

**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:

«Управління проєктом розробки сервісу з шерінгу електросамокатів»

Студента 2-го курсу групи УП-21

Купін Дмитро Олександрович

(прізвище, ім'я, по батькові)

Науковий керівник:

Д.Т.Н., професор

(науковий ступінь, вчене звання)

Данченко О.Б.

(прізвище, ім'я, по батькові)

(підпис студента)

(дата)

(підпис)

Попередній захист:

(Висновок: "До захисту в Екзаменаційній комісії")

Завідувач кафедри
технологій управління

(дата)

(підпис)

(прізвище, ініціали)

Київ – 2023

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет інформаційних технологій

Кафедра технологій управління

Освітній рівень Магістр

Спеціальність 122 Комп'ютерні науки

Освітньо-професійна програма Управління проектами

ЗАТВЕРДЖУЮ

Завідувач кафедри
професор Віктор МОРОЗОВ

“23” 06 2022 року

**ЗАВДАННЯ
НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ**

Студент: Дмитро КУПІН

Групи: УП-21

1. Тема кваліфікаційної роботи: «Управління проектом розробки сервісу з шерінгу електросамокатів». Затверджена на засіданні кафедри технологій управління протоколом №14 від 23.06.2022 р.

2. Строк подання студентом готової роботи “15” травня 2023 р.

3. Цільова установка та вихідні дані до роботи: дослідження характеристики об'єкту управління планування виконання проекту (календарне планування, зміст, бюджет, ресурси) та планування управління іншими галузями управління проектами.

4. Зміст роботи: аналіз та оцінка об'єкту управління, обґрунтування доцільності та життєздатності проекту, визначення

сутності проекту розробки сервісу з шерінгу електросамокатів, проведення аналізу предметної галузі, огляд існуючих вирішень проблеми систем шерінгу самокатів, аналіз конкурентів, аналіз галузі за методом 5 сил портера, проведення swot-аналізу, аналіз зацікавлених сторін проекту, заходи контролю якості, проведення маркетингових досліджень, проведення pest-аналізу, аналіз зовнішнього середовища, аналіз внутрішнього середовища, проведення інвестиційних досліджень, розробка плану виконання проекту, побудова ієрархічної структури робіт проекту, побудова організаційної структури проекту, календарне планування проекту, планування ресурсів проекту, дослідження технологій управління проектом, управління ризиками проекту, управління зацікавленими сторонами проекту, розробка програмного забезпечення проекту, опис структури та модулів програмного забезпечення проекту, створення беклогу проекту, розробка каркасного інтерфейсу, тестування системи.

5. Перелік графічного матеріалу: титульна сторінка, мета дипломної роботи, життєвий цикл проекту, дерево проблем, дерево цілей, каркасний інтерфейс, економічне обґрунтування, маркетингове дослідження, організаційна структура, ієрархічна структура робіт, управління ризиками, календарне планування, використання бюджету, розробка продукту, висновки.

6. Календарний план виконання роботи:

№	Назва роботи	План виконання роботи
1.	Вивчення літературних джерел з предмету дослідження	25.02.2023-01.03.2023
2.	Збір і вивчення матеріалів досліджуваного підприємства	01.03.2023-01.03.2023

3.	Складання розгорнутого плану кваліфікаційної роботи	01.03.2023-07.03.2023
4.	Ознайомлення наукового керівника з розгорнутим планом кваліфікаційної роботи. Внесення змін.	07.03.2023
5.	Підготовка розділу 1 “Аналіз та оцінка об’єкту управління”	14.03.2023-28.03.2023
6.	Підготовка розділу 2 “Розробка плану виконання проєкту”	28.03.2023-12.04.2023
7.	Підготовка розділу 3 “Дослідження технологій управління проєктом”	12.04.2023-26.04.2023
8.	Підготовка розділу 4 “Розробка програмного забезпечення проєкту”	26.04.2023-01.15.2023
9.	Оформлення кваліфікаційної роботи	01.05.2023 - 14.05.2023
10.	Передача кваліфікаційної роботи рецензенту для рецензування	13.05.2023
11.	Передача кваліфікаційної роботи науковому керівникові	15.05.2023
12.	Попередній захист кваліфікаційної роботи	15.05.2023
13.	Захист роботи	22.05.2023

Дата видачі завдання: 25.02.2023 р.

Керівник роботи: д.т.н., професор Данченко О. Б.

(підпис)

Завдання прийняв до виконання:

Студент групи УП-21 Дмитро КУПІН

(підпис)

АНОТАЦІЯ

Кваліфікаційної роботи магістра на тему:

«Управління проектом розробки сервісу з шерінгу електросамокатів»

Студент : Дмитро КУПІН

Науковий керівник : Данченко О. Б.

Рік захисту - 2023

Темою даної кваліфікаційної роботи є – «Розробка сервісу з шерінгу електросамокатів».

Метою даної роботи є створення зручного та доступного сервісу шерінгу електросамокатів, який дозволить користувачам легко та швидко орендувати транспортні засоби для переміщення по місту.

Ціль проекту – розробка та впровадження системи шерінгу самокатів.

Об'єкт дослідження – процес розробки та створення сервісу з шерінгу електросамокатів.

Предмет дослідження – процеси управління проектом розробки системи шерінгу самокатів, сюди входять процеси управління змістом, термінами, вартістю, ризиками, зацікавленими сторонами.

Новизна отриманих результатів полягає в управлінні розробки системи, якої нема на українському ринку, що робить її потрібною та актуальною.

Практичне значення отриманих результатів. Описана система є можливим рішенням для поєднання усіх найкращих аспектів шерінгу електросамокатів у конкурентноспроможному проекті.

Кваліфікаційна робота складається з анотації, вступу, чотирьох розділів, висновків, переліку використаних джерел і додатків.

У *першому розділі* було проведено аналіз існуючих вирішень проблеми шерінгу самокатів, аналіз конкурентів, маркетинговий аналіз та

інвестиційні дослідження. Побудовано дерево проблем і цілей, визначено мету, цілі і продукт проєкту.

У *другому розділі* було розроблено WBS-життєвий цикл проєкту, побудовано ієрархічна структура проєкту, організаційна структура проєкту. Було проведено календарне планування та планування вартістю, ресурсами проєкту.

У *третьому розділі* результати управління зацікавленими сторонами і ризиками.

У *четвертому розділі* було описано процес розробки продукту, який містить структурну схему потоків в системі, структуру бази даних, а також логічну, фізичну та концептуальну модель бази даних. Описано структуру модулів програмного забезпечення, розроблено каркасний інтерфейс продукту та описано процес тестування продукту.

Висновки було оформлено за результатами роботи. Проєкт є актуальним на сьогоднішній день.

Структура та обсяг роботи. Кваліфікаційна робота складається з 97 сторінок з урахуванням додатків, 20 таблиць і 29 рисунків.

Ключові слова : *система, шерінг самокатів, тестрейл, код, управління, планування.*

ЗМІСТ

ЗМІСТ.....	4
АНОТАЦІЯ.....	4
ТАБЛИЦЯ СКОРОЧЕНЬ ТА ПОЯСНЕНЬ	8
ВСТУП	9
РОЗДІЛ 1. АНАЛІЗ ТА ОЦІНКА ОБ'ЄКТУ УПРАВЛІННЯ.....	11
1.1 Обґрунтування доцільності та життєздатності проекту.....	11
1.2 Проведення аналізу предметної галузі	14
1.2.1 Огляд існуючих вирішень проблеми систем шерінгу самокатів.....	14
1.2.2 Аналіз конкурентів	15
1.2.3 Аналіз галузі за методом 5 сил Портера	17
1.2.4 Проведення SWOT-аналізу.....	20
1.2.5 Аналіз зацікавлених сторін проекту	22
1.2.6 Заходи контролю якості	24
1.3 Проведення маркетингових досліджень.....	25
1.3.1 Проведення PEST -аналізу.....	25
1.3.2 Аналіз зовнішнього середовища	31
1.3.3 Аналіз внутрішнього середовища.....	32
1.4 Проведення інвестиційних досліджень	33
1.5 Побудова дерева проблем та дерева цілей.....	37
1.6 Мета, цілі та результат проекту.....	39
РОЗДІЛ 2. РОЗРОБКА ПЛАНУ ВИКОНАННЯ ПРОЄКТУ	41
2.2 Побудова ієрархічної структури робіт проекту.....	42
2.3 Побудова організаційної структури проекту	43
2.4 Календарне планування проекту.....	46
2.5 Планування ресурсів проекту	51
РОЗДІЛ 3. ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ УПРАВЛІННЯ ПРОЄКТОМ... 54	
3.1 Управління ризиками проекту.....	55

3.2 Управління зацікавленими сторонами проекту.....	60
РОЗДІЛ 4. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПРОЄКТУ	65
4.1 Опис структурної схеми інформаційних потоків в системі	65
4.2 Структура бази даних.....	66
4.3 Опис структури та модулів програмного забезпечення проекту.....	69
4.4 Створення беклогу проекту	73
4.5 Розробка каркасного інтерфейсу.....	75
4.6. Тестування системи	77
ВИСНОВКИ.....	79
ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ.....	82
ДОДАТОК А.....	88
ДОДАТОК Б.....	90
ДОДАТОК Г.....	94
ДОДАТОК Д.....	95

ТАБЛИЦЯ СКОРОЧЕНЬ ТА ПОЯСНЕНЬ

Скорочення	Пояснення
ПЗ	Програмне забезпечення
ЖЦ	Життєвий цикл
ЛСС	Логіко-структурна схема
ОСТ	Організаційна структура робіт
БД	База даних
ІРС	Ієрархічна структура робіт
МВ	Матриця відповідальності
ІА	Інвестиційний аналіз
ВС	Внутрішнє середовище
ЗЦС	Зацікавлені сторони

ВСТУП

Актуальність роботи: Україна перебуває у стані трансформаційних процесів у всіх сферах суспільного життя, що вимагає вивчення успішного міжнародного досвіду впровадження змін. Однією з найбільш актуальних сфер, що потребує цифровізації, є транспорт. У сучасному світі шерінг електросамокатів є ефективним та екологічно чистим видом транспорту, який набуває все більшої популярності серед населення. Однак, в Україні наявна можливість зайняти лідируюче місце через конкуренцію з іншими новими сервісами та привнести покращення в цю нішу [1].

Метою даного проекту є розробка та впровадження сервісу з шерінгу електросамокатів в Україні з метою забезпечення населення ефективним та екологічно чистим видом транспорту, що сприятиме покращенню якості життя та зменшенню впливу на навколишнє середовище. Дана робота містить аналіз ринку шерінгу електросамокатів в Україні, розробку концепції та бізнес-плану сервісу, визначення його основних функцій та переваг перед іншими видами транспорту, а також оцінку потенційного економічного та екологічного ефекту від впровадження даного проекту.

Об'єкт дослідження – управління проектом розробки системи шерінгу електросамокатів.

Предмет дослідження – процеси управління проектом розробки системи шерінгу електросамокатів, сюди входять процеси управління змістом, термінами, вартістю, ризиками, зацікавленими сторонами.

Завдання дослідження:

- Аналіз предметної галузі.
- Інвестиційні дослідження.
- Маркетинговий аналіз.
- Розробка організаційної структури проекту.
- Розробка концепції проекту.

- Опис життєвого циклу.
- Опис цілей, мети та продукту проєкту.
- Календарне планування.
- Планування вартістю проєкту.
- Планування ресурсів проєкту.
- Управління ризиками.
- Управління зацікавленими сторонами.
- Опис структури бази даних.
- Опис структури і модулів системи.
- Створення беклогу проєкту.
- Опис розробки каркасного інтерфейсу.
- Опис тестування.

Новизна отриманих результатів полягає в управлінні розробки системи, яка долає недоліки конкурентів на українському ринку, що робить її потрібною та актуальною.

Практичне значення отриманих результатів. Описана система є можливим рішенням для шерінгу електросамокатів.

РОЗДІЛ 1. АНАЛІЗ ТА ОЦІНКА ОБ'ЄКТУ УПРАВЛІННЯ

1.1 Обґрунтування доцільності та життєздатності проекту.

Визначення сутності проекту розробки сервісу з шерінгу електросамокатів.

Проект розробки сервісу з шерінгу електросамокатів є досить актуальним та життєздатним, оскільки в сучасному світі зростає попит на екологічно чисті види транспорту, а також на альтернативні способи пересування в містах з високим трафіком.

Послуги шерінгу електросамокатів є одним зі способів задоволення цих потреб. Це передбачає надання користувачам можливості користуватися електросамокатами за певну плату на певний час, що дає можливість зменшити кількість пробок, викидів CO₂ в атмосферу та полегшити рух пішоходів та велосипедистів по місту.

Сутність проекту полягає в розробці та впровадженні в місто сервісу з шерінгу електросамокатів, що включатиме в себе створення мобільного додатку для замовлення електросамокатів, встановлення спеціальних станцій для їх зарядки та паркування, організацію сервісу обслуговування та ремонту електросамокатів, а також рекламну та маркетингову кампанії для просування сервісу серед потенційних користувачів.

Доцільність проекту базується на популярності та рості попиту на альтернативні способи пересування в містах, а також на тенденції до зменшення використання автомобілів з двигунами з внутрішнім згорянням. Крім того, шерінг електросамокатів є екологічно чистим та вигідним для міської інфраструктури, оскільки не потребує великих затрат на будівництво нових доріг або парковок [2].

Отже, проект розробки сервісу з шерінгу електросамокатів є доцільним та життєздатним, оскільки має потенціал зменшити транспортні проблеми в містах, покращити екологічну ситуацію та

забезпечити зручний та швидкий спосіб пересування для мешканців та гостей міста.

Проект передбачає розробку комплексного сервісу з шерінгу електросамокатів, що буде інтегрований з міською інфраструктурою та буде включати в себе встановлення спеціальних станцій для паркування та зарядки електросамокатів, розробку мобільного додатку для замовлення та оплати послуг, а також організацію сервісу обслуговування та ремонту електросамокатів.

Одним з ключових елементів успіху проекту буде його маркетингова та рекламна кампанія, спрямована на залучення максимальної кількості користувачів та популяризацію сервісу серед мешканців та гостей міста.

Очікуваним результатом проекту є забезпечення мешканців та гостей міста зручним та швидким способом пересування, покращення екологічної ситуації та зменшення транспортних проблем в місті. Крім того, проект може мати позитивний економічний вплив на місто, забезпечуючи нові робочі місця та збільшення туристичного потоку [2].

Назва проекту: "ShareScoot: управління розробкою сервісу з шерінгу електросамокатів".

Модульна система дозволить розробити сервіс ефективніше та забезпечить можливість додавати нові функції в майбутньому. Основними модулями ShareScoot можуть бути наступні:

1. Модуль управління електросамокатами - цей модуль включає в себе систему відстеження розташування електросамокатів, контроль рівня заряду та стану техніки. Також він забезпечує можливість блокування та розблокування електросамокатів для користувачів.
2. Модуль оплати та білінгу - цей модуль включає в себе систему оплати послуг, що включає в себе різні способи оплати, такі як кредитні картки та електронні гаманці. Крім того, він забезпечує ведення білінгу, створення рахунків та відслідковування фінансової

інформації.

3. Модуль мобільного додатку - цей модуль дозволяє користувачам швидко та зручно замовляти електросамокати та оплачувати послуги. Додаток має бути доступним для iOS та Android платформ, містити карту міста з розташуванням електросамокатів та станцій, а також забезпечувати можливість створення облікового запису та відстеження історії користування.
4. Модуль аналітики та звітності - цей модуль забезпечує збір та аналіз даних про користування сервісом, що дозволяє підвищити ефективність та забезпечити належну рівність використання. Він також забезпечує можливість створення звітів про роботу сервісу та заробітку, що дозволяє аналізувати ефективність бізнесу та приймати належні рішення щодо його розвитку.
5. Модуль клієнтської підтримки - цей модуль забезпечує зв'язок з користувачами та надає підтримку клієнтів через різні канали, такі як телефон, електронна пошта та чат. Він також дозволяє вирішувати технічні проблеми та відповідати на запитання користувачів щодо сервісу.
6. Модуль реклами та маркетингу - цей модуль забезпечує розробку та розповсюдження рекламних кампаній для залучення нових користувачів. Він також дозволяє аналізувати ефективність рекламних кампаній та виробляти стратегії маркетингу для збільшення кількості користувачів та заробітку.
7. Модуль безпеки - цей модуль забезпечує безпеку користувачів та техніки. Він включає в себе систему перевірки водійського посвідчення користувачів, систему контролю швидкості та відстані між електросамокатами на дорозі, а також систему сповіщення про аварії та надзвичайні ситуації.

Кожен з цих модулів взаємодіє з іншими для забезпечення оптимальної

роботи сервісу з шерінгу електросамокатів. Разом вони дозволять створити життєздатний та зручний сервіс для користувачів та бізнесу.

1.2 Проведення аналізу предметної галузі

1.2.1 Огляд існуючих вирішень проблеми систем шерінгу самокатів

В Україні вже декілька років успішно функціонують різні системи шерінгу самокатів, серед них можна виділити такі, як Bolt, Lime та Bird.

Bolt, що раніше відома як Taxify, запустила свій сервіс шерінгу електросамокатів в Україні в квітні 2019 року. Ця компанія створила додаток, який дозволяє користувачам легко знайти доступні самокати в місті, замовити їх та відкрити замок за допомогою смартфона. Bolt заявляє, що її електросамокати мають великий запас ходу і є досить стійкими та надійними для щоденного використання.

Bird - це американська компанія, яка також запустила свій сервіс шерінгу електросамокатів в Україні. За допомогою мобільного додатку користувачі можуть знайти доступні самокати в місті та відкрити їх замок за допомогою QR-коду. Компанія заявляє, що її електросамокати мають високу міцність та довговічність, а також були розроблені з урахуванням безпеки користувачів.

Обидві компанії пропонують доступні ціни за користування своїми сервісами та зробили значний внесок у популяризацію шерінгу електросамокатів в Україні.

Однак Bolt та Bird мають певні різниці в їх підходах до бізнесу. Наприклад, Bolt, окрім електросамокатів, пропонує також замовлення таксі, тоді як Bird зосереджується тільки на електросамокатах та електромопедах [3].

1.2.2 Аналіз конкурентів

Аналіз переваг та недоліків конкурентів показано у таблиці 1.1. [4]

Таблиця 1.1 Переваги та недоліків конкурентів

Назва	Переваги	Недоліки
1	2	3
Bolt	<ul style="list-style-type: none">• Зручний спосіб оплати• Система лояльності• Відносно велика кількість самокатів• Можна забронювати електросамокат	<ul style="list-style-type: none">• Ціна(іноді замовити таксі дешевше)• Мала кількість міст України з доступом до сервісу• Достатня кількість самокатів з технічними несправностями• Можна забронювати електросамокат лише на 3хв
Bird	<ul style="list-style-type: none">• Якісні самокати	<ul style="list-style-type: none">• Дуже велика вартість сервісу• Немає зручного способу сплати• Мала кількість самокатів• Доступ лише в двох містах України

Продовження Табл 1.1 Переваги та недоліків конкурентів

Lime	Середня вартість сервісу	<ul style="list-style-type: none"> • Мала кількість самокатів • Якість самокатів • Мала кількість міст України з доступом до сервісу
------	--------------------------	---

Конкурентний аналіз включає огляд конкурентів на ринку електросамокатів в Україні, зокрема Bolt та Bird.

Якісні характеристики електросамокатів Bolt та Bird досить схожі. Обидва стартапи пропонують електросамокати зі швидкістю до 25 км/год, з дальністю поїздки до 25 км та з дисковими гальмами. Щодо вартості, в обох сервісах вартість поїздки розраховується за мінімальний час використання та відстань поїздки.

Як і Bolt, Bird - це глобальна компанія зі штаб-квартирою в США, що надає послуги шерінгу електричних транспортних засобів, таких як електросамокати та електровелосипеди. Bird працює в більш ніж 200 містах у всьому світі, та лише в деяких містах України, таких як Київ та Харків.

У порівнянні з Bolt, Bird має більш високі ціни за хвилину користування, а також не зручний спосіб оплати, де спочатку необхідно поповнити акаунт користувача на фіксовану суму(100 грн або 150 грн), тоді як у Bolt окрім поповнення акаунту можна прив'язати карту і не обмежуватись балансом акаунту.

Негативні аспекти Bolt можна вдосконалити шляхом розширення географії обслуговування, адже на даний момент сервіс працює лише в окремих містах України. Також, більша прозорість у відносинах зі співробітниками та покращення сервісу підтримки клієнтів можуть сприяти збільшенню довіри споживачів та залученню нових користувачів.

Негативними аспектами Lime є також як і у всіх попередніх конкурентів дуже мала географія доступу до електросамокатів, досить повільна та погана підтримка користувачів, зіткнення користувачів з випадковими зняттями грошей з їх карти та неможливість користувачів повернути гроші.

У загальному, Volt та Bird є сильними конкурентами у сфері шерінгу електричних транспортних засобів, але кожна з компаній має свої переваги та недоліки. При розробці проєкту електросамокатів в Україні важливо враховувати ці фактори та забезпечити якісні та зручні послуги для користувачів.

Наш проєкт відрізняється від конкурентів за рахунок використання електросамокатів, що мають переваги порівняно зі звичайними самокатами, зокрема швидкість та дальність їзди. Розширення географії обслуговування. Також, можливість замовлення та оплати через мобільний додаток, більша інформативність про безпеку та правила руху на електросамокаті, людяна підтримка користувачів, а також програма лояльності можуть залучити нових клієнтів та забезпечити збереження старих [4].

1.2.3 Аналіз галузі за методом 5 сил Портера

Аналіз галузі шерінгу електросамокатів в Україні за методом 5 сил Портера дозволяє оцінити рівень конкуренції та привабливості цієї галузі.

Аналіз проводиться на основі п'яти основних сил, які впливають на галузь [5]:

1. Сила конкуренції у галузі
2. Загроза появи нових конкурентів.
3. Загроза заміни продуктів або послуг.
4. Переговорна сила постачальників.
5. Переговорна сила покупателів.

