

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук

«До захисту допущено»

В.о. завідувача кафедри

Оксана ШОВКОПЛЯС

(підпис)

червня 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня бакалавр

зі спеціальності 122 - Комп'ютерних наук,
освітньо-професійної програми «Інформатика»
на тему: «Сайт-каталог побутової техніки»
здобувача групи ІН-06-2 Нозадзе Кахабер Давидович

Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання на
відповідне джерело.

Кахабер НОЗАДЗЕ

(підпис)

Керівник,
доцент кафедри комп'ютерних наук

Наталія Барченко

(підпис)

Суми – 2024

Сумський державний університет
Центр заочної, дистанційної та вечірньої форм навчання
Кафедра комп'ютерних наук

«Затверджую»

В.о. завідувача кафедри

Оксана ШОВКОПЛЯС

(підпис)

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня бакалавра

зі спеціальності 122 - Комп'ютерних наук, освітньо-професійної програми «Інформатика»

здобувача групи ІН-06-2 Нозадзе Кахабер Давидович

1. Тема роботи: «Сайт-каталог побутової техніки»

затверджую наказом по СумДУ від «22» квітня 2024 р. № 0414-VI

2. Термін здачі здобувачем кваліфікаційної роботи до 29 травня 2024 року

3. Вхідні дані до кваліфікаційної роботи _____

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)

1) Аналіз проблеми предметної області, постановка й формування завдань дослідження.

2) Огляд технологій, що використовуються для розробки вебсайту. 3) Розробка каталогу

побутової техніки. 4) Аналіз результатів.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____

6. Консультанти до проекту (роботи), із зазначенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання «__» _____ 202_ р.

Завдання прийняв до виконання _____

(підпис)

Керівник _____

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Термін виконання	Примітка
1	<i>Аналіз предметної області, постановка й формування завдань дослідження</i>	06.05.24- 9.05.24	
2	<i>Огляд технологій, що використовуються для розробки вебсайту</i>	10.05.24- 13.05.24	
3	<i>Розробка каталогу побутової техніки</i>	14.05.24- 22.05.24	
4	<i>Аналіз отриманих результатів</i>	23.05.24- 27.05.24	
5	<i>Оформлення пояснювальної записки до кваліфікаційної роботи</i>	28.05.25- 31.05.24	

Здобувач вищої освіти

(підпис)

Керівник

(підпис)

АНОТАЦІЯ

Записка: 49 стр., 28 рис., 4 додатка, 20 використаних джерел.

Обґрунтування актуальності теми роботи – Тема кваліфікаційної роботи є актуальною, оскільки сайт-каталог дозволяє споживачам швидко і зручно знаходити та порівнювати необхідні пристрої для дому, забезпечуючи інформоване рішення завдяки оглядам і рейтингу продукції.

Об’єкт дослідження — інтернет-каталог побутової техніки.

Мета роботи — розробка інтернет-каталог побутової техніки

Методи дослідження — інструменти розробки вебсайтів, проектування та наповнення БД.

Результати — розроблено інтернет-каталог, який надає користувачам інформацію про побутову техніку. Надає можливість переглядати товар, шукати його. Реалізовано особистий кабінет, через який можна переглянути відібраний товар. Було проведено тестування методом чорного ящика.

ІНТЕРНЕТ-КАТАЛОГ ПОБУТОВОЇ ТЕХНІКИ, HTML, CSS, JAVASCRIPT, PHP, MYSQL.

Зміст

ВСТУП.....	2
1. ІНФОРМАЦІЙНИЙ ОГЛЯД.....	3
1.1 Аналіз принципів побудови ІС	3
1.2 Аналіз аналогів	4
1.3 Постановка задачі	7
2 ВИБІР МЕТОДІВ РОВ'ЯЗАННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ.....	9
2.1 Вимоги до інтернет-каталогу	9
2.2 Інструменти розробки вебсайтів	9
3 ПРОГРАМНА РЕАЛІЗАЦІЯ	18
3.1 Визначення структури додатка	18
3.2 Структура бази даних	22
3.3 Розробка дизайну	23
3.3 Розробка функціоналу системи	29
3.4 Тестування	31
ВИСНОВКИ.....	33
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	34
Додаток А Код html	36
Додаток Б Код CSS	43
Додаток В Код JS	44
Додаток Г Код PHP	45

ВСТУП

Актуальність. Актуальність сайтів побутової техніки зростає у зв'язку з постійним розвитком інтернету. Сучасні сайти дозволяють замовляти, слідкувати та обробляти замовлення через інтернет автоматично, за допомогою скриптів та сторінок коду. Це приводить к зменшенню навантаженню на працівників та економії ресурсів, які можна використати на інших напрямках.

Об'єкт дослідження. Інформаційні системи для сайтів побутової техніки.

Предмет дослідження. Розробка та впровадження інформаційної системи для сайту побутової техніки, яка допоможе автоматизувати процеси роботи магазину.

Гіпотеза. Розроблена інформаційна система для сайту побутової техніки значно вдосконалить процес управління бізнесом, забезпечить високий рівень точності та швидкості обробки замовлень, спростить взаємодію з клієнтами.

Новизна. Система сприяє підвищенню ефективності продажів, оптимізації та автоматизації роботи адміністрації, покращенню загальної якості обслуговування клієнтів.

Структура. Дана робота складається зі вступу, інформаційного огляду, вибору методу розв'язання поставленої задачі, програмної реалізації, висновків, списку використаних джерел та додатків.

1. ІНФОРМАЦІЙНИЙ ОГЛЯД

1.1 Аналіз принципів побудови ІС

Побудова інтернет-каталогів є важливою частиною веб-систем і додатків для організації інформації та навігації користувачів. Основні принципи побудови інтернет-каталогів:

- Ієрархічна структура: Каталоги зазвичай мають ієрархічну структуру, подібну до дерева. Це означає, що інформація організована у вигляді категорій та підкатегорій, які можуть мати підкатегорії ще більш низького рівня. Користувачі можуть навігувати між різними рівнями глибини каталогу [1].

- Класифікація інформації: Кожен елемент у каталозі повинен бути чітко класифікований і віднесений до відповідної категорії або підкатегорії. Це допомагає користувачам легше знаходити потрібну інформацію.

- Пошукова система: Практично всі сучасні інтернет-каталоги також мають пошукову систему. Вона дозволяє користувачам швидко знаходити інформацію за ключовими словами або фразами, що значно полегшує навігацію.

- Метадані і описи: Кожен елемент у каталозі повинен мати метадані та короткий опис. Це допомагає користувачам отримати загальне уявлення про зміст безпосередньо на сторінці елемента.

- Фільтри і сортування: Надаючи користувачам можливість фільтрувати та сортувати результати пошуку, ви дозволяєте їм точніше налаштовувати пошуковий запит.

- Можливість додавання інформації: Деякі інтернет-каталоги дозволяють користувачам додавати нову інформацію або редагувати існуючу. Це може бути корисним для покращення актуальності та повноти каталогу.

- Адаптивний дизайн: Важливо, щоб інтерфейс каталогу був адаптивним і добре виглядав як на десктопних, так і на мобільних пристроях.

– Навігація та спрощення: Забезпечення простої та легкої навігації є ключовим. Важливо мати наявність хлібних крихт, кнопок "назад" та "вперед", а також інших елементів для спрощення навігації[2].

– Зручний пошук: Покращений пошук, який враховує синоніми, орфографічні помилки та інші фактори, може покращити відповідність результатів пошуку.

– Аналітика та відстеження користувацької активності: Моніторинг користувацької активності може надати важливу інформацію для подальшого вдосконалення каталогу.

Варто зазначити, що це лише ознаки або рекомендації до побудови інтернет-каталогів а не вимоги, тому чітко дотримуватися кожного пункту не обов'язково.

1.2 Аналіз аналогів

Так як, дуже важко знайти інтернет-каталог присвячений саме побутовій техніці, було прийнято рішення досліджувати вебсторінки великих побутових магазинів, таких як: VISTA та EPICENTR.

Вебсайт «VISTA»[3] – офіційний вебсайт однієї з найбільших мереж будівельно-господарчих маркетів у місті Суми, яка працює з 2001 року. Вебсайт являє собою інтернет-магазин для багатьох речей, в тому числі будівельних. На головній сторінці (рисунок 1.1) можна знайти навігаційне меню, інформацію про знижки, нові колекції.

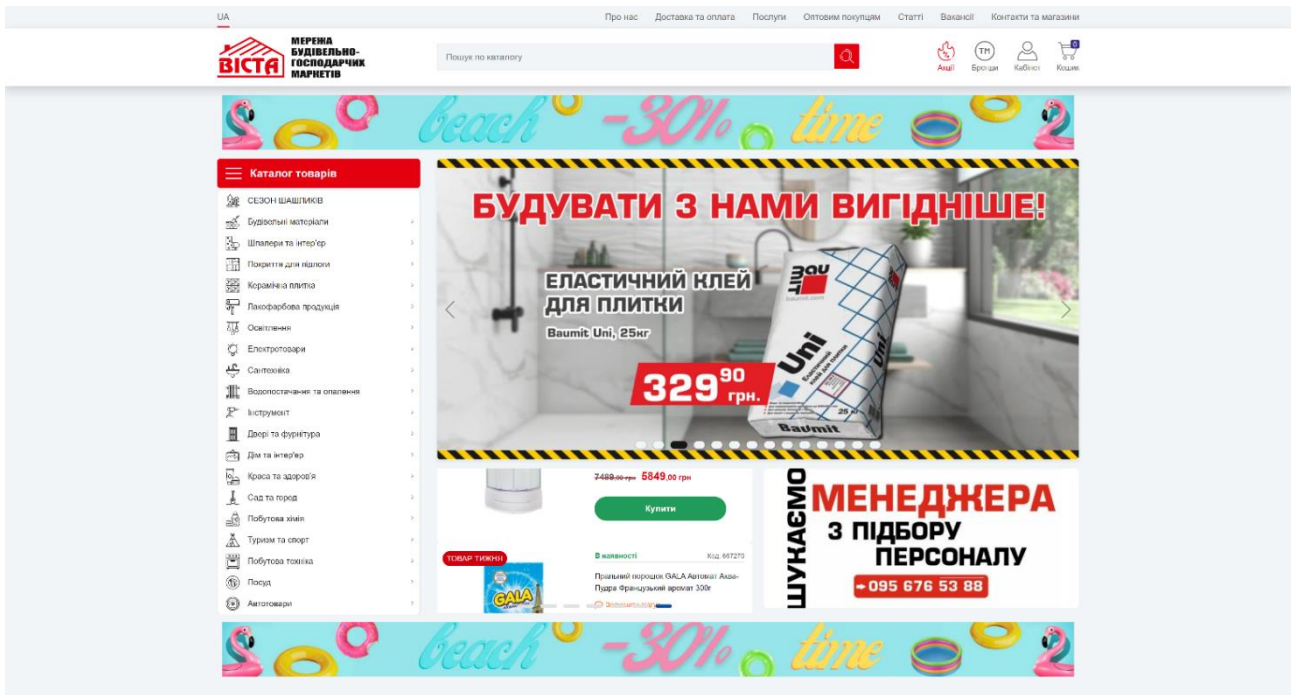


Рисунок 1.1 – Головна сторінка «VISTA»

На вебсторінках магазину (рис. 1.2) наявний пошук за допомогою категорії.

