

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет інформаційних технологій

Кафедра технологій управління

Спеціальність 122 «Комп'ютерні науки»

Освітньо-наукова програма «Управління проєктами»

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему:

**“Дослідження процесів управління проєктом розробки і впровадження
системи обліку міської транспортної інфраструктури”**

Студента 2-го курсу групи УП-21

Семчишина Віталія Руслановича

Науковий керівник:

кандидат техн. наук, професор
кафедри технологій управління

Морозов Віктор Володимирович

(підпис студента)

(дата)

(підпис)

Попередній захист:

(Висновок: “До захисту в Екзаменаційній комісії”)

Завідувач кафедри
технологій управління

(підпис)

Морозов В.В.

(прізвище, ініціали)

(дата)

Київ – 2026

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
Факультет інформаційних технологій**

Кафедра технологій управління
Освітній рівень Магістр
Спеціальність 122 Комп'ютерні науки
Освітня програма Управління проєктами

ЗАТВЕРДЖУЮ

Завідувач кафедри
професор Віктор МОРОЗОВ

“08” грудня 2025 року

**ЗАВДАННЯ
НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ**

Студент: Семчишин Віталій Русланович

Група: УП-21

1. Тема кваліфікаційної роботи: «Дослідження процесів управління проєктом розробки і впровадження системи обліку міської транспортної інфраструктури». Затверджена протоколом від №4 від «15» грудня.2025 р.
2. Строк подання студентом готової роботи - “15” травня 2026 р.
3. Цільова установка та вихідні дані до роботи: дослідження характеристик об'єкта управління та особливостей застосування сучасних методологій, методів і засобів управління проєктами. Робота включає календарне планування, управління змістом, бюджетом та ресурсами, а також оптимізацію управлінських процесів для досягнення поставлених цілей у задані строки та в межах виділених ресурсів.
4. Зміст роботи: анотація, вступ, перший розділ (аналіз ІТ-середовища та існуючих рішень, обґрунтування доцільності проєкту, SWOT та PEST аналіз, формування технічного завдання, кошторисні розрахунки), другий розділ (розробка концептуальних моделей, математична постановка задачі,

оптимізація розподілу бюджету), третій розділ (розробка інформаційного та програмного забезпечення, проектування баз, декомпозиція проєкту, опис модулів), висновки (заклучна частина), перелік використаних інформаційних джерел, додатки.

5. Перелік графічного матеріалу (слайдів): титульна сторінка, мета та завдання дипломної роботи, огляд галузі мобільних додатків у сфері здоров'я, життєвий цикл проєкту, дерево проблем, дерево цілей, економічне обґрунтування, маркетингове дослідження, організаційна структура підприємства, організаційна структура команди проєкту, ієрархічна структура робіт, управління ризиками, календарно-сітьова модель проєкту, використання бюджету, висновки.

6. Календарний план виконання роботи:

№ з/п	Назва частин роботи	%	Виконання роботи	
			За планом	Фактично
1	Вивчення рекомендованої літератури та ознайомлення з методичними матеріалами	3	01.09.2025-15.09.2025	05.09.2025-18.09.2025
2	Збір і вивчення матеріалів досліджуваної теми	5	16.09.2025-30.09.2025	19.09.2025-05.10.2025
3	Складання плану виконання кваліфікаційної роботи магістра	3	01.10.2025-05.10.2025	06.10.2025-08.10.2025
4	Узгодження з науковим керівником плану кваліфікаційної роботи та внесення відповідних змін	2	06.10.2025-10.10.2025	09.10.2025-15.10.2025
5	Підготовка розділу 1	15	11.10.2025-30.11.2025	16.10.2025-25.11.2025
6	Підготовка розділу 2	15	01.12.2025-31.01.2026	26.11.2025-05.02.2026
7	Підготовка розділу 3	15	01.02.2026-31.03.2026	06.02.2026-25.03.2026
8	Підготовка розділу 4	15	01.04.2026-20.04.2026	26.03.2026-15.04.2026
9	Оформлення кваліфікаційної роботи	7	21.04.2026-30.04.2026	16.03.2026-30.04.2026
10	Подання роботи на	3	01.05.2026-	21.04.2026-

	перевірку науковому керівнику, внесення змін		07.05.2026	05.05.2026
11	Подання роботи рецензенту	1	08.05.2026-12.05.2026	06.05.2026-10.05.2026
12	Попередній захист роботи	1	05.05.2026	05.05.2026
13	Захист роботи	2	20.05.2026	20.05.2026

Дата видачі завдання “9” грудня 2025 р.

Керівник роботи:

кандидат техн. наук, професор кафедри технологій управління

Морозов Віктор Володимирович

(підпис)

Завдання прийняв до виконання:

студент групи УП-21 Семчишин Віталій Русланович

(підпис)

ЗМІСТ

АНОТАЦІЯ	7
ТАБЛИЦЯ СКОРОЧЕНЬ ТА ПОЯСНЕНЬ	9
ВСТУП	10
РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТА ОБҐРУНТУВАННЯ ДОЦІЛЬНОСТІ ТА ЖИТТЄЗДАТНОСТІ ПРОЄКТУ	14
1.1 Аналіз ІТ-середовища області обліку транспортної інфраструктури. Порівняння існуючих рішень.....	14
1.2 Обґрунтування доцільності розробки запропонованого рішення	17
1.3 SWOT та PEST аналіз запропонованого рішення.....	19
1.4 Постановка задачі дослідження, формулювання технічного завдання на розробку у вигляді паспорту проєкту	26
1.5 Концепція проєкту	31
РОЗДІЛ 2. МАТЕМАТИЧНА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ	39
2.1 Розробка концептуальних моделей інформаційної системи. Визначення підсистем та надсистем.....	39
2.2 Формалізація математичних моделей та постановка задачі в математичному вигляді	41
2.3 Формалізація задачі оптимального розподілу бюджету	45
2.4 Обґрунтування обраного підходу до ефективного управління проєктами	48
2.5 Створення кошторису проєкту інструментом Altinvest.....	51
РОЗДІЛ 3. РОЗРОБКА ІНФОРМАЦІЙНОГО ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПРОЄКТУ	58
3.1 Розробка концептуальної моделі бази даних проєкту.....	58
3.2 Побудова логічної моделі бази даних проєкту.	61

3.3 Побудова фізичної моделі бази даних з використанням СУБД PostgreSQL.	65
3.4 Декомпозиція проєкту по продуктам.....	68
3.5 Опис модулів ПЗ та їх взаємодії.....	69
3.6 Програмна розробка.....	71
РОЗДІЛ 4. ЗАСТОСУВАННЯ ТЕХНОЛОГІЙ УПРАВЛІННЯ ІТ	
ПРОЄКТОМ	79
4.1 Розробка організаційної структури компанії та проєкту	79
4.2 Життєвий цикл проєкту	83
4.3 Ієрархічна структура робіт проєкту. Матриця відповідальності	85
4.4 Розробка календарного плану. Планування термінів проєкту	90
4.5 Управління ризиками проєкту	95
4.6 Моніторинг відхилень при виконання проєкту	98
ВИСНОВКИ	103
ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ	106

АНОТАЦІЯ

Кваліфікаційної роботи магістра на тему:

«Дослідження процесів управління проєктом розробки і впровадження системи обліку міської транспортної інфраструктури»

Студент: Семчишин Віталій Русланович

Науковий керівник: Морозов Віктор Володимирович

Рік захисту - 2026.

Темою даної роботи було обрано «Дослідження процесів управління проєктом розробки і впровадження системи обліку міської транспортної інфраструктури».

Метою даної роботи є розробка та впровадження автоматизованої інформаційної системи обліку міської транспортної інфраструктури, що дозволить забезпечити централізований збір даних, оптимізувати планування ремонтних робіт та підвищити прозорість розподілу бюджетних коштів за рахунок використання інструментів управління проєктами та геоінформаційного аналізу. *Ціль проєкту* – розробка та впровадження інформаційної системи обліку міської транспортної інфраструктури

Об'єктом дослідження є процеси управління проєктом розробки та впровадження інформаційної системи обліку міської транспортної інфраструктури. *Предметом дослідження* є сукупність управлінських процесів, методів та інструментів планування, реалізації та моніторингу даного проєкту, зокрема процеси управління змістом, часом, вартістю, ресурсами, якістю та проєктними ризиками, що

Наукова новизна роботи полягає у новому підході до управління проєктами впровадження систем обліку міської інфраструктури. Основна перевага запропонованого методу — здатність ефективно поєднувати технічні потреби міста з його фінансовими можливостями, що дозволяє уникнути зайвих витрат та зробити управління інфраструктурою прозорим та прогнозованим.

Практичне значення отриманих результатів. Отримані результати мають високий ступінь готовності до практичного впровадження у муніципальну сферу. Розроблені матеріали можуть бути використані як початкова проєктна документація при формуванні технічних завдань для оголошення тендерів на розробку ІТ-систем у територіальних громадах.

У першому розділі проаналізовано ІТ-середовище сфери обліку міської транспортної інфраструктури, проведено порівняльний аналіз існуючих рішень, здійснено SWOT та PEST аналіз, а також обґрунтовано доцільність проєкту та сформовано технічне завдання у вигляді паспорта проєкту.

У другому розділі розроблено концептуальні моделі інформаційної системи, здійснено математичну формалізацію задачі дослідження, зокрема задачі оптимального розподілу бюджету, а також сформовано кошторис проєкту за допомогою інструментарію Altinvest.

У третьому розділі проведено проєктування інформаційного та програмного забезпечення: розроблено концептуальну, логічну та фізичну моделі бази даних у СУБД PostgreSQL, здійснено декомпозицію проєкту за продуктами та описано архітектуру модулів програмного рішення.

У четвертому розділі розроблено технології управління ІТ-проєктом: сформовано організаційну структуру, ієрархічну структуру робіт (WBS) та матрицю відповідальності, розроблено календарний план, план управління ресурсами та ризиками, а також впроваджено механізми моніторингу відхилень при виконанні проєкту.

За результатами даної роботи було оформлено висновки. Проєкт є актуальним на сьогоднішній день та фінансово вигідним для реалізації.

Структура та обсяг роботи. Кваліфікаційна робота складається з 107 сторінок, 4 розділів, 32 рисунків, 25 таблиць, 40 джерел.

Ключові слова: управління проєктами, інформаційна система, облік елементів міської транспортної інфраструктури, муніципальне управління.

ТАБЛИЦЯ СКОРОЧЕНЬ ТА ПОЯСНЕНЬ

Скорочення	Пояснення
БД	База даних
ІТ	Інформаційні технології
ТЗ	Технічне завдання
ГІС	Геоінформаційна система
КМДА	Київська міська державна адміністрація
СУБД	Система управління базами даних
ІТС	Інтелектуальні транспортні системи
МВП (MVP)	Мінімально життєздатний продукт (Minimum Viable Product)
ERD (ER)	Діаграма зв'язку сутностей (Entity-Relationship Diagram)
EAV	Модель зберігання даних «Сутність-Атрибут-Значення» (Entity-Attribute-Value)
API	Програмний інтерфейс застосунку
SPA	Односторінковий вебзастосунок (Single Page Application)
WBS	Ієрархічна структура робіт (Work Breakdown Structure)
PBS	Структура розподілу продукту (Product Breakdown Structure)
RACI	Матриця розподілу відповідальності (Responsible, Accountable, Consulted, Informed)
KPI	Ключові показники ефективності (Key Performance Indicators)
NPV	Чиста приведена вартість (Net Present Value)
IRR	Внутрішня норма рентабельності (Internal Rate of Return)
PB	Період окупності (Payback Period)

ВСТУП

В умовах стрімкої урбанізації та зростання транспортного навантаження, міста України стикаються з критичними проблемами в управлінні інфраструктурою. На сьогодні більшість процесів управління залишаються недостатньо інформатизованими, а відсутність централізованого обліку стану об'єктів (дорожніх знаків, розмітки, світлофорів, покриття, зупинок тощо) значно ускладнює роботу комунальних служб. Така ситуація призводить до зниження прозорості використання бюджетних коштів та прийняття управлінських рішень, що не завжди базуються на актуальних даних [3,4]. У той же час, світовий досвід впровадження Smart City технологій на основі геоінформаційних систем (ГІС) та IoT-рішень доводить, що автоматизація збору та аналізу даних є ключем до ефективного міського господарства [13, 38].

На цьому тлі виникає потреба у створенні комплексного рішення, яке дозволило б здійснювати ефективний облік елементів транспортної інфраструктури, вести контроль за виконанням ремонтних робіт, аналізувати навантаження на інфраструктуру та формувати пріоритети для бюджетування. Таке рішення має забезпечити підвищення прозорості процесів, підтримку прийняття обґрунтованих управлінських рішень та більш ефективне використання публічних коштів.

Підставою для проведення дослідження є потреба органів місцевого самоврядування у створенні адаптивних інформаційних інструментів. Значущість дослідження обумовлена необхідністю переходу від реактивного управління (виправлення наслідків аварійності) до превентивного, що базується на точних даних. Розв'язання цієї задачі має безпосередній вплив на економічну ефективність міського управління та якість життя мешканців. Проведення даного дослідження є необхідним з огляду на те, що успішність впровадження складних інформаційних систем залежить не лише від технічних рішень (вибору СУБД чи алгоритмів), а й від якості управління

самим проєктом — від формування технічного завдання до моніторингу відхилень [8].

Враховуючи наведену вище інформацію, стає очевидною доцільність та перспективність розробки інформаційної системи для управління процесами обліку міської транспортної інфраструктури. Готовий продукт має стати інструментом, що допоможе відповідальним фахівцям та органам управління формувати комплексний план обслуговування та розвитку об'єктів інфраструктури, забезпечуючи при цьому можливість відстежувати ефективність виконання робіт у вигляді аналітичної статистики. Це має сприяти підвищенню якості міського середовища та відповідальності при використанні ресурсів.

Метою даної роботи є розробка та впровадження автоматизованої інформаційної системи обліку міської транспортної інфраструктури, що дозволить забезпечити централізований збір даних, оптимізувати планування ремонтних робіт та підвищити прозорість розподілу бюджетних коштів за рахунок використання інструментів управління проєктами та геоінформаційного аналізу.

Об'єктом дослідження є процеси управління проєктом розробки та впровадження інформаційної системи обліку міської транспортної інфраструктури.

Предметом дослідження є сукупність управлінських процесів, методів та інструментів планування, реалізації та моніторингу даного проєкту, зокрема процеси управління змістом, часом, вартістю, ресурсами, якістю та проєктними ризиками, що забезпечують успішну цифрову трансформацію міської інфраструктури [32].

Завдання дослідження:

- аналіз сучасного стану предметної галузі: дослідити світові тенденції та актуальні технології у сфері управління міською інфраструктурою для виявлення проблем існуючих систем.

- систематизація вимог: зібрати, формалізувати та проаналізувати потреби всіх зацікавлених сторін щодо майбутньої системи обліку.
- математичне моделювання: розробити математичні моделі, що дозволять аналізувати інформаційні зв'язки в системі та оптимізувати розподіл бюджетних коштів.
- проектування архітектури: розробити архітектурне рішення інформаційної системи, обґрунтувати вибір методологій аналізу та інструментів підтримки життєвого циклу розробки.
- розробка програмного забезпечення: створити логічну та фізичну моделі бази даних, провести декомпозицію системи на окремі функціональні модулі.
- управління проектом: розробити організаційну структуру, матрицю відповідальності та календарний план, що забезпечить ефективне виконання всіх завдань.
- оцінка ефективності: провести розрахунок вартості проекту, оцінити економічну доцільність та визначити потенційні ризики впровадження системи.

Перелік застосованих методів дослідження. Для досягнення поставленої мети в роботі використано комплекс взаємодоповнюючих методів: системного аналізу — для вивчення міської інфраструктури як цілісного об'єкта; математичного моделювання та формалізації — для розробки концептуальних моделей та оптимізації бюджету; декомпозиції — для структурування робіт проекту за допомогою WBS; інвестиційного аналізу — для фінансових розрахунків в Altinvest; проектного управління (PMBOK) — для розробки організаційної структури та календарного планування в MS Project; а також порівняльного аналізу та методів моніторингу — для обґрунтування технічних рішень та забезпечення контролю якості на всіх етапах впровадження системи.

Отримані результати. Результати дослідження є базовою платформою для розгортання інформаційної системи на реальних об'єктах. Виконана

робота поєднує комплексний теоретичний аналіз із прикладними розробками, що дозволяє використовувати створені структурні моделі, бази даних та плани управління проєктом як готовий фундамент для інтеграції в діяльність відповідних інфраструктурних організацій.

Наукова новизна результатів полягає в удосконаленні підходу до управління проєктами впровадження систем обліку міської інфраструктури, що, на відміну від традиційних методів, поєднує математичне моделювання бюджетних витрат із гнучкими інструментами проєктного менеджменту. Особливістю запропонованого підходу є здатність забезпечити баланс між складними технічними вимогами до обліку інфраструктурних об'єктів та реальними обмеженнями міського бюджету, що дозволяє мінімізувати фінансові ризики та зробити процеси управління міським господарством прозорими й обґрунтованими на кожному етапі реалізації проєкту.

Практичне значення отриманих результатів. Отримані результати мають високий ступінь готовності до практичного впровадження у муніципальну сферу. Розроблені матеріали — зокрема структура бази даних, ієрархічна структура робіт (WBS) та кошторисні розрахунки — можуть бути використані як початкова проєктна документація при формуванні технічних завдань для оголошення тендерів на розробку ІТ-систем у територіальних громадах.

Апробація результатів. Результати кваліфікаційної роботи магістра були презентовані на конференції IT&Is та опубліковані в тезах. Тема доповіді: «Сучасні підходи до ефективного управління проєктами в умовах цифрової трансформації» [17].

Публікації. Результати роботи були опубліковані у тезах доповідей на науково-технічній конференції.

РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТА ОБҐРУНТУВАННЯ ДОЦІЛЬНОСТІ ТА ЖИТТЄЗДАТНОСТІ ПРОЄКТУ

1.1 Аналіз ІТ-середовища області обліку транспортної інфраструктури. Порівняння існуючих рішень

Міські адміністрації великих міст (зокрема Києва) потребують інтегрованих систем для інвентаризації транспортних об'єктів – доріг, мостів, світлофорів, автобусних зупинок, дорожньої розмітки тощо. Така система повинна забезпечувати просторовий облік усіх об'єктів (наприклад, у геоінформаційному середовищі) та зберігати їх паспортні дані. Важливо також моніторити технічний стан цих об'єктів: регулярно фіксувати ушкодження покриття, зношення конструкцій, стан покриття тощо. Досягнення цього можливе через поєднання польових обстежень, датчиків IoT та аналізу історичних даних. Наприклад, сучасні GIS-платформи дозволяють автоматизовано «накладати» інформацію про пошкодження доріг на інтерактивну карту та аналізувати мережу в цілому, відображаючи такі параметри як рівність покриття чи міцність полотна [9, 19, 39].

Транспортне моделювання відіграє ключову роль у прогнозуванні впливу стану інфраструктури або ремонтних робіт на систему руху. За допомогою моделей (наприклад, PTV Visum/Vissim, AnyLogic, OpenSIM або цифрових двійників) можна імітувати затримки на розв'язках, корки, альтернативні маршрути тощо. Такі моделі дозволяють поставити «віртуальні перекриття» ділянок і подивитися, як від цього зміниться пасажиропотік, індекс мобільності (середня кількість поїздок на людину на день) чи індекс заторів. Наприклад, платформи цифрових двійників використовують реальні дані з камер і датчиків доріг, щоб моделювати потоки руху й насичення трафіком. Це дає змогу в режимі реального часу переналаштовувати світлофори і перенаправляти транспорт, мінімізуючи затори [13, 22].

Окремим завданням є оптимізація планування ремонтів і профілактичних робіт, щоб мінімізувати вплив на пасажиропотік. Міські управлінці прагнуть проводити роботи в години найменшого навантаження або поєднувати їх із іншими роботами, аби не паралізувати рух. Сучасні

рішення з управління активами й GIS дозволяють аналізувати транспортні потоки та обирати критичні ділянки для ремонту завчасно [21]. Наприклад, завдяки геоаналітиці спеціалісти можуть виявляти «вузькі місця» з високим навантаженням і планувати там роботи так, щоб вони припадали на час локально низького трафіку [18].

Правильно налаштований смартфон може стати персональним асистентом, помічником з планування власного часу та управління всіма сферами життя. Зокрема, все більшої популярності набирають додатки, які допомагають відслідковувати свій стан здоров'я, надають додаткову інформацію та поради щодо функціонування нашого організму.

Існуючі IT-рішення

GIS-платформи та системи управління активами: Наприклад, рішення від Esri, Cityworks чи український «GIS автомобільних доріг» від Magneticonemt дозволяють збирати, зберігати і візуалізувати просторові дані про мережу доріг та об'єктів інфраструктури [9]. Такі системи надають інструменти паспортизації доріг, мостів, світлофорів, зупинок тощо та відображають їх на карті. Через інтерфейс користувач може бачити стан об'єктів і вносити оновлення. Наприклад, модуль «GIS автомобільних доріг» призначений саме для обліку, паспортизації та супроводу життєвого циклу автошляхів (міських і заміських). Ці платформи зазвичай підтримують автоматичний пошук інформації по шарах («дороги», «світлофори» тощо) і дозволяють аналізувати мережу: оцінювати стан покриття чи виявляти проблемні ділянки за параметрами рівності або зчеплення проїзної частини. Системи класу Enterprise Asset Management (EAM) чи CMMS (IBM Maximo, Cityworks, FacilityForce тощо) доповнюють ці можливості модулями планування обслуговування, логістики ремонтних бригад, збирання технічних звітів.

Транспортне моделювання і цифрові двійники: Для оцінки впливу обмежень руху використовуються моделювальні інструменти (макро- та мікро-моделі). Компанії на кшталт PTV Group (Visum/Vissim), AnyLogic,

MatSim або відкриті рішення (SUMO, MATSim) дозволяють створювати деталізовані моделі руху в місті. В Україні також починають впроваджувати концепцію цифрового двійника руху. Зокрема, компанія «Про мобільність» розробляє цифрові моделі реальної мобільності, на яких тестуються різні сценарії рішень: транспорт, переміщення людей і авто “у цифровому форматі” [20]. Ці інструменти здатні програвати варіанти будівництва нових доріг чи перекриттів, оцінюючи їх ефект на пасажиропотік та затори.

Системи цифрових двійників і «smart city»-платформи: Деякі рішення поєднують дані моніторингу в реальному часі із моделями. Наприклад, платформи на кшталт GovPilot чи продукти IBA Ukraine надають можливість створити єдиний цифровий двійник міської інфраструктури. Вони інтегрують інформацію від IoT-датчиків (камера, магістральні лічильники трафіку) та дають аналітику: прогноз заторів, навантаження на громадський транспорт, необхідність профілактики на дорогах [17]. Це дозволяє приймати рішення, ґрунтуючись на даних (наприклад, перенастроїти світлофори чи розгорнути об'їзди) та сигналізувати про наближення критичного зносу об'єктів (мости, покриття) заздалегідь [17].

Інші технології: Для збирання даних використовуються мобільні додатки та датчики (білярдні камери, акустичні датчики, моніторинг бордюрів тощо). Рамки типу Tridium Niagara пропонують платформи для поєднання різних систем міста (енергетика, водопостачання, транспорт) на єдиному протоколі [22]. Такі підходи дозволяють інтегрувати існуючі «острівці» інформації в єдине рішення.

Структурне порівняння ключових функцій існуючих підходів наведено у таблиці 1.1.

Порівняння ключових функцій існуючих підходів

Категорія рішення	GIS + Asset Managem.	Транспортне моделювання (PTV, SUMO)	Цифрові двійники/ Smart City	Рішення IoT/ інтеграція (Tridium)
Інвентаризація інфраструктури	Так карти доріг, об'єктів	Ні дані імпортуються	Так інтеграція даних	Так підключає різні сенсори
Моніторинг стану	Так дані оглядів, сенсори	Ні використовують гіпотези	Так реальний час	Так поточні спостереження
Транспортне моделювання	Ні не моделюють трафік	Так розрахунок потоків, сценарії	Так динамічне моделювання	Ні самі по собі
Планування робіт і мінімізація впливу	Так розклад робіт, пріоритезація	Так оцінка впливу на рух	Так автоматизація оптимізації	Так допомагають зібрати дані для інших
Примітки	Esri ArcGIS, Magneticonemt, Cityworks	Використовуються для сценаріїв і оцінки	GovPilot, ІВА, рішення «Про мобільність»	Платформи інтеграції різноманітних систем

1.2 Обґрунтування доцільності розробки запропонованого рішення

Київ за рівнем заторів входить до трійки найпроблемніших міст світу – у 2021 році середня річна затримка руху на вулицях столиці сягала 56% [24]. Такі показники призводять до великих соціально-економічних витрат: даремно витрачається час громадян, збільшуються викиди шкідливих газів та вартість перевезень. Водночас існуюча у Києві вулично-дорожня мережа за «якісного управління трафіком» має резерв для підвищення пропускної здатності майже на 20% [25]. Це означає, що за рахунок інтелектуальних

систем (ІТС) можна суттєво оптимізувати рух без масштабних будівництв. Київська влада вже анонсувала впровадження ІТС, що дозволять прогнозувати затори та перенаправляти потоки через штучний інтелект і моніторинг у режимі реального часу [25].

Сучасна ситуація вимірюється низкою ініціатив: Київ долучився до європейської програми SUN4Ukraine (EU Smart Cities) з метою побудови кліматично-нейтрального та ефективного міста [16]. Також Україна разом із ЄІБ реалізує великі проєкти з модернізації електротранспорту та розширення інфраструктури (трамваїв, тролейбусів, метро), щоб зберегти екологічно чистий транспорт і не перевантажувати дороги. Усе це потребує прозорих інструментів планування та управління ресурсами. Наприклад, міський застосунок «Київ Цифровий» [23, 3, 38] вже надає сервіс «Рух транспорту», що показує рух громадського транспорту в реальному часі та час прибуття автобусів/тролейбусів. Проте даного функціоналу недостатньо: необхідна інтегрована платформа, яка охоплюватиме мета-модельний каталог об'єктів, інвентаризацію з прогнозуванням стану, моделювання транспортної ситуації, планування ремонтів, аналіз варіантів змін мережі, та оптимізацію бюджетного розподілу.

Без подібної системи управління різні аспекти інфраструктури лишаються розрізненими: інспектори ведуть облік стану «на папері», тендери проводяться окремо, а рішення щодо маршрутів чи ремонтів приймаються на основі уривчастих даних. Натомість комплексний web-based інструмент дозволить:

- централізовано зберігати інформацію про всі транспортні об'єкти (дороги, зупинки, транспортні засоби тощо) за гнучкою мета-моделлю;
- оперативно оновлювати технічний стан у єдиній БД (наприклад, формувати електронні карти об'єктів з усіма їх характеристиками і виконаними ремонтами) та прогнозувати зношення;

- моделювати рух з урахуванням реальних даних про стан доріг і трафік (використовуючи ІТС-алгоритми, як то передбачення заторів і перенаправлення потоків);
- планувати та узгоджувати технічні роботи з підрядниками (з інтеграцією тендерної платформи) так, щоб мінімізувати перешкоди руху;
- порівнювати сценарії (наприклад, вплив будівництва нового вело доріжки, зміни маршрутів, розширення транспорту) на ключові показники міської мобільності;
- оптимально розподіляти бюджет, автоматично генеруючи кілька варіантів інвестування в ремонт і розвиток мережі з оцінкою їх ефективності.

Усе це відповідає міжнародній практиці «розумних міст» – масштабному впровадженню ІТС, що покращує мобільність та екологію [24]. Системний підхід також відкриває можливості масштабування на інші українські міста: рішення можна адаптувати до різних масштабів і місцевих реалій за рахунок конфігурування мета-моделі об’єктів. Таким чином, поєднання згаданих модулів у єдиний web-сервіс не лише знизить витрати часу та коштів для мешканців і держави, а й узгодить наявні ініціативи (Цифровий Київ, транспортні програми, інвестиційні проекти) в єдину ефективну систему управління.

1.3 SWOT та PEST аналіз запропонованого рішення

1.3.1 PEST-аналіз

Було проведено аналіз зовнішнього макросередовища підприємства (для політичних, економічних, соціо-культурних та технологічних факторів) та об’єктивної оцінки експертів впливу факторів методом PEST аналізу [1, 26] – таблиці 1.2-1.6. Зробимо висновки, яким чином кожен із факторів може впливати на галузь та підприємство, а також згенеруємо ряд можливих дій з метою уникнення негативного впливу фактору. Результати дослідження наведені у таблиці 1.7.

Таблиця 1.2

Характер та ступінь впливу політичних факторів

Фактори впливу політичного середовища	Характер впливу фактору	Оцінка експертів ступеня впливу фактору (max = 3 б.)			Середній бал
		Експерт 1	Експерт 2	Експерт 3	
Кількісні та якісні обмеження на імпорт	-	2	2	1	-1.67
Бюрократизація і рівень корупції	-	3	3	2	-2.67
Стійкість політичної влади та існуючого уряду	-	2	2	2	-2.00
Податкова політика держави	-	3	2	2	-2.33
Вірогідність розвитку військових дій в країні	-	3	3	3	-3.00
Тенденції до регулювання або дерегулювання галузі	+	2	2	2	+2.00
Антимонопольне та трудове законодавство	-	2	2	2	-2.00
Майбутнє і поточне законодавство, що регулює правила роботи в галузі	-	2	3	2	-2.33
Підтримка інноваційних компаній з боку держави	+	2	2	1	+1.67
Лібералізація зовнішньоекономічного співробітництва	+	2	2	3	+2.33

Переважають негативні фактори, пов'язані з корупцією, військовими ризиками та податковою політикою. Проте державна підтримка інновацій забезпечують позитивні перспективи для проєкту.