Таблиця 1.2. Аналіз за методом 5 сил Портера

Параметр	Значення рівня	Характеристика	Рекомендовані заходи для компанії
1	2	3	4
Загроза конкуренції	Середня	Проект не є унікальним на українському ринку, проте серед конкурентів є багато недоліків, які з легкістю можна подалати та зайняти на ринку свою нішу.	Розвиватися та вдосконалюватися, залучати клієнтів за допомогою реклами та інших маркетингових заходів, вдосконалювати послуги та систему підтримки користувачів.
Рівень і загрози внутрішнього ринку своєї конкуренції	Низький	Наявність конкурентів, які мають власну базу користувачів та свої недоліки.	Розширювати асортимент послуг та додавати нові функції для приваблення клієнтів.

Продовження таблиці 1.2. Аналіз за методом 5 сил Портера

1	2	3	4
Загроза ринкової влади споживачів	Низький	Клієнтам притаманна низька ризикованість.	Розвивати систему зворотного зв'язку з клієнтами та швидко реагувати на їхні запити та пропозиції.
Загроза з боку постачальників.	Високий	Ризик від підняття цін на поставки, зрив поставок, неякісна сировина	Пошук нових постачальників та укладання довгострокових контрактів, щоб зменшити вплив змін цін.

Отже, підсумовуючи проведений аналіз галузі шерінгу електросамокатів в Україні за методом 5 сил Портера, можна зробити наступні висновки:

1. Загроза нових конкурентів є середньою, оскільки на ринку вже є декілька гравців, але їм можна легко нав'язати конкуренцію через реалізацію більш зручного та вигідного сервісу.
2. У більшості міст в Україні не існує серйозних правових обмежень на використання електросамокатів, що створює сприятливі умови для розвитку галузі.
3. Існує питання стандартизації технічних параметрів електросамокатів та їх розміщення на вулицях, що може призвести до зниження вартості та підвищення якості послуг.
4. Залежність від постачальників та виробників електросамокатів може призвести до збільшення витрат на закупівлю техніки та послуг.

5. Наявність великих гравців на ринку, таких як Lime та Bolt, може створювати перешкоди для нових учасників ринку.

Загалом, галузь шерінгу електросамокатів в Україні є перспективною та має потенціал для подальшого розвитку. Однак, існують питання щодо регулювання галузі та вирішення технічних проблем, які потребують уваги та дослідження [6].

1.2.4 Проведення SWOT-аналізу

Було виконано аналітичний метод, який дозволяє оцінити сильні та слабкі сторони проєкту, а також можливості і загрози, які впливають на нього.

Таблиця 1.3. SWOT-аналіз

	Можливості (O):	Загрози (T):
<p>Сильні сторони (S):</p> <p>1. Наявність передумов для успішного ведення бізнесу у вигляді доступу до інфраструктури.</p> <p>2. Зручна пристосована організаційно-правова форма ведення бізнесу</p> <p>3. Висока доступність для користувачів.</p>	<p>1. Підтримка декількох варіантів монетизації.</p> <p>2. Насичення сервісу соціальними можливостями для підтримки залученості користувачів.</p> <p>3. Колаборації із провайдерами зв'язку.</p> <p>Проведення промо-акцій, відкритих демо-тестів для всіх бажаючих.</p> <p>3. Використання переваг вдалого місцезнаходження для конфігурування та оптимальної роботи серверів по всій Україні.</p>	<p>1. Недостатньо ефективне сприйняття продукту проєкту цільовою аудиторією.</p> <p>2. Ймовірність низького попиту через небажання змінювати провайдера для забезпечення високошвидкісного інтернету.</p> <p>3. Поява нових конкурентів.</p>

Продовження Таблиці 1.3. SWOT-аналіз

<p>4.Розширені можливості за рахунок влучної пропозиції сервісу.</p>	<p>4.Підвищення кваліфікації персоналу за рахунок проведення семінарів та тренінгів</p>	<p>4. Недостатній рівень сумісності з духом і основною діяльністю виробника.</p> <p>5. Низький рівень взаємозв'язку з материнською компанією</p>
<p>Слабкі сторони (W):</p> <p>1.Відсутність досвіду у веденні транспортних додатків.</p> <p>2.Нестача кваліфікованих кадрів, які працювали раніше над створенням продуктів такого типу.</p> <p>3.Забезпечення недостатньо високого рівня якості, як наслідок ризиків, пов'язаних з постачальником й виробником електросамокатів.</p> <p>4.Значна залежність від фінансових показників проекту</p>	<p>1.Набуття досвіду через knowledge-sharing сесії з колегами з інших редакцій.</p> <p>2. Розробка нових рішень та інструментів для оновлення та підтримання в інфраструктури в належному стані.</p> <p>3. Введення новітніх систем трекінгу.</p>	<p>1. Розробка алгоритму дій у випадку, якщо ринок не ефективно сприйме продукт проекту.</p> <p>2. Розробити заходи щодо зменшення залежності від діяльності основного виробника.</p> <p>3.Визначити основні джерела фінансування та визначити основні фінансові показники проекту для оцінки ефективності роботи.</p>

Проведений аналіз підтвердив, що є можливість зайняти певну позицію на ринку та покращити зручність інфраструктури багатьох міст України [7,8].

1.2.5 Аналіз зацікавлених сторін проєкту

Зацікавленими сторонами проєкту є: Користувачі, постачальники програмних продуктів, інвестори та команда проєкту.

Користувачі хочуть отримати якісний програмний продукт що задовольняє їх вимоги.

Постачальники програмних продуктів хочуть отримати готовий програмний продукт, яким захочуть користуватись їх клієнти.

Інвестори хочуть отримати прибуток та мати можливості для подальших інвестицій.

Команда проєкту хоче створити якісний програмний продукт, отримати досвід та навички [9].

Вимоги зацікавлених сторін до якості

Вимоги користувачів до якості: правильна робота програми, відповідність можливостей програми вимогам до неї, відповідність вимогам бізнесу, широкий функціонал, зручний інтерфейс, зворотній зв'язок, вчасні оновлення, інструкції для користувачів, переваги над програмами-конкурентами.

Вимоги постачальників до якості: задоволення потреб користувача, якісна реалізація, висока якість програми, переваги над програмами-конкурентами

Вимоги інвесторів до якості: окупність проєкту, високі прибутки, залучення нових користувачів, можливість подальших інвестицій.

Вимоги проектної команди до якості: вчасна реалізація в межах бюджету, якісне управління і розвиток, чітко поставлені задачі, оплата праці, здобуття нових знань.

Визначення зацікавлених сторін

Визначення зацікавлених сторін (Stakeholder Analysis) є важливою складовою управління проектом. Цей процес дозволяє ідентифікувати всіх осіб та організації, які можуть бути зацікавлені в проекті, а також визначити їхні очікування та вплив на проект.

Визначення зацікавлених сторін допомагає забезпечити успіх проекту, оскільки дозволяє управлінцям проекту зрозуміти потреби, очікування та вимоги зацікавлених сторін. Це може допомогти забезпечити, що проект відповідає потребам зацікавлених сторін та забезпечити їх задоволеність результатами проекту [9].

Крім того, визначення зацікавлених сторін дозволяє управлінцям проекту виявити потенційні ризики та проблеми, пов'язані з проектом, і розробити плани їх управління. Це може допомогти зменшити вплив негативних факторів на проект та забезпечити його успішне завершення [9,10].

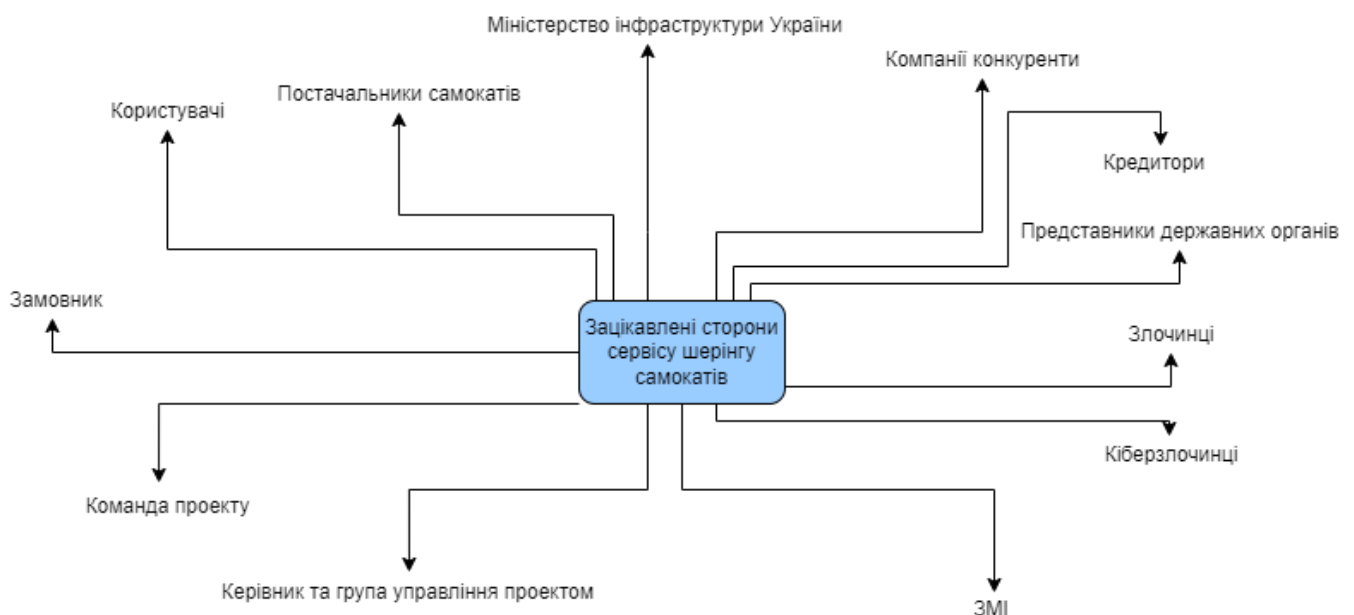


Рис. 1.1 Зацікавлені сторони проекту

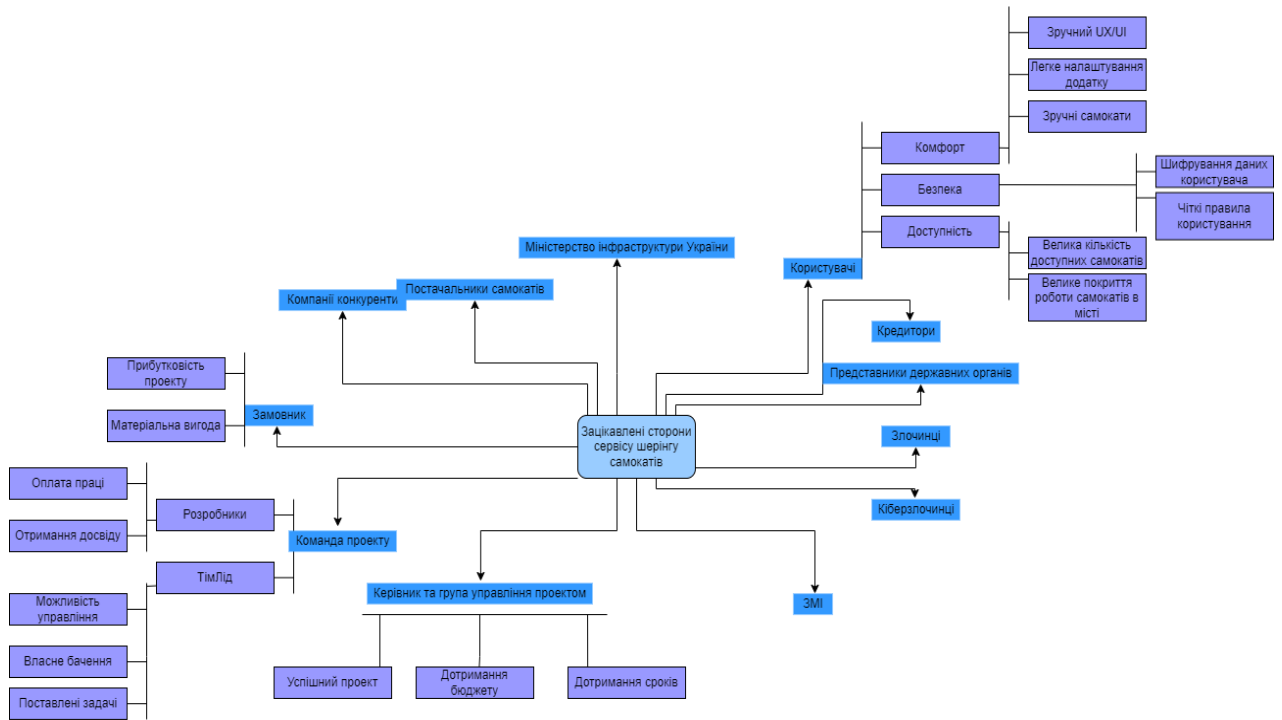


Рис 1.2 Вимоги зацікавлених сторін до якості проекту

1.2.6 Заходи контролю якості

Діаграми причинно-наслідкових зв'язків, також звані діаграмами «риб'ячий скелет» або діаграмами Ішікави.

Опис проблеми, розташований в голові «риб'ячого скелета», використовується в якості відправної точки для відстеження джерела проблеми до першопричини, що вимагає вживання заходів. Опис проблеми, як правило, являє собою виклад проблеми як недоробки, яку необхідно усунути, або мети, яку необхідно досягти.

Пошук причин здійснюється шляхом вивчення опису проблеми й пошуку відповідей на питання «чому» до тих пір, поки не буде ідентифікована першопричина, що вимагає прийняття заходів, або до тих пір, поки не будуть вичерпані всі обгрунтовані можливості на кожній частині ри�'ячого скелета.

Діаграми «риб'ячий скелет» часто виявляються корисними під час пошуку зв'язку небажаних ефектів, розглянутих як особлива варіація, зі

встановленою причиною, відносно якої команди проекту повинні виконати коригуючі дії для усунення даної особливої варіації.

Проектом, для аудиту є проект зі створення сервісу шерінгу самокатів.

Причини, що теоретично могли привести до зниження ефективності наведено нижче на діаграмі Ішікави [16]:

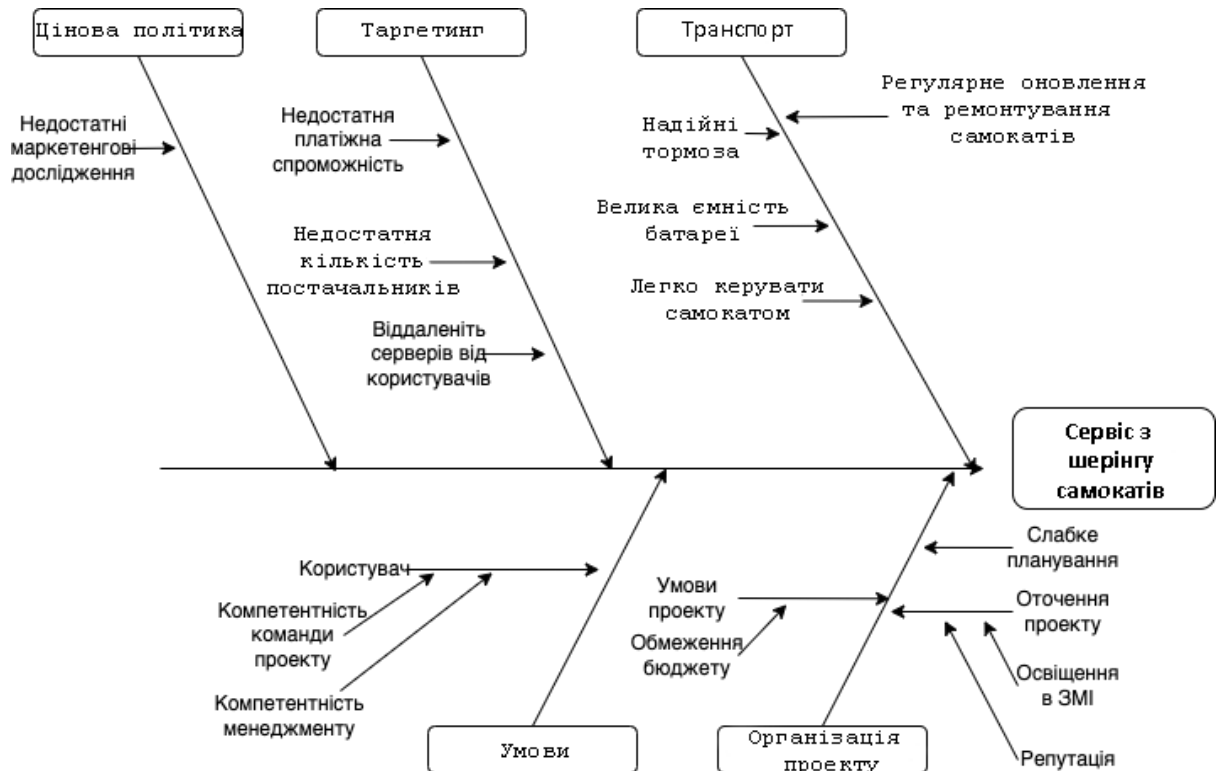


Рис 1.3. Діаграма Ішікави

1.3 Проведення маркетингових досліджень

1.3.1 Проведення PEST -аналізу

PEST-аналіз – це методика макроекономічного аналізу зовнішнього середовища організації, яка використовується для визначення політичних (Political), економічних (Economic), соціальних (Social) та технологічних (Technological) його аспектів, які впливають на бізнес компанії та надають розуміння ринку, місця і потенціалу компанії та перспективних напрямків розвитку [11,12].

Таблиця 1.4 Pest-аналіз

№	Зацікавлені сторони	Потреба	Вигода
1	Інвестори	Створення продукту для задоволення потреб споживачів.	Отримання стабільного прибутку від надання послуг.
2	Команда проекту	Розширення бази клієнтів для отримання нових проєктів та отримання досвіду.	Вдалий випуск продукту в результаті успішної реалізації проєкту та аналізу ринку.
3	Користувачі	Отримання сервісу для полегшення пересування.	Економія часу, соціалізація.
4	Конкуренти	Копіювання і поліпшення ідей конкурента.	Просування власного продукту. Отримання матеріальних винагород та збільшення аудиторії продукту.
5	Кіберзлочинці	Отримання інтелектуальної власності та персональних даних.	Вимагання матеріальних відшкодувань.

Характер та ступінь впливу політичних факторів на компанію [14]

Таблиця 1.5

Фактори впливу політичного середовища	Характер впливу фактору	Оцінка експертів ступеня впливу фактору				Середній бал
		Експерт 1	Експерт 2	Експерт 3	Експерт 4	
Політика щодо впровадження локдаунів	+	3	2	2	3	+2.5
Податкова політика	-	1	1	2	2	-1.

держави						5
Тенденції до регулювання або дерегулювання галузі	-	3	2	1	2	-2
Зміни в трудовому законодавстві	-	1	1	3	1	-1.5
Майбутнє і поточне законодавство, що регулює правила роботи в галузі	+	2	1	1	2	+1.5
Підтримка інноваційних компаній з боку держави	+	3	3	2	3	+2.75
Зміни положення ІТ компаній в країні	+	3	3	3	3	+3
Соціальна взаємодія	+	1	2	3	1	+1.75
Підтримка державою оздоровчих ініціатив	+	1	1	1	1	+1

Характер та ступінь впливу економічних факторів на компанію

Таблиця 1.6

Фактори впливу економічного середовища	Характер впливу фактору	Оцінка експертів ступеня впливу фактору				Середній бал
		Експерт 1	Експерт 2	Експерт 3	Експерт 4	
Зміни матеріального положення суспільства	+	1	1	1	2	+1,25
Тарифи на електроенергію	-	1	2	1	1	-1,25
Динаміка зайнятості	+	1	2	3	2	+2
Рівень безробіття	-	1	1	1	1	-1
Динаміка ВВП	+	2	1	1	1	+1,25

Характер та ступінь впливу соціально-культурних факторів на компанію

Таблиця 1.7

Фактори впливу соціально-культурного середовища	Характер впливу фактору	Оцінка експертів ступеня впливу фактору				Середній бал
		Експерт 1	Експерт 2	Експерт 3	Експерт 4	
Соціальна взаємодія	+	3	2	3	3	+2,75
Зміни базових цінностей	+	1	2	1	1	+1,25

Продовження Таблиці 1.7

Стилі в способі і рівні життя	+	3	3	3	3	+3
Відношення до роботи і відпочинку	+	1	2	3	2	+2
Рівень освіти	+	1	1	1	1	+1
Працевдатність населення	+	1	1	1	2	+1

Характер та ступінь впливу технологічних факторів на компанію

Таблиця 1.8

Фактори впливу технологічного середовища	Характер впливу фактору	Оцінка експертів ступеня впливу фактору				Середній бал
		Експерт 1	Експерт 2	Експерт 3	Експерт 4	
Розвиток технологій	+	1	2	1	2	+1.5
Законодавство в галузі	+	2	1	2	1	+1.5
Витрати на розробку ПЗ	-	2	2	1	2	-1.75
Нові патенти	+	1	2	3	2	+2
Доступ до технологій	+	3	2	3	2	+2.5
Розвиток конкурентних технологій	+	1	2	1	2	+1.5
Зміни тенденцій індустрії	-	1	1	1	2	-1

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

Таблиця 1.9

Політичні		Економічні	
Фактор	Вага	Фактор	Вага
Зміни положення ІТ компаній в країні	+ 3	Динаміка ВВП	+ 2
Зміни в трудовому законодавстві	- 1.5	Тарифи на електроенергію	- 1.25
Соціально-культурні		Технологічні	
Фактор	Вага	Фактор	Вага
Соціальна взаємодія	+ 3	Доступ до технологій	+ 2,5
Працевдатність населення	+ 1	Витрати на розробку ПЗ	- 1.75

Підсумки здійсненого PEST-аналізу для компанії [15]:

Таблиця 1.10

Фактори	Зміни в галузі	Зміни в організації	Дії
Політичні	1. Спроба держави блокувати обхідні шляхи в трудовому праві 2. Мотивація стартапів	1. Офіційний найм працівників 2. Виникнення сприятливих передумов до розвитку	1. Створення та впровадження контрмір для боротьби з впливом уряду 2. Участь у національних тендерах

Продовження Таблиці 1.10

Економічні	<ol style="list-style-type: none"> 1. Залежність від економічного стану країни 2. Залежність від електроресурсів в країни 	<ol style="list-style-type: none"> 1. Необхідність оптимізувати прибутки 2. Перехід до більш оптимального використання енергії 	<ol style="list-style-type: none"> 1. Створення спеціального відділу по оптимізації прибутків 2. Реновації офісів та відмова від використання дизельного транспорту
Соціально-культурні	<ol style="list-style-type: none"> 1. Зростання популярності здорового способу життя 2. Збільшення працездатності населення 	<ol style="list-style-type: none"> 1. Проведення тематичних тимблдінгів 2. Активізація маркетингового відділу 	<ol style="list-style-type: none"> 1. Розширення та вихід на нові ринки з новими пропозиціями 2. Збільшення плати за преміум підписку
Технологічні	<ol style="list-style-type: none"> 1. Збільшення доступу до технологій 2. Збільшення витрат на розробку ПЗ 	<ol style="list-style-type: none"> 1. Відкриття нових відділів розробки 2. Оптимізація відділу кадрів для відбору працівників 	<ol style="list-style-type: none"> 1. Впровадження нових фіч до додатку 2. Скорочення термінів розробки або відмова від запланованого функціоналу

1.3.2 Аналіз зовнішнього середовища

Цільовий ринок складається з виробників самокатів, сервісних центрів, користувачів шерінгу, провайдери серверів, держава. Виробники електросамокатів мають потребу у сбуті продукції на ринок крупними

замовленнями. Сервісні центри мають потребу у наданні своїх послуг задля отримання прибутку та розширення свого бізнесу. Мешканці міста мають потребу у зручному та дешевому пересуванні містом. Держава має потребу у набутті прибутку з податків з цілої низки задіяних бізнесів. Провайдери серверів мають потребу у наданні своїх послуг задля отримання прибутку та розширення свого бізнесу [16].

