

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ  
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ  
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається  
В.о. завідувача кафедри КНІ  
\_\_\_\_\_ Рязанцев О.І.  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

**МАГІСТЕРСЬКА РОБОТА**

НА ТЕМУ:

АЛГОРИТМИ ТА ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ СИСТЕМИ АВТОМАТИЗАЦІЇ  
ТРАНСПОРТНОЇ ЛОГІСТИКИ

Освітній ступінь “Магістр”  
Спеціальність 123 - “Комп’ютерна інженерія”

Науковий керівник роботи:

\_\_\_\_\_

(підпис)

Деркач М.В.

(ініціали, прізвище)

Консультант з охорони праці та  
безпеки в надзвичайних ситуаціях:

\_\_\_\_\_

(підпис)

Критська Я.О.

(ініціали, прізвище)

Здобувач вищої освіти:

\_\_\_\_\_

(підпис)

Лисак В.В.

(ініціали, прізвище)

Група:

КІ-19Дм

Северодонецьк 2021

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки  
Кафедра Комп'ютерних наук та інженерії  
Освітній ступінь магістр  
Спеціальність 123 «Комп'ютерна інженерія»  
(шифр і назва)

**ЗАТВЕРДЖУЮ:**

Т.в.о. завідувача кафедри КНІ  
Кардашук В.С.  
«    » 20 р.

**З А В Д А Н Н Я  
НА МАГІСТЕРСЬКУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ**

Лисаку Владиславу Вікторовичу  
(прізвище, ім'я, по батькові)

1. Тема роботи Алгоритми та інформаційна технологія системи автоматизації  
транспортної логістики

керівник проекту (роботи) к.т.н., доц. Деркач Марина Володимирівна  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затвержені наказом вищого навчального закладу від "05" 10 2020 р. № 140/15.15

2. Строк подання здобувачем вищої освіти роботи 11.01.2021 року

3. Вихідні дані до роботи матеріали науково-дослідної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно  
розробити) Аналіз функціональних можливостей автоматизації транспортної логістики.  
Дослідження алгоритмів транспортної логістики. Система автоматизації транспортної  
логістики. Охорона праці

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

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці та безпека в надзвичайних ситуаціях	ст.викл. Критська Я.О.		

7. Дата видачі завдання 05.10.2020

Керівник \_\_\_\_\_  
(підпис)

Завдання прийняв до виконання \_\_\_\_\_  
(підпис)

**КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту ( роботи )	Примітка
1	Аналіз літературних джерел з теми ДП	05.10.2020-17.10.2020	
2	Технічне завдання на розробку	18.10.2020-21.10.2020	
3	Розробка розділу охорони праці	22.10.2020-25.10.2020	
4	Аналіз функціональних можливостей автоматизації транспортної логістики	26.10.2020-02.11.2020	
5	Дослідження алгоритмів транспортної логістики	03.11.2020-10.11.2020	
6	Розробка системи автоматизації транспортної логістики	11.11.2020-29.12.2020	
7	Оформлення пояснювальної записки	30.12.2020-04.01.2021	
8	Оформлення автореферату	05.01.2021-07.01.2021	
9	Оформлення презентації	08.01.2021-11.01.2021	

Здобувач вищої освіти \_\_\_\_\_  
( підпис )

Лисак В.В.  
(прізвище та ініціали)

Науковий керівник \_\_\_\_\_  
( підпис )

Деркач М.В.  
(прізвище та ініціали)

## АНОТАЦІЯ

**Лисак В.В. Алгоритми та інформаційна технологія системи автоматизації транспортної логістики.**

Робота присвячена вирішенню завдання пошуку місця для паркування, оскільки з підвищенням рівня урбанізації це стає все більш складним завданням, особливо в великих містах. В роботі розглянуто алгоритми транспортної логістики, а саме алгоритм пошуку максимально можливої пропускної здатності мережі, зокрема алгоритм Форда-Фалкерсона, та алгоритми пошуку найкоротшої відстані, такі як алгоритм Дейкстри, алгоритм Флойда — Уоршелла та алгоритм Суурбалле. Проведено тестування алгоритмів, для цього попередньо визначено транспортну мережу існуючих центрів паркування міста Сєвєродонецьк. Оскільки в ході тестування визначено, що час виконання алгоритму Дейкстри швидше для всіх пошукових відстаней, то саме цей алгоритм інтегрується з алгоритмом пошуку максимально можливої пропускної здатності мережі. Це дає оптимальне рішення щодо шляху до найближчого парковочного центру, мінуючи найбільш навантажені ділянки дороги транспортної мережі на прикладі міста Сєвєродонецьк і розширює функціонал сервісу. Система автоматизації призначена для отримання інформації про кількість вільних місць для паркування в режимі реального часу; про час, коли доступні місця для паркування, щоб користувачі могли зарезервувати місце на вибраній автопарковці. Розробка спрямована на підвищення якості наданих послуг у сфері автотранспорту.

**Ключові слова:** система автоматизації, алгоритми транспортної логістики, інтеграція, відстань, час.

## ANNOTATION

**Lysak V.V. Algorithms and information technology of transport logistics automation system.**

The work is devoted to solving the problem of finding a place for parking, because with the increasing level of urbanization it becomes more and more difficult, especially in large cities. The paper considers algorithms of transport logistics, namely the algorithm for finding the maximum possible network bandwidth, in particular the Ford-Falkerson algorithm, and the shortest distance search algorithms, such as the Dijkstra algorithm, the Floyd-Worshell algorithm and the Suurballe algorithm. Testing of algorithms is carried out, for this purpose the transport network of the existing parking centers of the Severodonetsk city is preliminary defined. Since the test determined that the execution time of the Dijkstra algorithm is faster for all search distances, it is this algorithm that integrates with the algorithm for finding the maximum possible network bandwidth. This provides an optimal solution for the path to the nearest parking center, bypassing the busiest sections of the road transport network on the Severodonetsk city example and expands the functionality of the service. The automation system is designed to obtain information about the number of free parking spaces in real time; about the time when parking spaces are available so that users can reserve a place in the selected car park. The development is aimed at improving the quality of services provided in the field of motor transport.

**Keywords:** automation system, transport logistics algorithms, integration, distance, time.

## ЗМІСТ

Вступ.....	6
Розділ 1. Аналіз функціональних можливостей автоматизації транспортної логістики.....	9
1.1 Аналіз впровадження інформаційних технологій в транспортний сектор країни.....	9
1.2 Переваги, недоліки транспортної логістики, архітектура системи.....	19
1.3 Огляд існуючих рішень щодо автоматизації транспортної логістики.....	23
1.4 Постановка задачі розробки.....	26
Розділ 2. Дослідження алгоритмів транспортної логістики.....	28
2.1 Алгоритми пошуку найкоротшого шляху.....	28
2.1.1 Алгоритм Дейкстри.....	30
2.1.2 Алгоритм Флойда – Уоршелла.....	31
2.1.3 Алгоритм Суурбалле.....	32
2.2 Тестування алгоритмів пошуку найкоротшого шляху.....	32
2.3 Алгоритм пошуку максимального потоку.....	36
2.4 Інтеграція алгоритмів Дейкстри та Форда-Фалкерсона.....	38
Розділ 3. Система автоматизації транспортної логістики.....	40
3.1 Функціонал системи автоматизації.....	40
3.2 Реалізація алгоритму Форда-Фалкерсона в системі.....	43
3.3 Інтеграція алгоритмів пошуку найкоротшої відстані та максимально можливої пропускної здатності транспортної мережі міста.....	44
3.4 Схема бази даних системи автоматизації транспортної логістики.....	46
Розділ 4. Охорона праці.....	50
4.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу.....	50
4.2 Електробезпека.....	51
4.3 Освітлення.....	53
4.4 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій.....	56

	5
4.5 Охорона навколишнього природного середовища. Загальні дані з охорони навколишнього природного середовища.....	61
4.6 Висновки до розділу 4.....	62
ВИСНОВКИ.....	63
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	64
Додаток А Слайди презентації.....	68
Додаток Б Алгоритм Форда-Фалкерсона.....	77
Додаток В Алгоритм Дейкстри.....	79
Додаток Г Міграції бази даних.....	81

## ВСТУП

Актуальність роботи обумовлена наявністю проблем, пов'язаних з неефективним використанням паркувальних місць, пошуком місця для паркування та нерівномірним розподілом автомобільного трафіку, оскільки з підвищенням рівня урбанізації це стає все більш складним завданням, особливо в великих містах, бо призводить до заторів транспортного руху. Задля вирішення цієї проблеми, необхідно реалізувати пошук найкоротшого шляху, максимального потоку, тобто оптимальних маршрутів. Такий пошук, а також оцінка пройденого шляху або відстані, що залишилось подолати до контрольної точки, отримується завдяки алгоритмам транспортної логістики. Окрім використання алгоритмів транспортної логістики для вирішення цієї проблеми останні десятиріччя ведуться активні розробки та впровадження технології IoT, що включають комп'ютерні технології, технології зв'язку, системи позиціонування та автоматичної навігації та локалізації, різноманітні засоби управління транспортними системами, архітектура яких створює основу ефективного впровадження широкого спектра послуг для користувача. Такі технології широко застосовуються в управлінні дорожнім рухом, завдяки тому, що містять засоби для інформування учасників дорожнього руху про поточний стан дорожньо-транспортної ситуації, а саме системи контролю та оптимізації маршрутів, засоби, технології управління при надзвичайних ситуаціях, системи автоматичного збору інформації про кількість пасажирів на борту та зупинках, визначення місцеперебування транспортного засобу, ідентифікація автомобіля, інтелектуальні парковки, тощо. Реалізація подібних сервісів спрямована на підвищення якості наданих послуг у сфері автотранспорту.

**Зв'язок роботи з науковими програмами, планами, темами.** Дослідження, результати яких викладені в магістерській роботі, проводилися у Східноукраїнському національному університеті імені Володимира Даля

відповідно до державних програм і планів НДР, а також міжнародного проекту:

– Проект Європейського Союзу ERASMUS+ ALIOT 573818-EPP-1-2016-1-UK-EPPKA2-CBHE-JP «Internet of Things: Emerging Curriculum for Industry and Human Applications» (2016-2019 pp.);

– НДР «Наукові основи, методи і засоби розгортання інтелектуальних транспортних систем для моніторингу міського транспорту та підтримки пасажирів» (Східноукраїнський національний університет імені Володимира Даля, ДР 0117U005537, 2017-2020 pp.).

**Методи дослідження.** Проведені в роботі дослідження базуються на методах теорії управління та навігації систем, а також теорії графів та баз даних.

**Наукова новизна** магістерської роботи полягає у наступному: удосконалено інформаційну технологію системи автоматизації транспортної логістики за рахунок інтеграції алгоритмів Дейкстри та Форда-Фалкерсона.

**Практичне значення одержаних результатів** полягає в тому, що основні наукові положення магістерської роботи реалізовано у вигляді розрахункових алгоритмів, які утворюють прикладну інформаційну технологію для системи автоматизації транспортної логістики на основі геоінформаційних даних.

**Апробація** результатів наукових досліджень проводилася на всеукраїнських і міжнародних науково-практичних конференціях «ІТ-Ідея» (Сєверодонецьк, 2018); «10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)» (Metz, France, 2019); «Advanced Technologies in Research and Education» (Сєверодонецьк-Каїр, 2019, 2020).

**Публікації.** За темою роботи з викладенням її основних результатів опубліковано 6 наукових праць, серед яких 1 англomовна праця включена до бази даних *Scopus*, 2 статті у науковому фаховому виданні України; 3 тез в збірниках матеріалів науково-практичних конференцій.



**Структура і обсяг роботи.** Магістерська робота складається зі вступу, 4 розділів, висновків, переліку посилань до розділів з 30 найменувань, 4 додатків. Загальний обсяг роботи складає 82 сторінки. Магістерська робота містить 14 рисунків та 2 таблиць.

# 1 АНАЛІЗ ФУНКЦІОНАЛЬНИХ МОЖЛИВОСТЕЙ АВТОМАТИЗАЦІЇ ТРАНСПОРТНОЇ ЛОГІСТИКИ

## 1.1 Аналіз впровадження інформаційних технологій в транспортний сектор країни

Структура транспортної системи будь-якого міста визначається видами транспорту, що використовуються та можуть бути схарактеризовані транспортними мережами, що її складають. Розрізняють в основному три види міського транспорту: пасажирський, вантажний та спеціальний (санітарні та пожежні машини, транспорт для прибирання вулиць тощо). Також за характером оточуючого середовища, що використовується для руху, пасажирський транспорт поділяють на наземний, водний, повітряний, а також транспорт, який використовує штучно створене середовище (найчастіше підземний). Пасажирський транспорт є найважливішою складовою соціальної інфраструктури великого міста. Стійка і ефективна робота транспорту є необхідною умовою стабілізації та підйому економіки, поліпшення умов і рівня життя населення. Сучасний пасажирський транспорт забезпечує перевезення людей, їх ручної поклажі і багажу в різних видах сполучення. Перевезення людей можуть здійснюватися, як на професійній основі, так і самостійно громадянами в особистих (побутових) цілях. Такі перевезення можуть бути комерційними та некомерційними. Комерційні перевезення виконуються перевізником з метою отримання економічного результату (вигоди) і поділяються на перевезення транспортом загального користування та технологічні перевезення. Некомерційні перевезення виконуються громадянами з метою задоволення особистих (побутових) потреб, а також організаціями в інтересах державної або муніципальної служби (перевезення військовослужбовців, поліцейських, чиновників і т.д.).

Якість обслуговування пасажирів та соціальна користь транспортних послуг є важливою соціальною і економічною проблемою для кожного міста, так як необхідно задовольнити вимоги всіх груп населення. Якістю обслуговування пасажирів, зокрема раціональним плануванням мереж маршрутів, зупиночних станцій та місць пересадки з одного виду транспорту на інший, транспортуванням та розподілом вантажів займаються міська логістика та районне планування як галузь суспільної географії.

Інтенсивність функціонування транспорту є важливою складовою життя міста, що впливає на розвиток і добробут населення. Ефективність транспортної системи забезпечує сталий економічний та соціальний розвиток суспільства. Успішна реалізація питань підвищення мобільності населення можлива тільки при умові стійкого та ефективного функціонування транспортної галузі як особливої складової виробничої структури країни, що суттєво підвищує рівень життя населення.

Україна має розвинену, розгалужену, потужну транспортну систему, яка при відповідному її розвитку повинна задовольнити в повному обсязі усі вимоги населення держави. Транспортні лінії (в основному залізничного та автомобільного транспорту) пронизують практично всю територію держави. Транспорт України представляє собою систему транспортних комунікацій, яка об'єднує всі види транспорту. Розміщення їх регіонами держави, а також їх структура повністю відповідають вимогам щодо здійснення пасажирських перевезень у внутрішньому і міжнародному сполученнях.

Однак з метою підвищення якості транспортних послуг всі складові транспортної системи потребують істотного вдосконалення та модернізації. Крім того, транспортна система відповідає вимогам стратегічного розвитку держави, пов'язуючи в єдине ціле територію країни, забезпечуючи необхідну мобільність населення, реалізацію транзитних можливостей держави, безпеку і стабільність пасажирських перевезень.

Загальна довжина шляхів сполучення за різними видами транспорту становить 200299,1 км, а саме: залізничний - 22,0 тис. Км, автомобільний - 169,4 тис. км; внутрішні річкові судноплавні шляхи - 2,3 тис. км; тролейбусні лінії - 4,4 тис. км; трамвайні лінії - 2,1 тис. км, метрополітенівським лінії - 99,1 км. Щільність шляхів сполучення (в км шляхів на 1 тис. км<sup>2</sup> території) складає залізничних колій - 36; річкових судноплавних шляхів - 4; автомобільних доріг загального користування з твердим покриттям - 273. Інтенсивність перевезення пасажирів (млн. пас-км на 1 км довжини шляхів) становить за видами шляхів сполучення: залізничні колії - 2,35; річкові судноплавні шляхи - 0,03, автомобільні дороги з твердим покриттям - 0,1; тролейбусні лінії - 2, 4; трамвайні колії - 3,2; метрополітенівським шляху - 22,5.

Середня відстань перевезення одного пасажирів (в км) становить по окремих видах транспорту:

- залізничний - 112;
- морський - 6;
- річковий - 26;
- авіаційний - 1687;
- автомобільний (автобуси) - 13;
- тролейбусний - 6;
- трамвайний - 6;
- метрополітен - 8.

Внесок в загальні обсяги транспортної системи (в %) кожного виду транспорту становить:

- залізничний - 5,7%;
- автомобільний (автобуси) - 46,5%;
- трамвайний - 13,9%;
- тролейбусний – 23,1%;
- метрополітен - 10,6%,

– морський, річковий і авіаційний транспорт перевезли значно менше 1% пасажирів кожен.

По-іншому виглядає розподіл пасажирообігу (в млрд. пас/км) за видами транспорту:

- залізничний - 51,7;
- автомобільний (автобуси) - 47,5;
- трамвайний - 6,6;
- тролейбусний - 10,8;
- метрополітен - 6,4;
- морський - 0,1;
- річковий - 0,01;
- авіаційний - 5,5.

Наведені дані свідчать про такий розподіл пасажирообігу за видами транспорту: залізничний - 40,2%; автомобільний (автобуси) - 36,9%; трамвайний - 5,1%; тролейбусний - 8,4%; метрополітен - 5,0%; авіаційний - 4,3%, а морський і річковий транспорт - значно менше 1% кожен.

Існує два шляхи збільшення ефективності транспортного обслуговування населення та організації дорожнього руху: збільшення пропускної спроможності транспортної мережі та раціональне використання існуючої мережі. Перше рішення пов'язане з великими матеріальними витратами на реконструкцію транспортних вузлів і магістралей. Другий напрямок - раціоналізація використання існуючих транспортних систем і оптимізація процесів перерозподілу навантажень на транспортну мережу. Наведені дані вказують на наявність тенденції зростання обсягів пасажирських перевезень, вимагає проведення в життя ефективних заходів за рахунок впровадження новітніх інформаційних технологій щодо вдосконалення діяльності транспортно-дорожнього комплексу країни.

Інформаційні технології (ІТ) мають значне значення для транспортних систем, оскільки вони забезпечують доступ до інформації про поїздки, засоби планування, можливості спільного використання видів транспорту, роботи на відстані, порівняння вартості транспортного способу, здійснення оплати, підвищення безпеки та здоров'я, а також для спілкування з подорожами. За останнє десятиліття спостерігається значне зростання доступності транспортних ІТ, зокрема завдяки концепції Інтернету речей (ІоТ). Основна функціональність ІоТ полягає в тому, щоб забезпечується міжмашинний зв'язок за допомогою Інтернету та додатків.

