

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
Навчально-науковий інститут
«Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»
галузь знань 12 «Інформаційні технології»
спеціальність 122 «Комп'ютерні науки»

Форма навчання: денна

КВАЛІФІКАЦІЙНА БАКАЛАВРСЬКА РОБОТА

на тему: «Розроблення інформаційної системи з визначення оптимальних параметрів чотирьохтактного двигуна мотоцикла»

здобувача **Борисенка Олександра Андрійовича**

(ПІБ)



(Підпис)

Науковий керівник:

к.е.н., доцент

_____ Тішков Б.О.

**Робота допущена до захисту перед
екзаменаційною комісією з**

атестації здобувачів вищої освіти

завідувач кафедри:

к.е.н., доцент

_____ Тішков Б.О.

Міністерство освіти і науки України
Київський національний економічний університет імені Вадима Гетьмана
Навчально-науковий інститут «Інститут інформаційних технологій в економіці»

Кафедра інформаційних систем в економіці

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»

галузь знань 12 «Інформаційні технології»

спеціальність 122 «Комп'ютерні науки»

ПОГОДЖЕНО:

Керівник проектної групи(гарант)
освітньо-професійної програми

Помазун О.М.
“ ” 2025 р.

ЗАТВЕРДЖУЮ:

Завідувач кафедри
Тішков Б.О.

“ ” 2025 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

здобувачу вищої освіти Борисенку Олександрю Андрійовичу

очної (денної) форми навчання

на підготовку кваліфікаційної бакалаврської роботи

на тему: «Розроблення інформаційної системи з визначення оптимальних параметрів чотирьохтактного двигуна мотоцикла»

Тему затверджено наказом ректора Університету від « 7 » березня 2025 р.
№ 466- ст.

Кваліфікаційна бакалаврська робота виконується на матеріалах отриманих під час навчання, даних з навчальних посібників та інтернет - ресурсів, а також на фактичному матеріалі, зібраному автором за темою дослідження при виконанні курсових робіт.

План кваліфікаційної бакалаврської роботи

Розділ I Характеристика та наліз предметної області

Розділ II Розробка вимог і моделювання інформаційної системи

Розділ III Проектування та реалізація компонентів системи

Об'єкт дослідження розроблення інформаційної системи.

Предмет дослідження є обробка, аналіз та оптимізація параметрів чотиритактного двигуна мотоцикла на основі сенсорних даних з використанням сучасних інформаційних технологій.

Мета кваліфікаційної бакалаврської роботи полягає в розробці інформаційної системи з визначення оптимальних параметрів чотирьохтактного двигуна.

Конкретні завдання, які здобувач повинен виконати для досягнення поставленої мети:

У розділі I слід надати загальну характеристику галузі технічної експлуатації чотиритактних двигунів мотоциклів, з акцентом на проблематику аналізу та оптимізації їх робочих параметрів. Необхідно визначити об'єкт дослідження - інформаційну систему для моніторингу та оптимізації параметрів двигуна та описати структуру управління процесом збору, обробки і візуалізації даних. Особливу увагу слід приділити сучасним проблемам у діагностиці та контролі стану двигунів, які виникають при ручному підході або використанні застарілих методів. Доцільно навести схему організаційної структури функціонування ІС, описати ролі основних користувачів та підсистем.

У розділі II необхідно провести аналіз потреб користувачів і сервісного обслуговування мотоциклів, на основі чого сформулювати функціональні та нефункціональні вимоги до інформаційної системи. Важливо здійснити постановку задачі з урахуванням доцільності автоматизації процесу аналізу параметрів двигуна, розробити алгоритм розв'язання задачі оптимізації, подати блок - схеми та математичну модель оцінки ефективності двигуна. Окремо потрібно побудувати інформаційну модель системи: діаграми прецедентів, класів, компонентів, матриці трасування вимог. Також має бути виконано опис вхідних (показники з датчиків) та вихідних даних (ефективність, рекомендації, графіки), вказано їх джерела, спосіб збору та вимоги до точності й достовірності.

У розділі III необхідно виконати проєктування та реалізацію компонентів інформаційної системи: інформаційного, технічного, програмного та організаційного забезпечення відповідно до методичних вимог. Слід описати реалізовані функціональні модулі системи: авторизацію, збір і збереження показників сенсорів, виконання оптимізації параметрів, формування звітів, відображення графіків ефективності. Важливо надати опис обраного технічного рішення (Arduino, Raspberry Pi, Flutter, Flask), структури бази даних, маршрутизації запитів API та мобільного інтерфейсу користувача.

**Завдання підготував
науковий керівник**

Тішков Богдан Олександрович
« 10 » березня 2025 р.

**Завдання одержав
здобувач**

Борисенко Олександр

Андрійович

« 10 » березня 2025 р..

Відгук
про кваліфікаційну бакалаврську роботу
здобувача навчально - наукового інституту
«Інститут інформаційних технологій в економіці»
освітньо-професійної програми
«Комп'ютерні науки»

Борисенка Олександра Андрійовича

на тему
«Розроблення інформаційної системи з визначення оптимальних параметрів
чотирьохтактного двигуна мотоцикла»

Кваліфікаційна бакалаврська робота Борисенка О.А., присвячена обробці, аналізу та оптимізації параметрів чотиритактного двигуна мотоцикла на основі сенсорних даних з використанням сучасних інформаційних технологій.

В роботі проведений аналіз системних вимог та обґрунтування методів проектування системи оцінювання знань. На основі узагальнень обґрунтовані пропозиції та проектні рішення для системи моніторингу програмного забезпечення та мережевого обладнання.

Постановка задач виконана у відповідності з вимогами стандартів та методичних вказівок, алгоритм вирішення задачі розроблено з використанням сучасних технологій.

У роботі розглянуті питання інформаційного, технічного, програмного та організаційного забезпечення інформаційної системи.

При виконанні кваліфікаційної бакалаврської роботи студент Борисенко О.А, показав високий рівень знань, уміння узагальнювати практичний досвід використання сучасних ІТ - технологій, самостійно впроваджувати в практику проектні рішення.

В цілому кваліфікаційна бакалаврська робота студента Борисенка О.А. відповідає сучасним вимогам, заслуговує позитивної оцінки і може бути допущений до захисту в екзаменаційній комісії з атестації здобувачів вищої освіти.

Науковий керівник

доцент, к.е.н. Тішков Б.О.

(підпис)

Рецензія
на кваліфікаційну бакалаврську роботу
здобувача вищої освіти

Борисенка Олександра Андрійовича

Тема «Розроблення інформаційної системи з визначення оптимальних параметрів чотирьохтактного двигуна мотоцикла»

Актуальність теми кваліфікаційної бакалаврської роботи і доцільність її розроблення: тема кваліфікаційної бакалаврської роботи є безперечно актуальною, оскільки підвищення ефективності та надійності роботи двигунів внутрішнього згоряння є важливим завданням для сучасної інженерії та технічного обслуговування. Розробка доступної інформаційної системи для моніторингу та оптимізації параметрів двигуна мотоцикла є доцільною, враховуючи зростаючу популярність цього виду транспорту в Україні, що зумовлює потребу в сучасних інструментах для діагностики та збільшення терміну служби техніки. Робота вирішує проблему обмеженості традиційних «ручних» методів діагностики, пропонуючи автоматизоване рішення на базі сучасних інформаційних технологій.

Якість проведеного дослідження: дослідження проведено на високому методичному та технічному рівні. Студент ретельно проаналізував предметну галузь, існуючі програмні та апаратні рішення, на основі чого сформулював детальні функціональні та нефункціональні вимоги до системи. Проектування архітектури системи, структури бази даних та програмних модулів було виконано з використанням сучасних CASE - засобів (Enterprise Architect) та методологій моделювання (UML). Вибір стеку технологій (Raspberry Pi, Arduino, Flutter, Flask, MariaDB) є обґрунтованим, прагматичним та повністю відповідає поставленим завданням, що свідчить про глибоке розуміння автором принципів побудови комплексних інформаційних систем.

Позитивні риси кваліфікаційної бакалаврської роботи: до сильних сторін проєкту слід віднести високу практичну значущість та комплексний підхід до розробки, що об'єднує апаратну частину (сенсори, мікроконтролери), серверну логіку (API) та клієнтську реалізацію (мобільний застосунок) в єдину

працездатну систему. Позитивно відзначається створення повноцінного прототипу, що демонструє всі ключові функції: від збору даних до їх аналізу, візуалізації та управління звітами.

Зауваження: як рекомендацію щодо подальшого розвитку проекту варто було б розглянути можливість розробки веб - версії інтерфейсу для надання доступу до системи зі стаціонарних комп'ютерів, що розширило б її застосування у сервісних центрах та навчальних лабораторіях.

Практична значимість висновків і рекомендацій: як рекомендацію щодо подальшого розвитку проекту варто було б розглянути можливість розробки веб - версії інтерфейсу для надання доступу до системи зі стаціонарних комп'ютерів, що розширило б її застосування у сервісних центрах та навчальних лабораторіях.

Кафедра інформаційно - вимірювальних
технологій КПІ ім. Ігоря Сікорського

Доцент, кандидат технічних наук


Козир Олег Васильович
(підпис, ПІВ)

АНОТАЦІЯ

випускного бакалаврської роботи студента 4 курсу
Навчально - наукового інституту «Інститут інформаційних технологій в
економіці»

Борисенка Олександра Андрійовича, виконаного на тему:
«Розроблення інформаційної системи з визначення оптимальних параметрів
чотирьохтактного двигуна мотоцикла»

Київ: кафедра інформаційних систем в економіці, 2025 р.

Випускна бакалаврська робота присвячена актуальній проблемі підвищення ефективності роботи чотиритактного двигуна мотоцикла шляхом використання засобів цифрової автоматизації та інформаційних технологій. У роботі розроблено інформаційну систему, яка дозволяє збирати, аналізувати, візуалізувати та оптимізувати параметри двигуна на основі даних сенсорів у режимі реального часу. Проект складається з трьох логічно пов'язаних між собою розділів.

У першому розділі подано характеристику предметної галузі — систем моніторингу параметрів ДВЗ, проведено аналіз сучасних підходів до автоматизованої обробки даних, а також розглянуто наявні програмні рішення в цій сфері.

Другий розділ має проєктний характер і містить опис мети, специфікації вимог, постановки задачі, логіки її розв'язання, а також побудову інформаційної моделі системи та її архітектури. На основі моделювання обґрунтовано вибір компонентів і визначено взаємодію підсистем.

У третьому розділі викладено реалізацію системи: описано інформаційне забезпечення (типи даних, структура бази, моделі), технічне забезпечення (використання Raspberry Pi, Arduino, датчиків), програмне забезпечення (мобільний застосунок на Flutter, API на Flask та MariaDB), а також організаційне забезпечення. Наведено схеми автоматизації,

документацію, проаналізовано ефективність функціонування розробленої системи та її переваги для практичного використання в сфері технічного обслуговування мототехніки.

У висновках надано рекомендації щодо впровадження інформаційної системи на підприємствах або в особистому використанні для забезпечення контролю та підвищення надійності роботи чотиритактних двигунів.

РЕФЕРАТ

Дипломна бакалаврська робота містить: 112 сторінки, 16 таблиць, 45 рисунків, перелік джерел посилань з 30 найменувань, 4 додатку.

РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ З ВИЗНАЧЕННЯ ОПТИМАЛЬНИХ ПАРАМЕТРІВ ЧОТИРЬОХТАКТНОГО ДВИГУНА МОТОЦИКЛА

Перелік ключових слів: оптимізація двигуна, чотиритактний двигун, інформаційна система, мобільний застосунок, Raspberry Pi 5, Arduino UNO, сенсори, Flask API, база даних, MariaDB, Flutter, діагностика, ефективність, параметри роботи двигуна.

Предметом дослідження є обробка, аналіз та оптимізація параметрів чотиритактного двигуна мотоцикла на основі сенсорних даних з використанням сучасних інформаційних технологій.

Об'єктом дослідження є розроблення інформаційної системи з визначення оптимальних параметрів двигуна мотоцикла.

Мета дипломної роботи полягає в розробці інформаційної системи з визначення оптимальних параметрів чотирьохтактного двигуна.

Інструменти розроблення дипломної роботи: Figma, Enterprise Architect, Draw.IO, PyCharm, Android Studio, MariaDB, HeidiSQL.

Завданнями випускного бакалаврського проекту є: аналіз предметної області та технічних вимог, розробка бази даних для збереження параметрів оптимізації, реалізація API для взаємодії між БД та мобільним клієнтом, створення мобільного застосунку, забезпечення зручної візуалізації даних та розрахунків ефективності.

Апаратні та програмні засоби, що використовувались при проектуванні це Raspberry Pi 5 для розгортання серверної частини та СКБД MariaDB, Arduino UNO з підключеними сенсорами (обертів, температури, тиску, рівня масла, Flask (Python) для створення REST API, Flutter (Dart) для реалізації

мобільного застосунку, Enterprise Architect для моделювання, ZeroTier для віртуальної мережевої взаємодії.

Результати досягнуті в процесі роботи: розроблено повнофункціональну прототипну систему, що об'єднує серверну логіку та мобільний застосунок. Забезпечено автоматизовану обробку даних, вивід графіків ефективності та збереження звітів в базі даних. Наукова новизна полягає в інтеграції недорогих апаратних засобів в єдину ІС з підтримкою адаптивного інтерфейсу та VPN - з'єднання. Реалізація системи дозволяє підвищити ефективність технічного обслуговування мотоциклів та проводити аналіз роботи ДВЗ з мінімальними витратами.

Одержані результати можуть бути використані у сервісних центрах з технічного обслуговування мототехніки для моніторингу технічного стану, в навчальному процесі для демонстрації принципів збору та обробки технічних параметрів, власниками мотоциклів для персонального контролю за роботою двигуна, в інших проєктах IoT (Internet of Things), пов'язаних з моніторингом і оптимізацією параметрів технічних систем.

Рік виконання бакалаврської роботи – 2025.

Рік захисту бакалаврської роботи – 2025.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ ВЕЛИЧИН І ТЕРМІНІВ.....	13
ВСТУП.....	14
РОЗДІЛ 1	17
ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ	17
1.1. Характеристика предметної галузі та об'єкта дослідження.....	17
1.2. Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі.....	26
1.2.1 Загальні характеристики обраних систем.....	26
1.2.2 Розробка концепції інформаційної системи.....	28
РОЗДІЛ 2	30
РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ	30
2.1. Аналіз і специфікація вимог до інформаційної системи	30
2.2. Постановка задачі.....	36
2.2.1. Опис постановки задачі	36
2.2.2. Алгоритм розв'язання задачі.....	43
2.3. Моделювання інформаційної підсистеми	47
2.3.1 Моделювання поведінки системи	47
2.3.2 Моделювання структури системи.....	48
2.3.3 Розподіл вимог за компонентами системи	50
РОЗДІЛ 3	53
ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ ПІДСИСТЕМИ.....	53
3.1. Інформаційне забезпечення	53
3.1.1 Загальна характеристика інформаційного забезпечення	53
3.1.2 Організація збору і передання первинної інформації.....	55
3.1.3 Побудова системи класифікації та кодування	55
3.1.4 Проектування форм первинних документів та відеокадрів.....	57
3.1.5 Структура інформаційних масивів.....	58
3.1.6 Вибір СКБД	62
3.1.7 Інфологічна модель бази (сховища) даних.....	63
3.1.8 Даталогічна модель бази (сховища) даних.....	64
3.2. Технічне забезпечення	66
3.2.1 Загальні положення та схема автоматизації.....	66
3.2.2 Структура комплексу технічних засобів	67
3.2.3 Опис автоматизованого робочого місця.....	70
3.2.4 Схема мережі передачі даних.....	71

3.3. Програмне забезпечення.....	73
3.3.1 Структура програмного забезпечення	73
3.3.2 Системне програмне забезпечення	74
3.3.3 Прикладне програмне забезпечення	76
3.3.4 Програмна документація.....	78
3.4. Результати реалізації інформаційної підсистеми.....	80
ВИСНОВКИ.....	97
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	99
ДОДАТКИ	103
Д О Д А Т О К А	103
Д О Д А Т О К Б	105
Д О Д А Т О К В	111
Д О Д А Т О К Г	112

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ ВЕЛИЧИН І ТЕРМІНІВ

API - інтерфейс прикладного програмування

CSV - значення, розділені комами

DDL - мова визначення даних

ECT - температура охолоджувальної рідини

FK - зовнішній ключ

HTTP - протокол передачі гіпертексту

IAT - температура впускного повітря

IoT (IoT) - інтернет речей

JSON - нотація об'єктів JavaScript

MAF - датчик масової витрати повітря

PK - первинний ключ

RPM - оберти за хвилину

SQL - мова структурованих запитів

UI - інтерфейс користувача

UML - уніфікована мова моделювання

USB - універсальна послідовна шина

VPN - віртуальна приватна мережа

VSS - датчик швидкості транспортного засобу

Wi-Fi - технологія бездротової локальної мережі

APM - автоматизоване робоче місце

ДБЖ - джерело безперебійного живлення

ДВЗ - двигун внутрішнього згорання

СКБД - система керування базами даних

ВСТУП

У сучасному світі ефективність експлуатації транспортних засобів, зокрема мотоциклів, значною мірою залежить від рівня автоматизації моніторингу та аналізу технічних параметрів двигуна. Особливо це стосується чотиритактних двигунів, які завдяки своїй конструкції потребують точного налаштування й регулярного обслуговування. Низька якість палива, перевантаження, перегрівання або зміни атмосферного тиску можуть призвести до зниження ефективності роботи двигуна, збільшення витрат пального і навіть до критичних поломок. У зв'язку з цим, виникає потреба в створенні автоматизованої інформаційної системи, яка дозволить проводити оперативний аналіз і оптимізацію параметрів двигуна на основі фактичних сенсорних даних.

Розробка інформаційної системи, оптимізацію параметрів чотиритактного двигуна мотоцикла набуває нового значення, враховуючи підвищений інтерес до малогабаритного транспорту в Україні, особливо в сільських районах, службах доставки, туристичних фірмах і кур'єрських службах. На тлі воєнних дій та економічної нестабільності, мотоцикл став ключовим засобом пересування для багатьох громадян, що зумовлює необхідність доступних рішень для технічного обслуговування, діагностики та збільшення терміну служби двигуна.

Проблематика оптимізації роботи двигунів внутрішнього згоряння широко висвітлювалась у працях таких українських і зарубіжних дослідників, як Л. І. Мазуренко [27], О. Є. Гамола [28], Д. С. Мінчев [29], які у своїх роботах приділяли увагу математичному моделюванню режимів роботи двигуна та аналізу технічних характеристик. З боку інформаційних технологій важливий внесок зробили фахівці в галузі систем моніторингу (наприклад, праці Крейга Сміта щодо розробки IoT-рішень, а також досвід впровадження Arduino/ESP-плат у реальному часі) [30]. Однак, недостатньо дослідженою залишається

інтеграція цих технологій у доступну мобільну інформаційну систему для кінцевих користувачів, що обмежує масштабування подібних рішень.

Мета дипломної роботи полягає в розробці інформаційної системи з визначення оптимальних параметрів чотирьохтактного двигуна.

