

Вінницький національний технічний університет

(повне найменування вищого навчального закладу)

Факультет інтелектуальних інформаційних технологій та автоматизації

(повне найменування інституту, назва факультету (відділення))

Кафедра комп'ютерних наук

(повна назва кафедри (предметної, циклової комісії))

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Інформаційна технологія доставки води по місту»

Виконав: студент 2-го курсу, групи ЗКН-22м
спеціальності 122 «Комп'ютерні науки»

(шифр і назва напрямку підготовки, спеціальності)

Пугач Б.В.

(прізвище та ініціали)

Керівник: к.т.н., доц. каф. КН

Озеранський В.С.

(прізвище та ініціали)

«02»

12

2023 р.

Опонент: к.т.н., доц. каф. АІТ

Барабан М.В.

(прізвище та ініціали)

«04»

12

2023 р.

Допущено до захисту

Завідувач кафедри КН

д.т.н., проф. Яровий А.А.

(прізвище та ініціали)

« 08.12

» 2023 р.

Вінниця ВНТУ - 2023 рік

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних наук
Рівень вищої освіти II-й (магістерський)
Галузь знань – 12 – Інформаційні технології
Спеціальність – 122 – Комп'ютерні науки
Освітньо-професійна програма – Системи штучного інтелекту

ЗАТВЕРДЖУЮ

Завідувач кафедри КН

д.т.н., проф. Яровий А.А.

(підпис)

“ 29 ” 08 2023 року

ЗАВДАННЯ НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

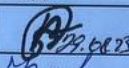
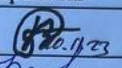
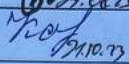
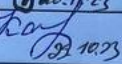
Пугачу Богдану Васильовичу

- 1 Тема роботи: «Інформаційна технологія доставки води по місту».
Керівник роботи: Озеранський Володимир Сергійович, к.т.н., доц. каф. КН
затверджені наказом вищого навчального закладу «18» 09 2023 року № 247
- 2 Строк подання студентом роботи 13.11.2023
- 3 Вихідні дані до роботи: кількість видів товару – не менше 5шт;
потужність бази адресатів – не менше 100чол; мова програмування - об'єктно-орієнтована.
- 4 Зміст пояснювальної записки (перелік питань, які потрібно розробити):
вступ, аналіз сучасного стану інформаційної технології доставки води по місту;
розробка інформаційної технології доставки води по місту; програмна реалізація
інформаційної технології доставки води по місту; економічна частина, висновки,
перелік використаних джерел, додатки.
- 5 Перелік графічного матеріалу (з точним зазначенням обов'язкових
креслень): схема алгоритму доставки води по місту; структура інформаційної
технології доставки води по місту; приклад роботи програмного засобу.

6 Консультанти розділів проекту (роботи)

Консультанти розділів роботи в таблиці 1.


Таблиця 1 - Консультанти роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-3	Озеранський В.С., к.т.н., доцент каф. КН	 09.09.23	 09.10.23
4	Кавецький В.В., к.е.н., доц. каф. ЕПВМ	 09.10.23	 09.10.23


7 Дата видачі завдання 29.08.23р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз сучасного стану інформаційної технології доставки води	01.09.23-04.09.23	
2	Розробка інформаційної технології доставки води	04.09.23-24.09.23	
3	Програмна реалізація інформаційної технології доставки води	25.09.23-15.10.23	
4	Підготовка економічної частини	21.10.23-29.10.23	
5	Апробація та/або впровадження результатів дослідження	30.10.23-06.11.23	
6	Оформлення пояснювальної записки, графічного матеріалу та презентації	06.11.23.-10.11.23	

Студент 
(підпис)

Пугач Б.В.
(прізвище та ініціали)

Керівник роботи 
(підпис)

Озеранський В.С.
(прізвище та ініціали)

АНОТАЦІЯ

УДК 621.374.415

Пугач Б.В. Інформаційна технологія доставки води по місту. Магістерська кваліфікаційна робота зі спеціальності 122 – комп'ютерні науки, освітня програма - Системи штучного інтелекту. Вінниця: ВНТУ, 2023. 104с.

На укр. мові. Бібліогр.: 20 назв; рис.: 21; табл. 15.

Дана магістерська кваліфікаційна робота присвячена розробці інформаційної технології доставки води по місту.

В роботі проаналізовано предметну область, зазначено актуальність дослідження, визначено основну проблему доставки води по місту. Запропоновано розробити нову інформаційну технологію, яка дозволить уникнути зазначених недоліків. Здійснено аналіз вимог до нової інформаційної технології доставки води по місту.

Обґрунтовано вибір технологій, проведено проектування архітектури, розроблено та реалізовано інформаційну технологію доставки води по місту у вигляді веб-сервісу. За основу серверної частини програми було обрано програмну платформу Node.js з використанням фреймворку Express. В якості системи управління базами даних було вирішено зстосовувати PostgreSQL. Для реалізації роботи телеграм бота було обрано бібліотеку Telegraf. В якості інструменту для розробки клієнтської частини додатку вибрано бібліотеку React.

В розділі економічної частини проведено оцінювання комерційного потенціалу розробки інформаційної технології оптичного розпізнавання тексту, спрогнозовано витрати на виконання наукової роботи та впровадження результатів, які склали 148543,59 грн, період окупності складає 1,93 року.

Ключові слова: інформаційна технологія, React, Node.js, доставка води, PostgreSQL.

ABSTRACT

Pugach B.V. Information technology of water delivery in the city. Master's thesis on specialty 122 - computer science, educational program - Artificial intelligence systems. Vinnytsia: VNTU, 2023. 104 p.

In Ukrainian speech Bibliography: 20 titles; Fig.: 21; table 15.

This master's thesis is devoted to the development of information technology for water delivery in the city.

The paper analyzes the subject area, indicates the relevance of the research, and defines the main problem of water delivery in the city. It is proposed to develop a new information technology that will avoid the mentioned shortcomings. An analysis of the requirements for the new information technology of water delivery in the city was carried out.

The choice of technologies was substantiated, architectural design was carried out, information technology for water delivery in the city in the form of a web service was developed and implemented. The Node.js software platform using the Express framework was chosen as the basis for the server part of the program. It was decided to use PostgreSQL as a database management system. The Telegraf library was chosen to implement the Telegram bot. The React library was chosen as a tool for developing the client part of the application.

In the section of the economic part, an assessment of the commercial potential of the development of the information technology of optical text recognition was carried out, the costs for the implementation of the scientific work and the implementation of the results were predicted, which amounted to UAH 148 543.59, the payback period is 1.93 years.

Keywords: information technology, React, Node.js, water delivery, PostgreSQL.

ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ СУЧАСНОГО СТАНУ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДОСТАВКИ ВОДИ ПО МІСТУ	7
1.1 Роль ІТ-технологій у веденні бізнесу.....	7
1.2 Аналіз існуючих програмних рішень доставки води	9
1.3 Аналіз вимог до інформаційної технології доставки води по місту.....	18
1.4 Висновок до розділу 1.....	20
2 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДОСТАВКИ ВОДИ ПО МІСТУ .	21
2.1 Розробка структури бази даних інформаційної технології доставки води по місту	21
2.2 Розробка структури інформаційної технології доставки води по місту.....	23
2.3 Аналіз вимог до інформаційної технології доставки води по місту.....	25
2.4 Висновок до розділу 2.....	28
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДОСТАВКИ ВОДИ ПО МІСТУ.....	29
3.1 Обґрунтування вибору засобів розробки.....	29
3.2 Обґрунтування вибору мови програмування	31
3.3 Обґрунтування вибору системи управління базами даних.....	33
3.4 Обґрунтування використання платформи Docker	34
3.5 Розробка серверної частини програмного забезпечення для доставки води по місту	38
3.6 Розробка клієнтської частини програмного забезпечення для доставки води по місту	45
3.7 Тестування програмного забезпечення для доставки води по місту	53
3.8 Висновок до розділу 3.....	59
4 ЕКОНОМІЧНА ЧАСТИНА.....	60
4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки	60
4.2 Розрахунок узагальненого коефіцієнта якості розробки	64

4.3 Розрахунок витрат на проведення науково-дослідної роботи.....	66
4.4 Розрахунок економічної ефективності науково-технічної розробки від її впровадження безпосередньо розробником (замовником).....	77
4.5 Висновок до розділу 4.....	83
ВИСНОВКИ.....	84
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	85
ДОДАТКИ.....	87
Додаток А (обов'язковий) Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень.....	88
Додаток Б (обов'язковий) Лістинг програми.....	89
Додаток В (обов'язковий) Ілюстративна частина.....	94
Додаток Г (довідниковий) Інструкція користувача.....	99

ВСТУП

Актуальність теми дослідження. На ринку послуг доставки в останні роки спостерігається інтенсифікація конкуренції, пов'язана як з успішною діяльністю великих міжнародних операторів, так і появою невеликих рішень для приватних підприємців. Одним із шляхів утримання бізнесу на плаву в сучасних реаліях є цифрова трансформація. Цифровізація процесів актуальна не лише на рівні окремих підприємств: цілі галузі обирають для себе цей шлях розвитку як єдину можливість відповідати умовам навколишнього світу, що стрімко змінюються. Завдяки цьому цифрова трансформація промисловості, роздрібної торгівлі, державного сектору та інших сфер уже сьогодні змінює життя кожної людини та кожної компанії.

Кожен намагається виділитися та показати себе різними способами у всіх інформаційних джерелах, а також у мережі Інтернет. Інтернет є невід'ємною частиною життя сучасної людини. Для того, щоб показати власну компанію потенційному споживачеві, необхідно розповісти аудиторії про діяльність організації. Розробка та впровадження інформаційної технології є досить ефективним способом просування компанії.

Використання інформаційних технологій є одним із прикладів сучасних рішень на шляху цифровізації бізнесу. Впровадження подібних речей може допомогти в залученні більшої кількості клієнтів, спростити ведення обліку та збільшити продуктивність обробки замовлень.

Доцільність даної роботи сформована та обумовлена бурхливим розвитком інформаційних технологій та широким впровадженням їх в усі сфери суспільства, в тому числі маленький та великий бізнес.

Основою серверної частини програми стане програмна платформа Node.js з використанням фреймворку Express. Системою управління базами даних виступить PostgreSQL. Для реалізації роботи телеграм бота використовуватиметься бібліотека Telegraf. В якості бібліотеки для написання клієнтської частини проекту буде використаний React.

Саме тому дуже доцільною є розробка інформаційної технології, яка вирішує задачу доставки води по місту.

Зв'язок роботи з науковими програмами, планами, темами. Магістерська робота виконана відповідно до напрямку наукових досліджень кафедри комп'ютерних наук Вінницького національного технічного університету 22 К1 «Моделі, методи, технології та пристрої інтелектуальних інформаційних систем управління, економіки, навчання та комунікацій» та плану наукової та навчально-методичної роботи кафедри.

Мета та завдання дослідження. Метою дослідження магістерської кваліфікаційної роботи є розширення функціональних можливостей програмного забезпечення для замовлення та доставки води по місту.

Для досягнення поставленої мети слід розв'язати такі завдання:

- проаналізувати предметну область доставки води по місту;
- проаналізувати існуючі програмні рішення для доставки води по місту;
- розробити структуру інформаційної технології доставки води по місту;
- розробити алгоритм функціонування інформаційної технології доставки води по місту;
- виконати програмну реалізацію запропонованої інформаційної технології доставки води по місту;
- провести тестування програмного продукту та виконати аналіз отриманих результатів;
- провести економічні розрахунки доцільності розробки інформаційної технології доставки води по місту.

Об'єкт дослідження – процес замовлення та доставки води по місту.

Предмет дослідження – програмні засоби для замовлення доставки води по місту.

Методи дослідження. У роботі використані такі методи наукових досліджень: теорія штучних нейронних мереж для реалізації інформаційної технології доставки води по місту, методи математичної статистики для обрахунків результатів

отриманих за допомогою програмного засобу, програмування на мовах високого рівня.

Наукова новизна одержаних результатів полягає в наступному:

Удосконалено інформаційну технологію доставки води по місту, що відрізняється від відомих використанням удосконаленої інформаційної моделі оптимізації замовлень, що забезпечило розширення функціональних можливостей програмного забезпечення для замовлення та доставки води по місту.

Практичне значення одержаних результатів:

1. Розроблено алгоритм доставки води по місту.
2. Здійснено програмну реалізацію інформаційної технології доставки води по місту.

Достовірність теоретичних положень магістерської кваліфікаційної роботи підтверджується строгістю постановки задач, коректним застосуванням математичних методів під час доведення наукових положень, строгим виведенням аналітичних співвідношень, порівнянням результатів з відомими, та збіжністю результатів моделювання з результатами, що отримані під час впровадження розроблених програмних засобів.

Особистий внесок здобувача. Усі результати, що наведені у магістерській кваліфікаційній роботі, отримані самостійно. У працях, які написано у співавторстві, здобувачу належать: особливості замовлення та доставки води по місту [1].

Апробація результатів роботи. Результати роботи були апробовані на конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2023)» (м. Вінниця, Україна, 2023 р.) [1].

Публікації. За результатами магістерської кваліфікаційної роботи опубліковано 1 тезу доповіді на конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2023)» [1].

1 АНАЛІЗ СУЧАСНОГО СТАНУ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДОСТАВКИ ВОДИ ПО МІСТУ

1.1 Роль ІТ-технологій у веденні бізнесу

Кожна людина має право на те, щоб пити якісну воду. Втім, у більшості випадків вода з крану, що потрапляє туди через міську систему водопостачання, залишає бажати кращого. В більшості випадків міська система постачання води застаріває і її модернізація вважається чимось дорогим та складним. Більшість методик очистки води, актуальних десятки років тому, наразі просто неактуальні, оскільки не здатні зробити рідину чистою у повному обсязі. Зазвичай така вода має неприємні смак та запах, а її вживання у довгостроковій перспективі може стати причиною серйозних проблем зі здоров'ям.

Внаслідок цього, сьогодні все більше людей шукають розумну альтернативу, яка б зробила доступ до питної води максимально простим. У якості найбільш поширених варіантів виступають домашній фільтр для води та бутильована покупна вода. Кожен з варіантів має як свої недоліки, так і переваги.

Все більше українців вважають за краще замовляти воду додому в бутлях завдяки вигідному співвідношенню якості продукту, зручності та вартості послуги. Для доставки використовуються полікарбонатні бутлі об'ємом 19л, це зручно і не доведеться турбуватися про поповнення запасів довгий час. Якщо раніше над питанням, яку доставку води обрати, розмірковували здебільшого офіс-менеджери та керівники підприємств, сьогодні дедалі частіше послугою користуються звичайні українці, які піклуються про власне здоров'я. Саме тому популярність спеціалізованих сервісів доставки води безперервно зростає. І це в свою чергу, веде до зростання попиту на використання інформаційних технологій в галузі доставки води.

Інформаційні технології в сучасному бізнесі представляють собою не тільки незамінний інструмент, але й ключовий елемент стратегії маркетингу для широкого кола компаній. Завдяки сталому розвитку технологій, вони стають дедалі більшим рушійним фактором успіху підприємств.

Інформаційні технології вирішують ряд складних завдань, допомагаючи оптимізувати бізнес-процеси та підвищувати ефективність підприємства. Автоматизація рутинних завдань дозволяє звільнити робочий час для стратегічного мислення та творчості співробітників.

Однією з основних переваг використання інформаційних технологій є можливість отримання та обробки великих обсягів даних. Аналітика та звітність, забезпечені цими технологіями, дозволяють бізнесу вчасно реагувати на зміни в ринкових умовах та приймати обґрунтовані стратегічні рішення.

Застосування інформаційних технологій в маркетингу дозволяє підприємствам побудувати ефективні кампанії з реклами та просування продуктів, зокрема в соціальних мережах та електронній комерції. Вони розширюють можливості спілкування з клієнтами, забезпечуючи персоналізований підхід та підвищуючи рівень задоволеності клієнтів.

Однак, разом із зростанням можливостей, постають і виклики, пов'язані зі збереженням безпеки інформації, захистом від кібератак та забезпеченням конфіденційності даних. Тому важливо враховувати не лише переваги, але й ризики використання інформаційних технологій та розвивати ефективні стратегії їх впровадження.

Усе більше компаній визнають, що впровадження сучасних інформаційних технологій є необхідним етапом для досягнення конкурентних переваг і підтримки сталого розвитку.

Програмною реалізацією інформаційних технологій може бути і веб-додаток – це клієнт-серверне програмне забезпечення, яке знаходиться на віддаленому сервері. Але в той же час інтерфейс користувача сервісу відображається в браузері. Це означає, що користувач не потребує встановлення додаткових програм, адже програма запускається через браузер.

Основні переваги використання веб-ресурсів для бізнесу:

1. Скорочення витрат. За рахунок автоматизації процесів та моментального обміну даними відбувається економія робочого часу працівників. Знижуються

матеріальні витрати (ті самі витрати на канцтовари та електроенергію). У результаті в компанії з'являються вільні ресурси.

2. Автоматизація. Вона, крім зниження витрат, тягне за собою підвищення ефективності. Пришвидшується робочий цикл.

3. Функціональність. Як правило, веб-сервіс створюють для вирішення кількох завдань. Тому ефективність організації збільшується комплексно. Закладені в продукт функції передбачають успіх за кількома напрямками.

Безумовно, не всім можуть стати у нагоді веб-сервіси. Однак є сфери, де програмні продукти справді допомагають вивести бізнес на новий рівень.

1.2 Аналіз існуючих програмних рішень доставки води

Інформаційні технології в галузі надання послуг з доставки води відіграють важливу роль у вдосконаленні процесів, оптимізації логістики та підвищенні якості обслуговування клієнтів. Новаторські рішення в цій галузі дозволяють компаніям прискорити та ефективно керувати всіма етапами процесу доставки води, починаючи від прийому замовлення і закінчуючи отриманням повідомлення від клієнта після успішної доставки.

Клієнти можуть замовляти воду за допомогою онлайн-платформ або мобільних додатків. Це забезпечує зручність і доступність для користувачів, які можуть швидко та легко вибрати тип води, обсяг і час доставки.

ІТ-рішення дозволяють як компаніям, так і клієнтам відстежувати стан замовлення в режимі реального часу. Це надає можливість точно визначити час прибуття води і попереджати клієнтів про будь-які зміни.

Спеціалізовані програмні системи допомагають оптимізувати маршрути доставки, що призводить до економії часу та ресурсів. Це особливо важливо для зменшення витрат пального та зниження викидів в атмосферу. ІТ-інструменти для ефективного управління запасами та складами дозволяють компаніям точно

визначати рівень запасів, уникати надмірної або недостатньої кількості продукції на складі.

Оплата за послуги доставки води через онлайн-платформи та мобільні додатки стала важливим елементом сучасного бізнесу, привносячи зручність та швидкість в процес взаємодії між клієнтами та підприємствами.

Можливість здійснювати оплату онлайн дозволяє клієнтам проводити транзакції в будь-який зручний для них час, без необхідності відвідування фізичних точок обслуговування чи використання готівки. Це особливо актуально в умовах сучасного темпу життя, де електронні платежі стають стандартом і дозволяють економити час і зусилля клієнтів.

Мобільні додатки, як інструмент для здійснення платежів, додатково покращують користувацький досвід. Вони надають можливість зручно вибирати опції, контролювати історію транзакцій та отримувати повідомлення про стан замовлення чи доставки. Додатки також можуть зберігати інформацію про улюблені замовлення або надавати знижки та бонуси, сприяючи лояльності клієнтів.

Оплата через інтернет дозволяє підприємствам автоматизувати фінансові процеси, зменшуючи ризики пов'язані з обробкою готівкових коштів та забезпечуючи точність в обліку фінансових операцій. Це сприяє підвищенню ефективності бухгалтерського обліку та дозволяє підприємствам фокусуватися на наданні якісних послуг замість рутинних фінансових операцій.

Загалом, електронні системи оплати в галузі доставки води роблять процес взаємодії між споживачами та підприємствами більш зручним, швидким та ефективним. Ці технології відіграють суттєву роль у підтримці сучасних стандартів обслуговування та відображають стрімкий розвиток та адаптацію бізнесу до вимог сучасного ринку.

В сучасному бізнес-середовищі використання інформаційних технологій стало ключовим фактором для покращення взаємодії компаній з клієнтами. Це не просто інструмент для автоматизації, але й потужний механізм для створення ретельної та персоналізованої взаємодії, яка веде до підвищення рівня задоволеності клієнтів і зміцнення їх лояльності.

Один із ключових аспектів використання інформаційних технологій у взаємодії з клієнтами - це можливість надавати ретельний зворотний зв'язок. Компанії можуть використовувати різноманітні канали комунікації, такі як електронна пошта, чати, соціальні мережі та онлайн-форми зворотного зв'язку, щоб спілкуватися з клієнтами. Це не лише дозволяє ефективно отримувати відгуки, але і створює можливість оперативно реагувати на їхні потреби та побажання.

Інформаційні технології також допомагають компаніям вирішувати можливі проблеми швидше та ефективніше. Системи керування заявками та технічною підтримкою дозволяють автоматизувати процес обробки звернень клієнтів, відстежувати їхні стани та швидко надавати відповіді чи рішення. Це важливо для створення позитивного враження у клієнтів та підтримки їхнього довіри до бренду.

Підприємства також можуть використовувати аналітику для вивчення поведінки клієнтів, їхніх вподобань та звичок. Це дозволяє персоналізувати пропозиції, надавати індивідуальний сервіс та адаптувати стратегії взаємодії. Все це сприяє покращенню якості обслуговування та підвищенню задоволеності клієнтів.

Загалом, інформаційні технології стають катализатором для створення глибокого та взаємовигідного партнерства між компаніями та їхніми клієнтами. Вони не лише спрощують комунікацію, але й допомагають зрозуміти потреби аудиторії та пристосовуватися до швидко змінюючихся вимог ринку.

Деякі компанії використовують інноваційні технології для зменшення впливу на навколишнє середовище, такі як використання електричних або гібридних автотранспортних засобів для доставки води.

Використання інформаційних технологій у сфері доставки води дозволяє компаніям забезпечити ефективність, точність і високий рівень обслуговування. Це допомагає вирішити завдання швидкої та надійної доставки води, що є важливим аспектом для задоволення потреб споживачів.

При аналізі уже існуючих рішень основними критеріями оцінки виступають:

- Інформативність;
- навігація (структурованість інформації, зручність користування);
- технічні можливості.

Інформаційна технологія повинна мати комерційну візуальну концепцію – фокус якої направлений на зручному та лаконічному відображенні всієї можливої інформації для здійснення замовлення, покупки та оплати доставки води клієнтом.

Як правило, під час розробки інформаційної технології у вигляді веб-ресурсу максимальна увага приділяється дизайну його головної сторінки. А вже на основі його розробляється дизайн типових внутрішніх сторінок. Метою головної сторінки веб-ресурсу є утримання уваги цільових відвідувачів і мотивація продовження знайомства з сайтом.

Проаналізуємо існуючі рішення веб-застосунків доставки води по місту.

Для аналізу було обрано кілька веб-ресурсів доставки води [3].

Веб-ресурс «mywatershop.com.ua» - офіційний сервіс доставки води групи компаній IDS Ukraine, яка є оператором на ринку видобутку та доставки води. Більше 26 років компанія пропонує покупцям тільки найкращу воду, розливу за європейськими стандартами якості. На сьогоднішній день компанія представляє 6 власних торгових марок, в тому числі такі відомі природні мінеральні води, як Моршинська, Миргородська та Aqua Life. Сервіс My Water Shop зосереджений на доставці води у бутлях 18,9 л, ПЕТ пляшках і склі, супутніх товарів і обладнання. Високий рівень сервісу забезпечує більше 200 000 задоволених клієнтів по всій Україні (рис. 1.1).

