

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТЕХНОЛОГІЙ

ННІ/факультет Навчально-науковий інститут економіки та бізнес-освіти
Кафедра Економіки та цифрового бізнесу
Спеціальність 122 «Комп'ютерні науки»
Форма навчання Денна

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

Тимофєєва Дениса Олександровича
(прізвище, ім'я, по батькові здобувача)

на тему Розробка сайту-каталогу виробника одягу

(повна назва теми)

за матеріалами _____

(повна назва бази дослідження)

науковий керівник к.т.н. Селезньов М.Є.
(наук. ступінь, вчене звання) (підпис) (прізвище, ініціали)

Робота допущена до захисту в ЕК

Протокол засідання кафедри
від 09 червня 2025р. № 12

Завідувач кафедри _____
(підпис)

к.е.н., доцент
наук. ступінь, вчене звання

Радько В.М.
прізвище, ініціали

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕКОНОМІКИ ТА БІЗНЕС-ОСВІТИ
(повне найменування вищого навчального закладу)

Кафедра економіки та цифрового бізнесу
Освітній ступінь бакалавр
Спеціальність 122 «Комп'ютерні науки»

ЗАТВЕРДЖУЮ
Завідувач кафедри _____ **В.М. Радько**

“07” квітня 2025 року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА ЗДОБУВАЧУ

_____ Тимофєєву Денису Олександровичу _____

1. Тема роботи «Розробка сайту-каталогу виробника одягу»

науковий керівник роботи _____ Селезньов Максим Євгенович _____,
затвержені наказом вищого навчального закладу від «04» квітня 2025 р. № 224-ст (д/ф)
№ 151-ст (з/ф)

2. Строк подання здобувачем роботи 31.05.2025р.

3. Зміст кваліфікаційної роботи бакалавра, об'єкт, предмет та мета дослідження:

Розділ 1 ТЕОРЕТИЧНІ АСПЕКТИ РОЗРОБКИ САЙТУ-КАТАЛОГУ ДЛЯ ВИРОБНИКА ОДЯГУ

Розділ 2 ПРОЄКТУВАННЯ ТА РОЗРОБКА САЙТУ-КАТАЛОГУ ВИРОБНИКА ОДЯГУ

Розділ 3 ФУНКЦІОНАЛ ТА ЕЛЕМЕНТИ ІНТЕРФЕЙСУ САЙТУ-КАТАЛОГУ

Об'єкт дослідження – процес розробки інформаційної системи у вигляді Web-ресурсу для представлення та адміністрування асортименту продукції

Предмет дослідження – функціональні та технічні аспекти розробки сайту-каталогу для виробника одягу

Мета кваліфікаційної роботи бакалавра – розробка зручного, функціонального сайту-каталогу, що дозволяє ефективно презентувати одяг, здійснювати пошук продукції та створювати замовлення.

4. Дата видачі завдання 04.04.2025р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи бакалавра	Строк виконання етапів роботи	Відмітка керівника про виконання етапів (дата, підпис)
1	Підготовка розділу 1	до 28.04.2025р.	27.04.2025
2	Підготовка розділу 2	до 16.05.2025р.	14.05.2025
3	Підготовка розділу 3	до 30.05.2025р.	29.05.2025
4	Реєстрація завершеної дипломної роботи	до 31.05.2025р.	30.05.2025
5	Отримання відгуку від наукового керівника	03-04.06.2025р.	04.06.2025
6	Отримання зовнішньої рецензії	05-06.06.2025р.	06.06.2025
7	Перевірка кваліфікаційної роботи на плагіат	02-09.06.2025р.	04.06.2025
8	Попередній захист кваліфікаційної роботи на кафедрі	03.06.2025р.	03.06.2025
9	Допуск кафедрою кваліфікаційної роботи до захисту	09.06.2025р.	09.06.2025
10	Підготовка студента до захисту в ЕК	до 17.06.2025р.	17.06.2025

Завдання підготував науковий керівник _____

(підпис)

Селезньов М. Є

(прізвище та ініціали)

Завдання одержав здобувач _____

(підпис)

Тимофєєв Д. О.

(прізвище та ініціали)

Примітки:

1. Форму призначено для видачі завдання здобувачу на виконання кваліфікаційної роботи бакалавра і контролю за ходом роботи з боку кафедри.
2. Розробляється керівником кваліфікаційної роботи. Видається кафедрою.
3. Формат бланка А4 (210×297 мм), 2 сторінки.

РЕФЕРАТ

Робота містить 48 сторінок, 18 рисунків, 35 джерел і 2 додатки.

Об'єкт дослідження: процес розробки інформаційної системи у вигляді Web-ресурсу для представлення та адміністрування асортименту продукції.

Предмет дослідження: функціональні та технічні аспекти розробки сайту-каталогу для виробника одягу.

Мета дослідження: розробка зручного, функціонального сайту-каталогу, що дозволяє ефективно презентувати одяг, здійснювати пошук продукції та створювати замовлення.

У рамках розробки сайту-каталогу використано базові Web-технології, такі як HTML, CSS та JavaScript. Це дозволило зробити проєкт легким для розгортання та підтримки, а також забезпечує високу гнучкість для подальших доповнень і змін. Передбачена також система авторизації адміністратора, яка обмежує доступ до функцій керування каталогом, що є необхідним елементом безпеки навіть у концептуальних проєктах.

Область застосування: результати роботи можуть бути використані при створенні повноцінної Web-платформи електронної комерції, орієнтованої на продаж одягу та аксесуарів з можливістю використання платіжних та логістичних сервісів.

WEB-ТЕХНОЛОГІЇ, WEB-РЕСУРС, САЙТ-КАТАЛОГ, ОДЯГ,
JAVASCRIPT, АВТОМАТИЗАЦІЯ, ЕЛЕКТРОННА КОМЕРЦІЯ,
ЦИФРОВІЗАЦІЯ БІЗНЕСУ.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	6
ВСТУП	7
РОЗДІЛ 1 ТЕОРЕТИЧНІ АСПЕКТИ РОЗРОБКИ САЙТУ-КАТАЛОГУ ДЛЯ ВИРОБНИКА ОДЯГУ	9
1.1 Актуальність розробки Web-ресурсів для виробників одягу в умовах сучасного цифрового ринку	9
1.2 Огляд існуючих цифрових рішень для онлайн-торгівлі	12
1.3 Вибір технологічного стеку розробки сайту-каталогу	15
Висновки до розділу 1	17
РОЗДІЛ 2 ПРОЄКТУВАННЯ ТА РОЗРОБКА САЙТУ-КАТАЛОГУ ВИРОБНИКА ОДЯГУ	19
2.1 Проєктування інформаційної моделі сайту-каталогу	19
2.2 Проєктування бази даних сайту-каталогу	29
2.3 Огляд необхідних алгоритмічних методів	32
Висновки до розділу 2	34
РОЗДІЛ 3 ФУНКЦІОНАЛ ТА ЕЛЕМЕНТИ ІНТЕРФЕЙСУ САЙТУ- КАТАЛОГУ	35
3.1 Розробка окремих модулів Web-каталогу	35
3.2 Огляд реалізованого функціоналу сайту-каталогу	39
3.3 Аналіз створеного сайту-каталогу та напрямки його подальшого розвитку	47
Висновки до розділу 3	50
ВИСНОВКИ	52
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	55
ДОДАТКИ	58

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

БД – База даних;

СКБД – Система керування базами даних;

API – Application Programming Interface;

CMS – Content Management System;

CRM – Customer Relationship Management;

SEO – Search Engine Optimization;

SPA – Single Page Application;

UML – Unified Modeling Language;

UX – User Experience.

ВСТУП

У сучасному світі інформаційні технології відіграють ключову роль у розвитку бізнесу, культури, освіти та практично всіх інших сфер діяльності. Зокрема, стрімке зростання електронної комерції та цифрових каналів комунікації зумовлює потребу у створенні зручних і доступних інтернет-ресурсів для взаємодії з споживачами. Для виробників одягу, як і для інших підприємств легкої промисловості, особливо важливо мати представництво в онлайн-просторі, що дозволяє презентувати продукцію широкій аудиторії, оперативно оновлювати асортимент і підтримувати зв'язок з клієнтами.

Одним з найбільш ефективних інструментів для досягнення цих цілей є Web-сайт у форматі каталогу. Такий ресурс не лише виконує інформативну функцію, а й може стати основою для побудови повноцінного цифрового середовища підприємства.

Сайт-каталог забезпечує можливість зручного перегляду товарів, фільтрації за певними критеріями, формування списку замовлень та взаємодії з адміністрацією ресурсу. Це значно полегшує процес знайомства потенційного клієнта з продукцією та сприяє формуванню лояльної аудиторії. При цьому, особливої актуальності розробка сайтів-каталогів набуває для малих та середніх виробників, які не завжди мають змогу використовувати дорогі комерційні платформи. Простий, але функціональний Web-ресурс з можливістю керування контентом через інтерфейс адміністратора - це доступне рішення, яке здатне задовольнити базові потреби бізнесу.

Об'єкт дослідження: процес розробки інформаційної системи у вигляді Web-ресурсу для представлення та адміністрування асортименту продукції.

Предмет дослідження: функціональні та технічні аспекти розробки сайту-каталогу для виробника одягу.

Метою дослідження є розробка зручного, функціонального сайту-каталогу, що дозволяє ефективно презентувати одяг, здійснювати пошук продукції та створювати замовлення.

В процесі дослідження були виконані наступні завдання:

- оцінити актуальність розробки Web-ресурсів для виробників одягу в умовах сучасного цифрового ринку;
- виконати огляд існуючих цифрових рішень для онлайн-торгівлі;
- обґрунтувати вибір технологічного стеку розробки сайту-каталогу;
- виконати проектування інформаційної моделі та бази даних сайту-каталогу;
- виконати розробку окремих модулів Web-каталогу;
- провести аналіз створеного сайту-каталогу та визначити напрямки його подальшого розвитку.

В процесі проведення дослідження використовувалися такі методики:

- аналіз предметної області;
- порівняльний аналіз;
- моделювання з використанням UML;
- системний аналіз.

Практична значущість результатів дослідження полягає в отриманому функціональному макеті сайту-каталогу, який можна використати як шаблон або основу для реального Web-проєкту. Розроблений Web-ресурс демонструє основні принципи побудови структури каталогу, реалізації фільтрації товарів, інтерактивної взаємодії з користувачем та створення замовлень, що робить його придатним для подальшої адаптації під потреби реальних виробників одягу або аналогічних комерційних структур.

РОЗДІЛ 1

ТЕОРЕТИЧНІ АСПЕКТИ РОЗРОБКИ САЙТУ-КАТАЛОГУ ДЛЯ ВИРОБНИКА ОДЯГУ

1.1 Актуальність розробки Web-ресурсів для виробників одягу в умовах сучасного цифрового ринку

У сучасну епоху цифрової трансформації майже всі сфери бізнесу активно інтегрують IT-рішення для оптимізації процесів, розширення аудиторії та підвищення конкурентоспроможності. Особливо це стосується галузей, де візуальне представлення товарів і бренду відіграє вирішальну роль - зокрема, індустрії моди. Сегмент виробництва одягу постійно розвивається: змінюються тренди, зростає конкуренція, збільшується кількість каналів комунікації з клієнтами. У такій динамічній екосистемі наявність сучасного Web-ресурсу - не просто побажання, а необхідність.

Одним із найбільш ефективних інструментів для представлення продукції у мережі є сайт-каталог. Він поєднує в собі функції цифрової вітрини, інформативного ресурсу та комунікаційної платформи. Завдяки зручній навігації, структурованій подачі інформації та можливості інтеграції з соціальними мережами чи зовнішніми сервісами, сайт-каталог суттєво розширює можливості компанії щодо просування своєї продукції.

Метою цієї роботи є розробка повнофункціонального Web-сайту-каталогу для виробника одягу, який дозволить ефективно презентувати асортимент продукції, підвищити впізнаваність бренду, забезпечити комфортну взаємодію з клієнтами та створити основу для подальшого розвитку цифрової інфраструктури компанії.

Такий сайт повинен:

- Відображати структурований асортимент товарів: поділ на категорії (наприклад, верхній одяг, повсякденний одяг, аксесуари), наявність підкатегорій, можливість додавання нових груп товарів без порушення логіки побудови.

- Надавати користувачу зручні інструменти навігації та пошуку - фільтри, сортування, ключове слово, категорії, колір, розмір, матеріал тощо.
- Мати детальні картки товарів з повною інформацією, декількома фотографіями, таблицями розмірів, рекомендаціями або супутніми товарами.
- Забезпечувати адаптивність інтерфейсу - сайт повинен коректно відображатись і залишатись функціональним на пристроях із різними розмірами екранів.
- Реалізовувати інтуїтивно зрозумілий та візуально привабливий інтерфейс, який формує у користувача позитивний досвід взаємодії з брендом.
- Підтримувати інтеграцію із засобами зв'язку - формою зворотного зв'язку, месенджерами, мапою з контактами, кнопками соцмереж тощо.
- Мати адміністративну частину, де можна додавати, редагувати або видаляти товари, керувати замовленнями (за потреби), додавати новини або оголошення.

Окремо слід зазначити важливість відповідності сайту технічним стандартам: безпека даних, швидке завантаження сторінок, коректна робота у різних браузерах, базова SEO-оптимізація, можливість масштабування проєкту.

Крім технічних аспектів, у фокусі розробки повинно бути розуміння потреб цільової аудиторії. Це означає: продумана структура розділів, доступна мова, естетичний стиль, що відповідає цінностям бренду, а також можливість емоційного залучення користувача.