Завданнями випускного бакалаврського проекту є: аналіз предметної області та технічних вимог, розробка бази даних для збереження параметрів оптимізації, реалізація API для взаємодії між БД та мобільним клієнтом, створення мобільного застосунку, забезпечення зручної візуалізації даних та розрахунків ефективності.

Предметом дослідження є обробка, аналіз та оптимізація параметрів чотиритактного двигуна мотоцикла на основі сенсорних даних з використанням сучасних інформаційних технологій.

Об'єктом дослідження є розроблення інформаційної системи.

У роботі застосовано комплекс методів: системного аналізу, структурного моделювання (UML), об'єктно - орієнтованого підходу до проектування, а також методи комп'ютерного моделювання, REST - архітектури для API та методи нормалізації баз даних.

Результати дослідження мають практичну вартість для приватних осіб, сервісних центрів та навчальних закладів. Теоретична основа проекту спирається на комбінацію IoT (Internet of Things), мобільних технологій та методів оптимізації. Методологічно система може бути пристосована до інших типів двигунів або транспортних засобів, завдяки відкритій структурі коду та модульності.

Інформаційна база сформована на основі технічної документації до Arduino UNO, Raspberry Pi 5, сенсорних модулів (датчик тиску, температури, обертів тощо), офіційної документації Flask, Flutter, MariaDB. Також використовувались результати академічних досліджень, офіційні форуми спільнот Arduino та Raspberry, приклади open source-проектів, аналітичні публікації у сфері транспортної інженерії, а також технічні статті про оптимізацію двигунів.

Результати кваліфікаційної бакалаврської роботи були апробовані шляхом представлення в збірнику матеріалів «VI Міжнародної науково – практичній конференції молодих вчених, аспірантів і студентів» з темою роботи: «Використання нейро – нечітких системи у визначенні оптимальних параметрів чоторитактного двигуна: як машина відчуває».

РОЗДІЛ 1

ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1. Характеристика предметної галузі та об'єкта дослідження

Предметною галуззю даного дослідження є процеси, пов'язані з функціонуванням, діагностикою та оптимізацією параметрів чотиритактних двигунів внутрішнього згоряння, зокрема тих, що використовуються у мотоциклах. У сучасному світі ефективність роботи двигунів є ключовим фактором для забезпечення високої надійності транспортних засобів, економії пального та зменшення негативного впливу на довкілля. Удосконалення робочих характеристик чотиритактних двигунів [1] сприяє зменшенню споживання палива та покращенню динаміки руху.

У сучасному транспортному секторі, де економічна ефективність, екологічна складова та надійність техніки напряму залежать від продуктивності двигуна, першочергове значення набуває завдання точного налаштування та оптимізації. Удосконалення робочих показників двигунів, зокрема зменшення споживання пального, збільшення потужності та крутного моменту, а також поліпшення динаміки руху, постійно перебуває у фокусі інженерних досліджень та практичних розробок. Зростаюча складність конструкцій сучасних двигунів, впровадження електронних систем управління та все суворіші екологічні норми, зменшують ефективність та збільшують трудовитрати на традиційні методи налаштування та діагностики «вручну».

У цьому контексті ключова роль переходить до сучасних інформаційних технологій та систем [2]. Вони забезпечують інструменти для автоматизованого збору великих обсягів даних з різноманітних датчиків двигуна, їх всебічної обробки, детального аналізу та інтелектуальної інтерпретації. Ефективне застосування інформаційних технологій стає

критичним для поліпшення якості та прискорення аналізу, забезпечення точності діагностики, а також визначення оптимальних режимів функціонування двигунів в різних умовах експлуатації. Розробка та впровадження спеціалізованих інформаційних систем, здатних не тільки аналізувати поточний стан, але й визначати оптимальні параметри для чотиритактного двигуна, є нагальною та актуальною задачею. Ці системи мають потенціал надавати корисні рекомендації щодо оптимізації параметрів практично в реальному часі, значно покращуючи процеси налаштування, технічного обслуговування та діагностики двигунів, що, своєю чергою, сприяє підвищенню якості взаємодії в ланцюгу «двигун - користувач».

Економічний зміст задач, при розв'язанні яких приймаються рішення в предметній галузі, та їх вплив:

- 1) Підвищення економічності пального: налаштування параметрів функціонування двигуна, таких як пропорції паливно-повітряної суміші та кут випередження запалювання [3] сприяє повноціннішому згоранню палива, що напряду зменшує його витрати на одиницю пройденого шляху. Це приносить суттєву фінансову вигоду для приватних власників автотранспорту, особливо у період коли ціни на енергоресурси нестабільні, так і для організацій з великими автопарками, де загальна економія на пальному може бути значною, зменшуючи експлуатаційні видатки.
- 2) Зменшення витрат на сервісне обслуговування та ремонт: функціонування двигуна в оптимальному режимі знижує навантаження на його складові та сповільнює процеси зносу. Інформаційна система, яка забезпечує точну діагностику та вчасне виявлення відхилень роботі двигуна.
- 3) Покращення експлуатаційних властивостей та конкуренто спроможності: оптимальні налаштування дозволяють повною мірою реалізувати потенціал двигуна, поліпшуючи його потужність. Для комерційного транспорту це може означати зростання продуктивності, для спортивної техніки - досягнення кращих

результатів, а для виробників - підвищення конкурентної здатності їх продукції на ринку.

Об'єктом дослідження є процес розроблення інформаційної системи, призначеної для аналізу параметрів роботи чотиритактного двигуна та їх оптимізації. Ця система спрямована на підвищення ефективності налаштування та діагностики двигунів через збір, обробку та аналіз даних.

Перелік об'єктів (галузей, підприємств, організацій, установ, бізнесу, технологічних об'єктів тощо) предметної галузі (таблиця 1.1):

Таблиця 1.1 – перелік об'єктів предметної галузі

Категорія	Об'єкти:
Технологічні об'єкти	Чотиритактні двигуни внутрішнього згоряння
Підприємства/організації	<ol style="list-style-type: none"> 1) Станції технічного обслуговування мотоциклів та автомобілів 2) Виробники мотоциклів та двигунів 3) Науково-дослідні установи, що займаються двигунобудуванням 4) Тюнінгові ательє та майстерні
Апаратні компоненти системи	<ol style="list-style-type: none"> 1) Мікроконтролери (наприклад, Arduino) 2) Мікрокомп'ютери (наприклад, Raspberry Pi 5) 3) Датчики (температури, тиску, обертів тощо)
Програмні компоненти системи	<ol style="list-style-type: none"> 1) Інформаційна система з визначення оптимальних параметрів 2) Модулі: збору даних, аналізу, оптимізації, звітування, користувацького інтерфейсу
Користувачі системи	<ol style="list-style-type: none"> 1) Інженери 2) Користувачі 3) Розробники 4) Адміністратори системи

Джерело: сформовано автором на основі виконаного дослідження

Також в проєкті було наведено опис рішень, що приймаються в предметній галузі (таблиця 1.2):

Таблиця 1.2 – опис рішень

№	Проблемні ситуації	Цілі розв'язання
1	Неоптимальна робота чотиритактного двигуна проявляється у наступних ознаках: підвищене споживання пального, яке перевищує вказані виробником показники або ж особисті очікування водія.	Створення інформаційної системи, що зможе автоматизувати збір повних даних про функціонування двигуна.
2	Дочасне зношення деталей двигуна внаслідок функціонування за неідеальних температур ,при детонації або з некоректною паливно-повітряною сумішшю.	Розробка алгоритмів для всебічного аналізу отриманих даних, які дозволять виявляти відхилення від стандартних показників.
3	Перевищення допустимих норм токсичності відпрацьованих газів, що призводить до екологічних проблем та можливих штрафних санкцій.	Впровадження математичних моделей та алгоритмів оптимізації для обчислення оптимальних значень ключових параметрів двигуна.
4	Складність та тривалість діагностики прихованих або плаваючих несправностей традиційними методами, що збільшує час простою техніки та вартість обслуговування.	Надання користувачеві наочних звітів, графіків та рекомендацій для прийняття обґрунтованих рішень стосовно налаштування або ремонту двигуна.
5	Відсутність можливості пристосувати параметри мотора до певних робочих умов (скажімо, високогір'я, спортивні змагання, застосування альтернативних видів палива) без використання спеціалізованого обладнання.	Підвищення загальної ефективності використання транспортних засобів завдяки зменшенню витрат палива та обслуговування, одночасно з покращенням динамічних властивостей та надійності.

Джерело: сформовано автором на основі виконаного дослідження

Рішення в межах конкретної галузі приймають різноманітні особи: інженери, користувачі, адміністратори системи та розробники. Кожна з цих категорій має свої пріоритети. Наприклад, власник мотоцикла, який використовує його для щоденних поїздок, може надавати перевагу економії пального та плавній роботі двигуна - інформаційна система дає змогу вибрати відповідну стратегію оптимізації. Спортсмен, навпаки, прагне до максимальної потужності та швидкої реакції двигуна, навіть за умови збільшеного споживання палива. Еколог зосереджується на зменшенні

шкідливих викидів, а механік СТО зацікавлений у швидкому й точному визначенні стандартних налаштувань для типових моделей двигунів. Щоб збалансувати такі, інколи протилежні, інтереси, інформаційна система використовує декілька підходів: багатокритеріальну оптимізацію, що дозволяє встановлювати вагові коефіцієнти для різних цілей або знаходити оптимальні рішення; візуалізацію компромісів - наприклад, графік залежності економії палива від приросту потужності для вибору збалансованої конфігурації; сценарний аналіз («що, якщо?»), який дає можливість моделювати різні налаштування та передбачати результати та збереження й використання профілів оптимізації, які дозволяють оперативно перемикатися між попередньо налаштованими конфігураціями в залежності від умов експлуатації або індивідуальних вподобань користувача.

В дипломній роботі наведено фактори, що впливають на середовище прийняття рішень, характер і ступінь такого впливу (табл.1.3). Таблиця організовує основні аспекти, що визначають ухвалення рішень під час налаштування робочих характеристик чотиритактного двигуна. Для кожної складової вказано найменування, стислий опис сутності, рівень впливу (значний, помірний), характер дії (прямий, непрямий, обмежувальний та інші). Ця структура надає можливість чітко розпізнати, як різноманітні умови - технічні, операційні, екологічні, людські та ресурсні - мають вплив на процес формування технічних рішень.

Таблиця 1.3 – опис факторів

Фактор	Опис	Характер впливу	Ступінь впливу
Технічний стан двигуна	Знос деталей, несправностей датчиків обмежує діапазон оптимальних налаштувань.	Прямий	Високий

Продовження таблиці 1.3

Кліматичні умови	Температура, вологість, атмосферний тиск впливають на процеси сумішоутворення та згоряння.	Опосередкований, потребує адаптації	Середній
Режим навантаження	Міський цикл, траса, спортивний режим вимагають різних налаштувань.	Динамічний	Високий
Якість палива	Невідповідність типу палива може спричинити детонацію або зниження потужності.	Критичний	Середній/Високий
Вимоги виробника	Жорсткі допустимі діапазони, вихід за які веде до втрати гарантії.	Обмежуючий	Високий
Екологічні норми	Обмеження на склад вихлопних газів обмежують параметри роботи двигуна.	Обмежуючий	Середній/Високий
Дані від датчиків	Неточні або зашумлені дані ведуть до помилкових висновків.	Фундаментальний	Високий

Продовження таблиці 1.3

Кваліфікація персоналу	Здатність інтерпретувати дані, розуміння системи, вплив УХ.	Залежить від інтерфейсу	Середній
Часові та ресурсні обмеження	Обмежений час на СТО, вартість обладнання.	Практичний	Середній

Джерело: сформовано автором на основі виконаного дослідження

Для взаємозв'язків рішень було наведено схему для кращого розуміння, яка ілюструє взаємозв'язки рішень в процесі оптимізації параметрів двигуна. Вона показує, як початкове рішення (наприклад, стратегія економії палива) призводить до зміни налаштувань системи, що викликає низку наслідків - як позитивних, так і потенційно ризикованих. У відповідь система або інженер приймає нові рішення, формуючи зворотний зв'язок. Схема підкреслює, що оптимізація - це динамічний процес із багатьма взаємопов'язаними кроками (рис. 1.1).

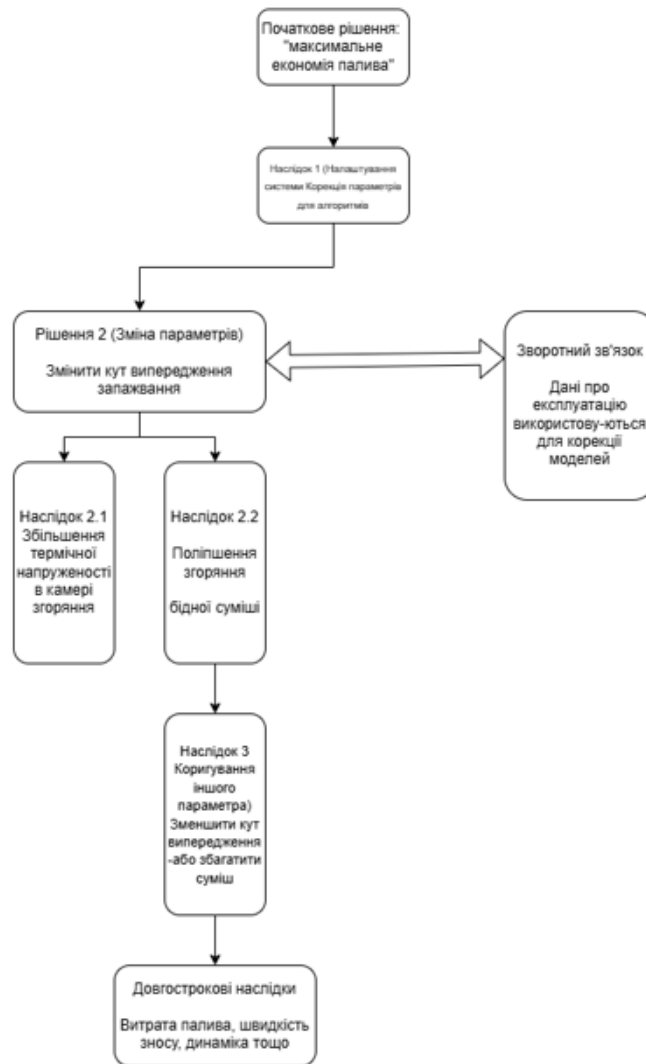


Рисунок 1.1 - Схема взаємозв'язків рішень

Інформація (внутрішня та зовнішня), що необхідна для прийняття обґрунтованих рішень:

- 1) Внутрішня: дані з датчиків двигуна, статична інформація про двигун (VIN-код, модель двигуна, рік випуску, об'єм), історичні дані (логи попередніх діагностик, історія ремонтів) та поточні налаштування системи.

До даних з датчиків двигуна відносяться:

1. Оберти колінчастого валу (RPM): визначає поточний режим роботи [4].
2. Температура охолоджувальної рідини (ECT): важлива для корекції паливоподачі та запалювання.

3. Температура впускного повітря (IAT): впливає на щільність повітря [5].
 4. Масова витрата повітря (MAF Sensor): точне вимірювання кількості повітря, що надходить у циліндри [6].
 5. Короткострокові та довгострокові паливні корекції (Short/Long Term Fuel Trim): показують, як ЕБК адаптує паливоподачу.
 6. Кут випередження запалювання (Ignition Timing Advance).
 7. Тиск палива в рампі.
 8. Швидкість транспортного засобу (VSS).
- 2) Зовнішня: технічні специфікації від виробника, нормативні документи, бази даних діагностичних кодів несправностей, науково - технічна література, дані про властивості палив та відгуки та досвід інших користувачів.

Як висновок у сфері оптимізації налаштувань чотиритактного двигуна постає необхідність ухвалення різних рішень для досягнення конкретної цілі, зниження споживання палива або збільшення показників потужності. Проте, це не є простим завданням, адже слід зважати на низку технічних обмежень, рівень зносу комплектуючих, а також на екологічні стандарти та багато іншого. Ситуації прийняття рішень при налаштуванні чотиритактного двигуна складні через велику кількість критеріїв (наприклад, потужність проти економічності), змінні умови (режим роботи, знос, погода), нестачу точних даних, багато можливих параметрів і варіантів, а також потребу у спеціальних знаннях та інструментах. У таких умовах можуть бути відкриті задачі (наприклад, пошук найкращих налаштувань під гонку чи економічну їзду), закриті задачі (перевірка відповідності після ремонту), кризові ситуації (раптове падіння потужності), а рішення приймаються або за чіткими даними, або з ризиком, або в умовах невизначеності чи навіть за нечітких цілей.

1.2. Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі

Для аналізу було розглянуто кілька подібних систем, що забезпечують базові функції моніторингу та діагностики двигуна. До них відносяться системи для автомобільних двигунів та інші телеметричні системи для обладнання внутрішнього згорання. Були розглянуті наступні приклади систем:

- 1) Bosch Motorsport Data Logger - дозволяє збирати та аналізувати телеметричні дані в режимі реального часу, але не включає персоналізовану оптимізацію двигуна для мотоциклів [7].
- 2) Dynojet Power Vision - призначена для аналізу роботи двигуна, тюнінгу та корекції подачі палива, але з обмеженою інтеграцією з іншими системами та обмеженою кількістю аналізованих параметрів [8].
- 3) Racetrak - платформа, яка може записувати телеметричні дані та налаштування двигуна для різних типів транспортних засобів, але не підтримує специфічні для мотоциклів функції та не має підходу до оптимізації, що базується на зміні навантаження або умов експлуатації [9].

1.2.1 Загальні характеристики обраних систем

Огляд трьох систем збору даних для автоспорту - Bosch Motorsport Data Logger, Dynojet Power Vision і Racetrak - ілюструє їхню універсальність і специфіку застосування.

- 1) Bosch Motorsport Data Logger характеризується високою точністю збору даних і широкими можливостями налаштування та ідеально підходить для професійного автоспорту і тестування автомобілів. Однак його висока вартість і складна інсталяція можуть поставити недосвідчених користувачів у не вигідне становище.
- 2) Dynojet Power Vision в першу чергу призначений для тюнінгу мотоциклів і автомобілів, пропонуючи простоту використання і зручну установку програмного забезпечення. Однак його можливості збору даних обмежені, що може бути недоліком для водіїв, які потребують більш поглибленого аналізу.
- 3) Racepak легко налаштовується і пропонує широкий спектр можливостей для збору даних. Можливості діагностики та аналізу даних зробили його популярним вибором серед гонщиків і тюнерів. Однак потреба в додаткових датчиках може збільшити загальну вартість системи.

Всі три системи надають можливості збору та аналізу даних, необхідні для поліпшення характеристик автомобіля в автоспорті. Вибір між цими системами залежить від конкретних потреб користувача. Професійні команди можуть віддати перевагу Bosch Motorsport за його функціональність, тоді як тюнери та аматори можуть знайти Dynojet або Racepak більш придатними для своїх цілей. Кожна система має свої сильні та слабкі сторони, що свідчить про важливість індивідуального підходу при виборі інструменту збору даних для автоспорту (таблиця 1.4).

