НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені ІГОРЯ СІКОРСЬКОГО»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

«На правах рукопису»
УДК 004.05

До захисту допущено:
В.о. завідувача кафедри
_____ Михайло НОВОТАРСЬКИЙ
«__» _____ 2024 р

Магістерська дисертація

на здобуття ступеня магістра

за освітньо-науковою програмою «Інженерія програмного забезпечення»
зі спеціальності 123 «Комп'ютерна інженерія»
на тему: «Система обробки логістичних перевезень та збуту нафтопродуктів»

Виконав:
студент ІІ курсу, групи ІО-32мп
Соболь Денис Дмитрович _____

Керівник:
Доц, к.т.н., доц.
Верба Олександр Андрійович _____

Консультант з нормоконтролю:
Проф., д.т.н., проф.,
Кулаков Юрій Олексійович _____

Рецензент:
Доц., к.т.н, доц.
Шимкович Володимир Миколайович _____

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент (-ка) _____

Київ – 2024 року

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМ. ІГОРЯ СІКОРСЬКОГО»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Рівень вищої освіти – другий (магістерський) за освітньо-науковою програмою

Спеціальність: 123 Комп'ютерна інженерія

Спеціалізація: Комп'ютерні системи та мережі

ЗАТВЕРДЖУЮ

Т. в. о. завідувача кафедри

_____*Михайло НОВОТАРСЬКИЙ*

(підпис)

"___" _____ 2024 р.

ЗАВДАННЯ

на магістерську дисертацію студенту

Соболю Денису Дмитровичу

1. Тема дисертації «*Система обробки логістичних перевезень та збуту нафтопродуктів»*_____

Науковий керівник дисертації доц. кафедри ОТ, к.т.н., доц., Верба О.А.

2. Строк подання студентом дисертації _____02.12.2024_____

3. Об'єкт дослідження є логістичний процес перевезення і збуту нафтопродуктів _____

4. Предмет дослідження: інформаційні та логістичні технології, що використовують для управління перевезеннями та збутом нафтопродуктів____

5. Перелік завдань, які потрібно розробити: Результати дослідження можуть бути використані для вдосконалення логістичних процесів в компаніях, які займаються перевезенням і збутом нафтопродуктів, сприяють зниженню витрат, оптимізації маршрутів транспортування, покращенню обслуговування клієнтів та забезпеченню екологічних вимог.

6. Консультанти розділів дисертації:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Нормоконтроль	проф. каф. ОТ, д.т.н., проф. Кулаков Ю.О.		

7. Дата видачі завдання 1 вересня 2024 року

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів дипломного проекту (роботи)	Строк виконання етапів проекту(роботи)	Примітки
1.	Затвердження теми роботи	01.09.2024	
2.	Вивчення та аналіз завдання	01.09.2024-15.09.2024	
3.	Розробка архітектури системи	15.09.2024-29.09.2024	
4.	Розробка підсистем	30.09.2024-13.10.2024	
5.	Програмна реалізація системи	14.10.2024-10.11.2024	
6.	Оформлення пояснювальної записки	11.11.2024-27.11.2024	
7.	Передзахист	12.12.2024	
8.	Захист	17.12.2024	

Студент _____ (підпис) Денис, СОБОЛЬ

Науковий керівник _____ (підпис) Олександр, ВЕРБА

РЕФЕРАТ

на магістерську дисертацію

виконану на тему:

Система обробки логістичних перевезень та збуту нафтопродуктів

Студентом: **Соболем Денисом Дмитровичем**

Робота складається із вступу та чотирьох розділів. Загальний обсяг роботи: 74 аркушів основного тексту, 21 таблиця. При підготовці використовувалася література з 46 різних джерел.

Актуальність дослідження. Системи обробки логістичних перевезень та збуту нафтопродуктів обумовлена зростаючою важливістю ефективного управління ланцюгами постачання в умовах сучасної економіки. Нафтопродукти є стратегічно важливим ресурсом, який використовується в багатьох галузях — від енергетики до транспорту і промисловості. Ефективне управління логістикою та збутом нафтопродуктів сприяє оптимізації витрат, зниженню ризиків та забезпеченню безперервності постачання, що особливо актуально в умовах нестабільних цін на нафту та конкуренції на ринку енергоносіїв. У сучасному світі глобалізація та стрімкий розвиток цифрових технологій призводять до значного ускладнення ланцюгів постачання. Це викликає необхідність впровадження автоматизованих систем управління логістикою, які дозволяють оперативно реагувати на зміни ринкових умов, враховувати географічні та політичні ризики, оптимізувати маршрути перевезень, а також підвищувати прозорість і точність обліку нафтопродуктів. Інтеграція цифрових рішень, таких як системи моніторингу, автоматизація планування та аналітики, дозволяє покращити управління ланцюгом постачання нафтопродуктів, що забезпечує високу ефективність операцій.

Тема дослідження. Розробка та вдосконалення систем обробки логістичних перевезень та збуту нафтопродуктів з використанням сучасних інформаційних технологій.

Мета дослідження. Методом дослідження є аналіз та оцінка наявних підходів до організації логістичних процесів і збуту нафтопродуктів, а також розробка рекомендацій для вдосконалення системи логістичних перевезень з урахуванням сучасних технологій управління та моніторингу.

Об'єкт дослідження. Логістичний процес перевезення і збуту нафтопродуктів у контексті діяльності паливно-енергетичних компаній.

Предмет дослідження інформаційні та логістичні технології, що використовують для управління перевезеннями та збутом нафтопродуктів, а також методи оптимізації процесів логістики нафтопродуктів.

Особистий внесок здобувача. Магістерське дослідження є самостійно виконаною роботою, у якій відображається створення унікальної в своїй галузі системи обробки замовлень на транспортні перевезення та торгівлі нафтопродуктами.

Наукова новизна: Результати дослідження можуть бути використані для вдосконалення логістичних процесів в компаніях, які займаються перевезенням і збутом нафтопродуктів, сприяють зниженню витрат, оптимізації маршрутів транспортування, покращенню обслуговування клієнтів та забезпеченню екологічних вимог. Впровадження рекомендованих технологій і методик.

Ключові слова: Система логістики, управління перевезеннями, збут нафтопродуктів, GPS-моніторинг, автоматизація логістики, управління замовленнями, оптимізація маршрутів, безпека транспортування, аналітика логістичних процесів, екологічна логістика.

ABSTRACT

for a master's thesis performed

on the topic: Logistics and Sales Management Platform for Petroleum Products

by student: **Sobol Denys Dmytrovych**

The work consists of an introduction and four chapters. Total amount of work: 74 sheets of the main text, 21 tables. Literature from 46 different sources was used in the preparation.

Actuality: Systems for processing logistics transportation and sales of petroleum products are due to the growing importance of effective supply chain management in the modern economy. Petroleum products are a strategically important resource used in many industries - from energy to transport and industry. Effective management of logistics and sales of petroleum products helps to optimize costs, reduce risks and ensure continuity of supply, which is especially relevant in conditions of unstable oil prices and competition in the energy market. In the modern world, globalization and the rapid development of digital technologies lead to significant complication of supply chains. This necessitates the implementation of automated logistics management systems that allow for rapid response to changes in market conditions, take into account geographical and political risks, optimize transportation routes, and increase transparency and accuracy of accounting for petroleum products. Integration of digital solutions, such as monitoring systems, automation of planning and analytics, allows to improve the management of the supply chain of petroleum products, which ensures high efficiency of operations.

The objectives of the study. Development and improvement of systems for processing logistics transportation and sales of petroleum products using modern information technologies.

The purpose of the study. The research method is the analysis and evaluation of existing approaches to the organization of logistics processes and sales of petroleum products, as well as the development of recommendations for improving the logistics transportation system taking into account modern management and monitoring technologies.

Object of study. The logistics process of transportation and sales of petroleum products in the context of the activities of fuel and energy companies.

Subject of study is information and logistics technologies used to manage the transportation and sales of petroleum products, as well as methods for optimizing the logistics processes of petroleum products.

Personal contribution of the applicant. The master's thesis is an independently completed work, which reflects the creation of a unique in its field system for processing orders for transportation and trade in petroleum products.

Scientific novelty: The results of the study can be used to improve logistics processes in companies engaged in the transportation and sale of petroleum products, contribute to reducing costs, optimizing transportation routes, improving customer service and ensuring environmental requirements. Implementation of recommended technologies and methods.

Keywords: Logistics system, transportation management, sales of petroleum products, GPS monitoring, logistics automation, order management, route optimization, transportation security, analytics of logistics processes, environmental logistics.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ 1

ВСТУП 2

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ 3

1.1 Огляд процесів логістики та збуту нафтопродуктів. Опис основних етапів процесу транспортування та збуту 3

1.2 Конкретизація проблем, які система повинна вирішити (наприклад, відстеження місцезнаходження транспорту, керування замовленнями) 7

1.3 Перелік функціональних (керування замовленнями, відстеження маршруту) та нефункціональних (надійність, безпека) вимог 11

ВИСНОВКИ ДО РОЗДІЛУ 1 17

РОЗДІЛ 2. РОЗРОБКА ІНТЕРФЕЙСУ КОРИСТУВАЧА 18

2.1 Основні сторінки та функціональність 18

2.2 Панель адміністратора для управління замовленнями та маршрутами 22

2.3 Відображення статусів замовлень та транспорту на карті 24

2.4 Поняття зручності та інтуїтивності інтерфейсу 25

ВИСНОВКИ ДО РОЗДІЛУ 2 29

РОЗДІЛ 3. ТЕХНІЧНА РЕАЛІЗАЦІЯ 30

3.1 Вибір технологій та інструментів. Вибір мови програмування, бази даних (наприклад, PostgreSQL або MySQL), а також інструментів для фронтенду (React або Angular) 30

3.2. Реалізація бекенду. API для обміну даними. Впровадження логіки управління замовленнями, моніторингом і маршрутизацією 38

3.3. Інтеграція з GPS-трекерами. Використання API для підключення до GPS-трекерів транспорту. 45

3.4. Розробка бази даних. Створення бази даних для зберігання інформації про маршрути, замовлення, транспорт 47

ВИСНОВКИ ДО РОЗДІЛУ 3 54

РОЗДІЛ 4. РОЗРОБКА СТАРТАП ПРОЕКТУ 55

4.1. Загальна характеристика стартап-проекту 55
4.2. Технологічний аудит ідеї проекту 57
4.3. Аналіз ринкових можливостей запуску стартап-проекту 58
4.4.Розробка ринкової стратегії проекту 63
4.5 Розроблення маркетингової програми стартап-проекту 66
ВИСНОВКИ ДО РОЗДІЛУ 4 67
ВИСНОВОК 69
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 71

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Логістичні компанії – компанії, головна діяльність яких обробка, контроль та виконання перевезень різних типів вантажів.

JS(Java Script) – мова програмування, використовується при розробці веб-сервісів.

JS-фреймворки – це інструменти мови Java Script, які мають у собі вже створені функції та компоненти, що допомагають вирішенню задач при розробці.

Node.js, Vue.js – фреймворки JavaScript

React – бібліотека JavaScript, за допомогою якої створюється зовнішній інтерфейс.

Angular – фреймворк мови TypeScript, який застосовують для створення веб-платформ.

PostgreSQL – система управління баз даних, застосовується задля обробки, управління та зберігання інформації.

Memcached – система, яка кешують інформацію, в результаті чого системи працюють значно швидше.

API – у певному сенсі «перекладач», слугує для зв’язку різних програм між собою.

ВСТУП

Система обробки логістичних перевезень та збуту нафтопродуктів є ключовим компонентом для оптимізації та контролю складних процесів, пов’язаних із транспортуванням, зберіганням та реалізацією паливних ресурсів. В умовах сучасної системи високого ринку нафтопродуктів, де попит супроводжується жорсткими стандартами, екологічними вимогами безпеки та економічними викликами, логістичні не мають лише базових функцій доставки, але й забезпечують прозорість та ефективність на кожному етапі ланцюга поставок. Інтеграція цифрових технологій дозволяє значно знизити операційні витрати, підвищити точність прогнозування запиту, а також оптимізувати маршрути та управління запасами.

До того ж системи обробки логістики та збуту нафтопродуктів дають можливість швидко адаптуватися до змін ринку, оперативно реагуючи на коливання ціни на нафту та нові вимоги до якості продукції. За допомогою таких рішень компанії можуть також досліджувати кожен етап логістичного процесу в реальному часі, знижуючи ймовірність затримок і мінімізуючи ризики витрат під час транспортування та зберігання. Ефективне використання даних, які збираються, сприяє підвищенню системи загальної рентабельності бізнесу, дозволяє більш раціонально розподіляти ресурси та адаптуватися до динамічних умов ринку. Таким чином, впровадження сучасної системи обробки логістичних перевезень і збуту нафтопродуктів є стратегічним кроком, що сприяє посиленню конкурентних позицій компанії та її стійкості в умовах глобалізованої економіки.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд процесів логістики та збуту нафтопродуктів. Опис основних етапів процесу транспортування та збуту

Аналіз предметної області логістики та збуту нафтопродуктів сприяє більш детальному вивченню процесів, що забезпечують доставку паливних ресурсів від виробника до кінцевого споживача. У цьому контексті ключовими аспектами є управління ланцюгами поставок, оптимізація транспортування та складська логістика, оскільки кожен із цих процесів має критичне значення для своєчасного та ефективного постачання нафтопродуктів. Галузь вимагає суворого утримання як економічних, так і екологічних стандартів, що змушує компанії постійно шукати нові способи зменшення операційних витрат і впливу на довкілля. Враховуючи високу вартість перевезення нафтопродуктів, особливо на далекій відстані, велике значення має оптимізація маршрутів та зменшення витрат на паливо й обслуговування транспорту. безсумнівно, зберігання цих нафтопродуктів вимагає використання спеціалізованих ємностей і суворого контролю за їх станом, щоб уникнути втрати або зниження якості продукції. Автоматизація процесів контролю та управління запасами, моніторингу в реальному часі, а також застосування сучасних інформаційних технологій, зокрема інтернет-речей (IoT) і машинного навчання, дозволяє підвищити ефективність кожного з цих етапів, зменшуючи людський фактор і знижуючи ризики помилок [1].

Аналіз предметної області зосереджується на тому, як мінливі ринкові умови, такі як коливання ціни на нафту та зміни в законодавчій базі, впливають на процеси логістики і збуту. Наприклад, вимоги до зменшення викидів CO_2 або підвищення енергоефективності змушують компанію розробляти екологічно стійкі рішення, які могли б відповідати сучасним стандартам безпеки. У зв’язку з цим налаштовується актуальність впровадження комплексних інформаційних систем для аналізу даних, які не лише підтримують оперативні рішення, а й сприяють побудові гнучких

стратегій для подолання можливих кризових ситуацій на ринку. Такий підхід забезпечує компанії можливість більшої адаптивності до ринкових умов, підвищуючи їх конкурентоспроможність.

Аналіз предметної області логістики та збуту нафтопродуктів враховує широкий спектр зовнішніх і внутрішніх факторів, які формують як загальну ефективність процесів, так і економічну рентабельність. Логістика нафтопродуктів є складною структурою, де інтеграція інформаційних технологій забезпечує безперервність і точність ланцюга постачань. Наприклад, контроль за тим, якою продукція на всіх етапах транспортування і зберігання є глобальним продуктом, навіть незначні відхилення в умовах зберігання можуть негативно вплинути на експлуатаційні характеристики нафтопродуктів. Використання автоматизованої системи моніторингу таких умов, як температура та тиск у резервуарах, дозволяє підтримувати стабільність якості, а також зменшувати ризики забруднення чи витоку. Ефективна логістика нафтопродуктів також забезпечує оптимальне планування транспортних ресурсів та аналіз попиту, що особливо важливо в умовах сезонних коливань споживання. Компанії прагнуть створювати прогнози на основі великих масивів даних, що збираються з різних джерел, таких як статистичні дані споживання, особливості поведінки ринку та загальні економічні тенденції. Це дозволяє не тільки оперативно реагувати на зміни в попиті, але й мінімізувати витрати, пов'язані із зберіганням та транспортуванням продукції. без того, використання штучного інтелекту для прогнозування попиту підвищеної точності в управлінських запасах та оптимізації ланцюгів поставок, що дає змогу компаніям досягти максимальної ефективності [2].

