

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Розробка інтернет-магазину сантехнічних виробів «Aquatech» з використанням React JS і Node.js

Виконав: студент

IV курсу, групи СН-41

спеціальності

122 Комп'ютерні науки

(шифр і назва спеціальності)

Лісовський С.В.

(підпис)

(прізвище та ініціали)

Керівник

Млинко Б.Б.

(підпис)

(прізвище та ініціали)

Нормоконтроль

Марценко С.В.

(підпис)

(прізвище та ініціали)

Завідувач кафедри

Боднарчук І.О.

(підпис)

(прізвище та ініціали)

Рецензент

(підпис)

(прізвище та ініціали)

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

(повна назва факультету)

Кафедра комп'ютерних наук

(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Боднарчук І.О.

(підпис)

(прізвище та ініціали)

«__» _____ 2024 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня Бакалавр

(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні науки

(шифр і назва спеціальності)

Студенту Лісовський Євген Володимирович

(прізвище, ім'я, по батькові)

1. Тема роботи Розробка інтернет-магазину сантехнічних виробів «Aquatech»
з використанням React JS і Node.js.

Керівник роботи Млинко Богдана Богданівна к.т.н., доц. кафедри КН

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «__» _____ 2024 року № _____

2. Термін подання студентом завершеної роботи _____

3. Вихідні дані до Літературні та інтернет джерела про створення інтернет-магазину
роботи

за допомогою технологій розробки React та Node.js

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1. Аналіз предметної області для розробки інтернет-магазину. 1.1 Аналіз предметної області. 1.2 Аналіз інтернет-ресурсів лідерів на ринку сантехнічних виробів. 1.3 Технології для розробки інтернет-магазину сантехнічних виробів. 1.4 Висновок до першого розділу

2. Проектування архітектури інтернет-магазину «Aquatech» 2.1 Взаємодія основних елементів архітектури. 2.2 Цілі та потреби інтернет-магазину сантехнічних виробів. 2.3 Функціональні вимоги до системи. 2.4 Діаграма варіантів використання. 2.5 Діаграма послідовностей.

2.6 Діаграма класів. 2.7 Висновки до другого розділу. 3. Розробка інтернет-магазину.

3.1 Початок роботи з React. 3.2 Створення клієнтської частини. 3.3 Дизайн інтерфейсу користувача. 3.4 Створення серверної частини. 3.5 Висновки до третього розділу. 4. Безпека

життєдіяльності, основи охорони праці. 4.1 Вимоги до освітлення виробничих приміщень. 4.2 Планування заходів із безпеки праці та контроль за їх виконанням. 4.3 Висновки до четвертого розділу. Висновки. Перелік джерел. Додатки

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

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи охорони праці	к.т.н., доц. Сенчишин В.С.	12.06.2024	15.06.2024

7. Дата видачі завдання 29 січня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	30.01.2024	Виконано
2.	Підбір джерел про розробку інтернет-магазину сантехнічних виробів	31.01.2024-03.02.2024	Виконано
3.	Опрацювання джерел по темі кваліфікаційної роботи	04.02.2024-06.02.2024	Виконано
4.	Виконання дослідження щодо розробки логічної архітектури	07.02.2024-11.02.2024	Виконано
	Розроблення інтернет-магазину сантехнічних виробів «Aquatech»		
5.	Оформлення розділу «Аналіз предметної області інтернет-магазину»	03.06.2024-05.06.2024	Виконано
6.	Оформлення розділу «Проектування архітектури інтернет-магазину»	06.06.2024-08.06.2024	Виконано
6.	Оформлення розділу «Розробка інтернет-магазину»	09.06.2024-11.06.2024	Виконано
7.	Виконання завдання до підрозділу «Безпека життєдіяльності»	12.06.2024-13.06.2024	Виконано
8.	Виконання завдання до підрозділу «Основи охорони праці»	14.06.2024-15.06.2024	Виконано
9.	Оформлення кваліфікаційної роботи	16.06.2024-17.06.2024	Виконано
10.	Нормоконтроль	18.06.2024-19.06.2024	Виконано
11.	Перевірка на плагіат	20.06.2024	Виконано
12.	Попередній захист кваліфікаційної роботи	21.06.2024	Виконано
13.	Захист кваліфікаційної роботи	26.06.2024	

Студент

(підпис)

Лісовський Є.В.

(прізвище та ініціали)

Керівник роботи

(підпис)

Млинко Б.Б.

(прізвище та ініціали)

АНОТАЦІЯ

Розробка інтернет-магазину сантехнічних виробів «Aquatech» з використанням React JS та Node.js // Кваліфікаційна робота освітнього рівня «Бакалавр» // Лісовський Євген Володимирович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група СН-41 // Тернопіль, 2024 // С. 74, рис. – 24, табл. – 1, кресл. – 0, додат. – 2, бібліогр. – 0.

Ключові слова: інтернет-магазин, сантехнічний виріб, розробка, архітектура, React, Node.js, MongoDB.

Кваліфікаційна робота присвячена дослідженню розробки інтернет-магазину сантехнічних виробів «Aquatech» та всіх необхідних елементів

Мета роботи: реалізація інтернет магазину сантехнічних виробів для продажу продукції з використанням сучасних технологій. таких, як React для створення клієнтської частини та Node.js для серверної частини.

В першому розділі кваліфікаційної роботи проаналізовано предметну область, проведений аналіз лідерів сантехнічних виробів та технологій для розробки інтернет-магазину.

В другому розділі кваліфікаційної роботи спроектовано архітектуру інтернет-магазину «Aquatech» за допомогою діаграми варіатів використання, діаграми послідовностей та діаграми класів.

В третьому розділі кваліфікаційної роботи розроблено клієнську та серверну частину інтернет-магазину.

В четвертому розділі кваліфікаційної роботи розглянуто вимоги до освітлення робочих приміщень та планування заходів із безпеки праці та контроль за їх виконанням.

ANNOTATION

Development of an online store of plumbing products "Aquatech" using React JS and Node.js // Qualification work for the educational level "Bachelor" / Lisovskyi Yevhen Volodymyrovych // Ternopil National Technical University named after Ivan Pului, Faculty of Computer Information Systems and Software Engineering, Department of Computer Science, group SN-41 // Ternopil, 2024 // P. 74, fig. 24, tables - 1, drawings - 0, supplementary - 0, bibliography - 0.

Keywords: online store, sanitary product, development, architecture, React, Node.js, MongoDB.

The qualification work is devoted to the study of the development of the online store of sanitary ware "Aquatech" and all the necessary elements

Purpose: to implement an online plumbing store for the sale of products using modern technologies such as React to create the client side and Node.js for the server side.

The first section of the qualification work analyzes the subject area, analyzes the leaders of sanitary products and technologies for the development of an online store.

In the second section of the qualification work, the architecture of the Aquatech online store is designed using a use case diagram, a sequence diagram, and a class diagram.

In the third section of the qualification work, the client and server parts of the online store were developed.

The fourth section of the qualification work considers the requirements for lighting of workspaces and planning of labor safety measures and control over their implementation.

ПЕРЕЛІК СКОРОЧЕНЬ І ТЕРМІНІВ

HTTP – це протокол, в якому описано правила передачі даних в інтернеті.

Javascript – високорівнева скриптова мова програмування, яка широко використовується для створення інтерактивних застосунків і інтернет-магазинів.

MongoDB – документо-орієнтована система керування базами даних (СКБД) з відкритим вихідним кодом, яка не потребує опису схеми таблиць

Node.js – середовище виконання JavaScript, яке дає змогу запускати код JavaScript на стороні сервера.

Npm – менеджер пакетів для JavaScript. З його допомогою розробники можуть легко встановлювати, оновлювати і видаляти бібліотеки та інструменти.

React – відкрита JavaScript бібліотека для створення інтерфейсів користувача

БД – база даних.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ДЛЯ РОЗРОБКИ ІНТЕРНЕТ-МАГАЗИНУ	8
1.1 Аналіз предметної області	8
1.2 Аналіз інтернет-ресурсів лідерів на ринку сантехнічних продуктів	9
1.3 Технології для розробки інтернет магазину сантехнічних виробів	13
1.4 Висновок до першого розділу.....	16
РОЗДІЛ 2. ПРОЄКТУВАННЯ АРХІТЕКТУРИ ІНТЕРНЕТ МАГАЗИНУ «AQUATECH»	17
2.1 Взаємодія основних елементів архітектури	17
2.2 Цілі та потреби інтернет магазину сантехнічних виробів.....	19
2.3 Функціональні вимоги до системи.....	20
2.4 Діаграма варіантів використання	23
2.5 Діаграми послідовності	26
2.6 Діаграма класів.....	29
2.7 Висновок до другого розділу	32
РОЗДІЛ 3. РОЗРОБКА ІНТЕРНЕТ МАГАЗИНУ	33
3.1 Початок роботи з React.....	33
3.2 Створення клієнтової частини	34
3.3 Дизайн інтерфейсу користувача	36
3.4 Створення серверної частини	46
3.5 Висновок до третього розділу.....	49
4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	50
4.1 Вимоги до освітлення виробничих приміщень.....	50
4.2 Планування заходів із безпеки праці та контроль за їх виконанням	52
4.3 Висновок до четвертого розділу.....	55
ВИСНОВОК.....	56
ПЕРЕЛІК ДЖЕРЕЛ	57
ДОДАТКИ	

ВСТУП

Актуальність теми. Розробка інтернет-магазину сантехнічних виробів є завжди актуальною, через те що, у наш час технології розвиваються неймовірно швидко, тим самим змушують сучасний світ слідувати новим тенденціями та вдосконалювати свої інтернет ресурси. Зміни відбуваються настільки стрімко, що компаніям доводиться постійно адаптуватися до нових умов, щоб залишатися конкурентоспроможними.

Електронна комерція – одна з галузей, яка зазнала серйозних змін у зв'язку технічним прогресом. Інтернет-магазини стали основним інструментом для багатьох підприємств, які прагнуть розширити ринок збуту і забезпечити зручність своїм клієнтам. Традиційні магазини під впливом інновацій втрачають своє лідерство на світову ринку, звичайною практикою для людей стало користуватися різними цифровими сервісами для виконання багатьох завдань.

Розробка та впровадження інтернет магазину сантехнічних виробів дає змогу бізнесу та компаніям демонструвати свою продукцію великій аудиторії через всесвітню павутину Інтернет. Привабливий контент, швидке оформлення замовлення, зручне керування, доставка допоможуть збільшити потік людей та популяризувати бренд.

Мета і задачі дослідження. Метою цієї кваліфікаційної роботи освітнього рівня «Бакалавр», є розробка інтернет-магазину сантехнічних виробів для продажу продукції з використанням сучасних технологій. Для успішної реалізації задачі використовується інноваційні інструменти такі, як React для створення клієнтської частини та Node.js для серверної частини.

Основні завдання відповідно до мети дослідження:

- вибрати сучасні інструменти для створення інтернет магазину;
- спроектувати логічну архітектуру всієї системи;
- розробити серверну та клієнтську частину;
- реалізувати захист даних.

Також слід забезпечити високу продуктивність та зручність користування системою.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ДЛЯ РОЗРОБКИ ІНТЕРНЕТ-МАГАЗИНУ

1.1 Аналіз предметної області

Сучасний ринок сантехнічних виробів вимагає інноваційного підходу до продажу та обслуговування клієнтів. З кожним роком споживачі стають все більш вимогливими до якості обслуговування. Нові технологічні впровадження розвиваються настільки швидко, що підприємства змушені постійно оновлювати власні електронні ресурси та адаптуватися до нових особливих рішень реалізації продукту. Для конкурентоспроможності на сантехнічному ринку слід постійно вдосконалювати свої інтернет платформи, щоб задовільними потреби користувачів. Розробка інтернет магазину не залишилася в стороні, оскільки фізичні магазини з кожним днем втрачають своє лідерство на ринку збуту. Використання електронних цифрових сервісів стало звичною практикою для технологічного світу.

У зв'язку з цим розробка сучасного і ефективного інтернет-магазину сантехнічних виробів є актуальним завданням. Такий магазин повинен відповідати найновішим стандартам якості. Також не слід забувати про зручність та безпеку, впроваджуючи високий рівень обслуговування клієнтів і можливість моментального реагування на їх потреби.

Особливості сучасних рішень для інтернет магазину:

- індивідуальний підхід: сучасний ринок диктує свої правила, користувачі звикли до того, що кожна електронна комерція хоче привернути їхню увагу за допомогою різних бонусів, акцій та спеціальних пропозицій. Для успішних продажів сантехнічних товарів слід віднайти свій персональний підхід, щоб втримати покупця та задовільнити його потреби.

- Інтеграція з соціальними мережами: у теперішній час соціальні мережі стали головним компонентом людського життя, тому для електронної комерції необхідно взаємодіяти з аудиторією через регулярні рекламні події та аналізувати відгуки для покращення якості продукції.

– Конфіденційність: захист персональних даних забезпечує швидке та якісне завершення транзакцій. Для цього використовують сучасні технології шифрування. Ці технології дають комплексне розуміння технічних процесів та допомагають приймати обґрунтовані рішення та оптимізувати всі елементи інтернет магазину.

Інтернет-магазини змушені регулярно вдосконалювати свої інструменти та функції, адаптуючись до сучасних змін, щоб забезпечити підтримку необхідних ключових компонентів. Крім перерахованих особливостей слід додати аналітичні інструменти для відстеження дій користувача на інтернет-ресурсі. Це дозволить зрозуміти його потреби та швидко створити пропозицію до вподобань. Також аналіз даних допомагає виробити стратегічні рішення для зростання популярності підприємства.

1.2 Аналіз інтернет-ресурсів лідерів на ринку сантехнічних продуктів

Для того, щоб створити ефективний та конкурентоспроможний інтернет-магазин сантехнічних виробів слід проаналізувати інтернет-ресурси провідних лідерів на ринку. Для лідерів прийнято обрати: Grohe, Rea та торгову мережу «Арс-кераміка» для того щоб, створити та вдосконалити власний інтернет магазин.

«Grohe» – компанія, яка розширює свої експортні канали, завдяки чому перші партії змішувачів уже надходять на міжнародні ринки. Це дозволило виробнику розширити свій асортимент продукції та закріпитися на нових ринках, зміцнюючи свою позицію як провідного виробника сантехнічного обладнання. Завдяки цим стратегіям компанія змогла не лише подолати труднощі, але й продовжувати зростати та інноваційно розвиватися.

Цей лідер в галузі сантехнічних товарів вирізняється інтернет-магазином, оскільки включає чітко структуровану категоризацію товарів та високу підтримку. Завдяки потужній пошуковій системі користувачі можуть швидко знаходити потрібні сантехнічні вироби, що в свою чергу покращує навігацію та взаємодію між системою. Всі ці аспекти допомагають компанії знаходитись

кращими на ринку сантехнічної продукції, забезпечуючи високу якість обслуговування, простоту використання і відповідають потребам сучасних споживачів.

На рисунку 1.1 представлено інтернет-магазин цього лідера.

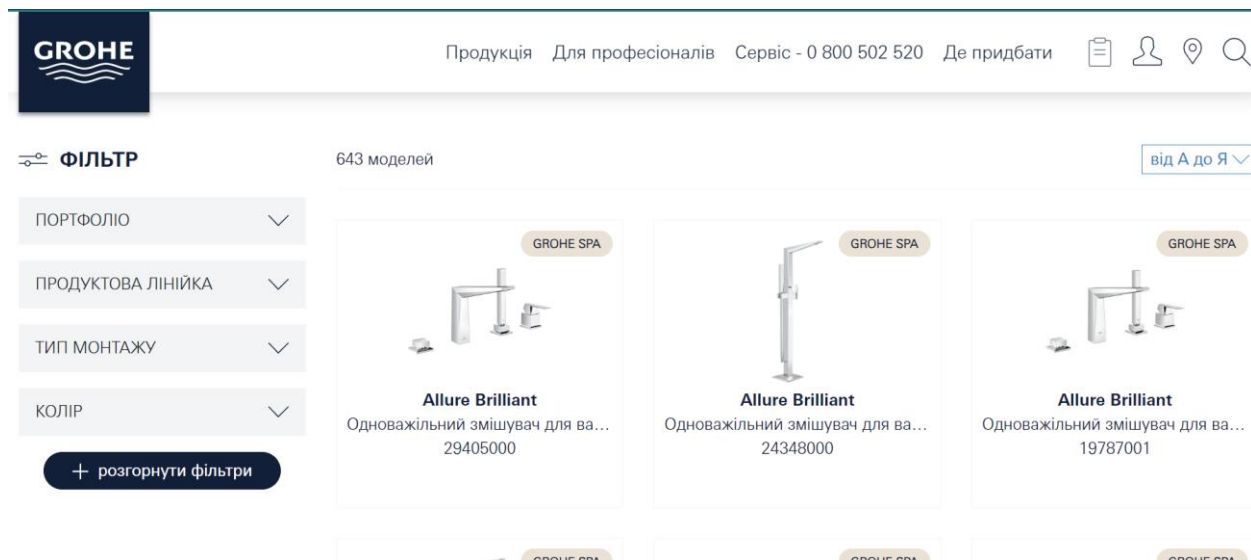


Рисунок 1.1 – Інтернет магазин компанії «Grohe»

Головні характеристики інтернет магазину:

– Дизайн та навігація: головне меню зроблено з чіткою структурою продукції, що дозволяє швидко віднайти необхідний сантехнічний виріб. Також ключовим аспектом для успішного дизайну є якість зображення товарів та детальний опис всіх компонентів, який в компанії Grohe зроблений на вищому рівні.

