сторінок, історію пошуку, зроблені покупки та кліки. Для цього використовуються аналітичні інструменти, які збирають анонімну інформацію, щоб дотримуватися стандартів конфіденційності.

На основі зібраних даних формується користувацький профіль. Далі завдяки аналітиці великих даних та алгоритмам машинного навчання користувачі сегментуються за інтересами, звичками та поведінковими характеристиками, що дозволяє забезпечити персоналізацію. Це стосується не тільки показу відповідних товарів, але і вибору рекламних банерів з акційними пропозиціями.

Наступний етап – це створення динамічних банерів, які підлаштовуються під конкретного користувача. Наприклад, якщо користувач цікавився певними товарами, система налаштовує банери з акціями саме на такі категорії, що збільшує вірогідність залучення.

Після цього проводиться аналіз ефективності: система відстежує, скільки разів користувач натискав на банер, скільки часу проводив на сторінці, що рекламується, а також інші показники.

Цей цикл забезпечує тісну взаємодію з базою даних, куди надходять всі користувацькі запити і де зберігається історія поведінки клієнтів. Це вимагає наявності потужних інструментів персоналізації, які дозволяють динамічно змінювати банери відповідно до інтересів кожного відвідувача.

4.3 Встановлення та налаштування системи «OpenCart»

Для встановлення OpenCart завантажуюмо останню версію з офіційного сайту та розпаковуємо її на сервер у потрібну директорію. Створюємо базу даних через панель керування хостингом і відкриваємо URL нашого сайту в браузері, щоб запустити майстер установки. Заповнимо дані для підключення до бази даних та створюємо адміністратора магазину. Після завершення інсталяції видаляємо папку «install» для безпеки, а згодом переходимо до панелі управління магазину за адресою /admin, використовуючи створений логін і пароль.

На рисунку 4.1. зображено вигляд головної сторінки для завантаження «OpenCart» з офіційного сайту.

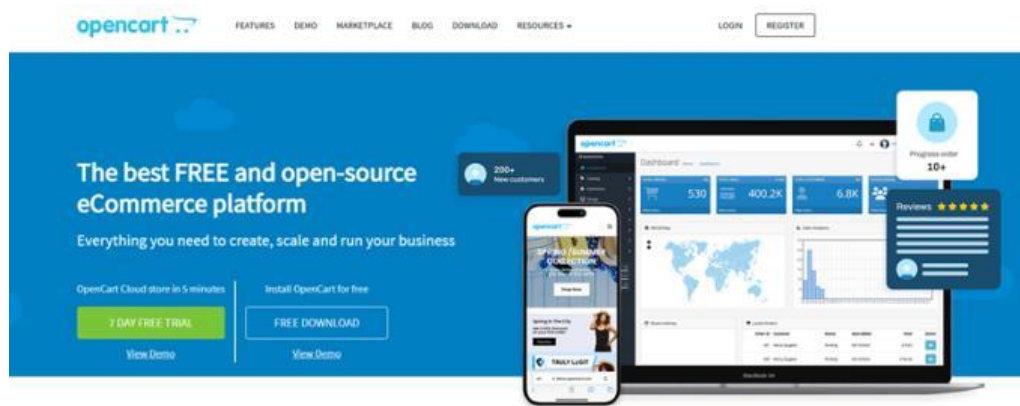


Рисунок 4.1 – Головна сторінка сайту «OpenCart»

Далі необхідно встановити «OpenServer», для того щоб встановити наш інтернет-магазин на локальний вебсервер. Спочатку завантажуюємо дистрибутив із офіційного сайту, обираємо версію Ultimate та розпаковуємо архів у робочу папку на комп'ютері [16]. Після розпакування відкриваємо папку з програмою та запускаємо файл OpenServer.exe.

Під час першого запуску налаштовуємо сервер, вибравши потрібні компоненти, такі як необхідні параметри PHP, PhpMyAdmin, Apache останньої актуальної версії, у налаштуваннях програми. Після цього запускаємо сервер й тепер можемо використовувати його для запуску нашого сайту.

Офіційну сторінку завантаження «OpenServer» представлено на рисунку 4.2.



Рисунок 4.2 – Офіційна сторінка завантаження «OpenServer»

4.4 Розробка бази даних

Наступним кроком стане створення бази даних. Для цього необхідно запустити «OpenServer» і знайти на панелі інструментів програму PhpMyAdmin. Після натискання відкриється вікно браузера з вкладкою авторизації в цій програмі. Вікно авторизації наведено на рисунку 4.3. За замовчуванням встановлено логін: root, пароль не потрібен.