Ще одним аспектом є забезпечення прозорості на всіх етапах ланцюга поставок. Системи управління, які не можуть відслідковувати пересування кожної партії продукції в реальному часі, сприяють підвищенню рівня довіри серед партнерів і споживачів. Це особливо актуально в контексті дотримання міжнародних екологічних стандартів, де точність звітності та контроль за

дотриманням норм вирішують роль. Крім того, розвиток цифрових платформ дозволяє об'єднувати різні елементи ланцюга поставок на єдиній платформі, що спрощує координацію між відділами та покращує взаємодію з постачальниками та клієнтами.

В результаті компанії, які успішно впроваджують комплексний підхід до обробки логістики і збуту нафтопродуктів, зберігають стратегічні переваги, оскільки можуть значно скоротити витрати, збільшити швидкість доставки і гнучко реагувати на ринкові виклики. Такий підхід дозволяє їм створити стабільну базу для довгострокового розвитку і бути більш конкурентоспроможними в мінливих умовах глобальної економіки.

Огляд процесів логістики та збуту нафтопродуктів охоплює весь ланцюг руху продукції від моменту її виробництва або виробництва до кінцевого споживача. На кожному етапі цього процесу зосереджено увагу на забезпеченні максимальної ефективності, мінімізації ризиків і дотриманні суворих стандартів якості та безпеки. Логістика нафтопродуктів включає складну мережу доставки, яка повинна виконувати такі фактори, як географічні особливості, доступність транспортної інфраструктури, попит на виробництво в різних регіонах і наявність потужностей для зберігання.

На початкових етапах виробництва чи виробництва нафтопродуктів компанії зіштовхуються із забезпеченням оперативної обробки продукції, щоб зберегти її якість та цінність. Далі процес включає транспортування нафтопродуктів на нафтобази або спеціалізовані склади, які часто розташовані поблизу споживачів або промислових центрів. Для цього використання використовують транспорт, зокрема трубопроводи, залізничний транспорт, морські танкери та автомобілі. Кожен вид транспорту має свої особливості, і його вибір залежить від факторів, таких як обсяг продукції, відстань до пункту призначення та терміновість доставки [3].

Зберігання нафтопродуктів на проміжних базах вимагає не тільки дотримання суворих екологічних норм, а й постійного моніторингу, оскільки нафтопродукти є

пожежобезпечними і можуть швидко втратити свої властивості при неналежних умовах зберігання. Тут важливу роль відіграють системи автоматизованого контролю, які відстежують параметри зберігання в реальному часі, запобігаючи втратам і зниженню якості.

Збут нафтопродуктів є фінальним етапом логістичного ланцюга, який передбачає продаж кінцевої продукції споживачу або дистриб'юторам. На цьому етапі особливу роль виконують маркетингові стратегії та прогнози попиту. Цифрові технології, зокрема системи управління складськими запасами та прогнозування попиту, не дозволяють компаніям адаптуватися до змін та забезпечити безперебійний збут. У підсумку, успішний огляд і оптимізація цих логістичних процесів дає можливість компаніям мінімізувати витрати, підвищити точність прогнозів і забезпечити стабільне постачання нафтопродуктів на ринок [4].

Процес транспортування та збуту нафтопродуктів розгортається в кількох ключових етапах, кожен із яких має свої особливості й виклики, потім цей сектор вимагає підтримки високих стандартів якості, точності й безпеки. На початковому етапі, після виробництва або видобутку, нафтопродукти доставляються до первинних сховищ або нафтобази, що розташовані в регіонах, які стратегічно зручні для подальшого розподілу. Цей етап передбачає використання транспортної інфраструктури, яка найкраще відповідає обсягу продукції та відстані – зокрема, це можуть бути трубопроводи для великих обсягів або ж залізничний і автомобільний транспорт для більшого гнучкого переміщення.

Нафтопродукти зберігаються на проміжних складах, де умови мають бути підконтрольні, щоб забезпечити стабільність влади пального. Системи моніторингу, впроваджені на цьому етапі, забезпечують контроль температури, тиску та рівня вологи, навіть незначні відхилення від норми можуть призвести до зниження якості продукції. Сучасні автоматизовані системи лише зменшують вплив людського фактора, але й дозволяють у режимі реального часу зберегти дані, які допомагають оперативно реагувати на будь-які зміни.

Наступним етапом є підготовка до доставки нафтопродуктів до кінцевих пунктів збуту. У цьому процесі велике значення має точне прогнозування попиту та ефективне планування маршрутів, щоб мінімізувати час і витрати на перевезення. Завдяки використанню технологій прогнозування та управління запасами компанії можуть оптимізувати ланцюг доставки, забезпечуючи своєчасну доставку та запобігаючи дефіциту продукції. Крім того, на цьому етапі випуску передбачені оптимальні способи транспортування для різних категорій пального. Наприклад, легкі фракції, як-от бензину, потребують інших умов перевезення, ніж більш в'язкі продукти, такі як мазут [5].

На завершальному етапі збуту нафтопродукти передаються в роздрібну або оптову мережу, де вже реалізуються кінцевими споживачами або промисловими підприємствами. Ефективна стратегія збуту вимагає від компаній здатності гнучко адаптуватися до ринкових коливань, отримати сезонний попит і забезпечити постачання необхідних продуктів у найбільш затребувані регіони. Важливою частиною цього етапу є маркетингова підтримка, спрямована на підвищення відомості споживачів щодо якості продукту та його особливостей, які допоможуть розширити конкурентний ринок.

Завдяки комплексному підходу до кожного з цих етапів підприємства мають можливість підвищити загальну ефективність процесів транспортування і збуту, мінімізувати операційні витрати та максимально задовольнити запитів споживачів, навіть в умовах нестабільних ринкових факторів.

1.2 Конкретизація проблем, які система повинна вирішити (наприклад, відстеження місцезнаходження транспорту, керування замовленнями)

Система обробки логістичних перевезень та збуту нафтопродуктів має на меті шкірний покрив цілу низку критичних проблем, які виникають через специфіку транспортування та продажу пального. Одним із центральних завдань є забезпечення безперервного моніторингу місця транспорту з продукцією, після

чого нафтопродукти перевозяться на далекі відстані, і будь-яке відхилення від маршруту або підтримка може призвести до порушення графіка поставок, збитків і незадоволення клієнтів. За допомогою GPS-навігації та трекінгових систем компанії можуть у реальному часі контролювати пересування кожної одиниці [6].

Іншою суттєвою проблемою є точний контроль над умовами зберігання та перевезення продукції, що є особливим значенням, тому що нафтопродукти мають певні хімічні властивості, які можуть змінюватися при температурних коливаннях або порушувати герметичність. Впровадження датчиків температури, тиску й вологості в сховищах і транспортних ємностях дозволяє уникати псування продукції та втрат через неконтрольовані умови зберігання. Ці датчики інтегровані з автоматизованими системами, які надсилають сповіщення у разі порушення встановлених параметрів, забезпечуючи постійний контроль за якістю нафтопродуктів.

Ще одним викликом є забезпечення прозорості та точності даних на всіх етапах логістики, оскільки велика кількість учасників процесу – від постачальників до кінцевих споживачів – потребує координації та обміну інформацією в режимі реального часу. Цифрові платформи не можуть централізувати цей обмін, що дає можливість користувачам з учасниками отримувати актуальну інформацію про статус завантаження, планові терміни доставки та можливості змін. Це не завершить тільки загальну ефективність операцій, але й забезпечує довіру між партнерами та знижує ризик помилок, пов’язаних із людським фактором [7].

Таким чином, комплексна система для логістики та збуту нафтопродуктів є потужним інструментом, який готовий долати ключові проблеми управління транспортом, контролю якості продукції, оптимізації запасів та забезпечення прозорості операцій, сприяючи стабільності й ефективності у всій ланці поставок.

Керування замовленнями у сфері логістики та збуту нафтопродуктів є критично важливим процесом, який забезпечує благополучне та точне задоволення потреб клієнтів. Кожне замовлення на нафтопродукти має відповідати жорстким

стандартам щодо обсягу, якості та термінів доставки, тому компанії зосереджуються на створених системах, які не тільки автоматизують обробку запитів, але й забезпечують гнучкість та адаптивність до мінливих ринкових умов. Перш за все, система керування замовленнями дозволяє компаніям оперативно реагувати на кількість запиту, аналізуючи історичні дані та прогнозуючи майбутні потреби. Це особливо важливо в галузі нафтопродуктів, де попит на паливо може змінюватися під впливом сезонних факторів, економічної ситуації або змінюватися в транспортних і промислових секторах. Процес оформлення замовлення включає ретельну координацію між усіма учасниками логістичного ланцюга, від виробників і постачальників до дистриб'юторів і кінцевих клієнтів. Компанії прагнуть забезпечити прозорість цього процесу за допомогою цифрових платформ, які дозволяють кожному учаснику отримувати доступ до актуальної інформації про статус замовлення, терміни доставки та можливість зміни в реальному часі. Такий підхід завершить довіру між партнерами, оскільки кожен може слідкувати за тим, як виконано замовлення, і планувати свої дії, виходячи з достовірних даних [8].

Безкоштовне забезпечення точності виконання кожного замовлення, системи керування замовленнями в галузі нафтопродуктів сприяють мінімізації операційних витрат. Автоматизація процесів обробки замовлень та інтеграція з іншими управлінськими системами, такими як управління запасами чи моніторинг транспорту, дозволяє уникнути надлишкових витрат на зберігання та зменшити кількість помилок через людський фактор. Крім того, завдяки розширенню аналітичних інструментів компанії можуть аналізувати продуктивність та ефективність виконання замовлень, виявляючи слабкі місця та шляхи для покращення обслуговування клієнтів. В остаточному підсумку ефективне керування замовленнями забезпечує стабільний та високий рівень обслуговування, що особливо важливо для утримання клієнтів і розширення ринкових позицій. Це створює стійку конкурентну перевагу, дозволяючи компаніям швидко адаптуватися

до змін ринку, реагувати на нові потреби клієнтів і підтримувати стабільне постачання продукції навіть в умовах коливання попиту.

Ефективне керування замовленнями в галузі нафтопродуктів включає глибоку інтеграцію з фінансовими системами та системами управління ризиками, що дозволяє компаніям утримувати фінансові зобов'язання і водночас бути гнучкими перед зовнішніми ринковими факторами. Наприклад, коливання ціни на нафту та валютні курси можуть суттєво вплинути на вартість закупівель і транспортування нафтопродуктів. Тому інтегровані системи керування замовленнями автоматично враховують актуальні ринкові умови, дозволяючи компаніям оперативно оновлювати прайс-листи, перераховувати витрати на кожне замовлення та адаптувати цінову політику під нові потреби. Це не лише оптимізує фінансові потоки, але й дає можливість клієнтам отримувати актуальні пропозиції в реальному часі, що сприяє гнучкості компанії та підвищує задоволеність клієнтів.

Ще одним аспектом керування замовленнями є аналітика поведінки клієнтів, що дозволяє компаніям краще розуміти потреби та пріоритети своїх споживачів. Завдяки збору та аналізу даних про замовлення, частоту запиту та обсяги закупівель кожен клієнт компанії може формувати індивідуальні комерційні пропозиції, пропонуючи спеціальні умови для постійних партнерів, знижки на великі обсяги або адаптовані графіки поставок. Це не тільки зміцнює ділові відносини, але й створює стійку лояльність клієнтів, які цінують персоналізований підхід і стабільність у співпраці [9].

Крім того, сучасні системи керування замовленнями включають елементи прогнозної аналітики, які дозволяють компаніям не лише відповідати на поточний запит, але й передбачати його зміни. Використовуючи алгоритми машинного навчання, ці системи можуть аналізувати тренди ринку та розраховувати великі обсяги попиту на майбутні періоди. Наприклад, з використанням історичних даних про сезонні піки попиту або зовнішні економічні фактори система може підказати, коли і на якому обсязі продукції слід готуватися, мінімізуючи ризик дефіциту чи

надлишкових запасів. Це дозволяє компаніям заздалегідь коригувати план поставок і знижувати витрати на зберігання, після планування стає більш точним і націленим на довгострокову стабільність.

Загалом, керування замовленнями в галузі нафтопродуктів є не просто процесом обробки запитів, а стратегічною складовою бізнесу, яка компенсує фінансову стійкість, адаптивність до ринкових змін і здатність передбачати майбутні потреби. Впровадження сучасних цифрових рішень дозволяє значно підвищити рівень обслуговування, оперативно реагувати на будь-які зміни та підтримувати ланцюг постачань, що особливо важливо в умовах висококонкурентного ринку нафтопродуктів [10].

1.3 Перелік функціональних (керування замовленнями, відстеження маршруту) та нефункціональних (надійність, безпека) вимог

Для системи обробки логістичних перевезень і збуту нафтопродуктів важливо врахувати комплексний набір функціональних можливостей, що забезпечують ефективне керування всіма етапами логістичного ланцюга – від отримання замовлення до його доставки. Такі системи мають бути адаптовані до особливостей галузі, враховуючи специфіку роботи з паливом, небезпечні вантажі, вимоги до безпеки та ретельного контролю всіх процесів.

Система має забезпечувати можливість створення та обробки замовлень, враховуючи різноманітність послуг, типи палива та вимоги клієнтів. Це включає в себе автоматизацію процесу прийому замовлень, їх класифікацію за типами та характеристиками, оптимізацію черги обробки для швидшого виконання. Система повинна дозволяти операторам оперативно коригувати інформацію про замовлення відповідно до змін вимог замовників, ситуаційних обставин або умов на ринку.

Важливою функцією є контроль над переміщенням вантажів. Система повинна надавати можливість не лише побудови оптимальних маршрутів, але й відстеження кожної фази перевезення, включаючи положення транспорту в

реальному часі, прогнозовані та фактичні часи прибуття. Особливо важливо контролювати дотримання умов транспортування, таких як температурний режим чи герметичність, оскільки будь-які відхилення можуть призвести до втрати якості продукту. Відстеження також дозволяє своєчасно виявляти затримки або інші відхилення, що допомагає мінімізувати ризики та забезпечити відповідальність за збереження вантажу [11].

Оскільки нафтопродукти потребують особливих умов зберігання, система повинна мати модулі управління складськими запасами, які відстежують поточні залишки, резерви та потреби у поповненні. Це включає автоматичне відображення даних про запаси та витрати, контроль за терміном придатності продуктів, а також можливість для управління операціями на складі. Система має передбачати моніторинг умов зберігання та контроль за безпекою, щоб уникнути будь-яких аварійних ситуацій.

При роботі з нафтопродуктами критично важливо забезпечити якість продуктів на всіх етапах – від закупівлі до доставки клієнту. Це включає контроль хімічного складу та властивостей нафтопродуктів, а також дотримання нормативів безпеки під час транспортування та зберігання. Система повинна мати модулі, що фіксують результати перевірок, попереджають про можливі відхилення та автоматизують процеси сертифікації продукції. Це забезпечує відповідність вимогам стандартів якості та запобігає можливим штрафам чи відкликанню продукції.

Система логістики повинна бути націлена на зниження витрат за рахунок ефективного планування ресурсів, розрахунку вартості послуг і оптимізації маршруту транспортування. Використовуючи інформацію про витрати на паливо, амортизацію транспорту та інші фактори, система дозволяє проводити аналітичний розрахунок вартості послуг та підвищувати точність бюджетного планування. Це, своєю чергою, дозволяє оптимізувати використання транспортних засобів, планувати графіки обслуговування та зменшувати загальні витрати [12].