– Функціональність компонентів: витривала пошукова структура з можливістю фільтрації за різними ключовими параметрами є необхідним елементом для лідера на ринку сантехнічних товарів. Також для довіри покупців впроваджено інтерактивні елементи, такі як калькулятор витрат води, віртуальні демонстрації продуктів.

– Інтерфейс: впроваджує легкість у використанні завдяки навігації та швидкий доступ до важливої інформації для відвідувача та включає можливість порівняння продукції, що допомагає користувачам прийняти обдумане рішення.

Інтернет магазин сантехніки «Grohe» є прикладом успішної платформи для продажу власної продукції, яка включає стильний сучасний дизайн, потужний функціонал на високу якість обслуговування.

Далі розглянуто лідера на ринку сантехнічних виробів компанію «Rea».

«Rea» – лідер сантехнічних виробів, який завдяки високій якості та широкому асортименту продукція здобув популярність і визнання серед покупців у багатьох країнах. Використання сучасних технологій виробництва та якісних матеріалів, а також багатовіковий досвід у галузі, сприяли значному успіху бренду на ринку.

На рисунку 1.2 візуально представлено інтернет-магазин цього виробника.

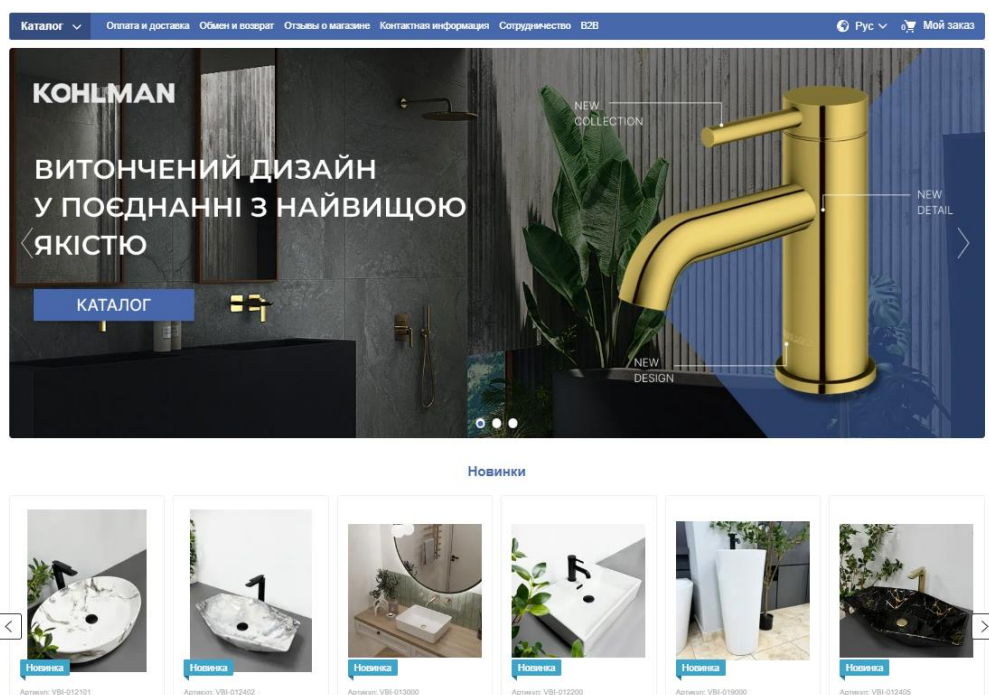


Рисунок 1.2 – Інтернет магазин виробника «Rea»

Основні характеристики інтернет платформи:

– Навігація та дизайн: в цієї компанії створений мінімалістичний та перенавантажений дизайн з акцентом на зображення продукції. Також розроблено зручне меню з категоріями популярний сантехнічних товарів, що в свою чергу дозволяє швидко знайти необхідний продукт. Для товару спроектовано детальний опис з вказівкою додаткової інформації.

– Функціональність основних елементів: виробник провадив розширений пошук з можливістю фільтрації за категоріями та популярними параметрами. Також розроблені інтерактивні елементи, такі як 3D перегляд продукції за допомогою відео та джерел в інтернет магазині. Слід відмітити, що функціональність зроблена продумано з відгуками користувачів про сантехнічні продукти найпопулярніші з яких виділяють рейтингом для зручності.

– Інструменти маркетингу: цей лідер сантехнічних товарів проваджує регулярні акцій від виробника та знижки для нових та постійних користувачів. Також добре спроектована система інформацій про розсилки з новинками та спеціальними пропозиціями продукції. Для охоплення великої клієнтської бази впроваджена інтеграція з популярними соціальними мережами.

Ще одним важливим аспектом є цілодобова підтримка клієнтів. Інтернет-магазин надає ефективне і якісне обслуговування, включаючи консультації з фахівцями, які можуть допомогти у виборі сантехнічної продукції, вирішити технічні проблеми та відповісти на питання стосовну виробу.

Після розглядання всіх основних компонентів інтернет магазину слід перейти до наступного лідера.

«Арс-Кераміка» – провідний лідер з постачання сантехнічного обладнання. Компанія пропонує великий асортимент товару відомих брендів та відповідає за якість продукції.

На рисунку 1.3 представлено інтернет магазин від компанії «Арс-Кераміки».

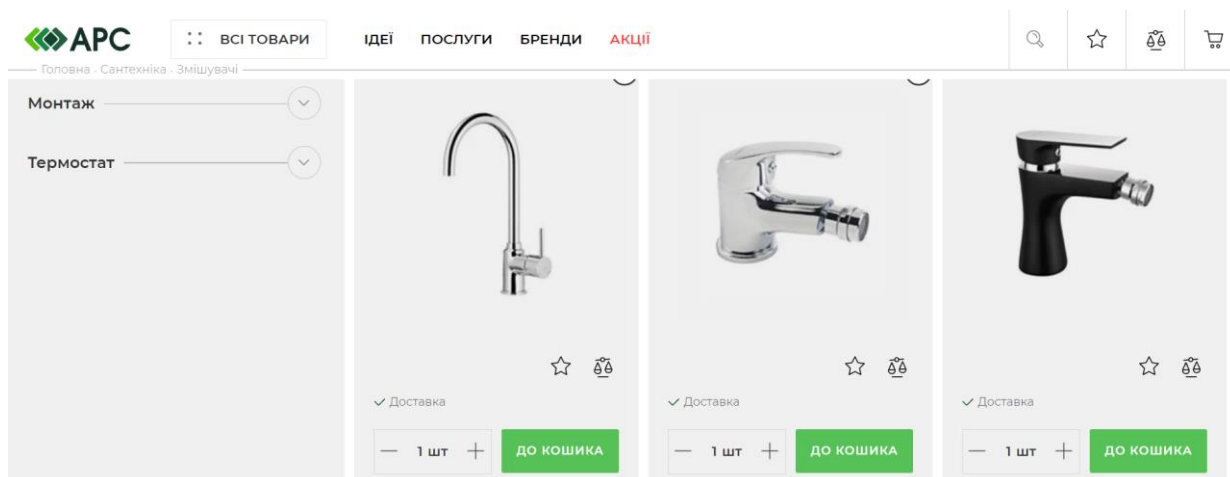


Рисунок 1.3 – Інтернет-магазин «Арс-Кераміки»

Інтернет-магазин вирізняється сучасним і привабливим дизайном, який відповідає світовим стандартам. Інтерфейс розроблений таким чином, щоб забезпечити зручність користування на різних пристроях, включаючи як персональні комп'ютери, так і мобільні пристрої. Платформа компанії також надає інтерактивні каталоги та віртуальні представлення, що дозволяють відвідувачам візуалізувати товари в реальних інтер'єрах.

Компанія регулярно організовує активні рекламні кампанії та спеціальні акції для підвищення популярності бренду і залучення нових клієнтів. Основні переваги цього інтернет-магазину включають високоякісні фотографії товарів, потужну пошукову систему з фільтрами та інтерактивні каталоги. Продукція відповідає високим стандартам якості, що забезпечує її довговічність і надійність у використанні.

Торгова мережа надає гарантію на всі товари та забезпечує післягарантійне обслуговування, що підвищує довіру клієнтів і сприяє тривалій співпраці.

1.3 Технології для розробки інтернет магазину сантехнічних виробів

Для створення ефективного сучасного інтернет магазину використовують інноваційні сучасні технології, такі як: React, Node.js, Express, Redux та для сховища даних MongoDB. Розглянемо їх детальніше.

На рисунку 1.4 відображено логотипи сучасних цих технологій.



Рисунок 1.4 – Логотипи сучасних технологій

React слугує компонентом для створення інтерфейсів користувача, розроблена компанією. Він дозволяє створювати багаторазові компоненти та управляти станом додатка за допомогою віртуального DOM, що забезпечує високу продуктивність [1-3].

Переваги цього компоненту:

- Модульність. Цей інструмент можна розробляти та тестувати окремо, що підвищує та спрощує підтримку коду. За допомогою цієї функції можна краще зосередитись на слабших частинах програми.

- Інкапсуляція. Для високої якості інструмент використовує спеціальні функції для більш організованості та спрощенню процесу розробки.

- Зрозуміла та легка структура. Завдяки односторонньому потоку даних структура додатку стає більш прозорою і зрозумілою. Також за допомогою цього потоку система забезпечує легке управління станом проєкту, що зменшує виникненню технічних несправностей.

Node.js виступає в ролі спеціальної платформи на базі JavaScript, що дозволяє виконувати код на стороні сервера. Основною перевагою Node.js є її асинхронна природа, яка дозволяє ефективно обробляти велику кількість одночасних запитів [4-6].

Головні переваги Node.js:

- неможливість блокування – у Node.js немає блокуючих викликів, що забезпечує безперервне виконання коду і знижує затримки при обробці запитів;

- ефективне використання ресурсів механізму впровадження та асинхронної природи дозволяє ефективно використовувати системні ресурси, що знижує витрати на інфраструктуру.

Express представляє мінімалістичний та гнучкий інструмент для Node.js. Він забезпечує набір інструментів для побудови веб-додатків та API, спрощуючи процес розробки серверної частини [7-9].

Цей елемент є потужною основою для серверної частини інтернет-магазину сантехніки завдяки своїй асинхронній природі. Це дозволяє обробляти тисячі запитів одночасно, забезпечуючи високу продуктивність і ефективне використання ресурсів. Величезна екосистема модулів та функцій пропонує

безліч готових рішень для різних завдань, що значно пришвидшує процес розробки. Висока швидкодія та можливість масштабування роблять Node.js ідеальним вибором для створення високонавантажених інтернет магазинів.

Ключові переваги Express [10-12]:

- оптимізація запитів Express дозволяє ефективно обробляти HTTP-запити, що забезпечує високу продуктивність та швидкодію серверних додатків;
- мінімальні накладні витрати завдяки своїй легкій структурі, Express має мінімальні накладні витрати, що забезпечує швидку роботу додатка.

Redux елемент бібліотеки для управління станом додатків. Вона допомагає підтримувати передбачуваний стан додатка, роблячи його більш зрозумілим і полегшуючи налагодження та тестування [13-15].

Інструмент додає структуру управління станом інтернет-магазину сантехніки, забезпечуючи передбачуваність та контрольованість. Також завдяки єдиному джерелу відстежувати зміни стану, що значно спрощує вирішення технічних питань інтернет магазину. Незмінність стану гарантує, що всі зміни здійснюються за допомогою чистих функцій, що робить додаток більш стабільним і передбачуваним.

Підтримка дозволяє обробляти асинхронні операції та розширювати можливості Redux без зміни основного коду.

Спеціальні функції. Представлені чистими функціями, тобто вони завжди повертають один і той же результат для одного і того ж вхідного стану та дії. Це робить їх легко трестованими та передбачуваними.

Простота тестування. За допомогою функції легко тестувати, оскільки вони не мають побічних ефектів і залежать тільки від вхідних параметрів.

Предбачуваність. Використання особливих чистих функцій для зміни стану забезпечує передбачувану поведінку додатка.

MongoDB представляє об'єктно-орієнтовану базу даних, яка в свою чергу забезпечує високу продуктивність, масштабованість та гнучкість у зберіганні та обробці даних [16].

Основні переваги MongoDB для системи інтернет магазину:

– Гнучкість. Зберігання даних у вигляді документів дозволяє легко адаптувати базу даних до потреб додатка, оскільки кожен документ може мати власну унікальну структуру, не обмежуючись жорсткими рамками схеми.

– Відсутність жорсткої схеми: Однією з ключових переваг MongoDB є відсутність вимоги до попереднього визначення схеми бази даних. Це означає, що структура даних може змінюватися в процесі розробки, не вимагаючи змін у вже існуючих документах.

– Адаптація до потреб інтернет магазину. Завдяки своїй структурі, відсутності жорсткої схеми та гнучкості у зберіганні даних, забезпечує ефективно та зручне управління складними даними.

Це робить її ідеальним вибором для сучасних інтернет магазинів та надає адаптації під мобільні додатки, де вимоги до даних можуть швидко змінюватися.

1.4 Висновок до першого розділу

Перший розділ кваліфікаційної роботи присвячений аналізу предметної області для розробки інтернет-магазину сантехнічних виробів. Аналіз показав, що сучасний ринок сантехніки вимагає інноваційних підходів до продажів та обслуговування клієнтів. Швидкий розвиток технологій змушує підприємства постійно оновлювати свої електронні ресурси та адаптуватися до нових умов ринку.

Аналіз інтернет-ресурсів провідних лідерів на ринку, таких як «Grohe», «Rea» та торгової мережі «Арс-Кераміка», показав, що успішні компанії використовують сучасний дизайн, потужні пошукові системи, інтерактивні елементи та забезпечують високий рівень обслуговування клієнтів.

Розглянуті сучасні технології для розробки інтернет-магазину, що в свою чергу дозволяють створювати високопродуктивні, масштабовані та гнучкі рішення, які відповідають потребам ринку. Використання технологій таких , як React, Node.js, Express, Redux та MongoDB забезпечує ефективну розробку та підтримку інтернет-магазину сантехнічних виробів.

РОЗДІЛ 2. ПРОЄКТУВАННЯ АРХІТЕКТУРИ ІНТЕРНЕТ МАГАЗИНУ «AQUATECH»

2.1 Взаємодія основних елементів архітектури

Архітектура інтернет-магазину є основою, на якій будується вся система. Вона визначає, як різні компоненти взаємодіють між собою. Також архітектура повинна забезпечувати просту підтримку та регулярні оновлення всього механізму для високої працездатності інтернет-магазину. Утримання такої архітектури дає змогу в подальшому виправити помилки, які виникатимуть на етапі розробки та додати нові функції, щоб забезпечити сумісність з новими технологіями. Потрібно врахувати ключові аспекти на етапі розробки:

- продуктивність;
- просте використання;
- безпека;
- підтримка.

В інтернет магазині сантехнічних виробів «Aquatech» важливу роль відіграє швидка обробка запитів для того, щоб збалансувати навантаження. Постійний моніторинг забезпечує стабільну роботу всіх функцій системи, що в свою чергу дає високу продуктивність.

Простота використання інтернет-магазину є важливим аспектом сучасного життя. Від простоти залежить задоволеність користувачів і їх лояльність до магазину. Інтуїтивно зрозумілий інтерфейс, зручна навігація, ефективний пошук, прозорий процес покупки і високоякісна підтримка забезпечують позитивний користувальницький досвід. Всі ці компоненти не тільки збільшують конверсії, але і сприяють збільшенню кількості повторних покупок і рекомендацій.

Також використання сучасних методів шифрування, багаторівнева швидка автентифікація і авторизація, захист від атак і постійний моніторинг безпеки дозволяють забезпечити високий рівень надійності системи. За допомогою цього знижується ризик, який сприяє підвищенню рейтингу інтернет-магазину серед конкурентів. Для підтримки всієї системи потрібно впровадити автоматизацію процесів, високоякісний сервіс і ефективно вирішувати проблеми користувачів,

надавати актуальну інформацію і забезпечувати стабільну роботу інтернет магазину [17].

У центрі створення магазину сантехнічних виробів «Aquatech» лежить розробка програмного забезпечення сайту. Оскільки інтернет-магазин повністю орієнтований на користувача, всі види діяльності та послуги орієнтовані на потреби та зручність користувачів. Слід врахувати, що для реалізації проєкту була розроблена повнофункціональна система, що складається з трьох ключових компонентів:

- клієнтська частина;
- серверна частина;
- база даних.

