

**Міністерство освіти і науки України
Луцький національний технічний університет
Факультет комп'ютерних та інформаційних технологій
Кафедра інженерії програмного забезпечення**

**КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «БАКАЛАВР»**

**РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ КАНЦТОВАРІВ «МОЛЬБЕРТ» З
ВИКОРИСТАННЯМ LARAVEL ТА MYSQL**

**DEVELOPMENT OF THE "MOLBERT" ONLINE STATIONERY STORE
USING LARAVEL AND MYSQL**

спеціальність 121 «Інженерія програмного забезпечення»
освітня програма «Інженерія програмного забезпечення»

Виконав: здобувач вищої освіти
групи ІПЗс-21
Левчук Віталій Юрійович

Керівник:
Ящук Андрій Анатолійович

Кваліфікаційну роботу
допущено до захисту

«10» 06 2025р.

Гарант освітньої програми:

к.т.н., доцент

Ліщина Наталія Миколаївна



Луцьк – 2025 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних та інформаційних технологій

Кафедра інженерії програмного забезпечення

Ступінь вищої освіти бакалавр

Галузь знань: 12 «Інформаційні технології»

Спеціальність: 121 «Інженерія програмного забезпечення»

Освітня програма: «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри



«20» 12 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Левчуку Віталію Юрійовичу

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи «Розробка інтернет-магазину канцтоварів «Мольберт» з використанням Laravel та MySQL»

Керівник роботи: доцент Ящук А.А.

затверджені наказом закладу вищої освіти від «19» грудня 2024 р. № 474/01-02

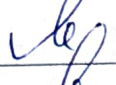
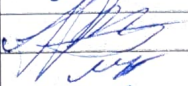
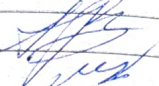
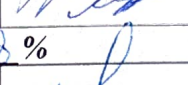
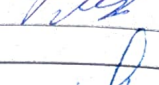
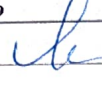
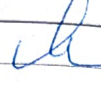
2. Строк подання здобувачем вищої освіти кваліфікаційної роботи бакалавра «10» червня 2025 р.

3. Вихідні дані до роботи: PHP 8.2, Laravel 8, MySQL 8.0, Blade, HTML5, CSS3, JavaScript, Swiper.js, Composer, Laravel Mix, Visual Studio Code, OpenServer, PHPUnit, методичні вказівки до виконання кваліфікаційної роботи бакалавра

4. Зміст розрахунково-пояснювальної записки (перелік питань, що потрібно розробити): аналіз предметної області, аналіз аналогічних програмних рішень, формування вимог до системи, розробка архітектури веб-додатку, реалізація функціональних модулів, тестування, підготовка документації, заходи безпеки.

5. Перелік графічного матеріалу: 24 рисунка

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
Аналіз предметної області	Яцук А.А.		
Специфікація вимог до розробленої системи	Яцук А.А.		
Розробка об'єкта проектування	Яцук А.А.		
Нормоконтроль	Вознюк А. В.		
Гарант ОП	Ліщина Н. М.		
Показник запозичень тексту		2,18 %	
Академічна доброчесність	Яцук А.А.		

7. Дата видачі завдання «20» грудня 2024 р.

№	Назва етапів кваліфікаційної роботи бакалавра	Строк виконання етапів роботи	Примітка
1	Огляд літературних джерел по темі кваліфікаційної роботи бакалавра	до 04.02.2025 р.	виконано
2	Аналіз проблеми розробки та впровадження об'єкту проектування	до 01.03.2025 р.	виконано
3	Обґрунтування вибору шляхів, технологій і засобів вирішення поставленого завдання	до 15.03.2025 р.	виконано
4	Розробка функціонально-структурної схеми роботи об'єкта проектування та проектування бази даних	до 29.03.2025 р.	виконано
5	Практична реалізація об'єкта проектування та розробка бази даних	до 26.04.2025 р.	виконано
6	Тестування та налагодження об'єкта проектування	до 03.05.2025 р.	виконано
7	Здача чистового варіанту кваліфікаційної роботи бакалавра на кафедрі	до 10.06.2025 р.	виконано

Здобувач вищої освіти



Віталій ЛЕВЧУК

Керівник кваліфікаційної роботи



Андрій ЯЦУК

АНОТАЦІЯ

Левчук В. Ю. Розробка інтернет-магазину канцтоварів «Мольберт» з використанням Laravel та MySQL. Рукопис. Кваліфікаційна робота бакалавра ОП «Інженерія програмного забезпечення» спеціальності 121 «Інженерія програмного забезпечення». Луцький національний технічний університет. Луцьк, 2025.

Кваліфікаційна робота бакалавра складається зі вступу, трьох розділів, висновків, списку використаних джерел.

У першому розділі проаналізовано предметну область електронної комерції, розглянуто функціональні особливості існуючих інтернет-магазинів та сформульовано завдання на кваліфікаційну роботу.

У другому розділі описано вимоги до програмного забезпечення, структуру бази даних, архітектуру системи, обґрунтовано вибір інструментів розробки та розкрито логіку взаємодії між компонентами.

У третьому розділі наведено приклади реалізації функціоналу вебдодатку, здійснено тестування, описано інструкцію з інсталяції, механізми захисту інформаційної системи, а також розроблено адміністративну панель для керування контентом.

У висновках узагальнено результати розробки, підтверджено досягнення поставленої мети та окреслено можливості подальшого розвитку системи.

Ключові слова: інтернет-магазин, Laravel, MySQL, Blade, PHP, електронна комерція, MVC, вебдодаток.

ABSTRACT

Levchuk V. Development of the "Molbert" online stationery store using Laravel and MySQL. Manuscript. Qualification work of the bachelor's program "Software Engineering" in the specialty "Software Engineering". Lutsk National Technical University. Lutsk, 2025.

The bachelor's qualification thesis consists of an introduction, three chapters, conclusions, a list of references.

The first chapter presents an analysis of the e-commerce domain, examines the functional features of existing online stores, and formulates the objectives.

The second chapter describes the software requirements, database structure, system architecture, and justifies the choice of development tools.

The third chapter presents the implementation of the main functionality, testing results, installation guide, security mechanisms, and the administrative panel for content management.

The conclusions summarize the results of the work, confirm the achievement of the goal, and outline prospects for further development of the system.

Keywords: online store, Laravel, MySQL, Blade, PHP, e-commerce, MVC, web application.

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАВДАНЬ НА КВАЛІФІКАЦІЙНУ РОБОТУ	9
1.1 Аналіз сучасного стану проблеми	9
1.2 Постановка завдання на кваліфікаційну роботу бакалавра	19
Висновок до розділу 1	21
РОЗДІЛ 2 СПЕЦИФІКАЦІЯ ВИМОГ ДО РОЗРОБЛЮВАНОЇ СИСТЕМИ.....	23
2.1 Аналіз, визначення вимог до розроблюваного програмного забезпечення та проєктування програмного забезпечення.....	23
2.2 Вибір засобів, методів і алгоритмів вирішення поставленого завдання.....	26
Висновок до розділу 2	33
РОЗДІЛ 3 РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ КАНЦТОВАРІВ «МОЛЬБЕРТ»	34
3.1 Практична реалізація об'єкта проєктування.....	34
3.2 Тестування та налагодження інформаційно-комп'ютерної системи ..	39
3.3 Розробка бази даних та взаємодія з нею засобами Laravel	41
3.4 Інструкція користувача з розгортання вебдодатку	44
3.5 Захист інформаційно-комп'ютерної системи.....	46
Висновок до розділу 3	47
ВИСНОВКИ.....	49
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	51

ВСТУП

Актуальність теми. У сучасних умовах стрімкого розвитку цифрових технологій електронна комерція стає невід’ємною частиною економічного середовища. Все більше компаній переходять на онлайн-формат торгівлі, що дозволяє оптимізувати витрати, охопити ширшу аудиторію та забезпечити зручність обслуговування. Актуальність теми полягає у створенні сучасного інтернет-магазину, який відповідатиме вимогам користувачів щодо функціональності, продуктивності, адаптивності та інформаційної безпеки.

Більшість наявних рішень мають обмеження щодо масштабованості або побудовані на застарілих технічних підходах, що створює потребу у реалізації нового рішення з гнучкою архітектурою та сучасним інтерфейсом.

Метою кваліфікаційної роботи є розробка вебдодатку – інтернет-магазину канцтоварів «Мольберт» – з використанням фреймворку Laravel та СУБД MySQL.

Об’єктом кваліфікаційної роботи є процес розробки веборієнтованих програмних систем для електронної комерції.

Предметом кваліфікаційної роботи виступає методика проєктування та реалізації інтернет-магазину з використанням архітектурного шаблону MVC, технологій Laravel, Blade, MySQL, HTML, CSS та JavaScript.

Завданням кваліфікаційної роботи є поетапна реалізація вебдодатку відповідно до поставленої мети, що включає виконання таких дій:

- провести аналіз предметної області;
- дослідити аналогічні онлайн-рішення та проаналізувати їх функціональні особливості;
- сформулювати вимоги до інтерфейсу, безпеки, адаптивності та продуктивності системи;
- спроектувати архітектуру веб-додатку та структуру бази даних відповідно до вимог;

- реалізувати основні модулі: каталог товарів, кошик, замовлення, кабінет користувача, адмінпанель;
- провести функціональне, інтеграційне та UX/UI-тестування веб-додатку;
- підготувати документацію з інсталяції, структури даних та принципів масштабування;
- впровадити заходи безпеки: автентифікацію, шифрування, захист від SQL-ін'єкцій, XSS, CSRF.

Практична значущість роботи полягає у створенні функціонального вебзастосунку, який може використовуватись як реальне рішення для організації онлайн-продажу товарів, або як основа для подальшого розвитку комерційних вебсистем. Робота спирається на сучасні методики веброзробки, включає принципи UX/UI-дизайну та демонструє можливість ефективної реалізації e-commerce функціоналу на базі Laravel.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАВДАНЬ НА КВАЛІФІКАЦІЙНУ РОБОТУ

1.1 Аналіз сучасного стану проблеми

Упродовж останніх років бурхливе зростання інформаційних і телекомунікаційних технологій кардинально змінило характер взаємодії між людьми та вплинуло на соціальні процеси в усьому світі. Засоби зв'язку, зокрема Інтернет, персональні комп'ютери й мобільний зв'язок, спричинили формування розгалуженої цифрової інфраструктури, що об'єднує окремих користувачів, компанії та навчальні заклади в єдиний комунікаційний простір. Таке інтенсивне впровадження технологій у повсякденне життя створило умови для активної участі кожного індивіда у глобальних мережевих процесах [1].

У сучасних економіках, особливо в технологічно розвинених країнах, спостерігається активне впровадження інтернет-торгівлі як одного з головних механізмів реалізації товарів і послуг. Щороку споживча активність у мережі зростає, і ця тенденція охопила також український ринок. Тут стрімко набирають обертів як великі бізнес-платформи, так і малі онлайн-магазини, а також інструменти для електронних розрахунків. Мережа Інтернет стала не лише середовищем для купівлі, а й ефективним каналом для просування продукції та комунікації з клієнтами [2].

Попит на цифрові сервіси зумовлений можливістю забезпечити індивідуальний підхід до кожного покупця, швидко та зручно оформити замовлення, зіставити ціни, ознайомитися з асортиментом і відгуками, не виходячи з дому.

Фактично, електронна комерція створює альтернативну інфраструктуру збуту, що вигідно відрізняється від традиційних форм торгівлі завдяки нижчій вартості товарів, ширшому вибору та гнучким умовам. Водночас вона не витісняє класичний ринок, а лише розширює можливості для споживача.

Однак у цифрових продажах є і свої слабкі сторони. Покупець не має змоги особисто оцінити фізичні характеристики товару – такі як текстура, колір, матеріал чи аромат. Це створює певні труднощі, особливо при замовленні парфумерії або косметичних засобів, де візуальна оцінка чи нюх є критично важливими.

Проте загалом, інтернет-торгівля несе позитивний вплив на сферу торгівлі. Відсутність реального магазину суттєво знижує собівартість товару через значно меншу орендну плату за рахунок відмови від торгового приміщення, меншої кількості працівників, більш лояльну податкову політику, що в свою чергу призводить до суттєвого зниження ціни для покупця та більшого прибутку для підприємця. Перевагою також є простота масштабування бізнесу.

Інтернет-торгівля також вплинула і на розвиток веб-розробки. Разом із тим, як у індустрію інтернет-торгівлі прийшли великі компанії, а разом з ними і великі гроші, стрімко почали розвиватися її методи та засоби. Причина цьому проста – пришвидшення темпів розробки та спрощення самого процесу призводить до більшого прибутку на одиницю часу. Вигідно це є і для замовника – менший час очікування та нижча ціна.