1.3.3 Аналіз внутрішнього середовища

Основою внутрішнього середовища є:

1. Замовник
2. Керівник та команда проекту (ведення проекту, розробка продукту)
3. Сервісний центр (надання послуг із ремонту та модифікування електросамокатів на постійній основі)
4. Виробники самокатів (постачання електросамокатів великими партіями на постійній основі, покращення власного продукту на основі відгуків великої низки користувачів, швидкий окуп платформи на якій створюються електросамокати завдяки великим об'ємам продажів)

Ресурси проекту поділяються на:

1. Трудові – команда проекту. Для розробки системи необхідно: Менеджер проекту, Аналітик, 2 Frontend-розробники, 2 Backend-розробники, Тестувальник, Дизайнер.
2. Матеріальні – обладнання для роботи, самокати. Серед технологій – сервери, ноутбуки, засоби інтернету для комунікації.

1.4 Проведення інвестиційних досліджень

Бізнес-планування та бізнес-проектування - це два ключових етапи в процесі створення нового бізнесу або розвитку існуючого.

Бізнес-планування - це процес розробки стратегії і плану дій для створення нового бізнесу або розвитку існуючого. Це включає в себе аналіз галузі, конкурентів, потенційного ринку, оцінку ризиків та прибутковості проекту, а також визначення необхідних ресурсів та стратегії їхнього залучення. Бізнес-планування допомагає зрозуміти, чи є ідея бізнесу реалізовуваною, які вимоги потрібні для запуску бізнесу, та оцінити його прибутковість [18].

Бізнес-проектування - це процес розробки конкретного проекту в межах стратегії, визначеної в бізнес-плані. Це включає в себе планування бюджету, розробку технічного завдання, складання графіку виконання робіт, визначення потрібних ресурсів та персоналу. Бізнес-проектування є більш конкретною стадією в процесі створення бізнесу, яка передбачає визначення детальних етапів та ресурсів для втілення стратегії, визначеної в бізнес-плані [19].

Побудова економічної моделі проекту здійснювалась за наступними припущеннями:

1. інтервал планування - місяць;
2. строк реалізації проекту – 10 міс.;
3. розрахунки ведуться з поточними цінами (із урахуванням інфляції);
4. ставка податку на додану вартість - 20%.
5. щорічний темп інфляції — 19.6 %;
6. зарплата виплачується працівникам 1 раз на місяць;

Дохід в межах проекту визначається продажом системи замовнику на 10 місяці проекту. Підсумковий дохід від продажу - 36 405 000 грн.

Інвестиційний аналіз

Інвестиційний аналіз проєкту було виконано в Microsoft Excel:

Виручка/Поточні затрати

Виручка від реалізації починає надходити з третього місяця і постійно зростає за рахунок збільшення кількості клієнтів [23].

Повні поточні витрати є протягом усього проєкту (рис. 1.4.).

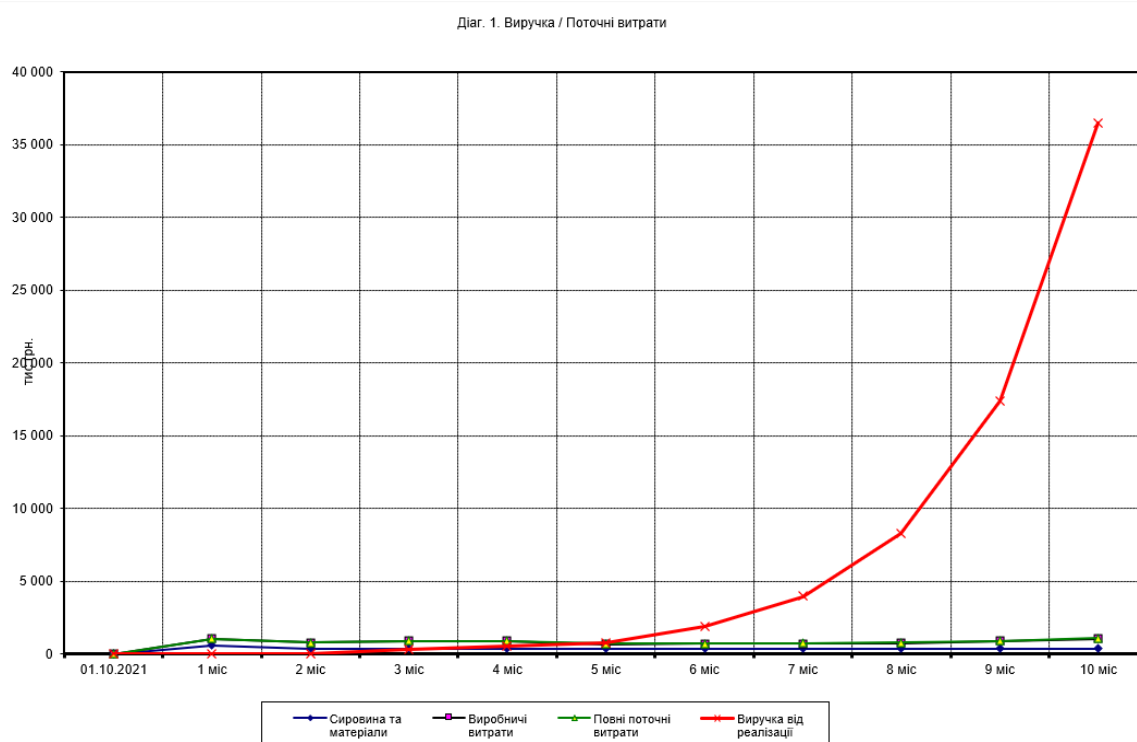


Рис. 1.4. Виручка/ Поточні затрати

Чистий капітал

Потреба в фінансуванні чистого оборотного капіталу незначно зростає до 5 місяця та поступово знижується до 10-го.

Чистий оборотний капітал наведено на рис. 1.5.

Діаг. 2. Чистий оборотний капітал

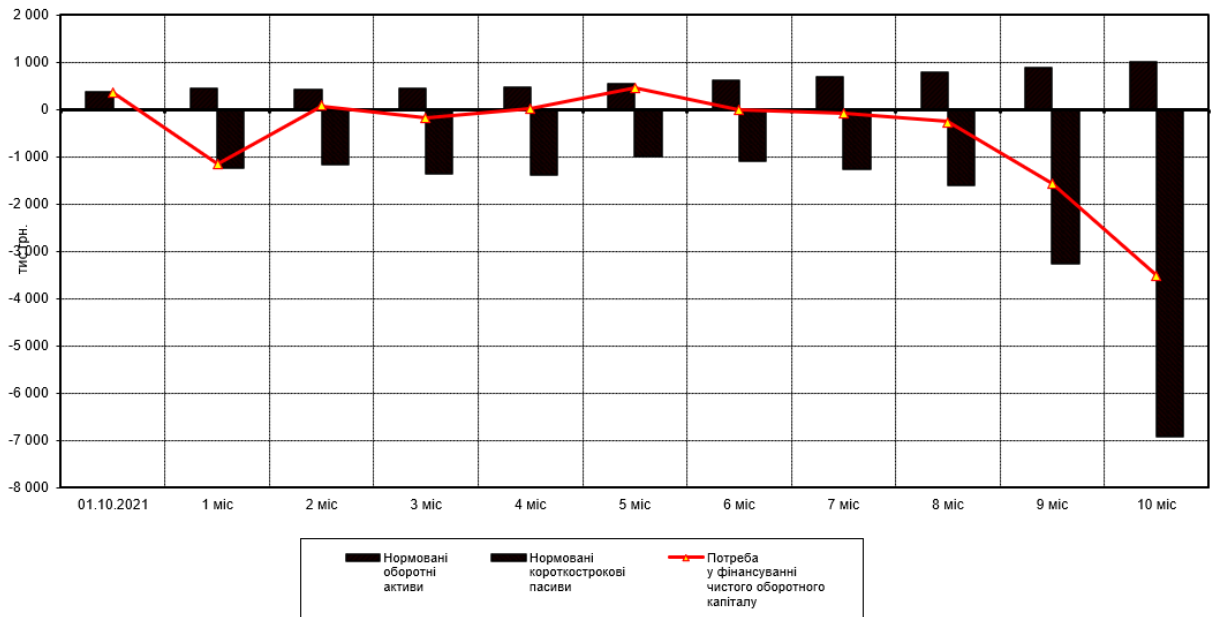


Рис. 1.5. Чистий оборотний капітал

Виплати по кредиту

На графіку погашення боргів показано як залишок заборгованості скорочується, починаючи з 4 місяця ведення діяльності проекту.

Діаг. 3. Обслуговування заборгованості

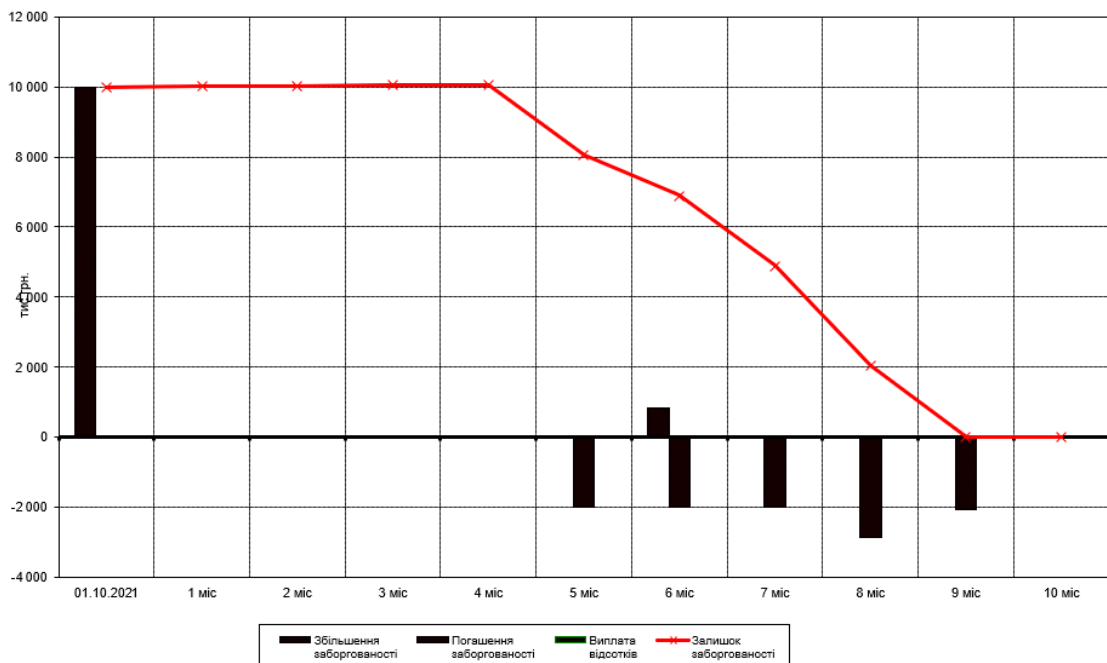


Рис. 1.6. Обслуговування заборгованості

Прибуток

Станом на 10 місяць чистий прибуток досягає 36 405 000 грн. Це показано на рис. 1.7.

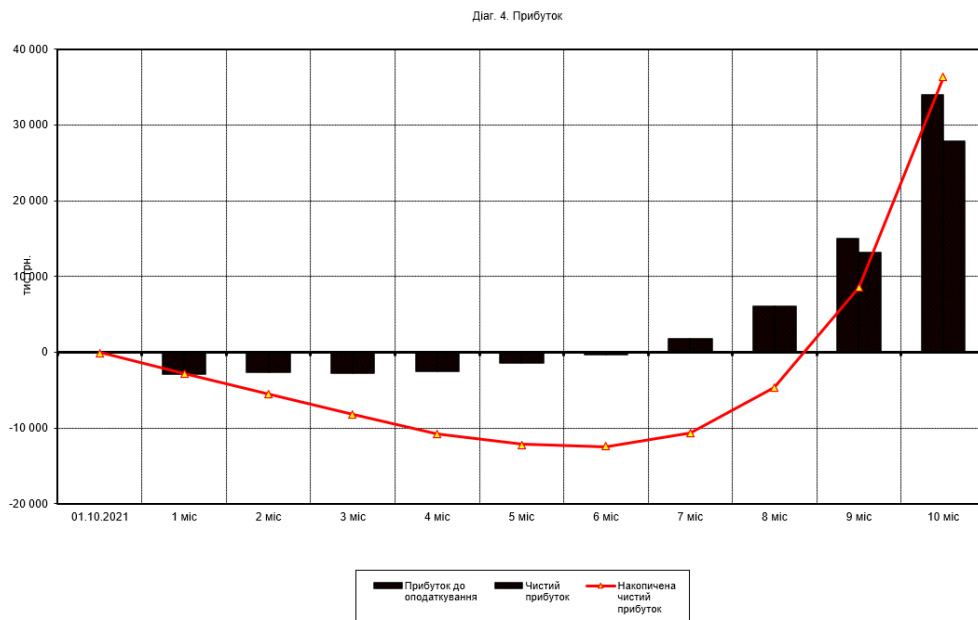


Рис. 1.7. Прибуток

Рух грошових активів

На графіку руху грошових коштів чітко видно первинний відтік, спричинений витратами до початку отримання прибутку, що потім змінюється на зростання об'ємів вільних грошових коштів.

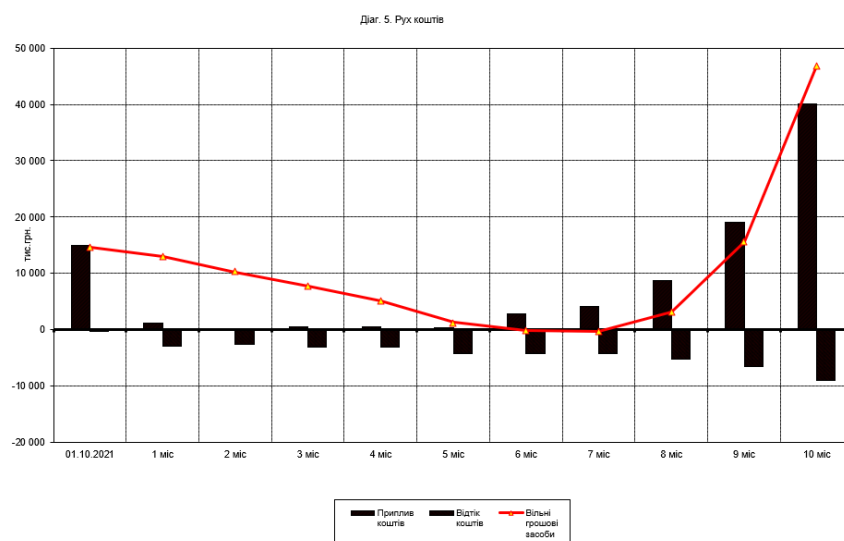


Рис. 1.8. Рух грошових коштів

Основні показники ефективності

Основні показники ефективності представлені на рисунку 1.8.

ПОКАЗНИКИ ЕФЕКТИВНОСТІ ПОВНИХ ІНВЕСТИЦІЙНИХ ВИТРАТ			
Простий строк окупності	років	0,7	
Дисконтований строк окупності	років	0,7	
NPV (чиста поточна вартість проєкту) включно з остаточною (ліквідаційною) вартістю проєкту	тис.грн.	37 846	0
IRR (внутрішня норма дохідності)			
- розрахункова на інтервал планування		24%	
- реальна річна		292%	
- номінальна річна		314%	
Максимальна ставка відсотків по кредитам в межах періода планування			
- розрахункова на інтервал планування		26%	0%
- реальна річна		309%	0%
- номінальна річна		332%	18%
Норма дохідності повних інвестиційних витрат		4292%	0%

Рис. 1.9. Показники ефективності

1.5 Побудова дерева проблем та дерева цілей

Дерево проблем - це інструмент аналізу проблем, що виникають в певній ситуації, проєкті або організації. Основна мета дерева проблем полягає в тому, щоб зрозуміти корінні причини проблем і знайти ефективні рішення для їх вирішення [37].

За допомогою дерева проблем можна візуалізувати проблему, розбити її на більш малі компоненти, встановити причинно-наслідкові зв'язки між цими компонентами та ідентифікувати кореневі причини проблем. Цей підхід дозволяє знайти оптимальні рішення, які не тільки вирішують конкретну проблему, але й допоможуть попередити її повторення у майбутньому.

Таким чином, метою створення дерева проблем є зрозуміти причини проблеми та розробити ефективні рішення для її вирішення. Це може допомогти підприємствам зменшити витрати, покращити якість продукту або послуги, збільшити ефективність процесів та підвищити задоволеність клієнтів.

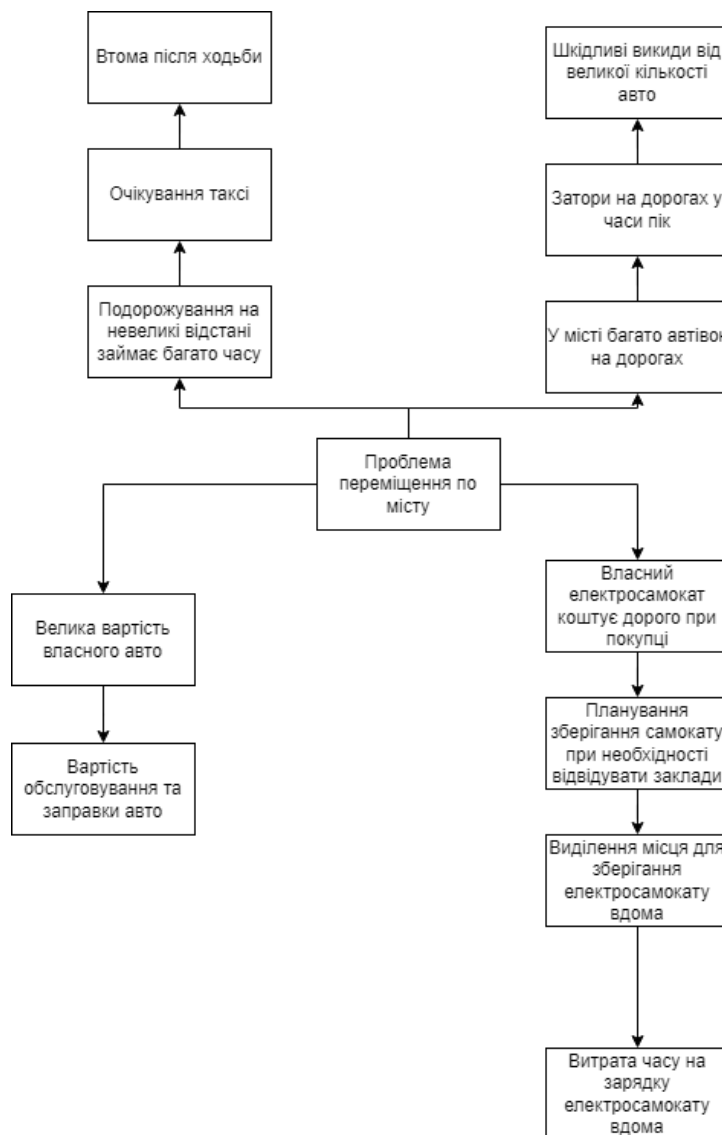


Рис. 1.10. Дерево проблем

Дерево цілей продукту розробляється з метою чітко визначити головну мету або ціль продукту, а також підцілі, необхідні для її досягнення. Це допомагає команді розробки продукту краще зрозуміти, що вони намагаються досягти та в який спосіб. Дерево цілей продукту є потужним інструментом, який дозволяє зосередитися на тому, що є дійсно важливим для продукту, а також підвищити ймовірність досягнення цілей продукту. Він допомагає команді продукту бути більш орієнтованою на результат та зрозуміти, як їх робота впливає на головну мету продукту [38,39].