Прогнозований облік використання ІоТ пристроїв (рис. 1.1):

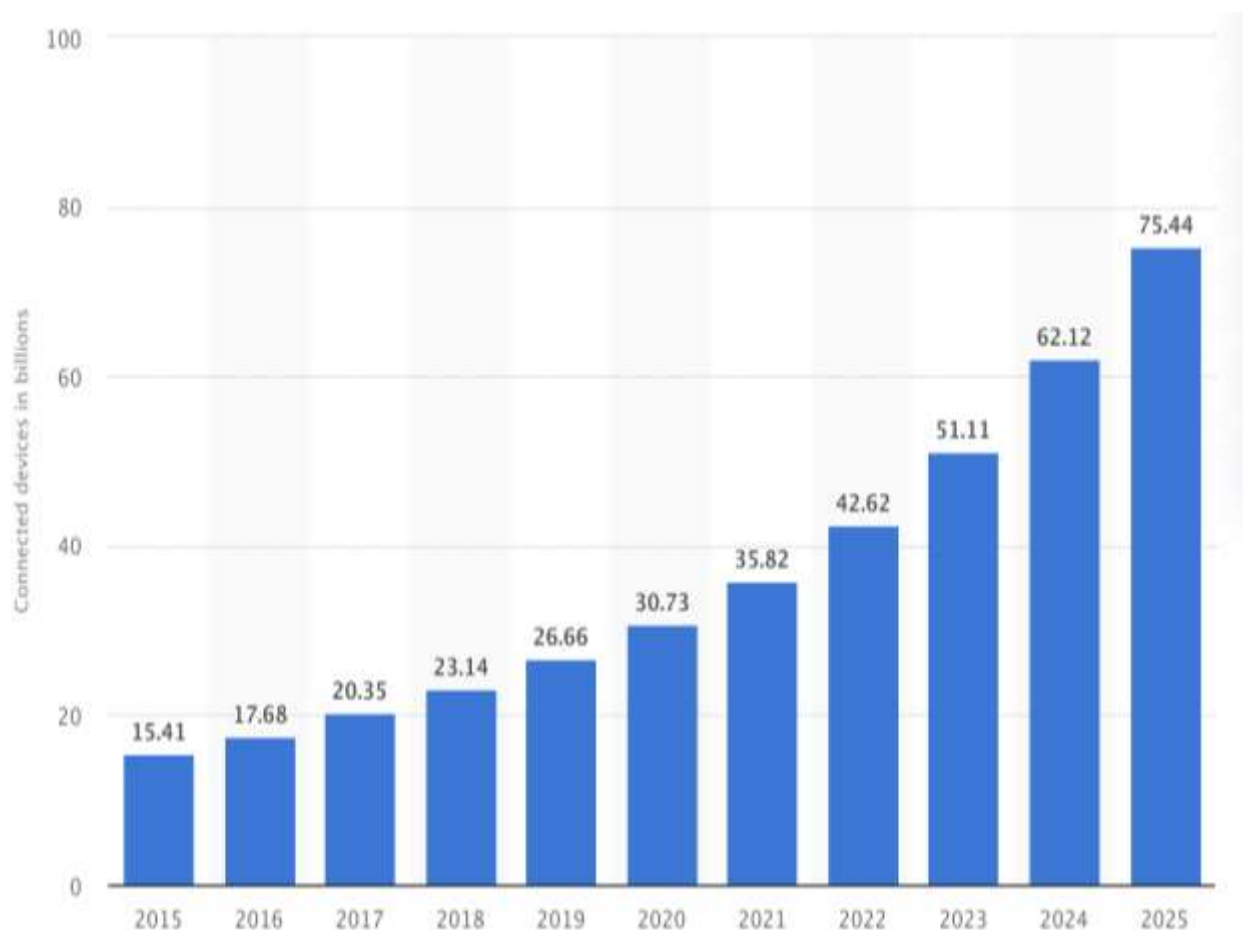


Рисунок 1.1 - Пристрої, підключені до Інтернету речей (ІоТ), встановлені в усьому світі з 2015 по 2025 рік (у мільярдах) (<https://www.statista.com/>)

Існує чимало доказів того, що ІТ глибоко змінили способи сприйняття та використання транспортних систем, а також мобільність; з далекосяжними наслідками для вибору виду транспорту та транспортного попиту.

Сервіси транспортної галузі охопили майже весь світ. Їх впровадження є дуже корисним за для розв'язання виникаючих проблем, завдяки сервісам, що надають можливості автоматичного планування маршрутів, моніторингу безпеки руху транспортних засобів, володіння інформацією про паркування, збору даних про транспортування. Різноманітні сервіси використовуються для багатьох транспортних задач, включаючи навігацію, місцеположення та місця для паркування транспортних засобів, збір даних про рух, інформацію про подорожі, планування маршрутів та інші.

Розумне паркування - одне з найбільш прийнятих та швидкозростаючих рішень Smart City у світі. Аеропорти, університети, торгові центри та міські гаражі - лише кілька суб'єктів господарювання, які почали усвідомлювати значні переваги технології автоматизованого паркування. Можливість підключення, аналізу та автоматизації даних, зібраних із пристроїв, що працюють та описуються як Інтернет речей, - це те, що робить можливим розумне паркування.

Розумне паркування передбачає використання недорогих датчиків, даних у режимі реального часу та додатків, що дозволяють користувачам контролювати доступні місця для паркування. Задля розуміння фактичного перебування автомобіля на парковочному майданчику є багато варіантів датчиків (рис. 1.2):

– Активні інфрачервоні датчики: Активні інфрачервоні датчики виявляють транспортні засоби, випромінюючи інфрачервону енергію та визначаючи кількість відбитої енергії. Використовуючи активні інфрачервоні датчики, можна вимірювати положення автомобіля. Однак недоліком датчика є

його чутливість до умов навколишнього середовища, таких як туман або обдування снігом, що впливає на роботу датчиків.

– Верхні радари можуть виявити присутність транспортного засобу за допомогою радіосигналів та лазерного відбиття світла. Система забезпечує точні результати в межах 7 метрів, але при більшій відстані та різких кутах відбиття її точність зменшується, отже, датчик живлення від мережі, закріплений на ліхтарній стійці, може надійно охоплювати лише 4-5 паркувальних місць.

– Датчики заземлення вважаються універсальним методом контролю за наповненістю стоянки. Бездротові датчики, що живляться від батареї, встановлені на кожному паркувальному місці та на них не впливають перешкоди прямої видимості. Близькість до виявленого об'єкта дозволяє датчикам заземлення забезпечувати точні результати за розумних витрат на обслуговування. В даний час наземні датчики вважаються найбільш гнучким та ефективним способом вирішення проблем паркування.



Рисунок 1.2 - Інтелектуальні системи паркування витрат та порівняння точності



Часто також для визначення перебування автомобіля використовують магнітометр, це датчик, який вимірює напруженість магнітного поля і допомагає зрозуміти орієнтацію об'єкта.

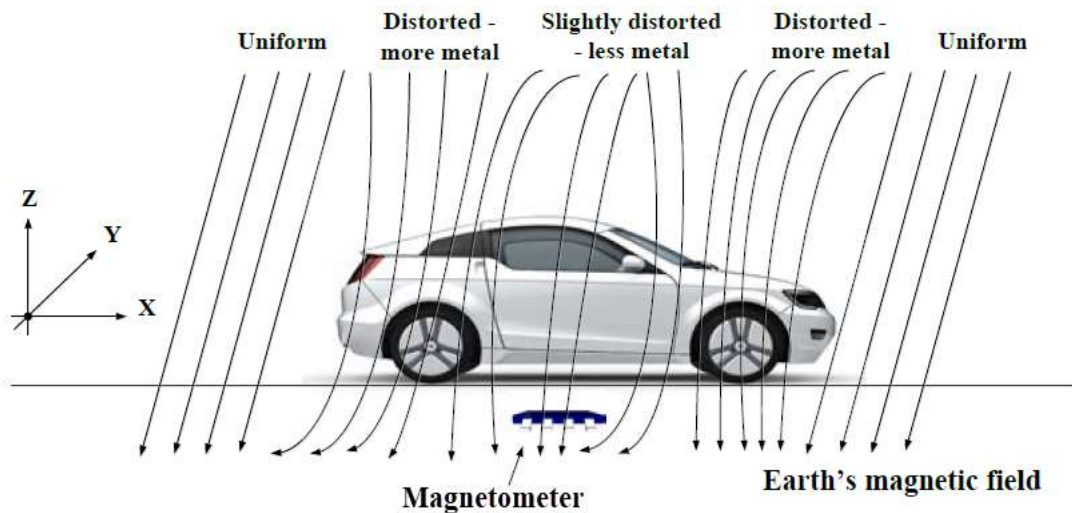


Рисунок 1.3 - Принципи виявлення транспортного засобу магнітним датчиком

Предмети, що містять залізо, як автомобілі, створюють спотворення на короткому діапазоні, яке можна виміряти. Величина цього спотворення залежить від типу чорного сплаву, відстані до датчика та розміру об'єкта (рис 1.3).

Метою визначення перебування автомобіля є автоматизація та зменшення часу, витраченого на ручний пошук оптимального паркінгу, місця та навіть ділянки. Деякі рішення включатимуть повний набір послуг, таких як онлайн-платежі, сповіщення про найближчі центри, про час паркування та навіть функції пошуку автомобілів для дуже великих партій. Рішення для паркування може мати велику користь як для користувача, так і для власника ділянки.

Існує безліч сервісів, які використовуються для багатьох транспортних завдань, включаючи навігацію, моніторинг безпеки дорожнього руху, інформацію про паркування, збір даних про трафік, інформацію про подорож в реальному часі, планування маршруту і транспортну логістику.

Щодо, оптимізації транспортних перевезень велике значення має правильно складений маршрут перевезення. Раціональні маршрути: оптимізують вантажопотоки в логістичних схемах; забезпечують завантаженість транспортного парку; знижують собівартість транспортування товарів.

Особливо важлива маршрутизація в автомобільних перевезеннях. Мобільність автотранспорту і пріоритет цього виду перевезень вимагає чіткої побудови завдання і точного виконання. При складанні маршруту потрібно враховувати все - від стану дорожнього покриття та наявності організованих стоянок автотранспорту, і до вимог по нормам проїзду мостів і переправ.

Транспортна логістика - це сукупність заходів по організації пасажиро- та товарообігу з однієї точки в іншу за оптимальним маршрутом.

Неефективне управління транспортною логістикою сприяє наступним проблемам:

1. Високі витрати на доставку. Зміст власного автопарку обходиться дорого, а вибрати оптимальний найманий транспорт складно.
2. Нераціональна завантаження транспорту - машини відправляються в рейс напівпорожніми.
3. Складно врахувати всі вимоги клієнтів: температурний режим, час доставки, нерівномірні обсяги замовлень та ін. В кожному маршруті багато деталей, які потрібно врахувати при плануванні.
4. Складені маршрути часто нелогічні і тому малоефективні - маршрутизація статична, одного разу складений маршрути не змінюються.
5. Не існує інструменту для комплексного управління: планування, контролю виконання планів і аналізу ефективності використання транспорту.
6. Великий вплив людського фактору на планування маршрутів і контроль виконання.

Для вирішення цих завдань використовуються логістичні програми (Transport Management System, TMS - системи управління транспортною логістикою). За даними компанії DSS, найбільша кількість компаній-замовників для автоматизації транспортної логістики вибирали систему Antor LogisticsMaster, а на другому місці за кількістю використання - система "БІТ-НОВА: Управління транспортною логістикою", на третьому - "1С-Парус: транспортна логістика та експедирування".

Найбільш оптимальне управління логістикою досягається шляхом інтеграції програми ANTOR LogisticsMaster з інформаційною системою моніторингу транспорту і мобільних об'єктів - ANTOR MonitorMaster. Це комплексне рішення легко інтегрується з обліковою системою підприємства, звідки надходять усі заявки на доставку товарів. Воно містить редаговану векторну електронну карту з прив'язаною до неї базою адрес. Система розраховує маршрути з урахуванням моделі транспортної мережі, пропуску транспорту в певні міські райони, вікна доставки, обсягу і ваги продукції. Рішення дозволяє управлінцям (супервайзерам) і диспетчерам отримати наочне порівняння запланованих і фактичних маршрутів на електронній карті в режимі "План-Факт". В результаті можна встановити відхилення маршруту від плану, усунути нецільове використання транспорту, необґрунтовані простой, слив палива, а також вирішувати ряд контрольних та аналітичних завдань.

Збільшення ефективності управління транспортом здійснюється завдяки більш широкому використанню технологій і інтегрованих систем, що забезпечують можливість контролю і аналітики. Уміння аналізувати рентабельність точок доставки, відстежувати ефективність використання транспортних ресурсів, контролювати дотримання KPI і оцінювати необхідність оптимізації логістичної інфраструктури (складів, обладнання, транспортних засобів і т.д.) все це може бути ключовим внеском у розвиток конкурентного ціноутворення продукції і виявлення можливостей для зміни.

Створення ефективного товарного потоку і мінімізації циклів виконання замовлення при успішному управлінні перевезеннями підвищує ефективність використання складської площі, сприяє економії грошових витрат на зберігання продукції і скорочує загальний час виконання замовлення. Підвищення гнучкості і ефективності управління вантажопереvezеннями створює можливість обслуговувати більшу кількість клієнтів, мінімізувати час завантаження і вивантаження, враховувати тимчасові вікна клієнтів і складу, враховувати характеристики вантажів і транспорту, а також формувати чіткий порядок відвантаження товару і виконання замовлень.

На території України подібні сервіси не розповсюджені так, як у країнах Європи, що робить цей ринок відкритим для нових пропозицій, а самі пропозиції новими і цікавими.

## **1.2 Переваги, недоліки транспортної логістики, архітектура системи**

З огляду на зростання міського населення і завантаженість доріг, а також підвищення якості послуг, розробка системи автоматизації транспортної логістики для центрів паркування є стратегічним питанням не тільки для дослідження, а й з економічної точки зору. Багато проблем з паркуванням і заторами на дорогах можна вирішити завдяки тому, що водії будуть володіти інформацією про наявність вільних місць в центрах паркування, про розташування центрів, а також, якщо водії будуть отримують відомості про найближчий центр, шлях до якого виключає ділянки доріг з можливими пробками (рис. 1.4).

Пошук найкоротшого шляху, максимального потоку, тобто оптимального маршруту, а також оцінка пройденого шляху або відстані, що залишилось подолати до контрольної точки, отримується завдяки алгоритмам транспортної логістики.

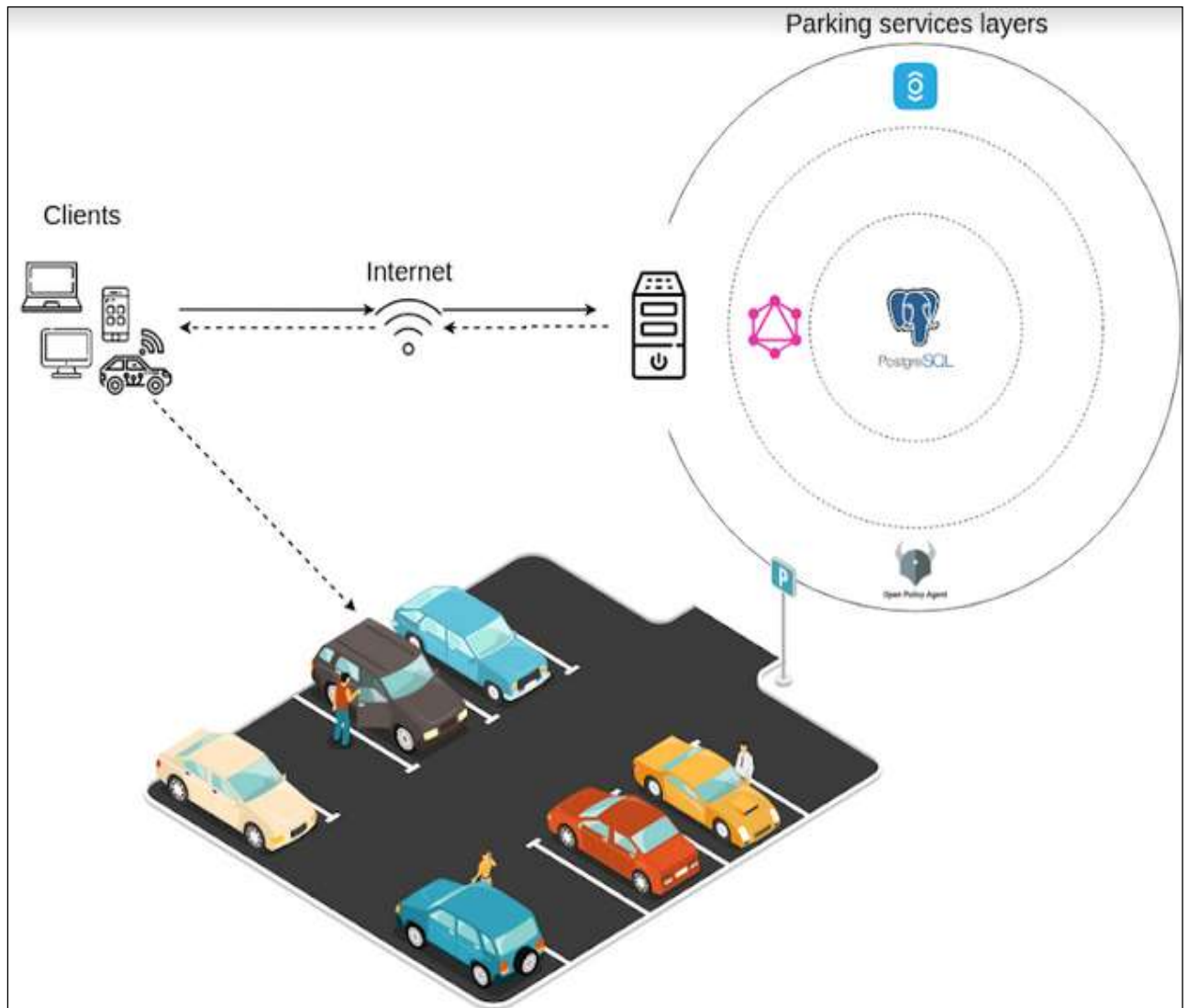


Рисунок 1.4 – Загальна концепція розробки

*Переваги сектору логістики.* Істина полягає в тому, що логістичний сектор має багато переваг, серед яких:

- Краще використання дистриб'юторської мережі: коли у вас є хороша логістична система з різними логістичними операторами, ви можете оптимізувати час разом із ланцюгом дистрибуції. Це означає, що існують різні компанії, присвячені логістиці та дистрибуції, які є основними на національному та міжнародному рівнях.

- Зменшення витрат: це можливо завдяки різним глобалізованим системам розподілу, оскільки вони зменшують транспортні витрати.

- Ефективніший логістичний ланцюг: якщо можливо здійснити більш ефективне управління логістикою, покращення як задоволення кінцевих споживачів, так і послуги.

- Транспортування та експрес-доставка: сьогодні як ніколи впроваджуються системи, що дозволяють терміновий транспорт. Таким чином, замовлення можуть досягти свого кінцевого пункту за набагато менше часу, ніж кілька років тому.

Інформаційні технології: технології допомагають розвивати сам сектор, покращуючи час і процеси.

*Недоліки логістичного сектора.* Істина полягає в тому, що є також деякі недоліки, які необхідно враховувати, зокрема:

- Координація: особливо у випадках міжнародної логістики можуть бути деякі збої в міжнародній координації.