Для того, щоб спростити розробку було придумано багато методів та засобів, зокрема CMS (Система керування вмістом). Система керування вмістом (або CMS – від англ. Content Management System) – це програмна платформа, призначена для створення, редагування та адміністрування сайтів [3]. У професійному середовищі її часто називають «двигуном» або «серцем» вебресурсу. На основі таких систем функціонує значна частина сайтів у глобальній мережі. Серед найвідоміших рішень варто згадати WordPress, Joomla! та OpenCart.

Серед основних переваг CMS-платформ варто виокремити такі:

- можливість створити повноцінний вебсайт без глибоких знань програмування;
- швидкість розгортання проєкту;

- економія коштів у порівнянні з індивідуальною розробкою;
- зручний інтерфейс для керування контентом.

Однак, попри значну кількість переваг, ці системи мають і свої недоліки:

- потребують регулярного оновлення та перевірки сумісності з розширеннями;
- можуть втрачати швидкодію при надмірному використанні плагінів або модулів;
- обмежений набір функцій, що не завжди дозволяє реалізувати нестандартні рішення;
- не завжди підходять для складних або спеціалізованих завдань.

Фактично CMS – це універсальний інструмент для побудови вебсайтів, орієнтований на користувачів без технічної підготовки. Завдяки відкритому вихідному коду більшість таких систем активно вдосконалюються спільнотами розробників по всьому світу. Це сприяє зростанню їх безпеки, функціональності та зручності. Популярні CMS також мають перевагу у вигляді великої кількості документації та підтримки. Натомість менш відомі рішення часто обмежені у функціоналі, мають менше навчальних матеріалів і рідше отримують критично важливі оновлення.

Є і інший спосіб створення сайтів без знань в програмуванні. Існує досить велика кількість платних та безкоштовних сервісів для створення сайтів – так званих конструкторів, до прикладу Wix, Squarespace, SITE123, IONOS MyWebsite Now та багато інших. Розглянемо плюси та мінуси такого підходу [4].

До плюсів можна віднести простоту розробки – із створенням сайту на подібному конструкторі зможе впоратись будь-хто, який вміє користуватися комп'ютером. Перевагою також є дешевизна – такий сайт можна створити абсолютно безкоштовно, або за невелику плату [5].

Однозначно можна сказати, що на даний момент ці технології є далекі від досконалості. До мінусів можна віднести те, що далеко не завжди можна реалізувати всі свої побажання, на відміну від традиційної розробки, де

можливості розробника обмежують лише його знання та навички. Очевидним мінусом також є низький рівень захисту таких магазинів – вони стають легкою здобиччю для хакерів. Цей мінус неважливий, якщо вам потрібен сайт-візитівка, проте є найсуттєвішим для магазину, адже ви як власник будете втрачати як гроші, так і клієнтів, які не захочуть користуватись вашим сервісом через страх втрати власних коштів або викрадення персональної інформації.

Тому на сьогоднішній день, та і в найближчому майбутньому, без програмістів у цій сфері не обійтись.

Перед розробкою власного проекту доцільно проаналізувати аналогічні продукти від інших компаній. Виявивши та врахувавши їх переваги та недоліки можна спростити планування та реалізацію власного сайту. Нижче наведені приклади схожих за концепцією інтернет-магазинів та їх аналіз.

Інтернет-магазин канцтоварів «7office» здійснює продаж та доставку канцтоварів по всій території України. Загалом, дизайн сайту є досить приємним : правильно підібрана гамма кольорів, інтуїтивно зрозумілий інтерфейс, хоча і з недоліками, про які буде сказано нижче. Для того, щоб проаналізувати роботу цього сайту, я відображу всі етапи покупки товару покупцем і розкажу про плюси і мінуси магазину.

На головній сторінці (рис. 1.1) нас зустрічає випадаючий список із категоріями товарів, стрічка пошуку за назвою, кнопки входу та реєстрації та кошик. Вище розташований логотип компанії, навігація по сайту та контактна інформація. Також є слайдер із інформацією про компанію та послуги, які вона надає. До мінусів на даному етапі можна віднести надто дрібний і тонкий шрифт кнопок навігації та відсутність будь-якої стилізації. З плюсів можна виділити вдало підібрану гамму, великий розмір слайдеру, який дозволяє розмістити великі та інформативні слайди, наявність контактної інформації, що достатньо зручно.

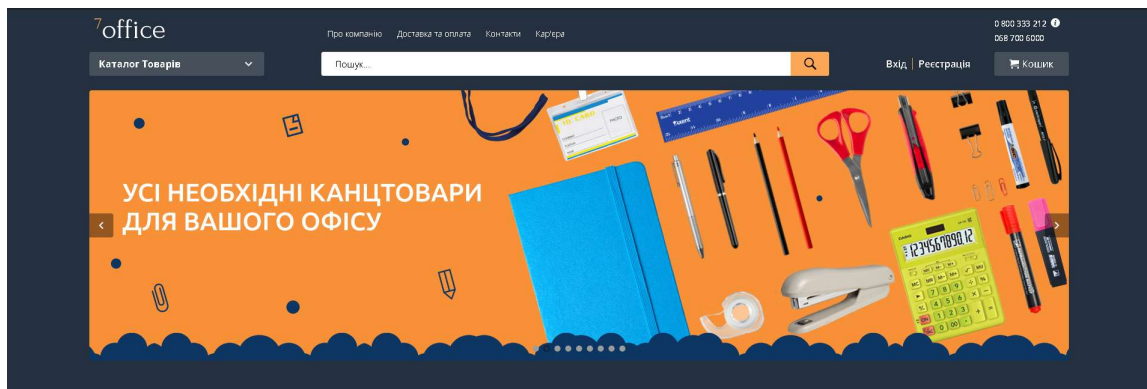


Рисунок 1.1 – Інтернет-магазин «7office» [6]

Нижче розташовані категорії товарів та їх фірми-виробники (рис. 1.2). При натисканні покупця перекидає на сторінку із товарами у цій категорії. Мінусом є нелогічне розташування кнопки «Вгору» – зазвичай, для зручності, вона завжди відображається з правого нижнього краю і зникає, коли користувач самотійно прогорне вгору. В даному випадку це просто статично розташована кнопка внизу перед так званим «підвалом» (footer) сайту. Також мінусом є надокучлива кнопка зворотнього відгуку, яку неможливо закрити і яка привертає багато уваги через свій яскравий колір. Вона якраз і розташована там, де мала б бути кнопка «Вгору».

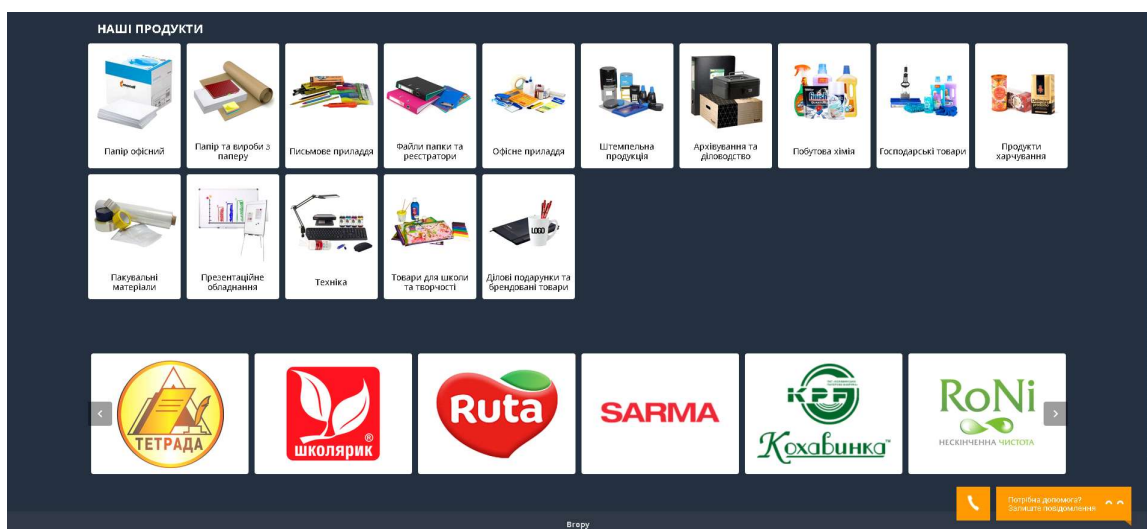


Рисунок 1.2 – Категорії товарів та їх фірми-виробники [6]

При переході на потрібну вам категорію вас перекидає на сторінку з

товарами (рис. 1.3). Досить гнучка система фільтрів допоможе покупцю знайти саме те, що йому потрібно. Відсортувати, до прикладу, письмове приладдя можна за торговою маркою, типом, матеріалом корпусу, кольором чорнила, товщиною лінії та іншими характеристиками, що притаманні саме письмовому приладдю.

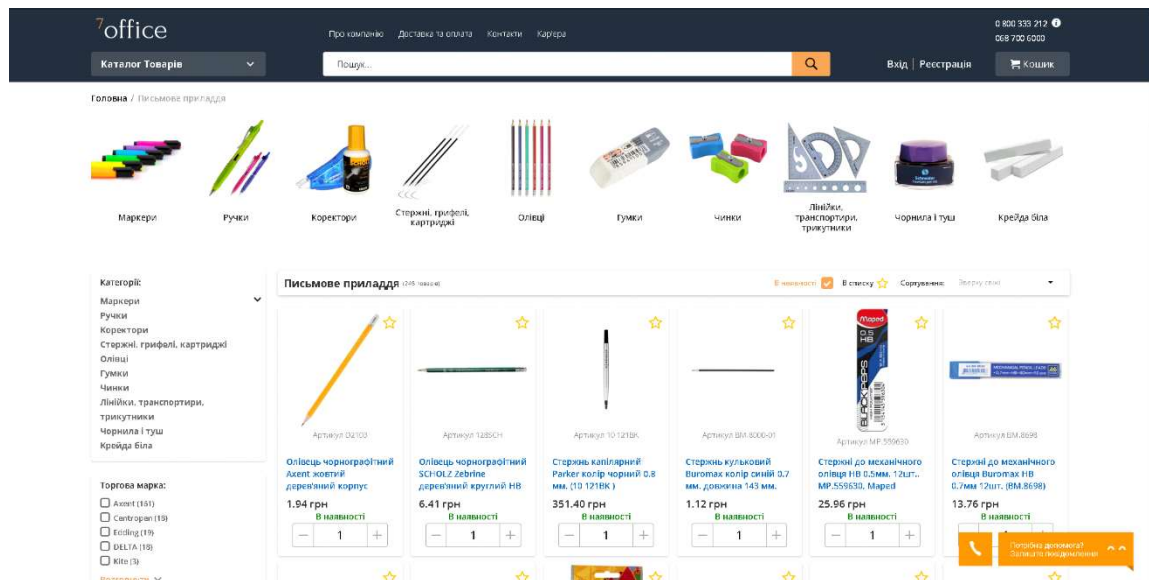


Рисунок 1.3 – Сторінка з товарами [6]

Проте, через велику кількість різних фільтрів та опцій у них, стає дуже незручним процес пошуку. Цілих 20 блоків із параметрами сортування, розташованих один під одним, займають місця майже в два рази більше, ніж самі товари (рис. 1.4). Це дезорієнтує користувача, змушуючи його бігати очима по великій кількості варіантів, чим збільшує час на налаштування потрібних фільтрів та викликає дискомфорт. Крім того, таке перетягування уваги на фільтри заважає сконцентруватись на перегляді самих товарів, адже потрібно прогорнути сторінку майже до кінця для того, щоб виставити потрібні параметри. Через це ускладнюється процес пошуку потрібного товару, а у підсумку – відлякує потенційного покупця, що несе за собою втрату прибутків.

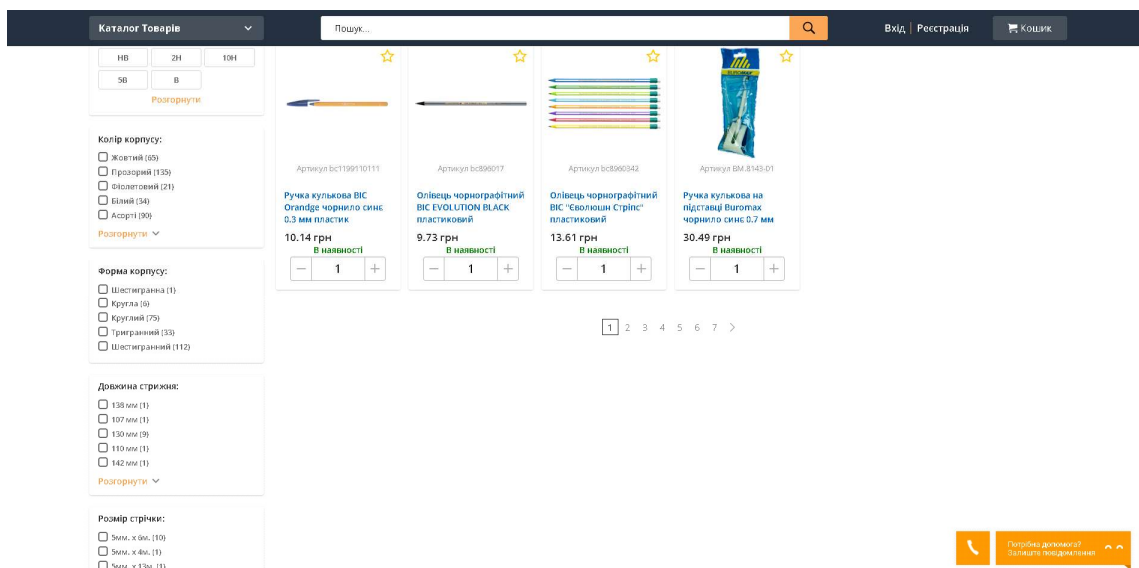


Рисунок 1.4 – Сторінка з товарами [6]

Після вибору потрібних товарів вони додаються до кошика, де можна оформити замовлення (рис. 1.5).