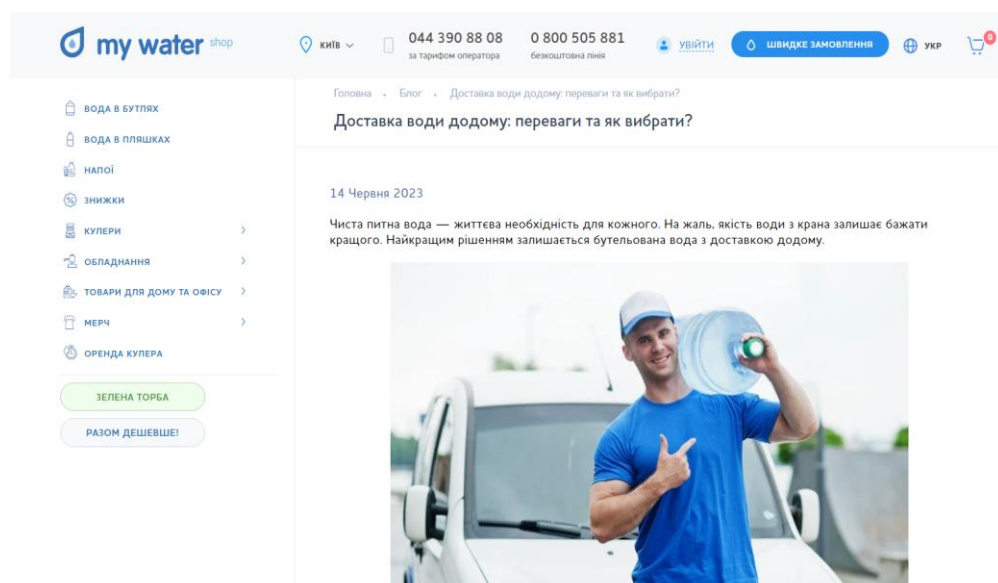


Рисунок 1.1 – Веб-ресурс доставки води mywatershop.com.ua

Компанія «ВОДО ОФІС СЕРВІС» здійснює доставку природної бутильованої води ТМ “Аквітанія” додому, в офіс, на виробництво, в заклади громадського харчування, медичні установи та школи у місті Львові.

Бутильована вода з доставкою – це найкраще рішення для вирішення проблеми забезпечення чистою та корисною водою дому та офісу. Можна замовити природну ідеально чисту карпатську воду у полікарбонатних бутлях ємністю 19 літрів для щоденного вживання та приготування їжі та напоїв. Полікарбонатний бутиль – це спеціальна тара зберігання та доставки питної води [4].

Разом з водою в бутлях можна замовити доставку супутніх товарів, а саме: каву мелену та в зернах “Poli“, “ЖОКЕЙ“ та “Jacobs“, чай “Althaus“, “Greenfield“, “TESS“, воду в пляшках “Рогатинська”, пластикові та паперові стакани, цукор, серветки та ін. (рис. 1.2)

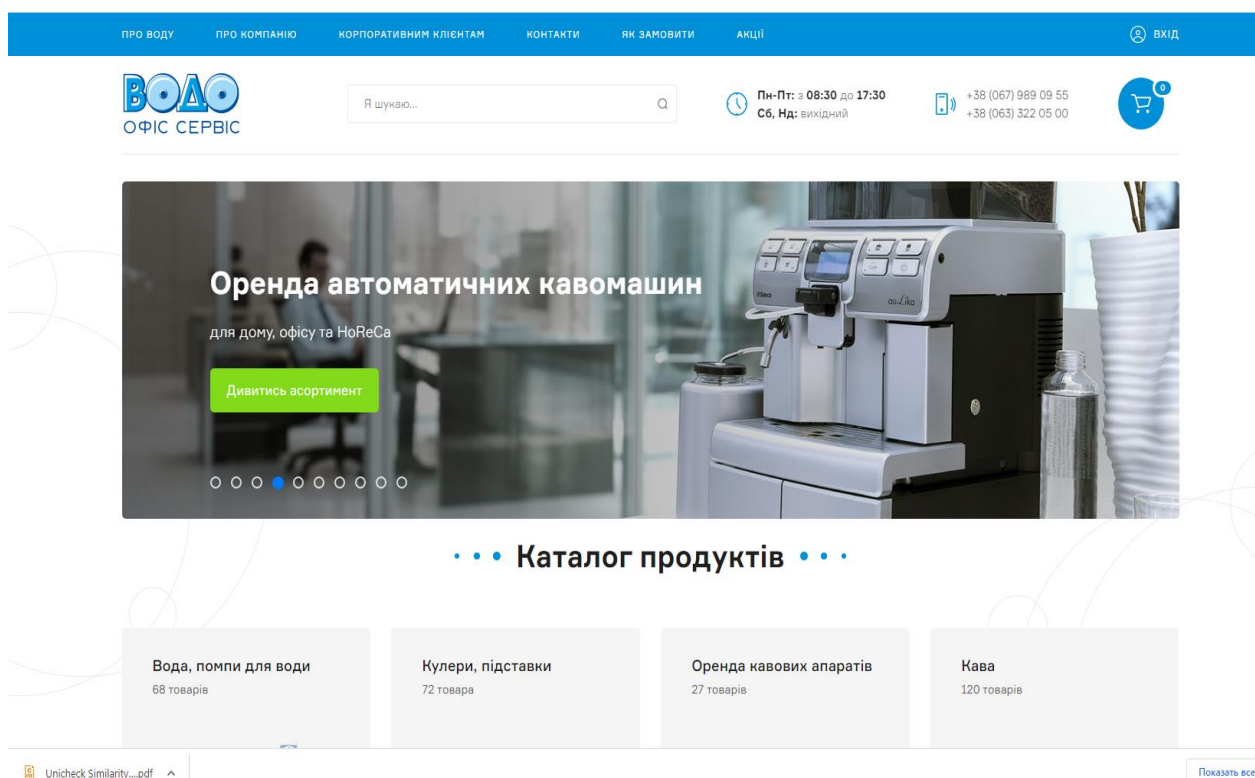


Рисунок 1.2 – Веб-ресурс доставки води ВОДО ОФІС СЕРВІС

Компанія також надає в оренду обладнання для приготування кави – кавові автомати SAECO та обладнання для зручного користування бутильованою водою – пластикові та керамічні диспенсери, помпи, бутилі, підставки, стаканотримачі та кулери для нагрівання та охолодження води.

Разом з тим, компанія ВОДО ОФІС СЕРВІС здійснює ремонт, технічне обслуговування, санітарну очистку кулерів та кавових автоматів.

Переваги сервісу полягають у наявності:

- чаю в пакетиках та листового Greenfield, Totti, Lovare, Карпатського чаю, Tess, Alokozay;

- кави меленої та в зернах Lavazza, Jacobs, Gimoca, Totti, Galca, Movenpick, Dallmayr, Ambassador;

- води в бутлях (19л) ТМ АКВІТАНІЯ та ТМ ДЖЕРЕЛЬНА;

- кулерів та диспенсерів для води, різноманітних аксесуарів (підставки для зручного розливу та зберігання бутлів в офісі, стаканороздавачі);

- води в пляшках – Моршинська, Vorjomi, Поляна Квасова, Pepsi, Coca-Cola, соки Sandora, Rich, молоко та ін.

- автоматичних кавомашин провідного світового виробника SAECO;

- виробів з паперу та одноразового посуду;

- продуктів харчування, печива, цукерок;

- побутової хімії для дому та офісу, обладнання та витратних матеріалів для прибирання;

- техніки для офісу.

Веб-ресурс доставки води по місту Вінниця «vinnitska.com».

Вода добувається зі свердловини глибиною 180 м., що відповідає Юрському періоду, в самому серці України, неподалік від відомого заповідника Конча-Заспа - місто Українка, Київська обл. Питна вода, під торговою маркою "Вінницька" відрізняється своєю чистотою і збалансованим складом, що гарантує її природну чистоту без шкідливих домішок. Процес фільтрації води відбувається на сучасному устаткуванні. У нашої води оптимальний сольовий склад і рівень кислотності рН 7,0 за рахунок чого вона не лише корисна, але і має м'який приємний смак.

Абсолютно чиста і без домішок, консервантів та інших шкідливих речовин, що підтверджується сертифікатами. Підходить для вживання в сирому вигляді, для приготування чаю, кави та їжі, а також для створення цілющих настоянок, фітобальзамів і т.д.

Головна сторінка веб-ресурсу має доволі простий та не перевантажений зайвою інформацією дизайн (рис. 1.3).

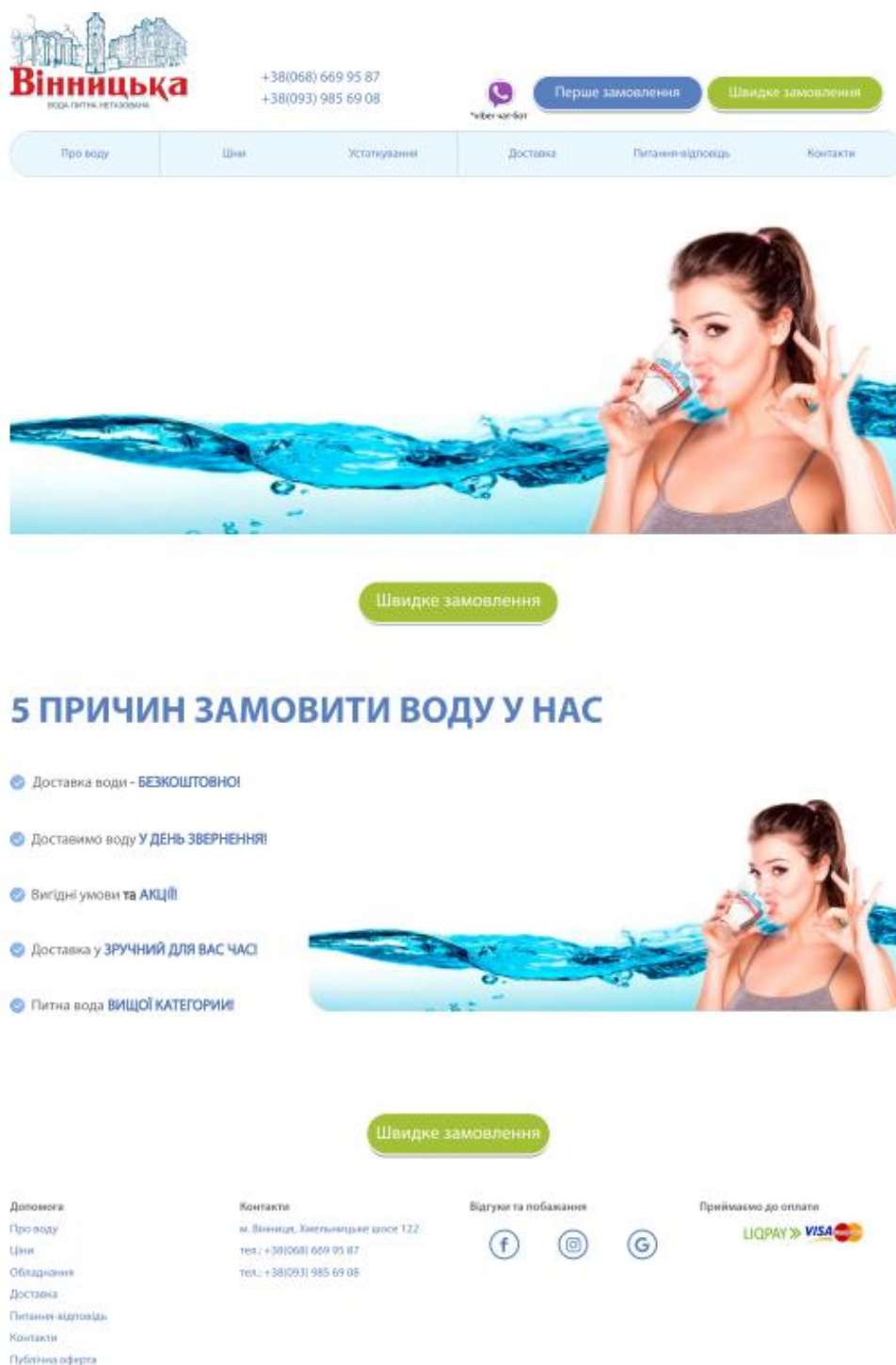


Рисунок 1.3 – Головна сторінка сайту доставки води vinnitska.com

При переході на сайт «vinnitska.com» користувач може ознайомитись з всією необхідною інформацією (рис. 1.4), натиснувши на відповідні пункти в рядку під

логотипом сайту (про воду, ціни, устаткування, доставка, питання-відповіді, контакти).



Рисунок 1.4 – Веб-ресурс доставки води vinnitska.com

Також гості сайту одразу можуть натиснути одну із двох кнопок для здійснення замовлення (перше замовлення, швидке замовлення). Форми замовлень що відкриваються при натисканні кнопок зображенні на рисунках 1.5 та 1.6.

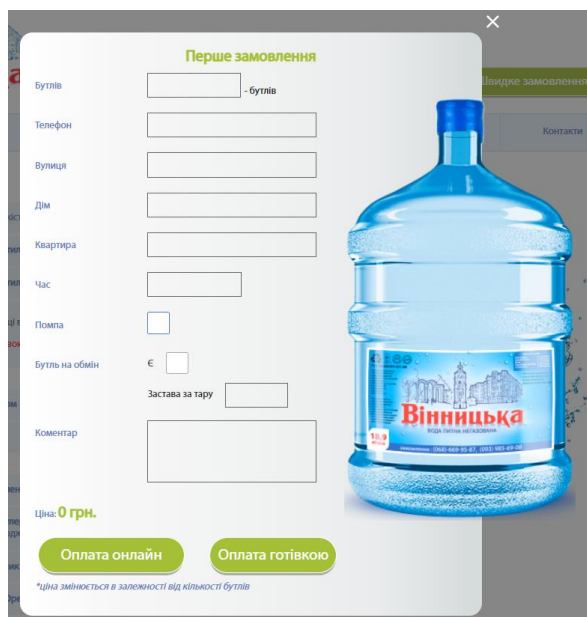
Також біля кнопок для здійснення замовлень розташоване посилання на чат бот для клієнтів для здійснення замовлення через месенджер Viber.

Нижня частина сторінки, що є традиційним елементом веб-сайтів, зазвичай призначена для розміщення важливої контактної інформації, яка дозволяє відвідувачам зв'язуватися з фірмою. Тут знаходяться адреса компанії, телефонні номери або електронні адреси, що спрощує шляхи комунікації між клієнтами та представниками бізнесу.

Зокрема, тут подаються точна адреса офісу та магазину, що створює додаткову впевненість у відвідувачів і надає їм можливість особистого візиту. Телефонні номери

дозволяють клієнтам швидко звертатися за допомогою телефонного дзвінка, щоб отримати докладнішу інформацію або вирішити питання.

Окрім того, на нижній частині сторінки часто розміщуються посилання на профілі фірми в різних соціальних мережах. Це робить інформацію більш доступною для користувачів, які використовують соціальні платформи для знаходження актуальних новин, пропозицій чи взаємодії з компанією.



Перше замовлення

Бутлів - бутлів

Телефон

Вулиця

Дім

Квартира

Час

Помпа

Бутль на обмін €

Застава за тару

Коментар

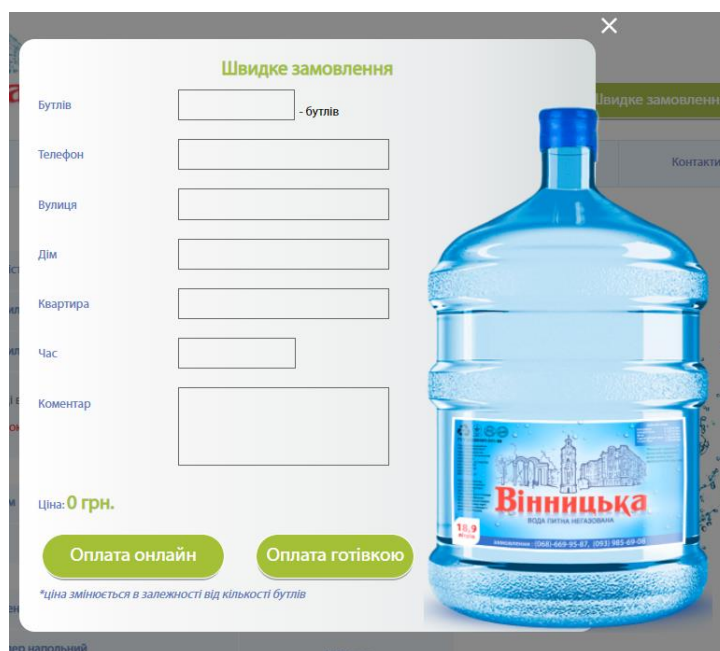
Ціна: 0 грн.

Оплата онлайн Оплата готівкою

*Ціна змінюється в залежності від кількості бутлів

Вінницька

Рисунок 1.5 - форма першого замовлення сайту vinnitska.com



Швидке замовлення

Бутлів - бутлів

Телефон

Вулиця

Дім

Квартира

Час

Коментар

Ціна: 0 грн.

Оплата онлайн Оплата готівкою

*Ціна змінюється в залежності від кількості бутлів

Вінницька

Рисунок 1.6 - форма швидкого замовлення сайту vinnitska.com

Форми першого та швидкого замовлення мають наступні спільні поля: бутлів, телефон, вулиця, дім, квартира, час, коментар. Форма першого замовлення має додаткові 3 поля: «помпа», «бутиль», «на обмін».

1.3 Аналіз вимог до інформаційної технології доставки води по місту

У ході виконання магістерської кваліфікаційної роботи необхідно вивчити предметну область доставки води по місту.

Інформаційні технології в сфері доставки води по місту відіграють важливу роль у покращенні ефективності, безпеки та якості обслуговування. Основні вимоги до інформаційної технології для замовлення та доставки води по місту включають:

- розробку зручної та інтуїтивно зрозумілої онлайн-платформи або мобільного додатка для замовлення води клієнтами;
- можливість відстеження статусу замовлення в режимі реального часу;
- розробку системи, яка автоматично оптимізує маршрути доставки для зменшення часу та витрат пального;
- можливу інтеграцію з GPS технологією для точного визначення місцезнаходження транспортних засобів;
- розробку системи, що повідомляє про можливі ризики нестачі води та дозволяє оперативно реагувати на це.
- збір та аналіз даних щодо замовлень, попиту та споживчої поведінки для покращення стратегій обслуговування та планування ресурсів.

Інтеграція цих технологій дозволить підприємствам в галузі замовлення та доставки води по місту покращити якість обслуговування, зменшити витрати та сприяти сталому використанню ресурсів.

Беручи до уваги існуючі рішення, можна виділити основні функціональні можливості якими має володіти нова інформаційна технологія доставки води по місту. Умовно можна виділити дві частини: перша частина доступна всім користувачам; друга частина, до якої має доступ лише власник.

Основною функцією для користувача є можливість здійснювати замовлення.

Розширення асортименту продукції організацій, що займаються доставкою води, включає в себе не лише основний продукт, а й супутні товари, які можуть значно поліпшити водопостачання клієнтів. Такі супутні товари включають у себе водяні фільтри для покращення якості води, насоси для забезпечення надійного тиску, кулери для зручного доступу до охолодженої води та інші аксесуари для забезпечення комфортного використання.

З метою максимальної зручності та доступності для клієнтів, на сайті компанії з доставки води обов'язково повинен бути розгорнутий каталог, в якому можна знайти всі доступні супутні товари. Каталог повинен бути структурований та зручний для навігації, щоб клієнти легко могли знайти необхідні їм продукти.

Додатково, важливо забезпечити можливість додавання декількох товарів у корзину для одного замовлення. Це значно спростить процес покупок для клієнтів, які можуть одним разом вибрати не лише воду, а й необхідні аксесуари. Механізми додавання товарів у корзину та подальшого оформлення замовлення повинні бути інтуїтивно зрозумілими та ефективними, забезпечуючи при цьому безпечність та конфіденційність інформації клієнта.

Застосування інформаційних технологій у цьому контексті може включати в себе не тільки функціональність для додавання товарів у корзину, але й системи персоналізованих рекомендацій, що допомагатимуть клієнтам знайти додаткові товари, які можуть їх зацікавити. Такий підхід покращить користувацький досвід та може позитивно вплинути на обсяги продажів супутніх товарів.

Важливим аспектом ефективного управління системою продажі супутніх товарів для власників чи адміністраторів компаній, які надають послуги доставки води, є наявність зручного та потужного інструментарію для керування товарним асортиментом та клієнтською базою.

Додатково, система повинна надавати можливість власнику або адміністратору переглядати список клієнтів і їхніх особистих даних. Це може включати інформацію про історію замовлень, включаючи деталі оплати та адреси доставки. Забезпечення доступу до цих даних дозволяє ефективно вести облік клієнтів, а також надавати персоналізований сервіс.

Однією з ключових функцій є можливість власнику або адміністратору переглядати замовлення, включаючи їхній стан та історію. Це дозволяє ефективно керувати процесом доставки, вчасно реагувати на зміни у замовленнях та спрощує обробку та відстеження кожного замовлення в режимі реального часу.

Використання інформаційних технологій для цих завдань спрощує і автоматизує багато рутинних операцій, забезпечуючи власникові чи адміністраторові більше часу та ресурсів для стратегічного планування та вдосконалення бізнес-процесів.

Також дуже важливою функцією для власника є можливість швидко дізнаватись про нові замовлення, але постійний моніторинг інтернет-ресурсів не є самим зручним рішенням для цього. Тому в додатку має бути присутня можливість миттєвого оповіщення відповідальних за обробку замовлень людей. Одним із таких рішень може бути повідомлення через месенджери такі як Telegram або Viber.

1.4 Висновок до розділу 1

В даному розділі проаналізовано предметну область, зазначено актуальність дослідження, визначено основну проблему доставки води по місту. Проаналізовано веб-ресурси аналогів: mywatershop.com.ua, ВОДО ОФІС СЕРВІС, vinnitska.com. Визначено їх переваги та недоліки. Запропоновано розробити нову інформаційну технологію, яка дозволить уникнути зазначених недоліків. Здійснено аналіз вимог до нової інформаційної технології доставки води по місту.

2 РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДОСТАВКИ ВОДИ ПО МІСТУ

2.1 Розробка структури бази даних інформаційної технології доставки води по місту

Вибір бази даних для інформаційної технології доставки води по місту є ключовим етапом у проектуванні ефективної та надійної системи. Рішення повинно враховувати обсяг даних, швидкість доступу, безпеку, та інші фактори.

Вибір бази даних залежить від конкретних вимог до проекту, таких як кількість даних, швидкість обробки, типи операцій, та розподіленість системи. Також важливо враховувати технічні навички команди розробників та сумісність бази даних з іншими інфраструктурними компонентами системи.

База даних в інформаційних технологіях у вигляді веб-ресурсу використовується для виконання наступних функцій:

- збереження інформації;
- читання збереженої інформації

База даних містить в собі набір таблиць, що містять інформацію про товари, категорії, коментарі, замовлення. Для побудови схеми бази даних розглянемо таблиці, що вона вміщує, а також їх призначення (табл. 2.1).

Таблиця 2.1 - Перелік таблиць бази даних та їх призначення

Назва таблиці	Призначення таблиці
ORDERS	Зберігає інформацію про замовлення
CUSTOMERS	Зберігає інформацію про клієнтів
REGIONS	Зберігає інформацію про райони міста
ITEMS	Зберігає інформацію про товари
CALLS	Зберігає інформацію про запит на дзвінок
PREDICTIONS	Зберігає інформацію про прогнозовані дзвінки

В таблиці 2.1 перераховано та описано всі таблиці бази даних веб застосунку доставки води по місту. Для схематичного зображення бази даних використовується діаграму класів що відображає таблиці бази даних та зв'язок між ними. Діаграма класів зображено на рисунку 2.1.