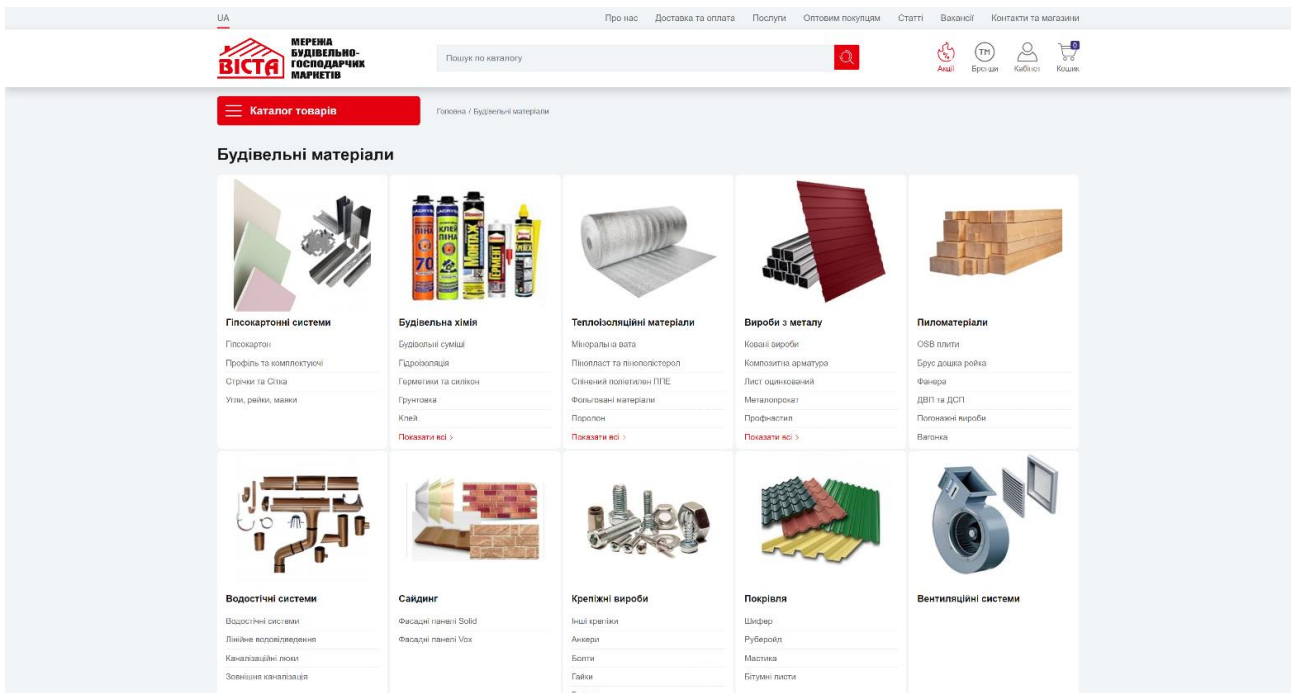


Рисунок 1.2 – Пошук товару за допомогою категорії

Також на вебсайті наявні такі функції як: зміна мови, список бажаних речей, кошик, «живий» чат підтримки клієнтів та власний кабінет з інформацією про замовлення.

Переваги сайту:

- 2 зрозумілий та простий до сприйняття дизайн;
- 3 зрозуміле навігаційне меню;
- 4 сайт швидко реагує на дії користувача;
- 5 є вся необхідна інформація та функції.

Вебсайт «EPICENTR»[4] – партнер державної програми «Відновлення» – пропонує своїм відвідувачам найбільший асортимент: Айфон 14, навушники AirPods 3, товари для виживання, шланги, товари для саду, а також спорттовари, все для походу, посуд і побутова хімія, пральний порошок, меблі, книги, дитячі товари і коляски, сантехніка, автотовари та побутове освітлення за найкращою ціною в Україні. В Епіцентрі представлено безліч вигідних пропозицій зі знижками по промокоду, акції та великі розпродажі, завдяки чому можна вигідно купити необхідні товари. На головній сторінці (рис 1.3), так само, як і на офіційному сайті «VISTA» можна знайти навігаційне меню, інформацію про знижки та нові колекції.

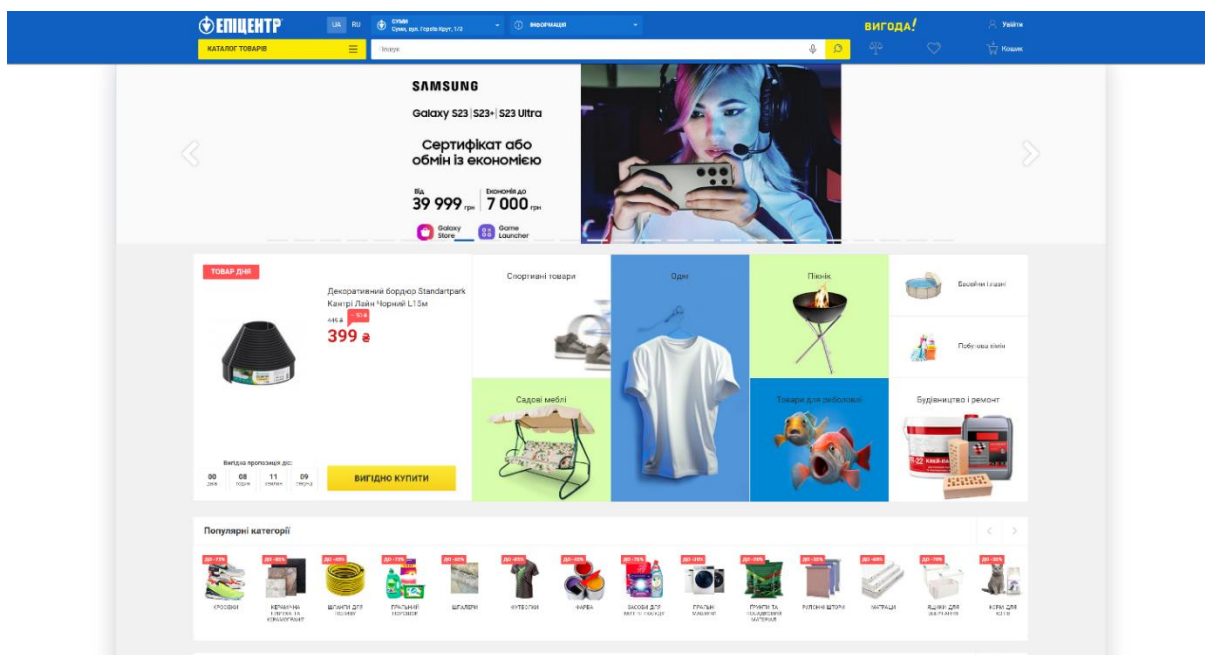


Рисунок 1.3 – Головна сторінка «EPICENTR»

На вебсторінках магазину (рис. 1.4) наявний пошук за допомогою категорій.

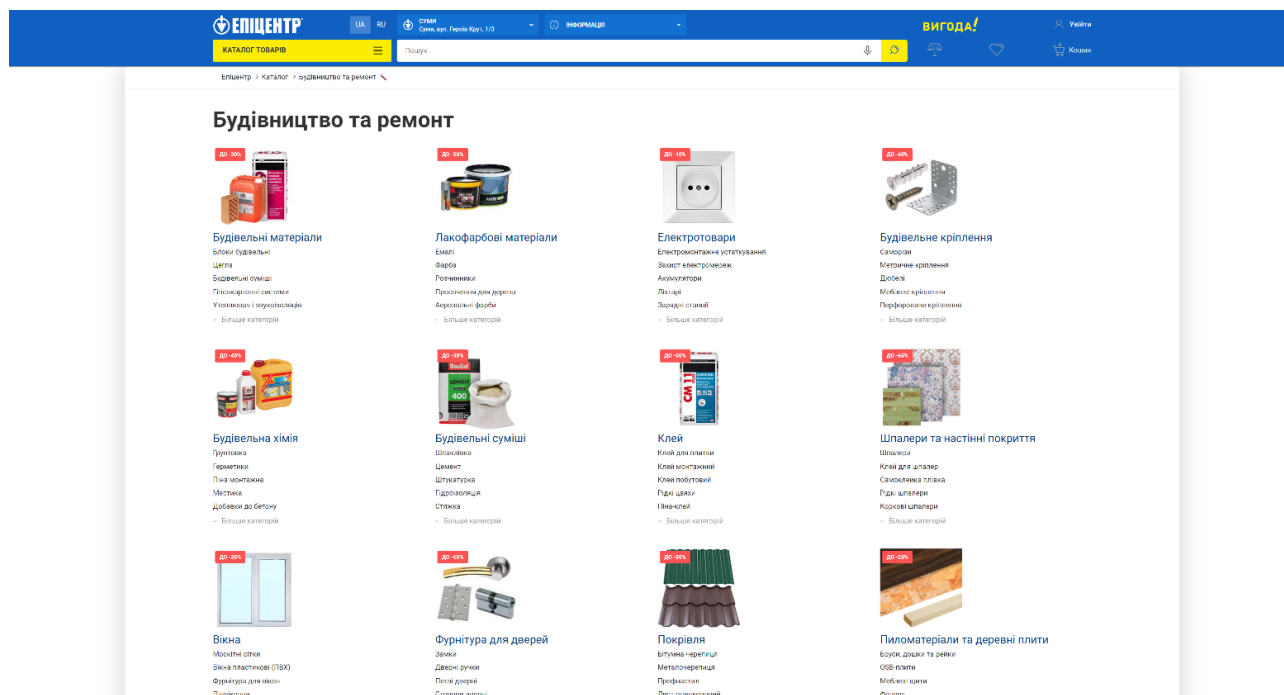


Рисунок 1.4 – Пошук товару за допомогою категорій

Також на вебсайті наявні такі функції як: зміна мови, кошик та власний кабінет з інформацією про замовлення.

Переваги сайту:

- 6 зрозумілий та простий до сприйняття дизайн;
- 7 зрозуміле навігаційне меню.

Недоліки сайту:

- 8 сайт повільно реагує на дії користувача;

1.3 Постановка задачі

Мета даної роботи: розробити інтернет-каталог побутової техніки. В результаті роботи необхідно реалізувати наступні задачі:

- Сформувані вимоги до проекту.
- Вибір та обґрунтування вибір середовища розробки.
- Вибір та обґрунтування технологій розробки для клієнтської та серверної частини.

- Розробити структуру вебсайту.
- Розробити структуру БД.
- Розробити дизайн вебсайту.
- Розробити функціонал.
- Провести тестування

2 ВИБІР МЕТОДІВ РОВ'ЯЗАННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ

2.1 Вимоги до інтернет-каталогу

1. Інтуїтивний інтерфейс: Інтерфейс повинен бути легким у використанні та зрозумілим для користувача. Користувач повинен з легкістю знаходити необхідні товари та категорії.
2. Пошукова система: Наявність потужної системи пошуку, яка дозволяє користувачам швидко знаходити продукти за різними критеріями, такими як модель, виробник, ціновий діапазон тощо.
3. Детальна інформація про товар: Кожен товар повинен мати докладний опис, технічні характеристики, відгуки користувачів, а також фотографії з різних ракурсів.
4. Безпека та конфіденційність: Забезпечення безпеки особистих даних користувачів, а також безпечних транзакцій при покупці товарів.
5. Мобільна сумісність: Сайт повинен бути адаптивним для мобільних пристроїв, щоб користувачі могли зручно переглядати каталог і робити покупки зі своїх смартфонів або планшетів.
6. Можливість збереження вибраних товарів: Функціонал "Обрані товари" для збереження продуктів, які користувачі хочуть придбати, але відкладають на майбутнє.
7. Особистий кабінет для користувачів, можливість зареєструватись або авторизуватись.
8. Додавати товари в «обрані» доступна тільки зареєстрованим користувачам.
9. Усі інформація про користувачів, товари має зберігатись в БД.

2.2 Інструменти розробки вебсайтів

Як середовище розробки було обрано текстовий редактор коду VS Code.

Visual Studio Code (VS Code)[6] — це популярний і розширюваний редактор вихідного коду, розроблений Microsoft. Він створений як легкий, але

потужний інструмент для розробників і широко використовується для різних мов програмування та завдань розробки. Ось деякі ключові функції та аспекти Visual Studio Code:

- Кросплатформенність: VS Code доступний для Windows, macOS і Linux, що робить його універсальним вибором для розробників на різних платформах.

- Open Source: Visual Studio Code є програмним забезпеченням з відкритим вихідним кодом, вихідний код якого доступний на GitHub. Це дозволяє спільноті робити внесок у його розвиток і створювати розширення.

