

ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«УНІВЕРСИТЕТ ЕКОНОМІКИ ТА ПРАВА «КРОК»»

КВАЛІФІКАЦІЙНА РОБОТА

Тема: «Вебсайт інтернет-магазину для продажу домашнього одягу з використанням багатокритеріальних фільтрів та рекомендаційної системи»

Ступінь вищої освіти – бакалавр
Спеціальність – 122 «Комп'ютерні науки»
Освітня програма «Комп'ютерні науки»

ПОЯСНЮВАЛЬНА ЗАПИСКА

Виконав: здобувача 4 курсу
групи КН-21

Карагужев Ілля

Керівник: старший викладач кафедри
комп'ютерних наук

Олег ЛУКУТІН

Засвідчую, що кваліфікаційна
робота оформлена відповідно
до ДСТУ 3008:2015 та не
містить запозичень з праць
інших авторів без відповідних
посилань.

Здобувач: _____

(підпис)

м. Київ – 2025 рік

ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«УНІВЕРСИТЕТ ЕКОНОМІКИ ТА ПРАВА «КРОК»»

ЗАТВЕРДЖУЮ:

завідувач кафедри комп'ютерних
наук

_____Сергій МІЧКІВСЬКИЙ

« ____ » ____ 20 ____ р

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Карагужев Ілля Сергійович

Тема роботи	Вебсайт інтернет-магазину для продажу домашнього одягу з використанням багатокритеріальних фільтрів та рекомендаційної системи
Номер та дата наказу про затвердження теми	№121-7 від 24 грудня 2024 року
Коротка постановка завдання	Розробити вебсайт інтернет магазину для продажу домашнього одягу, що включає каталог товарів, кошик, систему замовлення, реєстрацію користувачів, багатокритеріальних фільтрів та рекомендаційної системи
Посилання на джерела інформації (не більше п'яти найменувань, які рекомендує науковий керівник)	1. E-commerce UX: Best Practices for Online Stores. Доступно за URL: https://www.nngroup.com/articles/ecommerce-ux/ (дата звернення 13.03.2025) 2. Web Development with React and Node.js. Доступно за URL: https://developer.mozilla.org/en-US/docs/Learn/Server-side/Node_server_without_framework (дата звернення 13.03.2025)
Вимоги до кваліфікаційної роботи	Кваліфікаційна робота має передбачати створення повнофункціонального веб-застосунку, який включає адаптивний дизайн, зручний користувацький інтерфейс, безпечну обробку платежів, а також адміністративну панель для керування товарами та замовленнями. Робота повинна продемонструвати застосування сучасних технологій веб-розробки та відповідати вимогам захисту даних і безпеки користувачів.

Дата видачі завдання 27 грудня 2024 р.

Керівник

Олег ЛУКУТІН

Здобувач освітнього ступеня бакалавра

Ілля КАРАГУЖЕВ

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання	Примітка
Підготовчий етап			
1	Вибір напрямку дослідження	02.12.2024 р.	<i>виконано</i>
2	Формування теми та призначення керівника	16.12.2024 р.	<i>виконано</i>
3	Затвердження теми кваліфікаційної роботи	23.12.2024 р.	<i>виконано</i>
4	Затвердження завдання на кваліфікаційну роботу	27.12.2024 р.	<i>виконано</i>
Основний етап			
5	Розробка концепції кваліфікаційної роботи	13.01.2025 р.	<i>виконано</i>
6	Підбір та вивчення джерел інформації з напрямку дослідження. Огляд існуючих аналогів	20.01.2025 р.	<i>виконано</i>
7	Затвердження розширеної постановки завдання. Підготовка та подання керівникові розділу 1 кваліфікаційної роботи	10.03.2025 р.	<i>виконано</i>
8	Проектування. Підготовка та подання керівникові розділу 2 кваліфікаційної роботи	24.03.2025 р.	<i>виконано</i>
9	Підготовка доповіді для експертизи стану виконання кваліфікаційної роботи (проміжний контроль)	31.03-04.04.2025 р.	<i>виконано</i>
10	Реалізація. Підготовка та подання керівникові розділу 3 кваліфікаційної роботи	07.04.2025 р.	<i>виконано</i>
11	Підготовка та подання керівнику першого варіанту всієї кваліфікаційної роботи	14.04.2025 р.	<i>виконано</i>
12	Доопрацювання кваліфікаційної роботи з урахуванням зауважень керівника та представлення керівникові доопрацьованого варіанту кваліфікаційної роботи	21.04.2025 р.	<i>виконано</i>
Завершальний етап			
13	Представлення рукопису для перевірки на плагіат	28.04-04.05.2025 р.	<i>виконано</i>
14	Підготовка презентації та доповіді на передзахист	05.05-11.05.2025 р.	<i>виконано</i>
15	Передзахист кваліфікаційної роботи	12.05-16.05.2025 р.	<i>виконано</i>
16	Доопрацювання роботи за результатами передзахисту	19.05-06.06.2025 р.	<i>виконано</i>
17	Експертиза роботи керівником та зовнішнім експертом	09.06-15.06.2025 р.	<i>виконано</i>
18	Доопрацювання доповіді та презентації для захисту	09.06-15.06.2025 р.	<i>виконано</i>
19	Захист кваліфікаційної роботи	16.06-22.06.2025 р.	<i>виконано</i>

Керівник

Олег ЛУКУТІН

Здобувач освітнього ступеня бакалавра

Ілля КАРАГУЖЕВ

Карагузев І.С. «Вебсайт інтернет-магазину для продажу домашнього одягу з використанням багатокритеріальних фільтрів та рекомендаційної системи»

Пояснювальна записка кваліфікаційної роботи за спеціальністю 122 – Комп’ютерні науки (освітня програма – Комп’ютерні науки) СО Бакалавр. – ВНЗ «Університет економіки та права «КРОК», навчально-науковий інститут інформаційних та комунікаційних технологій, кафедра комп’ютерних наук, Київ, 2025.

У даній роботі розглянуто актуальні проблеми електронної комерції на прикладі розробки інтернет-магазину домашнього одягу. Реалізовано вебсайт для електронної комерції із застосуванням сучасних технологій, що включає систему управління замовленнями, каталог продукції, інтеграцію з платіжними системами та систему рекомендацій для користувачів.

Ключові слова: інтернет-магазин, електронна комерція, управління замовленнями, адаптивний інтерфейс, каталог продукції.

Рис. 22. Бібліограф.: 9 найм.

Karaguzhev I.S. «Website for an online store selling homewear, using multi-criteria filters and a recommendation system»

Explanatory note of the qualification work in the specialty 122 – Computer Science (educational program – Computer Science) SO Bachelor. – Higher Educational Institution “University of Economics and Law “KROK”, Educational and Scientific Institute of Information and Communication Technologies, Department of Computer Science, Kyiv, 2025.

This work considers the current problems of e-commerce using the example of developing an online store for home clothing. The web application for e-commerce was implemented using modern technologies, including an order management system, product catalog, payment system integration, and a recommendation system for users.

Keywords: online store, e-commerce, order management, adaptive interface, product catalog.

Fig. 22. Bibliography: 11 Items.

ЗМІСТ

ВСТУП.....	6
РОЗДІЛ 1 ПОСТАНОВКА ЗАВДАННЯ НА РОЗРОБКУ ПРОГРАМНОГО ПРОДУКТУ	8
1.1 ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ	8
1.2 ОГЛЯД АНАЛОГІВ.....	9
1.3 ПОСТАНОВКА ЗАДАЧІ	12
Висновки до розділу 1.....	13
РОЗДІЛ 2 ПРОЄКТУВАННЯ ВЕБСАЙТУ ІНТЕРНЕТ-МАГАЗИНУ	15
2.1 ПРОЄКТУВАННЯ СТРУКТУРИ ІНТЕРНЕТ-МАГАЗИНУ «ЗАТИШОК & СТИЛЬ» ...	15
2.2 МОДЕЛЮВАННЯ ПРОЦЕСІВ	18
2.3 СТВОРЕННЯ ПРОТОТИПІВ	19
2.4 ПРОЄКТУВАННЯ ДИЗАЙНУ ТА ІНТЕРФЕЙСУ	21
Висновок до розділу 2.....	23
РОЗДІЛ 3 РЕАЛІЗАЦІЯ ВЕБСАЙТУ ІНТЕРНЕТ-МАГАЗИНУ	25
3.1 ВИБІР ТЕХНОЛОГІЙ ДЛЯ СТВОРЕННЯ ВЕБРЕСУРСУ	25
3.2 СТВОРЕННЯ СТРУКТУРИ БАЗИ ДАНИХ.....	27
3.3 РЕАЛІЗАЦІЯ ФІЛЬТРІВ ТА КОРЗИНИ	27
3.4 ТЕСТУВАННЯ РЕАЛІЗОВАНОЇ СИСТЕМИ	32
ВИСНОВКИ	39
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	40

ВСТУП

Актуальність теми. Електронна комерція в сучасних умовах стрімко розвивається і стає невід’ємною частиною глобальної економіки. Онлайн-продаж товарів та послуг перетворився на динамічну галузь, де успіх залежить від упровадження інноваційних інструментів управління, аналітики даних і зручних для користувача веб-рішень. Зокрема, сфера продажу домашнього одягу в інтернеті останніми роками демонструє стабільне зростання, адже все більше людей прагнуть забезпечити собі комфорт, не витрачаючи час на офлайн-шопінг. Таким чином, тема розробки інтернет-магазину домашнього одягу є особливо актуальною з огляду на зростаючу популярність онлайн-покупок та необхідність модернізації механізмів взаємодії з клієнтами.