Таблиця 1.3

Характер та ступінь впливу економічних факторів

Фактори впливу економічного середовища	Характер впливу фактору	Оцінка експертів ступеня впливу фактору (max = 3 б.)			Середній бал
		Експерт 1	Експерт 2	Експерт 3	
1	2	3	4	5	6
Темпи зростання економіки	-	2	2	3	-2.33
Рівень інфляції	-	3	3	2	-2.67
Курси основних валют	-	3	3	3	-3.00
Рівень наявних доходів населення	-	2	2	2	-2.00

1	2	3	4	5	6
Ступінь глобалізації та відкритості економіки	+	3	2	3	+2.67
Монетарна та фіскальна політика держави	-	3	3	2	-2.67
Рівень розвитку підприємництва та бізнес середовища	+	2	3	2	+2.33
Цінова конкуренція з боку зарубіжних компаній	-	2	2	2	-2.00
Інвестиційний клімат в галузі	-	2	3	2	-2.33

Серед економічних факторів є як позитивні, так і негативні. Валютна нестабільність і висока інфляція створюють значні ризики, тоді як відкритість ринку та підтримка бізнес-середовища забезпечують перспективи для реалізації проекту.

Таблиця 1.4

Характер та ступінь впливу соціально-культурних факторів

Фактори впливу соціально-культурного середовища	Характер впливу фактору	Оцінка експертів ступеня впливу фактору (max = 3 б.)			Середній бал
		Експерт 1	Експерт 2	Експерт 3	
Рівень підготовки молодих спеціалістів в галузі	+	2	3	2	+2.33
Вимоги до якості продукції та рівня сервісу	+	3	3	3	+3.00
Рівень міграції та імміграційні настрої	-	3	2	3	-2.67
Культура формування заощаджень і кредитування суспільства	-	2	2	2	-2.00
Розвиток релігії та інших вірувань	+	1	1	1	+1.00
Ставлення до імпортованих товарів і послуг	+	2	2	1	+1.67
Спосіб життя і звички споживання	+	2	3	2	+2.67
Темпи росту населення	-	2	3	2	-2.33

Соціально-культурні фактори переважно позитивно впливають на проєкт завдяки високим стандартам якості та сучасним споживчим звичкам. Проте проблеми з міграцією та низькою фінансовою спроможністю населення створюють певні ризики для реалізації.

Таблиця 1.5

Характер та ступінь впливу технологічних факторів

Фактори впливу технологічного середовища	Характер впливу фактору	Оцінка експертів ступеня впливу фактору (max = 3 б.)			Середній бал
		Експерт 1	Експерт 2	Експерт 3	
Рівень інновації та технологічного розвитку галузі	+	3	3	2	+2.67
Ступінь використання, впровадження та передачі технологій	+	2	2	3	+2.33
Доступ до новітніх технологій	+	3	3	3	+3.00
Витрати на дослідження та розробки	-	2	2	2	-2.00
Можливість виробництва якісно нової продукції (розвиток конкурентних технологій)	+	3	2	3	+2.67
Законодавство в галузі технологічного оснащення галузі	+	2	2	1	+1.67

Технологічні фактори здебільшого позитивно впливають на проєкт, особливо завдяки інноваціям і доступу до сучасних технологій. Проте обмеження фінансування досліджень та розвитку та повільне оновлення законодавства можуть створювати перешкоди.

Таблиця 1.6

Фактори, які спричиняють найбільший вплив на проєкт за результатами оцінювання PEST-аналізу

Політичні		Економічні	
Фактор	Вага	Фактор	Вага
Вірогідність розвитку військових дій в країні	-3.00	Курси основних валют	-3.00
Лібералізація зовнішньоекономічного співробітництва	+2.33	Ступінь глобалізації та відкритості економіки	+2.67
Соціально-культурні		Технологічні	
Фактор	Вага	Фактор	Вага
Вимоги до якості продукції та рівня сервісу	+3.00	Доступ до новітніх технологій	+3.00
Рівень міграції та імміграційні настрої	-2.67	Витрати на дослідження та розробки	-2.00

Підсумки здійсненого PEST-аналізу

Фактори	Зміни в галузі	Зміни в організації	Дії
Політичні	1. Урядові ініціативи щодо цифровізації управління міською інфраструктурою (Дія.City, Smart City тощо) 2. Підвищення уваги до прозорості бюджетних витрат та процедур закупівель	1. Потреба відповідати регуляторним вимогам електронного обліку та звітності 2. Можливість отримання підтримки через державні програми цифровізації	1. Побудова рішення відповідно до вимог цифрового документообігу та прозорості 2. Підготовка до участі у державних та міських цифрових ініціативах
Економічні	1. Обмежені бюджети муніципалітетів на обслуговування інфраструктури 2. Доступність грантів ЄС і міжнародної технічної допомоги для smart-рішень	1. Потреба пропонувати економічно обґрунтовані рішення, які демонструють ефект на витрати/мобільність 2. Створення потенціалу для залучення донорського фінансування	1. Розробка модуля оптимізації розподілу бюджету з моделюванням результатів 2. Підготовка грантових пропозицій та партнерств з містами
Соціально-культурні	1. Зростаючі очікування громадян щодо якісної та безпечної міської інфраструктури 2. Підвищення зацікавленості суспільства у відкритих даних та smart-city рішеннях	1. Необхідність розробки користувацьких інтерфейсів для взаємодії громадян з системою 2. Підвищення важливості публічної аналітики та прозорості рішень	1. Додати до системи інтерфейси відкритих даних та модулів для громадського моніторингу 2. Розробити інструменти візуалізації для презентацій містянам
Технологічні	1. Активне впровадження ГІС, IoT, інструментів транспортного моделювання (Aimsun, PTV Visum тощо) 2. Поширення web-based архітектури та API-орієнтованих сервісів	1. Необхідність сумісності з існуючими системами міської інфраструктури (ГІС, транспортні моделі) 2. Потреба підтримки масштабування та інтеграції з іншими модулями смарт-міста	1. Архітектура на базі відкритих API та мікросервісів, web-GIS підтримка 2. Побудова бази на гнучкій мета-моделі для масштабованості системи

В результаті PEST-аналізу було виявлено фактори, що найбільше позитивно впливають на компанію: лібералізація зовнішньоекономічного співробітництва, ступінь глобалізації та відкритості економіки, вимоги до якості продукції та рівня сервісу, доступ до новітніх технологій. Ці фактори треба обов'язково враховувати при управлінні проєктом, оскільки вони напряду впливають на його розвиток та успіх.

До факторів, що мають найбільш негативний вплив належать: вірогідність розвитку військових дій в країні, курси основних валют, рівень міграції та імміграційні настрої, витати на дослідження розробки. Ці фактори слід враховувати при управлінні ризиками в проєкті.

1.3.2 Аналіз внутрішнього та зовнішнього середовища

Для більш детальної оцінки зовнішнього та внутрішнього середовища проєкту проведемо SWOT-аналіз [2] – таблиця 1.8.

Таблиця 1.8

SWOT аналіз концепції продукту

Внутрішні сильні сторони	Потенційні зовнішні можливості
1. Наявність потужних GIS/Asset-Management інструментів, що забезпечують просторовий аналіз мережі доріг і активів. Дозволяють інтегрувати дані про технічний стан і навантаження. 2. Сучасні технології цифрових двійників дозволяють проводити «what-if»-моделювання ремонтів та інфраструктурних змін з метою мінімізації негативного впливу на рух. 3. Потенціал зниження експлуатаційних витрат через профілактичні ремонти і оптимізацію робіт (завдяки прогнозам стану і навантаження). 4. Можливість залучення зовнішнього фінансування на «розумні міста» (гранти ЄС, ЕІВ тощо) як драйвер впровадження.	1. Інтеграція «актив–дані–модель–прийняття рішення»: формування єдиної системи, яка охоплює облік, моніторинг, моделювання і планування робіт. Це дозволить прискорити реагування на проблеми та прозоро інформувати громадськість. 2. Розвиток аналітики та AI: обробка великого масиву даних (відеопотоки, сенсорики) для виявлення дефектів і прогнозування зносу в автоматичному режимі. 3. Використання відкритих даних (Open Data): відкриття та стандартизація інформації про міську інфраструктуру може стимулювати появу нових сервісів. 4. Співпраця з міжнародними проєктами (досвід інших міст, «схеми роботи» під час реконструкції).

Внутрішні слабкі сторони	Потенційні зовнішні загрози
1. Фрагментарність рішення: багато несумісних систем та форматів даних у містах. Низький рівень цифрової зрілості місцевих департаментів. 2. Висока вартість впровадження і утримання. Необхідність підготовки та навчання персоналу. Бюджетні цикли можуть затримувати оновлення. 3. Обмежена інтеграція: наявні рішення часто охоплюють лише окремі задачі (наприклад, моделювання без урахування реального моніторингу) 4. Низький початковий попит і скепсис користувачів. Спротив змінам в усталених процесах комунальних служб і перевізників.	1. Війна та нестабільність можуть призвести до затримок у проєктах і руйнувань інфраструктури, що потребують іншого підходу до планування. 2. Кадрова криза: нестача фахівців з транспортного моделювання та аналітики, адже багато спеціалістів переорієнтовані на інші завдання. 3. Кібербезпека: розвиток IoT і цифрових сервісів відкриває нові вектори атак (на датчики, служби даних). 4. Можливі збої в роботі комунікацій або невідповідність технічних стандартів, що ускладнить міжсистемну взаємодію.

Аналіз показує, що ринок має ряд рішень для обліку і моніторингу транспортної інфраструктури (ГІС і CMMS-платформи, транспортні симулятори, платформи цифрових двійників). Однак значущою прогалиною залишається єдина інтегрована система, що поєднувала б інвентаризацію об'єктів, постійний моніторинг їх стану і моделювання руху у спільному середовищі з аналітикою. Відсутні готові комплексні рішення, які закривали б усі потреби міст: від засобів оперативного збору інформації (мобільні інспекції, сенсори) до потужного алгоритму оптимального планування ремонтів з урахуванням мінімального впливу на трафік. Саме це визначає запит на нові рішення – наприклад, систему «міський цифровий двійник» із елементами штучного інтелекту для пріоритезації об'єктів ремонту за умовами мобільності [10].

1.4 Постановка задачі дослідження, формулювання технічного завдання на розробку у вигляді паспорту проєкту

1.4.1 Задача дослідження

Задача дослідження полягає у проектуванні архітектури та розробці методології управління впровадженням інтегрованої Web-системи обліку міської транспортної інфраструктури. Основною науково-прикладною проблемою є створення інструменту, здатного консолідувати розрізнені дані (ГІС, IoT-сенсори, бюджетні системи) у єдиний цифровий контур. Необхідно забезпечити модульність архітектури для інтеграції з існуючими міськими сервісами (Prozorro, е-квиток, центр моніторингу трафіку) при дотриманні високих вимог до безпеки та масштабованості, що дозволить мінімізувати суб'єктивізм при прийнятті управлінських рішень та підвищити ефективність планування бюджетних видатків.

Дана діяльність класифікується як проєкт, оскільки вона є цільовою, динамічною та обмеженою в часі (24 місяці) діяльністю, що спрямована на створення унікального продукту — інформаційної системи обліку міської транспортної інфраструктури. Проєкт характеризується жорсткими обмеженнями ресурсів (бюджет 10 млн грн, внутрішній персонал), висуває високі вимоги до якості (стабільність та захищеність системи впродовж 20-річного життєвого циклу) та вимагає міждисциплінарної координації між сферами ІТ, транспортного моделювання, інженерії та фінансового обліку

1.4.2 Паспорт проєкту

Назва проєкту: «Розробка і впровадження інформаційної системи обліку і обслуговування міської транспортної інфраструктури».

Керівник проєкту: Семчишин Віталій Русланович.

Замовник: Київська міська державна адміністрація (КМДА).

Бізнес-ціль:

- забезпечення пріоритетності інвестицій у розвиток інфраструктури;
- зменшення витрат на обслуговування інфраструктури на 15% за 5 років;
- зменшення кількості непередбачуваних поломок об'єктів у 2 рази;

- зниження індексу заторів на 20 годин за 2 роки;
- збільшення щоденного пасажиропотоку на 25% впродовж 5 років;
- зниження вартості аудиту при зборі даних на 50% за 10 років;
- оптимізація штату адміністративних співробітників на 30%.

Тривалість: 24 місяці.

Бюджет: 10 000 000 грн

Ключовий результат: Готова Web-система, розгорнута у захищеному хмарному середовищі, з функціоналом обліку, планування робіт, моделювання та аудиту.

Критерії успіху: Випуск стабільної альфа-версії, повне усунення помилок, відповідність стандартам безпеки, забезпечення стійкої роботи системи протягом 20+ років з можливістю регулярних оновлень.

Припущення та обмеження: Міждисциплінарний підхід (ІТ, транспорт, будівництво), обмеження внутрішніх ресурсів персоналу, відповідність законодавству України про доступ до публічної інформації та захист персональних даних.

Високорівневі ризики: Технічна несумісність із застарілими системами міста, фрагментація вхідних даних, кіберзагрози, зміна бюджетних пріоритетів.

1.4.3 Структура та архітектура системи

Web-система з багаторівневим рішенням (клієнт – сервер), бажано на сучасних фреймворках (SPA для інтерфейсу, REST/API для обміну). Доцільно використовувати мікросервісну архітектуру або чітко розділені модулі, що спілкуються через API [9, 28, 30]. Основні компоненти: база даних (SQL/NoSQL) з просторовими даними (ГІС-сервер), бекенд-сервіси з алгоритмами аналізу та моделювання, веб- та мобільний фронтенд для користувачів [10]. Як основа даних може служити реляційна СУБД (PostgreSQL + PostGIS) або NoSQL (наприклад, MongoDB) з геопросторовими розширеннями. Архітектуру слід передбачити модульною: наприклад, окремі

сервіси для каталогу об'єктів, інвентаризації, транспортного моделювання, планування робіт, аналізу альтернатив і управління бюджетом [11].

1.4.4 Функціональні вимоги

1. *Моніторинг стану інфраструктури*: Система повинна автоматично збирати дані зі встановлених датчиків і пристроїв.
2. *Планування обслуговування*: Модуль для генерації графіків обслуговування та прогнозування можливих збоїв.
3. *Інтерактивна карта*: Відображення поточного стану інфраструктури в реальному часі.
4. *Сповіщення*: Налаштування автоматичних сповіщень про критичні ситуації.
5. *Аналітика*: Генерація звітів на основі зібраних даних із можливістю візуалізації.

1.4.5 Типи користувачів

Система має підтримувати кілька ролей з відповідними правами доступу. Наприклад:

- *Адміністратори (міська влада)*: повний доступ до налаштувань системи, завантаження даних, створення сценаріїв, фінансових правил і звітів.
- *Планувальники та аналітики*: доступ до модулів моделювання, порівняння альтернатив, формування звітів; можуть налаштовувати параметри симуляцій.
- *Інспектори й технічний персонал*: доступ до модуля інвентаризації: можуть вводити/редагувати стан об'єктів, завантажувати фото і заміри, отримувати завдання на перевірки.
- *Підрядники та постачальники*: доступ до планування робіт — отримують техзавдання, статуси тендерів, можуть подавати документи та звіти про виконання.

- *Аналітики і широке коло громадськості*: обмежений доступ до перегляду звітів або відкритих даних (наприклад, показники заторів, бюджету) для прозорості (за потребою – data.gov.ua).

1.4.6 Інтеграції

Система повинна з'єднуватися з існуючими ІТ-рішеннями та сервісами міста. Зокрема:

- *ГІС-платформи*: інтеграція з міською картою (OpenStreetMap, Esri, чи локальним геопорталом Києва) для візуалізації доріг, інфраструктури і результатів моделювання.
- *Центр організації дорожнього руху*: отримувати дані із систем відеомоніторингу та датчиків (камери, детектори) для актуалізації інформації про транспортні потоки.
- *Електронний квиток*: синхронізація з системою е-квитка (Kyiv Smart Card) для врахування пасажиропотоків.
- *Прозорро*: обмін даними з електронною платформою держзакупівель Prozorro для публікації тендерів і звітів щодо витрачених коштів.
- *ERP/Фінанси*: передача даних в бюджетні системи КМДА та управління фінансами для автоматичного планування і розрахунків.
- *IoT та мобільні додатки*: можливість отримувати дані від смарт-сенсорів (наприклад, лічильників трафіку або підключень трамваїв) і передавати інформацію від мобільних додатків (наприклад, додатку «Київ Цифровий» для зворотного зв'язку від пасажирів).

1.4.7 Безпека

Оскільки система працює з критичними даними і забезпечує містобудівні рішення, необхідно забезпечити:

- *Аутентифікацію та авторизацію*: впровадження системи RBAC (Role-Based Access Control) з багаторівневим доступом. Обов'язково шифрування даних при зберіганні та передачі (HTTPS, SSL/TLS).

- *Журнали подій і аудит*: фіксація всіх змін у системі (хто і коли вніс дані чи змінив модель), регулярні резервні копії.
- *Відповідність стандартам*: відповідність законодавству України (зокрема закони про доступ до публічної інформації та про захист персональних даних, якщо будуть оброблятися дані про пасажирів чи інспекторів) [1, 2].
- *Кіберзахист*: захист від DDoS-атак, антивірусна та антивзламна безпека. Можливе використання сертифікації (наприклад, ISO 27001) для критичних компонентів [7].
- *Навчання користувачів*: підготовка інструкцій і тренінгів для користувачів-працівників, щоб уникнути помилок і забезпечити коректність введених даних.

1.4.8 Масштабованість

Система має бути спроектована з урахуванням можливості подальшого розвитку та розгортання:

- *Горизонтальне розширення*: підтримка кластерів баз даних і балансування навантаження, щоб обробляти великі масиви просторових даних та запитів (наприклад, при попиті від десятків користувачів або розширенні на декілька міст).
- *Модульність*: архітектура дозволяє додавати нові модулі (наприклад, «велосипедна інфраструктура» чи «автономний транспорт») без переробки всієї системи.
- *Регулярне оновлення*: гнучкий випуск оновлень, можливість масштабування на хмарну інфраструктуру (AWS, Azure, локальні дата-центри) з автоматичним розгортанням.
- *Багатомовність і багатомістність*: система готова до додавання нового мовного інтерфейсу (при розширенні на інші міста) та підтримує розмежування даних по містах/регіонах (multi-tenant).

Урахування українських реалій (сильна фрагментація систем, війна, бюджетні обмеження) підтверджує необхідність єдиного цифрового інструменту управління транспортною інфраструктурою. Така система дозволить використати дані з усіх доступних джерел (камери, «Київ Цифровий», інспектори, фінанси) для ефективного прийняття рішень і оптимізації ресурсів. Світові та локальні приклади показують: інтелектуальні транспортні системи зменшують затори і покращують мобільність [24]. Реалізація запропонованих модулів у вигляді web-системи забезпечить прозорість витрачання коштів, підвищить оперативність ремонту та розвитку мережі, а також прискорить масштабування успішних рішень на інші українські міста (у рамках «смарт сіті») для досягнення сталого розвитку та безпеки транспорту [25].

1.5 Концепція проєкту

1.5.1 Дерево причин та наслідків

Для обґрунтування доцільності розробки інформаційної системи та ідентифікації ключових точок управлінського впливу було побудовано дерево причин та наслідків (див. рис. 1.1). Центральною проблемою дослідження визначено **неефективний облік і обслуговування міської транспортної інфраструктури**, що є системним бар'єром для розвитку сучасного урбаністичного середовища.

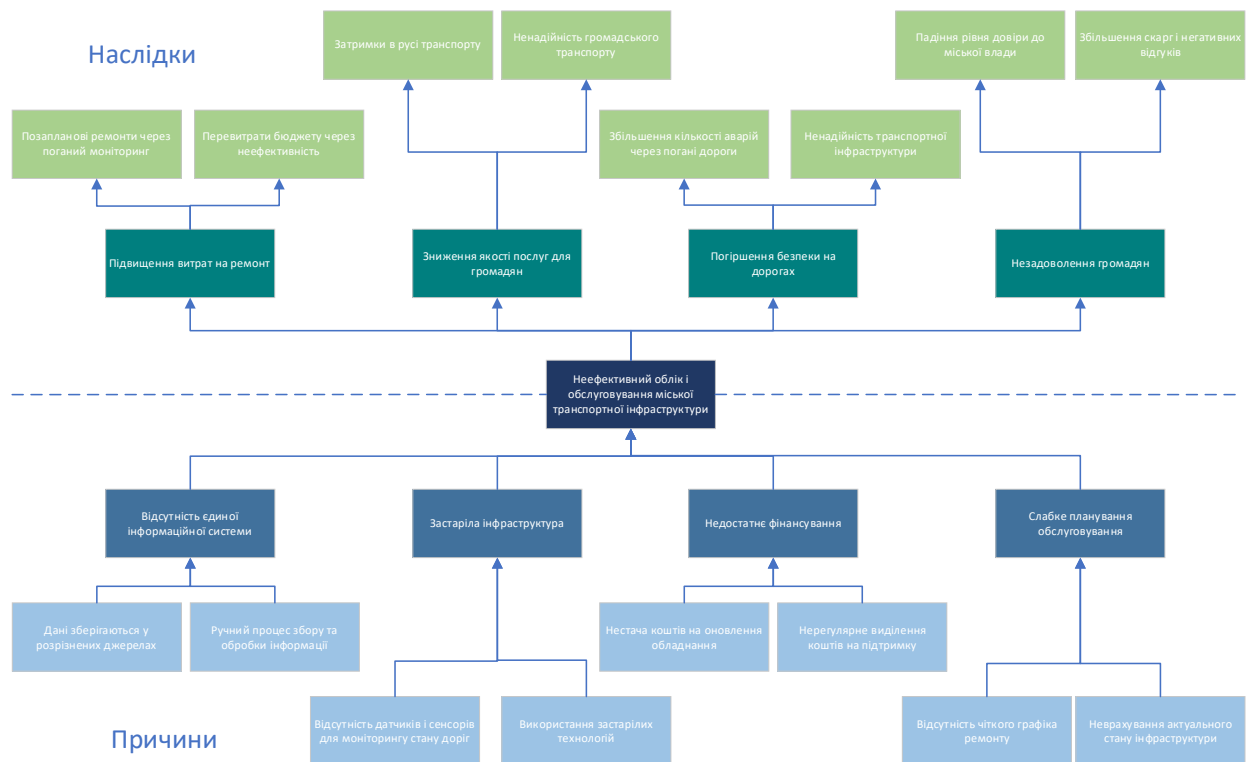


Рис. 1.1. Дерево причин та наслідків

Аналіз фундаментальних причин

Детальний розгляд архітектури проблеми дозволяє виділити чотири базові блоки причин, що детермінують незадовільний стан галузі.

Відсутність єдиної інформаційної системи. Дана причина є ключовою з точки зору ІТ-менеджменту. На сьогодні дані про інфраструктуру зберігаються у розрізних джерелах (паперові архіви, локальні таблиці різних відомств), що унеможлиблює отримання цілісної картини. Ручний процес збору та обробки інформації призводить до високої ймовірності помилок, втрати актуальності даних та значних часових затримок у прийнятті рішень.

Застаріла інфраструктура. Фізичний знос об'єктів посилюється відсутністю сучасних засобів моніторингу. Брак датчиків і сенсорів для автоматичного відстеження стану дорожнього покриття чи роботи світлофорних об'єктів змушує муніципальні служби працювати у режимі «реактивного» обслуговування (усунення наслідків замість запобігання). Використання застарілих технологій інвентаризації не дозволяє інтегрувати дані у сучасні ГІС-платформи.

Недостатнє та нераціональне фінансування. Проблема не обмежується лише дефіцитом коштів. Нестача ресурсів на оновлення обладнання часто є наслідком нерегулярного та необґрунтованого виділення коштів. Без цифрового обліку неможливо довести пріоритетність тих чи інших видатків, що створює замкнене коло недофінансування критичних вузлів.

Слабке планування обслуговування. Відсутність чітких графіків ремонту та ігнорування актуального стану об'єктів призводить до того, що обслуговування проводиться або занадто пізно, або не в повному обсязі. Це прямий наслідок дефіциту оперативної інформації, яка мала б надходити з автоматизованої системи.

Аналіз проміжних та критичних наслідків

Неефективність на рівні причин неминуче трансформується у негативні наслідки, які мають прямий вплив на життєдіяльність міста та бюджетний процес.

Економічний аспект (Підвищення витрат на ремонт). Через відсутність превентивного моніторингу дрібні пошкодження перетворюються на аварійні ситуації, що потребують капітальних вкладень. Позапланові ремонти завжди коштують дорожче за регламентні, а неефективне планування призводить до перевитрат бюджету, які могли б бути спрямовані на розвиток нових об'єктів.

Операційний аспект (Зниження якості послуг для громадян). Затримки в русі громадського та приватного транспорту стають регулярними. Ненадійність інфраструктури знижує мобільність населення, що негативно позначається на економічній активності міста.

Аспект безпеки (Погіршення безпеки на дорогах). Це найбільш критичний наслідок. Збільшення кількості аварій через поганий стан доріг або некоректну роботу засобів регулювання трафіку (світлофори, розмітка) несе пряму загрозу життю та здоров'ю громадян. Транспортна інфраструктура втрачає свою основну функцію — безпечне забезпечення зв'язності територій.

Соціально-політичний аспект (Незадоволення громадян). Системні збої в роботі транспорту призводять до падіння рівня довіри до міської влади.

Збільшення кількості скарг та негативний інформаційний фон змушують муніципалітет працювати в режимі «гасіння пожеж», замість реалізації стратегічних проєктів розвитку Smart City.

Висновок щодо стратегії розв’язання проблеми

Побудоване дерево причин та наслідків чітко демонструє, що вирішення проблеми лише шляхом збільшення фінансування (екстенсивний шлях) не принесе довгострокового результату, оскільки корінь проблеми лежить у площині управління інформацією та процесами.

Впровадження інформаційної системи, що є об’єктом даного дипломного проєкту, дозволить розірвати ланцюги негативних наслідків у точці «Відсутність єдиної системи» та «Слабке планування». Автоматизація обліку, інтеграція з IoT-датчиками та створення цифрових паспортів об’єктів дозволять перейти до моделі предиктивного (прогнозного) обслуговування. Це, своєю чергою, стабілізує витрати бюджету, підвищить безпеку на дорогах та забезпечить стійкий розвиток транспортної мережі міста на десятиліття вперед.

1.5.2 Постановка цілей

Центральною точкою розробки стратегії впровадження системи є декомпозиція глобальної місії проєкту на конкретні, вимірювані завдання. Головна ціль проєкту полягає у **покращенні управління міською транспортною інфраструктурою через впровадження сучасної інформаційної системи**, що дозволить перейти від реактивного до предиктивного (прогнозного) управління.

Для візуалізації ієрархії завдань та встановлення зв’язків між ними було побудовано Дерево цілей (див. Рис. 1.2). Структура дерева передбачає поділ генеральної цілі на чотири стратегічні вектори:

- Автоматизація обліку: створення цифрового фундаменту через електронні реєстри та автоматизований збір даних.

- Оптимізація обслуговування: впровадження модулів планування та прогнозування зносу об'єктів.
- Підвищення ефективності ресурсів: аналіз трафіку та надання рекомендацій щодо оптимізації маршрутів.
- Інформаційна підтримка: забезпечення управлінців та громади аналітичними дашбордами через веб-інтерфейси.

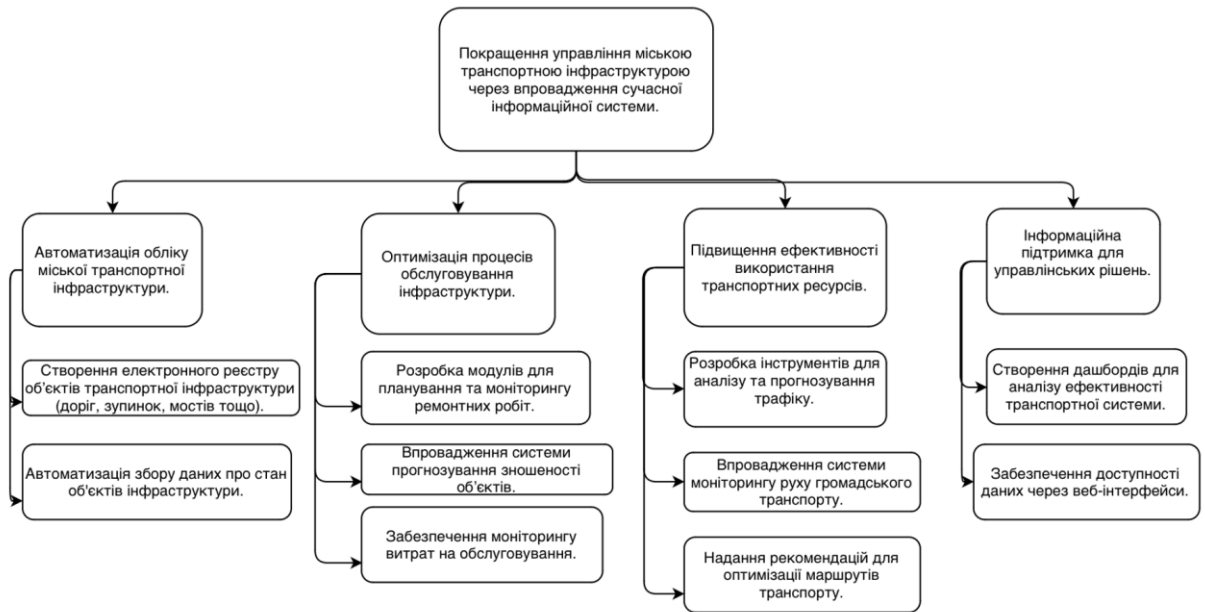


Рис. 1.2. Дерево цілей.

Для детального аналізу параметрів успіху проєкту, враховуючи часові та бюджетні обмеження, розроблено логіко-структурну схему цілей (див. Табл. 1.9). Вона дозволяє чітко визначити індикатори досягнення результатів та ідентифікувати потенційні ризики на кожному рівні логіки.