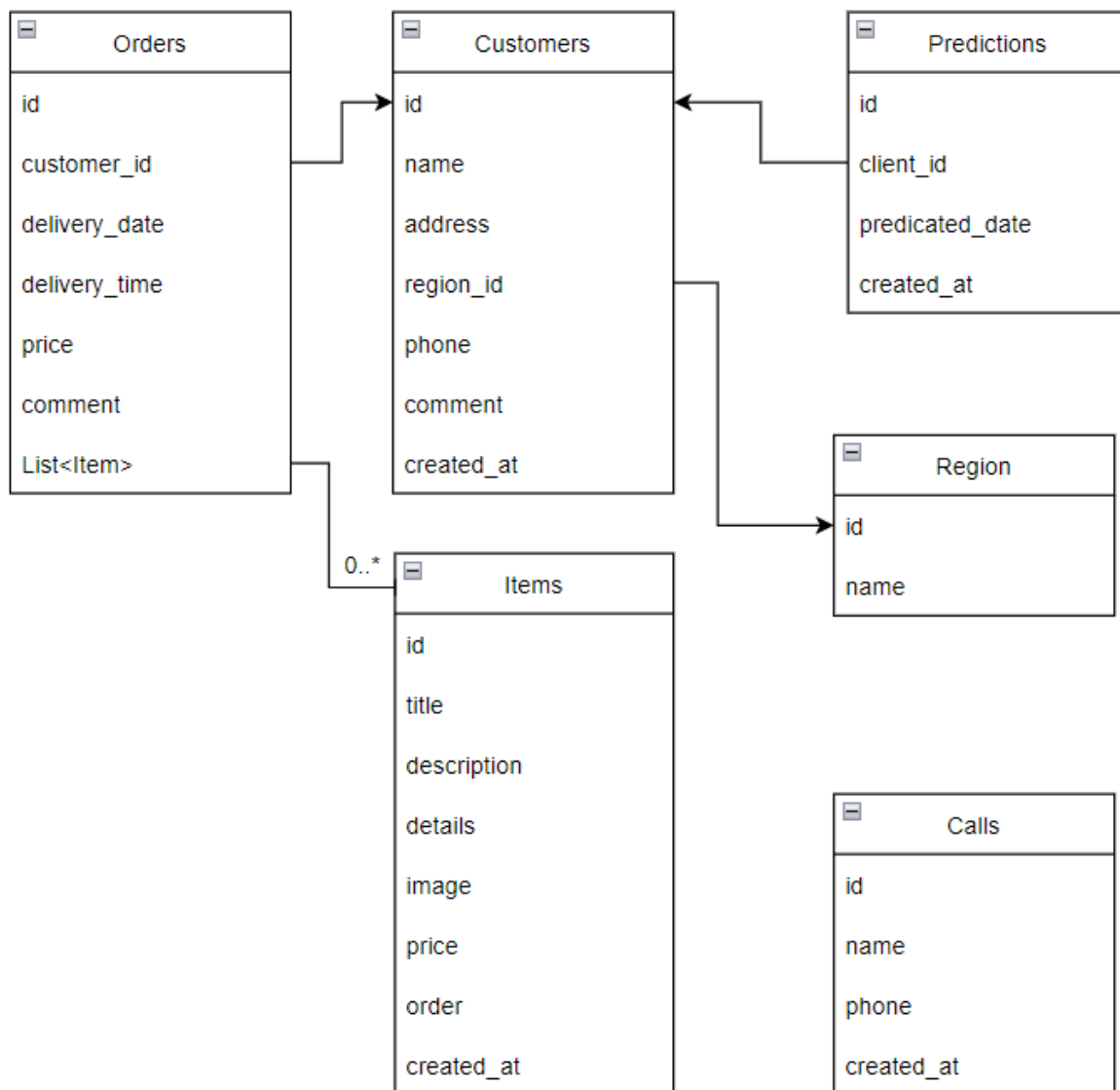


Рисунок 2.1 – Діаграма класів

З рисунку 2.1 видно що таблиця ORDERS пов'язана з таблицею CUSTOMERS причому одне замовлення може відноситись тільки одного клієнта. Також таблиця ORDERS має зв'язок один до багатьох з таблицею ITEMS, тому що можна додати декілька товарів до одного замовлення. Таблиця PREDICTION також має зв'язок один до одного з таблицею CLIENTS. CLIENTS в свою чергу має зв'язок з таблицею

REGION і оскільки один користувач може жити в одному регіоні, то вони мають зв'язок один до одного. Таблиця CALLS є окремою, оскільки запит на дзвінок здійснюється без уточнення інформації і містить лише номер телефону.

2.2 Розробка структури інформаційної технології доставки води по місту

Структура інформаційної технології доставки води по місту відображає призначення та взаємодію головних структурних модулів між собою та з користувачем. В інформаційній технології доставки води по місту є наступні модулі: модуль графічного інтерфейсу, модуль введення запиту користувача, модуль обробки запиту, модуль доступу до даних, модуль підготовки інформації для виводу, модуль що відповідає за роботу телеграм бота, модуль виведення результатів. Точкою входу в програму є головна сторінка, з якої починається навігація програмою по інших її складових. Розроблена загальна структурна схема функціонування веб застосунку доставки води по місту зображено на рисунку 2.2.

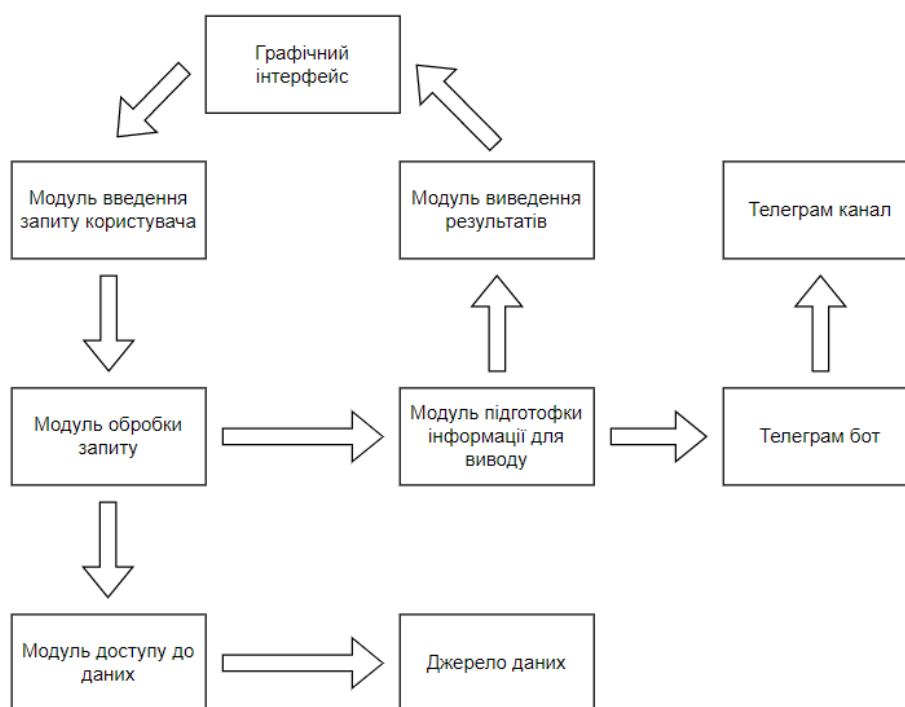


Рисунок 2.2 – Структура інформаційної технології доставки води по місту

Взаємодія модулів на прикладі здійснення замовлення зображена на рисунку 2.3.



Рисунок 2.3 – Діаграма послідовності здійснення замовлення

Після заповнення форми замовлення користувачем із графічного інтерфейсу через модуль введення інформації запит відправляється на обробку сервером. Сервер приймає цей запит в модулі обробки. Якщо запит був оброблений успішно, то він зберігається в базі за допомогою модуля збереження даних. Потім дані потрапляють в модуль підготовки інформації для виводу де відбувається генерація відповіді сервера на отриманий запит. Даний результат відправляється на клієнт з подальшим повідомленням для користувача. Також з модуля підготовки інформації підготовлене повідомлення перенаправляється в модуль телеграм боту, який здійснює його подальшу відправку власнику в телеграм.

2.3 Аналіз вимог до інформаційної технології доставки води по місту

Розробка та впровадження інформаційної технології доставки води по місту вимагає врахування різноманітних аспектів для забезпечення ефективності, надійності та зручності для користувачів. Ось деякі загальні вимоги до інформаційної технології доставки води по місту:

- забезпечення можливості онлайн-замовлення води через веб-сайт та мобільні додатки для різних платформ;
- інтуїтивно зрозумілий та дружній інтерфейс для користувачів;
- можливість відстеження статусу замовлення в режимі реального часу;
- автоматичні сповіщення користувачам про підтвердження замовлення, стан доставки та завершення замовлення;
- можливість вибору різних типів води (питної, каліброваної, газованої тощо) та об'ємів у замовленні;
- опції додаткових супутніх товарів (фільтри, насоси, кулери);
- безпечна система електронної оплати з використанням різних методів оплати (кредитні карти, електронні гроші, банківські перекази тощо);
- ефективна система контролю за запасами води та супутніми товарами на складі;
- автоматизована система перезамовлення для уникнення нестачі;
- можливість генерації звітів про замовлення, витрати води, обсяги продажу та інші ключові метрики;
- інструменти аналізу даних для планування маркетингових кампаній та стратегій розвитку;
- ефективна система обробки та відстеження замовлень з можливістю швидкої реакції на зміни або непередбачені обставини;
- система клієнтської підтримки через різні канали (онлайн-чат, телефон, електронна пошта).

Забезпечення виконання цих вимог допоможе створити інноваційну та конкурентоздатну інформаційну технологію для доставки води по місту, сприяючи покращенню якості обслуговування та задоволеності клієнтів.

Розробимо UML діаграму діяльності інформаційної технології доставки води по місту.

Unified Modeling Language (UML) - це стандартизована мова моделювання, яка використовується для візуалізації, проектування та документування програмних систем. Використання UML-діаграм сприяє розробці інформаційної технології доставки води по місту, надаючи зручний та зрозумілий спосіб представлення різних аспектів системи. До типових UML-діаграм відносяться:

- UML-діаграма використання - використовується для ідентифікації та моделювання функціональності системи з точки зору її користувачів - визначає ролі користувачів та їхні взаємодії з системою;

- UML-діаграма класів (Class Diagram) - моделює структуру системи шляхом визначення класів, їхніх властивостей, методів та взаємозв'язків. Ілюструє основні сутності та їхні залежності в системі;

- UML-діаграма послідовності (Sequence Diagram) - показує послідовність обміну повідомленнями між об'єктами системи. Ілюструє взаємодії в часі та порядок виконання операцій;

- UML-діаграма станів (State Diagram) - використовується для моделювання різних станів об'єктів та переходів між ними відповідно до подій. Допомагає відобразити реакції системи на події або зміни у внутрішньому стані;

- UML-діаграма діяльності (Activity Diagram) - використовується для моделювання послідовності дій або діаграмування бізнес-процесів. Відображає потік роботи та взаємодії між різними діями та об'єктами;

- UML-діаграма розгортання (Deployment Diagram) - моделює фізичне розташування компонентів системи на апаратному обладнанні. Визначає, як компоненти розподілені та взаємодіють на різних вузлах.

Використання цих UML-діаграм дозволяє команді розробників та зацікавленим сторонам краще зрозуміти, спільно працювати та узгоджувати різні аспекти

інформаційної технології доставки води по місту, що сприяє ефективному та системному розвитку проекту.

Діаграма варіантів використання - прийом для комунікації поведінки системи з точки зору кінцевого користувача за допомогою візуалізації зовнішньої видимої поведінки системи. Діаграма випадків використання - це проста діаграма високого рівня. Він не відображає деталей випадків використання чи кроків для досягнення певної мети. Натомість він відображає лише взаємозв'язки між випадками використання, акторами та системами.

При розгляді сценарію використання інформаційної технології доставки води по місту можна виділити два види користувачів клієнта та адміністратор. Діаграма варіантів використання зображена на рисунку 2.4.

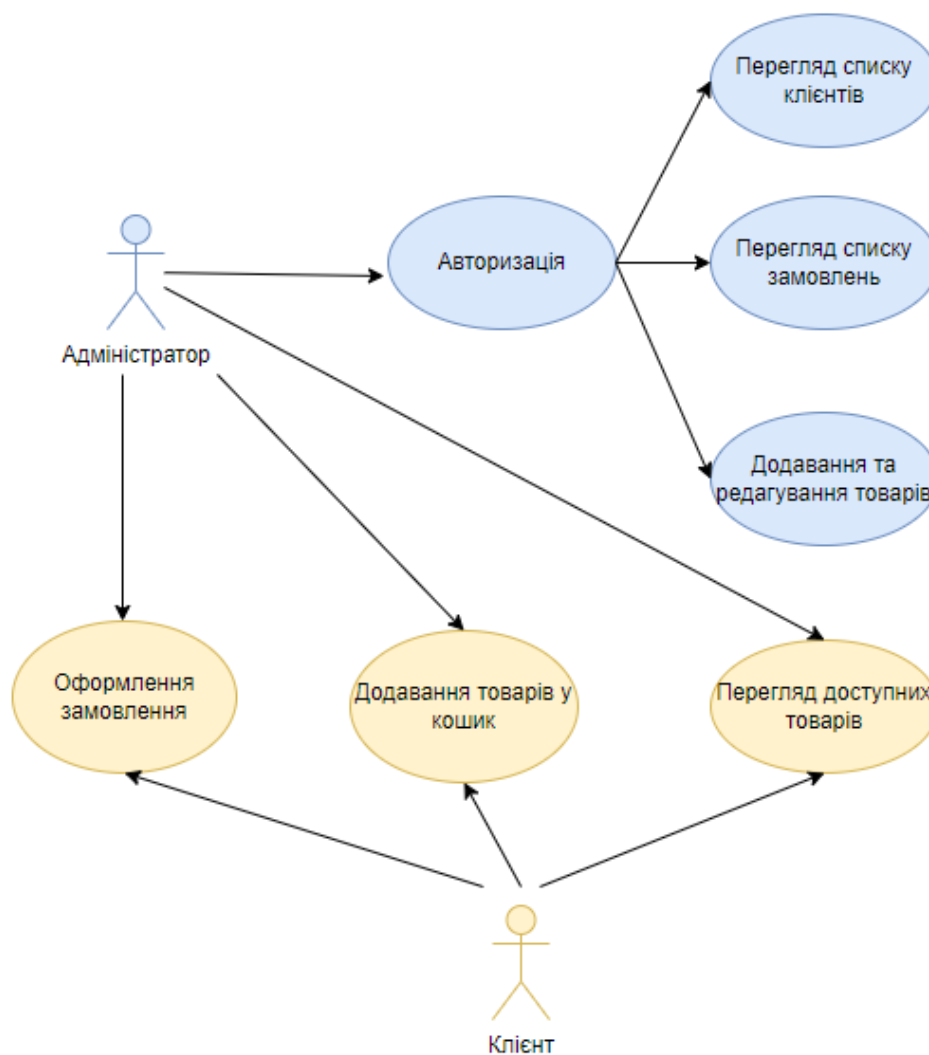


Рисунок 2.4 - Діаграма варіантів використання

Клієнт має можливість виконувати наступні дії:

- перегляд списку доступних товарів;
- додавання чи видаляти товари з кошика;
- створювати замовлення.

Також застосунок має містити модуль доступний адміністратору. Для обмеження доступу до цього ресурсу сторінки з боку користувачів, адміністратор спочатку має пройти етап авторизації, після чого для нього відкриваються наступні можливості:

- перегляд списку клієнтів;
- перегляд списку замовлень;
- додавання та редагування товарів.

Розглядаючи сценарії використання інформаційної технології для доставки води по місту, можна визначити два ключових види користувачів системи: клієнти та адміністратори. Кожна з цих груп відіграє важливу роль у забезпеченні ефективної та зручної роботи сервісу. Клієнти є основними використовувачами інформаційної технології доставки води. Клієнти можуть користуватися мобільним додатком чи веб-порталом для здійснення замовлення, вказання адреси доставки, вибору кількості та типу води. Адміністратори відповідають за ефективне управління та координацію процесів доставки води. Вони використовують інформаційну технологію для моніторингу замовлень, визначення оптимальних маршрутів доставки, а також для ведення обліку запасів води.

Взаємодія між цими двома групами користувачів дозволяє створити збалансовану та високоефективну систему доставки води, де кожна сторона виграє від використання інформаційної технології.

2.4 Висновок до розділу 2

В даному розділі, в контексті веб застосунку доставки води по місту, було розроблено інформаційну технологію доставки води по місту, розроблено структуру бази даних, структурну схему, яка відображає модулі проекту, та діаграму сценаріїв використання.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДОСТАВКИ ВОДИ ПО МІСТУ

3.1 Обґрунтування вибору засобів розробки

Під час розробки магістерської кваліфікаційної роботи використовувався редактор коду Visual Studio Code.

Visual Studio Code - редактор вихідного коду, розроблений Microsoft для Windows, Linux і macOS. Позичується як «легкий» редактор коду для кроссплатформеної розробки веб і хмарних додатків. Включає в себе відладчик, інструменти для роботи з Git, підсвічування синтаксису, IntelliSense і засоби для рефакторингу. Має широкі можливості для налаштувань: користувацькі теми, поєднання клавіш і файли конфігурації. Розповсюджується безкоштовно, розробляється як програмне забезпечення з відкритим вихідним кодом (рис. 3.1).

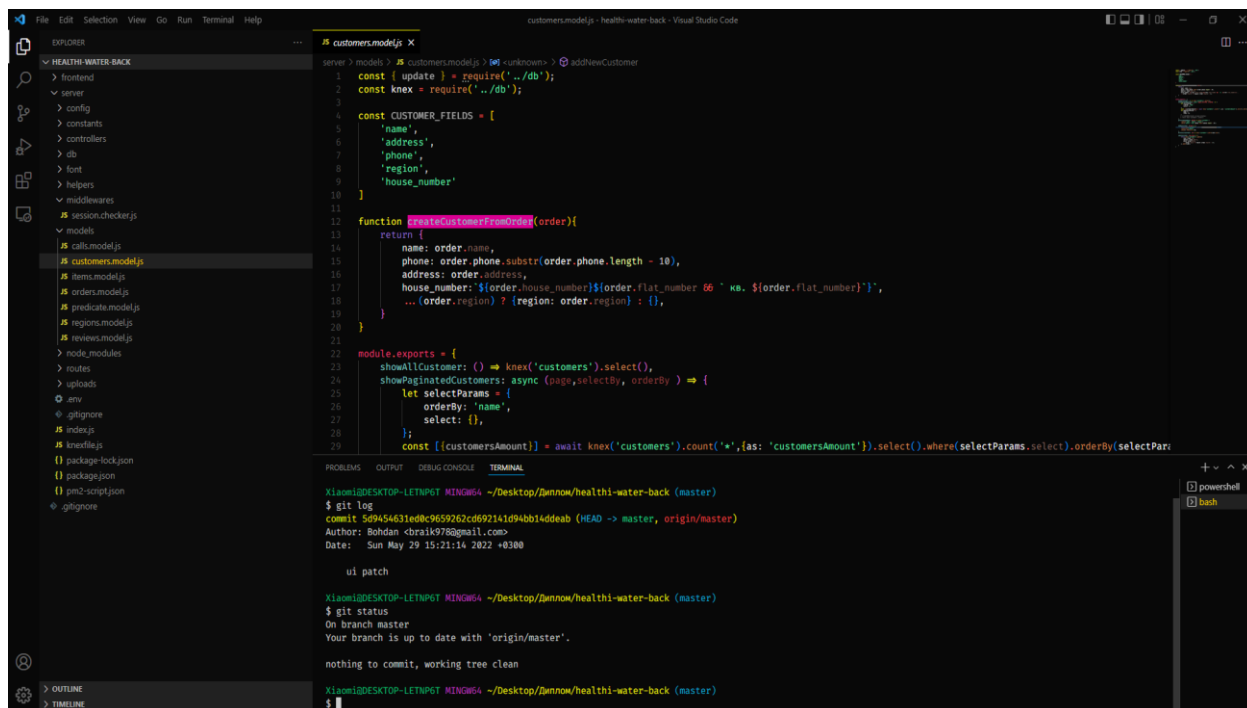


Рисунок 3.1 – вікно редактора коду Visual Studio Code

VS Code не варто плутати з Visual Studio - це IDE середовище, дуже потужне і масштабне, але водночас і важке. Назви схожі, тому що в обох засобів розробки один і той же виробник, але продукти різні за своєю суттю. VS Code «менше важить», простіше в освоєнні і підходить у тому числі для розробників-початківців.

Редактор коду існує для всіх популярних операційних систем: Windows, Linux та MacOS. Він безкоштовний, на відміну більшості версій «старшого брата» Visual Studio.

З VS Code працюють програмісти різними мовами. Наприклад, ним активно користуються веб-розробники, які пишуть HTML/CSS, JavaScript, PHP. Але редактор підтримує набагато більше мов: Python, Go, Ruby, C#, TypeScript і так далі. Він працює і з розширеннями та фреймворками для популярних мов – наприклад, з React JS та Vue.js, з мовами стилів SCSS та LESS, які доповнюють CSS.

Теоретично користуватися VS Code може розробник практично будь-якою з сучасних мов. Але на практиці його застосовують там, де не потрібні потужності повноцінної IDE.

VS Code дозволяє легко писати, формувати та редагувати код різними мовами. З його допомогою можна швидко створити проект та структуру файлів у ньому, він підсвічує синтаксис коду та допомагає автоматично правити помилки. У ньому є можливості для налагодження та запуску коду деякими мовами.

Редактор легко розширюється, тому до перелічених функцій можна додати нові – досить просто завантажити потрібне доповнення з офіційного каталогу. Доповнення також розповсюджуються безкоштовно.

Основні можливості:

- Інтеграція налаштувань з Sublime, Atom. Після установки VS Code перевіряє наявність другого IDE. Якщо знайдена Sublime або Atom VS Code запропонує частково перенести налаштування і швидкі клавіші в VS Code.

- Розширення. Зручна система додавання розширень. Досить перейти у вкладку з розширеннями і написати в пошуку необхідне доповнення.

- Git. Зручна робота з системою контролю версій. Робота з нею така ж, як і в Visual Studio 2019. Відображення коммітів, історії, гілок і т.д.

- Пошук. Розумний пошук з можливістю заміни за різними критеріями.
- Debug (Запуск додатків). Для веб проектів вже інтегровано налагодження. Для інших необхідно встановити розширення. Відображення використовуваних змінних при дебазі. Відображення помилок (для Деяких мов потрібно встановити розширення).
- Інтегрований термінал. Можна працювати через неї з Git і налагоджувати додаток.
- Налаштування. Велика кількість налаштувань, як і в SublimeText і Atom. Застосовуються моментально, перезапуск IDE НЕ потрібно.
- Візуальне оформлення. Можна кардинально змінити вигляд VS Code. Також є реалізовані теми, які застосовуються в один клік.

3.2 Обґрунтування вибору мови програмування

Для розробки проекту було обрано мову програмування JS. JavaScript (Джава сценарій) — назва реалізації стандарту мови програмування ECMAScript компанії Netscape, базується на принципах прототипно-орієнтованого програмування. Найпоширеніше і найвідоміше застосування мови — написання сценаріїв для веб-сторінок, але, також, використовується для впровадження сценаріїв керування об'єктами вбудованими в інші програми.

JavaScript - високорівнева мова програмування, яка підтримує імперативний, функціональний, подієво-орієнтований підходи. Вона має динамічну типізацію та застосовується для запису послідовних операцій — «сценаріїв» чи «скриптів». Такі послідовності зазвичай інтерпретуються, а не компілюються, а тому не потребують додаткових програм чи інструментів перетворення в інший рівень кодування.

Кожен веб-застосунок чи сайт побудований з використанням трьох технологій - HTML, CSS та JavaScript. Остання виступає «мозком» розробки й відповідає за інтерактивність й взаємодію з користувачем.

Варто зазначити, що все частіше компанії не обмежуються роботою із JavaScript лише в браузері, а й користуються платформою Node.js, середовищем

виконання JavaScript-коду, для написання серверних застосунків. Яскравими прикладами проєктів за такої комбінації є Netflix та PayPal.

Тож багатоплановість застосування — це ще кілька балів до перспективності JavaScript.

На сьогоднішній день ця мова дуже активно розвивається і застосовується не тільки при розробці веб сторінок, а й для написання серверних програм та кросплатформених додатків для персональних комп'ютерів та мобільних платформ.

Можливості JS:

- застосування різноманітних ефектів на сайті;
- додавання/видалення тегів;
- зміна вмісту HTML-елементів, зміна стилів;
- реагування на події, які пов'язані з тими чи іншими діями користувачів, для прикладу, клік мишки;
- виконання запитів до сервера;
- запам'ятовування даних зі сторони клієнта.

Node.js - середовище виконання JavaScript, побудоване на движку V8. Node.js використовує подійно-орієнтовану модель з неблокуючим введенням/виведенням, що робить його легким і ефективним.