- Багатонаціональні компанії та великі компанії: звичайно, сектор охоплюється дуже великими компаніями. Як середні, так і малі компанії мають дуже важкий доступ.

- Вартість транспорту: чим більша відстань для подорожі, тим більша її вартість. Це ускладнює отримання конкурентоспроможної ціни.

- Законність: у сфері логістики законодавство має багато, що сказати, як на рівні законів, митної політики, так і щодо входу та виходу товару.

Система автоматизації транспортної логістики збільшуватиме ефективність управління транспортним сектором, зокрема для сервісів паркування це буде ключовим внеском у розвиток конкурентоспроможності та виявлення таких можливостей:

- Оптимізоване паркування - користувачі знаходять найкраще доступне місце, економлячи час, ресурси та зусилля. Стоянка ефективно заповнюється, і комерційні та корпоративні організації можуть правильно використовувати простір.

– Скорочений рух - потік руху збільшується, оскільки менше автомобілів потрібно їздити навколо у пошуках відкритого місця для паркування.

– Зниження рівня забруднення - пошук місця для паркування спалює близько одного мільйона барелів нафти на день. Оптимальне рішення для паркування суттєво зменшить час водіння, тим самим зменшуючи кількість щоденних викидів транспортних засобів і, зрештою, зменшуючи глобальний екологічний вплив.

– Покращений досвід користування - розумне рішення щодо паркування інтегрує весь досвід користування в єдину дію. Оплата водія, ідентифікація місця, пошук місця та сповіщення про час безперешкодно стають частиною процесу прибуття до пункту призначення.

– Нові потоки доходу - завдяки технології інтелектуального паркування можливе багато нових потоків доходу. Наприклад, власники ділянок можуть увімкнути багаторівневі варіанти оплати залежно від місця паркування. Крім того, програми винагород можуть бути інтегровані в існуючі моделі, щоб заохотити повторних користувачів.

– Інтегровані платежі - користувачі, що повертаються, можуть замінювати щоденні платежі готівкою вручну на виставлення рахунків та оплату заявок зі свого телефону. Це також може забезпечити програми лояльності клієнтів та цінні відгуки користувачів.

– Підвищена безпека - працівники автостоянки та охоронці містять дані про ділянки в режимі реального часу, які можуть допомогти запобігти порушенням правил паркування та підозрілій діяльності. Камери розпізнавання номерних знаків можуть збирати відповідні кадри. Крім того, зменшення пошуку паркомісця на вулицях може зменшити кількість аварій, спричинених відволіканням уваги на пошук місця для паркування.

– Дані в режимі реального часу та аналіз тенденцій - з часом розумне рішення для паркування може створювати дані, які розкривають кореляцію та тенденції користувачів та партій. Ці тенденції можуть виявитись неоціненними для власників партій щодо того, як внести корективи та вдосконалення водіїв.

– Зниження витрат на управління - більша автоматизація та менша ручна діяльність економить на витратах робочої сили та вичерпанні ресурсів.

– Покращений сервіс та імідж бренду - безперебійний досвід дійсно може підняти для користувача імідж бренду корпоративних чи комерційних організацій. Незалежно від того, чи буде це пункт роздрібної торгівлі, аеропорту чи корпоративного офісу, відвідувачі, безсумнівно, будуть вражені передовими технологіями та факторами зручності.

### **1.3 Огляд існуючих рішень щодо автоматизації транспортної логістики**

На практиці часто виникає проблема визначення максимального потоку транспортної мережі. Цією задачею займаються вчені багато років, відомі роботи таких науковців: Ting Kien Hua, Hui Zhou, Noraini Abdullah, Jianhua Yin, Wei Wu, Biyuan Yaota інші. Для вирішення такої задачі найчастіше застосовують алгоритм Форда-Фалкерсона, завдяки якому можливо відшукати максимально можливу пропускну здатність мережі, у тому числі і для транспортних мереж міст.

До транспортної задачі пошуку найкоротшої відстані належать такі алгоритми, як алгоритм Дейкстри, алгоритм Суурбалле, алгоритм Флойда — Уоршелла та алгоритм Беллмана — Форда. Протягом багатьох років науковці різних країн світу використовують перелічені алгоритми для вирішення проблем транспортної логістики.



Автори Chu-Hsing Lin, Jui-Ling Yu, Jung-Chun Liu, Chia-Jen Lee роботи «A genetic algorithm approach for finding the shortest driving time on mobile devices» хоча й використали у своїй роботі генетичний алгоритм, щоб знайти найкоротший час у водінні з різноманітними сценаріями реальних умов руху та різної швидкості транспортного засобу, але зазначили, що це пов'язано з тим, що портативні пристрої мають обмежені ресурси, тобто неможливо використовувати для обчислення точних оптимальних рішень відомий алгоритм Дейкстри, який зазвичай використовується для пошуку найкоротшого шляху з картою з відомим числом вершин.

Інший приклад використання алгоритмів транспортної логістики є робота «Towards Online Shortest Paths Computation» авторів Leong Hou U, Hong Jun Zhao, Man Lung Yiu, Yuhong Li, Zhiguo Gong, де використовується двонаправлений пошук, що зменшує простір пошуку, виконуючи алгоритм Дейкстри одночасно вперед від початкової вершини і назад від кінцевої вершини.

Двонаправлений пошук обчислює найкоротший шлях від початкової одиниці до цільового вузла в роботі «Optimization of Shortest-Path Search on RDBMSBased Graphs» авторами Kwangwon Seo, Jinhyun Ahn, Dong-Hyuk Im. Неодноразово застосовуючи процес FEM (frontier – expand–злиття), що складається з операторів SQL для трьох операцій (обмеження, розширення та злиття), вони змогли реалізувати алгоритм Дейкстри в СУБД.

Алгоритм Дейкстри також використовується в роботі «CANDS: Continuous Optimal Navigation via Distributed Stream Processing» авторами Dinyu Yang, Dongxiang Zhang, Kian-Lee Tan, Jian Cao, Frederic Le Mouel для обчислення найкоротшого шляху між двох прикордонних вершин, що має назву ярлик, а коли відбувається оновлення стану дороги, оновлюються ярлики, щоб забезпечити коректність локальної оптимальності.

Автори Chengyi Gao, Hakki C. Cankaya, J. P. Jue в роботі «Survivable inter-domain routing with Suurballe-based intra-domain disjointness information in multi-

domain optical networks» пропонують алгоритм пошуку пари несуміжних міждоменних шляхів з мінімальною сумарною вартістю на основі матриці для кожного домену, що включає інформацію, яка генерується алгоритмом Сурбалле між внутрішньодоменними агрегованими посиланнями.

В роботі «Evaluation of algorithms for multipath route selection over the Internet» автори Fabian Helfert, Heiko Niedermayer, G. Carle вивчають різноманітні алгоритми вибору шляху, включаючи адаптовані версії алгоритму Сурбалле.

У роботі «Route planning for electric vehicle efficiency using the Bellman-Ford algorithm on an embedded GPU» авторами Adam Chambers, Matthew Eavis-O'Quinn, V. Roberge, M. Tarbouchi представлено використання алгоритму Беллмана-Форда для планування маршрутів руху на основі енергоефективності. Для подолання підвищеної обчислювальної складності та забезпечення розумних швидкостей обробки в цій роботі використовується вбудований графічний блок обробки та паралельна реалізація Беллмана-Форда на вбудованій системі графічного процесора. Цей метод дозволяє будувати маршрути із значним зменшенням енергетичних потреб, зберігаючи при цьому ефективність традиційних програм планування маршрутів.

Автори С. Okonji, O. J. Omotosho, A. C. Ogbonna в роботі «A Comparative Analysis of Four Path Optimization Algorithms» порівнюють чотири найбільш популярні алгоритми оптимізації шляху / маршруту (алгоритм оптимізації колонії мурашок, алгоритм Дейкстри, алгоритм Беллмана-Форда та алгоритм Сурбалле) для оперативного переміщення обладнання та персоналу від бази до місця стихійного лиха для ефективного реагування на надзвичайні ситуації, особливо в контексті країн, що розвиваються.

До сьогодні пошук оптимального шляху залишається досить складним завданням, це пов'язано з поганими дорожніми мережами, поганими та

застарілими навігаційними системами, несправними транспортними засобами та заторами на дорогах.

#### **1.4 Постановка задачі розробки**

**Мета** магістерської роботи є дослідження алгоритмів транспортної логістики та практична реалізація пошуку оптимального маршруту до вільного центру паркування на найкоротшій відстані від водія, виключаючи ділянки дороги, де здійснюється максимальний рух транспорту в сервісі для центрів паркування.

Поставлені та вирішені наступні **задачі**:

- 1) проаналізувати стан впровадження сучасних інформаційних технологій в транспортний сектор країни;
- 2) проаналізувати архітектуру, задачі, переваги, недоліки існуючих рішень щодо автоматизації транспортної логістики;
- 3) дослідити алгоритми транспортної логістики;
- 4) реалізувати пошук оптимального маршруту в сервісі.

**Об'єкт** – інтеграція алгоритмів транспортної логістики для пошуку оптимального маршруту.

**Предмет** – технологія IoT для забезпечення зв'язку між центром паркування та автовласниками.

Сервіс реалізує наступні функції:

1. Пошук вільних місць для паркування.
2. Пошук оптимального маршруту до центру паркування на найкоротшій відстані.
3. Резервування місць в режимі реального часу.

Основна мова програмування серверної частини була обрана - Golang, та API побудовано на технології GraphQL. Задля автоматичного розгортання,

масштабування та управління застосунками у контейнерах Docker було обрана відкрита система Kubernetes. Розподільником запитів до серверної частини використовується Nginx, а для реалізації інтерфейсу клієнта використовується JavaScript бібліотека React. В якості СУБД для розширюючих моделей використовується NoSQL - MongoDB, задля сутностей - де потрібна підтримка транзакційності SQL - Postgres.

Використання технології IoT визначає можливість здійснювати збір інформації у режимі реального часу щодо наявності вільних місць для паркування та оптимального маршруту до найближчого центру, що, у свою чергу, підвищує продуктивність та надійність міської інфраструктури.

## 2. ДОСЛІДЖЕННЯ АЛГОРИТМІВ ТРАНСПОРТНОЇ ЛОГІСТИКИ

Пошук найкоротшого шляху або максимального потоку, тобто оцінка оптимального маршруту до контрольної точки, отримується завдяки алгоритмам транспортної логістики. Найкоротший запит шляху через динамічну дорожню мережу є головною проблемою для оптимізації систем дорожнього руху в режимі реального часу. Існуючі рішення покладаються або на централізовану систему індексів із значними накладними витратами на попередні обчислення, або на розподілену систему обробки графів. Використання найкоротших алгоритмів шляху в навігації зводиться до проектування графу можливих шляхів і розмістити його на карті. Після отримання графу є можливість знайти найкоротший шлях між усіма вузлами. Вузли представляють точку або область на карті. Якість розрізу оцінюється на основі двох основних факторів:

- вміння знаходити оптимальний шлях;
- обчислювальна складність.

### 2.1 Алгоритми пошуку найкоротшого шляху

Задачу пошуку найкоротшого шляху можна вирішити, використовуючи теорію графів, завдяки наступним алгоритмам:

- алгоритм Дейкстри;
- алгоритм Торупа;
- алгоритм Рамана;
- алгоритм Флойда;
- алгоритм Беллмана-Форда;
- алгоритм Левіта.

Представлені алгоритми легко виконуються при невеликій кількості вершин графу. При збільшенні кількості вершин задача пошуку найкоротшого шляху стає дедалі складніше. У зв'язку з цим, існують модифікації існуючих підходів, а саме алгоритм Флойда – Уоршелла та алгоритм Суурбалле. Існує кілька модифікацій алгоритму Дейкстри з різною оцінкою часу виконання, серед них – алгоритми Торупа й Рамана.

Найкращих результат має алгоритм Торупа, в якому використовуються спеціальні структури даних (ієрархічна система кошиків, фібоначчівська купа), а обробка вершин графа виконується відповідно до ієрархії компонентів графа. Час роботи алгоритму оцінюється:

$$\Theta(|E| + V \cdot \log \log V) \quad (2.1)$$

*Алгоритм Беллмана-Форда* застосовується також і до графів, що містить ребра від'ємної ваги, на відміну від алгоритму Дейкстри. Якщо граф містить від'ємний цикл, то, зрозуміло, найкоротшого шляху до деяких вершин може і не існувати (у зв'язку з тим, що вага найкоротшого шляху має дорівнювати мінус нескінченності); втім, цей алгоритм можна модифікувати, щоб він сигналізував про наявність циклу від'ємної ваги, або навіть виводив сам цей цикл. Час роботи алгоритму оцінюється:

$$\Theta(E \cdot V) \quad (2.2)$$

*Алгоритм Левіта*. На кожному кроці підтримується три безлічі вершин:  $V_0$  - вершини, відстань до яких вже обчислено (але, можливо, не остаточно);  $V_1$  - вершини, відстань до яких обчислюється;  $V_2$  - вершини, відстань до яких ще не вираховано. Вершини в безлічі  $V_1$  зберігаються у вигляді двобічної черги (deque). Час роботи алгоритму оцінюється:

$$\Theta(E \cdot \log V) \quad (2.3)$$

### 2.1.1 Алгоритм Дейкстри

Отже, алгоритм Дейкстри являє собою  $n$  ітерацій, на кожній з яких вибирається непомічена вершина з найменшою величиною, ця вершина позначається, і потім проглядаються всі ребра, які виходять із даної вершини, і вздовж кожного ребра робиться спроба поліпшити значення на іншому кінці ребра. Час роботи алгоритму складається з:

- $n$  раз пошук вершини з найменшою величиною серед всіх непомічених вершин, тобто серед  $V$  вершин;
- $m$  раз проводиться спроба релаксації.

При простій реалізації цих операцій на пошук вершини буде витрачатися  $\Theta(V)$  операцій, а на одну релаксацію -  $\Theta(1)$  операцій, і підсумкова асимптотика алгоритму становить:

$$\Theta(V^2 + E) \quad (2.4)$$

Алгоритм на графах, що знаходить найкоротші шляхи від однієї з вершин графа до всіх інших. Алгоритм працює тільки для графів без ребр від'ємної ваги.

$$d(x) = \min\{d(x); d(y) + a_{y,x}\} \quad (2.5)$$

Кожній вершині  $X$  в ході алгоритму присвоюється число  $d(x)$ , яке дорівнює довжині найкоротшого шляху з вершини  $S$  в вершину  $X$  і включає тільки пофарбовані вершини. Спочатку передбачається  $d(s) = 0$  і  $d(x) = \infty$  для всіх інших вершин графа. Далі фарбуємо вершину  $S$  і вважаємо  $y = S$ , де  $y$  - остання пофарбована вершина.

Для кожної нефарбованої вершини  $X$  перераховується величина  $d(x)$  за формулою (1). Якщо  $d(x) = \infty$  для всіх нефарбованих вершин, то алгоритм закінчується, тому що відсутні шляхи з вершини  $S$  в нефарбовані вершини. Інакше зафарбовується та вершина, для якої величина  $d(x)$  є мінімальною. Зафарбовується і дуга, що веде в цю вершину і вважається  $y = x$ . Якщо  $y = t$ , найкоротший шлях з  $S$  в  $t$  знайдений.

На кожному кроці обробляється найближча, ще необроблена вершина. При її обробці всі ще невідвідані сусіди додаються в чергу для відвідування.

### 2.1.2 Алгоритм Флойда – Уоршелла

Алгоритм Флойда - Уоршелла є ефективним для розрахунку всіх найкоротших шляхів в щільних графах, коли має місце велика кількість пар ребр між парами вершин. У разі розріджених графів з ребрами невід’ємної ваги кращим вибором вважається використання алгоритму Дейкстри для кожного можливого вузла. Час роботи алгоритму оцінюється:

$$\Theta(|V|^3), \text{ де } V - \text{кількість вершин.} \quad (2.6)$$

Динамічний алгоритм для знаходження найкоротших відстаней між усіма вершинами зваженого орієнтованого графа.

$$d_{ij}^k = \min(d_{ij}^{k-1}, d_{ik}^{k-1} + d_{kj}^{k-1}) \quad (2.7)$$

Перед  $k$ -ою фазою ( $k=1\dots n$ ) вважається, що в матриці відстаней збережені довжини таких найкоротших шляхів, які містять в якості внутрішніх вершин тільки вершини з безлічі  $\{1, 2, \dots, k-1\}$ .



На вхід програмі подається граф, заданий у вигляді матриці суміжності - двовимірного масиву розміру  $n \times n$ , в якому кожен елемент задає довжину ребра між відповідними вершинами.

Передбачається, що якщо між двома якими-небудь вершинами немає ребра, то в матриці суміжності буде записано яке-небудь велике число; тоді це ребро завжди буде не вигідно брати, і алгоритм спрацює правильно.

### 2.1.3 Алгоритм Суурбалле

Алгоритм Суурбалле - це алгоритм знаходження двох непересічних шляхів в орієнтованому графі з невід'ємними вагами, так що обидва шляхи пов'язують ту ж саму пару вершин і мають мінімальну загальну довжину. Основна ідея алгоритму Суурбалле - використання алгоритму Дейкстри для пошуку шляху, модифікація ваг ребр графа і потім прогін алгоритму Дейкстри вдруге.

Вихід алгоритму формується шляхом комбінування двох шляхів, відкидання ребр, які проходяться в протилежних напрямках цими шляхами, і використання решти ребр для формування шляхів, які і служать виходом алгоритму.

Час роботи алгоритму оцінюється:

$$O(|E| + |V| \log |V|), \text{ де } E - \text{кількість ребр.} \quad (2.8)$$

## 2.2 Тестування алгоритмів пошуку найкоротшого шляху

Для тестування визначено транспортну мережу міста Сєверодонецьк для існуючих центрів паркування за допомогою орієнтованого зв'язного графу без петель і паралельних ребр.

Таблиця 2.1 – Геоінформаційні дані транспортної мережі міста

<b>A</b>	Chemists Avenue, 61, Severodonetsk, Luhansk region, 93400	48.952048, 38.508595	{ "B": 5, "D": 3, "G": 4, "H": 8 }
<b>B</b>	Egorova Street, 2, Severodonetsk, Luhansk Region, 93400	48.959746, 38.491267	{ "C": 3 }
<b>C</b>	Chemists Avenue, 46, Severodonetsk, Luhansk region, 93400	48.950842, 38.501561	{ "E": 2, "F": 1 }
<b>D</b>	Mendeleeva Street, 31, Severodonetsk, Luhansk Region, 93400	48.951124, 38.498814	{ "C": 2, "F": 3, "B": 4 }
<b>E</b>	Fedorenko Street, 10, Severodonetsk, Luhansk region, 93400	48.946164, 38.492806	{ "F": 1 }
<b>F</b>	Central Avenue, 46g, Severodonetsk, Luhansk region, 93400	48.945318, 38.494179	{ "G": 6 }
<b>G</b>	Gvardiysky Avenue, 38, Severodonetsk, Luhansk region, 93400	48.942418, 38.518193	{ "H": 2 }
<b>H</b>	Vilesova Street, 20-22, Severodonetsk, Luhansk Region, 93400	48.936952, 38.525698	nil

Визначено контрольні точки, що є вершинами графу, географічні координати та адреси яких, перелічені у табл. 2.1. Мережа, зображена на рис. 2.1, складається з 8 вершин і 13 дуг.