Таблиця 1.4 - Загальні характеристики обраних систем

Характеристика	Bosch Motorsport Data Logger	Dynojet Power Vision	Racepak
Призначення	Моніторинг та аналіз даних у motorsport	Оптимізація налаштувань двигуна та моніторинг	Збір даних, моніторинг та аналіз для автоспорту

Продовження таблиці 1.4

Канали збору даних	До 100 і більше	Залежить від конкретної моделі (базово 3-4)	До 50 і більше
Типи даних	Температура, тиск, оберти, прискорення, GPS тощо	Обероти двигуна, тиск наддува, температура масла	Температура, тиск, оберти, прискорення, GPS тощо
Дисплей	Можливість підключення до зовнішніх дисплеїв	Вбудований кольоровий дисплей	Дисплеї для моніторингу в реальному часі
Програмне забезпечення	Bosch Motorsport Software для аналізу даних	Power Core Software для налаштувань	Racepak Data Analysis Software
Способи підключення	CAN, Ethernet, RS232	USB, Bluetooth	CAN, USB, Bluetooth
Налаштування	Гнучке налаштування параметрів	Просте налаштування через програмне забезпечення	Гнучке налаштування під конкретні потреби
Аналіз даних	Візуалізація, створення графіків і звітів	Аналіз роботи двигуна та його налаштувань	Візуалізація, графіки, звіти
Можливість діагностики	Так, зчитування кодів помилок	Так, базова діагностика	Так, зчитування кодів помилок
Переваги	Висока точність, великий обсяг даних	Простота використання, налаштування без комп'ютера	Широкі можливості збору даних, гнучкість
Недоліки	Висока вартість, складність налаштування	Обмежені можливості збору даних	Може вимагати додаткових датчиків
Застосування	Професійний motorsport, тестування автомобілів	Тюнінг мотоциклів і автомобілів	Гоночні автомобілі, мотоцикли, тюнінг

Джерело: сформовано автором на основі виконаного дослідження

1.2.2 Розробка концепції інформаційної системи

На основі аналізу вимог до системи та доступного апаратного забезпечення (Arduino, Raspberry Pi 5, датчики, 3D-модель двигуна), пропонується наступна концепція інформаційної системи для збору, аналізу та оптимізації параметрів двигуна.

Система побудована за принципом клієнт-серверної архітектури:

- 1) Raspberry Pi 5 виступає як серверна частина, що містить базу даних, скрипти обробки даних з Arduino та API для взаємодії з мобільним додатком.
- 2) Arduino збирає дані з підключених датчиків (температура, оберти, вібрації тощо) і передає їх на Raspberry Pi через USB.
- 3) Мобільний додаток встановлюється на телефон і підключається до Raspberry Pi через API, щоб отримувати дані в реальному часі, переглядати історію вимірювань та отримувати рекомендації щодо налаштувань двигуна.

Основні компоненти системи:

- 1) База даних - для зберігання показників з датчиків, історії замірів, профілів налаштувань та рекомендацій.
- 2) Аналітичний модуль – Python - скрипти на Raspberry Pi, що обробляють вхідні дані та формують базові звіти.
- 3) Модуль оптимізації - базується на попередніх результатах та сценаріях, дозволяє оцінювати ефективність конфігурацій.
- 4) API - інтерфейс - реалізується за допомогою Flask [10], забезпечує взаємодію між Raspberry Pi та мобільним додатком.

Таким чином, на основі аналізу технічної літератури, характеристик існуючих рішень та практичного досвіду побудови системи з використанням Raspberry Pi, Arduino та мобільного додатку, можна зробити висновок про доцільність розробки спеціалізованої інформаційної системи, яка відповідатиме особливостям роботи з макетом двигуна. Така система повинна забезпечити зручний збір та аналіз даних із сенсорів, формування рекомендацій щодо оптимізації роботи двигуна, бути простою у впровадженні та обслуговуванні, а також підтримувати мобільний доступ для користувача.

РОЗДІЛ 2

РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1. Аналіз і специфікація вимог до інформаційної системи

Інформаційна система для визначення найкращих параметрів чотиритактного двигуна - це технічно складний програмний продукт, ефективність якого напряму залежить від досягнення бізнес-цілей всіх зацікавлених сторін, що беруть участь у її розробці, впровадженні та використанні.

Внутрішні стейкхолдери, як от розробники, зацікавлені у створенні не просто працездатного рішення, а високоякісної, стабільної та масштабованої інформаційної системи. Їхні бізнес-цілі включають повноцінну розробку функціоналу, забезпечення технічної підтримки, своєчасне оновлення компонентів та покращення системи відповідно до нових вимог. Важливо також забезпечити стабільне фінансування для подальшої підтримки та розширення функціональності.

Керівництво підприємства або наукової установи, яка впроваджує систему, розглядає її як стратегічний інструмент для підвищення ефективності аналізу двигунів, зменшення витрат на технічне обслуговування, а також поліпшення якості діагностики та модернізації. Автоматизація аналізу параметрів двигуна дозволяє оперативно приймати обґрунтовані рішення щодо технічного стану, сприяє зменшенню впливу людського фактора та оптимізації експлуатаційних процесів.

Зовнішні стейкхолдери, наприклад, інженери, механіки, діагностичні спеціалісти та користувачі системи, очікують на просту у використанні, інтуїтивно зрозумілу систему, яка дозволяє швидко вносити або імпортувати параметри двигуна, проводити верифікацію, аналіз та отримувати результати

оптимізації у вигляді наочних звітів. Однією з ключових вимог користувачів є наявність чіткої візуалізації результатів та можливості порівняння кількох варіантів оптимізації.

Регуляторні органи (або внутрішні стандартизаційні підрозділи) ставлять вимоги до точності обробки даних, збереження історії вимірювань, прозорості алгоритмів та відповідності технічним нормам. Крім того, особливу увагу приділяється безпеці обробки інформації та відповідності системи технічним стандартам.

Для досягнення згоди між усіма стейкхолдерами розробляються бізнес-вимоги [11], які враховують потреби кожної сторони. Розробники прагнуть створити стабільну та технічно гнучку архітектуру, керівництво - до зменшення витрат та підвищення якості технічного аналізу, користувачі - до зручного, функціонального та швидкого інструменту, а контролюючі органи - до відповідності вимогам безпеки та стандартам точності.

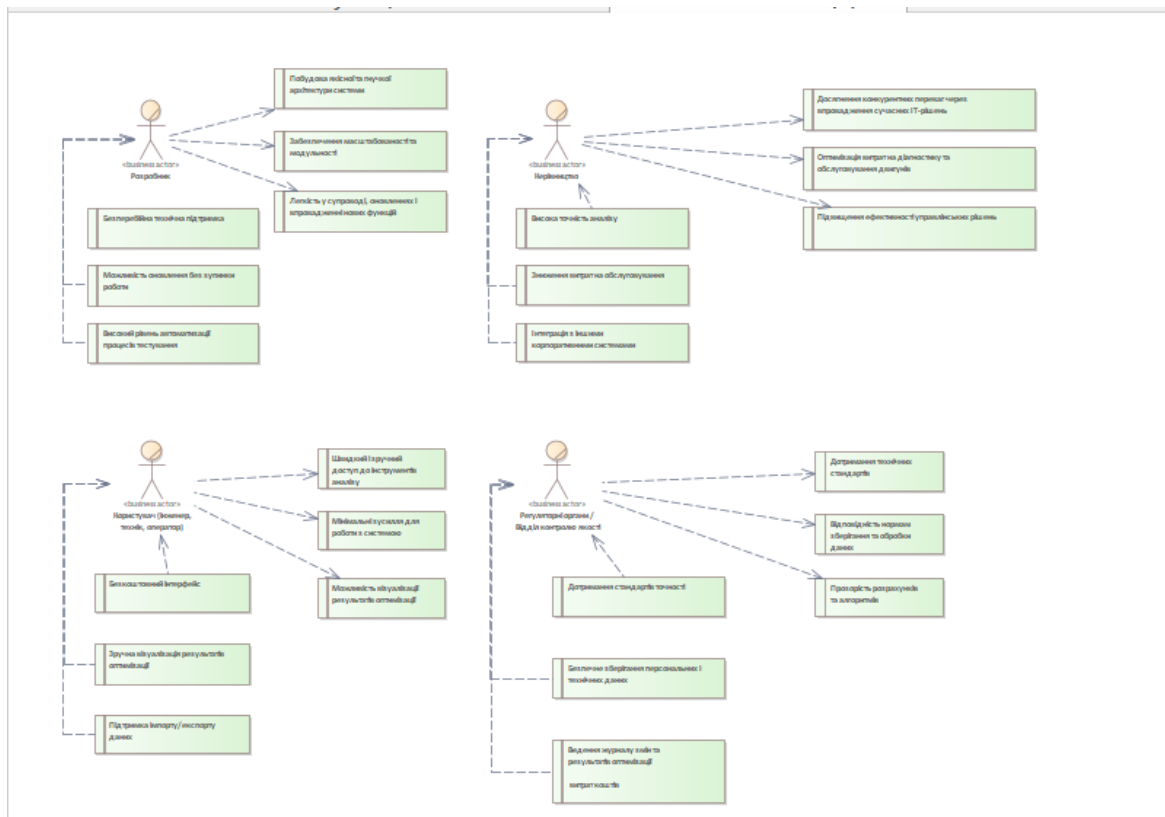


Рисунок 2.1 – Діаграма бізнес вимог і цілей

Ця схема описує ключові вимоги до сучасної IT-системи для оптимізації роботи двигунів. Система розрахована на різних користувачів: інженерів, операторів, а також регуляторні органи, які контролюють якість. Для зручності

вона має простий інтерфейс, зрозумілу візуалізацію результатів у вигляді графіків, а також можливість імпорту та експорту даних для подальшого аналізу.

Важливою частиною є безпека - система повинна надійно зберігати як технічні дані, так і персональну інформацію. Вона має відповідати всім технічним стандартам і нормам, при цьому забезпечуючи високу швидкість роботи алгоритмів.

По суті, діаграма бізнес-вимог і бізнес-цілей - це візуальний інструмент, який показує, чого саме хочуть досягти всі учасники проєкту (керівництво, розробники, користувачі, контролюючі органи), і які вимоги потрібно врахувати, щоб це стало можливим. Вона допомагає чітко зрозуміти очікування кожної сторони, узгодити пріоритети та сфокусуватись на найважливішому. Така діаграма дозволяє ефективно спланувати систему, зменшити зайві витрати, уникнути непорозумінь і створити рішення, яке буде одночасно корисним, зручним і безпечним у роботі з двигунами [12].

Функціональні вимоги - це опис того, що саме повинна робити інформаційна система для задоволення потреб користувачів і досягнення поставлених цілей. Вони формулюють конкретні функції, дії або сервіси, які має виконувати система в межах свого призначення [13].

Основні характеристики функціональних вимог:

- 1) Описують поведінку системи при певних умовах.
- 2) Зосереджені на функціональності: введення, обробка, збереження, вивід даних.
- 3) Зрозумілі для користувачів і розробників — часто є основою для побудови Use Case діаграм.
- 4) Мають бути перевіряльними — тобто такими, які можна протестувати [14].

Функціональні вимоги до інформаційної системи оптимізації параметрів чотирьохтактного двигуна деталізують основні підсистеми та ключові дії, які система повинна виконувати для забезпечення ефективної, точної та зручної роботи з параметрами двигуна.

Система охоплює весь цикл роботи з даними, починаючи від їх введення й завершуючи отриманням оптимізованих результатів, враховуючи як технічні особливості, так і потреби аналізу й зручності використання.

- 1) Підсистема «Аналізу Параметрів» фокусується на виявленні нетиповостей у вхідних даних та здійсненні всебічного аналізу параметрів двигуна. До її функціоналу належать: «Аналіз Параметрів Двигуна», «Виявлення аномалій даних» та «Візуалізація вхідних даних». Це дає змогу інженеру глибше розуміти стан двигуна та приймати рішення, базуючись на зібраних даних.
- 2) Підсистема «Введення та Верифікації Даних» відповідає за забезпечення якості та повноти інформації. Вона дозволяє користувачеві вводити параметри вручну або імпортувати їх з файлу, а також гарантує перевірку правильності введених значень через модуль «Верифікація Вхідних Даних». Крім того, передбачено збереження сеансів, що дає можливість уникнути втрати проміжних результатів під час роботи з великим обсягом даних.
- 3) Підсистема «Звітності» відповідає за формування, порівняння та експорт результатів оптимізації. Користувач може створити зведений звіт, експортувати його у форматах PDF або CSV, а також порівняти результати декількох запущених процесів оптимізації для аналізу динаміки й ефективності.
- 4) Підсистема «Оптимізації» є центральною частиною системи, що реалізує алгоритми оптимізації параметрів двигуна. Вона містить модулі вибору стратегії оптимізації, прогнозування результатів, забезпечення зменшення витрат палива та збереження профілів оптимізації для повторного використання. Важливою складовою є перевірка відповідності отриманих результатів специфікаціям виробника.

Загалом, архітектура системи забезпечує гнучкість використання, дотримується стандартів точності та безпеки та дає можливість адаптувати процеси під конкретні потреби користувача. Більш детальну інформацію щодо

функціональних вимог можна знайти на діаграмі, зображеній на рис. 2.2, а також у специфікаціях вимог, створених за допомогою Specification Manager в середовищі Enterprise Architect (додаток А).

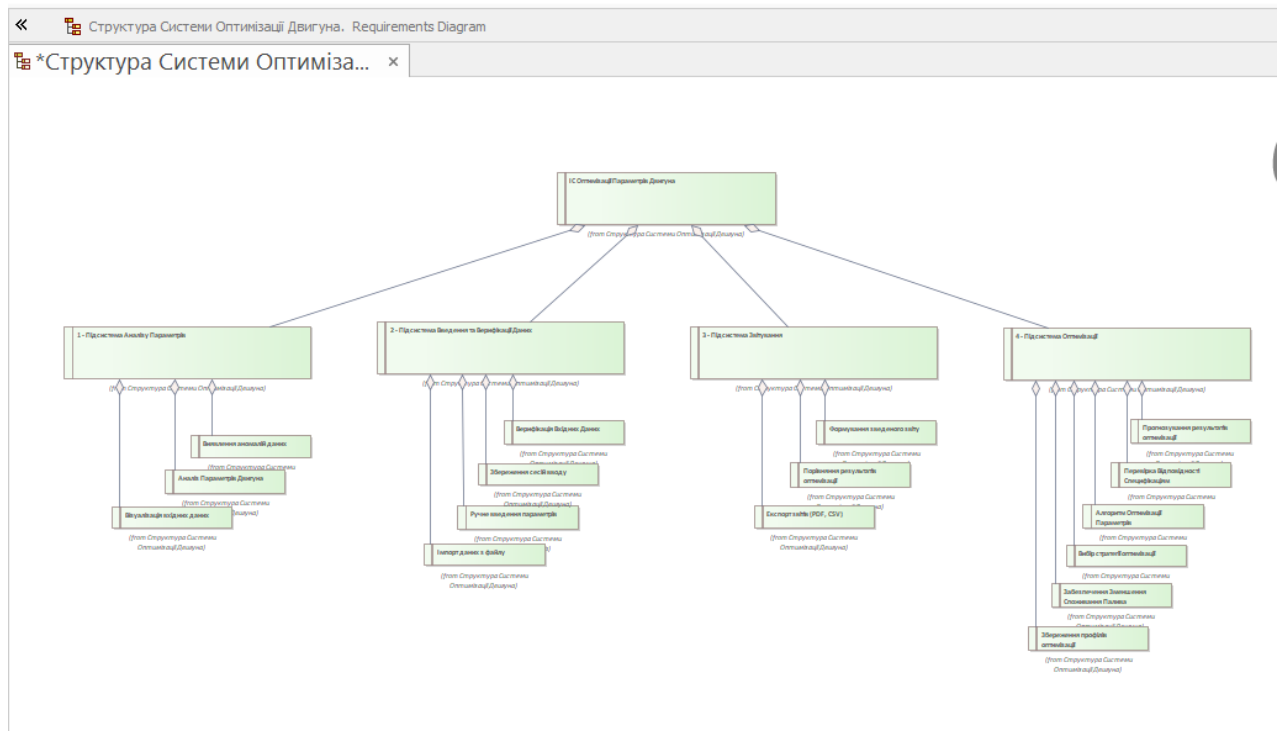


Рисунок 2.2 – Діаграма функціональних вимог

Використання Specification Manager у середовищі Enterprise Architect дало змогу чітко структурувати та класифікувати функціональні вимоги до інформаційної системи з визначення оптимальних параметрів чотирьохтактного двигуна. Всі вимоги згруповано по підсистемах, що відображає логічну архітектуру проєкту: введення та верифікація даних, аналіз, оптимізація і звітування.

Нефункціональні вимоги відіграють вирішальну роль, визначаючи важливі аспекти системи, що впливають на її функціонування, захищеність, швидкість роботи та зручність для користувачів. Ці вимоги зосереджуються не на конкретних діях системи, а на якісних показниках її діяльності [15].

- 1) Підсистема «Зручності» зосереджується на тому, щоб взаємодія з ІС була інтуїтивною та комфортною для інженерів. Сюди входять вимоги щодо українського інтерфейсу, логічної навігації та мінімальної кількості кліків для виконання основних операцій.

Інтуїтивно зрозумілий дизайн скорочує час навчання персоналу та зменшує ймовірність помилок під час роботи.

- 2) Підсистема «Безпеки» гарантує захист даних і транзакцій. Вона передбачає шифрування облікових записів, передавання даних тільки безпечними протоколами (HTTPS, SSH) та обов'язкову ролеву авторизацію користувачів. Це забезпечує відповідність законодавчим вимогам щодо персональних і технічних даних, а також сприяє підвищенню довіри до системи.
- 3) Підсистема «Продуктивності» визначає обмеження на тривалість ключових операцій. Алгоритм оптимізації повинен виконуватися не більше ніж за 5 секунд, а час відгуку інтерфейсу користувача (UI) - не перевищувати 0,5 секунди у 90% випадків. Такі показники дають змогу інтегрувати систему у виробничі процеси без затримок і простоїв.
- 4) Підсистема «Надійності» відповідає за стабільну роботу ПЗ. Система повинна забезпечувати 99% доступності протягом доби, а у разі збою - відновлювати сесію оптимізації з останнього збереженого етапу. Щоденне резервне копіювання даних мінімізує ризик їх втрати.
- 5) Підсистема «Масштабованості» гарантує можливість додавати нові алгоритми або апаратні модулі без зміни ядра програми. Це дозволяє розширювати функціонал разом із появою нових типів двигунів чи сенсорів.

Загалом сукупність нефункціональних вимог формує якісну оболонку навколо функціональних можливостей: підвищує швидкість роботи, гарантує безпеку, спрощує роботу користувачів і забезпечує довгострокову придатність системи до розвитку. Детальну ієрархію можна переглянути на діаграмі нефункціональних вимог (рис. 2.3), а їх текстові специфікації – у Specification Manager Enterprise Architect (додаток А).