Node.js — це однопоточне кросплатформове середовище виконання з відкритим вихідним кодом і бібліотека, яка використовується для запуску вебдодатків, написаних на JavaScript, поза браузером клієнта. Звучить трохи складно, правда? Простіше кажучи, Node.js — це програмне середовище, яке дозволяє запускати програми, написані мовою Javascript, поза браузером. Історично програми, написані на Javascript, на відміну від інших мов програмування, можна було запустити лише у браузерах, які мали спеціальний вбудований движок виконання коду цієї мови. Поза браузером Javascript, можна сказати, не працював. При розробці Node.js за основу було взято двигун виконання JavaScript під назвою V8, який був створений компанією Google і використовувався в браузері Google Chrome. Оскільки після створення Node.js Javascript код можна запустити фактично в будь-якому

середовищі, за допомогою цієї бібліотеки можна написати не лише фронтенд, а й серверну частину вебпрограми.

Node.js - це платформа на основі подій, що означає, що все, що відбувається в Node.js, є реакцією на подію. Цикл подій називається серцем Node.js. Він безстроково отримує запити та обробляє їх, а потім повертає відповіді відповідним клієнтам. Архітектура Node.js зображена на рисунку 3.2

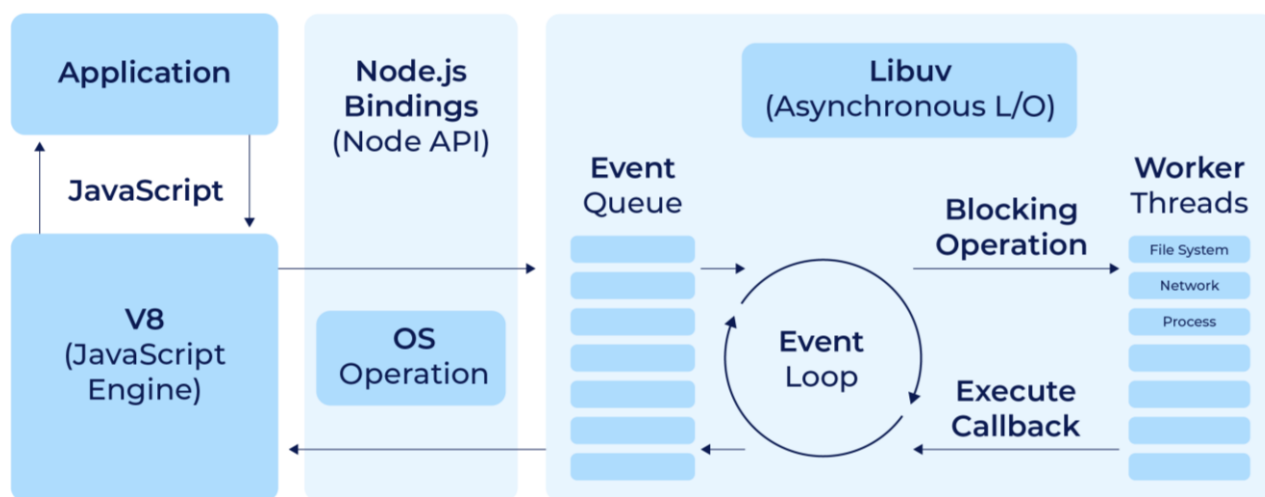


Рисунок 3.2 – Архітектура платформи Node.js

Іншими словами: Node.js дає можливість писати сервери за допомогою JavaScript з неймовірною продуктивністю. Як сказано у документації: Node.js - середовище виконання, яке використовує той самий рушій V8 Javascript, який ви можете знайти в браузері Google Chrome. Але це не достатньо для успіху Node.js - Node.js використовує libuv, мультиплатформенну бібліотеку з фокусом на асинхронний I/O.

2.3 Обґрунтування вибору системи управління базами даних

Для вибору системи управління базами даних розгляним найпопулярніші із них:

- MySQL;

- PostgreSQL;
- SQLite.

MySQL – це популярний сервер баз даних, який використовується в різних додатках. MySQL використовує SQL для комунікації з іншими програмами. Понад те, MySQL має свої власні розширені функції SQL для того щоб забезпечити користувачам додатковий функціонал. У цьому документі ми розглянемо як провести первинну установку MySQL, налаштувати бази даних і таблиці, і створити нових користувачів.

SQLite - компактна вбудована реляційна база даних. Вихідний код бібліотеки переданий в суспільне надбання. Є чисто реляційною базою даних. Слово «вбудовується» означає, що SQLite не використовує парадигму клієнтсервер. Тобто движок SQLite не є окремо працюючим процесом, з яким взаємодіє програма, а надає бібліотеку, з якої програма компонується і движок стає складовою частиною програми.

PostgreSQL – об'єктно-реляційна система керування базами даних. Є альтернативою як комерційним СКБД (Oracle Database, Microsoft SQL Server, IBM DB2 та інші), так і СКБД з відкритим кодом (MySQL, Firebird, SQLite). Postgres має безліч можливостей. Створений з використанням об'єктно-реляційної моделі, він підтримує складні структури та широкий спектр вбудованих та визначених користувачем типів даних. Він забезпечує розширену ємність даних і заслужив довіру дбайливим ставленням до цілісності даних.

Зважаючи на вище перелічені характеристики для подальшої розробки було обрано PostgreSQL як систему управління базами даних.

3.4 Обґрунтування використання платформи Docker

Docker надає можливість пакувати та запускати програму в слабо ізольованому середовищі, яке називається контейнером. Ізоляція та безпека дозволяють запускати багато контейнерів одночасно на даному хості. Контейнери легкі та містять все необхідне для запуску програми, тому немає необхідності турбуватись про те, що в

даний момент встановлено на хості. З його допомогою можна легко ділитися контейнерами під час роботи, і гарантовано всі отримають той самий контейнер, який працює так само.

Docker - це програмне забезпечення, яке дає можливість на певній ділянці пам'яті ізолювати встановити необхідну ОС (операційну систему), версію Java, налаштувати змінні оточення, встановити різні залежності і дати доступ тільки за певних умов. При цьому дану програму абсолютно не буде хвилювати, що відбувається навколо.

Docker спрощує життєвий цикл розробки, дозволяючи розробникам працювати в стандартизованих середовищах, використовуючи локальні контейнери, які надають ваші програми та послуги. Контейнери чудово підходять для безперервної інтеграції та безперервної доставки (CI/CD) робочих процесів.

Docker використовує архітектуру клієнт-сервер (рис. 3.3). Клієнт Docker спілкується з демоном Docker, який виконує важку роботу зі створення, запуску та розповсюдження контейнерів Docker. Клієнт і демон Docker можуть працювати в одній системі, або можна підключити клієнта Docker до віддаленого демона Docker. Клієнт Docker і демон спілкуються за допомогою REST API, через сокети UNIX або мережевий інтерфейс. Іншим клієнтом Docker є Docker Compose, який дозволяє працювати з додатками, що складаються з набору контейнерів.

Основною частиною всієї системи Docker є Docker Engine. Docker Engine — це програма, яка відповідає архітектурі клієнт-сервер. Він встановлюється на хост-машині. У Docker Engine є три компоненти:

- Сервер: це демон докерів під назвою `dockerd`. Він може створювати зображення докерів і керувати ними. Контейнери, мережі тощо.
- Rest API: використовується для інструктування демона докера, що робити.
- Інтерфейс командного рядка (CLI): це клієнт, який використовується для введення команд докера.

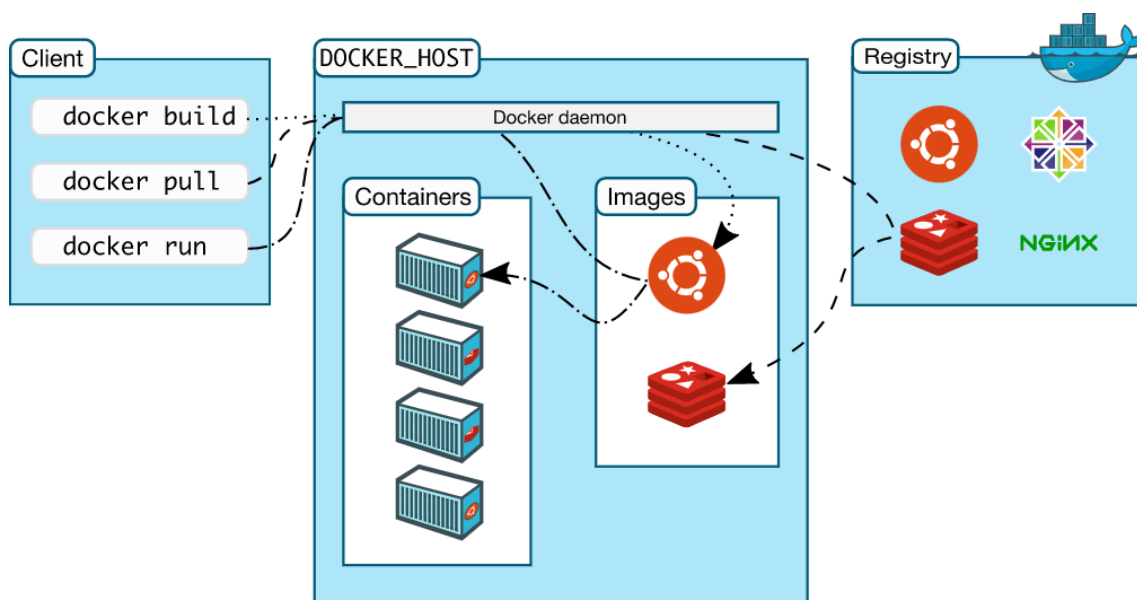


Рисунок 3.3 – архітектура Docker

Користувачі Docker можуть взаємодіяти з Docker через клієнта. Коли виконуються будь-які команди docker, клієнт надсилає їх до демона dockerd, який їх виконує. API Docker використовується командами Docker. Клієнт Docker може спілкуватися з більш ніж одним демоном.

Реєстри Docker – це місце, де зберігаються зображення Docker. Це може бути публічний реєстр докерів або приватний реєстр докерів. Docker Hub — це місце за замовчуванням для зображень докерів, загальнодоступного реєстру його магазинів. Ви також можете створити та запустити власний приватний реєстр.

Коли ви працюєте з Docker, ви використовуєте зображення, контейнери, томи, мережі; все це об'єкти Docker.

Зображення Docker – це шаблони лише для читання з інструкціями щодо створення контейнера Docker. Образ Docker можна витягнути з центру Docker і використовувати як він є, або ви можете додати додаткові інструкції до базового образу та створити нове та змінене зображення Docker. Ви можете створювати власні зображення докерів також за допомогою файлу docker. Створіть dockerfile з усіма інструкціями, щоб створити контейнер і запустити його; це створить ваше власне зображення Docker.

Зображення Docker має базовий шар, доступний лише для читання, а верхній шар можна записати. Коли ви редагуєте файл Docker і перебудуєте його, лише змінена частина перебудовується у верхній шар.

Після запуску образу докера він створює контейнер докерів. Усі програми та їх середовище працюють всередині цього контейнера. Ви можете використовувати Docker API або CLI, щоб запускати, зупиняти, видаляти контейнер докерів.

Нижче наведено зразок команди для запуску контейнера `docker ubuntu`:

```
docker run -i -t ubuntu /bin/bash
```

Коли ви виконуєте команди `docker pull` або `docker run`, потрібний образ докера витягується з налаштованого реєстру. Коли ви виконуєте команду `docker push`, образ докера зберігається в налаштованому реєстрі.

Постійні дані, створені докером що використовуються контейнерами Docker, зберігаються в томах. Вони повністю керуються докером через `docker CLI` або `Docker API`. Вони працюють на контейнерах Windows і Linux. Замість того, щоб зберігати дані в доступному для запису шарі контейнера, завжди є хорошим варіантом використовувати для них томи. Вміст тому існує за межами життєвого циклу контейнера, тому використання обсягу не збільшує розмір контейнера.

Ви можете використовувати прапор `-v` або `--mount`, щоб запустити контейнер з томом. У цьому прикладі команди ви використовуєте том `geekvolume` з контейнером `geekflare`.

```
docker run -d --name geekflare -v geekvolume:/app  
nginx:latest
```

Мережа Docker — це шлях, через який спілкуються всі ізольовані контейнери. У `docker` є в основному п'ять мережевих драйверів:

1) **Bridge**: це мережевий драйвер за замовчуванням для контейнера. Ви використовуєте цю мережу, коли ваша програма працює на автономних контейнерах, тобто декілька контейнерів, які спілкуються з одним хостом докерів.

2) **Host**: цей драйвер усуває мережеву ізоляцію між контейнерами докерів і хостом докерів. Він використовується, коли вам не потрібна мережева ізоляція між хостом і контейнером.

3) **Overlay**: ця мережа дає змогу службам роя спілкуватися один з одним. Він використовується, коли контейнери запущені на різних хостах Docker або коли сервіси роя формуються кількома додатками.

4) **None**: цей драйвер вимикає всю мережу.

5) **macvlan**: цей драйвер призначає mac-адресу контейнерам, щоб вони виглядали як фізичні пристрої. Трафік маршрутизується між контейнерами через їхні mac-адреси. Ця мережа використовується, коли ви хочете, щоб контейнери виглядали як фізичний пристрій, наприклад, під час міграції налаштування VM.

Використання Docker при розробці серверної частини веб застосунку допоможе спростити оптимізувати життєвий цикл розробки та полегшити процеси розгортання проекту на фінальних стадіях розробки.

3.5 Розробка серверної частини програмного забезпечення для доставки води по місту

Більшість Node.js-фреймворків не мають фіксованої структури каталогів і може бути непросто від початку проекту дотримуватися правильної структури. У зв'язку з цим дуже важливо на перших етапах розробки правильно структурувати проект на Node.js, щоб уникнути складнощів, коли ваші програми почнуть рости. Для реалізації серверної частини веб застосунку доставки води по місту була створена файлова структура яка зображена на рисунку 3.4.

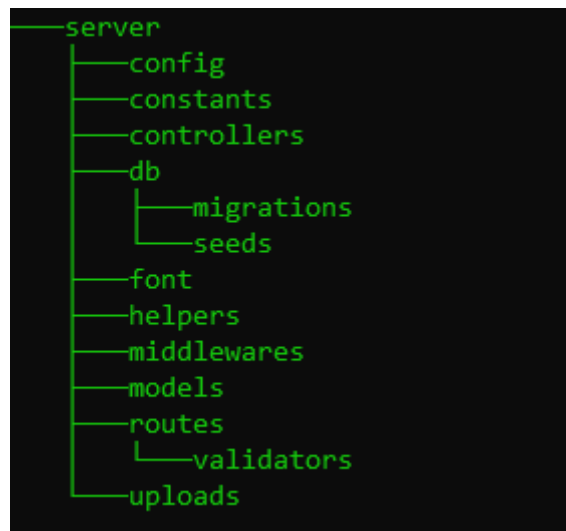


Рисунок 3.4 – Файлова структура серверної частини проекту

Кожна папка має своє призначення:

- config – для збереження загальних даних конфігурації проекту такі як порт на якому працюватиме сервер, дані підключення до бази даних і тд.;
- constants – для збереження постійних змінних які використовуються в проекті;
- controllers – для збереження файлів контролерів (MVC);
- db – для збереження файлів підключення до бази даних а також файлів міграції та сідів;
- helpers – для збереження файлів додаткової додаткової логіки яка не стосується бізнесу;
- midelvares – для збереження файлів з функціями проміжкової обробки;
- models – для збереження файлів моделей (MVC);
- routers – для збереження файлів кінцевих точок (endpoints);
- routers/validators – для збереження файлів з логікою валідаторів даних які відправляє користувач;
- uploads – для збереження зображень які завантажує клієнт.

Серверна частини являє собою REST API який реалізує кінцеві точки які наведені в таблиці 3.1.

Таблиця 3.1 – Опис всіх наявних кінцевих точок

Маршрут	Метод	Призначення
order/	GET	Список замовлень
order/	POST	Додавання замовлення
order/:id	DELETE	Видалення замовлення
order/pdf	GET	Отримання списку замовлень в pdf
items/	GET	Список товарів
items/	POST	Додавання товару
items/:id	PUT	Редагування товару
items/:id	DELETE	Видалення товару
customers/	GET	Список товарів
customers/	POST	Додавання нового клієнта
customers/:id	PUT	Редагування клієнта
customers/:id	DELETE	Видалення клієнта
regions/	GET	Список районів міста
regions/	POST	Додавання району міста
regions/:id	PUT	Редагування району міста
regions/:id	DELETE	Видалення району міста
review/	GET	Список відгуків
review/	POST	Додавання відгуку
review/:id	PUT	Редагування відгуку
review/:id	DELETE	Видалення відгуку
admin/	GET	Отримання сесії авторизації
admin/login	POST	Авторизація
admin/logout	GET	Видалення сесії авторизації

Нижче приведений код маршрутів для роботи з товарами:

```

const itemController = require('../controllers/item.controller');
const multer = require("multer");

```

```

const authCheck =
require('../middlewares/session.checker');
const storage = multer.diskStorage({
  destination: function (req, file, cb) {
    cb(null, 'uploads/')
  },
  filename: function (req, file, cb) {
    cb(null, Date.now() + '.jpg') //Appending .jpg
  }
})
const upload = multer({
  dest:"uploads",
  storage: storage,
  limits:{
    fileSize: 1024 * 1024
  }});
const router = require('express').Router();
router.post('/:id', authCheck, upload.single('image'),
itemController.createItem);
router.post('/', authCheck, upload.single('image'),
itemController.createItem);
router.get('/', itemController.showItems);
router.delete('/:id', authCheck,
itemController.deleteItems);
module.exports = router;

```

При обробці кінцевих точок для додавання та редагування товарів використовується проміжкова функція (middleware) для завантаження зображень. Дана функція реалізована з використанням бібліотеки multer.js.

Далі наведено код функції контролера для додавання товару:

```
createItem : async (req, res) => {
```

```

const fileData = req.file;
const id = req.params.id;
let item;
if(id){
    const [oldItem] = await
itemsController.getItemById(id);
    const newItem = {
        ...req.body,
        ...(fileData ? {image: fileData.filename}
: {}))
    }
    if(!fileData) {
        delete newItem.image;
    }

    item = await updateItem( newItem, id)
    if(fileData && oldItem.image){
        await removeUploaded(oldItem.image);
    }
} else {
    item = await createItem(
        {
            ...req.body,
            ...(fileData ? {image:
fileData.filename} : {}))
        }
    );
}
res.status(200).send(item);
}

```

Дана функція контролера є універсальною для додавання і редагування товару. В ній відбувається перевірка на наявність параметру “id” в тілі запиту і в разі її наявності виконує редагування існуючого товару, в протилежному випадку додає новий.

Робота з базою даних відбувається в файлах моделей (models).

```
module.exports = {
    showAllItems:      ()          =>
knex('items').select().orderBy('order'),
    addNewItems:      (props,      image)    =>
knex('items').insert(datePipe(props, image)).returning('id')
    .then(([id]) =>{
        return knex('items').where('id',id)
    }),
    updateItem:      (props,      id)        =>
knex('items').update(datePipe(props)).where({id}).returning(
'id')
    .then(([id]) =>{
        return knex('items').where('id',id)
    }),
    getItemById:      (id)            =>
knex('items').select().where({id}),
    deleteItemsById:      (id)        =>
knex('items').where({id}).del(),
}
```

Запити до бази даних реалізовані з використанням бібліотеки knex.js.

При додаванні замовлення дані одночасно зберігаються в базу даних і відправляється повідомлення на в телеграм канал. Код, який реалізує генерацію повідомлення та його відправлення наведений нижче:

```
add: async (req, res) => {
    const errors = validationResult(req);
```

```

    if (!errors.isEmpty()) {
        res.status(422).json({ errors: errors.array()
    });

        return;
    }
    const order = req.body;
        const customer = await
customerModule.getCustomerByPhone(order.phone);
    if(customer.length < 1){
        await customersModel.addNewCustomer(order);
    }

        const message =
`Замовлення\n${createMessage(order)}`
    await Promise.all([
        sendToTelegram(message),
        orderModule.addNewOrder(order),
    ]);
    res.status(200);
    res.json({message: 'order sent'})
},

```

Реалізація методу відправки повідомлення `sendToTelegram` винесений в окремий `telegram.helper.js` файл наведений далі.

```

const Telegraf = require('telegraf');
const { CHAT_ID, TOKEN } = require('../constants')
const Bot = new Telegraf(TOKEN);
exports.sendToTelegram = (message) =>
Bot.telegram.sendMessage(CHAT_ID, message);

```

Метод реалізований з використанням бібліотеки `telegraf.js`.

Також важливою частиною при обробці даних з клієнта на сервері є валідація. На `express.js` вона дуже просто реалізується, для цього можна використати бібліотеку `express-validator`. Код валідатора додавання товарів зображений нижче:

```
const { check } = require('express-validator');
module.exports = [
  check('title').isString().notEmpty(),
  check('description').isString().notEmpty(),
  check('image').notEmpty(),
  check('price').isNumeric().notEmpty(),
];
```

Даний валідатор задає наступні правила для полів:

- поле `title` – має бути текстовим рядком і не бути пустим;
- поле `description` – має бути текстовим рядком і не бути пустим;
- поле `image` – не має бути пустим;
- поле `price` – має бути числом і не бути пустим;

Для повного контролю над даними валідація зазвичай поводить як на сервері так і на клієнті. Код валідації на клієнті розглядається в наступному підрозділі.

3.6 Розробка клієнтської частини програмного забезпечення для доставки води по місту

Так само як і при розробці серверної частини, під час створення проекту клієнта з використанням `React.js`, потрібно спочатку створити файлову структуру проекту (рис. 3.5).



Рисунок 3.5 – Файлова структура клієнтської частини проекту

Кожна папка має ім'я яке відповідає їх призначенню:

`public` – для збереження загальних статичних файлів проєкті такі як лого сайту, `manifest.js` і тд.;

`src/api` – для збереження файлів реалізації запитів до серверу;

`src/common` – для збереження компонентів та стилів які можуть використовуватись декілька разів;

`src/constants` – для збереження постійних змінних для використання у проєкті;

`src/helpers` – для збереження додаткової логіки (корзина, валідація);

`src/pages` – для збереження компонентів сторінок сайту;

`src/static` – для збереження статичних медіа файлів;

`src/translations` – зберігає файли з перекладами.

