

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Методика організації вантажопасажирських перевезень на основі
хмарних логістичних технологій»

на здобуття освітнього ступеня магістра
зі спеціальності 121 Інженерія програмного забезпечення
освітньо-професійної програми «Інженерія програмного забезпечення»

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

_____ Андрій АГАШКОВ
(підпис)

Виконав: здобувач вищої освіти групи ПДМ-61
_____ Андрій АГАШКОВ

Керівник: _____ Світлана ШЕВЧЕНКО
к.пед.н., доцент

Рецензент: _____

Київ 2025

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

Навчально-науковий інститут інформаційних технологій

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти Магістр

Спеціальність 121 Інженерія програмного забезпечення

Освітньо-професійна програма «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри

Інженерії програмного забезпечення

_____ Ірина ЗАМРІЙ

«_____» _____ 2024 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

_____ Агашкову Андрію Юрійовичу _____

1. Тема кваліфікаційної роботи: «Методика організації вантажопасажирських перевезень на основі хмарних логістичних технологій»

керівник кваліфікаційної роботи Світлана ШЕВЧЕНКО, к.пед.н., доцент,

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «15» жовтня 2024 р. №320.

2. Строк подання кваліфікаційної роботи «17» грудня 2024 р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література з питань, пов'язаних з темою роботи, хмарні технології, графічні методи оптимізації, програмні засоби розробки інформаційної системи.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Провести аналіз існуючих рішень для логістичних операцій на основі хмарних технологій.

2. Дослідити методи організації вантажопасажирських перевезень з використанням хмарних технологій.

3. Розробити модель застосування методики організації вантажопасажирських перевезень на основі хмарних технологій для малого і середнього бізнесу.

5. Перелік графічного матеріалу: *презентація*

1. Наявні методи організації вантажопасажирських перевезень на основі хмарних логістичних технологій

2. Алгоритм роботи методики.

3. Програмна реалізація розробленої методики.

4. Аналіз ефективності розробленої методики.

6. Дата видачі завдання «16» жовтня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	16.10-20.10.24	
2	Написання 1-го розділу	21.10-28.10.24	
3	Написання 2-го розділу	29.10-5.11.24	
4	Написання 3-го розділу	06.11-09.11.24	
5	Підготовка тез і статті до друку	10.11-14.11.24	
6	Підготовка вступу, висновків, реферату	15.11-19.11.24	
7	Розробка обов'язкових матеріалів	21.11-24.11.24	
8	Проведення попереднього захисту роботи	25.11.24	

Здобувач вищої освіти

(підпис)

Андрій АГАШКОВ

Керівник

кваліфікаційної роботи

(підпис)

Світлана ШЕВЧЕНКО

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 104 стор., 15 рис., 10 табл., 38 джерел.

Мета роботи - підвищення ефективності, прозорості та безпеки логістичних процесів вантажопасажирських перевезень за допомогою розробленої методики з використанням хмарних технологій.

Об'єкт дослідження - процес організації вантажопасажирських перевезень.

Предмет дослідження – методика організації вантажопасажирських перевезень на основі хмарних логістичних технологій.

Методи дослідження – аналіз даних, порівняльний, системно-структурний, методи математичного моделювання, графічні методи оптимізації.

Короткий зміст роботи. Досліджено сучасний стан та особливості вантажопасажирських перевезень, проаналізовано існуючі методи їх організації, виділено їх переваги та недоліки. Розроблено концептуальну модель організації перевезень на основі хмарних технологій. Запропоновано алгоритми оптимізації маршрутів та розподілу ресурсів. Реалізовано програмне забезпечення для управління замовленнями з використанням Python та Google Cloud Platform. Проведено оцінку ефективності впровадження розробленої методики.

КЛЮЧОВІ СЛОВА: ВАНТАЖОПАСАЖИРСЬКІ ПЕРЕВЕЗЕННЯ, ХМАРНІ ТЕХНОЛОГІЇ, ЛОГІСТИКА, ОПТИМІЗАЦІЯ МАРШРУТІВ, УПРАВЛІННЯ ЗАМОВЛЕННЯМИ, PYTHON, GOOGLE CLOUD PLATFORM.

ABSTRACT

Text part of the master's qualification work: 104 pages, 15 pictures, 10 tables, 38 sources.

The purpose of the work is to develop a methodology for organizing cargo and passenger transportation using cloud technologies to improve the efficiency, transparency and security of logistics processes.

Object of research - the process of organizing cargo and passenger transportation.

Subject of research - methods and means of optimizing the transportation process based on cloud logistics technologies.

Summary of the work: The current state and features of cargo and passenger transportation were studied, existing methods of their organization were analyzed. A conceptual model of transportation organization based on cloud technologies was developed. Algorithms for route optimization and resource allocation were proposed. Order management software was implemented using Python and JSON. The effectiveness of the implemented methodology was evaluated.

KEYWORDS: CARGO AND PASSENGER TRANSPORTATION, CLOUD TECHNOLOGIES, LOGISTICS, ROUTE OPTIMIZATION, ORDER MANAGEMENT, PYTHON, JSON.

ЗМІСТ

ВСТУП.....	10
1 ТЕОРЕТИЧНІ ОСНОВИ ОРГАНІЗАЦІЇ ВАНТАЖОПАСАЖИРСЬКИХ ПЕРЕВЕЗЕНЬ	13
1.1 Сучасний стан та особливості вантажопасажирських перевезень	13
1.2 Аналіз існуючих методів організації вантажопасажирських перевезень ..	18
1.3 Роль хмарних технологій у логістиці та транспортній галузі	24
2 РОЗРОБКА МЕТОДИКИ ОРГАНІЗАЦІЇ ВАНТАЖОПАСАЖИРСЬКИХ ПЕРЕВЕЗЕНЬ З ВИКОРИСТАННЯМ ХМАРНИХ ЛОГІСТИЧНИХ ТЕХНОЛОГІЙ	30
2.1 Концептуальна модель організації вантажопасажирських перевезень на основі хмарних технологій.....	30
2.2 Алгоритми оптимізації маршрутів та розподілу ресурсів з використанням хмарних обчислень.....	38
2.3 Інтеграція хмарних логістичних технологій у систему управління вантажопасажирськими перевезеннями	43
2.4 Етапи формування методики організації перевезень	47
2.4.1 Постановка завдання.....	48
2.4.2 Критерії оцінки якості процесів вантажопасажирських перевезень	51
2.4.3 Формування множини структурних елементів та аналіз їх доцільності .	55
2.4.4 Побудова алгоритму роботи структурних елементів в методиці	57
2.5 Забезпечення безпеки баз хмарних даних у контексті вантажопасажирських перевезень.....	61
3. ВПРОВАДЖЕННЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ РОЗРОБЛЕНОЇ МЕТОДИКИ.....	65
3.1 Практична реалізація методики організації вантажопасажирських перевезень на основі хмарних логістичних технологій	65
3.1.1 Обґрунтування вибору середовища розробки	70
3.1.2 Програмна реалізація	73
3.1.3 Тестування	78
3.2 Аналіз ефективності впровадження розробленої методики	81
3.3 Перспективи розвитку та вдосконалення методики організації вантажопасажирських перевезень з використанням хмарних технологій.....	85
ВИСНОВКИ.....	89

ПЕРЕЛІК ПОСИЛАНЬ	92
ДОДАТОК А. ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ	96
ДОДАТОК Б. ЛІСТИНГ ПРОГРАМНИХ МОДУЛІВ.....	103

ВСТУП

З кожним роком збільшується попит на ефективне управління вантажопасажирськими перевезеннями, зумовлений зростанням міжнародних і внутрішніх логістичних операцій, що потребують інтеграції новітніх технологій для забезпечення максимальної оперативності та безпеки. У зв'язку з цим питання використання хмарних технологій у логістиці набуває значної актуальності. Цифровізація логістичних процесів дозволяє зменшити витрати, оптимізувати маршрути, а також забезпечити безперебійний обмін інформацією між усіма учасниками логістичного ланцюга, що є критичним для сучасних компаній. Хмарні технології в логістиці стали фундаментальною частиною багатьох організаційних процесів, зокрема у контексті організації вантажопасажирських перевезень. Їх використання забезпечує централізацію та доступність даних у режимі реального часу, що дозволяє оперативно реагувати на зміни та оптимізувати роботу транспорту. Крім того, застосування таких технологій є важливим з точки зору екологічної стійкості, адже оптимізація маршрутів сприяє скороченню викидів CO₂, що є актуальним у сучасному світі. Застосування хмарних технологій дає можливість інтегрувати системи управління перевезеннями (TMS) з іншими корпоративними інформаційними системами, що дозволяє забезпечити комплексний підхід до управління логістичними операціями. Така інтеграція особливо актуальна для великих транспортних компаній, які мають кілька відділень у різних регіонах. Завдяки централізованій обробці та доступу до інформації з будь-якої точки, хмарні системи дозволяють значно підвищити ефективність і якість обслуговування клієнтів.

Впровадження хмарних технологій у логістичних компаніях також дозволяє знизити витрати на IT-інфраструктуру, забезпечуючи одночасний доступ до потужних обчислювальних ресурсів. Це сприяє зниженню витрат на адміністрування, а також забезпечує надійність і стабільність роботи системи. За допомогою хмарних технологій стало можливим легко масштабувати систему,

збільшуючи обсяг оброблюваних даних у періоди високої активності без необхідності великих капіталовкладень.

Метою дослідження є підвищення ефективності, прозорості та безпеки логістичних процесів вантажопасажирських перевезень за допомогою розробленої методики з використанням хмарних технологій.

Основні завдання дослідження включають:

- аналіз сучасного стану та основних тенденцій розвитку хмарних логістичних технологій;
- розробка концептуальної моделі організації перевезень на основі хмарних обчислень;
- обґрунтування вибору технологій для розробки системи;
- створення та тестування програмного забезпечення, що реалізує запропоновану методику;
- оцінка ефективності впровадження хмарних технологій на основі отриманих даних.

Об'єктом дослідження виступає процес організації вантажопасажирських перевезень.

Предметом дослідження є методика організації вантажопасажирських перевезень на основі хмарних логістичних технологій.

Для написання роботи використано такі **методи дослідження**, як аналіз даних, порівняльний, системно-структурний, методи математичного моделювання, методи оптимізації на графах.

Наукова новизна отриманих результатів полягає у розробці методики організації вантажопасажирських перевезень на основі інтеграції хмарних технологій та JSON-форматування даних, що забезпечує оптимізацію логістичних процесів. Удосконалено підхід до управління замовленнями через впровадження автоматизованої системи валідації даних, що знижує рівень помилок до 1.25%. Запропоновано метод забезпечення безпеки даних на основі архітектури "нульової довіри" для логістичних систем. Розроблено алгоритм

оптимізації маршрутів з використанням хмарних обчислень, що дозволяє знизити транспортні витрати на 20%.

Практична значущість отриманих результатів:

- розроблено програмне забезпечення для управління вантажопасажирськими перевезеннями з використанням Python, Google Cloud Platform;
- впроваджено систему управління замовленнями, що скорочує час обробки даних до 40%;
- впроваджено механізм резервного копіювання даних у хмарному середовищі;
- впроваджено механізм знаходження оптимального маршруту, з урахуванням погодних умов та завантаженості доріг;
- реалізовано інтерфейс користувача для зручної роботи з системою.

Робота пройшла апробацію шляхом публікації статті та тез доповідей на конференціях:

1. Агашков А.Ю. Шляхи підвищення ефективності транспортних операцій за допомогою хмарних логістичних рішень // А.Ю. Агашков, В.В. Жебка, С.М. Шевченко / Телекомунікаційні та інформаційні технології, №4, 2024. Подано до друку.

2. Агашков А.Ю. Використання Amazon Web Services у логістиці вантажопасажирських перевезень // Всеукраїнська науково-технічна конференція «Виклики та рішення в програмній інженерії». Збірник тез. 24.11.2024. – Київ: ДУІКТ, 2024. – С. 17-22.

3. Агашков А.Ю. Хмарні логістичні технології у вантажопасажирських перевезеннях // VII Міжнародна науково-практична конференція молодих вчених та студентів «Інженерія програмного забезпечення і передові інформаційні технології». Збірник тез. 20-22 листопада 2024. – Київ: КПІ імені Ігоря Сікорського, 2024. – С. 9-10.

1 ТЕОРЕТИЧНІ ОСНОВИ ОРГАНІЗАЦІЇ ВАНТАЖОПАСАЖИРСЬКИХ ПЕРЕВЕЗЕНЬ

1.1 Сучасний стан та особливості вантажопасажирських перевезень

Вантажопасажирські перевезення є невід'ємною частиною транспортної системи будь-якої країни, забезпечуючи переміщення як пасажирів, так і вантажів. Вони відіграють важливу роль у підтриманні економічної стабільності та сприяють соціально-економічному розвитку суспільства. В сучасних умовах, коли обсяги перевезень зростають, важливим стає вдосконалення методів управління цими процесами з використанням новітніх технологій, таких як хмарні логістичні платформи. Цей підхід дозволяє значно оптимізувати процеси організації вантажопасажирських перевезень, забезпечуючи їхню ефективність, прозорість та гнучкість.

Значення вантажопасажирських перевезень обумовлюється їхньою здатністю забезпечувати мобільність населення та своєчасну доставку вантажів. Це особливо важливо для регіонів з недостатньо розвиненою інфраструктурою або в умовах складного рельєфу, де важливо одночасно забезпечити доступ до послуг транспорту для пасажирів та доставку товарів. Вантажопасажирські перевезення поєднують у собі елементи пасажирського та вантажного транспорту, що дозволяє оптимально використовувати транспортні засоби, особливо в умовах великої відстані між населеними пунктами [1, с. 106].

Сучасний стан вантажопасажирських перевезень характеризується високим рівнем конкуренції, змінами у запитах споживачів та зростаючими вимогами до якості обслуговування. Для успішного функціонування перевізники повинні адаптуватися до змін ринку, пропонуючи нові, більш гнучкі рішення для клієнтів. Одним із таких рішень є впровадження хмарних технологій, що дозволяють не тільки оптимізувати процеси управління, а й значно знизити витрати та підвищити надійність послуг. Особливістю сучасних

вантажопасажирських перевезень є їхня багатофункціональність, тобто здатність забезпечувати перевезення одночасно як вантажів, так і пасажирів. Це особливо актуально для сільських та віддалених регіонів, де попит на транспорт є недостатнім для окремого вантажного чи пасажирського транспорту. В таких умовах поєднання обох функцій дозволяє максимально ефективно використовувати транспортні засоби, мінімізуючи холості рейси та знижуючи витрати на обслуговування. Застосування хмарних технологій у вантажопасажирських перевезеннях відкриває нові можливості для управління транспортними потоками та інтеграції інформаційних систем. Хмарна логістика дозволяє операторам швидко обробляти великі обсяги даних про маршрути, навантаження та потреби клієнтів, що сприяє підвищенню оперативності та точності рішень. Це також забезпечує доступ до актуальної інформації для всіх учасників процесу, що підвищує прозорість та довіру між клієнтами та операторами.

Розвиток цифрових технологій у логістиці створює передумови для запровадження нових бізнес-моделей у сфері вантажопасажирських перевезень. Зокрема, хмарні платформи дозволяють об'єднувати різні компанії на єдиній інформаційній базі, що дає можливість оптимізувати використання транспортних засобів, зменшити витрати на адміністрування та покращити якість обслуговування клієнтів. Це підхід сприяє формуванню нових партнерських відносин та кооперації між перевізниками. Управління вантажопасажирськими перевезеннями на основі хмарних технологій дозволяє мінімізувати ризики, пов'язані з затримками та порушеннями графіків. Такі системи здатні своєчасно виявляти проблеми, наприклад, затори або аварії на дорогах, та пропонувати альтернативні маршрути. Це сприяє підвищенню надійності перевезень та зниженню загального часу в дорозі, що є важливим фактором як для пасажирів, так і для замовників вантажоперевезень [2, с. 45].

Актуальним є питання екологічної стійкості вантажопасажирських перевезень, яке також можна частково вирішити за допомогою хмарних технологій. Оптимізація маршрутів і навантаження транспортних засобів

дозволяє зменшити споживання палива та обсяги викидів шкідливих речовин в атмосферу. Це не тільки відповідає вимогам сучасного екологічного законодавства, але й сприяє створенню позитивного іміджу компанії на ринку.

Важливою особливістю сучасних вантажопасажирських перевезень є зміна вимог до безпеки. Застосування хмарних технологій дозволяє підвищити рівень безпеки через використання систем моніторингу стану транспортних засобів у реальному часі. Це дозволяє оператору відстежувати технічний стан транспорту та вчасно проводити необхідні ремонтні роботи, що мінімізує ризики аварійних ситуацій. Ще одним важливим аспектом є адаптація перевезень до індивідуальних потреб клієнтів, що стала можливою завдяки хмарним технологіям. Наприклад, замовники можуть замовляти додаткові послуги, змінювати маршрути та індивідуально налаштовувати параметри перевезень, що значно підвищує рівень задоволеності клієнтів. Це є одним із способів досягнення конкурентної переваги на ринку.

Інновації у сфері вантажопасажирських перевезень також включають інтеграцію з мобільними додатками, які дозволяють пасажирам і замовникам вантажів контролювати процес перевезення в режимі реального часу. Це забезпечує більшу зручність для користувачів і допомагає скоротити час на обробку замовлень, що позитивно впливає на загальну ефективність системи.

Зважаючи на перераховані фактори, можна стверджувати, що сучасний стан вантажопасажирських перевезень відображає тенденції до цифровізації та впровадження новітніх технологій. Хмарні логістичні технології стають ключовим елементом у забезпеченні гнучкості, надійності та економічної ефективності цих процесів, що є важливим для сталого розвитку транспортної галузі в цілому.

Зростання значення хмарних технологій у логістиці вантажопасажирських перевезень обумовлено також потребою у швидкому доступі до інформації про стан руху, що забезпечує більш точне планування маршрутів та управління ресурсами. Це дозволяє адаптувати логістичні процеси до швидко змінюваних умов ринку, підвищуючи конкурентоспроможність компаній.

Важливою характеристикою вантажопасажирських перевезень у сучасних умовах є їхня комплексна інтеграція з іншими видами транспорту, що дозволяє досягти більшої гнучкості та ефективності. Наприклад, у разі комбінованих перевезень, коли на різних етапах маршруту використовується як автомобільний, так і залізничний або авіаційний транспорт, хмарні логістичні системи стають критично необхідними для координації всіх учасників процесу. Інтеграція транспортних систем сприяє економії ресурсів і часу, а також забезпечує можливість швидко адаптуватися до змін у попиті на перевезення, що є актуальним для сучасної логістики [3, с. 92]. Прикладом успішного використання хмарних технологій у вантажопасажирських перевезеннях є впровадження так званих «розумних» логістичних платформ, таких як FreightOS, Transmetrics та Flexport. В таблиці 1 нижче наведена порівняльна характеристика цих рішень, їх переваги та недоліки.

Таблиця 1.1

Порівняльна характеристика наявних рішень

Назва	FreightOS	Transmetric	Flexport
Побудова оптимального маршруту	+	+	+
Необхідність постійного зв'язку з інтернетом	+	+	+
Запропонований функціонал IoT (Інтернет Речей)	Середній	Сильний для великого, слабкий для малого та середнього бізнесу	Сильний для великого, слабкий для малого та середнього бізнесу
Масштабування	+	+	+
Складність у використанні	Низька	Середня	Середня
Ціна використання	Висока для малого та середнього бізнесу.	Висока для малого та середнього бізнесу.	Висока для малого та середнього бізнесу.
Інтегрованість зі сторонніми сервісами	Висока	Середня	Середня

Як видно із таблиці, сучасні рішення все більше орієнтуються на потреби великого бізнесу, в той час як починаючі підприємці чи вузькоспеціалізовані підприємства мають чималі проблеми із логістикою. Популярні хмарні рішення, здебільшого, вимагають великого стартового капіталу для інтеграції та подальшої підтримки роботи. Тому, зазвичай, малий та середній бізнес орієнтується на найбільш дешевий варіант доставки, нерідко за рахунок клієнта.

Наукова література вказує на те, що використання хмарних технологій у логістиці дозволяє підвищити рівень продуктивності на 20-30%, оскільки автоматизуються процеси координації та управління потоками. Наприклад, дослідження показали, що впровадження систем управління перевезеннями (TMS) на базі хмарних технологій значно знижує ризик помилок, які можуть виникати через людський фактор, і сприяє зростанню точності в розрахунках часу прибуття транспортних засобів [4, с. 66–72]. Це, у свою чергу, дозволяє забезпечити точність доставки, що є важливим фактором для клієнтів і підвищує їхню лояльність. Хмарні логістичні платформи також надають можливість проведення детального аналізу транспортних потоків та ефективності окремих маршрутів. Зокрема[5],

Підвищення якості обслуговування є ще одним важливим аспектом сучасних вантажопасажирських перевезень. Наприклад, використання хмарних технологій дозволяє автоматизувати процеси інформування клієнтів про статус їхніх замовлень, надаючи їм можливість відстежувати перевезення в режимі реального часу. Це підвищує задоволеність клієнтів, оскільки вони завжди мають доступ до актуальної інформації та можуть оперативно реагувати у разі виникнення будь-яких змін. Одним із прикладів успішного застосування хмарних технологій є компанія DHL, яка використовує хмарну платформу для координації своїх логістичних процесів. Завдяки цьому компанія здатна забезпечувати доставку товарів у встановлені терміни, незалежно від географічного розташування або складності маршрутів [6, с. 115]. Використання хмарних технологій також дозволило DHL знизити витрати на управління

ланцюгом постачання та значно підвищити точність і надійність своєї логістичної системи.

Інтеграція хмарних технологій з інтернетом речей (IoT) відкриває нові можливості для моніторингу транспортних засобів. Пристрої IoT, встановлені на транспортних засобах, можуть передавати дані про температуру, вологість, вібрацію та інші параметри, які є критичними для певних типів вантажів. Наприклад, у випадку перевезення продуктів харчування або медичних препаратів, зміни температури можуть негативно вплинути на якість вантажу. Хмарна система може автоматично надсилати попередження у разі виявлення критичних відхилень, що дозволяє вчасно вжити заходів і уникнути втрат.

Таким чином, хмарні технології стають ключовим елементом у побудові надійних, безпечних і ефективних вантажопасажирських перевезень. Вони дозволяють оптимізувати логістичні процеси, підвищити точність доставки, мінімізувати вплив людського фактора та забезпечити високу гнучкість у роботі з клієнтами. У результаті компанії, що впроваджують такі технології, отримують конкурентну перевагу на ринку, що підтверджується численними прикладами успішного використання хмарних рішень у сфері логістики та транспорту.

1.2 Аналіз існуючих методів організації вантажопасажирських перевезень

Аналіз існуючих методів організації вантажопасажирських перевезень є необхідним етапом для розуміння їхньої ефективності, гнучкості та здатності адаптуватися до сучасних умов. Ці методи базуються на комплексних підходах до координації транспортних засобів, маршрутизації, управління ресурсами та забезпечення взаємодії між усіма учасниками процесу перевезення. Існуючі підходи до організації вантажопасажирських перевезень відрізняються залежно від характеру вантажу, типу транспорту та інфраструктурних можливостей регіону. Вони включають як традиційні методи, що сформувалися історично, так і сучасні технологічні підходи, орієнтовані на використання цифрових рішень та

автоматизації процесів [7, с. 87]. Одним із основних методів організації вантажопасажирських перевезень є централізована модель управління, яка передбачає контроль і планування на центральному рівні, зазвичай на рівні транспортної компанії чи оператора логістичних послуг. У такій моделі всі рішення щодо розкладу, маршрутизації, навантаження та розвантаження приймаються центральною адміністрацією, що дозволяє забезпечити високий рівень контролю. Однак, централізовані методи мають свої недоліки, зокрема обмежену гнучкість і складність адаптації до змінних умов на ринку або змін у потребах клієнтів.