Клієнтська частина магазину, розробленої за допомогою React, взаємодіє з сервером через протокол Node.js за допомогою HTTP-запиту. Вузол server.js обробляє запити та взаємодіє з базою даних MongoDB, використовуючи мову запитів.

На рисунку представлено 2.1 представлено взаємодію основних елементів.

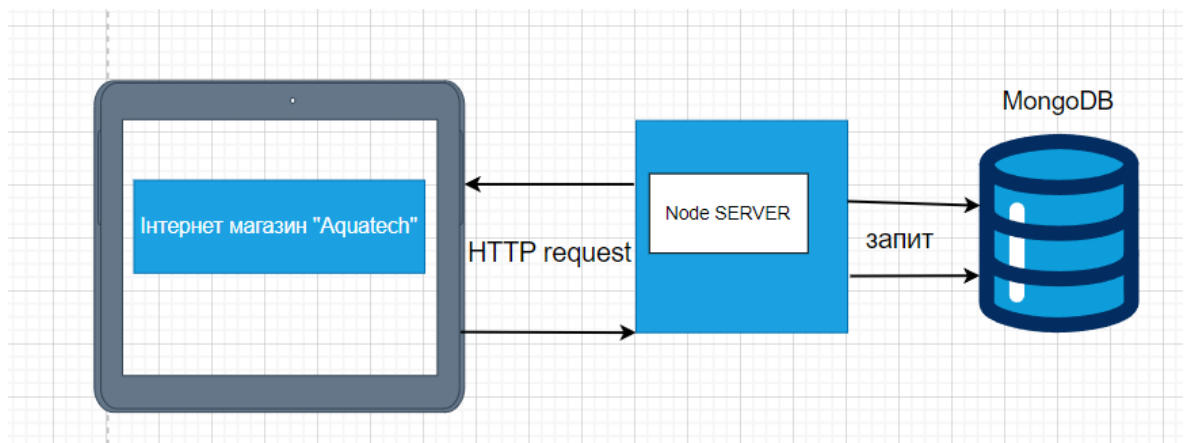


Рисунок 2.1 – Взаємодія основних елементів

Структура інтернет-магазину важлива для забезпечення ефективної, надійної та масштабованої роботи. Чіткий поділ між клієнтською і серверною частинами спрощує обслуговування і розвиток системи. Він використовує новітні технології, такі як React та Node.js забезпечує високу продуктивність і здатність швидко реагувати на запити користувачів. MongoDB забезпечує гнучке

зберігання даних і просте масштабування. Це особливо важливо для динамічного інтернет-магазину.

2.2 Цілі та потреби інтернет магазину сантехнічних виробів.

Основні цілі та потреби інтернет-магазину сантехнічних виробів включають широкий спектр функціональних вимог, спрямованих на забезпечення ефективної роботи всіх складових інформаційної системи для задоволення потреб користувачів. Розглянемо цей спектр детальніше.

Продаж товарів. Сантехнічні вироби займають лідерську позицію на ринку і користуються попитом в повсякденному житті, тому інформаційна система інтернет-магазину має давати змогу купити товар онлайн без труднощів для користувача. Система повинна забезпечувати швидкий і безпечний процес оформлення замовлення, оплати і доставки товарів. Всі ці параметри досягаються за рахунок налагодженої роботи з платіжними системами, які гарантують захист грошових транзакцій завдяки інтерфейсу для навігації.

Менеджмент продукту. Для вдосконалення стратегії розвитку магазину система повинна надавати аналіз даних про продаж певного товару, поведінку користувача, характеристики на які він опирається при покупці. За допомогою збору та обробці даних про кількість замовлень та продажі створюється популярність товару, середній час перебування на сайті та певну задовільну суму, яку готовий платити користувач за сантехнічний виріб. В подальшому все це допоможе оптимізувати магазин та надавати найкращі ціни на товар.

Управління замовленнями та доставкою. Сантехнічні вироби досить об'ємні, деякі з них мають велику вагу, що в свою чергу ускладнює зручність для покупця. Тому було прийняте рішення на всі важко підйомні сантехнічні вироби зробити безкоштовну доставку, щоб збільшити продажі та конверсії замовлень.

Звітність продукту. На основі менеджменту продукту слід формувати регулярні звіти, що допоможуть виявити тренди, зрозуміти вподобання клієнтів та приймати обґрунтовані рішення щодо розширення асортименту, Також звіти

допомагають аналізувати конкурентів та включати в роботу маркетингові ідеї для вдосконалення роботи інтернет магазину.

Комунікація з клієнтами. Для продажу товару онлайн слід впровадити кілька каналів зв'язку з клієнтами через електронну пошту, телефон, соціальні мережі. З огляду на високу конкуренцію обслуговування займає важливу роль, потрібно вирішувати проблеми та надавати консультації щодо вибору продукту. Також інтеграція з різними каналами зв'язку дозволяють підтримувати високий рівень обслуговування.

Захист персональних даних. Безпека є головним чинником для довіри користувачів до інтернет-магазину. Механізм повинен забезпечити захист платіжних операцій та іншої важливої конфіденційної інформації. Для цього потрібно дотримуватись стандартів безпеки при розробці магазину.

2.3 Функціональні вимоги до системи

Проаналізувавши всі цілі інтернет магазину сантехнічних виробів було прийнято рішення включити основні пріоритети до системи яка відповідає очікуванням та надає високу якість і надійність. Для проектування інтернет магазину важливо чітко визначити вимоги до системи, щоб забезпечити її безперервну та надійну роботу.

Наступним кроком визначасмо пріоритети для того, щоб зосередити зусилля на функціоналі самого магазину. Доповнення та уточнення вимог допоможе створенню масштабної системи для забезпечення якісного обслуговування та втіленню всіх поставлених задач.

Проектування інтернет-магазину це не тільки асортимент товару, послуги та ціни. Ринок сантехнічних виробів великий, тому потрібно враховувати сучасні вимоги та задовільнити покупців за допомогою функціональних характеристик. Функціональні вимоги є базовими та вимагають різних аспектів, котрі в свою чергу забезпечують розширені можливості для користувачів та адміністраторів. Також вони сприяють успішному запуску нововведень та подальшому розвитку інтернет магазину у довгостроковій перспективі.

Нижче представлено таблицю з вимогами до програми.

Таблиця 2.1 – Вимоги до програми

Посилання	Опис	Пріоритет 1 -має бути обов'язково, 2 -можна додати, 3- не обов'язково
F1	Аунтефікація даних Реєстрація та вхід в систему.	1
F2	Додавання, зберігання та оновлення даних в корзині	1
F3	Завантаження нового списку на платформі	1
F4	Створення замовлення та його подальшій шлях	1
F5	Адміністрування та консоль управління	2
F6	Платіж за допомогою банківських карт	2
F7	Доставка сантехнічної продукції	3

На основі цієї таблиці вимог функціоналу інтернет магазину сантехнічних виробів можна зробити детальний аналіз кожного елементу з урахуванням пріоритетів та важливості для системи. Перед етапом розробки важливим кроком є відтворення вимог, оскільки вони показують над якими компонентами слід працювати найбільше щоб відтворити базову структуру.

Реєстрація облікових записів та вхід в систему для користувачів є невід'ємною частиною будь-якої платформи для ідентифікації особистості і персоналізації відвідувачів. Без цієї функції покупці не зможуть взаємодіяти з системою. Також не потрібно нехтувати реєстрацією, вона повинна включати унікальну інформацію про користувача, таку як пароль, електронна пошта, ім'я. Важливо передбачити можливість відновлення особистих даних, вразі втрати паролю та підтримку різних методів автентифікації.

Далі необхідно спроектувати додавання, редагування та видалення товарів у корзині для ефективного процесу покупок. Користувачі повинні мати доступ для керування власною корзиною, щоб підібрати, ознайомитись з продуктом та перейти до фінального процесу покупки. Також вся структура повинна підтримувати збереження стану кошика при повторному вході в систему для всіх відвідувачів. Важливим етапом буде збереження історії покупок та перегляду сантехнічного товару для користувача.

Також до функціоналу інтернет магазину включені адміністратори, які мають можливість додавати нові товари до каталогу продукції магазину, включаючи детальну інформацію. Для нормального функціонування потрібна постійна підтримка та оновлення для забезпечення актуальності та конкурентоспроможності. Щоб все працювало без перебоїв слід передбачити можливість завантаження декількох зображень в різних форматах для одного типу товару.

Система має працювати злагоджено, тому основний процес оформлення замовлень є ключовим для відтворення всі функцій. Користувачі мають мати можливість швидко та легко завершити покупку та вразі хорошого сервісу залишити відгук для покращення індексації товару. Також система зобов'язана включати при продажі сантехнічного товару доставку для запобігання труднощів під час вибору продукції. Адміністративна консоль повинна мати розширені функції звітності та аналітики для відстеження покупок та моніторингу роботи інтернет магазину.

Часто користувачі реєструють кілька облікових записів, щоб запобігти цьому слід впровадити до інтернет магазину підтвердження даних за допомогою електронної пошти. Це запобігає створенню фальшивих аккаунтів і забезпечує достовірність даних. Також для захисту інформації слід використати HTTPS, щоб зробити транзакції та обробку особистої інформації безпечними. Для успішної роботи потрібно використати регулярне оновлення SSL-сертифікатів для моніторингу безпеки з'єднання.

Встановлення інтеграції з платіжними системами покращує зручність та безпеку платежів, що збільшить кількість покупок та хороших відгуків від

користувачів. Система повинна підтримувати механізм функціональності для обробки повернень платежів.

Функціональні вимоги до системи інтернет магазину дозволяють визначити базові та необхідні впровадження для забезпечення надійної роботи. Також визначення пріоритетів відграє важливу роль для системи та допомагає зосередити напрямок проектування на найбільших недопрацьованих функціях для покращення та успішний розвиток інтернет магазину. Доповнення та уточнення вимог дозволить створити комплексну сантехніко-орієнтовану систему.

2.4 Діаграма варіантів використання

Діаграма варіантів використання магазину показує взаємодію користувача з системою і охоплює основні функції, що забезпечують ефективну роботу проекту. Вона включає в себе два основних типи учасників: користувач і адміністратор, а також ряд випадків використання.

Структура надає платформу для купівлі та продажу сантехнічних виробів. Користувачі можуть переглядати доступні товари, додавати в кошик, замовляти та керувати профілями. Адміністратор має доступ до розширених функцій, які дозволяють йому керувати вмістом, особливими товарами, замовленнями та інформацією про користувачів. Діаграми варіантів допомагають візуалізувати основні сценарії взаємодії користувача з системою інтернет-магазину, сприяючи розробці більш ефективної та зрозумілої архітектури програми.

Вона надає чітке уявлення про функціональні можливості системи, що дозволяє розробникам і дизайнерам легко визначати необхідні компоненти та забезпечувати їхню інтеграцію для досягнення максимального комфорту користувачів. Крім того, діаграма варіантів використання забезпечує важливу інформацію для планування та тестування системи. Вона допомагає ідентифікувати потенційні проблеми та зони ризику, а також забезпечує можливість їхнього виправлення на ранніх етапах розробки.

В ході проектування архітектури прийнято рішення зобразити діаграму варіантів використання.

Нижче на рисунку 2.2 представлена її реалізація

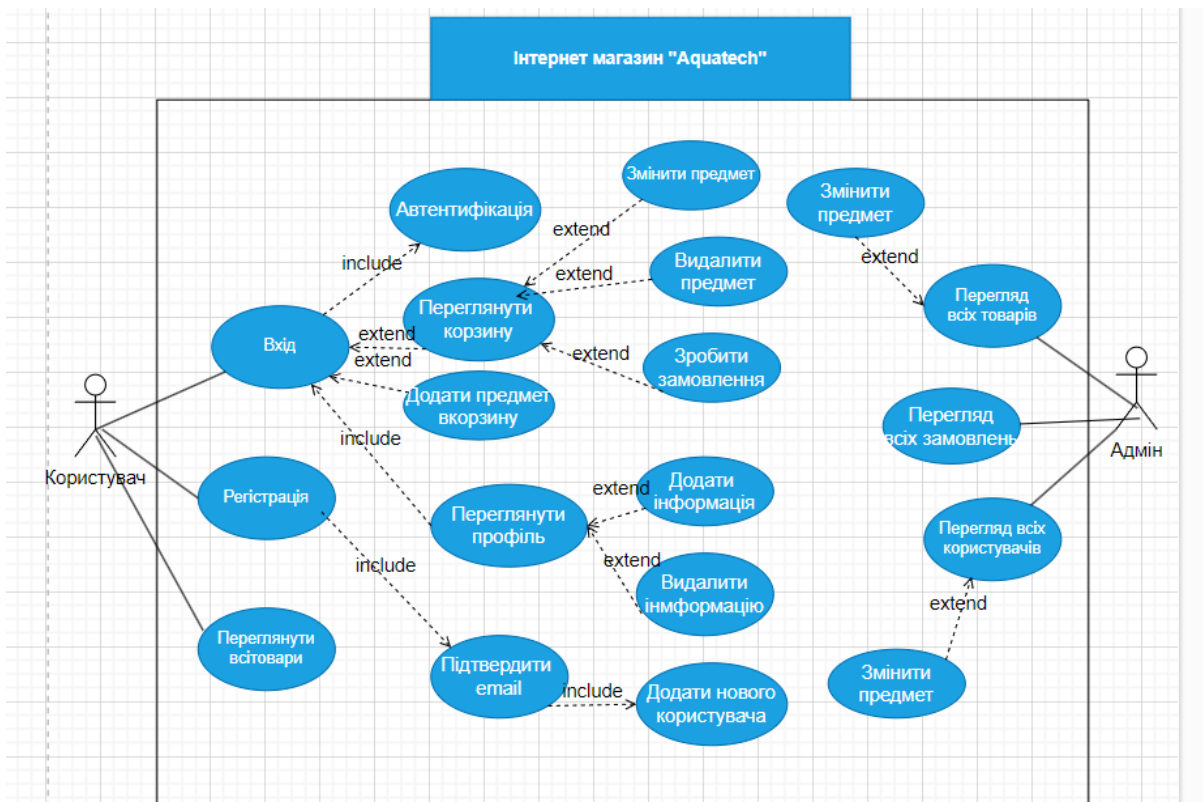


Рисунок 2.2 – Діаграма використання інтернет магазину «Aquatech»

Для подальшої роботи над магазином сантехнічних виробів було прийнято рішення описати зв'язки використання для користувача і адміністратора, дозволяє зрозуміти основні функціональні вимоги та взаємозв'язки між різними компонентами системи.

Розглянемо варіанти використання для користувача

Вхід. Користувач може увійти в систему для доступу до персоналізованих функцій. Дія рахується базовою та надає доступ до інших функцій.

Автентифікація. Проходить перевірка користувача перед наданням доступу до платформи

Реєстрація. Якщо користувач ще ні разу не відвідував інтернет магазин система пропонує йому зареєструватися в ній, ввести свої персональні дані та зберегти обліковий запис.

Переглянути корзину. Якщо користувач увійшов в інтернет магазин він може переглянути доступні сантехнічні товари.

Додати предмет в корзину. Користувач може додати товар до корзини, який він планує в подальшому купити.

Зробити замовлення. Користувач може оформити замовлення на товари, що знаходяться в корзині.

Переглянути профіль. Користувач може переглянути свій профіль і при потребі ввести нові дані або редагувати старі.

Підтвердити пошту. Після реєстрації користувач підтверджує свою особу переходячи на електронну пошту для активації облікового запису.

Розглянемо варіанти використання для адміністратора

Перегляд всіх товарів. Адміністратор може переглядати всі товар, які знаходяться в системі для керування ними.

Змінити характеристики товару. Адміністратор може змінювати інформацію про товари, такі як опис, ціна, наявність та інші технічні або описові параметри.

Видалити предмет. Адміністратор при потребі може видалити предмет з всієї системи або перекинути його в резервне сховище.

Перегляд всіх замовлень. Адміністратор при необхідності переглядає всі замовлення користувача за весь період для збору даних.

Перегляд всіх користувачів. Адміністратор може переглядати інформацію про всіх користувачів.

Додати нову інформацію. Адміністратор може додавати нову інформацію в обліковий профіль користувача.

Видалити інформацію: Адміністратор може видаляти інформацію з профілю користувача, якщо вона містить не коректну інформацію

Додати нового користувача. Адміністратор може додавати нових користувачів в систему для того, щоб зробити з нього іншого адміністратора або запросити за посиланням.

На діаграмі варіантів використання беруть участь два зв'язки «include» та «extend». Перший з них повинен виконати певну дію, це може бути базова дія,

наприклад вхід в систему. Другий може комплектуватися додатковою дією в певних умовах.