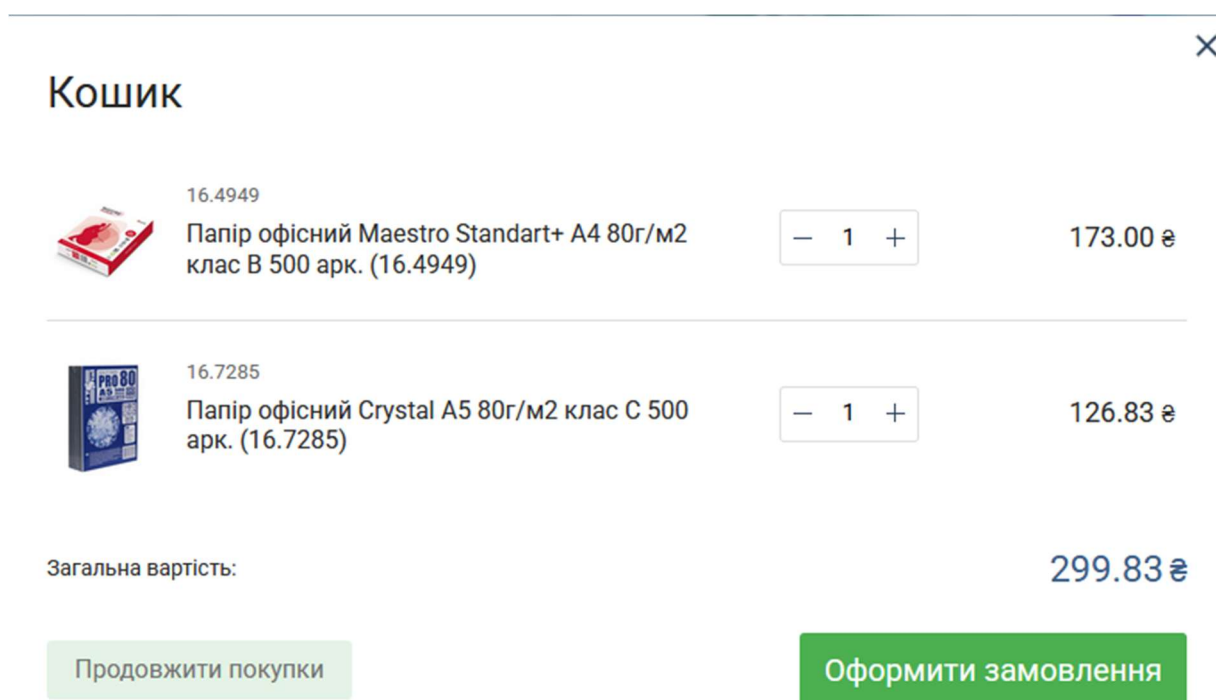


Рисунок 1.5 – Кошик [6]

Проте, перед тим як оформити замовлення, потрібно зареєструватись (рис. 1.6). Є можливість авторизуватись з допомогою

аккаунту Google або Facebook, що є досить зручним способом.

Головна / Вхід

РЕЕСТРАЦІЯ

Швидкий та зручний спосіб реєстрації в соцмережах:

або

Ім'я: Прізвище:

Телефон: Email:

Пароль:

Реєструючись, ви погоджуєтесь з [політикою конфідційності](#)

Рисунок 1.6 – Форма реєстрації [6]

Після реєстрації можна оформити замовлення (рис. 1.7). Форма приємна оку, інтуїтивно зрозуміла. Є також можливість вислати рахунок на вашу електронну пошту.

ОФОРМЛЕННЯ ЗАМОВЛЕННЯ

1 КОНТАКТНА ІНФОРМАЦІЯ

Ім'я*

Прізвище*

Назва компанії*

Код ЄДРПОУ*

Код ПІН*

Номер телефону*

Email*

7milkamouse7@gmail.com

☒ Платіть ПДВ

2 СПОСІБ ОПЛАТИ

☐ Оплата LiqPay

☐ Оплата готівкою або на карту

3 СПОСІБ ДОСТАВКИ

☒ Вести вручну

Місто

Вулиця

Будинок

Офіс

7milk mouse

КОНТАКТНА ІНФОРМАЦІЯ
Юридична особа

СПОСІБ ОПЛАТИ
Безготівкова оплата

СПОСІБ ДОСТАВКИ
Кур'єр 7Office

2 Товари на суму **397.992 €**

ДО СПЛАТИ: **397.992 €**

☒ Оформляючи замовлення, я приймаю умови [угоди користувача](#)

☐ Отримати рахунок на E-mail

Рисунок 1.7 – Оформлення замовлення [6]

Загалом, підсумовуючи вище сказане, можна сказати, що «7office» є хорошим прикладом того, як повинен працювати інтернет-магазин. Якщо не брати до уваги несуттєві недоліки по дизайну і оформленню сторінок, в цілому сайт функціонує правильно і принцип його роботи можна взяти за основу при створенні свого власного. Проте, в його роботі спостерігаються низька

швидкодія та деякі баги із кнопками (інколи спрацьовують не з першого натискання).

Інтернет-магазин канцтоварів «ОфісЦентр» здійснює продаж та доставку канцтоварів по всій території України. Одразу при потраплянні на сайт (рис. 1.8) в очі кидаються його три основні проблеми: застарілий, неестетичний дизайн та надлишкове нагромадження інформації у вигляді тексту та картинок.

Дрібні шрифти, які важко читати навіть на великому екрані, дуже заважають здійснювати покупки, а величезна палітра кольорів, яка утворилась внаслідок великої кількості різнобарвних картинок не дає сконцентруватись.

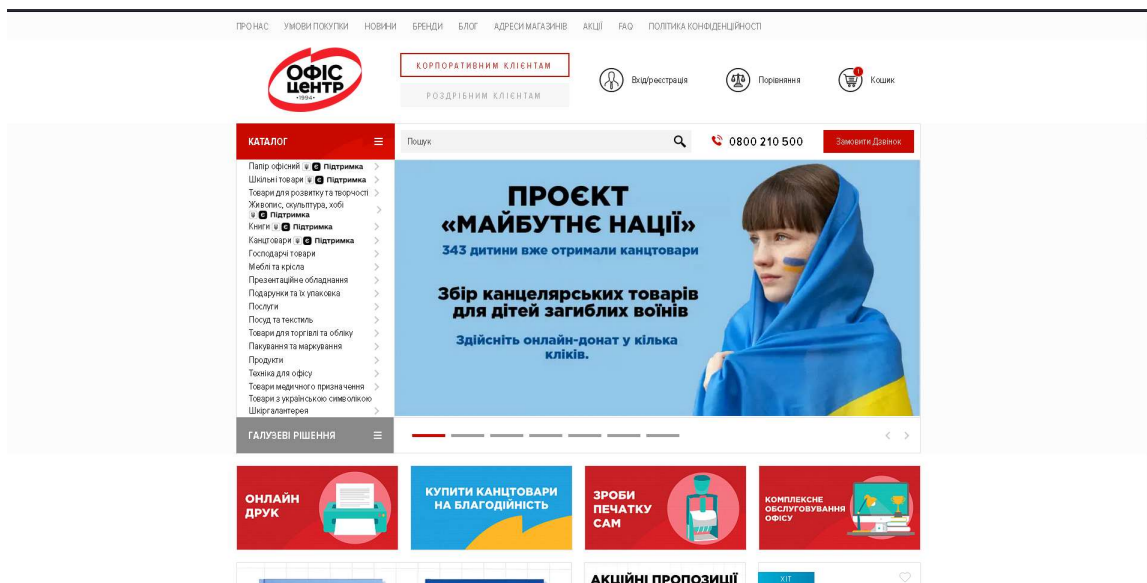


Рисунок 1.8 – Інтернет-магазин канцтоварів «ОфісЦентр» [7]

На сторінці з вибором товару (рис. 1.9) ситуація дещо краща. Окрім шрифтів у «шапці», решта виглядають нормально та легкі для читання. Реалізована також система фільтрів та сортування, виконана вона значно краще, ніж на попередньому сайті: опції фільтрів зроблені у вигляді випадаючого при натисканні списку, що суттєво економить місце та є набагато зручнішим у використанні. Є також система оцінок товарів від покупців, що також допомагає у виборі. Можна обрати кількість товарів, що будуть відображатись на одній сторінці, змінити відображення товарів із блоків у

список, додати товар у обрані та порівняти товари.

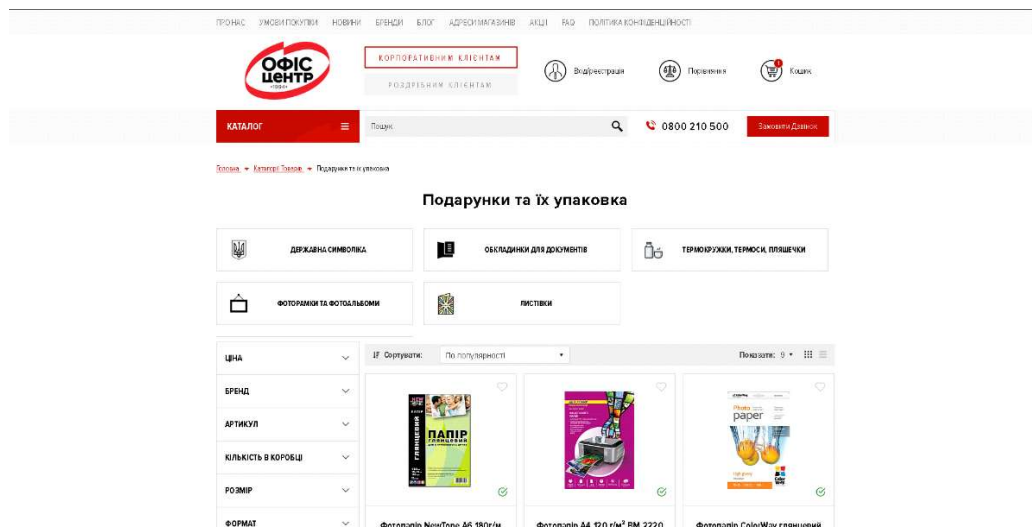


Рисунок 1.9 – Сторінка з вибором товару [7]

Проте є і дрібні недоліки. Наприклад, реалізація кнопок із рекомендованими категоріями товарів (рис. 1.10). При наведенні на одну з них, замість кнопки, яка знаходиться нижче з'являється список товарів із цієї категорії, але, через те, що між ними є відстань, досить складно навестись на потрібну категорію, адже при відведенні миші список швидко зникає.

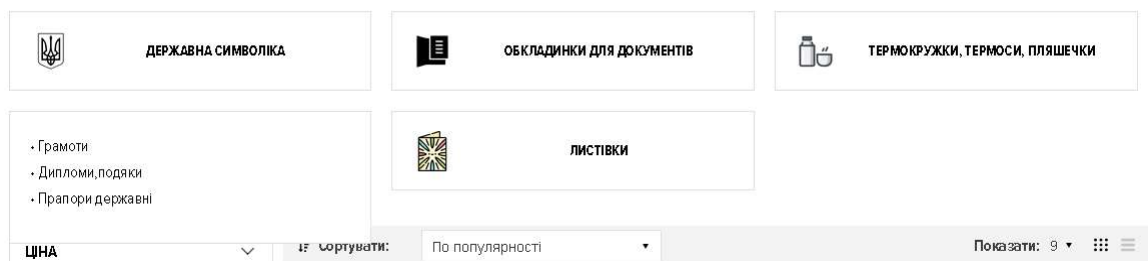


Рисунок 1.10 – Кнопки із рекомендованими категоріями товарів [7]

Після вибору потрібних вам товарів ви потрапляєте у кошик (рис. 1.11). Кошик є незручним та незрозумілим у користуванні. Перше, що кидається у очі – блок із ціною, вводом промокоду та навігаційними кнопками знаходиться зліва, тоді як зрозуміліше є розташування їх справа, як у переважної більшості інтернет-магазинів.