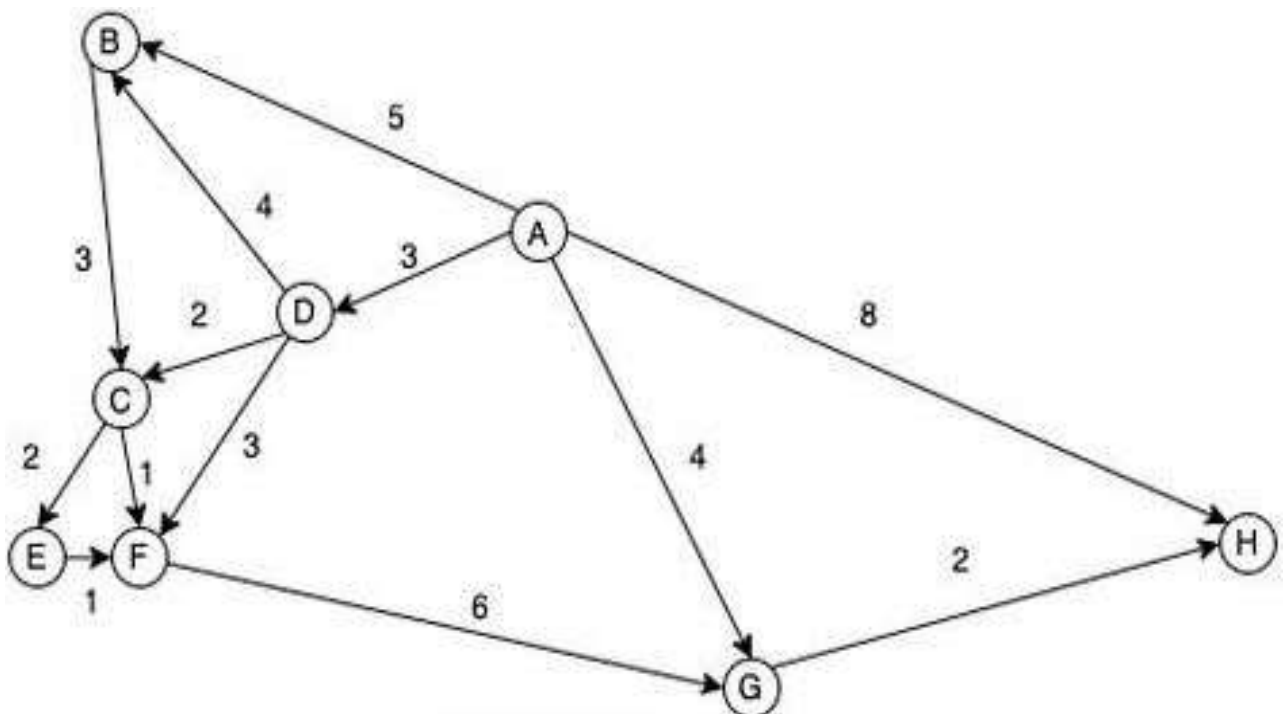
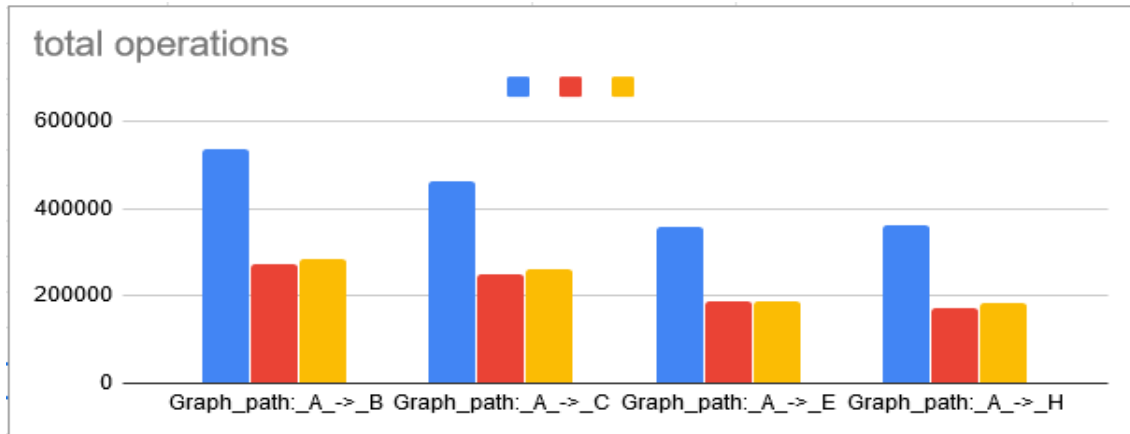
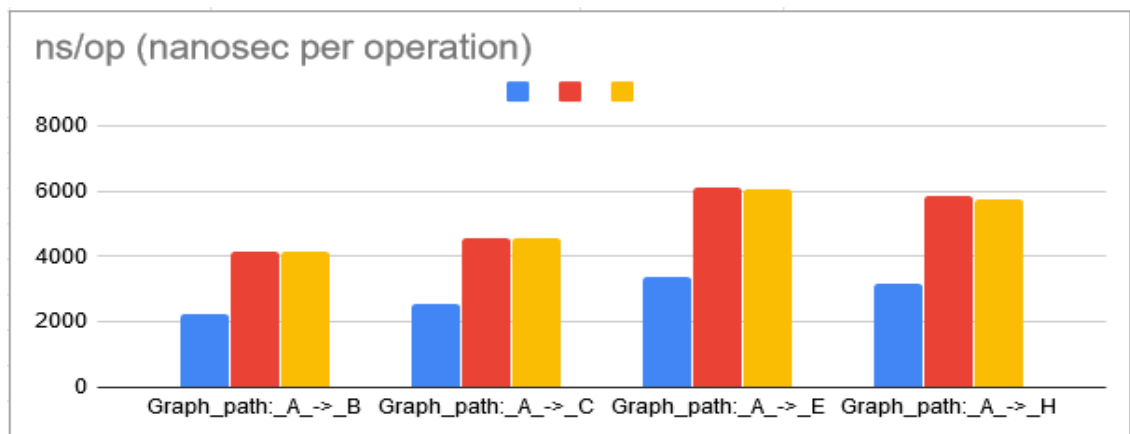


Рисунок 2.1 - Граф транспортної мережі

В ході тестування перелічені алгоритми порівняні за такими параметрами, як кількість операцій (рис. 2а) та час виконання алгоритму (рис. 2б). Отримані результати наведено у табл.2.2.



а)



б)

Рисунок 2.2 - Отриманні результати порівняння алгоритмів

Алгоритм Флойда - Уоршелла (на рис. 2.2 - червоні стовпці) є ефективним для розрахунку всіх найкоротших шляхів в щільних графах, коли має місце велика кількість пар ребр між парами вершин. У разі розріджених графів з ребрами невід'ємної ваги кращим вибором вважається використання алгоритму Дейкстри (на рис. 2.2 - сині стовпці) для кожного можливого вузла. Такий же висновок отримаємо завдяки діаграмам на рис. 2.2.

Таблиця 2.2 - Результати порівняння алгоритмів

Benchmark_Shortest_Path/	dijkstra		floyd		suurbale	
1.total_operations						
2. ns/op (nanosec per operation)	1	2	1	2	1	2
Graph_path: _A_->_B	536676	2238	271456	4130	283383	4143
Graph_path: _A_->_C	463988	2536	248994	4544	260944	4550
Graph_path: _A_->_E	358706	3398	187286	6125	186771	6060
Graph_path: _A_->_H	362128	3184	173468	5867	183522	5774

Незважаючи на те, що пошук найкоротшої відстані для різних тестових наборів згідно алгоритму Флойда – Уоршелла виконується за меншу кількість операцій, час виконання алгоритму Дейкстри швидше для всіх пошукових відстаней.

Тому саме алгоритм Дейкстри використовується для пошуку шляху від місцеперебування автовласника до найближчого паркувального центру. Для розглянутого графу отримано шлях крізь дуги  $(v_A, v_D)$ ,  $(v_D, v_C)$ ,  $(v_C, v_F)$ .

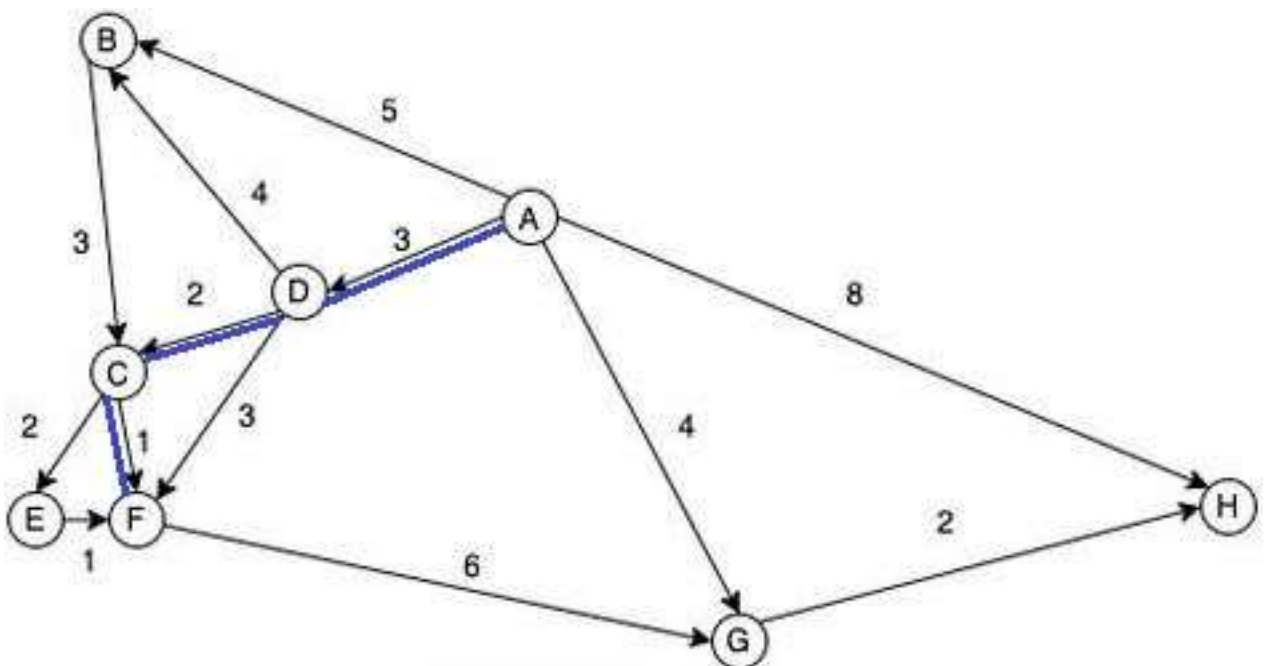


Рисунок 2.3 - Найкоротший шлях

### 2.3 Алгоритм пошуку максимального потоку

Принцип роботи алгоритму Форда-Фалкерсона полягає в наступному: починається з нульового потоку, а потім на кожній ітерації треба збільшити потік мережі. Тобто на кожному кроці є такий ланцюг, що збільшуватиме величину потоку  $f$  уздовж дуг цього ланцюга до насиченості.

Ланцюг з джерела в стік, в якому всі дуги допустимі, є збільшувальним ланцюгом. Якщо виконується одна з наступних умов, дуга з вершини  $u$  до вершини  $v$  є допустима:

1. Дуга узгоджена, якщо  $f(u, v) < c(u, v)$ ;
2. Дуга неузгоджена, якщо  $f(u, v) > 0$ ,

де  $f(u, v)$  - величина потоку, що можна пропустити по дузі,  $c(u, v)$  - залишкова пропускна здатність.

За збільшувальним ланцюгом можна пропустити потік величиною  $Q$ :

$$Q = \min\{q(u, v)\} \quad (2.9)$$

де  $q(u, v) = \{c(u, v) - f(u, v), \text{ якщо дуга узгоджена}; f(u, v), \text{ якщо дуга неузгоджена}\}$ .

Для того, щоб величина потоку мережі збільшилася на  $Q$ , треба на кожній узгодженій дузі ланцюга збільшити і на кожній неузгодженій зменшити потік на  $Q$ . Для знаходження збільшувального ланцюга використовується "Метод розстановки позначок". Процес розміщення міток в джерелі мережі розпочинається і в її стоці закінчується. Підтверджується існування збільшувального ланцюга з джерела в стік, як тільки стік виявився поміченим. Стан мітки на вершини мережі містить необхідну інформацію, достатню для відновлення цього ланцюгу та визначення величини зміни потоку. Можливе перебування вершини мережі в одному з 3-х станів: "непомічена", "позначена" і "переглянута".

Тобто, алгоритм Форда-Фалкерсона здійснює пошук такої множини потоків по дугах, щоб величина  $Q(v_s)$  - пропускна здатність транспортної мережі - була максимальною:

$$Q(v_s) \rightarrow \max \quad (2.10)$$

що відповідає мінімальному розрізу графа  $S \rightarrow \min$ .

Розріз вирізняє  $v_s$  від  $v_t$ , якщо вершини  $V_s, V_t$  відносяться до різних сторін розрізу:  $v_s \in V_s, v_t \in V_t, V = V_s \cup V_t$ . При цьому пропускна здатність розрізу графа визначається як:

$$c(S) = \sum_{e_j \in (V_s \rightarrow V_t)} c_j \quad (2.11)$$

де  $S$  – розріз,  $c(S)$  – пропускна здатність розрізу графа.

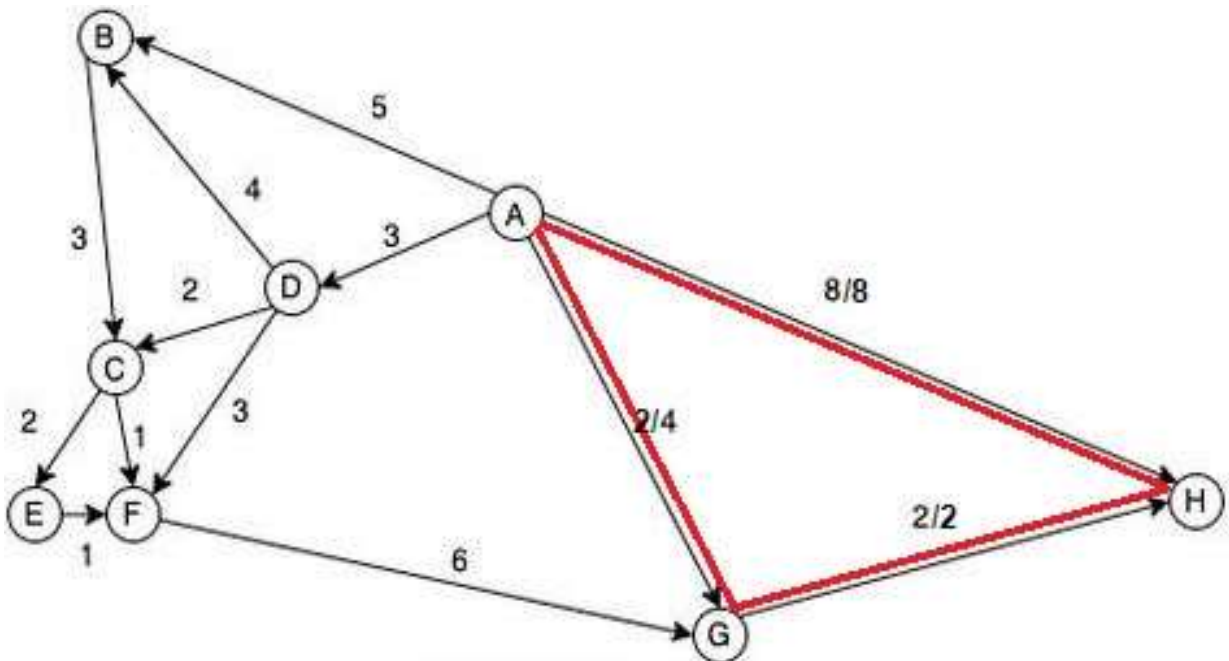


Рисунок 2.4 - Мінімальний розріз графу

Перевіримо роботу алгоритму для тієї ж транспортної мережі існуючих центрів паркування міста Сєвєродонецьк. В результаті, отримано максимальний потік, величина якого  $Q(v_s) = 10$ , дуги  $(vA, vG)$ ,  $(vG, vH)$ ,  $(vA, vH)$  утворюють мінімальний розріз, тим самим визначено найбільш навантажені ділянки дороги.

## 2.4 Інтеграція алгоритмів Дейкстри та Форда-Фалкерсона

Інтеграція двох вищезгаданих алгоритмів транспортної логістики з математичної точки зору виглядає наступним чином. Для транспортної системи міста застосовано графову модель:

$$G=(V,E), \quad (2.12)$$

з множиною вузлів

$$v=|V|, \quad (2.13)$$

з множиною дуг

$$e=|E|, \quad (2.14)$$

де  $v$  - кількість вузлів,  $e$  - кількість дуг,  $| \cdot |$  - знак потужності множини.

Результат роботи алгоритму Дейкстри виражається наступною множиною вузлів:

$$v_D=\{A, D, C, F\}, \quad (2.15)$$

а результат роботи алгоритму Форда-Фалкерсона:

$$v_{FF} = \{A, G, H\}. \quad (2.16)$$

Початковий вузол позначимо  $v_0$ , для розглянутого графу  $v_0 = \{A\}$ .

Тоді для того, щоб отримати оптимальне рішення щодо транспортного шляху, використовуємо вираз:

$$(v_D \cap v_{FF}) \setminus v_0 = \emptyset. \quad (2.17)$$

Згідно формулі 2.17 отримуємо порожню множину, що свідчить про вільний від пробок шлях до найближчого центру паркування (рис. 2.5). В іншому випадку, якщо множина непорожня, отримаємо набір вузлів, через які проходять навантажені транспортним потоком ділянки доріг.

Після інтеграції алгоритмів Дейкстри та Форда-Фалкерсона, отримано оптимальне рішення щодо шляху до найближчого паркувального центру, мінуючи найбільш навантажені ділянки дороги транспортної мережі на прикладі міста Северодонецьк. Це розширило функціонал сервісу, який надає інформацію про кількість вільних місць для паркування в режимі реального часу; про час, коли доступні місця для паркування, щоб користувачі могли зарезервувати місце на вибраній автопарковці.



Рисунок 2.5 – Результат інтеграції алгоритмів на прикладі м. Северодонецьк



### **3. СИСТЕМА АВТОМАТИЗАЦІЇ ТРАНСПОРТНОЇ ЛОГІСТИКИ**

#### **3.1 Функціонал системи автоматизації**

Огляд існуючих сервісів дозволив врахувати світовий досвід для реалізації сервісу. Розробка спрямована на підвищення якості наданих послуг у сфері автотранспорту. Головною відмінністю пропонованого рішення є отримання шляху до центру паркування, що знаходиться найближче до поточного місця розташування автовласника, виключаючи ділянки доріг з максимальним транспортним потоком, таким чином, частково розв'язати проблему заторів, що виникають в наслідок неефективного руху автомобілів в загальному потоці при пошуку місця для паркування.