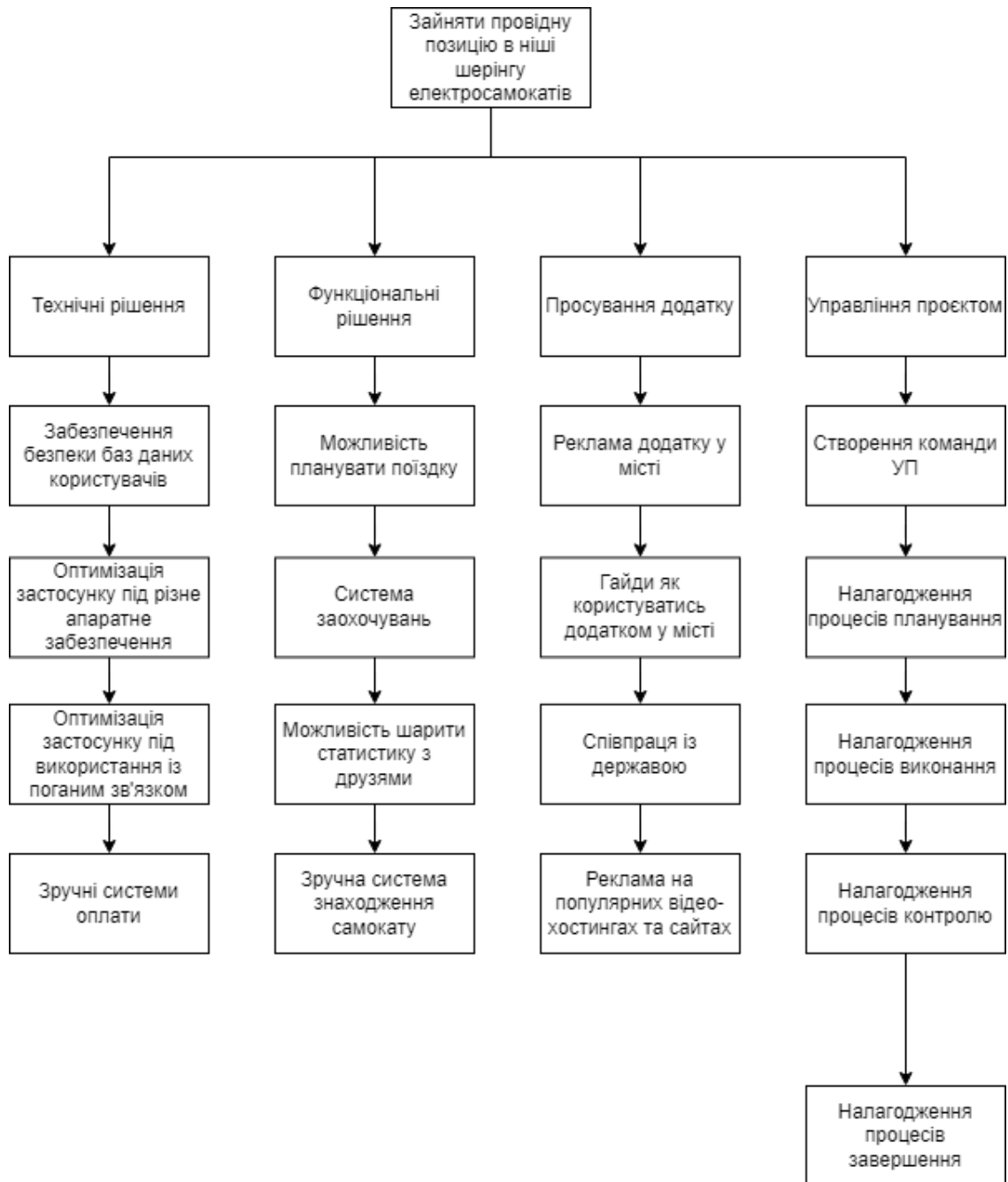


Рис. 1.11. Дерево цілей

1.6 Мета, цілі та результат проєкту

Основна мета проєкту:

Зайняти провідну позицію на ринку України. Створити потужний, надійний, функціональний, доступний кожному сервіс, що дозволить

брати в шерінгу електросамокати користувачам з будь-якого девайса [40,42].

Цілі проекту:

- Отримати 10 000 завантажень із цифрових платформ розповсюдження протягом перших 2 місяців після релізу.
- Отримати понад 90% позитивних відгуків.
- Досягнути CSI - індексу задоволеності клієнтів щонайменше 80%.
- Підвищити рівень Retention rate на 25% протягом 6 місяців з моменту релізу.

Результати проекту:

- Головний результат - випуск додатку, що буде доступний на кожную мобільну платформу.
- Сервіс має низький поріг входу для новачка.
- Заключено контракт з виробником електросамокатів.
- Ведеться сумісна діяльність з муніципальною владою.

РОЗДІЛ 2. РОЗРОБКА ПЛАНУ ВИКОНАННЯ ПРОЄКТУ

2.1 Розробка життєвого циклу проєкту

Розробка життєвого циклу проєкту - це процес планування та виконання проєкту від початку до кінця з урахуванням всіх етапів та ризиків, які можуть виникнути в процесі його виконання [43].

Таблиця 2.1. Життєвий цикл проєкту

Фаза	Дії	Тривалість
1	2	3
1. Ініціація	Початок роботи над концепцією проєкту. Формування статуту проєкту. Проведення маркетингових досліджень. Розробка організаційної структури та матриці відповідальності.	30 днів
2. Планування	Створення дорожньої карти проєкту. Створення команди проєкту. Розробка календарного графіку. Проведення організаційної наради. Планування бюджету.	54 днів
3. Проведення закупівель	Закупівля необхідного обладнання для команди розробки.	30 днів
4. Виконання	Розробка програмного забезпечення, управління проєктом, якістю, ризиками, закупівлями, командою, процесами, комунікаціями.	80 днів
5. Контроль	Постійне відслідковування виконання задач. Контроль витрат ресурсів та своєчасність виконання задач. Проведення проміжних нарад.	Протягом всього ЖЦ

Продовження таблиці 2.1. Життєвий цикл проекту

1	2	3
6. Тестування	Тестування програмного забезпечення для контролю якості продукту.	80 днів
7. Впровадження ПЗ	Підготовка клієнтського інтерфейсу, підготовка серверу, бази даних і допоміжних сервісів. Технічне впровадження	14 днів
8. Закриття проекту	Реліз проекту. Оплата та закриття всіх рахунків. Оформлення кінцевої документації по проекту. Фінальна нарада.	7 днів

2.2 Побудова ієрархічної структури робіт проекту

Побудова ієрархічної структури робіт проекту (WBS) відбувається з метою детального розбиття проекту на окремі етапи та роботи з метою їх легшого керування, планування та контролю. WBS дозволяє розглядати проект як сукупність окремих елементів, кожен з яких може бути керований та контрольований окремо. Це дозволяє забезпечити більш точну оцінку тривалості та вартості проекту, а також визначити необхідні ресурси та завдання для кожного елемента проекту. Крім того, WBS дозволяє структурувати проект таким чином, що кожен елемент проекту відповідає конкретній меті проекту та може бути оцінений за ступенем виконання [44].

Рисунок 2.1 стає ілюстрацією основних етапів та робіт в проекті

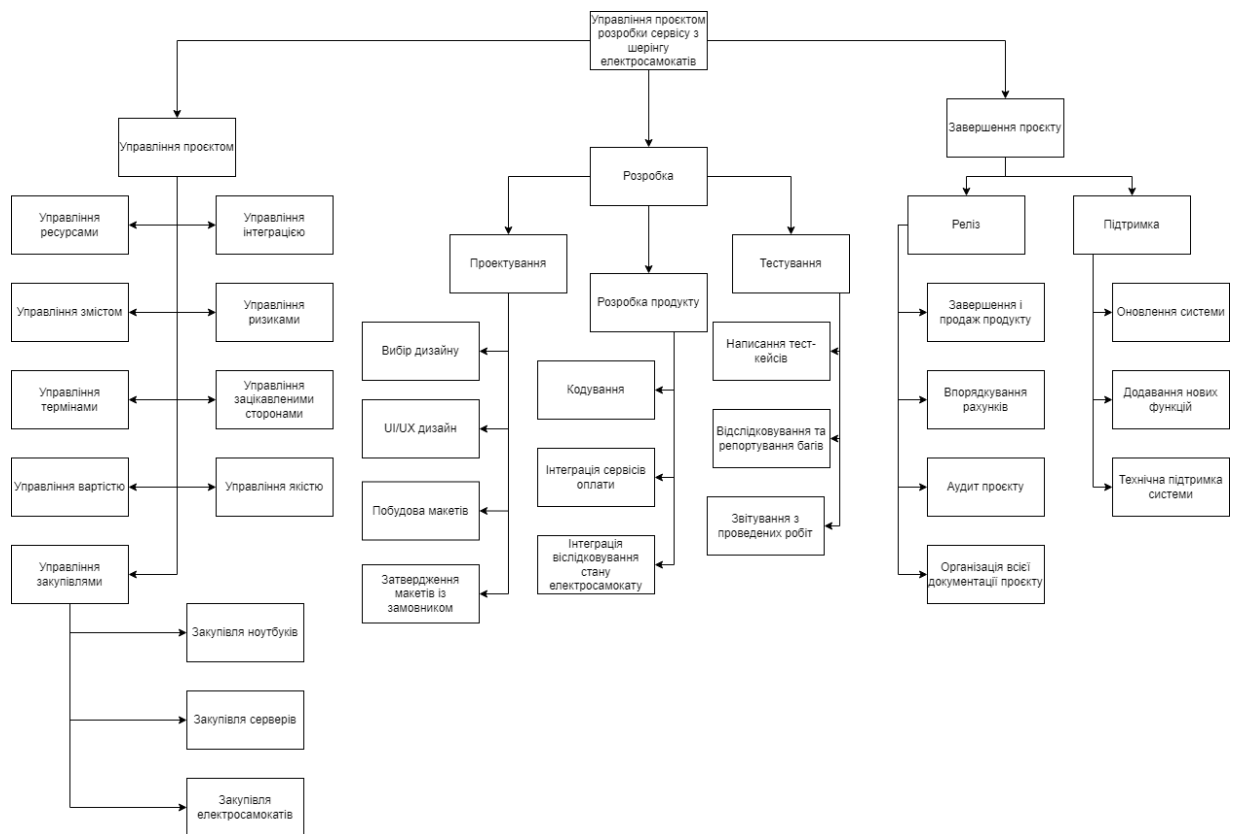


Рисунок 2.1. WBS проекту

2.3 Побудова організаційної структури проекту

Опис організаційної структури управління проектом

Матрична структура дає можливість гнучко маневрувати людськими ресурсами за рахунок перерозподілу їх між проектами, але при умові збереження їх адміністративної належності відповідним функціональним відділам. Особливістю матричної структури управління являється відсутність у керівника проекту контролю над персоналом, зайнятим проектом. Керівник проекту визначає, що і коли повинно бути виконано, а функціональний керівник вирішує, хто і як буде виконувати роботу.

За всі кінцеві результати роботи по здійсненню проекту, включаючи витрати виробництва, витрачений час і якість проекту, відповідає керівник проекту. Використовуючи цей тип структури важливо постійно слідкувати, щоб фактичні дані відповідали плановим, добре налагодити систему контролю за ходом виконання робіт по проекту, якістю

виконання, витратами та термінами. Керівник проекту повинен володіти детальною інформацією відносно всього проекту, а керівники відділів – відносно робіт, що виконуються їх відділами. На основі цих даних складаються звіти, які обговорюються керівниками проектів зі своїми групами. Обговорення можуть проводитись щотижня, а при необхідності – щоденно [36,45].

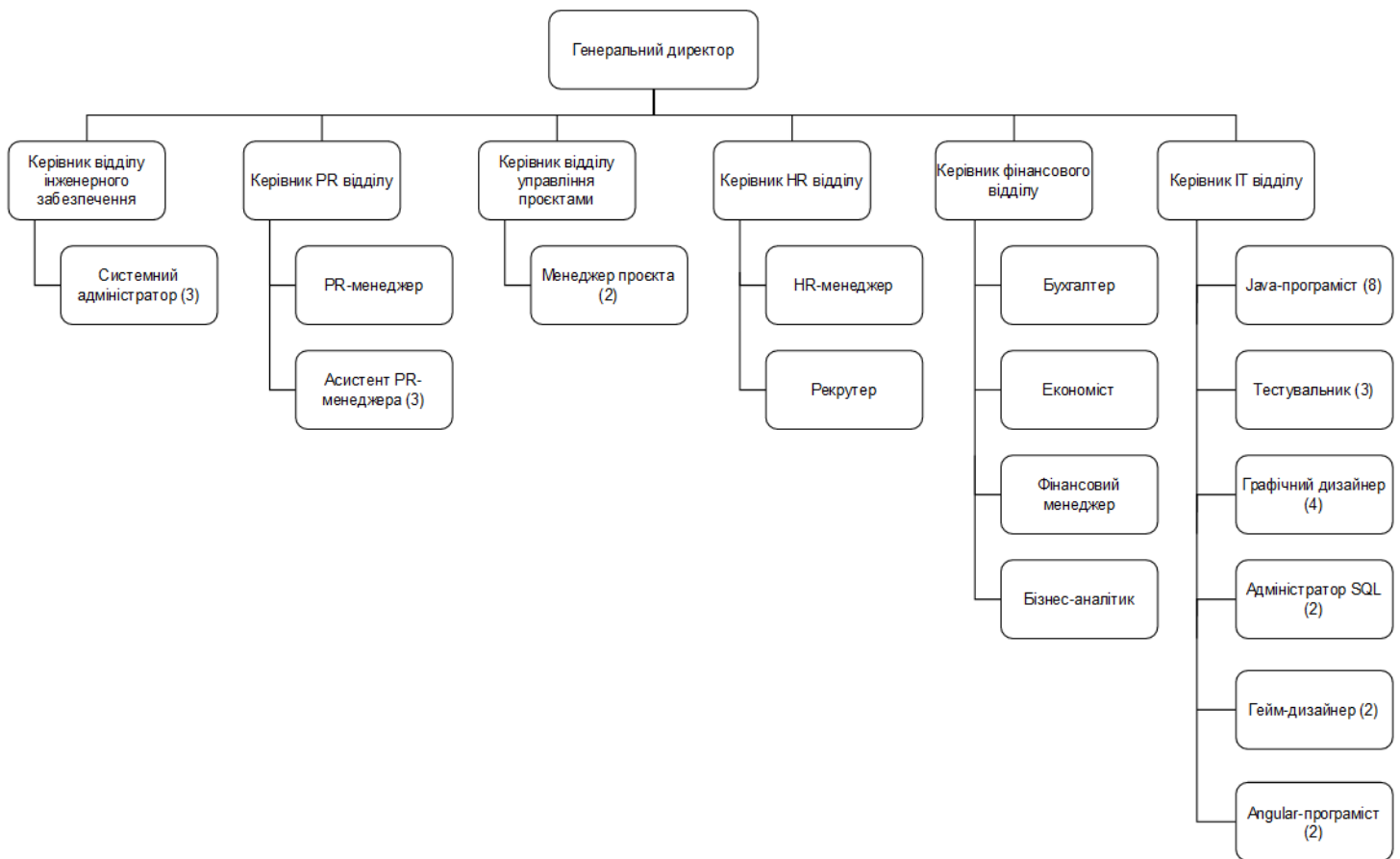


Рис.2.2. Організаційна структура компанії

До функцій керівника проекту (КП) відносяться наступні:

- надання консультаційної допомоги замовнику в розробці і реалізації проекту;
- вибір проектувальників і підрядчиків;
- складання планів робіт із проекту;

- оформлення і підписання актів, що підтверджують виконання робіт (етапів, підетапів, здачу об'єктів у дослідну і промислову експлуатації);
- складання матеріалів для висновку контрактів із замовниками і виконавцями (контрагентами) на проведення робіт із проекту;
- координація роботи всіх учасників проекту;
- постійний контроль за дотриманням термінів виконання робіт по проекту; науково-технічним рівнем і якістю розробок; витратами;
- приймання виконаних виконавцями і контрагентами робіт зі створення проекту;
- оформлення звітної документації на виконані роботи.

Керівник проекту зобов'язаний:

- домагатися від відділів, що беруть участь у проекті, виконання їхніх завдань по проекту;
- стежити за тим, щоб робота відділів велася відповідно до плану-графіка, кошторису витрат і специфікацій;
- вчасно виявляти труднощі, помилки, нестачу ресурсів, низьку якість робіт;
- вчасно вносити корективи в проект, якщо в цьому виникає необхідність;
- інформувати всіх зацікавлених осіб про хід роботи над проектом.

Однією з проблем матричних організаційних структур є перевантаження функціональних підрозділів. У випадку дисбалансу між обсягом робіт, що потрібно виконати для різних проектів, і можливостями у відповідних функціональних підрозділах виникають конфлікти між КП і керівниками підрозділів. У ряді випадків цю проблему можна вирішити за допомогою більш якісного планування робіт і ресурсів.

Розподіл відповідальності, функцій і повноважень з управління проектом [45,49].

Таблиця 2.2 Розподіл відповідальності

Роль	Функції
Менеджер проекту	Відповідальність за планування, виконання та завершення проекту. Контролює терміни виконання робіт. Представляє інтереси кінцевих користувачів і інших зацікавлених в продукті сторін.
Аналітик	Відповідає за розробку вимог до продукту, визначення функціональності та інтерфейсу користувача
Розробник front-end	Створення клієнтської частини програмного забезпечення (верстка, інтерфейс користувача).
Розробник back-end	Програмування і створення серверної частини програмного забезпечення.
Тестувальник	Займається тестуванням програмного забезпечення з метою виявлення помилок в його роботі і їх подальшого виправлення.
Дизайнер	Створює графічні інтерфейси, основне завдання яких – вирішувати задачі користувачів.

2.4 Календарне планування проекту

Календарне планування проекту відбувається з метою створення чіткого графіка виконання робіт проекту протягом певного періоду часу. Його основна мета полягає в тому, щоб забезпечити вчасне виконання всіх етапів проекту та досягнення поставлених цілей у відведений термін.

Календарне планування дозволяє узгодити роботу різних команд, планувати ресурси та знаходити кращі шляхи розподілу завдань для досягнення оптимальних результатів. Крім того, календарний план є основним інструментом для контролю за виконанням робіт та вирішення проблем, що виникають в процесі проекту. Завдяки календарному плануванню можна забезпечити ефективне виконання проекту та досягнення його мети вчасно і в межах бюджету [47].

План проекту був виконаний за допомогою програмного забезпечення Microsoft Project [46].

Загалом проект має тривалість 173 дні, поділяється на 4 великі етапи.

Режим задачі	Task Name	Длительнс	Начало	Окончание	Предшествен	Названия ресурсов	уба
	▲ Створення системи для шерінгу електросамокатів	173 дней	Вс 04.12.22	Вт 01.08.23			
	Початок	0 дней	Вс 04.12.22	Вс 04.12.22			
	▲ Розробка концепції	18 дней	Вс 04.12.22	Вт 27.12.22			
	▲ Визначення цілей проекту	5 дней	Вс 04.12.22	Чт 08.12.22			
	Побудова дерева цілей	5 дней	Вс 04.12.22	Чт 08.12.22	1	Менеджер проекту	
	▲ Побудова дерева проблем	6 дней	Пт 09.12.22	Пт 16.12.22			
	PEST аналіз	3 дней	Пт 09.12.22	Вт 13.12.22	4	Менеджер проекту	
	SWOT аналіз	3 дней	Ср 14.12.22	Пт 16.12.22	6	Менеджер проекту	
	▲ Визначення задач проекту	7 дней	Пн 19.12.22	Вт 27.12.22			
	Розбиття проекту на підзадачі	4 дней	Пн 19.12.22	Чт 22.12.22	7	Backend програміст;Frontend	
	Визначення термінів виконання підзадач	3 дней	Пт 23.12.22	Вт 27.12.22	9	Backend програміст;Frontend	
	▸ Процес підготовки	14 дней	Ср 28.12.22	Пн 16.01.23			
	▸ Процес виконання	133 дней	Вт 17.01.23	Чт 20.07.23			
	▸ Завершення	8 дней	Пт 21.07.23	Вт 01.08.23			
	Фініш	0 дней	Вт 01.08.23	Вт 01.08.23	70		

Рис. 2.3 Етап розробки концепції проекту

Режим задачі	Task Name	Длительнс	Начало	Окончание	Предшествен	Названия ресурсов	уба
	▲ Створення системи для шерінгу електросамокатів	173 дней	Вс 04.12.22	Вт 01.08.23			
	Початок	0 дней	Вс 04.12.22	Вс 04.12.22			
	▸ Розробка концепції	18 дней	Вс 04.12.22	Вт 27.12.22			
	▲ Процес підготовки	14 дней	Ср 28.12.22	Пн 16.01.23			
	▲ Постановка вимог	4 дней	Ср 28.12.22	Пн 02.01.23			
	Визначення функціональних вимог	2 дней	Ср 28.12.22	Чт 29.12.22	10	Менеджер проекту;Backend i	
	Визначення нефункціональних вимог	2 дней	Пт 30.12.22	Пн 02.01.23	13	Менеджер проекту	
	Розробка календарного плану	4 дней	Вт 03.01.23	Пт 06.01.23	14	Менеджер проекту	
	▸ Створення статуту проекту	6 дней	Пн 09.01.23	Пн 16.01.23			
	▸ Процес виконання	133 дней	Вт 17.01.23	Чт 20.07.23			
	▸ Завершення	8 дней	Пт 21.07.23	Вт 01.08.23			
	Фініш	0 дней	Вт 01.08.23	Вт 01.08.23	70		

Рис. 2.4 Етап процесу підготовки

Режим задачі	Task Name	Длительнс	Начало	Окончание	Предшественн	Названия ресурсов
	Створення системи для шерінгу електросамокатів	173 дней	Вс 04.12.22	Вт 01.08.23		
	Початок	0 дней	Вс 04.12.22	Вс 04.12.22		
	Розробка концепції	18 дней	Вс 04.12.22	Вт 27.12.22		
	Процес підготовки	14 дней	Ср 28.12.22	Пн 16.01.23		
	Процес виконання	133 дней	Вт 17.01.23	Чт 20.07.23		
	Розробка моделі	12 дней	Вт 17.01.23	Ср 01.02.23		
	Проектування	10 дней	Вт 17.01.23	Пн 30.01.23		
	Створення UML діаграм проекту	10 дней	Вт 17.01.23	Пн 30.01.23		
	Створення діаграми класів	3 дней	Вт 17.01.23	Чт 19.01.23		
	Виділення бізнес-об'єктів	2 дней	Вт 17.01.23	Ср 18.01.23	18	Backend програміст
	Визначення зв'язків	1 день	Чт 19.01.23	Чт 19.01.23	24	Backend програміст
	Діаграма компонентів	1 день	Пт 20.01.23	Пт 20.01.23	25	Backend програміст
	Діаграма об'єктів	1 день	Пн 23.01.23	Пн 23.01.23	26	Backend програміст
	Створення USE-CASE діаграми	2 дней	Вт 24.01.23	Ср 25.01.23		
	Виділення ролей	1 день	Вт 24.01.23	Вт 24.01.23	27	Backend програміст
	Визначення функцій	1 день	Ср 25.01.23	Ср 25.01.23	29	Backend програміст
	Діаграма діяльності	2 дней	Чт 26.01.23	Пт 27.01.23	30	Backend програміст
	Створення діаграми розгортання	1 день	Пн 30.01.23	Пн 30.01.23	31	Backend програміст
	ER-модель	2 дней	Вт 31.01.23	Ср 01.02.23	32	Backend програміст
	Формалізація бізнес-логіки	2 дней	Чт 02.02.23	Пт 03.02.23	33	Backend програміст
	Створення ядра програми	49 дней	Пн 06.02.23	Чт 13.04.23		
	Загальні сутності	15 дней	Пн 06.02.23	Пт 24.02.23	34	Backend програміст
	Запити до БД	24 дней	Пн 27.02.23	Чт 30.03.23		
	Створення	10 дней	Пн 27.02.23	Пт 10.03.23	36	Backend програміст
	Моніторинг	7 дней	Пн 13.03.23	Вт 21.03.23	38	Backend програміст
	Оптимізація	7 дней	Ср 22.03.23	Чт 30.03.23	39	Backend програміст
	Клієнтська логіка	5 дней	Пт 31.03.23	Чт 06.04.23	40	Backend програміст
	Адміністративна логіка	5 дней	Пт 07.04.23	Чт 13.04.23	41	Backend програміст
	Створення презентаційного шару	50 дней	Вт 17.01.23	Пн 27.03.23		
	Розробка дизайну	15 дней	Вт 17.01.23	Пн 06.02.23		Дизайнер