Мета та завдання дослідження. Метою роботи є розробка та впровадження ефективного інтернет-магазину домашнього одягу, який демонструватиме інноваційні підходи до систем пошуку, персоналізації та управління асортиментом. Для досягнення цієї мети поставлено такі завдання:

- дослідити сучасні методи розробки веб-застосунків із акцентом на пошукових алгоритмах та персоналізації;
- розробити зручну структуру каталогу товарів з інструментами швидкого пошуку та фільтрації;
- реалізувати інтегровану систему управління товарними ресурсами з урахуванням наявних технологій автоматизації складу;
- створити адаптивні механізми рекомендацій на основі аналізу поведінки користувачів (big data, машинне навчання тощо);
- налаштувати безпечну та зручну систему замовлень, платежів і доставки.

Об’єкт та предмет дослідження. Об’єктом дослідження є процес створення інтернет-магазину домашнього одягу, а предметом – застосування сучасних пошукових алгоритмів та інструментів управління товарними ресурсами для досягнення максимальної зручності та ефективності взаємодії з користувачами.

Методологічна основа дослідження. У роботі використовуються методи алгоритмічного пошуку, патерни проєктування веб-застосунків, а також хмарні технології для зберігання та обробки даних. Значну увагу приділено дослідженню алгоритмів оптимізації пошуку (зокрема, пошук із урахуванням синонімів, морфологічного аналізу) та системам управління товарними запасами (інвентаризація, налаштування складу, відстеження логістики). **Практичне значення роботи.**

Розробка інтернет-магазину домашнього одягу демонструє можливості застосування інноваційних технологій у сфері електронної комерції та може слугувати базовою платформою для подальших досліджень у галузі оптимізації пошуку, персоналізованих рекомендацій, а також побудови ефективних моделей взаємодії з клієнтами.

Структура роботи. Кваліфікаційна робота складається зі вступу, трьох розділів, висновків та списку посилань (9 найменувань). Пояснювальна записка містить 21 рисунок, що демонструють концепцію архітектури веб-додатка, схеми бази даних і приклади інтерфейсу користувача. Загальний обсяг пояснювальної записки становить 40 сторінок, основний зміст викладено на 39 сторінках.

РОЗДІЛ 1

ПОСТАНОВКА ЗАВДАННЯ НА РОЗРОБКУ ПРОГРАМНОГО ПРОДУКТУ

1.1 Опис предметної області

За даними досліджень глобальних маркетингових агентств, ринок електронної комерції невинно зростає (рис. 1.1). Особливо це помітно у сегменті одягу та аксесуарів для дому, де покупці дедалі частіше роблять вибір на користь онлайн-покупок.

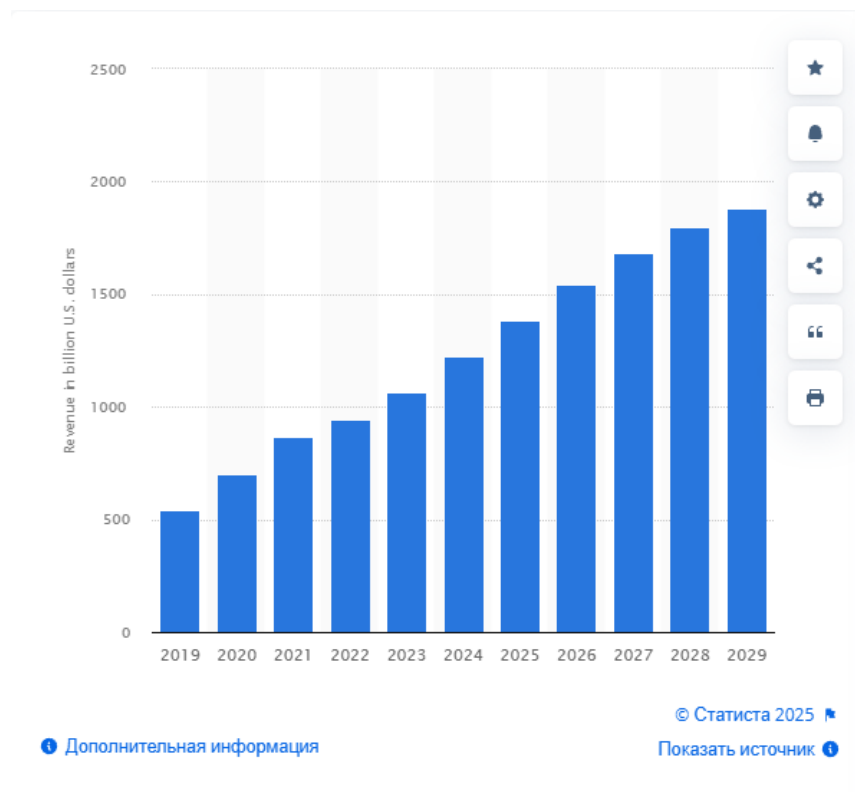


Рисунок 1.1 – Зростання обсягів продажу в сегменті домашнього одягу (онлайн) у світі, 2018–2025 рр.

Джерело: [1]

Головними перевагами інтернет-магазину в порівнянні з офлайн-форматом є зручність доступу, широкий вибір товарів та можливість швидко порівняти пропозиції. У випадку з домашнім одягом особливо важливу роль відіграють такі аспекти, як комфорт, якість тканин та розмірні таблиці. Тому

завданням даного проєкту є створення платформи, яка надаватиме користувачам необхідну інформацію в зручному та інтуїтивному форматі, а також пропонуватиме персоналізовані рекомендації на основі їхніх вподобань та історії пошуку.

1.2 Огляд аналогів

Перш ніж розпочати розробку інтернет-магазину, було здійснено аналіз кількох популярних платформ, які успішно працюють на ринку домашнього одягу в онлайн-сегменті:

H&M Home (рис. 1.2) – це підрозділ H&M, що пропонує стильні та доступні товари для дому: меблі, текстиль, декор тощо. Продукцію можна придбати онлайн на офіційному сайті. H&M Home також випускає колаборації з дизайнерами, наприклад, нещодавню колекцію з Соqі Соqі.



Рисунок 1.2 - H&M

Джерело: [2]

Позитивні аспекти: зручний інтерфейс, наявність фільтрів за кольором, розміром, матеріалом; швидке оформлення замовлення.

Недоліки: інколи відсутня локалізація певних артикулів, обмежена комунікація з клієнтом після покупки.

Oysho (Homewear) – це категорія домашнього одягу бренду Oysho, що входить до групи Inditex (рис. 1.3). Включає піжами, халати, нічні сорочки та комфортний одяг для дому. Virізняється стильним дизайном, натуральними матеріалами та м'якими кольорами [3].



Рисунок 1.3 – Oysho

Джерело: [3]

Позитивні аспекти: стильний дизайн платформи, інтегровані lookbook-галереї, що дають змогу оцінити товари в реальному застосуванні.

Недоліки: недостатньо гнучка система рекомендацій, відсутність подробиць про склад тканини у швидкому перегляді товару.

Marks & Spencer (Loungewear) – це колекція зручного одягу для дому та відпочинку. Включає піжами, костюми, халати та м'який трикотаж. Відзначається якісними тканинами, класичним дизайном і комфортним кроєм. Доступний онлайн та в магазинах M&S.