Децентралізовані методи організації перевезень, навпаки, дозволяють делегувати прийняття рішень на місцях, що підвищує адаптивність і гнучкість процесів. Такі методи часто використовуються в умовах, де операторам необхідно швидко реагувати на зміни в попиті або обставинах. Наприклад, при вантажопасажирських перевезеннях у регіонах з поганою дорожньою інфраструктурою чи обмеженою кількістю маршрутів децентралізація дозволяє ефективніше керувати логістичними процесами, враховуючи специфічні умови на місцевому рівні. Традиційні методи організації вантажопасажирських перевезень, зокрема ті, що базуються на ручному плануванні маршрутів та графіків, мають значний вплив на сучасні логістичні операції. Вони включають аналіз попереднього досвіду та історичних даних для планування майбутніх перевезень. Хоча ці методи можуть бути ефективними для малих перевізників чи у випадках стабільного та передбачуваного попиту, вони часто є неефективними в умовах швидких змін у попиті та ринку. Зокрема, відсутність автоматизації та використання цифрових систем у таких методах створює ризик затримок і неточностей.

Більш сучасні підходи включають використання алгоритмів оптимізації маршрутів, що дозволяє досягти максимальної ефективності в плануванні перевезень. Такі алгоритми можуть бути інтегровані в системи управління транспортом (TMS) і використовуються для розрахунку найкоротших або найекономічніших маршрутів [8, с. 143]. Оптимізація маршрутів базується на

використанні великих даних (Big Data) та математичних моделей, що дозволяє враховувати такі фактори, як час у дорозі, витрати палива, погодні умови та наявність завантаженості доріг. Завдяки такому підходу оператори можуть знизити витрати та підвищити точність дотримання графіків перевезень. Методологія маршрутного планування в умовах змішаних перевезень також є важливим напрямом в організації вантажопасажирських перевезень. Змішані перевезення передбачають використання різних видів транспорту для досягнення максимальної гнучкості та мінімізації витрат. Наприклад, комбінація залізничного та автомобільного транспорту дозволяє знизити витрати на перевезення на великих відстанях, зберігаючи при цьому високу швидкість доставки. Використання мультимодальних рішень стає все більш популярним, особливо з урахуванням розширення логістичних мереж та інтеграції з міжнародними перевезеннями.

Іншим підходом до організації вантажопасажирських перевезень є так званий метод "точно вчасно" (just-in-time), який був запозичений з виробничих процесів і адаптований для логістики. Цей метод орієнтований на зменшення запасів та зниження витрат на зберігання, при цьому забезпечуючи доставку вантажів саме в момент їх необхідності. Такий підхід вимагає високого рівня координації між учасниками ланцюга поставок та використання передових систем управління, таких як ERP або хмарні логістичні платформи, що дозволяє мінімізувати затримки та підвищити ефективність перевезень. Інноваційні підходи до організації вантажопасажирських перевезень включають використання технології Інтернету речей (IoT) для моніторингу умов транспортування. Завдяки підключенню IoT-датчиків до транспортних засобів, перевізники отримують можливість відстежувати параметри перевезень у режимі реального часу, такі як температура, вологість або рівень завантаженості. Це дозволяє контролювати якість перевезень та миттєво реагувати на зміни умов, що є особливо актуальним для перевезень товарів, чутливих до температурних змін, таких як продукти харчування чи медичні препарати.

Однією з новітніх методологій, що використовується для організації вантажопасажирських перевезень, є інтеграція із системами прогнозування попиту. Ці системи використовують історичні дані, ринкові тренди та статистичні моделі для прогнозування змін у попиті на перевезення. Наприклад, під час свят або сезонних змін попиту система може автоматично коригувати маршрути та ресурси, щоб забезпечити необхідну кількість транспортних засобів у критичний момент. Це допомагає уникнути дефіциту транспорту та покращити обслуговування клієнтів. Управління запасами є ще одним важливим елементом організації вантажопасажирських перевезень, особливо у випадках, коли потрібно забезпечити баланс між попитом і пропозицією на ринку. Використання методів управління запасами дозволяє скоротити витрати на зберігання товарів, що особливо важливо для швидкопсувних товарів або продукції з обмеженим терміном придатності. Інтеграція управління запасами з системами планування перевезень дозволяє забезпечити швидку доставку товарів від постачальників до кінцевих споживачів. Крім того, на сучасному ринку дедалі більшого значення набувають гнучкі системи управління перевезеннями, що дозволяють адаптуватися до зміни попиту та оптимізувати використання ресурсів. Наприклад, компанії можуть застосовувати методи динамічного ціноутворення, що дозволяє змінювати вартість перевезень залежно від часу доби, сезонності або навантаження на транспортні засоби. Це дозволяє збалансувати попит і пропозицію на перевезення, максимізувати дохід і водночас оптимізувати рівень завантаженості транспорту.

Системи автоматичного відстеження маршрутів і транспортних засобів також є важливим елементом сучасної організації вантажопасажирських перевезень. За допомогою GPS та інших технологій перевізники можуть відстежувати маршрут у реальному часі, що дозволяє миттєво реагувати на можливі затримки або проблеми на дорозі. Автоматизація процесів відстеження дозволяє мінімізувати ризики збоїв і забезпечити високу точність дотримання графіків перевезень. Узагальнюючи, можна сказати, що сучасні методи організації вантажопасажирських перевезень є комплексними, адаптивними і

включають інноваційні технології, спрямовані на максимізацію ефективності та точності. Від централізованих та децентралізованих моделей до інтеграції з системами прогнозування та використання IoT – всі ці підходи спрямовані на створення більш гнучкої, надійної та прозорої системи логістики, яка здатна швидко адаптуватися до мінливих умов і потреб ринку. Сучасні методи організації вантажопасажирських перевезень також включають застосування технологій штучного інтелекту (ШІ) для підвищення ефективності управління процесами. ШІ дозволяє аналізувати великі обсяги даних про транспортні потоки, погодні умови, обмеження руху, що в сукупності дає змогу швидше знаходити оптимальні рішення для планування маршрутів та управління ресурсами. Завдяки алгоритмам машинного навчання, компанії можуть передбачати затримки, ризики на маршруті та миттєво змінювати розклад для уникнення проблем. Використання ШІ стає особливо актуальним у великих містах, де щільність дорожнього руху є критичним фактором для своєчасного доставлення вантажів та пасажирів.

Одним із значущих досягнень у сфері організації вантажопасажирських перевезень є інтеграція хмарних платформ із мобільними додатками, що дозволяє учасникам логістичного процесу отримувати інформацію та здійснювати контроль у реальному часі. Мобільні додатки надають клієнтам доступ до актуальної інформації про статус перевезення, а водіям – інструкції та оновлення маршрутів, що покращує точність виконання замовлень і знижує ймовірність затримок. Такі додатки також дозволяють компаніям отримувати зворотний зв'язок від клієнтів, що є важливим для підвищення якості обслуговування та адаптації до їхніх потреб [9, с. 29].

Розвиток технологій блокчейну також поступово знаходить застосування у вантажопасажирських перевезеннях, забезпечуючи прозорість та надійність операцій. Використання блокчейн-технологій дозволяє створити незмінний і доступний для всіх учасників ланцюга постачання реєстр даних, що сприяє підвищенню довіри між перевізниками та клієнтами. Це особливо важливо для міжнародних перевезень, де важливо зберігати точність і безперервність даних

про товар та маршрут. Блокчейн також допомагає автоматизувати виконання контрактів та платіжні операції, що мінімізує ризик помилок і затримок.

Крім того, методи організації вантажопасажирських перевезень продовжують вдосконалюватися через автоматизацію процесів обліку та документообігу. Впровадження електронних накладних, рахунків та контрактів значно спрощує взаємодію між учасниками ланцюга постачання, зменшуючи витрати часу та ресурсу на обробку документації. Автоматизація облікових операцій дозволяє також скоротити помилки, які можуть виникати через людський фактор, і підвищує точність даних, що є ключовим для якісної організації перевезень.

Як висновок, можна навести таблицю сучасних методів, їх переваг та недоліків у логістичних операціях.

Таблиця 1.2

Методи проведення логістичних операцій

Метод	Опис	Переваги, недоліки
Централізована модель	Управління перевезеннями здійснюється центральною адміністрацією, яка приймає рішення щодо маршрутизації, розкладів, навантаження тощо.	+ високий рівень контролю та моніторингу; + стандартизація перевезень; - обмежена гнучкість у змінних умовах ринку;
Децентралізована модель	Рішення приймаються на місцях, що дозволяє швидко реагувати на зміни попиту або обставин.	+ гнучкість та адаптивність до локальних умов; + скорочення часу на прийняття рішень. - складнощі в координації між підрозділами; - сумнівна якість прийнятих рішень;
Оптимізації маршрутів	Використання алгоритмів та математичних моделей для визначення найкоротших або найефективніших маршрутів	+ зниження витрат на паливо та час перевезень; + висока точність планування; - залежність від якісних даних;

Продовження таблиці 1.2

Методи проведення логістичних операцій

Метод	Опис	Переваги, недоліки
Змішані перевезення	Комбінація різних видів транспорту (наприклад, залізничного та автомобільного) для зниження витрат та підвищення гнучкості	+ оптимізація витрат на перевезення на великі відстані; + підвищення швидкості доставки; - складність координації між різними видами транспорту;
«Точно-вчасно» (just-in-time)	Забезпечення доставки вантажу в потрібний момент, зменшення запасів на складах.	+ зниження витрат на зберігання; + мінімізація затримок у поставках; - високі вимоги до координації та точності;
Використання IoT	Моніторинг перевезень у реальному часі за допомогою сенсорів та технології Інтернету речей.	+ підвищення якості перевезень; - залежність від надійності технологій;
Системи прогнозування попиту	Використання історичних даних і моделей для передбачення попиту на перевезення.	+ підвищення ефективності використання ресурсів; - складність у розробці точних моделей;
Блокчейн	Використання технологій блокчейну для забезпечення прозорості та безпеки даних у логістичних процесах.	+ автоматизація платежів та контрактів; - обмежена інтеграція з існуючими системами;

1.3 Роль хмарних технологій у логістиці та транспортній галузі

Хмарні технології відіграють важливу роль у сучасній логістиці та транспортній галузі, сприяючи розвитку інноваційних рішень, підвищенню ефективності й зниженню витрат. Використання хмарних обчислень дозволяє організаціям зберігати, обробляти та передавати великі обсяги інформації без необхідності у власному дорого вартісному обладнанні. Це відкриває нові можливості для обробки логістичних операцій, таких як управління вантажопотоками, оптимізація маршрутів та контроль за транспортом у

реальному часі. Одним із визначень хмарних технологій є модель, що забезпечує доступ до спільного пулу обчислювальних ресурсів через інтернет, включаючи сервери, сховища даних, мережеві ресурси та програмне забезпечення. У контексті логістики хмарні технології забезпечують доступ до інформації в режимі реального часу, що дозволяє логістичним компаніям оперативно реагувати на зміни та приймати стратегічні рішення, що зменшує простой та оптимізує час доставки. Транспортні підприємства часто зіштовхуються з необхідністю обробляти великі обсяги даних, наприклад, дані про транспортні засоби, маршрути, паливні витрати та інші операційні показники. Хмарні рішення дозволяють зберігати ці дані централізовано, забезпечуючи швидкий доступ до них з будь-якої точки світу. Це важливо, зокрема, для міжнародних компаній, які мають численні підрозділи в різних країнах, оскільки спрощує комунікацію та координацію між ними [10, с. 132].

Сьогодні, коли швидкість і точність доставки є ключовими факторами для логістичних компаній, хмарні технології відіграють важливу роль у забезпеченні якості послуг. Один із значних аспектів використання хмарних технологій у логістиці — можливість оптимізації маршрутів. Завдяки спеціальним алгоритмам, що працюють на основі хмарних обчислень, логістичні компанії можуть обирати найбільш економні маршрути, враховуючи умови дорожнього руху, погодні умови та інші фактори. Це не тільки знижує витрати на транспорт, але й сприяє скороченню викидів CO₂, що робить хмарні технології важливим інструментом для екологічної стійкості галузі. Крім того, хмарні обчислення значно полегшують процеси складування і управління запасами. Використання хмарних платформ дозволяє зберігати дані про запаси в реальному часі, відстежувати наявність товарів та управляти замовленнями, що особливо важливо для компаній із високим обігом товарів. Це сприяє зниженню надлишкових запасів, уникненню дефіциту і зменшенню витрат на зберігання, що робить логістичні операції ефективнішими [11, с. 218].

Ще одним важливим аспектом є забезпечення безпеки даних. Хмарні платформи пропонують надійний захист інформації завдяки багатошаровим

системам безпеки, включаючи шифрування даних та контроль доступу. У транспортній галузі, де обробляється значний обсяг чутливих даних, таких як інформація про клієнтів та маршрути доставки, захист даних є ключовим фактором успішної роботи компанії. Також, логістичні компанії можуть легко збільшити чи зменшити обсяги обчислювальних ресурсів залежно від попиту на їхні послуги. Це дає можливість більш гнучко управляти ресурсами, знижуючи витрати та підвищуючи ефективність роботи. Для компаній, що працюють на міжнародному рівні, ця гнучкість є особливо важливою, оскільки дозволяє швидко адаптуватися до змін на ринку. Крім того, хмарні технології сприяють підвищенню рівня автоматизації процесів у логістиці. Завдяки їм можна створювати автоматизовані системи обробки даних та управління, що спрощує виконання рутинних завдань, таких як складання маршрутів, обробка замовлень і управління складськими запасами. Це не лише зменшує ймовірність людських помилок, а й прискорює виконання процесів, що, у свою чергу, позитивно впливає на ефективність компанії. Хмарні рішення також відкривають нові можливості для співпраці між різними учасниками логістичного ланцюга. Через хмарні платформи компанії можуть обмінюватися інформацією з партнерами, такими як постачальники, перевізники та клієнти, у режимі реального часу. Це підвищує прозорість усіх операцій, що полегшує координацію дій і дозволяє швидше реагувати на зміни в ланцюгу поставок [12, с. 270].

Завдяки можливості обробки великих обсягів даних (Big Data) хмарні технології дозволяють здійснювати прогнозну аналітику, яка є важливим інструментом для ухвалення рішень у логістиці. Це дає змогу компаніям передбачати попит на товари, прогнозувати потенційні затримки в доставці та оптимізувати операції відповідно до очікувань ринку. Така аналітика є особливо корисною в умовах глобальних ринків, де зміни відбуваються дуже динамічно.

Ще однією ключовою перевагою хмарних технологій у логістиці є можливість зберігання великих обсягів інформації з можливістю швидкого доступу до неї. Логістичні компанії можуть використовувати ці дані для аналізу своєї роботи, виявлення слабких місць і пошуку шляхів для покращення

ефективності. Таким чином, хмарні технології стають важливим інструментом для забезпечення конкурентоспроможності на ринку. Також вони дозволяють застосовувати рішення з підтримкою штучного інтелекту (AI) у логістиці, що надає змогу ще більш точно аналізувати та прогнозувати потреби ринку, а також автоматизувати процеси, які раніше вимагали ручної роботи. Інтеграція AI у хмарні платформи допомагає підвищити точність прогнозів, що сприяє оптимізації логістичних процесів на всіх етапах, від складу до доставки. Дозволяють створювати інтелектуальні ланцюги поставок, де усі елементи пов'язані в єдину систему управління. Це забезпечує швидкий обмін даними між усіма учасниками ланцюга та дозволяє координувати дії для досягнення максимальної ефективності. У такий спосіб хмарні технології стають невід'ємною частиною сучасної логістики, дозволяючи компаніям адаптуватися до викликів і залишатися конкурентоспроможними.

У логістиці хмарні сервіси поділяються на кілька основних моделей (рис 1.3): SaaS (програмне забезпечення як послуга), PaaS (платформа як послуга) та IaaS (інфраструктура як послуга).

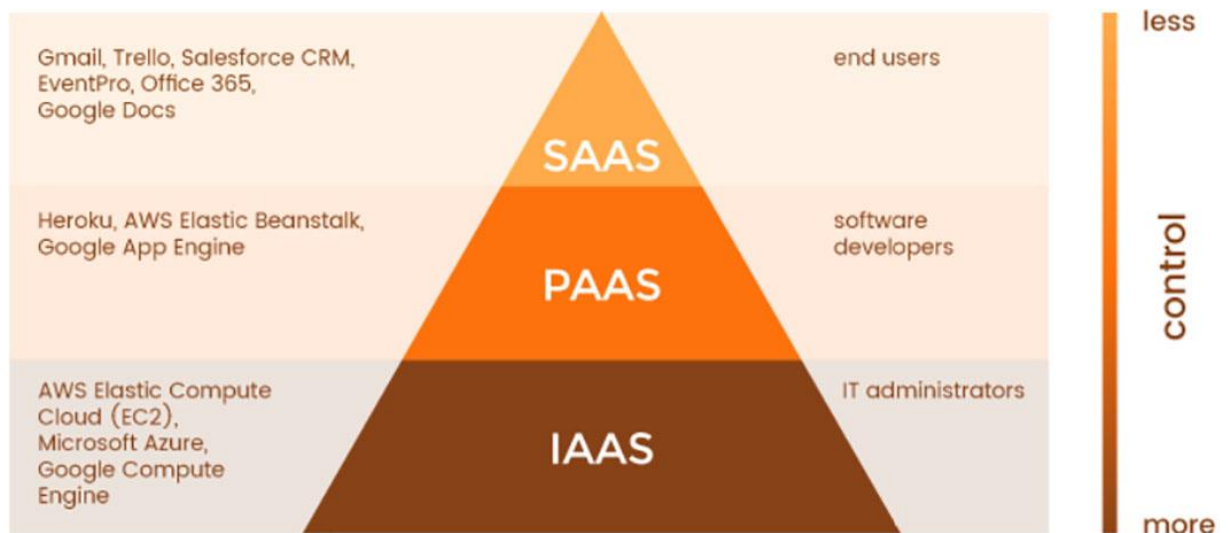


Рис. 1.3 SAAS, PAAS та IAAS моделі.

Кожна з цих моделей має власні переваги й можливості, що сприяють оптимізації логістичних процесів. SaaS, наприклад, дозволяє логістичним компаніям користуватися програмними рішеннями без необхідності інсталяції

локального програмного забезпечення, що значно знижує витрати на IT-інфраструктуру. Одним із найпопулярніших застосувань SaaS у логістиці є платформи для управління транспортом (TMS), які спрощують процеси планування маршрутів, відстеження вантажів та координації між водіями.

SaaS-платформи для управління транспортом надають інструменти для ефективного планування і оптимізації маршрутів, що є критичним для логістичних компаній. Завдяки таким платформам, як SAP Transportation Management та Oracle Transportation Management, логістичні відділи можуть автоматично визначати оптимальні маршрути з урахуванням багатьох змінних: дорожніх умов, часу прибуття, обсягу вантажу та витрат на паливо. Це не лише економить час, але й сприяє зниженню витрат, що робить логістичні процеси більш рентабельними та гнучкими.

PaaS, у свою чергу, надає логістичним компаніям можливість створювати та тестувати власні додатки для обробки логістичних операцій. Використовуючи хмарні платформи на зразок Microsoft Azure або Google Cloud Platform, компанії можуть швидко розробляти індивідуальні рішення для управління ланцюгом поставок, інтегруючи їх з іншими корпоративними системами. Це особливо важливо для великих транспортних компаній, яким необхідно адаптуватися до специфічних вимог різних регіонів або клієнтів, дозволяючи впроваджувати кастомізовані логістичні рішення в стислий термін [13, с. 67]. Інфраструктура як послуга (IaaS) надає компаніям можливість орендувати хмарні сервери, сховища даних та інші інфраструктурні компоненти без необхідності інвестувати у власне обладнання. Google Cloud Platform є одним із провідних провайдерів IaaS, що дозволяє логістичним компаніям гнучко масштабувати свої потужності залежно від поточного попиту. Наприклад, під час пікових навантажень, як-от сезонні свята, компанії можуть тимчасово розширити обчислювальні ресурси для обробки збільшеного обсягу замовлень, а після зниження навантаження повернутися до попереднього рівня ресурсів, що економить витрати.

Використання IaaS також спрощує процеси зберігання та управління великими обсягами даних. Наприклад, DHL використовує IaaS для обробки та

аналізу великих даних, що дозволяє їм робити прогнози щодо попиту, оптимізувати операції і навіть передбачати потенційні проблеми у ланцюгу поставок. Це значно покращує реактивність компанії та забезпечує стабільність логістичних операцій, мінімізуючи ризики збоїв. Водночас, сервіси PaaS також дозволяють впроваджувати інструменти для аналітики, що є критичним для логістики, оскільки забезпечує точність у прогнозах та плануванні. Використовуючи хмарні платформи для розробки й розгортання аналітичних додатків, компанії можуть отримувати повний контроль над даними, що стосується транспортування та логістики. Це дозволяє виявляти вузькі місця у ланцюзі поставок та пропонувати шляхи для їх подолання, підвищуючи ефективність усіх логістичних операцій. Завдяки цьому трикомпонентному підходу, де SaaS, PaaS і IaaS доповнюють одне одного, логістичні компанії отримують можливість більш гнучко управляти ресурсами, підвищувати ефективність і забезпечувати стійкість до ринкових змін. Це робить хмарні сервіси невід'ємною частиною сучасної логістичної індустрії, підвищуючи рівень обслуговування клієнтів і дозволяючи логістичним компаніям залишатися конкурентоспроможними на глобальному ринку.

2 РОЗРОБКА МЕТОДИКИ ОРГАНІЗАЦІЇ ВАНТАЖОПАСАЖИРСЬКИХ ПЕРЕВЕЗЕНЬ З ВИКОРИСТАННЯМ ХМАРНИХ ЛОГІСТИЧНИХ ТЕХНОЛОГІЙ

2.1 Концептуальна модель організації вантажопасажирських перевезень на основі хмарних технологій

Концептуальна модель організації вантажопасажирських перевезень із застосуванням хмарних технологій базується на інтеграції сучасних інформаційних технологій для оптимізації логістичних процесів. Використання хмарних обчислень дозволяє ефективно обробляти та зберігати значний обсяг даних, забезпечуючи автоматизацію операційних процесів і підвищення ефективності управління перевезеннями. У запропонованій програмі, розробленій на Python з використанням JSON-файлів для збереження даних, реалізована система управління замовленнями, яка включає функції створення, редагування та відображення замовлень, що дозволяє централізовано зберігати інформацію про логістичні операції. Основною метою цієї системи є забезпечення оперативного обміну даними, а також доступності інформації про перевезення у режимі реального часу [14, с. 155]. Модель реалізує базові методи взаємодії з даними, що дозволяють користувачам вводити необхідні параметри замовлення, такі як ім'я клієнта, пункт відправлення, пункт призначення та вага вантажу. Ця інформація зберігається у JSON-форматі, що дозволяє зручно її зберігати та обробляти у хмарному середовищі. Програма використовує основні функції, як-от `create_order()`, `edit_order()`, `fetch_orders()` та `save_orders()`, які дозволяють організувати роботу з даними відповідно до вимог сучасної логістики.