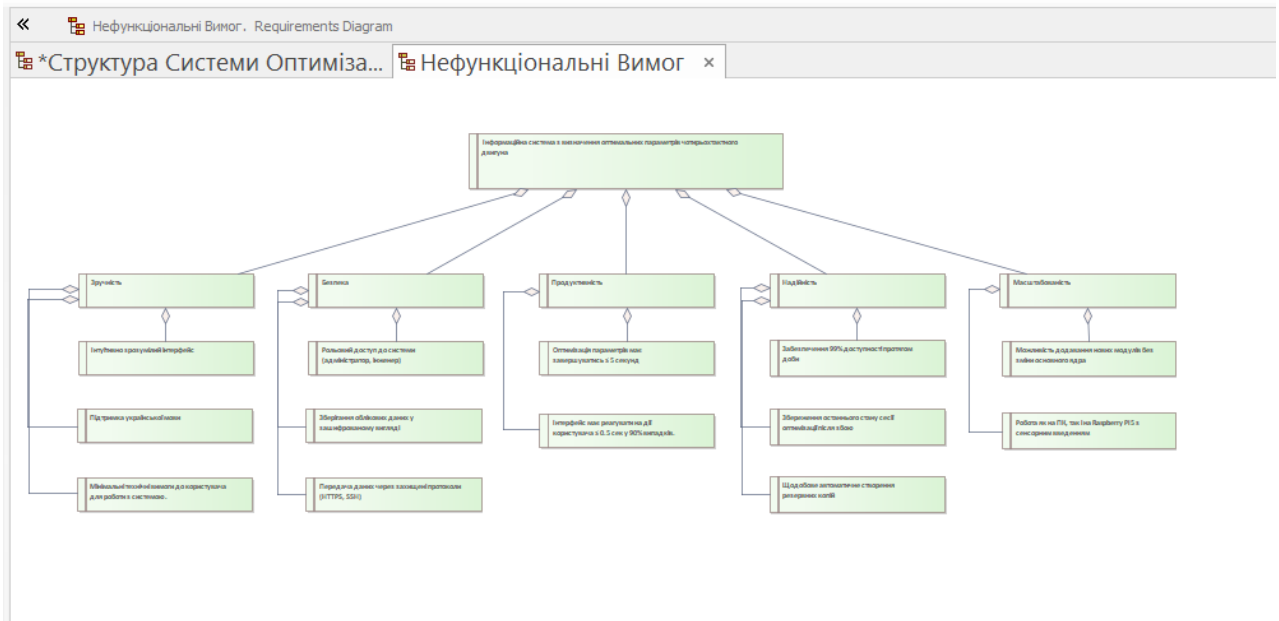


Рисунок 2.3 – Діаграма нефункціональних вимог

Використання «Specification Manager» в Enterprise Architect дозволило ретельно організувати та впорядкувати нефункціональні вимоги до інформаційної системи, зосереджені на налаштуванні найкращих характеристик чотиритактного двигуна. Всі вимоги розбиті по необхідних підсистемах, що підкреслює найважливіші риси якості системи: зручність використання, безпеку, швидкість роботи, надійність, можливість розширення.

2.2. Постановка задачі

2.2.1. Опис постановки задачі

Спроектвана інформаційна система створена для визначення найкращих параметрів чотиритактного двигуна мотоцикла з метою збільшення його продуктивності, зменшення споживання палива та запобігання нештатним ситуаціям. Система здійснює автоматизований збір та

обробку даних в режимі реального часу, використовуючи датчики (температура, тиск, оберти, витрата палива), аналізує ці показники, виявляє відхилення від норми, генерує рекомендації щодо налаштувань двигуна та демонструє результати у зрозумілій формі (графіки, сповіщення, діаграми). Ключовими об'єктами управління є безпосередньо двигун, сенсори, модуль оптимізації та користувачі системи, які взаємодіють з нею для реалізації отриманих порад. Розв'язання задачі відбувається безперервно — моніторинг в реальному часі, а оптимізація - періодично або за бажанням користувача. Згенерована інформація застосовується для щоденного контролю стану двигуна, регулювання його параметрів та планування технічного обслуговування. Автоматичне функціонування може припинитися у разі виходу з ладу датчиків, серйозних збоїв або зупинки двигуна. Система також інтегрується з іншими підсистемами, як-от технічне обслуговування чи безпека, для комплексного підходу до забезпечення справності мотоцикла. Основні функції автоматизовані, тоді як персонал здійснює контроль, оцінку рекомендацій та технічне втручання за потреби.

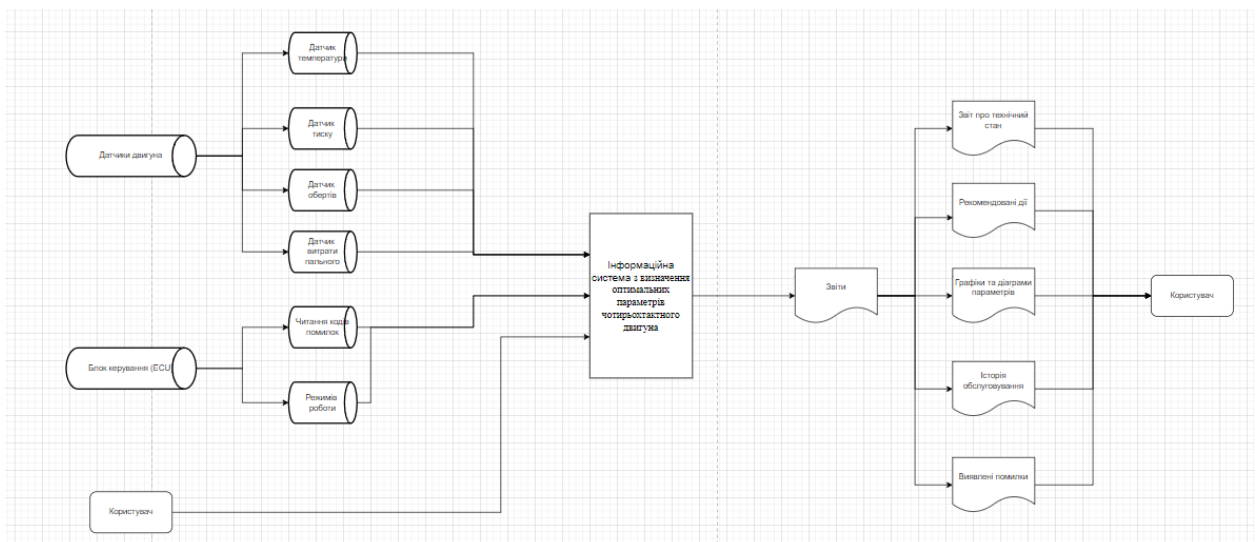


Рисунок 2.4 – Інформаційна модель системи

Призначення вихідної інформації полягає у забезпеченні користувачів актуальними даними про стан роботи двигуна мотоцикла, виявленні несправностей та наданні рекомендацій для оптимізації. Вихідна інформація включає такі типи повідомлень:

- 1) Звіти про стан двигуна – містять показники роботи двигуна у вигляді графіків і діаграм, що відображають основні параметри, такі як температура, тиск, обороти, і витрата палива.
- 2) Рекомендації для оптимізації – містять запропоновані налаштування для покращення ефективності роботи двигуна.
- 3) Критичні повідомлення – інформація про виявлені аномалії або критичні стани, що потребують негайного втручання.
- 4) Графік витрати палива – інформація про витрати палива в різні проміжки часу при різних станів роботи двигуна.
- 5) Повідомлення про профілактичне обслуговування – містить інформацію на основі зібраних даних, про те коли варто провести обслуговування.
- 6) Результати діагностики системи – інформація про виявлені недоліки або несправності модулів та компонентів системи.
- 7) Огляд статистики роботи двигуна – статистичні дані для майбутнього аналізу роботи двигуна та його покращення.

Таблиця 2.1 - перелік і опис вихідних повідомлень

№ з/п	Назва вихідного повідомлення	ID	Форма подання і вимоги до неї	Періодичність видання	Термін видання і допустимий час затримки	Користувач і інформації
1	Звіт про стан двигуна	ZP001	Файл, PDF/Excel, точність 99%	Щодня	В кінці кожного робочого дня	Механіки, технічні спеціалісти
2	Рекомендації для оптимізації	RO002	Файл, текстовий формат, точність 95%	Щотижнево	Після аналізу раз на тиждень	Механіки, технічні спеціалісти

Продовження таблиці 2.1

3	Критичне повідомлення	KP003	Повідомлення, текст/звук, негайне	У разі критичних подій	Негайно після виявлення аномалії	Механіки, технічні спеціалісти, адміністратори
4	Графік витрати палива	GF004	Файл, графічний формат (PNG), точність 98%	Щомісяця	Перший день кожного місяця	Технічні спеціалісти, аналітики
5	Повідомлення про профілактичне обслуговування	PP005	Повідомлення, текст, точність 97%	Щоквартально	За тиждень до запланованого обслуговування	Механіки, адміністратори
6	Результати діагностики системи	RD006	Файл, PDF, текстовий і графічні дані, точність 96%	Після завершення кожного діагностичного	Негайно після завершення діагностики	Механіки, технічні спеціалісти
7	Огляд статистики роботи двигуна	OS007	Файл, текстовий і графічний формат, точність 98%	Щорічно	В кінці кожного року	Механіки, технічні спеціалісти, аналітики

Джерело: сформовано автором на основі виконаного дослідження

Опис структурних одиниць вихідних повідомлень:

- 1) Температура двигуна – входить до звіту про стан двигуна, точність вимірювання $\pm 1^{\circ}\text{C}$, ID: ZP001.
- 2) Тиск у циліндрах – входить до звіту про стан двигуна, точність вимірювання ± 0.1 бар, ID: ZP001.
- 3) Рекомендована зміна налаштувань оборотів – входить до рекомендацій для оптимізації, точність розрахунку 95%, ID: RO002.

- 4) Повідомлення про перегрів двигуна – входить до критичних повідомлень, точність визначення перегріву 98%, ID: KP003.
- 5) Середня витрата палива – входить до графіка витрати палива, точність обчислення 97%, ID: GF004.
- 6) Дата наступного обслуговування – входить до повідомлення про профілактичне обслуговування, ID: PP005.
- 7) Інформація про несправності – входить до результатів діагностики системи, ID: RD006.

Основне призначення вхідних повідомлень — надання актуальних даних про ключові показники роботи двигуна, що дозволяє системі оперативно реагувати на зміни, виявляти аномалії та пропонувати оптимізаційні рішення.

Призначення та способи одержання вхідних повідомлень:

- 1) Дані про температуру двигуна - призначені для моніторингу температурного режиму двигуна з метою запобігання його перегріву та аварійним ситуаціям. Одержуються у реальному часі за допомогою спеціального датчика температури, встановленого на двигуні.
- 2) Дані про тиск у циліндрах - використовуються для контролю рівня тиску в циліндрах, що допомагає підтримувати оптимальні параметри згоряння палива і, відповідно, ефективну роботу двигуна. Інформація надходить від датчиків тиску, встановлених на кожному циліндрі, і оновлюється у реальному часі.
- 3) Дані про обороти двигуна - забезпечують дані про швидкість обертання двигуна, що дозволяє контролювати його робочий режим. Ці дані надходять у реальному часі з датчика обертів, що знаходиться на валу двигуна.
- 4) Дані про витрату палива - призначені для відстеження щоденної витрати палива, що дозволяє системі оцінити економічність роботи двигуна та виявляти можливі проблеми з паливною системою. Одержуються від датчика витрати палива, який передає інформацію раз на добу.

- 5) Інформація про критичні значення - повідомлення про аварійні стани, коли значення основних параметрів двигуна перевищують допустимі межі. Система автоматично генерує ці повідомлення на основі даних, що надходять від датчиків, і зберігає їх в аварійному логі.
- 6) Довідкові дані по оптимальних налаштуваннях - це набір рекомендаційних значень для налаштування параметрів двигуна, які сприяють підвищенню ефективності його роботи. Довідкові дані зберігаються у базі знань системи, і користувач має до них постійний доступ.
- 7) Статистика використання двигуна - надає зведені дані про роботу двигуна за місяць, що дозволяє проводити аналіз ефективності експлуатації двигуна. Ці дані генеруються зовнішньою системою аналізу та надходять у вигляді Excel-файлу щомісяця.

Таблиця 2.2 - перелік і опис вхідних повідомлень

№ з/п	Назва вхідного повідомлення	Ідентифікатор	Форма подання	Термін і частота надходження	Джерело
1	Дані про температуру двигуна	DP001	Файл, текстовий формат	У реальному часі	Датчик температури
2	Дані про тиск у циліндрах	DP002	Файл, текстовий формат	У реальному часі	Датчик тиску
3	Дані про обороти двигуна	DP003	Файл, текстовий формат	У реальному часі	Датчик обертів
4	Дані про витрату палива	DP004	Файл, текстовий формат	Щоденно	Датчик витрати палива

Продовження таблиці 2.2

5	Інформація про критичні значення	KP005	Документ, текстовий формат	Після виявлення критичних значень	Аварійний лог системи
6	Довідкові дані по оптимальних налаштуваннях	DR006	База даних	Постійний доступ	База знань
7	Статистика використання двигуна	SU007	Файл, Excel	Щомісяця	Зовнішня система аналізу роботи двигуна

Джерело: сформовано автором на основі виконаного дослідження

Опис структурних одиниць вхідних повідомлень:

- 1) Температура двигуна – показник температури двигуна, точність вимірювання $\pm 1^{\circ}\text{C}$, джерело: Датчик температури, ID: DP001.
- 2) Тиск у циліндрах – показник тиску у циліндрах, точність вимірювання ± 0.1 бар, джерело: Датчик тиску, ID: DP002.
- 3) Обороти двигуна – показник швидкості обертання двигуна, точність вимірювання ± 10 об/хв, джерело: Датчик обертів, ID: DP003.
- 4) Витрата палива – показник витрати палива за день, точність вимірювання ± 0.05 л, джерело: Датчик витрати палива, ID: DP004.
- 5) Критичне повідомлення про аварійний стан – інформація про критичні значення параметрів, джерело: Аварійний лог системи, ID: KP005.
- 6) Оптимальні налаштування двигуна – довідкові значення параметрів для оптимізації, джерело: База знань, ID: DR006.
- 7) Місячна статистика використання двигуна – аналіз роботи двигуна за місяць, точність 95%, джерело: Зовнішня система аналізу роботи двигуна, ID: SU007.

2.2.2. Алгоритм розв'язання задачі

Представлена таблиця «Перелік масивів використовуваної інформації» містить структуровану інформацію про восьми основних масивах даних інформаційної системи для визначення оптимальних параметрів чотирьохтактного двигуна.

Таблиця 2.3 - перелік масивів використовуваної інформації

№	Масив	Ідентифікатор	Максимальна кількість записів
1	Параметри робочого циклу двигуна	ENGINE_CYCLE_PARAMS	2 000
2	Дані з датчиків температури та тиску	SENSOR_TEMP_PRESS	15 000
3	Характеристики паливної суміші	FUEL_MIX_PROPERTIES	5 000
4	Геометричні параметри циліндрів	CYLINDER_GEOMETRY	500
5	Результати розрахунків оптимізації	OPTIMIZATION_RESULTS	10 000
6	Налаштування системи управління	CONTROL_SYSTEM_CONFIG	1 000

Продовження таблиці 2.3

7	Історія операцій користувача	USER_OPERATIONS_LOG	25 000
8	Бібліотека математичних моделей	MATH_MODELS_LIB	800

Джерело: сформовано автором на основі виконаного дослідження

Таблиця включає 6 масивів результатної інформації, що охоплюють оптимальні параметри двигуна, звіти з аналізу ефективності, графіки характеристик, рекомендації з налаштування, протоколи випробувань та статистичні дані роботи. Структура та оформлення повністю відповідають наданому зразку з відповідним заголовком, нумерацією таблиці та поясненням призначення масивів.

Таблиця 2.4 - перелік масивів результатної інформації

Масив	Ідентифікатор	Максимальна кількість записів
Оптимальні параметри двигуна	OPTIMAL_ENGINE_PARAMS	1 500
Звіти з аналізу ефективності	EFFICIENCY_REPORTS	5 000
Графіки характеристик двигуна	ENGINE_CHARTS	3 000
Рекомендації з налаштування	TUNING_RECOMMENDATIONS	2 500
Протоколи випробувань	TEST_PROTOCOLS	10 000
Статистичні дані роботи	PERFORMANCE_STATISTICS	8 000

Джерело: сформовано автором на основі виконаного дослідження

Визначення основних параметрів двигуна - розробка математичної моделі для розрахунку базових показників, таких як:

Потужність двигуна (P) розраховуватиметься за формулою 2.1:

(2.1)

$$P = \frac{2\pi nM}{60},$$

де n — кількість обертів колінчастого вала за хвилину, M — крутний момент ($H \cdot m$).

ККД двигуна розраховуватиметься за формулою 2.2:

(2.2)

$$\mu = \frac{P_{\text{корисна}}}{P_{\text{затрачена}}} * 100\%,$$

де $P(\text{корисна})$ – корисна потужність, $P(\text{затрачена})$ – потужність, витрачена на тертя, нагрівання, тощо.

Витрата палива (G) розраховуватиметься за формулою 2.3:

(2.3)

$$G = \frac{Q}{H},$$

де Q - кількість теплоти згоряння палива, H - теплотворна здатність палива.

Термодинамічна модель процесу згоряння – використання рівняння теплового балансу:

(2.4)

$$Q_{\text{вп}} = Q_{\text{корисна}} + Q_{\text{втрати}},$$

де $Q_{\text{вп}}$ - теплота згоряння палива, $Q_{\text{корисна}}$ - енергія, що передається на колінчастий вал, $Q_{\text{втрати}}$ - втрати тепла через вихлопи та охолодження.

Алгоритм визначення оптимальних параметрів чотирьохтактного двигуна базується на ітераційному методі оптимізації з використанням математичних моделей термодинаміки робочого циклу. Логіка алгоритму включає послідовні етапи: введення початкових параметрів двигуна (об'єм циліндра, ступінь стиснення, склад паливної суміші), розрахунок термодинамічних характеристик для кожного такту роботи двигуна, обчислення показників ефективності (потужність, крутний момент, питома витрата палива), порівняння отриманих результатів з цільовими значеннями та корекція параметрів у разі необхідності. Формування результатів

здійснюється шляхом збереження оптимальних параметрів у відповідних масивах та генерації звітів з рекомендаціями щодо налаштування двигуна.