Рисунок 1.4 - Marks & Spencer

Джерело: [4]

Позитивні аспекти: добре структурований каталог, наявність відгуків покупців, деталізовані описи товарів.

Недоліки: застарілий пошуковий механізм, інколи складно знайти конкретну модель через нечітку систему фільтрації.

Узагальнюючи, можна виділити такі загальні переваги розглянутих аналогів:

- зручне сортування товарів;
- привабливий дизайн та маркетингові акції;
- інтеграція швидких платежів та різних платіжних систем.

Водночас недоліками залишаються:

- недостатня персоналізація результатів пошуку;
- обмежена адаптивність під індивідуальні розмірні таблиці;
- відсутність гнучких рекомендаційних сервісів (machine learning).

Особливості нашого інтернет-магазину:

1. адаптивна система пошуку з можливістю морфологічного аналізу, що дасть змогу більш точно відображати потрібні товари;
2. інтеграція рекомендаційного модуля, який аналізує вподобання користувачів, історію покупок та пошукові запити, пропонуючи релевантні товари;

3. автоматизована система управління складом, що дає змогу оперативно відстежувати залишки, формувати замовлення та сповіщати клієнтів про наявність товару.

	Створення власної вітрини	Налагодження роботи з великою кількістю онлайн-партнерів	Вихід на 1-2 ТОП-магазини	Рішення для пакетного виходу на сотні майданчиків
Наявність єдиної точки для реклами	Є	Немає	Є	Є
Операційні витрати на роботу з каналом	Високі	Високі	Середні	Низькі
Наявність конкурентів у точках онлайн-продажів	Немає	Майже немає	Можливі	Майже немає
Швидкість початку роботи з каналом	Низька	Низька	Середня	Висока
Можливості з охоплення онлайн-ринку	Низькі	Високі	Вище середнього	Високі
Хто обробляє замовлення та продажі	Ви	Ви	Ви	Ви
Зміни у власних бізнес-процесах	Серйозні	Серйозні	Середні	Незначні

Рисунок 1.5 - Цифрова трансформація ритейлу: ретроспектива і прогнози розвитку на 2021 рік.

Джерело: [5]

1.3 Постановка задачі

У межах цієї роботи передбачається розробка інтернет-магазину домашнього одягу із такими ключовими функціями:

Оптимізована пошукова система:

- застосування алгоритмів пошуку з урахуванням можливих помилок у введенні та синонімів;
- гнучкі фільтри (за розміром, кольором, матеріалом, ціновим діапазоном);
- динамічний каталог товарів;
- постійне оновлення інформації про наявність та нові надходження завдяки автоматизованій прив'язці до складу;
- механізм швидких прев'ю (pop-up-вікон) з основними характеристиками товару;

- керування запасами та складом;
- інтеграція з внутрішньою системою управління товарними позиціями;
- автоматичне створення замовлень постачання, коли запаси конкретного товару досягають критичної позначки.

Система рекомендацій:

- персоналізація на основі історії переглядів, покупок та відгуків.
- використання алгоритмів машинного навчання (collaborative filtering, content-based filtering тощо);
- коректне відображення каталогу та всіх функцій магазину на пристроях із різною роздільною здатністю (десктопи, планшети, смартфони);

Безпечний процес оформлення замовлень:

- підтримка https-протоколу, системи автентифікації та авторизації клієнтів;
- інтеграція безпечних платіжних шлюзів (liqpay, raupal, wayforpay тощо).

Модуль зворотного зв'язку та аналітики

- збір даних про поведінку користувачів, аналіз популярних товарів та відстеження конверсій;
- реалізація кабінету адміністратора для перегляду ключових показників (KPI) магазину.

Таким чином, даний інтернет-магазин має поєднувати оптимізовані алгоритми пошуку, сучасні механізми рекомендацій і зручну систему керування складськими залишками. Це дозволить не лише спростити вибір покупця, а й підвищити ефективність бізнес-процесів, що в підсумку покращить досвід користувачів та конкурентоспроможність проєкту.

Висновки до розділу 1

У цьому розділі проведено комплексний аналіз предметної області, розглянуто популярні аналоги та сформульовано технічні й функціональні

вимоги до інтернет-магазину домашнього одягу. Сучасні тенденції в інтернет-торгівлі свідчать про необхідність впровадження інноваційних технологій, зокрема механізмів персоналізації та оптимізованого пошуку, аби задовольнити зростаючі потреби користувачів.

Огляд аналогів (H&M Home, Oysho, Marks & Spencer) продемонстрував, що навіть провідні бренди мають низку обмежень, як-от недостатня гнучкість пошуку та відсутність повноцінної персоналізації. Наш проєкт покликаний вирішити подібні недоліки й запропонувати новий рівень комфорту для покупців, який включає інтегровану систему управління складом, рекомендаційні сервіси та максимально зручний каталог товарів.

Поставлені завдання охоплюють широкий спектр рішень: від алгоритмів пошуку до системи рекомендацій і автоматизованого управління логістикою. Такий підхід дає змогу створити цілісну та динамічну платформу, що задовольнить потреби користувачів і сприятиме розвитку електронної комерції в сегменті домашнього одягу.

РОЗДІЛ 2

ПРОЄКТУВАННЯ ВЕБСАЙТУ ІНТЕРНЕТ-МАГАЗИНУ

2.1 Проєктування структури інтернет-магазину «Затишок & Стиль»

Для реалізації вебсайту інтернет-магазину була побудована модель даних, яка враховує типову структуру товарного каталогу. Основними сутностями системи є: пошук і фільтрація товарів, рекомендації на основі вподобань, детальний опис товарі, кошик та оформлення замовлення.

Пошук і фільтрація товарів (рис 2.1). Користувачі повинні мати змогу шукати товари за назвою, категорією, ціновим діапазоном, кольором чи матеріалом. Усе максимально просто і зручно.

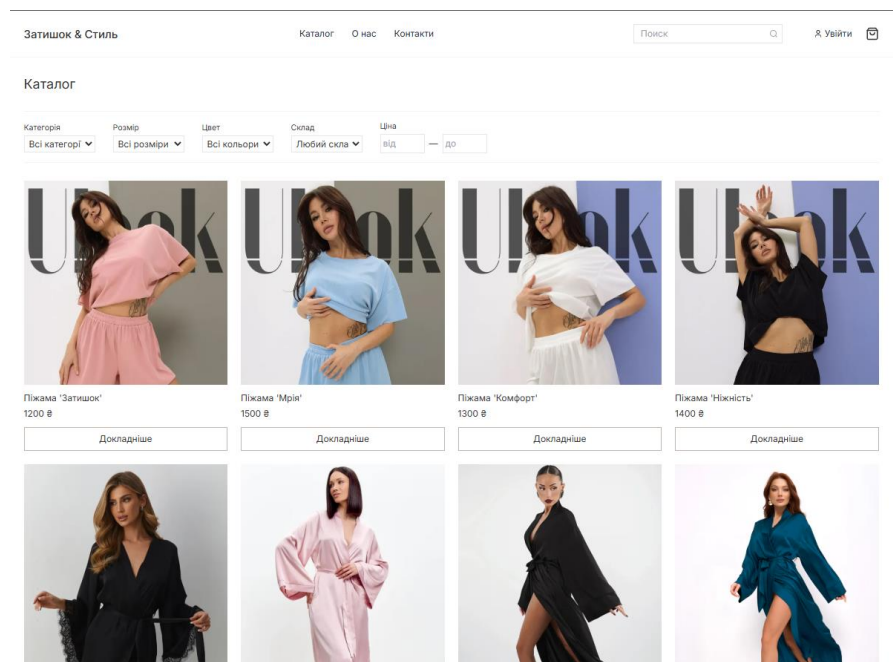


Рисунок 2.1 – Пошук та фільтрація

Джерело: розроблено автором

Рекомендації на основі вподобань. Система має вміти аналізувати вибрані товари, минулі замовлення або навіть дії користувачів, щоб запропонувати щось дійсно корисне. Це не просто магазин, а асистент стилю

та затишку.

Детальний опис товарі (рис 2.2.). У кожного товару має бути повна інформація – від характеристик і фото до відгуків клієнтів. Ми хочемо, щоб користувач був максимально впевнений у виборі.



Рисунок 2.2 – Опис товару

Джерело: розроблено автором

Кошик та оформлення замовлення (рис 2.3 та рис. 2.4). Додавання до кошика, редагування його вмісту, вибір способу доставки і оплати мають бути простими кроками, а не викликом.