Рис. 2.5 Етап виконання проекту - перша частина

Режим задачі	Task Name	Длительнс	Начало	Окончание	Предшественн	Названия ресурсов
	Створення системи для шерінгу електросамокатів	173 дней	Вс 04.12.22	Вт 01.08.23		
	Початок	0 дней	Вс 04.12.22	Вс 04.12.22		
	Розробка концепції	18 дней	Вс 04.12.22	Вт 27.12.22		
	Процес підготовки	14 дней	Ср 28.12.22	Пн 16.01.23		
	Процес виконання	133 дней	Вт 17.01.23	Чт 20.07.23		
	Розробка моделі	12 дней	Вт 17.01.23	Ср 01.02.23		
	Формалізація бізнес-логіки	2 дней	Чт 02.02.23	Пт 03.02.23	33	Backend програміст
	Створення ядра програми	49 дней	Пн 06.02.23	Чт 13.04.23		
	Створення презентаційного шару	50 дней	Вт 17.01.23	Пн 27.03.23		
	Розробка дизайну	15 дней	Вт 17.01.23	Пн 06.02.23		Дизайнер
	Розробка контенту	13 дней	Вт 17.01.23	Чт 02.02.23		
	Текстовий контент	7 дней	Вт 17.01.23	Ср 25.01.23		Frontend Програміст
	Візуальний контент	6 дней	Чт 26.01.23	Чт 02.02.23	46	Frontend Програміст
	Верстка	35 дней	Вт 07.02.23	Пн 27.03.23		
	форумання HTML	15 дней	Вт 07.02.23	Пн 27.02.23	47;46;44	Frontend Програміст
	css	10 дней	Вт 28.02.23	Пн 13.03.23	49	Frontend Програміст
	javascript	10 дней	Вт 14.03.23	Пн 27.03.23	50	Frontend Програміст
	З'єднання серверної та клієнтської частин	17 дней	Пт 14.04.23	Пн 08.05.23		
	Організація контролерів	10 дней	Пт 14.04.23	Чт 27.04.23	42;51	Backend програміст;Frontend
	Приєм платежів	7 дней	Пт 28.04.23	Пн 08.05.23	53	Backend програміст;Frontend
	Автоматизація розгортання	3 дней	Вт 09.05.23	Чт 11.05.23		
	Налаштування СКВ	1 день	Вт 09.05.23	Вт 09.05.23	54	Backend програміст
	Налаштування Jenkins	2 дней	Ср 10.05.23	Чт 11.05.23	56	Backend програміст
	Тестування	40 дней	Пт 12.05.23	Чт 06.07.23		
	Визначення обмежень	3 дней	Пт 12.05.23	Вт 16.05.23	57	Тестувальник
	Написання тестів	16 дней	Ср 17.05.23	Ср 07.06.23	59	Тестувальник
	Тестування навантаженням	6 дней	Пт 12.05.23	Пт 19.05.23	57	Backend програміст
	Тестування функціональності	10 дней	Чт 08.06.23	Ср 21.06.23	60	Тестувальник
	Формування звіту по тестуванню	1 день	Чт 22.06.23	Чт 22.06.23	62	Тестувальник
	Статичний аналіз коду	1 день	Пн 22.05.23	Пн 22.05.23	61	Backend програміст
	Виправлення знайдених помилок	10 дней	Пт 23.06.23	Чт 06.07.23	63;64	Backend програміст;Frontend
	Додавання лендінгів	5 дней	Пт 07.07.23	Чт 13.07.23	65	Frontend Програміст
	Внутрішня оптимізація	5 дней	Пт 14.07.23	Чт 20.07.23	66	Frontend Програміст
	Завершення	8 дней	Пт 21.07.23	Вт 01.08.23		
	Фініш	0 дней	Вт 01.08.23	Вт 01.08.23	70	

Рис. 2.6 Етап виконання проекту - друга частина

Режим задачі	Task Name	Длительнс	Начало	Окончание	Предшествент	Названия ресурсов
	▲ Створення системи для шерінгу електросамокатів	173 дней	Вс 04.12.22	Вт 01.08.23		
	Початок	0 дней	Вс 04.12.22	Вс 04.12.22		
	▸ Розробка концепції	18 дней	Вс 04.12.22	Вт 27.12.22		
	▸ Процес підготовки	14 дней	Ср 28.12.22	Пн 16.01.23		
	▸ Процес виконання	133 дней	Вт 17.01.23	Чт 20.07.23		
	▲ Завершення	8 дней	Пт 21.07.23	Вт 01.08.23		
	Аудит	7 дней	Пт 21.07.23	Пн 31.07.23	67	Менеджер проекту
	Закриття	1 день	Вт 01.08.23	Вт 01.08.23	69	Менеджер проекту
	Фініш	0 дней	Вт 01.08.23	Вт 01.08.23	70	

Рис. 2.7 Етап завершення проекту

Старт проекту заплановано на 04.12.22

Базовий план було зафіксовано, тож на графіку Ганта його відмічено сірим кольором [48].

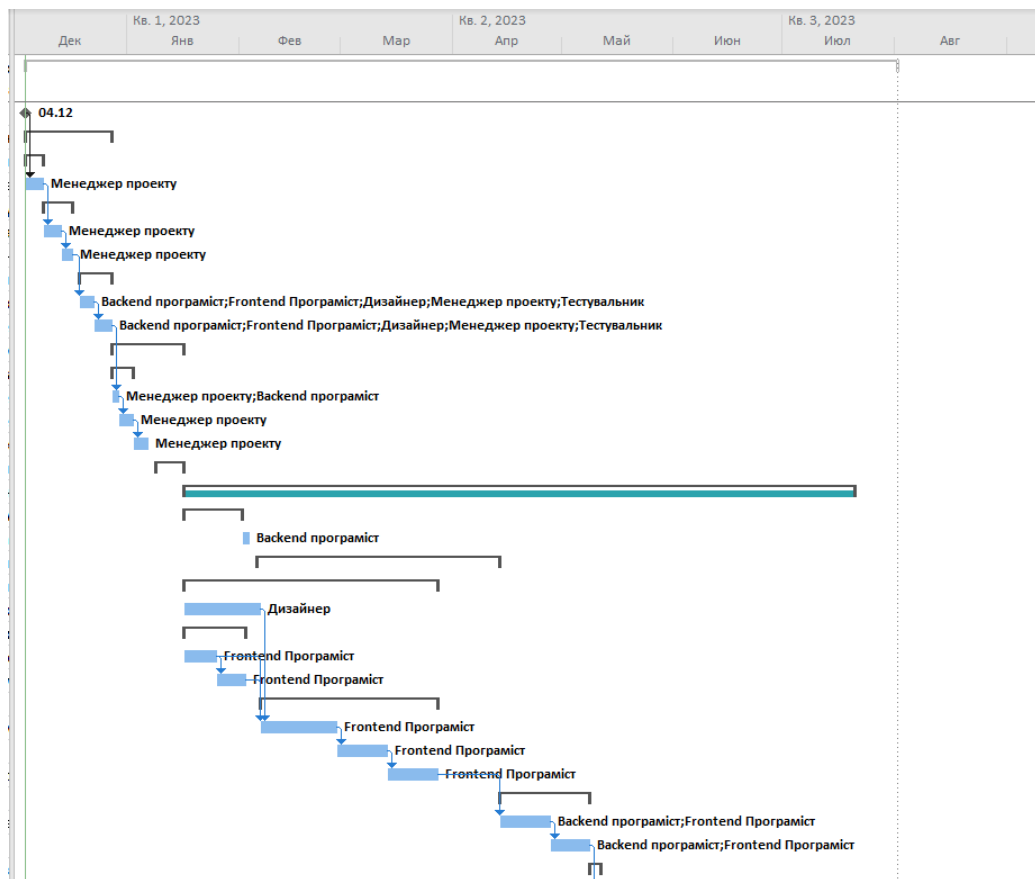


Рис. 2.8 Фрагмент календарного плану

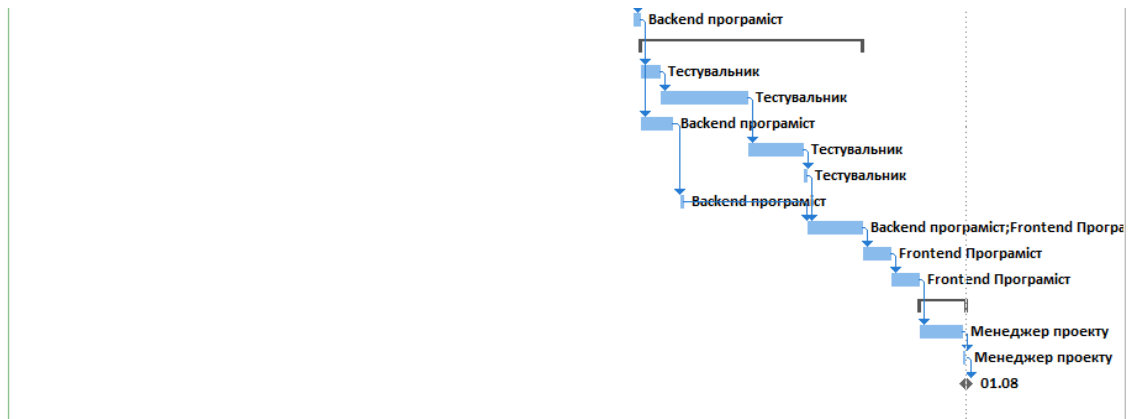


Рис. 2.9. Фрагмент календарного плану. Шостий крок проекту

Кінець проекту за результатами планування відбудеться – 01.08.2023, тривалість проекту складає 173 днів. Розглянемо фрагмент сітвової діаграми проекту (рис. 2.10) [49].

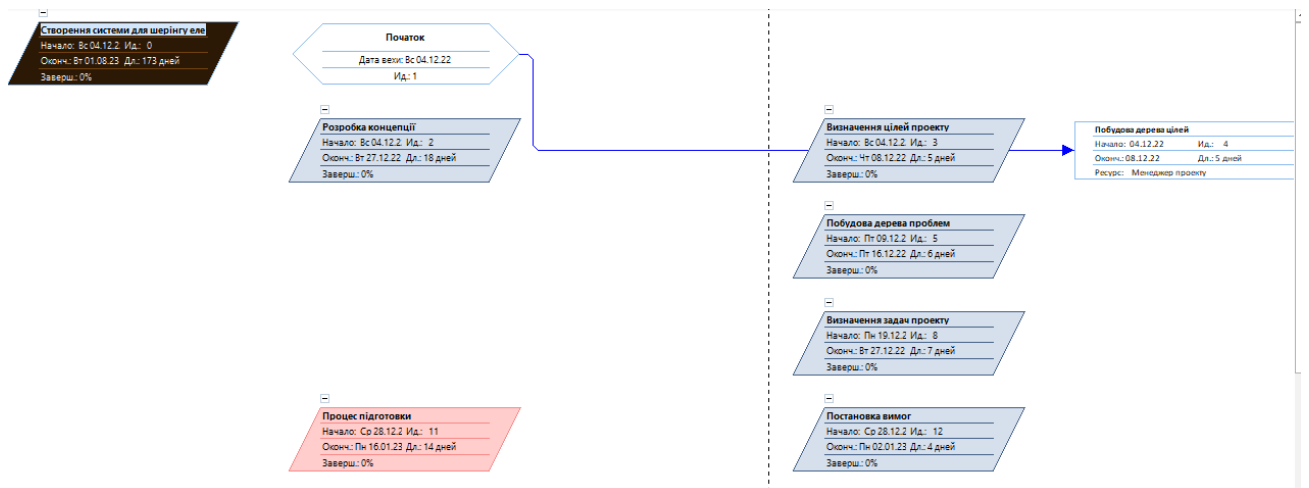


Рис. 2.10. Фрагмент сітвової діаграми проекту

Сітвova діаграма показує послідовність розробки робіт по проекту і їх залежність одна від одної. Розглянемо календар проекту на грудень 2022 року (рис. 2.11).

27	28	29	30	01 Дек	02	03
04	05	06	07	08	09	10
Створення системи для шерінгу електросамокатів; 173 днів						
Початок	Побудова дерева цілей; 5 днів				PEST аналіз; 3 днів	
11	12	13	14	15	16	17
Створення системи для шерінгу електросамокатів; 173 днів						
PEST аналіз; 3 днів			SWOT аналіз; 3 днів			
18	19	20	21	22	23	24
Створення системи для шерінгу електросамокатів; 173 днів						
Розбиття проекту на підзадачі; 4 днів					Визначення термінів виконання підзадач; 3 днів	
25	26	27	28	29	30	31
Створення системи для шерінгу електросамокатів; 173 днів						
Визначення термінів виконання підзадач; 3 днів			Визначення функціональних вимог; 2 днів		Визначення нефункціональних вимог; 2 днів	
01 Янв	02	03	04	05	06	07

Рис. 2.11. Календар проекту на грудень 2022 року

2.5 Планування ресурсів проекту

Основною метою планування ресурсів проекту є забезпечення достатньої кількості ресурсів, таких як людські, фінансові та матеріальні, для успішного виконання проекту в межах встановлених обмежень щодо часу, бюджету та інших ресурсів. Це включає оцінку ресурсів, необхідних для виконання кожного завдання проекту, розподіл ресурсів між завданнями та ресурсів між членами команди проекту, а також контроль за використанням ресурсів під час виконання проекту. Планування ресурсів є важливим етапом управління проектом, оскільки допомагає забезпечити ефективне використання ресурсів та зменшити ризики відхилень від плану через нестачу ресурсів [49,50].

Команда проекту з шерінгу самокатів складається з наступних ролей та виконавців:

1. Менеджер проекту - 1 особа: відповідає за планування, контроль виконання проекту та спілкування з клієнтом.

2. Аналітик - 1 особи: відповідає за розробку вимог до продукту, визначення функціональності та інтерфейсу користувача.
3. Дизайнер - 1 особи: відповідає за розробку дизайну інтерфейсу користувача та візуалізацію даних.
4. Розробник (Frontend) - 2 особи: відповідає за розробку фронтенду, включаючи HTML, CSS, JavaScript та React.
5. Розробник (Backend) - 2 особи: відповідає за розробку бекенду, включаючи базу даних, API, логіку бізнесу та безпеку.
6. Тестувальник – 1 особа: відповідає за тестування вимог, тестування функціональних та нефункціональних особливостей продукту, написання тестової документації.

Отже, загальна кількість людей в команді складається з 8 осіб.

Для реалізації проекту використовується Agile-методологія, Scrum.

Основні причини вибору цієї методології:

1. Гнучкість та адаптивність. Agile-методології дозволяють швидко реагувати на зміни вимог та умов ринку і корегувати процес розробки відповідно до цих змін.
2. Висока ефективність. Кожен етап розробки включає в себе повторення циклу планування-розробки-тестування, що забезпечує високу якість продукту та покращення процесів розробки з кожним циклом.
3. Командна співпраця. Agile-методології стимулюють співпрацю та комунікацію між усіма членами команди проєкту, що дозволяє досягати більшої ефективності.

Матеріальні ресурси для даного проєкту це ноутбуки і сервери.

Для даного проєкту був обраний тип проєктного фінансування у формі обмеженого регресу. Система проєктного фінансування була обрана

тому, що в рамках цієї економічно більш ефективної системи можна використовувати різні форми кредитування, залучати різних кредиторів, передбачати вигідні способи розміщення інвестиційних ресурсів. Форма обмеженого регресу на позичальника, була обрана, через те, що в даній формі відповідальність за результат несуть всі учасники проекту, що призводить до більшої продуктивності роботи.

Ризиками проектного фінансування представленого проекту є:

1. Економічні. Нездатність покрити всі проектні витрати прибутком від проекту;
2. Правові. Можливі проблеми зі створенням на платформі контенту, що порушує авторські права.
3. Непередбачувані. В дану групу ризиків крім непередбачуваних обставин (наприклад проблеми з виходом на ринок) також належать політичні ризики.

На основі вище наведеної інформації, сформовано таблицю [51,52].

Табл 2.3 Фінансування проекту

Назва витрат	Ціна за одиницю (грн.)	Кількість
Розробник back-end	60000	2
Розробник front-end	50000	2
Менеджер проекту	70000	1
QA тестувальник	30000	1
Дизайнер	30000	1
Аналітик	35000	1
Загальна вартість:	2 450 000	
Купівля обладнання:		
Сервер	40000	
Ноутбуки	279 000	
Загальна вартість:	319 000	

Продовження Табл 2.3 Фінансування проєкту

Додаткові витрати:		
Маркетингова компанія	375 000	
Інше (оренда приміщень)	40000	1
Загальна вартість:	535 000	
ВСЬОГО:	3 338 000	

РОЗДІЛ 3. ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ УПРАВЛІННЯ ПРОЄКТОМ

3.1 Управління ризиками проєкту

Управління ризиками проєкту має на меті ідентифікувати, аналізувати, оцінювати, мінімізувати та контролювати ризики, які можуть вплинути на успішне завершення проєкту. Це важлива складова будь-якого проєкту, оскільки допомагає знизити можливі збитки, запобігти проблемам та збільшити шанси на успішне завершення проєкту. Управління ризиками включає в себе кілька кроків:

1. Ідентифікація ризиків - визначення потенційних ризиків, які можуть виникнути під час виконання проєкту.
2. Аналіз ризиків - оцінка ймовірності та впливу ризиків на проєкт, визначення того, наскільки серйозним може бути вплив кожного ризику.
3. Оцінка ризиків - визначення потенційних наслідків кожного ризику, розробка стратегій їх управління та визначення пріоритету ризиків.
4. Мінімізація ризиків - прийняття заходів для зменшення ймовірності виникнення ризиків та зниження їх впливу на проєкт.
5. Контроль ризиків - встановлення механізмів моніторингу та контролю за ризиками, що допомагають вчасно виявляти проблеми та вживати заходів для їх вирішення.

Управління ризиками дозволяє забезпечити ефективне виконання проєкту та зменшити можливість виникнення небажаних наслідків, тому є невід'ємною частиною процесу управління проєктом [15].

Результати ідентифікації ризиків описано у таблиці 3.1.

Таблиця 3.1

Тип ризику	Ризикова подія	Сила впливу	Керованість
Програмні ризики	Не дотримання контролю версій	Висока	Висока
	Закінчення ліцензій на сфері розробки	Висока	Середня
	Проблеми переносу додатку на мультиплатформу	Середня	Висока
	Нема стандарту на середовищами розробки для співпрацівників	Низька	Висока
Апаратні ризики	Проблема з передачею GPS сигналу	Висока	Низька
	Падіння серверів через багату кількість користувачів	Висока	Середня
	Енергозатратний додаток	Середня	Середня
	Низький рівень інтернет трафіку	Висока	Низька
Внутрішні ризики проекту (команда)	Недостатня кваліфікація співробітників	Висока	Висока
	Не дотримання документації проекту	Середня	Висока
	Поганий менеджмент проекту	Висока	Висока
	Низька мотивація співробітників	Середня	Середня
Зовнішні (оточення)	Зміна податкових правил України	Низька	Низька
	Валютна нестабільність	Середня	Низька
	Поява значного конкуренту	Висока	Низька
	Змінення правил дистриб'ютера додатку	Висока	Низька
Форс мажори	Військовий стан в країні	Висока	Низька
	Локдаун через вірус	Висока	Низька
	Аварія на електростанції	Висока	Низька
	Зміна потреб ринку	Середня	Низька
Ризики кібербезпеки	Фішингові листи на пошту	Висока	Висока
	Використання небезпечного ПЗ	Висока	Середня
	Сніффінг трафіку в офісі розробки	Висока	Середня
	Злив інформації про ПЗ від співпрацівників	Середня	Середня

Після визначення можливих ризикових подій відбувається оцінка ризиків. На меті цього є виділення найбільш небезпечних ризиків з точки зору втрат і визначення на чому необхідно зосередитись.

Для оцінки ризикової події виставимо оцінки за шкалою (додаток А) за ознаками затримки у часі, частоти, фінансових втрат та ймовірності за проєкт. Результати оцінювання ризикових подій описано у таблиці 3.3 [14].