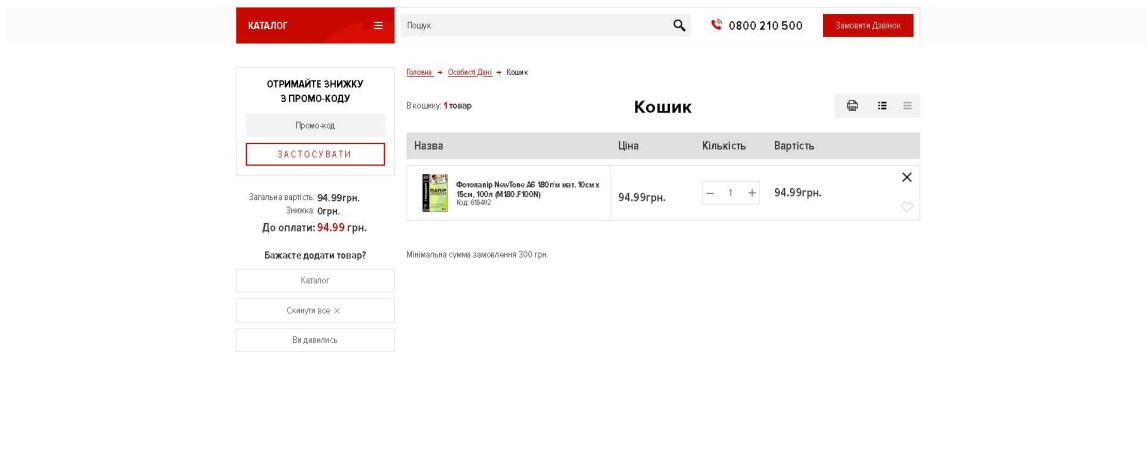


Рисунок 1.11 – Кошик [7]

Якщо сума замовлення менша, ніж 300 гривень, кнопка «Замовити» просто відсутня і з'являється лише тоді, коли сума достатня, що є досить незручним, адже клієнт може просто не зрозуміти, як оформити замовлення. Коли все ж таки ви набрали товарів більш як на 300 гривень, кнопки з'являються одразу дві (рис. 1.12), що також нелогічно.

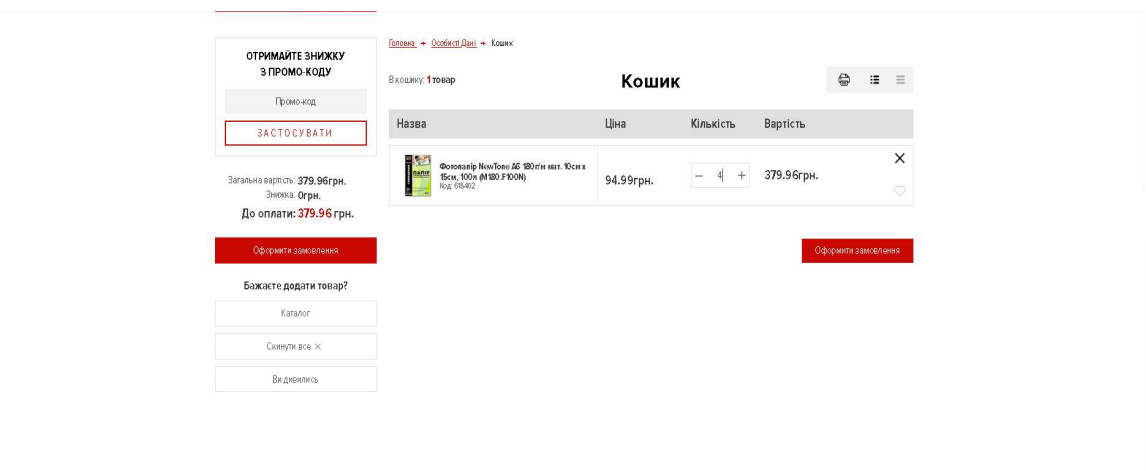


Рисунок 1.12 – Кошик із кнопками «Замовити» [7]

1.2 Постановка завдання на кваліфікаційну роботу бакалавра

Метою кваліфікаційної роботи є розробка сучасного веб-додатку – інтернет-магазину канцтоварів «Мольберт» – із використанням фреймворку

Laravel та СУБД MySQL. Розробка повинна забезпечити зручну взаємодію з інтерфейсом, ефективне управління товарами, безпечну обробку замовлень та відповідність актуальним вимогам користувачів і веб-розробки.

Для досягнення поставленої мети необхідно виконати наступні завдання:

- провести аналіз предметної області. Ознайомитися з особливостями ведення онлайн-торгівлі канцтоварами, сформулювати перелік основних функціональних потреб потенційних користувачів та адміністрації інтернет-магазину;

- дослідити існуючі програмні рішення. Проаналізувати популярні онлайн-магазини схожого спрямування, оцінити їх інтерфейс, функціональність, зручність використання та швидкодію. Визначити переваги, недоліки та можливості для вдосконалення;

- сформулювати вимоги до системи. Визначити технічні, функціональні та нефункціональні вимоги до веб-додатку, включаючи вимоги до інтерфейсу користувача, безпеки, адаптивності, масштабованості та підтримки;

- розробити архітектуру веб-додатку. Спроекувати логічну та фізичну архітектуру системи, включаючи структуру бази даних, компоненти веб-додатку, взаємодію між ними. Вибір архітектурних шаблонів та обґрунтування їх застосування у рамках Laravel;

- реалізувати ключові функціональні модулі. Створити окремі частини системи: головна сторінка, каталог товарів з фільтрацією, кошик, обробка замовлень, кабінет користувача та адміністративна панель для управління товарами і користувачами;

- забезпечити тестування веб-додатку. Виконати функціональне, інтеграційне та UX/UI-тестування. Перевірити систему на відповідність вимогам, її надійність, захищеність та стійкість до помилок;

- розробити супровідну документацію. Підготувати інструкції зі встановлення, налаштування та використання інтернет-магазину. Описати

структуру БД, логіку модулів, вимоги до серверного середовища та можливості масштабування системи в майбутньому;

– розробити та впровадити заходи безпеки. Проаналізувати потенційні загрози інформаційній системі, визначити критичні вектори атак та реалізувати механізми захисту від несанкціонованого доступу. Налаштувати автентифікацію користувачів, обмеження прав доступу, шифрування конфіденційних даних, а також захист від типових веб-вразливостей (SQL-ін'єкцій, XSS, CSRF). Забезпечити дотримання сучасних стандартів безпеки з урахуванням вимог GDPR та інших регламентів.

Висновок до розділу 1

У цьому розділі було проведено глибокий аналіз предметної області електронної комерції, зокрема ринку інтернет-торгівлі канцтоварами. Розглянуто динаміку розвитку онлайн-продажів, визначено їхню актуальність для сучасного ринку та обґрунтовано доцільність створення спеціалізованого веб-додатку.

На основі аналізу було виявлено основні тенденції розвитку інтернет-магазинів, зокрема популяризацію CMS-систем і онлайн-конструкторів, а також їхні переваги та обмеження в контексті створення функціонального та безпечного веб-сервісу. Особливу увагу приділено порівнянню готових рішень із повноцінною розробкою програмного забезпечення за допомогою фреймворку Laravel, що дозволяє реалізувати гнучку архітектуру й індивідуальний функціонал.

З метою визначення оптимальної моделі реалізації, було здійснено аналіз існуючих інтернет-магазинів схожого призначення. У процесі дослідження окремо розглянуто дизайн, структуру, функціональність, зручність користування та недоліки таких платформ, як «7office» та «ОфісЦентр». Проведений аналіз дозволив виділити ключові аспекти, які варто реалізувати у власному рішенні, а також уникнути типових помилок при проєктуванні.

У завершальній частині розділу було сформульовано постановку завдання на кваліфікаційну роботу, де окреслено основну мету проєкту – створення функціонального, безпечного та зручного у використанні веб-додатку інтернет-магазину канцтоварів. Сформовано перелік технічних і функціональних вимог, визначено етапи реалізації, а також критерії оцінки успішності реалізації поставленого завдання.

РОЗДІЛ 2

СПЕЦИФІКАЦІЯ ВИМОГ ДО РОЗРОБЛЮВАНОЇ СИСТЕМИ

2.1 Аналіз, визначення вимог до розроблюваного програмного забезпечення та проєктування програмного забезпечення

Розроблюване програмне забезпечення – це веб-застосунок для реалізації функціоналу інтернет-магазину канцтоварів. Система має забезпечити ефективну взаємодію між користувачем, адміністратором та системою обробки замовлень. Основними вимогами є зручність навігації, безпечність обробки даних, швидкодія, масштабованість та зрозумілий інтерфейс.

На основі аналізу потреб цільових користувачів та бізнес-вимог сформульовано такі функціональні вимоги:

- авторизація та реєстрація користувачів;
- каталог продукції з можливістю фільтрації;
- перегляд детальної інформації про товари;
- кошик для оформлення замовлень;
- обробка замовлень та статусів доставки;
- адміністративна панель для керування товарами та замовленнями.

Для побудови моделі взаємодії в системі було розроблено декілька діаграм, серед яких діаграма прецедентів, діаграма послідовності, діаграма активності та також була розроблена схема БД.

Діаграма прецедентів, яка відображає ролі користувачів (гостя, зареєстрованого клієнта, адміністратора) та їх взаємодії із системою, зображена на рисунку 2.1. Дана діаграма досить детально відображає принцип роботи інтернет-магазину від етапу замовлення до фактичного отримання товару покупцем. Описані всі варіанти взаємодії кожного актора з інформаційною системою.

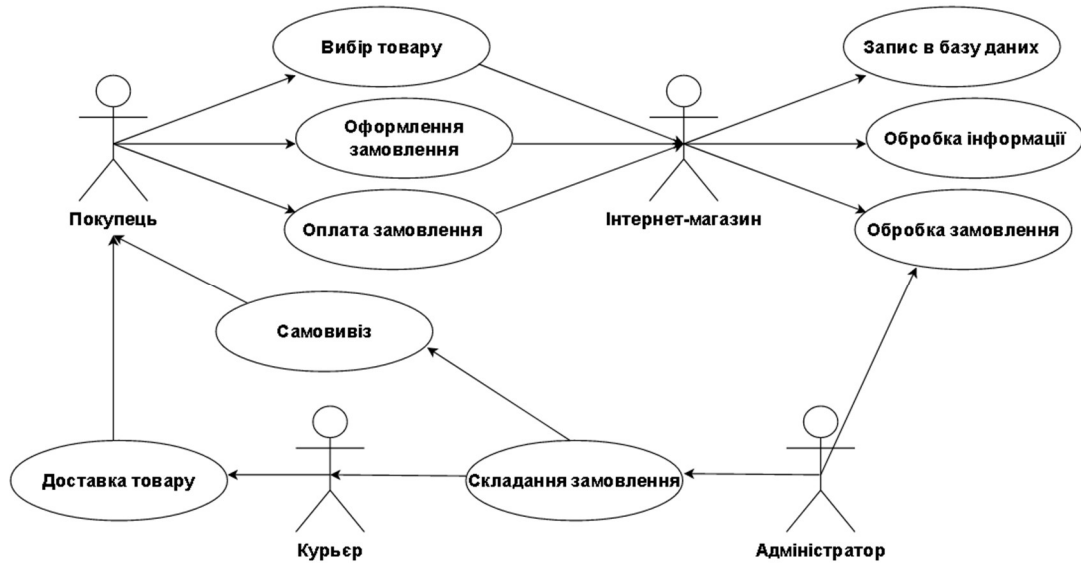


Рисунок 2.1 – Діаграма прецедентів

Діаграма послідовності, що моделює часову послідовність обміну повідомленнями між об'єктами, зображена на рисунку 2.2.