2.5 Діаграми послідовності

Для того, щоб інтернет-магазин функціонував ефективно, необхідно ретельно спроектувати його архітектуру та взаємодію між різними компонентами системи. Одним з важливих інструментів для цього є діаграма послідовностей, яка демонструє процеси, що відбуваються в структурі. Також показує як різні компоненти взаємодіють між собою для виконання певних завдань. Враховуючи основну мету інтернет магазину «Aquatec» забезпечення користувачів можливістю швидко і безпечно купувати сантехнічні товари, які необхідні для буденного життя. Було прийнято рішення реалізувати цю діаграму для кращого розуміння та оптимізації роботи системи, забезпечуючи її надійністю та продуктивністю. Нижче на рисунку 2.3 наведено діаграму послідовностей.

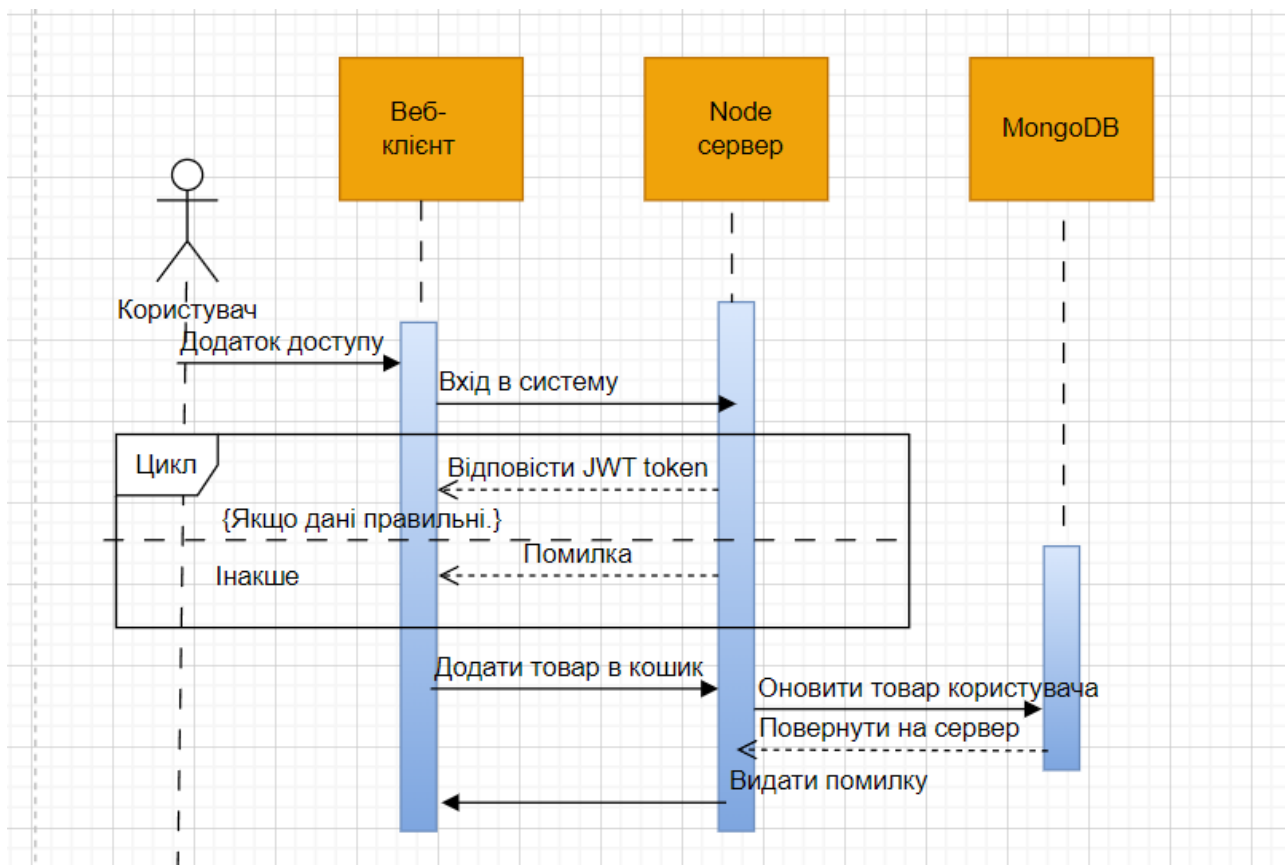


Рисунок 2.3 – Діаграма послідовностей

На діаграмі інтернет магазину сантехнічних виробів зображено основні компоненти:

- Веб-клієнт це спеціальний інтерфейс користувача за допомогою якого він взаємодіє з сервером. Цей клієнт відповідає за отримання запитів від користувача. Після цього він надсилає їх на сервер та відображає результат обробки запитів. Компонент надає простоту і зручність використання користувачам, котрі відвідали інтернет магазин.

- Node сервер це компонент який обробляє запити від веб-клієнта, забезпечує логіку бізнес процесів у структурі, аутенфікує користувачів. Також володіє значною функцією та генерує токени JWT для забезпечення безпеки передачі даних. Всі запити які він обробляє стають захищеними та досягають свого кінцевого призначення.

- MongoDB слугує так званим сховищем всіх даних, зберігає в собі інформацію про користувачів, характеристики товарів та замовлень [18]. Ця модель забезпечує швидке та гнучке зберігання оброблених даних. За допомогою неї можна з легкістю масштабувати всю систему інтернет магазину та надати високу продуктивність при обробці важкої інформації.

- Користувач: компонент який включає в себе всі дії, які повинен виконувати відвідувач магазину. Відповідає за операції входу, перегляд та додавання товару в корзину. Користувач взаємодіє з веб-інтерфейсом клієнта, який забезпечує зручний доступ до функцій інтернет-магазину.

Проаналізуємо ключові функції цієї діаграми:

- Запит на вхід: користувач при в ході в систему вводить свої облікові дані логін і пароль та надсилає запит через веб-клієнт. Веб-клієнт приймає його дані та надсилає HTTP-запит на сервер для перевірки автентичності [19].. Якщо всі дані правильні користувач успішно увійшов в іншому випадку система попросить його перевірити правильність введених даних. Ця процедура буде описана в обробці запиту на сервер.

- Обробка запиту на сервер: Node сервер отримує запит на вхід і перевіряє дані які ввів користувач. Спершу сервер шукає ці дані у сховищі MongoDB, щоб знайти користувач з таким обліковим записом. Після цього при успішному

знаходженні генерує JWT токен для підтвердження автентичності користувач і відправляє цей токен назад до веб-клієнта. Якщо сервер не знайшов відповідних даних в базі то просто відправить повідомлення про помилку авторизації та попросить відвідувача ввести їх повторно.

– Відповідь користувачу веб-клієнт отримує надісланий токен або повідомлення про помилку входу. Якщо надійде JWT токен то користувач авторизується і отримує доступ до всіх функцій платформи. При помилці просто відобразить повідомлення про невдалий вхід.

– Процедура додавання товару в корзину – після успішного входу відвідувач може обрати підходящий для нього товар. Далі надсилається запит на додавання товару в корзину через веб-клієнт. Вся ця дія займає невеликий інтервал часу в інтернет магазині на сторінці товару користувач просто використає кнопку «Додати до корзину». Далі формується HTTP-запит з інформацією про товар і надсилає його на сервер.

– Обробка запиту та взаємодія з MongoDB: Сервер отримує запит на додавання товару, але спочатку йому слід звернутися до сховища для оновлення інформації про корзину. Також це включає додавання нового товару зі списку у корзину користувача. БД оновлює надіслану інформацію про товари і повертає результат оброки на сервер. Після цього надходить успішне додавання товару або поверне помилку у разі, якщо додавання не вдалося. Наступним кроком йде відповідь клієнту за допомогою обробки відповіді від MongoDB і формується відповідний HTTP-запит.

– Відображення результату: компонент сервер надсилає підтвердження успішного додавання товару до корзину або повідомлення про помилку, яке включає в себе опис проблеми. Клієнт отримує результат запиту і демонструє користувачу відповідь повідомлення. Якщо все пройшло без помилок, то він отримує повідомлення про успішне додавання товару в корзину. Також користувач побачить оновлений кошик або вікно з помилкою.

Забезпечення коректної взаємодії між цими компонентами є важливим фактором для роботи інтернет магазину, підвищує задоволеність користувачів і демонструє високий рівень безпеки.

2.6 Діаграма класів

Структура бази даних та взаємодія між основними сутностями є головним показником сучасного конкурентоспроможного інтернет магазину. Необхідно щоб всі елементи працювали ефективно без втрати швидкості, тому було прийнято рішення спроектувати діаграму класів. Для інтернет-магазину сантехнічних виробів діаграма дозволяє зрозуміти, як всі дані організовані в структурі та як вони налагоджують зв'язок між собою. В свою чергу це важливо для забезпечення інформації про сантехнічні товари та аргументованої оцінки про їх якість від користувача. На рисунку 2.4 представлено діаграму класів інтернет магазин сантехнічних виробів.

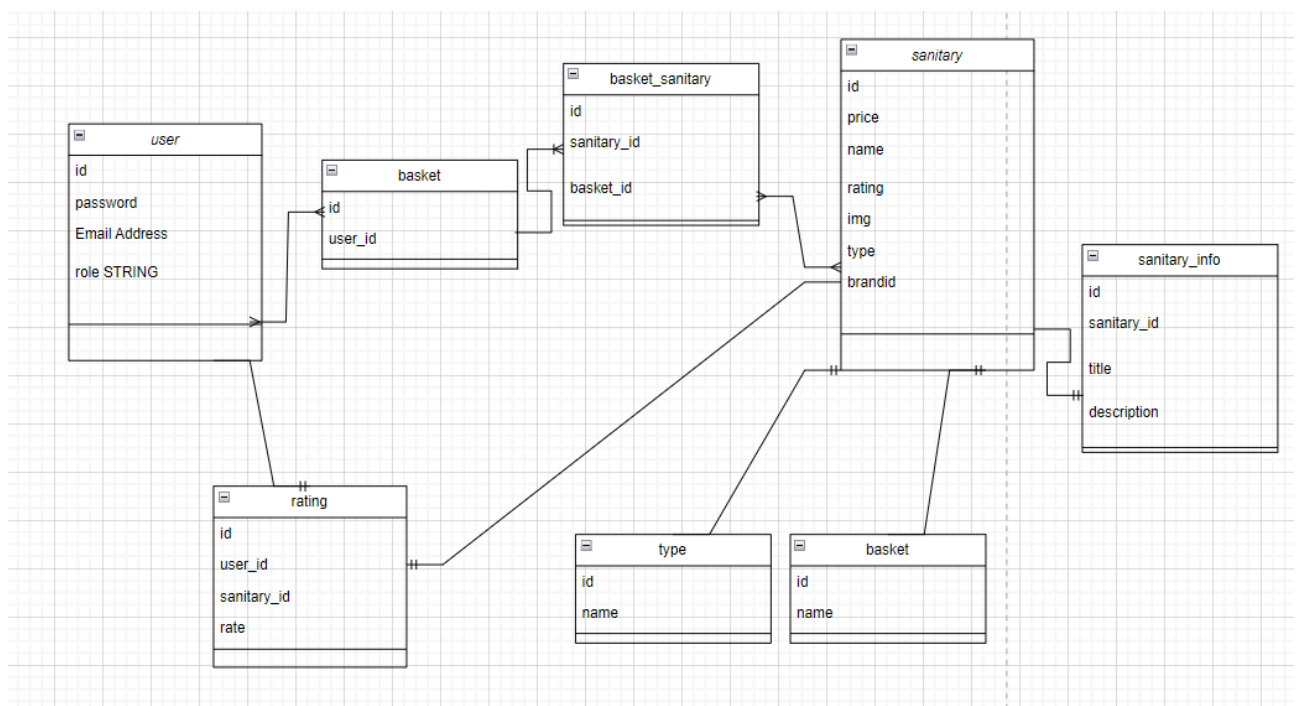


Рисунок 2.4 – Діаграма класів

На цій діаграмі представлені різні типи відношень між таблицями баз даних:

– Один до багатьох – для цього типу характерно, те щоб кожен запис в одній таблиці мав багато відповідних записів іншій таблиці, але кожен цей запис в іншій таблиці має лише один відповідний який використаний у першій з таблиць. Тип зв'язку переважно застосовується, коли один запис може мати

багато пов'язаних даних. Для прикладу замовлення в інтернет магазині може містити багато продукції.

– Один до одного – цей тип запису переважно використовується, коли відповідна інформація пов'язана з одним записом поділена між двома таблицями для зручності роботи даних. Кожен запис в одній таблиці має характерний запис в іншій таблиці і навпаки. Кожен запис в одній таблиці має відповідний запис в іншій таблиці і навпаки. Цей тип зв'язку зазвичай використовується, коли інформація, пов'язана з одним записом, розподілена між двома окремими таблицями для зручності роботи з даними [20]. Також забезпечує цілісність даних, які не дублюються і зберігаються в окремих таблицях для кращого керування системи. Чутливі дані знаходяться окремо від головних записів та обмежують доступ до них.

– Багато до багатьох – такий тип зв'язку в реляційних сховищах даних означає, що кожен запис в одній таблиці містить багато відповідних записів в іншій або навпаки. Завдяки такому зв'язку запис пов'язується з багатьма записами в іншій таблиці та отримує інформацію від неї. Часто для реалізації цього компоненту включають в роботу додаткову таблицю, яка містить зовнішні ключі обох з таблиць.

В інтернет магазині «Aquatech» є кілька основних компонентів, кожен з яких представлений на діаграмі класів. Всі вони виконують важливу роль для працездатності всього механізму, тому було прийнято рішення розглянути їхні спеціальні функції.

Користувач представляє всіх відвідувачів інтернет магазину, які можуть бути потенційними покупцями або виконувати роботу адміністратора. Кожен користувач має свій власний унікальний ідентифікатор. Також для входу в обліковий запис користувачу необхідно мати спеціальні елементи пароль та електронну пошту для підтвердження особистості. З першу система не розуміє чи користувач є звичайний покупець чи адміністратором. Для вирішення цієї проблеми було прийнято рішення надати класу користувач спеціальний атрибут «role», який визначає його права в системі. Зв'язок між користувачем та корзиною вказує на те, що один користувач може мати кілька корзин.

Сховище товарів представляє корзину в якій містяться товари, вибрані користувачем для подальших дій покупки або зміни сантехнічного товару на інший якщо його характеристики не відповідають бажаним. Цей атрибут має кілька ідентифікаторів свій власний та користувача, який вказує, якому відвідувачу належить корзина. Зв'язок між і користувачем є відношенням багато-до-одного, який демонструє те, що один користувач може мати кілька корзин, але кожна корзина належить тільки одного користувача.

«Sanitary» демонструє товари які користувач може придбати в інтернет магазині. Кожен продукт має унікальний ідентифікатор та свої відповідні характеристики. Для характеристики товарів належать їхня ціна , назва, рейтинг, зображення, тип та ідентифікатор компанії від якої представлений товар. Товари мають певний зв'язок через клас «basket_sanitary» через відношення багато-до-багатьох, що в свою чергу показує, що один вид товару може бути у багатьох корзинах і один кошик може містити велику кількість продукції.

«Basket-sanitary» є ключовим компонентом у структурі сховища даних для інтернет магазину сантехнічних виробів. Він реалізує відношення багато-до-багатьох між корзинами покупців, що дозволяє одному виду товару бути присутнім у кількох корзинах. Також слід відмітити, що структура бази даних завжди відображає актуальну кількість товарів завдяки відповідному запису який видаляє запис після відмови користувача. За допомогою цього класу можна отримати всю необхідну інформацію про сантехнічний продукт, та аналіз даних на основі поведінки відвідувача, які товари найчастіше додаються в корзину, але з певних причин не купуються. З огляду на це можна використати ці дані для збільшення продажів та виправлення помилок.

У системі присутній рейтинг товару, який дозволяє користувачам оцінювати товар для кращого розуміння зацікавленості в ньому відвідувачів та актуальності на ринку. Думка користувачів є найголовнішим аспектом для продажу сантехнічного продукту. Вона допомагає прийняте обґрунтоване рішення для покупки завдяки інформації про популярність і якість товару. Для користувача клас rating дає змогу залишити багато відгуків про різні товари. Це значить, що один користувач в силі дати експертну оцінку та підсилити

репутацію магазину. Допустимо, відвідувач зробив покупку сантехнічного виробу і хоче залишити відгук про якість товару. Ця оцінка створює новий запис у цьому класі з ідентифікатором цього користувача, ідентифікатором продукту і значенням, яке ввів користувач. Якщо інший покупець залишить додаткову оцінку то середній рейтинг товару оновиться і відобразиться на екрані.

Клас «type» забезпечує логічну структуру інтернет магазину, полегшує пошук товару для покупця та спрощує процес керування для адміністратора. Наприклад адміністратор створює нову замітку та додає новий елемент, кожен з цих елементів буде мати посилання на тип цієї замітки. Це все допомагає швидко відображати продукт під відповідну категорію. Кожен сантехнічний виріб в інтернет магазині належить до певного типу, що в свою чергу дозволяє вдосконалити каталог та полегшує навігацію для користувача.

Клас «sanitary_info» забезпечує зберігання та керування додатковою інформацією про сантехнічні вироби. Він надає користувачам характеристики та особливі відомості про товар, що підвищує розуміння покупця і довіру до продукції.