Завдяки зручному інтерфейсу, користувач зможе швидко знаходити необхідний товар, застосовуючи фільтри за категоріями, вибираючи розміри та інші параметри. Це значно покращує користувацький досвід (UX), адже користувачі отримують швидкий доступ до того, що їм цікаво, а також можуть замовити товар без зайвих зусиль [1-4].

Переваги використання сайту-каталогу для виробників одягу:

1. Систематизація асортименту. Сайт дозволяє чітко структурувати продукцію за категоріями, такими як жіночий, чоловічий, дитячий одяг,

аксесуари, що дає змогу зручно сортувати і подавати товар залежно від запитів користувачів.

2. Оперативне оновлення асортименту. Web-сайт дозволяє виробнику в реальному часі оновлювати інформацію про наявність товару, нові надходження, зміни в цінах чи акціях. Це дозволяє швидко адаптувати пропозицію до потреб ринку та зміни сезону.

3. Інформування клієнтів. Через сайт можна ефективно інформувати клієнтів про нові надходження, акції чи спеціальні пропозиції, що є важливим для стимулювання повторних покупок та підтримки лояльності споживачів.

4. Залучення нових клієнтів і збільшення довіри. Наявність якісного, професійно оформленого сайту-каталогу підвищує імідж бренду і додає довіри серед користувачів, оскільки багато покупців сьогодні орієнтуються на професіоналізм компанії, зокрема через її сайт.

5. Спрощення процесу замовлення та зв'язку. Сайт-каталог дозволяє значно спростити процес покупки та звернення до менеджерів, а також інтегрувати онлайн-консультації, системи для оформлення замовлень, що знижує час, необхідний для обробки заявки та підвищує рівень обслуговування.

Незважаючи на численні переваги, в Україні багато виробників одягу все ще не мають власного сайту або мають тільки обмежене представлення через соціальні мережі, що не дозволяє ефективно реалізувати потенціал онлайн-продажів. Більшість брендів використовують Instagram, Facebook чи інші платформи для представлення продукції. Однак соціальні мережі мають обмежений функціонал, що не дозволяє організувати повноцінну структуру каталогу, фільтрацію товарів або зручне оформлення замовлення.

Такі платформи також мають недоліки в частині довіри. Користувачі надають перевагу брендам, які мають професійно оформлений Web-сайт. Власний сайт додає компанії солідності і викликає більше довіри у потенційних покупців, оскільки вони можуть бути впевнені в безпеці своїх персональних даних і фінансових транзакцій.

Таким чином, сайт-каталог є основним інструментом для представлення продукції в Інтернеті, адже на відміну від звичайних сайтів-візиток, які надають лише загальну інформацію про компанію, сайт-каталог дає змогу користувачеві детально ознайомитися з товаром, побачити його характеристики, варіанти кольорів, розмірів і стилів, а також оцінити наявність товару в асортименті. Окрім цього, сайт-каталог дозволяє автоматизувати процес продажів, знижуючи залежність від зовнішніх ресурсів або платформ. При цьому створення сайту-каталогу - це не лише технічне завдання, а комплексна робота, що поєднує аналіз ринку, дизайн-мислення, знання користувацьких потреб і сучасні ІТ-інструменти. Успішна реалізація такого проекту стане важливим кроком у цифровій трансформації виробника одягу, сприятиме зростанню клієнтської бази та зміцненню позицій компанії на ринку.

1.2 Огляд існуючих цифрових рішень для онлайн-торгівлі

Індустрія виробництва та продажу одягу є однією з найбільш динамічних і конкурентних галузей сучасної економіки. Завдяки постійному змінюванню модних тенденцій, новим технологіям виробництва і зростаючій ролі цифрових комунікацій, виробники одягу змушені шукати нові способи залучення клієнтів, підвищення впізнаваності бренду та збільшення обсягів продажу. У цих умовах онлайн-інструменти, зокрема Web-сайти і інтернет-магазини, займають ключове місце у стратегії компаній, орієнтуючись на забезпечення ефективної взаємодії з кінцевим споживачем.

Перед тим, як розпочати розробку сайту-каталогу для виробника одягу, важливо детально проаналізувати вже існуючі рішення на ринку. Це дозволить не лише оцінити сучасні тенденції у Web-дизайні та функціональності сайтів-каталогів, а й виокремити сильні та слабкі сторони існуючих платформ, що є важливим етапом для створення конкурентоспроможного продукту. Важливо звернути увагу як на великі міжнародні бренди, так і на українських виробників, оскільки їхній досвід може дати корисні ідеї для майбутнього проекту.

Великі міжнародні компанії, такі як Zara, H&M, Bershka, Pull&Bear, Uniqlo, надають користувачам доступ до сучасних та повнофункціональних інтернет-магазинів, які водночас виконують роль каталогів. Їхні сайти розроблені з урахуванням передових технологій та орієнтовані на максимальну зручність користувачів. Ось основні характеристики таких сайтів:

- Навігація та розподіл товарів. Всі сайти мають чітко структуровану навігацію, яка дозволяє швидко знаходити потрібні категорії товарів та підкатегорії (наприклад, за типом одягу, сезоном, ціною, тощо). Це забезпечує зручність для користувачів та знижує ймовірність втрати потенційних покупців через складну навігацію.

- Адаптивний дизайн. Web-сайти великих брендів адаптовані під всі типи пристроїв - комп'ютери, планшети, смартфони. Це забезпечує зручність перегляду незалежно від того, з якого пристрою користувач заходить на сайт. Адаптивний дизайн також сприяє високому рівню SEO-оптимізації, що дозволяє залучити більше трафіку.

- Розширені фільтри. Більшість сайтів великих брендів мають розширену систему фільтрації товарів, що дозволяє шукати товар за різними параметрами: розміром, кольором, типом тканини, ціною, новизною, популярністю тощо. Це дозволяє користувачам швидко знаходити товар, що їх цікавить.

- Інтерактивні елементи. Модні бренди активно використовують 3D-проглядання товарів, галереї, анімовані зображення при наведенні на товар, що дозволяє користувачеві побачити товар з різних ракурсів. Такі функції додають користувачеві можливість краще уявити, як виглядатиме товар у реальному житті.

- Інтеграція з платіжними системами та доставкою. Сайти великих брендів мають інтеграцію з платіжними системами, що дозволяє здійснювати онлайн-платежі, а також з системами доставки, що забезпечує користувачеві зручність і можливість відслідковувати своє замовлення.

- Авторизація та особисті кабінети. Багато сайтів дозволяють користувачам створювати особисті кабінети, де можна зберігати адреси доставки, історію покупок, а також створювати список бажаних товарів.

Однак такі функціональні можливості, як розширена фільтрація, 3D-проглядання та інтеграція з платіжними системами, можуть бути занадто складними та ресурсомісткими для невеликих виробників, особливо без наявності власних ІТ-фахівців.

З іншого боку, малі українські виробники часто створюють спрощені сайти-каталоги, створені, в основному, за допомогою готових CMS (систем управління контентом), таких як WordPress, або Joomla. Такі сайти можуть бути менш функціональними, але вони більш доступні для малого бізнесу. Основні характеристики таких сайтів:

- Сайт-візитівка з каталогом. Багато українських брендів мають сайт, який є лише статичним каталогом з обмеженою кількістю товарів. Це часто виглядає як просто демонстрація товарів без інтерактивних можливостей, що обмежує можливості користувачів.

- Прості фільтри: Якщо фільтрація товарів є на сайті, то вона часто обмежена лише базовими категоріями (наприклад, «одяг для жінок» або «акційні товари»). Це створює незручності для користувачів, оскільки вони не можуть швидко знайти потрібний товар за конкретними параметрами.

- Обмежена функціональність мобільної версії. Багато сайтів не мають повноцінної адаптації для мобільних пристроїв, що знижує комфортність використання, особливо з урахуванням високої популярності мобільних покупок.

- Якість фотографій та описів товарів. На деяких сайтах є недостатньо якісні фото товарів, що може впливати на рішення користувачів. Окрім цього, не завжди є докладні текстові описи, які б детально розкривали характеристики товару.

Способи зв'язку. Зазвичай зв'язок з клієнтом здійснюється через номер телефону, месенджери або форму зворотного зв'язку. Це менш зручно для користувача, ніж інтегровані чати чи системи замовлень.

При цьому, ринок Web-розробки сьогодні пропонує безліч можливостей для створення індивідуальних рішень для сайту-каталогу. Для більш складних завдань можливе використання кастомних рішень на фреймворках типу React, Vue або Laravel, що дозволяє створювати гнучкіші і масштабовані проекти з високою продуктивністю і безпекою.

Проведений аналіз свідчить, що хоча на ринку вже існує багато сайтів-каталогів, більшість локальних виробників не мають сучасного, зручного у використанні, функціонального сайту. Проблеми, з якими стикаються українські виробники одягу, включають недостатню систематизацію товарів, незручну навігацію, неякісну візуалізацію продукції та обмежену функціональність мобільної версії. Це створює передумови для розробки конкурентного рішення - адаптивного, простого в адмініструванні сайту-каталогу, який дозволить бренду покращити своє представлення в онлайн-просторі, зробити процес покупок зручнішим та ефективнішим для кінцевих користувачів.

1.3 Вибір технологічного стеку розробки сайту-каталогу

Вибір технологічного стеку для розробки сайту-каталогу є важливим кроком, який впливає на швидкість створення, підтримку, масштабованість, безпеку та зручність для користувачів. Для початківців важливо обрати технології, які є простими для освоєння, мають доступну документацію та дозволяють реалізувати базові функції, такі як відображення товарів, фільтрація, пошук та обробка запитів.

Сучасні Web-додатки зазвичай базуються на клієнт-серверній архітектурі, де фронтенд відповідає за інтерфейс користувача, а бекенд - за обробку даних. У випадку сайту-каталогу фронтенд відображатиме каталог товарів, а бекенд забезпечуватиме доступ до бази даних для отримання інформації про продукти та категорії.

Для фронтенду можна використовувати різні інструменти, але для навчальних цілей найкраще почати з основного технологічного стеку: HTML

(для розмітки сторінки), CSS (для стилізації), JavaScript для інтерактивних елементів. Ці інструменти є базовими та широко доступними для вивчення. Для спрощення розробки дизайну можна використати фреймворки, такі як Bootstrap або Tailwind CSS, які надають готові компоненти для швидкої розробки [5-8].

Альтернативи, як-от React.js, Vue.js чи Angular, є потужними, але можуть бути складнішими на початковому етапі через необхідність вивчення додаткових концепцій, таких як віртуальний DOM чи двосторонній зв'язок даних.

Для бекенду може бути обрано PHP, оскільки це мова, спеціально розроблена для Web-розробки, з великою кількістю навчальних матеріалів та підтримкою більшості хостингів. Фреймворки, як-от Laravel, можуть бути використані пізніше для розширення функціоналу.

Також для реалізації бекенду може бути використаний Python з фреймворками Django чи Flask, який є простим для вивчення завдяки читабельному синтаксису, але може вимагати більше зусиль для налаштування хостингу. Крім того широко використовується Node.js з Express.js, який дозволяє використовувати JavaScript для бекенду, що може спростити розробку.

Для створення бази даних може бути обрана MySQL, оскільки це реляційна СКБД, яка є безкоштовною, простою у налаштуванні та широко підтримується. Вона ідеально підходить для зберігання структурованих даних, таких як товари та категорії [9-10].

Таким чином при виборі технологічного стеку для цього проекту керувалися наступними критеріями

- простота освоєння,
- доступність розгортання,
- достатній функціонал для реалізації базових завдань,
- можливість розширення в майбутньому,
- популярність та наявність активної спільноти.

Обраний стек відповідає всім цим критеріям, забезпечуючи баланс між доступністю інструментів розробки та їх практичним застосуванням.

Загалом, стек HTML, CSS, JavaScript, PHP та MySQL є оптимальним вибором для прототипу проєкту сайту-каталогу, враховуючи його простоту та доступність. Альтернативи, як-от Python чи Node.js, можуть бути розглянуті далі для розширення функціоналу [11-13].

Таким чином, вибір технологічного стеку є ключовим етапом, який визначає не лише швидкість створення проєкту, але й його підтримку, масштабованість, безпеку та зручність для користувачів. Для такого проєкту, як сайт-каталог, важливо обрати технології, які поєднують простоту освоєння, доступність документації та можливість реалізації базового функціоналу, як-от відображення товарів, фільтрація, пошук та обробка запитів.

Висновки до розділу 1

Проведений огляд теоретичних аспектів розробки сайту-каталогу для виробника одягу, охопив актуальність створення Web-ресурсів, аналіз існуючих цифрових рішень та обґрунтування вибору технологічного стеку. У сучасних умовах цифрової трансформації сайт-каталог є необхідним інструментом для виробників одягу, оскільки він забезпечує систематизацію асортименту, оперативне оновлення інформації, залучення нових клієнтів, спрощення процесу замовлення та підвищення впізнаваності бренду. На відміну від соціальних мереж, які мають обмежений функціонал, сайт-каталог пропонує структуровану подачу товарів, розширені можливості фільтрації, адаптивний дизайн і професійний імідж, що сприяє зростанню довіри споживачів і конкурентоспроможності компанії.

Аналіз існуючих рішень на ринку показав, що великі міжнародні бренди, такі як Zara чи H&M, використовують повнофункціональні сайти з адаптивним дизайном, розширеними фільтрами, інтерактивними елементами та інтеграцією з платіжними системами, що забезпечує високий рівень користувацького досвіду. Натомість малі українські виробники часто обмежуються простими сайтами на CMS, як-от WordPress, із базовою функціональністю, що не завжди

відповідає потребам сучасних користувачів через обмежену навігацію, неякісні фото чи відсутність мобільної адаптації. Це створює потребу в розробці конкурентоспроможного сайту-каталогу, який поєднує простоту, функціональність і сучасний дизайн.