Таблиця 1.9

Логіко-структурна схема цілей

	Показники досягнення	Вимірювачі (індикатори)	Припущення та ризики
--	----------------------	-------------------------	----------------------

Загальна ціль	Покращення управління міською транспортною інфраструктурою через впровадження інформаційної системи	За допомогою розробленого продукту зекономлено 15% коштів на обслуговування інфраструктури за 5 років користування. Об'єкти транспортної інфраструктури виходили з ладу на 50% рідше ніж до впровадження.	Наявність фінансування, підтримка з боку місцевої влади, готовність співпрацювати з технічними партнерами.
Конкретні цілі	1. Автоматизація обліку інфраструктури 2. Оптимізація обслуговування 3. Підвищення ефективності транспортних ресурсів	1. Зниження витрат на обслуговування на інфраструктури на 15% за 5 років.	Затримки у процесі впровадження через технічні або організаційні причини.
Результати	1. Єдиний електронний реєстр інфраструктури 2. Аналітичні звіти для управлінців 3. Поліпшення ефективності громадського транспорту	Зменшення індексів заторів на 20 годин за два роки Пасажиропотік в місті зріс на 25%.	Ризики пов'язані з низькою якістю зібраних даних або складністю навчання персоналу.
Дії та засоби	Розробка та впровадження інформаційної системи обліку, навчання персоналу, тестування системи	Штат адміністративних працівників відповідальних за облік об'єктів транспортної інфраструктури зменшився на 30%.	Можливе перевищення витрат на розробку або недостатній рівень кваліфікації персоналу.

Запропонована логічна структура дозволяє не лише реалізувати технічні вимоги до системи, а й забезпечити довгостроковий соціально-економічний ефект. Проєкт орієнтований на досягнення стійкого результату (життєвий цикл понад 20 років), що робить його важливою частиною стратегії розвитку «Smart City» та цифрової трансформації міського господарства.

1.5.3 Мета, цілі та продукт проєкту

Місія проєкту: трансформувати підходи до управління міським господарством через цифровізацію та створення інтелектуального середовища для обслуговування транспортної інфраструктури.

Мета проєкту: підвищити рівень керованості та надійності транспортної мережі шляхом впровадження інструментів автоматизованого обліку, предиктивної аналітики стану об'єктів та інтеграції міських сервісів у єдину Web-платформу.

Для деталізації завдань сформуємо основні цілі проєкту за методологією SMART:

- Конкретність (Specific): За допомогою розробленої системи забезпечити повну цифровізацію реєстру об'єктів транспортної інфраструктури (дороги, мости, зупинки) з прив'язкою до ГІС-координат.
- Вимірюваність (Measurable): Досягти скорочення витрат часу на формування щоквартальних звітів про стан інфраструктури з декількох робочих днів до 10 хвилин за рахунок автоматичної генерації аналітики.
- Досяжність (Achievable): Забезпечити інтеграцію системи з мінімум трьома існуючими міськими сервісами (Prozorro, Київ Цифровий, ГІС-портал) протягом першого року впровадження.
- Релевантність (Relevant): Зменшити частку позапланових (аварійних) ремонтних робіт на 20% у перший рік експлуатації завдяки модулю прогнозування зношеності.
- Обмеженість у часі (Time-bound): Випустити повнофункціональну стабільну версію системи та завершити навчання персоналу профільних департаментів протягом 24 місяців з моменту старту проєкту.

Продуктом проєкту є багатомодульна Web-система (клієнт-серверна архітектура), що забезпечує автоматизацію обліку, моніторинг технічного стану та стратегічне планування обслуговування транспортної інфраструктури міста.

РОЗДІЛ 2. МАТЕМАТИЧНА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

2.1 Розробка концептуальних моделей інформаційної системи.

Визначення підсистем та надсистем

Система - Система обліку міської транспортної інфраструктури.

Аналіз системи з фізичної точки зору.

Надсистема: КМДА, міністерство транспорту України

Зовнішні зв'язки системи з надсистемою.

Зовнішні фактори впливу:

- маршрути;
- зупинки;
- заправки;
- графік роботи;
- транспорт;
- станції технічного обслуговування.

Повний перелік всіх підсистем.

Перелік підсистем та їх елементів для кожної підсистеми наведено у таблиці 2.1.

Таблиця 2.1

Елементи підсистем

Користувацький інтерфейс (UI)	Аналізуючий модуль	Бази даних	Співробітники
Панель управління. Карта об'єктів інфраструктури. Модуль заявок на обслуговування. Система сповіщень. Історія обслуговування.	Алгоритм моніторингу стану. Модуль аналізу та прогнозування. Алгоритм пріоритезації. Модуль ефективності.	Інвентаризація об'єктів. Журнал обслуговування. Дані про стан інфраструктури. Метадані ресурсів.	Директор. Менеджер. Бухгалтер. Розробники.

Подамо у вигляді системи процес функціонування системи – рис. 2.1

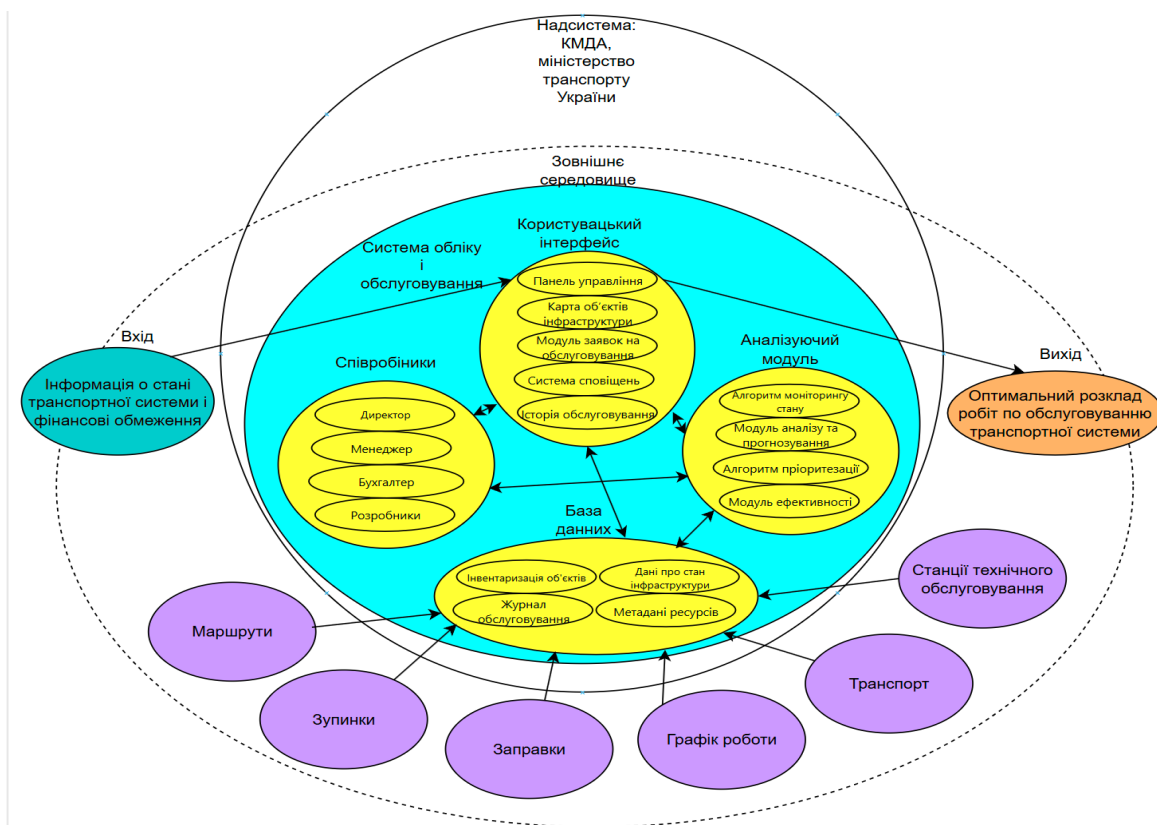


Рис. 2.1. Склад ІТ системи обліку і обслуговування міської транспортної інфраструктури

Опишемо підпроцеси як елементи системи.

Запит на аналіз стану транспортної інфраструктури: запит формується для отримання актуальної інформації про стан об'єктів транспортної інфраструктури (доріг, мостів, світлофорів тощо) та наявних фінансових обмежень.

Збір і моніторинг даних про об'єкти: постійний збір і оновлення даних через сенсори, сторонні джерела та ручні перевірки. Фіксація стану кожного об'єкта в реальному часі для подальшого аналізу.

Аналіз отриманих даних: аналіз інформації про рівень зносу, інтенсивність використання та поточний стан об'єктів. Розподіл об'єктів за пріоритетністю ремонту на основі критичності та ризику аварійності.

Оцінка фінансових і ресурсних обмежень: визначення наявного бюджету та доступності технічних ресурсів і персоналу для обслуговування об'єктів.

Узгодження плану витрат з поточними фінансовими можливостями.

Пріоритезація об'єктів для обслуговування: формування черги об'єктів для обслуговування на основі критичності їх стану та наявних ресурсів. Визначення термінів виконання для кожного об'єкта з урахуванням бюджету.

Оптимізація плану обслуговування: система автоматично формує оптимальний план обслуговування, враховуючи пріоритетність об'єктів, маршрути для бригад, доступні ресурси та фінансові обмеження.

Визначення і розподіл ресурсів: призначення необхідних ресурсів (матеріалів, персоналу, техніки) для виконання завдань на кожному об'єкті. Перевірка наявності та стану ресурсів.

Оптимізація маршрутів обслуговування: логістичний модуль розраховує оптимальні маршрути для обслуговуючого персоналу, мінімізуючи час та витрати на пересування між об'єктами.

Затвердження і формування остаточного графіка робіт: після узгодження і затвердження оптимізований план робіт оформлюється як остаточний графік для виконання обслуговування.

Надсистема для процесу і зовнішні зв'язки з нею.

Результатом роботи системи є оптимальний розклад робіт по обслуговуванню транспортної системи. Сформований розклад включає послідовність, дати і маршрути для обслуговування кожного об'єкта, а також розподіл ресурсів. Цей розклад забезпечує економічно ефективне і своєчасне виконання робіт з обслуговування інфраструктури.

2.2 Формалізація математичних моделей та постановка задачі в математичному вигляді

Формальний опис ІТ-системи.

Процес створення оптимального розкладу транспортної системи представлений у вигляді моделі “конуса”. Нехай $P = P^p, P^s$, де P^p - це сукупність процесів, що стосуються до створення оптимального розкладу

транспортної системи, а P^S – їх кількість, що в результаті призводить до кількості процесів пов'язаними зі створенням та подальшою підтримкою продукту: $P^S = P_1^S, P_2^S, \dots, P_j^S$, де j – кількість процесів.

Багато ресурсів у моделі "конуса" включають наявні та залучені (придбані) в рамках проекту матеріали, програмне забезпечення, робочу силу та інформаційні ресурси, які матеріалізуються в кінцевому продукті та утворюють нову бажану вартість у вигляді ІТ-проекту продукту (розподілена інформаційна система - PIS).

$R = R^M, R^H, R^P, R^I$, де R^M – сукупність матеріальних ресурсів, що задіяні у проекті.

$R^M = r_1^M, r_2^M, \dots, r_j^M$, де j – кількість видів матеріальних ресурсів, необхідних на проекті;

R^H – Сукупність людських ресурсів, що беруть участь у проекті. $R^H = \{r_1^H, r_2^H, \dots, r_j^H\}$, де j – кількість видів людських ресурсів, залучених до проекту;

R^P – Набір програмних ресурсів, що використовуються в проекті. $R^P = r_1^P, r_2^P, \dots, r_j^P$, де j – кількість видів програмних ресурсів у проекті;

R^I – Набір інформаційних ресурсів у проекті. $R^I = r_1^I, r_2^I, \dots, r_j^I$, де j – кількість видів інформаційних ресурсів у проекті.

Зацікавлені сторони, які мають прямий вплив не тільки на функціональність майбутнього продукту проекту, але і на успіх усього проекту, суттєво впливають на успіх ІТ-проекту. Сюди входять: КМДА, міністерство транспорту України, DB-розробники, Front/Back-end розробники, DevOps, проектний менеджер. Нехай $O = O^M, O^H$, де O – зацікавлені сторони.

O^H – Група зацікавлених сторін, що беруть участь у проекті. $O^H = \{o_1^H, o_2^H, \dots, o_j^H\}$, де j – кількість таких зацікавлених сторін;

O^M – Група зацікавлених сторін, що не беруть пряму участь у проекті. До такого типу можна віднести водіїв, автомеханіків та інших працівників

транспортної системи. $O^H = \{o_1^M, o_2^M, \dots, o_j^M\}$, де j – кількість таких зацікавлених сторін.

Для ІТ-проектів створення складних ІТ-продуктів характеризується сервісним компонентом, який вимагає стратегічного сервісу, служб розробки, переходів, операцій та постійних вдосконалень.

Нехай $S = S^H, S^M, S^P$, де S^H – набір послуг спрямованих на підтримку існуючої ІТ-інфраструктури. $S^H = \{s_1^H, s_2^H, \dots, s_j^H\}$, де j – кількість таких послуг/процесів;

S^M – набір послуг, що підтримують додаток. $S^M = \{s_1^M, s_2^M, \dots, s_j^M\}$, де j – кількість таких послуг/процесів;

S^P – набір служб, що надають підтримку користувачам. $S^P = \{s_1^P, s_2^P, \dots, s_j^P\}$, де j – кількість таких послуг/процесів.

Варто також і проговорити використання технологій для створення ІТ-продукту. До даної групи входить розробка, управління, тестування та експлуатація. Нехай $Z = Z^H, Z^M, Z^P$.

Z^H – набір технологій для розробки та тестування продукту проекту. $Z^H = \{z_1^H, z_2^H, \dots, z_j^H\}$, де j – кількість таких технологій;

Z^M – набір технологій управління проектами. $Z^M = \{z_1^M\}$;

Z^P – сукупність технологій реалізації та підтримки проектного продукту. $Z^P = \{z_1^P, z_2^P, \dots, z_j^P\}$, де j – кількість технологій, що використовуються для впровадження та підтримки проектного продукту.

Компоненти конфігурації ІТ-продукту пов'язані з визначенням елементів, їх параметрів та відношенням до розробленої інформаційної системи. Те саме стосується елементів проекту та середовища проекту. $K = K^P, K^S, K^E, K^{DP}, K^{DS}$, де K^P – набір параметрів проекту, $K^P = \{k_1^P, k_2^P, \dots, k_r^P\}$, де r – кількість параметрів проекту; K^S – набір параметрів проектного продукту, $K^S = \{k_1^S, k_2^S, \dots, k_d^S\}$, де d – кількість параметрів продукту проекту; K^E – це набір параметрів зовнішнього середовища проекту, $K^E = \{k_1^E, k_2^E, \dots, k_u^E\}$, де u – кількість параметрів зовнішнього середовища

проекту; K^{DP} – це набір вимог до проекту, $K^{DP} = \{k_1^{DP}, k_2^{DP}, \dots, k_o^{DP}\}$, де o – це кількість вимог (умов, можливостей та обмежень), яким повинен відповідати проект; K^{DS} – це набір вимог до продукту проекту, $K^{DS} = \{k_1^{DS}, k_2^{DS}, \dots, k_p^{DS}\}$, де p – це кількість вимог (умов, можливостей та обмежень), яким повинен відповідати проектний продукт

Доступність (при створенні продукту) визначається набором оцінок рівня виконання функцій та вимог (надійність, підтримка, справність, продуктивність, безпека).

Нехай $A = A^H, A^M, A^P, A^R, A^N$, де A^H - Сукупність оцінок, пов'язаних з надійністю. $A^H = \{a_1^H, a_2^H, \dots, a_j^H\}$, де j – кількість оцінок надійності;

A^M - Сукупність оцінок рівня підтримки. $A^M = \{a_1^M, a_2^M, \dots, a_j^M\}$, де j – кількість можливостей підтримки;

A^P - Сукупність оцінок справності. $A^P = \{a_1^P, a_2^P, \dots, a_j^P\}$, де j – кількість оцінок справності;

A^R - Сукупність оцінок ефективності. $A^R = \{a_1^R, a_2^R, \dots, a_j^R\}$, де j – кількість оцінок ефективності;

A^N - Сукупність оцінок безпеки. $A^N = \{a_1^N, a_2^N, \dots, a_j^N\}$, де j – кількість оцінок безпеки;

Виходячи з цього, математичний опис запропонованої моделі «конус» можна представити наступним чином: $M = \{X, Y, H\}$,

Де $X = \{P, R, O, S, Z, K, A\}$ - набір вхідних параметрів моделі [16];

$Y = \{C_p, T_p, Q\}$ – сукупність вихідних параметрів, на його основі ми визначимо ефективність процесів управління ІТ-проектами, де C_p – планова вартість створення елементів проекту, T_p – запланована тривалість життєвого циклу проекту, а Q – якість проекту.

H – являє собою набір каналів зв'язку між елементами моделі управління ІТ-проектами каналів зв'язку.

$H = h_1, h_2, \dots, h_e$, де e – кількість прямих зв'язків між усіма елементами моделі.

$\underline{H} = \underline{h_1}, \underline{h_2}, \dots, \underline{h_e}$, де e – кількість зворотних зв'язків між усіма елементами моделі.

У цьому випадку набір вихідних параметрів проектної моделі також може бути представлений у формі $X = \{x_{i_1} | i_1 = 1, 2, \dots, N_1\}$, де N_1 – кількість областей знань моделі M_1 .

Кінцева вартість:

$$C_p = \sum_{i_1=1}^{N_1} \sum_{j_1=1}^{T_p} \sum_{i_2=1}^e \left(C_1(x_{i_1}, t_{j_1}) + C_2(h_{i_2}) \right) \rightarrow \min \quad (2.1)$$

При обмеженнях:

$$\forall (x_{i_1} \in X) \cup (q_{i_1} \in Q) \exists t_{j_1} \in T_p, T_p \geq 0 \text{ та } C_p \leq C_b, C_b \geq 0, \quad (2.2)$$

Де C_b – бюджетна вартість проекту (інвестиції),

C_1 – функція витрат на створення елементів вхідних параметрів з $\{X\}$ та час $t_{j_1} \in T_p$,

C_2 – це функція витрат каналів зв'язку між елементами моделі з $\{X\}$.

2.3 Формалізація задачі оптимального розподілу бюджету

У столиці України – м. Києві – розвинена, але перевантажена транспортна мережа, в якій погані умови на дорогах та низька пропускна здатність призводять до значних заторів у години пік. В рамках проекту «Система обліку міської транспортної інфраструктури» ставиться задача оптимально розподілити річний бюджет міста B між об'єктами транспорту (дорогами, мостами, світлофорами, зупинками тощо) з метою мінімізації середнього часу поїздки пасажирів. Для формального опису цієї задачі введемо наступні складові:

- **Множина об'єктів інфраструктури.** Нехай $I = 1, 2, \dots, n$ – кінцевий набір об'єктів транспортної інфраструктури (автомагістралі, мости, перехрестя із світлофорами, пасажирські зупинки тощо). Кожному об'єкту $i \in I$ можна поставити у відповідність значимість (вагу) $w_i \geq 0$,

що відображає його внесок у середній час поїздки (наприклад, пропорційно транспортному потоку або довжині сегмента).

- **Параметри стану об'єктів.** Кожен об'єкт i характеризується технічним станом $c_i \in [0,1]$ (зношеністю), де $c_i = 1$ відповідає новому об'єкту, а $c_i = 0$ – повністю виведеному з експлуатації. Наприклад, для доріг часто використовують **індекс стану покриття (PCI)** від 0 до 100, який можна масштабувати на $[0,1]$. У простій моделі стан після ремонту визначатимемо як функцію від початкового стану c_i^0 і вкладень:

$$c_i = \min\{c_i^0 + \alpha_i x_i, 1\}, \quad (2.3)$$

де $x_i \geq 0$ – сума інвестицій (витрат на ремонт) для об'єкта i , а параметр $\alpha_i > 0$ показує ефективність витрат у покращенні стану. Така лінійна модель означає, що зі збільшенням фінансування стан зростає (до максимуму 1) з лінійною або слабо нелінійною залежністю.

- **Вплив інвестицій на стан.** Інвестиції x_i спрямовані на поліпшення стану: чим більше витрачено на i -й об'єкт, тим вищий його c_i . У більш загальному випадку можна припустити нелінійну модель покращення (напр., насичення), але для формулювання задачі достатньо вважати монотонним зростанням $f_i(x_i) = c_i$. Зазвичай враховують, що для кожного об'єкта існує максимум корисного відновлення.
- **Функція впливу стану на середній час поїздки.** Технічний стан об'єктів впливає на швидкість руху і затримки, а отже – на загальний час подорожі. Позначимо t_i^0 – базовий час проїзду об'єкта i за умови ідеального стану ($c_i = 1$). Тоді реальний час проїзду t_i є невід'ємною функцією $t_i(c_i)$, яка зазвичай спадає з ростом c_i . Наприклад, грубий наближений варіант – зворотно пропорційна залежність

$$t_i(c_i) = \frac{t_i^0}{c_i}, \quad (2.4)$$

що відображає ефект уповільнення при погіршенні покриття. Емпіричні дослідження підтверджують значимий вплив погіршеного покриття на

швидкість руху: так, у Khan et al. при переході від хорошого покриття до дуже поганого швидкість зменшилася на **8.8%**, а у Vock et al. було зафіксовано, що зростання середнього показника шорсткості IRI на одне стандартне відхилення призводить до **11%** падіння середньої швидкості. З цих результатів випливає, що покращення стану інфраструктури (підвищення c_i) сприятиме зниженню середнього часу поїздки. Формально середній час поїздки можна вважати середньоваговим за об'єктами:

$$T(c) = \frac{\sum_{i \in I} w_i t_i(c_i)}{\sum_{i \in I} w_i}, \quad (2.5)$$

або, еквівалентно (без нормування), мінімізувати сумарний час $\sum_i w_i t_i(c_i)$.

- **Бюджетні обмеження.** Задано річний бюджет B на транспортну інфраструктуру. Інвестиційні витрати мають задовольняти лінійне обмеження

$$\sum_{i \in I} x_i \leq B, \quad x_i \geq 0 \quad \forall i \in I \quad (2.6)$$

При цьому варто врахувати практичні верхні межі на x_i (неможливість надмірного ремонту) через обмеження стану: $c_i \leq 1$ дає $x_i \leq (1 - c_i^0)/\alpha_i$.

- **Цільова функція.** Метою є мінімізувати середній час поїздки, тобто

$$\min_{\{x_i\}} T(c(x)) = \min_x \sum_{i \in I} w_i t_i(c_i^0 + \alpha_i x_i), \quad (2.7)$$

за умов бюджетного обмеження. Звичайно, $t_i(\cdot)$ – невід’ємна спадаюча функція, тому задача є (потенційно) нелінійною оптимізацією з лінійними обмеженнями. У особливому випадку лінійних залежностей ($t_i \propto 1/c_i$, $c_i \propto x_i$) це задача з опуклою ціллю, яку можна вирішувати стандартними методами оптимізації.

Таким чином, математична модель виглядає так: задати $I = \{1, \dots, n\}$, $c_i^0 \in [0, 1]$, $t_i^0 > 0$, $w_i > 0$, ввести змінні $x_i \geq 0$ та функції

$$c_i = c_i^0 + \alpha_i x_i, \quad t_i = t_i(c_i), \quad (2.8)$$

де $c_i \leq 1$ та $t_i(c_i)$ – заданий спадаючий графік (напр., $t_i(c_i) = t_i^0/c_i$). Тоді рішення шукається із формулювання:

$$\min_{x_1, \dots, x_n} \sum_{i=1}^n w_i t_i(c_i^0 + \alpha_i x_i) \quad (2.9)$$

$$\text{за умов } \sum_{i=1}^n x_i \leq B, \quad x_i \geq 0, \quad c_i^0 + \alpha_i x_i \leq 1.$$

Постановка охоплює вказані елементи: множину об'єктів I , параметри стану c_i^0 , залежність стану від інвестицій через α_i , функцію впливу стану $t_i(c_i)$ на час поїздки, бюджетне обмеження та цільову функцію мінімізації сумарного часу.

Вибір моделі обґрунтуємо наступним чином. Врахування функції $t_i(c_i)$ дозволяє використовувати результати попередніх досліджень, згідно з якими погіршення стану покриття збільшує час поїздки і зменшує середню швидкість. Вплив інвестицій моделюється за допомогою коефіцієнта α_i , що відображає чутливість стану до фінансування (аналогічні моделі застосовуються у дослідженнях дорожнього утримання). Запропонована оптимізаційна модель узгоджується з підходами з математичного планування ресурсів: наприклад, Ma et al. використовували динамічне програмування для мінімізації затримок від ремонтних робіт на дорогах, а тут ми формулюємо нерухому (одноетапну) задачу з подібною метою.

Завдяки таким визначенням модель може бути розв'язана стандартними методами нелінійної оптимізації. Вона має властивість розширюваності – наприклад, у разі багаторічного планування можна ввести індекс часу і моделювати еволюцію c_i по роках, або додати сценарії попиту та розподілу потоків за дорогами. Описана формалізація служить концептуальним каркасом для рішення задачі розподілу бюджетних коштів на транспортну інфраструктуру міста Києва з метою зменшення затримок у русі і підвищення ефективності транспортної системи [14, 29, 35].

2.4 Обґрунтування обраного підходу до ефективного управління проєктами

Сучасний етап розвитку інформаційних технологій характеризується високим рівнем невизначеності вимог, динамічністю технологічного стеку та критичними вимогами до швидкості виведення продуктів на ринок (Time-to-Market). У контексті розробки системи обліку міської інфраструктури ситуація ускладнюється необхідністю інтеграції великих масивів геопросторових

даних, дотриманням жорстких бюджетних обмежень муніципального замовника (КМДА) та забезпеченням високої безпеки даних.

Як зазначено у дослідженнях [1, 27, 32, 34], традиційні методології управління проєктами, такі як Waterfall (каскадна модель), орієнтовані на детальне попереднє планування та незмінність обсягів робіт. Це дозволяє чітко прогнозувати бюджет, що є ключовою вимогою для державного сектору, проте створює ризики при виявленні нових технічних нюансів на етапі реалізації. З іншого боку, гнучкі методології (Agile, Scrum) забезпечують високу адаптивність, але часто не дають гарантій щодо фінальних термінів та повної вартості проєкту. Аналіз публікацій останніх років свідчить, що понад 50% ІТ-проєктів стикаються з проблемами саме через невідповідність обраної моделі управління специфіці продукту [1].

Архітектура фреймворку

Фреймворк складається з трьох взаємопов'язаних рівнів – рис.2.2.



Рис. 2.2. Гібридна модель управління проєктом

Процесний рівень (РМВОК-компонент): формалізоване ініціювання, планування, виконання, моніторинг і завершення проєкту; використання шаблонів планів управління обсягом, ризиками, комунікацією згідно з РМВОК [1].

Ітеративний рівень (Scrum-компонент): впровадження спринтів, щоденних стендапів, беклогу продукту та ретроспектив; ролі Product Owner, Scrum Master і команда розробки забезпечують гнучкість та швидку адаптацію [4].

Технологічний рівень (цифрова підтримка): інтеграція Jira для керування задачами й беклогом, GitLab для контролю версій і CI/CD, Microsoft Teams — для комунікації [12]–[13]; розробка дашборду KPI з автоматизованим збором даних із зазначених платформ, що забезпечує моніторинг у режимі реального часу [14].

Цифрові платформи та інструменти автоматизації моніторингу.

Ефективність запропонованого фреймворку безпосередньо залежить від використання сучасних цифрових платформ, що дозволяють здійснювати моніторинг у реальному часі. У межах дослідження обґрунтовано використання наступного інструментарію.

Jira Software та GitLab: ці платформи виступають єдиним джерелом істини (Single Source of Truth) для команди. Вони дозволяють візуалізувати потік робіт через Канбан-дошки, відстежувати продуктивність команди (Velocity Chart) та автоматизувати процес розгортання коду (CI/CD).

Хмарна аналітика та обчислення: враховуючи необхідність обробки значних обсягів просторових даних, використання хмарних ресурсів дозволяє динамічно масштабувати обчислювальні потужності. Це критично для математичного моделювання транспортних потоків, де складність обчислень може змінюватися залежно від обраного району моделювання.

Середовища для колаборації: Microsoft Teams та Slack забезпечують безперервну комунікацію, що є фундаментальним фактором для гнучких команд.

Експериментальне підтвердження та ефективність

Експериментальні дослідження впровадження подібних гібридних рішень у сучасних ІТ-компаніях демонструють позитивну динаміку ключових

показників успішності [1]. Зокрема, аналіз ретроспективних кейсів дозволяє стверджувати, що застосування такого фреймворку забезпечує:

- Дотримання часових та бюджетних обмежень завдяки чіткому плануванню верхнього рівня.
- Підвищення якості продукту за рахунок постійного зворотного зв'язку та адаптивності командної роботи.
- Мінімізацію ризиків на кожному етапі реалізації проєкту завдяки автоматизованому моніторингу відхилень.

Таким чином, обраний гібридний підхід є методичним фундаментом для подальшої математичної постановки задачі та розробки системи, оскільки він дозволяє збалансувати технічну складність об'єкта з управлінською гнучкістю процесу розробки [17].

2.5 Створення кошторису проєкту інструментом Altinvest

Для проведення детального фінансово-економічного аналізу та побудови прогнозу життєдіяльності проєкту було використано професійну систему інвестиційного моделювання Alt-Invest [20, 36]. Цей інструментарій дозволив автоматизувати розрахунок грошових потоків, оцінити рентабельність та визначити чутливість проєкту до зовнішніх факторів.

2.5.1 Формування операційного плану (доходи та витрати)

Визначимо основні показники для розрахунків [14] – рис 2.3.