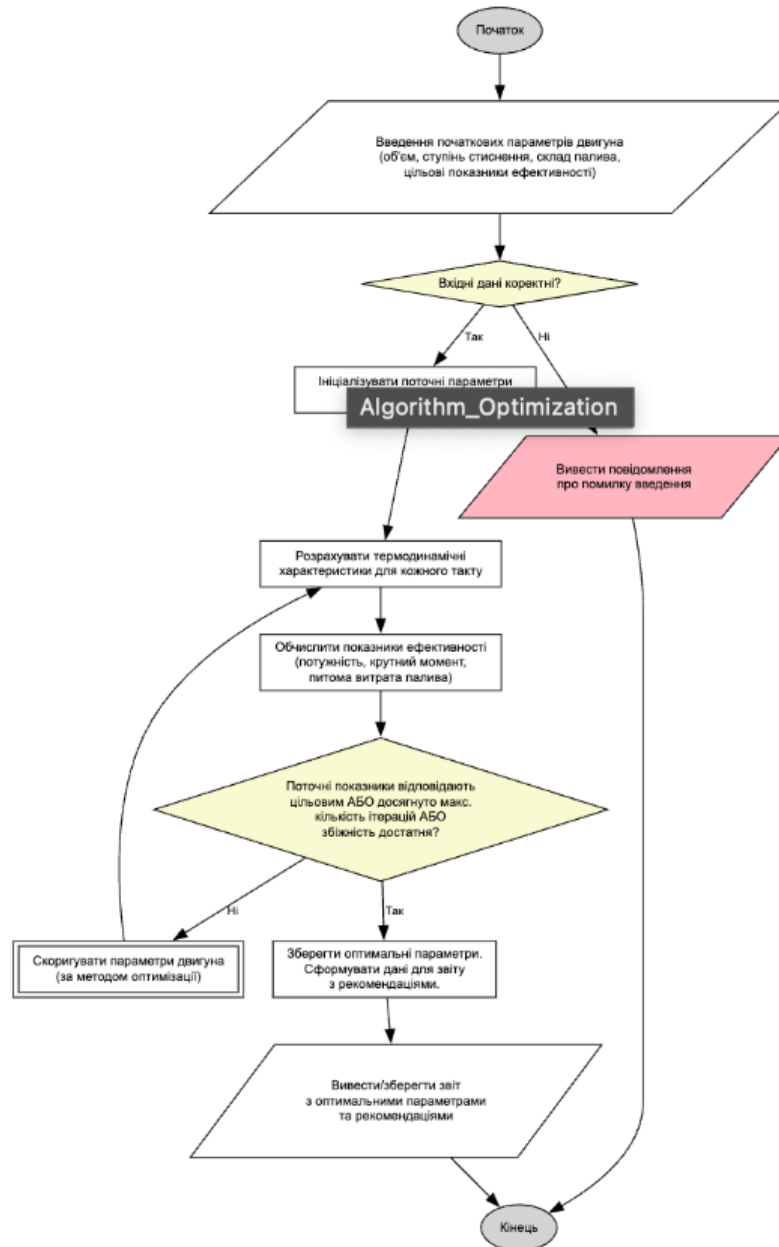


Рисунок 3.5 – Алгоритм роботи

Діаграми послідовностей демонструють порядок взаємодії між об'єктами системи під час реалізації конкретного сценарію [16]. Вони візуалізують, як ініціативи користувача чи інших системних елементів активують ланцюг викликів методів між об'єктами, враховуючи їхню часову послідовність. Ці діаграми є важливими для розуміння логіки бізнес-процесів, розподілу обов'язків між компонентами та моделювання поведінки системи у відповідь на дії користувача (Додаток Б).

2.3.2 Моделювання структури системи

Діаграма класів представляє логічну організацію системи – класи, що формують модель. Це статичний знімок, який демонструє сутності, їхні властивості й функції, а не процес реалізації. На діаграмі класів можливо відобразити взаємозв'язки між класами та інтерфейсами через узагальнення, агрегацію й асоціацію, що є корисним для ілюстрації спадковості, композиції або залежності.

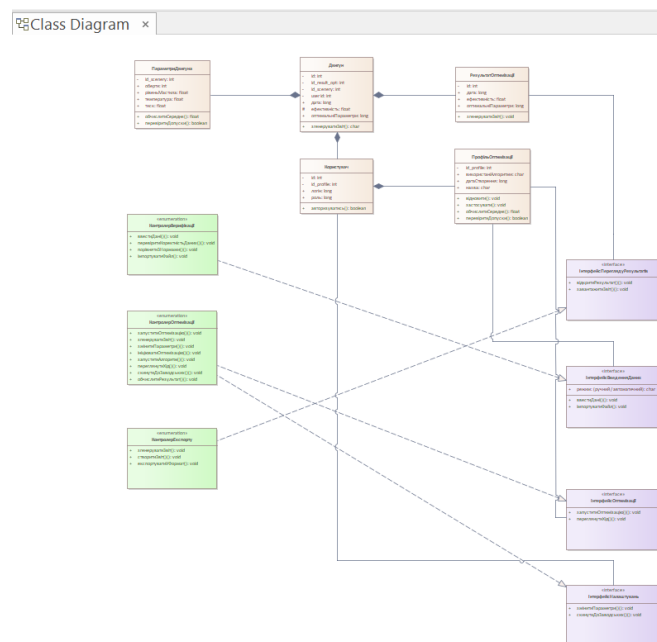


Рисунок 2.7 – Діаграма класів

Блок визначає набір ознак, що використовуються для опису системи, підсистеми, компонента або іншого інженерного об'єкта, що цікавить. Ці

ознаки можуть включати як структурні, так і поведінкові ознаки, такі як властивості, операції та прийоми, що представляють стан системи та поведінку, яку система може демонструвати.

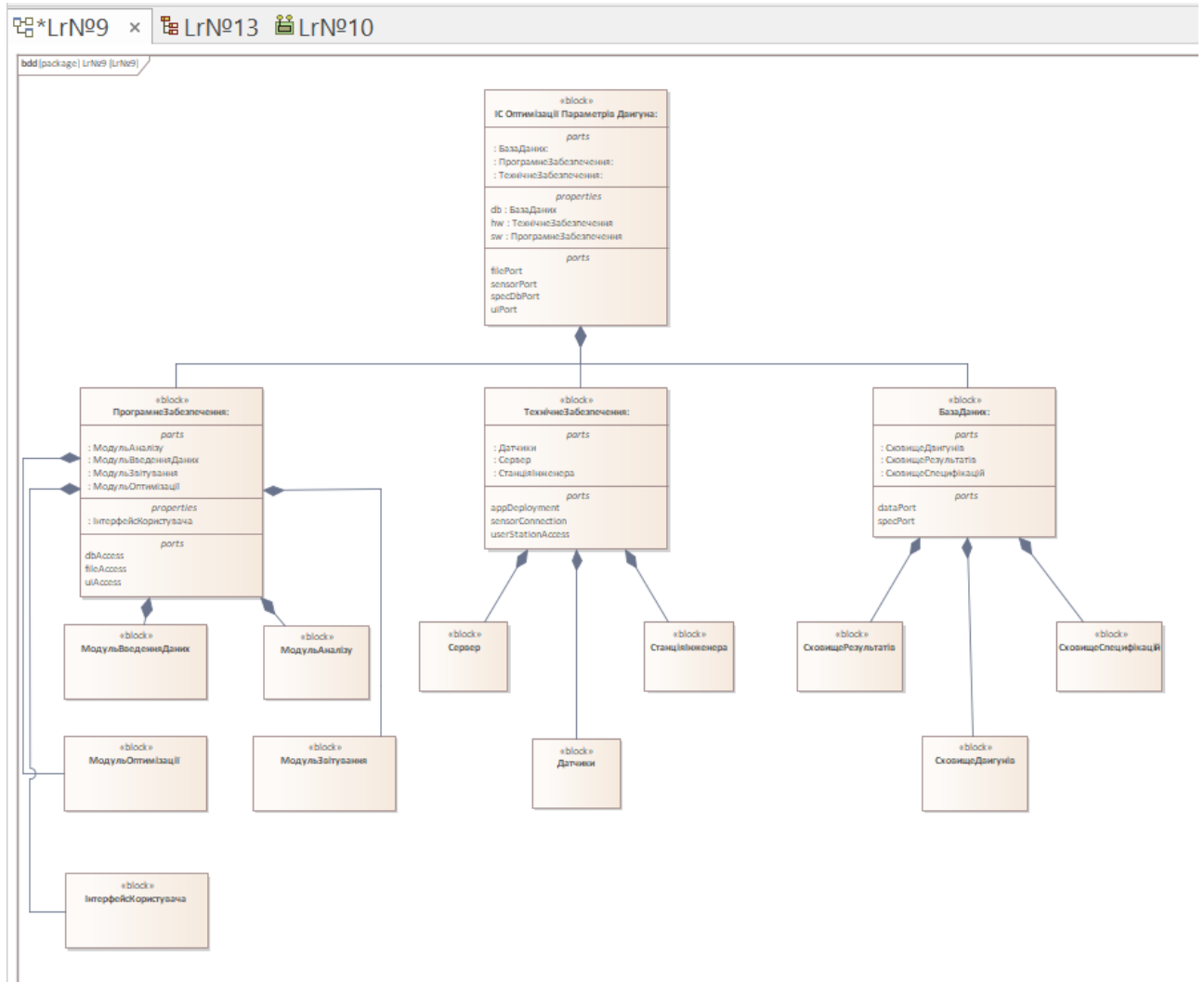


Рисунок 2.8 – Діаграма визначення блоків

Для інформаційної системи з визначення оптимальних параметрів чотиритактного двигуна така діаграма внутрішніх блоків відобразить структуру та взаємодію основних компонентів (рис. 2.9).

Контекстом цієї діаграми є головний системний блок Engine, всередині цього блоку представлені його ключові частини (parts), котрі є екземплярами блоків, визначених на діаграмі BDD. Взаємодія між цими частинами, а також між системою та зовнішнім світом відбувається через порти (ports) – чітко визначені точки взаємодії на межах блоку та його складових. На діаграмі показані ключові порти: порти системи (Engine): uiPort (для контакту з користувачем), filePort (для файлових операцій), sensorPort (для даних із

датчиків), specDbPort (для доступу до специфікацій). Конектори (connectors), зображені лініями, показують, як ці порти з'єднані між собою, утворюючи шляхи взаємодії. Наприклад, конектор між Engine::uiPort та sw::uiAccess показує, як запити від користувача передаються програмному забезпеченню. Конектор між sw::dbAccess та db::dataPort (а також db::specPort, згідно з діаграмою) демонструє, як програмне забезпечення взаємодіє з базою даних для читання та запису інформації (рис. 2.9).

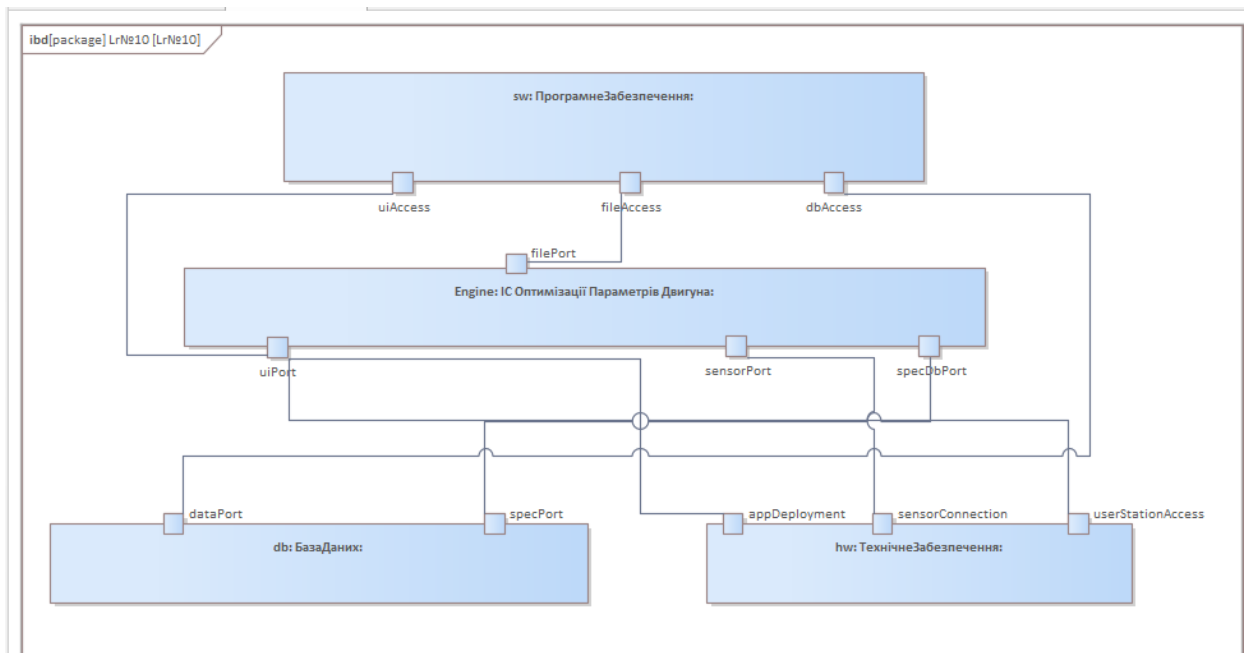


Рисунок 2.9 – Діаграма внутрішніх блоків

2.3.3 Розподіл вимог за компонентами системи

На представленій діаграмі трасування (рис. 2.10) візуалізовано цілісний логічний ланцюг взаємозв'язку між різними шаблями специфікації системи – від функціональних вимог аж до сценаріїв тестування. Така організація дає змогу відслідковувати, як кожна вимога перетворюється на поведінку системи через прецеденти використання (Use Cases), втілюється відповідними класами, пов'язана з інтерфейсами користувача, та, в кінцевому підсумку, перевіряється тестовими кейсами. Ліворуч на діаграмі розміщено функціональні вимоги,

згруповані за підсистемами (аналіз параметрів, верифікація, звітування, оптимізація), які конкретизують очікувану реакцію системи. Кожна з них взаємопов'язана з варіантами використання, що описують конкретні дії, які користувач повинен виконати (наприклад, «Виконати Оптимізацію Параметрів», «Імпортувати Дані Двигуна», «Переглянути Аномалії Даних» та інші). Ці прецеденти представлені у вигляді еліпсів у центральній частині діаграми. Далі, між прецедентами та класами моделі (класи-сутності, контролери, граничні класи) встановлено зв'язки, які демонструють, які саме компоненти забезпечують реалізацію поведінки, визначеної в сценарії. З правого боку діаграми розташовані тестові кейси, які перевіряють реалізацію відповідних сценаріїв (рис. 2.10).

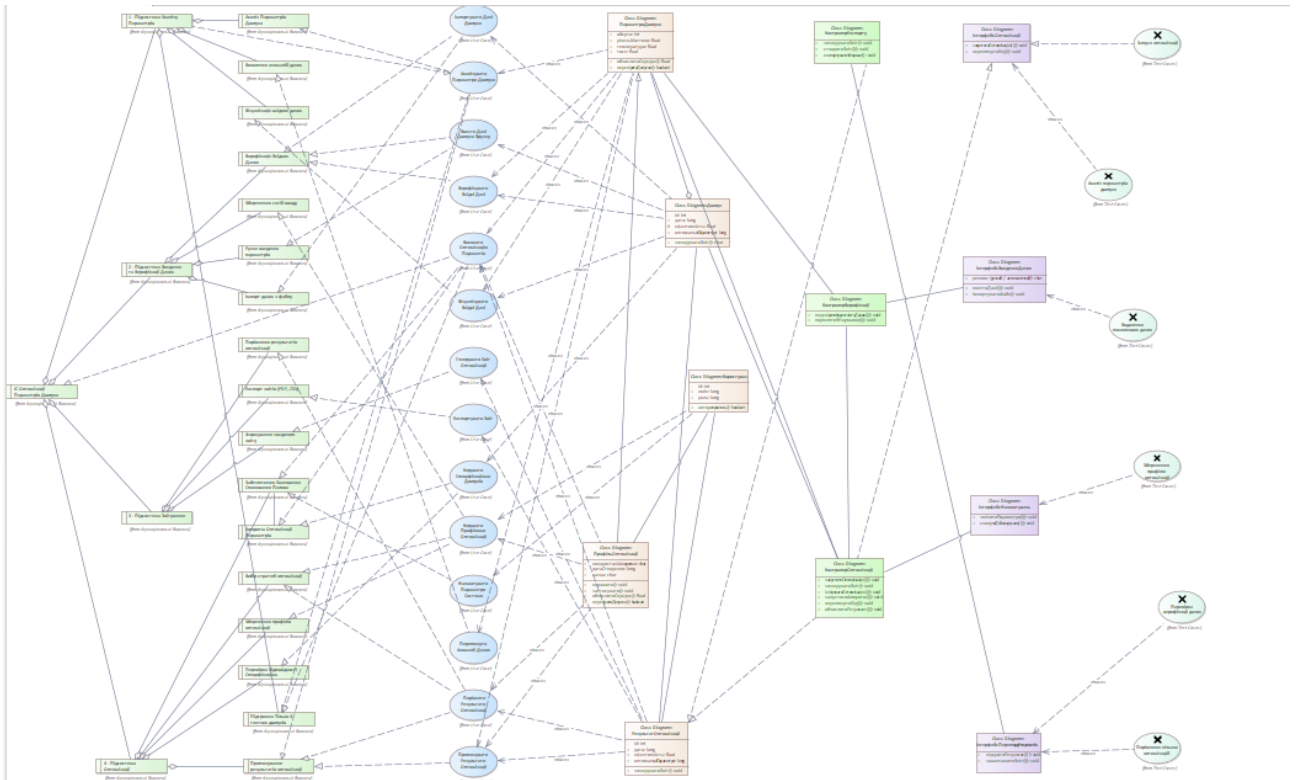


Рисунок 2.10 – Діаграма трасування

Матриці взаємозв'язків, що доповнюють діаграму трасування (див. рис. 2.11 – 2.13):