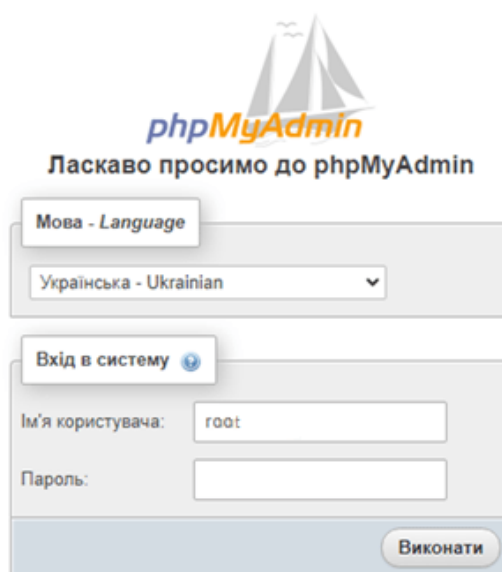


Рисунок 4.3 – Вікно авторизації в PhpMyAdmin

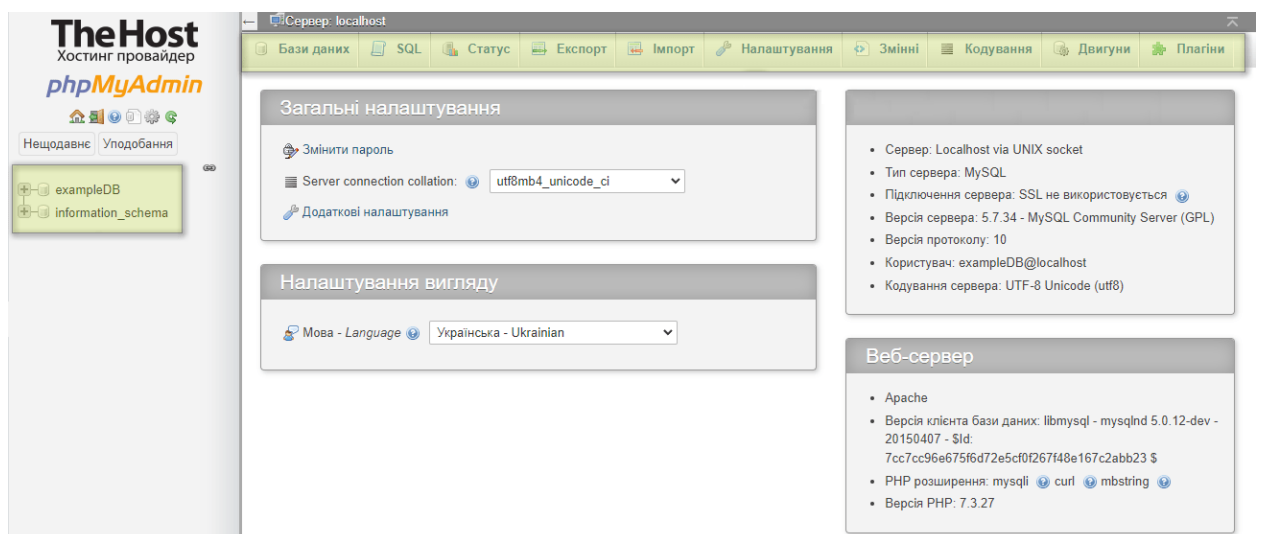


Рисунок 4.4 – Інтерфейс головної сторінки в PhpMyAdmin

Після входу в базу даних необхідно створити сутності, описані в розділі 3.2: «categories», «customers», «comment», «favorites», «content_favorites», «delivery», «manufacturer», «orders», «order_status», «product». Кожну сутність слід налаштувати відповідно до їхньої структури та властивостей. Важливо задати зв'язки між цими сутностями для забезпечення коректної взаємодії даних, що дозволить створити цілісну і функціональну базу даних для подальшого використання.

На рисунку 4.5 створена сутність «categories».

	#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию
☐	1	category_id	int(32)			Нет	Нет
☐	2	name	varchar(255)	utf8_general_ci		Нет	Нет
☐	3	discription	varchar(255)	utf8_general_ci		Нет	Нет
☐	4	image	blob			Нет	Нет

Рисунок 4.5 – Сутність «categories»

На рисунку 4.6 створена сутність «comment».

	#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию
☐	1	comment_id	int(32)			Нет	Нет
☐	2	customers_id	int(32)			Нет	Нет
☐	3	product_id	int(32)			Нет	Нет
☐	4	date_com	date			Нет	Нет
☐	5	comment	varchar(255)	utf8_general_ci		Нет	Нет

Рисунок 4.6 – Сутність «comment»

На рисунку 4.7 створена сутність «content_favorites».

	#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию
☐	1	content_favorites_id	int(32)			Нет	Нет
☐	2	product_id	int(32)			Нет	Нет

Рисунок 4.7 – Сутність «content_favorites»

На рисунку 4.8 створена сутність «customers».

	#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию
☐	1	customers_id	int(32)			Нет	Нет
☐	2	name	varchar(255)	utf8_general_ci		Нет	Нет
☐	3	surname	varchar(255)	utf8_general_ci		Нет	Нет
☐	4	password	varchar(255)	utf8_general_ci		Нет	Нет
☐	5	email	char(255)	utf8_general_ci		Нет	Нет
☐	6	phone_number	int(32)			Нет	Нет

Рисунок 4.8 – Сутність «customers»

На рисунку 4.9 створена сутність «delivery».

	#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию
☐	1	delivery_id	int(32)			Нет	Нет
☐	2	delivery_name	varchar(255)	utf8_general_ci		Нет	Нет

Рисунок 4.9 – Сутність «delivery»

На рисунку 4.10 створена сутність «favorites».

	#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию
☐	1	favorites_id	int(32)			Нет	Нет
☐	2	content_favorites_id	int(32)			Нет	Нет
☐	3	customers_id	int(32)			Нет	Нет

Рисунок 4.10 – Сутність «favorites»

На рисунку 4.11 створена сутність «manufacturer».

	#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию
☐	1	manufacturer_id	int(32)			Нет	Нет
☐	2	manufacturer_name	varchar(255)	utf8_general_ci		Нет	Нет

Рисунок 4.11 – Сутність «manufacturer»

На рисунку 4.1 створена сутність «order».

	#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию
☐	1	orders_id	int(32)			Нет	Нет
☐	2	customers_id	int(32)			Нет	Нет
☐	3	delivery_id	int(32)			Нет	Нет
☐	4	product_id	int(32)			Нет	Нет
☐	5	status_id	int(32)			Нет	Нет
☐	6	date_order	date			Нет	Нет
☐	7	address	varchar(255)	utf8_general_ci		Нет	Нет

Рисунок 4.12 – Сутність «orders»

На рисунку 4.13 створена сутність «order_status».

	#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию
☐	1	status_id	int(32)			Нет	Нет
☐	2	status_name	varchar(255)	utf8_general_ci		Нет	Нет

Рисунок 4.13 – Сутність «order_status»

На рисунку 4.14 створена сутність «product».

	#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию
☐	1	product_id	int(32)			Нет	Нет
☐	2	category_id	int(32)			Нет	Нет
☐	3	manufacturer_id	int(32)			Нет	Нет
☐	4	product_name	varchar(255)	utf8_general_ci		Нет	Нет
☐	5	price	float			Нет	Нет
☐	6	attributes	varchar(255)	utf8_general_ci		Нет	Нет
☐	7	description	varchar(255)	utf8_general_ci		Нет	Нет
☐	8	image	blob			Нет	Нет

Рисунок 4.14 – Сутність «product»

4.5 Під'єднання Google Analytics до вебресурсу

Щоб підключити наш вебресурс до Google Analytics, переходимо на сайт Google Analytics і входимо в обліковий запис Google [17]. Після цього натискаємо «Створити обліковий запис», щоб створити новий обліковий запис. Головна сторінка реєстрації зображена на рисунку 4.15.