2.7 Висновок до другого розділу

В другому розділі кваліфікаційної роботи встановлено, що чітке розмежування клієнтської та серверної частини, використання сучасних технологій. Розроблено детальний опис діаграми послідовностей та діаграми класів, щоб зрозуміти взаємодію між основними компонентами системи та для подальшого вдосконалення інтернет-магазину. Запропонована архітектура та функціональні вимоги гарантують надійну та ефективну роботу системи, задовольняючи потреби як користувачів, так і адміністраторів.

Ці результати створюють міцну основу для подальшого розвитку інтернет-магазину «Aquatech», забезпечуючи його конкурентоспроможність.

РОЗДІЛ 3. РОЗРОБКА ІНТЕРНЕТ МАГАЗИНУ

3.1 Початок роботи з React

Розробка інтерфейсу користувача за допомогою React важливий етап, який визначає, наскільки зручно користувачам буде здійснювати покупки. Інтерфейс має бути інтуїтивно зрозумілим, функціональним та візуально привабливим для користувачів. Перед розробкою потрібно врахувати кілька етапів. Розглянемо їх нижче.

Встановлення Node.js. Щоб працювати з повноцінним функціоналом React потрібно спершу інсталиювати ці компоненти. Це дозволить запускати Javascript на сервері, а компонент «npm» використовувати для управління пакетами важливо необхідними для проєкту.

```
PS D:\project\AquaTech> node -v
v20.13.1
PS D:\project\AquaTech> npm -v
10.5.2
```

Рисунок 3.1 – Створення функцій для проєкту

Встановлення React App. Інструмент який дозволяє швидко та ефективно створювати нові проєкти на React. Він автоматично підбирає та налаштовує характеристики середовища для створення проєкту.

Ініціалізація React. На цьому етапі слід встановити React на персональний комп'ютер. Далі слід ініціалізувати та створити проєкт. Нижче на рисунку представлено компоненти після ініціалізації.

```
"scripts": {
  "start": "node backend/server.js",
  "server": "nodemon backend/server.js",
  "client": "npm start --prefix frontend",
  "dev": "concurrently \"npm run server\" \"npm run client\"",
  "data:import": "node backend/seeder.js",
  "data:destroy": "node backend/seeder.js -d",
  "build": "npm install && npm install --prefix frontend && npm run build --prefix frontend"
},
```

Рисунок 3.2 – Компоненти після ініціалізації

Створення компонентів інтерфейсу. Цей етап відіграє важливу роль в розробці інтернет магазину. Тут потрібно обрати основні елементи які будуть потрібні та розділити їх для зручності на окремі компоненти. Вони будуть незалежними модулями, котрі містять в собі логіку та відображають структурованість та повторне використання коду.

3.2 Створення клієнтової частини

Клієнтська частина інтернет-магазину сантехнічних виробів використовує сучасні технологія, такі як React, що дозволяє створити динамічний і швидкий інтерфейс. Вона забезпечує для користувача комфортне перебування на платформі та легкість пошуку інформації товару.

Основні завдання клієнтської частини:

- розробка інтуїтивно зрозумілого дизайну інтерфейсу;
- забезпечення високої продуктивної системи;
- реалізація навігації пошуку;
- інтеграція з серверною частиною;
- забезпечення безпеки даних.

Створення проєкту на «Aquatech» зображено на рисунку 3.3.

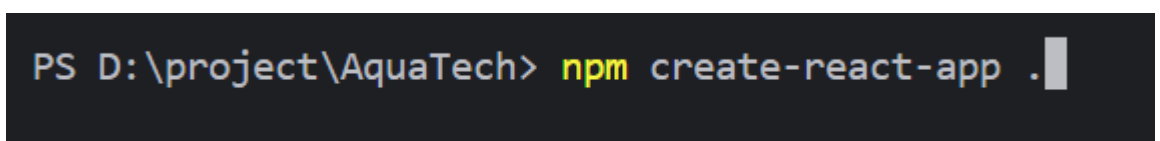


Рисунок 3.3 – Створення проєкту для інтернет-магазину

Розробка клієнтської частини є комплексним процесом, який створює базову структуру проєкту, розробку компонентів інтерфейсу, включає інтеграцію з сервером. Використання сучасних методів реалізації та підходів дозволяє створити зручний та продуктивну систему, що забезпечує успіх інтернет-магазину.

Нижче на рисунку 3.4 представлено розпакування проєкту після використання команди створення.

Важлива дія яка включає в себе подальшу реалізацію інтерфейсу інтернет магазину.

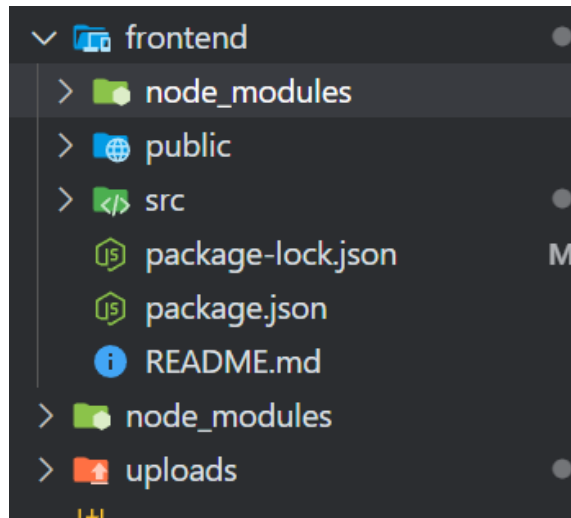


Рисунок 3.4 – Розпакування проєкту

Після налаштування середовища розробки необхідно згрупувати та організувати файли та директорії проєкту, щоб забезпечити зручність роботи та підтримку коду. Розглянемо декілька директорій

- src/components: директорія для зберігання всіх елементів інтерфейсу;
- src/pages: директорія для сторінок додатку;
- src/styles: директорія для стилів сторінок;
- src/services: директорія для файлів, які взаємодіють з API.

Нижче на рисунку 3.5 наведено детальний опис організації проєкту, який допоможе структурувати всі елементи правильно та ефективно .

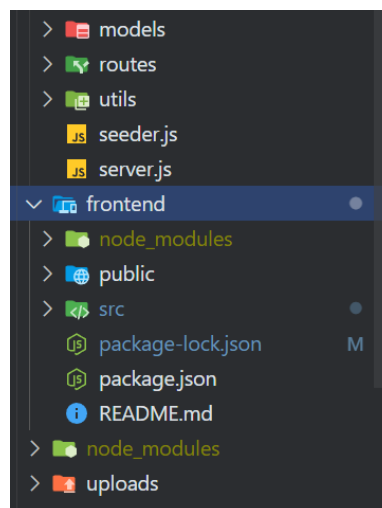


Рисунок 3.5 – Організація проєкту

Організація всіх файлів та директорій проєкту є важливим етапом у розробці клієнтської частини інтернет магазину сантехнічних виробів. Правильна структура проєкту допомагає уникнути помилок та забезпечує ефективну роботу проєкту. Логічна структура дає змогу легко відтворювати необхідні файли і компоненти. З правильно організованою структурою можна з легкістю добавляти нові компоненти. В директоріях знаходяться файли, які слугують провідником та обслуговуються сервером. Це все сприяє хорошу функціоналу без ризику втратити контроль над базовою структурою. Чітка організація директорій та файлів полегшує розробку. Добре організований інтернет магазин легше підтримувати, оновлювати та впроваджувати нові функції для покращення працездатності.

3.3 Дизайн інтерфейсу користувача

Для інтернет магазину сантехнічних виробів важливим аспектом, який відповідає за зовнішній вигляд та утримування відвідувачів на головній сторінці є інтерфейс. Для ефективності роботи ресурсу дизайн повинен бути інтуїтивно зрозумілим, функціональним та естетично привабливим. Розглянемо основні принципи і елементи інтерфейсу для інтернет магазину:

- Простота та зручність – користувач який перший раз відвідав магазин повинен легко знаходити важливу для нього інформацію, переглядати вміст товару та додавати його в корзину без перешкод для кінцевого замовлення.

- Доступність – для цього потрібно впровадити кольорову гаму яка буде доступної для всіх користувачів за допомогою шрифту, розміру тексту та якісного зображення продукції.

- Мінімалізм – важлива інформація не має зливатись з фоном головної сторінки. Також вона повинна бути виділена і легкодоступна. Всі елементи інтерфейсу мають мати відповідне місце розташування, бути консистентними та не сильно об'ємними. Правильно розроблена сторінка забезпечує ефективну роботу з елементами і спрощує процес розробки.

На рисунку 3.6 зображено головну сторінку інтернет магазину «Aquatech».

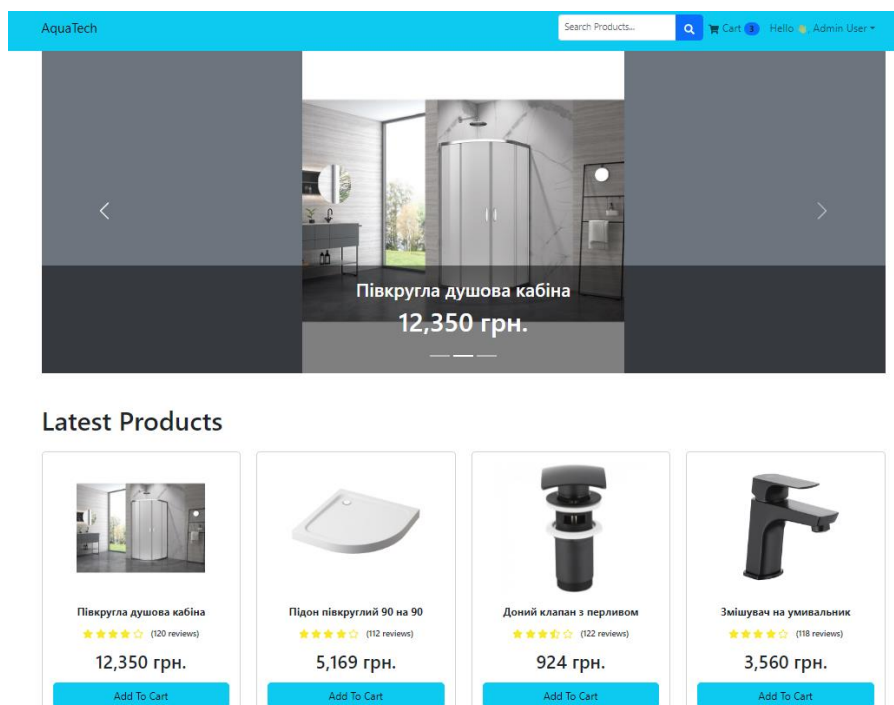


Рисунок 3.6 – Головна сторінка інтернет магазину «Aquatech»

Головна сторінка є ключовим компонентом інтернет магазину. Вона інтуїтивно зрозуміла, добре оформлена та приваблива. На головній сторінці присутні основні елементи такі як рядок пошуку, панель навігації, список товарів.

Основні елементи інтернет магазину сантехнічних виробів:

– Панель навігації – елемент є найважливішим на головній сторінці. Він забезпечує зручний і швидкий доступ до основних розділів інтернет магазину та допомагає орієнтуватися користувачам та знаходити важливу інформацію. Більшість посилань розташовані у верхній частині сторінки, щоб відвідувачі могли легко їх знаходити. Також присутній активний стан, щоб користувачі могли бачити на якій сторінки вони знаходяться. Всі посилання та кнопки на цій панелі є інтерактивними та надають зворотній зв'язок при взаємодії з ними.

– Рядок пошуку – компонент який дозволяє користувачам швидко знаходити необхідні товари за допомогою введення ключових слів. Також під час введення спеціальних слів відвідувачі можуть натиснути на кнопку пошуку для виконання поставленої цілі пошуку. Цей елемент повинен бути інтегрований з серверним компонентом для отримання відповідних результатів пошуку.

Функція підбору відповідних слів для спрощення пошуку покращує систему, пропонуючи варіанти певного товару під час введення.

– Список товарів – відображає всі доступні товари у вигляді карток з основною інформацією про кожен продукт. Він мусить бути зручним та функціональним, щоб забезпечити швидку взаємодію між користувачем і товаром. Цей елемент інтернет магазину містить зображення виробу, його ціну, тип, найменування та кнопку для зберігання до кошика. Картки на яких зображений продукт повинні бути інтерактивними, надаючи покупцеві можливість переглянути детальні характеристики та взаємодіяти з іншими функціями. Список легко інтегрується з рядком для швидкого пошуку товарів за ключовими словами в системі. Також важливим чинником є фільтрація товарів яка дозволяє користувачам обмежити відображення сантехнічних виробів за певними критеріями. Все це в комплексі покращує думку відвідувача про інтернет магазин.

Головна сторінка інтернет магазину повинна бути ретельно спроектована та реалізована для забезпечення інтуїтивної навігації, привабливий дизайн та функціонал забезпечить більшу конверсію покупок та прихильників продукту.

Лістинг коду 3.1 – Програмний код для зберігання загальної кількості.

```
const ProductListPage = () => {
  const [currentPage, setCurrentPage] = useState(1);
  const [totalPage, setTotalPage] = useState(0);
  const [total, setTotal] = useState(0);
  const [limit, setLimit] = useState(0);
  const [skip, setSkip] = useState(0);

  const { data, isLoading, error } = useGetProductsQuery({
    limit,
    skip
  });
};
```

Наступним кроком слід візуально розглянути сторінку з детальним описом про товар.

Сторінка характеристики товару є впливовим елементом на думку користувача і продуктивність самої системи. Візуально її представлено на рисунку 3.7.

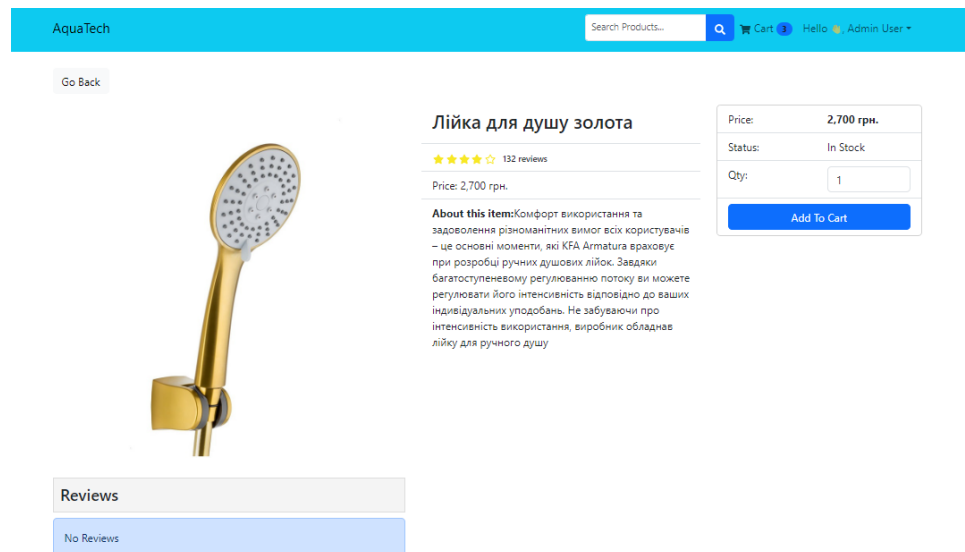


Рисунок 3.7 – Характеристики товару

Продуманий структурований дизайн, включає всі необхідні елементи та акцентує увагу на зручності використання. Розглянемо основні компоненти цієї сторінки більш детально.

Давайте розглянемо основні компоненти цієї сторінки детальніше.

Основні компоненти сторінки:

- зображення товару;
- назва виробу;
- ціна сантехнічного продукту;
- тип товару;
- кількість на залишку;
- кнопка Додати в кошик;
- повернення на головну сторінку.

Зображення є основним візуальним аспектом, що привертає увагу відвідувача і допомагає йому уявити як виглядає сантехнічний виріб. Висока якість зображення допомагає збільшити довіру до продукту.

Назва виробу переважно написана жирним шрифтом для привернення уваги покупця, щоб акцентувати його увагу на потрібному його продукті. Ціна

також має бути добре помітною, щоб не змушувати користувача довго шукати вартість продукту.

Лістинг 3.2 – Програмний код для входу користувача

```
export const usersApiSlice = apiSlice.injectEndpoints({
  endpoints: builder => ({
    login: builder.mutation({
      query: data => ({
        url: `${USERS_URL}/login`,
        method: 'POST',
        body: data
      })
    })
  })
})
```

Опис товару містить важливу інформацію про технічні характеристики та специфіку сантехнічного продукту, щоб наголосити на певній інформації, яку простий відвідувач може не знати.

Далі було прийняте рішення представити вхід та реєстрацію в інтернет магазині. Ці компоненти зображені на рис 3.8.