Для реалізації проєкту обрано класичний технологічний стек HTML, CSS та JavaScript, який добре підходить для прототипування Web-проєктів на етапі розробки. Цей вибір обґрунтований простотою використання, широкою підтримкою хостингів, доступністю навчальних матеріалів і можливістю реалізації базового функціоналу, такого як відображення товарів, фільтрація та додавання товарів. При цьому, використання CSS-фреймворку Bootstrap спрощує створення адаптивного дизайну. Альтернативні варіанти, такі як Node.js, Python чи Vue.js, можуть бути складнішими для початківців, але залишають простір для подальшого розвитку проєкту.

Таким чином, розробка сайту-каталогу для виробника одягу є актуальним завданням, яке поєднує сучасні IT-рішення, аналіз потреб цільової аудиторії та практичні навички Web-розробки. Обраний технологічний стек забезпечує міцний фундамент для створення функціонального прототипу, який відповідає сучасним стандартам, а також закладає основу для подальшого масштабування та вдосконалення.

РОЗДІЛ 2

ПРОЄКТУВАННЯ ТА РОЗРОБКА САЙТУ-КАТАЛОГУ ВИРОБНИКА ОДЯГУ

2.1 Проєктування інформаційної моделі сайту-каталогу

На початковому етапі розробки інформаційної системи, зокрема сайту-каталогу для виробника одягу, важливим є чітке визначення сутностей предметної області, їх властивостей, взаємозв'язків та ролі в системі. Це дозволяє сформуванню основи для подальшого проєктування структури сайту, логіки взаємодії між компонентами та ефективного відображення інформації, що стосується продукції та користувацьких процесів.

У контексті Web-сайту-каталогу об'єкти поділяються на кілька логічних груп, кожна з яких має свої особливості представлення та взаємодії:

1. Продукція (товари). Центральним об'єктом системи є товар - одиниця одягу або аксесуару, яка представлена в каталозі. Кожен товар має атрибути, серед яких: назва, опис, артикул, категорія (наприклад, жіночий одяг, чоловічий одяг, дитячий одяг, аксесуари), наявність (у розмірах/кольорах), ціна, фотографії, матеріал, розмірна сітка.

Цей об'єкт є динамічним - його дані можуть оновлюватися, змінюватися або видалятися з каталогу відповідно до актуального стану асортименту.

2. Категорії товарів. Для забезпечення зручної навігації товари класифікуються за категоріями. Категорії мають ієрархічну структуру, що дозволяє користувачу швидше знайти потрібну продукцію та реалізувати багаторівневу фільтрацію.

3. Користувачі системи. У системі передбачається щонайменше два типи користувачів:

Звичайний відвідувач (гостьовий користувач) - переглядає каталог, шукає товари, може звернутися через форму.

Адміністратор - має доступ до керування контентом: додавання, редагування та видалення товарів, оновлення інформації на сайті, перегляд запитів від користувачів.

Для кожного типу користувача визначаються дозволені дії, що є важливою складовою інформаційної безпеки.

4. Взаємодія користувача з інтерфейсом. Важливо розглядати не лише “фізичні” об’єкти, а й процеси. До таких належать: пошук товарів, перегляд картки товару, фільтрація, звернення до адміністрації, додавання товару до обраного (за потреби). Ці дії реалізуються за допомогою інтерактивних елементів інтерфейсу та створюють відгуки системи (запити, повідомлення, дії в журналі).

5. Додаткові об’єкти системи. До додаткових, але важливих об’єктів можна віднести: бренди/постачальники; зображення та мультимедійні елементи пов’язані з товарами, форма зворотного зв’язку та ін..

Всі вищенаведені об’єкти пов’язані між собою. Наприклад: товар належить до певної категорії; один товар може мати кілька зображень; кожен товар має унікальний ідентифікатор; користувач взаємодіє з товарами через інтерфейс, залишає запит або коментар; адміністратор обробляє запити, оновлює картки товарів. Ці взаємозв’язки формують основу для подальшого концептуального та логічного моделювання системи.

Оскільки мова йде про сайт виробника одягу, важливо врахувати особливості галузі: велика кількість візуального контенту (зображення товарів); часті оновлення асортименту; варіативність продукції (розміри, кольори); потреба у швидкій та гнучкій фільтрації; наявність сезонних та акційних товарів. Це впливає на побудову моделей інформаційної структури, архітектури для кастомної реалізації та підхід до організації бази даних.

Інформаційна модель - це формалізоване подання структури даних, що описує об’єкти дослідження, їх властивості та взаємозв’язки між ними. У контексті створення сайту-каталогу для виробника одягу така модель дозволяє чітко визначити, які дані необхідно зберігати, як ці дані будуть пов’язані між

собою, та яким чином вони будуть опрацьовуватись у процесі функціонування Web-застосунку.

Концептуальне проектування системи є важливим етапом, на якому визначається логічна структура програмного забезпечення, взаємозв'язки між об'єктами, а також ключові компоненти, необхідні для реалізації функціональності сайту-каталогу.

Чітке визначення об'єктів предметної області дозволяє сформуванню цілісного уявлення про функціонування системи. Далі ці об'єкти можуть бути використані для побудови інформаційної моделі, визначення атрибутів даних, проектування логіки взаємодії та побудови UML-діаграм.

UML-діаграма класів відображає основні класи (об'єкти) програмної системи, їх атрибути (властивості) та методи (функціональні можливості), а також зв'язки між ними. Вона дозволяє чітко структурувати програмну логіку та визначити, як різні частини системи взаємодіють між собою. У нашій системі виділено такі класи:

1. Product (Продукт)

Атрибути:

- id: int – унікальний ідентифікатор продукту;
- name: string – назва продукту;
- description: string – опис продукту;
- price: float – ціна продукту;

Методи:

- addProduct() – додавання нового продукту;
- updateProduct() – оновлення інформації про продукт;
- deleteProduct() – видалення продукту з каталогу.

Цей клас є центральним, оскільки система орієнтована на демонстрацію та керування товарами.

2. Category (Категорія)

Атрибути:

- id: int – унікальний ідентифікатор категорії;

- name: string – назва категорії.

Методи:

- addCategory() – додавання нової категорії;
- deleteCategory() – видалення категорії.

Категорії слугують для структурування товарів за групами (наприклад, "Чоловічий одяг", "Жіночий одяг" тощо).

3. User (Користувач)

Атрибути:

- id: int – унікальний ідентифікатор користувача;
- email: string – електронна адреса користувача.

Методи:

- login() – вхід у систему;
- register() – реєстрація нового користувача.

Цей клас репрезентує звичайного користувача системи, який може переглядати товари, шукати їх, надсилати повідомлення.

4. Admin (Адміністратор)

Атрибути:

- role: string – роль адміністратора (наприклад, “контент-менеджер”).

Методи:

- manageProduct() – керування продуктами (створення, редагування, видалення);
- manageCategory() – керування категоріями.

Адміністратор виконує дії, пов’язані з наповненням каталогу, оновленням інформації про товари та структуруванням сайту.

5. Message (Повідомлення)

Атрибути:

- id: int – унікальний ідентифікатор повідомлення;
- sender: string – ім’я або email відправника;
- content: string – текст повідомлення.

Методи:

- sendMessage() – відправити повідомлення;
- readMessage() – переглянути повідомлення.

Повідомлення є засобом комунікації між користувачем і адміністратором, або формою зворотного зв'язку.

Зв'язки між класами:

- Клас Product має зв'язок з Category, оскільки кожен товар належить до певної категорії.
- Admin взаємодіє з Product і Category, оскільки він має функції керування ними.
- User може надсилати Message, які потім читає Admin.

Отримана діаграма класів представлена на рисунку 2.1. Ця UML-діаграма є важливою частиною концептуального проектування, оскільки дозволяє узагальнити логіку роботи системи і слугує основою для майбутньої реалізації.

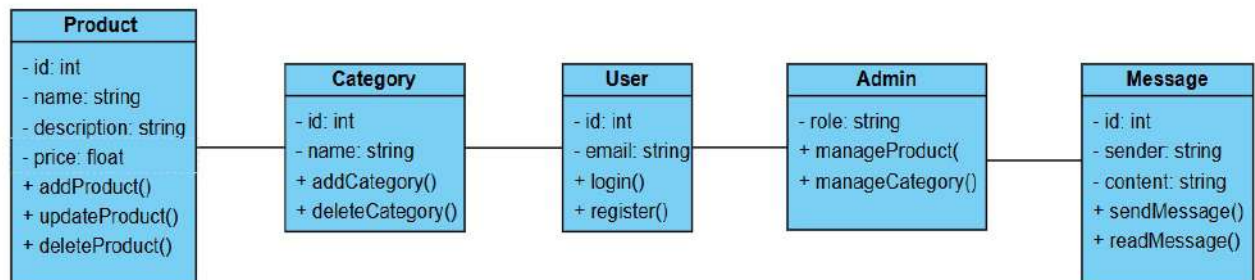


Рис. 2.1. Діаграма класів сайту-каталогу виробника одягу

Далі доцільно створити діаграму варіантів використання (Use Case Diagram), яка є одним з ключових інструментів у процесі проектування інформаційних систем. Вона дозволяє візуалізувати взаємодію користувачів (акторів) із системою, показуючи, які саме функції (випадки використання) система має реалізовувати для кожного типу користувача.

У класичному вигляді ця діаграма складається з наступних елементів:

- Акторів, зображених як чоловічки, або прямокутники з підписом ролі.
- Випадків використання, представлених у вигляді еліпсів із підписаними діями.
- Системи, окресленої прямокутником, у якому розміщуються всі випадки використання.
- Ліній зв'язку, які показують, які актори взаємодіють із якими функціями.

У контексті Web-сайту-каталогу для виробника одягу виділяються два основні актори: Користувач (відвідувач або зареєстрований покупець) та Адміністратор (особа, що керує вмістом сайту)

Користувач - це особа, яка заходить на сайт з метою перегляду товарів, пошуку, комунікації або замовлення. Вона має доступ до таких функцій:

- Реєстрація. Створення облікового запису для подальшого використання сайту (наприклад, щоб зберігати обране, відстежувати повідомлення, отримувати знижки тощо).
- Авторизація. Вхід до особистого кабінету після попередньої реєстрації.
- Перегляд товарів. Огляд продукції, представленої в каталозі, включаючи фото, описи, ціни.
- Пошук товарів. Використання пошукового рядка або фільтрів для знаходження товарів за заданими параметрами (категорії, розмір, ціна тощо).
- Надсилання повідомлень адміністратору. Можливість залишити відгук, поставити запитання або зробити пропозицію через контактну форму.

Адміністратор - це відповідальна особа, яка управляє контентом сайту і слідкує за його актуальністю. До його функцій входить:

- Управління продуктами. Додавання нових товарів, редагування вже існуючих позицій (зміна опису, фото, ціни), видалення неактуальних товарів.
- Управління категоріями. Створення, редагування або видалення категорій товарів, що дозволяє краще організувати структуру каталогу.

– Перегляд повідомлень. Доступ до повідомлень, отриманих від користувачів через сайт. Це можуть бути запити, скарги, пропозиції тощо.

Система-каталог - це центральна частина, яка об'єднує всі функції. Саме вона реагує на запити користувачів та адміністратора і забезпечує відповідні дії. Кожен випадок використання реалізується в межах цієї системи.

На рисунку 2.2 представлена розроблена діаграма випадків використання для сайту каталогу виробника одягу.

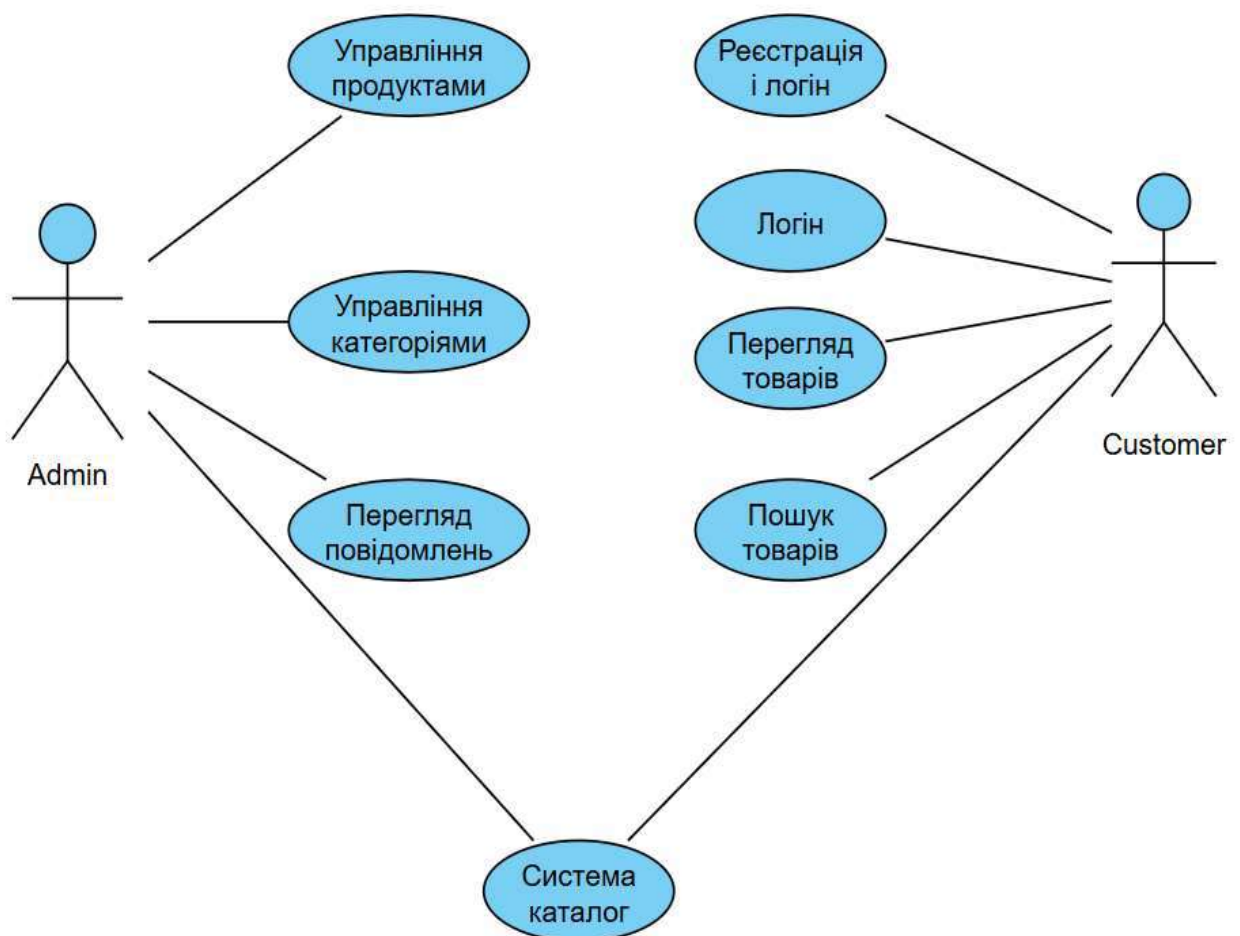


Рис. 2.2. Діаграма випадків використання

На наступному етапі доцільно створити діаграму послідовності прецеденту, яка відображає взаємодію між основними об'єктами системи під час додавання нового продукту адміністратором. Вона демонструє порядок виклику функцій, перевірок та обміну даними між користувачем, системою,

адміністратором та об'єктом продукту. Створена діаграма послідовності прецеденту додавання нового продукту адміністратором представлена на рисунку 2.3.