Створено систему, що використовується для інформування клієнтів про стан вільних місць кожного окремого центру паркування, а також здійснює управління місцем паркування, як безпосередньо власником автотранспорту, так і власником центру.

Сервіс реалізує наступні функції:

1. Пошук найближчого центру паркування.
2. Пошук максимального потоку міста.
3. Перевірка наявності вільних місць для паркування.
4. Бронювання паркувальних місць в режимі реального часу.

Головна мета роботи системи автоматизації транспортної логістики - це спростити сервіс паркування співробітників і забезпечити надійність завчасного бронювання. Система, структура якої зображена на рисунку 3.1, пропонує максимально мінімізовані дії для початку паркування автовласників:

1. Реєстрація компанії.
2. Розташування гаражу компанії.
3. Реєстрація працівників.

4. Редагування календаря для бронювання місця паркування для автовласників.

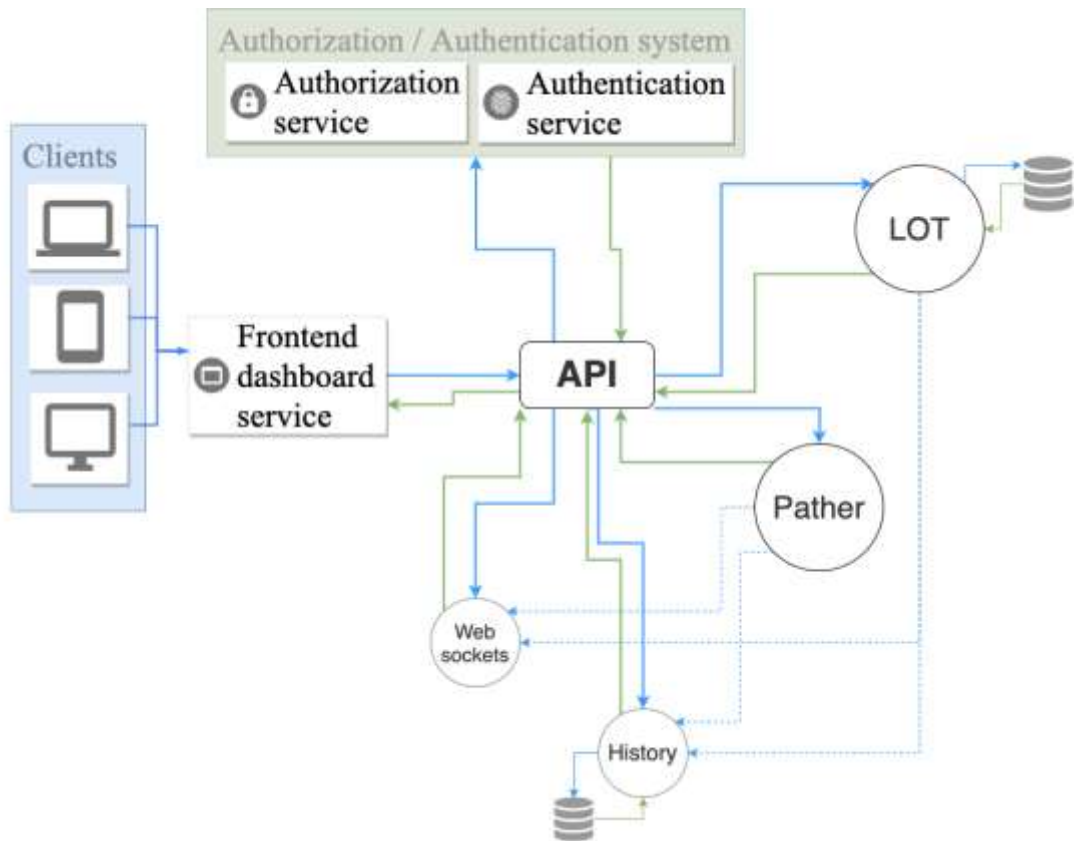


Рисунок 3.1 – Структура системи для паркувального центру

Основаючись на алгоритмах транспортної логістики, а також враховуючи усі перелічені переваги та недоліки, була розроблена структура системи автоматизації транспортної логістики для сервісів паркування (рис. 3.1). Розроблена система передбачає функціонал пошуку найкоротшого шляху від місця клієнта до найближчого центру парковці з вільними місцями.

Компоненти системи відповідають за наступне:

- “Auth” – автентифікація користувача у системі;
- “Accounting” - відповідальність за модель аккаунту, обробка інформації та властивостей користувача;
- “Locator” - відстежує поточне положення клієнту. Цей модуль відповідає за отримання даних із системи. Побудований як звичайний декодер

даних положення до використання в інших частинах сервісу, як правило, для отримання стану паркувальних місць;

- “Lot” - відстежує стан стоянки. Відповідає за управління, блокування та звільнення паркувальних місць;

- “Grapher” – пошук найкоротшого шляху до паркувального майданчика алгоритмом Дейкстри та за допомогою максимального потоку алгоритму Форда-Фолькерсона.;

- “History” - збереження історії дій користувача;

- “Websocket” - ріалтайм системи.

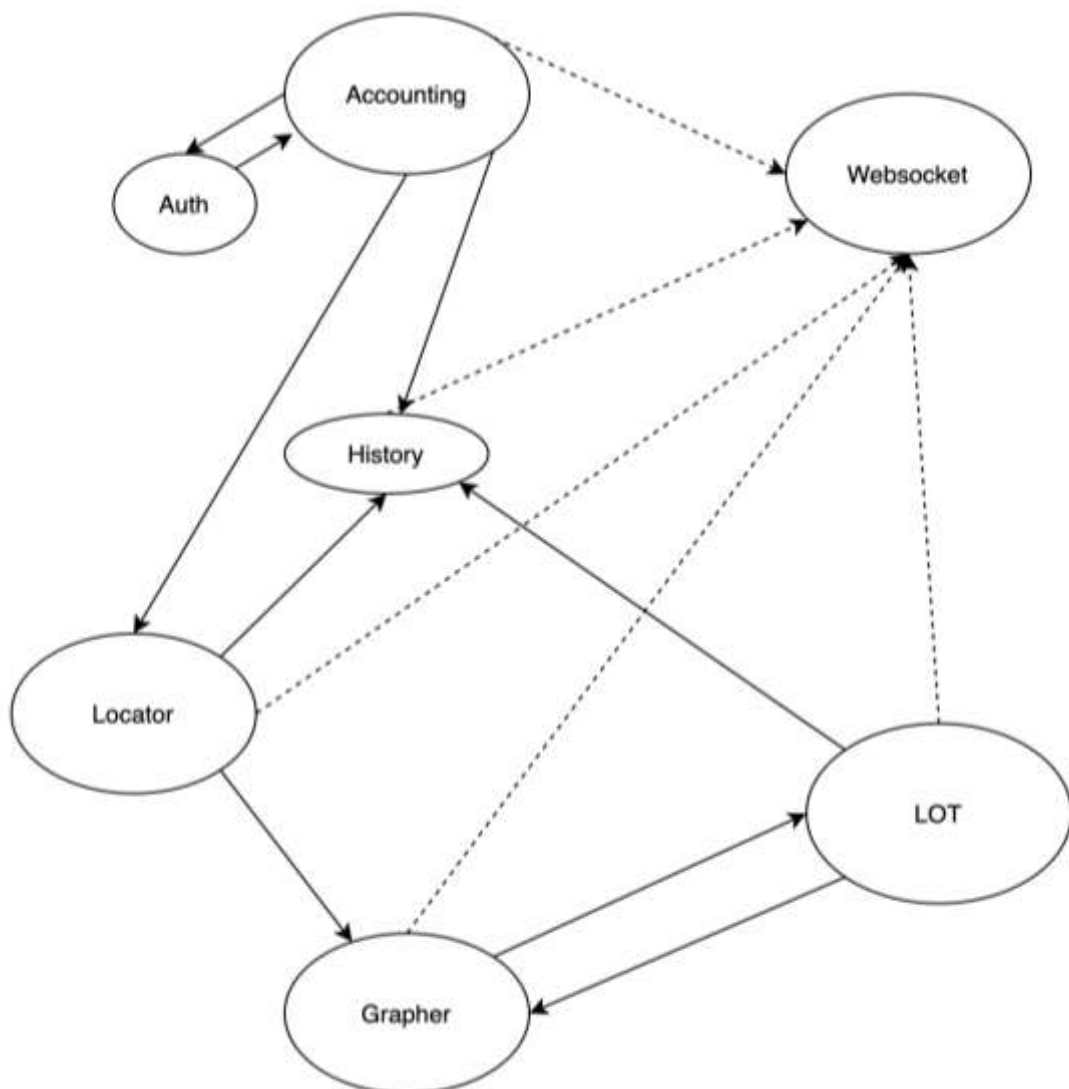


Рисунок 3.2 - Схема взаємодії компонентів системи автоматизації транспортної логістики для сервісів паркування

### 3.2 Реалізація алгоритму Форда-Фалкерсона в системі

Graph - структура даних, що використовується для представлення графу, VertexEdges var - це карта, де кожен ключ є вершиною, а значення - карта, де ключі - це вершини, до яких можна дістатися з основної вершини ключа, і значення ваги для цього ребра, властивість Vertices використовується як набір, який містить усі доступні вершини на графу. Властивість NegEdges вказує, чи містить граф негативні ребра чи ні.

```
type Graph struct {
    RawEdges      []Edge
    Vertices      map[uint64]bool
    VertexEdges   map[uint64]map[uint64]float64
    Undirected    bool
    NegEdges      bool
}
```

Edge - представлення краю одного з ребр графу, містить вершини для зважених графів.

```
type Edge struct {
    From  uint64
    To    uint64
    Weight float64
}
```

Distance - структура використовується для представлення відстані вершини від одної до іншої на графу.

```
type Distance struct {
    From  uint64
    Dist  float64
}
```

Функція MinCutMaxFlow (лістинг коду якої, див. дод. А) обчислює і повертає мінімальну кількість ребр, які потрібно видалити, щоб отримати два

неперервні набори вершин. Можливо видалити будь-який можливий шлях від початку до вершини `dest`, видаливши ці краї.

Сума всіх ребр, які потрібно видалити, відповідає пропускній здатності між початковим та кінцевим вузлами, максимальному потоку між двома ребрами.

`maxFlow` - використовується для `MinCutMaxFlow` для обчислення всіх можливих шляхів між двома точками.

```
func (gr *Graph) maxFlow(orig, dest uint64, undirEdges
map[uint64][]uint64, flows map[uint64]map[uint64]float64,
visitedEdges map[uint64]bool, path []uint64) {
if orig == dest {
    gr.recalcFlows(path, flows)
    return
}
for _, t := range undirEdges[orig] {
    if _, yetVisited := visitedEdges[t]; yetVisited {
        continue
    }
    visitedEdges[t] = true
    path = append(path, t)
    gr.maxFlow(t, dest, undirEdges, flows, visitedEdges,
path)
    path = path[:len(path)-1]
    delete(visitedEdges, t)
}
}
```

### **3.3 Інтеграція алгоритмів пошуку найкоротшої відстані та максимально можливої пропускної здатності транспортної мережі міста**

`Pather` – сервіс відповідальний за пошук коротшого шляху від місця знаходження клієнта до паркувального майданчика, на базі двох алгоритмів Дейкстри та Форда-Фалкерсона. При запиті найкоротшого шляху програмі потрібні вхідні дані – місцезнаходження клієнта, та лист паркувальних майданчиків.

Маючи усі необхідні дані – запускається пошук шляху. З початку два алгоритми починають працювати паралельно, та після видачі перших результатів потрібно переконатися, що знайдені шляхи не співпадають.

Потрібно виключити максимально завантажені ділянки шляху, знайдені алгоритмом Форда-Фалкерсона, зі знайденого найкоротшого шляху алгоритмом Дейкстри. Якщо ділянки співпадають, то потрібно збільшити вагу для пересічних ділянок шляху та відправити дані на пошук нового коротшого шляху через алгоритм Дейкстри (рис. 3.3).

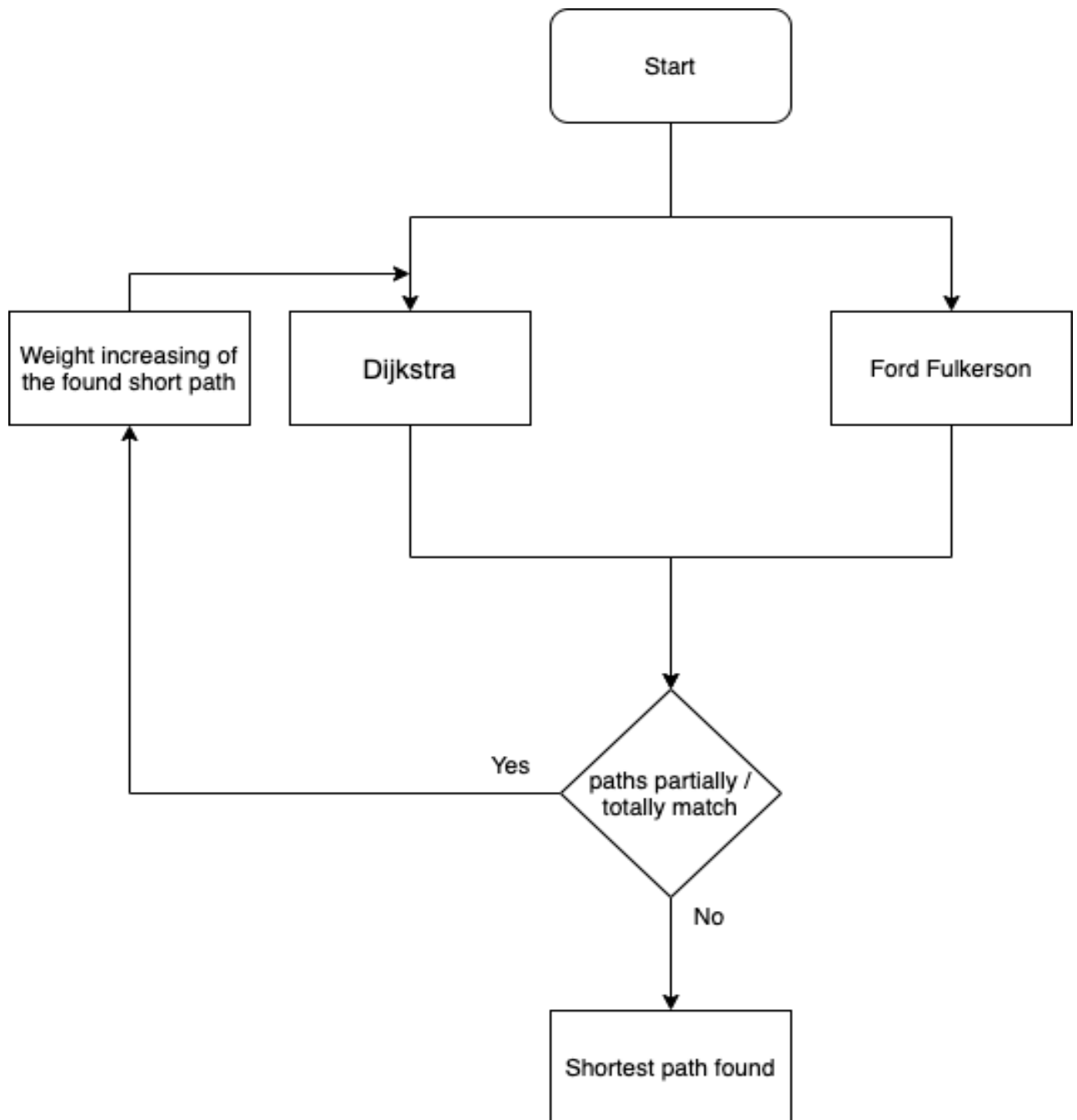


Рисунок 3.3 – Інтеграція алгоритмів для пошуку найкоротшого шляху від клієнта до паркувального майданчика

### 3.4 Схема бази даних системи автоматизації транспортної логістики

Один з ключових етапів розробки - створення бази даних, тому структурована схема бази даних зображена на рис 3.4.

Типова стоянка складається з одного або декількох блоків, які далі поділяються на поверхи. Кожен поверх містить кілька відділів, які допомагають водіям зорієнтуватися та запам'ятати місця для паркування. Вони, як правило, маркуються буквами, такими як "А", "В", "С" тощо.

Крім того, поверх містить кілька унікальних номерів - паркувальних місць.

Деякі з цих слотів призначені для людей з обмеженими можливостями.

Визначено 8 сутностей бази даних, які володіють наступними атрибутами:

- **Organizations** – сутність власника паркувального майданчику:

id – РК, унікальний ідентифікатор;

name – назва;

meta – динамічний набір даних організації;

created\_at, updated\_at – дата створення, оновлення організації.

- **Parkings** – сутність паркувального центру:

id – РК, унікальний ідентифікатор;

latitude, longitude – широта, довжина паркувального центру;

created\_at, updated\_at – дата створення, оновлення центру;

organization\_id – FK відношення центру до організації.

- **Spots** – сутність паркомісця:

id – РК, унікальний ідентифікатор;

meta – динамічний набір даних організації;

created\_at, updated\_at – дата створення, оновлення місця;

parking\_id - FK відношення центру до паркувального центру.

- **Wallets** – сутність гаманця клієнту:

- id – PK, унікальний ідентифікатор;  
 created\_at, updated\_at – дата створення, оновлення гаманця.
- **Assets** – сутність цінності(валюти):  
 id – PK, унікальний ідентифікатор;  
 meta – динамічний набір даних цінності;  
 created\_at, updated\_at – дата створення, оновлення цінності.
  - **Wallet\_assets** – співвідношення гаманця до цінності, та її кількість:  
 wallet\_id - FK відношення до сутності гаманця;  
 asset\_id - FK відношення до сутності цінностей;  
 amount – кількість цінностей на гаманці.
  - **Reservations** – сутність резерву паркомісця:  
 id – PK, унікальний ідентифікатор;  
 start\_at, end\_at – початок, кінець резерву;  
 created\_at, updated\_at – дата створення, оновлення резерву;  
 amount – ціна бронювання;  
 asset\_id - FK відношення до цінності;  
 spot\_id - FK відношення до паркомісця, яке заброньоване;  
 transaction\_id - FK відношення транзакції в рамках резервації.
  - **Transactions** – сутність транзакцій:  
 id – PK, унікальний ідентифікатор;  
 created\_at, updated\_at – дата створення, транзакції;  
 creator – FK, гаманець(покупця) створення транзакції;  
 status – стан транзакції.

Вирішена одна з найважчих задач водія – мінімально витратити час на пошук вільного місця та оптимального маршруту до центру паркування - за рахунок реалізації системи автоматизації транспортної логістики для центрів паркування.