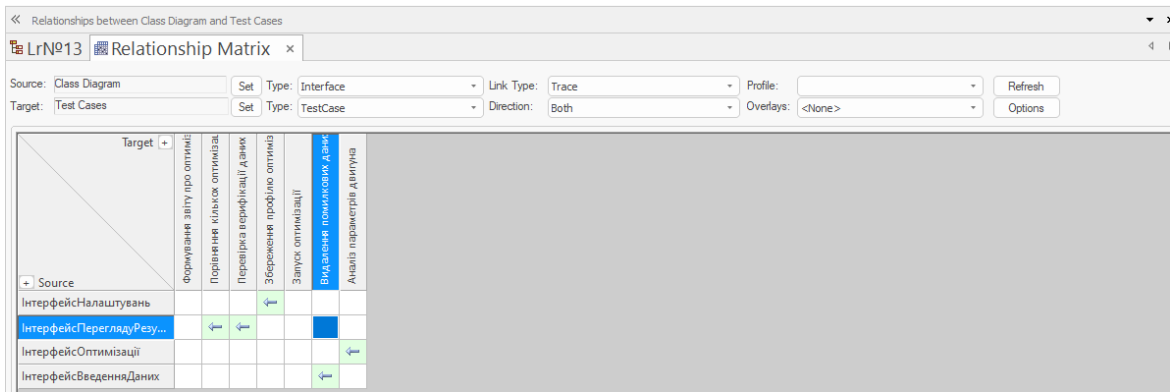


Рисунок 2.11 - Зв'язок між Інтерфейсами та Тест Кейсами

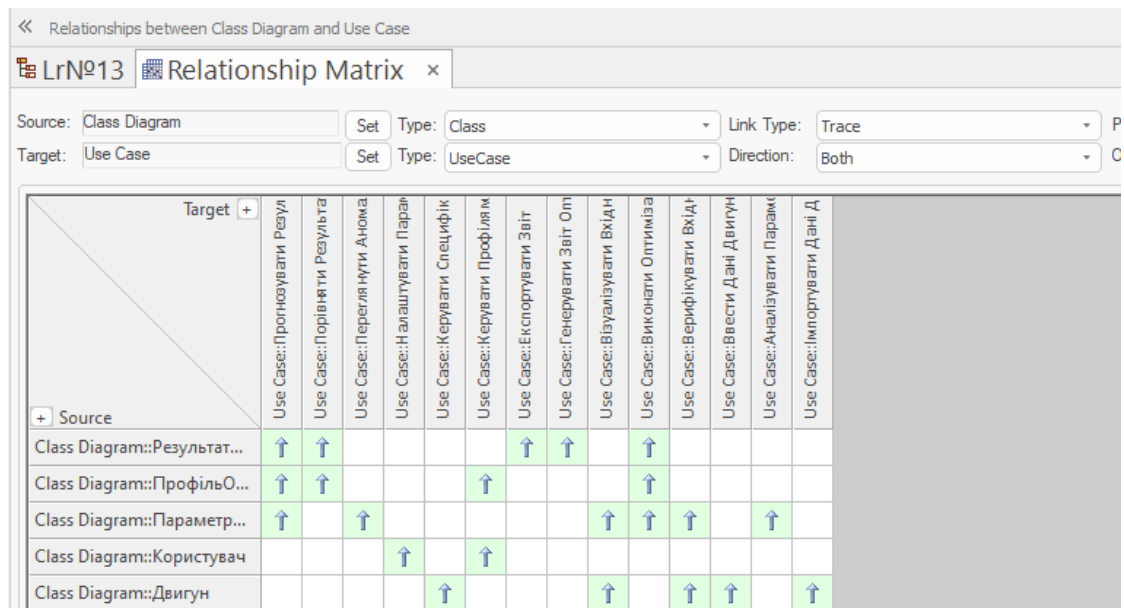


Рисунок 2.12 - Зв'язок між Класами та Use Cases

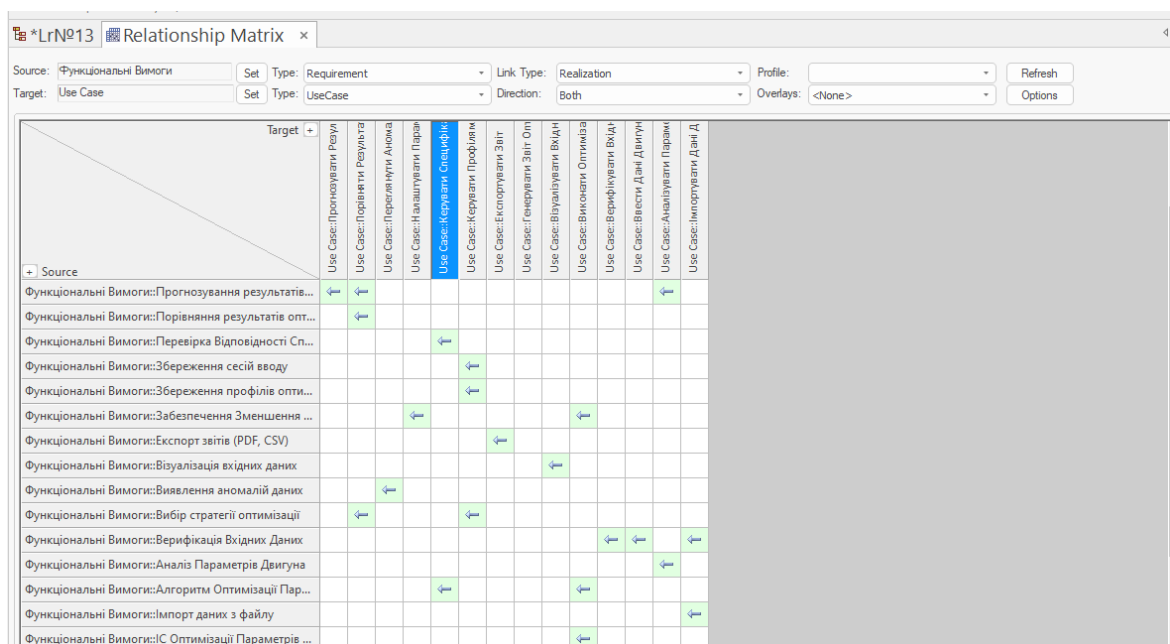


Рисунок 2.13 - Зв'язок між Вимогами та Use Cases

РОЗДІЛ 3

ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ ПІДСИСТЕМИ

Для ефективного функціонування інформаційної системи з визначення оптимальних параметрів чотиритактного двигуна реалізовано чотири ключові забезпечуючі підсистеми: інформаційне, технічне, програмне та організаційне забезпечення.

Інформаційне забезпечення включає структуру вхідних і вихідних даних, таких як параметри двигуна (оберти, температура, тиск, рівень мастила), а також результати оптимізації, які зберігаються у базі даних MariaDB. Технічне забезпечення системи складається з пристроїв збору даних - сенсорів, підключених через Arduino, а також сервера на Raspberry Pi 5, який виконує роль хоста бази даних і API. Програмне забезпечення охоплює серверну частину, реалізовану на Flask (Python), мобільний клієнт (Flutter), а також допоміжні бібліотеки для обміну даними та візуалізації графіків ефективності. Організаційне забезпечення визначає ролі користувачів (адміністратор, інженер, користувач), регламент взаємодії з системою, процеси авторизації, обробки запитів та аналізу отриманих результатів оптимізації.

3.1. Інформаційне забезпечення

3.1.1 Загальна характеристика інформаційного забезпечення

Інформаційне забезпечення (ІЗ) інформаційної системи, призначеної для визначення оптимальних параметрів чотиритактного двигуна мотоцикла, охоплює набір методів, інструментів і даних, що використовуються для збору,

зберігання, обробки та візуалізації робочих параметрів двигуна. ІЗ ґрунтується на фіксації ключових параметрів: кількість обертів (RPM), температура двигуна, тиск мастила, рівень оливи, а також обчислені показники ефективності.

Тип носія інформації - електронний. Дані зберігаються у базі даних MariaDB, розміщеній на вбудованому пристрої Raspberry Pi 5. Для гарантування надійності збереження даних впроваджено механізм автоматичного збереження результатів оптимізації в режимі реального часу через REST API [17].

Обґрунтування вибору СКБД полягає в простоті розгортання MariaDB на Linux-платформах, її продуктивності на малопотужних пристроях, а також підтримці стандарту SQL. Контроль достовірності даних здійснюється на рівні валідації в API та через контролери введення у мобільному застосунку.

Забезпечено перевірку допустимих значень, обмеження та типів даних. Загальна схема ІЗ представлена у вигляді взаємодії (рис. 3.1):

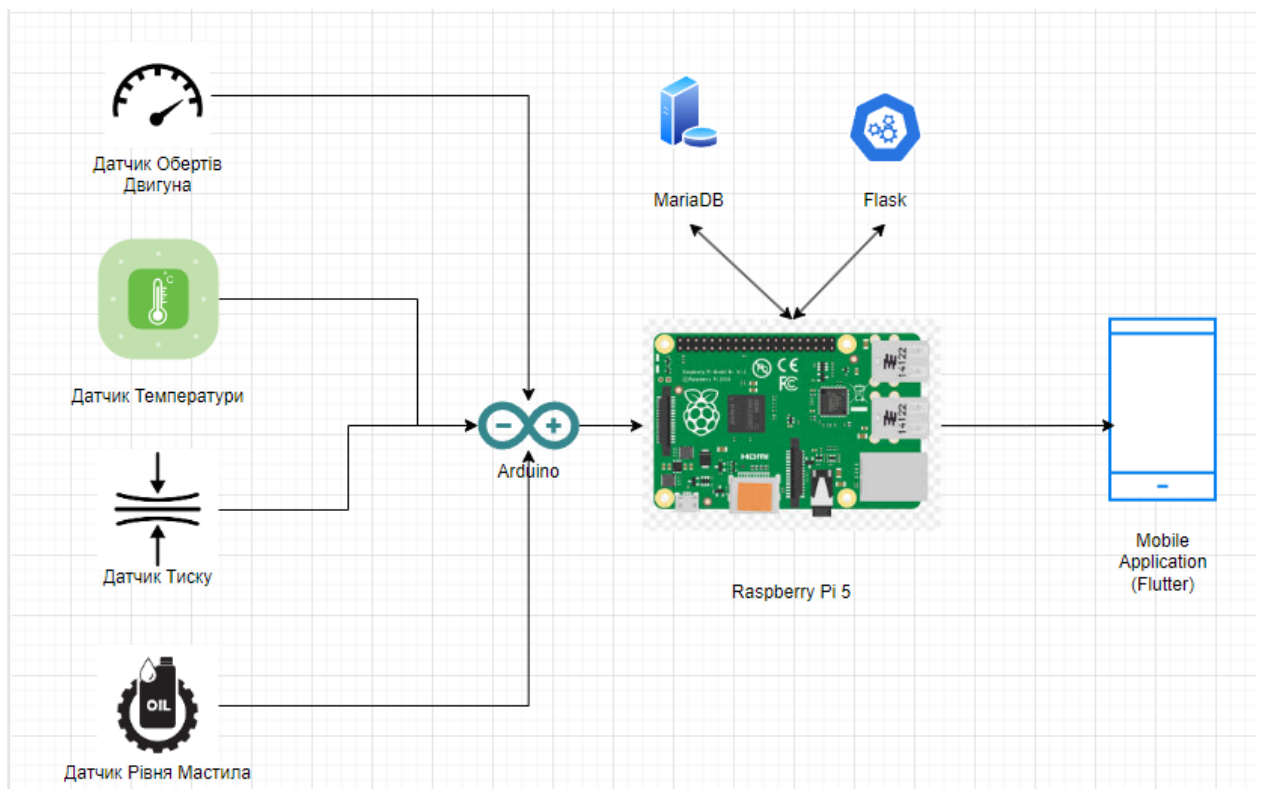


Рисунок 3.1 – Загальна схема ІЗ

3.1.2 Організація збору і передавання первинної інформації

Збір первинної інформації здійснюється за допомогою сенсорних пристроїв, які зчитують фізичні параметри роботи двигуна. Джерелами інформації є:

- 1) сенсори RPM, температури, тиску масла і рівня мастила, з'єднані з Arduino;
- 2) мобільний застосунок, з якого здійснюється оптимізація та ручне введення значень.

Передавання інформації відбувається через протокол HTTP у форматі JSON. Мобільний клієнт надсилає дані до серверного API, який обробляє запити та зберігає їх у базі даних. Передача даних є асинхронною, здійснюється по запиту користувача або автоматично при зборі даних з сенсорів.

Періодичність збору: за ініціативою користувача, або 1 раз під час запуску оптимізації. Носій - оперативна пам'ять пристроїв, а також довготривале зберігання в базі даних MariaDB.

3.1.3 Побудова системи класифікації та кодування

У розробленій інформаційній системі використано систему класифікації об'єктів. Вона гарантує чітку ідентифікацію вхідних даних, користувачів і оптимізаційних звітів. Класифікація базується на функціональних характеристиках, забезпечуючи ефективне зберігання, обробку та аналіз інформації про параметри функціонування двигуна.

Основні класифіковані об'єкти:

- 1) Користувачі системи (роль, логін, ID).

- 2) Оптимізаційні результати (записані показники сенсорів і розрахована ефективність).
- 3) Звіти оптимізації (пов'язані з результатами, мають часову мітку).
- 4) Показники сенсорів (температура, оберти, тиск, рівень мастила).
- 5) Типи ролей користувачів (адміністратор, інженер, користувач).

Найменування класифікаторів:

- 1) `USR_` - класифікатор користувачів (наприклад, `USR_001`, `USR_002`).
- 2) `OPT_` - ідентифікатор результату оптимізації (`OPT_0001`, `OPT_0002`).
- 3) `RPT_` - звіт оптимізації (`RPT_0001`, `RPT_0002`).
- 4) `SNS_` - класифікація сенсорних даних (`SNS_TEMP`, `SNS_RPM`, `SNS_PRS`, `SNS_OIL`).
- 5) `ROLE_` - класифікатор ролей (`ROLE_ADMIN`, `ROLE_ENGINEER`, `ROLE_USER`).

Таблиця 3.1 - Структура та довжина коду

Код	Призначення	Структура	Приклад
<code>USR_XXX</code>	Користувач	7 символів	<code>USR_012</code>
<code>OPT_XXXX</code>	Результат оптимізації	8 символів	<code>OPT_0045</code>
<code>RPT_XXXX</code>	Оптимізаційний звіт	8 символів	<code>RPT_0045</code>
<code>SNS_TEMP</code>	Температура двигуна	8 символів (постійні)	<code>SNS_TEMP</code>
<code>ROLE_ENGINEER</code>	Роль користувача	змінна довжина	<code>ROLE_USER</code>

Джерело: сформовано автором на основі виконаного дослідження

У додатку до документа надані приклади повних класифікаторів для користувачів, типів сенсорних даних та ролей (див. Додаток В).

3.1.4 Проектування форм первинних документів та відеокадрів

У розробленій інформаційній системі з оптимізації параметрів чотиритактного двигуна формується кілька основних документів - як первинних, так і машинограм. Вони забезпечують обробку, аналіз і збереження даних, а також їх подання користувачам.

Первинні документи - це вхідні дані, що надходять у систему від сенсорів, які вимірюють параметри роботи двигуна. Основні поля:

- 1) Ідентифікатор документа: DOC_SENS.
- 2) Назва: дані з сенсорів.
- 3) Джерело: мікроконтролер Arduino (через ESP до Flask API).
- 4) Форма представлення: JSON - структура, наприклад:

```
{
  "user_login": "user2025",
  "rpm": 6100,
  "temperature": 97.5,
  "pressure": 2.0,
  "oil_level": 55.0,
  "timestamp": "2025-05-30T13:24:00"
}
```

Рисунок 3.2 – Код JSON структури представлення даних

- 5) Періодичність надходження: щохвилини (або при ініціації вручну).
- 6) Користувач: система/інженер.

Машинограма - це результат обчислень оптимізації, який генерується автоматично на основі отриманих параметрів. Її мета - представлення ефективності роботи двигуна.

- 1) Ідентифікатор: MR_OPTIM.
- 2) Назва: результат оптимізації.
- 3) Форма представлення: запис у БД optimization_results + optimization_reports.
- 4) Основні поля:
 1. RPM, температура, тиск, мастило.

2. Ефективність (в %).
3. Коментар/резюме.
- 5) Приклад машинограми: $EFF = 85.4\%$. RPM: 6000, Temp: 95.0°C, Pressure: 2.1 bar, Oil: 60.0%.
- 6) Періодичність: за запитом користувача або після симуляції.
- 7) Користувачі: інженери, користувачі.

Відеокадри у класичному вигляді не використовуються, але реалізовано графічне відображення результатів у вигляді динамічного графіка ефективності (через `fl_chart` у Flutter):

- 1) Ідентифікатор: `CH_EFF_USER`.
- 2) Назва: графік ефективності.
- 3) Джерело: дані з таблиці `optimization_reports`.
- 4) Користувач: авторизований користувач.
- 5) Періодичність оновлення: динамічно при запиті на сторінці.

Також результати оптимізації можуть експортуватись у форматі PDF.

3.1.5 Структура інформаційних масивів

Для функціонування інформаційної системи з аналізу та оптимізації параметрів чотиритактного двигуна використовується ряд масивів (таблиць БД), основними з яких є: `users`, `sensor_readings`, `optimization_results`, `optimization_reports`.

Таблиця 3.2 – опис масиву «результати оптимізації»

Найменування	Ідентифікатор у програмі	Умовне позначення у формулах	Формат	Бізнес - правила				Логічні чи семантичні зв'язки
				Первинний/вто	Умова на значення	Обов'язкове	Індексне поле	

Продовження таблиці 3.2

ID	id	id	int	PK	> 0	так	ІНД	-
Логін користувача	user_login	u	varchar(50)	FK	is not null	так	ІДД	-> users.login
Оберти	rpm	n	int	-	1000 - 9000	так	-	-
Температура	temperature	T	float	-	0 - 150	так	-	-
Тиск	pressure	p	float	-	0.5 - 5	так	-	-
Рівень мастила	oil_level	oil	float	-	0 - 100	так	-	-
Ефективність	efficiency	E	float	-	0 - 100	так	-	-> optimization_reports.optimization_id
Час створення	created_at	t	datetime	-	not null	так	--	-

Джерело: сформовано автором на основі виконаного дослідження

Таблиця 3.3 – опис масиву «користувачі»

Найменування	Ідентифікатор у програмі	Умовне позначення у формулах	Формат	Бізнес - правила				Логічні чи семантичні зв'язки
				Первинний/вто	Умова на значення	Обов'язкове	Індексне поле	

Продовження таблиці 3.3

Логін користувача	login	L	VARCHAR(50)	-	Унікальне, не null	так	ІНД	-
Пароль	password	P	VARCHAR(50)	-	Не менше 4 символів	так	-	-
Роль	role	R	VARCHAR(20)	-	admin/user/engineer	так	-	-
ПІБ	full_name	FIO	VARCHAR(100)	-	-	ні	-	-
Телефон	phone	PH	VARCHAR(20)	-	-	ні	-	-

Джерело: сформовано автором на основі виконаного дослідження

Таблиця 3.4 – опис масиву «дані сенсорів»

Найменування	Ідентифікатор у програмі	Умовне позначення у формулах	Формат	Бізнес - правила				Логічні чи семантичні зв'язки
				Первинний/вто	Умова на значення	Обов'язкове	Індексне поле	
Логін користувача	user_login	UL	VARCHAR(50)	FK	існує в users	так	ІНД	-> users.login
Оберти двигуна	rpm	RPM	INT	-	4000 - 7000	так	-	-
Температура	temperature	T	FLOAT	-	85 - 110	так	-	-

Продовження таблиці 3.4

Тиск	oil_level	P	FLOAT	-	1.8 - 2.5	так	-	-
Рівень оливи	phone	OIL	FLOAT	-	40 - 70	так	-	-
Час зчитування	timestamp	TS	DATETIME	-	-	так		

Джерело: сформовано автором на основі виконаного дослідження

Таблиця 3.5 – опис масиву «звіти оптимізації»

Найменування	Ідентифікатор у програмі	Умовне позначення у формулах	Формат	Бізнес - правила				Логічні чи семантичні зв'язки
				Первинний/вто	Умова на значення	Обов'язкове	Індексне поле	
ID звіту	id	ID	INT	PK	AUTO_INCREMENT	так	ІНД	-
ID оптимізації	optimization_id	OID	INT	FK	існує в optimization_results	так	-	-> optimization_results.id
Параметри	parameters	PRM	TEXT	-	-	так	-	-
Резюме	summary	SUM	TEXT	-	-	так	-	-
Час створення	created_at	CRT	DATETIME	-	-	так	-	-

Джерело: сформовано автором на основі виконаного дослідження

3.1.6 Вибір СКБД

Для реалізації інформаційної системи з аналізу та оптимізації параметрів чотиритактного двигуна мотоцикла було обрано систему керування базами даних MariaDB. Основними критеріями вибору стали: сумісність із мовою SQL, відкритий вихідний код (open-source), стабільність роботи на платформі Raspberry Pi, а також підтримка необхідних механізмів безпеки, зокрема контролю доступу, автентифікації користувачів та можливість створення резервних копій.

Переваги вибору MariaDB:

- 1) Продуктивність та надійність при роботі з невеликими обсягами даних.
- 2) Легкість інтеграції з Python (Flask), що забезпечує взаємодію з REST API.
- 3) Гнучкість у проектуванні схеми БД, підтримка зовнішніх ключів та перевірок цілісності.
- 4) Безкоштовне використання без ліцензійних обмежень.

До обмежень, які можуть вплинути на результати проектування, відносяться: обмеження по апаратному забезпеченню (обсяг ОЗП, продуктивність процесора Raspberry Pi), потреба у попередньому налаштуванні прав доступу до бази, відсутність вбудованих механізмів для масштабування в хмарних середовищах. Проте для задач локальної системи оптимізації, що обслуговує обмежене число клієнтів, MariaDB є оптимальним рішенням.