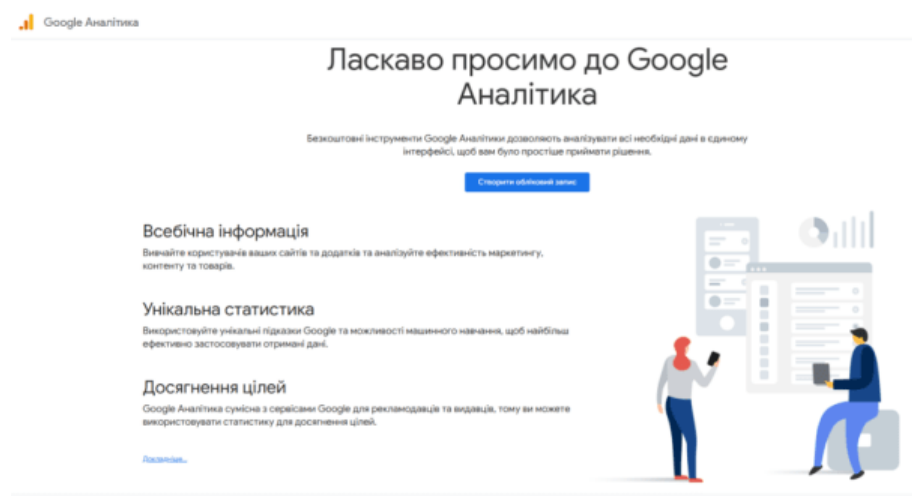


Рисунок 4.15 – Сторінка реєстрації «Google Analytics»

Далі заповнюємо поля з основною інформацією, такою як назва ресурсу, URL сайту, часовий пояс і галузь, до якої належить ваш сайт. Ці етапи представлені на рисунках 4.15 – 4.19.

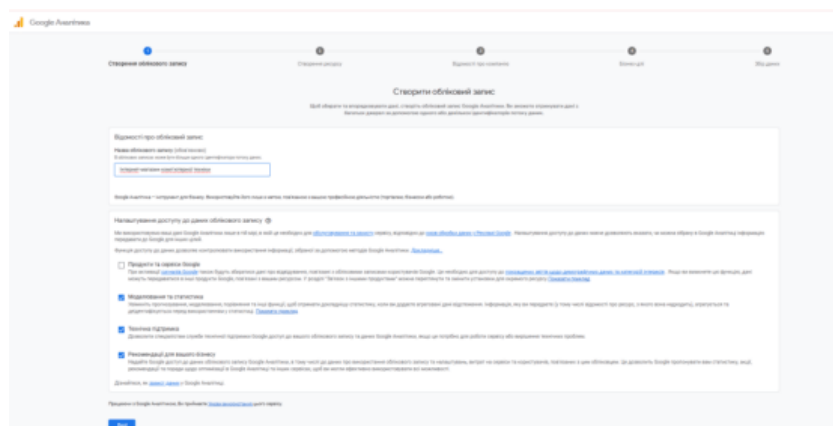


Рисунок 4.15 – Сторінка створення назви облікового запису «Google Analytics»

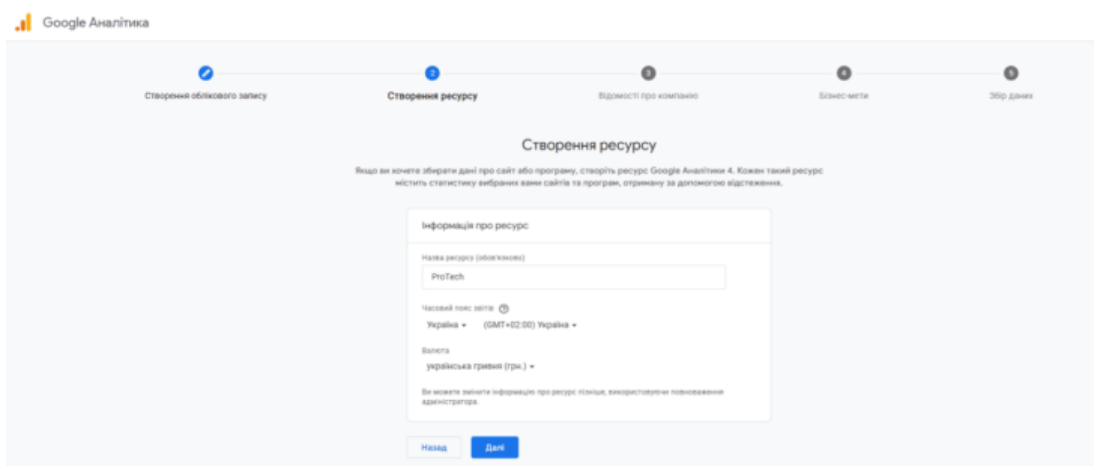


Рисунок 4.16 – Сторінка створення ресурсу на «Google Analytics»

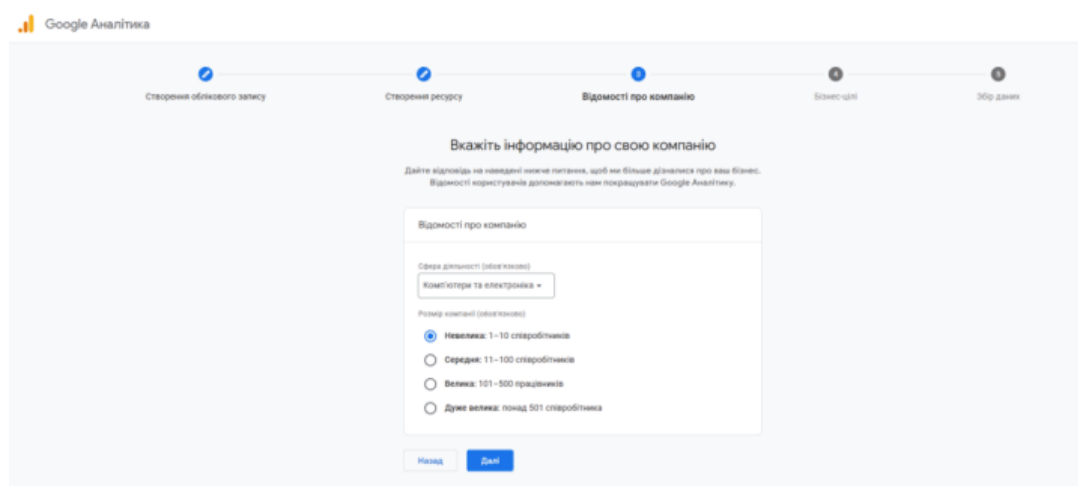


Рисунок 4.17 – Сторінка інформації про компанію на «Google Analytics»