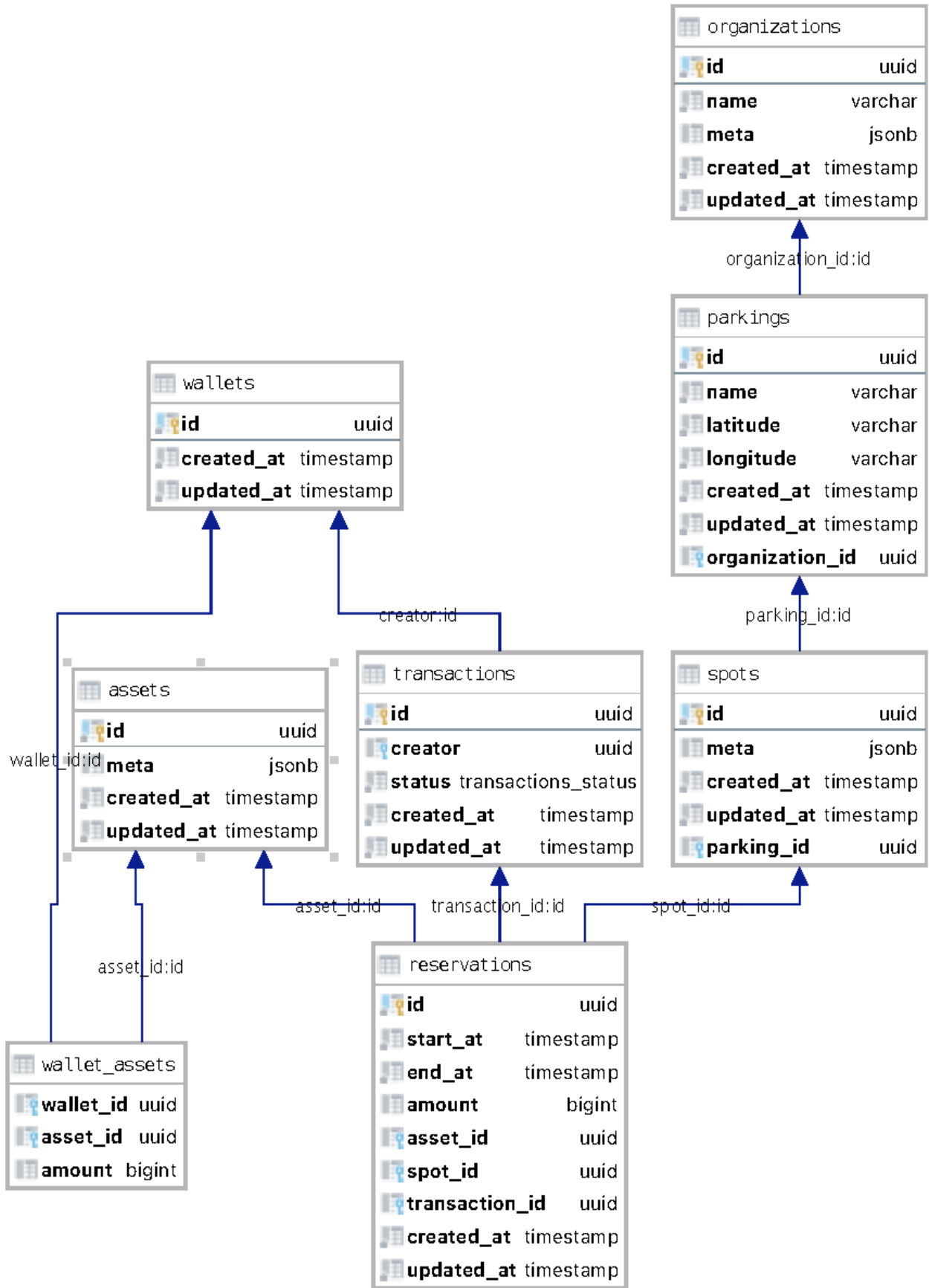


Рисунок 3.4 – Схема бази даних

Такий сервіс надає інформацію про наявність найближчих паркувальних центрів, кількість вільних місць для паркування в режимі реального часу, максимально навантажених ділянках дороги; користувачі також можуть зарезервувати місце на обраній парковці.

Стосовно розробленої системи автоматизації транспортної логістики, основна функціональність IoT полягає в тому, що за допомогою Інтернету та додатків між центром паркування та автовласниками забезпечено взаємозв'язок. Тобто сервіс володіє інформацією про центр паркування зі своїми паркомісцями, забезпечує збереження та передачу шляхом Інтернет-запиту на гаджети автовласників інформації про оптимальний маршрут до найближчого центру паркування, де є вільні місця.

## 4 ОХОРОНА ПРАЦІ

В даному розділі проведено аналіз потенційних небезпечних причин. Розглянуті заходи, які дозволяють забезпечити гігієну праці та виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики. Так як завданням дипломної роботи є розробка не тільки макету інформаційного табло на базі світлодіодного дисплею, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для серверу, де буде знаходитися веб-додаток програмно-апаратного комплексу.

### 4.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу

Основними характеристиками персонального комп'ютера є наступні:

1. Робоча напруга  $U=+220\text{В} \pm 5\%$ .
2. Робочий струм  $I=2\text{А}$ .
3. Споживана потужність  $P=350\text{ Вт}$ .

Роботу користувача розробленої підсистеми слід віднести до категорії Ia (легкі фізичні роботи) відповідно до даної категорії відносяться всі види діяльності, які виконуються сидячи й не вимагають фізичного напруження.

При експлуатації даного програмного продукту відповідно існують наступні небезпечні й шкідливі виробничі фактори:

1. Фізичні:
  - Підвищений рівень напруги електричної мережі, замикання якої може відбутися через тіло людини.
  - Підвищена або знижена вологість повітря.
  - Підвищена або знижена рухливість повітря.
  - Підвищений рівень статичної електрики.
  - Підвищена напруженість електричного й магнітного полів.

- Відсутність або нестача природного світла.
  - Знижена освітленість робочої зони.
  - Підвищений рівень шуму на робочому місці.
  - Підвищений рівень електромагнітного випромінювання.
  - Знижена контрастність.
2. Психофізіологічні:
- Фізичні перевантаження: статичні та динамічні.
  - Нервово-психічні перевантаження: розумове перенапруження, монотонність праці, перенапруження аналізаторів та емоційні перевантаження.

## 4.2 Електробезпека

Основним небезпечним фактором при роботі з ЕОМ є небезпека ураження людини електричним струмом, яка посилюється тим, що органи чуття людини не можуть на відстані виявити наявності електричної напруги на устаткуванні.

Проходячи через тіло людини, електричний струм справляє на нього складну дію, що є сукупністю термічної (нагрів тканин і біологічних середовищ), електролітичної (розкладання крові і плазми) і біологічної (подразнення і збудження нервових волокон та інших органів тканин організму) дій.

Тяжкість ураження людини електричним струмом залежить від цілого ряду чинників:

1. Значення сили струму.
2. Електричного опору тіла людини і тривалості протікання через нього струму.
3. Роду і частоти струму.
4. Індивідуальних властивостей людини і навколишнього середовища.

Відповідно до, приміщення для ЕОМ відноситься до приміщень без підвищеної небезпеки, тобто до приміщень, у яких відсутні умови, що створюють підвищену або особливу небезпеку. Небезпека ураження

електричним струмом існує всюди, де використовуються електроустановки, тому приміщення без підвищеної небезпеки не можна назвати безпечними.

Електробезпека забезпечується:

1. Відповідною конструкцією електроустановок.
2. Застосуванням технічних способів і засобів захисту.
3. Організаційними і технічними заходами.

Конструкція електроустановок відповідає умовам їхньої експлуатації і забезпечує захист персоналу від дотику до струмоведучих частин.

Основними технічними способами й засобами захисту від ураження електричним струмом, які використовуються окремо або в поєднанні один з одним, є:

1. Захисне заземлення.
2. Вирівнювання потенціалів.
3. Мала напруга.
4. Електричне розділення мереж.
5. Захисне відключення.
6. Ізоляція струмоведучих частин.
7. Компенсація струмів замикання на землю.
8. Захисні пристрої.
9. Попереджувальна сигналізація, блокування, знаки безпеки.
10. Ізолюючі захисні і запобіжні пристосування.

Основними технічними способами і засобами захисту від ураження електричним струмом, що передбачаються в даному дипломному проєкті, є:

1. Захисне заземлення.
2. Занулення.
3. Захисне відключення.
4. Ізоляція струмоведучих частин.

Занулення в комплексі із захисним відключенням зменшує напругу дотику і обмежує час, в перебігу якого людина, торкнувшись до корпусу, може потрапити під дію напруги.

### 4.3 Освітлення

Трудова діяльність людини завжди протікає в певних метеорологічних умовах, які визначаються поєднанням температури повітря, швидкості його руху і відносної вологості, тиском і тепловим випромінюванням від нагрітих поверхонь. Оскільки експлуатація проектного програмного засобу відбувається в приміщенні, то ці показники в сукупності (за винятком тиску) називаються мікрокліматом виробничого приміщення. В даний час основним нормативним документом нормалізації мікроклімату є ДБН В.2.5-67:2013. [25] «Опалення, вентиляція та кондиціонування».

Тяжкість праці характеризує сукупну дію всіх елементів, що складають умови праці, на працездатність людини, його здоров'я, життєдіяльність і відновлення робочої сили. У такому представленні поняття тяжкості праці однаково застосовне як до розумової, так і до фізичної праці. Оптимальні норми мікроклімату, згідно санітарних норм ДСН 3.3.6.042-99 [26], в робочій зоні, забезпечувані для робіт легкої категорії 1б приведені в табл. 4.3.1.

Таблиця 4.3.1 - Оптимальні норми мікроклімату

Період року	Температура, °С	Відносна вологість, %	Швидкість руху повітря, м/с, не більш
Холодний і перехідної	21 - 23	60 - 40	0,1
Теплий	22 - 24	60 - 40	0,2

У приміщенні, де знаходяться ЕОМ, повітрообмін реалізується за допомогою природної організованої вентиляції (вентиляційні шахти) і установки кондиціонера БК-2000. Цей метод забезпечує приток потрібної кількості свіжого повітря, що визначається в ДБН (30 кубічних метрів на годину на одного працюючого).

Для захисту від електромагнітного випромінювання передбачаються наступні заходи:

1. Застосування нових плазмових моніторів;
2. Віддалення робочого місця не менше, ніж на 0,4 - 0,5 м, оскільки напруженість електричного поля зменшується при віддаленні від джерела поля;
3. Встановлення раціональних режимів роботи персоналу (обмеження часу перебування);
4. Раціональне розміщення в робочому приміщенні устаткування, що випромінює електромагнітну енергію.

Оскільки рівень шуму не перевищує гранично допустимих величин, які встановлені санітарними нормами ДСН 3.3.6.037-99 [27], заходи для зниження шуму не проводяться.

Для зниження стомлюваності обслуговуючого персоналу в приміщеннях, де розташовані обчислювальні засоби, передбачається використовувати спокійні колірні поєднання і покриття, що не дають відблисків.

Світло є природною умовою існування людини. Воно впливає на стан вищих психічних функцій і фізіологічні процеси в організмі. Хороше освітлення діє тонізуюче, створює гарний настрій, покращує протікання основних процесів вищої нервової діяльності.

Збільшення освітленості сприяє поліпшенню працездатності навіть в тих випадках, коли процес праці практично не залежить від зорового сприйняття. При поганому освітленні людина швидко втомлюється, працює менш продуктивно, виникає потенційна небезпека помилкових дій і нещасних випадків.

У проєкті, що розробляється, передбачається використовувати суміщене освітлення. У світлий час доби використовуватиметься природне освітлення приміщення через віконні отвори, в решту часу використовуватиметься штучне освітлення. Штучне освітлення створюється газорозрядними лампами.

Штучне освітлення в робочому приміщенні передбачається здійснювати з використанням люмінесцентних джерел світла в світильниках загального

освітлення, оскільки люмінесцентні лампи мають високу потужність (80 Вт), тривалий термін служби (до 10000 годин), спектральний складом випромінюваного світла, близький до сонячного. При експлуатації ЕОМ виконується зорова робота IV в розряді точності (середня точність). При цьому нормована освітленість на робочому місці ( $E_n$ ) рівна 200 лк. Джерелом природного освітлення є сонячне світло. У приміщенні, де розташовані ЕОМ передбачається природне бічне освітлення, рівень якого відповідає ДБН В.2.5-28:2018 [28]. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє ДБН В.2.5-28:2018 [28] і для даного приміщення в світлий час доби достатньо природного освітлення. Світильники загального освітлення розташовуються над робочими поверхнями в рівномірно-прямокутному порядку. Для організації освітлення в темний час доби передбачається обладнати приміщення, довжина якого складає 8 м, ширина 5 м, світильниками ЛПО2П, оснащеними лампами типу ЛБ (дві по 80 Вт) з світловим потоком 5400 лм кожна.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників  $n$  виробляється по формулі (4.3.1):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M} \quad (4.3.1)$$

де  $E$  – нормована освітленість робочої поверхні, визначається нормами – 300лк;

$S$  – освітлювана площа, м<sup>2</sup>;  $S = 24$  м<sup>2</sup>;

$Z$  – поправочний коефіцієнт світильника ( $Z = 1,15$  для ламп розжарювання та ДРЛ;  $Z = 1,1$  для люмінесцентних ламп) приймаємо рівним 1,1;

$K$  – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;



$U$  – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575;

$M$  – число люмінесцентних ламп в світильнику – 2;

$F$  – світловий потік лампи – 5400лм (для ЛБ-80).

Підставивши числові значення у формулу (4.3.1), отримуємо:

$$n=300*24*1.1*1.55400*0.575*2=1.91$$

Приймаємо освітлювальну установку, яка складається з 2-х світильників, які складаються з двох люмінесцентних ламп загальною потужністю 160 Вт, напругою – 220 В.

#### **4.4 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій**

Застосовують різні електричні захисні засоби від ураження струмом:

- ізолюючі - ізолюють людини від струмоведучих або заземлених частин, а так-же від землі;
- основні - володіють ізоляцією, здатної довго витримувати робочу напругу електроустановки і тому ними дозволяється стосуватися струмоведучих частин, що знаходяться під напругою;
- запобіжні - володіють ізоляцією нездатною витримати робоча напруга електроустановки, і тому вони не можуть самостійно захищати людину від ураження струмом цим напругою. Їх значення - посилити захисні дії основних і ізолюючих засобів, разом з якими вони повинні застосовуватися, причому при використанні основних захисних засобів достатньо застосування одного заходи захисного засобу.

Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).

Відповідно до класифікації приміщень за ступенем небезпеки ураження електричним струмом [29], приміщення в якому проводяться всі роботи

належить до першого класу (без підвищеної небезпеки). Під час роботи використовуються електроустановки з напругою живлення 36 В, 220 В, і 360 В. Опір контуру заземлення повинен мати не більше 4 Ом.

Розрахунок проводять за допомогою методу коефіцієнта використання (екранування) електродів. Коефіцієнт використання групового заземлення  $\eta$  - це ставлення чинної провідності цього заземлення до найбільш можливої його провідності при нескінченно великих відстаней між його електродами. Коефіцієнт використання вертикальних заземлювачів  $\eta_v$  у залежності від розміщення заземлювачів і їх кількості знаходиться в межах 0,4 ... 0,99. Взаємну екрануючого дії горизонтального заземлювача (сполучної смуги) враховують за допомогою коефіцієнта використання горизонтального заземлювача  $\eta_c$ .

Послідовність розрахунку:

Визначається необхідний опір штучних заземлювачів  $R_{шт.з.}$ :

$$R_{шт.з.} = \frac{R_d \cdot R_{пр.з.}}{R_{пр.з.} - R_d}, \quad (4.4.1)$$

де  $R_{пр.з.}$  – опір природних заземлювачів;  $R_d$  – допустимий опір заземлення. Якщо природні заземлювачі відсутні, то  $R_{шт.з.} = R_d$ .

Підставивши числові значення в формулу (4.4.2), отримуємо:

$$R_{шт.з.} = \frac{4 \cdot 40}{40 - 4} \approx 4 \text{ Ом.} \quad (4.4.2)$$

Опір заземлення в значній мірі залежить від питомого опору ґрунту  $\rho$ , Ом • м. Приблизне значення питомої опору глини приймаємо  $\rho = 40$  Ом • м (табличне значення).

Розрахункова питомий опір ґрунту,  $\rho_{розр.}$ , Ом • м, визначається відповідно для вертикальних заземлювачів  $\rho_{розр.в.}$ , і горизонтальних  $\rho_{розр.г.}$ , Ом • м по формулі:

$$\rho_{\text{розр.}} = \psi \cdot \rho, \quad (4.4.3)$$

де  $\psi$  – коефіцієнт сезонності для вертикальних заземлювачів і кліматичної зони з нормальною вологістю землі, приймається для вертикальних заземлювачів  $\rho_{\text{розр.в}}=1,7$  і горизонтальних  $\rho_{\text{розр.г}}=5,5$  Ом•м.

$$\rho_{\text{розр.в}} = 1,7 \cdot 40 = 68 \text{ Ом} \cdot \text{м}$$

$$\rho_{\text{розр.г}} = 5,5 \cdot 40 = 220 \text{ Ом} \cdot \text{м}$$

Розраховується опір розтікання струму вертикального заземлення  $R_{\text{в}}$ , Ом, по (4.4.4).

$$R_{\text{в}} = \frac{\rho_{\text{розр.в}}}{2 \cdot \pi \cdot l_{\text{в}}} \cdot \left( \ln \frac{2 \cdot l_{\text{в}}}{d_{\text{ст}}} + \frac{1}{2} \cdot \ln \frac{4 \cdot t + l_{\text{в}}}{4 \cdot t - l_{\text{в}}} \right), \quad (4.4.4)$$

де  $l_{\text{в}}$  – довжина вертикального заземлювача (для труб - 2–3 м;  $l_{\text{в}}=3$  м);  $d_{\text{ст}}$  – діаметр стрижня (для труб - 0,03–0,05 м;  $d_{\text{ст}}=0,05$  м);  $t$  – відстань від поверхні землі до середини заземлювача, яка визначається за формулою (4.4.5):

$$t = h_{\text{в}} + \frac{l_{\text{в}}}{2}, \quad (4.4.5)$$

де  $h_{\text{в}}$  – глибина закладення вертикальних заземлювачів (0,8 м); тоді

$$t = 0,8 + \frac{3}{2} = 2,3$$

$$R_{\text{в}} = \frac{68}{2 \cdot \pi \cdot 3} \cdot \left( \ln \frac{2 \cdot 3}{0,05} + \frac{1}{2} \cdot \ln \frac{4 \cdot 2,3 + 3}{4 \cdot 2,3 - 3} \right) = 18,5 \text{ Ом}$$

Визначається теоретична кількість вертикальних заземлювачів  $n$  штук, без урахування коефіцієнта використання  $\eta_b$ :

$$n = \frac{2 \cdot R_b}{R_d} = \frac{2 \cdot 18,5}{4} = 9,25$$

$\Gamma$  визначається коефіцієнт використання вертикальних електродів групового заземлення без урахування впливу сполучної стрічки  $\eta_b = 0,57$  (табличне значення).