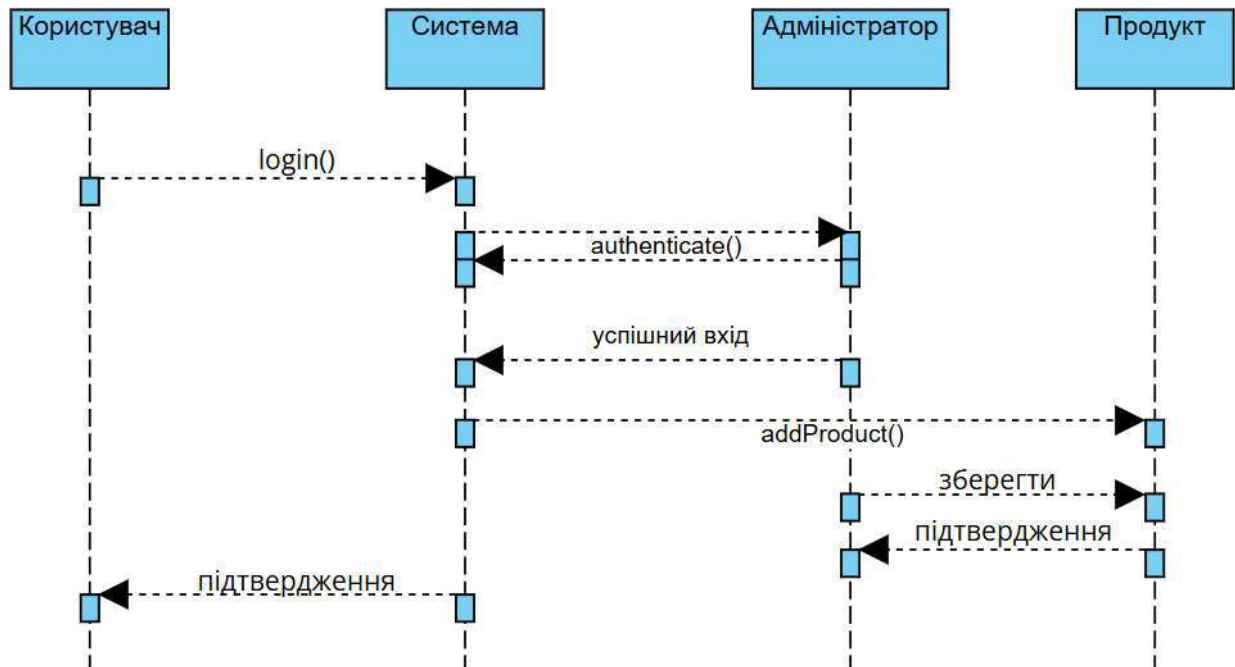


Рис. 2.3. Діаграма послідовності прецеденту додавання нового продукту адміністратором

Представлена діаграма послідовності прецеденту додавання нового продукту адміністратором містить наступне:

Учасники (Actors):

- Користувач - ініціює вхід у систему.
- Система - основна логіка, що керує авторизацією та обробкою дій.
- Адміністратор - має доступ до керування продуктами.
- Продукт - об'єкт, що зберігається в базі після додавання.

Послідовність дій:

1. Користувач надсилає запит login() до системи.
2. Система перевіряє облікові дані, вкликаючи authenticate() для перевірки адміністратора.

3. Після успішної автентифікації, система підтверджує вхід користувача.
4. Адміністратор ініціює дію `addProduct()`, яка надсилається в систему.
5. Система передає дані об'єкта `Product` і відбувається збереження інформації.
6. Після успішного збереження система надсилає підтвердження адміністратору.

Ця діаграма чітко ілюструє покроковий процес, який відбувається під час додавання нового товару до каталогу. Вона демонструє, як взаємодіють різні компоненти системи, що дозволяє краще зрозуміти логіку роботи додатку та важливість кожного об'єкта.

Далі доцільно створити діаграму діяльності, яка є важливим інструментом моделювання і використовується для візуалізації послідовності дій, що відбуваються під час взаємодії користувача з інформаційною системою. У цьому випадку діаграма буде описувати типовий сценарій поведінки користувача при здійсненні покупки в інтернет-магазині (рис. 2.4).

Процес починається з входу користувача на сайт (вказаний як вузол «Старт»), після чого виконується низка послідовних дій:

1. Перегляд каталогу товарів. Користувач знайомиться з продукцією, представленою на сайті. Він може переглядати загальні категорії, новинки, популярні товари або скористатися інструментами пошуку.
2. Пошук товару. Користувач вводить ключові слова у пошукове поле, щоб швидко знайти потрібну річ. Це може бути назва товару, категорія або характеристика (наприклад, «жіноче пальто»).
3. Фільтрація результатів. Після виводу результатів пошуку користувач застосовує фільтри (за ціною, кольором, брендом, розміром), щоб зменшити кількість варіантів і швидше знайти бажаний товар.
4. Вибір продукту. Серед доступних варіантів користувач обирає один конкретний товар для ознайомлення.
5. Перегляд детальної інформації. На сторінці продукту користувач бачить опис, фото, ціну, наявність розмірів, а також може прочитати відгуки.

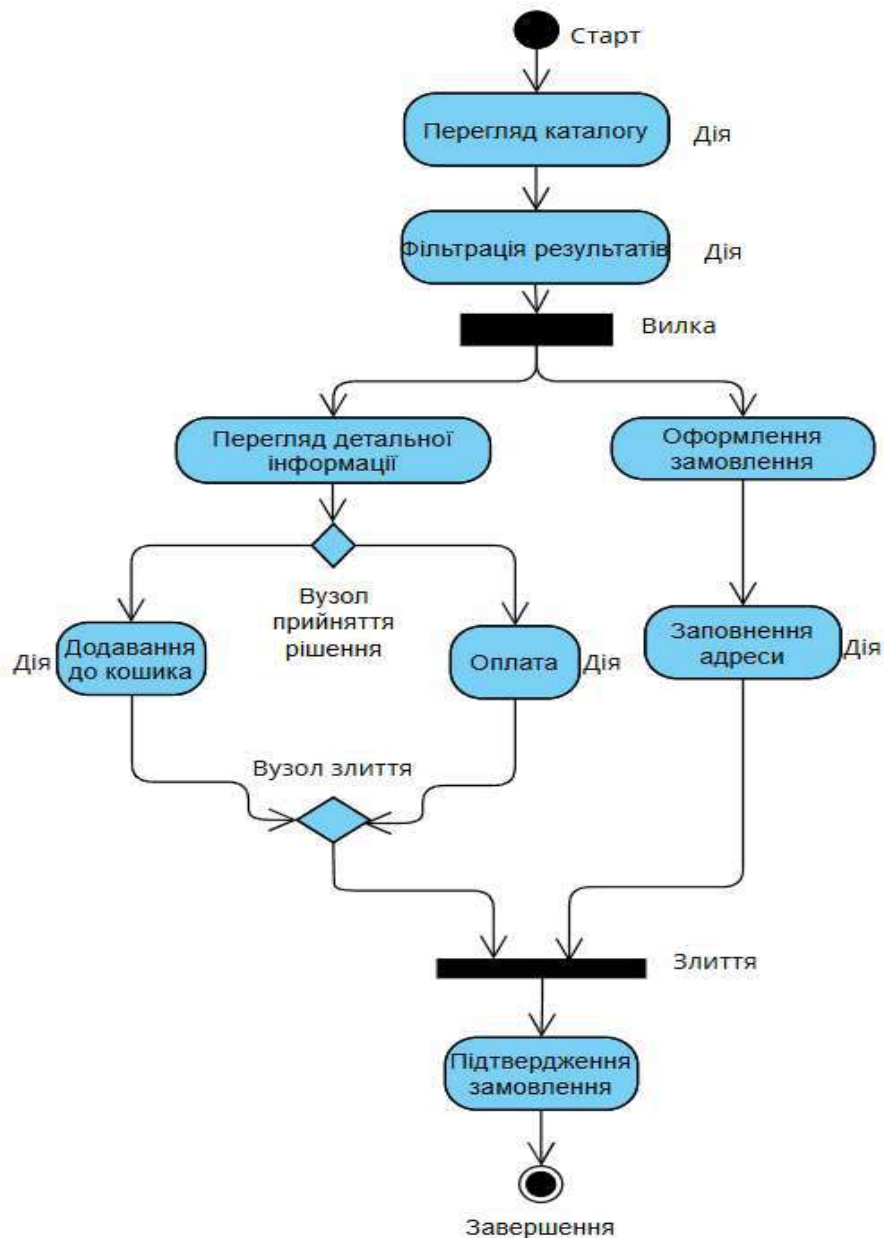


Рис. 2.4. Діаграма діяльності

6. Додавання до кошика. задоволений вибором, користувач додає товар до кошика. У разі потреби можна додати кілька одиниць або декілька різних товарів.

7. Перегляд кошика. Користувач переходить до кошика, щоб переглянути обрані позиції, їх кількість і остаточну вартість. Можна редагувати замовлення або видаляти товари.

8. Оформлення замовлення. Натиснувши на кнопку «Оформити замовлення», користувач переходить до етапу введення персональних даних.

9. Заповнення адреси доставки. Користувач вводить адресу або вибирає відділення поштової служби, куди має бути доставлено замовлення.

10. Оплата. Користувач обирає спосіб оплати - це може бути онлайн-оплата банківською карткою, або післяплата.

11. Підтвердження замовлення. Після успішної оплати система відображає повідомлення про успішне завершення замовлення. Користувач також отримує підтвердження на електронну пошту або в особистому кабінеті.

12. Завершення. На цьому процес покупки завершується, і користувач може або вийти з системи, або повернутись до каталогу.

Ще одним важливим аспектом побудови інформаційної моделі є опис інформаційних потоків - шляхів, якими дані передаються між компонентами системи. Для сайту-каталогу актуальними будуть такі інформаційні потоки:

- Користувач → Сайт: запит на перегляд товарів, використання фільтрів.
- Сайт → База даних: запит на отримання даних про товари певної категорії.
- Адміністратор → Система: введення або редагування товару, додавання зображень, оновлення цін.
- Користувач → Контактна форма: запит на зворотний зв'язок, який зберігається в таблиці БД.

Чітке розуміння цих інформаційних потоків дозволяє ефективно організувати архітектуру взаємодії між модулями застосунку.

Таким чином, створені UML-діаграми досить повно описують інформаційну модель сайту-каталогу виробника одягу, та можуть слугувати якісною основою для подальшої Web-розробки.

2.2 Проєктування бази даних сайту-каталогу

Проєктування бази даних сайту-каталогу доцільно розпочати зі створення ER-діаграми, яка використовується для моделювання логічної структури бази даних. Вона дозволяє візуально представити сутності системи, їх атрибути та

зв'язки між ними. У контексті сайту-каталогу виробника одягу ця діаграма допомагає спроектувати зберігання інформації про користувачів, товари, повідомлення, адміністраторів тощо. Вона буде містити наступні основні сутності та атрибути:

1. Користувач (User)

– Атрибути: id, ім'я, email, пароль.

– Користувач є базовим учасником системи, який може переглядати продукти і надсилати повідомлення.

2. Адміністратор (Admin)

– Атрибути: id, ім'я, email, роль.

– Адміністратор має розширені права, включаючи керування категоріями і продуктами, а також перегляд повідомлень.

3. Продукт (Product)

– Атрибути: id, назва, опис, ціна, розмір, колір, фото.

– Це центральна сутність, що відображає товар, доступний у каталозі.

4. Категорія (Category)

– Атрибути: id, назва.

– Категорії допомагають структурувати товари, наприклад: чоловічий одяг, жіночий одяг, аксесуари тощо.