- Багата екосистема: VS Code має величезну екосистему розширень, які забезпечують підтримку різних мов програмування, інструментів і фреймворків. Ці розширення розширюють можливості редактора та роблять його придатним для різних робочих процесів розробки.

- Вбудована інтеграція Git: Visual Studio Code містить вбудовану підтримку Git, що дозволяє розробникам працювати зі сховищами Git безпосередньо з редактора. Це спрощує контроль версій і співпрацю.

- Можливість налаштування: код VS можна налаштувати. Користувачі можуть налаштувати параметри, сполучення клавіш і теми відповідно до своїх уподобань. Ринок розширень пропонує численні теми та розширення для налаштування.

- Інструменти налагодження: VS Code підтримує налагодження для різних мов програмування та платформ. Розробники можуть встановлювати точки зупинки, перевіряти змінні та покроково виконувати код під час сеансів налагодження.

Visual Studio Code — це універсальний інструмент, який обслуговує широкий спектр розробників і сценаріїв розробки, від веб-розробки до науки про дані, розробки ігор тощо. Його популярність пояснюється його простотою, продуктивністю та великими можливостями налаштування, що робить його вибором для багатьох програмістів та інженерів програмного забезпечення.

HTML, аббревіатура від HyperText Markup Language (мова розмітки гіпертексту)[7], є стандартною мовою розмітки для створення веб-сторінок. Вона використовується для створення структури та вигляду веб-сторінок, а також для розміщення різноманітних мультимедійних вмістів, таких як тексти, зображення, відео, аудіо тощо.

HTML базується на концепції тегів, які використовуються для визначення різних елементів веб-сторінки. Кожен тег має відкриваючий та закриваючий тег, і між ними зазвичай розміщується контент. Наприклад, тег `<p>` використовується для визначення абзацу тексту:

```
<p>Це текст абзацу.</p>
```

У HTML також існують беззакривні теги, які не потребують закривання. Наприклад, тег `` для вставки зображень:

```

```

Крім того, HTML дозволяє використовувати атрибути, щоб надати додаткові властивості елементам. Наприклад, атрибут `href` в тезі `<a>` використовується для визначення посилання:

```
<a href="https://www.example.com">Посилання на приклад</a>
```

HTML також дозволяє використовувати CSS (Cascading Style Sheets) для стилізації вигляду веб-сторінок і JavaScript для додавання інтерактивності. В сучасному веб-розробці HTML поєднується з CSS та JavaScript для створення багатофункціональних та стильних веб-сторінок і додатків.

CSS, або Cascading Style Sheets (каскадні таблиці стилів)[8], є мовою, яка використовується для опису вигляду та форматування веб-сторінок, написаних мовою HTML або XML. Вона дозволяє розділити структуру веб-сторінки від її представлення, що робить процес створення та зміни вигляду сторінок більш простим та ефективним.

Основними концепціями CSS є селектори і властивості. Селектори використовуються для вибору елементів HTML, до яких будуть застосовані стилі, а властивості визначають вигляд тих елементів. Наприклад, у CSS можна

задати кольори, розміри шрифтів, відступи, вирівнювання, фонові зображення та багато іншого.

Основні переваги використання CSS включають:

- Розділення змісту та представлення: CSS дозволяє розділити структуру веб-сторінки від її вигляду, що полегшує редагування та управління вмістом.

- Повторне використання коду: Стили, визначені у CSS, можна застосовувати до багатьох сторінок, що зменшує потребу в повторному введенні коду та сприяє консистентності вигляду.

- Зручність управління: CSS дозволяє легко змінювати вигляд всіх елементів на сторінці, змінюючи лише один файл CSS, що робить процес редагування та оновлення більш простим.

- Покращення доступності: CSS надає більше можливостей для поліпшення доступності веб-сторінок, таких як контроль за розміром шрифтів, кольорами та іншими аспектами дизайну.

В сучасному веб-дизайні CSS використовується разом з HTML та JavaScript для створення багатофункціональних та стильних веб-сторінок та додатків.

JavaScript (JS)[9] - це високорівнева, інтерпретована мова програмування, яка використовується для розробки веб-додатків та веб-сторінок. Вона є однією з трьох основних технологій веб-розробки, разом з HTML та CSS. JavaScript використовується для створення динамічного контенту, взаємодії користувача з веб-сторінкою, анімації, обробки подій та багатьох інших речей.

Основні характеристики JavaScript включають:

- Інтерактивність: JavaScript дозволяє створювати веб-сторінки, які реагують на дії користувача. Наприклад, ви можете створити валідатор форм, який перевіряє правильність введених даних до відправки форми[10].

- Динамічний контент: За допомогою JavaScript можна динамічно змінювати вміст веб-сторінок без перезавантаження сторінки. Це дозволяє

створювати багатофункціональні веб-додатки, які працюють швидше та ефективніше.

- Маніпуляція DOM: JavaScript надає доступ до Document Object Model (DOM), що представляє структуру HTML-документа. За допомогою JavaScript можна додавати, видаляти та змінювати елементи сторінки, що дозволяє динамічно змінювати її вміст.

- Асинхронність: JavaScript підтримує асинхронне програмування, що дозволяє виконувати деякі операції без очікування завершення попередніх операцій. Це особливо важливо для виконання мережових запитів та роботи з асинхронними подіями, такими як клік користувача чи завантаження ресурсів.

- Підтримка багатьох платформ: JavaScript підтримується практично всіма сучасними веб-браузерами і може використовуватися як на клієнтській, так і на серверній стороні (за допомогою Node.js)[11].

JavaScript є надзвичайно потужним і гнучким інструментом, який забезпечує безліч можливостей для розробки веб-додатків та веб-сайтів різного рівня складності. Він використовується в широкому спектрі застосувань, від простих анімацій до складних односторінкових додатків.

PHP (гіпертекстовий препроцесор)[12] — це популярна серверна мова сценаріїв, яка в основному використовується для веб-розробки. Він відомий своєю простотою, гнучкістю та широким застосуванням у створенні динамічних веб-додатків. Ось деякі ключові аспекти та особливості PHP:

- Сценарії на стороні сервера: PHP — це мова сценаріїв на стороні сервера, що означає, що вона виконується на веб-сервері, а не в браузері користувача. Він зазвичай використовується для створення динамічних веб-сторінок шляхом обробки даних із баз даних, надсилання форм тощо.

- Можливість вбудовування в HTML: PHP-код можна вбудовувати безпосередньо в HTML-файли, що полегшує змішування серверної логіки з розміткою HTML. Код PHP укладено в теги `<?php i ?>`.

– Відкритий вихідний код: PHP є відкритим вихідним кодом і вільно доступний для використання, зміни та розповсюдження. Це сприяло його широкому застосуванню та великій спільноті розробників.

– Кросплатформенність: PHP працює на різних операційних системах, включаючи Windows, macOS, Linux і Unix, що робить його універсальним і доступним для розробників на різних платформах.

– Велика стандартна бібліотека: PHP постачається з великою стандартною бібліотекою функцій і розширень, які спрощують типові завдання веб-розробки, такі як робота з базами даних, обробка HTTP-запитів і обробка рядків.

– Безпека: PHP має функції безпеки та функції, які допомагають розробникам захистити свої веб-додатки від поширених уразливостей, таких як впровадження SQL і міжсайтовий сценарій (XSS).

– Продуктивність: в останніх версіях PHP значно підвищив продуктивність, і за правильного використання він може ефективно обробляти велику кількість веб-запитів. Кешування коду операції та інші методи оптимізації можуть ще більше підвищити продуктивність PHP.

– Сумісність: PHP має зворотну сумісність, тобто старіший код PHP зазвичай може працювати на новіших версіях мови, що зменшує ризик поломки існуючих програм під час оновлення.

– Веб-хостинг: PHP широко підтримується постачальниками веб-хостингу, що полегшує розгортання програм PHP на різних платформах хостингу.

PHP зазвичай використовується для створення широкого діапазону веб-додатків, від простих веб-сайтів до складних систем керування вмістом (CMS), платформ електронної комерції, сайтів соціальних мереж тощо. Простота використання та широка підтримка спільноти роблять його популярним вибором для веб-розробників у всьому світі [13]. Однак важливо дотримуватися найкращих практик і вказівок щодо безпеки під час розробки додатків PHP, щоб переконатися, що вони безпечні та придатні для обслуговування.

MySQL — це система управління реляційною базою даних (RDBMS) з відкритим кодом, яка широко використовується для керування та зберігання структурованих даних. Вона відома своєю швидкістю, надійністю та простотою використання, що робить її однією з найпопулярніших систем баз даних у світі. Ось деякі ключові аспекти та особливості MySQL[14]:

- Відкритий вихідний код: MySQL є програмним забезпеченням з відкритим вихідним кодом, що означає, що воно вільно доступне для використання, модифікації та розповсюдження. Це сприяло його широкому застосуванню та великій спільноті користувачів і розробників.

- Реляційна база даних: MySQL — це система керування реляційною базою даних, що означає, що вона зберігає дані в таблицях із рядками та стовпцями. Це дозволяє розробникам визначати зв'язки між таблицями та виконувати складні запити за допомогою мови структурованих запитів (SQL).

- Кросплатформенність: MySQL доступний у різних операційних системах, включаючи Windows, macOS, Linux і Unix. Ця крос-платформна сумісність робить його придатним для широкого діапазону середовищ розробки.

- Масштабованість: MySQL є масштабованим, що означає, що він може обробляти як маломасштабні, так і великомасштабні програми. Він підтримує такі функції, як реплікація та кластеризація для створення високодоступних і масштабованих архітектур баз даних.

- Продуктивність: MySQL відома своїми можливостями швидкого пошуку та обробки даних. Він включає такі функції, як індексування, оптимізація запитів і кешування для підвищення продуктивності бази даних.

- Безпека даних: MySQL забезпечує надійні функції безпеки, включаючи автентифікацію користувачів, контроль доступу та шифрування. Це дозволяє адміністраторам баз даних визначати точні привілеї доступу для користувачів і обмежувати несанкціонований доступ до даних [15]/

MySQL широко використовується у веб-розробці, системах керування контентом (CMS), платформах електронної комерції, програмах, керованих

даними, тощо. Це універсальний і надійний вибір для зберігання та керування структурованими даними, а його комбінація з PHP особливо популярна для створення динамічних веб-додатків.

phpMyAdmin — це безкоштовний веб-інструмент із відкритим кодом для керування базами даних MySQL і MariaDB. Він забезпечує зручний графічний інтерфейс для виконання різноманітних завдань керування базами даних без використання командного рядка [16]. Ось деякі ключові аспекти та функції phpMyAdmin:

- Веб-інтерфейс: доступ до phpMyAdmin здійснюється через веб-браузер, що дозволяє вам керувати своїми базами даних з будь-якого місця, де є підключення до Інтернету. Завдяки цьому адміністраторам і розробникам зручно працювати з базами даних віддалено.

- Управління базами даних: за допомогою phpMyAdmin ви можете виконувати основні завдання керування базами даних, такі як створення, видалення та перейменування баз даних, а також створення, зміна та видалення таблиць.

- Маніпулювання даними: забезпечує зручний інтерфейс для вставки, оновлення та видалення записів у таблицях бази даних. Ви також можете виконувати SQL-запити безпосередньо через інтерфейс.

- Імпорт і експорт: phpMyAdmin дозволяє імпортувати та експортувати дані в різних форматах, включаючи SQL, CSV, JSON і XML. Це корисно для резервного копіювання та перенесення вмісту бази даних.

- Керування користувачами: ви можете створювати облікові записи користувачів і керувати ними, призначати привілеї та контролювати доступ до певних баз даних і таблиць. Ця функція має вирішальне значення для підтримки безпеки бази даних [17].