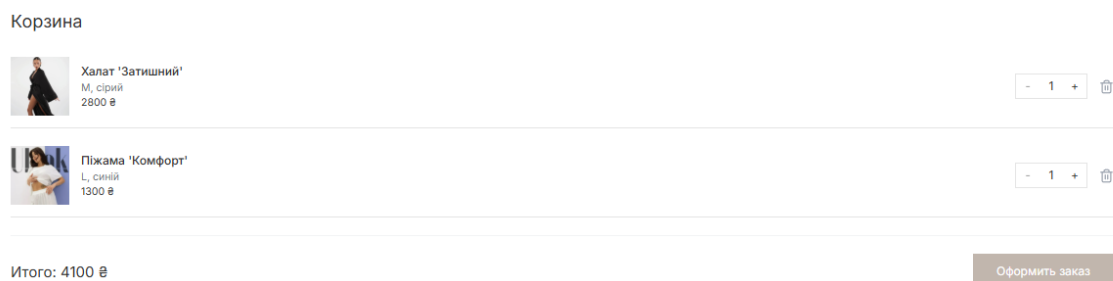


Рисунок 2.3 – Кошик

Джерело: розроблено автором

Оформлення замовлення

Ім'я

Телефон

Адреса доставки

Спосіб доставки

Кур'єр (500 грн)

Самовідвезення (безкоштовно)

Спосіб оплати

Онлайн картою

При отриманні

Підтвердити замовлення

Рисунок 2.4 – Оформлення замовлення

Джерело: розроблено автором

Щоб краще візуалізувати, як саме користувач буде взаємодіяти з сайтом, було створено діаграму прецедентів (рис 2.5). Вона дозволяє побачити основні функції системи з точки зору користувача і зрозуміти, як саме кожна дія буде реалізована.

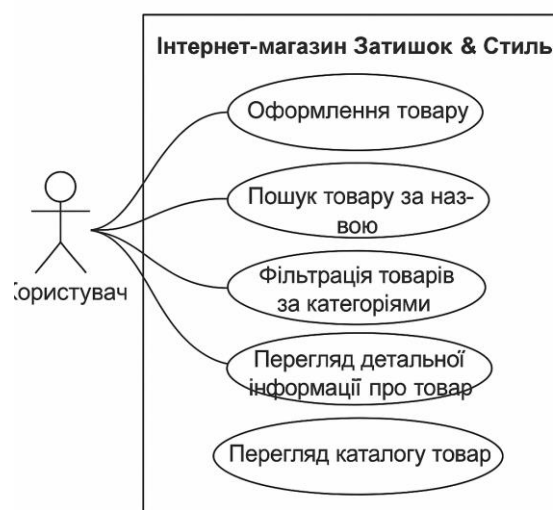


Рисунок 2.5 – Діаграма прецедентів

Джерело: розроблено автором

Для рекомендаційної системи передбачено використання алгоритмів типу content-based filtering, які аналізують характеристики переглянутих або обраних товарів і пропонують подібні за стилем, кольором чи матеріалом.

2.2 Моделювання процесів

Діаграма діяльності (рис 2.6.) – це вид діаграм у моделюванні програмного забезпечення, який застосовується для опису послідовності дій, які відбуваються у межах системи. Вона допомагає чітко побачити, який шлях проходить користувач, які дії доступні в певний момент і як система реагує на введені дані або прийняті рішення.

У нашому випадку, мова йде про веб-додаток інтернет-магазину «Затишок & Стиль», який спеціалізується на товарах для створення домашнього затишку, інтер'єру, а також стильних речей для побуту.



Рисунок 2.6 – Діаграма діяльності

Джерело: Автор

Діаграма діяльності демонструє, як користувач взаємодіє із сайтом: від початкового входу на головну сторінку – до оформлення покупки. Нижче представлено ключовий сценарій:

- початок взаємодії. Користувач відкриває головну сторінку магазину. На цьому етапі він приймає рішення: чи входити в особистий кабінет, чи продовжити без авторизації;
- реєстрація або авторизація. Якщо користувач ще не має акаунта, система пропонує зареєструватися. Зареєстровані користувачі проходять авторизацію;
- огляд і пошук товарів. Користувач може переглядати каталог, застосовувати фільтри за категоріями (наприклад, “Декор”, “Текстиль”, “Світильники”) або шукати товари за назвою;
- перегляд товару. Клікнувши на вподобаний товар, користувач переходить на сторінку з детальним описом – фото, матеріали, доступні кольори, відгуки тощо;
- прийняття рішення. Якщо товар сподобався, користувач додає його до кошика. Цей крок доступний лише після входу до акаунта;
- оформлення замовлення. Після перегляду вмісту кошика користувач переходить до оформлення покупки – вказує адресу, обирає спосіб доставки та оплати;
- завершення процесу. Після підтвердження покупки та оплати – взаємодія завершується.

2.3 Створення прототипів

Перед реалізацією функціоналу було створено прототипи інтерфейсу за допомогою середовища розробки Figma (рис. 2.7 та 2.8). Створення прототипів дало змогу візуалізувати загальну структуру сайту, перевірити логіку переходів між сторінками та розташування ключових функціональних елементів інтерфейсу, а також оцінити їхню ергономічність і зручність для кінцевого користувача.

Під час розробки було змодельовано такі основні сторінки вебсайту:

- головну сторінку з банером, навігаційним меню та блоками популярних категорій;
- сторінку каталогу товарів із можливістю багатокритеріальної фільтрації;
- сторінку картки товару з докладним описом, фото й відгуками;
- сторінку оформлення замовлення з вибором способу доставки й оплати;
- персональний кабінет користувача для перегляду історії покупок і керування профілем.

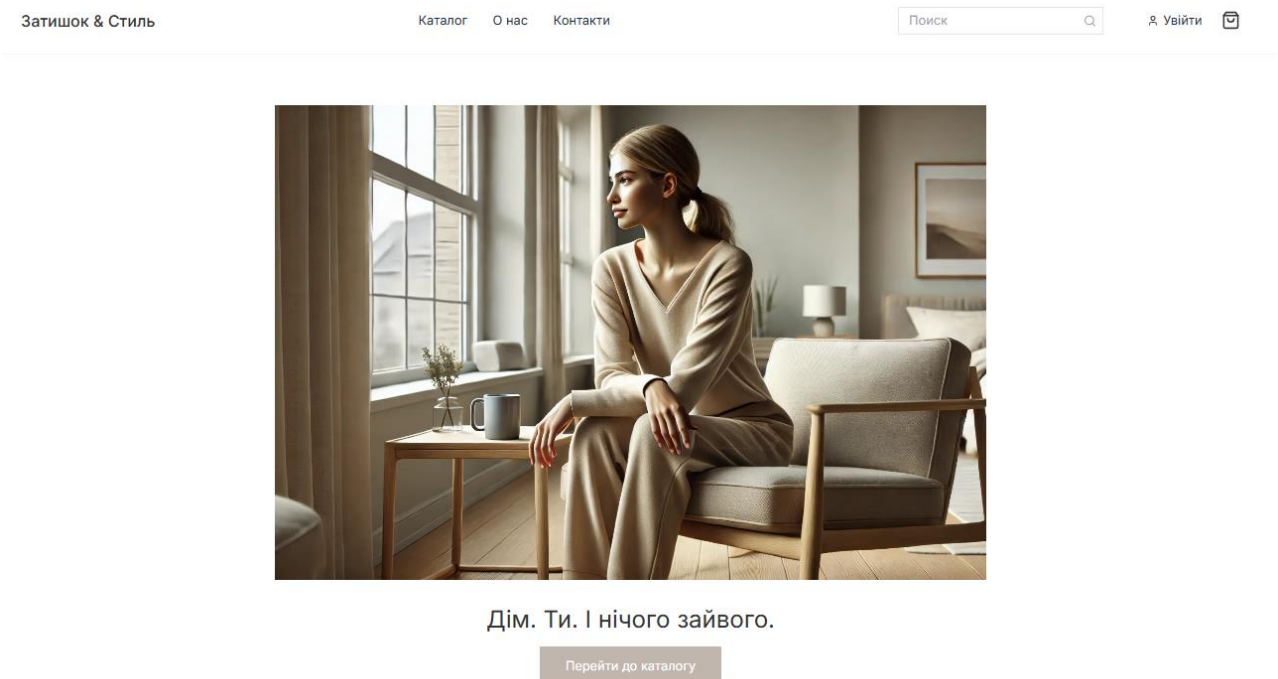


Рисунок 2.7 – Прототип

Джерело: розроблено автором

Створений прототип було використано для проведення попереднього тестування користувацьких сценаріїв, що дозволило своєчасно виявити й виправити потенційні недоліки у розташуванні інтерфейсних елементів, логіці взаємодії та навігації.