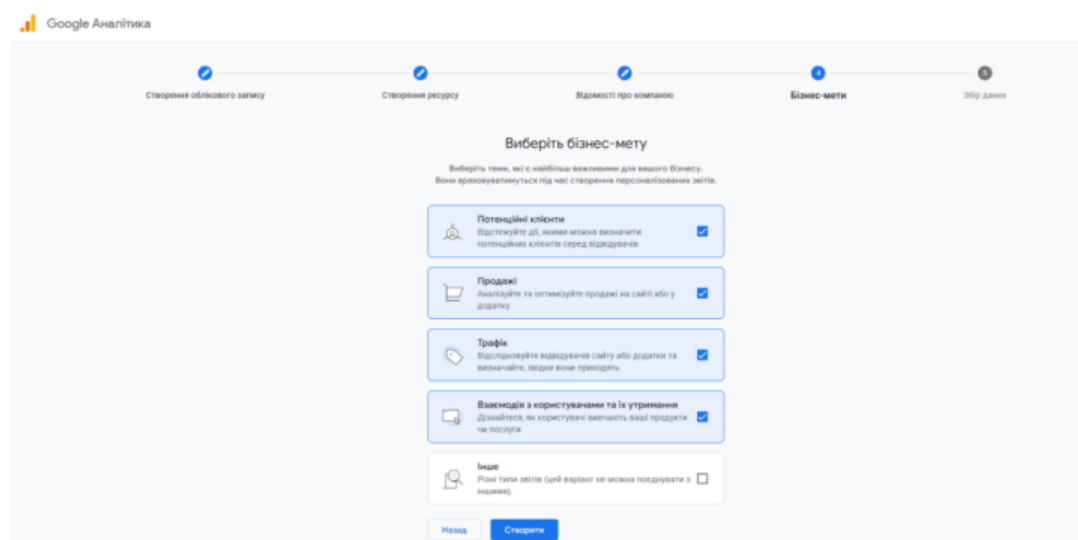


Рисунок 4.18 – Сторінка з визначенням мети вебресурсу на «Google Analytics»

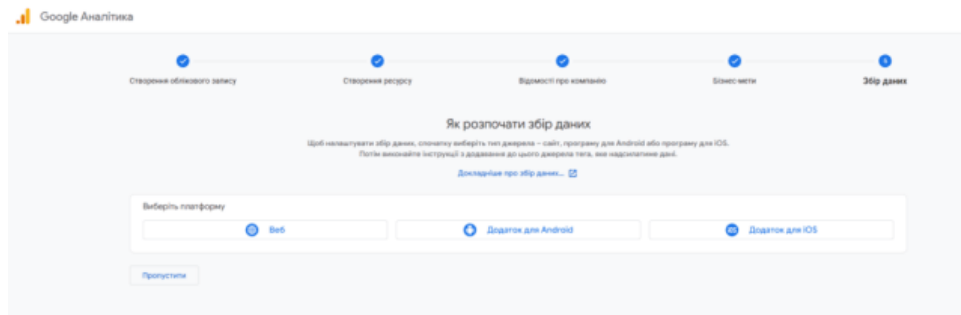


Рисунок 4.19 – Вибір платформи на «Google Analytics»

Обираємо платформу «Веб», де потрібно вказати URL вебсайту та назва потоку. Натискаємо «Створити та продовжити» для наступних налаштувань. Загальний вид сторінки зображено на рисунку 4.20.

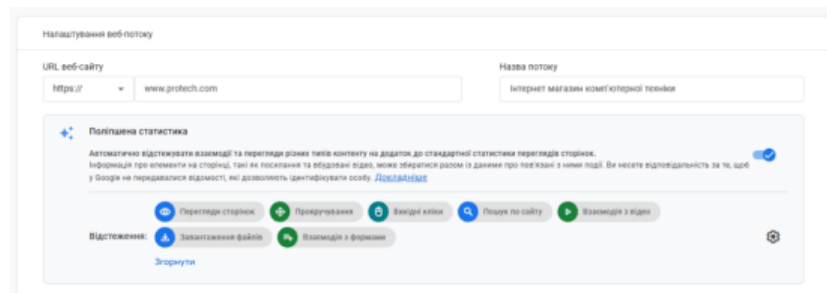


Рисунок 4.20 – Сторінка налаштування потоку даних на «Google Analytics»

Після створення облікового запису, надано унікальний ідентифікатор відстеження (Tracking ID), який має вигляд G-CJK4LJC85K. Цей ідентифікатор використовується для підключення нашого сайту до Google Analytics. Відомості про вебресурс представлено на рисунку 4.21.

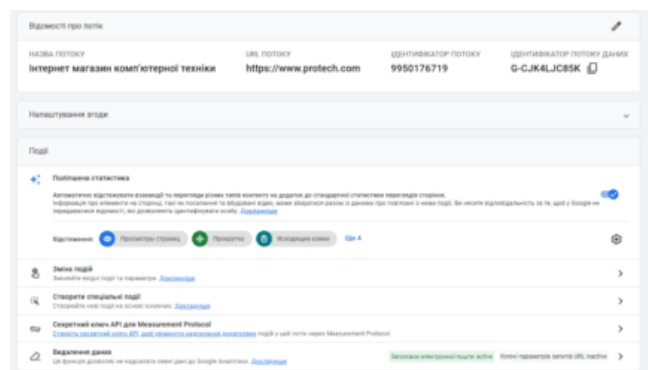


Рисунок 4.21 – Відомості про вебресурс

Далі потрібно встановити код відстеження на наш сайт. Копіюємо наданий Google Analytics код, який надано у лістингу 4.1 і вставляємо його у HTML сайту в секцію <head>. За допомогою цього, з'являється можливість збирати дані про відвідувачів сайту.