Код структури головної сторінки (рис. 3.6) реалізованої за допомогою EJS наведений нижче:

```

<>

    <FirstScreen
        itemCount={itemsInBasket.length}
        fastOrder={fastOrder}
                                showBasketModal={() =>
setShowBasketModal(true)}
        scrollToItems={scrollToItems}
    />
    <Action twoBottleAction={fiveBottleAction} />
    <Advantages />
    <Element name="items">
        <Items items={sortBy(Object.values(items),
i => i.order)} addItemToBasket={addItemToBasket}/>
    </Element>
    <ThreeSteps />
    <Reviews />
    <Footer />
        <FeedbackModal show={showFeedbackModal}
onHide={() => setShowFeedbackModal(false)}/>
        <MainFormModal show={showMainFormModal}
sendOrder={sendOrder}
                                onHide={() =>
setShowMainFormModal(false)} />
        <BasketModal itemsInBasket={itemsInBasket}
changeCounter={changeCounter} submitBasket={submitBasket}
                                show={showBasketModal} onHide={()
=> setShowBasketModal(false)} />

```

```

<Snackbar
    anchorOrigin={{ vertical: 'top',
horizontal: 'center' }}
    autoHideDuration={3000}
    open={!showSuccessModal}
    onClose={() => setShowSuccessModal('')}
  >
    <Alert onClose={() =>
setShowSuccessModal('')} severity="success">
      {showSuccessModal}
    </Alert>
  </Snackbar>
</>

```



Рисунок 3.6 – головна сторінка програмного забезпечення

Логіка корзини повністю реалізована на клієнті з використанням localStorage. localStorage об'єкт який є властивістю об'єкту window який дозволяє зберігати дані на локальному сховищі браузера. Код наведений нижче:

```
import {setToLocal} from "./localStorage.js";

const addToBasket = (id) => {
  const itemsString = localStorage.getItem('items');
  if (itemsString) {
    const items= itemsString.split(',');
    if (!items.includes(id)){
      setToLocal({ items: [...items, id].join(',')
});
    }
  } else {
    setToLocal({ items: id });
  }
}

const deleteFromBasket = (id) => {
  const itemsString = localStorage.getItem('items');
  if (itemsString) {
    const items = itemsString.split(',');
    setToLocal({ items: items.filter(itemId => +itemId
!== id).join(',') });
  }
}

const deleteAllFromBasket = (id) => {
  setToLocal({ items: '' });
}
```

```

const getItemsFromBasket = () => {
  const items = localStorage.getItem('items');
  if (items) return items.split(',').map(Number);
  return [];
}

export {
  addToBasket,
  deleteFromBasket,
  getItemsFromBasket,
  deleteAllFromBasket
}

```

Далі наведений код здійснення запитів до сервера з клієнта для роботи з замовленнями:

```

import fetch from '../helpers/fetch';
import urls from './urls/urls';

export async function getOrdersAction() {
  const response = await fetch(urls.orders,
    {
      method: 'GET',
      headers: {
        'Content-Type': 'application/json;
charset=UTF-8',
      },
    })
  return response.json();
}

```

```
export async function saveOrderAction(order) {
  if (!order) throw new Error('No order!');

  const response = await fetch(urls.orders,
    {
      method: 'POST',
      headers: {
        'Content-Type': 'application/json;
charset=UTF-8',
      },
      body: JSON.stringify(order),
    })
  return response.json();
}

export async function deleteOrderAction(id) {
  if (!id) throw new Error('No id!');

  const response = await fetch(urls.orders + id,
    {
      method: 'DELETE',
      headers: {
        'Content-Type': 'application/json;
charset=UTF-8',
      },
    })
  return response.json();
}
```

Запити виконані за допомогою Fetch API. API Fetch дозволяє JavaScript спілкуватися із сервером за допомогою HTTP-запитів і є кращою заміною класу XMLHttpRequest. Запити виконуються методом `fetch()`, який повертає Promise.

Також на клієнті була реалізована валідація форм, код наведений нижче:

```
module.exports = {
  required: (val) => val && (val.toString().trim()),
  number: (val) => !!Number(val),
  phone: (val) => val ?
!(val.replace(/[\d]/g, '').length < 12) : false,
  image: (file) => file && file.type && (file.type ===
'image/jpeg' || file.type === 'image/png'),
  validateAll: (vals, fields) => {
    let allRight = true
    Object.keys(fields).forEach(field => {
      const validate = module.exports[fields[field]]
      if (!validate(vals[field])) allRight = false
    });
    return allRight
  }
}
```

В кодї наведений клас з реалізацією методів валідації, серед яких:

- `required` – для валідацію на наявність будь яких даних в полі, тобто поле має бути заповненим обов'язково;
- `number` – перевірка чи значення є числом;
- `phone` – перевірка чи значення являє собою номер телефону;
- `image` – перевірка завантаженого файлу на відповідність формату картинки.

3.7 Тестування програмного забезпечення для доставки води по місту

Одразу після переходу користувача по посиланню на сайт відкривається головна сторінка яка складається з блоку заголовку, в якому відображенні кнопки для здійснення замовлення, корзина, та елементи з логотипом власника. Якщо спуститись трохи нижче, можна побачити каталог, який представлений у вигляді «каруселі» (рис. 3.7).

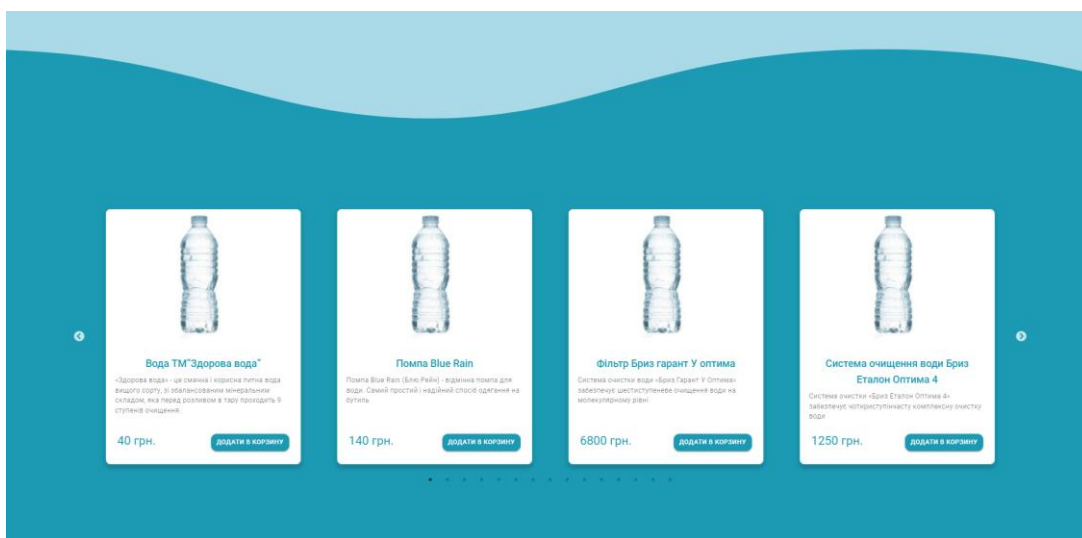


Рисунок 3.7 – Каталог товарів

Обравши необхідний товар, користувач може натиснути кнопку з написом “Додати товар”. Після натискання, має з’явитись спливаюче оповіщення про успішність додавання. Тестування додавання товару у корзину зображено на рисунку 3.8.

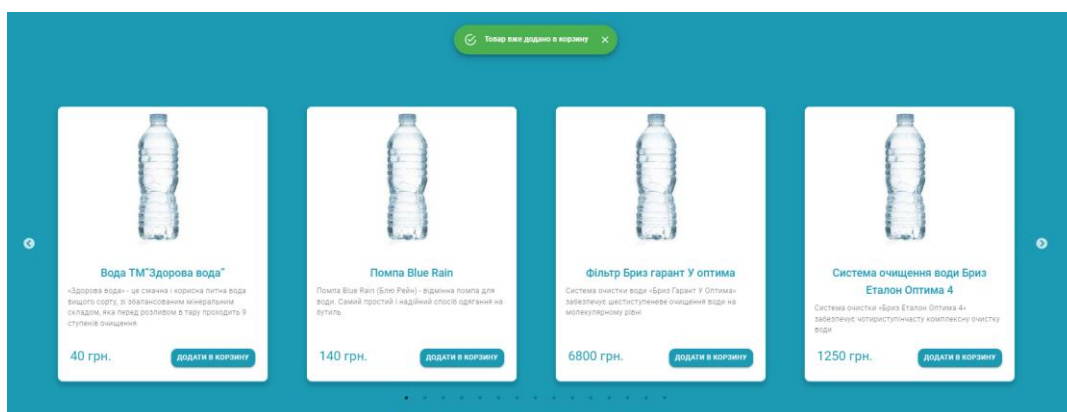


Рисунок 3.8 – Тестування додавання товару в корзину

Після того як товар бу доданий, біля значка корзини з'явиться помітка з кількістю доданих товарів. Користувач може перейти до корзини натиснувши на її іконку, після чого відкриється модальне вікно з списком обраних товарів, з можливістю додавання або зменшення одиниць певної продукції, або взагалі видалити її. В нижній лівій частині діалогового вікна відображається загальна сума замовлення, а у нижній правій – Кнопка для переходу до заповнення решти інформації яка для цього необхідна. Тестування кошика зображено на рисунку 3.9.

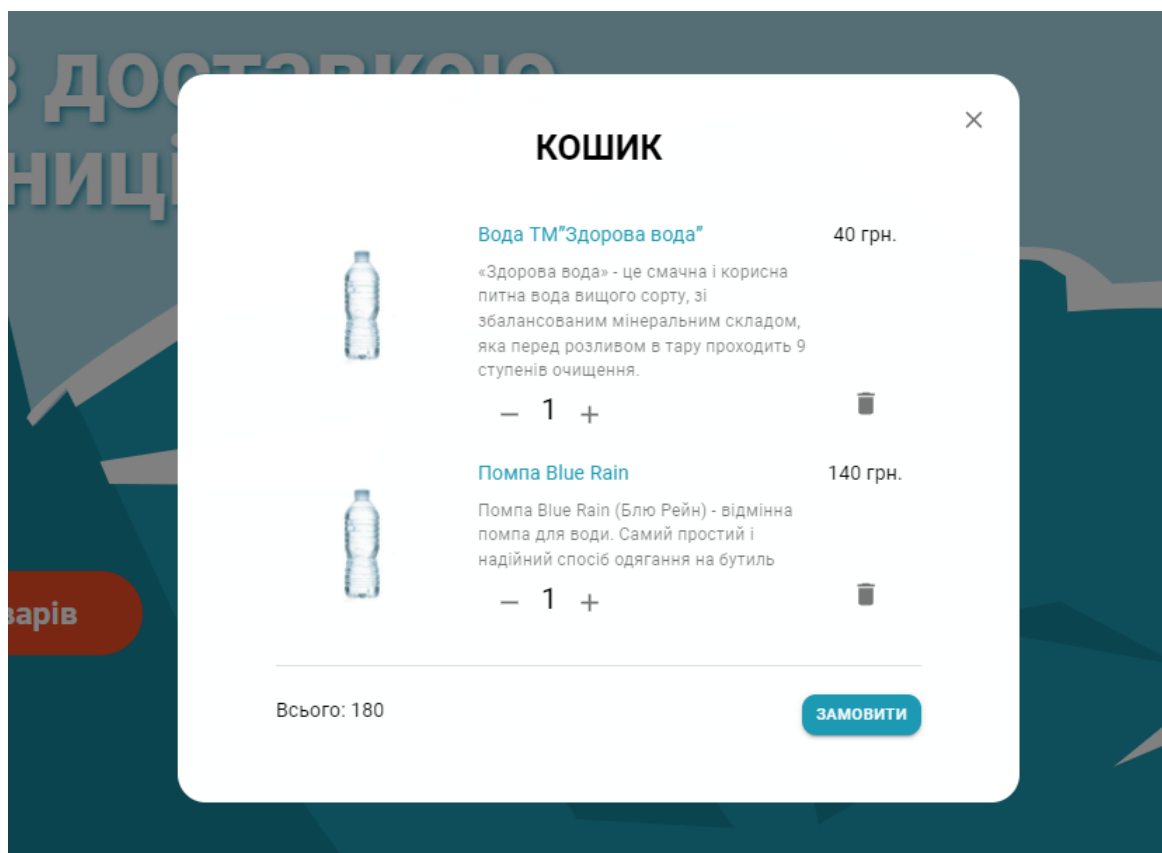


Рисунок 3.9 – Тестування кошика товарів

Коли користувач натискає кнопку замовити повинне відкритись вікно з формою замовлення (рис. 3.10).

ОФОРМЛЕННЯ ЗАМОВЛЕННЯ

Ім'я та прізвище: _____ Телефон: _____

07.06.2022 Час доставки: _____
dd.mm.y

Район: _____

Адреса доставки / вулиця: _____ № буд. _____ № кв. _____

Коментар: _____

ЗАМОВИТИ

Рисунок 3.10 – Тестування форми замовлення

Якщо незаповнені важливі поля, і натиснути кнопку з написом “Замовити” спрацює валідація, і відповідні поля вводу підсвітяться червоним кольором. Тестування валідації форми замовлення зображено на рисунку 3.11.

ОФОРМЛЕННЯ ЗАМОВЛЕННЯ

Ім'я та прізвище: _____ Телефон: _____
Тільки літери Неповний номер

06.06.2022 Час доставки: _____
dd.mm.y Обов'язкове поле

Район: _____
Обов'язкове поле

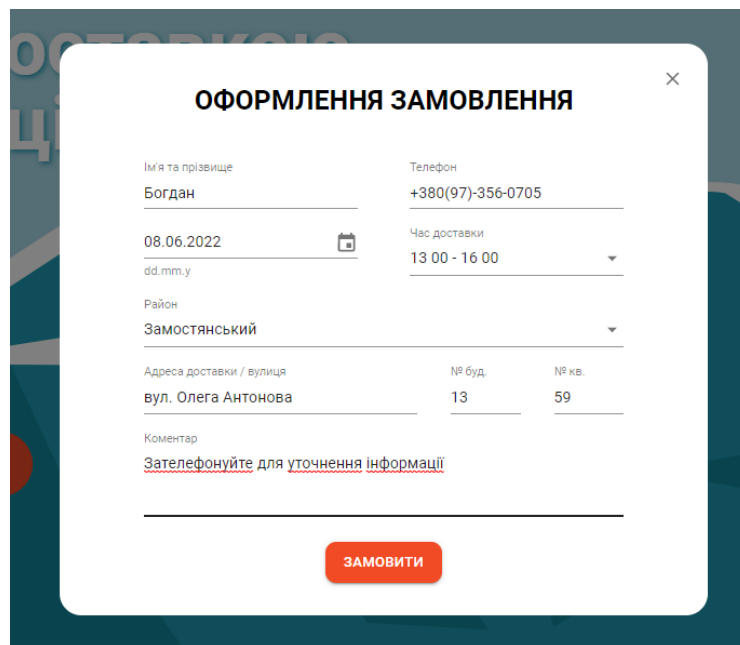
Адреса доставки / вулиця: _____ № буд. _____ № кв. _____
Обов'язкове поле Обов'язкове поле

Коментар: _____

ЗАМОВИТИ

Рисунок 3.11 – Тестування валідації форми замовлення

Якщо заповнити всі необхідні поля слідуючи інструкціям, замовлення буде успішно відправлене на сервер. Тестування оформлення замовлення зображено на рисунку 3.12.



ОФОРМЛЕННЯ ЗАМОВЛЕННЯ

Ім'я та прізвище: Богдан
Телефон: +380(97)-356-0705

08.06.2022
Час доставки: 13 00 - 16 00

Район: Замостянський

Адреса доставки / вулиця: вул. Олега Антонова
№ буд.: 13
№ кв.: 59

Коментар: [Зателефонуйте для уточнення інформації](#)

ЗАМОВИТИ

Рисунок 3.12 – Тестування оформлення замовлення

У разі успішного опрацювання замовлення сервером запис про нього буде доданий в базу даних, а також власник сайту отримає сповіщення в телеграм. Тестування отримання сповіщення про замовлення в телеграм зображено на рисунку 3.13.

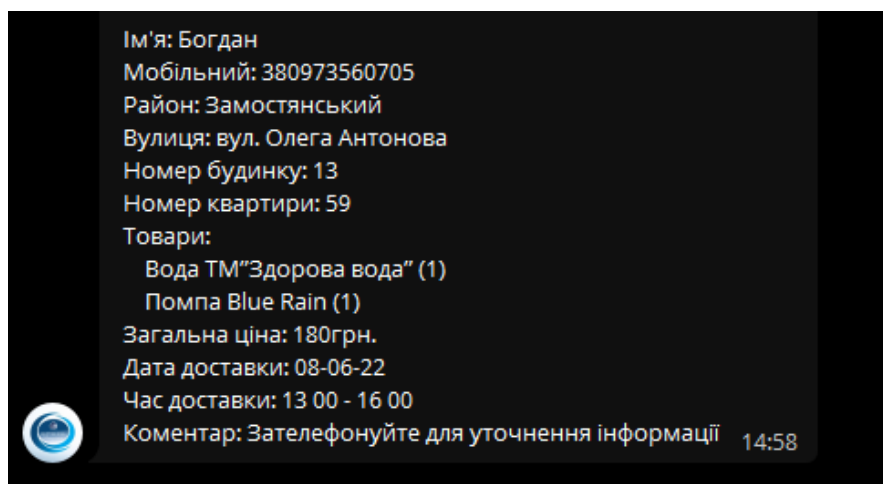


Рисунок 3.13 – Тестування отримання сповіщення про замовлення в телеграм

Ще однією важливою частиною сайту є сторінка яка доступна для перегляду лише для власника сайту. Для цього необхідно додати “/admin” до адресу сайту, після чого відкриється сторінка авторизації. Сторінка авторизації зображена на рисунку 3.14.

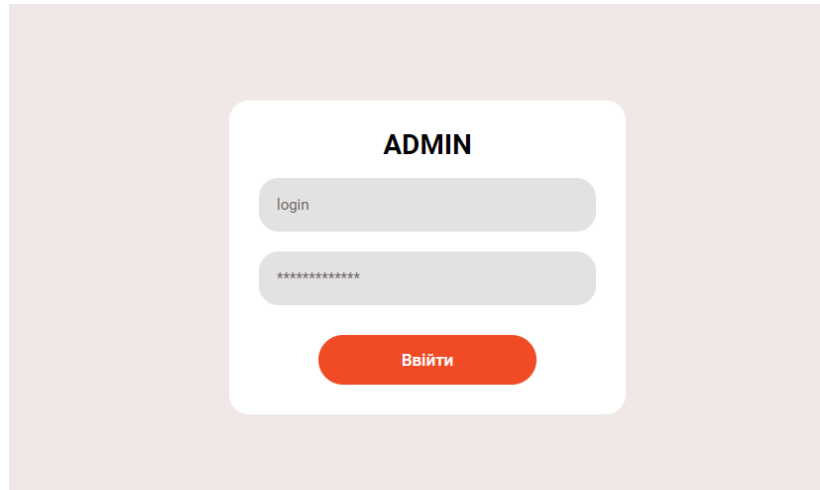


Рисунок 3.14 – Сторінка авторизації

Після авторизації власник отримає доступ до сторінок з списком замовлень, клієнтів та товарів (рис. 3.15 та 3.16).

ADMIN PANEL												
VODA												
Замовлення												
Actions	Ім'я	Телефон	Район	Адреса	№ дому	№ кв.	Товари	Ціна	Дата	Час	Коментар	Створено
	Мерзон Ірина	380731234692	Івдій	Лепі Ратушної	16	3	Вода ТМ"Здорова вода" (2),Ломпа Blue Rain (1)	220	14/03/2022	13 00 - 16 00	1 під'їзд	14/03/2022
	Лелюк Юлія	380930740958	Замостянський	Олега Антонова	20	1	Вода ТМ"Здорова вода" (3)	120	28/03/2022	16 00 - 20 00		27/03/2022
	Богдан	380973560705	Замостянський	вулиця Олега Антонова	13	59	Вода ТМ"Здорова вода" (3)	120	30/05/2022	13 00 - 16 00	Комент	29/05/2022
	мєвєєєєєє	380343434343	Староміський	улиця	34	43	Фільтр Брїч гарант У оптима (1),Ломпа Blue Rain (1)	6940	30/05/2022	13 00 - 16 00		29/05/2022

Рисунок 3.15 – Сторінка зі списком замовлень

Actions	Фото	Ім'я	Пріоритет	Ціна	Опис	Деталі	Створено
		Вода ТМ "Здорова вода"	1	40	«Здорова вода» - це смачна і корисна питна вода вищого сорту, зі збалансованим мінеральним складом, яка перед розливом в тару проходить 9 ступенів очищення.		30.05.2021
		Помпа Blue Rain	2	140	Помпа Blue Rain (Блю Рейн) - відмінна помпа для води. Самий простий і надійний спосіб одягання на бутиль.		15.05.2021
		Фільтр Бриз гарант У оптима	3	6800	Система очистки води «Бриз Гарант У Оптима» забезпечує шестиступеневе очищення води на молекулярному рівні.		15.05.2021
		Система очищення води Бриз Еталон Оптима 4	4	1250	Система очистки «Бриз Еталон Оптима 4» забезпечує чотириступінчасту комплексну очистку води.		15.05.2021
		Купити в мережі VIVO V13,5L 6L 6LW	5	9800	Характеристики VIVO X12-FC BLACK, кулер для води компресорний, підлоговий, вид настільний. Тип установки - Верхнє		14.06.2021

Рисунок 3.16 – Сторінка зі списком товарів

Власник може переглядати, редагувати, видаляти, зберігати звіти локально на свій пристрій. Під час редагування є можливість завантаження зображення з пристрою. Редагування товару зображено на рисунку 3.17.

Actions	Фото	Ім'я	Пріоритет	Ціна	Опис	Деталі	Створено
<input checked="" type="checkbox"/>	<input type="checkbox"/>	Вода ТМ "Здорова вода"	1	40	«Здорова вода» - це смачна і корисна питна вода вищого		Дата ставиться автоматично
		Помпа Blue Rain	2	140	Помпа Blue Rain (Блю Рейн) - відмінна помпа для води. Самий простий і надійний спосіб одягання на бутиль.		15/05/2021
		Фільтр Бриз гарант У оптима	3	6800	Система очистки води «Бриз Гарант У Оптима» забезпечує шестиступеневе очищення води на молекулярному рівні.		15/05/2021
		Система очищення води Бриз Еталон Оптима 4	4	1250	Система очистки «Бриз Еталон Оптима 4» забезпечує чотириступінчасту комплексну очистку води.		15/05/2021

Рисунок 3.17 – Тестування редагування товару

В формі є можливість завантаження зображення з локальної файлової системи. Після відправки форми файл буде збережений на сервері.

Проведемо порівняння функціональних можливостей програм для доставки води по місту (табл. 3.1)

Таблиця 3.1 – Аналіз функціональних можливостей програм для доставки води по місту

	Авторизація клієнта	Інформація про ціни на доставку води	Он-лайн замовлення	Сповіщення в месенджери	Зворотній зв'язок через месенджери
MyWaterShop	+	+	+	-	-
ВОДО ОФІС СЕРВІС	-	+	+	-	-
vinnitska.com	+	+	+	-	-
Розроблене програмне забезпечення	+	+	+	+	+

З таблиці 3.1 видно, що розроблений програмний продукт володіє новими функціональними можливостями: Сповіщення про статус замовлення в месенджери Telegram/Viber; Зворотній зв'язок через месенджер Telegram/Viber до веб-ресурсу.

3.8 Висновок до розділу 3

У даному розділі було здійснено програмну реалізацію інформаційної технології доставки води по місту у вигляді веб-ресурсу. Розроблено серверну та клієнтську частини веб-ресурсу для доставки води по місту, а також його дизайн та розширений функціонал.

Також в даному розділі було здійснено тестування веб-ресурсу для доставки води по місту, в результаті чого суттєвих проблем та неполадок не виявлено, тому можна зробити висновок, що всі функції працюють відповідно до їх функціоналу. розроблений програмний продукт володіє новими функціональними можливостями: Було здійснено розширення функціоналу додаванням двох функцій: Сповіщення про статус замовлення в месенджери Telegram/Viber; Зворотній зв'язок через месенджер Telegram/Viber до веб-ресурсу.

4 ЕКОНОМІЧНА ЧАСТИНА

Науково-технічна розробка має право на існування та впровадження, якщо вона відповідає вимогам часу, як в напрямку науково-технічного прогресу та і в плані економіки. Тому для науково-дослідної роботи необхідно оцінювати економічну ефективність результатів виконаної роботи.

Магістерська кваліфікаційна робота за темою «Інформаційна технологія доставки води по місту» відноситься до науково-технічних робіт, які плануються для використання безпосередньо самим розробником (замовником), тобто її результатами буде користуватися тільки розробник (або замовник). У цьому випадку нам потрібно довести ефективність інвестицій, вкладених у цей проект самим розробником (замовником).

Для цього випадку необхідно виконати такі етапи робіт:

- 1) провести технологічний аудит власної науково-технічної розробки, тобто встановити її науково-технічний рівень;
- 2) розрахувати витрати на здійснення науково-технічної розробки;
- 3) провести розрахунок економічної ефективності науково-технічної розробки у випадку її впровадження розробником (замовником) на власному підприємстві та обґрунтувати економічну доцільність впровадження розробником (замовником) розробленого науково-технічного проекту.

4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту дослідження за темою «Інформаційна технологія доставки води по місту» є оцінювання науково-технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням 5-ти бальної системи оцінювання за 12-ма критеріями, наведеними в табл. 4.1 [19].

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
Технічна здійсненність концепції					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено працездатність продукту в реальних умовах
Ринкові переваги (недоліки)					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в	Технічні та споживчі властивості продукту трохи гірші, ніж в	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в	Технічні та споживчі властивості продукту значно кращі, ніж в
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає

Продовження таблиці 4.1

Практична здійсненість					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки потрібно звести до таблиці 4.2.