Визначається необхідна кількість вертикальних заземлювачів з урахуванням коефіцієнта використання  $n_b$ , шт:

$$n_b = \frac{2 \cdot R_b}{R_d \cdot \eta_b} = \frac{2 \cdot 18,5}{4 \cdot 0,57} = 16,2 \approx 16$$

Визначається довжина сполучної стрічки горизонтального заземлювача  $l_c$ , м:

$$l_c = 1,05 \cdot L_b \cdot (n_b - 1), \quad (4.4.6)$$

де  $L_b$  – відстань між вертикальними заземлювачами, (прийняти  $L_b = 3$  м);  
 $n_b$  – необхідну кількість вертикальних заземлювачів.

$$l_c = 1,05 \cdot 3 \cdot (16 - 1) \approx 48$$

Визначається опір розтіканню струму горизонтального заземлювача (сполучної стрічки)  $R_r$ , Ом:

$$R_r = \frac{\rho_{\text{розр.г}}}{2 \cdot \pi \cdot l_c} \cdot \ln \frac{2 \cdot l_c^2}{d_{\text{см}} \cdot h_r}, \quad (4.4.7)$$

де  $d_{см}$  – еквівалентний діаметр смуги шириною  $b$ ,  $d_{см} = 0,95b$ ,  $b = 0,15$  м;  
 $h_r$  – глибина закладення горизонтальних заземлювачів (0,5 м);  $l_c$  - довжина  
 сполучної стрічки горизонтального заземлювача  $l_c$ , м

$$R_r = \frac{220}{2 \cdot \pi \cdot 48} \cdot \ln \frac{2 \cdot 48^2}{0,95 \cdot 0,15 \cdot 0,5} = 8,1$$

Визначається коефіцієнт використання горизонтального заземлювача  $\eta_c$  відповідно до необхідної кількості вертикальних заземлювачів пв.

Коефіцієнт використання сполучної смуги  $\eta_c = 0,3$  (табличне значення).

Розраховується результуючий опір заземлюючого електрода з урахуванням сполучної смуги:

$$R_{зар} = \frac{R_b \cdot R_r}{R_b \cdot \eta_c + R_r \cdot n_b \cdot \eta_b} \leq R_d. \quad (4.4.8)$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі [29-30], так як виконується умова:  $R_{зар} < 4$  Ом, а саме:

$$R_{зар} = \frac{18,5 \cdot 8,1}{18,5 \cdot 0,3 + 8,1 \cdot 16 \cdot 0,57} = 1,9 \leq R_d$$

При виникненні пожеж при роботі на ПЕОМ від таких можливими джерел запалювання як: іскри і дуги коротких замикань; перегрів провідників, резисторів і інших радіодеталей ПЕОМ [29-31], від тривалого перевантаження і наявність перехідного опору; іскри при розмиканні і розмиканні ланцюгів; розряди статичної електрики; необережне поводження з вогнем, а також вибухи газоповітряних і пароповітряних сумішей.

Важливу увагу слід звернути на пожежну безпеку підприємства в цілому і окремих його приміщень. У приміщеннях не повинно накопичуватися сміття,

непотрібну папір, мотлох та ін. Речі, які не використовуються у виробничому процесі. Наявний вільний аварійний вихід за межі приміщення в разі пожежі, бути передбачені вогнегасники. Вони повинні бути в робочому стані і перевірятися відповідно до норм. У приміщеннях повинна бути пожежна сигналізація, вогнегасник. У разі виникнення пожежі необхідно повідомити в найближчу пожежну частину, убезпечити інших працівників і по можливості прийняти кроки щодо запобігання можливих наслідків та усунення пожежі.

#### **4.5 Охорона навколишнього природного середовища. Загальні дані з охорони навколишнього природного середовища**

Діяльність за темою магістерської роботи в процесі її виконання впливає на навколишнє природне середовище і регламентується нормами діючого законодавства [34,37].

Основним екологічним аспектом в процесі діяльності за даними спеціальностями є процеси впливу на атмосферне повітря та процеси поводження з відходами, які утворюються, збираються, розміщуються, передаються на видалення (знешкодження), утилізацію, тощо в ІТ галузі.

В приміщенні немає впливу на атмосферне повітря при нормальних умовах праці, бо не використовуються сканери, принтери та інші джерела викиду забруднюючих речовин в повітря робочої зони. В процесі діяльності користувача виникають процеси поводження з відходами ІТ галузі. Види відходів, утворення, яких можливо:

- відпрацьовані люмінесцентні лампи - I клас небезпеки;
- батарейки та акумулятори (малі) -III клас небезпеки;
- змінні носії інформації - IV клас небезпеки;
- відпрацьований ізолюючий матеріал, дроти та кабелі - IV клас небезпеки;
- макулатура - IV клас небезпеки;
- побутові відходи - IV клас небезпеки.

#### 4.6 Висновки до розділу 4

В результаті проведеної роботи було зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Було визначено параметри і певні характеристики приміщення для роботи над запропонованим проектом написаному в кваліфікаційній роботі, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника.

Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Було наведено значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці робітника, а також – наведені інструкції з охорони праці, техніки безпеки при роботі на комп'ютері. Проведено аналіз впливу на навколишнє природне середовище під час виконання магістерської роботи.

## ВИСНОВКИ

Оскільки з підвищенням рівня урбанізації все більш складним завданням, особливо в великих містах, стає проблема пошуку місця для паркування, то реалізовано систему автоматизації транспортної логістики. Це дає оптимальне рішення щодо оптимального маршруту до вільного центру паркування на найкоротшій відстані від водія, виключаючи ділянки дороги, де здійснюється максимальний рух транспорту.

В роботі розглянуто алгоритми транспортної логістики, а саме алгоритм пошуку максимально можливої пропускної здатності мережі, зокрема алгоритм Форда-Фалкерсона, та алгоритми пошуку найкоротшої відстані, такі як алгоритм Дейкстри, алгоритм Флойда — Уоршелла та алгоритм Суурбалле. Проведено тестування алгоритмів, для цього попередньо визначено транспортну мережу існуючих центрів паркування міста Сєверодонецьк. Оскільки в ході тестування визначено, що час виконання алгоритму Дейкстри швидше для всіх пошукових відстаней, то саме цей алгоритм інтегрується з алгоритмом пошуку максимально можливої пропускної здатності мережі.

Розроблена система автоматизації транспортної логістики використовується для моніторингу та інформування про стан доступності кожного окремого центру паркування, а також здійснює управління місцем паркування, як безпосередньо власником автотранспорту, так і власником центру.



## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Dutzik T. A New Way to Go. The Transportation Apps and Vehicle-sharing. Tools that Are Giving More Americans. The Freedom to Drive Less / T. Dutzik, T. Madsen, P. Baxandall // U.S. PIRG Education Fund, 2013.
2. Siuhi S. Opportunities and challenges of smart mobile applications in transportation / S. Siuhi, J. Mwakalonge // Journal of traffic and transportation engineering (english edition), 2016. – No. 3 (6). – P. 582 - 592.
3. National Highway Transportation Safety Administration (NHTSA), 2013. Safercar. Available at: <http://www.safercar.gov/>
4. Tennessee Department of Transportation (TDOT), 2014. TDOT SmartWay Mobile App. Available at: <http://www.tdot.state.tn.us/tdotsmartway/mobile/>
5. Ji Y. Understanding drivers' perspective on parking guidance information / Y. Ji, W. Guo, P. Blythe, D. Tang, W. Wang // IET Intell. Transp. Syst., 2014. – Vol. 8, no. 4. – P. 398–406.
6. Rajabioun T. On-street and off-street parking availability prediction using multivariate spatiotemporal models / T. Rajabioun, P. Ioannou // IEEE Trans. Intell. Transp. Syst., 2015. – Vol. 16, no. 5. – P. 2913–2924.
7. Harshal V. Parking Guidance system using Internet of Things / V. Harshal, G. Mulay, V. Gohokar // International Conference on Inventive Computation Technologies, 2016. – P. 518-523.
8. Abdullah N. Using Ford-Fulkerson Algorithm and Max Flow-Min Cut Theorem to Minimize Traffic Congestion in Kota Kinabalu, Sabah / N. Abdullah, T. K. Hua // Journal of Information System and Technology Management (JISTM), 2017. – № 2(4). – P. 18-34.
9. Yao B. Path Optimization Algorithms Based on Graph Theory / B. Yao, J. Yin, H. Zhou, W. Wu // International Journal of Grid and Distributed Computing, 2016. – Vol. 9, No. 6. – P. 137-148.

10. Zhang M. Decision support approach for integrated maintenance program of urban rail transit // *International Journal of Computing*. – 2017. - Volume. 16, Issue. 3. - pp. 143–151.
11. Lin C. A genetic algorithm approach for finding the shortest driving time on mobile devices / C.-H. Lin, J.-L. Yu, J.-C. Liu, C.-J. Lee // *Scientific research and essays*. – 2011. – Vol. 6(2). – pp. 394-405.
12. Derkach M. Parking Guide Service for Large Urban Areas / M. Derkach, V. Lysak, I. Skarga-Bandurova, I. Kotsiuba // *10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*. - Metz, France, 2019. – pp. 567 – 571.
13. Деркач М.В. Розробка мобільного додатку для приватних центрів паркування / М.В. Деркач, В.В. Лисак, І.С. Скарга-Бандурова // *Вісник Східноукраїнського національного університету імені Володимира Даля*. – Северодонецьк: СХУ ім.В.Даля, 2018. – № 6 (247). – С.38-41.
14. Деркач М.В. Створення сервісу для підвищення якості наданих послуг центрів паркування / М.В. Деркач, В.В. Лисак // *ІТ-Ідея – 2018: збірник науково-практичних праць*. – Северодонецьк: СХУ ім.В.Даля, 2018. – С.67–69.
15. Lysak V. Interaction Golang with PostgreSQL database / V. Lysak, M. Derkach // *Advanced Technologies in Research and Education: collection of research materials of the Second International Conference*. – Severodonetsk: Publishing House of the Volodymyr Dahl East Ukrainian National University, 2019. – P. 96-98.
16. Lysak V. Implementation of Dijkstra's algorithm for Golang / V. Lysak, M. Derkach // *Advanced Technologies in Research and Education: collection of research materials of the Third International Conference*. – Severodonetsk: Publishing House of the Volodymyr Dahl East Ukrainian National University, 2020.
17. Лисак В.В. Інтеграція алгоритмів транспортної логістики для сервісу паркування / В.В. Лисак, М.В. Деркач, І.С. Скарга-Бандурова // *Вісник Східноукраїнського національного університету імені Володимира Даля*. – Северодонецьк: СХУ ім.В.Даля, 2020. – № 8 (264).

18. Закон України Про охорону праці;. Вводиться в дію Постановою ВР №2695-ХІІ від 14.10.92, ВВР, 1992, №49, ст.669. - Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/2694-12](http://www.url:https://zakon.rada.gov.ua/laws/show/2694-12)

19. Кодекс законів про працю України. Затверджується Законом № 322-VIII від 10.12.71 ВВР, 1971. Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/322-083](http://www.url:https://zakon.rada.gov.ua/laws/show/322-083). Закон України; Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності;

20. Наказ від 21 грудня 2000 року N 2180-III. Режим доступу: [www. URL: https://dnaop.com/html/2065/doc-zakon-ukrajini-pro-zagalynoobovjazkove-derzhavne-socialyne-strahuvannya-vid-neshhasnogo-vipadku-na-virobnictvi-ta-profesijnogo-z](http://www.url:https://dnaop.com/html/2065/doc-zakon-ukrajini-pro-zagalynoobovjazkove-derzhavne-socialyne-strahuvannya-vid-neshhasnogo-vipadku-na-virobnictvi-ta-profesijnogo-z)

21. Про затвердження Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці НПАОП 0.00-4.12-05. Наказ від 26.01.2005 №15. Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/z0231-05](http://www.url:https://zakon.rada.gov.ua/laws/show/z0231-05)

22. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПІН 3.3.2.007-98. Затверджено Постановою Головного державного санітарного лікаря України 10 грудня 1998 р. N 7. Режим доступу: [www. URL: https://zakon.rada.gov.ua/rada/show/v0007282-98](http://www.url:https://zakon.rada.gov.ua/rada/show/v0007282-98)

23. Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99. Постанова N 42 від 01.12.99. Режим доступу: [www. URL: https://zakon.rada.gov.ua/rada/show/va042282-99](http://www.url:https://zakon.rada.gov.ua/rada/show/va042282-99)

24. Електробезпека в будівлях і спорудах. Вимоги до захисних заходів від ураження електричним струмом. Наказ від 1 липня 2016 року N 204. Режим доступу: [www. URL: http://epicentre.co.ua/dstu/doc28522.html](http://www.url:http://epicentre.co.ua/dstu/doc28522.html)

25. ДБН В.2.5-28:2018 «Природне і штучне освітлення». Режим доступу: [www. URL: http://www.minregion.gov.ua/wp-content/uploads/2018/12/V2528-1.pdf](http://www.minregion.gov.ua/wp-content/uploads/2018/12/V2528-1.pdf)

26. НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями». Зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за №508/31960. Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/z0508-18](https://zakon.rada.gov.ua/laws/show/z0508-18)

27. ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою». Наказ від 15.06.2016 №158. Режим доступу: [www. URL: https://zakon.rada.gov.ua/rada/show/v0158858-16](https://zakon.rada.gov.ua/rada/show/v0158858-16)

28. Закон України «Про охорону навколишнього природного середовища». Вводиться в дію Постановою ВР № 1268-ХІІ від 26.06.91, ВВР, 1991, № 41, ст.547. Режим доступу: [www. URL:https://zakon.rada.gov.ua/laws/show/1264-12](https://zakon.rada.gov.ua/laws/show/1264-12)

29. Закони України «Про охорону навколишнього природного середовища». Вводиться в дію Постановою ВР № 4005-ХІІ від 24.02.94, ВВР, 1994, № 27, ст.219. Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/4004-12](https://zakon.rada.gov.ua/laws/show/4004-12)

30. Закон України «Про відходи». Відомості Верховної Ради України (ВВР), 1998, № 36-37, ст.242. Режим доступу: [www. URL: https://zakon.rada.gov.ua/laws/show/187/98-вр](https://zakon.rada.gov.ua/laws/show/187/98-вр)

## ДОДАТОК А

### Слайди презентації



Рисунок А.1 – Слайд 1

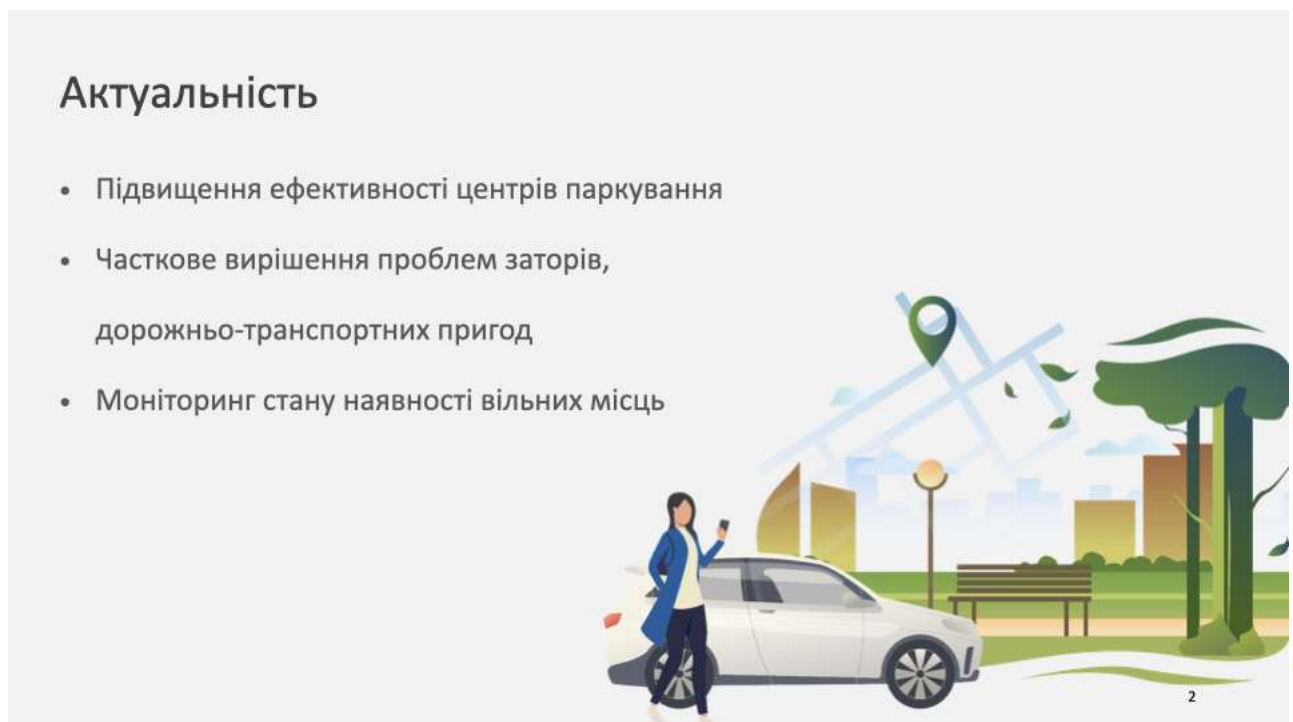


Рисунок А.2 – Слайд 2

## Постановка задачі

**Метою** магістерської роботи є дослідження алгоритмів транспортної логістики та практична реалізація пошуку оптимального маршруту до вільного центру паркування.

**Об'єкт** - інтеграція алгоритмів транспортної логістики для пошуку оптимального маршруту.

**Предмет** - технологія IoT для забезпечення зв'язку між центром паркування та автовласниками.



3

Рисунок А.3 – Слайд 3

Поставлені та вирішені наступні **задачі**:

- 1) проаналізувати стан впровадження сучасних інформаційних технологій в транспортний сектор країни;
- 2) проаналізувати архітектуру, задачі, переваги, недоліки існуючих рішень щодо автоматизації транспортної логістики;

3) дослідити алгоритми транспортної логістики;

4) реалізувати пошук оптимального маршруту в сервісі



4

Рисунок А.4 – Слайд 4

## Зв'язок роботи з науковими програмами, темами

Дослідження, результати яких викладені в магістерській роботі, проводилися у Східноукраїнському національному університеті імені Володимира Даля відповідно до державних програм і планів НДР, а також міжнародного проекту:

- Проект Європейського Союзу ERASMUS+ ALIOT 573818-EPP-1-2016-1-UK-EPPKA2-SBHE-JP «Internet of Things: Emerging Curriculum for Industry and Human Applications» (2016-2019 рр.);
- НДР «Наукові основи, методи і засоби розгортання інтелектуальних транспортних систем для моніторингу міського транспорту та підтримки пасажирів» (Східноукраїнський національний університет імені Володимира Даля, ДР 0117U005537, 2017-2020 рр.).

5

Рисунок А.5 – Слайд 5

## Апробація результатів наукових досліджень

За темою роботи опубліковано 6 наукових праць, серед яких 1 англomовна праця включена до бази даних Scopus, 2 статті у науковому фаховому виданні України; 3 тез в збірниках матеріалів науково-практичних конференцій.

Результати наукових досліджень представлено на всеукраїнських і міжнародних науково-практичних конференціях:

- ✓ «ІТ-Ідея» (Сєверодонецьк, 2018);
- ✓ «10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)» (Metz, France, 2019);
- ✓ «Advanced Technologies in Research and Education» (Сєверодонецьк-Каїр, 2019, 2020).