- Редактор запитів SQL: phpMyAdmin містить вбудований редактор запитів SQL із підсвічуванням синтаксису та доповненням коду, що полегшує написання та виконання складних запитів до бази даних.

phpMyAdmin — це цінний інструмент для адміністраторів баз даних, розробників і всіх, кому потрібно взаємодіяти з базами даних MySQL або MariaDB. Це спрощує завдання керування базами даних і забезпечує інтуїтивно зрозумілий спосіб роботи з реляційними базами даних. Багато провайдерів веб-хостингу включають phpMyAdmin як частину своїх пакетів хостингу, що полегшує користувачам керування своїми базами даних без встановлення додаткового програмного забезпечення.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Визначення структури додатка

Під час проведення аналізу аналогів інтернет-магазинів побутових приладів, стало зрозуміло уявлення майбутнього продукту, його структуру та дизайн. Для кожного вебсайту існує своя структура, яка включає, до себе, ряд вебсторінок, перехід між якими відбувається за рахунок гіперпосилань. Якісно розроблена структура вебсайту надає користувачу можливість без технічних перешкод користуватися вебсайтом.

Структура вебсайту - це організована система розташування та взаємозв'язку його сторінок, контенту та функціональних елементів. Вона визначає, які сторінки доступні на сайті, як вони пов'язані між собою та як користувачі можуть переміщатися по сайту.

Структура вебсайту зазвичай представляє собою деревоподібну ієрархію сторінок(рис. 2.1). На вершині ієрархії знаходиться головна сторінка (home page), звідки можна перейти до різних розділів або підсторінок. Розділи можуть мати свої підрозділи, що дозволяє поділити контент на логічні частини.



Рисунок 3.1 – Структура інтернет-магазину будівельних товарів

Крім ієрархічної структури, вебсайт може мати також глобальні елементи навігації, які дозволяють користувачам легко переходити між різними сторінками незалежно від їх місця в ієрархії. Це можуть бути меню, пошукові поля, посилання на популярні сторінки тощо.

Структура вебсайту визначається розробниками та дизайнерами з урахуванням потреб користувачів, логіки сайту та організації контенту. Вона має бути логічною, зрозумілою та зручною для навігації, щоб користувачі могли легко знайти необхідну інформацію та взаємодіяти з сайтом.

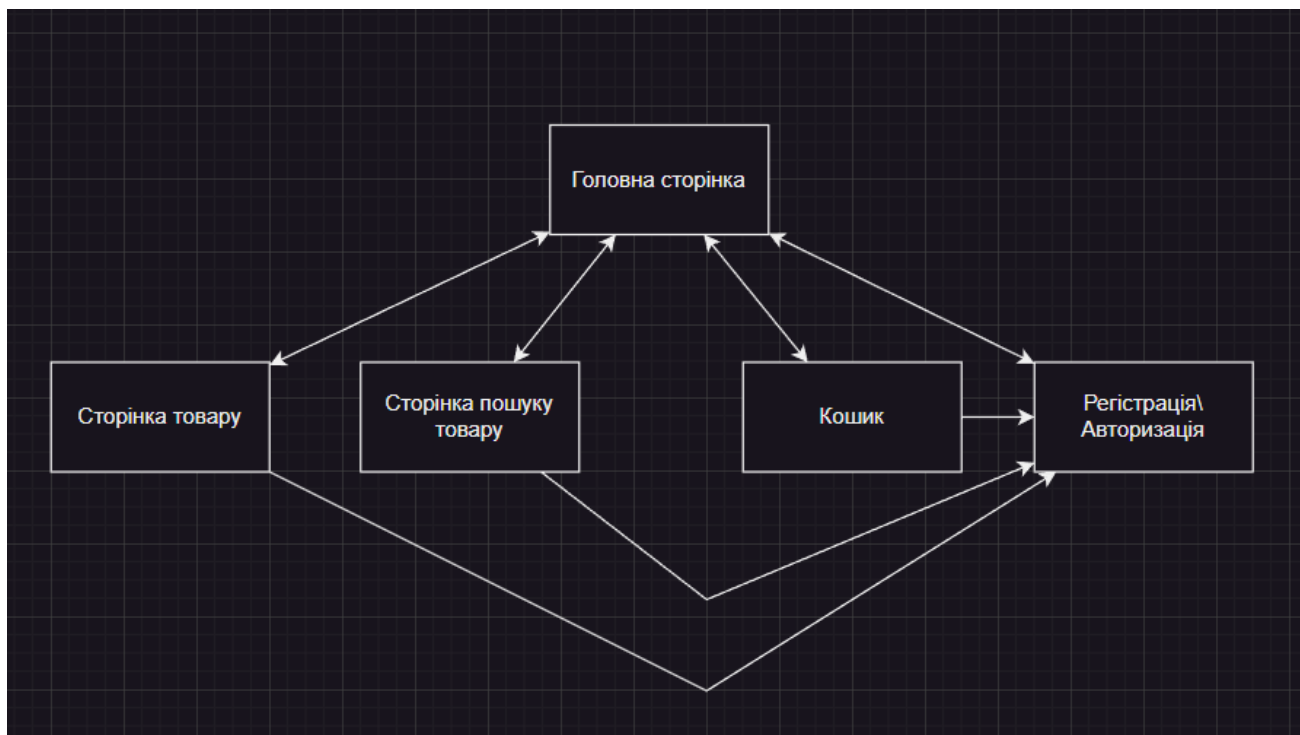


Рисунок 3.2 – Навігаційна мапа інтернет-магазину побутових товарів

З рисунка 2.1 можна зробити певні висновки, а саме, що для користувача доступні 5 вебсторінок, одна з яких доступна тільки після авторизації на вебсайті.

Структура самих вебсторінок складається з наступних блоків: шапка та підвал вебсайту, для усіх вебсторінок, та відрізняючи блоків. На рисунках 2.3-2.7 зображена структура вебсторінок даного інтернет-магазину.