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

Критерії	Експерт (ПІБ, посада)		
	1	2	3
	Бали:		
1. Технічна здійсненність концепції	4	5	4
2. Ринкові переваги (наявність аналогів)	3	3	3
3. Ринкові переваги (ціна продукту)	4	4	3
4. Ринкові переваги (технічні властивості)	3	3	4
5. Ринкові переваги (експлуатаційні витрати)	2	2	3
6. Ринкові перспективи (розмір ринку)	3	3	3
7. Ринкові перспективи (конкуренція)	2	2	2
8. Практична здійсненність (наявність фахівців)	5	5	5
9. Практична здійсненність (наявність фінансів)	2	3	2
10. Практична здійсненність (необхідність нових матеріалів)	4	5	5
11. Практична здійсненність (термін реалізації)	3	4	5
12. Практична здійсненність (розробка документів)	4	5	4
Сума балів	39	44	43
Середньоарифметична сума балів $СБ_c$	42,0		

За результатами розрахунків, наведених в таблиці 4.2, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в таблиці 4.3 [19].

Таблиця 4.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів $СБ_c$, розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41...48	Високий
31...40	Вище середнього
21...30	Середній
11...20	Нижче середнього
0...10	Низький

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Інформаційна технологія доставки води по місту» становить 42,0 бала, що, відповідно до таблиці 4.3, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки високий).

4.2 Розрахунок узагальненого коефіцієнта якості розробки

Окрім комерційного аудиту розробки доцільно також розглянути технічний рівень якості розробки, розглянувши її основні технічні показники. Ці показники по-різному впливають на загальну якість проектної розробки.

Узагальнений коефіцієнт якості (B_n) для нового технічного рішення розрахуємо за формулою [20]:

$$B_n = \sum_{i=1}^k \alpha_i \cdot \beta_i, \quad (4.1)$$

де k – кількість найбільш важливих технічних показників, які впливають на якість нового технічного рішення;

α_i – коефіцієнт, який враховує питому вагу i -го технічного показника в загальній якості розробки. Коефіцієнт α_i визначається експертним шляхом і при

цьому має виконуватись умова $\sum_{i=1}^k \alpha_i = 1$;

β_i – відносне значення i -го технічного показника якості нової розробки.

Відносні значення β_i для різних випадків розраховуємо за такими формулами:

- для показників, зростання яких вказує на підвищення в лінійній залежності якості нової розробки:

$$\beta_i = \frac{I_{ni}}{I_{ai}}, \quad (4.2)$$

де I_{ni} та I_{ai} – чисельні значення конкретного i -го технічного показника якості відповідно для нової розробки та аналога;

- для показників, зростання яких вказує на погіршення в лінійній залежності якості нової розробки:

$$\beta_i = \frac{I_{ai}}{I_{ni}}; \quad (4.3)$$

Використовуючи наведені залежності можемо проаналізувати та порівняти техніко-економічні характеристики аналогу та розробки на основі отриманих наявних та проектних показників, а результати порівняння зведемо до таблиці 4.4.

Таблиця 4.4 – Порівняння основних параметрів розробки та аналога.

Показники (параметри)	Одиниця вимірювання	Аналог	Проектований пристрій	Відношення параметрів нової розробки до аналога	Питома вага показника
Процесор (тактова частота)	ГГц	3,5	1,5	2,33	0,25
Оперативна пам'ять	Мб	16	4	4	0,3
Об'єм диску	Гб	100	40	2,5	0,15
Мережевий канал	\	1	2	2	0,05
Швидкість мережевого каналу	Мб/с	100	50	2	0,25

Узагальнений коефіцієнт якості (B_n) для нового технічного рішення складе:

$$B_n = \sum_{i=1}^k \alpha_i \cdot \beta_i = 2,33 \cdot 0,25 + 4 \cdot 0,3 + 2,5 \cdot 0,15 + 2 \cdot 0,05 + 2 \cdot 0,25 = 2,76.$$

Отже за технічними параметрами, згідно узагальненого коефіцієнту якості розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 2,76 рази.

4.3 Розрахунок витрат на проведення науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної роботи на тему «Інформаційна технологія доставки води по місту», під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуємо за відповідними статтями.

До статті «Витрати на оплату праці» належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці.

Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників (Z_o) розраховуємо у відповідності до посадових окладів працівників, за формулою [19]:

$$Z_o = \sum_{i=1}^k \frac{M_{ni} \cdot t_i}{T_p}, \quad (4.4)$$

де k – кількість посад дослідників залучених до процесу досліджень;

M_{ni} – місячний посадовий оклад конкретного дослідника, грн;

t_i – число днів роботи конкретного дослідника, дн.;

T_p – середнє число робочих днів в місяці, $T_p=21$ дні.

$$Z_o = 17120,00 \cdot 15 / 21 = 12228,57 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці 4.5.

Таблиця 4.5 – Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Число днів роботи	Витрати на заробітну плату, грн
Керівник проекту	17120,00	815,24	15	12228,57
Інженер-дослідник (програміст)	16850,00	802,38	15	12035,71
Консультант (менеджер служби доставки)	17000,00	809,52	4	3238,10
Фахівець 1-ї категорії	8500,00	404,76	15	6071,43
Всього				33573,81

Основна заробітна плата робітників

Витрати на основну заробітну плату робітників (Z_p) за відповідними найменуваннями робіт НДР на тему «Інформаційна технологія доставки води по місту» розраховуємо за формулою:

$$Z_p = \sum_{i=1}^n C_i \cdot t_i, \quad (4.5)$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

t_i – час роботи робітника при виконанні визначеної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C_i можна визначити за формулою:

$$C_i = \frac{M_M \cdot K_i \cdot K_c}{T_p \cdot t_{зм}}, \quad (4.6)$$

де M_M – розмір прожиткового мінімуму працездатної особи, або мінімальної місячної заробітної плати (в залежності від діючого законодавства), прийmemo $M_M=6700,00$ грн;

K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду (табл. Б.2, додаток Б) [19];

K_c – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

T_p – середнє число робочих днів в місяці, приблизно $T_p = 21$ дн;

$t_{зм}$ – тривалість зміни, год.

$$C_1 = 6700,00 \cdot 1,10 \cdot 1,35 / (21 \cdot 8) = 59,22 \text{ грн.}$$

$$З_{pl} = 59,22 \cdot 4,32 = 255,84 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці 4.6.

Таблиця 4.6 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
Встановлення обладнання для проведення досліджень	4,32	2	1,10	59,22	255,84
Інсталяція програмного забезпечення розробки програмного забезпечення	6,50	3	1,35	72,68	472,44
Встановлення цифрових обчислювальних систем та мережевого доступу	3,15	4	1,50	80,76	254,39
Відлагодження програмних модулів аналітичного дослідження	5,20	5	1,70	91,53	475,94
Підготовка тестового дослідження наявної бази даних	1,50	3	1,35	72,68	109,02
Формування бази даних результатів випробування	8,00	3	1,35	72,68	581,46
Всього					2149,10

Додаткова заробітна плата дослідників та робітників

Додаткову заробітну плату розраховуємо як 10 ... 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$Z_{\text{доо}} = (Z_o + Z_p) \cdot \frac{H_{\text{доо}}}{100\%}, \quad (4.7)$$

де $H_{\text{доо}}$ – норма нарахування додаткової заробітної плати. Прийmemo 10%.

$$Z_{\text{доо}} = (33573,81 + 2149,10) \cdot 10 / 100\% = 3572,29 \text{ грн.}$$

Нарахування на заробітну плату дослідників та робітників розраховуємо як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$Z_n = (Z_o + Z_p + Z_{\text{доо}}) \cdot \frac{H_{\text{зн}}}{100\%} \quad (4.8)$$

де $H_{\text{зн}}$ – норма нарахування на заробітну плату. Приймаємо 22%.

$$Z_n = (33573,81 + 2149,10 + 3572,29) \cdot 22 / 100\% = 8644,94 \text{ грн.}$$

До статті «Сировина та матеріали» належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби і предмети праці, які придбані у сторонніх підприємств, установ і організацій та витрачені на проведення досліджень за темою «Інформаційна технологія доставки води по місту».

Витрати на матеріали (M), у вартісному вираженні розраховуються окремо по кожному виду матеріалів за формулою:

$$M = \sum_{j=1}^n H_j \cdot C_j \cdot K_j - \sum_{j=1}^n B_j \cdot C_{\epsilon j}, \quad (4.9)$$

де H_j – норма витрат матеріалу j -го найменування, кг;

n – кількість видів матеріалів;

C_j – вартість матеріалу j -го найменування, грн/кг;

K_j – коефіцієнт транспортних витрат, ($K_j = 1,1 \dots 1,15$);

B_j – маса відходів j -го найменування, кг;

C_{vj} – вартість відходів j -го найменування, грн/кг.

$$M_1 = 3,0 \cdot 174,90 \cdot 1,1 - 0 \cdot 0 = 577,17 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці 4.7.

Таблиця 4.7 – Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за 1 кг, грн	Норма витрат, кг	Величина відходів, кг	Ціна відходів, грн/кг	Вартість витраченого матеріалу, грн
Папір канцелярський офісний (A4) BUROMAX Premium	174,90	3,0	0	0	577,17
Папір для заміток (A5) BUROMAX Premium Light	118,00	3,0	0	0	389,40
Начиння канцелярське BUROMAX Premium	187,00	3,0	0	0	617,10
Органайзер офісний	200,00	3,0	0	0	660,00
Картридж для принтера	2098,00	2,0	0	0	4615,60
Диск оптичний	27,00	5,0	0	0	148,50
USB-пам'ять Microtech 64 GB	179,00	2,0	0	0	393,80
Тека для паперів	92,00	3,0	0	0	303,60
Всього					7705,17

Витрати на комплектуючі (K_6), які використовують при проведенні НДР на тему «Інформаційна технологія доставки води по місту», розраховуємо, згідно з їхньою номенклатурою, за формулою:

$$K_6 = \sum_{j=1}^n H_j \cdot C_j \cdot K_j \quad (4.10)$$

де H_j – кількість комплектуючих j -го виду, шт.;

C_j – покупна ціна комплектуючих j -го виду, грн;

K_j – коефіцієнт транспортних витрат, ($K_j = 1,1 \dots 1,15$).

$$K_6 = 1 \cdot 2850,00 \cdot 1,05 = 2992,50 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці 4.8.

Таблиця 4.8 – Витрати на комплектуючі

Найменування комплектуючих	Кількість, шт.	Ціна за штуку, грн	Сума, грн
Роутер ХТ-МВ-0312	1	2850,00	2992,50
Кабель USB для передачі даних	1	356,00	373,80
Всього			3366,30

До статті «Спецустаткування для наукових (експериментальних) робіт» належать витрати на виготовлення та придбання спецустаткування необхідного для проведення досліджень, також витрати на їх проектування, виготовлення, транспортування, монтаж та встановлення.

Балансову вартість спецустаткування розраховуємо за формулою:

$$B_{\text{спец}} = \sum_{i=1}^k C_i \cdot C_{\text{пр.і}} \cdot K_i, \quad (4.11)$$

де C_i – ціна придбання одиниці спецустаткування даного виду, марки, грн;

$C_{\text{пр.і}}$ – кількість одиниць устаткування відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує доставку, монтаж, налагодження устаткування тощо, ($K_i = 1,10 \dots 1,12$);

k – кількість найменувань устаткування.

$$V_{\text{снец}} = 9650,00 \cdot 1 \cdot 1,05 = 10132,50 \text{ грн.}$$

Отримані результати зведемо до таблиці 4.9.

Таблиця 4.9 – Витрати на придбання спецустаткування по кожному виду

Найменування устаткування	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Мережеве обладнання передачі цифрових даних	1	9650,00	10132,50
Всього			10132,50

До статті «Програмне забезпечення для наукових (експериментальних) робіт» належать витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення, (програм, алгоритмів, баз даних) необхідних для проведення досліджень, також витрати на їх проектування, формування та встановлення.

Балансову вартість програмного забезпечення розраховуємо за формулою:

$$V_{\text{прг}} = \sum_{i=1}^k C_{\text{инрг}} \cdot C_{\text{прг.}i} \cdot K_i, \quad (4.12)$$

де $C_{\text{инрг}}$ – ціна придбання одиниці програмного засобу даного виду, грн;

$C_{\text{прг.}i}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i = 1,10 \dots 1,12$);

k – кількість найменувань програмних засобів.

$$V_{\text{прг}} = 5890,00 \cdot 1 \cdot 1,05 = 6184,50 \text{ грн.}$$

Отримані результати зведемо до таблиці 4.10.

Таблиця 4.10 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Середовище програмування Visual Studio Code	1	5890,00	6184,50
Платформа Docker	1	8560,00	8988,00
Всього			15172,50

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_{б}}{T_{г}} \cdot \frac{t_{вик}}{12}, \quad (4.13)$$

де $Ц_{б}$ – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$ – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_{г}$ – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$$A_{обл} = (27599,00 \cdot 1) / (3 \cdot 12) = 766,64 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці 4.11.

Таблиця 4.11 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Персональний комп'ютер Core I3-2700	27599,00	3	1	766,64
Робоче місце інженера-розробника (дослідника програмного забезпечення)	6560,00	5	1	109,33
Графічні пристрої виводу інформації	7500,00	4	1	156,25
Офісна оргтехніка	8329,00	4	1	173,52
Приміщення лабораторії розробки та дослідження	405000,00	30	1	1125,00
ОС Windows 11	5750,00	3	1	159,72
Прикладний пакет Microsoft Office 2021 Professional Plus	5125,00	3	1	142,36
Всього				2632,83

Витрати на силову електроенергію (B_e) розраховуємо за формулою:

$$B_e = \sum_{i=1}^n \frac{W_{yi} \cdot t_i \cdot C_e \cdot K_{eni}}{\eta_i}, \quad (4.14)$$

де W_{yi} – встановлена потужність обладнання на визначеному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

C_e – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії), прийmemo $C_e = 7,50$ грн;

K_{eni} – коефіцієнт, що враховує використання потужності, $K_{eni} < 1$;

η_i – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

$$B_e = 0,24 \cdot 240,0 \cdot 7,50 \cdot 0,95 / 0,97 = 432,00 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці 4.12.

Таблиця 4.12 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
Персональний комп'ютер Core I3-2700	0,24	240,0	432,00
Робоче місце інженера-розробника (дослідника програмного забезпечення)	0,08	240,0	144,00
Графічні пристрої виводу інформації	0,12	2,0	1,80
Офісна оргтехніка	0,32	2,0	4,80
Всього			582,60

До статті «Службові відрядження» дослідної роботи на тему «Інформаційна технологія доставки води по місту» належать витрати на відрядження штатних працівників, працівників організацій, які працюють за договорами цивільно-правового характеру, аспірантів, зайнятих розробленням досліджень, відрядження, пов'язані з проведенням випробувань машин та приладів, а також витрати на відрядження на наукові з'їзди, конференції, наради, пов'язані з виконанням конкретних досліджень.

Витрати за статтею «Службові відрядження» відсутні.

Витрати за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації» відсутні.

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуємо як 50...100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_{\text{в}} = (Z_o + Z_p) \cdot \frac{H_{\text{ив}}}{100\%}, \quad (4.15)$$

де $H_{\text{ив}}$ – норма нарахування за статтею «Інші витрати», прийmemo $H_{\text{ив}} = 50\%$.

$$I_{\text{в}} = (33573,81 + 2149,10) \cdot 50 / 100\% = 17861,46 \text{ грн.}$$

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін.

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуємо як 100...150% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{\text{нзв}} = (Z_o + Z_p) \cdot \frac{H_{\text{нзв}}}{100\%}, \quad (4.16)$$

де $H_{\text{нзв}}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати», прийmemo $H_{\text{нзв}} = 100\%$.

$$B_{\text{нзв}} = (33573,81 + 2149,10) \cdot 100 / 100\% = 35722,91 \text{ грн.}$$

Витрати на проведення науково-дослідної роботи на тему «Інформаційна технологія доставки води по місту» розраховуємо як суму всіх попередніх статей витрат за формулою:

$$B_{\text{заг}} = Z_o + Z_p + Z_{\text{дод}} + Z_{\text{н}} + M + K_{\text{в}} + B_{\text{спец}} + B_{\text{прз}} + A_{\text{обл}} + B_{\text{е}} + B_{\text{св}} + B_{\text{сп}} + I_{\text{в}} + B_{\text{нзв}}. \quad (4.17)$$

$$B_{\text{заг}} = 33573,81 + 2149,10 + 3572,29 + 8644,94 + 7705,17 + 3366,30 + 10132,50 + 15172,50 + 2632,83 + 582,60 + 0,00 + 0,00 + 17861,46 + 35722,91 = 141116,41 \text{ грн.}$$

Загальні витрати ZB на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховується за формулою:

$$ZB = \frac{B_{\text{заг}}}{\eta}, \quad (4.18)$$

де η - коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, прийmemo $\eta = 0,95$.

$$ZB = 141116,41 / 0,95 = 148543,59 \text{ грн.}$$

4.4 Розрахунок економічної ефективності науково-технічної розробки від її впровадження безпосередньо розробником (замовником)

При виконанні даної роботи за темою «Інформаційна технологія доставки води по місту» розглядається ситуація, коли замовник певної науково-технічної розробки використовує її тільки на своєму підприємстві (чи в організації) і не виводить її на ринок. У цьому випадку позитивним результатом від впровадження цієї науково-технічної розробки може бути покращення певних економічних та фінансових показників діяльності підприємства.

Для визначення величини майбутнього економічного ефекту та ефективності розробки визначимо певні характеристики підприємства (табл. 4.13).

Таблиця 4.13 – Вихідні дані замовника

Показник	Рік розробки	1-й рік	2-й рік	3-й рік	4-й рік
прогнозоване зростання кількості виготовленої продукції (наданих послуг) завдяки використанню нової чи модернізованої технології, застосування спеціальних технічних чи програмних засобів у році що аналізується, шт	-	10000	15000	11000	7500
кількість випущеної продукції (наданих послуг) у році до впровадження результатів нової науково-технічної розробки, шт	785000	-	-	-	-
зміна ціни реалізації продукції (зростання чи зниження) від впровадження результатів науково-технічної розробки у періоди часу, що аналізуються, грн	-	0,30	0,30	0,30	0,30
зміна собівартості виготовлення продукції (зростання чи зниження) від впровадження результатів науково-технічної розробки у періоди часу, що аналізуються, грн	-	-0,30	-0,30	-0,30	-0,30
ціна реалізації продукції за існуючими підходами у році до впровадження результатів нової науково-технічної розробки, грн	1,50	-	-	-	-
собівартість виготовлення продукції за існуючими підходами у році до впровадження результатів нової науково-технічної розробки, грн	0,50	-	-	-	-
загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн	148543,59	-	-	-	-

В цьому випадку основу майбутнього економічного ефекту будуть формувати такі показники: ΔN – прогнозоване зростання кількості виготовленої продукції (наданих послуг) завдяки використанню нової чи модернізованої технології, застосування спеціальних технічних чи програмних засобів у році що аналізується, шт; N – кількість випущеної продукції (наданих послуг) у році до впровадження

результатів нової науково-технічної розробки, 785000 шт; $\Delta\Pi_{я}$ – підвищення прибутку підприємства в результаті зменшення собівартості одиниці продукції, що виготовляється на підприємстві (в розрахунку на одиницю продукції) у році, що аналізується, грн. Причому підвищення прибутку підприємства можна представити як: $\Delta\Pi_{я} = \pm\Delta C - (\pm\Delta S)$, де $\pm\Delta C$ – зміна ціни реалізації продукції (зростання чи зниження) від впровадження результатів науково-технічної розробки у періоди часу, що аналізуються, грн; $\pm\Delta S$ – зміна собівартості виготовлення продукції (зростання чи зниження) від впровадження результатів науково-технічної розробки у періоди часу, що аналізуються, грн;

Підвищення прибутку підприємства в результаті зменшення собівартості одиниці продукції в 1-й рік

$$\Delta\Pi_{я} = 0,30 - (-0,30) = 0,60 \text{ грн.}$$

Підвищення прибутку підприємства в результаті зменшення собівартості одиниці продукції в 2-й рік

$$\Delta\Pi_{я} = 0,30 - (-0,30) = 0,60 \text{ грн.}$$

Підвищення прибутку підприємства в результаті зменшення собівартості одиниці продукції в 3-й рік

$$\Delta\Pi_{я} = 0,30 - (-0,30) = 0,60 \text{ грн.}$$

Підвищення прибутку підприємства в результаті зменшення собівартості одиниці продукції в 4-й рік

$$\Delta\Pi_{я} = 0,30 - (-0,30) = 0,60 \text{ грн.}$$

$\Pi_{я}$ – величина чистого прибутку, що його отримує підприємство від виробництва та реалізації одиниці продукції, грн. Причому приблизно можна прийняти $\Pi_{я} = C - S$, де C – ціна реалізації продукції за існуючими підходами у році до впровадження результатів нової науково-технічної розробки, 1,50 грн; S – собівартість виготовлення продукції за існуючими підходами у році до впровадження результатів нової науково-технічної розробки, 0,50 грн.

$$\Pi_{я} = 1,50 - 0,50 = 1,00 \text{ грн.}$$

Збільшення чистого прибутку підприємства $\Delta\Pi_i$ для кожного із років, протягом яких очікується отримання позитивних результатів від можливого впровадження науково-технічної розробки, розраховуємо за формулою:

$$\Delta\Pi_i = (\Delta\Pi_{\text{я}} \cdot N + \Pi_{\text{я}} \cdot \Delta N)_i, \quad (4.19)$$

де $\Delta\Pi_{\text{я}}$ – покращення основного якісного показника від впровадження на підприємстві результатів науково-технічної розробки у році що аналізується;

N – основний кількісний показник, який визначає обсяг діяльності підприємства у році до впровадження результатів нової науково-технічної розробки;

$\Pi_{\text{я}}$ – основний якісний показник, який визначає результати діяльності підприємства у кожному із років після впровадження науково-технічної розробки;

ΔN – зміна основного кількісного показника діяльності підприємства в результаті впровадження науково-технічної розробки у році що аналізується.

Збільшення чистого прибутку підприємства в 1-й рік впровадження

$$\Delta\Pi_i = 0,60 \cdot 785000 + 1,00 \cdot 10000 = 481000,00 \text{ грн.}$$

Збільшення чистого прибутку підприємства в 2-й рік впровадження

$$\Delta\Pi_i = 0,60 \cdot 785000 + 1,00 \cdot (10000 + 15000) = 496000,00 \text{ грн.}$$

Збільшення чистого прибутку підприємства в 3-й рік впровадження

$$\Delta\Pi_i = 0,60 \cdot 785000 + 1,00 \cdot (10000 + 15000 + 11000) = 507000,00 \text{ грн.}$$

Збільшення чистого прибутку підприємства в 4-й рік впровадження

$$\Delta\Pi_i = 0,60 \cdot 785000 + 1,00 \cdot (10000 + 15000 + 11000 + 7500) = 514500,00 \text{ грн.}$$

Приведена вартість збільшення всіх чистих прибутків $ПП$, що їх може отримати розробник від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_{i=1}^T \frac{\Delta\Pi_i}{(1 + \tau)^t}, \quad (4.20)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

T – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau = 0,1$;

t – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

$$\begin{aligned} \text{ПП} &= 481000,00/(1+0,1)^1 + 496000,00/(1+0,1)^2 + 507000,00/(1+0,1)^3 + \\ &+ 514500,00/(1+0,1)^4 = 437272,73 + 409917,36 + 380916,60 + 351410,42 = 1579517,11 \text{ грн.} \end{aligned}$$

Величина початкових інвестицій PV , які розробник має вкласти для впровадження і комерціалізації науково-технічної розробки:

$$PV = k_{\text{инв}} \cdot ZB, \quad (4.21)$$

де $k_{\text{инв}}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо $k_{\text{инв}} = 2$;

ZB – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 148543,59 грн.

$$PV = k_{\text{инв}} \cdot ZB = 2 \cdot 148543,59 = 297087,19 \text{ грн.}$$

Абсолютний економічний ефект $E_{\text{абс}}$ для розробника від можливого впровадження науково-технічної розробки становитиме:

$$E_{\text{абс}} = \text{ПП} - PV \quad (4.22)$$

де ПП – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 1579517,11 грн;

PV – теперішня вартість початкових інвестицій, 297087,19 грн.

$$E_{abc} = III - PV = 1579517,11 - 297087,19 = 1282429,92 \text{ грн.}$$

Внутрішня економічна дохідність інвестицій E_e , які можуть бути вкладені розробником у впровадження та комерціалізацію науково-технічної розробки:

$$E_e = T_{ж} \sqrt[1 + \frac{E_{abc}}{PV}] - 1, \quad (4.23)$$

де E_{abc} – абсолютний економічний ефект вкладених інвестицій, 1282429,92 грн;

PV – теперішня вартість початкових інвестицій, 297087,19 грн;

$T_{ж}$ – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримування позитивних результатів від її впровадження, 4 роки.

$$E_e = T_{ж} \sqrt[1 + \frac{E_{abc}}{PV}] - 1 = (1 + 1282429,92/297087,19)^{1/4} = 0,518.$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій τ_{min} :

$$\tau_{min} = d + f, \quad (4.24)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2023 році в Україні $d = 0,1$;

f – показник, що характеризує ризикованість вкладення інвестицій, прийmemo 0,1.