5. Повідомлення (Message)

– Атрибути: id, відправник, тема, текст, час.

– Повідомлення використовується для зворотного зв'язку користувача з адміністрацією сайту.

Між сутностями у ER-діаграмі сайту-каталогу виробника одягу будуть наступні зв'язки:

1. Користувач - Повідомлення:

– Один користувач може відправити декілька повідомлень.

– Це зв'язок один-до-багатьох: один користувач - багато повідомлень.

2. Адміністратор - Продукт/Категорія:

– Один адміністратор може створювати або редагувати декілька продуктів та категорій.

– Це зв'язок один-до-багатьох.

3. Категорія - Продукт:

– Кожна категорія містить декілька продуктів, але кожен продукт належить одній категорії.

– Це зв'язок один-до-багатьох.

Розроблена ER-діаграма сайту-каталогу виробника одягу представлена на рис. 2.5.

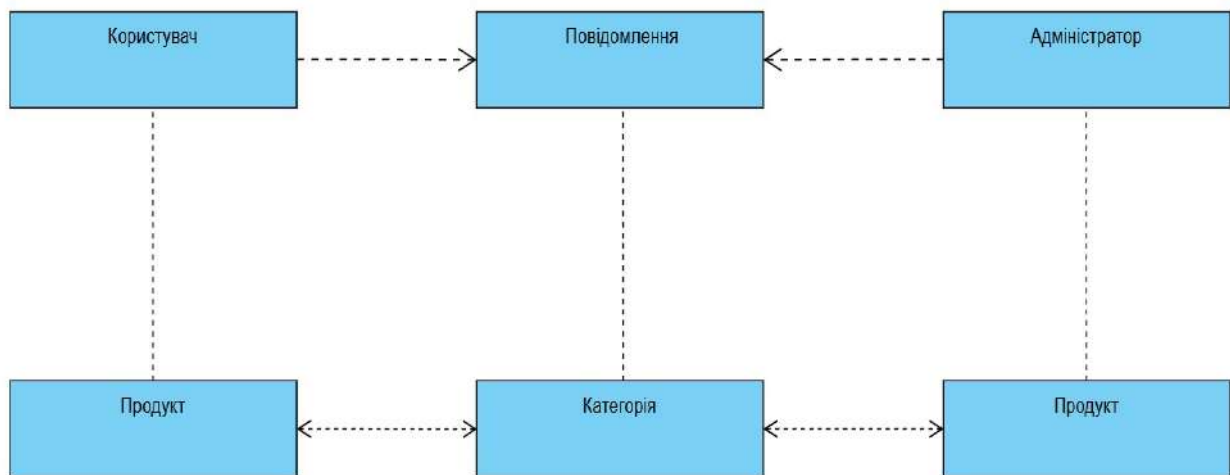


Рис. 2.5. ER-діаграма сайту-каталогу виробника одягу

Представлена ER-діаграма є основою для проектування структури бази даних, наприклад, у реляційній СУБД (такій як MySQL або PostgreSQL). Вона дозволяє легко перейти до побудови таблиць і забезпечує логічну цілісність даних у системі.

На основі аналізу сутностей та зв'язків між ними, на даному етапі, можна описати модель бази даних, яка визначить структуру таблиць, їх поля, типи даних і зв'язки. Зокрема база даних сайту-каталогу виробника одягу буде містити наступні таблиці з відповідними полями:

1. Таблиця products.

Поля: id, name, description, price, size, color, category_id, created_at.

2. Таблиця categories.

Поля: id, name, description.

3. Таблиця users.

Поля: id, name, email, password, role.

4. Таблиця orders.

Поля: id, user_id, status, created_at.

5. Таблиця order_items.

Поля: id, order_id, product_id, quantity.

6. Таблиця product_images.

Поля: id, product_id, image_url.

7. Таблиця contacts.

Поля: id, name, email, message, created_at.

Використання такої структури дозволяє ефективно реалізувати необхідний функціонал сайту-каталогу: відображення продукції, пошук, фільтрацію, керування контентом, обробку звернень користувачів, ведення історії замовлень тощо.

2.3 Огляд необхідних алгоритмічних методів

При розробці сайту-каталогу виробника одягу потенційно існує можливість змістити акцент на клієнтську обробку інформації за допомогою вбудованих інструментів JavaScript та HTML. Такий підхід дозволяє реалізувати ключові можливості системи - пошук, фільтрацію, сортування, валідацію та зберігання даних – зменшивши (або виключивши) використання серверної логіки чи зовнішніх сховищ.

В цьому випадку фільтрація товарів може бути реалізована за допомогою перебору масиву об'єктів, що зберігають інформацію про кожен товар (назва, категорія, розмір, колір, ціна тощо). На основі вибраних користувачем критеріїв (наприклад, категорія "Сукні", колір "Червоний", розмір "М") виконується

логічна перевірка для кожного елемента. Якщо товар відповідає всім умовам, він відображається в результатах пошуку. Такий підхід не створює значного навантаження і ефективний при роботі з невеликими масивами, що цілком підходить для навчального проєкту.

Сортування товарів так само може здійснюється клієнтською стороною за допомогою методів JavaScript. Це дозволяє миттєво змінювати порядок відображення товарів відповідно до вибраних параметрів без оновлення сторінки.

Також можна реалізувати тимчасове зберігання даних у браузері користувача. Завдяки цьому користувач може, наприклад, зберегти вибрані товари в кошику навіть після перезавантаження сторінки.

Ключовою частиною взаємодії користувача із сайтом є форми: реєстрації, входу, оформлення замовлення чи надсилання повідомлень. Для забезпечення правильності введених даних застосовується валідація на клієнтському рівні. Зокрема валідація може бути реалізована як через HTML-атрибути, так і через JavaScript-умови з перевітками довжини полів, формату email, тощо. Це дозволяє значно підвищити зручність і безпеку взаємодії без участі серверної частини.

Форма зворотного зв'язку або звернення користувача до адміністратора також може бути реалізована без використання БД. При цьому повідомлення зберігаються у тимчасовому масиві об'єктів або передаються через сторонній API. У структурі кожного повідомлення зберігається ім'я, email, тема, текст повідомлення та дата відправлення. Надалі повідомлення можуть бути передані на email адміністратора.

Застосування таких алгоритмічних методів дозволяє реалізувати деякі ключові функції сайту-каталогу на клієнтському рівні. Такий підхід підходить для малих комерційних та навчальних проєктів і забезпечує достатню функціональність для базових потреб користувача, суттєво спрощуючи логіку серверної частини та навантаження на неї.

Висновки до розділу 2

Проведений аналіз дозволив виділити ключові об'єкти системи (користувач, адміністратор, продукт, категорія товарів, повідомлення тощо) та описати їхні характеристики, що стало основою для побудови інформаційної моделі та структурування даних.

У межах концептуального проєктування розроблено низку UML-діаграм, які відображають логіку роботи майбутньої системи. Зокрема, діаграма класів деталізує структуру основних сутностей і зв'язків між ними; ER-діаграма окреслює логіку зберігання даних; діаграма послідовності демонструє взаємодію між користувачем і системою в конкретних сценаріях; діаграма діяльності ілюструє основні етапи виконання ключових дій. Ці моделі забезпечують цілісне бачення архітектури й поведінки системи, що є критично важливим для наступного етапу розробки.

Окремо розглянуто алгоритмічні аспекти функціонування додатку, зокрема логіку обробки вхідних і вихідних даних, алгоритми реєстрації, авторизації, пошуку та сортування товарів. Описаний підхід здатен забезпечити достатню функціональність для базових потреб користувача та суттєво спростити логіку серверної частини та навантаження на неї.

Отримані результати забезпечать міцне концептуальне та алгоритмічне підґрунтя для реалізації Web-сайту, оскільки, завдяки побудові моделей і опису функціональних процесів, забезпечено цілісне бачення системи, що дозволяє перейти до безпосередньої розробки та впровадження програмного продукту.

РОЗДІЛ 3 ФУНКЦІОНАЛ ТА ЕЛЕМЕНТИ ІНТЕРФЕЙСУ САЙТУ-КАТАЛОГУ

3.1 Розробка окремих модулів Web-каталогу

Web-каталог складається з кількох основних модулів, які забезпечують основний функціонал сайту: авторизація адміністратора, управління каталогом товарів, робота з кошиком та оформлення замовлення.

3.1.1 Модуль авторизації

Модуль авторизації забезпечує обмеження доступу до адміністративних функцій сайту-каталогу, таких як додавання, редагування та видалення товарів. Доступ до цих функцій дозволено лише авторизованим користувачам, що гарантує базову безпеку та контроль за вмістом каталогу.

Реалізація модуля побудована на основі HTML, CSS та JavaScript. Для збереження стану авторизації використовується механізм localStorage браузера. Перевірка логіна і пароля здійснюється на стороні клієнта. У даному проєкті для тестування використовується фіксована пара логін/пароль: admin / 1234 (рис. 3.1).

Основні функції модуля авторизації:

- Форма авторизації - складається з полів для введення логіна і пароля, а також кнопки підтвердження входу.
- Перевірка облікових даних - введені дані порівнюються з заздалегідь заданими значеннями.
- Збереження сесії - після успішного входу у localStorage зберігається маркер авторизації (loggedIn: true).
- Перенаправлення користувача - після успішного входу користувача перенаправляє на сторінку каталогу (catalog.html).
- Виведення повідомлення про помилку - у разі неправильного введення даних відображається відповідне повідомлення.

```

<!DOCTYPE html>
<html lang="uk">
<head>
  <meta charset="UTF-8" />
  <title>Вхід</title>
  <link rel="stylesheet" href="styles.css" />
</head>
<body>
  <div class="login-container">
    <h2>Авторизація</h2>
    <form id="login-form">
      <input type="text" id="username" placeholder="Логін" required />
      <input type="password" id="password" placeholder="Пароль" required />
      <button type="submit">Увійти</button>
    </form>
    <p id="error-message" style="color: red; display: none;">Невірний логін або пароль</p>
  </div>
  <script>
    const form = document.getElementById('login-form');
    const errorMessage = document.getElementById('error-message');

    form.addEventListener('submit', function (e) {
      e.preventDefault();
      const username = document.getElementById('username').value;
      const password = document.getElementById('password').value;

      if (username === 'admin' && password === '1234') {
        localStorage.setItem('loggedIn', 'true');
        window.location.href = 'catalog.html';
      } else {
        errorMessage.style.display = 'block';
      }
    });
  </script>
</body>
</html>

```

Рис. 3.1. Код модуля авторизації

HTML-форма містить поля для введення логіна і пароля. За допомогою JavaScript обробляється подія натискання кнопки "Увійти". Якщо дані правильні - зберігається статус авторизації та виконується перехід на іншу сторінку. У разі помилки виводиться попередження червоним текстом.

3.1.2 Модуль роботи з каталогом товарів

Для забезпечення зручної навігації по каталогу та швидкого пошуку товарів за різними параметрами на сторінці реалізовано набір елементів фільтрації (рис.3.2). Ці елементи форми дозволяють користувачу вибрати категорію товару, максимальну ціну, колір та розмір для фільтрації каталогу. Кнопка «Застосувати фільтри» активує функцію, яка перебирає всі товари і відображає лише ті, що відповідають обраним критеріям.

```

<label for="category-filter">Категорія:</label>
<select id="category-filter">
  <option value="all">Усі</option>
  <option value="shirts">Футболки</option>
  <option value="pants">Штани</option>
  <option value="jackets">Куртки</option>
  <option value="shoes">Взуття</option>
  <option value="headdres">Головні убори</option>
</select>

<label for="price-filter">Макс. ціна:</label>
<input type="number" id="price-filter" placeholder="Введіть ціну" min="0" />

<label for="color-filter">Колір:</label>
<select id="color-filter">
  <option value="all">Усі</option>
  <option value="red">Червоний</option>
  <option value="blue">Синій</option>
  <option value="green">Зелений</option>
</select>

<label for="size-filter">Розмір:</label>
<select id="size-filter">
  <option value="all">Усі</option>
  <option value="S">S</option>
  <option value="M">M</option>
  <option value="L">L</option>
  <option value="XL">XL</option>
</select>

<br /><br />
<button id="apply-filters">Застосувати фільтри</button>
</div>

```

Рис. 3.2. Код модуля каталогу

Самі товари в каталозі представлені як окремі блоки з класом `product` із набором атрибутів, які містять детальну інформацію про товар (рис. 3.3).

```

<div class="products">
  <!-- Футболка червона -->
  <div
    class="product"
    data-name="Футболка"
    data-category="shirts"
    data-color="red"
    data-size="M"
    data-price="1000"
    data-image="shirts-red.png"
    data-description="Зручна червона футболка з бавовни."
  >
    <a href="product.html?name=Футболка&color=red" class="product-link">
      
      <h3>Футболка</h3>
      <p>Ціна: 1000 грн</p>
    </a>
    <button class="add-to-cart" data-name="Футболка" data-price="1000">
      Додати в кошик
    </button>
  </div>

```

Рис. 3.3 Блок коду, який відповідає за відображення товару

3.1.3 Модуль роботи з кошиком

Кошик забезпечує функціонал вибору товарів користувачем перед оформленням замовлення. Користувач може додавати товари, переглядати список, видаляти позиції, а також бачити загальну суму. Відповідний модуль має наступні функції:

1. Збереження кошика у localStorage.
2. Додавання товарів у кошик.
3. Відображення списку товарів у кошику з їх даними.
4. Видалення товарів із кошика.
5. Підрахунок загальної вартості.

3.1.4 Модуль оформлення замовлення

Модуль оформлення замовлення відповідає за прийом замовлення від користувача: введення контактних даних, вибір способу оплати та перевірку коректності введених даних. Цей модуль має наступні функції:

- Валідація введених даних (ім'я, адреса, телефон).
- Підтвердження успішного оформлення замовлення.
- Очищення кошика після замовлення.