Сучасні логістичні системи включають модулі для інтерактивної взаємодії з клієнтами, що дозволяє замовникам відслідковувати статус своїх замовлень у реальному часі. Інтерфейси самообслуговування дають змогу клієнтам отримувати інформацію про маршрут, очікуваний час доставки, відправляти запити на підтримку або змінювати деталі замовлення. Крім того, важлива інтеграція з іншими партнерами – постачальниками, страховими компаніями, митними органами – що дозволяє забезпечити безперервність процесів. Ефективна система обробки логістичних перевезень і збуту нафтопродуктів є комплексним рішенням, що охоплює всі аспекти управління поставками, контролю якості, обліку витрат та взаємодії з клієнтами.

Нефункціональні вимоги для системи обробки логістичних перевезень і збуту нафтопродуктів є не менш важливими, ніж функціональні, оскільки вони забезпечують стабільність, безпеку і надійність роботи. Враховуючи специфіку нафтопродуктів та ризики, пов’язані з їх транспортуванням та зберіганням, такі вимоги мають бути максимально строгими та всебічно забезпеченими. Надійність системи є критичним параметром, оскільки вона має забезпечувати безперервність роботи навіть у разі непередбачених ситуацій, таких як технічні збої, обриви зв’язку чи втручання ззовні. Надійність передбачає високу відмовостійкість: система повинна мати резервні сервери, дублюючі механізми збереження даних та автоматичне переключення на резервні компоненти у випадку несправності основних. Це особливо важливо для безперервного відстеження перевезень, щоб жоден із замовлень не залишився без контролю. Система повинна швидко відновлюватися після збоїв та забезпечувати захист від втрати даних навіть у форс-мажорних ситуаціях [13].

Безпека має ключове значення, з огляду на чутливість даних і ризики, пов’язані з небезпечними вантажами. Система повинна захищати всі канали передачі інформації за допомогою сучасних методів шифрування, щоб запобігти несанкціонованому доступу до конфіденційної інформації, зокрема даних про

маршрути перевезення, склади та обсяги вантажів. Інформація про клієнтів та операції також має бути захищеною відповідно до регламентів захисту персональних даних. Важливо передбачити багаторівневий захист з автентифікацією, наприклад, з використанням двохфакторної авторизації, а також моніторинг спроб доступу. Контроль прав доступу дозволяє обмежити можливості перегляду та змінювання даних залежно від ролей користувачів, запобігаючи випадкам зловживань.

Продуктивність системи повинна бути адаптованою до пікових навантажень, коли кількість запитів з боку користувачів або зовнішніх систем може значно зрости. Наприклад, у випадках масштабних відвантажень чи великих замовлень система має забезпечувати достатню швидкодію для обробки всіх запитів без затримок. Це досягається за рахунок оптимізації алгоритмів, розподіленої обробки даних та масштабованої архітектури, яка дозволяє системі швидко адаптуватися до зростання обсягу операцій. Важливо забезпечити масштабованість, щоб система могла інтегрувати нові функції або збільшувати кількість користувачів і обсяги оброблюваних даних без необхідності повної перебудови. Система повинна бути побудована на модульних принципах, що дозволяють додавати нові компоненти або розширювати потужність серверів, коли це необхідно, забезпечуючи при цьому високу сумісність з іншими системами та платформами.

Інтерфейс користувача також повинен відповідати вимогам зручності, доступності та інтуїтивності, особливо з огляду на специфіку галузі, де оператори та водії можуть мати різний рівень технічної підготовки. Інтерфейс має бути зрозумілим і швидким у використанні, з мінімальною кількістю кроків для виконання типових операцій. Це забезпечує економію часу та знижує можливість помилок, що особливо важливо в умовах термінових перевезень або небезпечних вантажів.

Функціональні вимоги до системи обробки логістичних перевезень і збуту нафтопродуктів є основою для забезпечення її стабільної роботи та відповідності

галузевим стандартам. Надійність системи є ключовим аспектом, адже від її безперебійної роботи залежить здатність компанії дотримуватися графіку поставок і гарантувати своєчасне виконання замовлень. Система має витримувати значні навантаження і забезпечувати стабільну роботу навіть у пікові періоди, коли обсяг замовлень та обробки даних може зростати. Це передбачає резервування серверів і налаштування регулярних резервних копій, що мінімізує ризики втрати інформації у разі технічних збоїв або аварій.

Безпека системи також є критичною вимогою, оскільки інформація про переміщення транспорту, статус замовлень і складські запаси є комерційно цінною та потенційно конфіденційною. Впровадження багаторівневої системи захисту даних включає використання протоколів шифрування, контроль доступу на основі ролей, а також регулярний аудит безпеки для виявлення потенційних вразливостей. Зокрема, система має захищати дані від кіберзагроз і несанкціонованого доступу, що гарантує безпеку інформації клієнтів і партнерів, знижуючи ризики витоку або маніпуляцій із даними.

Крім цього, масштабованість є важливим аспектом, оскільки система повинна бути здатна до швидкого розширення у випадку збільшення обсягу операцій чи виходу компанії на нові ринки. Система має підтримувати додавання нових користувачів, збільшення кількості замовлень і розширення функціоналу без втрати продуктивності. Це забезпечує гнучкість і дозволяє компанії адаптуватися до зростання бізнесу та змін ринкового середовища без необхідності значної перебудови інфраструктури.

Ефективність роботи системи, особливо швидкість обробки запитів і відгук на запити користувачів, також є визначальними вимогами, які впливають на продуктивність компанії в цілому. Затримки в обробці замовлень або низька швидкість завантаження даних можуть знижувати якість обслуговування і викликати незадоволення клієнтів. Оптимізація часу відгуку і налаштування для

забезпечення швидкої роботи інтерфейсу є важливими параметрами, що підвищують загальний рівень ефективності.

Нарешті, вимога до підтримуваності системи передбачає легкість її оновлення та модифікації з мінімальними перериваннями роботи. Система повинна бути розроблена так, щоб регулярні оновлення і покращення могли виконуватися без значних затримок або простоїв. Це дає можливість компанії впроваджувати нові технології та функції, залишаючись конкурентоспроможною та готовою до адаптації в умовах динамічних змін у логістиці та збуті нафтопродуктів [14].

ВИСНОВКИ ДО РОЗДІЛУ 1

При аналізі предметної області для створення системи обробки логістичних перевезень та збуту нафтопродуктів важливо розібрати весь процес логістики та збуту нафтопродуктів. Головними аспектами в ланцюгу поставок є місцезнаходження складів, оптимальна логістика та цінова політика. Саме від цих головних аспектів залежить ціна кінцевого продукту. В процес логістики та збуду нафтопродуктів відносяться наступні етапи: виробництво нафтопродуктів, або ж імпорт нафтопродуктів з-за кордону; транспортування нафтопродуктів на бази зберігання нафтопродуктів; розвезення нафтопродуктів до кінцевих споживачів. Для різних компаній необхідний різний спектр послуг, через це система надає змогу як замовляти нафтопродукти в кінцеву точку, так і налаштовувати логістику власного товару за допомогою системи.

Головними проблемами в галузі є мала автоматизація процесів виконання угод, тяжкий контроль над процесом, надійність підрядчиків та компаній та прозорість співпраці між клієнтами та торгівельними компаніями. Ці проблеми може вирішити розроблена система, що зробить процес оформлення угод більш зручним та прозорим як для клієнтів, так і для продавців.

Для вирішення головних проблем система має відповідати певним вимогам, щоб задовільнити потреби кінцевого споживача. Ці вимоги поділяються на функціональні та нефункціональні. До функціональних вимог відносяться можливість постійного керування замовлення та відстеження маршруту в реальному часі. Також важливо забезпечити нефункціональні вимоги, а саме надійність співпраці, а також безпека вантажів клієнтів.

РОЗДІЛ 2

РОЗРОБКА ІНТЕРФЕЙСУ КОРИСТУВАЧА

2.1 Основні сторінки та функціональність

Розробка інтерфейсу користувача для системи управління логістикою і збутом нафтопродуктів є складним процесом, який вимагає врахування багатьох аспектів для забезпечення зручності та ефективності роботи. Оскільки система буде використовуватися операторами, менеджерами, аналітиками та іншими фахівцями, інтерфейс має відповідати вимогам кожної з цих груп користувачів. Головна мета інтерфейсу — зробити складні процеси керування замовленнями, відстеження маршрутів, управління запасами і моніторингу транспорту інтуїтивно зрозумілими та швидкими у виконанні. Це досягається через створення чіткої структури навігації, яка дозволяє користувачам без зусиль перемикатися між функціями та отримувати доступ до всіх необхідних даних [15].

Одним із найважливіших елементів є інформаційна панель (дашборд), яка в режимі реального часу відображає ключові показники та дані: статус замовлень, місцезнаходження транспорту, рівень запасів, а також можливі відхилення або критичні події. Такий підхід дозволяє користувачам швидко оцінити ситуацію, приймати оперативні рішення та зосередитися на пріоритетних завданнях. Інформація має бути подана в стислому і зрозумілому вигляді, наприклад, за допомогою графіків, карт та індикаторів стану, які забезпечують візуальне представлення даних для швидкого аналізу. Це особливо важливо в умовах логістики, де оперативність і точність мають вирішальне значення.

Крім того, інтерфейс повинен бути адаптованим до різних пристроїв, зокрема комп’ютерів, планшетів і смартфонів, що дозволить працівникам з легкістю отримувати доступ до системи в офісі, у дорозі або безпосередньо на об'єкті. Важливим аспектом є налаштування інтерфейсу під конкретні ролі користувачів. Наприклад, диспетчери можуть мати доступ до функцій відстеження транспорту та

управління маршрутами, а менеджери з продажу — до деталей замовлень і звітів щодо збуту. Такий розподіл прав доступу робить інтерфейс більш зручним і запобігає перевантаженню непотрібною інформацією [16].

Окрему увагу приділяють функціям пошуку та фільтрації даних, що дозволяють користувачам швидко знаходити конкретні замовлення, транспортні засоби або клієнтів, відфільтровуючи інформацію за параметрами, як-от дата, маршрут, статус або тип нафтопродукту. Це значно знижує час на обробку запитів і підвищує продуктивність. Користувачам також важливо мати можливість налаштовувати інтерфейс відповідно до своїх потреб, як-от вибирати, які дані виводити на головну панель, або налаштовувати повідомлення про важливі події.

Розробка інтерфейсу передбачає ретельне тестування, щоб забезпечити його стабільну роботу та відповідність вимогам користувачів. Врахування зворотного зв'язку від операторів і менеджерів на кожному етапі розробки дозволяє виявляти недоліки та вносити корективи, покращуючи загальну зручність системи. Загалом, інтерфейс користувача в системі логістики та збуту нафтопродуктів має поєднувати функціональність, простоту та інтуїтивність, дозволяючи користувачам ефективно керувати всіма аспектами постачання та забезпечуючи високий рівень контролю за операціями.

Для створення базового інтерфейсу користувача системи обробки логістичних перевезень та збуту нафтопродуктів можна використати веб-технології, такі як HTML, CSS і JavaScript, в поєднанні з фреймворками для інтерфейсу користувача, такими як React або Vue.js. Нижче наведено приклад базової структури інтерфейсу користувача, де є панель керування з основними компонентами, такими як моніторинг статусу замовлень, відстеження транспорту та управління запасами.

Цей код демонструє основи структури, яку можна легко масштабувати та налаштовувати відповідно до більш складних вимог.

Для повноцінної системи обробки логістичних перевезень та збуту нафтопродуктів можна розробити кілька основних сторінок, кожна з яких буде виконувати специфічні функції для різних користувачів. Така система зазвичай містить панель керування, сторінки для управління замовленнями, відстеження транспорту, управління запасами, а також сторінки аналітики та звітності. Розглянемо основні сторінки та функціональність, яку вони можуть надавати [17].

1. Головна панель керування (Dashboard)

Функціональність: Це основна сторінка, що слугує як центральний центр управління, де користувачі можуть отримати загальний огляд статусу системи в реальному часі. Тут відображаються ключові показники, такі як загальний статус замовлень, рівень запасів на складі, поточне місцезнаходження транспорту, а також попередження або критичні ситуації (наприклад, затримки доставки або низький рівень запасів).

Інтерфейс: Інформаційна панель із графіками, діаграмами і показниками. Може включати візуальні індикатори стану, що показують актуальні дані, зокрема, карти для відстеження транспорту та динамічні графіки для управління запасами.

2. Управління замовленнями (Orders Management)

Функціональність: На цій сторінці користувачі можуть переглядати всі активні, завершені та заплановані замовлення. Вона дозволяє створювати нові замовлення, редагувати наявні, а також відстежувати статус їх виконання. Користувачі можуть сортувати та фільтрувати замовлення за датою, клієнтом, обсягом і статусом (наприклад, «в процесі», «готовий до відправки», «доставлений»).

Інтерфейс: Таблиця з інформацією про замовлення, з можливістю переходу до детальної інформації для кожного замовлення. Для зручності також можуть бути додані функції пошуку, сортування та фільтрації.

3. Відстеження транспорту (Transport Tracking)

Функціональність: Ця сторінка забезпечує відстеження пересування транспортних засобів у режимі реального часу. Вона дозволяє отримати інформацію про місцезнаходження кожного транспортного засобу, його маршрут, статус доставки та очікуваний час прибуття. Функціонал може також включати сповіщення у випадку, якщо транспорт відхиляється від маршруту або затримується [18].

Інтерфейс: Карта з маркерами, що позначають місцезнаходження транспортних засобів, а також панель з детальною інформацією про кожен транспортний засіб. Користувачі можуть натискати на маркери для перегляду додаткових даних, таких як водій, вантаж і пункт призначення.

4. Управління запасами (Inventory Management)

Функціональність: Сторінка управління запасами надає можливість перегляду поточного рівня запасів, відстеження стану нафтопродуктів на складах, а також аналізу історії використання та поповнення. Функціонал дозволяє планувати запаси на основі прогнозів попиту та автоматично формувати замовлення на поповнення, якщо запаси досягають критично низького рівня.

Інтерфейс: Діаграми та графіки, що відображають рівень запасів з можливістю перегляду історичних даних та прогнозів. Список складів і поточного стану продукції на кожному з них, а також індикатори для низьких запасів або критичних рівнів.

5. Налаштування користувача та ролей (User and Role Management)

Функціональність: Ця сторінка дозволяє адміністраторам системи керувати обліковими записами користувачів, призначати права доступу та ролі. Вона надає можливість обмежити доступ до певних функцій системи залежно від ролі користувача, що підвищує безпеку та контроль.

Інтерфейс: Форми для додавання нових користувачів, вибору їх ролей і встановлення рівня доступу. Список активних користувачів із можливістю редагування або видалення їхніх профілів.

6. Сповіщення та попередження (Notifications and Alerts)

Функціональність: Система сповіщень надає користувачам інформацію про важливі події, як затримки доставки, відхилення від маршруту, низький рівень запасів або невідповідність умов зберігання. Сповіщення можуть бути інтегровані з електронною поштою або месенджерами для миттєвого інформування відповідальних осіб.

Інтерфейс: Панель сповіщень з коротким описом події та посиланнями на детальну інформацію. Користувачі можуть налаштовувати типи сповіщень, які вони бажають отримувати.

2.2 Панель адміністратора для управління замовленнями та маршрутами

Панель адміністратора для управління замовленнями та маршрутами є ключовим інструментом, який забезпечує централізований контроль над усіма логістичними процесами, від обробки замовлень до координації маршрутів транспорту. Цей інтерфейс дозволяє адміністраторам легко контролювати та керувати кожним замовленням, надаючи можливість відстежувати його статус, терміни виконання та будь-які зміни, які можуть виникнути під час виконання. Панель має бути інтуїтивно зрозумілою і добре організованою, щоб адміністратори могли швидко знаходити інформацію і своєчасно реагувати на будь-які затримки чи зміни в процесі доставки.