$\tau_{min} = 0,1 + 0,1 = 0,2 < 0,518$ свідчить про те, що внутрішня економічна дохідність інвестицій E_e , які можуть бути вкладені розробником у впровадження та комерціалізацію науково-технічної розробки вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою «Інформаційна технологія доставки води по місту» доцільно.

Період окупності інвестицій $T_{ок}$ які можуть бути вкладені розробником у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_г}, \quad (4.25)$$

де $E_г$ – внутрішня економічна дохідність вкладених інвестицій.

$$T_{ок} = 1 / 0,518 = 1,93 \text{ р.}$$

$T_{ок} < 3$ -х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати розробника профінансувати впровадження даної розробки для застосування в діяльності підприємства.

4.5 Висновок до розділу 4

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Інформаційна технологія доставки води по місту» становить 42,0 бала, що свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки високий).

При оцінюванні за технічними параметрами, згідно узагальненого коефіцієнту якості розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 2,76 рази.

Також термін окупності становить 1,93 р., що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати розробника до впровадження даної розробки при отриманні ефекту в розмірі 1282429,92 грн.

Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою «Інформаційна технологія доставки води по місту».

ВИСНОВКИ

У ході виконання магістерської кваліфікаційної роботи розроблено інформаційну технологію доставки води по місту.

В роботі першому розділі проаналізовано предметну область замовлення та доставки води по місту, зазначено актуальність дослідження, визначено основну проблему доставки води по місту. Здійснено аналіз переваг та недоліків сучасних програмних засобів, призначених для замовлення та доставки води по місту. Запропоновано розробити нову інформаційну технологію, яка дозволить уникнути зазначених недоліків. Здійснено аналіз вимог до нової інформаційної технології доставки води по місту. Розроблено структуру інформаційної технології доставки води по місту, розроблено структуру бази даних а також діаграму сценаріїв використання. Розроблено алгоритм функціонування інформаційної технології доставки води по місту. Здійснено програмну реалізацію інформаційної технології доставки води по місту у вигляді веб-ресурсу. Обґрунтовано вибір мови програмування JavaScript та середовища розробки Visual Studio Code. Розроблено серверну та клієнтську частини веб-ресурсу для доставки води по місту, а також його дизайн та функціонал. Здійснено тестування веб-ресурсу для замовлення та доставки води по місту, при цьому було створено 5 найменувань різних марок води для оформлення замовлення, в результаті чого суттєвих проблем та неполадок не виявлено. В розроблений програмний продукт було додано 2 нових функціональних можливості: Сповіщення про статус замовлення в месенджері Telegram; Зворотній зв'язок через месенджер Telegram до веб-ресурсу.

В роботі виконано розрахунок загальних витрат на завершення науково-технічної роботи та оформлення її результатів, які складають 148543,59 гривень, період окупності витрат складає 1,93 року. При оцінюванні за технічними параметрами, згідно узагальненого коефіцієнту якості розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 2,76 рази. В результаті аналізу розрахунків можна зробити висновок, що розроблений програмний продукт за ціною дешевший за аналоги і є високо конкурентоспроможним.

Отже всі поставлені задачі було виконано, мету роботи досягнуто.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. В.С. Озеранський, Б.В. Пугач. Інформаційна технологія доставки води по місту. Матеріали Всеукраїнської науково-практичної інтернет-конференції «Молодь в науці: дослідження, проблеми, перспективи (МН-2023)» – [Електронний ресурс]. – <https://conferences.vntu.edu.ua/index.php/mn/mn2024/paper/viewFile/19718/16312>
2. Інформатика. Комп'ютерна техніка. Комп'ютерні технології. Баженов В.А., Венгерський П.С., Горлач В.М. та інші. - Київ, «Каравела», 2003.
3. Аналіз середовища розробки мобільних додатків Android Studio [Електронний ресурс]. – 2019. – Режим доступу до ресурсу: <https://sibac.info/studconf/tech/1/69356>.
4. Панос Л. Алгоритми для початківців. Теорія і практика для розробника / Луридас Панос. – Львів: Ексмо, 2015. – 608 с. – (100072). – (978-5-04-089834-3; кн. 978).
5. Об'єктно - орієнтоване програмування [Електронний ресурс] // Програмування по-українськи. – 2011. – Режим доступу до ресурсу: <https://programming.in.ua/programming/basisprogramming/25-oop.html>.
6. The pros and cons of using Angular in web dev [Електронний ресурс]:[Веб-сайт] – Електронні дані. — Режим доступу: <https://ddi-dev.com/blog/programming/pros-and-cons-of-angular-web-app-development/>
7. Reasons to Consider Angular Framework For Web Development in 2021 [Електронний ресурс]:[Веб-сайт] – Електронні дані. — Режим доступу: <https://www.monocubed.com/angular-framework-for-web-development/>
8. Why You Should Use React.js For Web Development [Електронний ресурс]:[Веб-сайт] – Електронні дані. — Режим доступу: <https://www.freecodecamp.org/news/why-use-react-for-web-development/> 63
9. The Good and the Bad of Vue.js Framework Programming [Електронний ресурс]:[Веб-сайт] – Електронні дані. — Режим доступу: <https://www.altexsoft.com/blog/engineering/pros-and-cons-of-vue-js/>

10. Reasons Why You Should Choose Vue.js for Web UI Development [Електронний ресурс]:[Веб-сайт] – Електронні дані. — Режим доступу: <https://appinventiv.com/blog/vue-js-for-web-ui-development/>

11. SQL vs. NoSQL Databases: What's the Difference? [Електронний ресурс]:[Веб-сайт] – Електронні дані. — Режим доступу: <https://www3.technologyevaluation.com/research/article/sql-vs-nosql-databases-what-s-the-difference.html>

12. SQL vs. NoSQL: Comparative Advantages and Disadvantages [Електронний ресурс]:[Веб-сайт] – Електронні дані. — Режим доступу: <https://bitnine.net/blog-computing/sql-vs-nosql-comparative-advantages-and-disadvantages/>

13. NOSQL vs SQL. Key differences and when to choose each [Електронний ресурс]:[Веб-сайт] – Електронні дані. — Режим доступу: <https://pandorafms.com/blog/nosql-vs-sql-key-differences>

14. A Perfect Fit: 7 Ways Brands Use Personalization to Create Tailor-Made Shopping Experiences[Електронний ресурс]:[Веб-сайт] – Електронні дані. — Режим доступу: <https://www.shopify.com/blog/personalized-shopping-experiences>

15. Online Shopping Personalization – Statistics and Trends[Електронний ресурс]:[Веб-сайт] – Електронні дані. — Режим доступу: <https://www.invespcro.com/blog/online-shopping-personalization>

16. Unified Modeling Language [Електронний ресурс]: Режим доступу: https://uk.wikipedia.org/wiki/Unified_Modeling_Language.

17. JavaScript [Електронний ресурс]: Режим доступу: <https://uk.wikipedia.org/wiki/JavaScript>.

18. Documentation for JS developers [Електронний ресурс]: Режим доступу: <https://developer.mozilla.org/en-US/>.

19. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

20. Кавецький В.В. Економічне обґрунтування інноваційних рішень: практикум / В.В. Кавецький, В.О. Козловський, І. В. Причепка – Вінниця : ВНТУ, 2016. – 113 с.

ДОДАТКИ

Додаток А (обов'язковий)

Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ
НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬНазва роботи: Інформаційна технологія доставки води по містуТип роботи: магістерська кваліфікаційна робота
(БДР, МКР)Підрозділ кафедра комп'ютерних наук, ФПТА
(кафедра, факультет)

Показники звіту подібності Unicheck

Оригінальність 83,8% Схожість 16,2%

Аналіз звіту подібності (відмітити потрібне):

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи



Пугач Б.В.

Керівник роботи



Озеранський В.С.

Опис прийнятого рішення

Магістерську кваліфікаційну роботу допущено до захисту

Особа, відповідальна за перевірку



Озеранський В.С.

Додаток Б (обов'язковий)

Лістинг програми

Серверна частина

```

const itemController = require('../controllers/item.controller');
const multer = require("multer");
const authCheck = require('../middlewares/session.checker');
const storage = multer.diskStorage({
  destination: function (req, file, cb) {
    cb(null, 'uploads/')
  },
  filename: function (req, file, cb) {
    cb(null, Date.now() + '.jpg') //Appending .jpg
  }
})
const upload = multer({
  dest:"uploads",
  storage: storage,
  limits:{
    fileSize: 1024 * 1024
  }});
const router = require('express').Router();
router.post('/:id', authCheck, upload.single('image'),
itemController.createItem);
router.post('/', authCheck, upload.single('image'),
itemController.createItem);
router.get('/', itemController.showItems);
router.delete('/:id' , authCheck, itemController.deleteItems);
module.exports = router;

createItem : async (req, res) => {
  const fileData = req.file;
  const id = req.params.id;
  let item;
  if(id){
    const [oldItem] = await itemsController.getItemById(id);
    const newItem = {
      ...req.body,
      ...(fileData ? {image: fileData.filename} : {})
    }
    if(!fileData) {
      delete newItem.image;
    }

    item = await updateItem( newItem, id)
    if(fileData && oldItem.image){
      await removeUploaded(oldItem.image);
    }
  } else {

```

```

        item = await createItem(
            {
                ...req.body,
                ...(fileData ? {image: fileData.filename} : {})
            }
        );
    }
    res.status(200).send(item);
}
module.exports = {
    showAllItems: () => knex('items').select().orderBy('order'),
    addNewItem: (props, image) => knex('items').insert(datePipe(props,
image)).returning('id')
        .then(([id]) =>{
            return knex('items').where('id',id)
        })),
    updateItem: (props, id) =>
knex('items').update(datePipe(props)).where({id}).returning('id')
        .then(([id]) =>{
            return knex('items').where('id',id)
        })),
    getItemById: (id) => knex('items').select().where({id}),
    deleteItemsById: (id) => knex('items').where({id}).del(),
}
add: async (req, res) => {
    const errors = validationResult(req);
    if (!errors.isEmpty()) {
        res.status(422).json({ errors: errors.array() });
        return;
    }
    const order = req.body;
    const customer = await
customerModule.getCustomerByPhone(order.phone);
    if(customer.length < 1){
        await customersModel.addNewCustomer(order);
    }

    const message = `Замовлення\n${createMessage(order)}`
    await Promise.all([
        sendToTelegram(message),
        orderModule.addNewOrder(order),
    ]);
    res.status(200);
    res.json({message: 'order sent'})
}
const Telegraf = require('telegraf');
const { CHAT_ID, TOKEN } = require('../constants')
const Bot = new Telegraf(TOKEN);
exports.sendToTelegram = (message) => Bot.telegram.sendMessage(CHAT_ID,
message);
const { check } = require('express-validator');
module.exports = [
    check('title').isString().notEmpty(),

```

```

    check('description').isString().notEmpty(),
    check('image').notEmpty(),
    check('price').isNumeric().notEmpty(),
  ];

```

Клієнтська частина

```

<>
  <FirstScreen
    itemCount={itemsInBasket.length}
    fastOrder={fastOrder}
    showBasketModal={() => setShowBasketModal(true)}
    scrollToItems={scrollToItems}
  />
  <Action twoBottleAction={fiveBottleAction} />
  <Advantages />
  <Element name="items">
    <Items items={sortBy(Object.values(items), i => i.order)}
  addItemToBasket={addItemToBasket}/>
  </Element>
  <ThreeSteps />
  <Reviews />
  <Footer />
    <FeedbackModal show={showFeedbackModal} onHide={() =>
setShowFeedbackModal(false)}/>
    <MainFormModal show={showMainFormModal} sendOrder={sendOrder}
onHide={() => setShowMainFormModal(false)} />
      <BasketModal itemsInBasket={itemsInBasket}
changeCounter={changeCounter} submitBasket={submitBasket}
show={showBasketModal} onHide={() =>
setShowBasketModal(false)} />

    <Snackbar
      anchorOrigin={{ vertical: 'top', horizontal: 'center' }}
      autoHideDuration={3000}
      open={!showSuccessModal}
      onClose={() => setShowSuccessModal('')}
    >
      <Alert onClose={() => setShowSuccessModal('')}
severity="success">
        {showSuccessModal}
      </Alert>
    </Snackbar>

  </>
import {setToLocal} from "../localStorage.js";

const addToBasket = (id) => {
  const itemsString = localStorage.getItem('items');
  if (itemsString) {
    const items = itemsString.split(',');
    if (!items.includes(id)) {
      setToLocal({ items: [...items, id].join(',') });
    }
  }
}

```

```

    }
  } else {
    setToLocal({ items: id });
  }
}

const deleteFromBasket = (id) => {
  const itemsString = localStorage.getItem('items');
  if (itemsString) {
    const items = itemsString.split(',');
    setToLocal({ items: items.filter(itemId => +itemId !==
id).join(',') });
  }
}

const deleteAllFromBasket = (id) => {
  setToLocal({ items: '' });
}

const getItemsFromBasket = () => {
  const items = localStorage.getItem('items');
  if (items) return items.split(',').map(Number);
  return [];
}

export {
  addToBasket,
  deleteFromBasket,
  getItemsFromBasket,
  deleteAllFromBasket
}

import fetch from '../helpers/fetch';
import urls from './urls/urls';

export async function getOrdersAction() {
  const response = await fetch(urls.orders,
  {
    method: 'GET',
    headers: {
      'Content-Type': 'application/json; charset=UTF-8',
    },
  })
  return response.json();
}

export async function saveOrderAction(order) {
  if (!order) throw new Error('No order!');

  const response = await fetch(urls.orders,
  {
    method: 'POST',
    headers: {
      'Content-Type': 'application/json; charset=UTF-8',
    },
  },

```



```

        body: JSON.stringify(order),
      })
    return response.json();
  }

export async function deleteOrderAction(id) {
  if (!id) throw new Error('No id!');

  const response = await fetch(urls.orders + id,
    {
      method: 'DELETE',
      headers: {
        'Content-Type': 'application/json; charset=UTF-8',
      },
    })
  return response.json();
}

module.exports = {
  required: (val) => val && (val.toString().trim()),
  number: (val) => !!Number(val),
  phone: (val) => val ? !(val.replace(/[\^\\d]/g, '').length < 12) : false,
  image: (file) => file && file.type && (file.type === 'image/jpeg' ||
file.type === 'image/png'),
  validateAll: (vals, fields) => {
    let allFight = true
    Object.keys(fields).forEach(field => {
      const validate = module.exports[fields[field]]
      if (!validate(vals[field])) allFight = false
    });
    return allFight
  }
}

```

Додаток В (обов'язковий)**ІЛЮСТРАТИВНА ЧАСТИНА****ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ДОСТАВКИ ВОДИ ПО МІСТУ**

Виконав: студент 2 курсу, групи ЗКН-22м

спеціальності 122 – Комп'ютерні науки
(шифр і назва напрямку підготовки, спеціальності)



Пугач Б.В.

(прізвище та ініціали)

Керівник: к.т.н., доцент каф. КН



Озеранський В.С.

(прізвище та ініціали)

« 08 » 12 2023 р.

Рисунки А.1 - Програми аналізу доставки води по місту

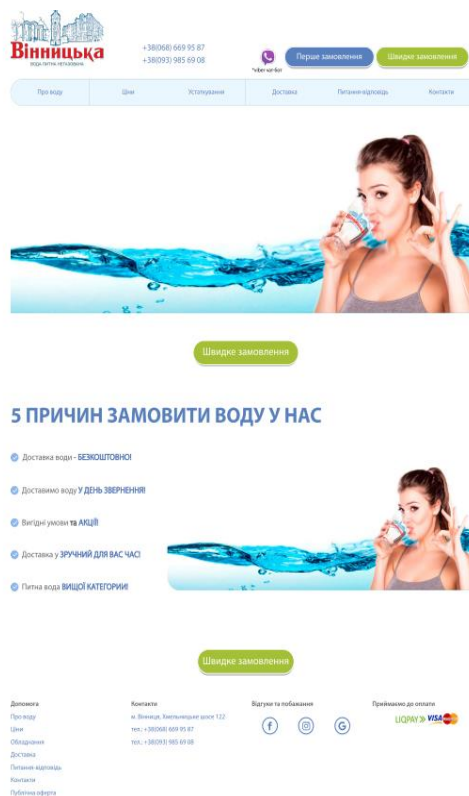
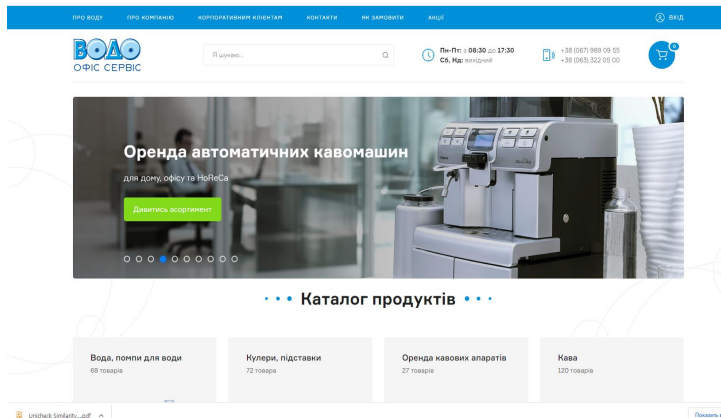
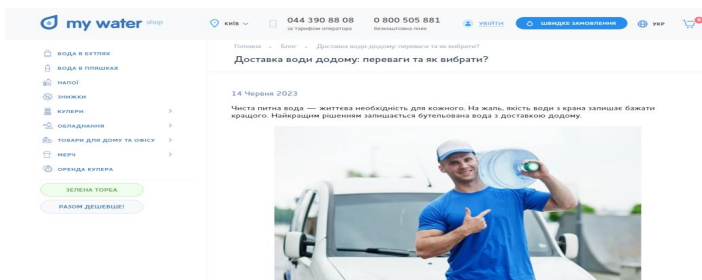


Рисунок В.1 - Програми аналогії доставки води по місту

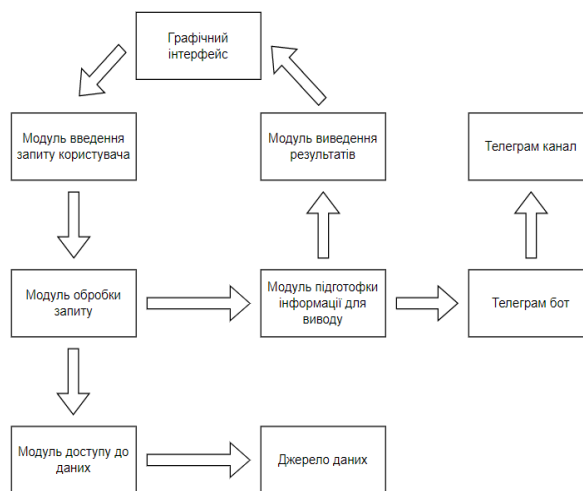


Рисунок В.2 - Структура інформаційної технології доставки води по місту

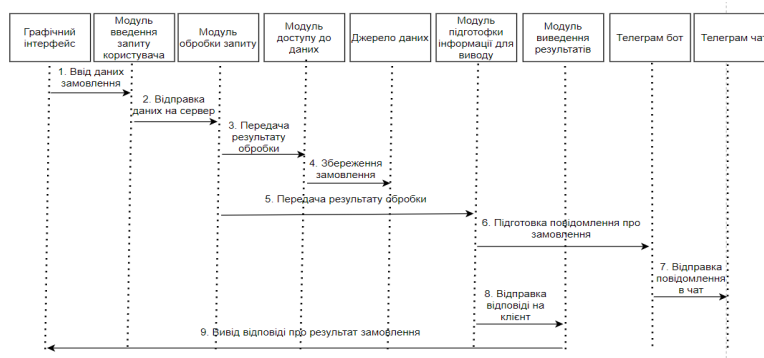


Рисунок В.3 - UML-Діаграма послідовності здійснення замовлення води

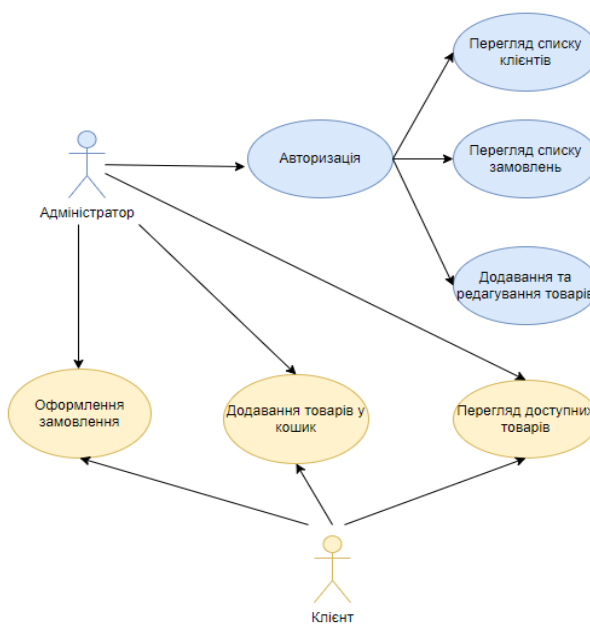


Рисунок В.4 - UML-Діаграма варіантів використання

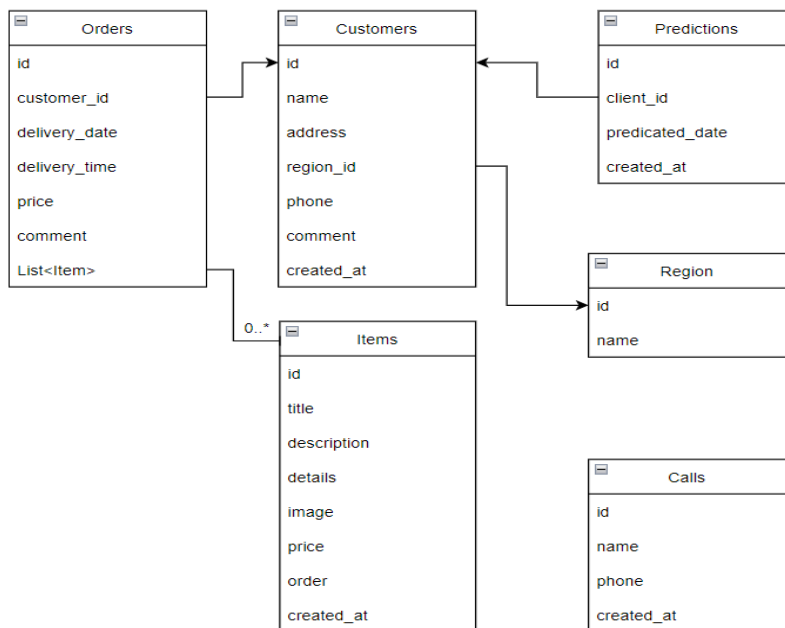


Рисунок В.5 - UML-Діаграма класів інформаційної технології доставки води по місту