Пошук продуктів... кошик 3 Увійти

Увійти

Адреса електронної пошти

Пароль

Залишати мене в обліковому записі. [Забули пароль?](#)

Новий клієнт? [зареєструватися](#)

Рисунок 3.8 – Вхід в структуру інтернет магазину

Вхід в систему здійснюється за допомогою окремої сторінки. Якщо користувач ще не має власного облікового запису то система попросить його

zareєструватися та ввести свої ідентифікаційні дані, але якщо користувач вже має свої облікові дані, то він може просто ввійти.

Сторінка входу складається з декількох важливих компонентів, які допомагають користувачам легко увійти в свій профіль:

- заголовок сторінки;
- електронна пошта;
- пароль;
- кнопка входу.

Сторінка реєстрації надає покупцям можливість створити новий профіль, завдяки спеціальній інформації та має такі компоненти:

- заголовок сторінки;
- поле для імені;
- поле для ведення електронної пошти;
- новий пароль;
- кнопка реєстрації.

Обидві сторінки мають мінімалістичний дизайн, що сприяє простоті використання. Вони виконані у дуже схожому стилі та мають базові чітко структуровані поля для введення інформації. Також сторінки забезпечують конфіденційність користувачів за допомогою механізму автентифікації та реєстрації в інтернет магазині.

Всі поля для зручності користувачів підписані, щоб інтуїтивно можна було зрозуміти, що і як робити. Крім того, обидві сторінки оптимізовані для різних пристроїв. Адаптивний дизайн гарантує, що всі елементи інтерфейсу залишаються зручними незалежно від моделі та розміру екрана пристрою.

Якщо брати до уваги безпеку, то сторінки входу на реєстрації підтримують сучасні протоколи захисту даних. Коли користувач вводить дані через захищені канали підтримки знижується ризик витоку конфіденційної інформації.

Система автентифікації забезпечує додатковий рівень. Для цього вона порівнює відповідність введених даних перед наданням доступу до особистого профілю користувача. Сторінки входу та реєстрації не лише виконують свої

основне призначення, але й забезпечують зручність та безпеку всього механізму для потреба користувача.

Після успішної реєстрації облікового запису відвідувач перенаправляється на сторінку підтвердження електронної пошти. Ця сторінка забезпечує верифікацію користувачів, дозволяючи переконатись, що вся введена інформація під час реєстрації є дійсною і належить відповідному користувачу. Також цей механізм запобігає утворенню кількох профілів користувача для однієї особи, в свою чергу це зменшує число фальшивих аккаунтів і позитивно впливає на інтернет магазин.

На рисунку 3.9 відображено успішне підтвердження електронної пошти користувач.

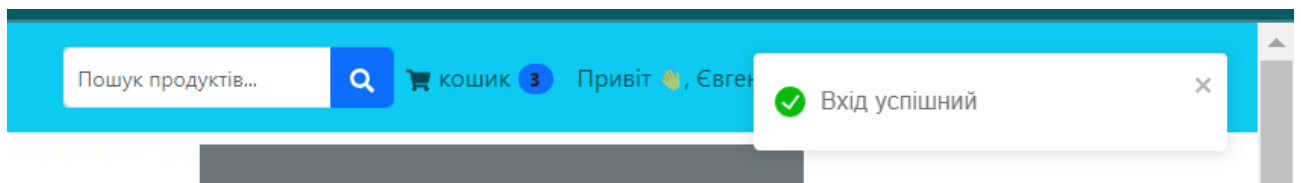


Рисунок 3.9 – Підтвердження про успішний вхід

Основне наповнення цієї сторінки:

- Повідомлення про успішне проходження реєстрації. Цей компонент складається з тексту розташованого у верхній частині сторінки, інформує користувача про те, що новий обліковий запис успішно зареєстрований в системі. Він виконаний в доброзичливішому тоні, що підвищує задоволеність відвідувача.

- Кнопка з підтвердженням електронної пошти. Компонент який містить заклик до підтвердження особистості перед тим, як користувач має отримати доступ до інтернет магазину. Він забезпечує певну безпеку, оскільки допомагає уникнути шахрайських дій.

- Кнопка для повторного надсилання. Корисний елемент сторінки який дозволяє користувачам повторно надсилати інформацію про підтвердження на їх електронну пошту. Слугує проміжним компонентом в тому випадку, коли користувач з першого разу не отримав листа через технічні проблеми системи.

Далі слід розглянути сторінку кошика в інтернет магазині. Вона забезпечує зручність та швидкий процес покупок, оскільки дає можливість користувачам легко керувати своєю корзиною.

Нижче на рисунку 3.10 реалізована ця сторінка.

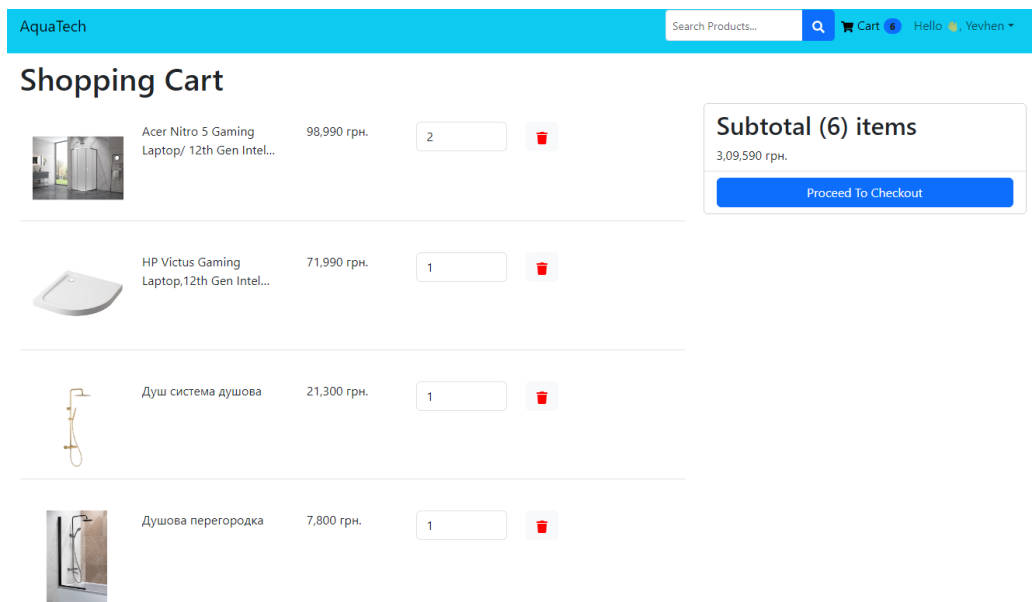


Рисунок 3.10 – Сторінка кошика товару

Сторінка кошика є головним елементом інтерфейсу для інтернет магазину. В ній користувачі можуть переглядати, редагувати та підтверджувати свої обрані продукти перед завершенням покупки.

Основні компоненти функціональності сторінки:

- список продукції;
- заголовок сторінки;
- характеристики про кожен товар;
- кількість вибраного сантехнічного виробу;
- загальна сума покупки;
- оформлення замовлення.

Також реалізовані додаткові функції, наприклад повернення на головну сторінку і можливість видалення товарів. Сторінка забезпечує зручне місцезнаходження необхідних елементів для перегляду або редагування вибраних товарів. Використання чіткої структури полегшує процес покупок для користувача та підвищує кількість успішних транзакцій.

Керування товарними пропозиціями в інтернет магазині сантехнічних виробів є важливим інструментом для користувачів. Було прийнято рішення детальніше розглянути сторінку яка дає змогу виконувати ці функцію. На рисунку 3.11 представлений інтерфейс цієї сторінки, включаючи функціонал для видалення продукції з підтвердженням дії.

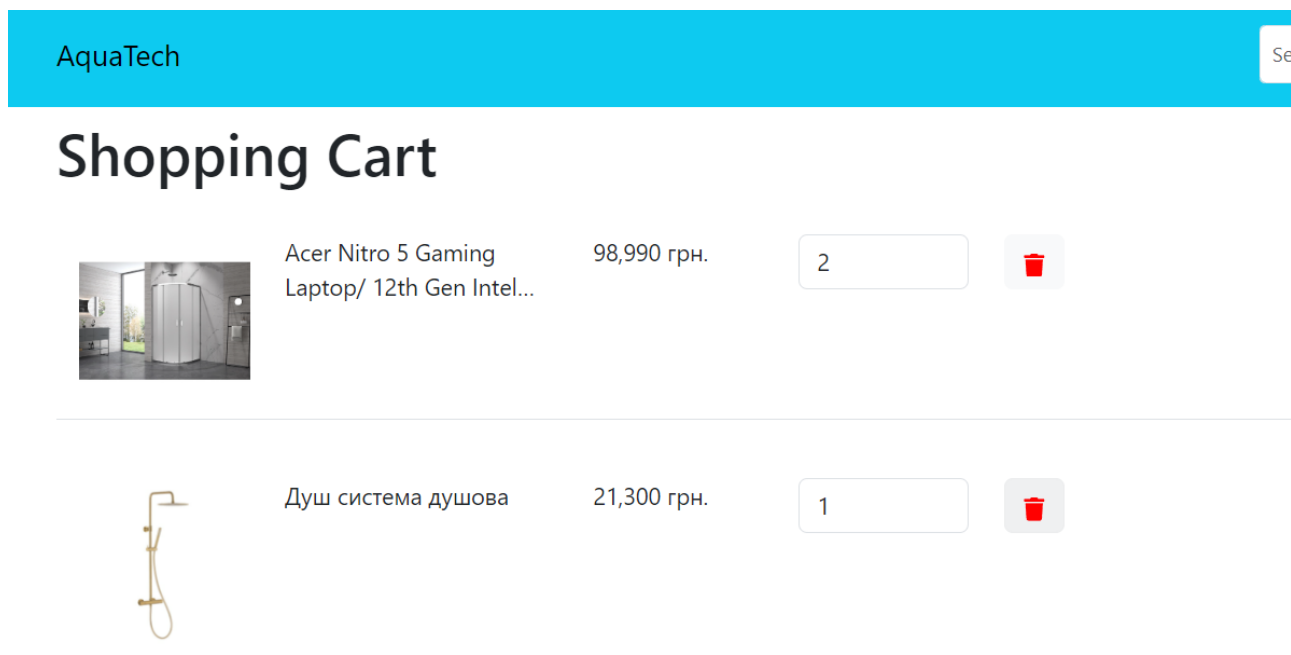


Рисунок 3.11 – Видалення продукції

Ключові компоненти цієї сторінки:

- профіль користувача;
- список продукції;
- історія замовлень;
- видалення сантехнічного продукту.

Ця сторінка дає змогу легко та ефективно стежити за замовленнями та підтримувати актуальність власного вибору. Також за допомогою зрозумілого дизайну користувач без перешкод виконає необхідний процес для видалення товару при необхідності. Пропонуються додаткові функції, такі як історія замовлень та кнопка додавання нових виробів для гнучкості процесу в системі. На кожній картці товару є кнопка видалення, яка дозволяє відвідувачам швидко видаляти непотрібні застарілі пропозиції.

Сторінка входу для адміністратора створена для того, щоб певна людина могла безпечно входити в систему інтернет магазину, використовуючи для цього свій обліковий запис і виконувати необхідні дії для покращення працездатності платформи. В ці дії входить управління контентом та користувачами, оновлення актуальності сантехнічних товарів та інші аспекти роботи інтернет магазину. Розглянемо детально всі складові цієї сторінки та функціональність. Нижче на рисунку 3.12 представлено адміністративну панель.

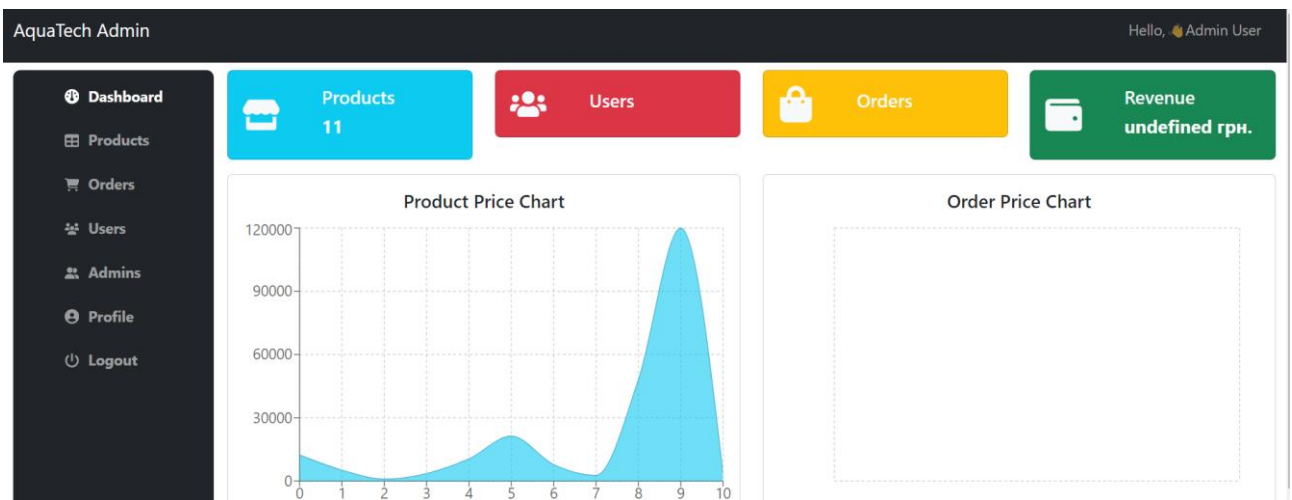


Рисунок 3.12 – Панель адміністратора інтернет-магазину

Правильна організація цієї сторінки дозволяє адміністратору без перешкод входити в систему магазину, забезпечуючи ефективне керування ключовими компонентами.

Основні елементи сторінки:

- форма для введення даних;
- заголовок;
- кнопка входу;
- посилання для повернення.

Адміністративна панель поєднує в собі простоту використання, високий рівень безпеки та інтуїтивно зрозумілий інтерфейс, що робить її ефективним механізмом для адміністратора. Також захищення каналів зв'язку, використання власного ключа доступу підвищує безпеку і надійність всієї системи для успішного управління необхідними аспектами інтернет магазину.

3.4 Створення серверної частини

Серверна частина інтернет магазину є головним етапом, щоб забезпечити взаємодію між клієнтською частиною та сховищем даних. Вона обробляє запити від клієнта, виконує спеціальні операції і повертає необхідні дані назад на клієнтську сторону. На рисунку 3.13 представлена частина реалізації базової серверної частини Node.js з використанням бібліотеки «mongoose» для роботи з базою даних MongoDB.

```
import mongoose from 'mongoose';

const connectDB = async () => {
  try {
    const connection = await mongoose.connect(process.env.MONGO_URI);
    console.log(
      `MongoDB connected successfully on host: ${connection.connection.host}, database: ${connection.connection.database}`
    );
    return connection;
  } catch (error) {
    console.error(`MongoDB connection error: ${error.message}`);
    process.exit(1);
  }
}
```

Рисунок 3.13 – Реалізація базової серверної частини

Для реалізації було використано наступні компоненти:

- створення моделі бази даних;
- реалізація маршрутів;
- запуск сервера;
- налаштування середовища;
- підключення до сховища даних.

Спочатку слід імпортувати необхідні бібліотеки яка забезпечує взаємодію з базою даних MongoDB. Вона дозволяє впровадити схеми для документів та взаємодіяти з ними через JavaScript.

Функціонал розроблений так, якщо під час підключення виникає помилка, вона обробляється у спеціальному блоці. Потім виводиться в консоль з повідомленням про помилку і процес завершиться та акцентує увагу на технічні несправності, які можна з легкістю покращити для працездатності сервера.

Далі було прийнято розглянути елемент відправки повідомлень через сервер для першого входу користувача до інтернет магазину. Цей компонент використовується для реєстрації в інтернет-магазині.

Детальніше його відображено на рисунку 3.14 .

```
import nodemailer from 'nodemailer';  
  
const transporter = nodemailer.createTransport({  
  host: process.env.EMAIL_HOST,  
  port: process.env.EMAIL_PORT,  
  secure: false,  
  auth: {  
    user: process.env.EMAIL_USER,  
    pass: process.env.EMAIL_PASS  
  }  
});  
  
export default transporter;
```

Рисунок 3.14 – Реєстрація в інтернет-магазині

Налаштування для відправки листів є відносно простим етапом, проте не менш важливим. Використання змінних середовищ для зберігання конфіденційних даних забезпечує безпеку і гнучкість налаштування. Після цього спеціальний транспортер легко відправляє електронні листи, використовуючи різні конфігурації. Цей підхід є прекрасним рішенням для реалізації функцій пов'язаних з автоматизацією відправки.

Для реалізації серверної частини слід врахувати безпечність всіх даних. Тому було прийнято рішення впровадити хешування паролів для захисту конфіденційної інформації у разі компрометації бази даних. Застосування хешування паролів знижує ризики пов'язані з витоків даних.

Для прикладу розглянемо створення панелі адміністратора із захищеним паролем.