Основна функціональна панель адміністратора включає можливість створення, редагування та видалення замовлень, що дозволяє адаптувати логістичний процес до потреб клієнтів або зовнішніх факторів, таких як зміна запиту чи форс-мажорні матеріали. Адміністратор може побачити деталі кожного замовлення: інформацію про клієнта, кількість замовленого продукту, кінцеву точку доставки та заплановані дати. Це гарантує збереження повної картини

замовлень та пріоритетність їх виконання, гарантуючи, що найбільш термінові замовлення обробляються своєчасно [20].

Крім управління замовленнями, панель надає адміністратору можливість контролювати маршрути транспорту в режимі реального часу. За допомогою інтегрованих карт, які показують поточне місце кожного транспортного потоку, адміністратор може відстежувати маршрут і тимчасово виявляти будь-які відхилення. Якщо транспортний засіб виконується через дорожні умови або технічні несправності, панель дозволяє оперативно перенаправити інший транспорт для виконання замовлення або повідомити клієнта про можливу затримку. Такий контроль рівня над маршрутами мінімізує ризики збоїв у постачанні та знижує операційні витрати, пов'язані із запізненнями.

Окрему увагу приділено оптимізації маршрутів, яка базується на аналізі попередніх даних та поточних умов руху. Панель адміністратора може давати рекомендації щодо найкращих маршрутів, враховуючи затори, стан доріг або інші фактори, які можуть включити на час доставки. Це дозволить зменшити витрати на паливо та прискорити процес доставки, що позитивно впливає на загальну продуктивність логістичної мережі.

Також панель адміністратора може включати модуль аналітики, що дозволяє аналізувати ефективність роботи системи в цілому. Адміністратор має доступ до звітів про середній час виконання замовлень, частоту затримок і фактори, які на них впливають. Це надає дані для прийняття стратегічних рішень, таких як перегляд розподілу ресурсів або зміни логістичної стратегії для підвищення ефективності.

Таким чином, панель адміністратора є потужним інструментом, що забезпечує прозорість, контроль і адаптивність у процесі управління замовленнями та маршрутами. Вона дозволяє адміністраторам підтримувати високу якість обслуговування, реагувати на зміни в режимі реального часу та оптимізувати використання ресурсів, забезпечуючи стабільність та ефективність усіх логістичних операцій [21].

2.3 Відображення статусів замовлень та транспорту на карті

Для ефективного керування логістикою в системі випускається відображення статусів замовлень і маршруту транспорту на інтегрованій карті. Така карта дозволяє адміністраторам у реальному часі контролювати сайт транспортних засобів, забезпечуючи видимість усіх ланцюгових поставок. Адміністратор може відразу побачити лише те, що знаходиться кожен транспортний засіб, але й статус його доставки, що відображається у маркерах із відповідними кольорами або індикаторами, які вказують на стан замовлення – наприклад, «в дорозі», «затримка», «доставлено». .

На карті кожне замовлення відображається як точка маршруту або маркер із детальною інформацією, включаючи статус вантажу, місце призначення, час прибуття та відомості про замовника. Адміністратори можуть натискати на маркери, щоб побачити розширену інформацію про замовлення, включаючи тип нафтопродукту, контактну інформацію водія та інші важливі дані, які допоможуть прийняти оперативні рішення. Наприклад, якщо на карті позначено затримку або відхилення від маршруту, адміністратор може оперативно розглянути альтернативні варіанти: скоригувати маршрут, вказати інший транспорт або повідомити клієнта про зміну графіка доставки [22].

Додатково функція відображення маршруту на карті дає можливість планувати оптимальний шлях, мінімізуючи витрати на паливо та скорочуючи час доставки. Система може автоматично розраховувати найвидіші маршрути, враховуючи поточний трафік, погодні умови і стан дороги, забезпечуючи найбільш ефективний шлях для кожного транспорту. Це дозволяє запобігти простою транспорту в заторах або на неефективних маршрутах, зменшуючи витрати та підвищуючи продуктивність.

Крім того, карта може бути інтегрована з модулем сповіщення, що повідомить адміністратора про критичні ситуації. Якщо транспортний засіб значно відхилився від маршруту або завантажився, система зараз надсилає сповіщення, надаючи адміністратору можливість тимчасово вжити заходів для утворення зриву поставок. Це завершиться точністю і швидкістю логістичних процесів, забезпечуючи більшу прозорість і контроль за всіма етапами доставки [23].

Такий підхід до інтеграції статусів замовлень і маршрутів транспорту на карті не тільки полегшує управління, але й покращує досвід клієнтів, дозволяючи компанії оперативно реагувати на будь-які зміни або затримки, забезпечуючи стабільність і ефективність доставки на ринок нафтопродуктів.

2.4 Поняття зручності та інтуїтивності інтерфейсу

Інтуїтивність інтерфейсу означає, що користувачі можуть легко зрозуміти, як працює система, навіть при першому використанні. Інтерфейс повинен відповідати їхньому очікуванню, заснованому на загальних принципах взаємодії з іншими програмами. Наприклад, кнопки, що забезпечують певні дії, мають бути чітко позначені та розміщені в місцях, де користувачі очікують їх знайти. Кольорові акценти та графічні елементи застосовуються для привернення уваги до важливих функцій або попередження, що дозволяє користувачам легко орієнтуватися в процесах та тимчасово реагувати на зміни.

Ключовим аспектом зручності та інтуїтивності є також забезпечення болю в дизайні. Якщо кожен розділ системи використовує однакові стилі, шрифти, кольори та шаблони розміщення, користувачі швидко адаптуються, розуміють, що чекати від інтерфейсу. Користувацький досвід стає передбачуваним, і це значно знижує стрес під час роботи з новою системою. Наприклад, якщо в усіх розділах кнопка «Зберегти» розташована в одному й тому місці та має один вигляд, користувачі швидко запам’ятають її розташування та більше не платять час на її пошук.

Проста і зрозуміла структура також важлива для зручності та інтуїтивності, оскільки вона дає користувачам можливість зосередитися на завданнях, а не на тому, як користуватися інструментом. Використання зрозумілих і коротких описів, читабельних шрифтів і відповідних іконок дозволяє швидко орієнтуватися в системі. Інтуїтивний інтерфейс зменшує кількість помилок, оскільки користувачі мають доступ до підказок і допоміжної інформації, коли їм це потрібно, що мінімізує потреби звертатися за допомогою або шукати інструкції [25].

Загалом, зручність та інтуїтивність є запорукою успішної роботи з інтерфейсом, оскільки вони стали користувачам ефективно виконувати свої завдання, не відмовляючись від складнощів у навігації чи інтерпретації функцій. Це задоволеність користувачів, підвищення їхньої продуктивності та створення враження від роботи з системою, яке є вирішальним для успішного впровадження будь-якого програмного продукту.

Додатково, зручність та інтуїтивність інтерфейсу охоплюють не тільки легкість у використанні, але й створення глибокого зв'язку між системою та користувачем, що виходить за межі базових функцій. Зручний інтерфейс адаптується до реальних сценаріїв роботи користувача: система враховує природні кроки, які він виконує у своїй повній діяльності, і пропонує саме ті інструменти, які йому потрібні в певний момент. Наприклад, для інтерфейсу логістичної системи необхідно забезпечити контекстні підказки – якщо користувач переглядає інформацію про замовлення, система автоматично пропонує додаткові дії, як-от підтвердження, перенесення або сповіщення клієнта про зміни. Це створює відчуття підтримки та зменшує когнітивні завантаження, оскільки користувачеві не потрібно щоразу шукати відповідні функції або згадувати процеси.

Інтуїтивність тісно пов'язана з принципом передбачуваності. Інтерфейс, який побудований на основі загальноприйнятих стандартів дизайну, формує відчуття довіри у користувача. Наприклад, такі елементи, як кнопки, іконки та панелі, що відповідають загальним правилам дизайну, дозволяють користувачу інтуїтивно

розуміти їх функції, які відсутні в новій системі. Коли інтерфейс побудований з урахуванням цих очікувань, користувачі можуть передбачити, що буде далі, і це значно зменшує стрес, пов'язаний із адаптацією до нового середовища. Важливо, що ці передбачення базуються на мінімально необхідному досвіді, тому користувач може почати роботу з інтерфейсом, не витрачаючи час на тривале навчання або відвідування з системою [26].

Інтерфейс також має сприяти «плавній адаптації» – нові користувачі повинні швидко відчувати себе комфортно, навіть якщо не мають досвіду роботи з конкретною системою. Елементи, як-от інтерактивні підказки, покрокові інструкції або «демонстраційний режим» для новачків, допомагають користувачам швидко перейти до основних функцій, відчуваючи себе при цій підтримці. Ця функція важлива для великих корпоративних систем, де навчання користувачів може забирати багато часу і ресурсів, а належне самонавчання економити час і дозволяє новим працівникам ефективніше включатися в роботу.

Важливою складовою є також функція доступності, коли інтерфейс розроблений з клієнтськими потребами користувачів з підвищеними можливостями, наприклад, людей з обмеженими можливостями. Інтерфейс, що відповідає стандартам доступності (як-от WCAG), виділяє чіткі контрастні кольори, доступність елементів управління для використання з клавіатурою та зручну навігацію для людей із порушеннями зору чи слуху. Система, яка відповідає вимогам доступності, розширює аудиторію користувачів і дозволяє різним людям працювати на однаковому рівні, що забезпечує інклюзивність і створює комфортні умови для всіх співробітників, незалежно від їхніх фізичних можливостей.

Добре розроблений інтерфейс виходить за рамки стандартних принципів зручності та інтуїтивності – він стає частиною робочого процесу, підходить під потреби користувача і робить взаємодію із системою природною, навіть максимально самою, якщо система складна. Це зараз до більшої продуктивності,

задоволення від роботи та впевненість у тому, що система підтримує користувача на кожному етапі роботи [27].

ВИСНОВКИ ДО РОЗДІЛУ 2

Отже, для розробки інтерфейсу користувача, необхідно розробити основні сторінки системи та налаштування їх функціональності для клієнтів, створити панель адміністратора для управління замовленнями та маршрутами для оформлення угод, налаштувати відображення статусів замовлень та відстеження транспорту в реальному часі, та головне – забезпечити зручність та інтуїтивність інтерфейсу для кращого опанування з боку споживачів.

До основних сторінок мають відноситись наступні: головна панель керування (головна сторінка, яка має забезпечити центральне управління), управління замовленнями (для створення, редагування та видалення замовлень, а також для перегляду статусу замовлень), відстеження транспорту (забезпечення відстеження пересування транспортних засобів у режимі реального часу), налаштування користувача та ролей (для керування обліковими записами користувачів), сповіщення та попередження(для надання інформації користувачам про важливі події).

Панель адміністратора для управління замовленнями та маршрутами забезпечує централізований контроль над усіма логістичними процесами. Він має дозволяти адміністратору контролювати та керувати усіма замовленнями на платформі, а також виконувати зміни в реальному часі. Тобто, до головних функцій адміністратора відноситься створення, редагування замовлень. Окрім цього адміністратор має можливість отримати всю необхідну інформацію про користувачів та про замовлення. Також адміністратор повинен мати можливість змінювати бази даних користувачів, продуктів та замовлень.

Інтуїтивність та зручність є головним аспектом для створення системи, оскільки раніше подібних систем не існувало, через що для споживачів даний продукт буде новим для використання. Саме зручність та інтуїтивність можуть спростити адаптацію нових користувачів до системи.

РОЗДІЛ 3

ТЕХНІЧНА РЕАЛІЗАЦІЯ

3.1 Вибір технологій та інструментів. Вибір мови програмування, бази даних (наприклад, PostgreSQL або MySQL), а також інструментів для фронтенду (React або Angular)

Технічна реалізація інтерфейсу користувача для логістичних систем з управління замовленнями та відстеження транспорту вимагає інтеграції кількох компонентів, які забезпечують стабільність, швидкість і функціональність. Одним із ключових аспектів є вибір архітектури: всім клієнт-серверну архітектуру, де фронтенд (інтерфейс користувача) працює як незалежний клієнт, який запитує і відображається з бекенду через API. Це дозволяє розділити обробку і зберігання даних, які можуть вимагати значних ресурсів, від візуальної частини системи, яка взаємодіє з користувачем.

Для фронтенду необхідно використовувати сучасні JavaScript-фреймворки, такі як React або Vue.js, які можуть створювати динамічні інтерфейси з інтерактивними компонентами. Компонентна структура цих фреймворків дозволяє розробляти окремі функціональні блоки — наприклад, панель відстеження замовлень або модуль управління запасами — як незалежні модулі, які можна повторно використовувати. Це також полегшує тестування та обслуговування коду, оскільки кожен компонент має чітко визначені параметри та функції. Реалізація інтерактивних карток для відстеження транспорту фактично здійснюється з використанням бібліотеки Leaflet або Google Maps API, що забезпечує користувачам візуальне уявлення про маршрут транспорту та статус доставки [28].

Бекенд, або серверна частина, часто реалізується на Node.js або Python, де сервери обробляють запити клієнтів, зберігають інформацію про замовлення, транспорт і маршрути, а також забезпечують основну логіку системи. API забезпечує зв’язок між фронтендом і бекендом, дозволяючи користувальному інтерфейсу запитувати дані та надсилати зміни, які оновлюються в базі даних.

Зазвичай використовувати RESTful або GraphQL API, щоб забезпечити швидкий обмін інформацією між клієнтом і сервером та мінімізувати обсяг даних, що передаються. Також важливе значення має кешування — наприклад, за допомогою Redis або Memcached зменшує завантаження на сервер і прискорює відповіді на часті запити.

Щодо бази даних, для управління логістичною системою доцільно використовувати реляційні бази даних, такі як PostgreSQL або MySQL, для структурованих даних про замовлення, клієнтів і продукти. Для великих обсягів нереляційних даних (наприклад, логів відстеження руху транспорту) можна використовувати базу NoSQL, як MongoDB, що дозволить гнучко обробити неструктуровані дані. Якщо система повинна працювати в реальному часі, важливу роль обробляє дані в потоковому режимі. Наприклад, платформи на зразок Apache Kafka допомагають передавати повідомлення між компонентами системи, підвищуючи високу продуктивність і можливість роботи з великим потоком даних без затримок.

Безпека є критичним аспектом технічної реалізації, оскільки система логістики часто обробляє конфіденційну інформацію. Для цього протоколу шифрування HTTPS і SSL для забезпечення захищеного з'єднання між користувачем і сервером. Крім того, для контролю доступу потрібно використовувати системи аутентифікації та авторизації, наприклад JWT (JSON Web Tokens) або OAuth, що гарантує доступ до системи лише авторизованим користувачам. такі заходи захищають дані від несанкціонованого доступу і підвищують надійність системи [29].

Завдяки грамотній реалізації інтерфейс цієї архітектури працює з мінімальними підтримками, підтримуючи стабільну роботу навіть при великому завантаженні. Технічні рішення, які лежать на основі системи, дозволяють створити надійну, швидку і безпечну платформу для логістичних операцій, яка легко адаптується до нових вимог та інтегрується з додатковими модулями і функціями.

Вибір технологій та інструментів для створення інтерфейсу логістичної системи має критичне значення для забезпечення її стабільності, продуктивності та можливості масштабування. Починається все з визначення потреби користувачів і типів завдань, які вони відмовляють у системі. Наприклад, у системі управління замовленнями та транспортом користувачі потребують швидкого доступу до даних у реальному часі, просто в обробці запитів і можливості виконувати дії без затримок. Для досягнення цієї мети виберіть технологію, яка підтримує високу продуктивність і забезпечує ефективну обробку даних.

На етапі розробки фронтенду вибір часто припадає на сучасні JavaScript-фреймворки, як-от React, Vue.js або Angular, які створюють динамічні та інтерактивні інтерфейси. React часто стає першим вибором через його продуктивність, розширюваність та багату екосистему. Використання компонентної структури дозволяє розробникам розділити складні інтерфейси на невеликі частини, які можна тестувати та вдосконалювати окремо. Це підхід завдяки повторному використанню компонентів, що є у великих логістичних системах, де багато модулів мають такі ж функції, як і управління даними про замовлення та маршрути [30].