3.1.7 Інфологічна модель бази (сховища) даних

Інфологічна модель бази даних постає як наслідок аналізу предметної галузі та проектування структури збереження інформації, необхідної для підтримки всіх функцій інформаційної системи, призначеної для оптимізації параметрів двигуна. Побудова моделі відбувалася за допомогою інструменту Enterprise Architect, який дозволяє візуалізувати логічні зв'язки між сутностями, атрибутами та ключами бази даних.

- 1) Users - містить інформацію про зареєстрованих користувачів системи (адміністратори, інженери, оператори).
- 2) sensor_readings - зберігає дані з сенсорів (оберти, температура, тиск, рівень мастила).
- 3) optimization_results - результати оптимізації параметрів на основі даних з сенсорів.
- 4) optimization_reports - текстові звіти з описом ефективності та параметрів.

Кожна сутність має первинні ключі (id або login), зовнішні ключі для забезпечення зв'язків (наприклад, user_login у звітах є зовнішнім ключем до users.login). Усі атрибути мають визначені типи даних, обмеження на значення (NOT NULL, CHECK) та індекси для підвищення швидкості вибірки даних (рис. 3.3).

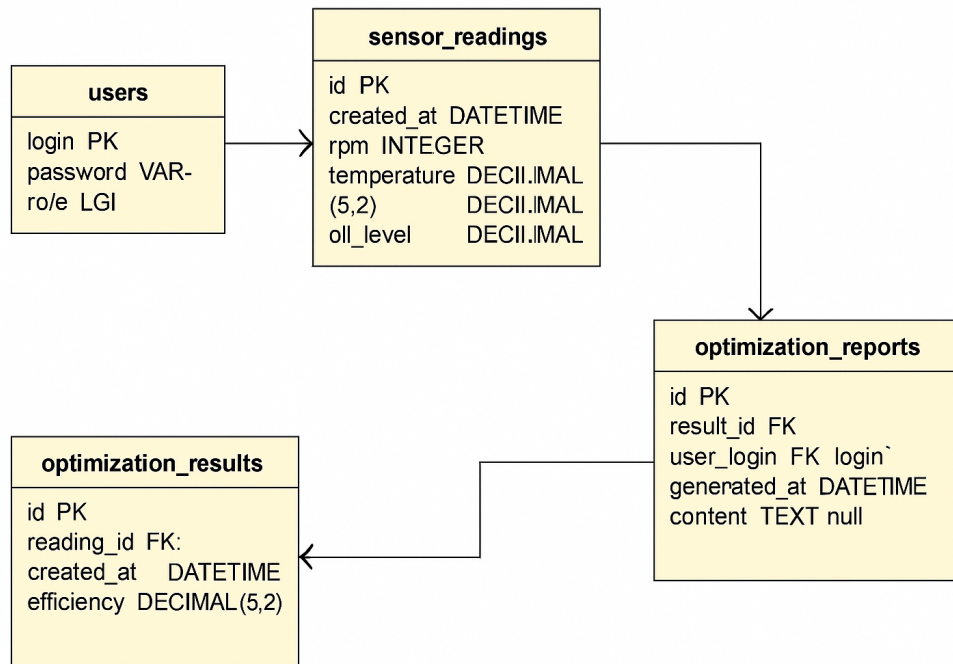


Рисунок 3.3 – Інфологічна модель БД

У рамках цієї бакалаврської роботи не проектувалось окреме сховище даних (OLAP), оскільки функціональність системи зосереджена на оперативному управлінні, оптимізації параметрів та збереженні історії вимірювань.

3.1.8 Даталогічна модель бази (сховища)даних

Даталогічна модель описує фізичну реалізацію структури бази даних у середовищі обраної СКБД MariaDB. У цій моделі враховано специфіку типів даних, зовнішніх ключів, індексів та обмежень на цілісність даних. Модель реалізована засобами DDL (Data Definition Language)[18], а структура створюється за допомогою SQL-команд CREATE TABLE.

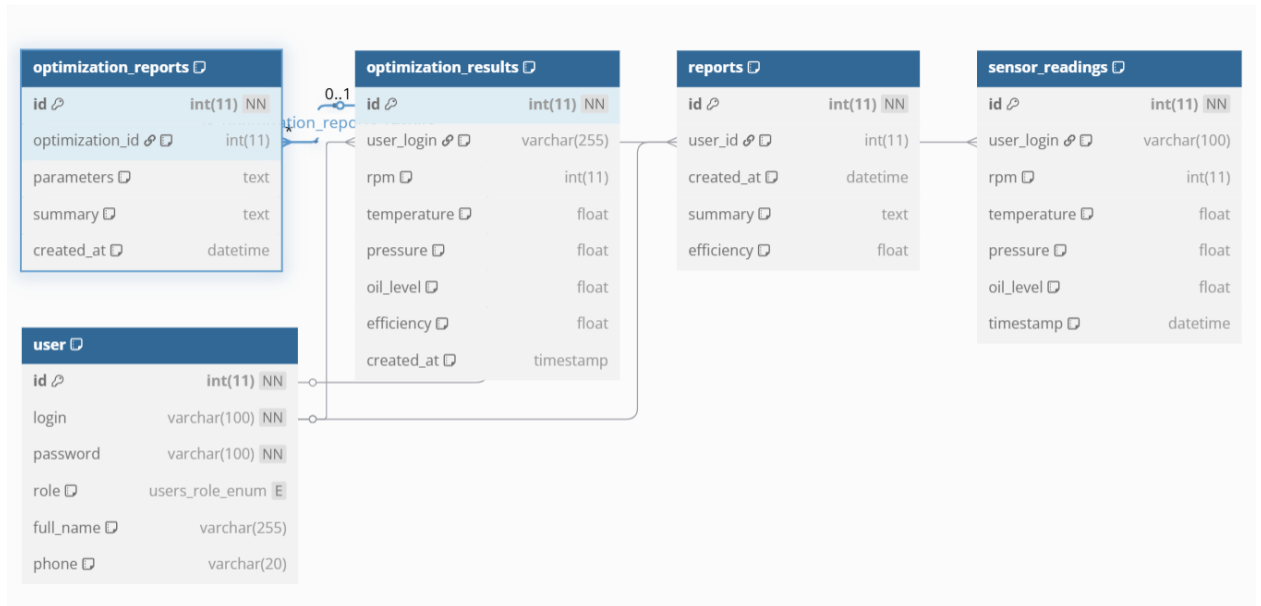


Рисунок 3.4 – Даталогічна модель БД

Характеристики реалізації:

- 1) Для полів login, user_login, optimization_id реалізовані зовнішні ключі (FK) з каскадним видаленням (ON DELETE CASCADE).
- 2) Поля id у кожній таблиці - первинні ключі (PK).
- 3) Використано відповідні типи даних (VARCHAR, INT, FLOAT, TIMESTAMP) згідно з вимогами до точності та ефективності зберігання.
- 4) Додано значення за замовчуванням для timestamp, created_at - автоматичне формування дати/часу.

Приклад Insert:

```
-- Створення користувача
INSERT INTO users (login, password, role, full_name, phone)
VALUES ('engineer1', 'pass123', 'інженер', 'Іваненко Іван', '+380671234567')
```

Реалізація виконувалась у середовищі HeidiSQL із подальшим тестуванням через API (Flask) та мобільний застосунок (Flutter). Всі таблиці повністю сумісні з MariaDB.

3.2. Технічне забезпечення

3.2.1 Загальні положення та схема автоматизації

Інформаційна система з аналізу й оптимізації параметрів чотиритактного двигуна мотоцикла є комплексом технічних і програмних засобів, які взаємодіють між собою через локальну або віртуальну мережу. В основі системи лежить серверна частина, що реалізована на одноплатному комп'ютері Raspberry Pi 5, на якому розгорнуто веб - сервер із Flask API та система управління базами даних MariaDB. Саме на цьому пристрої зберігаються результати оптимізації, звіти, облікові записи користувачів і сенсорні показники.

Передача первинної інформації відбувається за допомогою мікроконтролера Arduino Uno, який оснащений низкою датчиків (оберти, температура, тиск, рівень мастила). Дані із сенсорів передаються до Raspberry Pi або через USB - з'єднання. Це забезпечує гнучкість в розташуванні обладнання та дозволяє використовувати систему в умовах як лабораторного, так і реального дорожнього середовища.

Користувачі взаємодіють із системою через мобільний застосунок, створений на базі фреймворку Flutter. Додаток підключається до API, розгорнутого на Raspberry Pi, і дозволяє здійснювати реєстрацію та авторизацію, переглядати параметри сенсорів у реальному часі, проводити розрахунки ефективності, створювати звіти та переглядати аналітику у вигляді графіків. З'єднання забезпечується або через локальну мережу або через віртуальну приватну мережу (VPN), реалізовану засобами ZeroTier, що дозволяє безпечний доступ до системи з будь - якого місця, де є Інтернет.

Таким чином, система складається з таких основних фізичних вузлів:

- 1) Сенсорний модуль: набір датчиків, підключених до Arduino, що вимірюють технічні параметри двигуна.

- 2) Контролер збору даних: Arduino Uno , який обробляє та передає дані.
- 3) Серверна частина: Raspberry Pi 5, який приймає, обробляє і зберігає дані.
- 4) База даних: MariaDB, що виконує функцію централізованого зберігання інформації.
- 5) API - сервер: Flask - програма, яка забезпечує обмін даними між мобільним застосунком і базою.
- 6) Клієнтський застосунок: мобільний застосунок Flutter, який слугує головним інтерфейсом користувача.

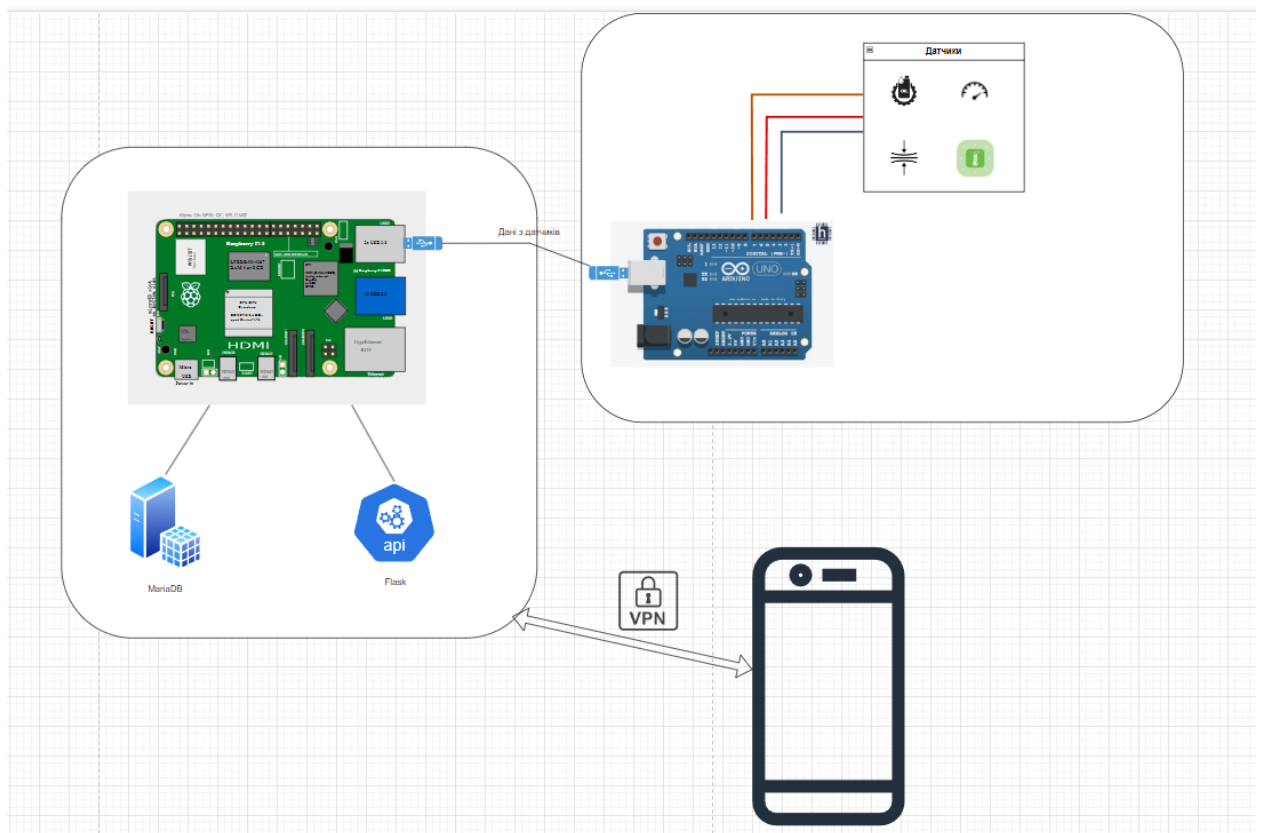


Рисунок 3.5 – Загальна схема автоматизації

3.2.2 Структура комплексу технічних засобів

Інформаційна система з аналізу та оптимізації параметрів чотиритактного двигуна мотоцикла реалізована з урахуванням принципів доступності, мобільності та енергоефективності. Центральним елементом

системи є одноплатний комп'ютер Raspberry Pi 5, який виступає в ролі локального сервера для розгортання веб - сервісів (Flask API) та системи управління базами даних MariaDB.

Передавання первинних даних здійснюється через мікроконтролер Arduino UNO, який зчитує показники з підключених датчиків: температури, тиску, обертів двигуна та рівня масла. Передавання інформації від Arduino до Raspberry Pi реалізовано через USB, а також може бути адаптоване до бездротової передачі (Wi-Fi) у разі застосування ESP32[19].

Користувачі системи (інженери, механіки, водії, адміністратори) взаємодіють з нею через мобільний застосунок, реалізований у середовищі Flutter. Застосунок підключається до API - сервера в межах локальної мережі або за допомогою віртуальної приватної мережі ZeroTier, що дозволяє забезпечити безпечний доступ поза межами фізичної мережі.

Технічні засоби, що використовуються:

- 1) Raspberry Pi 5 - 4-ядерний ARM Cortex-A76, 4–8 ГБ ОЗП, підключення через Ethernet або Wi-Fi.
- 2) Arduino UNO - мікроконтролер з 14 цифровими та 6 аналоговими входами.
- 3) Сенсори:
 1. датчики температури (DS18B20) [20]:

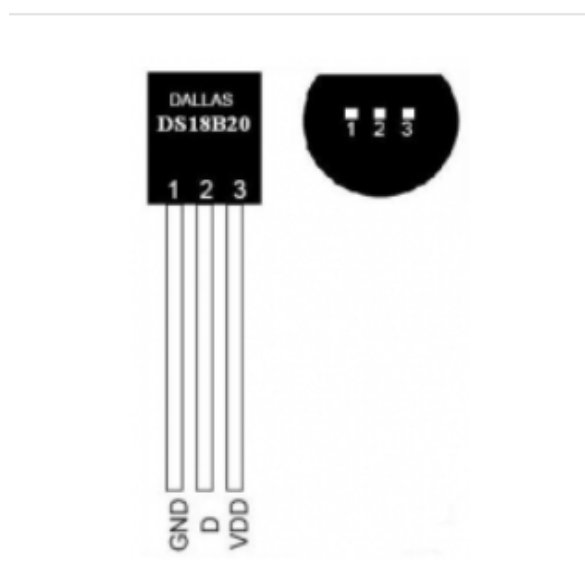


Рисунок 3.6 – Датчик температури

2. тиску (BMP280) [21]:



Рисунок 3.7 – Датчик тиску

3. обертів (оптичні/магнітні) [22]:



Рисунок 3.8 – Енкодер

4. рівня масла (аналоговий/цифровий) [23]:

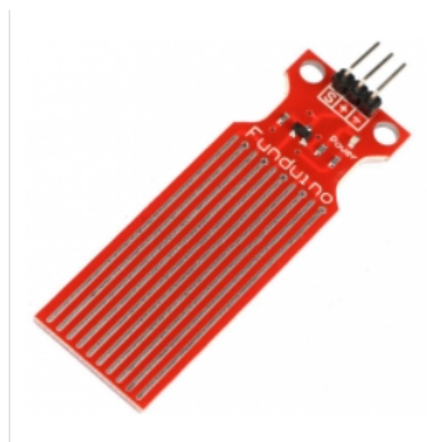


Рисунок 3.9 – Датчик рівня рідини

4) Мобільні пристрої - смартфони Android/iOS з попередньо встановленим застосунком.

5) Маршрутизатор / Wi-Fi точка доступу - для створення локальної мережі.

У випадку критичних виробничих умов (висока вібрація, пил, температура) Raspberry Pi та інші пристрої можуть розміщуватися у захисних корпусах з активним охолодженням, що відповідають стандартам IP54 або вище [24].

Система не потребує постійної присутності окремої команди. Вона функціонує завдяки діям звичайних користувачів, які використовують власні пристрої. Адміністрування Raspberry Pi можна проводити дистанційно, через адміністраторський доступ. Якщо виникає потреба у розширенні масштабів, передбачено варіанти: установка кількох вузлів Raspberry Pi для збільшення потужності або перенесення серверної частини на хостинг - платформу.

3.2.3 Опис автоматизованого робочого місця

Інформаційна система з аналізу та оптимізації параметрів чотиритактного двигуна мотоцикла передбачає використання декількох типів автоматизованих робочих місць (АРМ), кожне з яких відповідає функціональній ролі користувача - інженер, адміністратор або звичайний користувач (водій) (таблиця 3.6).

Таблиця 3.6 – АРМ і їм опис

№	Тип АРМ	Пристрій	Основні функції	Мін. вимоги до пристрою	Тип підключення	Безпека / доступ
---	---------	----------	-----------------	-------------------------	-----------------	------------------

Продовження таблиці 3.6

1	Адміністратор	Raspberry Pi 5	Сервер БД, API, керування користувачами, моніторинг	Cortex-A76, 4 ГБ RAM, microSD 32+ ГБ	Ethernet / Wi-Fi	Обмежений доступ по SSH, VPN
2	Інженер	Смартфон / планшет (Android/iOS)	Аналіз, запуск оптимізації, перегляд графіків та звітів	Android 9+, 3+ ГБ RAM, Wi-Fi / LTE	Wi-Fi / 4G / VPN (ZeroTier)	Авторизація, HTTPS / VPN
3	Водій / користувач	Смартфон	Перегляд результатів оптимізації, статусів двигуна	Android 8+, базові характеристики	Wi-Fi / 4G / 5G	Авторизація в додатку
4	Сенсорний вузол	Arduino UNO + датчики	Збір даних: RPM, t°C, тиск, рівень масла	USB/UART зв'язок, живлення 5В	Пряме з'єднання з Raspberry	Локальне, захищене оточення
5	Точка доступу	Wi-Fi роутер / мобільна мережа	Зв'язок між мобільними пристроями та Raspberry Pi	Wi-Fi 802.11n/ac	Локальна / мобільна мережа	WPA2 / VPN

Джерело: сформовано автором на основі виконаного дослідження

3.2.4 Схеми мережі передачі даних

Передача даних у системі здійснюється через локальну Wi-Fi мережу (802.11n/ac) або за допомогою віртуальної приватної мережі ZeroTier VPN, що дозволяє встановлювати з'єднання між пристроями навіть у різних

підмережах, зокрема за умов мобільного доступу до Raspberry Pi. Обмін даними реалізовано наступним чином: Arduino UNO передає дані на Raspberry Pi 5 через USB; Raspberry Pi 5 взаємодіє з мобільним застосунком Flutter через Flask REST API по протоколу HTTP, при цьому підключення клієнтів відбувається через локальну мережу або інтернет-з'єднання через ZeroTier. В системі передбачено один канал обміну між сенсором та Raspberry Pi (через USB або Wi-Fi), а також до 5 - 10 одночасних підключень користувачів до API. Вибрані апаратні засоби включають: Raspberry Pi 5 з Gigabit Ethernet для дротового з'єднання сервера, Xiaomi Router 4A (802.11ac) для Wi - Fi підключення мобільних клієнтів, модуль ESP8266/NodeMCU для бездротової передачі з сенсорів та ZeroTier One для організації VPN-доступу. Ethernet - з'єднання реалізовано через кабель UTP - 5e/6 із роз'ємом RJ - 45, локальна IP - адреса призначається динамічно (DHCP) або задається статично, а у випадку віддаленого доступу застосовується віртуальна IP-адреса від ZeroTier[25] з підтримкою NAT traversal[26] (рис. 3.10).