Лістинг 4.1 – Фрагмент коду від Google Analytics для відстеження сайту

```
<!-- Google tag (gtag.js) -->
<script async src="https://www.googletagmanager.com/gtag
/js?id=G-CJK4LJC85K"></script>
<script>
  window.dataLayer = window.dataLayer || [];
  function gtag(){dataLayer.push(arguments);}
  gtag('js', new Date());
  gtag('config', 'G-CJK4LJC85K');
</script>
```

Щоб переконатися, що підключення працює, переходимо в Google Analytics у розділ «Realtime» (або «У реальному часі»). Відкриваємо сайт в новій вкладці браузера і бачимо активних користувачів на сайті в режимі реального часу.

4.5 Інтеграція TensorFlow для персоналізації інтернет-магазину

TensorFlow є важливим інструментом для реалізації персоналізованої реклами в онлайн-магазинах техніки, оскільки він забезпечує потужну платформу для створення моделей машинного навчання. За допомогою цієї бібліотеки машинного навчання можна обробляти великі обсяги даних про поведінку користувачів, їхні уподобання та історію покупок. Це дозволяє створювати моделі, які прогнозують інтереси клієнтів і формують персоналізовані

рекомендації, наприклад, відображення товарів, які можуть зацікавити конкретного користувача.

Застосування TensorFlow також допомагає оптимізувати витрати на рекламу. Використовуючи дані про поведінку клієнтів, алгоритми можуть визначати найбільш ефективні пропозиції, які будуть відображатися на банерах або в інших рекламних блоках сайту. Це підвищує рівень конверсій і задоволеність клієнтів, роблячи взаємодію з платформою максимально зручною та релевантною. У цілому TensorFlow забезпечує надійний інструмент для впровадження інноваційних підходів у маркетингу, які відповідають сучасним потребам бізнесу.

Для підключення TensorFlow до проєкту інтернет-магазину встановлюємо його у нашому середовищі розробки. Спочатку встановлюємо TensorFlow за допомогою команди `pip install tensorflow`, а також Flask для розгортання API (`pip install flask`). Після цього створюємо та натреновуємо модель рекомендації і налаштовуємо Flask для обслуговування API, щоб наш сайт міг надсилати запити до моделі і отримувати рекомендації.

У лістингу 4.2 наведено код для створення REST API, який приймає запити з ID користувача і повертає рекомендації на основі передбачень моделі.

Лістинг 4.2 – Створення REST API

```
pip install tensorflow flask
from flask import Flask, request, jsonify
import tensorflow as tf
model = tf.keras.models.load_model("recommendation_model.h5")
app = Flask(__name__)
@app.route('/recommend', methods=['POST'])
def recommend():
    data = request.get_json()
    prediction = model.predict([data['user_id']])
    return jsonify(prediction.tolist())
if __name__ == '__main__':
    app.run(debug=True)
```

4.6 Реалізація користувацького інтерфейсу

Користувацький інтерфейс є ключовим компонентом будь-якого програмного забезпечення, адже саме він визначає, наскільки зручно користувачі можуть взаємодіяти із системою. У цьому підрозділі розглядаються підходи, інструменти та принципи, які використовуються для розробки інтуїтивного, естетичного та функціонального інтерфейсу.

Як видно на рисунку 4.22, був реалізований прототип головної сторінки сайту інтернет-магазину «ProTech». Розберемо цю сторінку детальніше по її складовим:

- верхня частина (header): містить логотип магазину, пошуковий рядок, номер телефону для зв'язку, іконку «особистий кабінет», а також кошик покупок.
- ліва частина: розташована вертикальна панель із каталогом товарів, у якій перелічено основні категорії, кожна категорія веде до відповідної сторінки.
- центральна частина: представлено текстовий блок з загальною інформацією про магазин;
- секція «Спеціально для вас підібрали»: розташована в центральній частині сторінки нижче текстового блоку, ця секція містить карусель із добіркою товарів, яка представляє персоналізовану рекламу, де кожен товар супроводжується зображенням, назвою, ціною, рейтингом;
- нижня частина сайту (footer): містить логотип магазину та три колонки з посиланнями: для клієнтів (доставка, гарантії, контакти), юридичну інформацію (публічна оферта, політика конфіденційності) та дані про продавця (сертифікації, карта сайту).

Наступний етап – реалізація сторінки категорії, що розміщена на рисунку 4.23, яка є ключовим елементом для структуризації контенту та покращення навігації користувачів.