Для відображення географічних даних, які мають важливе значення для відстеження маршрутів транспорту, використовують бібліотеки картографії, такі як Leaflet чи Mapbox. Листівка є легкою та продуктивною бібліотекою, яка швидко завантажується та має інтуїтивно зрозумілу документацію. Вона дозволяє інтегрувати карти, додавати маркери для кожного транспортного засобу, прокладати маршрути та відображати статуси. Mapbox, у свою чергу, пропонує більш розширені можливості, зокрема у випадку роботи з великими обсягами даних або візуалізацією складних маршрутів, і може бути адаптований для потреб масштабних проектів.

Бекенд системи, відповідні для обробки логіки та зберігання даних, часто реалізуються на Node.js або Python (зокрема, на фреймворку Django або Flask для

Python), остання помилкова технологія добре масштабується і підтримує високі завантаження. Node.js забезпечує асинхронну обробку, що є критично важливим для системи з інтенсивним використанням запитів у реальному часі, зокрема коли йдеться про оновлення маршруту даних. Для швидкої обробки великих обсягів інформації та передачі даних між компонентами системи можуть використовуватися брокери, такі як Apache Kafka або RabbitMQ, які підтримують стабільний потік даних, необхідний для логістичної системи.

Вибір бази даних базується на природі даних. Реляційні бази, такі як PostgreSQL, добре підходять для зберігання структурованих даних, як-от інформації про замовлення, клієнтів та продукти. Ця база забезпечує надійність транзакцій і високу швидкість запитів, що ідеально підходить для критично важливих бізнес-даних. У випадку зберігання неструктурованих або великих обсягів даних, таких як журнали трекінгу транспорту чи інформації про маршрути, використовується база NoSQL, наприклад MongoDB, яка швидко працює з великими масивами даних і дозволяє легко розширюватися з ростом завантаження. Завдяки гнучкості в зберіганні бази даних NoSQL оптимізують обробку запитів, що значно забезпечує швидкість доступу до історичних даних і забезпечує надійну підтримку системи з аналітичними завданнями [31].

Забезпечення безпеки даних і доступу до системи також вибір технологій. Протоколи безпеки, такі як HTTPS і SSL, захищають з'єднання між користувачем і сервером, а також гарантують конфіденційність даних. Аутентифікація користувачів часто реалізується через JSON Web Tokens (JWT), що забезпечує швидкий і захищений доступ користувачів до системи, дозволяючи одночасно контролювати ролі та рівні доступу.

Загальний успіх вибору технологій та інструментів залежить від їх здатності підтримувати, забезпечувати масштабну продуктивність і безпеку. Тому важливо, щоб усі компоненти були взаємодоповнюваними, дозволяючи гнучко адаптувати

систему до збільшення вимог і нових викликів, які постають у сфері логістики та управління замовленнями.

Вибір мови програмування для розробки логістичних систем з управління замовленнями та відстеженням маршруту транспорту обумовлений потребою забезпечити високу продуктивність, масштабованість та стабільність. У подібних системах часто використовують мови, які здатні обробляти великі обсяги даних і підтримувати асинхронну архітектуру. JavaScript (зокрема, платформа Node.js) та Python є популярними виборами завдяки своїй універсальності й здатності обробляти запити в режимі реального часу.

Node.js особливо підходить для завдань, де необхідна обробка в режимі реального часу, після чого його архітектура заснована на неблокуючих подіях і асинхронно введених/виведених, що робить його ідеальним для обробки одночасних запитів від користувачів. Це дозволяє системі ефективно обслуговувати кількість клієнтів, які можуть одночасно відслідковувати сайт транспорту або переглядати статус замовлень. Node.js має активну екосистему пакетів (npm), що дозволяє швидко інтегрувати готові рішення для таких завдань, як управління маршрутами, підключення до бази даних або обробки даних користувачів [32].

Python, зокрема з використанням фреймворків Django або Flask, надає можливість створювати серверну частину систем, де необхідна висока продуктивність та надійність. Python пропонує потужні бібліотеки для обробки даних, машинного навчання й аналітики, що дозволяє інтегрувати інтелектуальні алгоритми оптимізації маршрутів і прогнозування запиту на основі даних минулих періодів. Django надає комплексний підхід до розробки, вбудовуючи в себе безліч корисних функцій, зокрема ORM (Object-Relational Mapping), що сприяє інтеграції з базою даних і значно прискорює розробку.

Щодо бази даних, вибір між PostgreSQL і MySQL залежить від сумісності та вимог до зберігання даних у проекті. PostgreSQL часто розглядається як кращий вибір у системах, де важлива складна логіка запитів, підтримка транзакцій і

дотримання строгих умов цільності даних. Наприклад, для логістичної системи важливо гарантувати, що дані про замовлення й маршрути завжди будуть цільними, що завершує виконання функцій на оперативність і точність процесів. PostgreSQL має потужні засоби для обробки складних запитів, підтримує JSON-дані та географічні координати (через розширення PostGIS), що дозволяє ефективно зберігати і використовувати дані для відстеження маршруту на карті. Це робить його універсальним вибором, який забезпечує надійність та гнучкість при роботі з великою кількістю різнотипних даних.

MySQL, у свою чергу, знає свою високу швидкість і стабільність при обробці великої кількості простих запитів, які можуть бути корисними, якщо акцент робиться на швидкій обробці транзакцій. Це особливо важливо в сценаріях, де необхідно постійно підтримувати оновлення статусу замовлень і відстеження місця транспорту. MySQL, як правило, підходить для проєктів, де не потрібна надана аналітика, але важливо швидко виконувати операції з читання та запису даних, як оновлення статусу замовлень або обробка великих обсягів клієнтських запитів. Рішення про вибір мови програмування та бази даних можна залежати від потреби у масштабованості. Наприклад, PostgreSQL добре виправляється зі збереженням великих обсягів складських даних, забезпечуючи безперервність роботи навіть за умов великого завантаження, що робить його ідеальним для проектів із деяким зростанням даних. Якщо в основному призначено швидке обслуговування й обробка величезної кількості запитів у реальному часі, то MySQL може мати оптимальний вибір завдяки швидкості та простоті у використанні. Вибір мови програмування і бази даних тісно пов'язаний із потребами конкретної системи та обсягами даних, з якими вона працює. Використання Node.js або Python на бекенді, в поєднанні з PostgreSQL або MySQL для зберігання, забезпечує надійність, масштабованість і гнучкість, необхідну для ефективної роботи системи управління замовленнями і логістикою [33].

Вибір інструментів для фронтенду, таких як React чи Angular, залежить від багатьох факторів, включаючи складність проекту, потребу в інтерактивності, а також вимоги до швидкості та зручності оновлення інтерфейсу. У логістичних системах з управління замовленнями й відстеженням транспорту важливо, щоб інтерфейс був здатний швидко реагувати на зміни даних у реальному часі, а також забезпечував просту навігацію для користувачів. React і Angular є провідними технологіями для досягнення цих цілей, хоча кожна з них має свої підходи і переваги.

React, розроблений Facebook, виділяється своїм компонентним підходом і гнучкістю, що дозволяє створювати динамічні інтерфейси з легкістю в масштабуванні. Компонентна архітектура React надає розробникам можливість розділити складні інтерфейси на окремих, повторно використаних компонентах, що забезпечує прозорість і спрощує розробку. Наприклад, для логістичних систем можна створити такі компоненти, як «Список замовлень», «Карта відстеження» або «Панель сповіщень», і кожен з них функціонуватиме незалежно, що сприяє управлінню їхнім станом. React також підтримує «односторонню передачу даних», завдяки чому кожен компонент отримує дані з центрального джерела, а будь-які зміни в системі миттєво відображаються в інтерфейсі без потреби в перезавантажених сторінках. Це важливо для логістичної системи, де користувачі мають швидкий доступ до інформації про статуси замовлень або маршрути транспорту [34].

React також має сильну екосистему інструментів і бібліотек, таких як Redux для управління станом або React Router для побудови навігації, що дозволяє створювати складні інтерфейси з великою кількістю інтерактивних елементів. Така система управління станом стає критично важливою в логістичних об’єктах, що дозволяє забезпечити синхронізацію даних у різних частинах інтерфейсу. Завдяки React можна ефективно керувати оновленнями даних, що дає можливість

відображати оновлені маршрути на карті чи нові замовлення без необхідності перезавантаження сторінок, що покращує загальний досвід користувача.

Angular, розроблений Google, також є потужним інструментом для побудови великих і складних веб-додатків. Це повноцінний фреймворк, який надає розробникам інтегроване середовище для створення функціональних і масштабованих інтерфейсів. Angular відомий своєю двосторонньою прив'язкою даних (двостороння прив'язка даних), яка дозволяє автоматично синхронізувати дані між моделлю та відображенням в інтерфейсі. Це означає, що будь-які зміни, внесені в інтерфейси, мить відображаються в моделі і навпаки. У логістичному контексті двостороння зв'язка може бути корисною, якщо потрібно автоматично оновлювати інформацію про або відображати зміни в режимі замовлення реального часу. Наприклад, зміни в статусі замовлення можуть миттєво з'явитися на панелі диспетчера, що забезпечує швидкість і точність прийнятих рішень.

Angular також пропонує модульну структуру, яка полегшує управління великими об’єктами розподілу та коду на зрозумілі частини, які можуть обробляти різні команди розробників. Це стає цінним у великих корпоративних проектах, де інтерфейс може мати різні модулі, як і окремі панелі для адміністраторів, диспетчерів і водіїв. Angular надає вбудовані інструменти для тестування, зокрема Jasmine і Karma, що покращують стабільність і надійність програм, завдяки процесу розробки інтерактивних компонентів можна швидко перевірити їх функціональність. Крім того, фреймворк Angular містить широкий набір інструментів для забезпечення безпеки та доступності, що є для системи, що обробляє конфіденційні дані замовлень та маршрутів.

React і Angular також мають переваги в продуктивності, завдяки використанню віртуального DOM в React і вдосконаленого механізму зміни виявлення в Angular. Це дозволяє зменшити обсяг рендеринга сторінок, забезпечуючи швидке відображення інтерфейсу та мінімізуючи затримку. Крім того, обидва фреймворки добре підтримують інтеграцію з API та бекендом, що

забезпечує швидкий обмін інформацією з сервером, таким як статус транспорту чи замовлення нових.

Таким чином, вибір між React і Angular залежить від конкретних вимог проєкту та уподібнення команди розробників. React забезпечує гнучкість і зосередженість на швидкій роботі з його динамічними компонентами, що робить оптимальним для інтерфейсів, де потрібна швидка взаємодія з даними. Angular, з його вбудованою структурою і розширеними можливостями, підходить для великих проектів, де важлива модульність і повноцінне середовище для розробки. У будь-якому випадку помилкові інструменти не можуть створити функціональні, інтуїтивно зрозумілі та надійні інтерфейси для системи логістики [35].

3.2. Реалізація бекенду. API для обміну даними. Впровадження логіки управління замовленнями, моніторингом і маршрутизацією

Реалізація бекенда для логістичних систем є основою, яка забезпечує обробку всіх критично важливих операцій, таких як управління замовленнями, моніторинг транспорту в реальному часі, оновлення статусів, а також обробку запитів від клієнтів і партнерів. Бекенд складається з серверної додатки, бази даних і API, що дозволяє обмінюватися даними з фронтендом, забезпечуючи безперервний потік інформації та інтеграцію різних модулів систем.

На етапі розробки серверної додатки важливо вибрати платформу, яка забезпечує масштабованість, продуктивність і здатність обробляти одночасні запити. Для цих цілей підходить Node.js, оскільки він працює на базі неблокуючої асинхронної архітектури, що дозволяє ефективно обслуговувати велику кількість користувачів одночасно. Це особливо важливо для логістичних систем, де всі користувачі можуть постійно оновлювати статуси замовлень і отримувати інформацію про маршрути в реальному часі. Кожен запит у Node.js обробляється окремим потоком, що дозволяє уникати блокувань при високому завантаженні та забезпечує стабільну роботу системи.

На сервері реалізуються API-ендпоїнти, які дозволяють фронтенду взаємодіяти з базою даних і отримувати оновлені дані. Наприклад, можна створити окремі ендпоінті для отримання списку замовлень, оновлення статусу конкретного замовлення, створення маршруту транспорту або додавання нового клієнта. Кожен ендпоінт повинен обробляти запити, перевіряти вхідні дані на відповідність критеріям безпеки та валідації, і лише після виконання цієї операції в базі даних. Така багаторівнева структура дозволяє знизити ризики помилок і забезпечити стабільність системи [36].

Для зберігання даних замовлення, маршрутизації та клієнтів бекенд підключається до реляційної бази даних, наприклад, PostgreSQL. PostgreSQL забезпечує надійність і високий рівень транзакційної обробки даних, що важливо для логістичних систем, де кожна операція — це критичний крок у ланцюзі постачань. Наприклад, якщо в системі реєструється нове замовлення, PostgreSQL дозволяє зберегти цю транзакцію в цілісному вигляді, з усіма пов’язаними даними, такими як клієнт, пункт призначення, статус доставки тощо. Для швидкого доступу до інформації та спрощення запитів можна налаштувати індекси на основних таблицях (наприклад, за статусом замовлення чи ID транспорту), що значно прискорює запити.

Щоб підтримувати обмін даними між компонентами бекенда та фронтенду, API можна реалізовувати на основі REST або GraphQL. REST-API є традиційним підходом, що дозволяє структурувати ендпоінті для логічних операцій (наприклад, створення, читання, оновлення та видалення даних), забезпечуючи стандартизовану взаємодію між компонентами. GraphQL, у свою чергу, забезпечує більшу гнучкість, не дозволяє клієнтам запитувати лише ті дані, які їм потрібні, і отримувати їх у форматах, які економлять обсяг передачі даних. Для логістичної системи GraphQL може бути особливо корисним, не дає можливість динамічно отримувати інформацію про маршрути, які змінюються в реальному часі [37].

Забезпечення безпеки бекенда є великим аспектом, особливо в системах, де обробляються конфіденційні дані про клієнтів і маршрути. Реалізація безпеки включає а

Крім цього, для забезпечення високої продуктивності бекенд-сервера можна використовувати кешування. Кешування за допомогою Redis або Memcached дозволяє зберегти часто запитувані дані, такі як замовлені статуси або інформацію про маршрути, що зменшує завантаження в базу даних і значно пришвидшує системні відповіді на запити. Це особливо корисно в реальних логістичних проектах, де дані можуть оновлюватися рідше, а запити від користувачів є частинами.

Також важливо забезпечити масштабованість бекенда для підтримки розширення користувачів і обсягів даних. цього можна досягти, розгортаючи додаткові екземпляри сервера або використовуючи контейнеризацію (наприклад, Docker), що дозволяє створювати легко масштабовані інфраструктури. Інтеграція із системами управління контейнерами, як Kubernetes, дозволяє автоматично балансувати завантаження та розподіляти ресурси для забезпечення стабільної роботи системи.

Реалізація бекенда для логістичної системи вимагає уваги до швидкості, надійності й захищеності кожного компонента, щоб забезпечити злагоджену роботу в усіх системах та підтримувати високу продуктивність і точність обробки даних.

API для обміну даними в логістичній системі з управління замовленнями та відстеженням транспорту виконує роль «посередника», забезпечуючи взаємодію між фронтендом і бекендом, а також підтримуючи безперервний потік даних між системами ефективних модулів. У логістичному середовищі, де обробка інформації в реальному часі є критичною, API відповідає за надання швидких відповідей на запити користувачів, відображення актуальних статусів замовлень і точного сайту транспорту.

При розробці API необхідно врахувати структуру даних і логіку, яка відповідає специфікі логістичних процесів. Наприклад, для системи управління замовленнями API дозволяє фронтенду запитувати дані про нові замовлення, статуси існуючих замовлень, а також інформацію про кожен етап доставки. Коли адміністратор або запитує клієнт статус замовлення, API звертається до бази даних для отримання актуальної інформації та повертає ці дані фронтенду у форматі, готовому для візуалізації. Це забезпечує прозорість ланцюга поставок і надає користувачам актуальні дані про прогрес доставки [38].