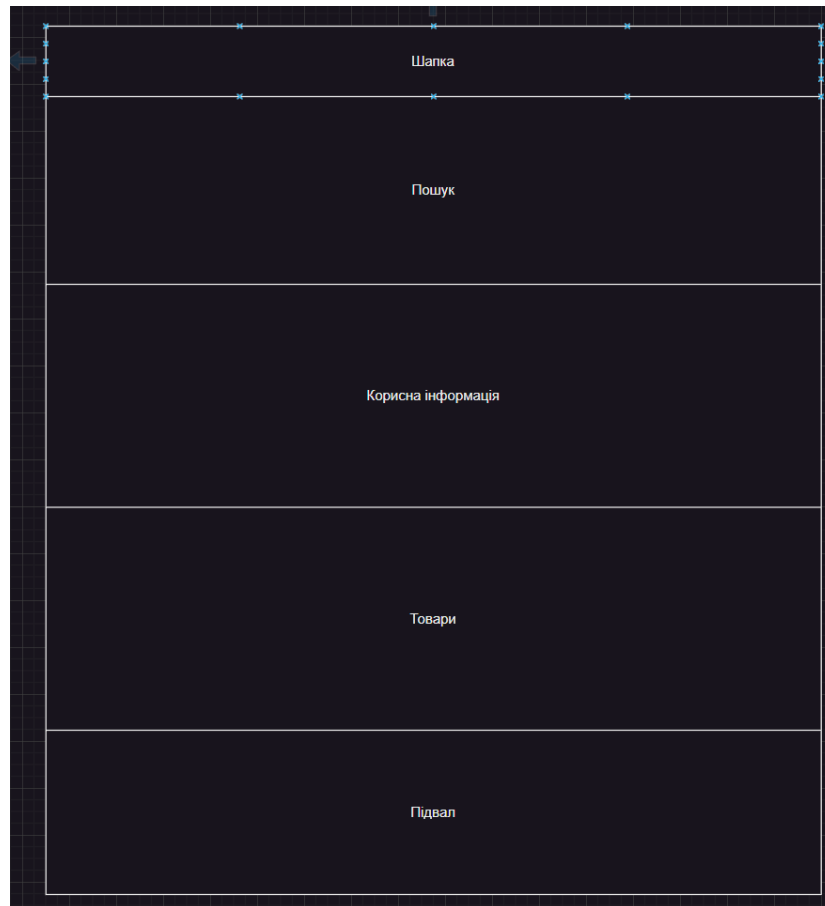


Рисунок 3.3 – Структура головної сторінки

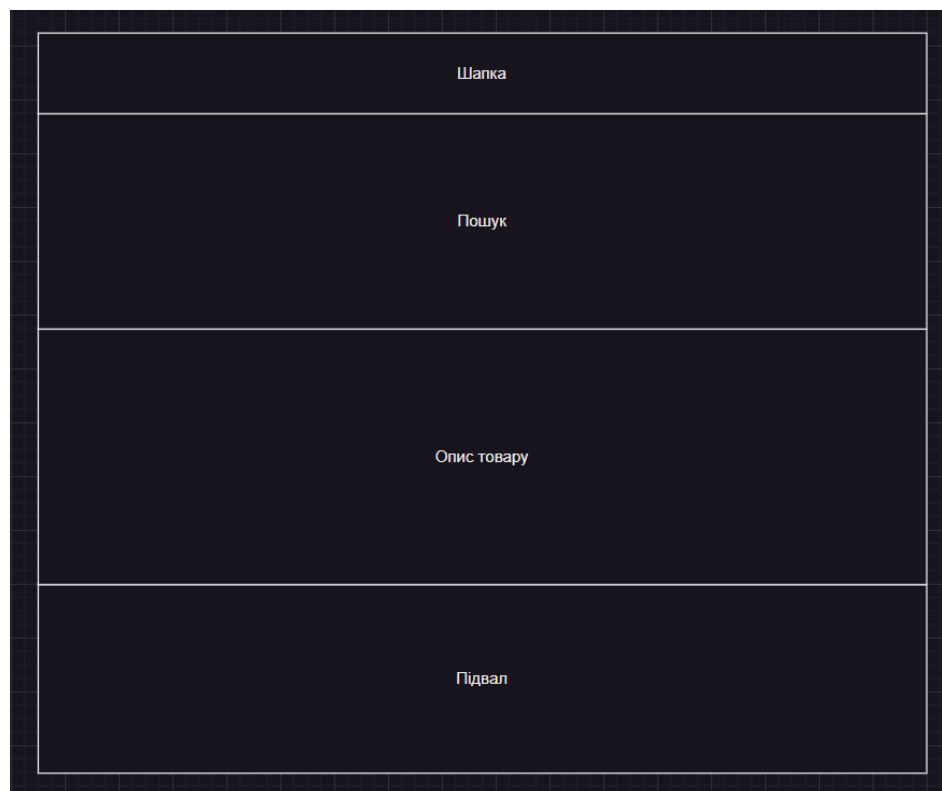


Рисунок 3.4 – Структура вебсторінки опису товару

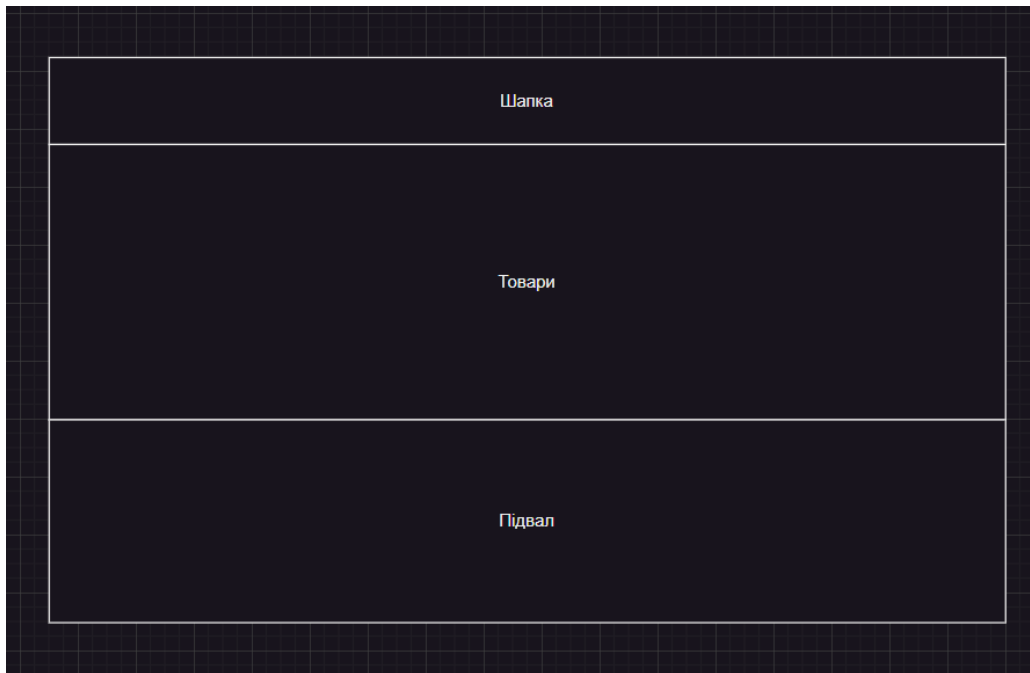


Рисунок 3.5 – Структура вебсторінки пошуку товару

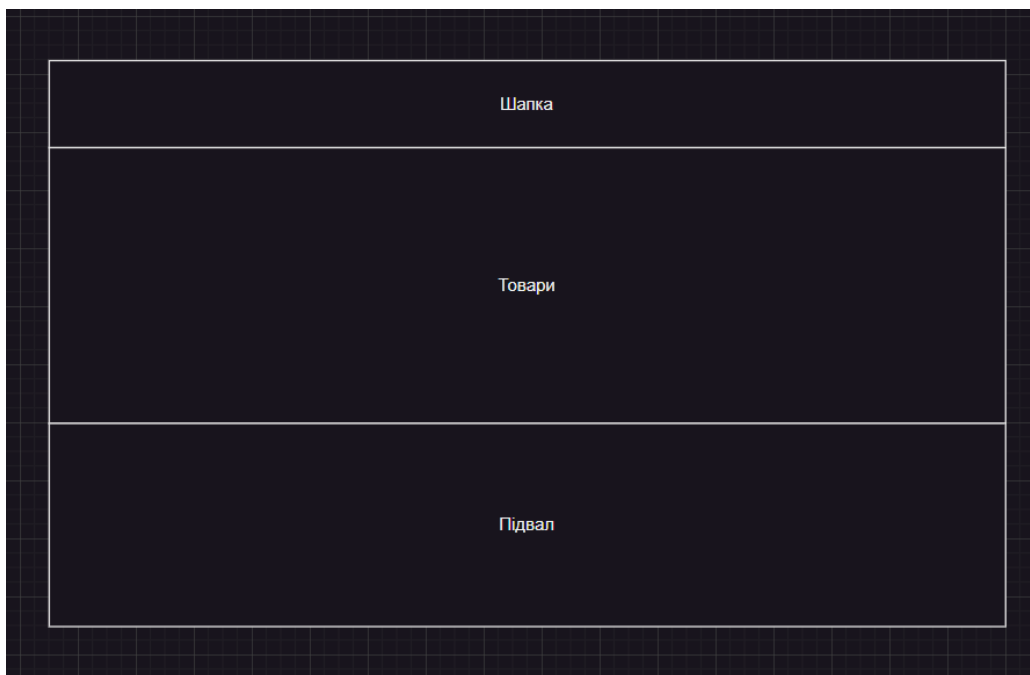


Рисунок 3.6 – Структура вебсторінки кошика



Рисунок 3.7 – Структура вебсторінки реєстрації\авторизації

3.3 Структура бази даних

Виходячи з вимог була розроблена БД, концептуальна модель зображена на рисунку 3.1.

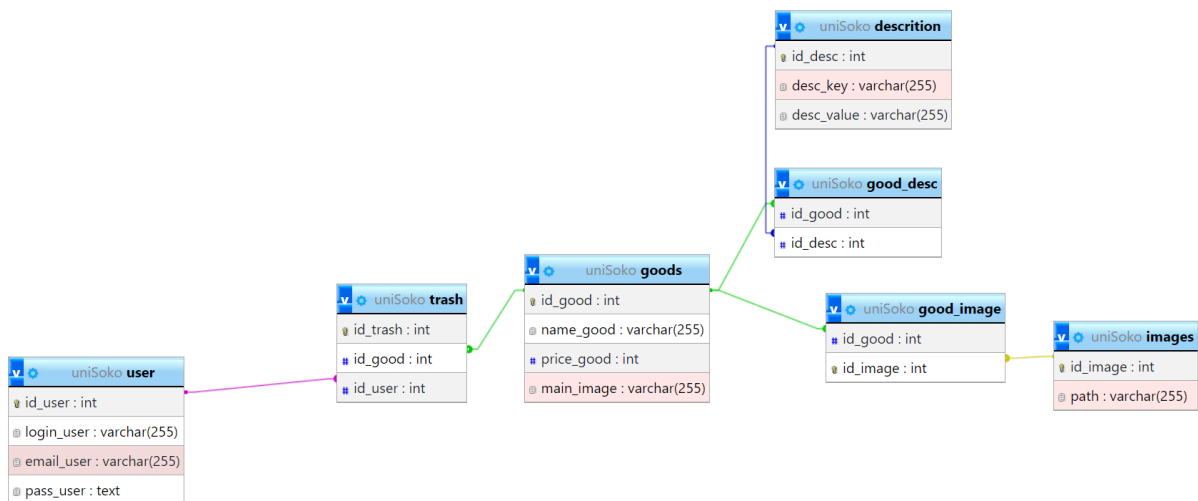


Рисунок 3.8 – Концептуальна модель БД

Дана БД складається з семи таблиць, а саме:

«User» - зберігається інформація про користувачів

«Trash» - корзина товарів

«Goods» - зберігається інформація про товари

«Description» - зберігається опис товарів

«goods_desc» - таблиця містить інформацію якому товару відповідає який опис(дане рішення було обране через те, що у багатьох товарів можуть бути однакові характеристики)

«goods_image» - таблиця містить інформацію якому товару відповідає яке зображення.

«images» - містить перелік усіх зображень.

Зв'язок між таблицями:

«User» - «Trash» - багато до одного.

«Trash» - «Goods» - багато до одного.

З вище описаного слідує, що зв'язок між таблицями «User» - «Goods» - багато до багатьох.

«Goods» - «goods_desc» - багато до одного.

«goods_desc» - «Description» - багато до одного.

З вище описаного слідує, що зв'язок між таблицями «Goods» - «Description» - багато до багатьох.

«goods_image» - «Goods» - багато до одного.

«goods_image» - «images» - багато до одного.

З вище описаного слідує, що зв'язок між таблицями «Goods» - «images» - багато до багатьох.

Після створення БД , можна переходити до створення коду, який буде взаємодія ти з БД та викликатись при певних діях користувача, наприклад при реєстрації користувача або додаванні товару.

3.3 Розробка дизайну

Дизайн веб-сайту - це процес створення та організації вигляду, оформлення та взаємодії елементів на веб-сторінках. Це включає в себе вибір

кольорів, шрифтів, графіки, компоновання елементів, стиль інтерфейсу та інші аспекти, що стосуються зовнішнього вигляду та способу презентації контенту на веб-сайті.

Дизайн веб-сайту має на меті забезпечити естетичний, привабливий та функціональний вигляд сайту, а також полегшити взаємодію користувача з ним. Він допомагає встановити ідентичність та брендування сайту, створити зручну навігацію, покращити читабельність контенту та забезпечити гармонійну композицію сторінки.

Дизайн веб-сайту також може включати в себе адаптивний або респонсивний дизайн, що означає його адаптацію до різних пристроїв та розмірів екранів. Це дозволяє сайту оптимально відобразитися на комп'ютерах, планшетах, смартфонах та інших пристроях.

Веб-дизайнер використовує різноманітні інструменти та технології, такі як графічні редактори, HTML/CSS, JavaScript, а також розуміння принципів веб-дизайну та взаємодії, для створення зовнішнього вигляду веб-сайту. Важливим аспектом дизайну веб-сайту є його спрямованість на користувачів, їхні потреби та взаємодію з сайтом.

Вебсайт був розроблений у єдиному стилі, основними кольорами стали: білий червоний та чорний. Для оформлення шрифтів використовувались наступні теми: Inter, Roboto, Montserrat Alternates.

Шапка вебсайту(рис. 2.9) представляє собою декілька блоків, які мають різне призначення. Логотип складається з тематичної назви компанії. Блок з кнопкою авторизацією\регістрацією або з кнопкою виходу, якщо користувач вже увійшов в свій аккаунт. Блок з кошиком, цифра якого вказує скільки даний користувач додав товарів, доступна тільки після авторизації. Розташування блоків зроблено за допомогою властивості flex. Фон даного блоку зроблений за допомогою градієнту.

Приклад створення градієнта:

```
background: linear-gradient(1.02deg, #fc0b0b6f -17.21%, rgba(74, 153, 211, 0) 75.91%), #FFFFFF;
```

Рисунок 3.9 – Шапка вебсайту

Підвал вебсайту (рис. 2.10) – містить опис компанії та інформацію для клієнта, правила користування інформацією на вебсайті.

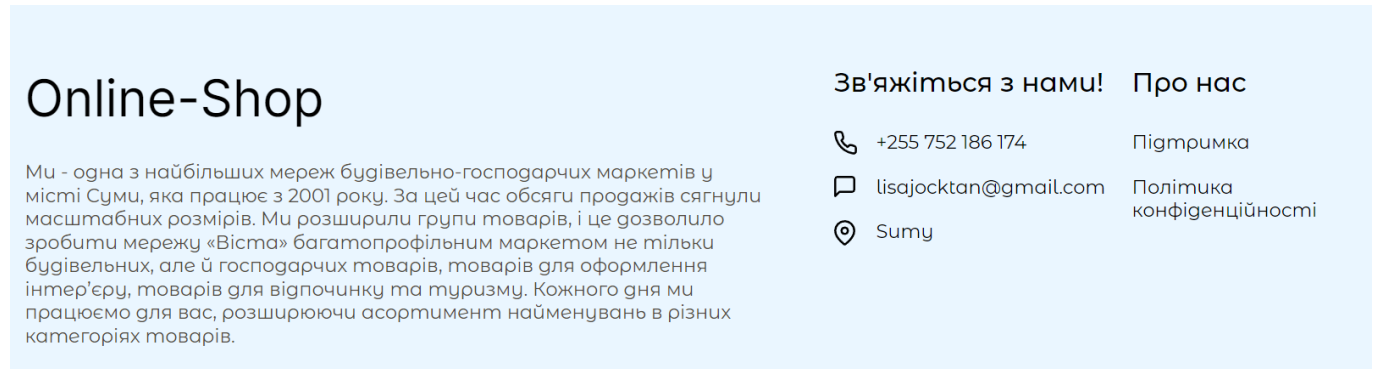


Рисунок 3.10 – Підвал вебсайту

Пошук товару (рис. 2.11) – поле пошуку це форма з полем вводу input типу text, після того як користувач введе символ або набір символів та натисне клавішу Enter, відбудеться переадресація на іншу вебсторінку. Приклад запити до БД з пошуком товару:

```
SELECT * FROM `goods` WHERE name_good like '%$search' OR name_good like '$search%' OR name_good like '$search' OR name_good like '%$search%';
```

Де \$search – значення поля вводу, яке передається за допомогою метода GET.

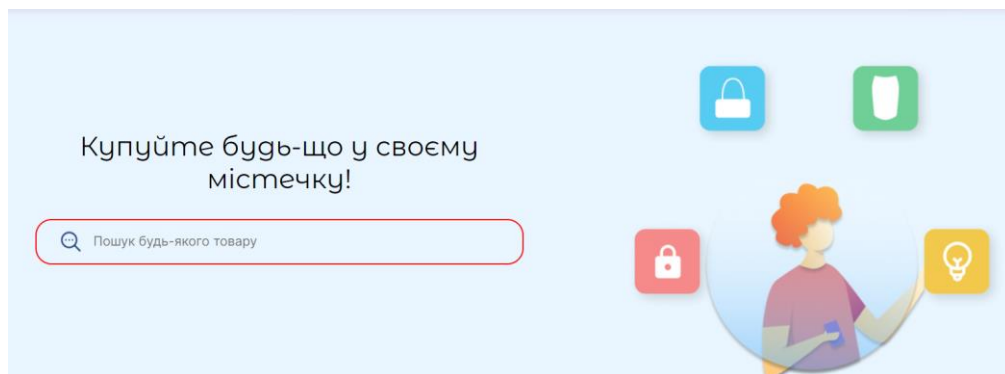


Рисунок 3.11 – Блок пошуку товару

Блок з формацією для клієнта(рис. 2.12) та блок з товарами (рис. 2.13) – даний блок, як було сказано вище було реалізована за допомогою властивості flex.