Рисунок 2.8 – Прототип

Джерело: Автор

2.4 Проєктування дизайну та інтерфейсу

Під час розробки інтерфейсу веб-додатку інтернет-магазину «Затишок & Стиль» основний акцент було зроблено на створенні зручного, естетичного та доступного користувацького досвіду. В основу проєктування покладено принципи сучасного UX/UI-дизайну, що дозволяють максимально спростити навігацію по сайту та зробити процес покупки інтуїтивно зрозумілим для будь-якого користувача.

Чітка ієрархія інформації. Структура каталогу, фільтрів та карток товарів побудована так, щоб користувач легко міг знайти потрібне. Важлива інформація – назва, ціна, кнопка «Докладніше» – розміщена в зонах першочергової уваги.

Інтуїтивна навігація. Меню, кнопка «Кошик», розділ обраного та особистий кабінет знаходяться на видному місці, що дає змогу швидко взаємодіяти із сайтом навіть вперше.

Адаптивність інтерфейсу. Сайт коректно відображається на смартфонах, планшетах і десктопах (рис 2.8). Елементи автоматично підлаштовуються під розмір екрана, що забезпечує зручність у будь-яких умовах користування.

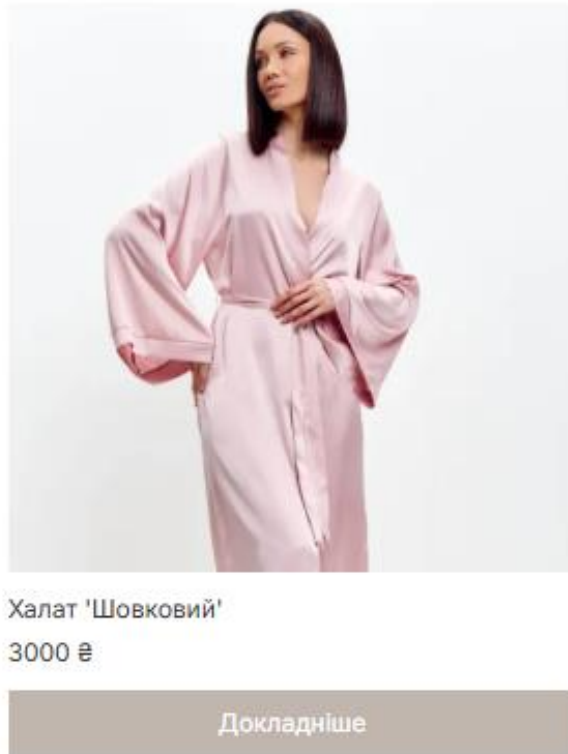


Рисунок 2.8 – Звичне розташування

Джерело: Автор



Рисунок 3.2 – Кошик

Джерело: Автор

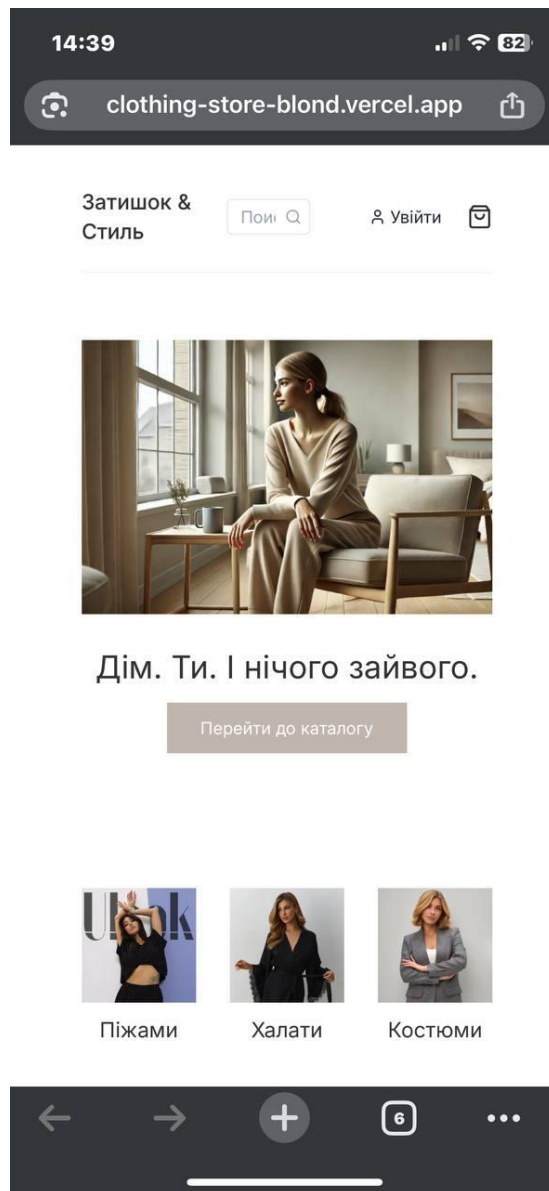


Рисунок 3.3 – Вигляд зі смартфона

Джерело: Автор

Доступність для всіх користувачів. Дизайн враховує потреби людей із порушеннями зору. Контрастні кольори, крупні шрифти та прості елементи навігації сприяють зручному перегляду вмісту незалежно від пристрою чи особливостей користувача.

Висновок до розділу 2

У другому розділі було детально розглянуто процес проектування вебсайту інтернет-магазину «Затишок & Стиль» – від базової структури до продуманого дизайну інтерфейсу.

У результаті аналізу функціональності та потреб користувача було сформовано чітке уявлення про те, якою має бути сучасна онлайн-платформа для продажу товарів домашнього вжитку: зручною, швидкою та естетично привабливою.

Створена структурна модель сайту враховує всі необхідні функції – від фільтрації товарів до рекомендаційної системи. Завдяки діаграмам прецедентів та діяльності вдалося наочно відобразити взаємодію користувача з системою та передбачити логіку всіх можливих сценаріїв.

Окрему увагу приділено етапу створення прототипів, що дало змогу попередньо оцінити юзабіліті сайту, перевірити розташування основних елементів та адаптивність макету.

Дизайн інтерфейсу було розроблено з урахуванням принципів UX/UI, що допомогло досягти балансу між візуальною привабливістю та практичною функціональністю. Обрана кольорова палітра – тепла, світла, ненав'язлива – відповідає загальній концепції бренду та створює атмосферу затишку з першого погляду на сторінку.

Загалом, розділ став основою для подальшої реалізації вебсайту: його логіка, структура і зовнішній вигляд уже сформовані, і тепер є чітке бачення, як усе має виглядати та працювати в реальному проєкті.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ ВЕБСАЙТУ ІНТЕРНЕТ-МАГАЗИНУ

3.1 Вибір технологій для створення вебресурсу

Для розробки вебсайту «Затишок & Стиль» було обрано сучасний стек технологій, який забезпечує швидкість роботи, надійність системи та зручність для кінцевого користувача.

Ключовим рішенням стала розробка сайту на базі Next.js –фреймворку для React, який поєднує можливості серверного та клієнтського рендерингу. Це дало змогу значно пришвидшити завантаження сторінок та покращити індексацію сайту пошуковими системами.

У якості бази даних обрано PostgreSQL, розгорнуту у хмарному середовищі Neon (рис 3.1.). Таке рішення дозволило забезпечити масштабованість проєкту, резервне копіювання та мінімізувати ризики втрати даних.