Для управління маршрутним транспортом API можна взаємодіяти з картографічними сервісами, як-от Google Maps або Mapbox, щоб забезпечити відображення поточного місця транспортної інформації на карті. У цьому випадку API використовує дані GPS або трекери, які передають дані про координати транспорту на сервері в реальному часі. API обробляє ці координати, оновлює інформацію в базі даних і передає її фронтенду, де користувачі можуть спостерігати за рухом транспорту. Для цього можна використовувати модель push-повідомлень або веб-сокетів, які забезпечують постійне оновлення даних без необхідності повторного завантаження сторінок.

Архітектура API робить, чим передаються запити і відповіді між клієнтом і сервером. У логістичних системах часто вимагається REST-архітектура для будівництва API, де кожен ресурс має свій унікальний ендпоінт (наприклад, /ordersдля замовлення або /trackingдля трекінгу транспорту), який дозволяє користувачам отримувати або змінювати дані через стандартні HTTP-запити (GET, POST, PUT, DELETE).). REST підходить для логістичних систем, забезпечує структуру запитів і відповідей в єдиному, стандартизованому форматі JSON, що полегшує інтеграцію різних систем і забезпечує безпечний обмін даними.

Іншим підходом до реалізації API є використання GraphQL, що дозволяє одночасно запитувати незадоволені дані та формувати складні запити для кількох ресурсів. Наприклад, якщо користувач хоче отримати детальну інформацію про

замовлення разом із відомостями про маршрут транспорту, GraphQL дозволяє зробити цей один запит замість того, щоб надсилати кілька запитів через REST. Це значно зменшує навантаження на сервер і обсяг переданих даних, що особливо важливо в системах з інтенсивним трафіком [39].

Безпека API є критичною частиною обміну даними в логістичній системі, де обробляються конфіденційні дані про замовлення та маршрути. Для захисту API використовує токени аутентифікації, такі як JWT (JSON Web Tokens), що забезпечує перевірку правового доступу користувачів. Після входу в систему користувач отримує токен, який додається до кожного запиту до API, що дозволяє серверу перевіряти автентичність і користувача. Це дозволяє обмежити доступ до різних функцій системи в залежності від рівня правого користувача (наприклад, клієнти можуть бачити тільки своє замовлення, а адміністратор – усі замовлення).

API також підтримує обробку великих обсягів даних через пагінацію та фільтрацію. Для запитів, які повертають довгі списки записів (наприклад, усі замовлення за останній місяць або всі поточні маршрути транспорту), API реалізує розбивку на сторінках і дозволяє фільтрувати дані за певними параметрами, такими як статус замовлення, дані чи призначення пункту. Це оптимізує роботу з великими масивами інформації, зменшує завантаження на сервер і забезпечує швидкість відповіді для користувачів, дозволяючи їм швидше знайти потрібні дані.

Для підвищення ефективності взаємодії між клієнтом і сервером API можна включати механізми кешування, які зберігають часто запитувані дані, такі як список замовлень чи статуси доставки. Завдяки цьому зменшується кількість звернень до бази даних, що приводить відповіді на повторювані запити та зменшує загальне завантаження на сервер.

Таким чином, API для обміну даними є основою, яка забезпечує надійний, швидкий і захищений доступ до інформації, підтримуючи всі функції логістичної системи. Це дозволяє користувачам легко отримувати актуальні дані та керувати

замовленнями і маршрутами в реальному часі, створюючи прозорий і ефективний процес обміну інформацією між усіма учасниками логістичного ланцюга.

Впровадження логіки управління замовленнями, моніторингом і маршрутизацією в логістичній системі є комплексним процесом, який охоплює кілька інтегрованих рівнів обробки даних, алгоритмів прийняття рішень і забезпечення точності та швидкості операцій. Логіка управління замовленнями включає детальну обробку даних про кожне замовлення: від моменту його створення до моменту доставки. Якщо користувач (наприклад, клієнт або оператор) створить нове замовлення, система повинна зафіксувати всі критичні дані, такі як пункт відправлення і призначення, обсяг продукції, час доставки та інші параметри. Для цієї бази даних створюється запис, який містить унікальний ідентифікатор замовлення та статус «створено», після чого замовлення проходить через усі повні етапи обробки.

Якщо замовлення тільки отримує статус «в процесі», воно автоматично вноситься в систему моніторингу, що дозволяє відстежувати його прогрес у реальному часі. Логіка моніторингу реалізується через постійний зв'язок між транспортними засобами та бекендом систем за допомогою GPS-трекерів. Система повністю отримує координати кожного транспортного потоку та оновлює дані в базі. Ці координати передаються на сервер через API і залишаються доступними для відображення на карті в інтерфейсі користувача. Моніторинг також індикатори таких проблем, як затримки на маршруті або відхилення від початково запланованого маршруту. У таких випадках система автоматично формує повідомлення або попередження для диспетчера, який може прийняти оперативне рішення.

Важливим аспектом є реалізація логіки маршрутизації, яка відповідає вибору оптимального маршруту для кожного замовлення, враховуючи такі параметри, як відстань, стан доріг, прогнозовані затори або погодні умови. Маршрутизація використовує дані з картографічних сервісів, як-от Google Maps API або Mapbox, і

розраховує декілька можливих варіантів доставки за конкретними потребами замовлення (наприклад, терміновість доставки або обмеження щодо певних доріг). Ця логіка також інтегрує алгоритми машинного навчання, які аналізують історичні дані, виявляючи найкращі маршрути на основі попередніх поїздок і обсягів трафіку. Ці додаткові системи прогнозують затримки та вибирають найефективніший маршрут у реальному часі [40].

Управління замовленнями також включає оновлення статусів замовлення на кожному етапі, від обробки до доставки. Під час проходження кожного етапу система змінює статуси в базах даних, які відображаються на панелі користувачів, дозволяючи диспетчерам і клієнтам бути в курсі поточного стану доставки. Наприклад, коли транспортний засіб досягає певної точки або переходить на відстань 5 кілометрів від точки призначення, система може автоматично змінити статус на «доставляється» і надіслати погодження клієнту про наближення замовлення.

Для контролю за виконанням кожного замовлення впроваджуються механізми запису й аналізу даних про час проходження кожного етапу маршруту, що дозволяє отримувати звіти про продуктивність логістичної мережі. Дані про виконання кожного замовлення використовуються для побудови аналітичних звітів, які виявляють точки затримок або проблемні ділянки маршруту, що дозволяє оптимізувати майбутні доставки. Завдяки такій логіці можна не тільки контролювати поточні замовлення, але й покращувати процеси доставки, враховуючи попередні результати [41].

У разі виникнення непередбачених умов, таких як технічні несправності транспорту або різких змін погодних умов, система підтримує логіку переадресації замовлень. Це означає, що у випадку збоїв вона автоматично перенаправляє замовлення на інші транспортні засоби або коригує маршрут. У таких випадках диспетчери підтримують повідомлення і можуть втрутитися в процес, якщо потрібно прийняти рішення вручну.

Завдяки комплексній логіці управління системами замовлення, моніторингу та маршрутизації забезпечується високий рівень автоматизації та прозорості процесів, що дозволяє користувачам завжди мати актуальні дані для прийняття рішень і швидко адаптуватися до змін умов.

3.3. Інтеграція з GPS-трекерами. Використання API для підключення до GPS-трекерів транспорту.

Інтеграція логістичної системи з GPS-трекерами дозволяє точно відстежувати сайт транспорту в реальному часі, що є критичним місцем для управління замовленнями та оптимізації маршрутів. GPS-трекери, встановлені на транспортних засобах, постійно фіксують координати, швидкість, напрямок руху та інші дані, які передаються на сервер за допомогою мобільного зв'язку або спеціалізованих IoT-протоколів, таких як MQTT. Ці дані забезпечують постійний потік інформації про кожен транспортний засіб і допомагають системі формувати загальну картину логістичних операцій.

Для інтеграції з GPS-трекерами використовується API, який виступає інтерфейсом для отримання та обробки даних з трекерів у реальному часі. Стандартні GPS-постачальники надають власний API, через який можна отримувати координати і додаткові дані про транспортні засоби. У такому випадку логістична система підключається до API GPS-постачальника і, використовуючи унікальні ідентифікатори кожного трекера (зазначте це серійний номер чи інший унікальний токен), запитує дані про місце. Наприклад, кожен транспортний засіб, що має GPS-трекер, реєструється в базі даних системи, а під час запиту API звертається до сервера GPS-постачальника, щоб отримати координати для відповідного ідентифікатора [42].

Ці запити переглядаються через HTTP-запити, часто у форматі JSON, де на кожен запит система отримує координати та додаткову інформацію, яка обробляється в реальному часі. Важливою особливістю API для роботи з GPS-

трекерами є можливість отримувати дані в потоках, що дозволяє оновлювати інформацію про веб-транспорт безперервно і з мінімальною затримкою. У деяких системах можуть використовуватися веб-сайти, що дозволяють GPS-пристроям передавати дані на сервер в режимі реального часу, створюючи постійний канал зв'язку, а не періодичні запити, як це буває з класичним REST API. Такий підхід є оптимальним для логістичних систем, де точне і швидке оновлення даних створює критичну роль.

Зберігши дані про координати, система оновлює їх у базі даних і робить доступними для інтерфейсу користувача, де диспетчери або адміністратори можуть відстежувати пересування транспортних засобів на карті. Цей процес не просто показує поточний сайт, а й підтримує функцію прогнозування: з використанням історичних систем даних можна аналізувати затримки, виявляти найбільш проблемні ділянки на маршрутах і автоматично рекомендувати альтернативні шляхи [43].

Крім регіонів, API для GPS-трекерів може надавати інші корисні дані, такі як швидкість транспорту, час простою, витрати палива, що особливо важливо для оптимізації логістики. Наприклад, якщо система відображає надто довгий час простою, вона може автоматично надіслати повідомлення диспетчеру про можливе затримання, що дозволяє диспетчерам швидко реагувати на проблеми в режимі реального часу.

Інтеграція з GPS також забезпечує безпеку та контроль над транспортом. Наприклад, система може контролювати швидкість і миттєво включити диспетчера про її перевищення, або ж реагувати на відхилення від запланованого маршруту, яке може свідчити про проблему або позаштатну травму. У таких випадках система може автоматично записувати всі дані та робити їх доступними для аналізу, дозволяючи диспетчерам мати повну картину ситуації та оперативно приймати рішення [43].

Інтеграція логістичної системи з GPS-трекерами через API забезпечує детальний і постійний контроль над транспортними засобами, надаючи точні дані для ефективного управління логістикою в реальному часі. Це дозволяє оптимізувати маршрути, мінімізувати затримки та знизити операційні витрати, створюючи надійну та прозору систему доставки.

3.4. Розробка бази даних. Створення бази даних для зберігання інформації про маршрути, замовлення, транспорт

Розробка бази даних для логістичних систем є одним із окремих етапів, крім того, база даних виступає центральним сховищем для всіх даних про замовлення, маршрути, транспортні засоби, статуси доставки і навіть системи користувачів. Для такої системи необхідно створити базу даних, яка не лише забезпечує збереження великого обсягу інформації, але й забезпечує швидкий доступ до даних, стабільність під час високого завантаження та підтримку складних запитів для аналітики та управління маршрутами. Основу бази даних складають кілька великих сутностей, зокрема замовлення, транспорт, маршрути та користувачі. Для забезпечення цільності даних і точного зв'язку між ними використовуються реляційні бази даних, такі як PostgreSQL або MySQL. Кожна сутність має свою таблицю в базі даних, яка містить усі атрибути, необхідні для її повного опису. Наприклад, таблиця замовлень! Додайте інформацію про ідентифікатор замовлення, тип товару, обсяг, клієнта, пункт відправлення та призначення, заплановану дату доставки та поточний статус. Це дозволяє системі в будь-який момент відобразити точне положення про посаду і всі деталі, пов'язані з ним.

Для забезпечення зв’язку між замовленнями і транспортними засобами в базі створені зовнішні ключі, що дозволяють забезпечити зв’язок між таблицями та виключити дублювання даних. Наприклад, додайте посилання на конкретний транспортний засіб, який його виконує, а сам транспортний засіб пов’язаний із водієм і маршрутом. Це дозволяє запитувати не тільки базову інформацію про

замовлення, але й отримувати комплексну інформацію, включаючи транспортний засіб, поточний статус маршруту та водія, який залишає перевезення [44].

Таблиця маршрутів зберігає інформацію про координати кожного пункту на маршруті, а також часові позначки для прогнозування часу. Це важливо для аналізу ефективності доставки, після збережених маршрутів і годин позначки не можна прораховувати середню швидкість транспорту, прогнозувати затримки та оптимізувати подальші маршрути. База даних зберігає також історичні дані про маршрути, що дозволяє використовувати їх для аналізу та покращення логістичних процесів. Наприклад, аналізуючи попередні маршрути і час доставки, система може визначити ділянки з підвищеним ризиком затримок і запропонувати альтернативні шляхи. Крім основних таблиць для замовлень і маршрутів, створити таблицю для зберігання даних про транспортні засоби. Ця таблиця містить інформацію про тип транспортного стресу, вантажопідйомність, витрати палива, технічний стан і дати останнього обслуговування. Це дозволяє системі не тільки вибирати відповідний транспортний засіб для кожного замовлення на основі обсягу продукції та відстані, але й контролювати технічне обслуговування транспорту, щоб уникнути проблем під час перевезення. Інформація про технічний стан зберігається з прив'язкою до маршрутів, що дозволяє підтримувати систему планування технічних перевірок, щоб уніфікувати використання транспортних засобів, які потребують ремонту [45].

Щоб оптимізувати запити до бази даних, використовують індекси для полів, які часто запитуються, такі як статус замовлення, ідентифікатор маршруту або номер транспортного продукту. Це забезпечує швидке виконання запитів навіть при значному обсязі даних. Крім того, індекси можуть прискорити складні запити, які об'єднують дані з кількох таблиць. Наприклад, індекси на шляху маршруту та замовлення прискорюють виконання запитів для відображення статусу доставки в реальному часі на основі останніх оновлених даних з GPS.

Для забезпечення цільності бази даних застосовуються операції, що гарантують збереження повноти даних навіть при високих навантаженнях або збої.

Наприклад, якщо під час оновлення даних про замовлення вимагається збій, транзакція забезпечує відкат до попереднього стану, запобігаючи некоректному збереженню даних. Це забезпечує безпеку інформації, що особливо важливо у великій логістичній системі, де навіть незначні помилки в даних можуть призвести до збою в доставці або помилок у маршрутизації.

Система баз даних також налаштована для роботи з географічними даними, що дозволяє інтегрувати координати маршрутів і підтримувати запити з геопросторовим аналізом. Наприклад, розширення PostGIS для PostgreSQL дозволяє зберегти і обробляти географічні координати, що полегшує маршрутизацію та оптимізацію логістичних шляхів на основі точного розташування. Це дозволяє точно відображати дані на карті, аналізувати маршрути, обчислювати відстані між пунктами та використовувати ці дані для вибору оптимальних шляхів доставки. База даних стає основою, що зберігає всі необхідні дані для управління логістикою, забезпечуючи стабільність, гнучкість і точність у всій системі. Це дозволяє користувачам відстежувати, планувати й аналізувати логістичні процеси в реальному часі, що є основою для успішної роботи логістичної мережі.

У цій схемі є таблиці для зберігання даних про замовлення, транспортні засоби, маршрути, а також для фіксації статусів і відстеження місцезнаходження транспорту. Цей приклад враховує зв’язки між таблицями, щоб забезпечити цілісність даних.