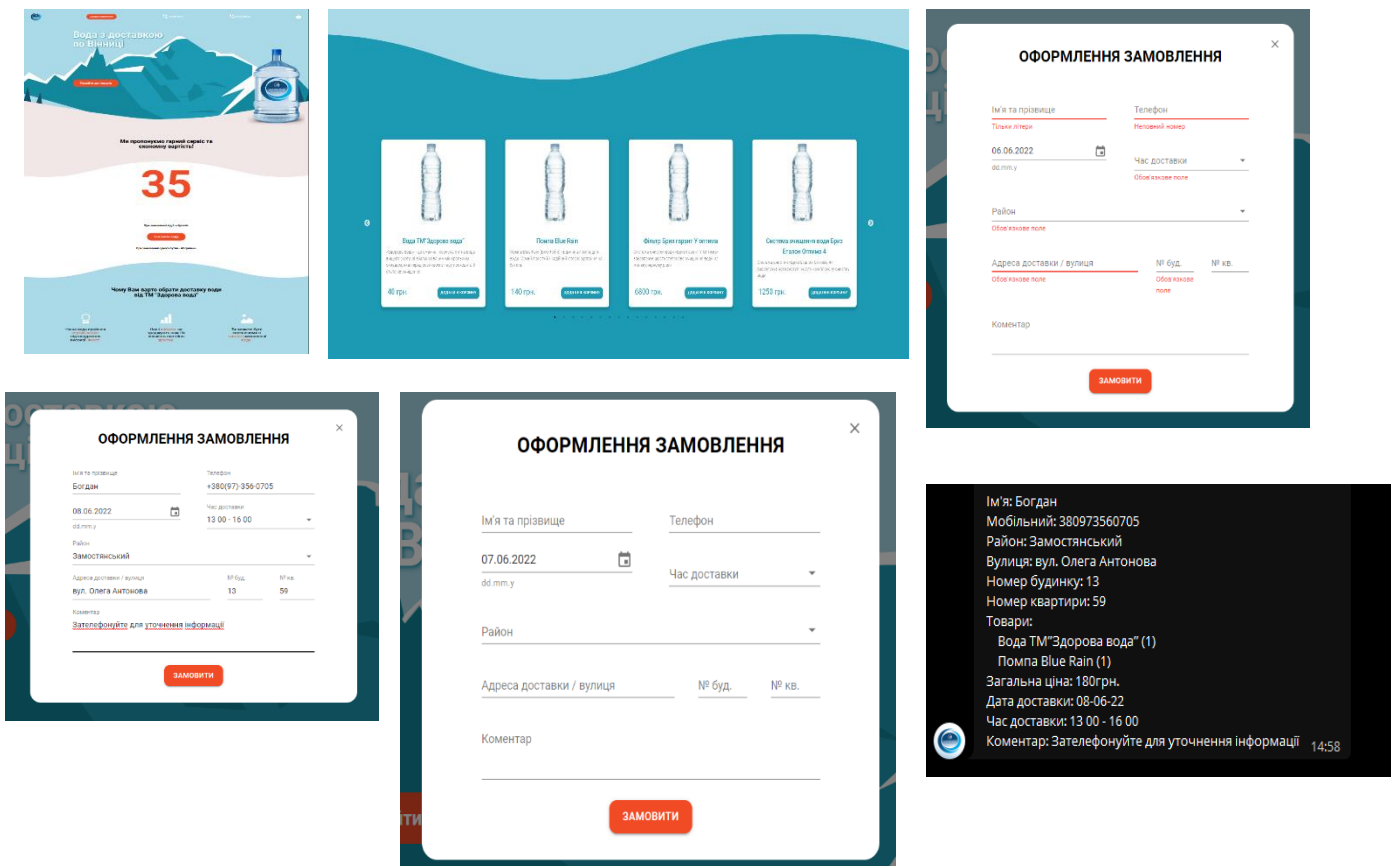
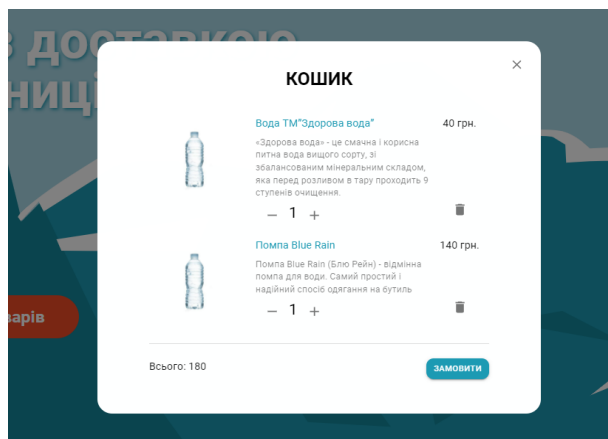


Рисунок В.6 - Програмне забезпечення для доставки води по місту



ADMIN PANEL

VODA

Сайт

Замовлення

Клієнти

Імовірні замовлення

Товари

Вихід

Замовлення

Search

АCTIONS	Ім'я	Телефон	Район	Адреса	М ² дому	М ² кв.	Товари	Ціна	Дата	Час	Коментар	Створено
	Медзон Франс	38073234892	Іщай	Лесі Ратушко	16	3	Вода ТМ "Здорова вода" (2), Помпа Blue Rain (1)	220	14/03/2022	13:00 - 16:00	1 під'їзд	14/03/2022
	Лілея Клім	38093074998	Завостинський	Олега Антєнова	20	1	Вода ТМ "Здорова вода" (3)	120	28/03/2022	16:00 - 20:00		27/03/2022
	Богдан	38097568705	Завостинський	Віра Олегівна	13	59	Вода ТМ "Здорова вода" (3)	120	30/05/2022	13:00 - 16:00	Комент	29/05/2022
	Іванівна	3803434343	Старосинський	улиця	34	43	Фільтр Blue Rain У оптом (1), Помпа Blue Rain (1)	6840	30/05/2022	13:00 - 16:00		29/05/2022

10 rows | 1-4 of 4

Рисунок В.7 - Програмне забезпечення для доставки води по місту

Додаток Г (довідниковий) Інструкція користувача

Одразу після переходу користувача по посиланню на сайт відкривається головна сторінка яка складається з блоку заголовку, в якому відображенні кнопки для здійснення замовлення, корзина, та елементи з логотипом власника (рис. Г.1).

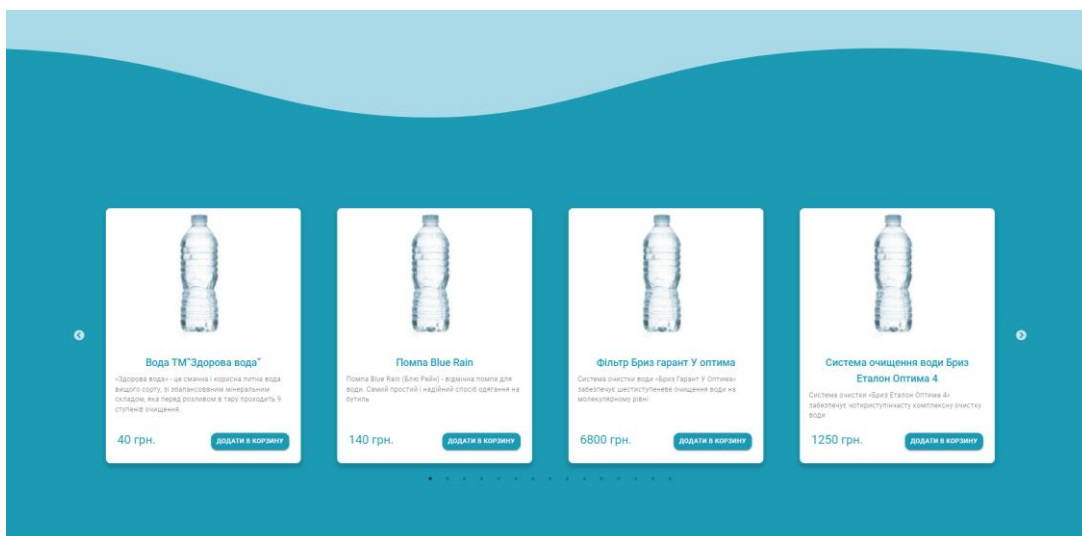


Рисунок Г.1 – Каталог товарів

Обравши необхідний товар, користувач може натиснути кнопку з написом “Додати товар”. Після натискання, має з’явитись спливаюче оповіщення про успішність додавання (рис. Г.2).

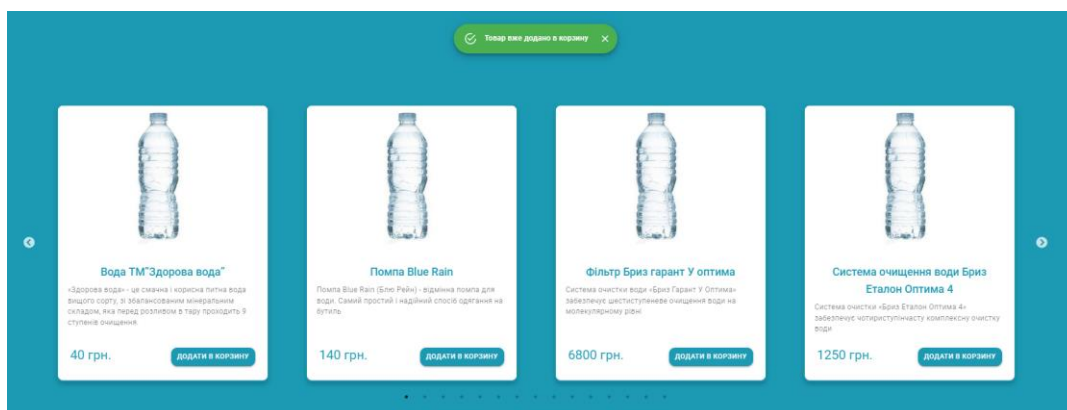


Рисунок Г.2 – Додавання товару в корзину

Після того як товар бу доданий, біля значка корзини з'явиться помітка з кількістю доданих товарів. Користувач може перейти до корзини натиснувши на її іконку, після чого відкриється модальне вікно з списком обраних товарів, з можливістю додавання або зменшення одиниць певної продукції, або взагалі видалити її. В нижній лівій частині діалогового вікна відображається загальна сума замовлення, а у нижній правій – Кнопка для переходу до заповнення решти інформації яка для цього необхідна (рис. Г.3).

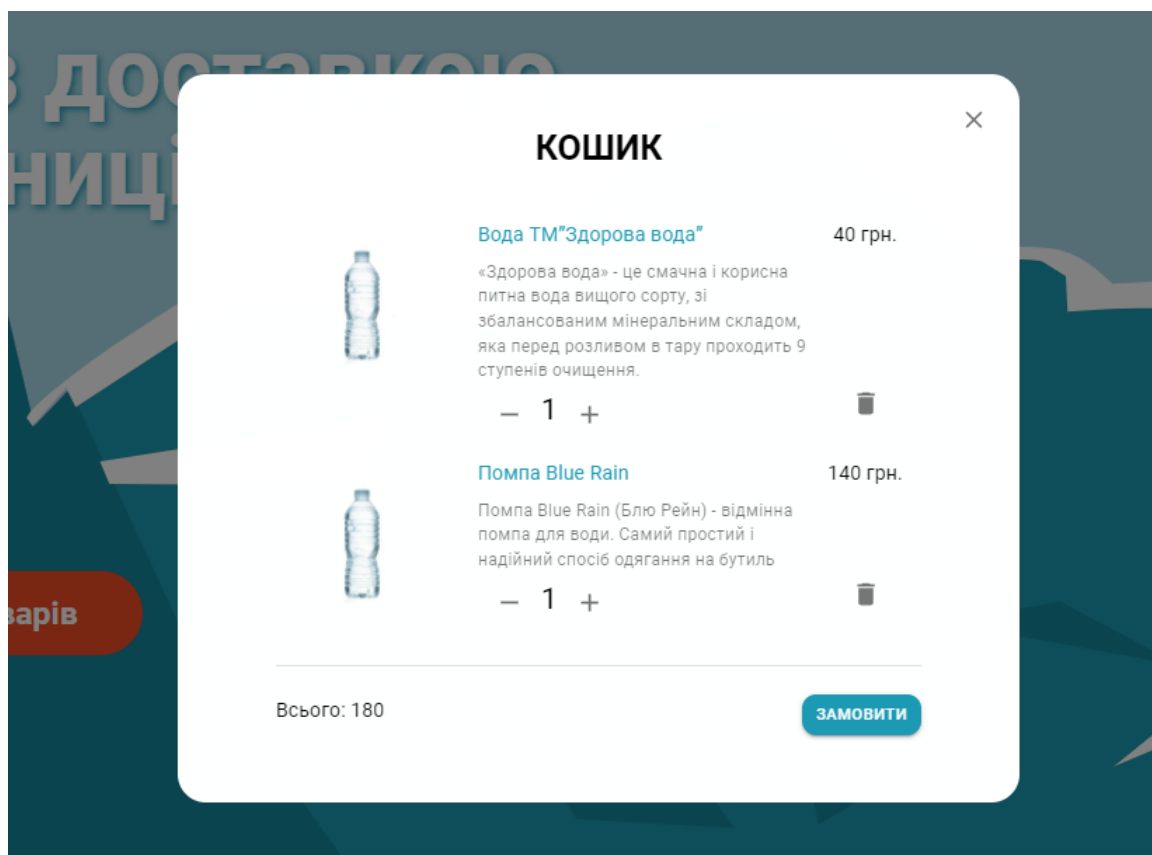


Рисунок Г.3 – Кошик товарів

Коли користувач натискає кнопку замовити повинне відкритись вікно з формою замовлення (рис. Г.4).

ОФОРМЛЕННЯ ЗАМОВЛЕННЯ

Ім'я та прізвище: _____ Телефон: _____

07.06.2022 Час доставки: _____
dd.mm.y

Район: _____

Адреса доставки / вулиця: _____ № буд. _____ № кв. _____

Коментар: _____

ЗАМОВИТИ

Рисунок Г.4 – Форма замовлення

Якщо незаповнені важливі поля, і натиснути кнопку з написом “Замовити” спрацює валідація, і відповідні поля вводу підсвітяться червоним кольором (рис. Г.5).

ОФОРМЛЕННЯ ЗАМОВЛЕННЯ

Ім'я та прізвище: _____ Телефон: _____
Тільки літери Неповний номер

06.06.2022 Час доставки: _____
dd.mm.y Обов'язкове поле

Район: _____
Обов'язкове поле

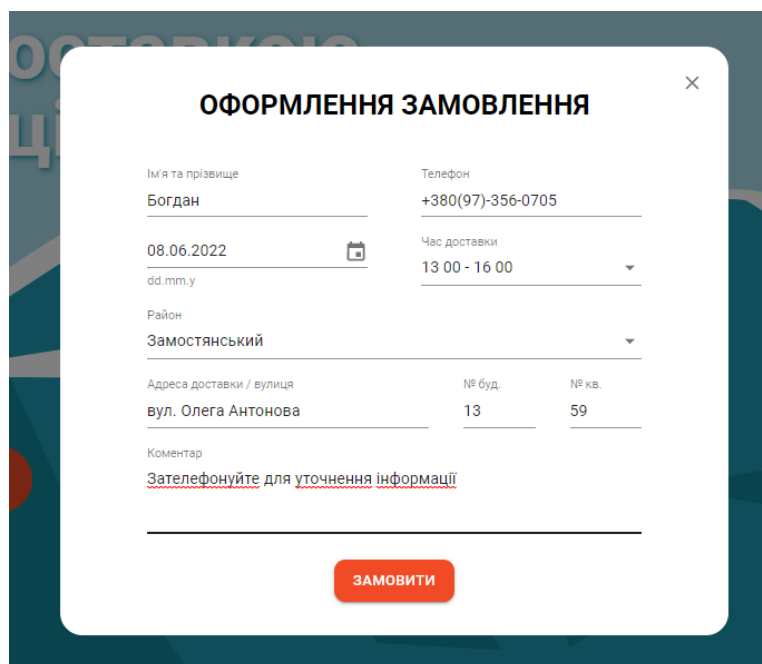
Адреса доставки / вулиця: _____ № буд. _____ № кв. _____
Обов'язкове поле Обов'язкове поле

Коментар: _____

ЗАМОВИТИ

Рисунок Г.5 – Валідація форми замовлення

Якщо заповнити всі необхідні поля слідуєчи інструкціям, замовлення буде успішно відправлене на сервер (рис. Г.6).



ОФОРМЛЕННЯ ЗАМОВЛЕННЯ

Ім'я та прізвище: Богдан

Телефон: +380(97)-356-0705

08.06.2022

Час доставки: 13 00 - 16 00

Район: Замостянський

Адреса доставки / вулиця: вул. Олега Антонова

№ буд.: 13

№ кв.: 59

Коментар: Зателефонуйте для уточнення інформації

ЗАМОВИТИ

Рисунок Г.6 – Оформлення замовлення

У разі успішного опрацювання замовлення сервером запис про нього буде доданий в базу даних, а також власник сайту отримає сповіщення в телеграм (рис. Г.7).

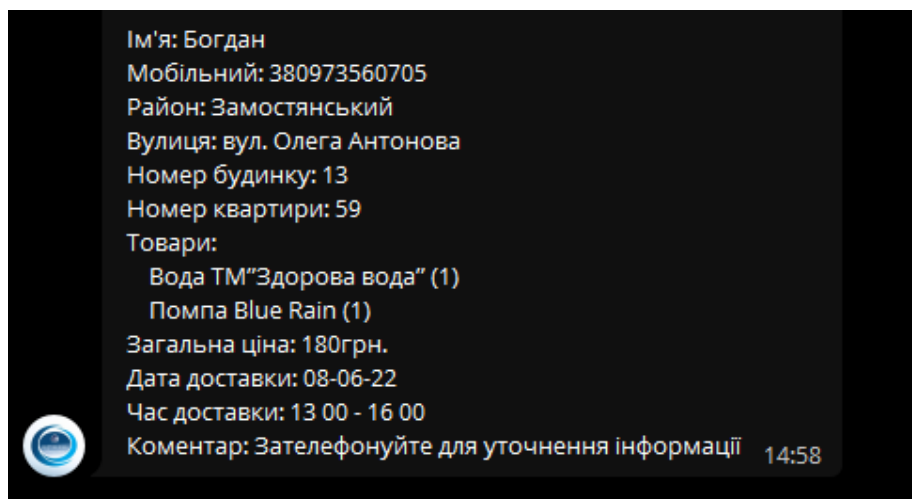


Рисунок Г.7 – Отримання сповіщення про замовлення в телеграм

Ще однією важливою частиною сайту є сторінка яка доступна для перегляду лише для власника сайту. Для цього необхідно додати “/admin” до адресу сайту, після чого відкриється сторінка авторизації (рис. Г.8).

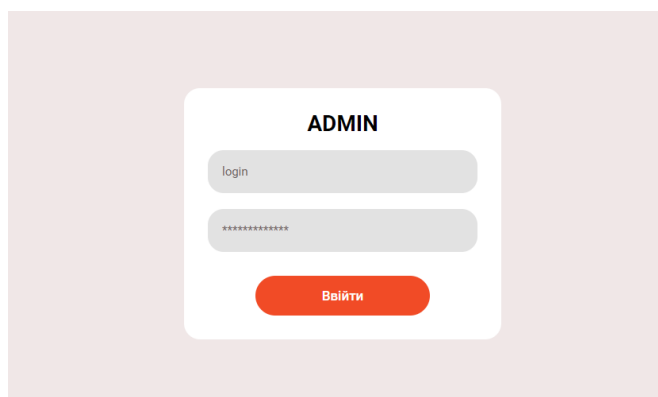
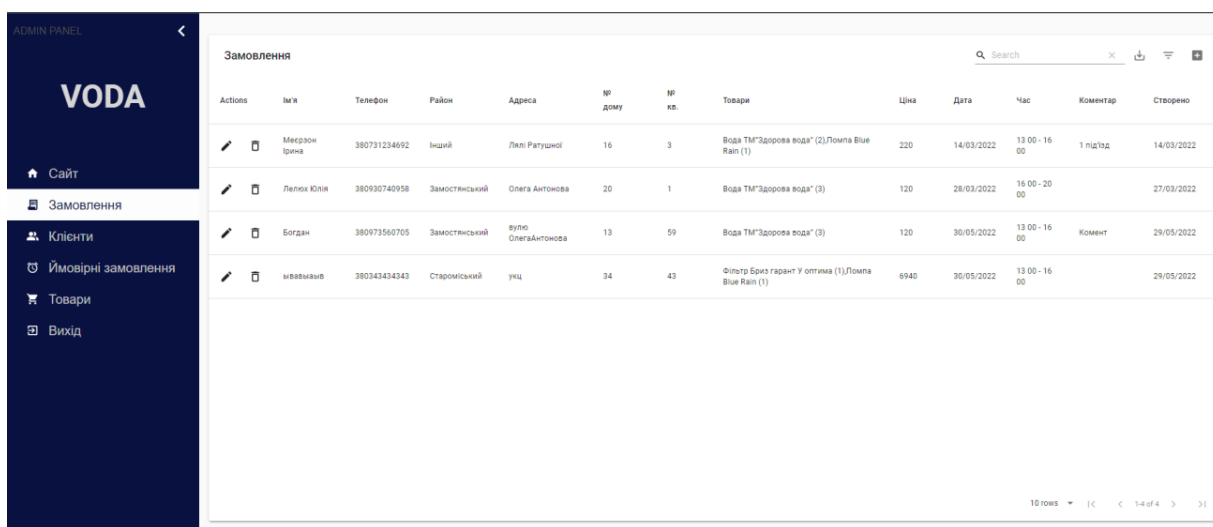


Рисунок Г.8 – Сторінка авторизації

Після авторизації власник отримає доступ до сторінок з списком замовлень, клієнтів та товарів (рис. Г.9, Г.10).



Actions	Ім'я	Телефон	Район	Адреса	№ дому	№ кв.	Товари	Ціна	Дата	Час	Коментар	Створено
	Мекрзон Ірина	380731234692	Іщій	Лілі Ратушної	16	3	Вода ТМ "Здорова вода" (2), Ломпа Blue Rain (1)	220	14/03/2022	12 00 - 16 00	1 під'юд.	14/03/2022
	Лелік Юлія	380930740958	Замостянський	Олега Антонова	20	1	Вода ТМ "Здорова вода" (3)	120	28/03/2022	16 00 - 20 00		27/03/2022
	Богдан	380973560705	Замостянський	Вулиця Олега Антонова	13	59	Вода ТМ "Здорова вода" (3)	120	30/05/2022	13 00 - 16 00	Комент	29/05/2022
	мввмвмвм	380343434343	Староміський	улиця	34	43	Фільтр Бриз гарант У оптима (1), Ломпа Blue Rain (1)	6940	30/05/2022	13 00 - 16 00		29/05/2022

Рисунок Г.9 – Сторінка зі списком замовлень

Actions	Фото	Ім'я	Пріоритет	Ціна	Опис	Деталі	Створено
		Вода ТМ "Здорова вода"	1	40	"Здорова вода" - це смачна і корисна питна вода вищого сорту, зі збалансованим мінеральним складом, яка перед розливом в тару проходить 9 ступенів очищення.		30.05.2021
		Помпа Blue Rain	2	140	Помпа Blue Rain (Блю Рейн) - відмінна помпа для води. Самий простий і надійний спосіб одгання на бутиль		15.05.2021
		Фільтр Бриз гарант У оптима	3	6800	Система очистки води «Бриз Гарант У Оптима» забезпечує шестиступеневе очищення води на молекулярному рівні		15.05.2021
		Система очищення води Бриз Еталон Оптима 4	4	1250	Система очистки «Бриз Еталон Оптима 4» забезпечує чотириступінчасту комплексну очистку води		15.05.2021
		Купити нову модель VIO X12-FC BLACK	5	9800	Характеристики VIO X12-FC BLACK, кулер для води компресорний, підлоговий, вид настільний. Тип установки - Верхнє		14.06.2021

Рисунок Г.10 – Сторінка зі списком товарів

Власник може переглядати, редагувати, видаляти, зберігати звіти локально на свій пристрій. Під час редагування є можливість завантаження зображення з пристрою. В формі є можливість завантаження зображення з локальної файлової систем. Після відправки форми файл буде збережений на сервері.