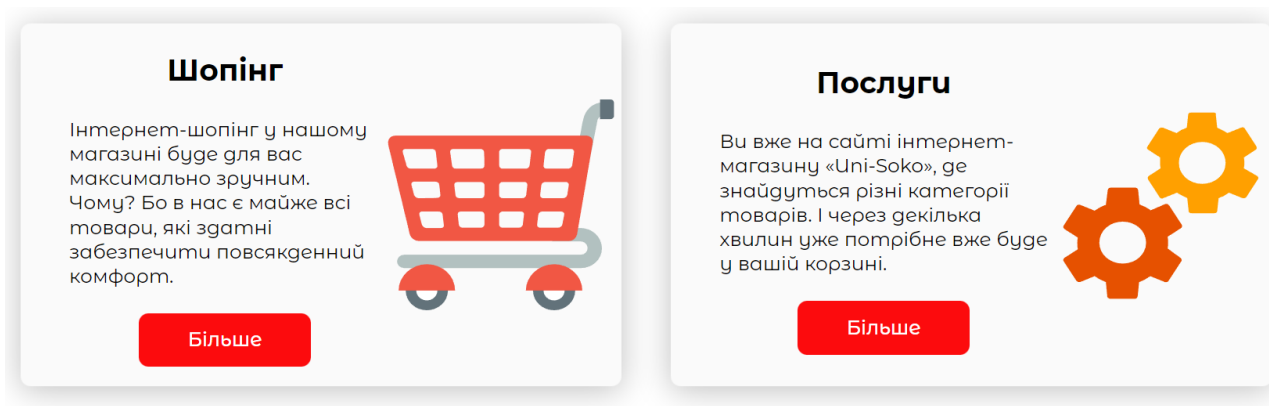


Рисунок 3.12 – Блок з формацією для клієнта

Товари

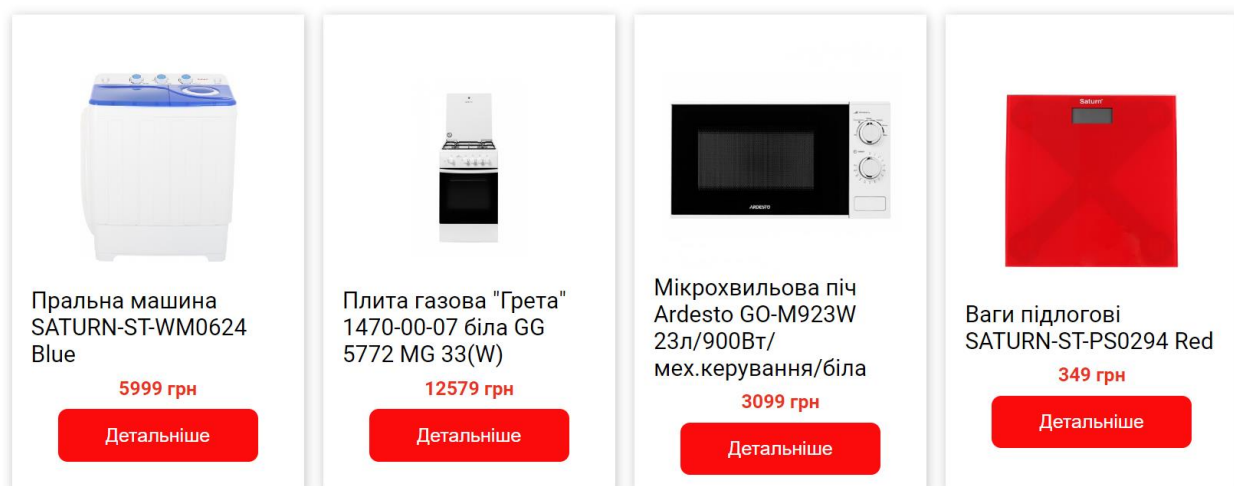



Рисунок 3.13 – Блок з формацією для клієнта

Вебсторінка з описом товару складається як і всі вебсторінки даного вебресурсу з шапки, підвалу вебсайту але корім їх на ній ще присутні два блок це блок з пошуком товару та блок з описом товару(рис. 2.14) – на ньому присутній слайдер та опис товару з можливістю придбати товар та зателефонувати.

Опис товару



Пральна машина SATURN-ST-WM0624 Blue


Вага:	22 кг
Гарантія:	24 місяці
Колір:	Білий з синім
Матеріал:	Пластик
Рівень шуму:	80 гБ

Подзвонити
Придбати

Рисунок 3.14 – Блок з інформацією про товар

Вебсторінка пошуку товару складається з одного додаткового блоку, за основу якого був узятий блок з головної сторінки з переліком існуючих товарів(рис. 2.13), даний блок зображений на рисунку 2.15.

Результати пошуку за: га



Плита газова "Грета" 1470-00-07 біла GG 5772 MG 33(W)

12579 грн

Детальніше

Рисунок 3.15 – Блок з шуканим товаром

Блок з переліком товарів, які були занесені до кошика(рис. 2.16), з вебсторінки кошик – за основу бере блок з головної сторінки який зображений на рисунку 2.13.

Товари

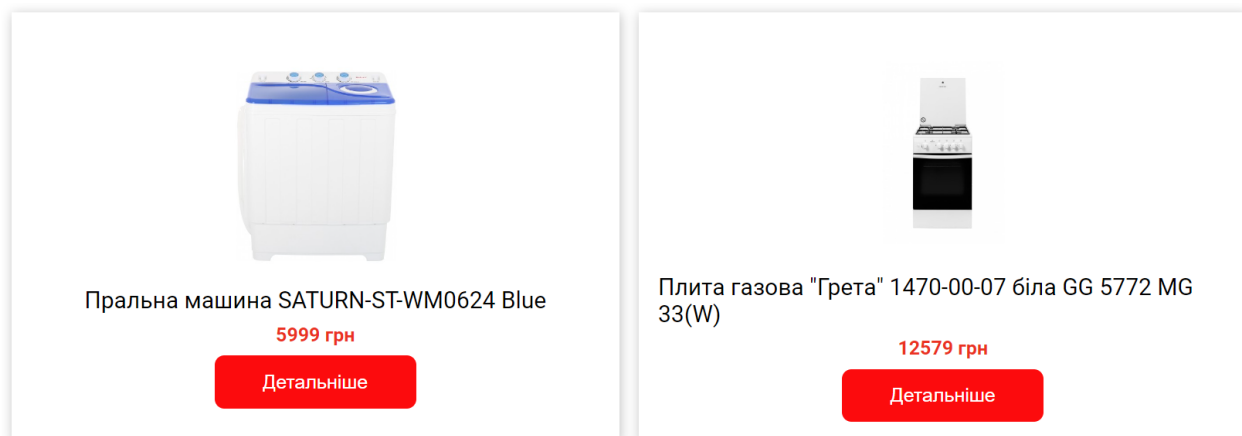


Рисунок 3.16 – Блок з вебсторінки кошик

Даний блок реєстрації\авторизації(рис. 2.17) надає користувачеві зареєструватися або авторизуватися на даному вебсайті.

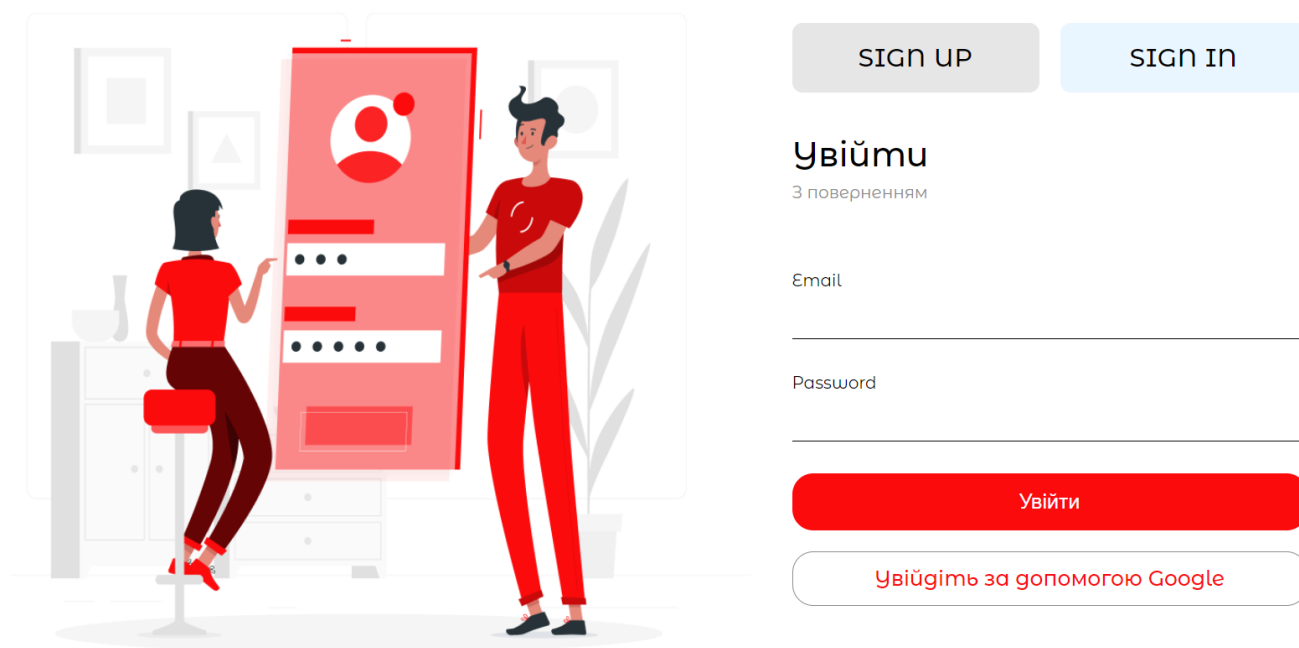


Рисунок 3.17 – Блок реєстрації\авторизації

Увесь код, описаних вище елементів та сценаріїв наведений у відповідних додатках: додаток А - гіпертекстова розмітка(HTML та PHP), додаток Б - таблиці каскадних стилів(CSS), додаток В - сценарії (JavaScript).

3.3 Розробка функціоналу системи

Для блоку реєстрації\авторизації та для усіх кнопок на вебсайті застосований псевдоклас `hover`, який при наведенні курсора миші на блок збільшує його. Приклад на рисунках 2.18 - 2.19.