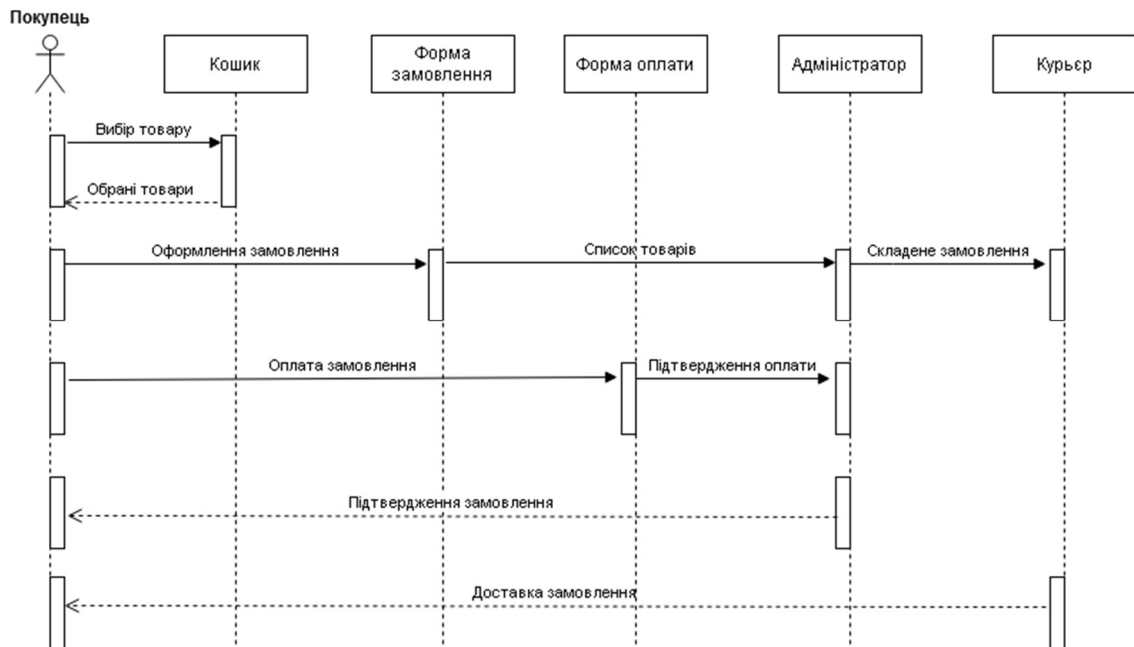


Рисунок 2.2 – Діаграма послідовності

Діаграму розгортання, яка демонструє розміщення компонентів системи в інфраструктурі, зображена на рисунку 2.3.

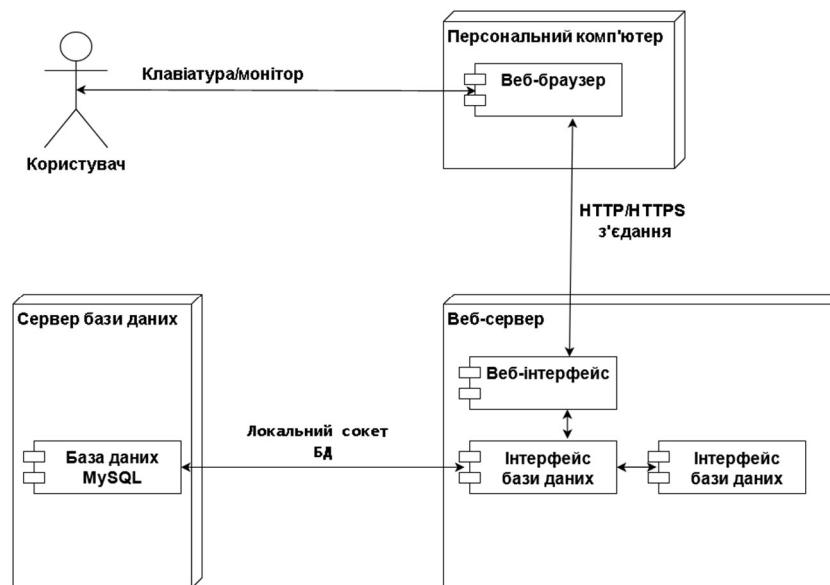


Рисунок 2.3 – Діаграма розгортання

Схему бази даних, яка відображає структуру таблиць, зв'язки між ними та типи даних, зображена на рисунку 2.4.

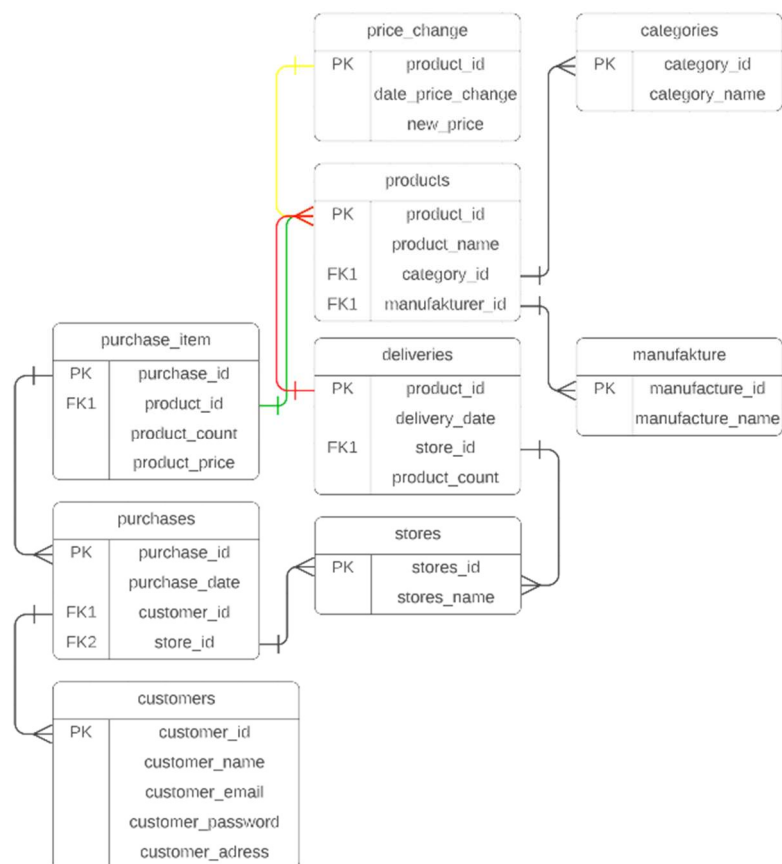


Рисунок 2.4 – Схема БД

У якості фреймворку для розробки обрано Laravel, що базується на архітектурі MVC (Model-View-Controller). Це дозволяє логічно поділити систему на модель даних, інтерфейс користувача та керуючу логіку.

Серед ключових технологій та інструментів, використаних у проєкті:

- Blade – шаблонізатор, що забезпечує динамічну генерацію сторінок;
- Eloquent ORM – засіб для взаємодії з базою даних, що реалізує шаблон ActiveRecord;
- Fluent – конструктор SQL-запитів;
- Artisan CLI – консольний інтерфейс для генерації коду;
- Laravel Sanctum – модуль автентифікації для токенованого доступу;
- PHPUnit – система модульного тестування;
- MySQL – реляційна СУБД, яка забезпечує зберігання даних та оптимізована для високонавантажених систем.

Застосування цієї технологічної бази дозволяє забезпечити гнучкість системи, високу продуктивність, розширюваність та захист від типових загроз (SQL-ін'єкцій, XSS).

2.2 Вибір засобів, методів і алгоритмів вирішення поставленого завдання

Laravel – це один із найпопулярніших фреймворків для PHP, який надає розробникам інструменти для створення сучасних, масштабованих і безпечних вебзастосунків. Завдяки модульності, відкритому коду, зручному синтаксису та розвиненій екосистемі, Laravel став стандартом де-факто для багатьох проєктів у сфері веброзробки.

З іншого боку, цей фреймворк надає розробникам гнучкі інструменти для побудови веб-систем, які відповідають високим стандартам якості. Його архітектура значно полегшує створення масштабованих та структурованих додатків, дозволяючи зосередитись на логіці бізнес-процесів, а не на технічних дрібницях. Завдяки інтуїтивно зрозумілому синтаксису Laravel дозволяє

писати чистий та легко підтримуваний код, що особливо важливо у командній розробці. Також варто відзначити велику екосистему, яка включає безліч офіційних та сторонніх пакетів, що значно розширюють функціональні можливості проєктів.

Його основне призначення – спрощення та прискорення створення вебрішень шляхом надання готових шаблонів архітектури, зокрема MVC (Model-View-Controller), вбудованих інструментів для аутентифікації, маршрутизації, шаблонізації, роботи з базами даних тощо. Однією з ключових переваг Laravel є чітке відокремлення логіки, подання та обробки даних, що значно покращує читабельність коду та підтримуваність застосунку [8].

З іншого боку, цей фреймворк надає розробникам гнучкі інструменти для побудови веб-систем, які відповідають високим стандартам якості. Його архітектура значно полегшує створення масштабованих та структурованих додатків, дозволяючи зосередитись на логіці бізнес-процесів, а не на технічних дрібницях. Завдяки інтуїтивно зрозумілому синтаксису Laravel дозволяє писати чистий та легко підтримуваний код, що особливо важливо у командній розробці. Також варто відзначити велику екосистему, яка включає безліч офіційних та сторонніх пакетів, що значно розширюють функціональні можливості проєктів.

Laravel пропонує такі потужні функції, як Artisan – консольний інтерфейс для генерації шаблонного коду та автоматизації завдань; Eloquent ORM – об'єктно-реляційний зв'язок із базою даних на основі патерна Active Record; Blade – швидкий та зручний шаблонізатор; засоби кешування, обробки сесій, валідації даних, логування, створення API та багато іншого [9].

З іншого боку, цей фреймворк надає розробникам гнучкі інструменти для побудови веб-систем, які відповідають високим стандартам якості. Його архітектура значно полегшує створення масштабованих та структурованих додатків, дозволяючи зосередитись на логіці бізнес-процесів, а не на технічних дрібницях. Завдяки інтуїтивно зрозумілому синтаксису Laravel дозволяє писати чистий та легко підтримуваний код, що особливо важливо у командній

розробці. Також варто відзначити велику екосистему, яка включає безліч офіційних та сторонніх пакетів, що значно розширюють функціональні можливості проєктів.

Фреймворк підтримує розширену систему маршрутів, яка дозволяє гнучко налаштовувати шляхи доступу до ресурсів та дій. За допомогою middleware можна керувати доступом, обмеженнями безпеки або перевіркою прав користувачів до виконання певних дій.

З іншого боку, цей фреймворк надає розробникам гнучкі інструменти для побудови веб-систем, які відповідають високим стандартам якості. Його архітектура значно полегшує створення масштабованих та структурованих додатків, дозволяючи зосередитись на логіці бізнес-процесів, а не на технічних дрібницях. Завдяки інтуїтивно зрозумілому синтаксису Laravel дозволяє писати чистий та легко підтримуваний код, що особливо важливо у командній розробці. Також варто відзначити велику екосистему, яка включає безліч офіційних та сторонніх пакетів, що значно розширюють функціональні можливості проєктів.

Laravel активно підтримує мультимовність, що дозволяє створювати вебресурси для кількох аудиторій одночасно. Через механізм локалізації можливо швидко додати переклади інтерфейсу, повідомлень та контенту.

З іншого боку, цей фреймворк надає розробникам гнучкі інструменти для побудови веб-систем, які відповідають високим стандартам якості. Його архітектура значно полегшує створення масштабованих та структурованих додатків, дозволяючи зосередитись на логіці бізнес-процесів, а не на технічних дрібницях. Завдяки інтуїтивно зрозумілому синтаксису Laravel дозволяє писати чистий та легко підтримуваний код, що особливо важливо у командній розробці. Також варто відзначити велику екосистему, яка включає безліч офіційних та сторонніх пакетів, що значно розширюють функціональні можливості проєктів.

Ще однією важливою складовою є механізм міграцій, який надає змогу контролювати версії структури бази даних. Завдяки цьому команди проєкту

можуть координовано вносити зміни до структури таблиць, не боячись втратити важливі дані.

З іншого боку, цей фреймворк надає розробникам гнучкі інструменти для побудови веб-систем, які відповідають високим стандартам якості. Його архітектура значно полегшує створення масштабованих та структурованих додатків, дозволяючи зосередитись на логіці бізнес-процесів, а не на технічних дрібницях. Завдяки інтуїтивно зрозумілому синтаксису Laravel дозволяє писати чистий та легко підтримуваний код, що особливо важливо у командній розробці. Також варто відзначити велику екосистему, яка включає безліч офіційних та сторонніх пакетів, що значно розширюють функціональні можливості проєктів.

В системі Laravel інтегровано систему модульного тестування за допомогою PHPUnit. Це дозволяє перевіряти кожен окремий компонент додатку на працездатність ще до розгортання його у продакшн.

З іншого боку, цей фреймворк надає розробникам гнучкі інструменти для побудови веб-систем, які відповідають високим стандартам якості. Його архітектура значно полегшує створення масштабованих та структурованих додатків, дозволяючи зосередитись на логіці бізнес-процесів, а не на технічних дрібницях. Завдяки інтуїтивно зрозумілому синтаксису Laravel дозволяє писати чистий та легко підтримуваний код, що особливо важливо у командній розробці. Також варто відзначити велику екосистему, яка включає безліч офіційних та сторонніх пакетів, що значно розширюють функціональні можливості проєктів.

Також Laravel підтримує RESTful API, має вбудовані можливості для автентифікації та авторизації, у тому числі через пакети Laravel Sanctum і Laravel Passport [10].

З іншого боку, цей фреймворк надає розробникам гнучкі інструменти для побудови веб-систем, які відповідають високим стандартам якості. Його архітектура значно полегшує створення масштабованих та структурованих додатків, дозволяючи зосередитись на логіці бізнес-процесів, а не на технічних

дрібницях. Завдяки інтуїтивно зрозумілому синтаксису Laravel дозволяє писати чистий та легко підтримуваний код, що особливо важливо у командній розробці. Також варто відзначити велику екосистему, яка включає безліч офіційних та сторонніх пакетів, що значно розширюють функціональні можливості проєктів.