Таблиця 3.3

Ризикова подія	Затримки у часі		Фінансові втрати		Ймовірність		Частота (за проект)		Важливість ризику (компл. показник)
	Якіс. оц.	Кільк. оц.	Якіс. оц.	Кільк. оц.	Якіс. оц.	Кільк. оц.	Якіс. оц.	Кільк. оц.	Якіс.оц.
	Війна в країні	вс	8	сс	5	к	10	нн	1
Локдаун через вірус	вн	7	вн	7	св	6	сн	4	42
Поганий менеджмент проекту	вв	9	вв	9	сн	4	сн	4	36
Поява значного конкуренту	нс	2	св	6	сс	5	нн	1	30
Сніффінг трафіку в офісі розробки	сс	5	сн	4	вн	7	сс	5	28
Злив інформації про ПЗ від співпрацівників	вс	8	вв	9	нв	3	нс	2	27
Не дотримання контролю версій	нв	3	вн	7	нв	3	нн	1	21
Використання небезпечного ПЗ	нс	2	нв	3	вн	7	нс	2	21
Зміна потреб ринку	вс	8	вв	9	нс	2	нс	2	18
Не дотримання документації проекту	вн	7	вн	7	нс	2	нв	3	14
Падіння серверів через велику кількість користувачів	вн	7	вн	7	нс	2	нс	2	14
Проблема з передачею GPS сигналу	вн	7	вн	7	нс	2	нс	2	14
Енергозатратний додаток	сс	5	сн	4	нв	3	нс	2	12
Зміна податкових правил України	нс	2	сн	4	нв	3	нн	1	12
Проблеми переносу додатку на мультплатформу	св	6	сс	5	нс	2	нн	1	10
Недостатня кваліфікація співробітників	вв	9	сс	5	нс	2	нс	2	10
Закінчення ліцензій на сфери розробки	к	#	к	#	нн	1	нн	1	10

Продовження Таблиці 3.3

Низька мотивація співробітників	сс	5	нв	3	нв	3	нв	3	9
Валютна нестабільність	св	6	вн	7	нн	1	нн	1	7
Низький рівень інтернет трафіку	вн	7	вн	7	нн	1	нн	1	7
Складність підтримки програмного продукту	св	6	нв	3	нс	2	сс	5	6
Аварія на електростанції	вн	7	св	6	нн	1	нс	2	6
Нема стандарту на середовищами розробки для співпрацівників	вн	7	нс	2	нв	3	нс	2	6
Фішингові листи на пошту	нв	3	нс	2	нн	1	нн	1	2

Після оцінки всіх можливих ризиків визначаємо найнебезпечніші ризики та розроблюємо протиризові міри. Розробимо міри на чотирьох стадіях: профілактика ризику, ранній ознаці симптому, при симптомі ризику та при проблемі ризику. Протиризові міри описані у таблиці 3.4 [15].

Таблиця 3.4

Ризикова подія	ПРЗ 1	Симптом (рання ознака)	ПРЗ 2	ПРЗ 3
	профілактика		при симптомі	при проблемі
Війна в країні	Заохочувати працівників виїхати з небезпечних регіонів. Допомога переїзду. Підготовка компанії до роботи у віддаленому режимі. Можливість виїхати за кордон.	Інформаційна агітація сусідніх стран. Збір військової сили на кордоні.	Проаналізувати критичні процеси та ключових працівників, увага на збереження матеріальних і нематеріальних ресурсів	Реорганізація структури і процесів максимальної кількості працівників із виконанням крит. робіт. План робіт із обмеж. персоналом. Перехід на закордон. сервери. Перехід на стабільні валюти.

Продовження Таблиці 3.4

Локдаун через вірус	Проводити інформаційну обізнаність про вірус та як вберегтись від нього. Зменшення кількості працівників в офісі. Регулярні тестування на вірус.	Збільшення кількості захворівших по світу, в Україні.	Проаналізувати можливість переходу на дистанційну роботу. Проаналізувати можливі кадрові тимчасові заміни.	Реорганізація структури і процесів, ознайомлення максимальної кількості працівників із виконанням критичних робіт. План робіт із обмеженою кількістю працівників.
Поганий менеджмент проекту	Підвищення кваліфікації ПМ.	Невідповідність факту до плану виконання робіт.	Реорганізувати обов'язки в команді згідно кваліфікаціям та слідкувати за їх виконанням відносно часового плану проекту.	Переоцінка ресурсів, аналіз можливостей команди при поточних ресурсах. Реорганізація роботи.
Поява значного конкуренту	Розповсюдження бренду. Цікава реклама. Позитивний образ компанії.	Зниження попиту на послуги компанії.	Аналіз можливих шляхів до збільшення частки ринку.	Запуск рекламної кампанії для поширення обізнаності про проект. Розширення комунікації з клієнтами в соц. мережах.
Сніффінг трафіку в офісі розробки	Тренінги для персоналу з кібербезпеки. Оновлювати антивірусні програми своєчасно. Проводити тестування безпеки.	Отримання зловмисником доступу до інформації.	Повідомити службу безпеки про симптоми і якнайшвидше визначити чи була хакерська атака.	Оновити програмне забезпечення. Виявити зловмисника і недоліки в системі. Оцінити втрати.

Протиризикові заходи були розроблені для сніффінгу трафіку, локдауну через вірус, військові дії та поганий менеджмент проекту, по

причині того, що саме ці ризики мають найбільший негативний вплив на реалізацію проєкту [17].

3.2 Управління зацікавленими сторонами проєкту

Управління зацікавленими сторонами проєкту (англ. stakeholder management) має на меті ідентифікацію, аналіз та взаємодію з усіма зацікавленими сторонами проєкту з метою забезпечення успішності проєкту та задоволення їх потреб. Зацікавлені сторони (стейкхолдери) проєкту можуть бути різних видів: замовник проєкту, співробітники, клієнти, конкуренти, державні органи, громадські організації тощо [10].

Основна мета управління зацікавленими сторонами полягає у виявленні потреб та очікувань стейкхолдерів проєкту, а також управлінні їх впливом на проєкт. Для цього зазвичай проводяться такі кроки:

1. Ідентифікація зацікавлених сторін проєкту.
2. Аналіз потреб та очікувань стейкхолдерів.
3. Визначення важливості стейкхолдерів для проєкту.
4. Розробка стратегій взаємодії зі стейкхолдерами.
5. Реалізація стратегій взаємодії та моніторинг їх ефективності.
6. Коригування стратегій взаємодії у разі зміни умов та потреб стейкхолдерів.

Управління зацікавленими сторонами є важливою складовою процесу управління проєктом, оскільки дозволяє врахувати інтереси всіх стейкхолдерів та забезпечити успішну реалізацію проєкту.

Таблиця 3.5

Пріоритет по зацікавленим сторонам	Зацікавлені сторони	Вимоги до якості проекту	Пріоритет по модулям проекту	Модуль проекту	Вимоги до якості продукту
1	Користувачі	Додаток з інтерфейсом картою	1	Модуль транспорту	Можливість оренди самоката
1	Користувачі	Додаток з інтерфейсом картою	1	Модуль транспорту	Можливість паркувати самокат
1	Користувачі	Додаток з інтерфейсом картою	1	Модуль транспорту	Можливість перегляду стану самоката
1	Користувачі	Інтуїтивно зрозумілий інтерфейс	2	Особистий акаунт	Можливість перегляду внесених власних даних
1	Користувачі	Інтуїтивно зрозумілий інтерфейс	2	Особистий акаунт	Можливість перегляду історії поїздок
1	Користувачі	Інтуїтивно зрозумілий інтерфейс	2	Особистий акаунт	Внесення змін до профілю
1	Користувачі	Інтуїтивно зрозумілий інтерфейс	2	Особистий акаунт	Внешення змін до аватарки профілю

Продовження таблиці 3.5

1	Користувачі	Інтуїтивно зрозумілий інтерфейс	2	Особистий акаунт	Внесення змін до персональної інформації
1	Користувачі	Швидкодія	1	Модуль транспорту	Підключення менше ніж через 2 хв.
1	Користувачі	Наявність соціальних фіч	8	Соціальний модуль	Підтримка користувача у чаті
1	Користувачі	Наявність соціальних фіч	8	Соціальний модуль	Окремий дашборд з ачивментами
1	Користувачі	Наявність соціальних фіч	8	Соціальний модуль	Можливість поділитись ачивментами
1	Користувачі	Рекомендації	9	Наявність рекомендацій та пропозицій	Аналіз поїздок
2	Постачальники самокатів	Наявність аналізайційних інструментів	7	Модуль звітності та візуалізації статистики користувачів	Експорт в Kibana
2	Постачальники самокатів	Наявність аналізайційних інструментів	7	Модуль звітності та візуалізації статистики користувачів	Можливість кастомізувати візуалізації статистики

Продовження таблиці 3.5

2	Постачальники самокатів	Особисті пропозиції	9	Наявність рекомендацій та пропозицій	Можливість проведення сезонних акцій та знижок
2	Постачальники самокатів	Наявність інструменту для зворотнього зв'язку	6	Модуль повідомлення про помилку та відгуків	Імплементация баг-репортів
3	Інтернет-провайдри	Забезпечення безпеки з'єднання	3	Модуль безпеки та швидкодії	Захищений протокол передачі даних SSL
3	Інтернет-провайдри	Забезпечення безпеки з'єднання	4	Модуль безпеки та швидкодії	Відповідність вимогам 384kb/s, 2% Packet loss, 100ms latency
3	Інтернет-провайдри	Забезпечення безпеки з'єднання	4	Модуль безпеки та швидкодії	Підтримка 2000 користувачів одночасно
3	Інтернет-провайдри	Наявність інструменту для звістності	7	Модуль звістності та візуалізації статистики користувачів	Експорт в Kibana
3	Інтернет-провайдри	Наявність інструменту для звістності	7	Модуль звістності та візуалізації статистики користувачів	Можливість кастомізувати візуалізації статистики

Продовження таблиці 3.5

4	Команда проекту	Реалізація проекту в строк	5	План-графік виконання проекту	Затвердження плану-графіку проекту та відстеження виконання за планом
4	Команда проекту	Успішність реалізації проекту	5	План-графік виконання проекту	Виконання проекту за планом-графіком
4	Команда проекту	Чітко поставлені задачі команді проекту	5	План-графік виконання проекту	Призначення відповідальною та надіслані повідомлення про терміни виконання задач за проектом

РОЗДІЛ 4. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПРОЄКТУ

4.1 Опис структурної схеми інформаційних потоків в системі

Діаграми потоку інформації або діаграми потоку даних — це візуальні методи, які використовуються для демонстрації того, як інформація або дані переміщуються системою.

Інформація може змінюватися під час переходу від входу до виходу системи. Ці діаграми корисні, щоб показати, як користувачі можуть взаємодіяти з продуктом.

Діаграми інформаційних потоків можна розділити на рівні, які показують прогресивний потік і підвищену деталізацію. Діаграми інформаційних потоків є цінними інструментами, оскільки вони можуть описати системний процес, не покладаючись на деталі, пов'язані з комп'ютерними системами.

Користувач взаємодіє з клієнтською програмою, яка надсилає запити на серверну частину програми. Серверна частина обробляє запити і зберігає інформацію про клієнтів і платежі в базу даних. Він також обробляє запити на обробку замовлень на ремонт і відкликання, зберігаючи інформацію про них у базі даних [53].

Серверна частина також обробляє запити на інформацію про місцезнаходження скутерів і їх заряд, отримуючи цю інформацію від спеціальних датчиків, встановлених на скутерах, і зберігаючи в базі даних.

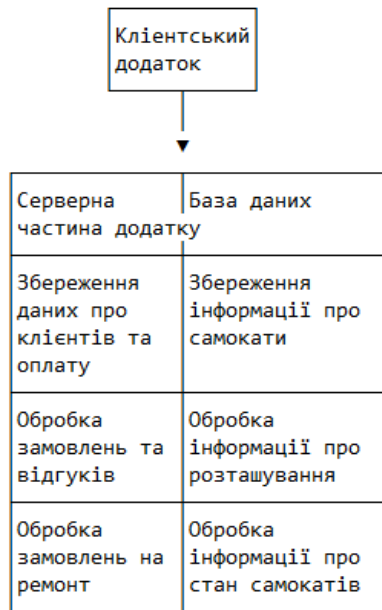


Рис. 4.1 Діаграма інформаційного потоку

4.2 Структура бази даних

Концептуальна модель бази даних (БД) створюється з метою описати взаємозв'язки між сутностями та їх атрибутами, які є ключовими для організації даних в БД. Основною метою створення концептуальної моделі БД є визначення та опис структури даних для певної сфери діяльності організації, що дозволяє розробити більш точну та ефективну структуру БД. Така модель може слугувати основою для подальшої розробки фізичної моделі БД та реалізації самої БД [54].

База даних створена для уявної компанії, яка займається задачею в оренду електросамокатів. Компанія у своєму розпорядженні має міні-парк самокатів, які розташовані на станціях оренди. На кожній станції присутні працівники, які займаються обробкою замовлень. Для спрощення дизайну бази даних була обрана опція переміщення лише між точками видачі (тобто орендар не може залишити самокат у випадковому місці, лише зі станції видачі на іншу станцію). Оплата відбувається за хвилину

користування + ціна розблокування. Оплата можлива як картою, так і готівкою.

Були обрані наступні сутності: самокат (scooter), орендар (customer), працівник (employee), станція оренди (rental_station), категорія скутера (category), а також сутність подорож (trip), яка характеризує зв'язок many-to-many[53].

Характеристика зв'язків.

One-to-many: *rental_station -> employee*

Many-to-many: *(customer, scooter, employee) - (trip)*, *scooter - category_scooter - category*.

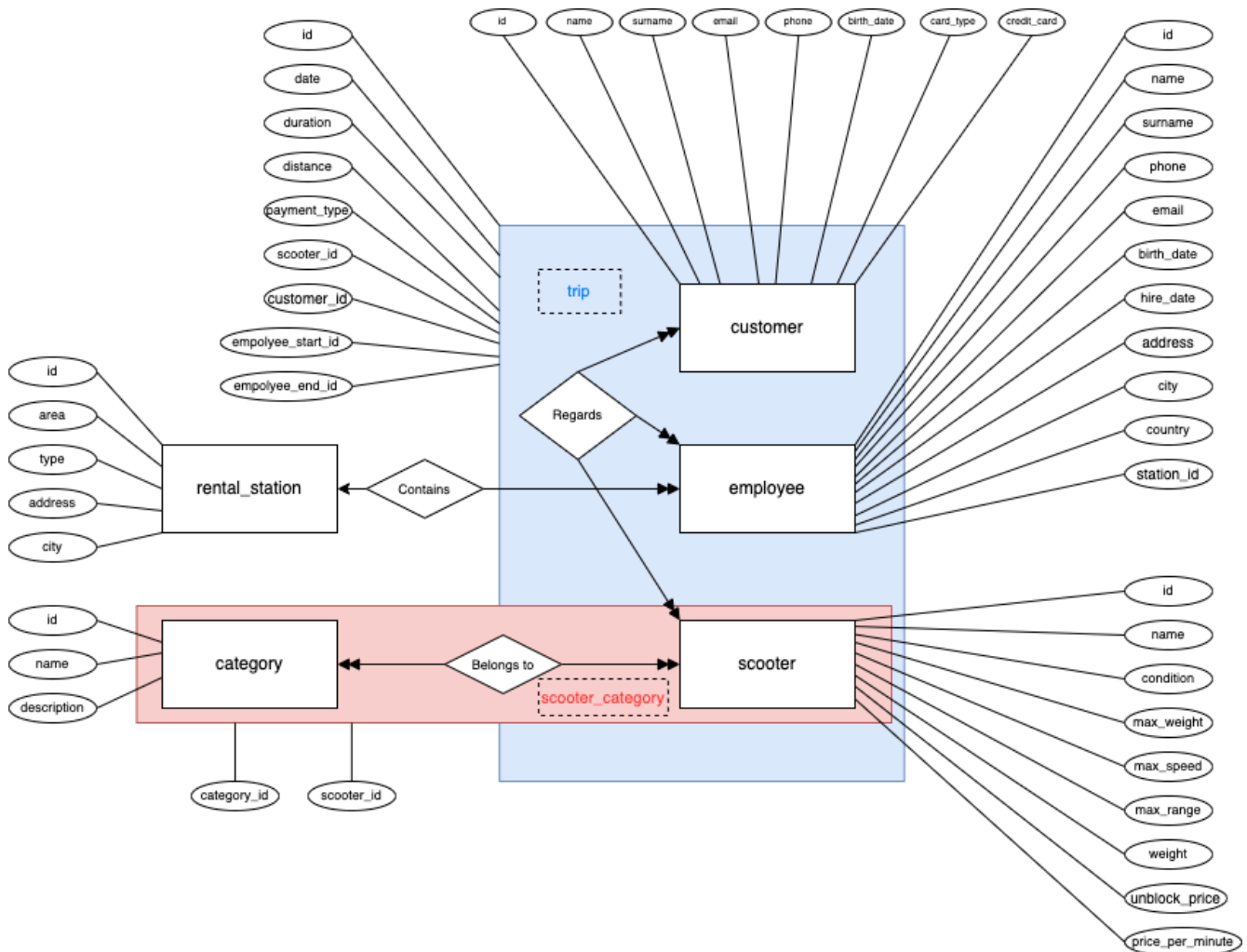


Рис. 4.2. Концептуальна модель бази даних

Створення логічної моделі бази даних (ЛМБД) проводиться з метою опису структури даних та взаємозв'язків між ними на рівні концепційної моделі, але з урахуванням специфіки конкретної СУБД, в якій буде реалізовуватись база даних. ЛМБД дає можливість узагальнено описати схему бази даних та відобразити належність атрибутів до сутностей, забезпечує зрозумілість структури бази даних для всіх учасників проекту, спрощує розробку та збереження схеми бази даних. Крім того, створення ЛМБД дозволяє провести аналіз правильності та повноти опису даних та взаємозв'язків між ними, що є важливим для подальшої роботи з базою даних та її розвитку [55].

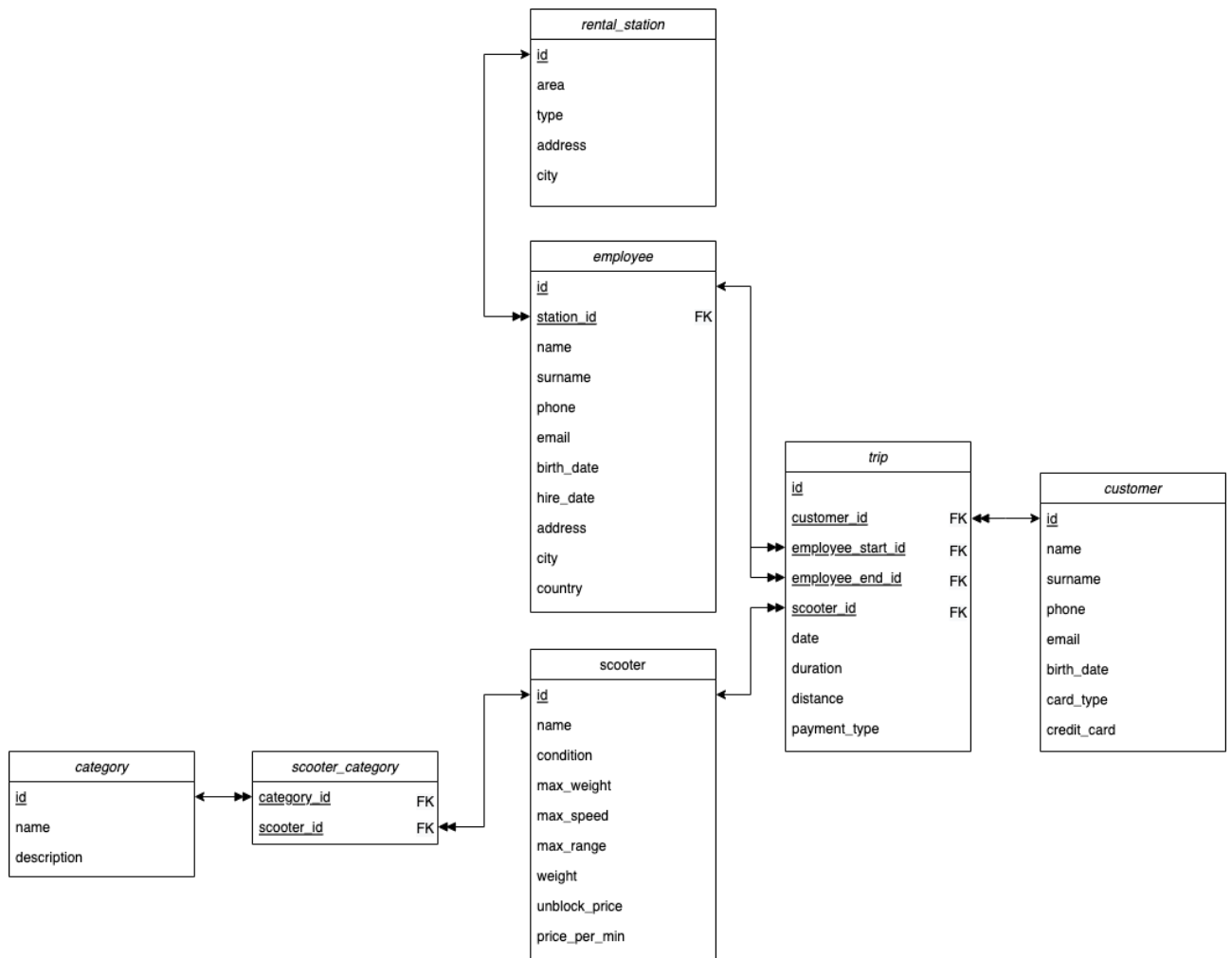


Рис. 4.3. Логічна модель бази даних

Фізична модель бази даних створюється з метою опису фізичної реалізації бази даних. Вона описує, як таблиці, стовпці та інші об'єкти бази

даних будуть зберігатися на диску, які індекси будуть створені та як будуть здійснюватися звернення до даних. Фізична модель бази даних дозволяє зменшити час на звернення до бази даних, збільшити її продуктивність та зменшити розмір збереженого на диску обсягу даних.