Проект розробки і впровадження системи обліку міської транспортної інфраструктури			
ЗАГАЛЬНІ ДАНІ			
Тривалість інтервалу планування проєкту (ІП)		дні	30
Строк життя проєкту (ЖЦП)	24	міс	+/-
Дата початку проєкту			01.06.2026
Місцева валюта (основна назва)			тис.грн.
Місцева валюта (додаткова назва)			грн.
Іноземна валюта (основна назва)			тис.дол.
Іноземна валюта (додаткова назва)			дол.
Валюта Результатів	Місцева		1
Метод розрахунків	Постійні ціни		1

Рис. 2.3. Параметри проєкту

Аналіз структури витрат (Costs). Витратна частина проєкту була детально деталізована за статтями, що дозволило точно визначити точку беззбитковості – деталі у таблиці 2.2. Усі витрати розподілені на:

- Прямі витрати на персонал: Це найбільша стаття видатків. До неї включено заробітну плату основної команди розробників (Backend, Frontend, QA, DevOps) та адміністративного персоналу. Сумарні витрати на оплату праці разом із нарахуваннями становлять понад 40 000 тис. грн за весь період проєкту.
- Експлуатаційні та загальногосподарські витрати: Включають оренду офісу, ліцензії на спеціалізоване ПЗ та оплату хмарних серверів (AWS/Azure) для розгортання системи. Ці витрати становлять приблизно 2 500 – 3 000 тис. грн щомісяця протягом фази активної розробки.
- Податки та відрахування: Модель автоматично враховує нарахування на заробітну плату та ПДВ, що дозволяє бачити реальний "чистий" фінансовий результат.

Таблиця 2.2

Витрати за категоріями

Стаття витрат	Кількість	Тривалість (к-сть місяців)	Ціна за шт./міс. – тис. грн.	Загальна сума, тис. грн.
1	2	3	4	5
Категорія витрат: трудові ресурси				52 026
Проектний менеджер	1	24	205	4 920
Системний архітектор	1	18	227,5	4 095
Розробники BE	3	18	182,0	9 828
Розробники FE	3	18	159,0	8 586
Тестувальники	2	15	114,0	3 420

Бізнес-аналітики	2	12	136,0	3 264
Тех. письменник / Тренер	2	7	91,0	1 274
Support Engineer	1		91,0	637
Категорія витрат: обладнання та інфраструктура				5 870
Хмарна інфраструктура (AWS/Azure)	1	24	117	2 830
Ліцензії на ПЗ та API	1	24	78	1 870
Витратні матеріали (офіс/зв'язок)	1	24	49	1 170
Категорія витрат: податки				20 579
Податок на додану вартість (ПДВ)	—	10–24	—	12 831
Податок на прибуток	—	18–24	—	3 065
Інші податки (майно, збори)	—	24	—	4 683
Категорія витрат: фінансові витрати				2 959
Виплачені відсотки (кредит 10 млн, 25%)	—	24	—	2 959
Загальні витрати				81 434

Трудові ресурси: Найбільша стаття витрат проєкту (~60%). Розрахунок включає динамічну зміну штату: наприклад, бізнес-аналітики працюють на першому етапі, а спеціалісти підтримки та тренери з'являються після запуску Альфа-версії.

Інфраструктура: Враховано оренду серверних потужностей та ліцензії, що конвертуються з іноземної валюти (долар) у гривню за курсом моделі.

Податкове навантаження: Проєкт передбачає значні виплати ПДВ (12,8 млн грн) та податку на прибуток, що відображає високу маржинальність продукту після релізу.

Обслуговування боргу: Відсотки за кредитом складають 2,96 млн грн, що є ціною підтримки ліквідності (Cash Flow) на етапах розробки.

Планування доходної частини (Sales). Стратегія доходів базується на етапності впровадження системи обліку міської інфраструктури. Оскільки проєкт має тривалий цикл розробки, надходження розподілені наступним чином.

Етап 1: Альфа-версія (10-й місяць). На цьому етапі замовнику передається мінімально життєздатний продукт (MVP), що приносить перший значний дохід у розмірі 22 000 тис. грн.

Етап 2: Фінальний реліз (18-й місяць). Після повного тестування та інтеграції всіх модулів здійснюється фінальна оплата за продукт у розмірі 30 000 тис. грн.

Етап 3: Сервісна підтримка (19–24 місяці). Перехід на модель регулярних платежів за технічне супроводження та оновлення системи. Щомісячний платіж становить 2 500 тис. грн, що забезпечує стабільність грошового потоку після завершення основної розробки. Графік Sales&Costs на рисунку 2.4.

Чистий прибуток та рентабельність – рис. 2.5. Завдяки збалансованій вартості продукту, проєкт демонструє позитивну динаміку прибутку. Незважаючи на збитковість у перші 9 місяців (інвестиційна фаза), вже після 10-го місяця накопичений чистий прибуток починає зростати, досягаючи 8 700 тис. грн на кінець 2-го року.

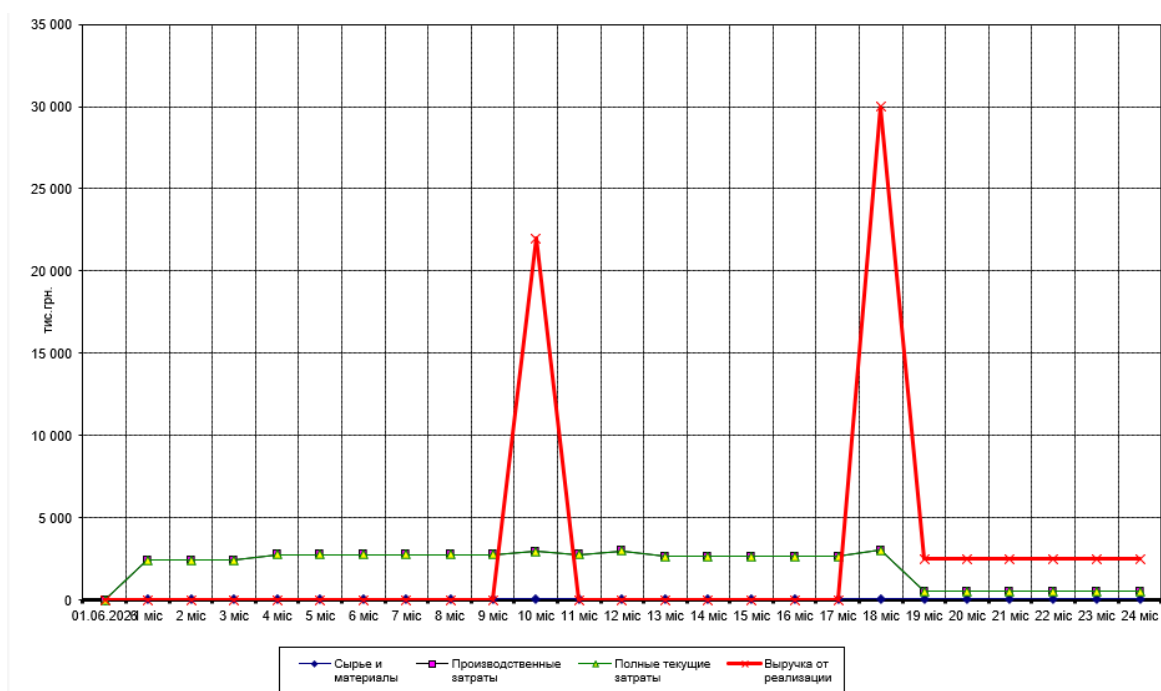


Рис. 2.4. Sales&Costs

Це підтверджує реалістичність моделі: високі витрати на висококваліфікованих ІТ-спеціалістів повністю перекриваються вартістю інноваційного продукту, що впроваджується в міську інфраструктуру.

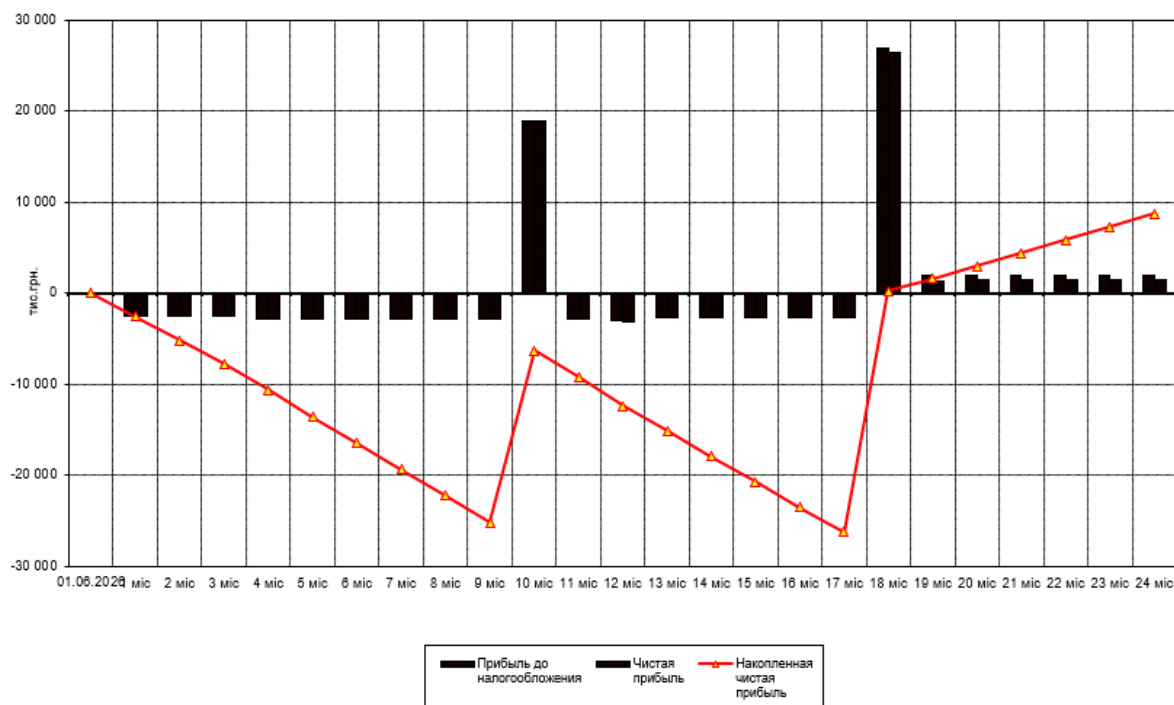


Рис. 2.5. Графік прибутку

2.5.2 Інвестиційний план та джерела фінансування

Особливістю даного проєкту є змішана модель фінансування, що мінімізує ризики касових розривів.

- Власний капітал та гранти: На старті проєкту залучено 1 000 тис. грн статутного капіталу та 40 000 тис. грн цільового грантового фінансування.
- Позиковий капітал: Для покриття пікових потреб у ліквідності передбачено кредитну лінію у розмірі 10 000 тис. грн під 25% річних.

2.5.3 Аналіз руху грошових коштів (Cash Flow)

Звіт про рух грошових коштів є найбільш інформативним показником життєздатності проєкту – рис.2.6. Завдяки вчасному фінансуванню, показник

вільних грошових коштів протягом усього періоду розробки залишається позитивним.

Запас міцності: Мінімальний залишок коштів на рахунку не опускається нижче 25 264 тис. грн, що свідчить про високу фінансову стійкість.

Фінальний результат: На кінець 24-го місяця накопичений залишок коштів становить 50 235 тис. грн.

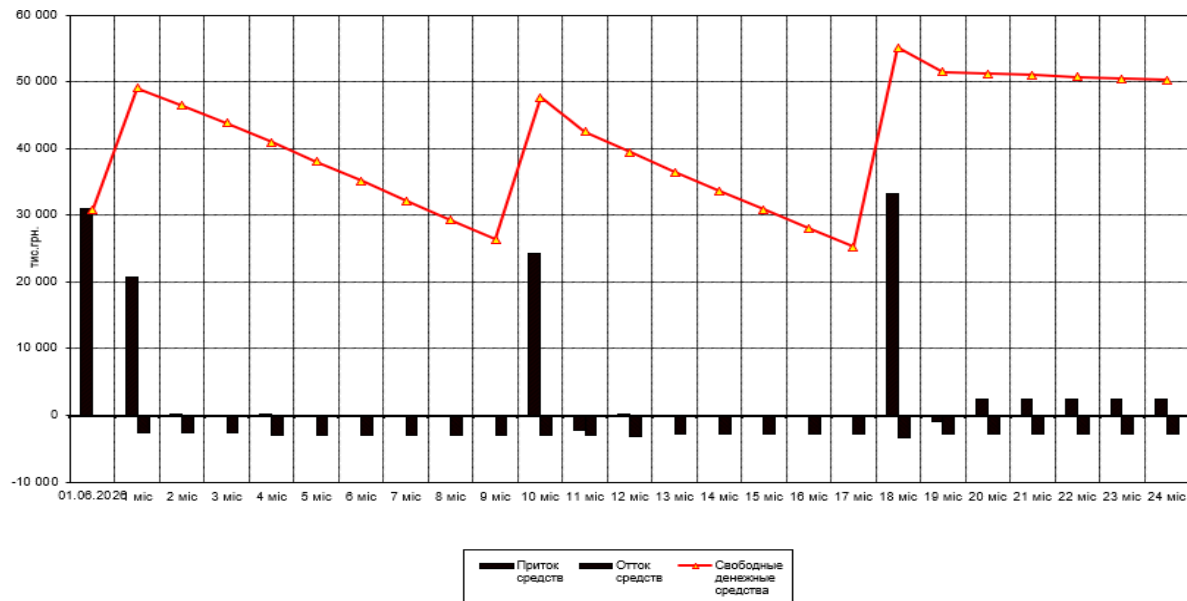


Рис. 2.6 Графік руху коштів

2.5.4 Ключові показники ефективності проєкту

За результатами моделювання отримано наступні інтегральні показники інвестиційної привабливості – рис.2.7.

Економічна ефективність (NPV): Показник чистої приведеної вартості на рівні **10 376 тис. грн** свідчить про те, що проєкт є не просто прибутковим, а здатним генерувати додану вартість понад встановлену ставку дисконтування (12%). Це означає, що доходи від реалізації системи обліку міської інфраструктури повністю покривають усі операційні та інвестиційні витрати з урахуванням часової вартості грошей.

Запас міцності та рентабельність (IRR): Внутрішня норма рентабельності (**54%**) суттєво перевищує як ставку дисконтування (12%), так і

вартість залученого кредитного капіталу (25%). Це вказує на високу стійкість проєкту до можливих ризиків (наприклад, зростання витрат або коливання валютного курсу) та робить його привабливим для потенційних інвесторів.

Швидкість повернення капіталу (PB): Період окупності у **1,5 року** є дуже коротким для складних ІТ-систем з муніципальним спрямуванням. Враховуючи, що загальний термін життєвого циклу моделі складає 24 місяці, проєкт виходить на повну самоокупність вже всередині другого року, що мінімізує тривалість перебування капіталу в зоні фінансового ризику.

	A	C	D	F	G	H	P	Q	R	X	Y	Z	AA	AB	AC	AD	AE	AF
	ОСНОВНІ ПОКАЗНИКИ ПРОЄКТУ			01.06.2026	1 МІС	2 МІС	10 МІС	11 МІС	12 МІС	18 МІС	19 МІС	20 МІС	21 МІС	22 МІС	23 МІС	24 МІС		ВСЬОГО
1355	Тривалість інтервалу планування		дні	30														
1357	Строк життєвого циклу проєкту		міс	24														
1360	Надходження від продажів		тис. грн.	0	0	0	22 000	0	0	30 000	2 500	2 500	2 500	2 500	2 500	2 500		67 000
1361	Поточні витрати		тис. грн.	0	2 456	2 456	2 985	2 765	3 012	3 067	540	540	540	540	540	540		52 249
1364	Податки та відрахування до позабюджетні фонди, мито		тис. грн.	0	841	841	1 172	952	1 040	1 573	618	621	625	628	632	636		21 270
1366	НДВ до бюджету (+) з бюджету (-)		тис. грн.	0	0	0	4 150	0	0	5 818	477	477	477	477	477	477		12 831
1367	Чистий прибуток		тис. грн.	0	-2 595	-2 595	18 876	-2 904	-3 151	26 474	1 374	1 392	1 411	1 430	1 450	1 469		8 700
1369	Те ж, зростаючим підсумком		тис. грн.	0	-2 595	-5 190	-6 334	-9 238	-12 389	174	1 548	2 940	4 351	5 781	7 231	8 700		8 700
1371	Дивиденди		тис. грн.	0	0	0	0	0	0	0	0	0	0	0	0	0		0
1373	Потреба у фінансуванні постійних активів		тис. грн.	0	0	0	0	0	0	0	0	0	0	0	0	0		0
1374	Потреба у фінансуванні чистого обігового капіталу		тис. грн.	137	-755	23	-2 412	2 208	-55	-3 417	3 455	-2	-2	-2	-2	-2		-
1376	Чисті доходи для повних інвестиційних витрат		тис. грн.	-137	-1 702	-2 480	21 426	-4 974	-2 959	30 029	-1 944	1 509	1 506	1 502	1 499	1 495	-535	11 659
1378	Те ж, зростаючим підсумком		тис. грн.	-137	-1 839	-4 319	-2 013	-6 988	-9 946	6 628	4 684	6 193	7 699	9 201	10 700	12 194	11 659	11 659
1379	Ставка порівняння (номінальна річна)		%	12%														
1381	NPV		тис. грн.	10376														
1382	IRR (номінальна річна)		%	54%														
1383	Простий строк окупності		лет	1,5														
1384	Дисконтований строк окупності		лет	1,5														
1386	Зростання статутного капіталу та цільове фінансування		тис. грн.	21 000	20 000	0	0	0	0	0	0	0	0	0	0	0		41 000
1387	Залучення кредитів		тис. грн.	10 000	0	0	0	0	0	0	0	0	0	0	0	0		10 000
1389	Погашення заборгованості		тис. грн.	0	0	0	0	0	0	0	-1 610	-1 632	-1 655	-1 678	-1 701	-1 724		-10 000
1391	Виплати відсотків по кредитах		тис. грн.	0	-137	-137	-137	-137	-137	-137	-137	-115	-93	-70	-47	-24		-2 959
1393	Загальний коефіцієнт покриття боргу		рази	-	357,87	338,82	347,60	310,39	287,85	402,51	30,45	30,31	30,17	30,03	29,89	29,74		
1395	Вільні грошові кошти		тис. грн.	30 864	49 024	46 406	47 613	42 501	39 405	55 155	51 464	51 225	50 983	50 738	50 488	50 235	0	50 235
1397																		

Рис. 2.7 Основні показники проєкту

Загальний фінансовий висновок. Проєкт характеризується високою фінансовою стійкістю та ефективністю. Завдяки оптимальному поєднанню власного капіталу, цільового фінансування (40 млн грн) та кредитної лінії (10 млн грн), вдалося досягти стабільного залишку вільних грошових коштів на рівні 50 235 тис. грн на кінець розрахункового періоду.

РОЗДІЛ 3. РОЗРОБКА ІНФОРМАЦІЙНОГО ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПРОЄКТУ

3.1 Розробка концептуальної моделі бази даних проєкту.

Діаграма зв'язку сутностей, також відома як ERD [7, 12], ER-діаграма або ER-модель, є типом структурної діаграми для використання в розробці бази даних. ERD містить різні символи та з'єднувачі, які візуалізують дві важливі відомості: основні сутності в системі та взаємозв'язки між цими сутностями. Коли ми говоримо про сутності в ERD, дуже часто маємо на увазі такі бізнес-об'єкти, як люди/ролі (наприклад, студент), матеріальні бізнес-об'єкти (наприклад, продукт), нематеріальні бізнес-об'єкти (наприклад, журнал) тощо. «Відношення» — це те, як ці сутності пов'язані один з одним у системі.

Діаграми зв'язків сутностей забезпечують візуальну відправну точку для проектування бази даних, яку також можна використовувати для визначення вимог до інформаційної системи в організації. Після розгортання реляційної бази даних ERD все ще може служити точкою відліку, якщо пізніше знадобиться будь-яке налагодження або реорганізація бізнес-процесу.

Однак, хоча ERD може бути корисною для організації даних, які можуть бути представлені реляційною структурою, вона не може належним чином представляти неструктуровані дані, а також навряд чи буде корисною сама по собі для інтеграції даних у вже існуючу інформаційну систему.

Зобразимо концептуальну модель бази даних за допомогою ER для каталогу об'єктів – рис. 3.1.

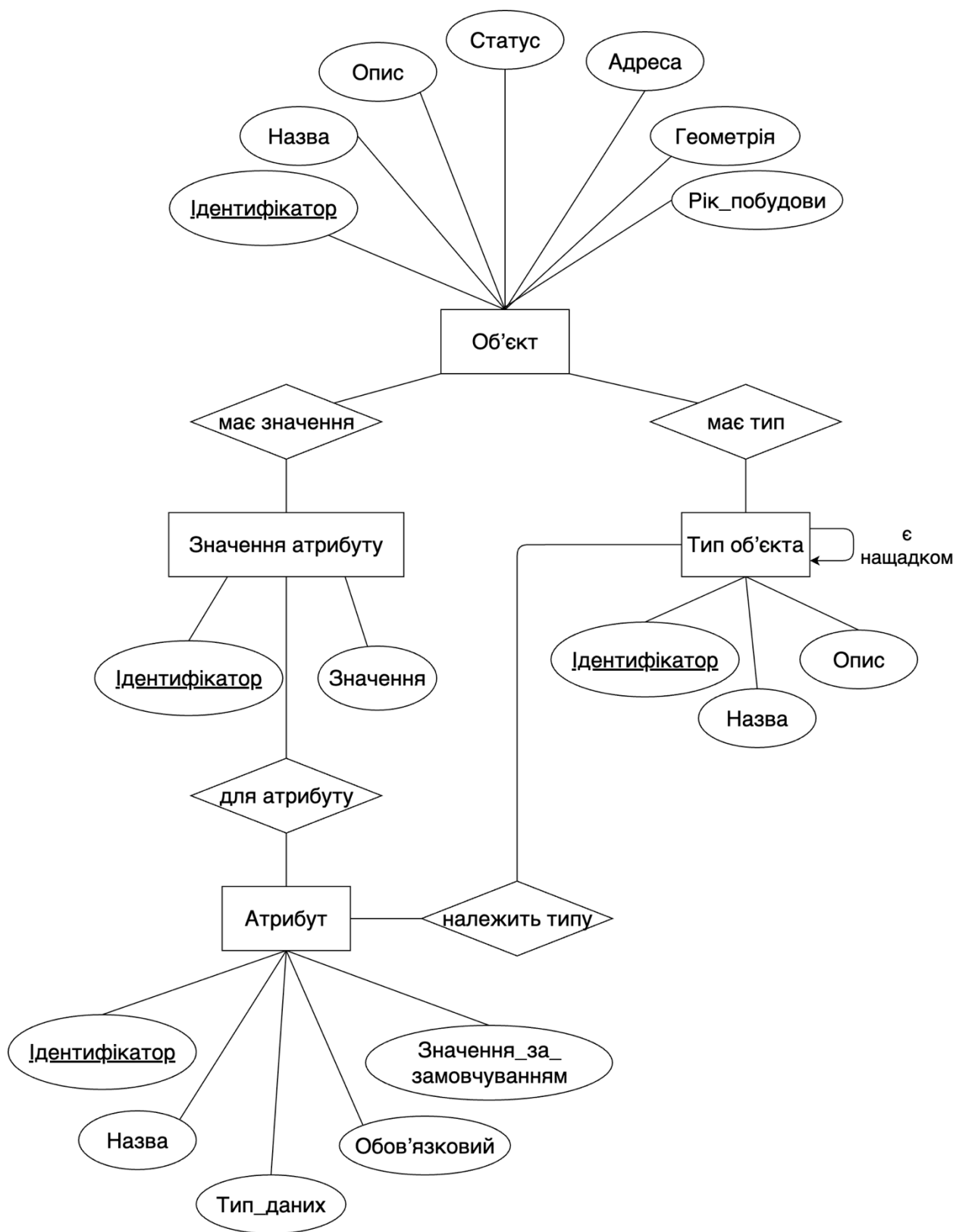


Рис. 3.1. Концептуальна модель бази даних каталогу об'єктів

Ця діаграма ілюструє концептуальну структуру метамодельного підходу до зберігання і організації даних про об'єкти транспортної інфраструктури. Вона дозволяє гнучко описувати будь-які типи об'єктів (наприклад, мости,

дороги, зупинки, світлофори), задаючи їм атрибути, зв'язки, геодані та значення. Основна ідея — розділення опису структури даних (метаданих) від самих даних, що дозволяє розширювати модель без зміни схеми бази даних або логіки застосунку. Детальний опис основних сутностей моделі наведено у таблиці 3.1.

Об'єкт — основна сутність, яка описує конкретну одиницю інфраструктури. Має базові поля: ідентифікатор, назву, опис, рік побудови, статус, адресу, геометрію (координати або полігон).

Тип об'єкта — категорія, до якої належить об'єкт (наприклад, "Зупинка" або "Трамвайна колія"). Типи можуть утворювати ієрархії через зв'язок «є підтипом».

Атрибут — визначає, які властивості доступні для об'єктів певного типу (наприклад, "довжина", "матеріал", "інтенсивність трафіку"). Атрибут описується назвою, типом даних, значенням за замовчуванням, обов'язковістю.

Значення атрибуту — конкретні значення для певного атрибуту в межах конкретного об'єкта.

Таблиця 3.1

Опис концептуальної моделі БД

Назва сутності	Опис сутності	Назва параметру	Опис параметру
1	2	3	4
Об'єкт	Основна одиниця інфраструктури, що підлягає обліку та моніторингу.	Ідентифікатор	Унікальний системний ідентифікатор об'єкта.
		Назва	Коротке найменування об'єкта для відображення.
		Опис	Розширені текстові відомості про об'єкт.
		Рік побудови	Календарний рік введення в експлуатацію.
		Статус	Поточний стан (діє, ремонт, консервація).
		Адреса	Текстове посилання на місцезнаходження.
		Геометрія	ГІС-дані (координати точки або контури полігону).

1	2	3	4
Тип об'єкта	Категорія, яка визначає логічну групу та структуру властивостей об'єкта.	Ідентифікатор	Унікальний ідентифікатор типу.
		Назва типу	Найменування категорії (напр., "Зупинка").
		Опис	Розширені текстові відомості про тип.
Атрибут	Опис конкретної властивості, що доступна для певного типу об'єктів.	Ідентифікатор	Унікальний ідентифікатор атрибуту.
		Назва атрибуту	Найменування характеристики (напр., "Матеріал").
		Тип даних	Формат значення (рядок, число, дата, логічне).
		Значення за замовчуванням	Значення, що встановлюється автоматично.
		Обов'язковий	Логічний прапор (True/False) щодо вимоги заповнення.
Значення атрибуту	Конкретне фактичне наповнення атрибута для певного екземпляра об'єкта.	Ідентифікатор	Системний ідентифікатор запису значення.
		Значення	Фактичні дані (напр., "Бетон", "12.5 м").

Основні зв'язки:

- Об'єкт має тип (зв'язок із Типом об'єкта)
- Тип об'єкта має атрибути
- Об'єкт має значення, які відповідають атрибутам
- Тип об'єкта може успадковувати інший тип

Ця модель є основою каталогу інфраструктури та підтримує динамічне розширення — можна додавати нові типи об'єктів, нові атрибути, змінювати властивості без втручання в код. Вона критично важлива для уніфікації даних у масштабній міській системі обліку.

3.2 Побудова логічної моделі бази даних проєкту.

Логічна модель ERD [13]— це детальна версія концептуальної моделі ERD. Логічна модель ER розроблена для збагачення концептуальної моделі

шляхом явного визначення стовпців у кожній сутності та введення операційних і транзакційних сутностей.

Зобразимо логічну модель бази даних – рис. 3.2. Детальний опис основних сутностей моделі наведено у таблицях 3.2-3.3.

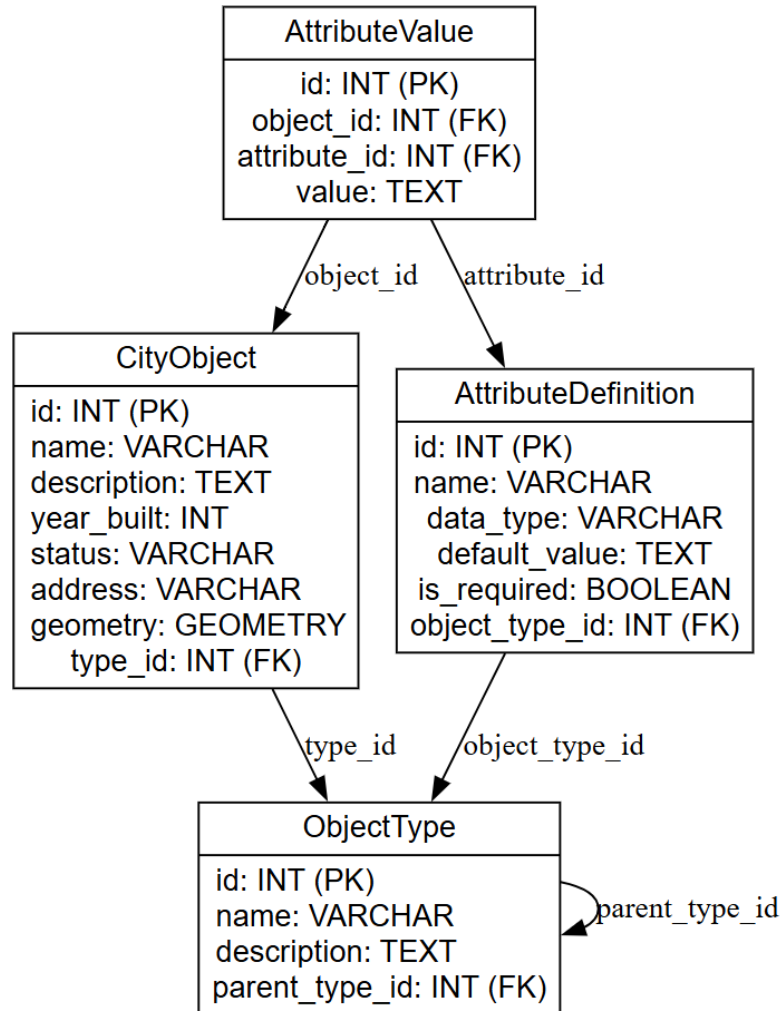


Рис. 3.2. Логічна модель бази даних каталогу інфраструктури

Основні сутності:

CityObject — основна таблиця, що представляє інфраструктурні об'єкти (мости, дороги, зупинки тощо). Містить такі поля: `id` (первинний ключ), `name`, `description`, `year_built`, `status`, `address` (основні атрибути об'єкта), `geometry` (геопросторове представлення), `type_id` (зовнішній ключ на тип об'єкта).