Серед численних реляційних систем управління базами даних особливе місце посідає MySQL – потужна, надійна та популярна платформа, що здобула визнання серед розробників усього світу. Ця СУБД із відкритим кодом має багаторічну історію розвитку та підтримується широкою спільнотою, що забезпечує її актуальність, стабільність та безпеку [11].

Однією з визначальних переваг MySQL є її здатність безперешкодно інтегруватися з найпоширенішими мовами програмування, такими як PHP. Це робить її особливо ефективною для створення динамічних вебзастосунків з високою функціональністю. Завдяки застосуванню стандартизованої мови SQL, розробники мають змогу зручно керувати великими масивами структурованих даних, виконувати складні операції з агрегації, фільтрації, об'єднання інформації тощо [12].

Сучасна цифрова інфраструктура неможлива без ефективних систем управління базами даних. Від вибору СУБД залежить стабільність, продуктивність і масштабованість багатьох інформаційних сервісів. З-поміж усіх доступних рішень MySQL продовжує залишатись одним з найнадійніших і найпопулярніших інструментів завдяки постійним оновленням, широкій підтримці спільноти та гнучкості конфігурації.

MySQL реалізована за архітектурною моделлю клієнт-сервер, що забезпечує чіткий розподіл обов'язків: серверна частина відповідає за зберігання та обробку інформації, а клієнтська – за взаємодію з користувачем. Така структура дозволяє створювати як невеликі вебсайти, так і розгалужені корпоративні платформи з багатьма точками доступу [13].

Підтримка транзакційного механізму дозволяє MySQL гарантувати консистентність даних, навіть за умови паралельного доступу численних

користувачів. А реалізовані засоби реплікації та резервного копіювання забезпечують додатковий рівень відмовостійкості й збереження критично важливої інформації.

Сучасні тенденції веброзробки дедалі частіше передбачають використання мікросервісної архітектури, де кожен модуль системи виконує окрему функцію та взаємодіє з іншими через API. У такій парадигмі MySQL демонструє високу гнучкість, оскільки може бути легко масштабована горизонтально, розподіляючи навантаження між кількома інстанціями бази даних. Її сумісність із контейнерними технологіями, такими як Docker і Kubernetes, дозволяє ефективно впроваджувати MySQL у DevOps-процеси.

Не можна ігнорувати той факт, що MySQL стала каталізатором для розвитку суміжних технологій, таких як ORMs, системи кешування (Memcached, Redis), а також інструменти автоматичного бекапу і моніторингу. Розмаїття готових бібліотек і утиліт, що підтримують MySQL, значно скорочує час на розгортання нових проєктів.

Фреймворк Laravel, який широко використовується для створення вебзастосунків на PHP, органічно поєднується з MySQL. Їх тандем дозволяє реалізовувати архітектурно продумані системи з мінімальними витратами на написання SQL-запитів завдяки ORM Eloquent. Остання забезпечує об'єктно-орієнтований доступ до даних, що підвищує зрозумілість коду та його обслуговування [14].

MySQL підтримує великий набір сучасних можливостей: складні індекси, зовнішні ключі, тригери, події, збережені процедури – усе це дозволяє адаптувати базу даних до специфіки конкретного проєкту без втрати продуктивності. Гнучкість у налаштуваннях робить її однією з найзручніших СУБД у сфері веброзробки.

Безпека також залишається пріоритетом: реалізовано розширені засоби шифрування, аутентифікації, контроль доступу на рівні користувача, підтримка SSL-з'єднань. Це дозволяє відповідати сучасним вимогам кіберзахисту та нормативним стандартам.

MySQL демонструє відмінну сумісність із хмарними платформами: її можна легко розгорнути у середовищах Amazon Web Services, Google Cloud Platform або Microsoft Azure. Це забезпечує зручне масштабування, балансування навантаження та автоматичне резервування даних [15].

Однією з унікальних переваг MySQL є підтримка різних механізмів зберігання (storage engines), які можуть бути обрані під специфіку конкретного завдання. Наприклад, InnoDB є основним і найбільш безпечним типом для транзакційних додатків із підтримкою зовнішніх ключів, тоді як MyISAM може використовуватися для проєктів, що вимагають максимальної швидкості при читанні. Такий підхід дозволяє досягати балансу між продуктивністю, стабільністю та надійністю в залежності від конкретного випадку.

MySQL має розвинений інструментарій для створення резервних копій та міграцій між серверами. Зокрема, утиліти mysqldump, mysqlpump або інструменти на кшталт MySQL Enterprise Backup дозволяють створювати як повні, так і інкрементальні резервні копії. Це важливо для забезпечення безперервності бізнес-процесів та швидкого відновлення у випадку втрати даних.

Для діагностики проблем з продуктивністю MySQL пропонує команду EXPLAIN, яка дозволяє оцінити, як саме оптимізатор запитів обробляє SQL-запит. Крім того, підтримка режиму профілювання (SHOW PROFILE) та журналів повільних запитів (slow query log) роблять систему зручною у налаштуванні й обслуговуванні складних додатків, де критично важливе швидке реагування на зміну навантажень.

Завдяки такому багатому функціоналу MySQL перетворилася з простої реляційної бази даних на комплексний інструмент, здатний задовольнити як потреби малого бізнесу, так і запити великих корпорацій. Вона активно використовується в проєктах з відкритим кодом, освітніх платформах та комерційних продуктах.

Популярність MySQL обумовлена не лише її технічними характеристиками, а й активною підтримкою спільноти, розвинутою

документацією та регулярними оновленнями, що дозволяє їй залишатися одним з лідерів серед СУБД у глобальному масштабі.

Висновок до розділу 2

У даному розділі було проведено аналіз функціональних і технічних вимог до розроблюваного веб-додатку інтернет-магазину канцтоварів. Здійснено вибір засобів, методів і технологій, що найкраще відповідають специфіці поставленого завдання.

На основі аналізу було обрано фреймворк Laravel, який забезпечує структуровану архітектуру на основі MVC, підтримує RESTful API, має вбудовані засоби автентифікації, валідації, роботи з базами даних, шаблонізатором і кешуванням. У якості системи управління базами даних обрано MySQL, яка є надійною, продуктивною і має розвинену підтримку транзакцій та індексації.

Було визначено основні алгоритми, які реалізуються в системі: автентифікація користувачів, формування замовлень, фільтрація товарів, робота з сесіями та кешуванням. Також сформовано методичну основу розробки із застосуванням об'єктно-орієнтованого програмування, принципів безпеки та тестування.

Отже, у результаті цього етапу було закладено фундамент програмного продукту: обрано технологічну платформу, інструменти реалізації та визначено архітектурні рішення, які забезпечують подальше ефективне впровадження інформаційної системи.

РОЗДІЛ 3

РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ КАНЦТОВАРІВ «МОЛЬБЕРТ»

3.1 Практична реалізація об'єкта проєктування

У процесі створення інтернет-магазину канцтоварів «Мольберт» було реалізовано гнучку архітектуру, побудовану на базі фреймворку Laravel, що є одним із найпопулярніших рішень у сучасному веб-програмуванні. Основну увагу при проєктуванні було приділено не лише технічним аспектам, а й зручності користувацької взаємодії, адаптивності до різних типів пристроїв, можливості подальшого масштабування системи та забезпеченню її стійкості й безпеки.

Розробка здійснювалася у середовищі PHP версії 8.2, з дотриманням принципів шаблону MVC (Model-View-Controller). Такий підхід дозволив розділити логіку роботи застосунку, його представлення для користувача та взаємодію з базою даних, що, в свою чергу, позитивно вплинуло на підтримуваність коду та полегшило подальші доопрацювання.

Особливу роль у процесі розгортання фронтенду відіграв інструмент Laravel Mix, який надає зручний інтерфейс для компіляції ресурсів проєкту та працює на базі Webpack. У конфігураційному файлі цього інструменту було вказано шляхи до відповідних стилів та JavaScript-скриптів, а також задано параметри оптимізації для підготовки застосунку до роботи в продуктивному середовищі.

На головній сторінці реалізовано блок з новинами магазину, в якому працівники з допомогою адмін-панелі можуть розміщати новини, оновлення або вигідні пропозиції. Його вигляд зображено на рисунку 3.1. Подібний принцип взаємодії адміністратора з сайтом реалізовано для усіх блоків з контентом. Крім того, є можливість додавати нові сторінки повністю засобами адмін-панелі.

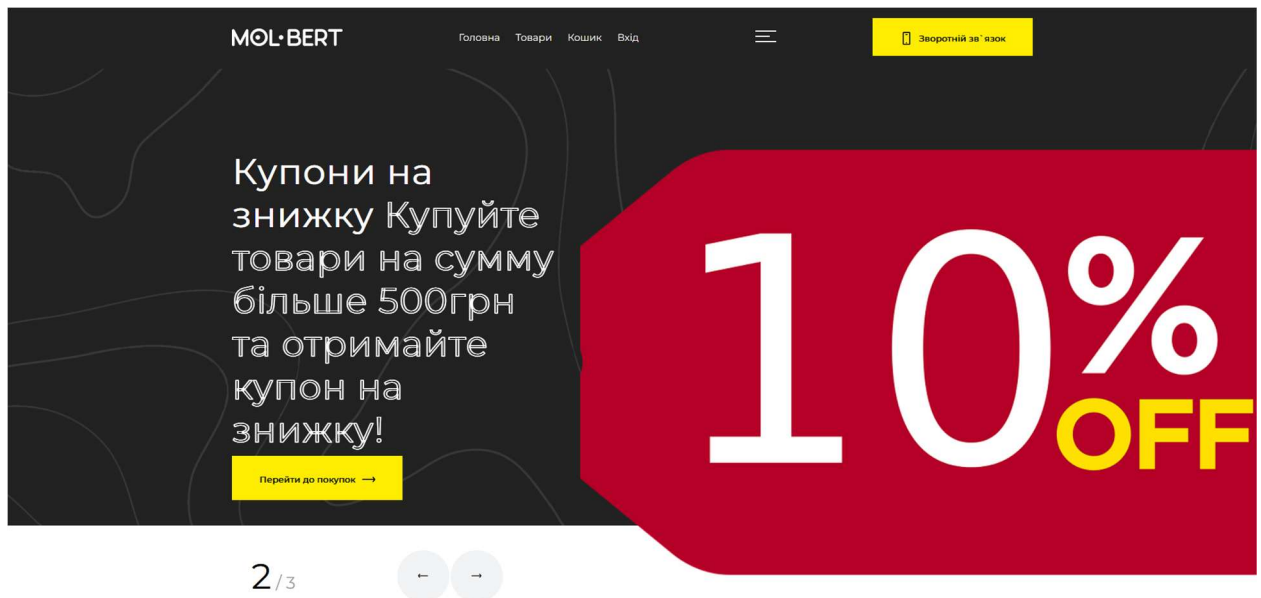


Рисунок 3.1 – Вигляд блоку новин

Програмна реалізація цього блоку представляє собою слайдер, реалізований з допомогою бібліотеки SwiperJS, слайди у якому генеруються динамічно отримуючи дані з бази. Нижче, у лістингу 3.1, наведений код слайдера новин.

Лістинг 3.1 – Розмітка блоку «Новини»

```
@if (!empty($sliders))
    @foreach ($sliders as $key => $slider)
        <div class="hero-bg-2 lazy" data-
bg="{{asset('assets/front/img/sliders/'.$slider->image)}}">
            <div class="kunteynir">
                <div class="hero-txt">
                    <span>{{convertUtf8($slider->title)}}</span>
                    <h1>{{convertUtf8($slider->text)}}</h1>
                    @if (!empty($slider->button_url) && !empty($slider-
>button_text))
                        <a href="{{ $slider->button_url }}" class="hero-boxed-
btn">{{convertUtf8($slider->button_text)}}</a>
                    @endif
                </div>
            </div>
        </div>
    @endforeach
@endif
```

кінець лістингу 3.1

Працівник магазину, який має доступ до адмін-панелі може додавати, видаляти або редагувати новини. На рисунку 3.2 зображено вигляд форми додавання/редагування новини.

Рисунок 3.2 – Форма додавання/редагування новин

У лістингу 3.2 наведена частина контроллера, яка відповідає за збереження слайдера.