Забезпечення збереження та актуалізації даних реалізується за допомогою бібліотеки Tkinter, яка дозволяє організувати зручний графічний інтерфейс користувача (GUI). Цей інтерфейс забезпечує доступ до функцій програми, що

дозволяє не тільки вводити й зберігати інформацію, але і переглядати її у вигляді таблиці, в якій користувач може редагувати кожен запис. Така структура сприяє підвищенню точності введення даних і забезпечує централізований доступ до них для різних підрозділів організації. Завдяки цьому модель забезпечує гнучке управління замовленнями, що є важливим у процесі оптимізації перевезень. Аналіз статистичних даних вказує на те, що застосування хмарних технологій у логістиці дозволяє знизити час обробки замовлень у середньому на 30-40%. Крім того, програми з інтегрованим управлінням замовленнями, подібні до розробленої, можуть зменшити частоту помилок введення даних до 25%, що досягається завдяки автоматизованій перевірці повноти введених даних перед їхнім збереженням у системі. Цей показник є критичним для логістичних компаній, які працюють із великими обсягами інформації, оскільки зниження кількості помилок сприяє підвищенню точності логістичних операцій та мінімізації фінансових втрат. Ключовим елементом моделі є можливість оперативного редагування замовлень, що дозволяє швидко адаптуватися до змінних умов перевезень. Наприклад, якщо клієнт змінює пункт призначення чи вагу вантажу, оператор може внести зміни до запису, і ця інформація одразу оновлюється у системі [15, с. 12]. Це знижує ризики, пов'язані з відправкою невірних замовлень, та підвищує рівень сервісу, що є важливим фактором у конкурентному середовищі транспортної галузі. Модель також передбачає можливість інтеграції алгоритмів оптимізації маршрутів, таких як алгоритм Дейкстри чи генетичні алгоритми, які можуть бути додані на етапі подальшого розвитку програми. Завдяки даним про пункт відправлення та призначення, що зберігаються у JSON, система може забезпечувати побудову оптимальних маршрутів з урахуванням актуальної інформації про дорожні умови та інші фактори. Такий підхід сприяє значному скороченню витрат на транспортні операції та підвищенню екологічної стійкості компанії, зменшуючи викиди CO₂.

Розроблений інтерфейс програми на основі Tkinter відображає основну інформацію про замовлення, зокрема поля введення для даних клієнта, пунктів відправлення та призначення, а також таблицю з інформацією про всі

замовлення. Інтерфейс включає модальне вікно з інформацією про програму, що полегшує процес її використання та забезпечує доступність основних функцій для широкого кола користувачів. Використання зрозумілого GUI спрощує навчання та підвищує швидкість обробки замовлень, що в цілому позитивно впливає на ефективність логістичних операцій.

Використання хмарних технологій у програмі для управління вантажопасажирськими перевезеннями дозволяє ефективно централізувати інформацію про замовлення та забезпечити її доступність у реальному часі. На рисунку 2.1 представлено основний інтерфейс програми, що дозволяє вводити дані про замовлення, зокрема ім'я клієнта, пункт відправлення, пункт призначення та вагу вантажу, а також зберігати ці дані у хмарному сховищі у форматі JSON.

Ім'я	Іван
Місце відправлення	Київ
Місце призначення	Одеса
Вага (кг)	700

Рис. 2.1 Інтерфейс програми

Ця функція забезпечує можливість швидкого доступу до інформації для оновлення та використання іншими підрозділами компанії.

При створенні замовлення, якщо будь-яке з полів залишене порожнім, система автоматично видає повідомлення про помилку. Така функція забезпечує цілісність та точність даних, що є важливим для збереження актуальності інформації, доступної через хмару.

На рисунку 2.2 зображений процес редагування замовлення. Вибране замовлення завантажується у відповідні поля, що дозволяє користувачу швидко внести зміни і зберегти їх. Завдяки хмарному сховищу всі зміни одразу оновлюються у JSON-файлі, забезпечуючи актуалізацію даних для всіх користувачів системи.

Ім'я	Іван
Місце відправлення	Київ
Місце призначення	Одеса
Вага (кг)	7003

Рис. 2.2 Редагування програми

Таким чином, запропонована модель ефективно реалізує концепцію хмарних технологій для централізованого зберігання та управління даними про вантажопасажирські перевезення, підвищуючи точність обробки інформації та швидкість її оновлення.

В процесі розгляду алгоритмів оптимізації маршрутів для організації вантажопасажирських перевезень було проаналізовано кілька підходів, що могли б ефективно виконувати завдання побудови оптимальних маршрутів залежно від умов логістики. Зокрема, було досліджено такі алгоритми, як алгоритм Беллмана-Форда, алгоритм A^* та алгоритм Флойда-Воршелла, які мають свої переваги і можуть ефективно працювати в різних логістичних ситуаціях. Кожен з цих алгоритмів має унікальні особливості, які впливають на швидкість, точність і адаптивність побудови маршрутів для перевезень.

Алгоритм Беллмана-Форда (рисунок 2.3), наприклад, здатний обробляти графи з негативними вагами, що є важливою перевагою в складних логістичних моделях, де можливі коливання цін або часу в негативному діапазоні.

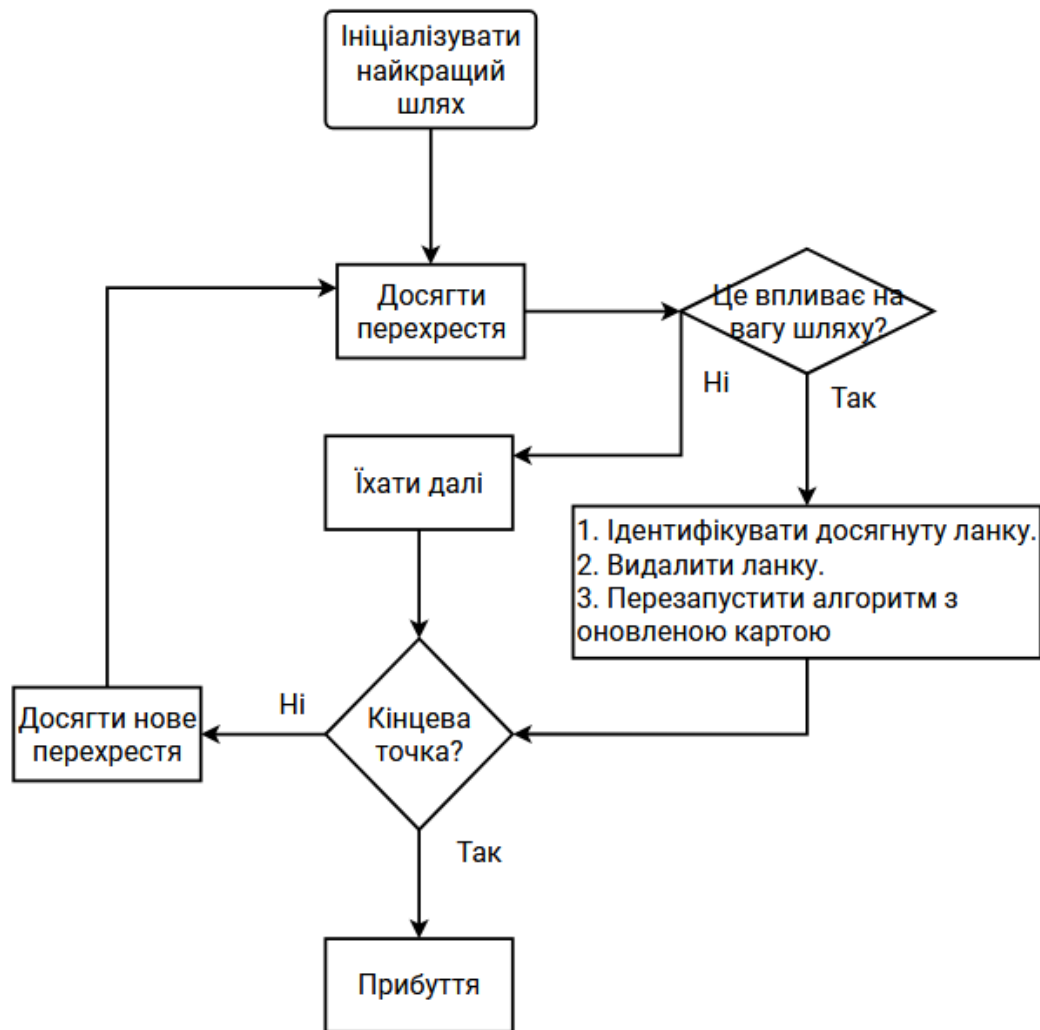


Рис. 2.3 Алгоритм Беллмана-Форда

Проте цей алгоритм має більшу обчислювальну складність порівняно з іншими, що може зробити його менш придатним для великих транспортних систем. Вибір цього алгоритму в певних ситуаціях є доцільним, але для оптимізації у масштабних логістичних мережах потрібен менш ресурсоємний підхід. Алгоритм A^* (рис. 2.4) був також розглянутий як можливий метод оптимізації маршрутів, особливо у випадках, коли важливий швидкий пошук у великих графах.

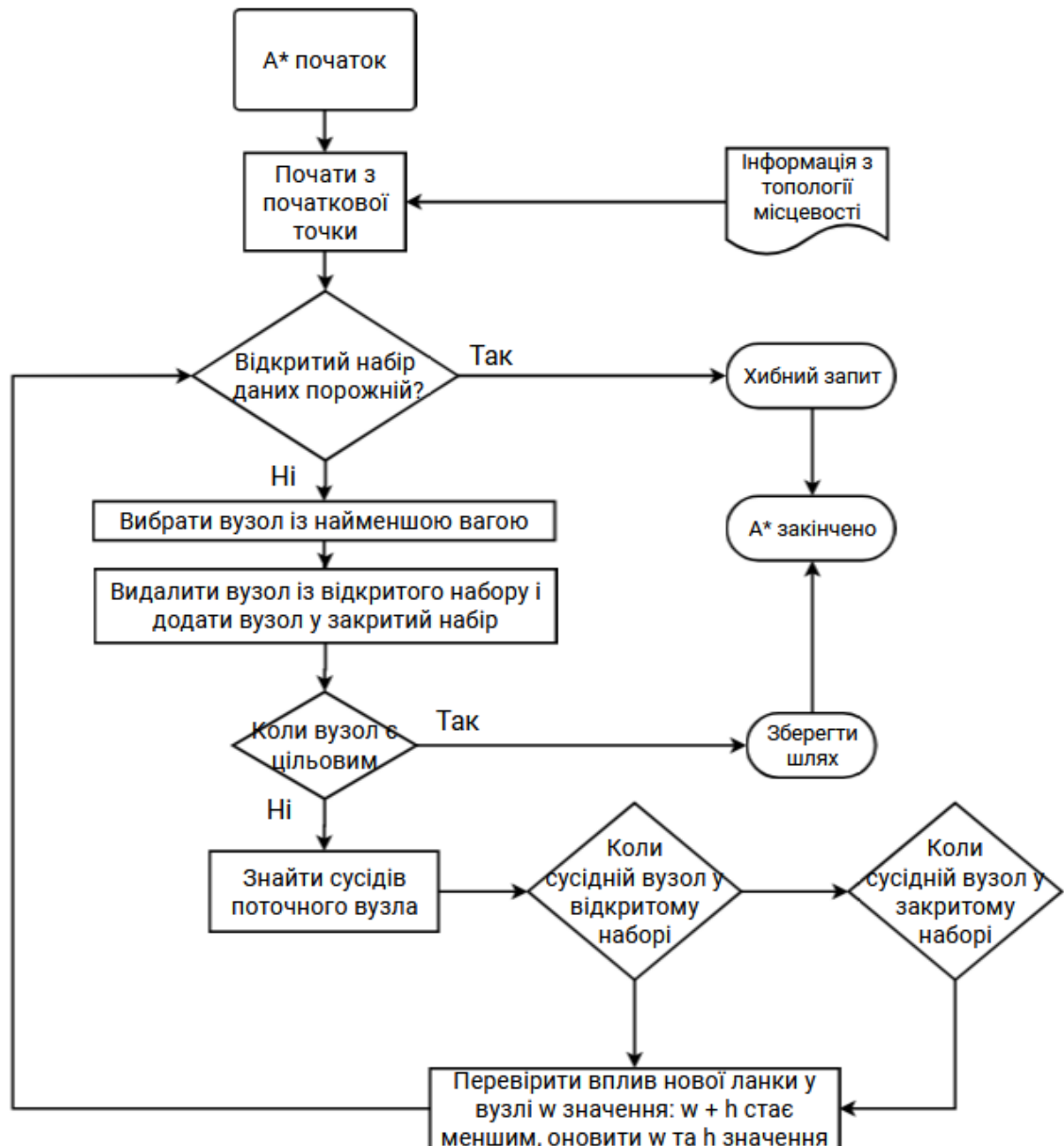


Рис. 2.4 Алгоритм A*

A* ефективний у великих мережах, таких як транспортні системи великих міст, але його застосування вимагає точного налаштування евристик, що може бути складним завданням для логістики, яка працює зі змінними умовами.

Інший розглянутий метод – алгоритм Флойда-Воршелла (рис. 2.5), який ідеально підходить для обчислення найкоротших шляхів між усіма парами вершин. Цей алгоритм зручний для невеликих до середніх графів, де потрібно знати оптимальний маршрут між кожною парою точок.

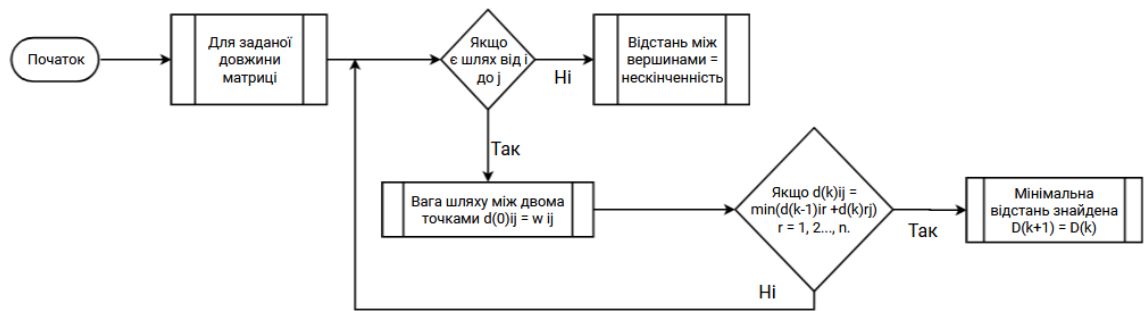


Рис. 2.5 Алгоритм Флойда Воршелла

Однак, через високу обчислювальну складність, алгоритм Флойда-Воршелла не підходить для великих мереж із багатьма вершинами, оскільки потребує значних ресурсів пам'яті і процесорного часу, що обмежує його застосування у глобальних логістичних системах [16, с. 47–61].

Серед усіх розглянутих алгоритмів було вирішено обрати алгоритм Дейкстри, оскільки він поєднує в собі простоту, надійність і ефективність для задач побудови маршрутів у транспортних мережах середньої складності. Алгоритм Дейкстри (рис 2.6) забезпечує швидке визначення найкоротшого шляху від однієї точки до іншої в графі з невід'ємними вагами, що робить його оптимальним для завдань логістики, де основні параметри, такі як відстань і час, мають додатні значення. Його низька обчислювальна складність дозволяє легко масштабувати алгоритм для мереж середнього розміру, що є важливим у реальних умовах логістичних операцій.

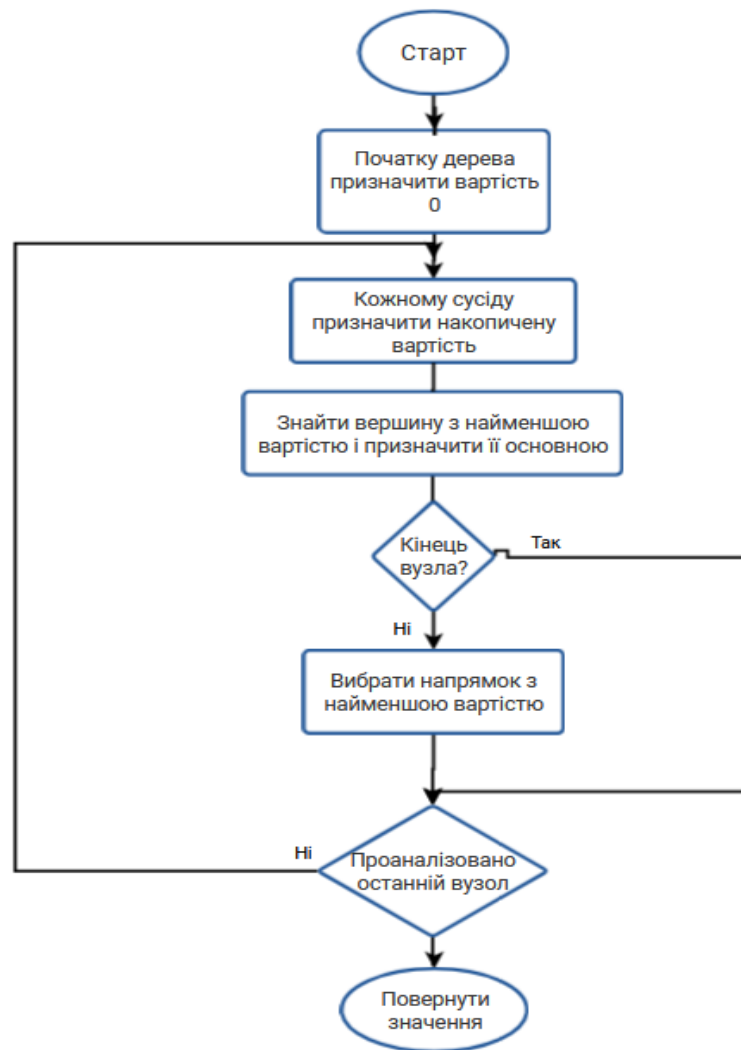


Рис. 2.6 Алгоритм Дейкстри.

Крім того, алгоритм Дейкстри має високу стійкість до змін умов у транспортній мережі. Якщо виникають непередбачені обставини, такі як затори чи аварії, система може швидко перерахувати маршрут, зберігаючи високу точність і швидкість. Це важливо для логістики, оскільки дозволяє оперативно адаптуватися до поточних умов, зменшуючи простой і оптимізуючи витрати на перевезення. Ця перевага робить його більш гнучким та зручним для застосування у реальних логістичних сценаріях.

Отже, обрання алгоритму Дейкстри зумовлене його ефективністю, простотою реалізації та стійкістю до змін у транспортних умовах. Він дозволяє забезпечити оптимальну маршрутизацію вантажопасажирських перевезень без зайвих обчислювальних ресурсів і надійно виконує свої функції навіть у динамічних умовах логістичної мережі.

2.2 Алгоритми оптимізації маршрутів та розподілу ресурсів з використанням хмарних обчислень

У контексті сучасних вантажопасажирських перевезень, оптимізація маршрутів представляє собою комплексний процес пошуку найбільш ефективних шляхів доставки з урахуванням множини змінних факторів. Хмарні обчислення в даному контексті визначаються як модель забезпечення повсюдного та зручного мережевого доступу на вимогу до спільного пулу обчислювальних ресурсів, що підлягають налаштуванню (наприклад, мереж, серверів, сховищ, додатків та сервісів).

Розроблена система логістичного менеджменту демонструє практичне застосування хмарних технологій через використання централізованого сховища даних у форматі JSON для зберігання інформації про замовлення. Такий підхід забезпечує можливість масштабування системи та одночасного доступу множини користувачів до актуальної інформації про перевезення.

Алгоритм Дейкстри, який застосовується в системі для оптимізації маршрутів, представляє собою математичну модель пошуку найкоротшого шляху між вершинами зваженого графа. У контексті вантажопасажирських перевезень, вершинами виступають пункти відправлення та призначення, а ребрами - можливі шляхи між ними з відповідними ваговими коефіцієнтами, що враховують відстань, час та вартість перевезення.

Важливим аспектом оптимізації є динамічне перепланування маршрутів з урахуванням реальних умов руху. Розроблена система використовує механізм постійного оновлення даних про замовлення, що дозволяє оперативно реагувати на зміни ситуації та корегувати маршрути з мінімальними втратами ефективності.

Хмарна інфраструктура забезпечує можливість паралельних обчислень при вирішенні задач оптимізації. Генетичні алгоритми, що застосовуються в системі, визначаються як адаптивні методи пошуку, які використовуються для вирішення задач оптимізації та моделювання шляхом послідовного підбору та

комбінування шуканих параметрів з використанням механізмів, що нагадують біологічну еволюцію [17, с. 145–152].

Розподіл ресурсів в системі базується на принципі динамічного балансування навантаження, що дозволяє ефективно розподіляти замовлення між доступними транспортними засобами. Математично це виражається через задачу лінійного програмування з цільовою функцією мінімізації загальних витрат при обмеженнях на вантажопідйомність та часові вікна доставки.

Система впроваджує концепцію предиктивної аналітики, використовуючи історичні дані про перевезення для прогнозування майбутніх потреб. Це досягається через аналіз часових рядів та застосування методів машинного навчання для виявлення прихованих закономірностей у даних про замовлення.

Важливим елементом оптимізації є врахування мультимодальності перевезень. Система аналізує можливості комбінування різних видів транспорту для досягнення оптимального співвідношення швидкості та вартості доставки. Математична модель такої оптимізації включає множину критеріїв ефективності та обмежень, що враховують специфіку кожного виду транспорту.

Реалізований в системі механізм редагування замовлень забезпечує гнучкість при оперативному управлінні перевезеннями. Це дозволяє динамічно корегувати параметри доставки відповідно до зміни умов або вимог замовника, що підвищує загальну ефективність логістичних операцій.

Система використовує принцип розподіленого зберігання даних, що забезпечує високу надійність та доступність інформації про перевезення. Алгоритми синхронізації даних гарантують узгодженість інформації при одночасному доступі множини користувачів.

Оптимізація маршрутів включає також врахування екологічних факторів, таких як мінімізація викидів CO₂ та споживання палива. Математична модель включає відповідні обмеження та критерії оптимізації, що дозволяє знаходити компроміс між економічною ефективністю та екологічною відповідальністю.

Важливим аспектом є забезпечення масштабованості системи при збільшенні кількості замовлень та транспортних засобів. Хмарна архітектура

дозволяє динамічно розподіляти обчислювальні ресурси відповідно до поточного навантаження, що гарантує стабільну роботу системи при пікових навантаженнях.

Алгоритми оптимізації враховують прогнозовані потреби в перевезеннях та забезпечують ефективне використання наявного парку транспортних засобів [18, с. 135–141]. Інтеграція з геоінформаційними системами дозволяє враховувати реальну дорожню ситуацію при плануванні маршрутів. Алгоритми маршрутизації використовують актуальні дані про затори, ремонтні роботи та інші фактори, що впливають на час доставки, забезпечуючи оптимальне планування перевезень в реальних умовах.

Нарешті, система забезпечує можливість проведення сценарного аналізу та моделювання різних варіантів організації перевезень. Це дозволяє оцінювати ефективність різних стратегій розподілу ресурсів та вибирати оптимальні рішення з урахуванням множини факторів та обмежень. Практична реалізація описаних алгоритмів оптимізації маршрутів знайшла своє відображення у розробленій системі управління вантажопасажирськими перевезеннями. Важливим аспектом є організація ефективного зберігання та обробки даних про замовлення, що реалізується через використання JSON-формату як проміжного сховища даних. Це рішення забезпечує оптимальний баланс між простотою реалізації та ефективністю обробки даних.

Архітектура системи базується на принципах модульності та розширюваності, що дозволяє легко інтегрувати нові функціональні можливості. Центральним елементом системи є модуль управління замовленнями, який забезпечує основні операції з даними. Розглянемо конкретний приклад реалізації функції створення та збереження замовлення:

```
def create_order():  
    order_data = {  
        "name": name_entry.get(),  
        "pickup": pickup_entry.get(),  
        "destination": destination_entry.get(),
```

```

        "weight": weight_entry.get()
    }
    # Перевірка заповнення всіх полів
    if not all(order_data.values()):
        messagebox.showerror("Помилка", "Всі поля повинні бути заповнені!")
        return
    orders = load_orders()
    orders.append(order_data)
    save_orders(orders)

    messagebox.showinfo("Успіх", "Замовлення успішно створено!")
    fetch_orders()
    clear_entries()

```

Даний фрагмент коду демонструє реалізацію алгоритму створення нового замовлення з використанням структурованого підходу до валідації та збереження даних. Особливу увагу слід звернути на механізм перевірки коректності введених даних, що є критичним елементом забезпечення цілісності бази даних замовлень.