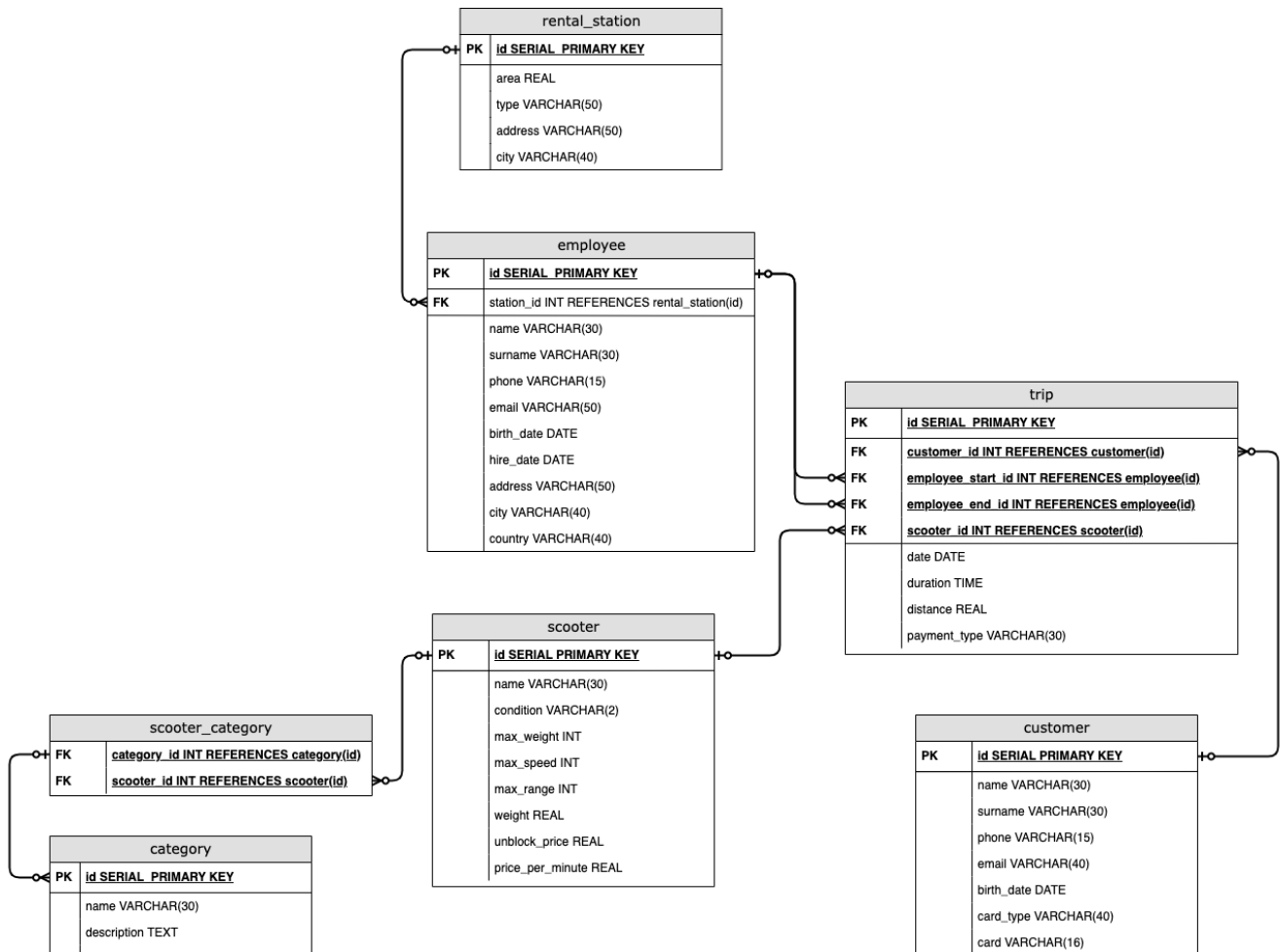


Рис. 4.4. Фізична модель бази даних

4.3 Опис структури та модулів програмного забезпечення проєкту

Основні модулі системи:

1. Модуль «Батарея» - цей модуль перевіряє правильності відображення рівня заряду батареї на карті та у додатку, коректність показання інформації про залишковий час роботи на одному заряді, а правильності підрахунку кількості кілометрів, які можна проїхати на

- одному заряді батареї, наявність повідомлення в додатку про низький рівень заряду батареї.
2. Модуль «Локація» - цей модуль перевіряє правильність відображення поточного місцезнаходження електросамоката на карті, коректність роботи системи знайдення ближчого доступного електросамоката для клієнта, можливість відстеження маршруту руху електросамоката на карті.
 3. Модуль «Назва» - перевірка правильності відображення назви електросамоката в додатку та на карті, перевірка коректності роботи системи генерації унікальної назви для кожного електросамоката.
 4. Модуль «Кількість»- перевірка коректності відображення кількості доступних електросамокатів на карті та в додатку, перевірка коректності роботи системи оновлення кількості доступних електросамокатів.
 5. Модуль «Дистанція» - перевірка коректності підрахунку відстані між двома точками на карті, перевірка коректності розрахунку вартості оренди електросамоката залежно від дистанції між початковою та кінцевою точками.
 6. Модуль «Пройдений шлях» - цей модуль перевіряє коректність підрахунку відстані, пройденої користувачем.
 7. Модуль «підтримки платежів» - цей модуль дозволяє користувачам здійснювати платежі за користування електросамокатами. Він повинен мати інтеграцію з різними платіжними системами та забезпечувати безпечність транзакцій.
 8. Модуль «управління користувачами» - цей модуль дозволяє адміністраторам відстежувати та керувати активністю користувачів,

у тому числі заблокувати доступ до сервісу для користувачів, які порушують правила користування.

9. Модуль «безпеки» - цей модуль включає систему відеоспостереження та інші заходи безпеки, щоб забезпечити безпечність користувачів та зменшити ризик втрати електросамокатів. Також він може містити систему автоматичної блокування електросамокатів у випадку виявлення незвичайної активності.

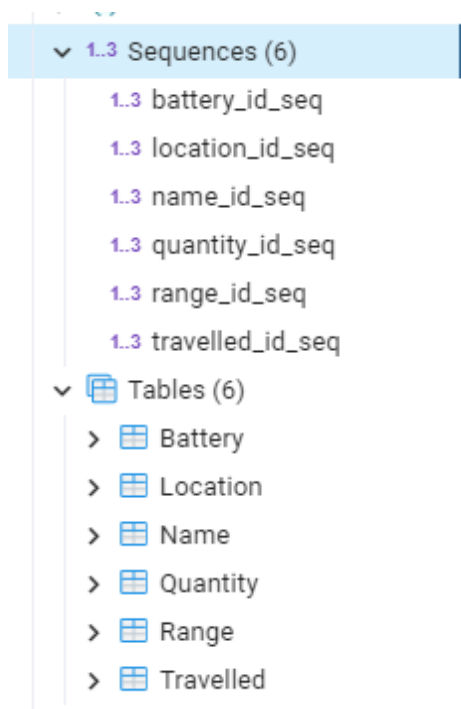


Рис. 4.5. Структура проекту

ER-діаграма (Entity-Relationship Diagram) - це модель даних, що використовується для візуалізації залежностей між різними сутностями (таблицями) в базі даних. Головна мета ER-діаграми - це демонстрація взаємозв'язків між таблицями та їх атрибутами.

ER-діаграма складається з сутностей (entity), зв'язків між цими сутностями (relationship) та атрибутів (attributes), які описують властивості цих сутностей.

Сутність - це об'єкт або поняття, яке відображається в таблиці бази даних, наприклад, користувач, продукт, замовлення тощо. Атрибут - це характеристика цих сутностей, наприклад, ім'я, номер телефону, вартість тощо.

Зв'язок між сутностями показує взаємозв'язок між двома таблицями, наприклад, зв'язок "один до багатьох" (one-to-many), "багато до багатьох" (many-to-many), "один до одного" (one-to-one) та інші.

ER-діаграма дозволяє візуалізувати залежності між таблицями та допомагає розробникам створювати більш структуровані бази даних. Крім того, це дає змогу краще зрозуміти взаємозв'язки між різними сутностями, спростити процес проектування та реалізації бази даних [49].

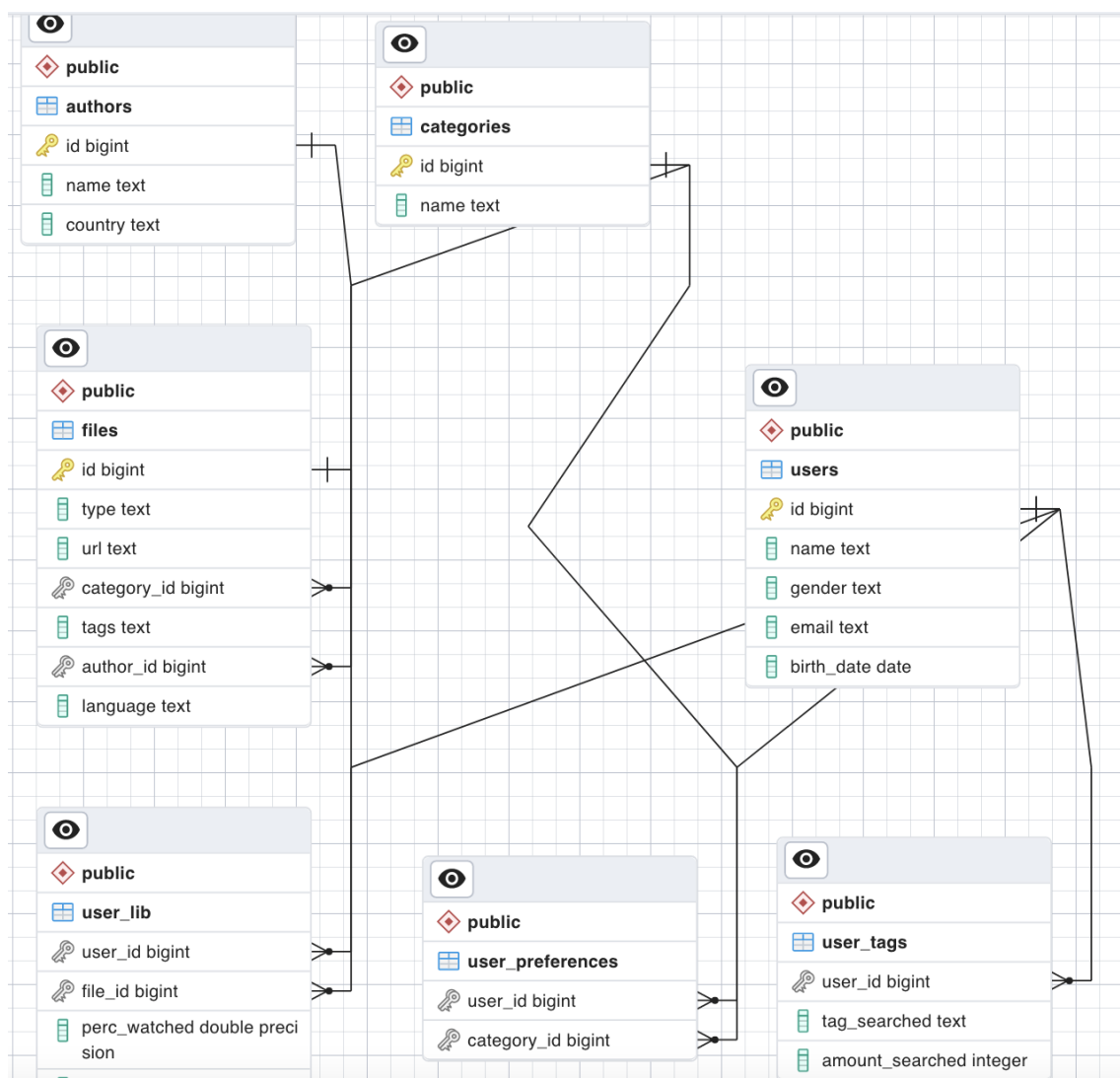


Рис. 4.6. ER діаграма, що демонструє зв'язки між таблицями

4.4 Створення беклогу проєкту

Процес створення сторімаппінгу беклогу проєкту з шерінгу самокатів:

1. Створення теми Визначаємо тему беклогу проєкту - шерінг самокатів.
2. Визначення користувачів Визначаємо цільових користувачів продукту та їх потреби. Наприклад, можуть бути такі користувачі: велосипедисти, студенти, туристи, місцеві жителі.
3. Створення "мапи історій" (story map) Створюємо карту історій, яка складається з основних функцій продукту та історій користувачів. Основні функції можуть включати такі пункти: замовлення самоката, оплата, перевірка наявності заряду, пошук місць, де можна залишити самокат, здача самоката.
4. Деталізація історій Кожну історію користувача розбиваємо на менші складові елементи, наприклад, для історії "Як велосипедист я хочу знайти вільний самокат, щоб поїхати на ньому" може бути такий список: перегляд мапи з вільними самокатами, фільтрація за місцем знаходження, бронювання самоката.
5. Планування релізів Створюємо план релізів, де розподіляємо історії користувачів на окремі релізи.
6. Призначення пріоритетів Для кожної історії користувача встановлюємо пріоритети в рамках релізу.
7. Оцінювання складності Оцінюємо складність реалізації кожної історії користувача та встановлюємо терміни виконання.
8. Планування спринтів Створюємо план спринтів, де розподіляємо історії.

Наступний етап – поділ беклогу на дві частини: MVP (Minimum Viable Product) та додатковий функціонал.

MVP - це мінімально необхідний функціонал для запуску товару ринку.

Це функціонал, який дозволяє користувачам виконувати основні завдання продукту. У випадку з нашим проектом шерінгу самокатів, MVP включає наступні функції:

- Реєстрація та авторизація користувачів
- Перегляд доступних самокатів на карті
- Бронювання самокатів
- Оренда самокату на певний період
- Відстеження стану заряду самокату в реальному часі
- Оплата оренди самокату

Додатковий функціонал буде включати такі функції, як:

- Можливість залишати відгуки про самокати
- Замовлення ремонту для самокату
- Можливість вибрати тип самокату (електричний чи звичайний)
- Програма лояльності для постійних користувачів
- Інтеграція із системами громадського транспорту
- Можливість зробити попереднє замовлення на самокат

Створення сторі-мапінгу для кожної частини.

Структура сторі-мапінгу для MVP:

1. Реєстрація та авторизація

- Створення облікового запису
- Авторизація у системі

2. Перегляд доступних самокатів

- Перегляд списку самокатів на карті
- Перегляд інформації про самокат

3. Бронювання та оренда самокату

- Бронювання самокату на певний період часу
- Оренда самокату на певний період

4. Відстеження стану заряду самокату

- Відображення рівня заряду самокату в реальному часі
- Оповіщення користувача про необхідність заміни самокату з низьким зарядом

5. Оплата оренди самокату

- Оплата оренди через програму

4.5 Розробка каркасного інтерфейсу

Каркасний інтерфейс - це структура, що визначає основні компоненти системи та взаємозв'язки між ними. Він дозволяє розбити систему на окремі частини та спрощує процес розробки, забезпечуючи зручний інтерфейс для взаємодії між компонентами. Це спрощує процес розробки, забезпечує більшу стабільність та масштабованість системи. Крім того, каркасний інтерфейс дозволяє використовувати різні технології та підходи до розробки окремих компонентів системи, що сприяє більш ефективному використанню ресурсів та прискоренню процесу розробки.

Екран входу/реєстрації

- Форма входу з полями для введення електронної пошти та пароля
- Посилання на сторінку реєстрації для нових користувачів
- Посилання на сторінку відновлення пароля для забутих паролів

Головний екран

- Карта з розташуванням доступних скутерів
- Пошук за адресою або назвою місця
- Фільтри за типом скутера, вартістю, відстанню та іншими параметрами
- Можливість замовити скутер на певний час

Екран бронювання скутера

- Інформація про скутер, включаючи його місцезнаходження та стан акумулятора
- Час, на який заброньовано скутер
- Попередження про необхідність своєчасного повернення скутера
- Кнопка підтвердження бронювання та оплати

Екран підтвердження платежу

- Інформація про вартість бронювання та остаточну суму платежу
- Поле для введення даних платіжної картки
- Кнопка завершення оплати та підтвердження бронювання

Екран під час водіння

- Інформація про скутер, включаючи його місцезнаходження та стан акумулятора
- Таймер для відстеження часу в дорозі
- Інформація про пройдену відстань за одну поїздку
- Кнопка завершення поїздки та повернення скутера

Екран оплати поїздки

- Інформація про вартість поїздки та остаточну суму оплати
- Поле для введення даних платіжної картки
- Кнопка для завершення оплати та підтвердження поїздки



Рис 4.7. Каркасный інтерфейс

4.6. Тестування системи

TestRail - це інструмент для керування тестуванням програмного забезпечення, що дозволяє зберігати тест-кейси, створювати тест-плани, виконувати тестування, відстежувати помилки та створювати звіти про прогрес тестування. TestRail надає гнучку інтеграцію з іншими інструментами тестування, такими як JIRA, Selenium, Jenkins та багатьма іншими.

Однією з головних переваг використання TestRail є можливість керувати тестами та відстежувати їх стан на різних етапах розробки проєкту. TestRail дозволяє організовувати тестові набори та структурувати їх за різними параметрами, такими як пріоритет, складність тестів, ризик, що дозволяє зробити тестування більш ефективним та зменшити ризик виявлення помилок в продукті після його випуску.

Також TestRail дозволяє відслідковувати статус виконання кожного тесту, пов'язувати їх з помилками, що були виявлені та визначати пріоритети

для їх вирішення. Це дозволяє команді розробників та тестувальників ефективніше взаємодіяти та планувати свою роботу, що зменшує час на виявлення та виправлення помилок [56.57].

Крім того, TestRail надає можливість створювати детальні звіти про стан тестування, які містять інформацію про кількість виконаних тестів, кількість виявлених помилок та їх статус, що дозволяє зробити аналіз ефективності тестування та прийняти відповідні рішення для покращення якості продукту.

Нижче наведено тест-кейси для перевірки системи шерінгу електросамокатів [58]:

1. Перевірка, що на карті правильно відображається місцезнаходження всіх доступних електросамокатів у системі шерінгу.
2. Перевірка, що інформація про кожен електросамокат (наприклад, заряд акумулятора) на карті відображається правильно.
3. Перевірка, що користувач може вибрати електросамокат на карті та переглянути детальну інформацію про нього.
4. Перевірка, що користувач може бронювати електросамокат на певний час.
5. Перевірка, що користувач може запустити електросамокат після успішної оплати.
6. Перевірка, що користувач може закінчити поїздку та повернути електросамокат на вільну стоянку.
7. Перевірка, що інформація про кожну успішну та неуспішну транзакцію оплати належним чином записується в системі.

ВИСНОВКИ

З ростом технологій та розвитком міських інфраструктур, з'явилася можливість використання електросамокатів як зручного, екологічного та економічного виду транспорту.

Управління проєктом розробки сервісу з шерінгу електросамокатів є актуальним завданням, яке потребує вирішення багатьох складних технічних та бізнес-питань. Використання сучасних технологій та аналіз даних дозволить розробити ефективний сервіс, який задовольнить потреби користувачів та забезпечить високий рівень безпеки.

Щодо статистики кількості зацікавлених користувачів, можна очікувати значний ріст популярності електросамокатів та сервісів шерінгу в майбутньому. За даними Statista, ринок шерінгу електросамокатів очікується зрости з 1,1 млн поїздок у 2019 році до 7,3 млн поїздок у 2025 році в Європі та США. Крім того, з введенням зелених зон та зменшенням кількості автомобілів у містах, розвиток шерінгу електросамокатів може стати однією з ключових тенденцій міської мобільності в майбутньому.

Звідома інформація про використання сервісів шерінгу електросамокатів в Україні є обмеженою. Проте, за даними аналітичної компанії "Ukrainian E-commerce Expert", ринок шерінгу електросамокатів в Україні почав розвиватися в 2018 році і зростає з кожним роком. У 2020 році обсяг цього ринку становив близько 3 млн доларів.

За даними сервісу Google Trends, інтерес користувачів до шерінгу електросамокатів в Україні також зростає. У першій половині 2021 року попит на пошукову фразу "шерінг електросамокатів" в Україні зрос на 17% порівняно з попереднім роком.

Отже, можна прогнозувати, що з розвитком інфраструктури та збільшенням свідомості користувачів щодо переваг екологічного транспорту, кількість зацікавлених користувачів шерінгу електросамокатів в Україні буде продовжувати зростати в майбутньому.

Під час дослідження було проаналізовано існуючі сервіси з шерінгу електросамокатів, виявлено їх переваги та недоліки. Було розглянуто технічні аспекти розробки сервісу, такі як вибір платформи, розробка інтерфейсу та забезпечення безпеки користувачів.

Результатом проекту є розроблений сервіс з шерінгу електросамокатів, який відповідає вимогам сучасного користувача та ринку. Особлива увага приділялась забезпеченню безпеки користувачів, тому були використані сучасні технології для захисту даних та попередження несанкціонованого доступу до сервісу.

Для подальшого розвитку проекту рекомендується провести додаткове дослідження ринку та конкурентних сервісів, вдосконалити інтерфейс та забезпечити максимально можливу зручність та швидкість роботи сервісу. Також рекомендується залучити додаткові інвестиції для подальшого розвитку та розширення сервісу.

У цілому, розробка сервісу з шерінгу електросамокатів є актуальною та перспективною темою в сучасному світі, а отримані результати та рекомендації можуть бути корисними для дослідження та розвитку подібних проектів в майбутньому.

В рамках кваліфікаційної роботи було проведено аналіз конкурентів та визначено, що на даний момент в країні можна конкурувати за рахунок створення більш зручного та привабливого для користувача сервісу, проведено аналітичний метод, який дозволяє оцінити сильні та слабкі сторони проекту, можливості і загрози, які впливають на нього, політичні, економічні, соціальні та технологічні аспекти, які впливають на нього, також аналіз внутрішнього та зовнішнього середовища.

Проведено інвестиційні дослідження та побудовано економічну модель проекту, визначено економічну вигоду від інвестиційних вкладень. Описано мету, цілі та продукт проекту. Побудовано дерево проблем та цілей

Розроблено ЖЦ проекту, який складається з 8 основних фаз – ініціація, планування, проведення закупівель, виконання, контроль, тестування, впровадження ПЗ, закриття проекту.

Побудовано WBS, яка складається з трьох основних компонентів – управління проектом, розробка та завершення проекту. Кожен компонент поділен на етапи.

Побудовано організаційну структуру проекту, описано потрібні ресурси для реалізації проекту і витрати на них, описано склад команди.

Проведено календарне планування проекту. Увесь проект займає 10 місяців та поділено на 10 спринтів.

Планування вартості здійснювалось шляхом розрахунку витрат по категоріях та зведено у єдиний кошторис.

Проведено управління зацікавленими сторонами, їх було поділено на первинні та вторинні сторони, визначено їх вплив на проект, очікування, стратегію взаємодії та розроблено план комунікації.

Проведено управління ризиками. Було визначено 4 найнебезпечніші ризики для проекту і їхній пріоритет. Описано протиризикові дії для подій, які несуть найбільш негативний вплив на проект.