ObjectType — типи об'єктів, які дозволяють класифікувати об'єкти за видами (наприклад, "Міст", "Світлофор"). Підтримується ієрархія типів через поле parent_type_id (зовнішній ключ на батьківський тип).

AttributeDefinition — опис атрибутів, які можна додавати до типів об'єктів. Атрибути мають такі властивості: id, name, data_type (string, number, date тощо), default_value, is_required (чи є обов'язковим), object_type_id (зовнішній ключ на тип об'єкта).

AttributeValue — значення атрибутів, задані для конкретного об'єкта. Поля: id, object_id (зовнішній ключ на об'єкт), attribute_id (зовнішній ключ на атрибут), value (конкретне значення, збережене у вигляді рядка).

Зв'язки:

- CityObject має зовнішній ключ type_id, який вказує на ObjectType.
- ObjectType може мати parent_type_id для побудови ієрархії типів.
- AttributeDefinition містить посилання object_type_id на тип об'єкта, до якого належить атрибут.
- AttributeValue пов'язує конкретний об'єкт (object_id) із конкретним атрибутом (attribute_id), зберігаючи його значення (value).

Таблиця 3.2

Загальний опис структури бази даних

Назва таблиці	Опис	Первинні ключі	Вторинні ключі
CityObject	Зберігає основні дані про фізичні одиниці інфраструктури та їх геометрію.	id	type_id
ObjectType	Визначає класифікацію об'єктів та підтримує ієрархічне успадкування.	id	parent_type_id
AttributeDefinition	Описує метадані: перелік доступних характеристик для кожного типу об'єкта.	id	object_type_id
AttributeValue	Містить фактичні значення додаткових характеристик для конкретних екземплярів об'єктів.	id	object_id, attribute_id

Опис полів логічної моделі бази даних

Назва таблиці	Назва поля	Опис поля
CityObject	id	Унікальний ідентифікатор об'єкта.
CityObject	name	Коротка назва інфраструктурної одиниці.
CityObject	description	Розширений опис технічних характеристик або особливостей.
CityObject	year_built	Рік спорудження або введення в експлуатацію.
CityObject	status	Поточний стан (експлуатується, ремонт, аварійний).
CityObject	address	Поштова адреса або топографічне посилання.
CityObject	geometry	Геопросторові дані об'єкта (координати точки, лінія або полігон).
CityObject	type_id	Посилання на категорію з таблиці ObjectType.
ObjectType	id	Унікальний ідентифікатор типу.
ObjectType	name	Назва категорії (наприклад: "Трамвайне полотно", "Міст").
ObjectType	description	Опис типу об'єкту.
ObjectType	parent_type_id	Посилання на батьківський тип для створення ієрархії.
AttributeDefinition	id	Унікальний ідентифікатор атрибута.
AttributeDefinition	name	Назва характеристики (наприклад: "Матеріал покриття").
AttributeDefinition	data_type	Тип даних значення (string, integer, boolean, date).
AttributeDefinition	default_value	Значення за замовчуванням, якщо інше не вказано користувачем.
AttributeDefinition	is_required	Ознака обов'язковості заповнення (Boolean).
AttributeDefinition	object_type_id	Посилання на тип об'єкта, до якого відноситься атрибут.
AttributeValue	id	Унікальний ідентифікатор запису значення.
AttributeValue	object_id	Посилання на конкретний об'єкт із CityObject.
AttributeValue	attribute_id	Посилання на опис атрибута з AttributeDefinition.
AttributeValue	value	Фактичне значення характеристики, збережене у текстовому форматі.

3.3 Побудова фізичної моделі бази даних з використанням СУБД PostgreSQL.

Фізична модель ERD [3, 19, 31, 40] представляє фактичний проєкт реляційної бази даних. Фізична модель даних розробляє логічну модель даних, призначаючи кожному стовпцю тип, довжину, значення тощо. Оскільки фізична модель ERD представляє, як дані мають бути структуровані та пов'язані в конкретній СУБД, важливо враховувати угоду та обмеження фактичної системи баз даних, у якій буде створена база даних. Необхідно переконатися, що типи стовпців підтримуються СУБД і в іменуванні сутностей і стовпців не використовуються зарезервовані слова.

Опишемо фізичну модель БД – рисунок 3.3.

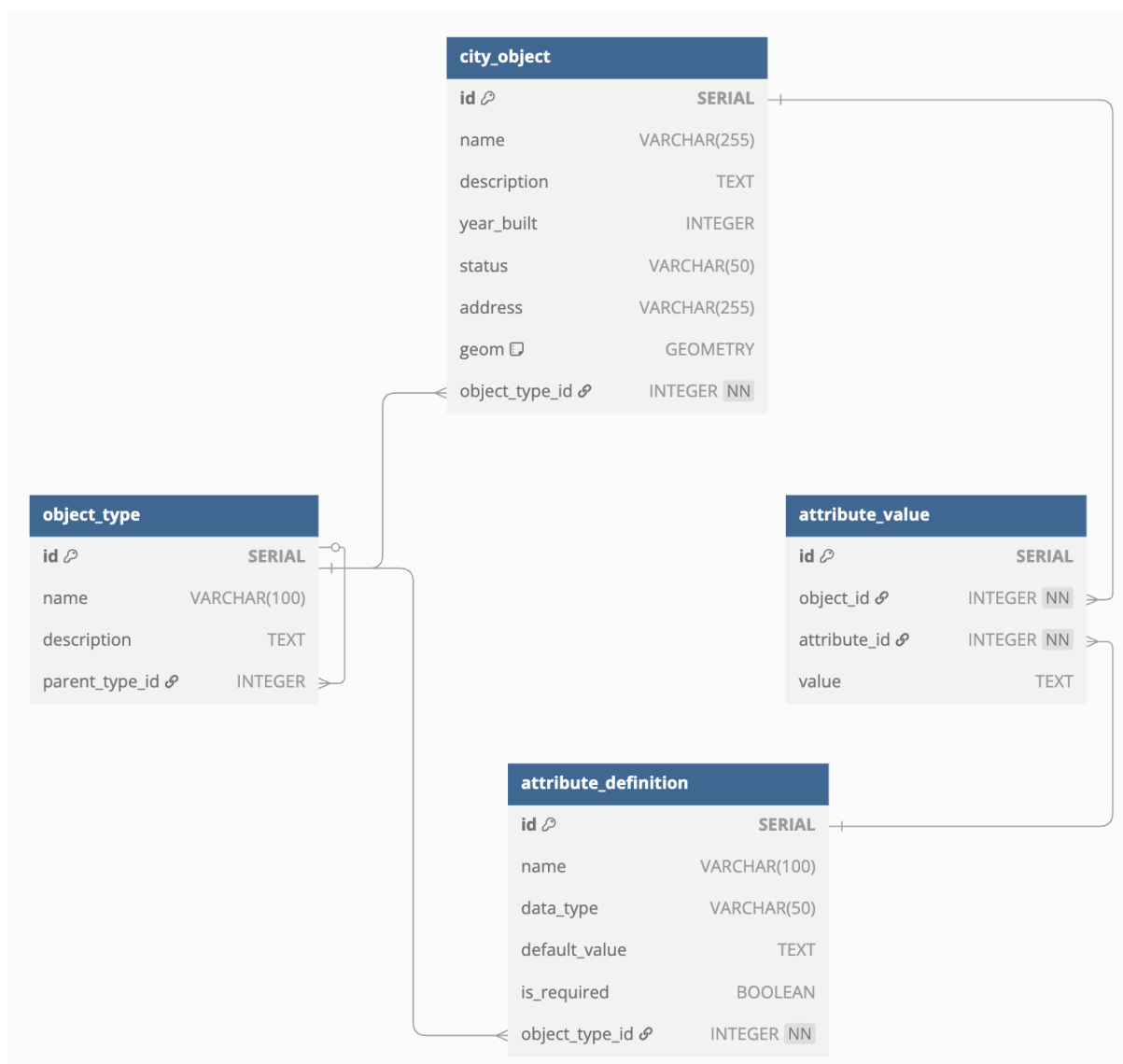


Рис. 3.3. Фізична модель бази даних каталогу інфраструктури

Ця фізична модель реалізує гнучку структуру зберігання даних про будь-які типи інфраструктурних об'єктів (дороги, мости, зупинки, світлофори, рухомий склад тощо) з можливістю динамічного розширення структури без зміни схеми бази даних.

Нижче наведено детальний опис фізичної моделі бази даних, структурований за кожною таблицею. Ця специфікація відповідає архітектурі EAV (Entity-Attribute-Value) з підтримкою геопросторових даних.

City_object — основна таблиця об'єктів – опис полів в табл.3.4. Містить базову інформацію про кожен об'єкт (назва, опис, рік побудови, адреса, статус, геометрія), а також зв'язок з відповідним типом через object_type_id.

Таблиця 3.4

Опис таблиці «City object» (Об'єкти інфраструктури)

№	Назва	Тип	Опис	Поле в БД
1	Ідентифікатор	INT (PK)	Унікальний номер об'єкта в системі	id
2	Назва об'єкта	VARCHAR	Коротка назва (напр., "Міст Патона")	name
3	Опис	TEXT	Детальні відомості про об'єкт	description
4	Рік побудови	INT	Рік введення в експлуатацію	year_built
5	Статус	VARCHAR	Поточний стан (діє, ремонт тощо)	status
6	Адреса	VARCHAR	Місцезнаходження об'єкта	address
7	Геометрія	GEOMETRY	Координати або полігон (PostGIS формат)	geometry
8	Тип об'єкта	INT (FK)	Посилання на категорію з ObjectType	type_id

Object_type — довідник типів об'єктів – опис полів в табл.3.5. Кожен тип може мати батьківський тип, що дозволяє реалізувати ієрархію (успадкування властивостей). Наприклад, тип "Транспортна споруда" може мати підтипи "Міст", "Переїзд", "Зупинка".

Опис таблиці «Object type» (Типи об'єктів)

№	Назва	Тип	Опис	Поле в БД
1	Ідентифікатор	INT (PK)	Унікальний номер типу	id
2	Назва типу	VARCHAR	Найменування (напр., "Зупинка")	name
3	Опис	TEXT	Детальні відомості про тип	description
4	Батьківський тип	INT (FK)	Посилання на вищу категорію в ієрархії	parent_type_id

Attribute_definition — дозволяє адміністратору системи визначати додаткові (динамічні) атрибути для конкретного типу об'єкта— опис полів в табл.3.6. Наприклад, для "Мосту" можна додати атрибути "матеріал", "довжина", "кількість опор" тощо. Атрибути мають тип даних, значення за замовчуванням і позначку про обов'язковість.

Таблиця 3.6

Опис таблиці «Attribute definition» (Опис додаткових атрибутів)

№	Назва	Тип	Опис	Поле в БД
1	Ідентифікатор	INT (PK)	Унікальний номер атрибута	id
2	Назва атрибута	VARCHAR	Назва поля (напр., "Матеріал")	name
3	Тип даних	VARCHAR	Формат (string, integer, date тощо)	data_type
4	Значення за замовчуванням	VARCHAR	Початкове значення	default_value
5	Обов'язковість	BOOLEAN	Чи є поле критичним для заповнення	is_required
6	Належність до типу	INT (FK)	До якої категорії об'єктів належить поле	object_type_id

Attribute_value — таблиця значень атрибутів для конкретних об'єктів. Вона реалізує зберігання динамічно заданих властивостей у форматі “об'єкт – атрибут – значення”. Детальний опис наведено у таблиці 3.7.

Опис таблиці «Attribute value» (Фактичні значення атрибутів)

№	Назва	Тип	Опис	Поле в БД
1	Ідентифікатор	INT (PK)	Унікальний номер запису значення	id
2	Об'єкт	INT (FK)	Посилання на конкретний CityObject	object_id
3	Атрибут	INT (FK)	Посилання на опис з AttributeDefinition	attribute_id
4	Значення	TEXT	Фактичний вміст характеристики	value

3.4 Декомпозиція проєкту по продуктам

Структура розподілу продукту — це ефективний інструмент, який детально описує фізичні компоненти конкретного продукту або системи, що розглядається [12]. Формальний PBS має форму ієрархії. Він починається з кінцевого продукту на вершині ієрархії, за яким йдуть підкатегорійні елементи продукту. Структура розбивки продукту подібна до структури розбивки роботи (WBS).

Подібно до WBS, структура розподілу продукту служить для зменшення складного проєкту або продукту на керовані компоненти. У результаті команди можуть отримати чітке розуміння продукту, його компонентів і того, що потрібно для забезпечення цих компонентів. На етапі планування в ньому перераховуються продукти, які будуть доставлені протягом життєвого циклу проєкту, все, що потрібно для успішного завершення проєкту [8].

На рисунку 3.4 показана PBS проєкту по продуктам.

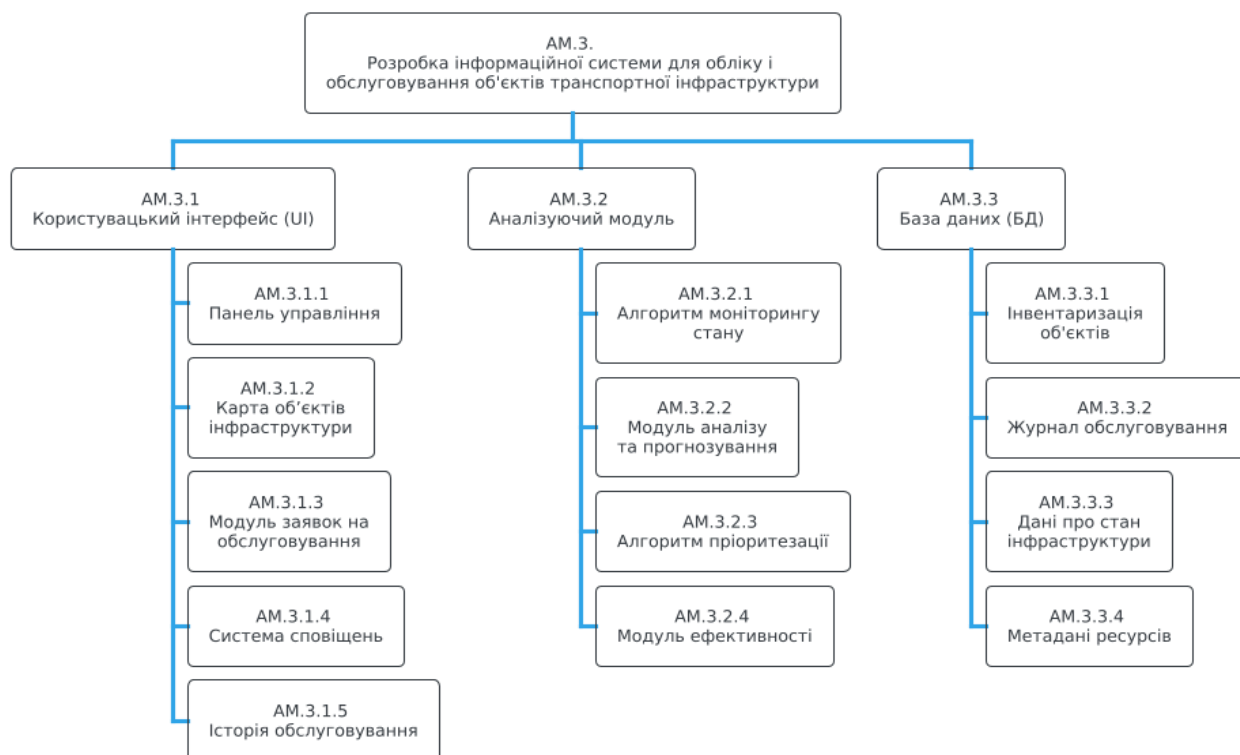


Рис. 3.4 PBS проекту по продуктам

Користувацький інтерфейс (UI). Це інтерфейс, з яким взаємодіють користувачі: адміністратори, інспектори, планувальники, підрядники тощо. Його функції: перегляд об'єктів на карті, введення даних, формування звітів, сценарне моделювання, візуалізація аналітики.

Аналізуючий модуль (Backend / API). Серверна логіка обробляє всі запити, забезпечує зв'язок з БД, здійснює бізнес-логіку (розрахунки, планування, сценарії «що, якщо»), а також взаємодіє з зовнішніми сервісами (камери, GPS, Prozorro API тощо).

База даних. Система використовує різні типи баз даних для зберігання структурованих, геопросторових, часових і аналітичних даних.

3.5 Опис модулів ПЗ та їх взаємодії.

Продуктом буде вважатись інформаційна система управління міською транспортною інфраструктурою.

Опишемо далі основні модулі.

Метамодельний каталог об'єктів: гнучка структура даних, що дозволяє описувати будь-які компоненти інфраструктури (дороги, мости,

світлофори, зупинки, трамвайні колії, рухомий склад тощо) з властивостями, зв'язками і довільними атрибутами. Це реалізується через «мета-модель» – схему опису, яку адміністратор системи може змінювати без переробки коду (наприклад, додаючи новий тип об'єкта чи властивості). Каталог інтегрується з ГІС: кожен об'єкт має геоприв'язку (координати, кластери). Така система дозволяє уніфікувати всі дані про транспортну інфраструктуру в БД.

Модуль інвентаризації: дає змогу інспекторам і технічному персоналу фіксувати/оновлювати стан об'єктів. Форма введення даних може працювати онлайн/оффлайн на мобільних пристроях. Набір функцій: створення технічних паспортів (відмітки про ремонти, інспекції, вікові графіки), система повідомлень про необхідність планових оглядів. Для прогнозування зносу може використовуватися модель на основі історичних даних (машинне навчання або аналітика): наприклад, аналіз темпів руйнування покриття чи корозії мостів. Електронна БД «транспортних об'єктів» з їхніми технічними характеристиками й історією експлуатації є фундаментом для наступних розрахунків.

Модуль транспортного планування: містить інструменти для моделювання пасажиропотоку та транспортних потоків по місту. Використовує транспортні моделі (макроскопічні/мікроскопічні), бере дані з реального часу (камери, сенсори, GPS громадського транспорту) і аналізує показники (індекс мобільності, рівень заторів, пасажиропотік на маршрутах). Враховує технічний стан мережі (наприклад, звуження руху через ремонт). Система здатна аналізувати сценарії («що, якщо»): наприклад, додавання нового маршруту чи зміна розкладу. Як результат генерує візуалізації (карти заторів, діаграми пасажиропотоку). Автоматизація маршрутизації дозволяє оптимізувати рух: наприклад, перенаправляти трафік на менш завантажені дороги у реальному часі.

Модуль планування технічних робіт: інтегрується з календарем і бюджетом. Дає змогу прокласти графік ремонту окремих об'єктів з урахуванням їх пріоритетності і ресурсів. Містить інтерфейс для підрядників –

вони отримують заявки, можуть подавати тендерні пропозиції (інтеграція з Prozorro API) і звітувати про виконання. Система також формує необхідні документи для тендерів (зведення по обсягам робіт, кошториси). Автоматично оцінюється вплив запланованих ремонтів на трафік (наприклад, що станеться при перекритті певної вулиці) через симуляцію руху.

Модуль порівняння альтернатив: призначений для аналізу «сценаріїв змін» у транспортній мережі. Користувач може задати кілька варіантів розвитку: наприклад, будівництво нової дороги проти розвитку громадського транспорту. Система візуалізує різницю – наприклад, графік зміни заторів чи викидів CO₂ в кожному сценарії. Тут корисні аналітика та ГІС-демонстрація: наприклад, накладення зон з обмеженням руху на карту. Метою є допомогти ухвалювати рішення на підставі порівняння кількісних показників.

Модуль розподілу бюджету: автоматизує фінансове планування. Користувач вводить сукупний бюджет на транспорт і пріоритети (ремонт доріг, закупівля електротранспорту, розвиток інтелектуальних систем тощо). Система пропонує кілька варіантів розподілу коштів з оцінкою їх впливу – наприклад, «як зміниться затримка руху, якщо виділити 30% бюджету на закупівлю нових трамваїв проти 20%». Це реалізується через аналітичні моделі, що зв'язують інвестиції з показниками ефективності (можна залучити дані з програми «Міський громадський транспорт»). В результаті отримуємо звітні діаграми чи таблиці порівняння кількох бюджетних сценаріїв.

3.6 Програмна розробка

3.6.1 Архітектурні рішення та структурна побудова інформаційної системи

Для забезпечення високої масштабованості, відмовостійкості та гнучкості при обробці великих масивів інфраструктурних даних, у проєкті обрано багаторівневу клієнт-серверну архітектуру з використанням мікросервісного підходу [28, 30]. Дане рішення дозволяє ізолювати окремі

функціональні блоки системи, забезпечуючи їх незалежну розробку, розгортання та масштабування.

Опис рівнів архітектури

Рівень представлення (Frontend). Реалізується за принципом SPA (Single Page Application) з використанням сучасних JavaScript-фреймворків (наприклад, React або Vue.js). Це забезпечує високу швидкість відгуку інтерфейсу та безперервну взаємодію з користувачем без перезавантаження сторінок. Для мобільних користувачів передбачено кросплатформний мобільний застосунок, що взаємодіє з тими ж API-сервісами.

Рівень прикладних сервісів (Backend & API). В основі лежить REST API, який виступає єдиним інтерфейсом обміну даними між клієнтською та серверною частинами. Архітектура передбачає поділ на автономні модулі (сервіси), що взаємодіють між собою:

- Сервіс каталогу об'єктів: відповідає за управління базовими сутностями інфраструктури.
- Сервіс інвентаризації: забезпечує збір та оновлення даних про технічний стан.
- Модуль транспортного моделювання: містить математичні алгоритми аналізу трафіку та прогнозного моделювання.
- Сервіс планування та аналізу альтернатив: відповідає за вибір оптимальних сценаріїв обслуговування.
- Модуль управління бюджетом: інтегрується з фінансовими моделями для розподілу ресурсів.

Рівень даних (Data Storage & GIS). Враховуючи специфіку проекту, ядром системи є реляційна СУБД PostgreSQL із розширенням PostGIS. Це дозволяє ефективно зберігати та обробляти геопросторові дані (точки, лінії, полігони) безпосередньо на рівні бази даних, виконуючи складні просторові запити. Для збереження неструктурованих даних (наприклад, технічної документації або фотофіксації об'єктів) можливе використання NoSQL-сховищ (MongoDB).

Архітектуру можна представити у вигляді наступних логічних блоків –
рис.3.5:

- Client Side: Web Browser (SPA), Mobile App.
- API Gateway: Маршрутизація запитів, автентифікація та авторизація користувачів.
- Microservices Layer: Набір незалежних Docker-контейнерів, кожен з яких реалізує окрему бізнес-логіку (Облік, Моделювання, Бюджет).
- Persistence Layer: ГІС-сервер, Центральна БД (PostgreSQL), Кеш-сервер (Redis) для швидкого доступу до картографічних даних.

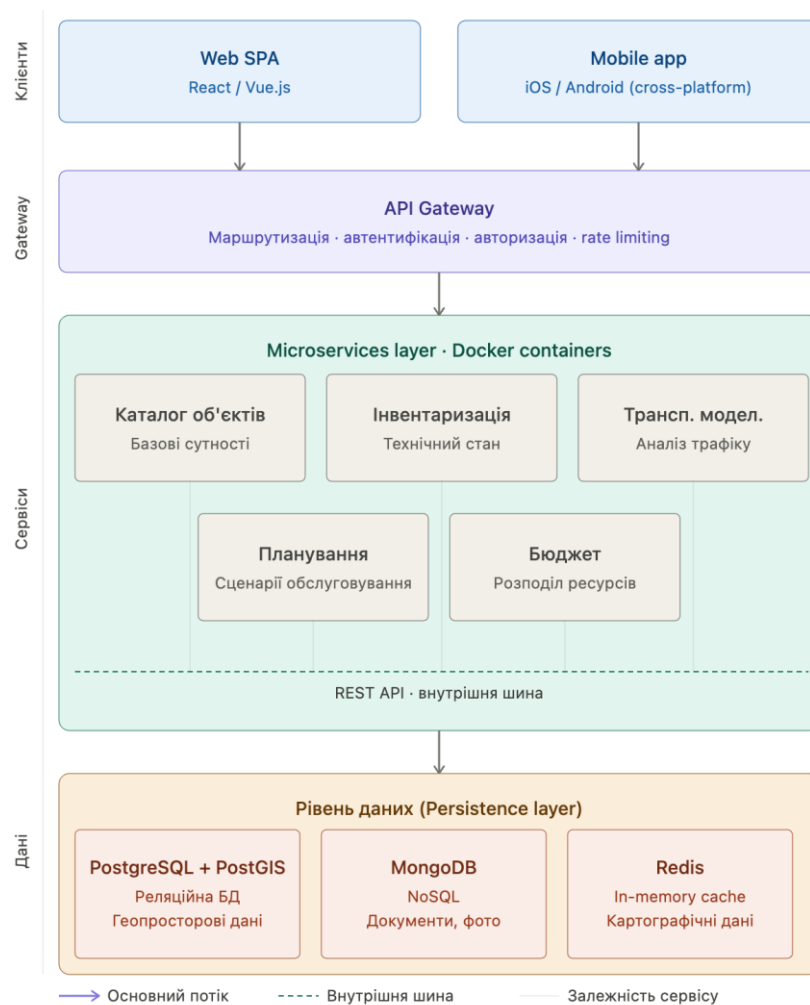


Рис. 3.5 Схематичне представлення архітектурного рішення

3.6.2 Алгоритм предиктивного обслуговування та оптимізації бюджету інфраструктури

Ефективне управління міською інфраструктурою вимагає переходу від моделі реагування на аварії до моделі предиктивного (прогнозного) обслуговування. Це дозволяє не лише забезпечити безпеку експлуатації об'єктів, а й раціонально використовувати обмежені бюджетні ресурси міста.

Метод визначення індексу технічного стану об'єкта. В основі алгоритму лежить обчислення інтегрального індексу технічного стану, який базується на динамічних атрибутах, що зберігаються в базі даних за схемою EAV. Кожному дефекту або параметру зносу присвоюється ваговий коефіцієнт, що відображає його вплив на загальну стійкість системи [15, 29].

Логіка процесу:

1. Система зчитує актуальні значення з таблиці AttributeValue.
2. Вираховується відхилення кожного параметра від еталонного стану.
3. На основі критичності об'єкта та його зносу формується черга на обслуговування.

Програмна реалізація (фрагмент Java):

```
public class MaintenancePredictor {
    private static final double CRITICAL_LEVEL = 40.0;

    public double calculateStatus(InfrastructureObject obj) {
        double currentHealth = 100.0;
        // Розрахунок зносу на основі динамічних атрибутів
        for (Double defectValue : obj.getDynamicAttributes().values()) {
            currentHealth -= (defectValue * 0.5); // 0.5 - базовий
коєфіцієнт ваги
        }
        return Math.max(0, currentHealth);
    }

    public boolean checkUrgency(double health) {
        return health < CRITICAL_LEVEL;
    }
}
```

Алгоритм оптимального розподілу бюджетних коштів. Коли сформовано список об'єктів, що потребують ремонту, постає задача вибору підмножини проєктів, які принесуть максимальну користь місту в межах заданого річного бюджету. Ця задача формалізується як класична «задача про

рюкзак» (Knapsack Problem), де вага — це вартість ремонту, а цінність — сукупний соціально-економічний ефект.

Реалізація методом динамічного програмування: для великої кількості об'єктів найбільш ефективним є метод динамічного програмування, що дозволяє знайти глобальний оптимум.

Програмна реалізація (фрагмент Java):

```
public List<InfrastructureObject> optimizeBudget (List <InfrastructureObject>
items, int budget) {
    int n = items.size();
    double[][] dp = new double[n + 1][budget + 1];
    for (int i = 1; i <= n; i++) {
        InfrastructureObject item = items.get(i - 1);
        int cost = (int) item.getEstimatedCost();
        double profit = (100 - calculateStatus(item)) *
item.getImportanceWeight();

        for (int j = 0; j <= budget; j++) {
            if (cost <= j) {
                dp[i][j] = Math.max(dp[i-1][j], dp[i-1][j-cost] + profit);
            } else {
                dp[i][j] = dp[i-1][j];
            }
        }
    }
    return extractOptimalList(dp, items, budget); // Метод для відновлення
списку обраних об'єктів
}
```

Запропонований алгоритм дозволяє автоматизувати процес прийняття рішень на рівні департаментів КМДА. Замість суб'єктивного вибору об'єктів для ремонту, система пропонує математично обґрунтований план.

3.6.3 Проєктування інтерфейсу користувача (UI/UX)

Інтерфейс розроблюваної системи спроектовано як багатофункціональне робоче середовище, що базується на принципах модульності та контекстного управління. Робочий простір розділений на функціональні зони, що дозволяє користувачу перемикатися між операційним обліком та стратегічним плануванням без втрати даних.