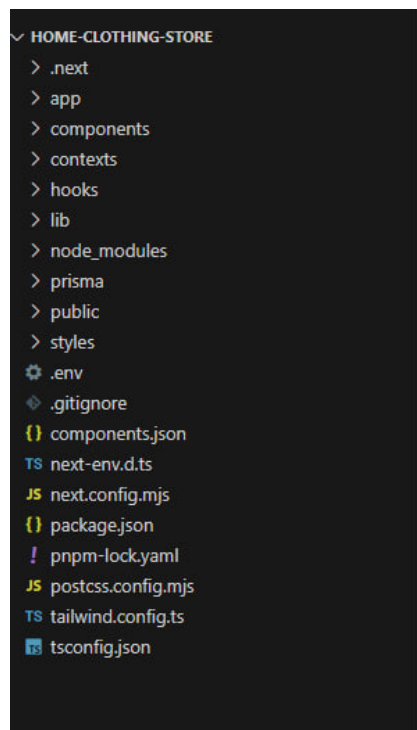


Рисунок 3.1 –База даних

Джерело: розроблено автором

Функціонал корзини реалізований за допомогою localStorage браузера. Завдяки цьому користувачі можуть додавати товари до корзини без обов'язкової авторизації, а дані зберігаються навіть після закриття або оновлення сторінки.

Керування версіями коду здійснювалося за допомогою Git, що дозволяло відслідковувати зміни та працювати над проєктом більш організовано.

Для розробки використовувалися середовище Visual Studio Code (рис 3.2.), сервіс Figma для проєктування інтерфейсу та Postman для тестування API-запитів.

Обраний набір технологій повністю відповідав поставленим завданням і дав можливість реалізувати всі необхідні функціональні можливості магазину.

```
1 "use client"
2
3 import type React from "react"
4
5 import { useState } from "react"
6 import Link from "next/link"
7 import { useRouter } from "next/navigation"
8 import { Header } from "@components/header"
9 import { useAuth } from "@contexts/auth-context"
10
11 export default function Login() {
12   const [email, setEmail] = useState("")
13   const [password, setPassword] = useState("")
14   const [error, setError] = useState("")
15   const { login } = useAuth()
16   const router = useRouter()
17
18   const handleLogin = async (e: React.FormEvent) => {
19     e.preventDefault()
20     setError("")
21
22     if (!email || !password) {
23       setError("Будь ласка, заповніть усі поля")
24       return
25     }
26
27     const success = await login(email, password)
28     if (success) {
29       router.push("/")
30     } else {
31       setError("Невірний email або пароль")
32     }
33   }
34
35   return (
36     <main className="min-h-screen bg-white text-[#333] font-['Inter',sans-serif]">
37       <Header />
38
39       <div className="container mx-auto py-12">
40         <div className="max-w-md mx-auto">
41           <h1 className="text-2xl mb-8 text-center">Вхід до акаунту</h1>
42         </div>
43     </main>
44   )
45 }
```

Рисунок 3.2 – Фрагмент коду (перегляд через “Visual Studio”)

Джерело: розроблено автором

3.2 Створення структури бази даних

База даних відіграє ключову роль у функціонуванні сайту, оскільки забезпечує збереження та обробку інформації про користувачів, товари й замовлення.

Структура бази даних була спроектована відповідно до потреб магазину та включає основні таблиці (рис 3.3.):

- users – для зберігання інформації про зареєстрованих користувачів, їхні контактні дані та ролі в системі;
- products – для опису товарів, їх характеристик, цін та зображень;
- categories – для класифікації товарів за типами (піжами, халати, костюми);
- orders – для збереження інформації про оформлені замовлення;
- orderitems – для фіксації складу кожного замовлення: перелік товарів, їх кількість і вартість.

Фільтрація товарів здійснюється безпосередньо через атрибути таблиці Products, тому створення окремої таблиці для фільтрів не потребувалося.

Робота з базою даних організована через API-інтерфейси, що дозволяє ефективно обробляти запити та забезпечує швидке отримання необхідної інформації.

3.3 Реалізація фільтрів та кошики

Щоб спростити пошук товарів і підвищити зручність користування сайтом, було впроваджено систему багатокритеріальної фільтрації.

Користувач має змогу одночасно обирати кілька параметрів для пошуку:

- категорію товару,
- розмір,
- колір,
- склад тканини,
- бажаний ціновий діапазон.

```

<div className="max-w-2xl">
  <form>
    <div className="grid grid-cols-1 gap-6 mb-8">
      <div>
        <label className="block mb-2">Ім'я</label>
        <input
          type="text"
          className="w-full border border-gray-200 p-2 focus:outline-none focus:border-[#c1b6ad]"
          required
        />
      </div>
      <div>
        <label className="block mb-2">Телефон</label>
        <input
          type="tel"
          className="w-full border border-gray-200 p-2 focus:outline-none focus:border-[#c1b6ad]"
          required
        />
      </div>
      <div>
        <label className="block mb-2">Адреса доставки</label>
        <textarea
          className="w-full border border-gray-200 p-2 h-24 focus:outline-none focus:border-[#c1b6ad]"
          required
        ></textarea>
      </div>
      <div>
        <label className="block mb-2">Спосіб доставки</label>
        <div className="space-y-2">
          <div className="flex items-center">
            <input type="radio" id="delivery1" name="delivery" className="mr-2" defaultChecked />
            <label htmlFor="delivery1">Кур'єр (500 грн)</label>
          </div>
          <div className="flex items-center">
            <input type="radio" id="delivery2" name="delivery" className="mr-2" />
            <label htmlFor="delivery2">Самовідвезення (безкоштовно)</label>
          </div>
        </div>
      </div>
      <div>
        <label className="block mb-2">Спосіб оплати</label>
        <div className="space-y-2">
          <div className="flex items-center">
            <input type="radio" id="payment1" name="payment" className="mr-2" defaultChecked />
            <label htmlFor="payment1">Онлайн карткою</label>
          </div>
          <div className="flex items-center">
            <input type="radio" id="payment2" name="payment" className="mr-2" />
            <label htmlFor="payment2">При отриманні</label>
          </div>
        </div>
      </div>
      <div>
        <button type="submit" className="px-8 py-3 bg-[#c1b6ad] text-white">
          Підтвердити замовлення
        </button>
      </div>
    </form>
  </div>
</main>
)
}

```

Рисунок 3.3 – Перегляд через Sublime Text

Джерело: розроблено автором

Фільтри реалізовані на базі Next.js (рис 3.4), з використанням серверного рендерингу та передачі параметрів через URL (рис 3.4). Після вибору критеріїв формується запит до бази даних PostgreSQL, де динамічно будується SQL-запит із відповідними умовами фільтрації.

```

interface Product {
  id: string
  name: string
  price: number
  image: string
  category: string
  sizes: string[]
  colors: string[]
  composition: string
}

function CatalogContent() {
  const searchParams = useSearchParams()
  const router = useRouter()
  const [products, setProducts] = useState<Product[]>([])
  const [loading, setLoading] = useState(true)
  const [allSizes, setAllSizes] = useState<string[]>([])
  const [allColors, setAllColors] = useState<string[]>([])
  const [allCategories, setAllCategories] = useState<string[]>([])
  const [allCompositions, setAllCompositions] = useState<string[]>([])

  useEffect(() => {
    const fetchProducts = async () => {
      try {
        const params = new URLSearchParams(searchParams.toString())
        const response = await fetch(`/api/products?${params.toString()}`)
        const data = await response.json() as Product[]
        setProducts(data)

        const sizes = Array.from(new Set(data.flatMap((p: Product) => p.sizes)))
        const colors = Array.from(new Set(data.flatMap((p: Product) => p.colors)))
        const categories = Array.from(new Set(data.map((p: Product) => p.category)))
        const compositions = Array.from(new Set(data.map((p: Product) => p.composition)))

        setAllSizes(sizes)
        setAllColors(colors)
        setAllCategories(categories)
        setAllCompositions(compositions)
      } catch (error) {
        console.error('Error fetching products:', error)
      } finally {
        setLoading(false)
      }
    }

    fetchProducts()
  }, [searchParams])

  const handleFilterChange = (key: string, value: string) => {
    const params = new URLSearchParams(searchParams.toString())
    if (value) {
      params.set(key, value)
    } else {
      params.delete(key)
    }
    router.push(`/catalog?${params.toString()}`)
  }
}

```

Рисунок 3.4 – Фільтри та категорії

Джерело: розроблено автором

Особливістю реалізації є:

- підтримка одночасного застосування кількох фільтрів,

- можливість швидкого скидання фільтрів одним натисканням,
- плавна підвантаження товарів без перезавантаження сторінки,
- оптимізована обробка запитів через налаштовані індекси в базі даних.

Завдяки такому підходу користувач отримує швидкий і зручний інструмент для пошуку необхідних товарів за кількома критеріями.