6

Рисунок А.6 – Слайд 6



Рисунок А.7 – Слайд 7

### Алгоритми пошуку коротшого шляху



**Эдсгер Вібе Дейкстра**



**Роберт В Флойд**  
алг. Алгоритм Флойда – Уоршелла



**Джон Суурбалле**

8

Рисунок А.8 – Слайд 8





Рисунок А.9 – Слайд 9

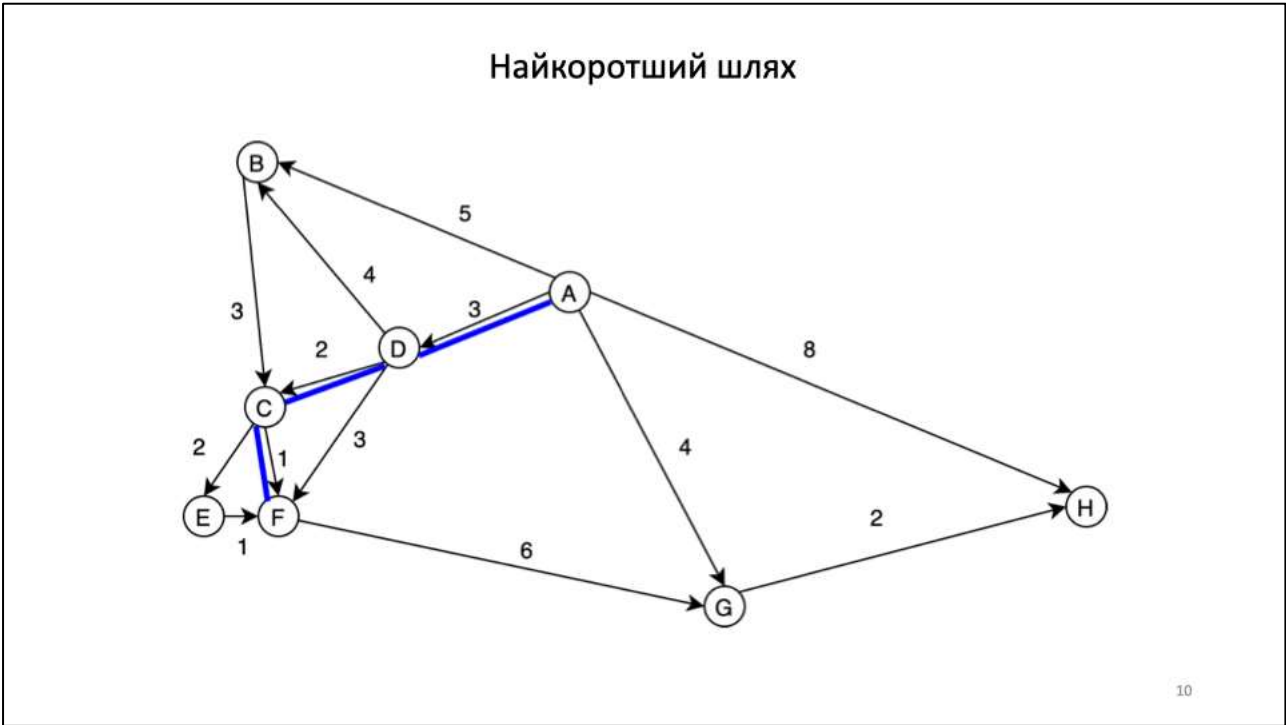


Рисунок А.10 – Слайд 10

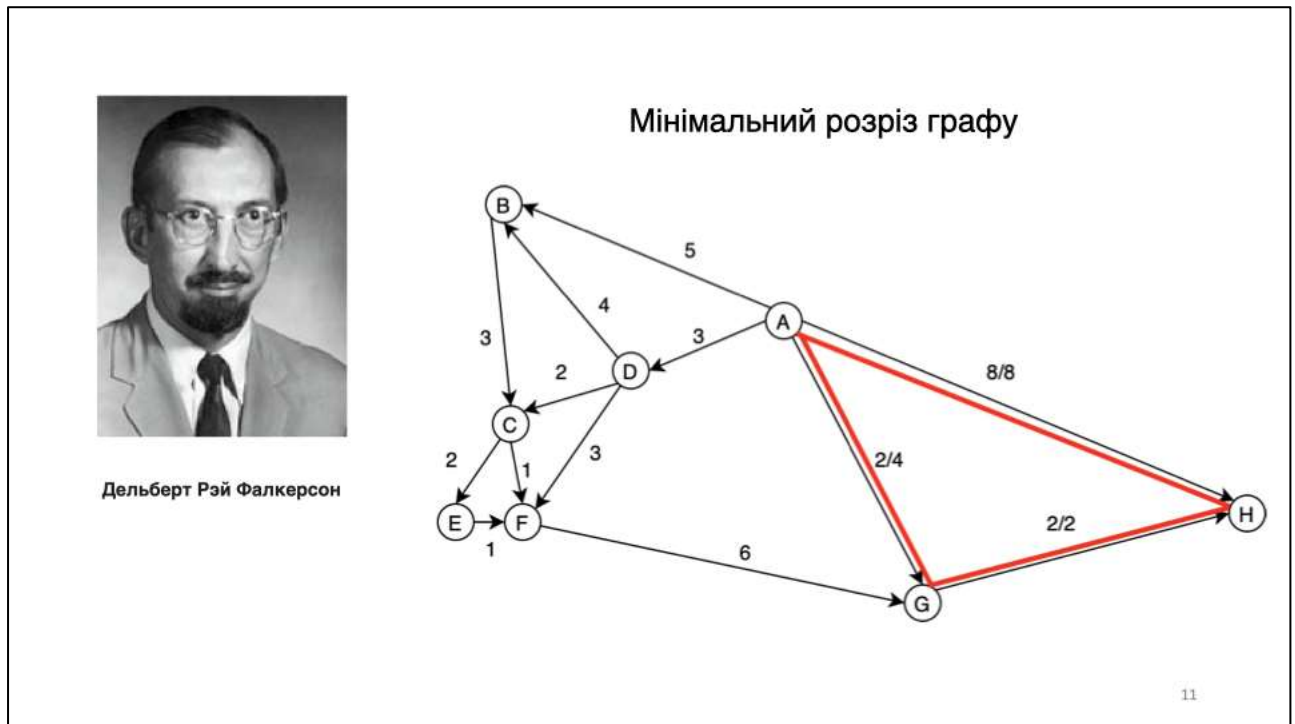


Рисунок А.11 – Слайд 11

Інтеграція алгоритмів транспортної логістики

Для транспортної системи міста застосовано графову модель:  $G=(V,E)$ , з множиною вузлів  $v \in V$ , з множиною дуг  $e \in E$ , де  $v$  - кількість вузлів,  $e$  - кількість дуг,  $|\cdot|$  - знак потужності множини.

Результат роботи алгоритму Дейкстри виражається наступною множиною вузлів:

$$vD=\{A, D, C, F\},$$

а результат роботи алгоритму Форда-Фалкерсона:

$$vFF=\{A, G, H\}.$$

Початковий вузол позначимо  $v_0$ , для розглянутого графу  $v_0=\{A\}$ .

Тоді для того, щоб отримати оптимальне рішення щодо транспортного шляху, використовуємо вираз:

$$(vD \cap vFF) \setminus v_0 = \emptyset.$$

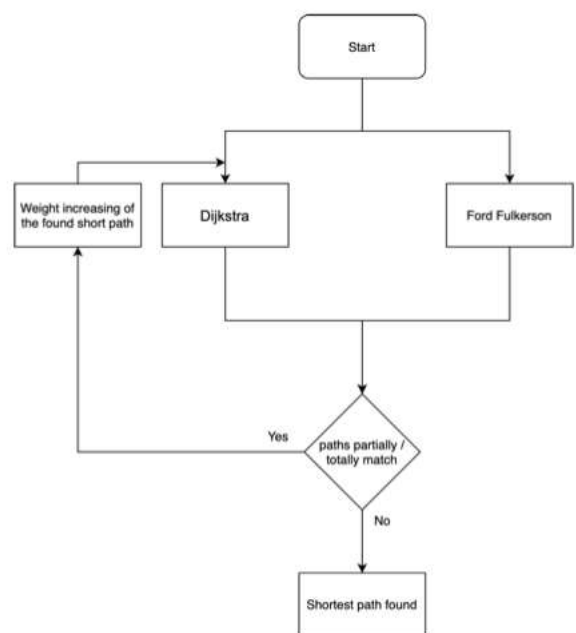


Рисунок А.12 – Слайд 12




Рисунок А.13 – Слайд 13



Рисунок А.14 – Слайд 14

## Можливості системи автоматизації транспортної логістики


- Реєструвати центри
- Переглядати стан місць
- Орендувати місця паркування
- Отримувати коротшій шлях до місця




15


Рисунок А.15 – Слайд 15

## Швидкий пошук паркомісця та коротшого шляху до нього





**зайнято**



**вільно**

**Інтуїтивно зрозумілий стан місць**

16

Рисунок А.16 – Слайд 16

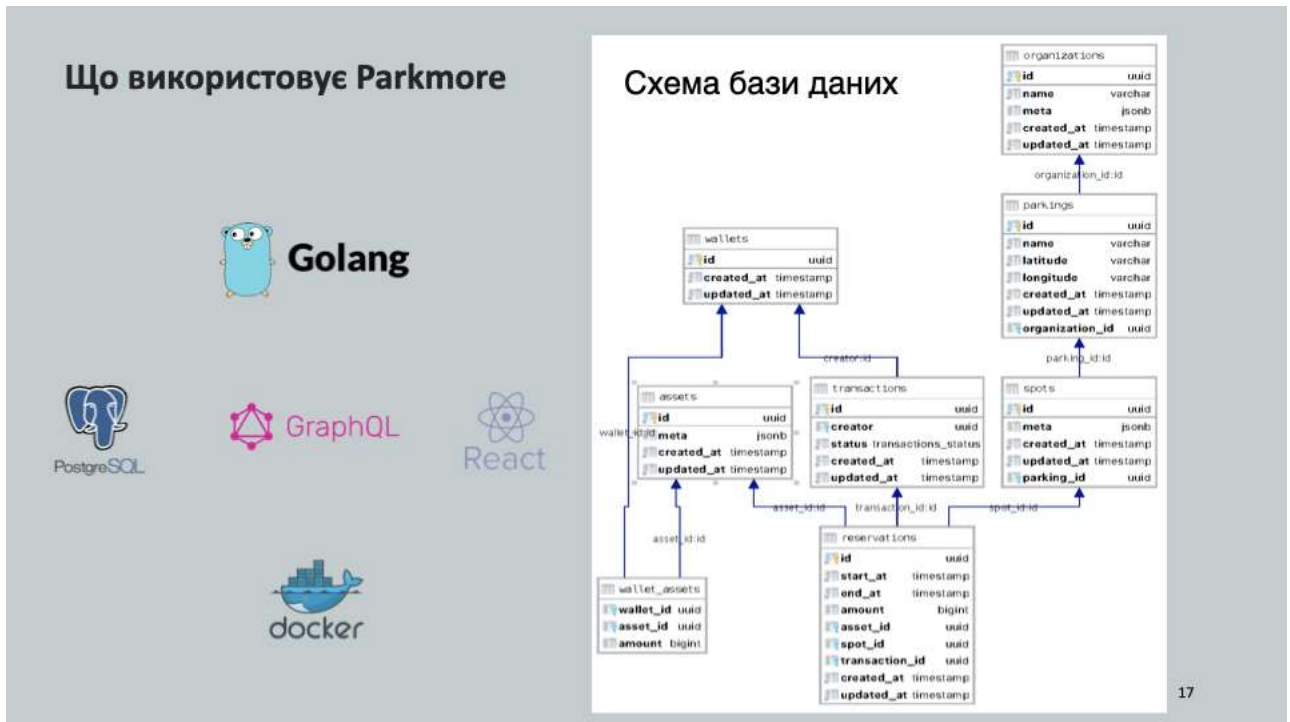


Рисунок А.17 – Слайд 17



Рисунок А.18 – Слайд 18

## ДОДАТОК Б

## Алгоритм Форда-Фалкерсона

```

1 func (gr *Graph) MinCutMaxFlow(orig, dest uint64, precision
2 float64) (maxFlowMinCut float64, flows
3 map[uint64]map[uint64]float64, cut []*Edge) {
4     // This map will contain the reverse edge relations
5     undirEdges := make(map[uint64][]uint64)
6     for f, dests := range gr.VertexEdges {
7         for t := range dests {
8             if _, ok := undirEdges[f]; !ok {
9                 undirEdges[f] = []uint64{t}
10            } else {
11                undirEdges[f] = append(undirEdges[f], t)
12            }
13            if _, ok := undirEdges[t]; !ok {
14                undirEdges[t] = []uint64{f}
15            } else {
16                undirEdges[t] = append(undirEdges[t], f)
17            }
18        }
19    }
20
21    flows = make(map[uint64]map[uint64]float64)
22    for f, vertices := range gr.VertexEdges {
23        for v := range vertices {
24            if _, ok := flows[f]; ok {
25                flows[f][v] = 0.0
26            } else {
27                flows[f] = map[uint64]float64{v: 0.0}
28            }
29        }
30    }
31    lastFlow := -1 - precision
32    maxFlowMinCut = 0.0
33    for maxFlowMinCut-lastFlow > precision {
34        gr.maxFlow(orig, dest, undirEdges, flows,
35 map[uint64]bool{orig: true}, []uint64{orig})
36
37        lastFlow = maxFlowMinCut
38        maxFlowMinCut = 0.0
39        for _, flow := range flows[orig] {
40            maxFlowMinCut += flow
41        }
42    }
43
44    queue := []uint64{orig}

```

```

45     cut = []*Edge{}
46     visited := map[uint64]bool{orig: true}
47     for len(queue) > 0 {
48         v := queue[0]
49         queue = queue[1:]
50         for t, w := range gr.VertexEdges[v] {
51             if _, ok := visited[t]; !ok && flows[v][t] != w
52         {
53             queue = append(queue, t)
54             visited[t] = true
55         }
56     }
57 }
58 for f, dests := range gr.VertexEdges {
59     for t, w := range dests {
60         _, fromVisit := visited[f]
61         _, toVisit := visited[t]
62         if fromVisit && !toVisit && flows[f][t] == w {
63             cut = append(cut, &Edge{
64                 From:    f,
65                 To:      t,
66                 Weight: w,
67             })
68         }
69     }
70 }
71
72 return
73 }
```

## ДОДАТОК В

### Алгоритм Дейкстри

```

1 func (gr *Graph) ShortestPath(origin, dest uint64) (path
2 []uint64, dist map[uint64]Distance) {
3     dist = make(map[uint64]Distance)
4     if !gr.NegEdges {
5         // We can use Dijkstra :)
6         queue := fibheap.New()
7         current := origin
8         dist[current] = Distance{
9             From: current,
10            Dist: 0,
11        }
12        ended := false
13        for !ended && current != dest {
14            for t, w := range gr.VertexEdges[current] {
15                if _, defined := dist[t]; !defined ||
16                dist[current].Dist+w < dist[t].Dist {
17
18                queue.DecreaseScore(dist[current].Dist+w, t)
19                    dist[t] = Distance{
20                        From: current,
21                        Dist: dist[current].Dist + w,
22                    }
23                }
24            }
25            _, currentT := queue.Min()
26            if currentT != nil {
27                current = currentT.(uint64)
28            } else {
29                ended = true
30            }
31        }
32    } else {
33        // Bellman-Ford implementation
34        dist = map[uint64]Distance{origin: Distance{
35            From: origin,
36            Dist: 0,
37        }}
38        queue := []uint64{origin}
39        for len(queue) > 0 {
40            visited := make(map[uint64]bool)
41            for t, w := range gr.VertexEdges[queue[0]] {
42                if _, ok := dist[t]; !ok || dist[t].Dist >
43                dist[queue[0]].Dist+w {
44                    dist[t] = Distance{

```



```

45         From: queue[0],
46         Dist: dist[queue[0]].Dist + w,
47     }
48     if _, ok := visited[t]; !ok {
49         visited[t] = true
50         queue = append(queue, t)
51     }
52 }
53 }
54 queue = queue[1:]
55 }
56 }
57
58 path = []uint64{dest}
59 node := dist[dest]
60 if _, ok := dist[dest]; ok {
61     for node.From != origin {
62         path = append(path, node.From)
63         node = dist[node.From]
64     }
65     path = append(path, origin)
66     for i := 0; i < len(path)/2; i++ {
67         path[i] ^= path[len(path)-i-1]
68         path[len(path)-i-1] ^= path[i]
69         path[i] ^= path[len(path)-i-1]
70     }
71 }
72
73 return
74 }

```

## ДОДАТОК Г

### Міграції бази даних

```
1 create table organizations
2 (
3     id uuid not null constraint organizations_pk primary key,
4     name      varchar      not null,
5     meta      jsonb,
6     created_at timestamp    not null,
7     updated_at timestamp    not null
8 );
9
10 create unique index organizations_id_uindex on organizations
11 (id);
12
13 create table parkings
14 (
15     id uuid not null constraint parkings_pk primary key,
16     name      varchar      not null,
17     latitude  varchar      not null,
18     longitude varchar      not null,
19     created_at timestamp    not null,
20     updated_at timestamp    not null,
21     organization_id uuid references organizations (id)
22 );
23
24 create unique index parkings_id_uindex on parkings (id);
25
26 create table spots
27 (
28     id uuid not null constraint spots_pk primary key,
29     meta      jsonb,
30     created_at timestamp    not null,
31     updated_at timestamp    not null,
32     parking_id uuid references parkings (id)
33 );
34
35 create unique index spots_id_uindex on spots (id);
36
37 create table wallets
38 (
39     id uuid not null constraint wallets_pk primary key,
40     created_at timestamp    not null,
41     updated_at timestamp    not null
42 );
43
44 create unique index wallets_id_uindex on wallets (id);
45
```

```
46 create table assets
47 (
48     id uuid not null constraint assets_pk primary key,
49     meta jsonb,
50     created_at timestamp    not null,
51     updated_at timestamp    not null
52 );
53
54 create unique index assets_id_uindex on assets (id);
55
56 create type transactions_status as enum ('Reserved',
57 'Canceled');
58
59 create table transactions
60 (
61     id uuid not null constraint transactions_pk primary key,
62     creator uuid references wallets (id),
63     status transactions_status not null,
64     created_at timestamp    not null,
65     updated_at timestamp    not null
66 );
67
68 create unique index transactions_id_uindex on transactions
69 (id);
70
71 create table reservations
72 (
73     id uuid not null constraint reservations_pk primary key,
74     start_at timestamp    not null,
75     end_at timestamp    not null,
76     amount bigint,
77     asset_id uuid references assets (id),
78     spot_id uuid references spots (id),
79     transaction_id uuid references transactions (id),
80     created_at timestamp    not null,
81     updated_at timestamp    not null
82 );
83
84 create unique index reservations_id_uindex on reservations
85 (id);
86
87 create table wallet_assets (
88     wallet_id uuid references wallets (id),
89     asset_id uuid references assets (id),
90     amount bigint
91 )
```