Для розробки продукту було здійснено опис структурної схеми інформаційних потоків в системі, описано структуру бази даних, структуру та модулі програмного забезпечення проекту, прописано беклог продукту, розроблено каркасний інтерфейс і проведено тестування продукту з виправленням помилок.

ПЕРЕЛІК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1. Google Trends по запиту Електросамокат [Електронний ресурс] – Режим доступу: <https://trends.google.com/trends/explore?geo=UA&q=електросамокат>
2. За екологію та проти зростання ДТП. Як регулюють електросамокати [Електронний ресурс] – Режим доступу: <https://suspihne.media/amp/61371-za-ekologiu-ta-proti-zrostanna-dtp-ak-u-riznih-krainah-reguluut-elektrosamokati/>
3. Forbes.ua В Україні працює шість сервісів шерингу електросамокатів. Пояснюємо на цифрах, як влаштований цей ринок [Електронний ресурс] – Режим доступу: <https://forbes.ua/innovations/v-ukraini-pratsyuemyat-servisiv-sheringu-elektrosamokativ-poyasnyuemo-na-tsifrakh-yak-vlashtovaniy-tsey-rinok-16062021-1844>
4. Ain.ua Дослідження електротранспорту в Україні. [Електронний ресурс] – Режим доступу: <https://ain.ua/2022/06/14/cherez-deficyt-palnogo-popyt-na-elektromobili-zris-v-7-raziv-a-na-elektrosamokaty-v-3/amp/>
5. 5 сил Портера – Дія.Бізнес. [Електронний ресурс] – Режим доступу: <https://business.diia.gov.ua/handbook/marketing/5-sil-portera>
6. Управління проектами: процеси планування проектних дій [Текст]: підручник / І.В. Чумаченко, В.В. Морозов, Н.В. Доценко, А.М. Чередніченко. – К.: Університет економіки та права «КРОК», 2014. – 673 с.
7. Катренко, А.В. Системний аналіз об'єктів та процесів комп'ютеризації [Текст]: навч. посіб. / А.В. Катренко. – Л.: Науковий світ, 2000. – 424 с.

8. Керівництво з управління інноваційними проектами та програмами P2M: перекл. з англ.; під ред. С.Д. Бушуєва. – К.: Науковий світ, 2009. – 173 с.
9. Розробка проекту: Аналіз зацікавлених сторін. [Електронний ресурс] – Режим доступу: <https://tamarisk.org.ua/rozrobka-proektu-analizatsikavlenih-storin/>
10. Управління зацікавленими сторонами. [Електронний ресурс] – Режим доступу: <https://studfile.net/preview/7153489/page:36/>
11. Управління ризиками в проектному менеджменті / Н. С. Скопенко, І.В. Євсєєва, В. О. Москаленко. – 2013. – С. 41-44.
12. Управління проектними ризиками / Кравченко М. О., Пиркова О. В,С. 1-6.
13. Ризики проекту: аналіз, оцінка та стратегії управління. [Електронний ресурс] – Режим доступу: <https://skillsetter.io/blog/risk-management-ua>
14. Оцінка ризиків проекту. [Електронний ресурс] – Режим доступу: https://pidru4niki.com/86060/investuvannya/otsinka_rizikiv_proektu
15. Управління проектними ризиками. [Електронний ресурс] – Режим доступу: <https://library.if.ua/book/96/6633.html>
16. Морозов В.В. Управління проектами: процеси планування проектних дій [Текст]: підручник / І.В. Чумаченко, В.В. Морозов, Н.В. Доценко, А.М. Чередніченко. – К.: Університет економіки та права «КРОК», 2014. – 673 с.
17. Морозов В.В., Чередніченко А.М. та інші. Формування, управління та розвиток команди проекту / Навчальний посібник. - К.: Таксон, 2009. 466 с.
18. Project Management Body of Knowledge (PMBOK) - Project Management Institute, Inc.

19. Бізнес-менеджмент: Навч. посіб. / Л.І. Федулова, В.Г. Федоренко, В.Ф. Гриньов, В.П. Сладкевич, В.Є. Воротін; Міжрегіон. акад. упр. персоналом.
20. Чатфілд К., Джонсон Т. Д., Сінді Льюїс. Microsoft Project 2019. Крок за кроком. ДМК Прес, 2021.
21. Хігні Дж. Основи управління проектами. Фабула, 2020. 269 с.
22. Верба В.А., Загородніх О.А. Проектний аналіз: Підручник. - К.-КНЕУ, 2000.- 322 с.
23. Гевко і.Б.: Операційний менеджмент: Навчальний посібник: - К.: Кондор, 2005. - 228 с.
24. Морозов В. В. Компоненти управління проектами: навчальний посібник для самостійної роботи студентів магістратури по спеціальності 8.000003 "Управління проектами" / В.В. Морозов. – К.: Університет економіки та права "КРОК", 2005. – 62 с.
25. Бушуев С. Д. Динамічне лідерство в управлінні проектами: Монографія / С. Д. Бушуев, В. В. Морозов // Українська асоціація управління проектами. – К., 1999. – 312 с
26. Батенко Л. П. Управління проектами: навч. посіб. / Л. П. Батенко, О. А. Загородніх, В. В. Ліщинська. – К.: КНЕУ, 2003. – 231 с.
27. Інвестиційно-інноваційні чинники формування розвитку й управління конкурентоспроможним потенціалом підприємства: Монографія / Д. В. Солоха, В. В. Морева, С. О. Чирков, В. Я. Козлова, О. В. Белякова. – Донецьк: СПД Дмитренко Л. Р., 2010. – 400 с.
28. 5 сил Портера –Дія.Бізнес. [Електронний ресурс] – Режим доступу: <https://business.diia.gov.ua/handbook/marketing/5-sil-portera>
29. Соклакова І. В. Стратегічний аналіз діяльності організації, 2020 – 242 с.
30. Роберт Грант. Сучасний стратегічний аналіз. 2017, – Вип. 9. – С. 670

31. Соклакова І. В. Стратегічний аналіз діяльності організації, 2020 – 210 с.
32. Кобилянський Л. С. Управління проектами : навчальний посібник / Л.С. Кобилянський. – К.: МАУП, 2002. – 200 с.
33. Управління комунікаціями проекту / В.І. Торкатюк, Л.П. Вороновская, С.В. Аксенова, О.В. Леуненко, М.В. Грицаненко, – С. 1-3.
34. Аналіз зовнішнього середовища підприємства. [Електронний ресурс] – Режим доступу: <https://buklib.net/books/22609/>
35. Костенко Т.Д., Підгора Є.О., Рижиков В.С., Панков В.А. Економічний аналіз і діагностика стану сучасного підприємства: Навч. посібник для студентів вищих навч. закладів. - К.: ЦНЛ, 2008. – 398 с.
36. Управління проектами / Петренко Н.О., Кустріч Л.О., Гоменюк М. О., – 2019, – 244с.
37. «Дерево проблем». [Електронний ресурс] – Режим доступу:
а. <https://hromada.canactions.com/derevo-problem-rishen/>
38. «Дерево цілей підприємства». [Електронний ресурс] – Режим доступу: <https://buklib.net/books/25617/>
39. «Діагностика та ідентифікація проблем». [Електронний ресурс] - Режим доступу: https://stud.com.ua/31874/menedzhment/diagnostika_identifikatsiya_proble m
40. «Мета й стратегія проекту». [Електронний ресурс] - Режим доступу: <https://buklib.net/books/34066/>

41. «Цілі та принципи управління проектами. Життєвий цикл проекту.»
[Електронний ресурс] - Режим доступу:
<https://buklib.net/books/22487/>
42. «Формулювання мети та завдань проекту». [Електронний ресурс] -
Режим доступу:
https://pidru4niki.com/1048030452762/ekonomika/formulyuvannya_meti_z_avdan_proektu
43. Павлова С.І. Підходи до поділу життєвого циклу проекту в проектному менеджменті. [Електронний ресурс] – Режим доступу:
<https://conf.ztu.edu.ua/wp-content/uploads/2018/12/192.pdf>
44. Ієрархічна структура робіт проекту. [Електронний ресурс] – Режим доступу:
https://stud.com.ua/21060/menedzhment/iyerarhichna_struktura_robit_proektu
45. Управління проектами / Л. П. Батенко, О. А. Загородніх, В. В. Ліщинська. – К.: КНЕУ, 2003. – 231 с.
46. MS Project: Огляд можливостей для ефективного планування.
[Електронний ресурс] – Режим доступу:
<https://blog.ganttpro.com/ru/obzor-ms-microsoft-project/>
47. Планування послідовності робіт. [Електронний ресурс] – Режим доступу: <https://buklib.net/books/24837/>
48. Лозова О. В. Управління ресурсами компанії у проектах. – 2018. – С. 67-71.
49. Управління проектами / Довгань Л. Є., Мохонько Г. А., Малик І. П.; навч. посіб. – К.: Центр учбової літератури, 2017. – 429 с.
50. Управління вартістю проектами. [Електронний ресурс] – Режим доступу: https://pidru4niki.com/87721/menedzhment/upravlinnya_vartistyu_proektiv

51. Управління витратами / Чорна М.В., Смірнова П.В., Бугріменко Р.М, 2017. – 166 с.
52. Оцінка вартості проєкту. [Електронний ресурс] – Режим доступу: https://pidru4niki.com/87722/menedzhment/otsinka_vartosti_proektu
53. Основні поняття реляційних БД: нормалізація, зв'язок та ключі. [Електронний ресурс] – Режим доступу: <https://bondarenko.dn.ua/osnovni-ponyattya-relyatsijnih-bd-normalizatsiyazv-yazok-ta-klyuchi/>
54. Етап концептуального проєктування. Основні поняття концептуального проєктування. Концептуальне проєктування. Об'єкти і їх властивості. Взаємовідношення об'єктів. [Електронний ресурс] – Режим доступу: <https://studfile.net/preview/5454386/page:5/>
55. Побудова логічної моделі даних. [Електронний ресурс] – Режим доступу: <https://studfile.net/preview/7144845/page:29/>
56. Що таке життєвий цикл багу. [Електронний ресурс] – Режим доступу: <https://www.qamadness.com/knowledge-base/what-is-a-bug-life-cycle/>
57. Вступ в тестування програмного забезпечення / Луїза Тамро, – 2018, – 368 с.
58. Тестування від А до Я. Тест-кейси. [Електронний ресурс] – Режим доступу: <https://ilarionhalushka.github.io/testing-theory>

ДОДАТОК А

Таблиця 3.2

Створення системи для шерінгу електросамокатів
Початок
Розробка концепції
Визначення цілей проекту
Побудова дерева цілей
Побудова дерева проблем
PEST аналіз
SWOT аналіз
Визначення задач проекту
Розбиття проекту на підзадачі
Визначення термінів виконання підзадач
Процес підготовки
Постановка вимог
Визначення функціональних вимог
Визначення нефункціональних вимог
Розробка календарного плану
Створення статуту проекту
Процес виконання
Розробка моделі
Формалізація бізнес-логіки
Створення ядра програми
Створення презентаційного шару
Розробка дизайну
Розробка контенту
Текстовий контент
Візуальний контент
Верстка
форумання HTML
css
javascript
З'єднання серверної та клієнтської частин
Організація контролерів
Прийом платежів
Автоматизація розгортання
Налаштування СКВ
Налаштування Jenkins
Тестування
Визначення обмежень
Написання тестів
Тестування навантаженням
Тестування функціональності
Формування звіту по тестуванню

Статичний аналіз коду
Виправлення знайдених помилок
Додавання лендінгів
Внутрішня оптимізація
Завершення
Аудит
Закриття
Фініш

ДОДАТОК Б

ОГОЛОШЕННЯ

про проведення відкритих торгів

Найменування замовника:		ТОВ «ShareScoot»		
Категорія замовника:		Юридична особа, яка здійснює діяльність в одній або декількох окремих сферах господарювання		
Ідентифікаційний код замовника в ЄДР:		43146904		
Місцезнаходження замовника:		Україна, 04071, місто Київ, вул.Костянтинівська, будинок 32, офіс 8		
Контактна особа замовника, уповноважена здійснювати зв'язок з учасниками:		Міць Павло Олександрович,+380952122264		
Вид предмета закупівлі:		Електро-транспорт		
Назва предмета закупівлі:		Електросамокат двоколісний		
Код за Єдиним закупівельним словником:		ДК 021:2015 34100000-8 Електротранспортні засоби		
Назва номенклатурної позиції предмета закупівлі	Код згідно з Єдиним закупівельним словником, що найбільше відповідає назві номенклатурної позиції предмета закупівлі	Кількість товарів або обсяг виконання робіт чи надання послуг	Місце поставки товарів або місце виконання робіт чи надання послуг	Строк поставки товарів, виконання робіт чи надання послуг
Електросамокат	ДК 021:2015 34100000-8 Електротранспортні засоби	25 шт	Україна, 04071, місто Київ, вул.Костянтинівська, будинок 32, офіс 8	до 1 червня 2023

Умови оплати:

Поставка товару		Післяплата	20	Календарні	100
Очікувана вартість предмета закупівлі:			450 000,00 UAH		
Джерело фінансування закупівлі: Власний бюджет (кошти від ведення господарської діяльності підприємства)					
Джерело фінансування закупівлі			Опис		Сума
Інше					450 000,00 UAH
Розмір мінімального кроку пониження ціни:		2000,00 UAH			
Математична формула для розрахунку приведеної ціни (у разі її застосування):		відсутня			
Кінцевий строк подання тендерних пропозицій:		10 Травня 2023 09:00			
Мова тендерної пропозиції:		українська			
Дата та час розкриття тендерних пропозицій:		відсутній , після завершення електронного аукціону			
Дата та час проведення електронного аукціону:		10 травня 2023 15:15			
Вид та умови надання забезпечення пропозицій учасників:		відсутні			
Розмір та умови надання забезпечення виконання договору про закупівлю:		відсутні			

ДОДАТОК В

Java модуль Сканування QR коду для активації електросамоката:

```
import java.util.EnumMap;
import java.util.Map;

import com.google.zxing.BarcodeFormat;
import com.google.zxing.BinaryBitmap;
import com.google.zxing.DecodeHintType;
import com.google.zxing.MultiFormatReader;
import com.google.zxing.NotFoundException;
import com.google.zxing.RGBLuminanceSource;
import com.google.zxing.ReaderException;
import com.google.zxing.Result;
import com.google.zxing.common.HybridBinarizer;
import com.google.zxing.qrcode.QRCodeReader;

public class QrScanner {

    public static void main(String[] args) {

        // Параметри для сканування QR коду
        Map<DecodeHintType, Object> hints = new EnumMap<>(DecodeHintType.class);
        hints.put(DecodeHintType.TRY_HARDER, true);
        hints.put(DecodeHintType.POSSIBLE_FORMATS, BarcodeFormat.QR_CODE);

        // Підключення камери
        Camera camera = new Camera();

        // Сканування QR коду
        while (true) {
            Image image = camera.capture();
            String result = decodeQRCode(image, hints);
            if (result != null) {
                System.out.println("Результат сканування: " + result);
                // Передача даних у форматі JSON для активації самоката
                activateScooter(result);
                break;
            }
        }
    }
}
```

```

        // Вимкнення камери
        camera.release();
    }

    private static String decodeQRCode(Image image, Map<DecodeHintType, Object>
hints) {
        try {
            // Створення об'єкту RGBLuminanceSource з отриманого зображення
            RGBLuminanceSource source = new RGBLuminanceSource(image.getWidth(),
image.getHeight(), image.getPixels());
            BinaryBitmap bitmap = new BinaryBitmap(new HybridBinarizer(source));

            // Створення об'єкту QRCodeReader та MultiFormatReader для сканування QR
коду
            QRCodeReader qrCodeReader = new QRCodeReader();
            MultiFormatReader multiFormatReader = new MultiFormatReader();
            multiFormatReader.setHints(hints);

            try {
                // Спроба сканування QR коду
                Result result = qrCodeReader.decode(bitmap);
                return result.getText();
            } catch (NotFoundException e) {
                // Якщо спроба з QRCodeReader не вдалася, спробувати MultiFormatReader
                try {
                    Result result = multiFormatReader.decode(bitmap);
                    return result.getText();
                } catch (ReaderException ex) {
                    // Якщо спроба із MultiFormatReader також не вдалася, поверну
                    // Спробуємо розпізнати QR-код за допомогою QRCodeReader
                    Result result = qrCodeReader.decode(bitmap);
                    return result.getText();
                } catch (IOException ex) {
                    System.out.println("Помилка при зчитуванні файлу: " + ex.getMessage());
                    return null;
                } catch (NotFoundException ex) {
                    System.out.println("QR-код не знайдено: " + ex.getMessage());
                    return null;
                }
            }
        }
    }
}

```

ДОДАТОК Г

Java модуль для пошуку електросамокату на карті

```
import com.google.maps.GeoApiContext;
import com.google.maps.model.LatLng;
import com.google.maps.model.MarkerOptions;
import com.google.maps.PlacesApi;
import com.google.maps.errors.ApiException;

public class ScooterMap {
    private final String API_KEY = "YOUR_API_KEY_HERE";

    public void addScooterMarker(double lat, double lng) throws ApiException,
        InterruptedException, IOException {
        GeoApiContext context = new
        GeoApiContext.Builder().apiKey(API_KEY).build();
        LatLng location = new LatLng(lat, lng);
        PlacesSearchResponse searchResponse =
        PlacesApi.nearbySearchQuery(context, location)
            .radius(100)
            .type(PlaceType.PARKING)
            .await();

        if(searchResponse.results.length > 0) {
            PlacesSearchResult place = searchResponse.results[0];
            LatLng placeLocation = place.geometry.location;
            MarkerOptions markerOptions = new MarkerOptions();
            markerOptions.position(placeLocation);
            markerOptions.title("Scooter Location");

            markerOptions.icon(BitmapDescriptorFactory.fromResource(R.drawable.scooter_ico
            on));

            // Додаємо маркер на карту
            mMap.addMarker(markerOptions);
        }
    }
}
```

ДОДАТОК Д

Java модуль оплати поїздки на електросамокаті

```
import java.math.BigDecimal;
import java.util.HashMap;
import java.util.Map;
import com.stripe.Stripe;
import com.stripe.exception.StripeException;
import com.stripe.model.Charge;

import com.braintreegateway.BraintreeGateway;
import com.braintreegateway.CreditCard;
import com.braintreegateway.Customer;
import com.braintreegateway.Result;
import com.braintreegateway.Transaction;
import com.braintreegateway.TransactionRequest;

public class PaymentModule {

    // Метод для ініціалізації BraintreeGateway
    public BraintreeGateway initBraintreeGateway() {
        final String merchantId = "your_merchant_id";
        final String publicKey = "your_public_key";
        final String privateKey = "your_private_key";

        return new BraintreeGateway(
            Environment.SANDBOX, // Використовуємо Sandbox для тестування
            merchantId,
            publicKey,
            privateKey
        );
    }

    // Метод для створення транзакції
    public Transaction createTransaction(String nonceFromClient, BigDecimal
amount) {
        BraintreeGateway gateway = initBraintreeGateway();

        TransactionRequest request = new TransactionRequest()
            .amount(amount)
            .paymentMethodNonce(nonceFromClient)
    }
}
```

```

        .options ()
        .submitForSettlement (true)
        .done ();

Result<Transaction> result = gateway.transaction ().sale (request);

if (result.isSuccess ()) {
    return result.getTarget ();
} else if (result.getTransaction () != null) {
    // Виняткова ситуація виникає, якщо транзакція не була успішно
    виконана, але була згенерована помилка
    throw new RuntimeException ("Transaction failed with message: " +
result.getTransaction ().getStatus ().toString ());
} else {
    throw new RuntimeException ("Transaction failed with message: " +
result.getMessage ());
}
}

// Метод для додавання кредитної карти
public String addCreditCard (String customerId, String cardholderName,
String cardNumber, int expirationMonth, int expirationYear, String cvv) {
    BraintreeGateway gateway = initBraintreeGateway ();

    Map<String, Object> customerParams = new HashMap<> ();
    customerParams.put ("id", customerId);

    CreditCardRequest request = new CreditCardRequest ()
        .customerId (customerId)
        .cardholderName (cardholderName)
        .number (cardNumber)
        .expirationMonth (String.valueOf (expirationMonth))
        .expirationYear (String.valueOf (expirationYear))
        .cvv (cvv);

    Result<CreditCard> result = gateway.creditCard ().create (request);
    if (result.isSuccess ()) {
        return result.getTarget ().getToken ();
    } else {
        throw new RuntimeException ("Credit card could not be added: " +
result.getMessage ());
    }
}
}

```

```

// Метод для отримання даних клієнта
public Customer getCustomer(String customerId) {
    BraintreeGateway gateway = initBraintreeGateway();

    Result<Customer> result = gateway.customer;
}

// Ключ API Stripe
Stripe.apiKey = "sk_test...";
// Загальна вартість електросамоката, яку необхідно сплатити
BigDecimal amount = new BigDecimal(1000);
// Код валюти, в якій проводиться транзакція
String currency = "usd";
// Карткові дані клієнта (ім'я, номер картки, дата закінчення дії,
CVC)
Map<String, Object> cardParams = new HashMap<String, Object>();
cardParams.put("number", "4242424242424242");
cardParams.put("exp_month", 12);
cardParams.put("exp_year", 2023);
cardParams.put("cvc", "123");
// Додаткова інформація про клієнта (ім'я, електронна пошта)
Map<String, Object> customerParams = new HashMap<String, Object>();
customerParams.put("name", "John Doe");
customerParams.put("email", "john.doe@example.com");
// Параметри транзакції (вартість, валюта, карткові дані, додаткова
інформація про клієнта)
Map<String, Object> chargeParams = new HashMap<String, Object>();
chargeParams.put("amount", amount);
chargeParams.put("currency", currency);
chargeParams.put("source", cardParams);
chargeParams.put("customer", customerParams);
try {
    // Створення транзакції в Stripe
    Charge charge = Charge.create(chargeParams);

    // Оплата успішно проведена
    System.out.println("Payment successful. Charge ID: " +
charge.getId());

} catch (StripeException e) {
    // Помилка при проведенні транзакції
    e.printStackTrace();
    System.out.println("Payment failed. Error message: " +
e.getMessage());}}

```