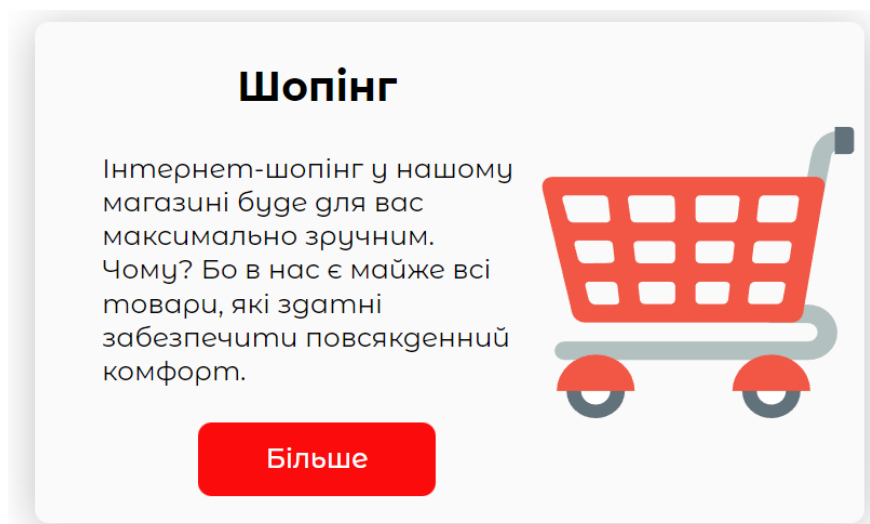


Рисунок 3.18 – Кнопка в стані спокою

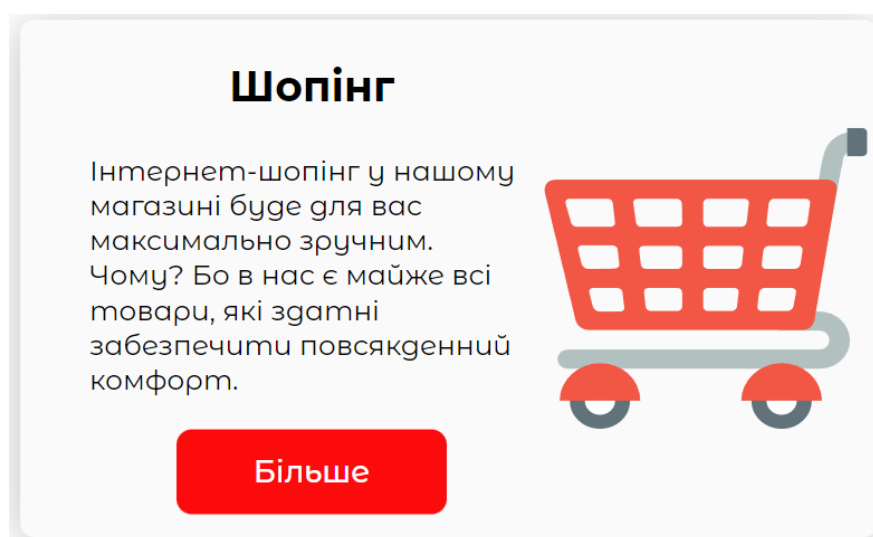


Рисунок 3.19 – Кнопка при наведенні

Пошук товару – реалізований за допомогою запиту до БД, який виводить усі товари у назві яких був знайдений певний символ, слово або набір слів. Важливо звернути увагу на таблицю кодування ASCII, а саме на мову введення символів в поле вводу для пошуку товарів.

Регістрація\авторизація – у даних функцій дуже схожа робота, після введення користувачем даних, в обох випадках дані різні, відбувається переадресація на системні файли create_user\signin відповідно, це програмні файли в них немає фронтенд частини, тому користувач не помічає дану переадресацію. В даних файлах прописана валідація даних які були передані, при неуспішній валідації – відбудеться переадресація на попередню вебсторінку та виведеться відповідна помилка(рис. 2.20), якщо валідація пройшла успішно то в першому випадку дані заносяться до таблиці user, в другому випадку створюється сесія user з одним полем id_user, яке містить в собі порядковий номер користувача в таблиці. Після чого відбувається переадресація на головну сторінку.

The image shows a login interface. At the top, there are two buttons: 'SIGN UP' (grey) and 'SIGN IN' (light blue). Below them is the heading 'Увійти' (Login) with the subtitle 'З поверненням' (With return). There are two input fields: 'Email' and 'Password'. Below the password field, a red-bordered box contains the error message: 'Не вірно вказаний логін або пароль.' (Incorrect login or password). Below this is a red 'Увійти' (Login) button and a button for 'Увійдіть за допомогою Google' (Login with Google).

Рисунок 3.20 – Приклад непройденної валідації

Для додавання товару у кошик спочатку необхідно авторизуватися, потім натиснути на відповідну кнопку на карточці товару(рис. 2.21), зелений полюс в верхньому правому куту. Після натиснення на дану кнопку відбудеться

додавання товару в кошик, а саме відповідних порядкових номерів товару та користувача в таблицю в БД.

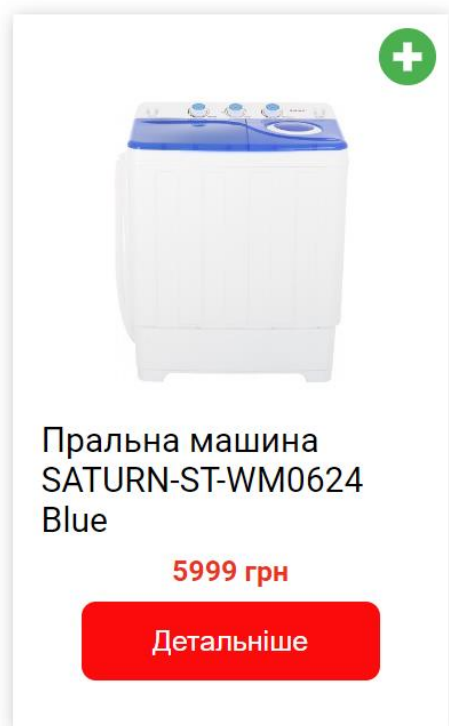
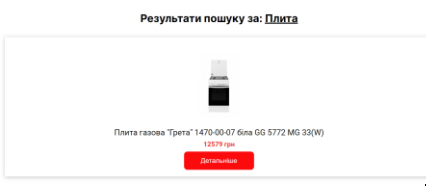
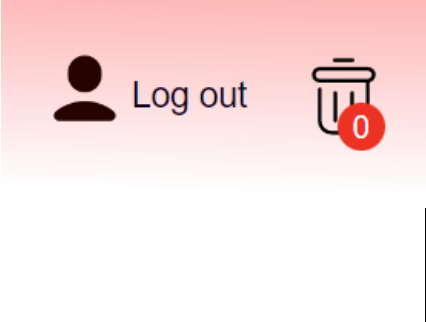
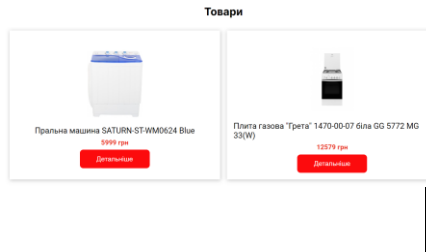
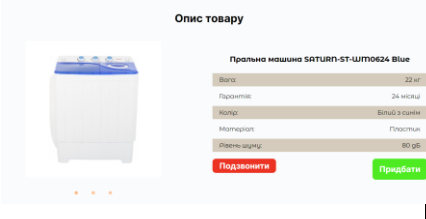


Рисунок 3.21 – Кнопка додавання товару

3.4 Тестування

Тестування - це фінальний крок розробки будь-якого продукту. Тестування даного програмного продукту відбувається методом білого ящика. Відповідні тести занесені у таблиці 2.1

Таблиця 2.1 - Тестування вебзастосунку

№	Умова, яка тестується	Очікуваний результат	Фактичний результат	1/0
1	Пошук товару	Переадресація та вивід усіх товарів з співпадінням у назві.		1
2	Регістрація аккаунта	Проведення валідації та занесення даних у таблиці з подальшою можливістю авторизації		1
3	Додавання товару у кошик	Додавання товару у кошик з можливістю перегляду доданих товарів		1
4	Перегляд обраного товару	Переадресація на вебсторінку та вивід даних з БД.		1

ВИСНОВКИ

Під час виконання даної роботи, було проведено аналіз принципів побудови ІС. Після отримання знань, основних характеристик, інтернет-каталогів, було проведено аналіз аналогів схожих за тематикою інтернет-каталогів. Сформовано задачі на дану роботу.

Після проведення аналізу принципів побудови та аналогічних проєктів, на основі даної інформації, було сформовано вимоги до майбутнього проєкту. На основі даних вимог були обрана інструменти та технології розробки інтернет-каталогу.

Першим кроком розробки, стало визначення структури додатку, на його основі було розроблено БД проєкту. Після чого почався процес розробки дизайну проєкту та його верстка. Після створення розмітки та додавання стилів, був реалізовано функціонал. В самому кінці розробки праведно тестування програмного продукту.

В результаті даної роботи було створено інтернет-каталог побутової техніки, який повністю відповідає вимогам та сучасним тенденціям у веброботці. Програмний продукт було протестовано, тестування недоліків не виявило.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. The Good and the Bad of Vue.js Framework Programming [Електронний ресурс] – Режим доступу до ресурсу: <https://www.altexsoft.com/blog/engineering/pros-and-cons-of-vue-js/>
2. Traditional and Digital Catalogues [Електронний ресурс] – Режим доступу до ресурсу: <https://www.bbpress.co.uk/news/pros-and-cons-of-traditional-and-digital-catalogues>
3. What Is A User Interface, And What Are The Elements That Comprise one? [Електронний ресурс] – Режим доступу до ресурсу: <https://careerfoundry.com/en/blog/ui-design/what-is-a-user-interface/>
4. What Is HTML? Hypertext Markup Language Basics Explained [Електронний ресурс] – Режим доступу до ресурсу: <https://www.hostinger.com/tutorials/what-is-html>
5. Hotline [Електронний ресурс] - Режим доступу до ресурсу: <https://hotline.ua/sport-igrovye-myachi/gala-bora-10-bv5671sc/>
6. Slam [Електронний ресурс] - Режим доступу до ресурсу: <https://slam.city/ru/mode/item-9471.html>
7. Epicentrk [Електронний ресурс] - Режим доступу до ресурсу: <https://epicentrk.ua/ua/shop/futbolnyy-myach-nike-flight-dc1496-100-r-5.html>
8. Azteca [Електронний ресурс] - Режим доступу до ресурсу: <https://azteca.com.ua/ua/about-us/>
9. Nicepage [Електронний ресурс] - Режим доступу до ресурсу: <https://nicepage.com/>
10. Donsport [Електронний ресурс] - Режим доступу до ресурсу: <https://donsport.ru/blog/kakim-sportom-zanyatsya-rebenku/>
11. Sethi R. Software Engineering: Basic Principles and Best Practices - Cambridge: Cambridge University Press, 2023. - 361 p.
12. Montagne Euripides. Systems Software: Essential Concepts. - Cognella Academic Publishing, 2022. — 228 p.

13. Tsui F., Karam O., Bernal B. Fundamentals Of Software Engineering. - 5th Edition. — Jones & Bartlett Publishers, 2022. — 450 p.
14. Miller Brian D., Ackerman Jason. Principles of Web Design. - 3rd Edition. — Allworth, 2022. — 280 p.
15. Smashing Magazine. Psychology of Web Design.- Smashing Media GmbH, 2021. — 271 p.
16. Abelson H., Sussman G.J., Henz M., Wrigstad T., Sussman J. Structure and Interpretation of Computer Programs: JavaScript Edition. - The MIT Press, 2022. - 640 p.
17. Adeolu O. Essential Guide to PHP for All Levels.- Young M. Computer Guru Institute, 2024. — 171.
18. Cao J. E-commerce Big Data Mining and Analytics. — Springer, 2023. — 368 p.
19. Martínez-López Francisco J. (ed.). Advances in Digital Marketing and eCommerce: Fourth International Conference, 2023 . — Springer, 2023. — 314 p.
20. Shuai Qinghong, Li Zhongjun. E-Commerce Industry Chain: Theory and Practice . — Springer, 2023. — 384 p.