Нижче на рисунку 3.15 наведено реалізацію цього процесу.

```
import bcrypt from 'bcrypt';  
⚡  
const users = [  
  {  
    name: 'Admin User',  
    email: 'admin@admin.com',  
    password: bcrypt.hashSync('admin123', 10),  
    isAdmin: true  
  },  
  {
```

Рисунок 3.15 – Захист паролю

Наступним кроком є створення масиву користувачів. З прикладу який представлено вище створюємо одного користувача з роллю адміністратора. Кожен об'єкт якого містить кілька полів:

- ім'я;
- електрона пошта;
- пароль;
- вказівник на користувача.

Для забезпечення високого рівня безпеки в інтернет-магазині сантехнічних виробів необхідно впровадити надійні методи хешування. Найефективнішим інструментом для цього завдання є «bcrypt».

Цей інструмент є одним з найпопулярніших та надійних методів хешування паролів в розвинутому світі технологій. Проста інтеграція та використання певних алгоритмів та можливість налаштовувати кілька циклів хешування забезпечують захист особистих даних різних видів атак.

Розглянемо компонент для отримання списку продуктів з бази даних з підтримкою пагінації та пошуку відповідних параметрів. Він забезпечує ефективно управління великої кількості продукту та надає користувачам зручні можливості для знаходження необхідних товарів. Для реалізації використовують асинхронні операції для перевірки параметрів запиту.

Реалізація компоненту представлена на рисунку 3.16.

```
const getProducts = async (req, res, next) => {
  try {
    const total = await Product.countDocuments();
    const maxLimit = process.env.PAGINATION_MAX_LIMIT;
    const maxSkip = total === 0 ? 0 : total - 1;
    const limit = Number(req.query.limit) || maxLimit;
    const skip = Number(req.query.skip) || 0;
    const search = req.query.search || '';

    const products = await Product.find({
      name: { $regex: search, $options: 'i' }
    })
      .limit(limit > maxLimit ? maxLimit : limit)
```

Рисунок 3.16 – Отримання списку продукції

Інструмент дозволяє зменшити навантаження на сервер і базу даних, забезпечуючи при цьому швидкий та легкий доступ користувачів до великого каталогу продукції. Такий метод використовує організацію пошуку та пагінації та є оптимальним для інтернет магазину сантехнічних виробів.

Також багато функцій які впроваджені в цей інструмент забезпечує ефективно управління великою системою. Вони отримують параметри запиту від користувача і використовують його для здійснення пошуку за допомогою ключового виразу. Це дозволяє користувача легко знаходити потрібні продукти за частковим збігом. Після обробки всіх параметрів функція виконує запит до сховища даних для отримання цілого списку, що відповідають критеріям пошуку. Також функції обмежуються кількістю результатів за допомогою ліміту пропускають непробні частину за допомогою реалізованих методів.

3.5 Висновок до третього розділу

У третьому розділі розглянуто процес створення серверної та клієнтської частини інтернет-магазину з використанням React, Node.js і MongoDB. Включено налаштування бази даних, реалізацію маршрутів, запуск сервера та забезпечення безпеки даних, включаючи хешування паролів.

4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Вимоги до освітлення виробничих приміщень

У ДБН України «Будинки та споруди. Громадські будинки та споруди», затверджених наказом Держбуду України від 4 серпня 1999 року, та у ДБН В.2.5-28-2006 «Інженерне обладнання будинків і споруд. Природне і штучне освітлення», затверджених наказом Міністерства будівництва, архітектури та житлово-комунального господарства України від 15 травня 2006 року № 168, встановлено нормативні показники освітленості для громадських і адміністративно-побутових споруд.

Освітлення – це отримання, розподіл та використання світлової енергії для забезпечення нормальних умов праці. Світло впливає на діяльність людини. При недостатньому освітленні людина працює менше продуктивно, швидко втомлюється, зростає потенційна небезпека помилкових дій і нещасних випадків. Погане освітлення може призвести до порушення функції зорового аналізатора, розвитку професійних захворювань. Хороше освітлення впливає на центральну нервову систему, поліпшує функції інших аналізаторів, сприяє бадьорому настрою, підвищує працездатність.

Освітлення має бути достатнім, рівномірним, щоб були видні дрібні деталі. Не має бути: надмірного освітлювального потоку, різких контрастів, затінення.

Під час проектування освітлення робочого місця необхідно враховувати: освітленість у різних площинах (горизонтальній і вертикальній), зміну світла в різний час дня і року, за різної погоди, а також пофарбування службових приміщень і предметів, що в них розміщені, або інші пристрої, за винятком приміщень, у яких за технологічними вимогами не допускається проникнення зовнішнього повітря.

У громадських будинках має бути забезпечено природне і штучне освітлення, а також інсоляція (опромінення поверхонь сонячним світлом). Проектування приміщень, де працюють працівники юридичних професій, не допускається без природного освітлення і тільки у виняткових випадках у

складах та спеціальних лабораторіях, жіночих туалетах дозволяється використання лише штучного освітлення. Природне освітлення може бути верхнім – через світлові ліхтарі даху, бічним – через вікна в стінах і комбінованим – через ліхтарі і вікна і забезпечувати його коефіцієнт не нижче ніж 1,5 %.

Вид необхідного природного освітлення встановлюється на основі розмірів (ширини) приміщення. За ширини приміщення до 12 м рекомендується бічне одностороннє освітлення, при ширині більше 12 м і до 24 м – бічне двостороннє. Якщо ж ширина приміщення більша 24 м, то освітлення бажано мати комбінованим. Існує п'ять типів освітлення службових приміщень: пряме, напівпряме, пряме косо, напівкосо, косо. Для більшості приміщень найбільш придатне використання напівкосого освітлення, коли від 10 до 40 % світла падає вниз і від 60 до 90 % світла спрямовується вверх, відбиваючись потім від стелі. За допомогою цього досягається добре розсіювання світла, зменшується осліплюючий ефект. Для різного виду робіт потребується різне освітлення, але варто пам'ятати, що природне освітлення є найбільш сприятливим. За умови його наявності вдається прочитувати на 5-10 % більше сторінок друкованого тексту, аніж за умови штучного освітлення. При недостатньому за нормами природному освітленні використовують додатково і штучне. Таке освітлення називається суміщеним.

Штучне освітлення також має відповідати гігієнічним вимогам, що визначені у ДБН В 2.5-28-2006. Воно може бути загальним, місцевим та комбінованим. Система загального освітлення приміщень передбачає розміщення світильників під стелею, таким чином, щоб забезпечити рівномірний світловий потік або його локалізацію над певною групою обладнання. Велике значення набуває висота встановлення світильників над підлогою. Найбільш раціональна висота для світильників з числом люмінесцентних ламп до чотирьох – 2,5 м, а при чотирьох і більше – 3,2 м.

Джерела штучного освітлення не повинні засліплювати. Для цього слід застосовувати відповідну арматуру й дотримуватися регламентованої висоти підвісу світильників. Найбільш раціональна висота для світильників з числом

люмінесцентних ламп до чотирьох – 2,5 м, а при чотирьох і більше – 3,2 м. Комбіноване штучне освітлення у кабінетах й робочих кімнатах судів повинно становити 400/200 лк. Конференцзали і зали засідань повинні бути облаштовані загальним освітленням не менше 300 лк.

4.2 Планування заходів із безпеки праці та контроль за їх виконанням

Планування робіт із безпеки праці передбачає встановлення послідовних рішень щодо створення безпечних і здорових умов праці та у свою чергу поділяється на перспективне оперативне, поточне, у яких визначені задачі підприємству в цілому і окремим структурним підрозділам, а також керівникам і спеціалістам.

Планування робіт здійснюється на основі:

- заходів, що забезпечують досягнення встановлених нормативів безпеки праці, гігієни праці та виробничого середовища;
- заходів, передбачених колективним договором;
- заходів по усуненню недоліків, виявлених при розслідуванні нещасних випадків, професійних захворювань і аварій.

Перспективне планування розробляється на 3-5 років і вміщує найбільш важливі, трудомісткі і довгострокові за терміном виконання заходи з охорони праці, виконання яких, зазвичай, вимагає спільної роботи кількох підрозділів підприємства. Можливість виконання заходів перспективного плану повинна бути підтверджена обґрунтованим розрахунком необхідного матеріально-технічного забезпечення і фінансових витрат із зазначенням джерел фінансування.

Комплексні довгострокові програми поліпшення умов праці та виробничого середовища підприємств передбачають:

- упровадження безпечної техніки і технологій;
- застосування ефективних інженерно-технічних засобів, що забезпечують досягнення встановлених нормативів охорони праці;
- проведення реконструкції санітарно-побутових приміщень;

- заходи щодо заміни шкідливих речовин і матеріалів чи усунення безпосереднього контакту працівників із ними;

- заходи організаційного характеру (вдосконалення системи навчання, впровадження раціональних режимів праці й відпочинку та ін.).

Поточне планування здійснюється у межах календарного року через розроблення відповідних заходів у розділі «Охорона праці» колективного договору.

Оперативне планування роботи з охорони праці розробляються за підсумками контролю стану охорони праці на об'єктах господарювання терміном, зазвичай, на місяць, квартал. Оперативні заходи щодо усунення виявлених недоліків зазначаються безпосередньо у наказі, постанові суб'єкта СУОПШ, які видаються за підсумками контролю, або у плані заходів. Функція планування, в основі якої лежить прогностичний аналіз, має вирішальне значення в системі управління охороною праці. Планування роботи з охорони праці поділяється на перспективне, поточне та оперативне.

Дійове управління охороною праці можна здійснювати тільки за наявності повної, своєчасної й точної інформації про стан охорони праці на об'єкті. Одержати таку інформацію, виявити можливі відхилення від норм та правил із охорони праці, перевірити додержання вимог законодавства з охорони праці та виконання відповідних планів, програм, приписів, прийнятих рішень можна тільки на підставі регулярного та об'єктивного контролю. Тому контроль за станом охорони праці є найбільш відповідальною та трудомісткою функцією процесу управління.

Контроль за станом охорони праці включає:

- виявлення порушення вимог законів і нормативних актів з охорони праці;
- оцінку рівня небезпечних виробничих факторів (НВФ) і шкідливих виробничих факторів (ШВФ) на робочих місцях;
- перевірку усунення раніше виявлених порушень;
- перевірку виконання працівником обов'язків з охорони праці;
- перевірку виконання планів робіт з охорони праці;

– перевірку забезпечення працівників засобами індивідуального захисту (ЗІЗ) і засобами колективного захисту (ЗКЗ).

Контроль за охороною праці може бути:

– відомчим, що здійснюється посадовими особами, повноважними представниками та службами міністерства або іншого центрального органу виконавчої влади, а також асоціації, корпорації, концерну або іншого об'єднання підприємств;

– регіональним, що здійснюється посадовими особами, повноважними представниками та службами місцевих органів виконавчої влади та органів місцевого самоврядування;

– громадським, що здійснюється виборними органами та представниками професійних спілок, інших громадських організацій;

– страховим, що здійснюється страховими експертами з охорони праці Фонду соціального страхування від нещасних випадків на виробництві та професійних захворювань;

– внутрішнім, що здійснюється в межах підприємства (установи, організації) відповідними службами, посадовими особами та громадськими інспекторами (уповноваженими трудових колективів) цього підприємства.

Внутрішній контроль поділяється на: оперативний (повсякденний); такий, що здійснюється службою охорони праці підприємства; громадський; адміністративно-громадський триступеневий.

Оперативний контроль із боку керівників робіт і підрозділів, а також інших посадових осіб підприємства здійснюється згідно із затвердженими посадовими обов'язками.

Служба охорони праці контролює виконання вимог з охорони праці в усіх структурних підрозділах та службах підприємства.

У справі створення здорових та безпечних умов праці значна роль відводиться громадському контролю, який покладено на громадських інспекторів із охорони праці або представників трудових колективів (якщо профспілкова організація на підприємстві не створювалася), а також комісію з

питань охорони праці підприємства (за рішенням загальних зборів або конференції працівників підприємства).

Адміністративно-громадський триступеневий контроль здійснюється на трьох рівнях (ступенях). На першому рівні контролю начальник виробничої дільниці (майстер) спільно з громадським інспектором профгрупи щоденно перевіряють стан охорони праці на виробничій дільниці. На другому рівні начальник цеху спільно з громадським інспектором та спеціалістами відповідних служб цеху (механік, електрик, технолог) два рази на місяць (згідно із затвердженим графіком) перевіряють стан охорони праці в цеху. На третьому рівні контролю щомісячно (згідно із затвердженим графіком) комісія підприємства під головуванням керівника (головного інженера) перевіряє стан охорони праці в цілому на підприємстві. До складу комісії входять: керівник служби охорони праці, голова комісії з охорони праці (або представник профкому), керівник медичної служби, працівник пожежної охорони та головні спеціалісти підприємства (технолог, механік, енергетик). Результати роботи комісії фіксуються в журналі триступеневого контролю і розглядаються на нараді. За результатами наради видається наказ по підприємству.

Слід зазначити, що крім контролю за станом охорони праці, на підприємстві здійснюється державний нагляд за дотриманням законодавчих та інших нормативно-правових актів з охорони праці.

4.3 Висновок до четвертого розділу

У четвертому розділі розглянуто важливі аспекти безпеки життєдіяльності та основи охорони праці в контексті виробничих приміщень та організації робочого процесу. Зокрема, значна увага була приділена вимогам до освітлення виробничих приміщень, що мають відповідати державним будівельним нормам України. Забезпечення належного рівня освітленості є критичним фактором, що впливає на продуктивність праці, зниження втомлюваності та ризиків професійних захворювань.

ВИСНОВОК

В ході виконання кваліфікаційної роботи створено інтернет-магазин сантехнічних виробів «Aquatech».

В першому розділі кваліфікаційної роботи освітнього рівня «Бакалавр»:

- проаналізовано предметну область в сучасних реаліях;
- розглянуто сучасні інструменти для розробки інтернет-магазину, зокрема React для клієнтської частини і Node.js для серверної;
- висвітлено особливості лідерів сантехнічних виробів на сучасному ринку.

В другому розділі кваліфікаційної роботи:

- спроектовано логічну архітектуру системи, що гарантує ефективну роботу та зручний інтерфейс для користувачів;
- створено діаграми класів, послідовностей, варіантів відповідностей для кращого представлення інтернет-магазину;
- визначено основні цілі та функції інтернет-магазину сантехнічних виробів.

В третьому розділі кваліфікаційної роботи:

- розроблено інтерфейс з урахуванням принципів мінімалізму, що робить його інтуїтивно зрозумілим і зручним для користувачів;
- створено клієнтську та серверну частину;
- реалізовано панель адміністратора для кращого керування інтернет-магазином.

У розділі «Безпека життєдіяльності, основи охорони праці», розглянуто важливі аспекти безпеки життєдіяльності та основи охорони праці. Встановлено дотримання нормативних вимог до освітлення виробничих приміщень для забезпечення необхідних умов праці.

ПЕРЕЛІК ДЖЕРЕЛ

1. Quick Start. React. URL: <https://react.dev/learn> (дата звернення: 12.06.2024: 20.06.2024).
2. Getting started with React - Learn web development. MDN Web Docs. URL: https://developer.mozilla.org/en-US/docs/Learn/Tools_and_testing/Client-side_JavaScript_frameworks/React_getting_started (дата звернення: 8.06.2024).
3. A JavaScript library for building user interfaces. React. URL: <https://legacy.reactjs.org> (дата звернення: 9.06.2024).
4. ReactJS Tutorial. Tutorialspoint. URL: <https://www.tutorialspoint.com/reactjs> (дата звернення: 10.06.2024).
5. React Alert – Material UI. MUI. URL: <https://mui.com/material-ui/react-alert/> (дата звернення: 10.06.2024).
6. React UI Component Library for Modern & Responsive Web Apps. Syncfusion. URL: <https://www.syncfusion.com/react-components> (дата звернення: 10.06.2024).
7. REACT info brochure. Issuu. URL: https://issuu.com/nokut/docs/react_info-brochure (дата звернення: 11.06.2024).
8. Node.js v22.3.0 Documentation. Node.js. URL: <https://nodejs.org/en/docs/> (дата звернення: 11.06.2024).
9. Node.js 20: Latest Features and Enhancements. Radixweb. URL: <https://radixweb.com/blog/what-is-new-in-nodejs-20> (дата звернення: 11.06.2024).
10. What is the Node.js runtime environment?. WhatIs. URL: <https://www.techtarget.com/whatis/definition/Nodejs> (дата звернення: 12.06.2024).
11. Main Node.js Advantages and Disadvantages. EPAM. URL: <https://anywhere.epam.com/en/blog/node-js-pros-and-cons> (дата звернення: 13.06.2024).
12. What you need to know about Node.js? Short guide on JavaScript runtime. Merixstudio. URL: <https://www.merixstudio.com/blog/what-you-need-know-about-nodejs-short-technology-guide> (дата звернення: 13.06.2024).