Навігаційна панель (Sidebar): Розміщена ліворуч і забезпечує миттєвий доступ до чотирьох основних розділів: «Головна» (ГІС-карта), «Каталог об'єктів» (EAV-реєстр), «Аналітика» (Предиктивний аналіз) та «Бюджет» (Оптимізація витрат).

Модуль ГІС-моніторингу (рисунк 3.6): Представляє собою інтерактивну карту Києва з геопросторовим відображенням активів. Об'єкти позначені колірними маркерами, що відповідають їхньому поточному статусу (діє/потребує ремонту). Праворуч розміщена панель швидкого перегляду характеристик обраного об'єкта.

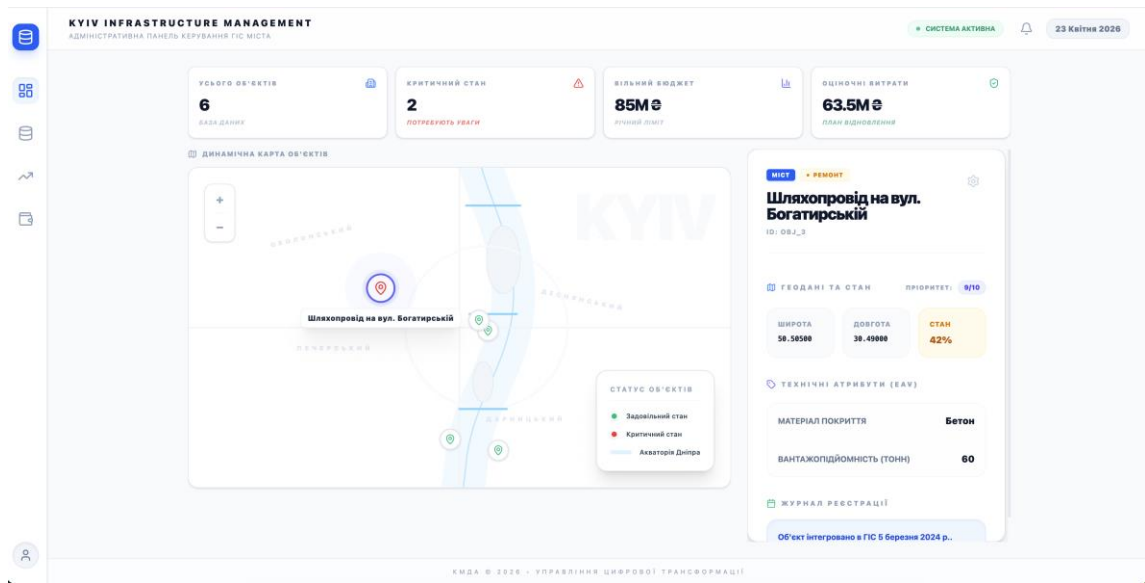


Рис. 3.6 Модуль ГІС-моніторингу

Каталог інфраструктури та управління метаданими (рисунк 3.7): Реалізує логіку EAV-моделі. Екран відображає табличний реєстр об'єктів із можливістю фільтрації за типом та районом. Форма редагування об'єкта динамічно адаптує набір вхідних полів залежно від обраного типу інфраструктури (наприклад, специфічні поля для мостів або зупинок).

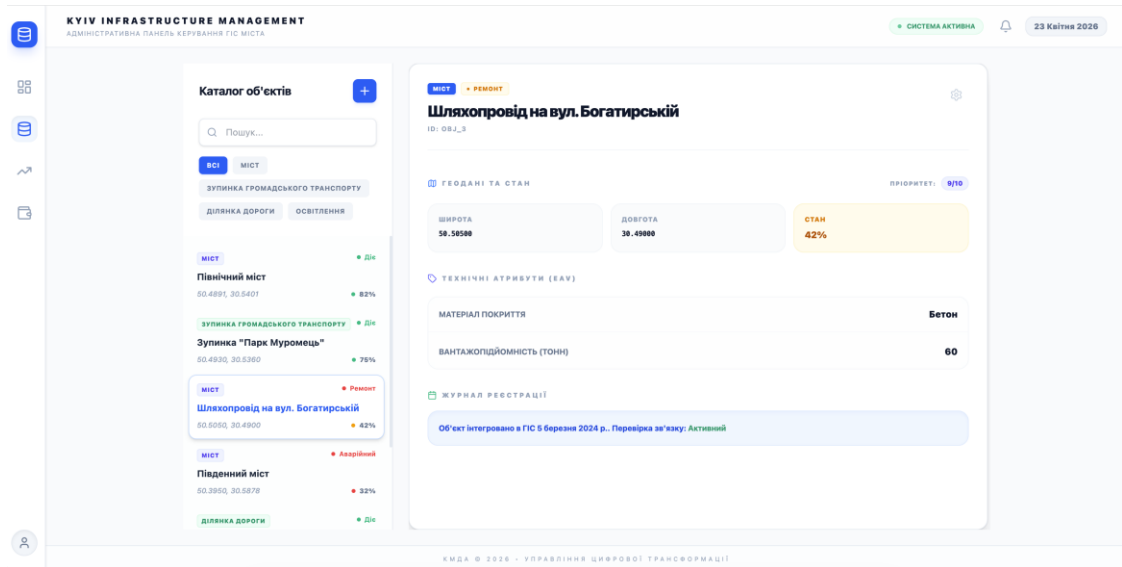


Рис. 3.7 Каталог інфраструктури та управління метаданими

Панель предиктивної аналітики (рисунок 3.8): Візуалізує результати роботи алгоритму оцінки технічного стану. Кожен об'єкт супроводжується «Індексом стану» та прогностичним графіком зносу, що дозволяє визначити оптимальний час для наступного обслуговування.

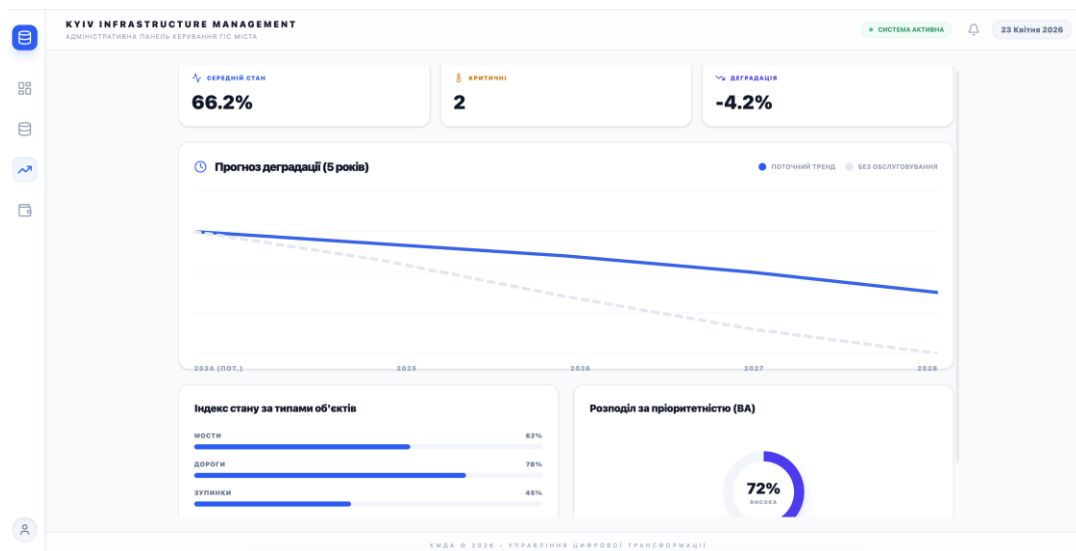


Рис. 3.8 Панель предиктивної аналітики

Модуль стратегічного планування бюджету (рисунок 3.9): Дозволяє моделювати сценарії фінансування. Користувач вводить доступний обсяг бюджету, після чого система, використовуючи метод динамічного програмування, автоматично формує список пріоритетних об'єктів для ремонту, що забезпечує максимальний соціально-економічний ефект.

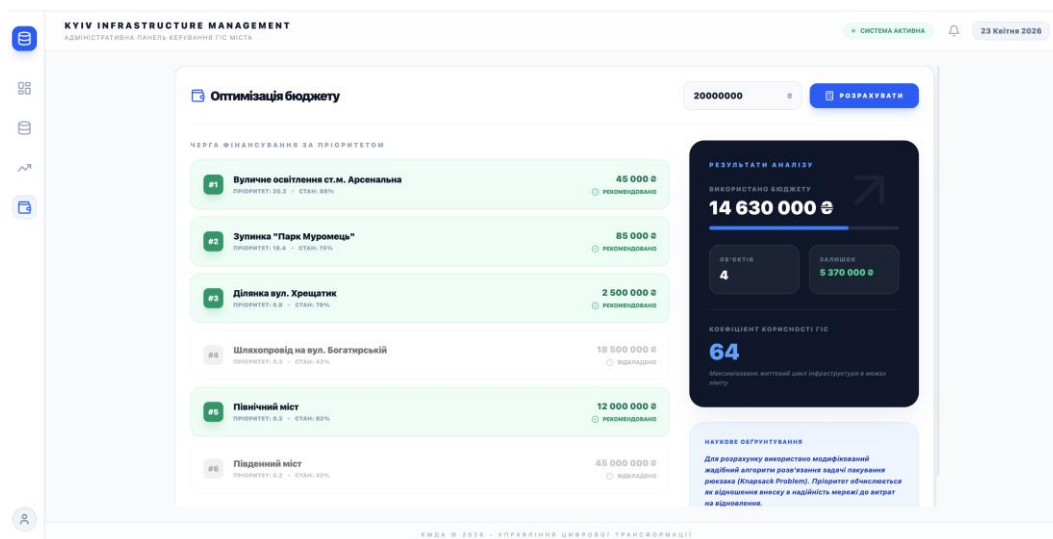


Рис. 3.9 Модуль стратегічного планування бюджету

Ефективність управління міською інфраструктурою безпосередньо залежить від швидкості доступу до даних та інтуїтивності інструментів їх

аналізу. При проєктуванні інтерфейсу розроблюваної системи основну увагу було приділено принципам ергономічності, мінімізації когнітивного навантаження на оператора та забезпеченню швидкого переходу між ГІС-даними та технічними атрибутами об'єктів.

РОЗДІЛ 4. ЗАСТОСУВАННЯ ТЕХНОЛОГІЙ УПРАВЛІННЯ ІТ ПРОЄКТОМ

4.1 Розробка організаційної структури компанії та проєкту

Ефективність розробки складних муніципальних систем, таких як ІС управління інфраструктурою міста, значною мірою залежить від обраної моделі управління та розподілу повноважень. Для реалізації даного проєкту було обрано **збалансовану матричну організаційну структуру**.

Матрична структура являє собою поєднання функціонального управління (за спеціалізацією) та проєктного управління. У збалансованому типі матриці повноваження між функціональними керівниками та менеджерами проєктів розподілені рівномірно.

Переваги обраної структури:

- Оптимізація ресурсів: Можливість залучати вузькопрофільних спеціалістів (наприклад, архітекторів ГІС або експертів з безпеки) на конкретні етапи розробки.
- Ефективна комунікація: Горизонтальні зв'язки між відділами дозволяють швидше вирішувати технічні проблеми.
- Професійний розвиток: Співробітники зберігають приналежність до своїх функціональних підрозділів (наприклад, відділу тестування), що сприяє обміну досвідом.

Ризики та їх нівелювання: Основний недолік — подвійне підпорядкування (функціональному керівнику та РМ) — вирішується чітким розмежуванням зон відповідальності та використанням інструментів Jira для автоматизованого обліку робочого часу [26, 37].

Згідно з рис. 2.1, компанія-виконавець складається з чотирьох ключових департаментів:

- Інженерний відділ: Об'єднує розробників (Front-end, Back-end, DevOps), відповідальних за технічну реалізацію та CI/CD процеси.

- Відділ аналізу та дизайну: Складається з бізнес-аналітиків та UI/UX дизайнерів, які працюють над EAV-моделлю та зручністю інтерфейсу.
- Відділ тестування (QA): Забезпечує контроль якості через ручне та автоматизоване тестування.
- Адміністративний відділ: Надає юридичну, фінансову та операційну підтримку.

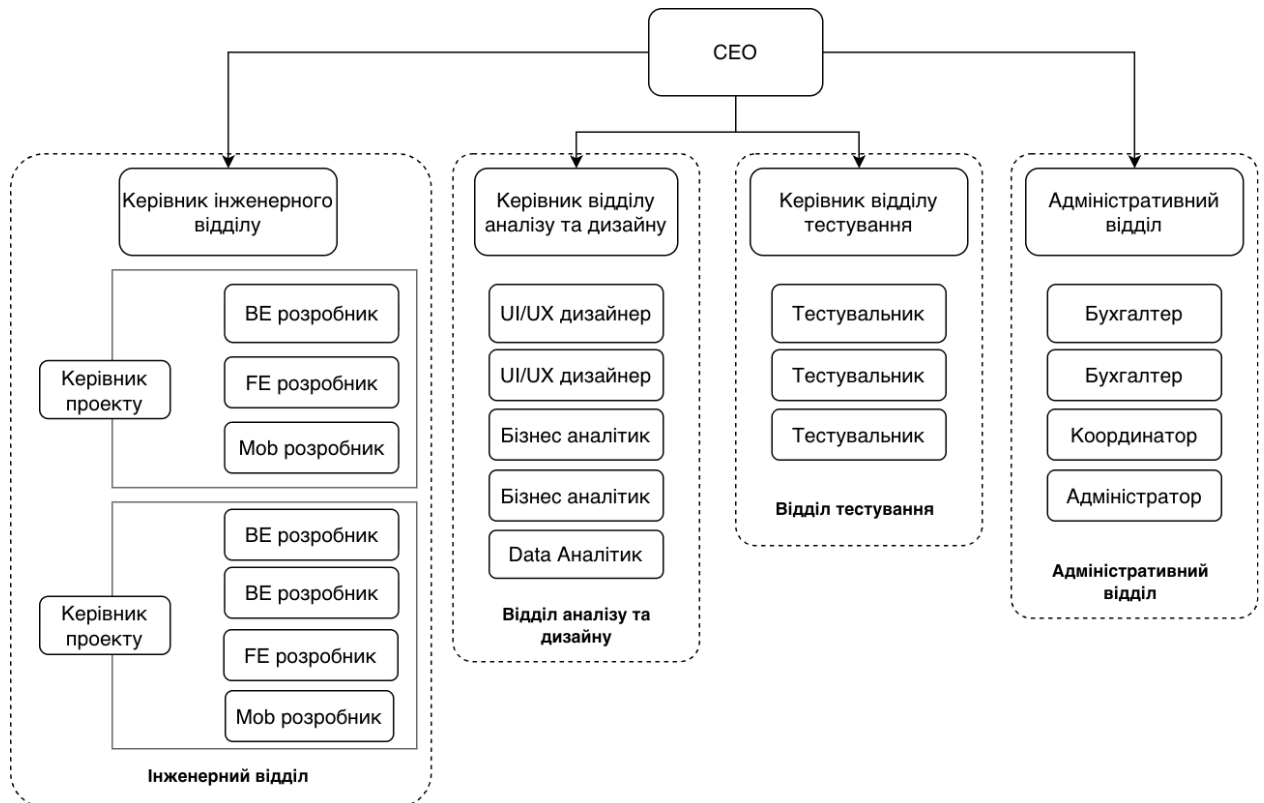


Рис. 4.1 Схема організаційної структури компанії-виконавця

Організаційна структура команди проєкту.

Організаційна структура команди проєкту, представлена на рис. 4.2, побудована за функціонально-ієрархічним принципом, що забезпечує чіткий розподіл відповідальності та ефективну комунікацію між спеціалістами. Загальна чисельність команди складає 15 осіб.

Всі підрозділи проєкту підпорядковуються безпосередньо Керівнику проєкту (Project Manager), який є центральною ланкою управління. Команда

розділена на кілька ключових напрямів (підкоманд), кожен з яких має чітко визначені завдання та зони відповідальності.

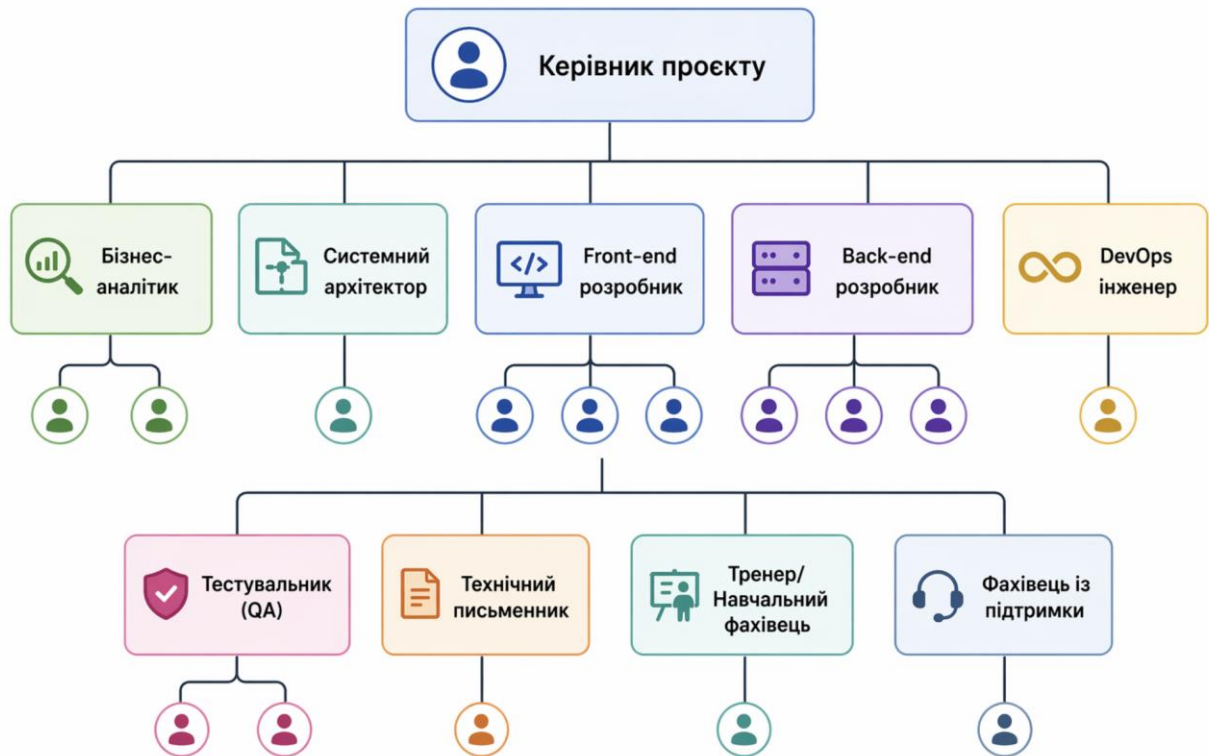


Рис. 4.2 Схема організаційної структури команди

Управлінський та аналітичний блок:

- Керівник проекту (PM): Відповідає за стратегічне планування, управління бюджетом та ризиками, координацію роботи всіх підкоманд та взаємодію із замовником (КМДА).
- Бізнес-аналітики (BA): Підкоманда у складі 2 фахівців. Вони відповідають за збір та аналіз вимог, формалізацію EAV-моделі та написання технічних специфікацій.
- Системний архітектор: 1 фахівець, який проєктує високорівневу архітектуру системи, схему бази даних та визначає технологічний стек.

Блок розробки (Development):

- Front-end розробники: Підкоманда з 3 спеціалістів. Вони відповідають за реалізацію клієнтської частини системи, ГІС-панелі та інтерактивних інтерфейсів (React).

- Back-end розробники: Найчисельніша підкоманда у складі 4 спеціалістів. Вони реалізують серверну логіку (Java/Python), API, алгоритми предиктивного аналізу та взаємодію з базою даних PostgreSQL.

Блок забезпечення якості та розгортання:

- DevOps інженер: 1 фахівець, який налаштовує CI/CD процеси, керує хмарною інфраструктурою (AWS/Azure) та відповідає за безпеку.
- Тестувальники (QA): Підкоманда з 2 спеціалістів, які проводять ручне та автоматизоване тестування для верифікації функціональності системи.

Блок впровадження та підтримки. Цей блок є критично важливим для успішної передачі продукту замовнику.

- Технічний письменник: 1 фахівець, який готує технічну документацію та інструкції користувача.
- Тренер / Навчальний фахівець: 1 фахівець, який відповідає за проведення тренінгів для співробітників КМДА та кінцевих користувачів.
- Фахівець із підтримки: 1 фахівець, який забезпечує технічну підтримку та обробку запитів користувачів на етапі впровадження та експлуатації.

Матриця взаємодії в команді

Обрана структура забезпечує баланс між спеціалізацією (вертикальне підпорядкування) та потребою в крос-функціональній взаємодії (горизонтальні зв'язки). Наприклад:

- Аналітики тісно співпрацюють із Back-end розробниками на етапі проєктування бази даних.
- Back-end та Front-end розробники взаємодіють через чітко визначені API-інтерфейси.
- DevOps забезпечує готовність інфраструктури для роботи тестувальників та розгортання фінальних версій продукту.

Така організація команди дозволяє ефективно реалізовувати складні математичні моделі та ГІС-функціонал, зберігаючи при цьому прозорість процесів управління та можливість швидкої адаптації до змін у вимогах.

4.2 Життєвий цикл проєкту

Сформуємо цілі та задачі проєкту в межах основних етапів життєвого циклу ІТ проєкту – таблиця 4.1 [6]

Таблиця 4.1

Опис життєвого циклу проєкту

Назва етапу ЖЦ	Ціль та задачі	Основні роботи
1	2	3
Визначення проєкту	Ціль: Чітко визначити обсяг, мету і вимоги до проєкту. Задачі: - Визначити ключові потреби замовника. - Сформулювати основні функціональні вимоги. - Підготувати технічну документацію	- Проведення зустрічей із замовником (КМДА) для уточнення вимог. - Створення концепції системи. - Аналіз потреб користувачів і стейкхолдерів. - Узгодження бюджету та основних параметрів проєкту (12.4 млн грн, 24 місяці). - Оцінка початкових ризиків та ресурсів.
Організація проєкту	Ціль: Організувати всі процеси та ресурси для початку проєкту. Задачі: Формування структури проєктної команди	- Складання плану управління проєктом. - Призначення керівника проєкту, бізнес-аналітиків, розробників, тестувальників та інших членів команди. - Визначення внутрішніх та зовнішніх підрядників для виконання окремих завдань. - Розподіл бюджету та ресурсів.
Управління якістю	Ціль: Забезпечити відповідність продукту всім вимогам і стандартам. Задачі: - Розробка системи контролю якості. - Встановлення ключових показників якості.	- Визначення критеріїв успішності системи. - Розробка плану тестування (включно з альфа-тестуванням).
Формування команди	Ціль: Створити ефективну проєктну команду. Задачі: - Залучення потрібних фахівців. - Забезпечення злагодженої роботи команди.	- Підбір кандидатів з необхідними навичками. - Проведення тренінгів та навчання для підвищення кваліфікації членів команди. - Організація комунікацій між командою та замовником.

1	2	3
Планування проекту	Ціль: Визначити часові рамки та ресурси для виконання проекту. Спроектувати продукт Задачі: - Розробити детальний план із зазначенням ключових віх. - Визначити ресурси та бюджет для кожного етапу. - Створити модель даних - Створити Архітектуру ПЗ	- Встановлення конкретних строків для кожної фази проекту, створення календарного плану проекту. - Оцінка ризиків та непередбачуваних витрат. - Створення плану на випадок надзвичайних ситуацій. - Дослідження предметної області з експертом, формування доменної моделі даних - Декомпозиція системи на частини. - Проектування бази даних - Проектування комунікації між частинами
Управління виконанням	Ціль: Забезпечити успішне виконання всіх запланованих завдань. Задачі: - Управління змінами та корекція плану.	- Перевірка відповідності ходу робіт із затвердженим планом. - Впровадження змін на основі зворотного зв'язку від замовника. - Організація регулярних нарад для обговорення прогресу.
Тестування	Ціль: Переконавшись, що продукт відповідає вимогам та функціональності замовника. Задачі: - Розробити та реалізувати план тестування. - Виявити і виправити всі помилки перед передачею продукту.	- Проведення функціонального тестування. - Альфа-тестування системи з виявленням ключових помилок. - Збір відгуків від користувачів після тестування. - Внесення корективів у програму та тестування виправленої версії.
Впровадження	Ціль: Забезпечити безперебійне впровадження продукту в реальне середовище експлуатації. Задачі: - Підготовка продукту до впровадження у робоче середовище. - Навчання кінцевих користувачів.	- Розгортання системи в захищеному хмарному середовищі. - Навчання співробітників КМДА користуванню системою. - Фінальна перевірка роботи системи перед офіційним запуском.
Завершення проекту	Ціль: Офіційне завершення проекту та передача готового продукту. Задачі: - Завершити всі завдання та формально закрити проєкт. - Підготувати підсумкову документацію та передати результати.	- Проведення фінального тестування. - Підготовка фінального звіту про виконання проекту. - Офіційна передача системи замовнику та закриття проекту.

4.3 Ієрархічна структура робіт проекту. Матриця відповідальності

Сформуємо ієрархічну структуру робіт проекту – рис.4.3 [32].

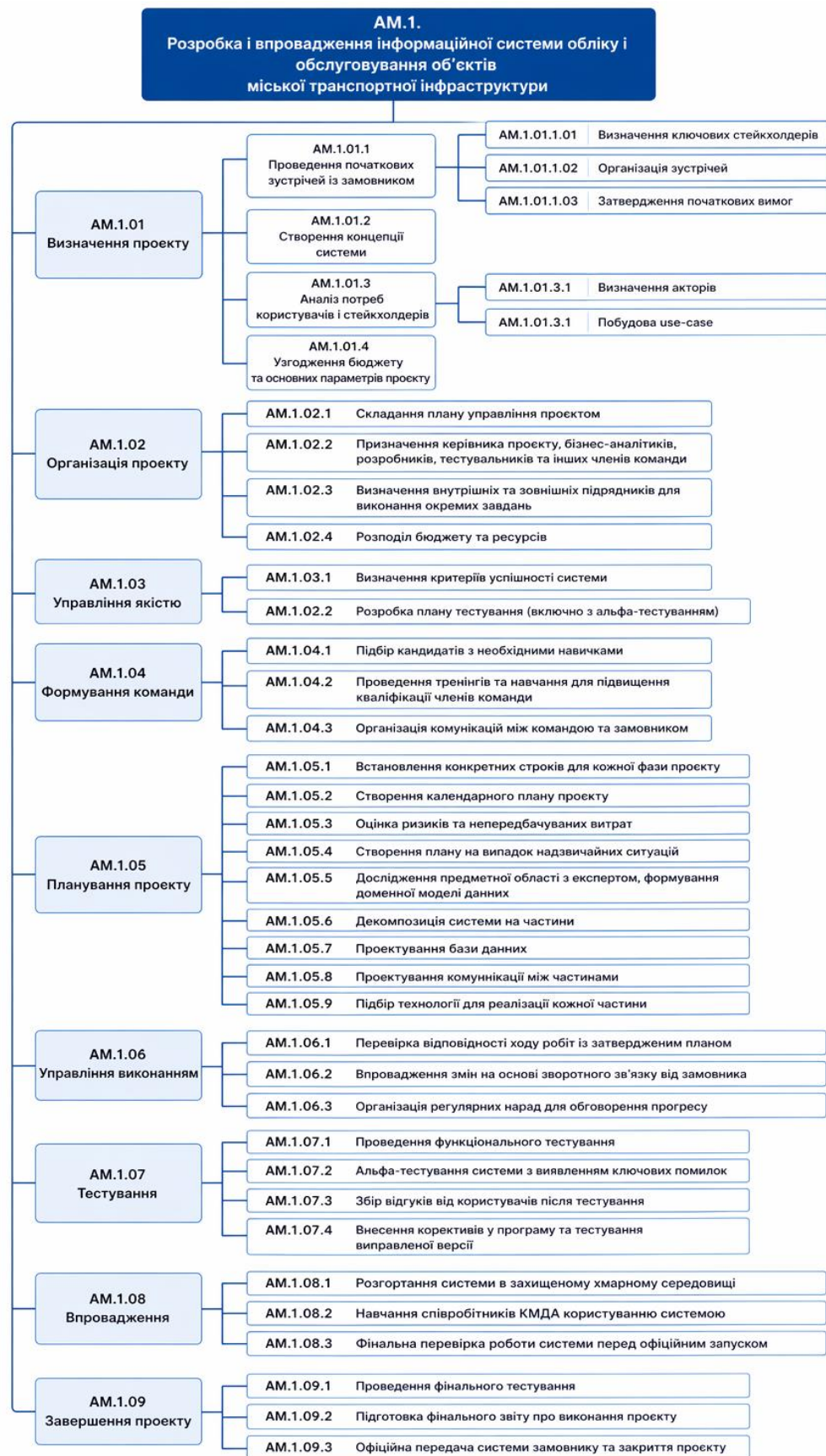


Рис. 4.3 WBS проекту по фазам життєвого циклу

Процес розпочинається з фази *визначення проєкту*, яка є фундаментом для всієї подальшої роботи, оскільки передбачає ідентифікацію ключових стейкхолдерів, проведення установчих зустрічей із представниками КМДА для фіксації вимог та формування загальної концепції системи. На цьому ж етапі проводиться аналіз потреб користувачів, будуються сценарії використання (use-cases) та затверджуються початкові параметри проєкту, включаючи бюджет у розмірі 12.4 млн грн.

Після формалізації цілей проєкт переходить до *фази організації*, де розробляється комплексний план управління, призначається керівний склад команди та визначаються внутрішні й зовнішні підрядники для специфічних робіт.

Наступним критичним етапом є *управління якістю*, що включає встановлення вимірюваних критеріїв успішності та розробку плану тестування, зокрема проведення альфа-тестування для виявлення критичних помилок на ранніх стадіях розробки.