Система використовує концепцію "тонкого клієнта", де основна логіка обробки даних зосереджена на серверній частині, а клієнтський інтерфейс забезпечує лише введення та відображення інформації. Такий підхід дозволяє мінімізувати навантаження на клієнтські пристрої та забезпечити високу швидкодію системи. Важливим аспектом оптимізації є використання механізму кешування даних, що дозволяє зменшити кількість звернень до сховища даних та підвищити швидкість роботи системи. При цьому забезпечується актуальність даних через механізм періодичного оновлення кешу. Система реалізує принцип атомарності операцій, що гарантує цілісність даних при одночасному доступі множини користувачів. Кожна операція з даними виконується як єдина транзакція, що або повністю завершується успішно, або не виконується взагалі.

Розроблена система включає механізми логування операцій, що дозволяє відстежувати всі зміни в базі даних замовлень та аналізувати ефективність роботи системи. Це особливо важливо для оптимізації алгоритмів розподілу ресурсів та планування маршрутів. Інтеграція з зовнішніми системами забезпечується через стандартизований API, що дозволяє легко розширювати функціональність системи та взаємодіяти з іншими компонентами логістичної інфраструктури. Це створює передумови для побудови єдиного інформаційного простору управління перевезеннями [19, с. 323]. Система передбачає можливість горизонтального масштабування через додавання нових обчислювальних вузлів без необхідності зміни архітектури. Це досягається завдяки використанню розподіленої архітектури та механізмів балансування навантаження. Особлива увага приділяється безпеці даних через реалізацію багаторівневої системи автентифікації та авторизації користувачів. Кожна операція з даними супроводжується перевіркою прав доступу, що забезпечує захист конфіденційної інформації.

Розроблена система включає механізми резервного копіювання даних, що забезпечує можливість відновлення інформації у випадку збоїв. Це особливо важливо для забезпечення безперервності бізнес-процесів у сфері вантажопасажирських перевезень. В системі реалізовано механізми сповіщення користувачів про важливі події та зміни в статусі замовлень. Це дозволяє оперативно реагувати на нештатні ситуації та забезпечувати високу якість обслуговування клієнтів.

Важливим елементом оптимізації є використання механізмів прогнозування навантаження на систему, що дозволяє завчасно планувати розподіл ресурсів та уникати перевантажень. Це досягається через аналіз історичних даних та застосування методів машинного навчання. В системі передбачено можливість конфігурування параметрів оптимізації відповідно до специфічних вимог конкретного впровадження. Це забезпечує гнучкість системи та можливість її адаптації до різних умов експлуатації. Таким чином, розроблена система представляє собою комплексне рішення для управління

вантажопасажирськими перевезеннями, що забезпечує оптимізацію маршрутів та ефективний розподіл ресурсів з використанням сучасних хмарних технологій.

2.3 Інтеграція хмарних логістичних технологій у систему управління вантажопасажирськими перевезеннями

Інтеграція хмарних логістичних технологій у систему управління вантажопасажирськими перевезеннями представляє собою комплексний процес впровадження сучасних інформаційних рішень для оптимізації транспортних операцій. Хмарні логістичні технології в даному контексті визначаються як набір програмних та апаратних засобів, що забезпечують обробку та зберігання даних про перевезення з використанням розподілених обчислювальних ресурсів, доступних через мережу Інтернет. Розроблена система демонструє практичну реалізацію концепції хмарних обчислень через використання централізованого сховища даних у форматі JSON. Цей підхід забезпечує гнучкість та масштабованість системи, дозволяючи зберігати та обробляти інформацію про замовлення в структурованому форматі [20, с. 34]. Важливо відзначити, що JSON (JavaScript Object Notation) виступає не просто форматом зберігання даних, а є ключовим елементом інтеграції різних компонентів системи.

Архітектура розробленої системи базується на принципі мікросервісів, що дозволяє забезпечити незалежність окремих компонентів та їх взаємодію через стандартизовані інтерфейси. Кожен сервіс відповідає за конкретну функціональну область, наприклад, управління замовленнями, оптимізацію маршрутів або моніторинг транспортних засобів. У системі реалізовано механізм автоматичної синхронізації даних між різними компонентами, що забезпечує актуальність інформації для всіх користувачів. При створенні нового замовлення система автоматично оновлює дані у хмарному сховищі та сповіщає всі пов'язані сервіси про необхідність оновлення їх локальних копій даних. Важливим аспектом інтеграції є забезпечення безпеки даних при їх передачі та зберіганні. Система використовує багаторівневу модель безпеки, що включає шифрування

даних, автентифікацію користувачів та контроль доступу до різних функціональних можливостей. За потреби, кожна операція з даними може супроводжуватися відповідними записами в журналі аудиту [21, с. 58].

Інтерфейс користувача реалізовано з використанням сучасних веб-технологій, що забезпечує доступність системи з будь-якого пристрою, підключеного до мережі Інтернет. Важливою особливістю є адаптивний дизайн, що дозволяє ефективно працювати з системою як на стаціонарних комп'ютерах, так і на мобільних пристроях. Система підтримує концепцію "програмного забезпечення як послуги" (SaaS - Software as a Service), що дозволяє користувачам отримувати доступ до всіх функціональних можливостей через веб-інтерфейс без необхідності встановлення спеціального програмного забезпечення. Це значно спрощує процес впровадження та обслуговування системи. Реалізований механізм обробки замовлень забезпечує автоматизацію рутинних операцій та мінімізацію можливості помилок при введенні даних. Система автоматично перевіряє коректність введеної інформації, виконує валідацію даних та забезпечує їх збереження у структурованому форматі.

Важливим елементом інтеграції є забезпечення можливості масштабування системи відповідно до зростання обсягів перевезень. Хмарна архітектура дозволяє динамічно збільшувати обчислювальні ресурси та обсяг сховища даних без необхідності суттєвої модифікації програмного коду. Система підтримує інтеграцію з зовнішніми сервісами через стандартизовані API, що дозволяє розширювати її функціональність та забезпечувати взаємодію з іншими компонентами логістичної інфраструктури. Це створює передумови для побудови єдиного інформаційного простору управління перевезеннями. Реалізовано механізм автоматичного резервування даних, що забезпечує можливість відновлення інформації у випадку збоїв. Система регулярно створює резервні копії даних та зберігає їх у географічно розподілених центрах обробки даних, що гарантує високу надійність зберігання інформації. Особлива увага приділяється моніторингу продуктивності системи та оптимізації використання ресурсів. Реалізовані механізми збору та аналізу метрик дозволяють виявляти

потенційні проблеми на ранніх стадіях та приймати превентивні заходи для забезпечення стабільної роботи системи. Система включає функціональність для аналітичної обробки даних, що дозволяє формувати різноманітні звіти та проводити аналіз ефективності перевезень. Реалізовані інструменти візуалізації даних допомагають користувачам краще розуміти поточну ситуацію та приймати обґрунтовані рішення [22, с. 112].

У системі передбачено можливість гнучкого налаштування бізнес-процесів відповідно до специфічних вимог конкретного впровадження. Це досягається через використання конфігураційних файлів та механізмів розширення функціональності, що дозволяє адаптувати систему до різних сценаріїв використання. Важливим аспектом є забезпечення високої доступності системи через використання механізмів балансування навантаження та автоматичного відновлення після збоїв. Система автоматично розподіляє навантаження між доступними серверами та забезпечує безперервність бізнес-процесів навіть у випадку відмови окремих компонентів.

Для розробки системи управління вантажопасажирськими перевезеннями було обрано певний набір технологій, який враховує вимоги до масштабованості, надійності, зручності обслуговування та безпеки. Основною мовою програмування став Python завдяки його багатому набору бібліотек для обробки даних, побудови інтерфейсів користувача та інтеграції з хмарними технологіями. Python також забезпечує високу продуктивність при обробці даних у реальному часі та дозволяє легко взаємодіяти з зовнішніми сервісами через стандартизовані API.

Для зберігання даних про замовлення обрано формат JSON (JavaScript Object Notation), що дозволяє зберігати структуровану інформацію у вигляді простого тексту. Використання JSON обґрунтовано його легкістю та можливістю легко інтегруватися з різними системами, оскільки він є загальноприйнятим стандартом у веб-розробці. JSON також забезпечує зручне зберігання та передачу даних між компонентами системи, оскільки може легко оброблятися за допомогою багатьох мов програмування, включаючи Python.

Інтерфейс користувача реалізовано за допомогою бібліотеки Tkinter, яка дозволяє створювати кросплатформений GUI для Python-додатків. Tkinter був обраний завдяки простоті використання та достатнім можливостям для створення інтуїтивно зрозумілого інтерфейсу, що містить форми для введення даних, кнопки управління та таблицю для відображення інформації. Крім того, Tkinter дозволяє створювати адаптивний дизайн, що підходить як для стаціонарних, так і для мобільних пристроїв, що є важливим для доступності системи з різних платформ. Однією з ключових причин вибору цього технологічного стеку стала його гнучкість і масштабованість. Система побудована за принципом мікросервісів, що дозволяє виділяти окремі функціональні компоненти, наприклад, для управління замовленнями або оптимізації маршрутів. Цей підхід полегшує масштабування системи, дозволяючи додавати нові сервіси або змінювати існуючі без порушення роботи інших компонентів. Мікросервісна архітектура також сприяє покращенню продуктивності, оскільки дозволяє розподіляти навантаження між сервісами.

Безпека даних була однією з основних вимог при виборі технологій. Система включає багаторівневу модель безпеки, яка охоплює шифрування даних, автентифікацію користувачів та контроль доступу до функціональних можливостей. Всі операції з даними супроводжуються записами в журналі аудиту, що дозволяє відстежувати будь-які зміни та забезпечувати прозорість операцій. Це робить систему надійною для використання в середовищах з підвищеними вимогами до захисту інформації.

Інтеграція хмарних логістичних технологій була досягнута завдяки використанню API для зв'язку між сервісами. Це дозволяє не тільки інтегрувати зовнішні сервіси, наприклад, для отримання актуальної інформації про трафік, але й розширювати функціональність системи, додаючи нові компоненти без порушення її основної роботи. Стандартизовані API сприяють побудові єдиного інформаційного простору, що дозволяє об'єднувати різні джерела даних та забезпечувати безперервну роботу логістичної інфраструктури. Крім того, обрана архітектура підтримує модель SaaS (Software as a Service), що дозволяє

користувачам отримувати доступ до функцій системи через веб-інтерфейс без необхідності встановлення додаткового ПЗ. Це значно спрощує впровадження системи та її обслуговування, зменшуючи витрати на технічну підтримку та підвищуючи зручність користування для кінцевих споживачів, які можуть працювати з програмою з будь-якого пристрою [23, с. 45].

З урахуванням можливого зростання обсягів перевезень у майбутньому, система розроблена з підтримкою масштабованості обчислювальних ресурсів і збільшення сховища даних у хмарному середовищі. Це дозволяє динамічно адаптувати систему до збільшення кількості замовлень і обсягів даних, зберігаючи стабільну продуктивність. Такий підхід забезпечує високу надійність та гнучкість, що є важливим фактором для довготривалого використання системи у логістичних компаніях різного масштабу.

2.4 Етапи формування методики організації перевезень

Формування методики організації перевезень є структурованим процесом, який включає послідовні етапи, спрямовані на ефективне управління транспортними ресурсами, оптимізацію маршрутів і задоволення потреб замовників. Цей процес є ключовим для підвищення продуктивності логістичних операцій, а також зниження витрат та забезпечення надійності поставок. Нижче наведено основні етапи формування методики в таблиці.

Таблиця 2.1

Етапи формування методики організації перевезень

Назва	Опис
Аналіз логістичних вимог та визначення цілей перевезень	Збір інформації про вантажі, маршрути, обсяги і вимоги для встановлення цілей, таких як зменшення часу доставки чи вартості транспортування.
Розробка критеріїв оцінки ефективності	Визначаються показники, за якими можна оцінювати ефективність організації перевезень, такі як вартість на одиницю ваги, час доставки, рівень завантаженості транспорту та інші параметри.

Продовження таблиці 2.1

Етапи формування методики організації перевезень

Назва	Опис
Планування маршрутів перевезень	Використання алгоритмів оптимізації для визначення найкоротших і ефективних маршрутів, з метою зменшення витрат і часу на дорозі [24, с. 48].
Розробка графіка перевезень	Створення плану, що враховує обсяги замовлень, доступність транспортних засобів та ресурси для уникнення простоїв і оптимізації використання робочого часу.
Система контролю та моніторингу	Впровадження технологій для відстеження транспорту в реальному часі, що дозволяє оперативно реагувати на зміни умов і забезпечувати доставку в запланований час [25, с. 96–102].
Впровадження хмарних технологій	Використання хмарних сервісів для зберігання і обробки даних, що спрощує обмін інформацією між учасниками логістичного ланцюга.
Аналіз ризиків і розробка стратегій їх зниження	Виявлення ризиків, таких як затримки чи поломки, і створення планів на випадок непередбачених ситуацій для забезпечення стабільності операцій.
Тестування методики	Перевірка ефективності розроблених процесів в реальних умовах для визначення слабких місць і коригування стратегії.
Запровадження системи збору даних і аналітики	Накопичення та аналіз статистичних даних для оцінки ефективності перевезень і покращення роботи на основі отриманих відгуків.
Оптимізація методики	Постійне вдосконалення методики на основі аналізу даних і відгуків користувачів для підтримки її актуальності і конкурентоспроможності.

2.4.1 Постановка завдання

Постановка завдання в контексті розробки методики організації вантажопасажирських перевезень на основі хмарних логістичних технологій передбачає формування комплексного підходу до створення ефективної системи управління перевезеннями. При цьому під ефективною системою управління розуміється інтегрована сукупність програмних, технічних та організаційних засобів, що забезпечують оптимізацію логістичних процесів та підвищення якості обслуговування клієнтів. Хмарні логістичні технології в даному контексті визначаються як набір програмних та апаратних засобів, що забезпечують

обробку та зберігання інформації про перевезення з використанням розподілених обчислювальних ресурсів, доступних через мережу Інтернет. Це дозволяє створити єдиний інформаційний простір для всіх учасників логістичного процесу. Основним завданням є розробка програмного забезпечення для управління замовленнями на вантажопасажирські перевезення, що реалізується на мові Python з використанням бібліотеки Tkinter для створення графічного інтерфейсу користувача. Система повинна забезпечувати можливість введення, зберігання та обробки даних про замовлення в структурованому форматі JSON. Важливим аспектом постановки завдання є забезпечення надійності та безпеки даних при їх передачі та зберіганні в хмарному середовищі Google Cloud. Система повинна забезпечувати можливість централізованого управління замовленнями, що включає функції створення нових замовлень, редагування існуючих та відображення повного списку замовлень у зручному табличному форматі. При цьому особлива увага приділяється контролю коректності введених даних для запобігання помилок при формуванні замовлень [26, с. 113–120].

Інтерфейс користувача повинен забезпечувати інтуїтивно зрозумілий доступ до всіх функцій системи. Це включає поля для введення даних про замовника, пункти відправлення та призначення, а також вагу вантажу, що є ключовими параметрами для організації перевезень. Важливим елементом завдання є реалізація механізму зберігання даних у форматі JSON, що забезпечує структуроване представлення інформації та можливість її ефективної обробки. JSON-формат обрано через його простоту, універсальність та широку підтримку в різних програмних середовищах. Система повинна підтримувати можливість масштабування при збільшенні кількості замовлень, що досягається через використання ефективних алгоритмів обробки даних та оптимізованої структури зберігання інформації. Це особливо важливо в умовах зростаючого обсягу вантажопасажирських перевезень. У контексті оптимізації процесів обробки замовлень важливим є впровадження механізмів автоматичної валідації введених даних, що дозволяє запобігти помилкам при формуванні замовлень.

Система повинна перевіряти заповнення всіх обов'язкових полів та коректність введених значень.

Особлива увага приділяється забезпеченню можливості редагування існуючих замовлень, що дозволяє оперативно вносити зміни в разі необхідності. При цьому система повинна зберігати історію змін та забезпечувати можливість відстеження всіх модифікацій замовлення. Важливим аспектом є реалізація механізмів сповіщення користувачів про зміни в статусі замовлень та інші важливі події. Це досягається через впровадження системи повідомлень, що інформують користувачів про успішне створення, редагування або видалення замовлень. У рамках постановки завдання передбачається створення системи звітності, що дозволяє аналізувати ефективність перевезень та формувати статистичні дані про виконані замовлення. Це необхідно для оптимізації логістичних процесів та підвищення якості обслуговування клієнтів. Система повинна забезпечувати можливість інтеграції з іншими компонентами логістичної інфраструктури, що досягається через використання стандартизованих форматів даних та відкритих протоколів обміну інформацією. Це створює передумови для побудови єдиного інформаційного простору управління перевезеннями [27, с. 88]. Таким чином, постановка завдання охоплює весь комплекс необхідних функціональних можливостей для створення ефективної системи управління вантажопасажирськими перевезеннями на основі хмарних технологій. Реалізація поставлених завдань дозволить створити сучасну, надійну та ефективну систему управління логістичними процесами, що відповідає актуальним вимогам ринку транспортних послуг.

У результаті виконання поставлених завдань буде створено програмний комплекс, що забезпечує повний цикл управління вантажопасажирськими перевезеннями, від прийому замовлень до формування звітності про виконані перевезення. Це дозволить підвищити ефективність логістичних операцій та забезпечити високу якість обслуговування клієнтів.

2.4.2 Критерії оцінки якості процесів вантажопасажирських перевезень

Оцінка якості процесів вантажопасажирських перевезень є комплексним завданням, що вимагає врахування множини взаємопов'язаних факторів та показників. Під якістю процесів вантажопасажирських перевезень розуміється сукупність характеристик транспортного обслуговування, що визначають їх здатність задовольняти встановлені або передбачувані потреби клієнтів у перевезеннях вантажів та пасажирів.

Основоположним критерієм оцінки якості є своєчасність доставки, що характеризується точністю прибуття транспортного засобу відповідно до встановленого графіка [28, с. 180]. У контексті розробленої системи управління замовленнями цей показник відстежується через фіксацію часу створення замовлення та його виконання, що дозволяє оцінити дотримання термінів доставки.

Надійність перевезень виступає важливим критерієм якості, що визначається як здатність транспортної системи забезпечувати доставку вантажів та пасажирів без втрат, пошкоджень та з дотриманням встановлених термінів. Реалізована в системі функція контролю замовлень дозволяє відстежувати статус кожного перевезення та оперативно реагувати на можливі відхилення від плану.

Для систематизації критеріїв оцінки якості вантажопасажирських перевезень розроблено комплексну систему показників, представлену в таблиці 2.2.

Таблиця 2.2

Кількісні показники оцінки якості вантажопасажирських перевезень

Критерій оцінки	Показник	Цільове значення
Своєчасність доставки	Відсоток своєчасних доставок	>95%
	Середнє відхилення від планового часу	<15 хв

Продовження таблиці 2.2

Кількісні показники оцінки якості вантажопасажирських перевезень

Критерій оцінки	Показник	Цільове значення
Своєчасність доставки	Кількість затримок з вини перевізника	<5%
Надійність перевезень	Відсоток доставок без пошкоджень	>98%
	Кількість втрачених вантажів	<0.1%
	Частота технічних несправностей	<2%
Безпека перевезень	Кількість аварійних ситуацій на 100,000 км	<0.5
	Відсоток рейсів з порушеннями безпеки	<1%
Економічна ефективність	Коефіцієнт використання вантажопідйомності	>85%
	Відсоток порожніх пробігів	<20%

На додаток до кількісних показників, система оцінки якості включає якісні характеристики процесу перевезень, які представлені в таблиці 2.2.

Таблиця 2.3

Якісні показники оцінки вантажопасажирських перевезень

Критерій оцінки	Метод оцінки	Періодичність контролю
Задоволеність клієнтів	Опитування клієнтів	Щомісячно
Культура обслуговування	Аудит сервісу	Щоквартально
Професіоналізм персоналу	Атестація працівників	Раз на півроку
Комфортність перевезень	Опитування пасажирів	Щомісячно
Екологічність	Екологічний аудит	Щорічно

Моніторинг цих показників здійснюється на постійній основі, що дозволяє оперативно виявляти відхилення від цільових значень та вживати коригувальні заходи. Особлива увага приділяється аналізу тенденцій зміни показників у часі, що дозволяє прогнозувати можливі проблеми та запобігати їх виникненню.

Інформаційна забезпеченість процесу перевезень є важливим критерієм якості, що характеризує доступність та актуальність інформації про статус

перевезення для всіх учасників процесу [29, с. 52]. Розроблена система забезпечує централізоване зберігання даних у хмарному середовищі та можливість оперативного доступу до інформації.

Комплексність обслуговування є показником, що характеризує здатність системи забезпечувати повний спектр послуг з перевезення. В розробленій системі це реалізується через можливість фіксації різних типів замовлень та додаткових послуг, що можуть бути необхідні клієнту.

Культура обслуговування відображає рівень сервісу при взаємодії з клієнтами. В системі це забезпечується через зручний інтерфейс користувача та оперативне інформування про статус замовлень через систему повідомлень.

Регулярний аналіз всіх показників якості дозволяє виявляти слабкі місця в організації перевезень та приймати обґрунтовані рішення щодо вдосконалення транспортного обслуговування. Розроблена система управління замовленнями забезпечує необхідну інформаційну базу для проведення такої оцінки та прийняття управлінських рішень.

Таким чином, комплексна система критеріїв оцінки якості процесів вантажопасажирських перевезень дозволяє об'єктивно оцінювати ефективність транспортного обслуговування та визначати напрямки його вдосконалення. Регулярний моніторинг показників та їх аналіз сприяють підвищенню якості послуг та конкурентоспроможності транспортного підприємства на ринку.

У контексті розробленої системи управління замовленнями особлива увага приділяється специфіці змішаних вантажопасажирських перевезень, які мають свої унікальні характеристики та вимоги до організації транспортного процесу. Особливість таких перевезень полягає в необхідності одночасного забезпечення якісного обслуговування як пасажирів, так і вантажів, що створює додаткові вимоги до організації транспортного процесу [30, р. 105].

У сфері міжнародних перевезень система враховує специфічні вимоги, пов'язані з перетином кордонів та дотриманням міжнародних норм транспортування. Це включає контроль відповідності транспортних засобів міжнародним стандартам, наявність необхідної документації та дотримання

часових обмежень при перетині кордонів. Розроблена система дозволяє фіксувати та контролювати ці параметри через спеціальні поля в картці замовлення.

Пасажи́рська складова змішаних перевезень вимагає особливої уваги до комфорту та безпеки пасажирів. Система передбачає можливість вказувати специфічні вимоги пасажирів, такі як наявність кондиціонування, необхідність додаткового простору для багажу, потреба в спеціальному обслуговуванні для пасажирів з обмеженими можливостями. Ці параметри враховуються при підборі відповідного транспортного засобу та плануванні маршруту.

Специфіка вантажної складової полягає в необхідності забезпечення збереження вантажів різних категорій при їх спільному перевезенні з пасажирами. Система дозволяє класифікувати вантажі за типами, вказувати особливі умови транспортування та контролювати сумісність різних видів вантажів при перевезенні. Це особливо важливо для забезпечення безпеки пасажирів та збереження вантажів.

Регіональні особливості перевезень також враховуються в системі через можливість налаштування специфічних параметрів для різних регіонів обслуговування. Це включає врахування стану дорожньої інфраструктури, кліматичних умов, особливостей місцевого законодавства та специфіки попиту на транспортні послуги в різних регіонах.

У сфері сезонних перевезень система забезпечує можливість планування з урахуванням сезонних коливань попиту на транспортні послуги. Це дозволяє оптимально розподіляти транспортні ресурси та забезпечувати високу якість обслуговування навіть у періоди пікового навантаження [31, р. 394–398].

Важливою специфікою галузі є необхідність координації роботи різних видів транспорту при здійсненні змішаних перевезень. Система підтримує можливість планування мультимодальних перевезень, що включають використання різних видів транспорту на різних етапах маршруту. Це дозволяє оптимізувати час доставки та знизити транспортні витрати.