13. MongoDB ClickHouse Docs. ClickHouse. URL: <https://clickhouse.com/docs/en/engines/table-engines/integrations/mongodb> (дата звернення: 13.06.2024).
14. MongoDB documentation guide. Postman. URL: <https://www.postman.com/mongodb-devrel/workspace/mongodb-public/documentation/17898583-25682080-e247-4d25-8e5c-1798461c7db4> (дата звернення: 14.06.2024).
15. MongoDB LIKE Statement Usage. Atlassian. URL: <https://www.atlassian.com/data/databases/how-to-use-a-sql-like-statement-in-mongodb> (дата звернення: 14.06.2024).
16. Express - Node.js web application framework.. URL: <http://expressjs.com/> (дата звернення: 14.06.2024).
17. Express. npm. URL: <https://www.npmjs.com/package/express> (дата звернення: 15.06.2024).
18. Redux Essentials, Part 1: Redux Overview and Concepts. Redux. URL: <https://redux.js.org/tutorials/essentials/part-1-overview-concepts> (дата звернення : 15.06.2024).
19. Learn about Sentry's Redux integration. Sentry. URL: <https://docs.sentry.io/platforms/javascript/guides/react/features/redux/> (дата звернення: 16.06.2024).
20. Building microservices with Nodejs Express and swagger. TopJavaTutorial. URL: <https://www.topjavatutorial.com/nodejs/building-microservices-with-nodejs-express-and-swagger/> (дата звернення: 15.06.2024).
21. MongoDB NestJS technical documentation. NestJS. URL: <https://docs.nestjs.com/techniques/mongodb> (дата звернення: 15.06.2024).
22. Learn JavaScript: Tutorials for all programmers. Stackify. URL: <https://stackify.com/learn-javascript-tutorials/> (дата звернення: 16.06.2024).
23. In Case You Missed It: Redux Has Cool New Features!. Redux. URL: <https://redux.io/redux-has-cool-new-features/> (дата звернення: 16.06.2024).
24. HTTP documentation. MDN Web Docs. URL: <https://developer.mozilla.org/en-US/docs/Web/HTTP> (дата звернення: 16.06.2024).

25. JWT authentication: Best practices and when to use it. LogRocket. URL: <https://blog.logrocket.com/jwt-authentication-best-practices/> (дата звернення: 16.06.2024).
26. Documentation. DiffusionData. URL: <https://docs.diffusiondata.com/> (дата звернення: 16.06.2024).
27. Sequence Diagram Tutorial, complete guide with examples. Creately. URL: <https://creately.com/guides/sequence-diagram-tutorial/> (дата звернення: 17.06.2024).
28. Fryz M., Mlynko B. Property Analysis of Conditional Linear Random Process as a Mathematical Model of Cyclostationary Signal // 2nd International Workshop on Information Technologies: Theoretical and Applied Problems (ITTAP 2022). Ternopil, Ukraine: CEUR Workshop Proceedings, 2022. Vol. 3309. P. 77–82.
29. Fryz M., Mlynko B. Determination of the characteristic function of discrete-time conditional linear random process and its application // Sci. J. TNTU. 2023. Vol. 109, № 1. P. 16–23.
30. Fryz M., Mlynko B. Property analysis of multivariate conditional linear random processes in the problems of mathematical modelling of signals // Technol. Audit Prod. Reserv. 2022. Vol. 3, № 2(65). P. 29–32.
31. Фриз М.Є., Млинко Б.Б. Умовні лінійні випадкові процеси з дискретним часом та їх властивості // Вісник Хмельницького національного університету. Серія: Технічні науки. 2022 (309), № 3. С. 7–12.
32. Fryz M., Mlynko B. Properties of Stationarity and Cyclostationarity of Conditional Linear Random Processes // 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET). Lviv-Slavske, Ukraine: IEEE, 2020. P. 166–170.
33. Fryz M., Scherbak L., Mlynko B., Mykhailovych T. Linear Random Process Model-Based EEG Classification Using Machine Learning Techniques // Proceedings of the 1st International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2023). Ternopil, Ukraine: CEUR Workshop Proceedings, 2023. Vol. 3468. P. 126–132.

34. Бабак В. П., Марченко Б. Г., Фриз М. Є. Теорія ймовірностей, випадкові процеси та математична статистика. К.: Техніка, 2004. 288 с.
35. Шестеренко О., Шестеренко В. Електропостачання промислових підприємств. Київ : Нова кн., 2004. 424 с.
36. Запорожець О. Основи охорони праці. Київ : Центр учб. літ., 2019. 264 с.

ДОДАТКИ

Реєстрація на головній сторінці в інтернет-магазині «Aquatech»

Ліснинг А.1 – Програмний код для реєстрації на сторінки

```
import React, { useState, useEffect } from 'react';
import { Link, useNavigate, useLocation } from 'react-router-dom';
import { Form, Button, Row, Col, InputGroup } from 'react-bootstrap';
import { useSelector, useDispatch } from 'react-redux';
import { useRegisterMutation } from '../slices/usersApiSlice';
import { setCredentials } from '../slices/authSlice';
import { toast } from 'react-toastify';
import { FaEye, FaEyeSlash } from 'react-icons/fa';
import FormContainer from '../components/FormContainer';
import Loader from '../components/Loader';
import Meta from '../components/Meta';

const RegisterPage = () => {
  const [name, setName] = useState('');
  const [email, setEmail] = useState('');
  const [password, setPassword] = useState('');
  const [confirmPassword, setConfirmPassword] = useState('');
  const [showPassword, setShowPassword] = useState(false);
  const [showConfirmPassword, setConfirmShowPassword] =
  useState(false);

  const dispatch = useDispatch();
  const navigate = useNavigate();

  const [register, { isLoading }] = useRegisterMutation();

  const { userInfo } = useSelector(state => state.auth);

  const { search } = useLocation();
  const searchParams = new URLSearchParams(search);
  const redirect = searchParams.get('redirect') || '/';
```

```

useEffect(() => {
  if (userInfo) {
    navigate(redirect);
  }
}, [userInfo, redirect, navigate]);

const togglePasswordVisibility = () => {
  setShowPassword(!showPassword);
};

const toggleConfirmPasswordVisibility = () => {
  setConfirmShowPassword(!showConfirmPassword);
};

const submitHandler = async e => {
  e.preventDefault();
  if (password !== confirmPassword) {
    toast.error('Passwords do not match!');
    return;
  } else {
    try {
      const res = await register({ name, email, password
    }).unwrap();
      dispatch(setCredentials({ ...res }));
      navigate(redirect);
      toast.success('Registration successful. Welcome!');
    } catch (error) {
      toast.error(error?.data?.message || error.error);
    }
  }
};

return (
  <FormContainer>
    <Meta title={'Register'} />
    <h1>Register</h1>
    <Form onSubmit={submitHandler}>

```



```

<Form.Group className='mb-3' controlId='name'>
  <Form.Label>Name</Form.Label>
  <Form.Control
    value={name}
    type='text'
    placeholder='Enter name'
    onChange={e => setName(e.target.value)}
  />
</Form.Group>
<Form.Group className='mb-3' controlId='email'>
  <Form.Label>Email address</Form.Label>
  <Form.Control
    value={email}
    type='email'
    placeholder='Enter email'
    onChange={e => setEmail(e.target.value)}
  />
</Form.Group>
<Form.Group className='mb-3' controlId='password'>
  <Form.Label>Password</Form.Label>
  <InputGroup>
    <Form.Control
      type={showPassword ? 'text' : 'password'}
      value={password}
      placeholder='Enter password'
      onChange={e => setPassword(e.target.value)}
    />
    <InputGroup.Text
      onClick={togglePasswordVisibility}
      id='togglePasswordVisibility'
      style={{ cursor: 'pointer' }}
    >
      {showPassword ? <FaEye /> : <FaEyeSlash />}
    </InputGroup.Text>
  </InputGroup>
</Form.Group>
<Form.Group className='mb-3' controlId='confirmPassword'>

```

```

<Form.Label>Confirm Password</Form.Label>
<InputGroup>
  <Form.Control
    type={showConfirmPassword ? 'text' : 'password'}
    value={confirmPassword}
    placeholder='Confirm password'
    onChange={e => setConfirmPassword(e.target.value)}
  />
  <InputGroup.Text
    onClick={toggleConfirmPasswordVisibility}
    id='toggleConfirmPasswordVisibility'
    style={{ cursor: 'pointer' }}
  >
    {showConfirmPassword ? <FaEye /> : <FaEyeSlash />}
  </InputGroup.Text>
</InputGroup>
</Form.Group>
<Button
  className='mb-3 w-100'
  variant='primary'
  type='submit'
  disabled={isLoading}
>
  Register
</Button>
</Form>
<Row>
  <Col>
    Already have an account?
    <Link
      to={redirect ? `\/login?redirect=${redirect}` :
        '/login'}
      className=' mx-2'
    >
      Sign In
    </Link>
  </Col>
</Row>
export default RegisterPage;

```

Вміст сторінок продукції інтернет-магазину

Ліснинг Б.1 – Програмний код вмісту сторінок інтернет-магазину

```
import { createSlice } from '@reduxjs/toolkit';

const initialState = {
  userInfo: localStorage.getItem('userInfo')
    ? JSON.parse(localStorage.getItem('userInfo'))
    : null
};

const authSlice = createSlice({
  name: 'auth',
  initialState,
  reducers: {
    setCredentials: (state, action) => {
      state.userInfo = action.payload;
      localStorage.setItem('userInfo',
JSON.stringify(action.payload));
    },
    logout: (state, action) => {
      state.userInfo = null;
      localStorage.removeItem('userInfo');
    }
  }
});

export const { setCredentials, logout } = authSlice.actions;

export default authSlice.reducer;
import { createSlice } from '@reduxjs/toolkit';
import { updateCart } from '../utils/cartUtils';

const initialState = localStorage.getItem('cart')
  ? JSON.parse(localStorage.getItem('cart'))
```

```
    : { cartItems: [], shippingAddress: {}, paymentMethod: 'Razorpay'
  };

const cartSlice = createSlice({
  name: 'cart',
  initialState,
  reducers: {
    addToCart: (state, action) => {
      const item = action.payload;
      const existItem = state.cartItems.find(x => x._id ===
item._id);

      if (existItem) {
        state.cartItems = state.cartItems.map(x =>
          x._id === existItem._id ? item : x
        );
      } else {
        state.cartItems = [...state.cartItems, item];
      }

      return updateCart(state);
    },
    removeFromCart: (state, action) => {
      const id = action.payload;
      state.cartItems = state.cartItems.filter(x => x._id !== id);

      return updateCart(state);
    },
    saveShippingAddress: (state, action) => {
      state.shippingAddress = action.payload;
      return updateCart(state);
    },
    savePaymentMethod: (state, action) => {
      state.paymentMethod = action.payload;
      return updateCart(state);
    },
    clearCartItems: (state, action) => {
```

```

        state.cartItems = [];
        return updateCart(state);
    }
}
});

export const {
  addToCart,
  removeFromCart,
  saveShippingAddress,
  savePaymentMethod,
  clearCartItems
} = cartSlice.actions;

export default cartSlice.reducer;
a
import { PRODUCTS_URL, UPLOAD_URL } from '../constants';
import { apiSlice } from './apiSlice';

export const productApiSlice = apiSlice.injectEndpoints({
  endpoints: builder => ({
    getProducts: builder.query({
      query: ({ limit, skip, search }) => ({
        url: PRODUCTS_URL,
        params: { limit, skip, search }
      }),
      providesTags: ['Product']
    }),
    getTopProducts: builder.query({
      query: () => ({
        url: `${PRODUCTS_URL}/top`
      }),
      providesTags: ['Product']
    }),
    getProductDetails: builder.query({
      query: productId => ({
        url: `${PRODUCTS_URL}/${productId}`

```

```

    }),
    providesTags: ['Product']
  )),
  createProduct: builder.mutation({
    query: productData => ({
      url: PRODUCTS_URL,
      method: 'POST',
      body: productData
    }),
    invalidatesTags: ['Product']
  )),
  updateProduct: builder.mutation({
    query: ({ productId, ...productData }) => ({
      url: `${PRODUCTS_URL}/${productId}`,
      method: 'PUT',
      body: { ...productData }
    }),
    invalidatesTags: ['Product']
  )),
  deleteProduct: builder.mutation({
    query: productId => ({
      url: `${PRODUCTS_URL}/${productId}`,
      method: 'DELETE'
    }),
    invalidatesTags: ['Product']
  )),
  uploadProductImage: builder.mutation({
    query: data => ({
      url: UPLOAD_URL,
      method: 'POST',
      body: data
    }),
    invalidatesTags: ['Product']
  )),
  createProductReview: builder.mutation({
    query: ({ productId, ...reviewData }) => ({
      url: `${PRODUCTS_URL}/reviews/${productId}`,

```

```

        method: 'POST',
        body: { ...reviewData }
    )),
    invalidatesTags: ['Product']
  })
})
});

export const {
  useGetProductsQuery,
  useGetProductDetailsQuery,
  useCreateProductMutation,
  useDeleteProductMutation,
  useUploadProductImageMutation,
  useUpdateProductMutation,
  useCreateProductReviewMutation,
  useGetTopProductsQuery
} = productApiSlice;
import { createSlice } from '@reduxjs/toolkit';

const initialState = {
  search: ''
};

export const searchProductSlice = createSlice({
  name: 'products',
  initialState,
  reducers: {
    searchProduct: (state, action) => {
      state.search = action.payload;
    },
    clearSearch: state => {
      state.search = '';
    }
  }
});

```

```
// Action creators are generated for each case reducer function
export const { searchProduct, clearSearch } =
searchProductSlice.actions;

export default searchProductSlice.reducer;
a
import { USERS_URL } from '../constants';
import { apiSlice } from './apiSlice';

export const usersApiSlice = apiSlice.injectEndpoints({
  endpoints: builder => ({
    login: builder.mutation({
      query: data => ({
        url: `${USERS_URL}/login`,
        method: 'POST',
        body: data
      }),
      invalidatesTags: ['User']
    }),
    logout: builder.mutation({
      query: () => ({
        url: `${USERS_URL}/logout`,
        method: 'POST'
      }),
      invalidatesTags: ['User']
    }),
    register: builder.mutation({
      query: data => ({
        url: `${USERS_URL}`,
        method: 'POST',
        body: data
      }),
      invalidatesTags: ['User']
    }),
    newPasswordRequest: builder.mutation({
      query: data => ({
        url: `${USERS_URL}/reset-password/request`,
```



```

        method: 'POST',
        body: data
    )),
    invalidatesTags: ['User']
  )),
  resetPassword: builder.mutation({
    query: ({ userId, token, password }) => ({
      url: `_${USERS_URL}/reset-
password/reset/${userId}/${token}`,
      method: 'POST',
      body: { password }
    )),
    invalidatesTags: ['User']
  )),
  profile: builder.mutation({
    query: data => ({
      url: `_${USERS_URL}/profile`,
      method: 'PUT',
      body: data
    )),
    invalidatesTags: ['User']
  )),
  getUserProfile: builder.query({
    query: async () => ({
      url: `_${USERS_URL}/profile`
    )),
    providesTags: ['User']
  )),
  getUsers: builder.query({
    query: () => ({
      url: USERS_URL
    )),
    providesTags: ['User']
  )),
  admins: builder.query({
    query: () => ({
      url: `_${USERS_URL}/admins`

```

```

    }),
    providesTags: ['User']
  )),
  getUserById: builder.query({
    query: userId => ({
      url: `${USERS_URL}/${userId}`
    }),
    providesTags: ['User']
  )),
  deleteUser: builder.mutation({
    query: userId => ({
      url: `${USERS_URL}/${userId}`,
      method: 'DELETE'
    }),
    invalidatesTags: ['User']
  )),
  updateUser: builder.mutation({
    query: ({ userId, ...userData }) => ({
      url: `${USERS_URL}/${userId}`,
      method: 'PUT',
      body: { ...userData }
    }),
    invalidatesTags: ['User']
  )),
  )));

```

```

export const {
  useLoginMutation,
  useLogoutMutation,
  useRegisterMutation,
  useNewPasswordRequestMutation,
  useResetPasswordMutation,
  useProfileMutation,
  getUserProfileQuery,
  getUsersQuery,
  deleteUserMutation,

```

```
    useUpdateUserMutation,  
    useGetUserByIdQuery,  
    useAdminsQuery  
  } = usersApiSlice;  
import React from 'react';  
import ReactDOM from 'react-dom/client';  
import 'bootstrap/dist/css/bootstrap.min.css';  
// import './assets/styles/bootstrap.custom.css';  
import './assets/styles/index.css';  
import reportWebVitals from './reportWebVitals';  
import Routes from './routes/Routes';  
import store from './store';  
import { Provider } from 'react-redux';  
import { HelmetProvider } from 'react-helmet-async';  
const root = ReactDOM.createRoot(document.getElementById('root'));  
  
root.render(  
  <React.StrictMode>  
    <HelmetProvider>  
      <Provider store={store}>  
        <Routes />  
      </Provider>  
    </HelmetProvider>  
  </React.StrictMode>  
)
```