SQL-скрипт для створення бази даних логістичної системи

```
-- Створення бази даних логістики
CREATE DATABASE logistics_system;
\c logistics_system;

-- Таблиця для зберігання інформації про замовлення
CREATE TABLE orders (
    order_id SERIAL PRIMARY KEY,
    client_name VARCHAR(255) NOT NULL,
```

```
    origin VARCHAR(255) NOT NULL,
    destination VARCHAR(255) NOT NULL,
    product_type VARCHAR(255) NOT NULL,
    quantity NUMERIC(10, 2) NOT NULL,
    order_status VARCHAR(50) DEFAULT 'Pending',
    created_at TIMESTAMP DEFAULT NOW(),
    updated_at TIMESTAMP
);

-- Таблиця для зберігання інформації про транспортні засоби
CREATE TABLE vehicles (
    vehicle_id SERIAL PRIMARY KEY,
    license_plate VARCHAR(20) UNIQUE NOT NULL,
    vehicle_type VARCHAR(50) NOT NULL,
    capacity NUMERIC(10, 2) NOT NULL,
    fuel_consumption NUMERIC(5, 2),
    last_service_date DATE,
    status VARCHAR(50) DEFAULT 'Available'
);

-- Таблиця для зберігання інформації про маршрути
CREATE TABLE routes (
    route_id SERIAL PRIMARY KEY,
    order_id INT REFERENCES orders(order_id) ON DELETE CASCADE,
    vehicle_id INT REFERENCES vehicles(vehicle_id) ON DELETE SET NULL,
    start_location VARCHAR(255) NOT NULL,
    end_location VARCHAR(255) NOT NULL,
    estimated_distance_km NUMERIC(10, 2),
    estimated_time_hours NUMERIC(10, 2),
    actual_distance_km NUMERIC(10, 2),
    actual_time_hours NUMERIC(10, 2),
    route_status VARCHAR(50) DEFAULT 'Scheduled',
    start_time TIMESTAMP,
    end_time TIMESTAMP
);

-- Таблиця для зберігання GPS-трекінгу транспортних засобів на маршруті
CREATE TABLE gps_tracking (
    tracking_id SERIAL PRIMARY KEY,
```

```
    route_id INT REFERENCES routes(route_id) ON DELETE CASCADE,
    vehicle_id INT REFERENCES vehicles(vehicle_id) ON DELETE CASCADE,
    latitude NUMERIC(9, 6) NOT NULL,
    longitude NUMERIC(9, 6) NOT NULL,
    speed NUMERIC(5, 2),
    recorded_at TIMESTAMP DEFAULT NOW()
);

-- Таблиця для зберігання історії змін статусів замовлень
CREATE TABLE order_status_history (
    history_id SERIAL PRIMARY KEY,
    order_id INT REFERENCES orders(order_id) ON DELETE CASCADE,
    previous_status VARCHAR(50),
    new_status VARCHAR(50),
    changed_at TIMESTAMP DEFAULT NOW(),
    changed_by VARCHAR(255)
);

-- Таблиця для зберігання інформації про водіїв
CREATE TABLE drivers (
    driver_id SERIAL PRIMARY KEY,
    driver_name VARCHAR(255) NOT NULL,
    phone_number VARCHAR(20) UNIQUE NOT NULL,
    license_number VARCHAR(50) UNIQUE NOT NULL,
    vehicle_id INT REFERENCES vehicles(vehicle_id) ON DELETE SET NULL
);

-- Таблиця для призначення водіїв на маршрути
CREATE TABLE route_assignments (
    assignment_id SERIAL PRIMARY KEY,
    route_id INT REFERENCES routes(route_id) ON DELETE CASCADE,
    driver_id INT REFERENCES drivers(driver_id) ON DELETE SET NULL,
    assigned_at TIMESTAMP DEFAULT NOW()
);

-- Таблиця для зберігання інформації про користувачів системи
CREATE TABLE users (
    user_id SERIAL PRIMARY KEY,
    username VARCHAR(255) UNIQUE NOT NULL,
```

```
    password_hash VARCHAR(255) NOT NULL,
    role VARCHAR(50) NOT NULL,
    created_at TIMESTAMP DEFAULT NOW()
);
```

Ось опис структури бази даних

orders: Таблиця для зберігання інформації про замовлення, включаючи замовника, місце призначення, тип продукту та статус замовлення. Статус оновлюється по мірі виконання замовлення (наприклад, Pending, In Progress, Completed).

vehicles: Таблиця для зберігання даних про транспортні засоби, включаючи тип транспорту, вантажопідйомність, витрати палива та статус доступності.

routes: Таблиця для зберігання даних про маршрути, включаючи маршрут, транспортний засіб, заплановані і фактичні відстань і час, а також статус маршруту (наприклад, Scheduled, In Progress, Completed).

gps_tracking: Таблиця для запису GPS-даних у реальному часі, включаючи координати, швидкість і час запису, що дозволяє відстежувати транспорт у русі.

order_status_history: Таблиця для зберігання історії змін статусів замовлень, що забезпечує відстеження дій для подальшого аналізу і звітності.

drivers: Таблиця для даних про водіїв, які пов’язані з транспортними засобами і мають унікальні номери водійських ліцензій.

route_assignments: Таблиця для призначення водіїв на конкретні маршрути з фіксацією часу призначення.

users: Таблиця для зберігання інформації про користувачів системи з ролями для контролю доступу (наприклад, адміністратор, диспетчер).

Ця база даних дозволяє не тільки зберігати інформацію про замовлення, маршрути і транспортні засоби, але й підтримує комплексний облік і аналітику, зберігаючи історію змін статусів, дані про місцезнаходження в реальному часі та історичні дані для аналізу. Це надає можливість логістичній системі легко

масштабуватися, зберігати точну інформацію для планування та оптимізації, а також підтримувати прозорість усіх логістичних процесів [46].

ВИСНОВКИ ДО РОЗДІЛУ 3

Для розробки системи важливо вибрати оптимальні інструменти та технології для реалізації проекту. Для даного продукту було вибрано клієнт-серверну архітектуру, що дозволяє розділити обробку та зберігання даних.

Розробку клієнт-частини можна реалізувати за допомогою фреймворків JavaScript, таких як React та Vue.js. Вони дозволяють реалізувати окремі необхідні функціональні блоки, які необхідні для системи. За допомогою цих фреймворків можна реалізувати панель відстеження замовлень, а також контроль виконання замовлень.

Для створення серверної частини проекту можна використати Node.js або Python. За допомогою даних технологій можна дозволити серверній частині обробляти запити клієнтів, зберігають інформацію про замовлення, транспорт і маршрути, а також забезпечують основну логіку системи. Для швидкого обміну інформацією використовують RESTful або GraphQL API, а використавши Redis або Memcached можна зменшити завантаження на сервер та збільшити швидкість обробки запитів.

Важливо створити якісну базу даних, щоб система працювала максимально коректно. Для цього можна використати реляційні бази даних, наприклад PostgreSQL або MySQL, що структурують інформацію. Для відстеження транспорту можна використати базу MongoDB, що обробляє неструктуровані дані.

У кожної з представлених технологій часто використовуються для розробки різних типів платформ. Кожна з них має свої переваги та недоліки. Важливо підібрати саме такі технології, які можуть реалізувати необхідний функціонал та щоб система працювала необхідним чином.

РОЗДІЛ 4

РОЗРОБКА СТАРТАП ПРОЕКТУ

4.1. Загальна характеристика стартап-проекту

Розробка стартап-проекту в галузі логістики, заснованого на системі для управління перевезеннями, замовленнями та маршрутизацією, передбачає створення комплексного рішення, яке забезпечить компаніям ефективний контроль над доставками та оптимізацію процесів у реальному часі. Такий проект може мати велике практичне застосування для логістичних операторів, транспортних компаній, а також для підприємств, що самостійно керують власними логістичними процесами. Стартап також може націлитися на автоматизацію перевезень для галузей із швидкопсувними товарами, забезпечуючи точний контроль над часом та умовами доставки. Більш детальна інформація по особливостям та характеристиці проекту наведена у таблицях 4.1 – 4.3.

Таблиця 4.1. Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Система LogiTrack автоматизації оформлення перевезень та торгівлі нафтопродуктів	1. Оформлення угод	Комунікація між логістами, клієнтами, продавцями та перевізником в одній системі
	2. Пропозиції послуг	Представлення продавцями та перевізниками своїх послуг
	3. Управління транспортними засобами	Контроль логістами перевезень напряму з перевізниками
	4. Доступ до Інтернету	Використання системи з різних пристроїв не прив’язуючись до локації
	5. Оформлення документації	Автоматичне створення накладних, договорів і митних документів
	6. Розрахунки з клієнтами	Автоматизація процесу виставлення рахунків та оплати

Таблиця 4.2. Опис унікальних властивостей продукту

№ п/п	Назва властивості	Пояснення
1	Конкурентне середовище	На платформі постачальники та перевізники виставляють свої пропозиції, що дає змогу клієнтам обрати найоптимальніший для себе варіант
2	Вузька галузь	Галузь нафтопродуктів занадто застаріла в Україні, тому досі в мережі немає подібних платформ з таким же функціоналом, тому в даній галузі немає конкурентів
3	Просте користування	Всі аспекти, що необхідні для укладання угод перевезення та торгівлі нафтопродуктів знаходяться на одній платформі
4	Контроль операцій	В платформі є можливість прямого спілкування між замовником та постачальником, що дає змогу повністю контролювати угоду на всіх етапах її оформлення.

Таблиця 4.3. Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№ п/п	Техніко-економічні характеристики ідеї	Потенційні товари/концепції конкурентів		W (слабка сторона)	N (нейтраль-на сторона)	S (сильна сторона)
		Мій проект	Конку-рент			
1	Бюджетне фінансування	Відсутність фінансування	Використання інвестицій	Відсутність фінансування	Часткове фінансування	Повне фінансування
2	Матеріальна база	Наявна повна матеріальна база	Часткова матеріальна база	Відсутність матеріальної бази	Часткова матеріальна база	Повна матеріальна база

Продовження таблиці 4.3.

3	Наявність маркетингової стратегії	Веб-реклама, реклама за допомогою партнерів	Достатнє використання реклами	Відсутність просування	Часткове просування	Активна трансляція
4	Використання бізнес-послуг	Наявна	Відсутня	Відсутність консультування	Часткове консультування	Повне ведення поекту

4.2. Технологічний аудит ідеї проекту

Таблиця 4.4. Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технології	Доступність технологій
1	Система LogiTrack автоматизації оформлення перевезень та торгівлі нафтопродуктів	WebStorm	Наявна	69.00$ за рік користування / тимчасове безкоштовне користування
2		Бібліотеки JS, Node.JS, PostgreSQL	Наявні	У відкритому доступі

Розробка LogiTrack здійснюється з використанням архітектури, що поєднує бекенд-сервер на Node.js з базою даних PostgreSQL, яка зберігає всі ключові дані про замовлення, маршрути та транспортні засоби. Використання вебсокетів або MQTT для реального часу дозволяє безперервно оновлювати дані про місцезнаходження транспортних засобів. Фронтенд, створений на React, забезпечує зручний і швидкий доступ до інформації, а також інтеграцію з Google Maps або іншими картографічними сервісами для візуалізації маршрутів у реальному часі.

4.3. Аналіз ринкових можливостей запуску стартап-проекту

Платформа LogiTrack є унікальною платформою в своєму середовищі, що надає сильні переваги над окремими постачальниками, які вручну шукають клієнтів. Дана платформа дозволить клієнтам знайти оптимальні пропозиції. Дослідження предметної галузі надає змогу платформі задовільнити потреби всіх користувачів системи. У таблиці 4.5 наведено попередню характеристику потенційного ринку стартап-проекту.

Таблиця 4.5. Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	3
2	Загальний обсяг продаж, $	100000
3	Динаміка ринку (якісна оцінка)	Не змінюється
4	Наявність обмежень для входу (вказати характер обмежень)	Клієнти мають своїх постачальників
5	Специфічні вимоги до стандартизації та сертифікації	Відсутні
6	Середня норма рентабельності в галузі (або по ринку), %	100
7	Попит	Високий
8	Технологія	На стадії активного розвитку
9	Зміни в міжнародній торгівлі і прямих інвестиціях	Часто змінюються
10	Інтенсивність конкуренції	Відсутня
11	Швидкість зміни умов ринку	Ринкові умови змінюються від багатьох факторів: ціни на нафту, становище в країні, валютні коливання, зміна податків
12	Доступність інвестицій	Необмежено

Таблиця 4.6. Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності поведінки потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Постачання палива	АЗС, сільські господарства, підприємства	Різні види нафтопродуктів, відмінність попиту в залежності від сезону	Зручне середовище, якість товару, швидкість виконання угод
2	Логістичний супровід	Заводи та під-приємства з галузі нафтопродуктів	Різні умови співпраці, окрема специфіка роботи різних компаній	Спеціалізовані автомобілі, зручна логістика, велика кількість пропозицій
3	Перевезення нафтопродуктів	Логістичні організації; компанії, що торгують нафтопродуктами	Цілі використання послуг, що залежать від специфіки послуг	Прозорий контроль процесу, конкурентний ринок, низькі ціни

Таблиця 4.7. Аналіз загроз запуску та роботи стартап-проекту

№ п/п	Фактор загрози	Опис загрози	Планове реагування компанії
1	Військовий стан в країні	Через постійні обстріли та війну в країні зменшується кількість споживачів	За допомогою маркетингу збільшувати кількість користувачів
2	Складність використання	Споживачі можуть мати певні проблеми в користуванні нової системи	Створення логічного та простого інтерфейсу, який буде зручним для користувачів

Продовження таблиці 4.7.

3	Коливання цін	Через багато факторів ціни на нафтопродукти можуть сильно коливатись	Створення сильнішої конкуренції на платформі
4	Поява конкурентів	Поява конкурентів в даній галузі може зменшити кількість клієнтів даної системи	Додавання функціоналу до платформи для більш зручного користування

Таблиця 4.8. Аналіз можливостей по реалізації стартап-проекту

№ п/п	Фактор можливості	Зміст можливості	Можлива реакція компанії
1	Використання бази клієнтів	Використання системи компаніями, що працюють в галузі	Спрощення процесу продажу та перевезення нафтопродуктів
2	Заохочення нових користувачів	Заохочення нових користувачів за допомогою головних переваг	Збільшення кількості продажів та надання якісних послуг
3	Ведення активної маркетингової компанії	За допомогою маркетингу додавання нових користувачів системи	Збільшення пропозицій та конкуренції в системі, збільшення якості наданих послуг, зменшення цін

Таблиця 4.9. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	Прояв характеристики конкурентного середовища	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
Тип конкуренції	Чиста конкуренція	Зменшення цін на послуги

Продовження таблиці 4.9.

Рівень конкурентної боротьби	Національний	Співпраця з компаніями по всій території України
Галузева ознака	Внутрішньогалузева	Надання гарантій співпраці з компаніями
Конкуренція за видами товарів	Товарно-родова	Збільшення кількості послуг, що надає платформа
Характер конкурентних переваг	Цінова	За допомогою конкуренції зменшення цін та збільшення якості

Таблиця 4.10. Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товаризамінники
	Продавці нафтопродуктів	Маркетингові компанії	Перевізники	Орієнто-ваність на клієнтські потреби	Міжнародні компанії
Висновки	Основною загрозою є співпраця продавців безпосередньо з клієнтами напряму, створення під кожного персональних умов, що призведе до зменшення попиту	Недостатня матеріальна база може призвести до створення не оптимального продукту	Великий ринок перевізників створює конкуренцію, та зменшення цін	При збільшені постачальників якість та ціни послуг покращуються, що впливає на вибір клієнтів. Також важливий подальший розвиток платформи	Може з’явитись після зміни положення країни, можливе подальше використання для розвитку системи

Таблиця 4.11. Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкуренто-спроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Цінова політика	Встановлення вартості послуг, що будуть вигідні для споживачів
2	Репутація виробника	Працювати з перевіреними постачальниками та перевізниками, щоб гарантувати клієнтам 100% надійність співпраці
4	Унікальність сервісних послуг	Надання споживачам послуг в одній системі, що є унікальним в галузі

Таблиця 4.12. Порівняльний аналіз сильних та слабких сторін стартап-проекту

№ п/п	Фактор конкуренто-спроможності	Бали 1-20	Порівняння рейтингу товарів-конкурентів						
			-3	-2	-1	0	+1	+2	+3
1	Лідерські позиції на ринку	19						+	
2	Цінова політика компанії	18					+		
3	Торговий маркетинг	17				+			
4	Репутація компанії	19						+	
5	Лояльність до бренду	18					+		
6	Інвестиційний бюджет	16			+				

На основі попереднього аналізу різних факторів загроз та можливостей стартап-проекту по застосуванню Система LogiTrack автоматизації оформлення

перевезень та торгівлі нафтопродуктів, у таблиці 4.13 наведено матрицю SWOT. Дана схема відображає загальну картину про розвитку проекту на ринку.