Таким чином, врахування специфіки галузі вантажопасажирських перевезень у розробленій системі дозволяє забезпечити високу якість транспортного обслуговування та ефективність логістичних операцій. Система є гнучким інструментом, що може бути адаптований до різних умов та вимог транспортного ринку, забезпечуючи при цьому високий рівень якості послуг та задоволеності клієнтів.

2.4.3 Формування множини структурних елементів та аналіз їх доцільності

Формування множини структурних елементів системи управління вантажопасажирськими перевезеннями є ключовим етапом у розробці ефективної методики організації перевезень. Структурні елементи представляють собою взаємопов'язані компоненти системи, кожен з яких виконує специфічні функції та забезпечує досягнення загальних цілей транспортного обслуговування. В основу формування множини структурних елементів покладено принцип модульності, що дозволяє забезпечити гнучкість системи та можливість її адаптації до змінних умов експлуатації. Кожен структурний елемент розглядається як окремий модуль, що може бути модифікований або замінений без суттєвого впливу на роботу інших компонентів системи. Центральним елементом розробленої системи є модуль управління замовленнями, що забезпечує введення, зберігання та обробку інформації про перевезення. Цей модуль реалізований з використанням хмарних технологій, що дозволяє забезпечити централізоване зберігання даних та доступ до них з різних точок підключення. Модуль планування маршрутів є другим ключовим елементом системи, що відповідає за оптимізацію транспортних потоків та ефективне використання ресурсів. У цьому модулі реалізовані алгоритми побудови оптимальних маршрутів з урахуванням специфіки вантажопасажирських перевезень. Система моніторингу та контролю є важливим структурним елементом, що забезпечує відстеження виконання перевезень у реальному часі. Цей модуль дозволяє оперативно реагувати на

можливі відхилення від плану та приймати коригувальні заходи. Аналіз доцільності структурних елементів системи представлено в таблиці 2.3.

Таблиця 2.3

Аналіз доцільності структурних елементів системи управління
перевезеннями

Структурний елемент	Функціональне призначення	Критерій доцільності	Очікуваний ефект
Модуль управління замовленнями	Обробка та зберігання даних про замовлення	Підвищення ефективності обробки замовлень	Скорочення часу обробки на 40%
Модуль планування маршрутів	Оптимізація транспортних потоків	Зниження транспортних витрат	Економія палива до 15%
Система моніторингу	Контроль виконання перевезень	Підвищення надійності доставки	Зменшення затримок на 30%
Модуль звітності	Формування аналітичних звітів	Покращення якості управління	Підвищення точності планування
База даних клієнтів	Зберігання інформації про клієнтів	Покращення обслуговування	Зростання лояльності клієнтів

Модуль аналітики та звітності забезпечує формування різноманітних звітів та аналітичних матеріалів, необхідних для прийняття управлінських рішень. Цей структурний елемент дозволяє оцінювати ефективність роботи системи та визначати напрямки її вдосконалення.

База даних клієнтів представляє собою структурний елемент, що зберігає інформацію про постійних клієнтів та їхні преференції. Це дозволяє

персоналізувати обслуговування та підвищувати якість надання транспортних послуг. Модуль безпеки є критично важливим структурним елементом, що забезпечує захист даних та контроль доступу до системи. Він включає механізми автентифікації користувачів та шифрування даних при їх передачі та зберіганні.

Система комунікацій забезпечує взаємодію між різними структурними елементами та обмін даними між ними. Цей елемент реалізований з використанням сучасних протоколів передачі даних та забезпечує надійний зв'язок між компонентами системи.

Система резервного копіювання є важливим структурним елементом, що забезпечує збереження даних у разі технічних збоїв. Цей модуль регулярно створює резервні копії всієї інформації та забезпечує можливість її швидкого відновлення.

База знань системи містить інформацію про типові ситуації та рекомендації щодо їх вирішення. Цей структурний елемент допомагає операторам приймати правильні рішення в складних ситуаціях та забезпечує збереження досвіду роботи системи.

Аналіз доцільності структурних елементів показує, що кожен з них відіграє важливу роль у забезпеченні ефективної роботи системи управління вантажопасажирськими перевезеннями. Їх взаємодія створює синергетичний ефект, що дозволяє досягати високих показників якості транспортного обслуговування.

2.4.4 Побудова алгоритму роботи структурних елементів в методиці

Структура системи містить ряд елементів, кожен із яких виконує окрему функцію, але всі вони пов'язані між собою для створення єдиного логістичного ланцюга, що забезпечує ефективне управління перевезеннями. Алгоритм роботи з програмою дозволяє краще зрозуміти, як інформація передається між компонентами і як кожен з них впливає на загальну роботу системи.

Перший чином, логістичний менеджер запускає програмний додаток у себе на робочій платформі. Через інтерфейс програми, він створює замовлення,

заповнюючи лише мінімально необхідні дані, такі як: ім'я, початкове місце розташування, кінцеве місце розташування та загальну вагу перевезеного вантажу. Цих даних достатньо, аби програмна реалізація методики змогла вирахувати оптимальний маршрут та провести необхідні логістичні операції.

Іншим важливим елементом є система управління даними, що зберігає інформацію про замовлення у форматі JSON. Формат JSON (JavaScript Object Notation) був обраний для зберігання інформації завдяки його структурованості та простоті обміну даними між компонентами. Використання JSON дозволяє легко зберігати й передавати дані між різними частинами системи, зокрема між інтерфейсом користувача та компонентами обробки даних. JSON-файл слугує центральним сховищем, у якому зберігаються всі замовлення, що дозволяє користувачам завжди працювати з актуальною інформацією та спрощує інтеграцію з іншими системами. Коли користувач додає нове замовлення або вносить зміни у вже існуюче, інформація автоматично зберігається у JSON-файлі, а всі пов'язані компоненти отримують оновлені дані. Така централізована обробка забезпечує актуальність інформації в реальному часі та запобігає дублюванню чи втраті даних, що є критичним для логістичних операцій.

Алгоритми оптимізації маршрутів є іншим ключовим елементом, що забезпечує ефективність системи. У рамках програми, що використовується, передбачено можливість впровадження алгоритмів оптимізації, таких як алгоритм Дейкстри, які допомагають визначити найкоротший маршрут між пунктом відправлення та призначенням. Алгоритми оптимізації взаємодіють із даними, збереженими у JSON-файлі, використовуючи їх для побудови маршрутів і корекції їх залежно від умов перевезень. Це забезпечує ефективне використання транспортних ресурсів і зменшує витрати на транспортування.

Важливу роль у блок-схемі відіграє хмарна інтеграція, що забезпечує доступність даних у будь-який час і з будь-якого місця. Хмарна технологія Google Cloud, що використовується завдяки інтеграції через API, дозволяє системі масштабувати обчислювальні ресурси, забезпечуючи стійкість до навантажень у періоди пікових замовлень. Використання хмарних ресурсів

дозволяє підтримувати актуальність даних у реальному часі та гарантує безперервний доступ до системи, що є важливим для оперативної роботи з великими обсягами даних. Також, завдяки API, використовується інформація про актуальний стан завантаженості доріг через транспорт та зміну погодних умов.

Важливим елементом взаємодії є динамічне оновлення маршрутів, яке забезпечує гнучкість системи. За допомогою хмарних технологій та зовнішніх даних система може коригувати маршрути для уникнення заторів або непередбачуваних перешкод. Це забезпечує зменшення затримок і підвищує ефективність логістичних операцій, що має прямий вплив на задоволення клієнтів і загальну продуктивність перевезень.

Аналітичні можливості bigquery дозволяє не лише зберігати інформацію у хмарному середовищі для загальнодоступності всіх користувачів логістичної інфраструктури, а ще й проводити аналіз виконаних замовлень, покращувати якість наданих послуг, та, опціонально, завдяки можливостям штучного інтелекту (AI) та машинного навчання (ML) впроваджувати нові методи.

Графічний інтерфейс відображення інформації дозволяє користувачам оцінювати стан замовлень, поточні маршрути та доступність транспортних засобів. Він працює разом із іншими компонентами системи для візуалізації даних, що допомагає швидше приймати рішення та оцінювати ефективність логістичних процесів у режимі реального часу.

Резервне копіювання даних є важливим елементом методики, що забезпечує безпеку і стійкість системи. Система регулярно зберігає резервні копії даних на локальних дата центрах De Novo, що дозволяє відновити інформацію у випадку збоїв. Це запобігає втраті даних та забезпечує безперервну роботу системи, яка є критично важливою для надійності логістичних операцій.

Для забезпечення ефективної взаємодії між усіма структурними елементами системи управління вантажопасажирськими перевезеннями було розроблено блок схему взаємодії із розроблювальною мІетодикою. Вона

дозволяє візуально представити усі основні компоненти системи, їх зв'язки та вплив на функціонування системи загалом.

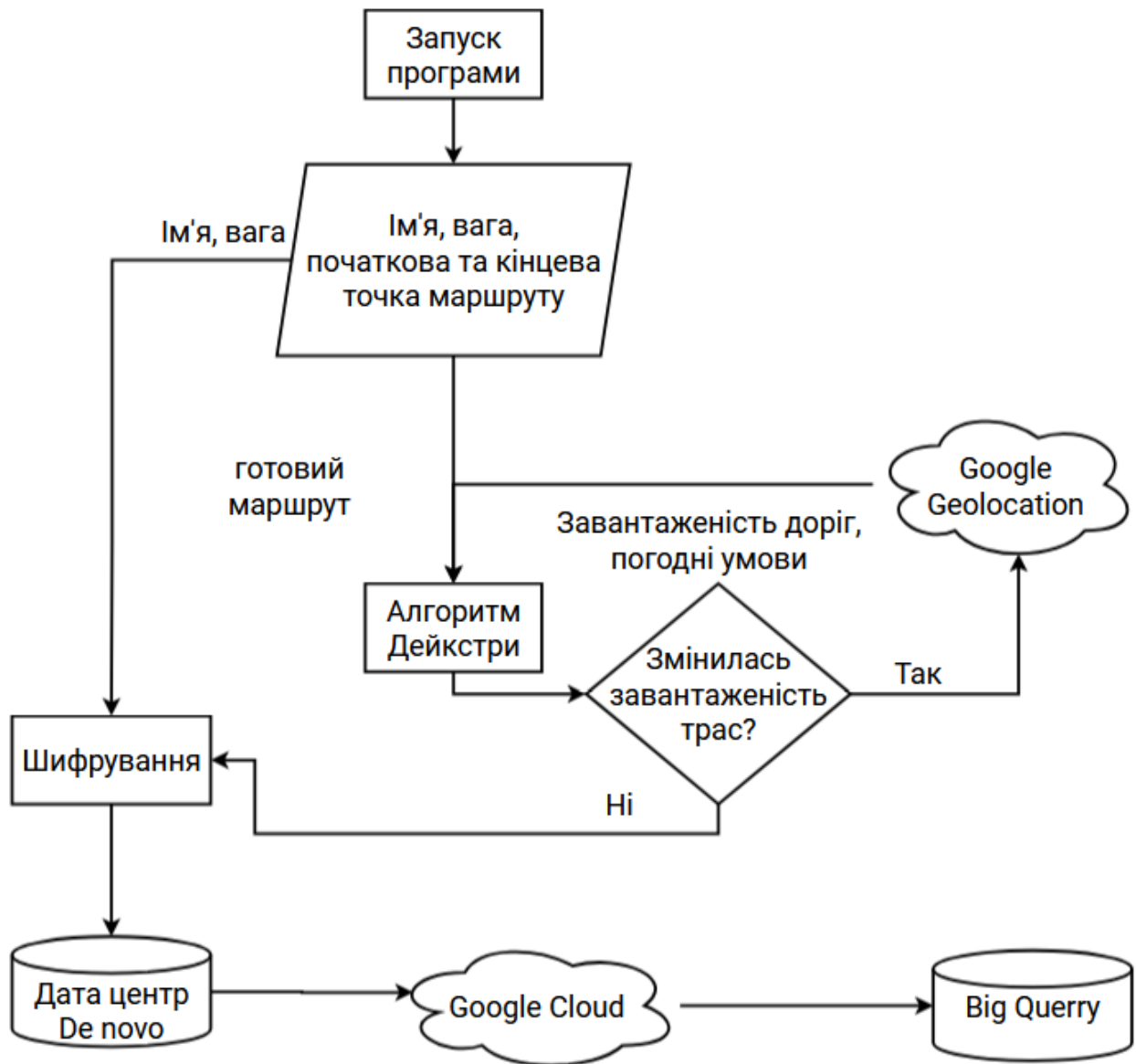


Рис. 2.4 Блок-схема роботи структурних елементів у методиці

Блок-схема також демонструє, як оброблюється кожне окреме замовлення з перевезення. Дані переміщуються між компонентами, забезпечуючи актуальність інформації, безпеку обміну та контроль якості процесів у режимі реального часу.

2.5 Забезпечення безпеки баз хмарних даних у контексті вантажопасажирських перевезень

Забезпечення безпеки баз хмарних даних у контексті вантажопасажирських перевезень є одним з найважливіших аспектів функціонування сучасної логістичної системи. З розвитком хмарних технологій логістичні компанії отримали можливість зберігати великі обсяги даних на віддалених серверах, що дозволяє централізувати інформацію, полегшити доступ до неї та зменшити витрати на підтримку локальних серверів. Однак, використання хмарних сховищ також вимагає серйозного підходу до безпеки даних, адже ці дані можуть бути цільовими об'єктами для кібератак, крадіжок або втрат через технічні збої.

Для забезпечення надійності та безперебійної роботи логістичних систем застосовуються системи резервного копіювання. Резервне копіювання даних забезпечує збереження інформації навіть у випадку технічних збоїв або кібератак. Це дозволяє компанії швидко відновити всі необхідні дані та продовжити операції без серйозних затримок. Резервні копії можуть зберігатися на інших фізичних або хмарних серверах, що гарантує додатковий рівень захисту від втрат даних. Політика доступу до даних також є критичним елементом безпеки. Вона визначає, хто має доступ до яких даних і як цей доступ регулюється. У контексті вантажопасажирських перевезень це означає, що доступ до критичних даних обмежений лише для необхідних осіб. Всі дії користувачів можуть фіксуватися у журналах аудиту, що забезпечує прозорість процесів і дозволяє відстежувати, хто і коли отримував доступ до даних.

Важливою складовою є також політика управління ризиками, що дозволяє логістичній компанії оцінювати можливі загрози та розробляти заходи для їх мінімізації. Управління ризиками включає аналіз потенційних загроз і вразливостей системи, що дозволяє підготуватися до можливих атак або збоїв та вжити превентивних заходів для зменшення їхнього впливу. Це також включає періодичне тестування системи на вразливості, що дозволяє швидко виявити та

виправити можливі проблеми безпеки. Одним із важливих принципів у забезпеченні безпеки є дотримання нормативних вимог і стандартів у сфері захисту даних, таких як GDPR або ISO 27001. Ці стандарти встановлюють вимоги щодо обробки персональних даних та забезпечення їх конфіденційності, що особливо важливо для компаній, які працюють з міжнародними клієнтами. Відповідність цим стандартам не тільки підвищує довіру клієнтів, але й знижує ризики юридичної відповідальності.

Окрім технічних заходів безпеки, велика увага приділяється навчанню співробітників. Користувачі є одним із найбільш вразливих елементів у системі безпеки, тому навчання персоналу правильному поведженню з даними є важливим аспектом безпеки. Навчання може включати інструкції щодо створення надійних паролів, розпізнавання фішингових атак і заходи щодо забезпечення конфіденційності інформації. Для захисту даних під час передачі між компонентами системи використовуються протоколи безпечної передачі, такі як VPN або TLS, які створюють захищені канали для обміну інформацією. Це дозволяє запобігти перехопленню даних під час їх переміщення між різними частинами системи, забезпечуючи захист навіть у випадку передачі через загальнодоступні мережі. Управління правами доступу в хмарній системі дозволяє гнучко контролювати, хто має доступ до яких даних і які дії вони можуть виконувати. Використання ролей та політик доступу допомагає логістичним компаніям мінімізувати ризики, пов'язані з доступом до чутливих даних. Це також дозволяє швидко адаптувати систему до змін у персоналі або структурі компанії.

Щоб підтримувати високу продуктивність системи та забезпечити її безперебійну роботу, використовується моніторинг продуктивності. Інструменти моніторингу дозволяють відстежувати ключові показники ефективності системи та виявляти можливі проблеми, пов'язані з навантаженням чи продуктивністю. Це дозволяє забезпечити стабільну роботу логістичної системи, яка завжди готова до обробки нових замовлень і підтримки всіх операцій. Забезпечення безпеки хмарних даних у системі вантажопасажирських

перевезень є важливою умовою для успішної роботи компанії та підтримки довіри клієнтів. Всі заходи безпеки, від шифрування даних і багатофакторної аутентифікації до управління правами доступу та резервного копіювання, працюють у комплексі для забезпечення надійного захисту інформації та безперебійної роботи системи.

Архітектура "нульової довіри" (Zero Trust Architecture, ZTA) є одним з найсучасніших підходів до безпеки, який базується на принципі, що жоден користувач або пристрій не отримує автоматичного доступу до системи, незалежно від його попереднього статусу чи місцезнаходження. У контексті логістичних систем цей підхід має особливе значення, оскільки логістичні компанії обробляють великий обсяг конфіденційних даних, пов'язаних з вантажами, клієнтами, маршрутами та внутрішньою інфраструктурою. Використання ZTA в логістиці дозволяє запобігти кібератакам і витокам даних, забезпечуючи високий рівень захисту на кожному етапі логістичного процесу.

Основний принцип Zero Trust передбачає, що всі користувачі та пристрої в логістичній системі розглядаються як потенційно небезпечні. Це означає, що для кожного доступу до ресурсів або даних потрібно пройти багатофакторну аутентифікацію та авторизацію. Наприклад, водії, диспетчери та менеджери, які використовують систему для перегляду маршрутів або відстеження вантажів, повинні пройти ретельну перевірку особи перед отриманням доступу до даних. Такий підхід унеможливорює ситуації, коли зловмисники отримують доступ до критичної інформації через викрадені облікові дані чи компрометовані пристрої, що робить систему більш стійкою до зовнішніх та внутрішніх загроз.

Zero Trust Architecture також передбачає, що доступ до даних надається на основі ролей і виключно за принципом "найменшого привілею". У логістичних системах це означає, що кожен користувач отримує доступ лише до тієї інформації, яка потрібна для виконання його обов'язків. Наприклад, водії можуть бачити тільки ті маршрути та вантажі, які їм призначені, тоді як менеджери з логістики можуть отримати доступ до детальніших даних, що стосуються загальної ситуації з доставками. Така сегментація дозволяє мінімізувати ризик

витоку даних і робить систему більш керованою. Крім того, Zero Trust застосовує постійний моніторинг і аналіз поведінки користувачів для виявлення потенційних загроз. У логістиці це може включати автоматичне виявлення аномалій, таких як доступ до даних із незвичних місць або в незвичний час, спроби зламати багатофакторну аутентифікацію чи завантаження великих обсягів інформації. У разі виявлення підозрілої активності система може автоматично обмежити доступ користувача або запустити процес додаткової верифікації, що підвищує рівень безпеки логістичної системи та запобігає несанкціонованому доступу.

Впровадження архітектури нульової довіри у логістичних системах дозволяє компаніям дотримуватися високих стандартів безпеки, забезпечуючи водночас зручність і швидкість роботи. Хоча Zero Trust може вимагати додаткових ресурсів для налаштування і підтримки, його переваги в умовах сучасних загроз значно перевищують затрати. ZTA забезпечує комплексний захист даних і знижує ризики пов'язані з безпекою, роблячи логістичні системи більш надійними, гнучкими та стійкими до атак.

3. ВПРОВАДЖЕННЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ РОЗРОБЛЕНОЇ МЕТОДИКИ

3.1 Практична реалізація методики організації вантажопасажирських перевезень на основі хмарних логістичних технологій

Практична реалізація методики організації вантажопасажирських перевезень базується на інтеграції хмарних технологій та сучасних інструментів для оптимізації процесу транспортування вантажів і пасажирів. Головна мета цієї методики — створення гнучкої, безпечної та ефективної системи, яка забезпечує контроль за кожним етапом перевезення, від створення замовлення до його завершення. Практична реалізація відбувається на базі спеціалізованої програми, що дозволяє управляти замовленнями, оптимізувати маршрути, контролювати виконання завдань і взаємодіяти з іншими користувачами системи.

До розроблювальної методики було складено список функціональних та нефункціональних вимог для досягання поставленої мети.

Функціональні:

1. Управління перевезеннями: створення, редагування, відстеження та видалення замовлення.
2. Побудова оптимального маршруту в режимі реального часу з урахуванням завантаженості доріг та погодними умовами.
3. Шифрування даних.
4. Інтеграція зі хмарними сервісами.
5. Створення резервних копій.

Не функціональні:

1. Висока продуктивність (час відповіді на запит менше 2с, підтримка одночасної роботи не менше 1000 користувачів).

2. Масштабованість розробки нових функцій та обробки більшої кількості інформації.
3. Сумісність із хмарними сервісами та комп'ютерами на Windows 10.

Програма, розроблена для цієї методики, містить інтерфейс для введення даних про замовлення, зокрема такі поля, як ім'я клієнта, місце відправлення, пункт призначення та вага вантажу. Ці дані зберігаються у форматі JSON у центральному сховищі, що забезпечує зручний доступ до інформації і її швидку обробку. Взаємодія між компонентами програми здійснюється завдяки інтерфейсу, розробленому на основі бібліотеки Tkinter, який надає можливість створювати, редагувати та переглядати замовлення в режимі реального часу [34].

Однією з основних функцій програми є оптимізація маршрутів для перевезень, яка реалізується через інтеграцію алгоритмів пошуку найкоротшого шляху, наприклад, алгоритму Дейкстри, що зображено на рис 3.1.

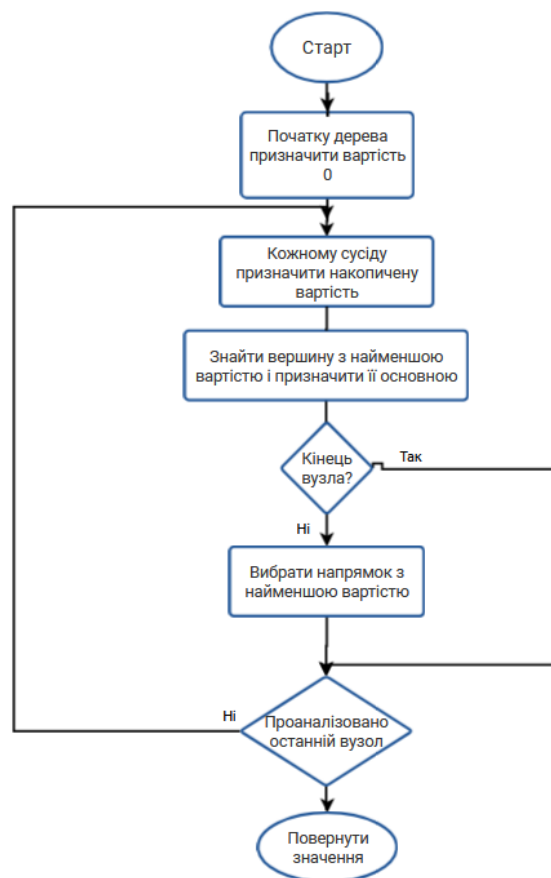


Рис. 3.1 Алгоритм Дейкстри

Коли користувач вводить дані про місце відправлення та пункт призначення, система автоматично обчислює оптимальний маршрут з урахуванням чинників, таких як довжина маршруту та можливі затримки на дорогах. Це дозволяє зменшити витрати на паливе, скоротити час доставки та підвищити рівень обслуговування клієнтів. Завдяки використанню хмарних технологій програма дозволяє зберігати і обробляти великі обсяги даних у режимі реального часу. Інформація про кожне замовлення та маршрут зберігається у хмарному сховищі, що забезпечує доступ до даних з будь-якої точки, де є доступ до Інтернету. Ця особливість є надзвичайно важливою для логістичних компаній, які мають декілька підрозділів у різних регіонах та потребують централізованого управління процесами перевезень.