Паралельно відбувається *формування команди*, що передбачає підбір 15 фахівців із необхідними компетенціями, організацію тренінгів для підвищення їхньої кваліфікації та налаштування каналів комунікації між виконавцями та замовником.

Фаза безпосереднього *планування проєкту* охоплює встановлення чітких часових рамок для кожного етапу розробки тривалістю 24 місяці, створення календарного графіка та глибоке дослідження предметної області спільно з експертами. У межах цього етапу розробляється доменна модель даних, проєктується архітектура бази даних на основі EAV-моделі та проводиться оцінка потенційних ризиків із формуванням резервного фонду.

Управління виконанням забезпечує безперервний контроль ходу робіт, впровадження необхідних змін на основі зворотного зв'язку від КМДА та проведення регулярних нарад для обговорення поточного прогресу.

Завершальні стадії проєкту фокусуються на верифікації та розгортанні продукту. *Фаза тестування* передбачає повну перевірку функціональності

системи, збір відгуків від перших користувачів та внесення фінальних коректив у програмне забезпечення. Під час *впровадження* система розгортається в захищеному хмарному середовищі AWS або Azure, на що виділено бюджет у розмірі \$30,000, а фахівці проводять навчання персоналу КМДА правилам експлуатації інтерфейсу. Життєвий цикл завершується фазою офіційного *закриття проєкту*, яка включає підготовку підсумкової звітності, передачу повної технічної документації та формальну передачу готового рішення замовнику.

Матриця RACI

На основі наданої ієрархічної структури робіт (WBS) для сформованої команди опишемо розподіл відповідальності для проєкту розробки ІС міської інфраструктури за допомогою матриці RACI – таблиця 4.2.

У рядках матриці наведено основні етапи та активності життєвого циклу проєкту, сформовані на основі WBS-структури, зокрема: ініціація, організація, управління якістю, планування, виконання, тестування, впровадження та завершення проєкту. У стовпцях представлено ключові ролі учасників команди: керівник проєкту, бізнес-аналітик, системний архітектор, розробники, FE/BE розробники, DevOps інженер, тестувальник та фахівець з підтримки.

У матриці використано наступні позначення:

- R (Responsible) — виконавець: безпосередньо працює над завданням.
- A (Accountable) — відповідальний: затверджує результат, відповідає за успіх етапу в цілому (тільки одна особа на завдання).
- C (Consulted) — консультант: надає експертну допомогу, двосторонній зв'язок.
- I (Informed) — поінформований: отримує повідомлення про результат, односторонній зв'язок.

Матриця відповідальності RACI

Виконавці проєкту		Етапи проєкту							
		Керівник проєкту	Бізнес аналітик	Системний архітектор	Розробники	FE/BE розробник	DevOps інженер	Тестувальник	Підтримка
1		2	3	4	5	6	7	8	9
Управління проєктом		R							
Ініціація	Початкові зустрічі	A	R	C	I	I	I	I	I
	Концепція системи	A	R	C	I	I	I	I	I
	Аналіз потреб	A	R	C	I	I	I	I	I
	Узгодження бюджету	A	C	I	I	I	I	I	I
Органі- зація	План управління	A	R	C	I	I	I	I	I
	Формування команди	A	R	I	I	I	I	I	I
	Ресурси та бюджет	A	C	I	I	I	I	I	I
Якість	Критерії якості	A	R	C	I	I	I	C	I
	План тестування	A	C	I	I	I	I	R	I
Формува- ння команди	Підбір кандидатів	A	R	C	I	I	I	I	I
	Навчання	A	C	I	I	I	I	I	I
	Комунікація	A	R	I	I	I	I	I	I
Планування	Строки	A	R	C	I	I	I	I	I
	Ризики	A	R	C	I	I	I	I	I
	Декомпозиція	A	R	C	I	I	I	I	I
	Архітектура	A	C	R	I	I	I	I	I
	База даних	A	C	R	C	R	I	I	I
	Технології	A	C	R	I	I	C	I	I
Викона- ння	Контроль виконання	A	R	I	I	I	I	I	I
	Управління змінами	A	R	C	I	I	I	I	I
	Мітинги	A	R	I	I	I	I	I	I

1		2	3	4	5	6	7	8	9
Тестування	Функціональне тестування	A	I	I	C	C	I	R	I
	Альфа-тестування	A	I	I	C	C	I	R	I
	Збір фідбеку	A	R	I	I	I	I	C	I
	Виправлення багів	A	I	I	R	R	I	C	I
Впровадження	Розгортання	A	I	C	I	I	R	I	I
	Фінальна перевірка	A	C	C	I	I	I	R	I
Завершення	Фінальне тестування	A	I	I	I	I	I	R	I
	Передача в підтримку	A	C	C	I	I	I	I	R

Керівник проєкту у більшості випадків виступає як Accountable, що відповідає його ролі у забезпеченні досягнення цілей проєкту, контролі виконання та прийнятті ключових управлінських рішень. Бізнес-аналітик здебільшого є Responsible на етапах ініціації та планування, оскільки саме він відповідає за формування вимог і взаємодію із замовником. Системний архітектор виконує роль Responsible або Consulted у питаннях технічного проєктування, вибору архітектурних рішень та технологій.

Розробники (зокрема FE/BE) залучені як Responsible на етапах реалізації функціоналу та виправлення дефектів. DevOps інженер відповідає за процеси розгортання та підтримки інфраструктури, виступаючи Responsible на етапі впровадження. Тестувальник є ключовим виконавцем (Responsible) у процесах тестування та контролю якості програмного продукту. Фахівець з підтримки залучається переважно на завершальних етапах проєкту та відповідає за подальший супровід системи після її впровадження.

Запропонована матриця RACI дозволяє підвищити прозорість процесів управління проєктом, забезпечити ефективну комунікацію між учасниками та оптимізувати розподіл відповідальності, що є критично важливим для успішної реалізації складних інформаційних систем.

4.4 Розробка календарного плану. Планування термінів проєкту

Для забезпечення чіткого контролю за термінами виконання робіт, координації дій команди та ефективного управління ресурсами було розроблено детальний календарний план проєкту. Основним інструментом планування обрано середовище Microsoft Project, що є галузевим стандартом у сфері управління проєктами.

Використання MS Project дозволило візуалізувати задачі у формі діаграми Ганта. Опис задач та залежностей наведено у таблиці 4.3 [23, 27].

Таблиця 4.3

Задачі проєкту

№	Назва задачі	Попередні роботи
1	2	3
1	Початок проєкту (віха)	
2	Аналіз потреб замовника	
3	Зустрічі зі стейкхолдерами (зацікавленими сторонами)	2
4	Визначення проблем та цілей проєкту	3
5	Визначення КРІ (ключових показників ефективності)	4
6	Підготовка звіту про ініціацію	5
7	Призначення менеджера проєкту (віха)	6FF
8	Узгодження цілей з КМДА	7
9	Ідентифікація стейкхолдерів	8
10	Підготовка документа про ініціацію (Статуту проєкту)	9
11	Визначення змісту (обсягу) проєкту	10
12	Оцінка попередніх ризиків	11
13	Проведення SWOT-аналізу	12
14	Визначення ключових потреб замовника (віха)	13FF
15	Формування візії проєкту	14SS
16	Аналіз наявних ІТ-систем	15
17	Визначення ролей у проєкті	16
18	Узгодження форматів взаємодії	17
19	Формулювання функціональних вимог (віха)	18FF
20	Аналіз користувачів та стейкхолдерів	19
21	Завершення зустрічей з КМДА (віха)	20
22	Завершення аналізу користувачів	21
23	Затвердження бюджету та параметрів (віха)	22FF
24	Визначення критеріїв успішності проєкту	23SS
25	Затвердження критичних обмежень	24
26	Початок етапу планування (віха)	25
27	Складання плану управління проєктом	25
28	Розробка технічної документації	27
29	Створення концепції системи	28
30	Розробка архітектури ПЗ	29

1	2	3
31	Розробка структури бази даних	30
32	Визначення підрядників	31
33	Формування команди проєкту	32
34	Розподіл бюджету та ресурсів	33
35	Розробка системи контролю якості	34
36	Визначення ключових показників якості	35
37	Розробка плану тестування	36
38	Відбір кандидатів	37
39	Розробка плану навчання	38
40	Організація структури комунікацій	39
41	Розробка детального графіка контрольних точок	40
42	Аналіз ризиків та надзвичайних ситуацій	41
43	Завершення технічного проєктування	42
44	Затвердження плану впровадження	43FF
45	Координація етапів та графіка робіт	44
46	Проведення внутрішнього перегляду планів	45
47	Проведення публічної презентації плану	46
48	Затвердження плану управління (віха)	47SS
49	Затвердження архітектури системи (віха)	48
50	Завершення етапу планування (віха)	49
51	Початок реалізації проєкту (віха)	50
52	Розробка модуля обліку дорожнього покриття	50
53	Розробка модуля обліку світлофорів та дорожніх знаків	50
54	Розробка модуля управління ремонтними роботами	50
55	Розробка підсистеми аналітики інцидентів	50
56	Інтеграція з геоінформаційною системою (ГІС)	50
57	Інтеграція з міською системою камер відеоспостереження	56
58	Розробка інтерфейсу адміністратора	57
59	Розробка інтерфейсу користувача (вебверсія)	50
60	Розробка інтерфейсу користувача (мобільна версія)	50
61	Налаштування автоматичного збору даних із датчиків	60
62	Розробка механізму планування технічного обслуговування	50
63	Розробка алгоритму прогнозування поломок	50
64	Розробка модуля звітності для менеджерів	58
65	Налаштування прав доступу та ролей користувачів	58
66	Впровадження push-сповіщень та нагадувань	65
67	Розробка дашборду в реальному часі	66
68	Впровадження API для зовнішніх з'єднань	67
69	Проведення інтеграційного тестування модулів	68; 52; 53; 54; 55; 59; 61; 62; 63; 64
70	Оптимізація продуктивності системи	69
71	Розробка механізмів резервного копіювання та відновлення	70
72	Функціональне тестування	71SF
73	Альфа-тестування	72
74	Збір відгуків	73

1	2	3
75	Коригування після тестування	74
76	Підготовка до розгортання	75
77	Навчання кінцевих користувачів	76
78	Розгортання у продуктивному середовищі	77
79	Тестування в «бойових» умовах	78
80	Підготовка фінальної документації	79
81	Узгодження з КМДА	80
82	Передача продукту замовнику (віха)	81
83	Початок етапу завершення (віха)	82
84	Фінальний аудит проєкту	82
85	Формування підсумкових звітів	84
86	Узгодження фінальних результатів	85
87	Закриття контрактів із підрядниками	86
88	Архівування документації	87
89	Оцінка виконання KPI	88
90	Підготовка висновків (lessons learned) щодо впровадження	89
91	Проведення фінальної зустрічі	90
92	Фінальна презентація для КМДА	91
93	Розподіл залишкових ресурсів	92
94	Закриття доступу до систем розробки	93
95	Підписання акта приймання-передачі	94
96	Формальне закриття проєкту	95
97	Завершення проєкту (віха)	96

Виставимо дату початку – 01.06.2026. Дата фінішу – 27.06.2028.

Тривалість проєкту – 543 дні.

Етап 1. Ініціація проєкту – рис .4.4.

Тривалість: червень — серпень 2026.

На цьому етапі закладається фундамент системи. Основна мета — визначити межі проєкту та отримати підтримку ключових зацікавлених сторін (КМДА).

Ключові задачі:

- Аналіз потреб замовника та визначення KPI.
- Розробка та затвердження Статуту проєкту.
- Ідентифікація стейкхолдерів та проведення SWOT-аналізу.
- Формування візії майбутньої ІТ-системи.

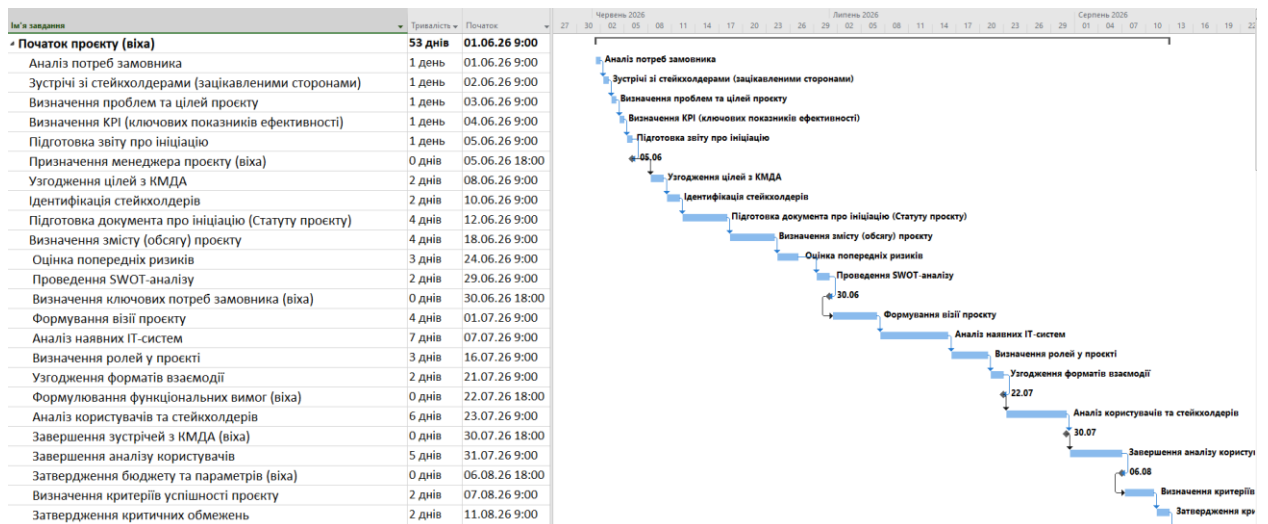


Рис. 4.4 Діаграма Ганта. Фрагмент етапу Ініціації проекту

Етап 2. Планування – рис. 4.5.

Тривалість: серпень 2026 — лютий 2027

Перехід від ідеї до технічного проектування. Тут створюється детальна

«дорожня карта» для команди розробки.

Ключові задачі:

- Розробка технічної архітектури та структури бази даних.
- Планування ресурсів, бюджету та системи управління ризиками.
- Формування команди та відбір підрядників.
- Створення плану тестування та навчання майбутніх користувачів.

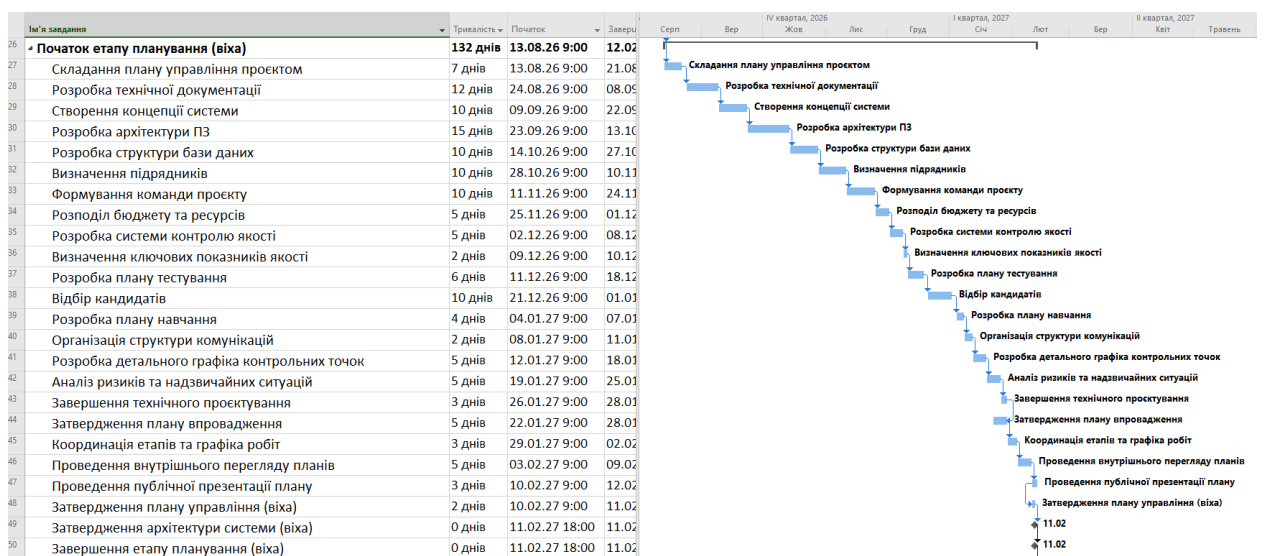


Рис. 4.5 Діаграма Ганта. Фрагмент етапу Планування проекту

Етап 3. Реалізація та впровадження (Активна фаза) – рис.4.6.

Тривалість: лютий 2027 — березень 2028.

Найбільш ресурсомісткий етап, де відбувається паралельна розробка всіх функціональних блоків системи.

Ключові задачі:

- Паралельна розробка модулів: облік дорожнього покриття, світлофорних об’єктів, ремонтних робіт та аналітики.
- Інтеграція з ГІС та системою відеоспостереження.
- Розробка мобільних та веб-інтерфейсів для користувачів.
- Тестування та розгортання: проведення альфа-тестів, навчання персоналу та запуск системи в «бойових» умовах (Production).

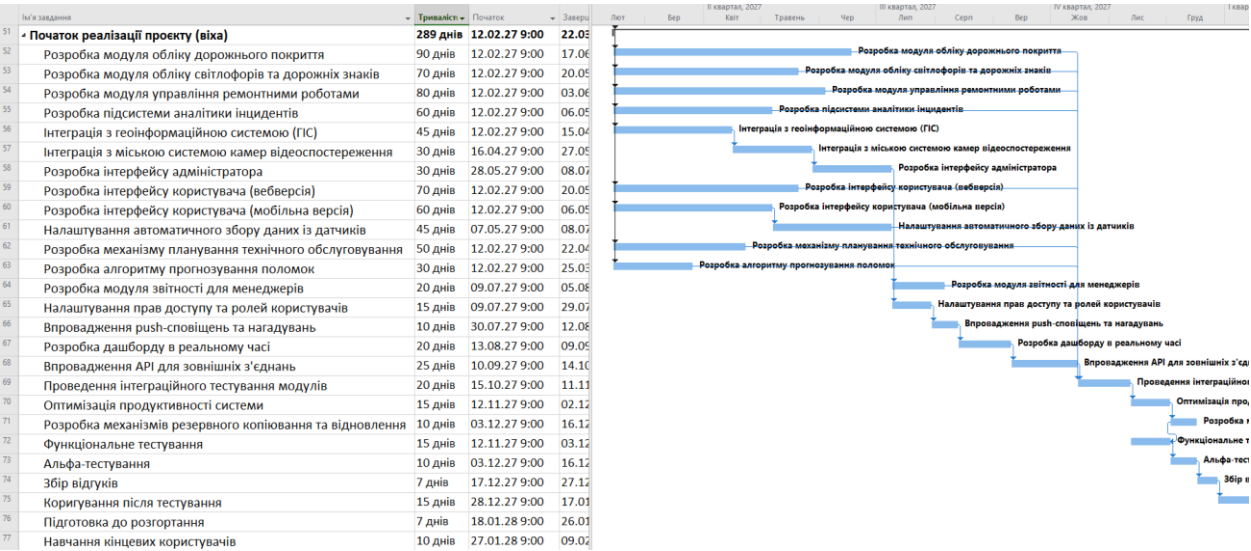


Рис. 4.6 Діаграма Ганта. Фрагмент етапу Реалізація та впровадження проекту

Етап 4. Завершення проекту – рис.4.7.

Тривалість: березень 2028 — червень 2028.

Етап формального закриття, передачі знань та оцінки досягнутих результатів.

Ключові задачі:

- Проведення фінального аудиту та перевірка відповідності KPI.
- Підготовка підсумкових звітів та документації Lessons Learned (вивчені уроки).

- Формальне підписання актів приймання-передачі з замовником.
- Архівування матеріалів та закриття контрактів.

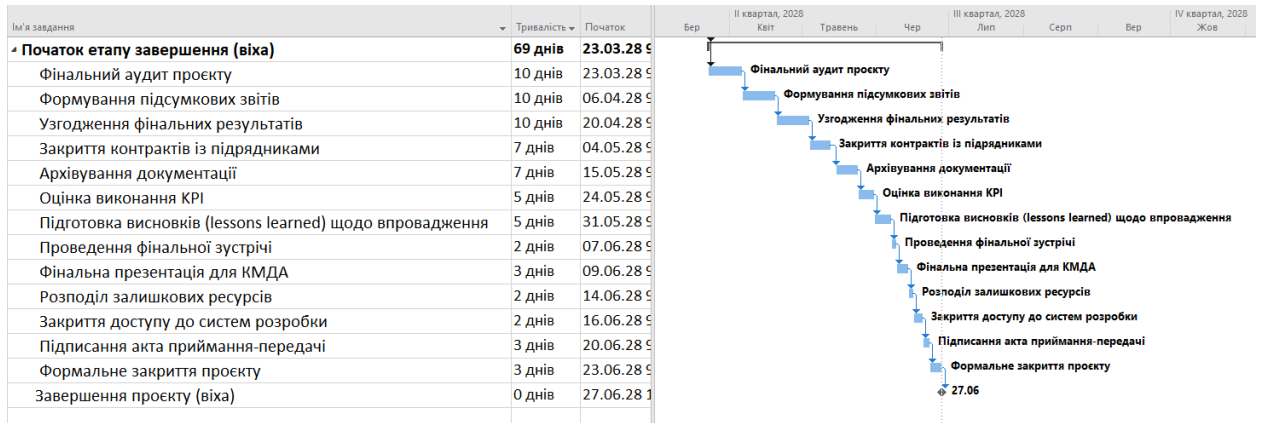


Рис. 4.7 Діаграма Ганта. Фрагмент етапу Завершення проекту

Наведемо також для візуалізації фрагмент календаря команди в активному етапі розробки – рис.4.8.

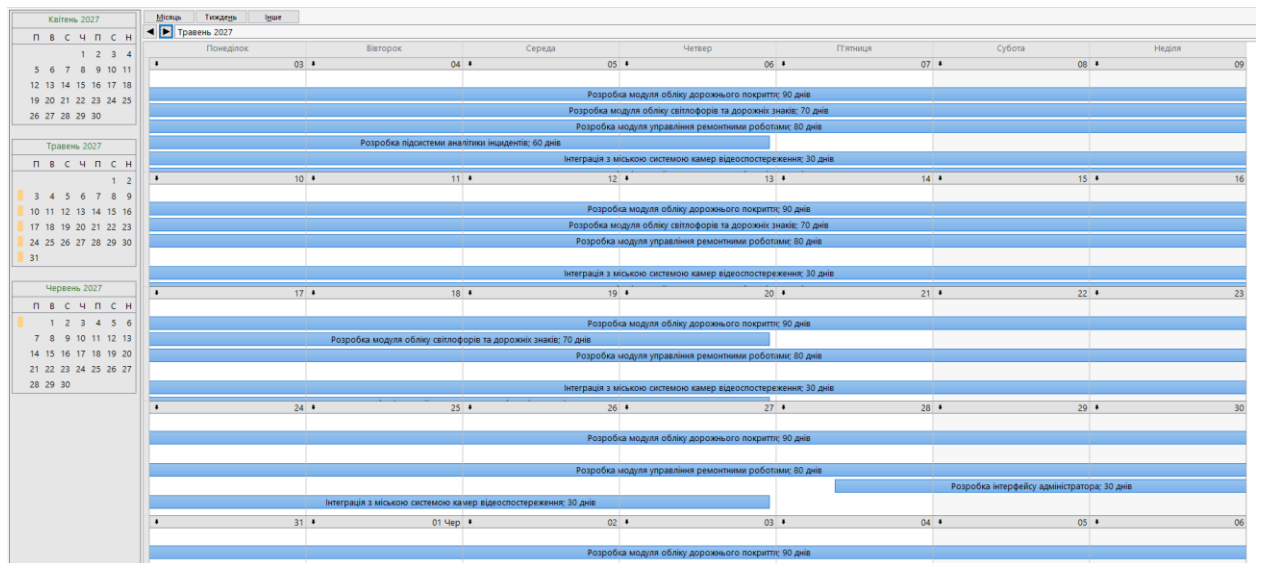


Рис. 4.8 Фрагмент календаря команди за травень 2028

4.5 Управління ризиками проекту

Процес управління ризиками в межах розробки інформаційної системи моніторингу об'єктів дорожньої інфраструктури спрямований на ідентифікацію потенційних загроз та розробку стратегій мінімізації їхнього впливу на часові межі, бюджет та якість кінцевого продукту. Ризик розглядається як ймовірнісна подія, що у разі виникнення може призвести до відхилення від запланованих показників проекту.

Системний підхід до ризиків передбачає реалізацію наступних ітераційних кроків [11, 18, 33]:

1. Виявлення (ідентифікація): складання вичерпного переліку загроз.
2. Оцінювання: визначення ймовірності та масштабів наслідків.
3. Ранжування: виділення найбільш критичних факторів.
4. Планування реагування: формування превентивних та корегувальних заходів.
5. Моніторинг: постійний нагляд за індикаторами ризику протягом усього життєвого циклу розробки.

Для ідентифікації ризиків було проведено консультації з фахівцями у сфері IT-інфраструктури та аналіз досвіду реалізації подібних муніципальних проєктів. Результати класифікації ризиків за категоріями наведено у таблиці 4.5.

Таблиця 4.5

Реєстр ідентифікованих ризиків проєкту

№	Категорія	Ризикова подія	Рівень впливу	Ступінь керованості
1	Технологічні	Несумісність модулів при інтеграції з ГІС	Високий	Високий
2	Технологічні	Критичні помилки у логіці обробки даних з камер	Високий	Середній
3	Технологічні	Недостатня продуктивність серверної частини	Середній	Високий
4	Технічні	Відмова серверного обладнання	Середній	Середній
5	Технічні	Втрата даних через відсутність резервування	Високий	Середній
6	Організаційні	Порушення термінів розробки (активна фаза)	Високий	Середній
7	Організаційні	Недостатня кваліфікація залучених розробників	Середній	Високий
8	Людський фактор	Хвороби або звільнення ключових спеціалістів	Високий	Середній
9	Зовнішні	Зміни у законодавстві щодо захисту персональних даних	Середній	Низький
10	Зовнішні	Економічна нестабільність та інфляційні процеси	Середній	Низький
11	Безпекові	Кібератаки на бази даних інфраструктури	Високий	Середній
12	Форс-мажор	Тривале порушення енергопостачання (блекаути)	Високий	Низький

Для проведення детального аналізу використовується бальна система оцінки (квазі-кількісний метод). Легенда оцінювання, що базується на комбінації якісних та кількісних показників, представлена у таблиці 4.6.

Таблиця 4.6

Шкала оцінювання ризикових подій

Якісна характеристика	Шифр	Кількісний еквівалент (бали)
Відсутній вплив	0	0
Дуже низький	ДН	1-2
Низький	Н	3
Нижче середнього	НС	4
Середній	С	5
Вище середнього	ВС	6
Високий	В	7-8
Дуже високий	ДВ	9
Критичний (катастрофічний)	К	10

На основі розробленої шкали проведено оцінювання найбільш імовірних ризиків, що можуть суттєво вплинути на реалізацію системи (таблиця 4.7). Важливість ризику визначається як добуток балів ймовірності та сумарного впливу.

Таблиця 4.7

Результати оцінювання пріоритетних ризиків

№	Ризикова подія	Часові втрати	Фінансові втрати	Ймовірність	Сумарна важливість
1	Форс-мажорні обставини (військові дії/блекаути)	9	7	6	96
2	Порушення термінів (активна фаза 18 міс.)	8	5	5	65
3	Кібератаки та витік конфіденційних даних	6	8	4	56
4	Хвороби/ротація кадрів у команді	5	4	6	54
5	Технічні збої серверної інфраструктури	7	6	3	39
6	Непередбачуване здорожчання хмарних послуг/ПЗ	3	7	4	40

Завершальним етапом управління є розробка комплексу заходів для нейтралізації загроз – таблиця 4.8. Протиризикові заходи (ПРЗ) поділяються на превентивні (профілактика), оперативні (при появі перших симптомів) та антикризові (при настанні події).

Таблиця 4.8

Результати оцінювання пріоритетних ризиків

№	Ризикова подія	Профілактика (ПРЗ-1)	Симптоми (ознаки)	Реакція на проблему (ПРЗ-3)
1	Порушення графіку розробки	Використання Agile-методології, щотижневий моніторинг Ганта	Накопичення невиконаних задач у беклозі	Перерозподіл ресурсів, залучення додаткових розробників
2	Форс-мажор (енергосистема)	Наявність альтернативних джерел живлення, хмарне деплоювання	Початок графіків відключень	Перехід на віддалену роботу з різних регіонів
3	Кадровий дефіцит	Документування коду (Wiki), уникнення концентрації знань в одній особі	Зниження швидкості розробки (Velocity)	Оперативний пошук фрілансерів або заміна з кадрового резерву
4	Безпекові ризики	Регулярні аудити безпеки, шифрування даних	Спроби несанкціонованого доступу в логах	Блокування IP-адрес, відновлення системи з бекапу

Впровадження описаної системи управління ризиками дозволяє підвищити життєздатність проєкту та гарантувати завершення активної фази розробки у встановлений 18-місячний термін.

4.6 Моніторинг відхилень при виконання проєкту

Для забезпечення контролю за ходом виконання робіт було проведено моделювання відхилень від базового плану. Моніторинг здійснювався на початкових етапах проєкту для перевірки здатності команди дотримуватися графіку та бюджету.

Сценарій моделювання відхилень:

1. Початок: проєкт стартував вчасно — 01.06.2026.

2. Дотримання плану: операції «Аналіз потреб замовника» та «Зустрічі зі стейкхолдерами» були завершені у чіткій відповідності до запланованих термінів.

3. Затримка ресурсу: через тимчасову непрацездатність провідного аналітика (лікарняний) початок задач, де він був ключовим виконавцем, змістився на 5 робочих днів.