Лістинг 3.2 – Контроллер слайдера

```
public function store(Request $request)
{
    $image = $request->image;
    $allowedExts = array('jpg', 'png', 'jpeg', 'svg');
    $extImage = pathinfo($image, PATHINFO_EXTENSION);

    $messages = [
        'language_id.required' => 'The language field is required'
```

```

];

$rules = [
    'language_id' => 'required',
    'image' => 'required',
    'title' => 'nullable',
    'title_font_size' => 'required|integer|digits_between:1,3',
    'text' => 'nullable',
    'text_font_size' => 'required|integer|digits_between:1,3',
    'button_text' => 'nullable',
    'button_text_font_size' => 'required|integer|digits_between:1,3',
    'button_url' => 'nullable|max:255',
    'serial_number' => 'required|integer',
];
$be = BasicExtended::first();
$validator = Validator::make($request->all(), $rules, $messages);
if ($validator->fails()) {
    $errmsgs = $validator->getMessageBag()->add('error', 'true');
    return response()->json($validator->errors());
}
$slider = new Slider;
$slider->language_id = $request->language_id;
$slider->title = $request->title;
$slider->title_font_size = $request->title_font_size;
$slider->button_text = $request->button_text;
$slider->button_text_font_size = $request->button_text_font_size;
$slider->button_url = $request->button_url;
if ($request->filled('image')) {
    $filename = uniqid() . '.' . $extImage;
    @copy($image, 'assets/front/img/sliders/' . $filename);
    $slider->image = $filename;
}
$slider->serial_number = $request->serial_number;
$slider->save();
Session::flash('success', 'Slider added successfully!');
return "success";
}

```

кінець лістингу 3.2

Аналогічним чином реалізовано усі інші блоки на головній сторінці, весь контент може адмініструватись без втручання в код, лише за допомогою адмін-панелі. Крім цього, реалізовано адміністрування товарів, їх категорій та тегів.

За допомогою створених у адмін-панелі категорій та тегів на сторінці товарів реалізована фільтрація товарів. Блок фільтрів (рис. 3.3) автоматично

генерується за допомогою даних які раніше були занесені в базу. Крім цього, реалізований пошук за ключовими словами.

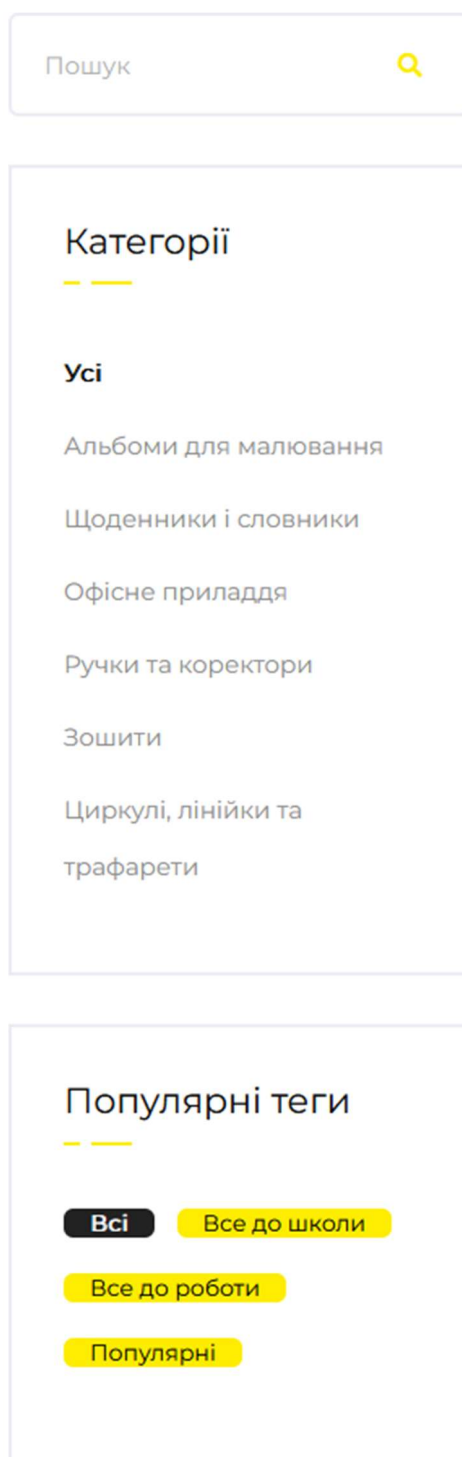


Рисунок 3.3 – Блок фільтрів

На сторінці товару користувач може дізнатись інформацію про товар, передивитись його фотографії, почитати коментарі інших користувачів та

побачити його оцінку (рис. 3.4). Для того щоб користувач міг залишити відгук про товар та оцінити його, попередньо йому потрібно авторизуватись.

Опис Відгуки (1)

Зошит 96 аркушів клітина Тешка. Тетрада TE214

☆☆☆☆☆

 milka – 28-05-2025
Чудовий зошит

Коментар

Comment

Рейтинг *

☆☆☆☆☆

Підтвердити

Рисунок 3.4 – Відгуки користувачів

3.2 Тестування та налагодження інформаційно-комп'ютерної системи

Після завершення основної реалізації програмного забезпечення інтернет-магазину канцтоварів було проведено комплексне тестування та налагодження функціональних модулів системи. Метою цього етапу є забезпечення коректної роботи всіх компонентів, виявлення можливих помилок та перевірка відповідності вимогам, визначеним на етапі проєктування.

Тестування проводилося в декілька етапів:

- ручне тестування – перевірка функціоналу через взаємодію з інтерфейсом;
- модульне тестування – за допомогою PHPUnit для верифікації окремих методів контролерів та моделей;

- інтеграційне тестування – перевірка коректної взаємодії між модулями (наприклад, реєстрація замовлення);
- тестування інтерфейсу користувача – перевірка відповідності верстки очікуваному вигляду, адаптивність, юзабіліті;
- тестування маршрутизації – перевірка всіх URL-адрес на наявність 404-помилки та коректність перенаправлень.

Основні перевірені сценарії включали:

- успішну реєстрацію та авторизацію користувача з підтвердженням email;
 - додавання товарів до кошика, збереження позицій між сторінками;
 - оформлення замовлення з вибором способу доставки та оплати;
 - валідацію форм (email, обов'язкові поля, довжина, формат);
 - роботу пошуку та фільтрації товарів за категоріями та тегами;
 - доступ до панелі адміністратора тільки для авторизованих користувачів;
 - роботу з промокодами та купонами на знижку;
 - відправку повідомлень на email після підтвердження дій.
- Під час тестування були виявлені незначні помилки, зокрема:
- некоректне відображення повідомлення про помилку при неправильному форматі email;
 - некоректне оновлення суми в кошику при зміні кількості товару;
 - дублювання записів у базі після повторного надсилання форми без захисту CSRF.

Усі помилки були усунуті шляхом додавання додаткових валідацій, перевірок стану форми та використання стандартних засобів захисту Laravel (наприклад, `csrf_token()` для захисту від повторного сабміту). Оновлення корзини синхронізовано з базою даних і сесіями користувача.

Для локального тестування та налагодження було розгорнуто проєкт у середовищі розробки на базі Laravel Sail (Docker), що дозволило максимально наблизити умови до реального серверного середовища.

Мінімальні системні вимоги для розгортання веб-додатку:

- операційна система: Ubuntu 20.04+, Windows 10+, macOS Mojave+;
- СКБД: MySQL 8.0 або MariaDB 10.5+;
- веб-сервер: Apache або Nginx;
- сервер пошти: SMTP-підтримка (Mailtrap, Gmail SMTP);
- Node.js: версія 16+;
- браузер: Google Chrome, Mozilla Firefox, Safari (останній реліз).

У результаті проведеного тестування та налагодження було підтверджено коректну роботу всіх модулів, відповідність розробленої системи функціональним вимогам та готовність до реального використання користувачами і адміністраторами платформи.

3.3 Розробка бази даних та взаємодія з нею засобами Laravel

В рамках створення інформаційної системи інтернет-магазину канцтоварів реалізація структури бази даних здійснюється засобами фреймворку Laravel, зокрема з використанням механізмів міграцій, моделей та ORM Eloquent.

Створення таблиць у базі даних виконується за допомогою міграцій. Міграції дозволяють описати структуру таблиць засобами PHP-коду, а не SQL-запитами, що значно полегшує версіонування та керування базою. Приклад створення таблиці Products за допомогою міграції наведено нижче (ліст. 3.3).

Лістинг 3.3 – Міграція таблиці Products

```
class CreateProductsTable extends Migration
{
    public function up()
    {
        Schema::create('products', function (Blueprint $table) {
            $table->bigIncrements('id');
            $table->string('title')->nullable();
            $table->string('slug')->nullable();
            $table->integer('language_id')->default(0);
            $table->integer('stock')->nullable();
        });
    }
}
```

```

        $table->string('sku')->nullable();
        $table->integer('category_id')->nullable();
        $table->text('tags')->nullable();
        $table->string('feature_image')->nullable();
        $table->text('summary')->nullable();
        $table->text('description')->nullable();
        $table->decimal('current_price', 11, 2)->default(0);
        $table->decimal('previous_price', 11, 2)->default(0);
        $table->integer('status')->default(1);
        $table->timestamps();

    });
}

public function down()
{
    Schema::dropIfExists('products');
}
}

```

кінець лістингу 3.3

Модель Product (ліст. 3.4) відображає таблицю products у вигляді класу. Вона визначає список полів, дозволених для масового заповнення (\$fillable), а також реляційні зв'язки з іншими моделями, як-от категорії, зображення товару та мова.

Лістинг 3.4 – Модель Product

```

<?php
namespace App;
use Illuminate\Database\Eloquent\Model;
class Product extends Model
{
    protected $fillable = [
        'title',
        'slug',
        'language_id',
        'stock',
        'sku',
        'category_id',
        'tags',
        'feature_image',
        'summary',
        'description',
        'current_price',
        'previous_price',
    ];
}

```

```

        'rating',
        'status',
        'meta_keywords',
        'meta_description',
        'type',
        'download_link',
        'download_file'
    ];

    public function category() {
        return $this->hasOne('App\Pcategory', 'id', 'category_id');
    }
    public function product_images() {
        return $this->hasMany('App\ProductImage');
    }

    public function language() {
        return $this->belongsTo('App\Language');
    }
}

```

кінець лістингу 3.4

Для створення нового товару в базу даних реалізовано метод `store()` у відповідному контролері. Він містить складну логіку перевірки даних, обробки зображень, перевірки типу товару (фізичний або цифровий), роботи з файлами та збереження пов'язаних записів, зокрема зображень у таблиці `product_images`.

Застосовуються функції валідації через фасад `Validator`, логіка перевірки поля `slug` на унікальність, а також обробка файлів, завантажених через форму. Файл зображення копіюється у відповідну директорію, а при типі «digital» виконується додаткова перевірка на формат і зберігання архіву.

Наприкінці створюється новий об'єкт `Product`, і через метод `create()` модель автоматично зберігається в таблицю `products`, після чого зберігаються пов'язані зображення.

Такий підхід дозволяє чітко відокремити логіку взаємодії з базою даних на рівні моделей, застосовуючи вбудовані засоби `Laravel`, що робить код більш чистим, зрозумілим та легким у супроводі.

3.4 Інструкція користувача з розгортання вебдодатку

Для того, щоб користуватись сайтом, його потрібно розмістити на хостингу. Для цього потрібно виконати певну послідовність дій.

Перед початком потрібно зареєструвати та активувати свій аккаунт на хостингу та додати домен.

Після цього потрібно перейти до файлового менеджера вашого хостингу та закинути архів із файлами проекту на один рівень із папкою `public_html`. Після цього розархівуйте його у ту ж саму папку Для прикладу назвемо проект `laravel`. Тоді структура папок буде такою, як на рисунку 3.5.

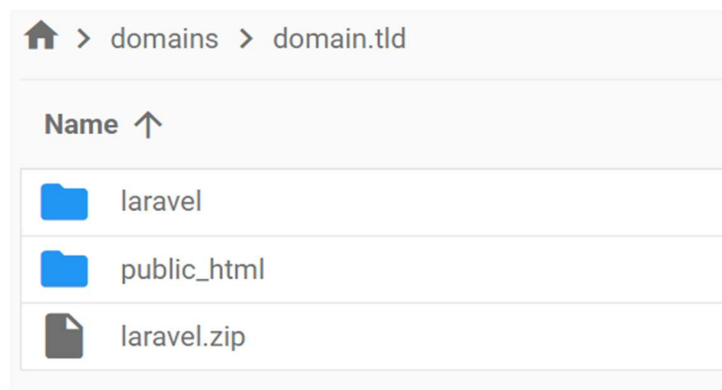


Рисунок 3.5 – Вікно файлового менеджера

Після цього відкрийте папку `laravel/public/` і перемістіть усі файли з неї в `public_html` (рис. 3.6).