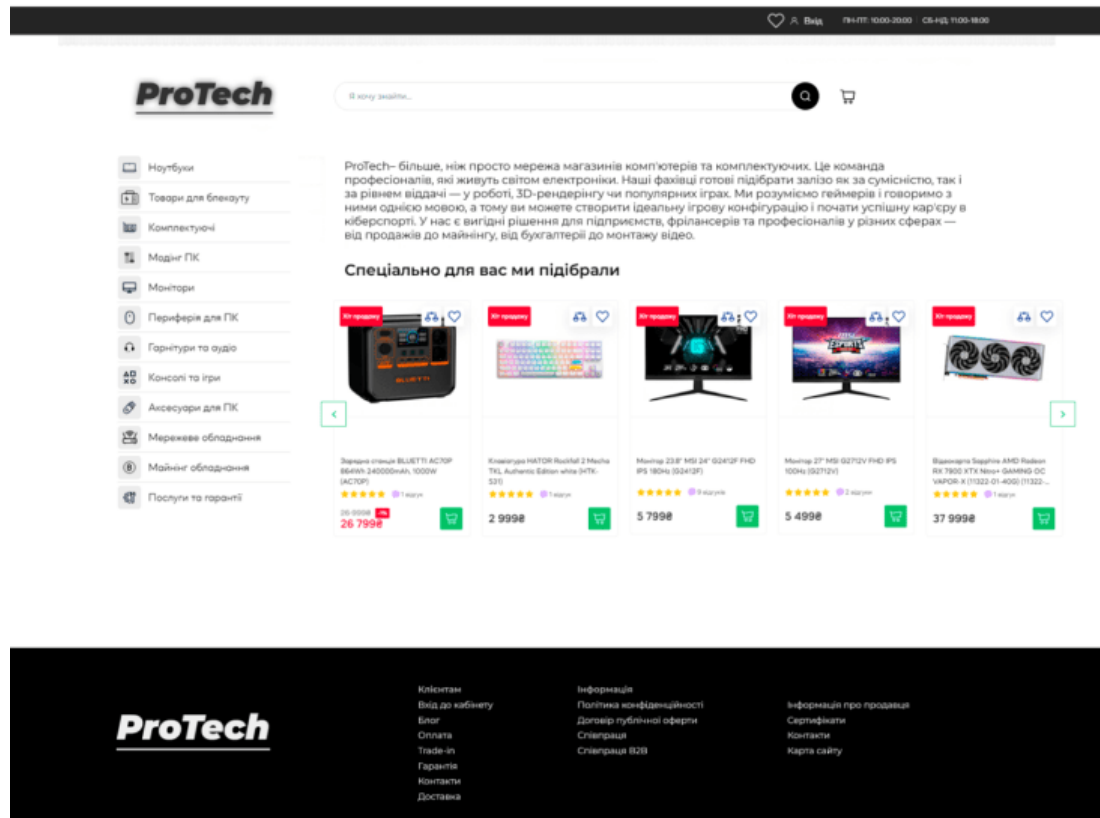


Рисунок 4.22 – Прототип головної сторінки сайту інтернет-магазину «ProTech»

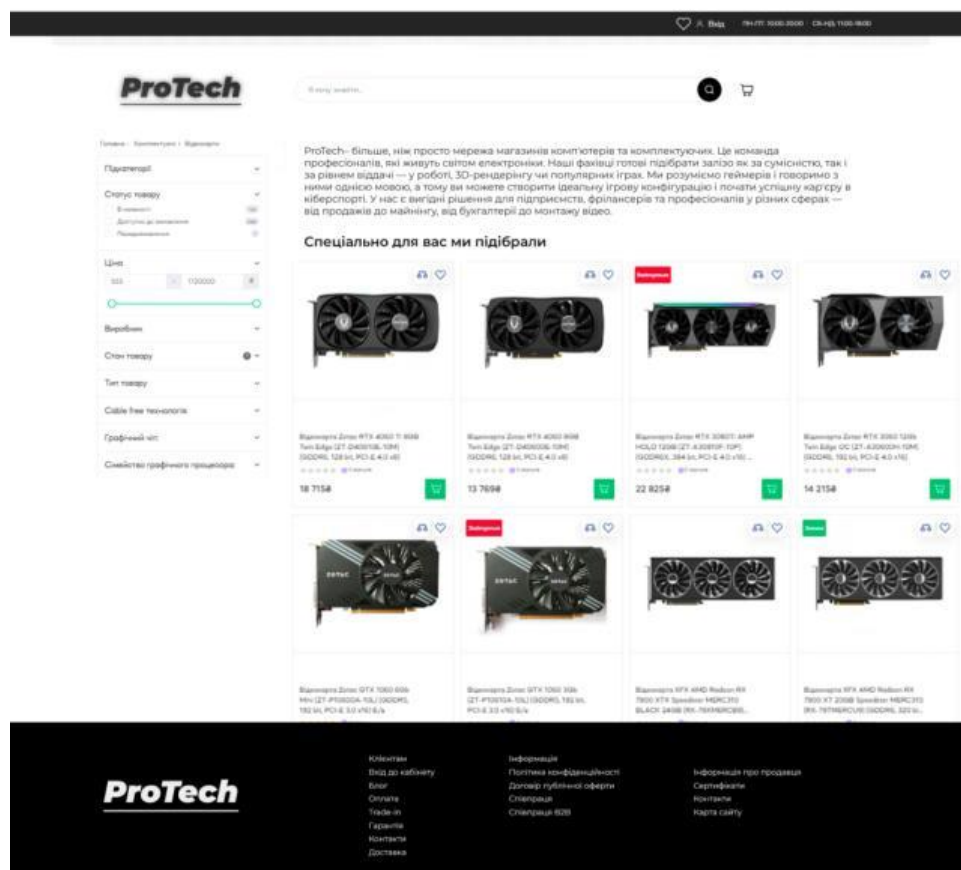


Рисунок 4.23 – Сторінка категорії сайту інтернет-магазину «ProTech»

На даній сторінці розміщується:

- у верхній частині (header) розташовані логотип, пошуковий рядок, контактна інформація, а також доступ до "Особистого кабінету", обраних товарів і кошика;
- ліва частина сторінки містить фільтри для зручного пошуку, серед яких стан товару, ціновий діапазон, виробник, тип та інші характеристики;
- у центральній частині є опис бренду, а нижче – список рекомендованих товарів із зображеннями, цінами, короткими характеристиками та кнопками для додавання в кошик чи обране;
- у підвалі (footer) розміщені посилання на корисну інформацію, зокрема про доставку, гарантію, контакти, блог, а також карта сайту.

На рисунку 4.24 зображена форма для оформлення замовлення, призначена для користувачів, які не мають облікового запису. Вона дозволяє швидко ввести необхідні дані для здійснення покупки, забезпечуючи зручність та простоту процесу, навіть якщо клієнт вперше відвідує сайт.

КЛІЄНТ

* Телефон

* Повне ПІБ

Email

☐ Не телефонувати для підтвердження замовлення ⓘ

Залиште свій коментар

Ваш коментар

Рисунок 4.24 – Форма оформлення замовлення без облікового запису

Наступним кроком є розробка сторінки детальної інформації про товар, яка дозволить користувачам ознайомитися з ключовими характеристиками продукту, переглянути його зображення, залишити відгуки чи поставити запитання, а також швидко оформити покупку. Сторінка товару представлена на рисунку 4.25.