4. Коригування термінів:

а. Задачі «Визначення проблем та цілей» та «Визначення КРІ» розпочалися із запізненням на 5 днів.

б. Задача «Аналіз наявних ІТ-систем» розпочалася на 5 днів пізніше, а її фактична тривалість збільшилася з 10 до 12 днів.

с. Задача «Формування візії проєкту» була виконана випереджальними темпами — на 3 дні швидше за план.

5. Підсумок етапу: фаза ініціації (R.1) була закрита із загальним відставанням від графіка на 17 днів.

Візуалізація планових показників (базового плану) наведена на рис. 4.9, а фактичний стан виконання робіт із урахуванням відхилень відображено на рис. 4.10.

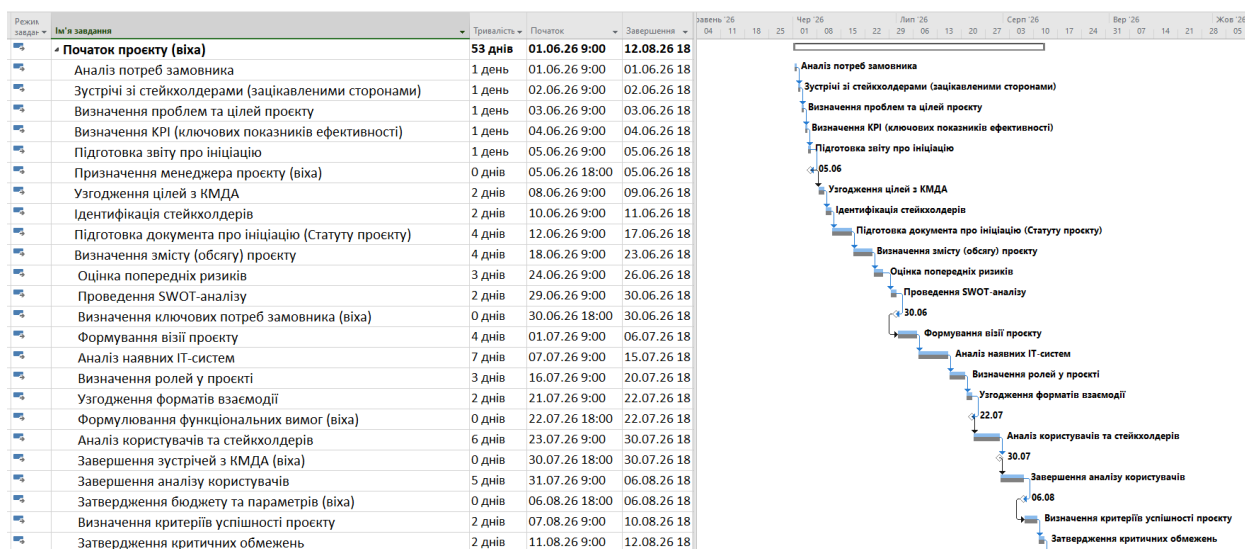


Рис. 4.9 Діаграма Ганта. Фрагмент етапу Ініціації проєкту – запланований

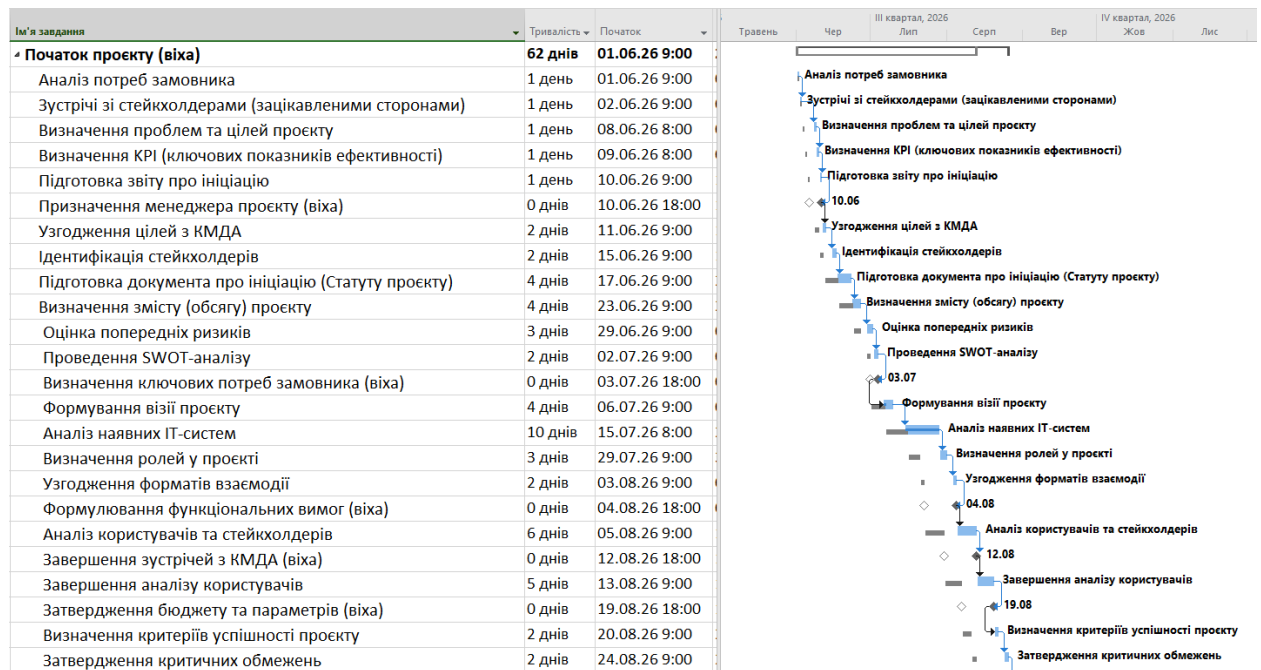


Рис. 4.10 Діаграма Ганта. Фрагмент етапу Ініціації проєкту – фактичний

За методом освоєного обсягу [21, 25] для змодельованого сценарію проведемо розрахунок – рисунок 4.11, та наведемо звіти – рис. 4.12-4.13.

- базова вартість;
- BCWS - Budgeted cost for work scheduled;
- BCWP- Budgeted cost for work performed;
- ACWP - Actual cost of work performed;
- SV - Schedule variance;
- CV - Cost variance;
- EAC - Estimate at completion;
- VAC – Variance at completion

	Ім'я завдання	Запланований обсяг – PV (BCWS)	Освоєний обсяг – EV (BCWP)	AC (ACWP)	SV	CV	EAC	BAC	VAC
1	Початок проєкту (віха)	355 275,00 ₴	238 325,00 ₴	238 325,00 ₴	-116 950,00 ₴	0,00 ₴	631 375,00 ₴	631 375,00 ₴	0,00 ₴
2	Аналіз потреб замовника	23 850,00 ₴	23 850,00 ₴	23 850,00 ₴	0,00 ₴	0,00 ₴	23 850,00 ₴	23 850,00 ₴	0,00 ₴
3	Зустрічі зі стейкхолдерами (зацікавленими сторонами)	23 850,00 ₴	23 850,00 ₴	23 850,00 ₴	0,00 ₴	0,00 ₴	23 850,00 ₴	23 850,00 ₴	0,00 ₴
4	Визначення проблем та цілей проєкту	13 600,00 ₴	13 600,00 ₴	13 600,00 ₴	0,00 ₴	0,00 ₴	13 600,00 ₴	13 600,00 ₴	0,00 ₴
5	Визначення KPI (ключових показників ефективності)	13 600,00 ₴	13 600,00 ₴	13 600,00 ₴	0,00 ₴	0,00 ₴	13 600,00 ₴	13 600,00 ₴	0,00 ₴
6	Підготовка звіту про ініціацію	6 800,00 ₴	6 800,00 ₴	6 800,00 ₴	0,00 ₴	0,00 ₴	6 800,00 ₴	6 800,00 ₴	0,00 ₴
7	Призначення менеджера проєкту (віха)	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴
8	Узгодження цілей з КМДА	34 100,00 ₴	34 100,00 ₴	34 100,00 ₴	0,00 ₴	0,00 ₴	34 100,00 ₴	34 100,00 ₴	0,00 ₴
9	Ідентифікація стейкхолдерів	13 600,00 ₴	13 600,00 ₴	13 600,00 ₴	0,00 ₴	0,00 ₴	13 600,00 ₴	13 600,00 ₴	0,00 ₴
10	Підготовка документа про ініціацію (Статуту проєкту)	27 200,00 ₴	27 200,00 ₴	27 200,00 ₴	0,00 ₴	0,00 ₴	27 200,00 ₴	27 200,00 ₴	0,00 ₴
11	Визначення змісту (обсягу) проєкту	27 200,00 ₴	27 200,00 ₴	27 200,00 ₴	0,00 ₴	0,00 ₴	27 200,00 ₴	27 200,00 ₴	0,00 ₴
12	Оцінка попередніх ризиків	54 525,00 ₴	54 525,00 ₴	54 525,00 ₴	0,00 ₴	0,00 ₴	54 525,00 ₴	54 525,00 ₴	0,00 ₴
13	Проведення SWOT-аналізу	13 600,00 ₴	0,00 ₴	0,00 ₴	-13 600,00 ₴	0,00 ₴	13 600,00 ₴	13 600,00 ₴	0,00 ₴
14	Визначення ключових потреб замовника (віха)	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴
15	Формування візії проєкту	30 650,00 ₴	0,00 ₴	0,00 ₴	-30 650,00 ₴	0,00 ₴	10 250,00 ₴	30 650,00 ₴	20 400,00 ₴
16	Аналіз наявних IT-систем	72 700,00 ₴	0,00 ₴	0,00 ₴	-72 700,00 ₴	0,00 ₴	218 100,00 ₴	181 750,00 ₴	-36 350,00 ₴
17	Визначення полей v	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴	0,00 ₴	30 750,00 ₴	30 750,00 ₴	0,00 ₴

Рис. 4.11 Розрахунок за методом освоеного обсягу

ПЕРЕВИЩЕННЯ ВИТРАТ

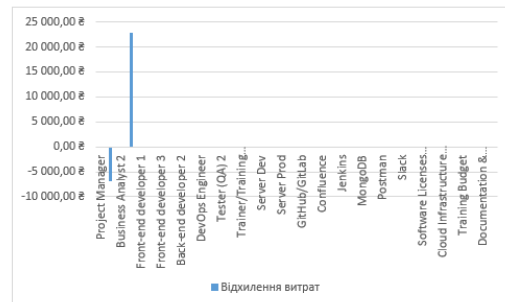
ВІДХИЛЕННЯ ВАРТІСТІ ЗАВДАНЬ

Відхилення вартості для всіх завдань верхнього рівня у проєкті.



ВІДХИЛЕННЯ ВАРТІСТІ РЕСУРСІВ

Відхилення вартості для всіх трудових ресурсів.



Ім'я	% виконання	Витрати	Базова вартість	Відхилення витрат
Початок проєкту (віха)	60%	647 325,00 ₴	631 375,00 ₴	15 950,00 ₴
Початок етапу планування (віха)	0%	1 500 875,00 ₴	1 500 875,00 ₴	0,00 ₴
Початок реалізації проєкту (віха)	0%	10 711 225,00 ₴	10 711 225,00 ₴	0,00 ₴
Початок етапу завершення (віха)	0%	696 500,00 ₴	696 500,00 ₴	0,00 ₴

Ім'я	Витрати	Базова вартість	Відхилення витрат
Project Manager	1 260 750,00 ₴	1 260 750,00 ₴	0,00 ₴
Business Analyst 1	639 200,00 ₴	646 000,00 ₴	-6 800,00 ₴
Business Analyst 2	122 400,00 ₴	122 400,00 ₴	0,00 ₴
System Architect	1 171 625,00 ₴	1 148 875,00 ₴	22 750,00 ₴
Front-end developer 1	2 329 350,00 ₴	2 329 350,00 ₴	0,00 ₴
Front-end developer 2	802 950,00 ₴	802 950,00 ₴	0,00 ₴
Front-end developer 3	802 950,00 ₴	802 950,00 ₴	0,00 ₴
Back-end developer 1	1 965 600,00 ₴	1 965 600,00 ₴	0,00 ₴
Back-end developer 2	1 319 500,00 ₴	1 319 500,00 ₴	0,00 ₴
Back-end developer 3	1 501 500,00 ₴	1 501 500,00 ₴	0,00 ₴

Рис. 4.12 Звіт по перевищенню витрат

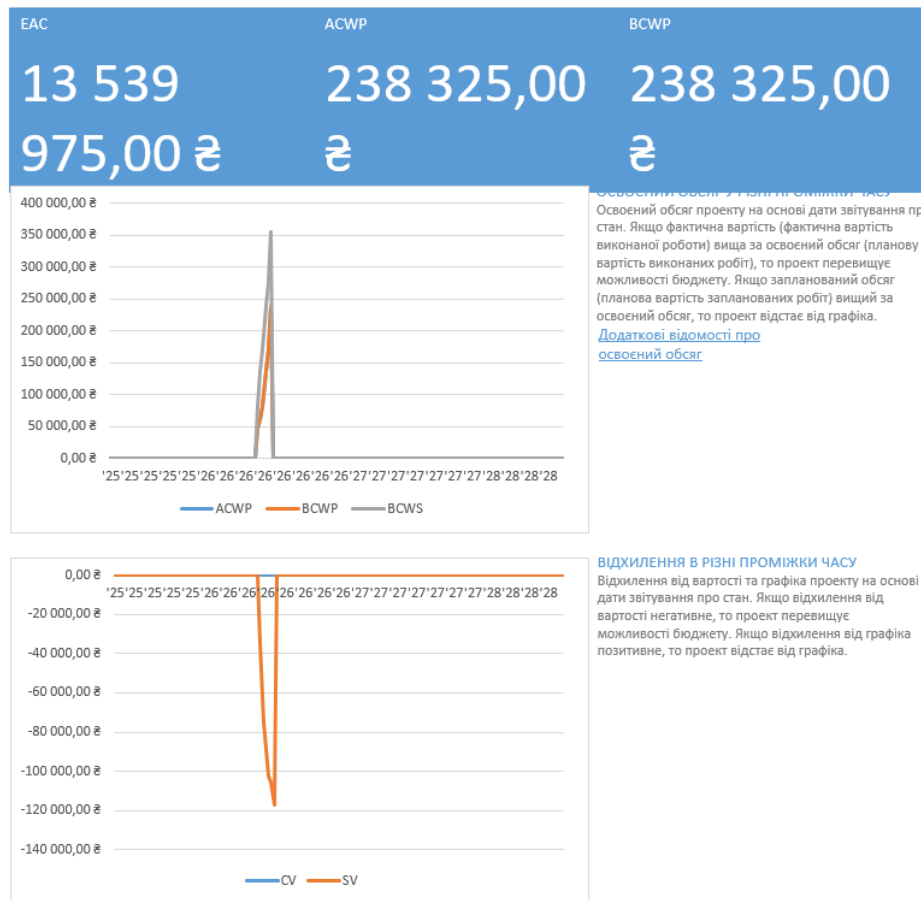


Рис. 4.13 Звіт по освоєному обсягу

Моделювання етапу ініціації виявило критичне відставання від графіка на 17 днів ($SV = -116\,950$ €), спричинене тимчасовою втратою ключового ресурсу та збільшенням фактичної тривалості окремих задач. Хоча на поточну дату витрати на виконані роботи відповідають базовому плану ($CV = 0$ €), звіт по ресурсах вже прогнозує загальну перевитрату проекту в розмірі 15 950 € через значне зростання вартості робіт системного архітектора.

Для вирівнювання ситуації на наступних фазах керівнику проекту необхідно застосувати методи стиснення розкладу (паралельне виконання задач або залучення додаткових ресурсів), щоб компенсувати втрачений час. Крім того, критично важливо переглянути стратегію управління ризиками для забезпечення взаємозамінності ключових фахівців та посилити контроль за витратами найдорожчих ресурсів.

ВИСНОВКИ

У кваліфікаційній роботі магістра вирішено актуальне науково-прикладне завдання — досліджено процеси управління проектом розробки та впровадження автоматизованої інформаційної системи обліку міської транспортної інфраструктури. Ознайомлення з результатами роботи дає підстави стверджувати, що поставлена мета — підвищити рівень керованості та надійності транспортної мережі шляхом цифровізації управління інфраструктурою — досягнута в повному обсязі.

На основі проведеного дослідження зроблено такі узагальнення, оцінки та пропозиції:

1. Доведено актуальність та економічну доцільність цифровізації обліку інфраструктури.

Аналіз поточного ІТ-середовища муніципального управління (на прикладі м. Києва) виявив, що відсутність єдиної інтегрованої системи призводить до реактивного (аварійного) обслуговування об'єктів інфраструктури, що спричиняє перевитрати бюджету та погіршення транспортної мобільності. За допомогою інструментів PEST та SWOT-аналізу визначено ключові фактори впливу, серед яких найвищий позитивний вплив має курс на впровадження технологій «Smart City», а найбільші загрози становлять воєнні ризики та обмеженість бюджетів [5, 13]. Автором сформовано паспорт проекту зі створення web-системи із бюджетом 10 млн грн та тривалістю 24 місяці, що інтегрує в собі ГІС, інструменти інвентаризації та модулі оптимізації фінансів.

2. Запропоновано математичну модель оптимізації бюджету та доведено економічну ефективність проекту.

Прикладним результатом роботи є формалізація задачі оптимального розподілу міського бюджету на транспортну інфраструктуру. Розроблену математичну модель зведено до задачі нелінійної оптимізації з лінійними обмеженнями (варіація задачі про рюкзак), де цільовою функцією виступає

мінімізація середнього часу поїздки громадян залежно від інвестицій у технічний стан об'єктів.

За допомогою професійного інструментарію Alt-Invest проведено інвестиційний аналіз. Доведено, що проєкт є економічно вигідним: чиста приведена вартість (NPV) становить 10 376 тис. грн, внутрішня норма рентабельності (IRR) досягає 54%, а дисконтований термін окупності складає лише 1,5 року. Вільний грошовий потік (Cash Flow) наприкінці 24-го місяця становить понад 50 млн грн, що свідчить про високу фінансову стійкість запропонованого гібридного (грантово-кредитного) підходу до фінансування.

3. Спроектовано адаптивне інформаційне та програмне забезпечення системи.

Важливим прикладним результатом дослідження є розробка архітектури системи на основі мікросервісного підходу. Враховуючи високу варіативність об'єктів інфраструктури, автором запропоновано та реалізовано метамодельну структуру бази даних на основі патерну EAV (Entity-Attribute-Value). Спроектовані концептуальна, логічна та фізична моделі бази даних (СУБД PostgreSQL + PostGIS) дозволяють динамічно розширювати систему без зміни програмного коду.

Крім того, розроблено алгоритм предиктивного (прогнозного) обслуговування, що обчислює інтегральний індекс технічного стану об'єкта, та спроектовано ергономічні UI/UX інтерфейси (модуль ГІС-моніторингу, каталог інфраструктури, панель аналітики), які забезпечують ефективну взаємодію стейкхолдерів із системою.

4. Застосовано сучасні технології управління ІТ-проєктом та оптимізовано організаційну структуру.

Для нівелювання високого рівня невизначеності автором обґрунтовано використання гібридного підходу до управління (PMBOK для стратегічного процесного рівня та Scrum для ітеративної розробки). Сформовано збалансовану матричну організаційну структуру команди проєкту чисельністю 15 осіб. Для забезпечення прозорості процесів розроблено ієрархічну

структуру робіт (WBS), декомпозовану за 9 фазами життєвого циклу, та побудовано матрицю відповідальності RACI, яка чітко розмежовує обов'язки між аналітиками, архітекторами, розробниками та інженерами з якості.

У середовищі MS Project розроблено детальний календарно-сітьовий графік проєкту загальною тривалістю 543 дні (понад 90 взаємопов'язаних задач).

5. Розроблено систему управління ризиками та апробовано метод освоєного обсягу (EVM).

Автором ідентифіковано та оцінено 12 ключових ризиків проєкту (технологічних, організаційних, безпекових та форс-мажорних). Для найбільш критичних загроз (блекаути, зрив термінів, плінність кадрів) розроблено матрицю протиризикових заходів (превентивних та антикризових).

Практична значущість застосованих методів контролю доведена через моделювання відхилень виконання проєкту. Використовуючи метод освоєного обсягу (Earned Value Management), змодельовано ситуацію затримки ресурсу на етапі ініціації, що призвела до відставання від графіка на 17 днів (SV = -116 950 грн). На основі отриманих звітів запропоновано управлінські рішення щодо застосування методів стиснення розкладу (fast-tracking) на наступних фазах для уникнення глобального зриву дедлайнів.

Виконана кваліфікаційна робота становить собою комплексне дослідження, яке поєднує глибокий теоретичний аналіз IT-середовища з практичною реалізацією архітектурних та управлінських завдань. Запропонована інформаційна система та розроблені документи (паспорт проєкту, WBS, календарний план, кошторис) мають високий ступінь готовності до впровадження у практику органів місцевого самоврядування.

Реалізація даного проєкту дозволить КМДА перейти до моделі предиктивного управління, скоротити експлуатаційні витрати на 15% за 5 років, мінімізувати аварійність транспортної інфраструктури та підвищити прозорість розподілу публічних фінансів.

ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1. Закон України «Про електронні комунікації» від 16.12.2020 № 1089-IX / Верховна Рада України. – Офіц. вид. – К. : Парлам. вид-во, 2021. – 120 с.
2. Закон України «Про публічні закупівлі» від 25.12.2015 № 922-VIII (із змінами) / Верховна Рада України. – Офіц. вид. – К. : Парлам. вид-во, 2021. – 68 с.
3. Національна транспортна стратегія України на період до 2030 року : схвалена розпорядженням Кабінету Міністрів України від 30 травня 2018 р. № 430-р. – К. : КМУ, 2018. – 45 с.
4. Стратегія розвитку міста Києва до 2025 року (в редакції рішення Київської міської ради від 13.05.2021 № 92/92). – К. : КМДА, 2021. – 112 с.
5. ДБН В.2.3-5:2018. Вулиці та дороги населених пунктів. – К. : Мінрегіонбуд України, 2018. – 86 с.
6. ДСТУ ISO 21500:2016. Настанови щодо управління проектами (ISO 21500:2012, IDT). – К. : ДП «УкрНДНЦ», 2017. – 45 с.
7. ДСТУ ISO/IEC 27001:2015. Інформаційні технології. Методи захисту. Системи управління інформаційною безпекою. Вимоги. – К. : ДП «УкрНДНЦ», 2016. – 32 с.
8. Наукова, навчально-методична, спеціальна література, видана українською мовою 8. Бірюков О. В. Управління ІТ-проектами в умовах цифрової трансформації : монографія / О. В. Бірюков. – К. : Центр учбової літератури, 2021. – 215 с.
9. Бойко В. І., Іванов О. М. Геоінформаційні системи в управлінні інфраструктурою : навч. посіб. / В. І. Бойко, О. М. Іванов. – Л. : Новий Світ-2000, 2019. – 302 с.
10. Бушуєв С. Д., Бушуєва Н. С. Управління проектами та програмами організаційного розвитку : підручник. – К. : Саміт-Книга, 2020. – 340 с.

11. Данченко О. Б., Заноза О. А. Оцінка ризиків в ІТ-проєктах: прикладні аспекти // Управління розвитком складних систем. – 2018. – № 35. – С. 12–19.
12. Єгорова О. В. Основи проєктування реляційних баз даних : підручник / О. В. Єгорова. – Х. : ХНУРЕ, 2017. – 245 с.
13. Ільїн О. В. Моделювання транспортних потоків у Smart City // Системні дослідження та інформаційні технології. – 2022. – № 1. – С. 78–85.
14. Карпенко М. В. Оптимізація інвестиційних рішень у муніципальному секторі за допомогою методу динамічного програмування // Економіка та держава. – 2019. – № 4. – С. 34–40.
15. Мельник А. О. Сучасні підходи до управління якістю програмного забезпечення : навч. посіб. – К. : КПІ ім. Ігоря Сікорського, 2020. – 198 с.
16. Морозов В. В., Кальян А. В. Гнучкі методології управління проєктами в умовах невизначеності // Вісник Київського національного університету. Серія: Фізико-математичні науки. – 2021. – № 3. – С. 102–109.
17. Морозов В. В., Семчишин В. Р. Сучасні підходи до ефективного управління проєктами в умовах цифрової трансформації // Матеріали науково-технічної конференції (ІТ&І). – К. : КНУ імені Тараса Шевченка, 2026. – С. 45–48.
18. Морозов В. В., Чередніченко А. М. Управління ризиками в складних ІТ-проєктах : підручник. – К. : Університет «КРОК», 2018. – 280 с.
19. Петренко І. С. Розробка ГІС-додатків на базі PostgreSQL та PostGIS / І. С. Петренко. – О. : Наука і техніка, 2020. – 312 с.
20. Савчук В. П. Оцінка ефективності інвестиційних проєктів: концепції та практика : монографія. – К. : КНЕУ, 2019. – 410 с.
21. Тарасенко О. М. Застосування методу освоєного обсягу для контролю ІТ-проєктів // Проблеми програмування. – 2017. – № 4. – С. 55–63.
22. Шевченко В. І. Цифрові двійники міської інфраструктури: теорія та практика / В. І. Шевченко. – Дніпро : Оксамит, 2023. – 260 с.

23. Beck K., et al. Agile Estimating and Planning. 2nd ed. Pearson Education, 2019. – 330 p.
24. Deemer P., Benefield G. The Scrum Primer: A Lightweight Guide to the Theory and Practice of Scrum. InfoQ, 2020. – 45 p.
25. Fleming Q. W., Koppelman J. M. Earned Value Project Management. 4th ed. Project Management Institute, 2016. – 235 p.
26. Kerzner H. Project Management: A Systems Approach to Planning, Scheduling, and Controlling. 13th ed. Wiley, 2022. – 848 p.
27. Larson E. W., Gray C. F. Project Management: The Managerial Process. 8th ed. McGraw-Hill Education, 2020. – 752 p.
28. Mak S. Microservices: Up and Running. O'Reilly Media, 2021. – 300 p.
29. Martello S., Toth P. Knapsack Problems: Algorithms and Computer Implementations. 2nd ed. Wiley, 2018. – 312 p.
30. Newman S. Building Microservices: Designing Fine-Grained Systems. 2nd ed. O'Reilly Media, 2021. – 614 p.
31. Obe R. O., Hsu L. S. PostGIS in Action. 3rd ed. Manning Publications, 2021. – 600 p.
32. Project Management Institute. A Guide to the Project Management Body of Knowledge (PMBOK® Guide). 7th ed. PMI, 2021. – 370 p.
33. Project Management Institute. Practice Standard for Project Risk Management. PMI, 2019. – 116 p.
34. Schwaber K., Sutherland J. The Scrum Guide: The Rules of the Game. Scrum.org, 2020. – 14 p.
35. Vercellis C. Business Intelligence: Data Mining and Optimization for Decision Making. Wiley, 2018. – 432 p.
36. Atlassian Jira Software Documentation. URL: <https://confluence.atlassian.com/alldoc/> (дата звернення: 12.04.2026).
37. Kyiv Smart City: Офіційний портал. URL: <https://kyivsmartcity.com> (дата звернення: 18.03.2026).

38. Magneticon EMT. Геоінформаційна система автомобільних доріг. URL: <https://magneticonemt.com> (дата звернення: 24.05.2025).
39. PostgreSQL Official Documentation. URL: <https://www.postgresql.org/docs/> (дата звернення: 05.04.2026).
40. Jeff Tao. Characteristics of Time Series Data [Електронний ресурс] / Jeff Tao // TDengine. – 2023. – Режим доступу до ресурсу: <https://tdengine.com/characteristics-of-time-series-data/>.
41. Syarif Yusuf Effendi. ARIMA for Time-Series Forecasting: Retail Sales Predictions [Електронний ресурс] / Syarif Yusuf Effendi // Medium. – 2023. – Режим доступу до ресурсу: <https://medium.com/@syarifyusuf/arima-for-time-series-forecasting-retail-sales-predictions-6e0da74232a0>.
42. Кужда Т. Довірчий інтервал для складання економічних прогнозів [Електронний ресурс] / Тетяна Кужда, Михайло Галушак // Соціально-економічні проблеми і держава. — 2014. — Вип. 2 (11). — С. 162-169. — Режим доступу до журн.: <http://sepd.tntu.edu.ua/images/stories/pdf/2014/14ktisep.pdf>.
43. Will Kenton. What Is PEST Analysis? Its Applications and Uses in Business [Електронний ресурс] / Will Kenton // Investopedia. – 2024. – Режим доступу до ресурсу: <https://www.investopedia.com/terms/p/pest-analysis.asp>.
44. D. M. Burghate, "Work Breakdown Structure: Simplifying Project Management," International Journal of Commerce and Management Studies, vol. 3, no. 2, pp. 1–5, 2018.
45. Adam Hayes. Autoregressive Integrated Moving Average (ARIMA) Prediction Model [Електронний ресурс] / Adam Hayes // Investopedia. – 2023. – Режим доступу до ресурсу: <https://www.investopedia.com/terms/a/autoregressive-integrated-moving-average-arima.asp>.

46. Rob Vettor. Architecting Cloud Native .NET Applications for Azure / Rob Vettor, Steve "ardalis" Smith. – Redmond, Washington: Microsoft Developer Division, 2023.
47. Billy Okeyo. Introduction to Blazor [Электронный ресурс] / Billy Okeyo. – 2023. – Режим доступа до ресурсу: <https://www.billyokeyo.com/posts/introduction-to-blazor/>.
48. Darshana Mihiran Edirisinghe. Blazor Series: Episode 1 — Introduction to Blazor [Электронный ресурс] / Darshana Mihiran Edirisinghe // Medium. – 2024. – Режим доступа до ресурсу: <https://medium.com/@darshana-edirisinghe/blazor-series-episode-1-introduction-to-blazor-8c54bbdeb875>.