3.1.5 Фільтрація товарів за категорією та ціною

Для зручності користувача на сайті реалізовано функціонал фільтрації товарів. Він дозволяє швидко знайти необхідні позиції за категорією (наприклад, футболки, сукні, штани) та ціною. Це особливо корисно для великих каталогів, де без системи фільтрів користувачеві довелось би витратити більше часу на перегляд усіх товарів. Основна логіка фільтрації реалізована на стороні клієнта (JavaScript). Усі товари зберігаються у масиві, і при зміні параметрів фільтра на

сторінці виконується перебір масиву з перевіркою відповідності критеріям (рис. 3.4).

```
// 2. Клік по кнопці "Застосувати фільтри" – фільтрує товари
document.getElementById('apply-filters').addEventListener('click', function () {
  const category = document.getElementById('category-filter').value;
  const price = document.getElementById('price-filter').value;
  const color = document.getElementById('color-filter').value;
  const size = document.getElementById('size-filter').value;

  const products = document.querySelectorAll('.product');
  products.forEach(product => {
    const productCategory = product.getAttribute('data-category');
    const productPrice = parseFloat(product.getAttribute('data-price'));
    const productColor = product.getAttribute('data-color');
    const productSize = product.getAttribute('data-size');

    const isCategoryValid = category === 'all' || productCategory === category;
    const isPriceValid = price === '' || productPrice <= parseFloat(price);
    const isColorValid = color === 'all' || productColor === color;
    const isSizeValid = size === 'all' || productSize === size;

    if (isCategoryValid && isPriceValid && isColorValid && isSizeValid) {
      product.style.display = 'block';
    } else {
      product.style.display = 'none';
    }
  });
});
```

Рис. 3.4. Код модуля фільтрів

Цей блок коду спрацьовує при натисканні кнопки "Застосувати фільтри". Після цього зчитуються значення фільтрів: категорія, ціна, колір, розмір. Потім усі товари на сторінці перебираються через процедуру `forEach`. Для кожного товару перевіряється, чи відповідає він умовам: якщо вибрано конкретну категорію, ціну, колір або розмір - товар має відповідати цим критеріям.

3.2 Огляд реалізованого функціоналу сайту-каталогу

Сайт-каталог одягу розроблено як односторінковий Web-додаток з підтримкою базових функцій інтернет-магазину, включаючи авторизацію, перегляд товарів, фільтрацію, керування кошиком, редагування товарів та

оформлення замовлення. Зокрема, було реалізовано наступний основний функціонал:

1. Головна сторінка. Після відкриття сайту користувач потрапляє на головну сторінку, яка виконує інформаційну та навігаційну функції. У верхній частині сторінки розміщено логотип компанії та меню навігації, яке дозволяє перейти до розділів сайту. Головна сторінка має приємний зовнішній вигляд, що створює загальне уявлення про стиль компанії та асортимент (рис.3.5).



Рис. 3.5. Головна сторінка сайту-каталогу виробника одягу

Один з основних пунктів меню - "Каталог", що веде до сторінки з переліком товарів. Однак, щоб перейти до каталогу, користувач спочатку повинен пройти авторизацію. Це обмеження реалізовано засобами JavaScript через перевірку змінної у localStorage, яка вказує, чи користувач увійшов до системи. Якщо спробувати перейти до каталогу без авторизації, відкривається відповідне вікно входу.

2. Авторизація. Форма авторизації реалізована у вигляді окремої форми (рис.3.6). Для входу достатньо ввести логін та пароль, що встановлені у коді.

Реєстрація нових користувачів не передбачена, оскільки проект розрахований на демонстрацію базового функціоналу.

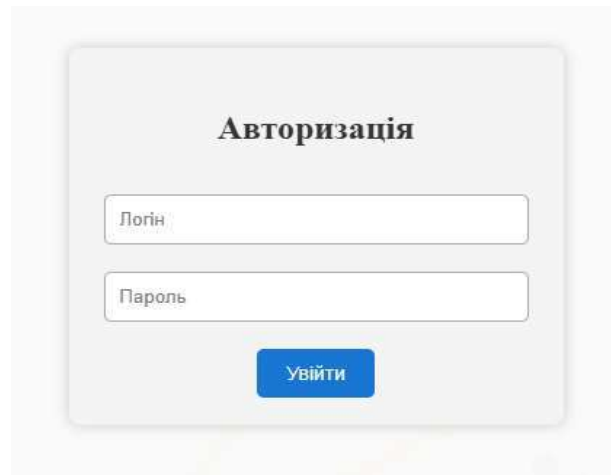


Рис. 3.6 Вікно авторизації

У разі правильного введення даних користувач автоматично перенаправляється на сторінку каталогу. У браузері зберігається маркер входу, який дозволяє отримати доступ до захищених розділів. Це зроблено через JavaScript-функцію, яка перевіряє `localStorage.getItem('loggedIn')` і виконує `window.location.href` у разі успішного входу.

3. Сторінка каталогу товарів. Після авторизації користувач отримує доступ до сторінки з каталогом одягу. Усі товари відображаються у вигляді карток із фото, назвою, ціною та кнопкою "Додати в кошик". Інтерфейс реалізовано з урахуванням адаптивності - товари коректно відображаються як на великих екранах, так і на мобільних пристроях (рис.3.7).

Для адміністраторів або авторизованих користувачів також доступні кнопки "Редагувати" та "Видалити", що дозволяють змінювати або прибирати товари без перезавантаження сторінки.

4. Фільтрація товарів. З метою зручного пошуку необхідного товару реалізовано функцію фільтрації (рис.3.8). Користувач може скористатися кнопкою "Фільтри", щоб відкрити панель з відповідними параметрами. Доступні поля вибору:

- Категорія - дозволяє відфільтрувати товари за типом одягу.
- Колір - обрати лише товари певного кольору.
- Розмір - підтримується вибір кількох популярних розмірів.
- Максимальна ціна - обмеження вартості товару.

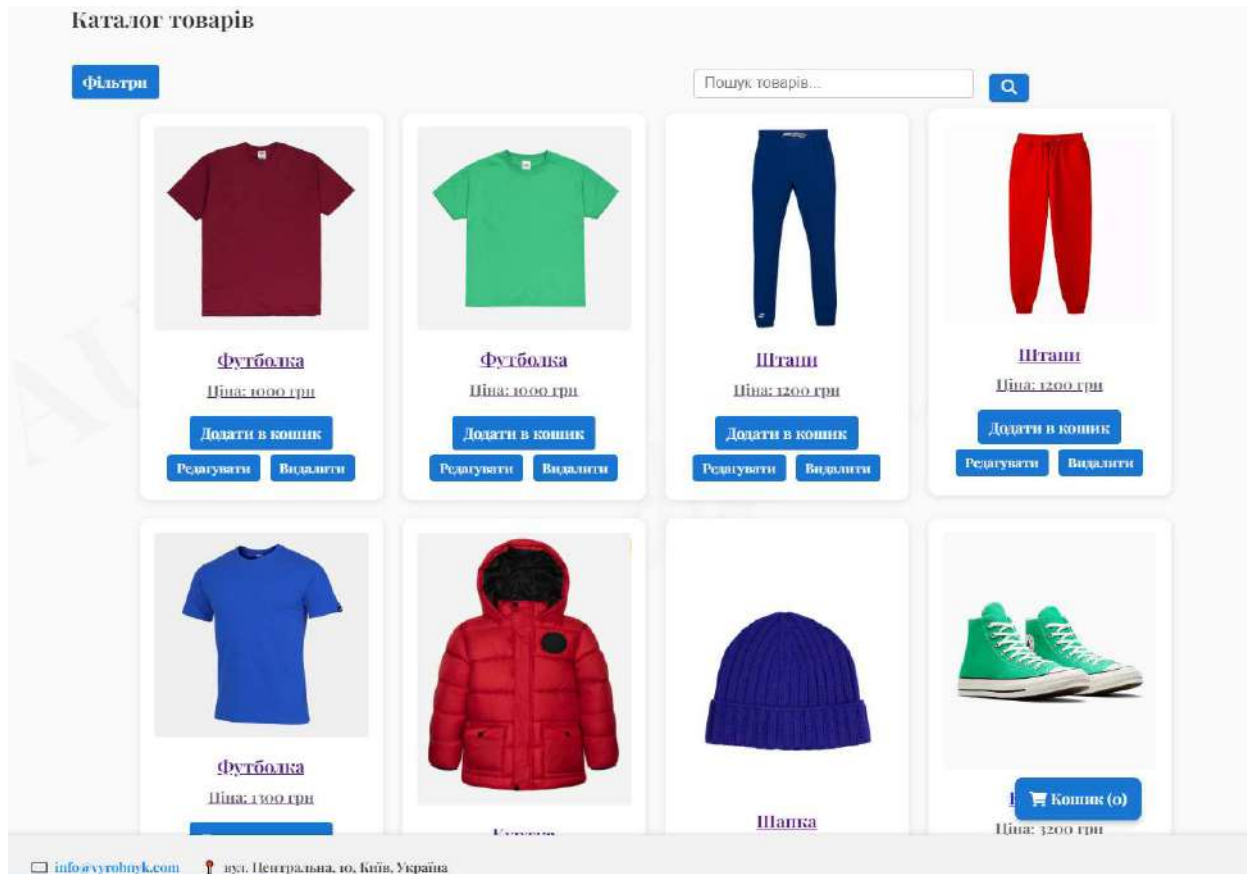


Рис. 3.7 Сторінка каталогу

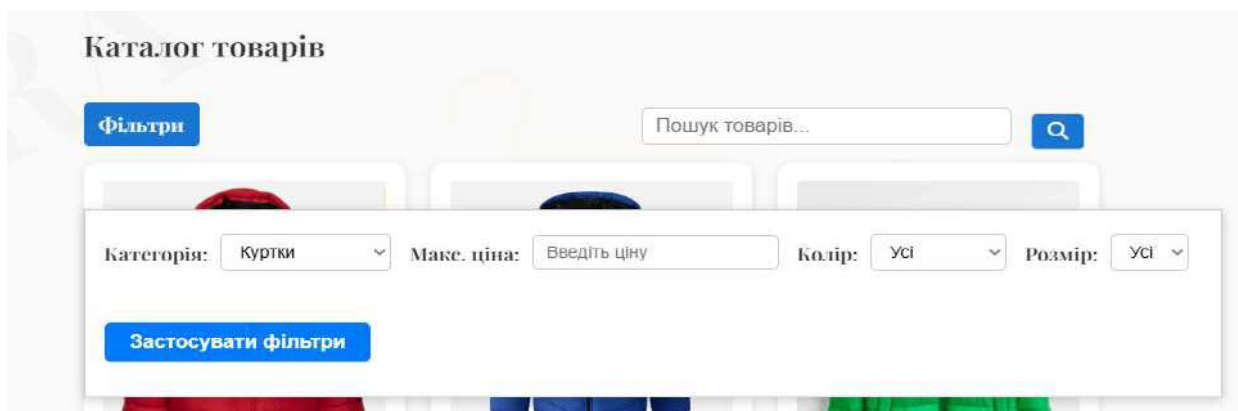


Рис. 3.8. Застосування фільтрів

Фільтрація працює динамічно: після натискання кнопки "Застосувати фільтри" товари на сторінці одразу змінюються відповідно до вибраних параметрів.

5. Кошик. У правому нижньому куті сторінки знаходиться інтерактивна кнопка "Кошик". Натискання на неї відкриває модальне вікно зі списком обраних товарів (рис. 3.9). Для кожного товару в кошику відображаються такі дані:

- Мініатюра зображення
- Назва
- Ціна
- Кнопка для видалення

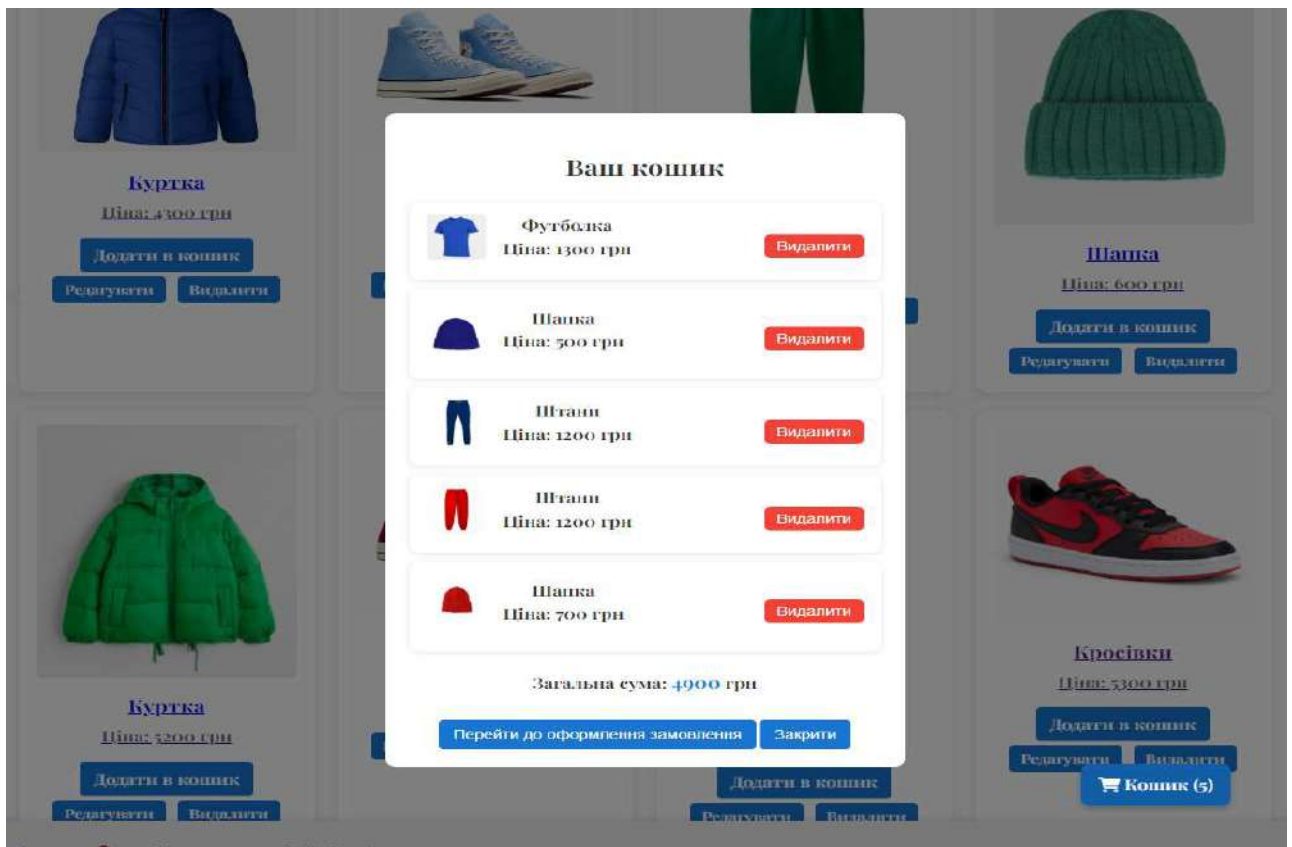


Рис. 3.9. Вікно кошику

У нижній частині модального вікна автоматично підраховується загальна сума замовлення, а також доступна кнопка "Оформити замовлення", яка веде до відповідної сторінки.