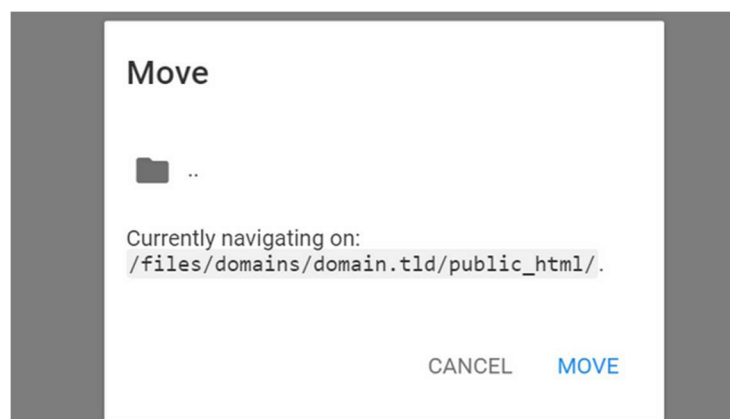
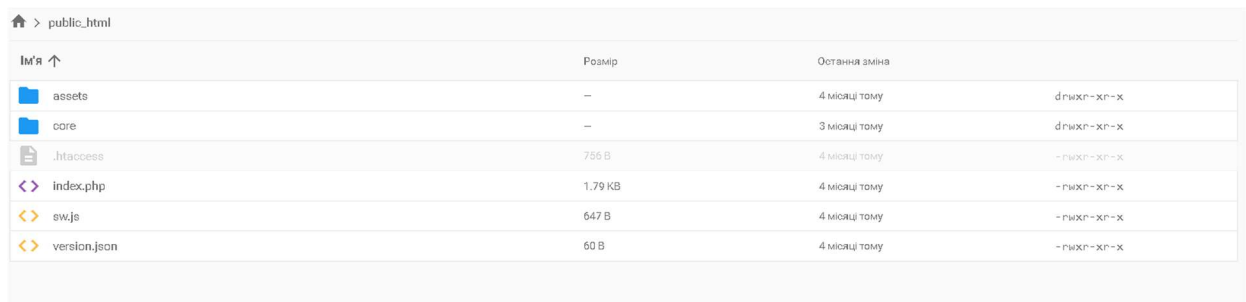


Рисунок 3.6 – Вікно переміщення файлів

Після переміщення структура файлів у папці `public_html` буде як на рисунку 3.7.



Ім'я ↑	Розмір	Остання зміна	
assets	—	4 місяці тому	d7wxp-xp-x
core	—	3 місяці тому	d7wxp-xp-x
.htaccess	755 B	4 місяці тому	-7wxp-xp-x
index.php	1.79 KB	4 місяці тому	-7wxp-xp-x
sw.js	647 B	4 місяці тому	-7wxp-xp-x
version.json	60 B	4 місяці тому	-7wxp-xp-x

Рисунок 3.7 – Вміст папки `public_html`

Потім потрібно створити базу даних. Щоб створити базу даних, перейдіть до Веб-сайти → Керування, знайдіть на бічній панелі «Керування» та натисніть на нього. Далі переконайтеся, що веб-сайт, якому буде призначено базу даних, вибрано в спадному меню ліворуч. У розділі «Створити нову базу даних MySQL і користувача бази даних» введіть ім'я бази даних MySQL, ім'я користувача та пароль, а потім натисніть «Створити».

Наступним кроком буде редагування файлу `.env`. У ньому потрібно змінити дані про базу даних на такі, як на рисунку 3.8.

```
APP_NAME='undocms '
APP_ENV=local
APP_KEY=base64:UzLL/xfwc35mARcz2bSuoJdZk70VwdCXKJPYRvzJTyg=
APP_DEBUG=true
APP_URL=http://localhost

LOG_CHANNEL=stack

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=u321947816_Rc6ZP
DB_USERNAME=u321947816_NzoLz
DB_PASSWORD="Rc6ZP"
```

Рисунок 3.8 – Файл `.env`

Після створення бази даних підключіться до свого облікового запису через SSH, перейдіть до каталогу `laravel` і введіть команду «`php artisan migrate`», щоб перенести всі файли до бази даних.

3.5 Захист інформаційно-комп'ютерної системи

Одним із ключових аспектів розробки веб-додатку є забезпечення надійного захисту інформації, зокрема даних користувачів, товарів, замовлень та облікових записів. У рамках проєкту інтернет-магазину канцтоварів захист реалізовано на кількох рівнях, із використанням вбудованих механізмів `Laravel`.

Основним інструментом для автентифікації користувачів у системі є стандартний механізм авторизації `Laravel`, доповнений бібліотекою `Sanctum`. Цей модуль дозволяє реалізувати безпечну перевірку користувачів за допомогою токенів доступу, що особливо важливо для API-запитів, мобільних додатків і SPA.

Реалізовано розмежування доступу до функціоналу на основі ролей користувачів. Звичайний користувач має доступ до перегляду товарів, оформлення замовлень, перегляду історії покупок. Адміністратор отримує розширені права – редагування товарів, керування замовленнями, категоріями, користувачами тощо.

Захист від несанкціонованого доступу реалізовано за допомогою `middleware`. Кожен маршрут захищено перевіркою ролі користувача. Для захисту форм від повторної відправки та зовнішніх атак застосовується CSRF-захист, який активовано автоматично у `Laravel`. Кожна форма включає токен, що перевіряється сервером при обробці запиту.

Паролі користувачів не зберігаються у відкритому вигляді – для їх хешування використовується функція `bcrypt()`, яка забезпечує надійний рівень захисту від зломів і витоку даних.

Валідація усіх вхідних даних відбувається перед обробкою запитів. Застосовуються як вбудовані правила валідації (required, numeric, email тощо), так і користувацькі функції. Це дозволяє уникати ін'єкцій, типових помилок при обробці форм, а також попереджати атаки типу XSS (вставка небезпечного HTML або скриптів).

Laravel Eloquent ORM автоматично екранує вхідні значення при формуванні SQL-запитів, тим самим виключаючи можливість SQL-ін'єкцій. Усі параметри запитів передаються через підготовлені вирази, що гарантує безпеку на рівні взаємодії з базою даних.

Конфіденційна інформація (ключі доступу, SMTP-дані, параметри БД) зберігається у .env файлі, який не передається у відкритий репозиторій. Це дозволяє централізовано змінювати налаштування середовища без втручання у код і зменшує ризики витоку даних.

Крім того, реалізовано обмеження на типи файлів при завантаженні зображень або архівів – перевіряється формат, розмір та вміст, щоб уникнути потенційного завантаження шкідливого ПЗ або виконуваних скриптів.

Завдяки впровадженним заходам система забезпечує базовий рівень інформаційної безпеки, відповідає сучасним стандартам захисту даних та гарантує стабільну роботу як для кінцевих користувачів, так і для адміністратора ресурсу.

Висновок до розділу 3

У даному розділі було здійснено практичну реалізацію веб-додатку – інтернет-магазину канцтоварів «Мольберт» – відповідно до попередньо сформованих вимог і поставлених завдань. Було описано структуру та компоненти системи, а також продемонстровано процес проектування та розробки ключових функціональних модулів.

Розробка здійснювалась із використанням фреймворку Laravel, що забезпечило гнучку архітектуру, модульність та високу продуктивність

системи. У процесі реалізації створено адаптивний інтерфейс для клієнтів і захищену адміністративну панель для керування товарами та замовленнями. Особливу увагу приділено взаємодії з базою даних засобами Eloquent ORM, налаштуванню маршрутів, валідації вхідних даних, захисту форм, механізмам авторизації та тестування.

Було реалізовано функціонал реєстрації, авторизації, фільтрації товарів, кошика, оформлення замовлень, email-сповіщень і захисту від типових веб-загроз. Також описано процес налагодження та тестування, що дозволило виявити і усунути критичні помилки до розгортання системи на хостингу.

У підсумку, розроблена інформаційна система є повноцінним веб-рішенням для електронної комерції, яке відповідає сучасним вимогам щодо зручності, безпеки та масштабованості, а також може бути розгорнуте у продакшн-середовищі для реального використання.

ВИСНОВКИ

У межах виконання кваліфікаційної роботи було реалізовано повноцінний веб-додаток – інтернет-магазин канцтоварів «Мольберт» – який відповідає сучасним вимогам до функціональності, безпеки, адаптивності та масштабованості.

Було здійснено аналіз предметної області електронної комерції з фокусом на ринок канцтоварів. Було зібрано інформацію про основні вимоги потенційних покупців і потреби адміністраторів онлайн-магазину. Це дозволило сформулювати чітке розуміння цільової аудиторії та функціональних очікувань від системи.

Проведено огляд і оцінку існуючих рішень у сфері електронної торгівлі, зокрема платформ, що спеціалізуються на продажі офісного приладдя. Аналіз конкурентів, а саме інтернет-магазини «7office» та «ОфісЦентр», допоміг визначити сильні сторони і недоліки їхніх рішень, що вплинуло на формування вимог до функціоналу й структури розроблюваного вебдодатку.

На основі отриманих результатів було сформульовано технічні, функціональні й нефункціональні вимоги до системи. Вони охоплюють адаптивність, масштабованість, безпеку, зручність користування, а також стабільність роботи й сумісність із сучасними вебтехнологіями.

Розроблено архітектуру інтернет-магазину, яка включає структуру бази даних, логіку взаємодії між компонентами та модель MVC. Обрано фреймворк Laravel, що забезпечив гнучке й ефективне управління логікою програми, зокрема в частині взаємодії з базою даних, обробки запитів та побудови маршрутизації.

У рамках реалізації було створено ключові модулі системи, включаючи головну сторінку, каталог із системою фільтрації, кошик, форму замовлення, особистий кабінет користувача та адмінпанель для управління товарами, категоріями, користувачами й замовленнями.

Після реалізації функціоналу проведено всебічне тестування: функціональне, модульне, інтеграційне й UX/UI. Це дозволило переконатися у працездатності додатку, відповідності заявленим вимогам та зручності користування.

Підготовлено супровідну документацію, що включає інструкції з інсталяції, налаштування та використання системи. Описано логіку роботи модулів, вимоги до середовища розгортання та варіанти масштабування для майбутнього розширення функціоналу.

Окремо реалізовано низку заходів для підвищення інформаційної безпеки. Забезпечено автентифікацію користувачів, контроль прав доступу, шифрування даних, а також захист від найпоширеніших вразливостей – SQL-ін'єкцій, міжсайтового скриптингу (XSS), міжсайтової підробки запиту (CSRF). При розробці враховано вимоги до конфіденційності відповідно до міжнародних стандартів.

Таким чином, поставлена мета була досягнута в повному обсязі: створено надійний, безпечний, функціональний інтернет-магазин канцтоварів, який відповідає вимогам сучасної веброзробки та має потенціал для подальшого вдосконалення.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. 19 ecommerce statistics you need to know in 2023. Oberlo. URL: <https://www.oberlo.com/blog/ecommerce-statistics> (дата звернення: 19.02.2025).
2. The impact of technological advancement on culture and society – Scientific Reports. Nature. URL: <https://www.nature.com/articles/s41598-024-83995-z> (дата звернення: 19.02.2025).
3. CMS – що це таке і як працює, види та приклади. HOSTiQ Wiki. URL: <https://hostiq.ua/wiki/ukr/cms/> (дата звернення: 22.02.2025).
4. Вибираємо конструктор сайтів: огляд 25 найкращих варіантів. SendPulse UA. Блог про email та інтернет-маркетинг. URL: <https://sendpulse.ua/blog/best-web-page-builders> (дата звернення: 22.02.2025).
5. Конструктори сайтів для створення вебсайту: безкоштовні та платні сайтбілдери – Ukrainian Digital Community. URL: <https://ukrainiandigital.com/konstruktory-saytiv-iaki-buvaiut-ta-iak-obraty/> (дата звернення: 23.02.2025).
6. Інтернет-магазин канцтоварів «7office». URL: <https://7office.ua/> (дата звернення: 23.02.2025).
7. Інтернет-магазин канцтоварів «ОфісЦентр». URL: <https://office-centre.com.ua/> (дата звернення: 23.02.2025).
8. Installation – laravel 12.x – the PHP framework for web artisans. Laravel. URL: <https://laravel.com/docs/12.x> (дата звернення: 3.03.2025).
9. Artisan console – laravel 10.x – the PHP framework for web artisans. Laravel. URL: <https://laravel.com/docs/10.x/artisan> (дата звернення: 3.03.2025).
10. Laravel sanctum – laravel 10.x – the PHP framework for web artisans. Laravel. URL: <https://laravel.com/docs/10.x/sanctum> (дата звернення: 3.03.2025).
11. DB-Engines ranking. URL: <https://db-engines.com/en/ranking> (дата звернення: 4.03.2025).
12. PHP: mysql (original) – manual. PHP. Hypertext Preprocessor. URL: <https://www.php.net/manual/en/book.mysql.php> (дата звернення: 4.03.2025).

13. GeeksforGeeks. Architecture of MySQL – GeeksforGeeks. GeeksforGeeks. URL: <https://www.geeksforgeeks.org/architecture-of-mysql/> (дата звернення: 7.03.2025).

14. Database: getting started - laravel 12.x – the PHP framework for web artisans. Laravel. URL: <https://laravel.com/docs/12.x/database> (дата звернення: 7.03.2025).

15. Хостинг MySQL – Amazon RDS для MySQL – AWS. Amazon Web Services, Inc. URL: <https://aws.amazon.com/ru/rds/mysql/> (дата звернення: 9.03.2025).