Додаток А Код html

```
<?php
session_start();
?>
<!DOCTYPE html>
<html lang="ua">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="/css/style.css">
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link
href="https://fonts.googleapis.com/css2?family=Inter:wght@400;500;700;900&family=Montserrat+Alternates:wght@400;500;700;900&family=Roboto:wght@400;500;700;900&display=swap" rel="stylesheet">
  <link rel="icon" type="image/png" href="/img/misc/shopping-cart.svg">
  <title>Uni-Soko</title>
</head>

<body>
  <?php
    require_once './include/connect.php'
  ?>
  <div class="wrapper">
    <div class="header">
      <div class="container">
```



```

<div class="header_body">
  <div class="header__logo"><a href="./index.php">Uni-Soko</a></div>
  <div class="header__info">
    <?php
      if(isset($_SESSION['user'])){
        $id_user = $_SESSION['user']['id'];
        $result_count_trash = $mysql->query("SELECT count(*) FROM
`trash` WHERE id_user = '$id_user'");
        $row_count_trash = $result_count_trash->fetch_assoc();
      ?>
      <div class="header__acc"><a href="./include/logout.php">Log
out</a></div>
      <form class="header__trash" action="./php/shopping_cart.php"
method="get">
        <button href="./trash.php" name="trash_button" value="<?php echo
$id_user ?>"></button>
        <div class="trash__counter"><?php echo $row_count_trash['count(*)']
?></div>
      </form>
      <?php
        }else{
      ?>
      <div class="header__acc"><a
href="./php/login.php">Register/Login</a></div>
      <?php
        }
      ?>
    </div>
  </div>
</div>

```

```
</div>
<div class="body">
  <div class="body__top">
    <div class="container">
      <div class="body__serach">
        <div class="serach__left">
          <div class="left__title">Купуйте будь-що у своєму містечку!</div>
          <form action="./php/search.php" method="get" class="left__form">
            <input type="text" name="serach_goods" class="form__input"
placeholder="Пошук будь-якого товару">
          </form>
        </div>
        <div class="search__right"></div>
      </div>
    </div>
  </div>
  <div class="body__info">
    <div class="container">
      <div class="info__cards">
        <div class="cards__card">
          <div class="cards__body">
            <div class="card__desc">
              <div class="card__title">Шопінг</div>
              <div class="card__text">Інтернет-шопінг у нашому магазині
буде для вас максимально зручним. Чому? Бо в нас є майже всі товари, які
здатні забезпечити повсякденний комфорт. </div>
              <div class="crad__button button button__shopping">Більше</div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```

```

        <div class="card__image"></div>
    </div>
</div>
<div class="cards__card">
    <div class="cards__body">
        <div class="card__desc">
            <div class="card__title">Послуги</div>
            <div class="card__text">Ви вже на сайті інтернет-магазину «Uni-
Soko», де знайдуться різні категорії товарів. І через декілька хвилин уже
потрібне вже буде у вашій корзині.</div>
            <div class="card__button button button__services">Більше</div>
        </div>
        <div class="card__image"></div>
    </div>
</div>
</div>
</div>
</div>
</div>
<div class="body__goods">
    <?php
        $result = $mysql->query("SELECT * FROM `goods`");
    ?>
    <div class="container">
        <div class="goods__body">
            <div class="goods__title">Товари</div>
            <div class="goods__items">

```

```

<?php
    while($row = $result->fetch_assoc()){
    ?>
    <div class="items__item">
        <form class="item__body item__from" method="get"
action="./php/goods.php">
            <div class="item__image"></div>
            <div class="item__title"><?php echo $row['name_good'] ?></div>
            <div class="item__price"><?php echo $row['price_good'] ?>
грн</div>
            <button name="btn_goods" value="<?php echo $row['id_good'] ?>"
class="item__button button">Детальніше</button>
        </form>
        <?php
            if(isset($_SESSION['user'])){
            ?>
            <form class="item__add" method="post"
action="./include/add_trash.php">
                <button name="add_trash" value="<?php echo $row['id_good']
?>">
                    
                </button>
            </form>
            <?php
                }
            ?>
        </div>
    <?php
        }

```

```

        ?>
    </div>
</div>
<div class="goods__more">
    <div class="more__line"></div>
    <div class="more__button">Більше товарів</div>
    <div class="more__line"></div>
</div>
</div>
</div>
</div>
<div class="footer">
    <div class="container">
        <div class="footer__body">
            <div class="footer__desc">
                <div class="desc__title">Online-Shop</div>
                <div class="desc__text">Ми - одна з найбільших мереж будівельно-
господарчих маркетів у місті Суми, яка працює з 2001 року. За цей час обсяги
продажів сягнули масштабних розмірів. Ми розширили групи товарів, і це
дозволило зробити мережу «Uni-Soko» багатoproфільним маркетом не тільки
будівельних, але й господарчих товарів та товарів для оформлення інтер'єру.
Кожного дня ми працюємо для вас, розширюючи асортимент найменувань в
різних категоріях товарів.</div>
            </div>
            <div class="footer__row">
                <div class="footer__lists">
                    <div class="lists__tilte">Зв'яжіться з нами!</div>
                    <div class="lists__list">
                        <li class="phone"><a
href="tel:+38066572186174">+38(066)-57-218-61-74</a></li>

```

```
<li class="mail"><a href="mailto:k.nozadze@student.sumdu.edu.ua">k.nozadze@gamil.com</a></li>
```

```
<li class="loc"><a href="https://www.google.com/maps?q=sumy&gs_lcrp=EgZjaHJvbWUqCggAEAA Y4wIYgAQyCggAEAA Y4wIYgAQyCggBEC4Y1AIYgAQyCggCEC4Y1AIYgAQ yCggDEC4Y1AIYgAQyBwgEEC4YgAQyBwgFEC4YgAQyBwgGEAA YgAQyBw gHEAA YgAQyBwgIEAA YgAQyBwgJEAAYgATSAQgxMDQyajBqN6gCALACA A&um=1&ie=UTF- 8&sa=X&ved=2ahUKEwjm2_iQIlb_AhWLv4sKHSf7DQEQ_AUoAnoECAEQBA" target="_blank">Sumy</a></li>
```

```
</div>
```

```
</div>
```

```
<div class="footer__lists">
```

```
<div class="lists__tilte">Про нас</div>
```

```
<div class="lists__list">
```

```
<li class=""><a href="#">Підтримка</a></li>
```

```
<li class=""><a href="#">Політика конфіденційності</a></li>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<script src="/js/moreGoods.js"></script>
```

```
</body>
```

```
</html>
```

```
<?php
```

```
require_once './include/reconnect.php'
```

```
?>
```

Додаток Б Код CSS

```
@import url("./nullStyle.css");
@import url("./_header.css");
@import url("./_search.css");
@import url("./_info.css");
@import url("./_goods.css");
@import url("./_footer.css");
@import url("./_user.css");

body {
  font-family: "Montserrat Alternates", sans-serif;
  height: 100%;
}

.wrapper {
  min-height: 100%;
  display: flex;
  flex-direction: column;
}

.container {
  max-width: 1300px;
  padding: 0px 10px;
  margin: 0 auto;
}

.body {
  flex: 1 1 auto;
}
```

Додаток В Код JS

```
const startHeight = 838
const goodsItem = document.querySelector('.goods__items')
const height = goodsItem.clientHeight
const moreButton = document.querySelector('.more__button')
let currentHeight = startHeight
if(height < startHeight){
    const buttonLine = document.querySelectorAll('.more__line')
    buttonLine.forEach(element => {
        element.style.display = 'none'
    });
    moreButton.style.display = 'none'
}else{
    if(height > startHeight){
        goodsItem.style.height = `${currentHeight}px`;
    }
    moreButton.addEventListener('click', ()=>{
        currentHeight+= startHeight;
        if(height < currentHeight){
            goodsItem.style.height = `${height}px`;
            return
        }
        goodsItem.style.height = `${currentHeight}px`;
    })
}
```


Додаток Г Код PHP

```
<!DOCTYPE html>
<html lang="ua">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="../css/style.css">
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link
href="https://fonts.googleapis.com/css2?family=Inter:wght@400;500;700;900&family=Montserrat+Alternates:wght@400;500;700;900&family=Roboto:wght@400;500;700;900&display=swap" rel="stylesheet">
  <link rel="icon" type="image/png" href="/img/misc/shopping-cart.svg">
  <title>Uni-Soko</title>
</head>

<body>
  <?php
    require_once '../include/connect.php'
  ?>
  <div class="wrapper">
    <div class="header">
      <div class="container">
        <div class="header_body">
          <div class="header__logo"><a href="../index.php">Uni-Soko</a></div>
          <div class="header__info">
```

```

<?php
    if(isset($_SESSION['user'])){
        $id_user = $_SESSION['user']['id'];
        $result_count_trash = $mysql->query("SELECT count(*) FROM
`trash` WHERE id_user = '$id_user'");
        $row_count_trash = $result_count_trash->fetch_assoc();
    ?>

    <div class="header__acc"><a href=" ../include/logout.php">Log
out</a></div>

    <form class="header__trash" action=" ../shoping_cart.php" method="get">
        <button href=" ../shopping_cart.php" name="trash_button"
value="<?php echo $id_user ?>"></button>

        <div class="trash__counter"><?php echo $row_count_trash['count(*)']
?></div>

    </form>

    <?php
        }else{
    ?>

    <div class="header__acc"><a
href=" ../login.php">Register/Login</a></div>

    <?php
        }
    ?>

</div>

</div>

</div>

<div class="body">
    <?php

```

```
$search = $_GET['serach_goods'];
```

```
$result = $mysql->query("SELECT * FROM `goods` WHERE name_good  
like '%$search' OR name_good like '$search%' OR name_good like '$search' OR  
name_good like '%$search%';");
```

```
?>
```

```
<div class="body__goods">
```

```
<div class="container">
```

```
<div class="goods__body">
```

```
<div class="goods__title">Результати пошуку за: <span><?php echo  
$search; ?></span></div>
```

```
<div class="goods__items">
```

```
<?php
```

```
while($row = $result->fetch_assoc()){
```

```
?>
```

```
<div class="items__item">
```

```
<form class="item__body item__from" method="get"  
action="./goods.php">
```

```
<div class="item__image"></div>
```

```
<div class="item__title"><?php echo $row['name_good'] ?></div>
```

```
<div class="item__price"><?php echo $row['price_good'] ?>
```

```
грн</div>
```

```
<button name="btn_goods" value="<?php echo $row['id_good'] ?>"  
class="item__button button">Детальніше</button>
```

```
</form>
```

```
</div>
```

```
<?php
```

```
}
```

```
?>
```

```

</div>
</div>
</div>
</div>
</div>
<div class="footer">
  <div class="container">
    <div class="footer__body">
      <div class="footer__desc">
        <div class="desc__title">Online-Shop</div>
        <div class="desc__text">Ми - одна з найбільших мереж
будівельно-господарчих маркетів у місті Суми, яка працює з 2001 року. За цей
час обсяги продажів сягнули масштабних розмірів. Ми розширили групи
товарів, і це дозволило зробити мережу «Uni-Soko» багатoproфільним
маркетом не тільки будівельних, але й господарчих товарів та товарів для
оформлення інтер'єру. Кожного дня ми працюємо для вас, розширюючи
асортимент найменувань в різних категоріях товарів.</div>
      </div>
      <div class="footer__row">
        <div class="footer__lists">
          <div class="lists__tilte">Зв'яжіться з нами!</div>
          <div class="lists__list">
            <li class="phone"><a
href="tel:+38066572186174">+38(066)-57-218-61-74</a></li>
            <li class="mail"><a
href="mailto:k.nozadze@student.sumdu.edu.ua">k.nozadze@gamil.com</a></li>
            <li class="loc"><a
href="https://www.google.com/maps?q=sumy&gs_lcrp=EgZjaHJvbWUqCggAEAA
Y4wIYgAQyCggAEAAAY4wIYgAQyCggBEC4Y1AIYgAQyCggCEC4Y1AIYgAQ
yCggDEC4Y1AIYgAQyBwgEEC4YgAQyBwgFEC4YgAQyBwgGEAAyG AQyBw
gHEAAyG AQyBwgIEAAyG AQyBwgJEAAYgATSAQgxMDQyajBqN6gCALACA
A&um=1&ie=UTF-
```

8&sa=X&ved=2ahUKEwjm2_iQIib_AhWLV4sKHSf7DQEQ_AUoAnoECAEQBA" target="_blank">Sumy

</div>

</div>

<div class="footer__lists">

<div class="lists__tilte">Про нас</div>

<div class="lists__list">

<li class="">Підтримка

<li class="">Політика конфіденційності

</div>

</div>

</div>

</div>

</div>

</div>

<script src="./js/moreGoods.js"></script>

<?php

require_once './include/reconnect.php'

?>

</body>

</html>