6. Оформлення замовлення. Сторінка оформлення замовлення (order.html) містить перелік усіх товарів, які користувач додав у кошик. Тут відображаються фотографії товарів, ціни та загальна вартість. Під основною частиною розташована форма, у якій користувач вводить свої контактні дані - ім'я, адресу доставки та телефон (рис.3.10).

The screenshot displays a shopping cart interface. At the top, there are three items listed in a vertical stack:

- Item 1: A pair of blue pants, labeled "Штани", with a price of "Ціна: 1200 грн" and a red "Видалити" button.
- Item 2: A pair of red pants, labeled "Штани", with a price of "Ціна: 1200 грн" and a red "Видалити" button.
- Item 3: A red hat, labeled "Шапка", with a price of "Ціна: 700 грн" and a red "Видалити" button.

Below the items, the total price is displayed as "Загальна сума: 4900 грн".

The form below the total price contains the following fields:

- "Ім'я:" followed by an empty text input field.
- "Адреса доставки:" followed by an empty text input field.
- "Телефон:" followed by an empty text input field.
- "Метод оплати:" followed by a dropdown menu currently showing "Готівка".
- A blue button at the bottom labeled "Оформити замовлення".

Рис. 3.10. Оформлення замовлення

Цей етап завершує процес покупки, хоча дані не надсилаються на сервер, оскільки сайт на поточному етапі позиціонується як тестовий проєкт без бекенда.

7. Редагування товарів. Редагування товарів можливе лише для авторизованих користувачів. Натискання на кнопку "Редагувати" відкриває модальне вікно, де можна змінити назву товару та його ціну. Після підтвердження зміни відображаються на сторінці без перезавантаження (рис. 3.11).

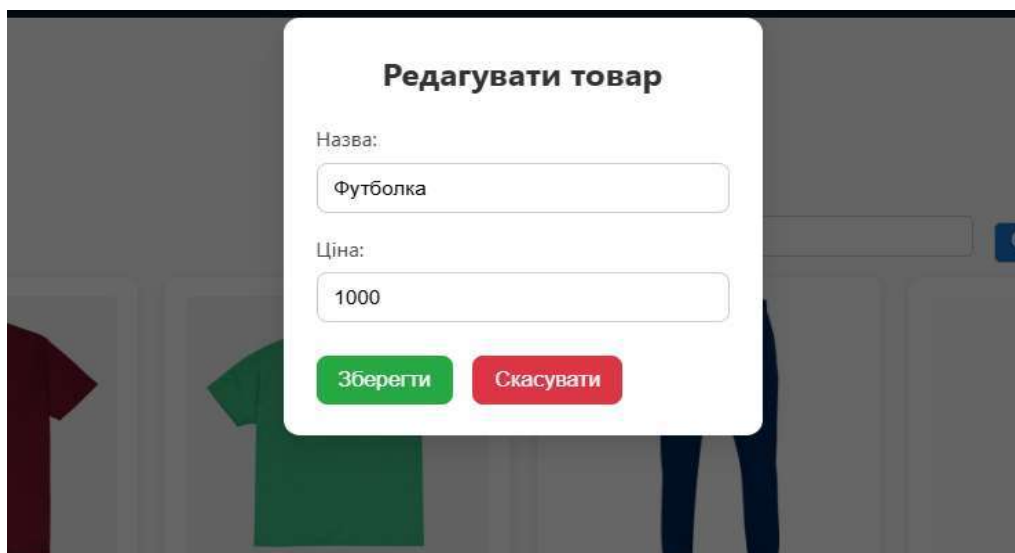


Рис. 3.11. Вікно редагування товару

Це дозволяє легко оновлювати інформацію про товари, що особливо корисно в умовах обмеженого функціоналу без бази даних.

8. Додавання товарів. Додавання нових товарів доступне лише для авторизованих користувачів, які здійснили вхід у систему. Для створення нового товару необхідно натиснути кнопку «Додати товар», яка відкриває модальне вікно з формою введення даних (рис. 3.12).

Форма містить обов'язкові поля для заповнення:

- Назва товару
- Ціна
- Категорія
- Колір
- Розмір
- Опис товару
- Зображення товару (вибір через файловий інпут)

Перед збереженням система перевіряє коректність введених даних: всі поля мають бути заповнені, ціна - числовим значенням, а також обов'язково повинно бути вибрано зображення товару. Якщо якісь дані відсутні або некоректні, користувач отримає відповідне повідомлення із проханням виправити помилки.



Додати новий товар

Назва:
Штани

Ціна:
2300

Категорія:
Штани

Колір:
Зелений

Розмір:
L

Виберіть файл 258614_Gensk...616064508.jpg

Опис:
|

Додати товар Відмінити

Рис. 3.12 Вікно додавання товару

Після натискання кнопки «Зберегти» обране зображення конвертується у формат base64 за допомогою об'єкта FileReader, що дозволяє відобразити картинку без необхідності завантаження на сервер. Потім створюється новий елемент товару, який додається до списку товарів на сторінці (рис. 3.13).

Новий товар містить такі елементи:

- Зображення товару;
- Назва;
- Ціна;
- Кнопка «Додати в кошик».

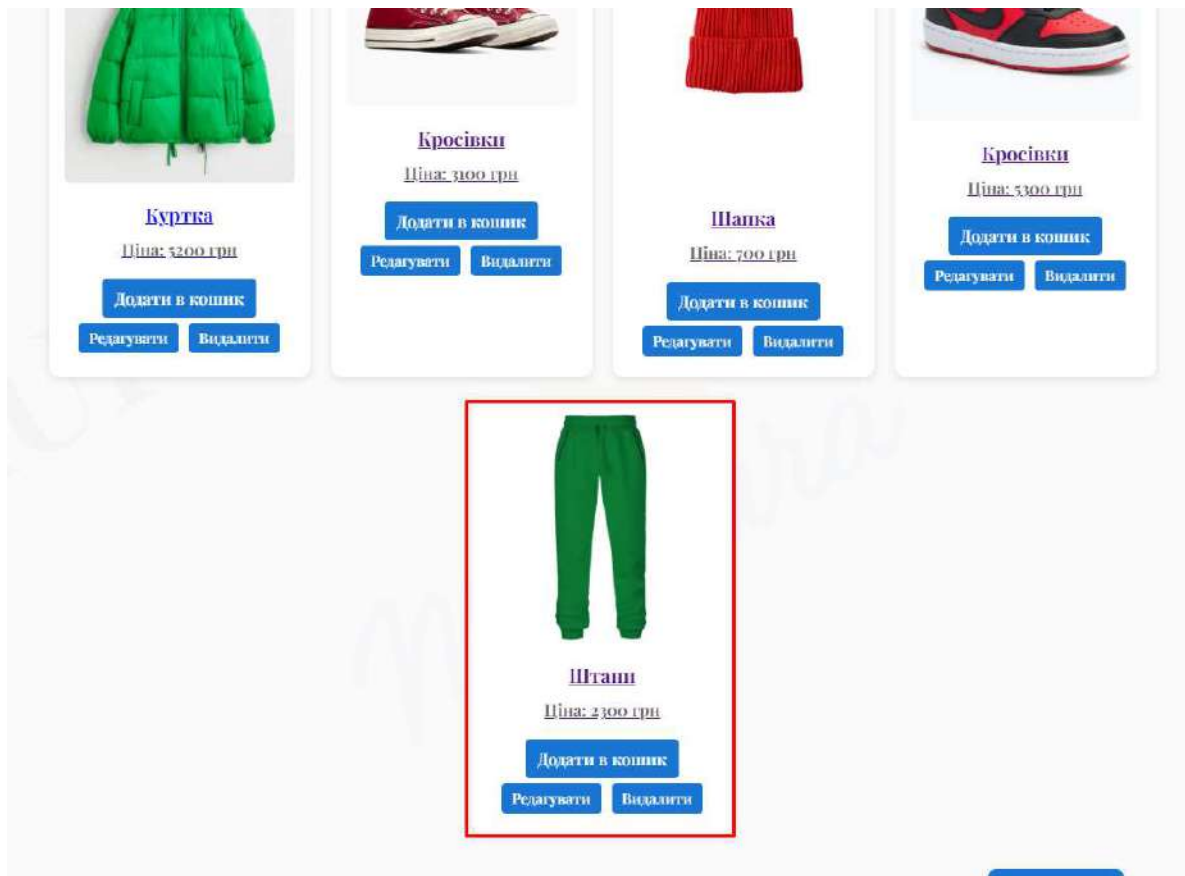


Рис. 3.13 Результат додавання товару

Оразу після додавання товару до DOM йому автоматично додаються кнопки «Редагувати» та «Видалити» з відповідними обробниками подій. Це дозволяє користувачу змінювати чи видаляти товар без перезавантаження сторінки.

Після успішного додавання форми введення автоматично очищується для можливого введення нового товару, а модальне вікно закривається. Таким чином, процес додавання нового товару є швидким, зручним і інтуїтивно зрозумілим.

3.3 Аналіз створеного сайту-каталогу та напрямки його подальшого розвитку

В рамках цієї роботи було створено повноцінний сайт-каталог для виробника одягу, що забезпечує базову взаємодію користувача з продукцією - перегляд товарів, фільтрацію, пошук, додавання до кошика та оформлення замовлення. Сайт відповідає сучасним вимогам до структури Web-ресурсу, має

адаптивний інтерфейс і демонструє практичну реалізацію ключових функцій електронної комерції. Аналізуючи отриманий результат за критеріями функціональності, зручності використання, технічної реалізації та можливостей масштабування, з'ясували наступне:

- Каталог товарів реалізовано у вигляді динамічного блоку, що дозволяє зручно відображати асортимент із використанням карток товарів.
- Фільтрація за категоріями дозволяє користувачеві швидко орієнтуватися у великій кількості товарів, спрощуючи процес пошуку необхідного виробу.
- Пошук за ключовими словами реалізований з використанням JavaScript, що дозволяє моментально фільтрувати результати без перезавантаження сторінки.
- Кошик функціонує через localStorage, що дає змогу користувачеві додавати та видаляти товари з кошика навіть без авторизації.
- Оформлення замовлення реалізується через форму з валідацією введених даних, що дозволяє зменшити кількість помилок при заповненні.
- Застосовано адаптивну верстку, що дозволяє переглядати сайт на мобільних пристроях та планшетах.
- Інтерфейс сайту витримано в єдиному стилі, з використанням читабельних шрифтів, контрастних кольорів і зрозумілої ієрархії елементів.
- Усі ключові функції доступні з головної сторінки або через навігаційне меню, що забезпечує інтуїтивну навігацію.
- Відсутнє реальне серверне збереження даних: усі дані зберігаються на стороні клієнта в localStorage, що не дозволяє зберігати замовлення, товари чи профілі користувачів у базі.
- Функція авторизації реалізована лише умовно - без реєстрації або системи ролей користувачів (існує тільки фіксований логін/пароль для адміністратора).
- Безпека даних на поточному етапі відсутня: дані не захищені від стороннього доступу або зміни через інструменти браузера.

- Відсутня система обробки замовлень (наприклад, надсилання на електронну пошту, формування історії замовлень).
- Сайт не підтримує багатомовність, що може обмежити охоплення потенційної аудиторії.

Визначені наступні можливі напрямки подальшого розвитку проєкту:

1. Реалізація серверної частини. Використання Node.js або іншого бекенд-інструменту для збереження та обробки даних. Інтеграція з MySQL або PostgreSQL для створення повноцінної бази товарів, замовлень і користувачів. Створення API для взаємодії фронтенду з сервером.

2. Поліпшення системи авторизації та безпеки. Реєстрація нових користувачів із можливістю входу та відновлення пароля. Впровадження різних ролей (адміністратор, покупець). Хешування паролів та захист від SQL-ін'єкцій.