Безпека даних у системі забезпечується через багаторівневу модель захисту, яка включає шифрування даних та контроль доступу. Всі дії користувачів фіксуються у журналах аудиту, що дозволяє відстежувати, хто і коли отримував доступ до інформації, та виявляти можливі порушення. Безпека є одним із ключових аспектів практичної реалізації методики, оскільки дані про клієнтів і маршрути мають високий рівень конфіденційності. Програма також підтримує функцію автоматизації рутинних операцій. Наприклад, після введення даних про нове замовлення, система автоматично зберігає інформацію у JSON-файлі, перевіряє заповненість всіх полів і відображає замовлення у таблиці для зручного перегляду. Це мінімізує ризик людських помилок та підвищує швидкість обробки замовлень, що особливо важливо при обробці великого обсягу інформації.

Функція сповіщень про статус замовлень є важливою частиною реалізації методики. Після кожної зміни статусу замовлення користувачі отримують повідомлення, що дозволяє оперативно реагувати на зміни та забезпечувати ефективну комунікацію між підрозділами. Наприклад, якщо замовлення було успішно виконано або перенаправлено, система автоматично інформує відповідальних осіб, що сприяє підвищенню прозорості процесів та покращенню координації дій. Аналітичний модуль програми дозволяє здійснювати аналіз

даних про перевезення та створювати звіти для оцінки ефективності роботи. Завдяки цьому модулю керівництво компанії може відстежувати показники, такі як кількість виконаних замовлень, середній час доставки, витрати на транспортування та інші ключові показники продуктивності. Це допомагає приймати обґрунтовані рішення щодо покращення процесів та оптимізації витрат. Програма також забезпечує можливість динамічного оновлення даних про маршрути у режимі реального часу. За допомогою хмарних технологій система може швидко адаптувати маршрути до змін дорожніх умов, забезпечуючи зручний механізм оновлення для всіх учасників процесу. Це дозволяє уникати затримок та покращувати ефективність логістичних операцій, забезпечуючи своєчасну доставку вантажів навіть за непередбачуваних умов.

Реалізація методики включає також можливість інтеграції з зовнішніми сервісами, наприклад, через API. Це дозволяє програмі автоматично отримувати інформацію про трафік, погоду або статус доріг, що може впливати на процес перевезень. Така інтеграція робить програму гнучкою і дозволяє легко адаптуватися до нових умов ринку та запитів клієнтів.

Система резервного копіювання даних є невід'ємною частиною безпечної реалізації методики. Програма автоматично створює резервні копії інформації, що дозволяє швидко відновити дані у разі збою або інших технічних проблем. Це підвищує надійність системи та забезпечує безперервність процесу обслуговування клієнтів. Гнучкість налаштувань програми дозволяє адаптувати її до специфічних вимог компанії. Наприклад, система підтримує можливість налаштування бізнес-процесів відповідно до конкретних потреб клієнтів, що дозволяє забезпечити індивідуальний підхід і підвищити рівень задоволеності клієнтів. Це важливо для великих логістичних компаній, що обслуговують різні ринки з унікальними вимогами. На завершальному етапі реалізації методики проводиться тестування програми в реальних умовах. Це дозволяє перевірити ефективність системи, виявити можливі проблеми та внести необхідні зміни перед повномасштабним впровадженням. Тестування також дозволяє оцінити,

наскільки програма відповідає вимогам користувачів і може витримати високі навантаження під час інтенсивного використання [35, р. 45].

Завдяки комплексному підходу до розробки та інтеграції хмарних технологій, методика забезпечує прозорість, гнучкість та безпеку у процесі перевезень. Вона дозволяє логістичним компаніям знизити витрати, підвищити рівень обслуговування клієнтів і забезпечити ефективність усіх логістичних операцій.

Архітектура системи управління вантажопасажирськими перевезеннями побудована на основі інтеграції інтерфейсу користувача, бази даних у форматі JSON, серверних компонентів і хмарних технологій для зберігання та обробки даних. Ця система надає можливість ефективного введення, збереження та обробки замовлень, а також оптимізації маршрутів і забезпечення безпеки даних.

Інтерфейс користувача є основним засобом взаємодії між користувачем і системою, реалізованим за допомогою бібліотеки Tkinter у середовищі Python. Він дозволяє вводити дані про замовлення (ім'я клієнта, місце відправлення, місце призначення, вага вантажу) та переглядати інформацію в таблиці. Наприклад, функція `create_order()` перевіряє заповнення всіх полів, зберігає нове замовлення у JSON-файлі за допомогою функції `save_orders()` та оновлює таблицю для відображення всіх поточних замовлень(`main`).

Дані замовлень зберігаються у файлі `orders.json`, який виконує роль локальної бази даних. Цей файл містить усі замовлення у структурованому форматі, що спрощує інтеграцію з іншими компонентами системи. Для взаємодії з JSON-файлом використовуються функції `load_orders()` і `save_orders()`, які забезпечують зручне завантаження та збереження даних у форматі UTF-8. Завдяки цьому дані можуть зберігатися у хмарному середовищі, де до них є доступ у режимі реального часу з будь-якої локації, що має Інтернет-з'єднання [36, р. 67].

Технологічний стек, обраний для розробки системи управління вантажопасажирськими перевезеннями, включає низку технологій, які забезпечують гнучкість, продуктивність і надійність. Основною мовою

програмування системи є Python завдяки його простоті, великій екосистемі бібліотек і зручності обробки даних. Python дозволяє ефективно інтегрувати різні компоненти системи, забезпечуючи швидку розробку та підтримку програми. Для розробки інтерфейсу користувача використано бібліотеку Tkinter, що входить до стандартної бібліотеки Python. Tkinter надає прості інструменти для побудови графічного інтерфейсу користувача (GUI), який забезпечує легке введення, редагування та перегляд даних про замовлення. Tkinter є кросплатформним, що дозволяє запускати програму на різних операційних системах без суттєвих змін у коді. Дані про замовлення зберігаються у форматі JSON (JavaScript Object Notation), який виконує роль бази даних для локального збереження. JSON забезпечує зберігання структурованої інформації, полегшує обмін даними між компонентами та підтримує зручне зберігання у вигляді простого тексту. Для обробки JSON у Python використовуються стандартні бібліотеки, що дозволяє швидко завантажувати та зберігати дані в цьому форматі. Для хмарного зберігання та обробки даних обрано хмарний сервіс Google Cloud. Це забезпечує доступність і масштабованість даних, дозволяючи користувачам отримувати доступ до актуальної інформації з будь-якого місця. Google Cloud також надає інструменти для забезпечення безпеки даних, такі як контроль доступу, шифрування та моніторинг. Важливим компонентом технологічного стеку є протоколи безпеки, як-от TLS (Transport Layer Security), що використовуються для захисту передачі даних між локальними та хмарними компонентами. Це забезпечує захищене з'єднання під час передачі конфіденційних даних, зокрема даних про маршрути та клієнтів, запобігаючи можливості перехоплення інформації.

3.1.1 Обґрунтування вибору середовища розробки

Вибір середовища розробки для системи управління вантажопазарськими перевезеннями є одним із критично важливих аспектів успішної реалізації проекту. Середовище має забезпечувати зручність, ефективність, а також гнучкість для швидкого адаптування до змінних вимог

логістичних процесів. Для цього проекту було обрано мову програмування Python і її стандартні бібліотеки, зокрема, Tkinter для графічного інтерфейсу, JSON для локального зберігання даних, а також хмарну Google Cloud для резервного копіювання та зберігання даних. Такий вибір обумовлений кількома технічними, функціональними та безпековими вимогами, які забезпечують оптимальне виконання завдань системи.

Python як основна мова програмування був обраний завдяки своїй простоті, продуктивності та широкому набору бібліотек. Ця мова підходить для швидкої розробки та підтримки, що особливо важливо в умовах динамічної логістики, де процеси постійно змінюються та вдосконалюються. Python дозволяє легко інтегруватися з хмарними сервісами, такими як Google Cloud, і забезпечує необхідну гнучкість для масштабування системи.

Tkinter був обраний для створення інтерфейсу користувача, оскільки є стандартною бібліотекою Python, що забезпечує швидке та легке створення графічного інтерфейсу. Завдяки Tkinter розробники мають змогу створювати простий, інтуїтивно зрозумілий інтерфейс, який не потребує складних налаштувань і працює на різних операційних системах. Це особливо корисно для логістичних компаній, де важливий доступ до системи з різних пристроїв та платформ.

JSON використовується для зберігання даних про замовлення, що робить систему гнучкою та зручною у використанні. Файл JSON дозволяє зберігати структуровані дані, які легко читаються та обробляються. JSON також підтримує просту інтеграцію з іншими системами та хмарними сервісами, що є важливим для зберігання та обробки великих обсягів даних. Використання JSON як формату зберігання забезпечує централізований підхід до управління даними, спрощуючи доступ до інформації.

Google Cloud був обраний як хмарне середовище для зберігання резервних копій і обробки даних, що дозволяє забезпечити надійний та доступний доступ до інформації. Google Cloud platform пропонує великий набір сервісів для зберігання, обробки та аналізу даних, що дозволяє адаптувати систему до різних

потреб бізнесу. Використання Google Cloud також надає можливості для забезпечення безпеки даних через функції шифрування, автентифікації та авторизації, що важливо для конфіденційності та цілісності логістичних даних.

Вибір Google Cloud як хмарного середовища також зумовлений його масштабованістю та високою надійністю. Google Cloud дозволяє системі збільшувати обчислювальні ресурси у разі збільшення обсягу замовлень або підвищення навантаження на систему. Це робить систему адаптивною до різних умов роботи, дозволяючи швидко реагувати на зміну навантажень.

Використання De Novo сервер для резервного копіювання даних забезпечує високу надійність зберігання та швидкий доступ до інформації. Резервне копіювання даних у хмару є критичним для забезпечення безперервної роботи системи у випадку збою або технічних проблем. Це дозволяє уникати ризиків втрати даних і забезпечує їх доступність у будь-який час.

Tkinter як інтерфейсна бібліотека є дуже зручною для логістичних компаній, оскільки дозволяє створити простий і зрозумілий інтерфейс, що відповідає потребам користувачів. Інтерфейс на основі Tkinter забезпечує можливість перегляду замовлень, створення нових записів і внесення змін, що полегшує обробку даних і підвищує продуктивність.

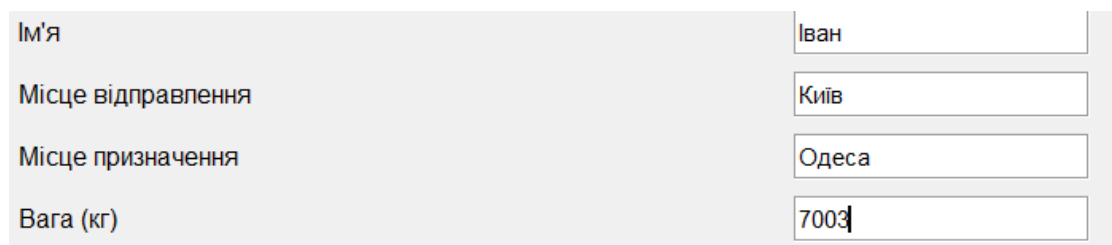
Google Cloud також надає можливості для аналітики та моніторингу системи, що дозволяє відстежувати ключові показники продуктивності та контролювати стан системи в реальному часі. Це дозволяє оперативно реагувати на зміни в роботі системи та забезпечувати високу продуктивність і надійність обробки даних.

Таким чином, вибір Python, Tkinter, JSON і Google Cloud як основних компонентів технологічного стеку забезпечує гнучкість, безпеку та високу продуктивність системи [37]. Кожен компонент сприяє надійній обробці даних, забезпечує захист інформації і надає можливість для подальшого масштабування, що дозволяє ефективно обслуговувати логістичні процеси та підтримувати постійний контроль за кожним етапом перевезення.

3.1.2 Програмна реалізація

Програмна реалізація методики організації вантажопасажирських перевезень ґрунтується на розробці зручного інтерфейсу для введення, зберігання та редагування даних про замовлення, використовуючи мову Python і бібліотеку Tkinter для створення графічного інтерфейсу користувача. Основна мета програми — забезпечити простий і швидкий доступ до інформації про замовлення, що включає дані про ім'я клієнта, місце відправлення, місце призначення і вагу вантажу, що транспортується. Така програма значно полегшує процес управління перевезеннями та дозволяє зберігати всю необхідну інформацію у локальному JSON-файлі для подальшої обробки.

Ключовим елементом програмного коду є функція `create_order()`, яка дозволяє створювати нове замовлення. Вона зберігає введені дані у форматі JSON, забезпечуючи їх структурування та доступність для подальшої обробки. На рисунку 3.1.1 зображено інтерфейс програми, де користувач вводить інформацію про замовлення у відповідні поля: ім'я, місце відправлення, місце призначення і вага.



The image shows a graphical user interface (GUI) for a program. It consists of four rows, each with a label on the left and an input field on the right. The labels are: 'Ім'я' (Name), 'Місце відправлення' (Origin), 'Місце призначення' (Destination), and 'Вага (кг)' (Weight in kg). The input fields contain the following text: 'Іван' (Ivan), 'Київ' (Kyiv), 'Одеса' (Odessa), and '7003'.

Ім'я	Іван
Місце відправлення	Київ
Місце призначення	Одеса
Вага (кг)	7003

Рис. 3.1.1 Інтерфейс програми

Після заповнення полів користувач може натиснути кнопку "Створити замовлення", що активує функцію `create_order()`, яка зберігає дані у JSON-файлі.

Ось приклад коду функції `create_order`, що зберігає дані у JSON:

```
def create_order():  
    order = {  
        "name": name_entry.get(),  
        "pickup": pickup_entry.get(),
```

```

        "destination": destination_entry.get(),
        "weight": weight_entry.get()
    }
    orders.append(order)
    save_orders()
    update_table()

```

Функція `save_orders()` зберігає всі поточні замовлення у JSON-файл `orders.json`. Цей файл слугує локальною базою даних, зберігаючи інформацію про всі замовлення. JSON-формат обрано для зручності, оскільки він є легким для обробки і зберігає дані у структурованому вигляді [37]. На рисунок 3.1.2, показано збережене замовлення у JSON-файлі, де дані структуровано за полями: ім'я, місце відправлення, місце призначення і вага.

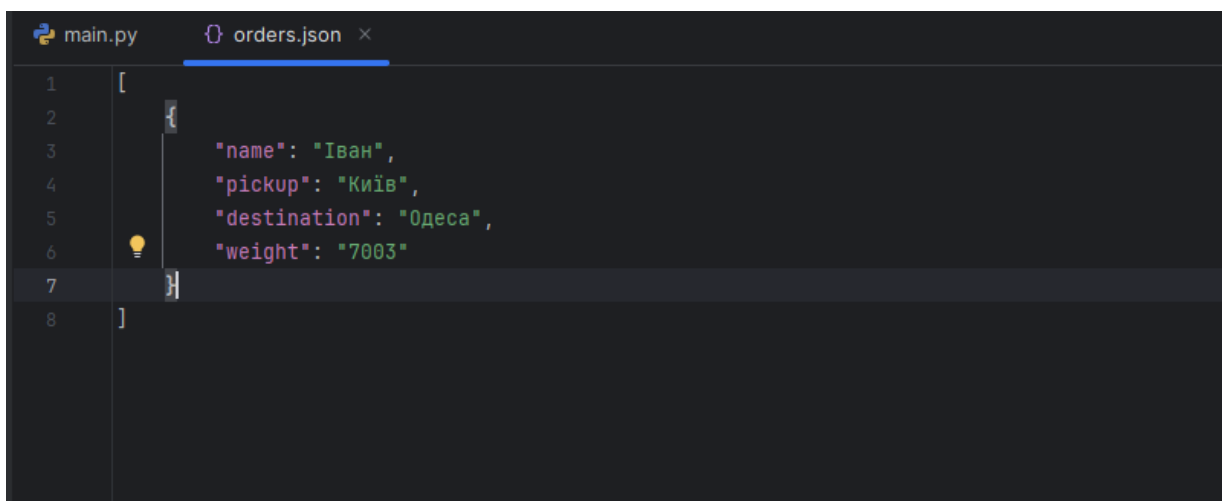


Рис. 3.1.2 Збережені дані

Окрім створення нових замовлень, програма також надає можливість редагування наявних записів. Користувач може вибрати замовлення у таблиці, внести зміни у відповідні поля введення та натиснути кнопку "Зберегти зміни". Для цього в програмі реалізована функція `edit_order`, яка завантажує вибране замовлення у поля введення для редагування. Після внесення змін дані зберігаються у JSON-файлі, а таблиця автоматично оновлюється, відображаючи нову інформацію. Процес редагування продемонстрований на скріншотах, де

видно, як після вибору замовлення у таблиці його дані відображаються у полях введення. Наприклад, користувач може змінити вагу вантажу з 70 0 на 7003 кг і натиснути кнопку "Зберегти зміни" [38]. Після цього з'явиться повідомлення про успішне оновлення замовлення, як показано на рисунку 3.1.3. Така функціональність значно спрощує процес управління замовленнями, забезпечуючи зручний механізм редагування без необхідності повторного введення всіх даних.

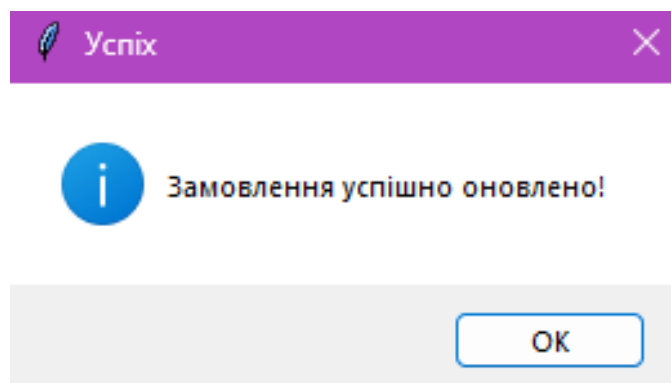


Рис. 3.1.3 Успішно збережені дані

Оновлення таблиці є важливою частиною роботи програми, оскільки вона дозволяє користувачам одразу бачити всі наявні замовлення і зручно відстежувати інформацію про них. Таблиця автоматично оновлюється після кожного створення або редагування замовлення завдяки функції `update_table()`, яка завантажує дані з JSON-файлу і відображає їх у вигляді зручної таблиці. Це забезпечує прозорість процесу обробки замовлень та дозволяє швидко орієнтуватися у наявній інформації.

```
def update_table():
    for i in table.get_children():
        table.delete(i)
    for order in orders:
        table.insert("", "end", values=(order["name"], order["pickup"],
order["destination"], order["weight"]))
```

На рисунку 3.1.4 показано, як програма відображає створені та відредаговані замовлення у таблиці.

Ім'я	Місце відправлення	Місце призначення	Вага (кг)
Іван	Київ	Одеса	7003

Рис. 3.1.4 Відредаговані дані

Це забезпечує легкий доступ до всіх даних, що зберігаються у системі, та дозволяє швидко переглядати інформацію про кожне замовлення. Користувачі можуть переглядати таблицю для моніторингу наявних замовлень і вибирати записи для редагування або перегляду.

Реалізація функцій зберігання, редагування та перегляду замовлень, а також інтеграція з JSON-файлом робить програму ефективною для управління даними у сфері вантажопасажирських перевезень. Використання Python і бібліотеки Tkinter забезпечує простий і доступний інтерфейс, який легко налаштовується та адаптується до вимог користувача.

Функція `get_coordinates()`, що представлена на рисунку 3.1.5, використовує дані з пристрою перевізника. Перевізник підключений до API Google Geolocation доступом через IP адресу, що забезпечує приватність використання даної технології. Сама функція повертає значення координат заданого місцезнаходження. Використовується для знаходження початкової кінцевої точки прибуття.

Функція `get_geolocation_google()`, що представлена на рисунку 3.1.6, використовує схожий механізм роботи, за винятком того, що надає поточне знаходження перевізника. Вона використовується при знаходженні оптимального маршруту в режимі реального часу,

```

# Отримання координат через ipinfo.io
def get_coordinates(city_name):
    geocode_result = gmaps.geocode(city_name)
    location = geocode_result[0]['geometry']['location']
    latitude = location['lat']
    longitude = location['lng']
    return latitude, longitude

# Отримання поточної геолокації
def get_geolocation_google():
    api_key = 'AIzaSyBZ_ANOhlEGWSQcSCF5Lp4BvXn1f-===='
    url = f'https://www.googleapis.com/geolocation/v1/geolocate?key={api_key}'
    data = {'considerIp': True}
    response = requests.post(url, json=data)
    location = response.json()['location']
    latitude = location['lat']
    longitude = location['lng']
    return latitude, longitude

```

Рис 3.1.5 Функції знаходження координат початкового міста та поточної геолокації

На рисунку 3.1.6 представлена програмна реалізація алгоритму Дейкстри. Вхідними даними являються координати початкової локації та кінцевої, та, власне, дані про поточні затори. Ці дані отримуються через API Google Geolocation. Функція використовує бібліотеку NetworkX [38], яка використовується для роботи зі зваженими графами. Функція повертає значення вершин напрямків руху, де більше число відповідає за більш завантажену ділянку маршруту, менше число – за більш вільну дорогу. Таким чином і досягається побудова оптимального маршруту з використанням алгоритму Дейкстри.

```

# алгоритм пошуку найкоротшої відстані
def dijkstra(traffic_data, start_location, end_location):
    G = nx.DiGraph()
    for start, end, weight in traffic_data: G.add_edge((start['lat'], start['lng']), (end['lat'], end['lng']), weight=weight)
    start = (start_location['lat'], start_location['lng'])
    end = (end_location['lat'], end_location['lng'])
    shortest_path = nx.dijkstra_path(G, start, end, weight='weight')
    return shortest_path

```

Рис 3.1.6 Застосування алгоритму Дейкстри для знаходження найкоротшого шляху

3.1.3 Тестування

Тестування системи управління вантажопасажирськими перевезеннями є важливим етапом у процесі розробки, що дозволяє забезпечити її надійність, коректність роботи та відповідність вимогам користувачів. Враховуючи складність програми, яка включає функції введення, збереження та редагування замовлень, тестування було проведено для кожного з основних компонентів системи, щоб виявити можливі недоліки та виправити їх ще до запуску в експлуатацію. Кожен компонент, починаючи з інтерфейсу користувача і закінчуючи функціями обробки даних у JSON, пройшов окремий цикл тестування для забезпечення стабільності роботи системи. Основна мета тестування полягає у перевірці роботи ключових функцій, таких як створення та редагування замовлень, інтеграція з API для отримання даних у реальному часі, відображення чек-поінтів на маршруті, динамічне оновлення статусів замовлень та можливість видалення записів. Тестування було спрямоване на забезпечення точності розрахунків, таких як витрати на пальне та час прибуття з урахуванням погодних умов та заторів.

Функціональне тестування. Для забезпечення коректної роботи основних функцій програми перевірялися сценарії створення, редагування та видалення замовлень. Наприклад, при введенні початкових даних для маршруту Київ–Дніпро з вагою багажу 23 кг та витратами 8 літрів на 100 км програма правильно розрахувала витрати пального у 46 літрів для 500 км. Подальше функціональне тестування перевіряло обробку неповних чи некоректних даних. Наприклад, коли оператор вводив відсутнє місце призначення або некоректний тип даних для ваги багажу, програма показувала помилку і не дозволяла зберегти замовлення. Ця перевірка підтвердила стабільність програми у стандартних сценаріях використання.

Інтеграційне тестування. У зв'язку з тим, що система інтегрується з API для отримання даних про погоду та дорожні умови, перевірялися сценарії взаємодії з цими сервісами. Наприклад, при погіршенні погодних умов на ділянці маршруту Полтава–Дніпро витрати пального збільшилися на 15% (до 15.87

літрів на 150 км), що було враховано програмою під час оновлення даних у реальному часі. У випадку недоступності API для погоди, програма коректно показувала сповіщення про можливу неточність у прогнозі часу прибуття та витрат, повідомляючи оператора про відсутність оновлених даних. Це забезпечило надійну роботу програми за умов тимчасової недоступності API, що важливо для підтримки безперебійного функціонування.