Однією з ключових функцій будь-якого інтернет-магазину є корзина товарів – механізм, який дозволяє користувачеві зберігати вибрані позиції перед оформленням замовлення. У рамках реалізації вебсайту «Затишок & Стиль» корзина була розроблена з урахуванням максимальної зручності для користувача та мінімального навантаження на сервер.

Особливості реалізації

Корзина реалізована на основі localStorage – вбудованого сховища браузера, яке дозволяє зберігати дані навіть після оновлення сторінки або закриття вкладки. Такий підхід дає змогу уникнути складних механізмів авторизації та зберігати вибір користувача незалежно від наявності акаунта.

Коли користувач додає товар у корзину:

- обрана позиція записується у localStorage разом із кількістю одиниць;
- дані автоматично зчитуються на сторінці "корзина" та відображаються у вигляді списку;
- користувач має можливість змінювати кількість товарів або видаляти позиції;
- підсумкова сума перераховується в режимі реального часу.

Технічна реалізація. У реалізації використовувалися можливості React (через Next.js) – зокрема React Context API для зберігання стану корзини та синхронізації її між сторінками. Це дозволяє:

- Зберігати стан корзини навіть при переході між різними розділами сайту;

- Оновлювати дані без перезавантаження;
- Підтягувати стан корзини одразу після відкриття сторінки.

Корзина також інтегрована з механізмом генерації замовлення: після натискання кнопки "Оформити", дані з localStorage можуть бути збережені на сервері (через API) або виведені у вигляді PDF/Word-документа для друку.

```

"use client"

import { useEffect, useState } from "react"
import Link from "next/link"
import { Header } from "@/components/header"
import { CartItem as CartItemType } from "@/lib/cart"
import { CartItem } from "@/components/cart-item"

export default function Cart() {

  const [cartItems, setCartItems] = useState<CartItemType[]>([])
  const [isLoading, setIsLoading] = useState(true)

  useEffect(() => {
    const savedCart = localStorage.getItem("cart")
    if (savedCart) {
      setCartItems(JSON.parse(savedCart))
    }
    setIsLoading(false)
  }, [])

  useEffect(() => {
    if (!isLoading) {
      localStorage.setItem("cart", JSON.stringify(cartItems))
    }
  }, [cartItems, isLoading])

  const updateQuantity = (id: string, size: string, color: string, newQuantity: number) => {
    setCartItems(prevItems =>
      prevItems.map(item => {
        if (item.id === id && item.size === size && item.color === color) {
          return { ...item, quantity: newQuantity }
        }
        return item
      })
    )
  }

  const removeItem = (id: string, size: string, color: string) => {
    setCartItems(prevItems => prevItems.filter(item =>
      !(item.id === id && item.size === size && item.color === color)
    ))
  }

  const total = cartItems.reduce((sum, item) => sum + item.price * item.quantity, 0)

  if (isLoading) {
    return <div className="flex justify-center items-center min-h-screen">Завантаження...</div>
  }

  return (
    <main className="min-h-screen bg-white text-[#333] font-['Inter',sans-serif]">
      <Header />

      <div className="container mx-auto px-4 py-12">
        <h1 className="text-2xl mb-8">Корзина</h1>

        {cartItems.length > 0 ? (
          <div className="space-y-6">
            <div className="space-y-6">
              {cartItems.map(item => (
                <CartItem
                  key={`-${item.id}-${item.size}-${item.color}`}
                  item={item}
                  onUpdateQuantity={updateQuantity}
                  onRemove={removeItem}
                />
              ))}
            </div>
          </div>
        ) : (
          <div className="text-center">
            Корзина порожня
          </div>
        )}

        <div className="text-right mt-8">
          Загальна сума: {total} грн
        </div>
      </div>
    </main>
  )
}

```

Рисунок 3.5 – Корзина товарів

Джерело: розроблено автором

3.4 ТЕСТУВАННЯ РЕАЛІЗОВАНОЇ СИСТЕМИ

Тестування програмного продукту є завершальним етапом його розробки і відіграє ключову роль у перевірці працездатності всієї системи. Його основна мета – переконатися, що всі реалізовані функції вебсайту «Затишок & Стиль» працюють відповідно до початкових вимог, не містять критичних помилок і забезпечують комфортну взаємодію для користувача.

Під час тестування передбачалося досягнення таких конкретних завдань:

- перевірити коректність роботи багатокритеріального фільтра: відображення відповідних товарів при зміні параметрів (категорія, розмір, колір, ціна тощо);
- переконатися у стабільній роботі кошика: додавання, редагування кількості та видалення товарів;
- оцінити збереження стану кошика за допомогою механізму **localStorage** – після оновлення сторінки або повторного входу;
- провести перевірку адаптивності – наскільки інтерфейс сайту коректно відображається на різних екранах (смартфон, планшет, ноутбук);
- виявити потенційні проблеми з валідацією форм і обробкою введених даних;
- заміряти швидкодію при взаємодії з фільтрами, перегляді товарів, відкритті сторінок;

Таким чином, тестування було спрямоване як на перевірку функціональності, так і на оцінку зручності та стабільності роботи сайту.

3.4.1 План тестування

З огляду на поставлені цілі, для перевірки якості реалізованої системи було обрано комбінований підхід до тестування, що включав ручні перевірки, використання інструментів браузера, а також моделювання типових дій користувача.

Серед основних типів тестування, які проводилися, варто виділити такі:

- функціональне тестування, яке охоплювало всі ключові сценарії взаємодії користувача з сайтом: додавання товарів до корзини, зміна їх кількості, очищення фільтрів, застосування фільтрації за кількома параметрами, оновлення підсумкової вартості, а також перевірка валідації при спробі оформити замовлення без заповнення обов'язкових полів;
- тестування збереження стану (localStorage), яке полягало у перевірці, чи зберігаються вибрані товари в корзині після оновлення сторінки, переходу між розділами сайту та повторного входу в браузер;
- Тестування продуктивності, в межах якого оцінювались швидкість рендерингу сторінок, оперативність оновлення товарів при фільтрації, а також час відгуку API-запитів до бази даних PostgreSQL.

Такий підхід дозволив охопити як технічну стабільність, так і зручність реального використання вебресурсу.

```

// Fetch similar products
const similarResponse = await fetch(`/api/products/similar?currentProductId=${resolvedParams.id}`)
if (!similarResponse.ok) throw new Error("Не вдалося знайти схожі товари")
const similarData = await similarResponse.json()
setSimilarProducts(similarData)
} catch (err) {
  setError(err instanceof Error ? err.message : 'Виникла помилка')
} finally {
  setLoading(false)
}
}

fetchData()
}, [resolvedParams.id])

if (loading) return <div>Завантаження...</div>
if (error) return <div>Помилка: {error}</div>
if (!product) return <div>Товар не знайдено</div>

return (
  <div className="min-h-screen bg-white text-[#333] font-['Inter',sans-serif]">
    <Header />
    <div className="container mx-auto px-4 py-8">
      <div className="grid grid-cols-1 md:grid-cols-2 gap-8">
        <div className="aspect-square overflow-hidden bg-gray-50">
          <Image
            src={product.image || "/placeholder.svg"}
            alt={product.name}
            width={800}
            height={800}
            className="object-cover w-full h-full"
          />
        </div>
        <div>
          <h1 className="text-3xl mb-4">{product.name}</h1>
          <p className="text-2xl mb-6">{product.price} €</p>
        </div>
      </div>
    </div>
  </div>
)

```

Рисунок 3.6 – Помилки

Джерело: Автор

Тестування форм та обробки помилок. Навмисно вводились некоректні або пусті значення у поля (наприклад, порожній email, нечислові значення у поле "Ціна") – для перевірки обробки винятків.

3.4.2 Результати тестування

За підсумками проведених перевірок можна стверджувати, що вебсайт «Затишок & Стиль» демонструє стабільну роботу та виконує всі ключові функції відповідно до початкових вимог. Основні дії – такі як додавання товарів до корзини, застосування фільтрів, обробка форм та навігація сторінками – відбуваються без збоїв або затримок. Інтерфейс залишається інтуїтивно зрозумілим і зручним навіть на мобільних пристроях.

Під час первинного тестування було зафіксовано кілька незначних проблем, які оперативно усунули в процесі доопрацювання:

- після видалення товару з корзини не оновлювалась підсумкова сума – проблему вирішено через оновлення стану у компоненті корзини;
- у мобільній версії фільтри перекривали основний контент – це усунуто шляхом введення додаткових умов приховування при ширині екрана нижче 768 пікселів;
- у формі оформлення замовлення відсутня перевірка правильності номера телефону – впроваджено відповідну валідацію формату вводу.