На цій сторінці можна побачити:

- у верхній частині, заголовок (header), який включає логотип, пошуковий рядок, номер телефону з графіком роботи, форму для входу в «особистий кабінет», кнопку закладки та кошик;
- у центральній частині сторінки розташована картка товару, яка містить зображення продукту, назву, технічні характеристики, детальний опис та кнопку «Купити» для додавання товару до кошика;
- нижче розташований блок «Відгуки та питання», де можна переглядати коментарі, ставити запитання або залишати свої відгуки.

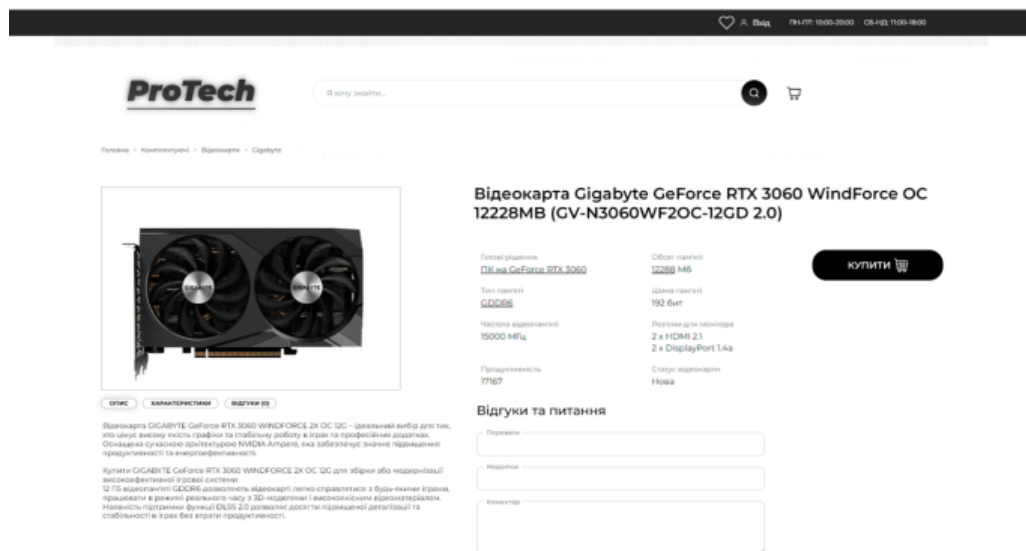


Рисунок 4.25 – Сторінка товару сайту інтернет-магазину «ProTech»

4.7 Програмна реалізація вебресурсу

У лістингу 4.3 наведено програмний код функції, що реалізує кошик товарів. Ця функція забезпечує додавання, видалення та управління товарами в кошику, що є ключовою частиною системи електронної комерції.

Лістинг 4.3 – Функція для реалізації товарного кошика

```
function add_to_cart($product_id, $count=1) {
    if(!empty($_SESSION['products'][$product_id])) {
        $_SESSION['products'][$product_id]['count']++;
    }
    else{
        $_SESSION['products'][$product_id]=array();

        $q="SELECT price FROM product
        WHERE id='$product_id'";
        $add_product=mysql_fetch_assoc(mysql_query($q));
        $_SESSION['products'][$product_id]['cost']
        =$add_product['price'];
        $_SESSION['products'][$product_id]['count']=$count;
    }
    update_cart();
}
```

У лістингу 4.2 наведено програмний код функції, яка реалізує підрахунок загальної вартості покупки. Ця функція обчислює підсумкову вартість товарів у кошику з урахуванням їх кількості, ціни та можливих знижок. Її використання спрощує процес фінансових розрахунків і забезпечує точність оформлення замовлення.

Лістинг 4.2 – Функція підрахунку загальної вартості покупки

```
function update_cart() {
    $_SESSION['products_incart']=count($_SESSION['products']);
    $_SESSION['cart_cost']=0;
    foreach($_SESSION['products'] as $key=>$value) {
        $_SESSION['cart_cost']+=$_SESSION['products'][$key]['cost']*
        $_SESSION['products'][$key]['count'];
    }
}
```

У лістингу 4.3 наведено програмний код функції, яка відповідає за занесення даних про реєстрацію користувача до бази даних. Ця функція забезпечує запис інформації, такої як ім'я, електронна пошта, пароль та інші реєстраційні дані, гарантуючи їхню безпечність та доступність для подальшої

обробки. Її впровадження є ключовим етапом у роботі системи управління користувачами.

Лістинг 4.3 – Функція занесення даних про реєстрацію до бази даних

```
<?php
if (isset($_POST['login'])) {
    $login = $_POST['login'];
    if ($login == '') {
        unset($login);
    }
}
if (isset($_POST['password'])) {
    $password = $_POST['password'];
    if ($password == '') {
        unset($password);
    }
}
if (empty($login) || empty($password)) {
    exit("Не вірно введені дані!");
}
$login = stripslashes($login);
$login = htmlspecialchars($login);
$password = stripslashes($password);
$password = htmlspecialchars($password);
$login = trim($login);
$password = trim($password);
include("bd.php");
$query = $db->prepare("SELECT customer_id FROM customers WHERE
name = ?");
$query->bind_param("s", $login);
$query->execute();
$result = $query->get_result();
if ($result->num_rows > 0) {
    exit("Цей логін зайнятий, введіть інший логін.");
}
$hashed_password = password_hash($password, PASSWORD_DEFAULT);
$query2 = $db->prepare("INSERT INTO customers (name, password)
VALUES (?, ?)");
$query2->bind_param("ss", $login, $hashed_password);
if ($query2->execute()) {
    echo "Вітаю з реєстрацією!";
} else {
    echo "Помилка.";
}
$query->close();
$query2->close();
$db->close();
?>
```

4.8 Ефекти персоналізації в інтернет магазині

Персоналізація в інтернет-магазинах є важливим інструментом підвищення ефективності продажів і покращення користувацького досвіду. Завдяки аналізу даних про поведінку клієнтів, їхні вподобання та історію покупок, система здатна формувати індивідуальні рекомендації та пропозиції [18]. У цьому розділі розглядаються результати впровадження персоналізації, її вплив на якість обслуговування користувачів і економічну вигоду для бізнесу, а саме:

- система використовує алгоритми машинного навчання TensorFlow для аналізу поведінки користувачів, історії їхніх покупок та переглядів товарів – як результат, кожному користувачу пропонуються індивідуально підібрані рекомендації, що значно полегшує вибір товарів та підвищує зручність користування інтернет-магазином;
- контент головної сторінки адаптується під уподобання користувачів, відображаючи актуальні акції, товари, які їх зацікавили раніше, та персоналізовані рекламні пропозиції;
- для кожного клієнта створюються налаштовані пошукові фільтри, які враховують попередній вибір товарів та категорій;
- упроваджено адаптивний дизайн, що забезпечує зручність навігації на будь-яких пристроях, додатково покращено функціонал для роботи з кошиком, переглядом замовлень і зміною персональних даних;
- система дозволяє адміністраторам інтернет-магазину отримувати детальну аналітику про поведінку користувачів та ефективність маркетингових стратегій завдяки інтеграції Google Analytics, що дозволяє швидко адаптуватись, враховуючи поведінку покупців.

Таким чином, результати проекту дозволяють створити унікальний користувацький досвід, покращити залученість клієнтів, підвищити рівень їхньої лояльності та оптимізувати бізнес-процеси компанії.

ВИСНОВКИ

У ході виконання роботи було розроблено інтернет-магазин, орієнтований на продаж техніки та комплектуючих, з акцентом на персоналізовані рекомендації та зручний користувацький інтерфейс. Реалізована система відповідає сучасним вимогам ринку електронної комерції.

При аналізі предметної області було проведено вивчення особливостей онлайн-торгівлі, сформовано функціональні та технічні вимоги до програмного забезпечення.

Під час моделювання програмного забезпечення, створено UML-діаграми прецедентів та діаграми послідовності, які відображають основні сценарії роботи системи.

У розділі проєктування бази даних – визначено структуру даних, зв'язки між таблицями та обґрунтовано вибір СУБД. Реалізована логічна модель бази даних відповідає вимогам масштабованості та продуктивності.

Для проєктування інтернет-магазину – використано платформу OpenCart, інтегровано модуль TensorFlow для персоналізації товарних рекомендацій, налаштовано Google Analytics для моніторингу ефективності роботи сайту.

Використання сучасних технологій, таких як PHP, MySQL, TensorFlow та OpenCart, дозволяє реалізувати стабільну, функціональну та зручну систему для користувачів і адміністраторів. Інтернет-магазин забезпечує високий рівень зручності, безпеки даних і можливості для подальшого розвитку.

Розроблений проєкт підтвердив ефективність застосування принципів персоналізації та адаптивного дизайну для підвищення продажів і залучення клієнтів. Отримані результати свідчать про доцільність впровадження сучасних ІТ-рішень у сферу електронної комерції для підвищення конкурентоспроможності бізнесу.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

- 1) Електронна комерція. URL: <https://ideadigital.agency/blog/konversiya-shho-tse-take-i-yak-yiyi-porahuvati/> (Дата звернення: 01.10.2024)
- 2) Рекламні пропозиції. URL: <https://www.ranktracker.com/uk/blog/6-ways-to-leverage-personalization-to-enhance-user-experience/> (Дата звернення: 01.10.2024)
- 3) Інтернет-магазин техніки ТТТ.ua. URL: <https://www.ttt.ua/ua> (Дата звернення: 01.10.2024)
- 4) Інтернет-магазин техніки МТА.ua. URL: <https://mta.ua> (Дата звернення: 01.10.2024)
- 5) CMS. Management, finance, economics: modern problems and ways of their solutions: collective monograph / Zhydovska N., – etc. – International Science Group. – Boston: Primedia eLaunch, 2021. – 615 с. (Дата звернення: 01.10.2024)
- 6) OpenCart. URL: <https://interkassa.com/blog/cms-opencart-plyusi-ta-minusi-oglyad-funkcionalu> (Дата звернення: 01.10.2024)
- 7) HTML. URL: https://css.in.ua/article/shcho-take-css_3 (Дата звернення: 01.10.2024)
- 8) CSS. URL: https://css.in.ua/article/shcho-take-html_10 (Дата звернення: 01.10.2024)
- 9) PHP. URL: <https://dou.ua/lenta/articles/how-to-learn-php/> (Дата звернення: 01.10.2024)
- 10) JavaScript. Теорія та методика навчання математики, фізики, інформатики: Збірник наукових праць. Випуск V: В 3-х томах. – Кривий Ріг: Видавничий відділ НМетАУ, 2005. – Т. 2: Теорія та методика навчання фізики. – 401 с. (Дата звернення: 01.10.2024)
- 11) Клієнт-серверна архітектура. URL: <https://training.qatestlab.com/blog/technical-articles/client-server-architecture/> (Дата звернення: 01.10.2024)
- 12) СУБД. URL: <https://foxminded.ua/systema-upravlinnia-bazamy-danykh/> (Дата звернення: 01.10.2024)

13) SQLite. URL: <https://freehost.com.ua/ukr/faq/wiki/chto-takoe-sqlite/> (Дата звернення: 01.10.2024)

14) MySQL. URL: <https://promoter.net.ua/articles/baza-danix-mysql.html> (Дата звернення: 01.10.2024)

15) Wireframe. URL: <https://goldwebsolutions.com/uk/blog/shho-take-vajrfrej-m-wireframe-ta-dlya-chogo-vin-potriben-u-protse-si-rozrobki-sajtu-chi-dodatku/> (Дата звернення: 01.10.2024)

16) OpenServer. URL: <https://ospanel.io/> (Дата звернення: 01.11.2024)

17) GoogleAnalytics. URL: <https://marketingplatform.google.com/about/analytics/> (Дата звернення: 01.11.2024)

18) Киричек Г.Г., Коновалова Є.А. Інтернет-магазин зі стратегією персоналізованої реклами. XII Міжнародна науково-практична конференція "Сучасні проблеми і досягнення в галузі радіотехніки, телекомунікацій та інформаційних технологій", 10-12 груд. 2024 : тези доп. – Запоріжжя. (Дата звернення 01.12.2024)