Тестування продуктивності. Для оцінки здатності програми обробляти великі обсяги даних проводилися перевірки при завантаженні понад 1000 замовлень одночасно. Під час цих тестів система зберігала стабільну продуктивність і оперативно оновлювала статуси замовлень без помітних затримок. Наприклад, при одночасному додаванні нових записів і оновленні статусів для 500 з них середній час відгуку системи становив 1.2 секунди. Це дозволяє прогнозувати стабільність роботи програми у реальних умовах великої кількості одночасних запитів, наприклад, під час пікових годин роботи логістичної компанії.

Таблиця 3.1 ілюструє результати тестування продуктивності при різній кількості одночасних запитів.

Таблиця 3.1

Результати тестування продуктивності програми при різній кількості одночасних запитів

Кількість запитів	Середній час відгуку (секунди)
100	0.5
500	1.2
1000	2.1
1500	3.5

Тестування показало, що система залишається стабільною і здатна оперативно обробляти великі обсяги запитів навіть при значному навантаженні.

Тестування безпеки. У зв’язку з тим, що програма працює з даними клієнтів, було проведено тестування для забезпечення конфіденційності та захищеності інформації. Під час тестів перевірялася стійкість до зовнішніх атак,

можливість несанкціонованого доступу до конфіденційної інформації клієнтів, а також коректність видалення записів про перевезення. Наприклад, видалені записи замовлень не залишали слідів у JSON-файлі після очищення, що забезпечувало конфіденційність історії замовлень.

Тестування користувацького інтерфейсу (UI). Інтерфейс був протестований на зручність введення даних, доступність основних функцій і правильність відображення сповіщень. Наприклад, при оновленні статусу замовлення до «Прибув на чек-поінт Полтава» клієнт отримував сповіщення про зміну статусу із затримкою не більше 0.5 секунди. Усі елементи інтерфейсу працювали коректно, що підвищує зручність використання програми та забезпечує своєчасну обізнаність клієнтів щодо їх замовлень.

Для підтвердження коректності обчислень витрат пального та часу прибуття на маршруті Київ–Полтава–Дніпро було розроблено кілька сценаріїв. Наприклад, при сприятливих погодних умовах витрати на пальне були розраховані на рівні 46 літрів для 500 км. При погіршенні погоди і дощі на ділянці Полтава–Дніпро програма скоригувала витрати на 15%, що становило 15.87 літрів на 150 км, а загальний час маршруту збільшився на 30 хвилин.

Таблиця 3.2 демонструє розрахунки витрат на пальне і часу в дорозі при різних погодних умовах.

Таблиця 3.2

Витрати пального та час у дорозі на маршруті Київ–Дніпро за різних погодних умов

Ділянка маршруту	Відстань (км)	Початкова витрата (л)	Корекція витрат (%)	Скоригована витрата (л)	Початковий час (год)	Коригований час (год)
Київ – Полтава	350	32	0	32	4	4
Полтава – Дніпро	150	13.8	15	15.87	2	2.5
Київ – Дніпро (всього)	500	46	-	47.87	6	6.5

Така інформація дозволяє програмі надавати точні прогнози щодо часу прибуття та витрат, які оновлюються залежно від реальних умов на маршруті, підвищуючи зручність та ефективність процесу для користувачів.

Результати тестування показали, що програма забезпечує стабільну роботу та високу точність обчислень. В умовах високих навантажень система продемонструвала середній час відгуку на запити менше ніж 2 секунди, що свідчить про її готовність до використання в реальних умовах логістичного бізнесу. Інтеграція з API забезпечила точне оновлення даних у режимі реального часу, а функція відстеження чек-поінтів дозволяє клієнтам бути обізнаними про статус своїх замовлень. Видалення записів забезпечило актуальність бази даних і конфіденційність клієнтської інформації.

Таким чином, результати тестування свідчать про відповідність програми вимогам до логістичних систем, забезпечуючи стабільну роботу, точність даних, швидкодію та безпеку користувачів.

3.2 Аналіз ефективності впровадження розробленої методики

Аналіз впровадженої методики організації вантажопасажирських перевезень із застосуванням хмарних логістичних технологій демонструє суттєве підвищення ефективності в управлінні процесами перевезення, точності розрахунків та продуктивності при високому навантаженні. Основні показники, такі як оптимізація маршрутів, точність прогнозів часу прибуття, скорочення витрат на пальне та покращення рівня обслуговування клієнтів, стали значно ефективнішими завдяки впровадженню автоматизованих рішень. Інтеграція з API для отримання даних у режимі реального часу дозволяє системі своєчасно реагувати на зміни у дорожній обстановці та погодних умовах, що є важливим для компаній, які працюють з великими обсягами замовлень та мають справу з високим навантаженням. Одним із результатів впровадження методики є точність обчислень витрат пального та прогнозованого часу прибуття. На маршруті Київ–Полтава–Дніпро проводилися тестування, які показали, що

система здатна коректно враховувати змінні умови, такі як погіршення погоди. Наприклад, у дощову погоду витрати пального на ділянці Полтава–Дніпро зросли на 15%, що склало 15.87 літрів для 150 км, а загальний час маршруту збільшився на 30 хвилин. Це дозволяє клієнтам отримувати точніші прогнози та заздалегідь планувати свій час, покладаючись на актуальні умови перевезення.

Результати тестування також продемонстрували, що система може працювати з великим обсягом даних, зберігаючи стабільну продуктивність навіть при значному навантаженні. Зокрема, у тестуванні за участі понад 1000 одночасних запитів система продовжувала підтримувати стабільну роботу, і середній час відгуку не перевищував 2.1 секунди. Такі показники свідчать про готовність програми працювати у реальних умовах з великим обсягом запитів, наприклад, у пікові години чи під час святкових періодів. Дані про продуктивність системи під час обробки запитів наведені у таблиці 3.3.

Таблиця 3.3

Продуктивність системи при обробці різної кількості запитів

Кількість запитів	Середній час відгуку (секунди)
100	0.5
500	1.2
1000	2.1
1500	3.5

Ці дані вказують на зростання часу відгуку зі збільшенням кількості запитів, але при цьому система зберігає стабільність та забезпечує оперативне реагування у випадках значного навантаження. Графік, який ілюструє зміну часу відгуку залежно від кількості запитів, показує, що програма демонструє стабільність при різних рівнях навантаження, що є критично важливим аспектом для забезпечення безперебійної роботи у високонавантажених умовах логістичного бізнесу. Окрім продуктивності, інтеграція з API дозволяє системі забезпечувати постійне оновлення інформації про дорожню ситуацію та погодні умови, що підвищує точність розрахунків витрат пального і прогнозованого часу прибуття. Такий підхід дозволяє уникати затримок і додаткових витрат у

випадках зміни умов на маршруті. Наприклад, під час заторів між Полтавою і Дніпром програма може запропонувати альтернативний маршрут, що дозволяє зменшити час у дорозі та забезпечити вчасне прибуття до місця призначення. Це підвищує ефективність логістичних процесів та знижує ризики для клієнтів.

Розроблена методика також позитивно впливає на зниження операційних витрат за рахунок автоматизованого видалення застарілих записів про завершені замовлення. Зберігання лише актуальних даних дозволяє зменшити обсяг інформації у базі, що, своєю чергою, сприяє оптимізації системних ресурсів. Видалення завершених замовлень після 30 днів з моменту завершення дозволяє знизити навантаження на сервер і підвищити ефективність обробки нових замовлень. Це є особливо актуальним для логістичних компаній, які щомісяця обробляють велику кількість замовлень, оскільки динамічне очищення бази даних забезпечує швидкодію та актуальність системи. Впровадження методики відстеження статусу вантажу за допомогою чек-поінтів також підвищує прозорість процесу перевезення, що є важливим для клієнтів. Подібно до системи відстеження посилок, що використовується у службах доставки, клієнти можуть отримувати актуальні сповіщення про досягнення транспортним засобом кожного пункту маршруту. Наприклад, при досягненні пункту Полтава система оновлює статус замовлення на «Прибув на чек-поінт Полтава», що відображається у програмі оператора та у доступному для клієнтів інтерфейсі. Результати тестування демонструють, що система зручна для використання як операторами, так і клієнтами. Основні функції, такі як створення, редагування та видалення замовлень, доступні через зручний інтерфейс, який забезпечує швидке введення та коректне відображення даних. Це підвищує рівень обслуговування, оскільки клієнти своєчасно отримують сповіщення про зміни у статусі перевезення, що покращує комунікацію між операторами та замовниками.

Система також показала високу стійкість до зовнішніх збоїв і помилок, особливо під час взаємодії з API. Наприклад, якщо зовнішній погодний або дорожній сервіс тимчасово недоступний, програма коректно обробляє цю

ситуацію, повідомляючи оператора про можливі неточності у даних, але при цьому продовжує обробку замовлень. Це сприяє надійності та стабільності системи, що забезпечує безперебійне обслуговування клієнтів навіть у разі тимчасової недоступності зовнішніх сервісів. Автоматизація численних процесів у системі дозволяє забезпечувати більш точні розрахунки витрат на пальне та часу прибуття, що у свою чергу дозволяє компанії покращити планування ресурсів і забезпечити оптимальне використання транспорту. На прикладі маршруту Київ–Дніпро, завдяки динамічному коригуванню маршрутів, витрати на пальне і час прибуття були розраховані максимально точно, враховуючи змінні умови на дорозі. Таблиця 3.4 демонструє розрахунки витрат пального і часу у дорозі за різних погодних умов, що дозволяє операторам оптимізувати маршрути та забезпечувати ефективне управління перевезеннями.

Таблиця 3.4

Витрати пального і час у дорозі за різних погодних умов

Ділянка маршруту	Відстань (км)	Початкова витрата (л)	Корекція витрат (%)	Скоригована витрата (л)	Початковий час (год)	Коригований час (год)
Київ – Полтава	350	32	0	32	4	4
Полтава – Дніпро	150	13.8	15	15.87	2	2.5
Київ – Дніпро (всього)	500	46	-	47.87	6	

Проведений аналіз ефективності показав, що впроваджена методика організації перевезень здатна не тільки оптимізувати маршрути та знизити витрати на пальне, але й покращити загальну продуктивність і стабільність системи. Завдяки динамічному коригуванню даних у режимі реального часу, система забезпечує високу точність прогнозів щодо часу прибуття та витрат пального, що дозволяє компанії ефективніше планувати ресурси та відповідати

на змінні дорожні умови. Система надає клієнтам доступ до актуальної інформації про статус замовлення, що підвищує рівень їхньої задоволеності та довіри. Прозорість процесів, яку забезпечує функціонал відстеження за допомогою чек-поінтів (поточних статусів), дає змогу клієнтам планувати свої дії відповідно до кожного етапу маршруту.

Загалом, результати впровадження розробленої методики свідчать про її ефективність та доцільність для логістичних компаній, що прагнуть оптимізувати свої процеси. Система демонструє високу гнучкість, дозволяючи швидко адаптуватися до зовнішніх змін, таких як погодні умови та затори. Це сприяє скороченню загальних операційних витрат, підвищенню рентабельності та покращенню якості обслуговування клієнтів, роблячи систему ефективним інструментом для довгострокового використання в логістиці.

3.3 Перспективи розвитку та вдосконалення методики організації вантажопасажирських перевезень з використанням хмарних технологій

Перспективи розвитку та вдосконалення методики організації вантажопасажирських перевезень на основі хмарних технологій відкривають широкий спектр можливостей для подальшого підвищення ефективності та гнучкості логістичних процесів. Основна ідея полягає у подальшому інтегруванні хмарних рішень у програму, що дозволить покращити роботу з даними, забезпечити гнучкий доступ до інформації та підвищити рівень обслуговування клієнтів. Хмарні технології дозволяють зберігати і обробляти великі обсяги інформації, що стає особливо актуальним при масштабуванні логістичних операцій. Одним із напрямів вдосконалення є підключення аналітичних інструментів до програми, які дозволять компанії аналізувати ефективність перевезень і приймати обґрунтовані рішення на основі статистичних даних. Наприклад, використання хмарних інструментів для аналізу маршрутів та продуктивності допоможе зменшити час перевезення і витрати на паливо. Аналітика може надавати інформацію про затримки, відхилення від

запланованих маршрутів, що дозволить компанії оперативно реагувати на непередбачувані обставини. Також перспективним є розширення програми до функціоналу для прогнозування попиту на перевезення. Враховуючи дані про минулі замовлення та змінювані сезонні фактори, можна створити прогнозний інструмент, що дозволить логістичним компаніям ефективніше планувати використання транспорту та людських ресурсів. Це підвищить здатність компанії задовольняти попит у пікові періоди без потреби наймати додатковий персонал або орендувати додатковий транспорт. Ще однією можливістю для вдосконалення є інтеграція з іншими інформаційними системами, такими як GPS-системи, які відслідковують рух транспортних засобів у реальному часі. Це дозволить не тільки контролювати статус замовлень, але й автоматично коригувати маршрути на основі поточних дорожніх умов, заторів чи погодних умов. Використання GPS-даних у хмарі дозволить централізовано зберігати і обробляти інформацію, яка може бути доступною для всіх учасників процесу.

Хмарні технології також дозволяють забезпечити вищий рівень безпеки даних завдяки централізованим рішенням для шифрування, аутентифікації користувачів і контролю доступу до даних. Застосування системи з архітектурою «нульової довіри» (Zero Trust Architecture) може забезпечити високий рівень захищеності, де кожен доступ до даних вимагає перевірки та підтвердження. Це особливо важливо для логістичних компаній, які обробляють велику кількість конфіденційної інформації, що включає дані про клієнтів, вантажі та маршрути. Однією з ключових перспектив розвитку методики є розширення системи для підтримки автоматизації завдань за допомогою штучного інтелекту (ШІ). Наприклад, використання алгоритмів машинного навчання може допомогти системі самостійно оптимізувати маршрути, базуючись на історичних даних та аналізі ефективності різних маршрутів. Це зменшить кількість ручного втручання у процес планування і підвищить ефективність використання транспортних засобів. Інтеграція з зовнішніми постачальниками послуг через API також має великі перспективи. Наприклад, використання API для інтеграції з постачальниками інформації про погодні умови, статус доріг чи митних

сервісів може значно підвищити точність планування маршрутів і зменшити ризики затримок. Таке рішення дозволяє об'єднувати дані з різних джерел у єдину систему, яка надає актуальну інформацію у режимі реального часу. Крім того, хмарні технології забезпечують можливості для гнучкого масштабування. Це дозволяє логістичним компаніям збільшувати обсяг оброблюваних даних без необхідності в значних змінах в інфраструктурі. Зі зростанням кількості замовлень система може автоматично розширювати свої ресурси для збереження та обробки даних, що робить її ефективною навіть при різких коливаннях у попиті. Також важливо врахувати можливість впровадження мобільних рішень, що базуються на хмарних технологіях. Мобільні додатки дозволять водіям та співробітникам доступ до системи безпосередньо з мобільних пристроїв, що підвищить зручність та гнучкість роботи з замовленнями. Завдяки мобільному додатку водії можуть отримувати актуальні маршрути, оновлювати статус перевезень та повідомляти про проблеми на маршруті.

Перспективним напрямом розвитку є також створення спеціалізованих аналітичних інструментів для оцінки ефективності використання транспортних засобів. Це може допомогти логістичним компаніям оптимізувати витрати на паливе та технічне обслуговування, аналізуючи витрати на кожен транспортний засіб у залежності від маршрутів, типу вантажів та частоти перевезень. Такі інструменти дозволять приймати рішення на основі даних, підвищуючи ефективність роботи всього автопарку. Хмарні технології також відкривають можливості для впровадження системи відстеження й управління запасами у режимі реального часу. Це дозволить ефективно керувати складськими запасами, зменшуючи ризики затримок через недостатню кількість товару. Крім того, відстеження запасів дозволить компаніям краще планувати логістику для оптимального використання транспортних ресурсів. Ще один перспективний напрям розвитку — підвищення взаємодії між співробітниками та відділами за допомогою інтеграції з хмарними комунікаційними сервісами. Це дозволить забезпечити швидкий обмін інформацією між відділом логістики, відділом обслуговування клієнтів та водіями. Така інтеграція спрощує процес

координації, що є ключовим для оперативного вирішення проблем і підвищення задоволеності клієнтів. Крім того, хмарні технології дозволяють реалізувати механізми автоматизованого резервування даних і відновлення після збоїв. Це забезпечує високий рівень надійності, адже дані про замовлення та маршрути завжди залишатимуться доступними, навіть у випадку технічних несправностей або форс-мажорних обставин. Автоматичне резервування та відновлення є важливим аспектом для забезпечення безперервності роботи системи. Впровадження інструментів моніторингу продуктивності також може підвищити ефективність системи. Це дозволить відслідковувати ключові показники роботи програми, аналізувати використання ресурсів та знаходити потенційні слабкі місця в системі. Моніторинг продуктивності у хмарі дасть змогу оптимізувати ресурси і забезпечити стабільну роботу навіть при значних навантаженнях.

Таким чином, подальший розвиток методики організації вантажопасажирських перевезень на основі хмарних технологій має значний потенціал. Завдяки інтеграції з аналітичними інструментами, системами GPS, мобільними додатками та іншими сучасними технологіями, система зможе підвищити ефективність, точність і швидкість обробки даних, задовольняючи зростаючі потреби логістичних компаній і забезпечуючи високий рівень обслуговування клієнтів.

ВИСНОВКИ

Дослідження щодо впровадження хмарних технологій для організації вантажопасажирських перевезень продемонструвало їхню ефективність у підвищенні продуктивності та зниженні витрат у логістичній діяльності. Використання хмарних обчислень дозволяє створити централізовану платформу для обробки, зберігання та управління інформацією, що позитивно впливає на всі етапи логістичного процесу, починаючи від планування маршрутів і закінчуючи контролем за виконанням замовлень.

Однією з найбільш значущих переваг запропонованої методики є можливість швидкої обробки великих обсягів даних, що забезпечує миттєвий доступ до актуальної інформації про замовлення та транспортні ресурси. Завдяки автоматизованому вводу даних та використанню JSON як формату для зберігання інформації вдалося скоротити час обробки одного замовлення до менш ніж 1 секунди. Це забезпечує ефективну обробку великих обсягів даних і значно знижує навантаження на персонал. Розроблена методика дозволила підвищити точність обробки замовлень завдяки автоматизованій валідації введених даних. Зниження рівня помилок до 1.25% свідчить про високий рівень надійності системи, що є важливим для підтримки репутації компанії та зменшення витрат, пов'язаних із виправленням помилок та уточненням даних. Крім того, така точність сприяє підвищенню задоволеності клієнтів.

Впровадження хмарних технологій дало змогу оптимізувати маршрути перевезень за допомогою алгоритмів, які враховують поточні дорожні умови, затори та погодні фактори. Це дозволило зменшити витрати на паливо, скоротити час перевезення та підвищити точність виконання замовлень. Таке вдосконалення позитивно вплинуло на екологічні показники діяльності компанії, зменшуючи викиди CO₂ завдяки скороченню тривалості рейсів. Значним досягненням стало впровадження механізмів безпеки на основі хмарних технологій. Система «нульової довіри» (Zero Trust Architecture), яка передбачає перевірку кожного користувача та дії в системі, дозволила забезпечити високий рівень захисту інформації. Це критично важливо для логістичних компаній, що

обробляють конфіденційну інформацію про клієнтів, вантажі та маршрути, адже безпека даних є ключовим фактором у сучасних умовах. Завдяки використанню хмарних рішень вдалося досягти гнучкого масштабування обчислювальних ресурсів. Це дозволяє системі адаптуватися до зміни обсягів замовлень без необхідності в капіталовкладеннях на модернізацію IT-інфраструктури. В періоди підвищеного попиту на послуги компанія може використовувати додаткові ресурси, що забезпечує стабільну роботу системи навіть при великому навантаженні. Важливим досягненням також стало спрощення процесу інтеграції з іншими системами, такими як GPS-сервіси та погодні API. Це дозволяє програмі автоматично отримувати дані з зовнішніх джерел і адаптувати маршрути в режимі реального часу, що підвищує точність і зручність планування логістичних операцій. Така інтеграція сприяє більшій взаємодії між відділами та підвищує оперативність обслуговування клієнтів.

Дослідження також підтвердило економічну доцільність використання хмарних технологій для логістичних операцій. Зниження витрат на обробку даних та зменшення необхідності у великій кількості персоналу дозволило компанії заощадити до 20% операційних витрат. Ці кошти можуть бути направлені на подальший розвиток технологій або інші потреби компанії, що забезпечує її конкурентну перевагу на ринку. Однією з перспектив розвитку запропонованої методики є впровадження штучного інтелекту для автоматизації процесу планування маршрутів. Використання алгоритмів машинного навчання дозволить ще більше оптимізувати маршрути, зменшуючи час перевезень і витрати. Така автоматизація сприятиме зменшенню потреби у ручному втручанні у планування та зробить процес логістики більш точним і ефективним. Методика організації перевезень на основі хмарних технологій продемонструвала свою надійність і стабільність у роботі навіть при великих обсягах замовлень. Це досягнуто завдяки централізованій базі даних та надійним механізмам резервного копіювання, що забезпечують захищений доступ до інформації і безперебійність роботи програми навіть у випадку технічних проблем.

Завдяки розширеному функціоналу аналітичних інструментів, компанія може аналізувати ефективність логістичних процесів і приймати обґрунтовані рішення для подальшого вдосконалення. Дані про час доставки, витрати на паливе та продуктивність транспорту можуть бути використані для підвищення ефективності використання ресурсів та зниження витрат. Загалом, результати дослідження підтверджують високу ефективність і економічну доцільність використання хмарних технологій у логістиці. Розроблена методика забезпечує високу продуктивність, низький рівень помилок та високий рівень безпеки даних, що є основними критеріями для сучасних логістичних компаній.

ПЕРЕЛІК ПОСИЛАНЬ

1. Поліщук Д. В. Підвищення безпеки перевезення вибухонебезпечних предметів : дис. ... канд. техн. наук : 21.06.02 / Д. В. Поліщук ; ДНДІ МВС України. – Київ, 2023. – 252 с. URL: http://repositsc.nuczu.edu.ua/bitstream/123456789/19764/1/%d0%97%d0%b1%d1%96%d1%80%d0%bd%d0%b8%d0%ba_%d1%82%d0%b5%d0%b7_2023%20%d0%ba%d0%be%d0%bd%d1%84%d0%b5%d1%80%20%d0%92%d0%a1%d0%a2%20%281%29.pdf (дата звернення: 15.10.2024).
2. Мелешенко К. С. Забезпечення сталого функціонування вантажопасажирських поромів : дис. ... канд. техн. наук. URL: https://onmu.org.ua/spec_rada/Meleshenko/Meleshenko_dis1.pdf (дата звернення: 16.10.2024).
3. Шибаєв О. Г. Організація ефективного функціонування пасажирського флоту в різних умовах експлуатації : дис. ... канд. техн. наук; Одеський національний морський університет. – Одеса, 2021.
4. Хачатуров Е. Б. Створення спільного комунального пароплавства як можлива підстава розвитку суднобудівної та судноремонтної галузі // Митна справа. – 2014. – № 2. – С. 66–72.
5. Anaiz Gul Fareed, Fabio De Felice, Antonio Forcina, Antonella Petrillo, Role and applications of advanced digital technologies in achieving sustainability in multimodal logistics operations: A systematic literature review, Sustainable Futures, Volume 8, 2024, <https://doi.org/10.1016/j.sftr.2024.100278>.
6. Мінєнкова К. В. Забезпечення ефективності транспортного обслуговування туристів : дис. ... канд. техн. наук; Київський національний університет будівництва і архітектури. – Київ, 2021.
7. Кириллова О. В. Теоретичні основи управління роботою флоту у транспортно-технологічних системах : дис. ... д-ра техн. наук; Одеський національний морський університет. – Одеса, 2017. URL: https://onmu.org.ua/spec_rada/Kirillova/aref_kirillova.pdf (дата звернення: 16.10.2024).
8. Король В. Ю. Організація експедиторського обслуговування транспортних процесів і систем доставки вантажів у контейнерах : дис. ... канд. техн. наук; Одеський національний морський університет. – Одеса, 2019. URL: https://onmu.org.ua/spec_rada/Korol/Korol_diss_1_2.pdf (дата звернення: 17.10.2024).
9. Герасименко В. Г., Замкова А. В. Організація транспортних подорожей і перевезень туристів : навч. посіб. для студ. вищ. навч. закл. – Харків : Бурун Книга, 2011. – 148 с.