Повторне тестування після внесення змін підтвердило стабільність роботи системи та її готовність до подальшого використання в умовах реального навантаження.

3.5 Перевірка працездатності та напрями подальшого розвитку системи

Після завершення реалізації ключових функціональних модулів сайту «Затишок & Стиль» було проведено тестування розробленої системи. Метою цього етапу стало виявлення можливих помилок у роботі застосунку, перевірка зручності інтерфейсу та визначення рівня готовності ресурсу до реального використання. Крім того, аналіз результатів розробки дозволив окреслити перспективні напрями вдосконалення проєкту.

3.5.1 Результати тестування

Комплексне тестування охоплювало функціональні, адаптивні та навантажувальні сценарії, а також перевірку роботи з localStorage і реакцію сайту на типові дії користувача. Основні функції – додавання товарів до корзини, фільтрація, редагування кількості товарів, очищення форм і адаптивне відображення – працюють стабільно, як на десктопних, так і на мобільних пристроях.

У процесі перевірки було виявлено низку незначних помилок (наприклад, некоректне оновлення суми при видаленні товару, конфлікт фільтрів на мобільному інтерфейсі), які були оперативно виправлені. Після повторної перевірки система продемонструвала стабільність і відповідність усім заявленим вимогам.

3.5.2 Потенціал розвитку та вдосконалення

Хоча система повністю функціональна та готова до використання, її архітектура дозволяє розширення функціоналу з мінімальними витратами. У результаті аналізу було сформовано низку перспективних напрямів розвитку:

Впровадження особистого кабінету користувача

На даний момент користувач може здійснювати покупки без реєстрації.

Проте реалізація системи авторизації й особистого кабінету дозволить:

- зберігати історію замовлень;
- створювати список обраного;
- прискорювати повторне оформлення;
- забезпечити вищий рівень персоналізації взаємодії.

Цей функціонал покращить досвід постійних покупців і дозволить реалізувати інші функції, залежні від автентифікації (наприклад, рейтинг товарів, персональні знижки).

Розширення рекомендаційної системи

У майбутньому доцільно реалізувати рекомендації на основі поведінкових даних користувача: попередніх переглядів, історії покупок, доданих до кошика товарів тощо. Для цього можуть бути використані:

- алгоритми машинного навчання;
- сторонні API з готовими сервісами персоналізації;
- аналіз агрегованих даних користувачів.

Це не лише покращить користувацький досвід, а й підвищить конверсію.

Інтеграція платіжних систем

На наступному етапі розвитку важливо підключити реальні платіжні сервіси, такі як LiqPay, WayForPay, Stripe або Fondy. Це дозволить:

- оформлювати онлайн-замовлення з оплатою на сайті;
- автоматично підтверджувати платежі та змінювати статус замовлення;
- зменшити кількість покинутих корзин.

Інтеграція платіжних систем є критично важливою для перетворення ресурсу на повноцінну комерційну платформу.

Аналітика та звітність для адміністратора

Додатковим напрямом удосконалення є створення адміністративного модуля з базовою статистикою та аналітикою. Такий функціонал може включати:

- загальну кількість замовлень за період;
- статистику переглядів товарів;
- аналіз популярних категорій;
- ефективність фільтрів;
- динаміку продажів.

Зібрані дані дадуть змогу приймати зважені бізнес-рішення щодо асортименту, маркетингу та цінової політики.

Розгортання на продакшн-хостингу та SEO-оптимізація

Щоб сайт міг ефективно працювати у відкритому доступі, необхідно забезпечити:

- розміщення на стабільному хостингу (наприклад, Vercel, Netlify або VPS-сервері);
- базову SEO-оптимізацію (title, meta-description, Open Graph, sitemap, robots.txt);
- швидке завантаження сторінок, зокрема на мобільних пристроях;
- доступність для людей з інвалідністю (a11y);
- підтримку кількох мов (наприклад, української та англійської).

Ці кроки підвищують видимість ресурсу в пошукових системах, що позитивно вплине на трафік і продажі.

Висновок до розділу 3

У цьому розділі було детально розглянуто процес реалізації вебсайту інтернет-магазину «Затишок & Стиль» – від вибору технологічного стеку до впровадження ключових функціональних модулів. Здійснено обґрунтований вибір Next.js як фреймворку для розробки інтерфейсу, PostgreSQL (Neon) для роботи з базою даних, а також localStorage як зручного інструменту для збереження стану корзини на стороні клієнта без потреби авторизації.

У рамках реалізації описано структуру бази даних, розробку багатокритеріального фільтра для ефективного пошуку товарів, створення адаптивного інтерфейсу, а також впровадження клієнтської корзини, яка зберігає дані між сесіями. Проведене тестування підтвердило стабільну роботу всіх основних функцій, відсутність критичних помилок, а також високу швидкодію ресурсу.

Окрему увагу було приділено аналізу подальших можливостей розвитку проєкту. Запропоновано низку удосконалень, які можуть бути реалізовані на наступних етапах:

- інтеграція платіжних систем для повноцінної онлайн-оплати;
- створення аналітичного модуля для адміністратора;
- розширення рекомендаційної системи;
- оптимізація для пошукових систем і публічне розгортання сайту.

Таким чином, у межах третього розділу було не лише реалізовано повноцінний MVP онлайн-магазину, а й закладено основу для його подальшої трансформації у масштабний комерційний продукт. Сайт відповідає сучасним вимогам до швидкості, зручності та функціональності, а також має значний потенціал для розширення як з технічного, так і з бізнесового боку.

ВИСНОВКИ

У межах кваліфікаційної роботи було розроблено повнофункціональний вебсайт інтернет-магазину домашнього одягу «Затишок & Стиль», що враховує актуальні потреби електронної комерції та реалізує сучасний підхід до управління асортиментом, пошуку й персоналізації.

У першому розділі проведено аналіз предметної області, розглянуто аналогічні платформи, визначено функціональні та технічні вимоги до системи. На цій основі було сформовано чітку постановку задачі й окреслено переваги розроблюваного рішення: багатокритеріальна фільтрація, рекомендаційна система й адаптивний інтерфейс.

У другому розділі розроблено структуру вебсайту, спроектовано базу даних, діаграми процесів і прецедентів. Створено прототипи інтерфейсу, що дозволило перевірити логіку роботи й користувацький шлях ще до реалізації.

У третьому розділі здійснено технічну реалізацію платформи на базі стеку Next.js, React і PostgreSQL. Описано реалізацію кошика й багатокритеріальних фільтрів, тестування функціоналу й адаптивного дизайну. За результатами тестування сайт стабільно виконує всі заявлені функції, коректно обробляє фільтрацію й оформлення замовлень, а завдяки localStorage користувацькі сесії залишаються збереженими навіть при оновленні сторінки.

Наприкінці окреслено напрями подальшого розвитку проєкту: впровадження платіжних сервісів, розширення рекомендаційної системи на основі машинного навчання, реалізація аналітики для адміністратора та SEO-оптимізація для підвищення видимості платформи.

Розроблений сайт відповідає сучасним вимогам зручності, продуктивності й безпеки, має потенціал масштабування та розвитку у повноцінний комерційний продукт для ринку електронної комерції.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Revenue of the e-commerce industry in the U.S. 2019-2029 // Global Apparel Market - Statistics & Facts. Statista. – URL: <https://www.statista.com/topics/2443/apparel-market/>
2. H&M Official Website. Категорія “Home”.
URL: https://www2.hm.com/en_gb/home.html
3. Oysho – Homewear Collection.
URL: <https://www.oysho.com/ua/homewear>
4. Marks & Spencer – офіційний веб-сайт. Розділ “Loungewear”.
URL: <https://www.marksandspencer.com/l/women/nightwear/loungewear>
5. E-commerce in Fashion and Apparel. Adobe Blog.
URL: <https://blog.adobe.com/en/topics/fashion>
6. Офіційна документація з веб-розробки. MDN Web Docs.
URL: <https://developer.mozilla.org/>
7. Патерни проєктування для веб-застосунків. Microsoft Docs.
URL: <https://learn.microsoft.com/uk-ua/azure/architecture/patterns/>
8. Machine Learning for Recommender Systems. Google Developers.
URL: <https://developers.google.com/machine-learning/recommendation>
9. E-commerce Security Best Practices. OWASP.
URL: <https://owasp.org/www-project-top-ten/>