Таблиця 4.13. SWOT- аналіз стартап-проекту

Сильні сторони: 1. Унікальність продукту 2. Зручність системи 3. Оптимізація процесу 4. Конкурентне середовище	**Слабкі сторони**: 1. Тяжка маркетингова стратегія для розширення бази клієнтів 2. Складна політична та економічна ситуація в країні
Можливості: 1. Перевести ринкові угоди в одну систему 2. Розширення функціоналу платформи 3. Зменшення ціни 4. Зріст якості наданих послуг	**Загрози**: 1. Зменшення споживачів 2. Поява конкурентів на ринку

Таблиця 4.14. Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Активне опанування ринку та укладення нових договорів із підприємцями	Висока	Короткі
2	Слідування стратегії по забезпеченню конкурентоспроможності	Середня	Короткі

4.4.Розробка ринкової стратегії проекту

При планування ринкової стратегії проекту варто зосередити увагу на цільових групах потенційних споживачів.

Таблиця 4.15. Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Підприємства, сільські господарства, автозаправні станції	Готовий	Високий	Середня	Тяжка
2	Транспортні компанії	Готовий	Високий	Середня	Проста
3	Логістичні компанії	Готовий	Середній	Низька	Середня
Обрані цільові групи: стратегія цільового маркетингу					

Згідно аналізу потенційних споживачів на запропонований продукт, було обрано стратегію цільового маркетингу, оскільки система створена для використання окремими компаніями, які працюють у вузькій галузі.

Таблиця 4.16. Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	Стратегія зростання	Прагнення компаній до збільшення попиту та пропозицій від користувачів системи	Стратегія концентрованого зростання, внутрішнє розширення споживачів, а також розширення функціоналу платформи	Стратегія цільового маркетингу

Таблиця 4.17. Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першо-прохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1	Так	Стратегія працювати з існуючими споживачами, оскільки нових майже не з'являється	Не буде, через відсутність конкурентів	Стратегія цінового лідера

Таблиця 4.18. Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Забезпечення високої якості послуг	Стратегія цільового маркетингу	Співпраця з надійними постачальниками та перевізниками	Якість, надійність, безпека
2	Вигідна ціна на продукти	Стратегія цінового лідера	Створення високої конкуренції на платформі	Вигідність, вибір, якість

4.5 Розроблення маркетингової програми стартап-проекту

Таблиця 4.19 Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити
1	Зручність платформи	Товар об'єднує всі необхідні функції для укладання угод	Зручність, зрозумілість
22	Конкуреція на ринку	Товар створює конкуренцію між перевізниками та постачальниками	Вигідні ціни, збільшена якість, надійність

Таблиця 4.20 Формування системи збуту стартап-проекту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Зорієнтованість на високоякісний продукт	Надання можливості використовувати свою платформу задля надання або ж отримання необхідних послуг	Значна	Цільова реклама, пряма агітація

Таблиця 4.21 Концепція маркетингових комунікацій стартап-проекту

№	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Застосування системи	Веб-застосунки	Вигідні пропозиції, надійність, якість	Представлення товару споживачам	Зручна логістично-торгівельна платформа

ВИСНОВКИ ДО РОЗДІЛУ 4

В розробці стартап–проекту лежить головна ідея використати систему як платформу для паливних підприємств, постачальників, логістичних компаній, перевізників як окремих автомобілей, так і автопарки, а також для клієнтів (автозаправних станцій, сільських господарств, заводів). Дана платформа забезпечить взаємостосунки між партнерами, що надасть змогу контролювати угоди в одному просторі, а окрім цього можливість створення конкуренції між перевізниками та постачальниками. Конкуренція позитивно впливає як на клієнтів, так і користувачів, що надають послуги.

Стартап-проект застосовуватиметься для оформлення замовлень, надання власних послуг, управління транспортними засобами, оформленнями документації та розрахунками з клієнтами. Серед головних властивостей платформа буде мати конкурентне середовище, вузьку галузь застосування, просте користування та контроль операцій. Для даного проекту важлива матеріальна база та наявність спеціалізованої маркетингової стратегії.

Розроблення системи буде виконуватись за допомогою JS-фреймворку Node.js з базою даних PostgreSQL, а клієнт-частина розроблена за допомогою React.

При ознайомлені ринку галузі було розглянуто головні характеристики для запуску даного стартапу в галузі нафтопродуктів. До потенційних клієнтів належать клієнти, що потребують нафтопродуктів (АЗС, сільські господарства, підприємства), компанії що використовують транспорт у внутрішніх цілях (нафтобази та підприємства), постачальники палива(торгові компанії, нафтобази). Попередньо ознайомлено з головними загрозами відтворення стартапу – військовий стан в країні, відсутність попередніх практик запровадження подібних систем, коливання цін та появи нових конкурентів. Для реалізації проекту такі можливості, як наявна матеріальна база, можливість заохочення нових користувачів, та проведення правильної маркетингової компанії.

Провівши повний аналіз можливостей платформи було досліджено, що головними перевагами проекту є унікальність продукту, зручність системи, оптимізація циклу проведення угод та створення умов для здорової конкуренції. Проект має на меті перевести ринкові угоди в одну систему, збільшити функціонал для виконання всіх необхідним потреб користувачів, зменшити ціну на внутрішньому ринку та підвищення якості наданих послуг.

ВИСНОВОК

Висновок із розробленої стратегії для продукту управління логістикою нафтопродуктів демонструє, що успішний вихід на ринок потребує глибокого розуміння потреб ключових споживачів, аналізу конкурентного середовища та адаптації маркетингової стратегії до конкретних вимог галузі. Такий продукт є не просто технологічним інструментом, а комплексним рішенням, яке інтегрує різні етапи логістики, відстеження, управління безпекою та замовлення маршрутизації, що особливо важливо для компаній, чия діяльність залежить від стабільності поставок і високих стандартів. Стратегія просування орієнтована на цінну пропозицію: продукт не тільки скорочує витрати і ефективність, а й надає можливості для довгострокового планування і гнучкої адаптації продукту до мінливих умов.

Впровадження даного продукту для таких груп споживачів, як логістичні компанії, нафтові підприємства та промислові об’єкти з власною логістикою, має на меті розв’язання найважливіших проблем: складність планування перевезень, нестача контролю в реальному часі, висока залежність від зовнішніх факторів, таких. як погодні умови і коливання ціна на паливо. Продукт орієнтований на забезпечення безпеки і точності, що є визначальним для збереження якості нафтопродуктів і дотримання нормативних вимог. Завдяки модулю та можливості адаптації продукту, він підходить для різних масштабів бізнесу: від малих компаній, які прагнуть до економії, до великих корпорацій, які потребують комплексного контролю над усіма аспектами логістики.

Маркетингова стратегія, спрямована на побудову репутації бренду через контент, участь у галузевих заходах та співпрацю з технологічними партнерами, є важливою складовою, яка дозволяє створити довіру до продукту. без стандартної реклами, акцент зроблений на глибокій експертній аналітиці, кейсах з реальними прикладами впровадження та постійній підтримці клієнтів, що забезпечує лояльність і готовність до тривалого партнерства. Це підсилює позицію продукту

на ринку, роблячи його привабливим не тільки як інструмент для вирішення поточних завдань, але і як довгострокове інноваційне рішення. Створений підхід об'єктом технічної інновації з розумінням галузевих потреб і стратегічним просуванням на ринку. Такий продукт не лише надійним клієнтам оптимізує логістичні процеси, а й покращує стабільну побудову та інфраструктуру ставок нафто конкурентів, забезпечуючи переваги для користувачів та створюючи цінність для ринку продукту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Агіновський А. Ю., Федько В. П., Митько О. А. Логістика торгівлі. Серія «Підручники і навчальні посібники». — Київ: Фенікс, 2021.

2. Г. В. Савицька, Аналіз господарської діяльності підприємства: Навчальний посібник / ред. Г. В. Савицька. — 7-ме вид., доп. та перероб. — Харків: Нове знання, 2022.

3. Анікін Б. А., Тяпухін А. П. Комерційна логістика: Підручник. — Київ: ТК Велбі, «Проспект», 2020.

4. Бондаренко Д. Дж., Клосс Д. Дж. Логістика. Інтегрований ланцюг поставок: переклад з англійської. — Київ: ЗАТ «Олімп-Бізнес», 2021.

5. Гаджинський А. М. Логістика: Підручник. — 4-те вид., доповнене та перероблене. — Київ: ІКЦ «Маркетинг», 2021.

6. Гордон М. П., Карнаухов С. Б. Логістика товарообігу. — Київ: Радіо і зв'язок, 2021. — 160 с.

7. Зюзькін Д. Т. Фінансове право: конспект лекцій. — Київ: МАУП, 2021.

8. Крикавський Є. В. Логістика. Основи теорії: Підручник — Львів: «Інтелект-Захід», 2021. — 416 с.

9. Лукінський В. С. та інші. Логістика на прикладах. — Київ: «Фінанси», 2021.

10. Лукінський В. С. Моделі та методи теорії логістики: Навчальний посібник / ред. В. С. Лукінського. — Київ, 2021.

11. Нестеренко Б. К. Теоретико-логістичні основи процесу товарообігу. — Самара: 2021. — 44 с.

12. Неруш Ю. М. Комерційна логістика. — Київ: «Банки і біржі» ЮНІТІ, 2020.

13. Ніколенко Д. Т., Ганько А. П. Логістичні системи: їх значення та ефективність // Логістика і бізнес / ред. Мільтон Л. Б. — Київ: Брандес, 2020. — сс. 32-35.

14. Основи логістики: Навчальний посібник / ред. Л. Б. Мільтон та В. І. Сергієнко. — Київ: Інфа-М, 2020.

15. Нільсон Б.Р. Практикум з логістики: Навчальний посібник.— 2-ге видання, доповнене та перероблене: Інфа-М, 2020.
16. Семоненко А. І. Підприємницька логістика. — СПб.: Політехніка, 2020. — 346 с.: іл.
17. Сирник А. І., Голдін В. І. Логістика. Основи теорії: навчальний посібник для вищих навчальних закладів. — СПб.: «Союз», 2021. — 544 с.
18. Сігур І. В. Економіка підприємства: Навчальний посібник. — 2-е видання, перероблене і доповнене — М.: Фінанси і статистика, 2020.
19. Строєнко В. Н., Українцев В. Б. Теоретичні основи логістики. — Ростов н/Д: «Фенікс», 2021.
20. Уотерс Д. Логістика. Управління ланцюгом постачань. — М.: ЮНІТІ–ДАНА, 2020. — 503 с.
21. Cooper J., Browne M., Peters M. European Logistics. — Oxford: Blackwell Publishers, 2020.
22. Jonsons J.C., Wood D.F. Contemporary Logistics, 4-те видання. — Нью-Йорк: MacMillan, 2021
23. Moller C., Johanser J., Paradigms in Logistics з Департаменту виробництва Університету Ольборга, Данія, 20019.
24. Stock J.R., Lamber D.M., Strategic Logistics Management, 2-е видання. — Homewood Cliffs, IL: Richard P. Irwin, 2020.
25. Бордовський М. В., Бондаренко А. А., Єрмоленко В. І. Геологія та геохімія нафти і газу: навчальний посібник. – Київ: Недра, 2020. – 288 с.
26. Добрянська. А. Ф., [інші автори]. Перетворення нафти в природі; ред. Добрянська А. Ф. – Київ: Гостоптехиздат, 2020. – 416 с.
27. Бенашвілі Є. М. Розділення вуглеводневих і гетероатомних сполук нафти. – Тбілісі: Мецніереба, 2020. – 152 с.
28. Геологія та геохімія нафти і газу. – Казань: Казанський університет, 2021. – 4 с.
29. Добрянський А. Ф. Хімія нафти. – Донецьк: Гостоптехиздат, 2020. – 224 с.

30. Максим'юк М. Р., Міцкевич Д. І., Міцкевич А. І. Нафтове забруднення поверхневих вод та шляхи подолання його наслідків // Наукові праці. Техногенна безпека – 2019. – Том 233, Випуск 221, Сторінки 37–40.
31. Білявський Г. О., Фурдуй Р. С., Костіков І. Ю. Основи екології: підручник, 2-ге видання. – Київ: Либідь, 2020. – 408 с.
32. Курілецький Л. Дослідження впливу нафтових плівок на випаровування з водної поверхні за допомогою евапориметра // Проблеми фонового моніторингу стану природного середовища: Збірка наукових праць, 2020, Випуск 8, Сторінки 136-144.
33. Мирник С. С., Немировська І. А. Особливості процесу випаровування з водної поверхні за наявності тонких плівок нафти // Водні ресурси, 2020, № 1, Сторінки 75–82.
34. Адаменко О. М., Рудько Г. І. Екологічна геологія. – Київ: Манускрипт, 2020. – 348 с.
35. Губко В. Н., Коваленко А. А., Екологія геологічного середовища: Навчальний посібник. – Херсон: БГУ, 2020. – 120 с.
36. Гукало В. С. Навчально-методичний посібник "Екологія нафтовидобувного комплексу". Київ: Видавництво «Шевченклвський університет», 2019 р. — 174 с.
37. Гірський А.І., Охорона навколишнього середовища в нафтовій та газовій промисловості / А.І. Булавінов, П.П. Макаренко, В.Ю. Щербаків. — Київ: Недра, 2021 р. — 483 с.
38. Звернення з відходами у Київській області: щорічний інформаційно-аналітичний збірник / укладач і відповідальний редактор Є.Ю. Білка. — Львів: Фонд соціально-екологічної реабілітації, 2020 р. — 76 с.
39. Дерій С.Л., Нафта як паливний ресурс і забруднювач навколишнього середовища. — Київ: Видавництво РУДН, 2020 р. — 131 с.
40. Тел Л.Г., Кім Б.І., Зоненко В.І., Охорона навколишнього середовища при будівництві та експлуатації газо-нафто провідних систем. — Київ: Недра, 2020 р.

41. Кам’янець Ф.А., Видалення нафтопродуктів з водної поверхні та ґрунту / Ф.А. Кам’янець, Є.І. Богомольний. — Київ: Інститут комп'ютерних досліджень: Регулярна і хаотична динаміка, 2020 р. — 525 с.
42. Науменко А.Н., Використання сорбентів для ліквідації нафтових розливів / А.Н. Нагірний, В.Н. Хлібороб: Київ — Нефтяне господарство, 2020 р. — 67 с.
43. Реабілітація ґрунтів, забруднених аварійними виливами нафтопродуктів, доц. Н.М. Гринчишин, к.с.-г.н., доц. О.Ф. Бабаджанова, к.т.н., ЛДУ БЖД. 2020.
44. Сковорода О.П., Методи очищення ґрунту після аварійних розливів нафти та нафтопродуктів, Політехнічний молодіжний журнал, 2020 № 02 (43).
45. Шубенко Д.Ю. Досвід та перспективи використання біопрепаратів для рекультивації ґрунтів, забруднених нафтою. 2020. Шевенківський університет, м. Київ.
46. Щербаков О. О. Регламент розробки нафтових і газових родовищ: Затверджено 15.03.2017 №118 Міністерством екології та природних ресурсів України. 2020.