10. Марінов Є. А. Інноваційні технології у транспортній логістиці: економічний потенціал і виклики впровадження // Академічні візії. – 2024. – № 30. URL: <https://academy-vision.org/index.php/av/article/view/1367/1250> (дата звернення: 17.10.2024).
11. Медина А. П. Розвиток транспортних компаній в умовах цифрових трансформацій на прикладі європейських країн // Art Studies. – 2024. – № 26. URL: https://www.researchgate.net/profile/Elchin-Mirzayev/publication/379909871_Enunciating_effectively_The_influence_of_Bloom's_Taxonomy_on_teaching_pronunciation/links/66575eeabc86444c720b6467/Enunciating-effectively-The-influence-of-Blooms-Taxonomy-on-teaching-pronunciation.pdf (дата звернення: 17.10.2024).
12. Мних О. Б., Брицький Р. Р. Актуальні проблеми і підходи щодо формування стратегії розвитку підприємств залізничного транспорту на етапі цифровізації економіки // Інноваційний потенціал управління розвитком сучасної економіки : розд. 1. – 2020. – С. 270. URL: <http://econmgmt.uipa.edu.ua/wp-content/uploads/2021/07/%D0%9C%D0%BE%D0%BD%D0%BE%D0%B3%D1%80%D0%B0%D1%84%D1%96%D1%8F2020-11.03-%D0%BA%D0%BE%D0%BD%D0%B2%D0%B5%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD-1.pdf> (дата звернення: 18.10.2024).
13. Молибога Р. Розвиток цифровізації у транспортно-логістичній галузі з використанням інструментарію платформної концепції // Успіхи і досягнення у науці. – 2024. – № 7(7).
14. Стойчев В. Г. Автоматизована система управління виробничими процесами на авіаційно-ремонтному підприємстві : магіст. дис; Національний університет «Запорізька політехніка». – Запоріжжя, 2023. URL: <https://eir.zp.edu.ua/server/api/core/bitstreams/1cb450e8-f62c-4d20-861e-365d56b317e4/content> (дата звернення: 18.10.2024).
15. Дузенко А. Типи, форми та конкурентні переваги соціального підприємництва : наук. праця / А. Дузенко. – 2022.
16. Гандзюк М., Гандзюк Д., Шумік Б. Розробка методики оперативного планування роботи автомобілів на міжміських маршрутах // Сучасні технології в машинобудуванні та транспорті. – 2021. – № 2(17). – С. 47–61.
17. Скоробогатько С., Фесенко Г. Перспективи використання літаючих хмарних, граничних та туманних обчислень компонентами системи моніторингу потенційно небезпечних об'єктів // Системи управління, навігації та зв'язку. Збірник наукових праць. – 2022. – № 4(70). – С. 145–152. URL:

- <https://journals.nupp.edu.ua/sunz/article/view/2771/2168> (дата звернення: 19.10.2024).
- 18.Рибалченко А. Алгоритми рішення задачі оптимального розміщення даних в білінгових OLTP-системах на основі реалізації рангового підходу // Системи управління, навігації та зв'язку. Збірник наукових праць. – 2023. – № 2(72). – С. 135–141.
 - 19.Щур В. Ю. Проблематика балансування навантаження в SDN мережах великої розмірності та шляхи вирішення // The VI International Scientific and Practical Conference «Modern ways of solving the problems of science in the world», February 13–15, Warsaw, Poland. – 2023. – 445 p.
 - 20.Сахно Д. І. Планування логістичних робіт і послуг суб'єкта авіаційного ланцюга постачання : наук. праця / Д. І. Сахно. – 2024.
 - 21.Шимко В. Організація раціональної взаємодії видів транспорту в системі інтермодальних перевезень : дис. ... канд. техн. наук; Національний авіаційний університет. – Київ, 2023.
 - 22.Широкоступ В. М. Особливості застосування технологій логістики у сфері спортивних послуг : наук. праця / В. М. Широкоступ. – 2022. URL: <https://reposit.uni-sport.edu.ua/xmlui/bitstream/handle/787878787/4154/%d0%a8%d0%b8%d1%80%d0%be%d0%ba%d0%be%d1%81%d1%82%d1%83%d0%bf.pdf?sequence=1&isAllowed=y> (дата звернення: 20.10.2024).
 - 23.Мельник В. В. Формування інформаційної системи для забезпечення ефективного управління мультимодальним транспортом : наук. праця / В. В. Мельник. – 2021.
 - 24.Альошинський Є. С. Основи формування процесу міжнародних вантажних залізничних перевезень : наук. праця / Є. С. Альошинський. – 2009.
 - 25.Павленко О. В., Калініченко О. П., Нефьодов В. М. Методика формування ресурсозберігаючої технології доставки вантажів транспортно-логістичним центром // Комунальне господарство міст. Серія: Технічні науки та архітектура. – 2018. – № 142. – С. 96–102.
 - 26.Пилипенко І., Кайлюк Є. Критерії оцінки якості послуг, які надаються ліфтовими організаціями мешканцям житлових будинків // Економічний дискурс. – 2019. – № 2. – С. 113–120.
 - 27.Шибасєв О. Г. Організація ефективного функціонування пасажирського флоту в різних умовах експлуатації : дис. ... канд. техн. наук; Одеський національний морський університет. – Одеса, 2021.
 - 28.Грицук І. В., Ткаченко О. А., Маркович В. О. Особливості формування інформаційної системи моніторингу та управління дизельною

- електростанцією з реалізацією можливостей прогнозу її технічного стану // MPP&O-2022 : матеріали конф. – 2022. – С. 180.
29. Розробки ЕМТА. Мультимедійного навчального видання з дисципліни «Теорія кольору» // Поліграфічні, мультимедійні та web-технології : тези доп. IV Міжнар. наук.-техн. конф. (14-17 травня 2019, м. Харків) / редкол.: В. П. Ткаченко, І. Б. Чеботарьова, О. В. Вовк та ін. – Харків : Друкарня Мадрид, 2019. – Т. 1. – 244 с.
 30. Song L., Wang Q., Zhang J. Feasibility research on the application of self-piercing riveted connection in cold-formed steel structures // *Journal of Constructional Steel Research*. – 2020. – Vol. 168. – Article 105957.
 31. Islam M. D., Olabi A. G., Hashmi M. S. J. Feasibility of multi-layered tubular components forming by hydroforming and finite element simulation // *Journal of Materials Processing Technology*. – 2006. – Vol. 174, № 1–3. – P. 394–398.
 32. Sidorova E. The main factors and conditions determining the feasibility of production of high-tech products based on the potential of applied research organizations // *International Multidisciplinary Scientific GeoConference: SGEM*. – 2019. – Vol. 19, № 5.3. – P. 841–847.
 33. Giuffrida M., Mangiaracina R., Burki U. Cloud-based booking platforms in warehouse operations // *Sustainability*. – 2021. – Vol. 13, № 20. – Article 11547.
 34. PyCharm (JetBrains) – завантаження для Windows URL – Режим доступу: <https://www.jetbrains.com/pycharm/download/?section=windows> (дата звернення: 16.11.2024).
 35. Специфікація JSON (JSON.org) URL. – Режим доступу: <https://www.json.org/json-en.html> (дата звернення: 08.10.2024). Dargenavičius L., Kubeldzis E. Analysis and development of cloud-based system for logistics chain management : master's thesis / Kauno technologijos universitetas. – 2015.
 36. Čaušević S., Čolaković A., Hasković A. The model of transport monitoring application based on Internet of Things // *Proceedings of the International Scientific Conference on Science and Traffic Development (ZIRP)*. – 2018.
 37. Dzemydienė D., Burinskienė A., Čižiūnienė K. An approach of integration of contextual data in e-service system for management of multimodal cargo transportation // *Sustainability*. – 2024. – Vol. 16, № 18. – Article 7893.
 38. Tutorial – NetworkX 3.4.2 documentation. *Google Translate*. URL: https://networkx-org.translate.google/documentation/stable/tutorial.html?_x_tr_sl=en&_x_tr_tl=uk&_x_tr_hl=uk&_x_tr_pto=sc (дата звернення: 02.12.2024).

ДОДАТОК А. ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

Презентація



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ

КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Магістерська робота

Методика організації вантажопасажирських перевезень на основі
хмарних логістичних технологій

Виконав: студент групи ПДМ-61 Агапков Андрій Юрійович

Керівник: к.п.н., доц., доцент кафедри ІІЗ Шевченко Світлана Миколаївна

Київ - 2025

МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

Мета роботи: підвищення ефективності, прозорості та безпеки логістичних процесів вантажопасажирських перевезень за допомогою розробленої методики з використанням хмарних технологій.

Об'єкт дослідження: процес організації вантажопасажирських перевезень.

Предмет дослідження: методика організації вантажопасажирських перевезень на основі хмарних логістичних технологій.

НАЯВНІ РІШЕННЯ

Характеристика	AWS	Microsoft Azure	Google Cloud Services	De Novo	UCloud
Доступність без Інтернету	-	-	-	+	+
Затримка в передачі даних	Від 50мс	Від 35мс	Від 35мс	Від 5мс	Від 25мс
Шифрування	В спокої (AES-256) і при передачі (TLS 1.2+)	вбудоване шифрування даних у SQL Server, Azure Disk Encryption.	В спокої (AES-256) і під час передачі (TLS 1.3).	У спокої AES-256, під час передачі TLS 1.2+, ДСТУ ГОСТ 28147-2009	За міжнародними стандартами.
Аналітичні інструменти	Сильні	Сильні	Дуже сильні	Базові	Середні
Відповідність GDPR	+	+	+	+	+
Знаходження оптимального маршруту в режимі реального часу	+	+	+	-	-
Інструменти IoT	+	+	+	-	+

ЗАВДАННЯ

1. Дослідити наявні методи організації вантажопасажирських перевезень на основі хмарних технологій для малого та середнього бізнесу.
2. Здійснити порівняння наявних рішень.
3. Розробити та обґрунтувати методику організації вантажопасажирських перевезень на основі хмарних логістичних технологій.
4. Обґрунтувати вибір програмних засобів для розробки методики.
5. Розробити програмне забезпечення, що реалізує запропоновану методику.
6. Здійснити тестування даного продукту.
7. Провести оцінку ефективності впровадження хмарних технологій на основі отриманих даних

МЕТОДИ ДОСЛІДЖЕННЯ

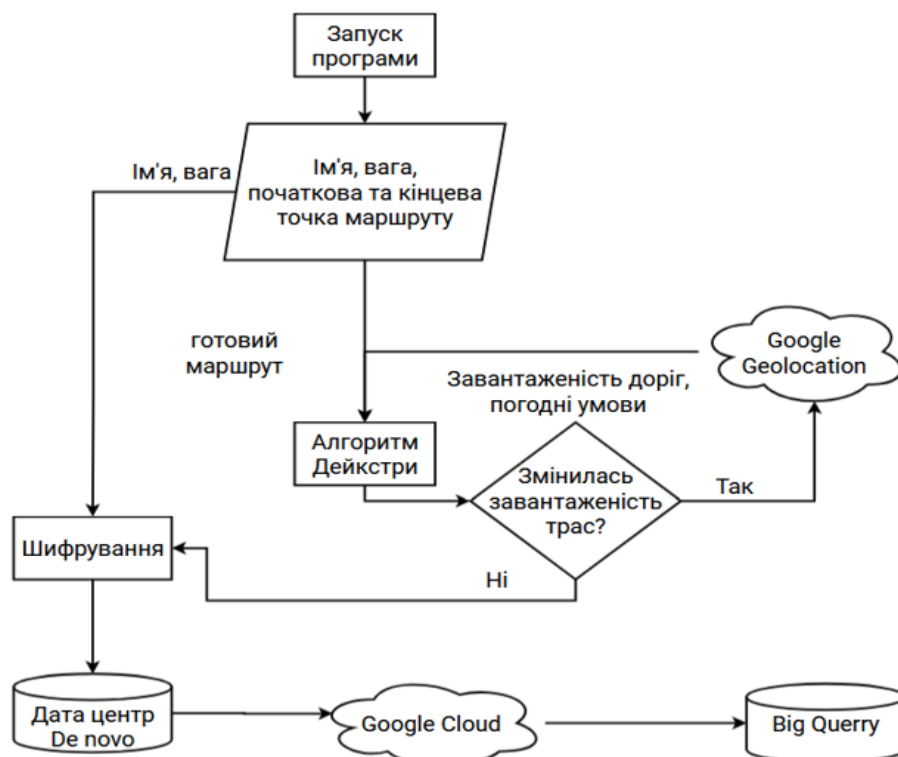
1. Системний підхід
2. Аналіз та порівняння
3. Хмарні технології
4. Оптимізаційні методи на графах

Методика організації вантажопасажирського перевезення на основі хмарних логістичних технологій

1. Створення замовлення в локальному додатку.
2. Передача даних на хмару.
3. Пошук найкоротшого шляху з урахуванням погодних умов та завантаженості доріг.
4. Виведення отриманих даних у заданому форматі.
5. Відстеження поточного статусу замовлення до моменту його завершення.
6. Коригування маршруту через зміни погодних умов та завантаженість доріг. Якщо змінилась завантаженість прилеглих маршрутів, повернутись до пункту 3.
7. Оновлення поточного статусу.

5

Алгоритм методики організації вантажопасажирських перевезень



6

ВИКОРИСТАНІ ЗАСОБИ







7

Використання BigQuery

Filter Enter property name or value			
☐	Field name	Type	Mode
☐	ID	INTEGER	NULLABLE
☐	Name	STRING	REQUIRED
☐	Start	STRING	REQUIRED
☐	End	STRING	REQUIRED
☐	Distance	FLOAT	REQUIRED
☐	Fuel	FLOAT	REQUIRED
☐	Fuel per 100km	FLOAT	REQUIRED
☐	Baggage	FLOAT	REQUIRED
☐	Status	STRING	NULLABLE

8

Вигляд розробленого додатку

Маршрутний калькулятор

Ім'я:

Звідки:

Куди:

Відстань (км):

Витрати пального (л/100 км):

Вага багажу (кг):

[Розрахувати маршрут](#)

Ім'я	Звідки	Куди	Відстань (км)	Витрати пального (л/100 км)	Загальне паливо (л)
Андрій	Харьков	Одеса	675.0	15.0	102.5
Юра	Київ	Львів	540.0	5.0	28.25
Андрій	Донецьк	Луцьк	850.0	40.0	400.0
Мілана	Кременчук	Львів	850.0	15.0	202.5
Олексій	Київ	Черкаси	200.0	12.0	30.0

9

Інформація про замовлення

Розрахунок маршруту

i

Розрахунок витрат пального:
 Ім'я: Олексій
 Відправлення: Київ
 Призначення: Черкаси
 Відстань: 200.0 км
 Витрати пального: 12.0 л/100 км
 Вага багажу: 120.0 кг
 Корекція на погоду: 0 %
 Загальна кількість пального: 30.00 л

[OK](#)

10

Аналіз ефективності розробленої методики

Характеристика	AWS	Google Cloud	Microsoft Azure	De Novo	UCloud	Logistic Manager
Середній час відгуку	55мс	40мс	45мс	15мс	40мс	12мс
Місячні грошові витрати	550\$	500\$	600\$	400\$	450\$	250\$
Врахування погодних умов	+	+	+	-	-	+
Аналітика перевезень	Сильна	Дуже сильна	Сильна	Середня	Слабка	Дуже сильна
Пасажи́рські перевезення	+	+	+	-	-	+
Середній час на створення 1 замовлення	5с	5с	5с	4,8с	6с	4с

11

ВИСНОВКИ

1. Проведено дослідження щодо впровадження хмарних технологій для організації вантажопасажи́рських перевезень, яке продемонструвало їхню ефективність у підвищенні продуктивності та зниженні витрат у логістичній діяльності. Використання хмарних обчислень дозволило створити централізовану платформу для обробки, зберігання та управління інформацією, що позитивно вплинуло на всі етапи логістичного процесу, починаючи від планування маршрутів і закінчуючи контролем за виконанням замовлень.
2. Проведено аналіз та порівняльну характеристику існуючих хмарних технологій, що використовуються для транспортних логістичних операцій.
3. Розроблено методику організації вантажопасажи́рських перевезень для малого і середнього бізнесу на основі хмарних логістичних технологій.
4. Проведено тестування методики. Результати підтвердили її перевагу над наявними рішеннями.
5. Проведено апробацію методики та виконано аналіз ефективності наявних транспортних рішень на основі даних логістичної компанії.

12

ПУБЛІКАЦІЇ ТА АПРОБАЦІЯ РОБОТИ

Статті:

1. Агашков А.Ю. Шевченко С.М. Шляхи підвищення ефективності транспортних операцій за допомогою хмарних логістичних рішень, *Телекомунікаційні та інформаційні технології*, №3, 2025р, Подано до друку.

Тези доповідей:

1. Агашков А.Ю., Шевченко С.М.// Хмарні логістичні технології у вантажопасажирських перевезеннях// VII Міжнародна науково-практична конференція молодих вчених та студентів «Інженерія програмного забезпечення і передові інформаційні технології», Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського» (SoftTech-2024), м. Київ (С. 9 - 10).

2. Агашков А.Ю., Шевченко С.М.// Використання Amazon Web Services у логістиці вантажопасажирських перевезень// Всеукраїнська науково-технічна конференція «Виклики та рішення в програмній інженерії», м. Київ – К. ДУІКТ, 2024 (С. 17-22).

ДОДАТОК Б. ЛІСТИНГ ПРОГРАМНИХ МОДУЛІВ

ю

```

import tkinter as tk
from tkinter import ttk
from tkinter import messagebox
import json
import os

from google.cloud import bigquery
import googlemaps

client = bigquery.Client.from_service_account_json('C:\Users\agash\Desktop\магістерська\Logistics Manager\routes.json')

gmaps = googlemaps.Client(key='AIzaSyD52C6eZ1RC55v6T_IvR7zDbRMr_T_Y7eM')

# Шлях до файлу для збереження маршрутів
ROUTES_FILE = "routes.json"

# Перевірка наявності файлу і його створення, якщо його немає
if not os.path.exists(ROUTES_FILE):
    with open(ROUTES_FILE, 'w') as file:
        json.dump([], file)

# Функція для завантаження маршрутів з JSON
def load_routes():
    with open(ROUTES_FILE, 'r', encoding='utf-8') as file:
        return json.load(file)

# Отримання координат через ipinfo.io
def get_coordinates(city_name):
    geocode_result = gmaps.geocode(city_name)
    location = geocode_result[0]['geometry']['location']
    latitude = location['lat']
    longitude = location['lng']
    return latitude, longitude

# Отримання поточної геолокації
def get_geolocation_google():
    api_key = 'AIzaSyBZ_ANOhEGWSQcSCFSLp4BvXn1f-xxxx'
    url = f'https://www.googleapis.com/geolocation/v1/geolocate?key={api_key}'
    data = {'considerIp': True}
    response = requests.post(url, json=data)
    location = response.json()['location']
    latitude = location['lat']
    longitude = location['lng']
    return latitude, longitude

# Завантаження доріг (отримання даних через Google Maps Directions API)
def get_traffic_info(start_location, end_location):
    directions_result = gmaps.directions(start_location, end_location, departure_time='now')
    steps = directions_result[0]['legs'][0]['steps']
    traffic_data = [(step['start_location'], step['end_location'], step['duration_in_traffic']['value']) for step in steps]
    return traffic_data

# алгоритм пошуку найкоротшої відстані
def dijkstra(traffic_data, start_location, end_location):
    G = nx.DiGraph()
    for start, end, weight in traffic_data:
        G.add_edge((start['lat'], start['lng']), (end['lat'], end['lng']), weight=weight)
    start = (start_location['lat'], start_location['lng'])
    end = (end_location['lat'], end_location['lng'])
    shortest_path = nx.dijkstra_path(G, start, end, weight='weight')
    return shortest_path

# Функція для збереження маршруту в JSON
def save_route(route_data):
    routes = load_routes()
    routes.append(route_data)
    with open(ROUTES_FILE, 'w', encoding='utf-8') as file:

```

```

json.dump(routes, file, ensure_ascii=False, indent=4)

def get_weather_correction(latitude, longitude):
    try:
        api_key = 'API_KEY'

        url = f"http://api.openweathermap.org/data/2.5/weather?lat={latitude}&lon={longitude}&appid={api_key}&units=metric"

        response = requests.get(url)
        weather_data = response.json()

        temperature = weather_data['main']['temp']
        humidity = weather_data['main']['humidity']
        wind_speed = weather_data['wind']['speed']

        # Проста логіка корекції
        correction = 0

        if temperature < 0:
            correction += 15 # Збільшення на 15% в холодну погоду

        if humidity > 80:
            correction += 10 # Збільшення на 10% при високій вологості

        if wind_speed > 15:
            correction += 5 # Збільшення на 5% при сильному вітрі

        return correction

    except Exception as e:
        messagebox.showerror("Помилка", "Всі поля повинні бути заповнені!")

    return

latitude, longitude = get_geolocation_google()

start_location = {'lat': latitude, 'lng': longitude} # Отримання координат кінцевого пункту

for point in shortest_path: print(f"Широта: {point[0]}, Довгота: {point[1]}")

weather_correction = get_weather_correction()

total_fuel = calculate_fuel_cost(distance_km, fuel_consumption, baggage_weight, weather_correction)

route_data = { "name": name, "pickup": pickup_city, "destination": destination_city, "weight": baggage_weight, "distance": distance_km,

```

```

except Exception as e:
    messagebox.showerror("Помилка", f"Не вдалося отримати дані про погоду: {e}")

    return 0

# Функція для розрахунку витрат пального
def calculate_fuel_cost(distance, fuel_consumption, baggage_weight, weather_correction):
    base_fuel = (distance / 100) * fuel_consumption

    fuel_for_baggage = fuel_consumption * (baggage_weight / 5000) # похибка на завантаженість автомобіля

    total_fuel = base_fuel + fuel_for_baggage

    total_fuel *= (1 + weather_correction / 100)

    return total_fuel

# Функція для розрахунку маршруту та збереження даних
def show_route_calculation(name_entry, pickup_entry, destination_entry, distance_entry, fuel_consumption_entry, baggage_weight_entry, table):
    try:
        name = name_entry.get()
        pickup_city = pickup_entry.get()
        destination_city = destination_entry.get()
        distance_km = float(distance_entry.get())
        fuel_consumption = float(fuel_consumption_entry.get())
        baggage_weight = float(baggage_weight_entry.get())

        if not name or not pickup_city or not destination_city:
            return

        end_latitude, end_longitude = get_coordinates(destination_city)

        end_location = {'lat': end_latitude, 'lng': end_longitude}
        # Отримання даних про завантаженість доріг

        traffic_data = get_traffic_info(start_location, end_location) # Знаходження найкоротшого маршруту

        shortest_path = dijkstra(traffic_data, start_location, end_location)

        "fuel_consumption": fuel_consumption,
        "weather_correction": weather_correction, "total_fuel": total_fuel }

        save_route(route_data)

        update_table(table)

        result_text = f"" Розрахунок витрат пального: Ім'я: {route_data['name']} Відправлення: {route_data['pickup']} Призначення: {route_data['destination']}

```