3. Розширення функціональності. Система обліку замовлень, повідомлення адміністратору про нові замовлення. Можливість прикріплення кількох зображень до одного товару. Рейтинг товарів, відгуки користувачів. Розділ “Новинки”, “Акції”, “Рекомендовані товари”.

4. Інтеграція з зовнішніми сервісами. Платіжні системи (LiqPay, Wayforpay, PayPal). Підключення до служб доставки (Нова Пошта, Укрпошта) з автоматичним розрахунком вартості доставки. Підключення систем аналітики (Google Analytics, Facebook Pixel).

5. Оптимізація та SEO-просування. Додавання мета-тегів і структурованих даних. Генерація sitemap.xml та robots.txt. Підвищення швидкості завантаження сторінок (оптимізація зображень, використання CDN).

6. Покращення інтерфейсу. Застосування сучасного фреймворку (React, Vue) для полегшення розширення функціональності. Створення адміністративної панелі для керування товарами та замовленнями. Додавання анімацій і мікроваємодій для покращення користувацького досвіду.

Таким чином, розроблений сайт-каталог може бути не лише демонстраційним прикладом для навчального проєкту, а й реальною основою для створення повнофункціонального інтернет-магазину. Застосовані технології та

структура сайту дозволяють легко масштабувати та вдосконалювати проєкт відповідно до нових вимог, що забезпечує його актуальність і гнучкість у подальшому використанні.

Висновки до розділу 3

Здійснено практичну реалізацію проєкту сайту-каталогу для виробника одягу, що охоплює розробку основних функціональних модулів, огляд реалізованого інтерфейсу та оцінку результатів. Робота над сайтом охопила повний цикл користувацької взаємодії - від перегляду асортименту до оформлення замовлення, що дозволяє розглядати проєкт як повноцінний клієнтський прототип онлайн-магазину.

Розглянуто розробку ключових функціональних компонентів. Було реалізовано механізм авторизації, який обмежує доступ до адміністративних функцій. Каталог підтримує динамічне відображення товарів, їх додавання, редагування та видалення без перезавантаження сторінки. Створено зручний інтерфейс кошика з підрахунком загальної вартості та переходом до оформлення. Для полегшення навігації користувача реалізовано пошук та фільтрацію товарів за категоріями.

Зроблений огляд реалізованого функціоналу, включаючи приклади його використання. Кожна функція супроводжується відповідними візуальними прикладами, що ілюструють зручність та логіку користувацької взаємодії. Окрему увагу приділено валідації даних під час оформлення замовлення, що підвищує надійність введення інформації.

Проведено аналіз створеного Web-каталогу. Було виявлено, що запропоноване рішення вдало поєднує простоту реалізації з високою функціональністю. Водночас окреслено перспективні напрямки для подальшого розвитку, серед яких: впровадження серверної частини з базою даних, реалізація повноцінного адміністрування через панель керування, інтеграція платіжних систем, адаптація до мобільних пристроїв з урахуванням специфіки UX-дизайну,

використання сучасних фреймворків (React, Vue) та можливість масштабування проєкту до повноцінного e-commerce рішення.

З технічного боку, сайт створено з використанням HTML, CSS, JavaScript і механізмів збереження даних у localStorage, що дозволило реалізувати повноцінний прототип без серверної частини. Такий підхід є доцільним на початку створення великого масштабованого проєкту, оскільки дозволяє зосередитися на клієнтській логіці та взаємодії користувача з інтерфейсом.

Отримані результати підтверджують ефективність обраного технологічного підходу та відкривають широкі можливості для подальшого розвитку системи як у технічному, так і в комерційному напрямках.

ВИСНОВКИ

У межах даної дипломної роботи розроблено сайт-каталог для виробника одягу, який виконує функції презентації товарів, зручного їх пошуку, формування замовлень та часткового адміністрування без використання серверної частини. Робота охопила всі ключові етапи створення Web-застосунку - від теоретичного обґрунтування потреби в сайті-каталозі до реалізації повністю функціонального клієнтського прототипу, перевірки його працездатності та аналізу можливих напрямків подальшого розвитку.

Підтверджена актуальність створення Web-каталогу саме для сфери виробництва одягу. З'ясовано, що використання лише соціальних мереж як каналу просування та комунікації з клієнтами не забезпечує достатнього рівня зручності, функціональності та масштабованості. Натомість, власний сайт-каталог дає змогу представити повний асортимент товарів, реалізувати детальні описи, структурувати продукцію за категоріями, забезпечити незалежність від сторонніх платформ та створити передумови для подальшої інтеграції з CRM-системами, платіжними інструментами тощо.

Порівняно підходи до побудови Web-додатків - монолітні рішення, SPA, використання CMS та фреймворків. Обґрунтовано вибір технологій, які є оптимальними для навчального проєкту: HTML, CSS та JavaScript для клієнтської частини, без використання серверної логіки чи баз даних. Такий підхід дозволяє зосередитися на використанні ключових навичок фронтенд-розробки, включаючи роботу з DOM, обробку подій, валідацію форм, маніпулювання структурованими даними тощо.

Приділено увагу етапу проєктування інформаційної структури системи. Побудовано UML-діаграми, які відображають сценарії використання, логіку обробки дій користувача, взаємозв'язки між сутностями (товар, користувач, замовлення). Це дозволило сформулювати чітке розуміння архітектури майбутньої системи. Окремо проаналізовано особливості побудови ER-моделі бази даних,

яка може бути використана при розширенні проєкту до повноцінного динамічного Web-застосунку.

Виконано реалізацію клієнтської частини сайту-каталогу. Web-інтерфейс створено з урахуванням принципів юзабіліті, адаптивності та зрозумілої навігації. Основні реалізовані функціональні можливості включають:

- відображення каталогу товарів у вигляді карток із зображеннями, описами, ціною;
- фільтрацію товарів за назвою або категорією;
- авторизацію адміністратора з обмеженням доступу до редагування товарів;
- можливість додавання, редагування та видалення товарів у режимі реального часу;
- роботу кошика з підрахунком загальної суми замовлення;
- оформлення замовлення з валідацією введених даних;
- збереження стану кошика, авторизації та товарів у localStorage.

Детально проілюстровано кожен етап взаємодії користувача з інтерфейсом, що дає змогу легко зрозуміти логіку роботи системи та її функціональність. Продемонстровано використання JavaScript для реалізації взаємодії з DOM-елементами, обробки подій і керування динамічним контентом.

В результаті перевірки виявлено, що всі заплановані функції працюють стабільно, а інтерфейс є зручним і зрозумілим для користувача. Тестування охопило як позитивні, так і негативні сценарії: введення коректних та некоректних даних, спроби доступу до закритих функцій без авторизації, очищення кошика, зміна кількості товарів тощо. Завдяки використанню localStorage забезпечено збереження даних між сесіями, що підвищує зручність користування.

До переваг створеного сайту-каталогу можна віднести просту реалізацію, повну автономність роботи без серверної частини, інтуїтивний інтерфейс і легкість у розгортанні. Обмеженнями є відсутність реальної бази даних,

відсутність захисту даних та неможливість реалізувати багатокористувацький режим. На основі цього визначено наступні перспективні напрямки розвитку:

- впровадження backend-частини на основі PHP або Node.js;
- збереження даних у реальній базі (MySQL, MongoDB);
- створення панелі адміністратора з авторизованим доступом;
- реалізація реєстрації користувачів і перегляду історії замовлень;
- підключення платіжних та логістичних систем;
- SEO-оптимізація сайту та адаптація до мобільних пристроїв;
- використання сучасних JavaScript-фреймворків для підвищення масштабованості.

Таким чином, результати роботи підтверджують досягнення її основної мети - створення функціонального сайту-каталогу одягу. Робота дозволила на практиці реалізувати ключові принципи Web-розробки, логіку побудови інтерфейсу користувача, структурування коду, проектування Web-систем та перевірку їх функціональності. Розроблений сайт може бути використаний як база для подальшого вдосконалення та комерційного використання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Братко А. В. Основи веб-дизайну : навч. посіб. Київ : КНЕУ, 2019. 230 с.
2. Горбенко І. В. Web-програмування : підручник. Харків : ХНУРЕ, 2021. 356 с.
3. Дяченко О. І. HTML та CSS. Створення та оформлення сайтів. Київ : Ліра-К, 2020. 192 с.
4. Крамаренко Т. Г. Основи розробки сайтів : навч. посіб. Київ : Каравела, 2021. 210 с.
5. Флэнаган Д. JavaScript. Подробное руководство. 6-е изд. Санкт-Петербург : Питер, 2021. 976 с.
6. Заяць С. В., Бондар В. М. Інтернет-технології : навч. посіб. Львів : ЛНУ, 2018. 288 с.
7. Миколюк О. В. Основи програмування на JavaScript. Івано-Франківськ : Фоліант, 2022. 180 с.
8. Кришталь С. П., Кушнір С. В. Веб-технології : навч. посіб. Київ : Академія, 2020. 252 с.
9. Сірик Я. Б. Основи проектування баз даних : навч. посіб. Тернопіль : ТНТУ, 2019. 144 с.
10. Костенко А. В. MySQL для початківців. Харків : Ранок, 2020. 160 с.
11. Марченко О. М. PHP для розробників. Київ : ДП "Інформатика", 2021. 248 с.
12. Сімонов С. В. Створення динамічних сайтів : навч. посіб. Дніпро : ДНУ, 2019. 198 с.
13. Кучерук Ю. В. Проектування веб-систем : монографія. Львів : ЛНУ, 2020. 312 с.
14. Bootstrap 5 Documentation. URL: <https://getbootstrap.com/docs/5.3/getting-started/introduction/> (дата звернення: 25.05.2025).

15. Mozilla Developer Network. HTML Reference. URL: (<https://developer.mozilla.org/en-US/docs/Web/HTML>) (дата звернення: 25.05.2025).
16. Mozilla Developer Network. CSS Reference. URL: (<https://developer.mozilla.org/en-US/docs/Web/CSS>) (дата звернення: 25.05.2025).
17. Mozilla Developer Network. JavaScript Reference. (developer.mozilla.org/en-US/docs/Web/JavaScript) (дата звернення: 25.05.2025).
18. W3Schools. HTML, CSS, JavaScript Tutorials. URL: (<https://www.w3schools.com/>) (дата звернення: 25.05.2025).
19. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання. Київ : Держспоживстандарт України, 2015. 16 с.
20. ISO/IEC 25010:2011. Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models. Geneva : ISO, 2011.
21. Nielsen J. Designing Web Usability. Indianapolis : New Riders, 2000. 432 p.
22. Krug S. Don't Make Me Think. Revisited. San Francisco : New Riders, 2014. 216 p.
23. Shopify. E-commerce UX Best Practices. URL: (<https://www.shopify.com/partners/blog/ux-best-practices>) (дата звернення: 25.05.2025).
24. WooCommerce Docs. URL: <https://woocommerce.com/documentation/> (дата звернення: 25.05.2025).
25. Magento DevDocs. URL: (<https://developer.adobe.com/commerce/>) (дата звернення: 25.05.2025).
26. WordPress Codex. URL: (<https://codex.wordpress.org/>) (дата звернення: 25.05.2025).
27. Євдокимов В. В. Аналіз інтерфейсів користувача в системах електронної комерції. Економіка і суспільство. 2022. № 40. С. 115–120.

28. Тимошенко Л. О., Гнатенко О. С. UX-дизайн як інструмент підвищення ефективності вебсайтів. Вісник Київського нац. ун-ту технологій та дизайну. 2021. № 3. С. 78–84.

29. Про захист персональних даних : Закон України від 01.06.2010 № 2297-VI. Відомості Верховної Ради України. 2010. № 34. Ст. 481.

30. Конституція України : офіц. текст. Київ : Парлам. вид-во, 2019. 92 с.

31. Про електронну комерцію : Закон України від 03.09.2015 № 675-VIII. Відомості Верховної Ради України. 2015. № 45. Ст. 410.

32. Google Developers. Web Vitals. URL: <https://web.dev/vitals/>) (дата звернення: 25.05.2025).

33. Lighthouse Performance Tool. URL: <https://developer.chrome.com/docs/lighthouse/>) (дата звернення: 25.05.2025).

34. Романенко О. П. Маркетинг у цифровому середовищі. Київ : КНЕУ, 2021. 216 с.

35. Гречанюк А. О. Основи розробки клієнт-серверних веб-додатків. Київ : НАУ, 2022. 184 с.

ЗГОДА здобувача(чки) вищої освіти
Державного університету економіки і технологій про
перевірку кваліфікаційної роботи на прояви
академічного плагіату
та розміщення в Репозитарії Університету

Я, Тимощев Денис Олександрович (ПШ),
підтримую політику Державного університету економіки і технологій
з академічної доброчесності і відкритого доступу.

Засвідчую, що кваліфікаційна бакалаврська (магістерська)
робота

Розробка сайту-каталогу виробника одягу

(назва роботи повністю) виконана самостійно та не містить академічного плагіату. Я не надавав(ла) і не одержував(ла) недозволену допомогу під час підготовки цієї роботи. Робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

Із чинним Положенням про запобігання та виявлення академічного плагіату в роботах здобувачів вищої освіти Державного університету економіки і технологій ознайомлений(а). Чітко усвідомлюю, що в разі виявлення у кваліфікаційній роботі порушення норм академічної доброчесності робота не допускається до захисту або оцінюється незадовільно.

Також я поінформований(на), що відповідно до «Положення про Репозитарій (електронну базу даних) Державного університету економіки і технологій» зазначена робота буде розміщена в Електронному архіві Університету (Репозитарії ДУЕТ). З умовами такого розміщення ознайомлений(на).

10.06.2025

Дата



підпис

Тимощев Д. О.

ініціали, прізвище (власноруч)