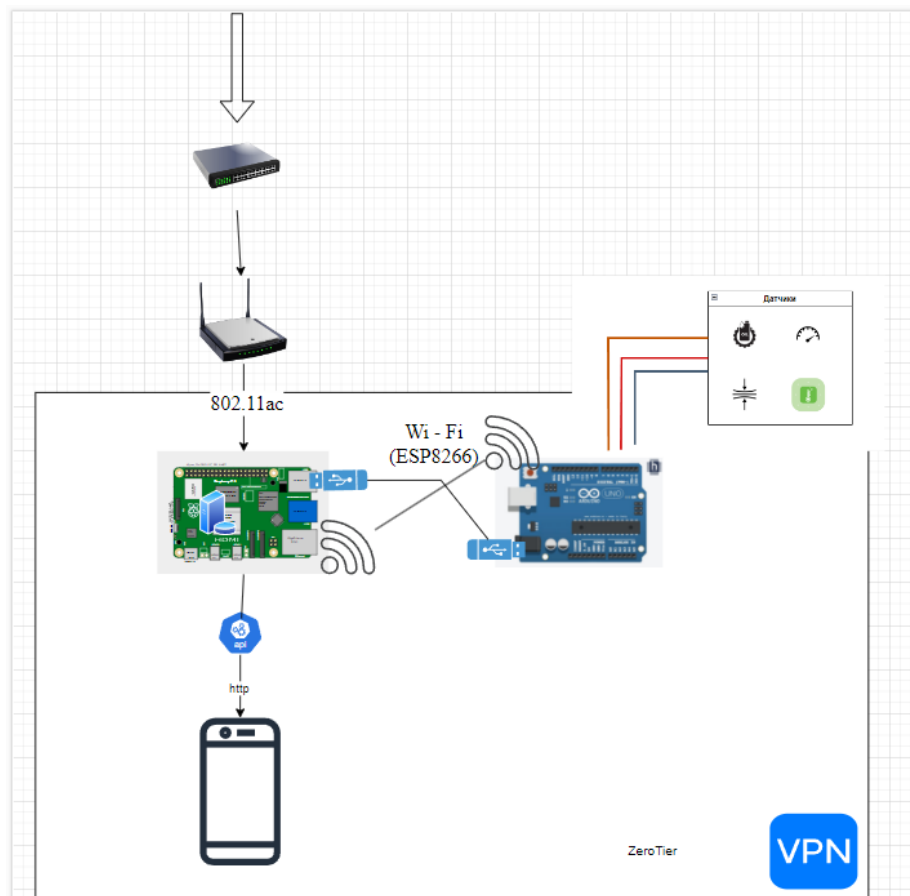


Рисунок 3.10 – Схема мережі передачі даних

3.3. Програмне забезпечення

3.3.1 Структура програмного забезпечення

Програмне забезпечення системи розподілено відповідно до тривірневої архітектурної моделі, яка включає:

- 1) Базове (системне) програмне забезпечення.
- 2) Прикладне (функціональне) програмне забезпечення.
- 3) Програмну документацію.

Базове системне програмне забезпечення (ПЗ) створює фундамент для стабільної роботи інформаційної системи, забезпечуючи доступ до апаратних ресурсів та підтримку серверних операцій. Воно включає операційну систему Raspberry Pi OS на базі Debian Linux, виконуюче середовище Python 3 з веб-сервером Flask для REST API, систему керування базами даних MariaDB 10.6, комунікаційне ПЗ ZeroTier VPN для організації захищеного каналу зв'язку, а також інтерфейси для взаємодії з Arduino через USB. На цій основі функціонує прикладне ПЗ, яке реалізує всі ключові можливості системи: серверна частина на Flask обробляє логіку оптимізації та маршрути для реєстрації, авторизації й роботи з сенсорними даними, тоді як мобільний клієнт, розроблений на Flutter (Dart), надає користувачам інтерфейс для доступу до сторінок оптимізації, перегляду звітів, графіків та профілю, взаємодіючи з API через HTTP та локально зберігаючи налаштування користувача.

Функціональні модулі системи охоплюють оптимізацію параметрів двигуна, розрахунок та візуалізацію ефективності, аналіз даних із сенсорів та авторизацію з розмежуванням прав доступу.

Тестування розробленої системи проводилося комплексно, поєднуючи ручні методи з частковою автоматизацією, та охоплювало ключові сценарії, такі як реєстрація та авторизація користувачів, процес оптимізації параметрів, коректна обробка невалідних вхідних даних та адекватне відображення

графіків ефективності; при цьому використовувалися тестові набори даних з різними межовими значеннями параметрів двигуна (температура, тиск, рівень оливи, оберти) та моделювалися нештатні ситуації, зокрема відсутність даних від сенсорів або обрив зв'язку. Програмна документація, створена за допомогою CASE - засобів, включає UML - діаграми (прецедентів, активностей, послідовностей), ER - діаграму інфологічної моделі бази даних з описом таблиць (users, optimization_reports, sensor_readings, optimization_results), діаграму потоків даних від мікроконтролера Arduino до мобільного застосунку, а також OpenAPI - специфікацію для API, формуляр, опис контрольного прикладу та структуру SQL - таблиць, які детально представлені у звіті (рис. 3.11).

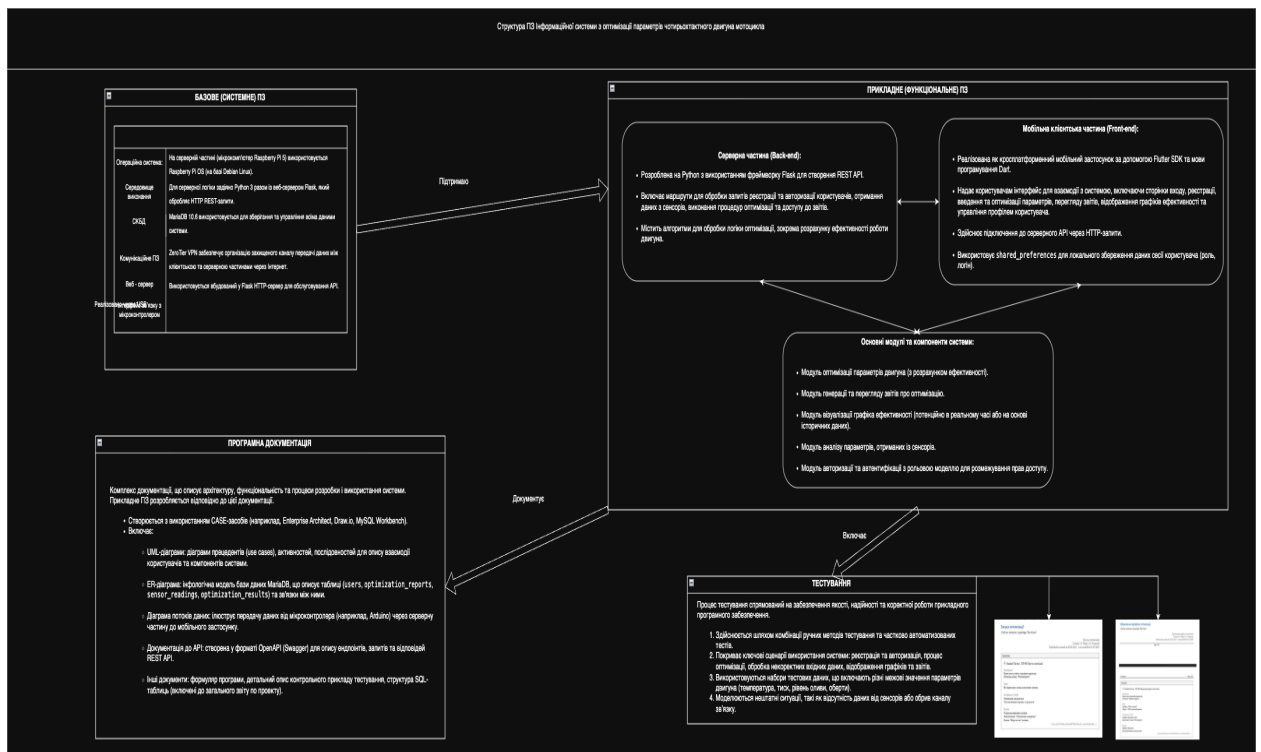


Рисунок 3.11 – Схема структури ПЗ

3.3.2 Системне програмне забезпечення

Для реалізації інформаційної системи аналізу та оптимізації параметрів чотиритактного двигуна мотоцикла було обрано комплекс системного

програмного забезпечення, що забезпечує надійну та безпечну роботу всіх її компонентів: серверна частина функціонує на Raspberry Pi 5 під керуванням Raspberry Pi OS (на базі Debian Linux) завдяки її продуктивності на обмежених ресурсах та відкритості, тоді як клієнтська частина представлена мобільним застосунком на Flutter для Android. Проектування інфологічної моделі бази даних здійснювалося за допомогою DiagramIo, а для моделювання системи та її компонентів (блок-схем, UML - діаграм) використовувалися Draw.io та Enterprise Architect. Серверне REST API розгорнуто на Python 3 з фреймворком Flask та бібліотеками flask-cors і mysql-connector-python, а кросплатформенний мобільний клієнт розроблено на Flutter SDK (Dart) з використанням shared_preferences, http та fl_chart. Захищений віддалений доступ до системи реалізовано через ZeroTier VPN, що створює віртуальну захищену мережу. Розробка ведеться в IDE PyCharm та Android Studio. Додатково, для роботи з графікою, схемами та документацією використовуються Figma та Draw.io, що в сукупності забезпечує повний цикл розробки та підтримки сучасної розподіленої інформаційної системи (таблиця 3.7).

Таблиця 3.7 – Системне програмне забезпечення

Категорія ПЗ	Назва / Засіб	Призначення
ОС сервера	Raspberry Pi OS	Робота серверної частини
ОС клієнту	Android	Платформа для запуску застосунку
СУБД	MariaDB + HeidiSQL	Збереження даних
Мережевий VPN	ZeroTier One	Захищене підключення до API

Продовження таблиці 3.7

Фреймворк API	Flask (Python)	Rest API та його реалізація
Мова програмування сервера	Python	Логіка та взаємодія з БД
Мобільна платформа	Flutter + Dart	Створення мобільного додатку
CASE - засіб	Enterprise Architect	Проектування і моделювання
Графічні інструменти	Figma, Draw.io	Розробка схем та дизайну
Інструменти IDE	PyCharm, Android Studio	Розробка серверного ПЗ та мобільного застосунку
Бібліотеки Flutter	http, fl_chart, shared_prefe	Відправка http запитів, побудова графіків

Джерело: сформовано автором на основі виконаного дослідження

3.3.3 Прикладне програмне забезпечення

У межах створення інформаційної системи для аналізу та оптимізації параметрів чотиритактного двигуна мотоцикла було розроблено комплекс прикладного програмного забезпечення, яке охоплює серверну частину, мобільний клієнт та логіку взаємодії з апаратною складовою (сенсорами, контролерами).

Програмне забезпечення загального призначення - сюди входить мобільний застосунок, розроблений за допомогою Flutter SDK. Він забезпечує доступ користувачів до основних функцій системи: авторизації, збору показників із сенсорів, оптимізації параметрів, перегляду звітів і візуалізації

ефективності. Flutter був обраний через його кросплатформеність, швидкість розробки та підтримку адаптивного UI.

Методо - орієнтоване ПЗ - використовується для реалізації математичної моделі обчислення ефективності двигуна на основі зібраних сенсорних даних (RPM, температура, тиск, рівень оливи). Алгоритми реалізовані на Python у серверному модулі (Flask API).

Проблемно - орієнтоване ПЗ - включає в себе модуль Flask API, розгорнутий на Raspberry Pi 5, який приймає, обробляє, зберігає запити та дані користувачів, реалізуючи всю бізнес-логіку. Він також забезпечує взаємодію з MariaDB, надаючи REST - інтерфейси для роботи з інформацією про користувачів, сенсори, оптимізацію та звіти.

ПЗ для роботи в глобальних мережах - система підтримує роботу як у локальній мережі, так і в глобальній через ZeroTier VPN, що дозволяє безпечно обмінюватися даними між мобільним застосунком і сервером незалежно від фізичного розташування.

ПЗ для адміністрування обчислювального процесу - для моніторингу роботи API, бази даних та пристроїв використовується системне адміністрування Raspberry Pi з ОС Raspberry Pi OS (Linux). Адміністратор має доступ до логів, журналів подій та може оновлювати серверну частину.

При розробці прикладного ПЗ було враховано такі критерії:

- 1) Інтеграція з обраною MariaDB.
- 2) Можливість роботи в режимі реального часу.
- 3) Інтерфейс користувача, оптимізований для мобільних пристроїв;
- 4) Масштабованість - потенційна підтримка нових сенсорів і додаткових аналітичних модулів.
- 5) Безпека - ізоляція локальної мережі або тунелювання трафіку через ZeroTier.

3.3.4 Програмна документація

Програмна документація є невід’ємною частиною інформаційної системи, що забезпечує повноцінне використання, обслуговування, адаптацію та розвиток прикладного програмного забезпечення. Вона охоплює опис структури програмних засобів, їх функціональні можливості, інструкції щодо встановлення, налаштування, запуску, а також вимоги до апаратного та інформаційного середовища.

У формулярі наведено основні відомості про програмний продукт (таблиця 3.8):

Таблиця 3.8 – формуляр

Назва	Інформаційна система з аналізу й оптимізації параметрів чотиритактного двигуна мотоцикла
Мова розробки	Dart (Flutter) для мобільного клієнта, Python (Flask) для API
База даних	MariaDB
Сервер	Raspberry Pi 5, ОС: Raspberry Pi OS (Linux)
Компоненти	Мобільний застосунок, серверний API та база даних
Дата створення	2025

Джерело: сформовано автором на основі виконаного дослідження

Призначення програми - система розроблена для комплексного аналізу та оптимізації роботи двигуна мотоцикла. Її ключові функції включають збір та обробку сенсорної інформації безпосередньо з датчиків, автоматизоване обчислення показників ефективності його роботи, а також збереження отриманих результатів оптимізації для подальшого аналізу. Крім того, система забезпечує формування детальних звітів та наочну графічну візуалізацію змін параметрів двигуна, надаючи структурований доступ до цих результатів різним категоріям користувачів, таким як інженери, водії та адміністратори.

Сфера застосування системи охоплює підприємства, що займаються обслуговуванням та ремонтом мототехніки, навчальні лабораторії при технічних закладах освіти, а також спеціалізовані майстерні. Використання системи спрямоване на покращення якості технічного обслуговування

двигунів, підвищення їх ефективності та оптимізацію експлуатаційних характеристик.

Обмеження на застосування - для забезпечення повноцінного та стабільного функціонування інформаційної системи необхідно дотримуватися мінімальних вимог до конфігурації апаратного та програмного забезпечення. Серверна частина, що базується на мікрокомп'ютері Raspberry Pi, вимагає наявності 4 - ядерного ARM - процесора з тактовою частотою не менше 1.5 ГГц, обсягу оперативної пам'яті від 4 до 8 ГБ, а також 32 ГБ постійної пам'яті на SD - карті або SSD - накопичувачі. Обов'язковою є наявність Ethernet або Wi - Fi адаптера для мережевого підключення та встановлена операційна система Raspberry Pi OS.

Для клієнтської частини, реалізованої у вигляді мобільного застосунку, необхідний мобільний пристрій під управлінням операційної системи Android версії 8.0 або вище, або iOS версії 13.0 або вище. Пристрій повинен мати щонайменше 2 ГБ оперативної пам'яті та забезпечувати підключення до локальної мережі або до віртуальної приватної мережі ZeroTier VPN для взаємодії з сервером.

Керівництво системного програміста - перед початком експлуатації інформаційної системи системному програмісту необхідно виконати ряд підготовчих кроків для належного налаштування серверного середовища. По-перше, слід встановити всі необхідні залежності Python, зокрема бібліотеки Flask та mysql-connector-python, які забезпечують роботу веб - сервісу та взаємодію з базою даних. По - друге, потрібно налаштувати систему керування базами даних MariaDB, що включає створення бази даних з назвою engine_optimizer та подальший імпорт її структури за допомогою наданого SQL - файлу. На завершення, важливо перевірити успішність підключення до розгорнутого API за його IP - адресою, як у локальній мережі, так і через ZeroTier VPN, для забезпечення доступності сервісу.

Мобільний застосунок, розроблений на платформі Flutter, компілюється у виконуваний файл формату .apk (для Android). Подальше встановлення

цього файлу здійснюється безпосередньо на клієнтський мобільний пристрій для забезпечення доступу користувачів до функціоналу системи.

Опис контрольного прикладу - для перевірки працездатності та коректності функціонування системи використовується контрольний приклад, що моделює типовий сценарій взаємодії. Цей приклад включає набір визначених вхідних даних: оберти двигуна (RPM) встановлені на рівні 6000 об/хв, температура двигуна становить 95°C, тиск оливи - 2.1 бар, а рівень оливи - 60%. За таких умов очікується, що система розрахує ефективність роботи двигуна на рівні 100%.

У результаті успішного виконання контрольного прикладу в базі даних має з'явитися відповідний збережений запис у таблиці `optimization_results`, а також сформований звіт у таблиці `optimization_reports`. Крім того, дані про ефективність та пов'язані параметри повинні коректно візуалізуватися у вигляді графіка в мобільному застосунку користувача.

Структура програмних модулів системи організована у вигляді окремих файлів, що відповідають за різні компоненти та функціональні частини. Основний файл серверної логіки `app.py` містить реалізацію Flask API з усіма необхідними маршрутами для обробки запитів від клієнтської частини. Компоненти інтерфейсу користувача мобільного застосунку, розроблені на Flutter, знаходяться у файлах з розширенням `.dart`, таких як `login_page.dart` (сторінка входу), `home_page.dart` (головна сторінка), `optimization_page.dart` (сторінка оптимізації) та інших подібних файлах. Структура бази даних MariaDB, включаючи таблиці, поля та зв'язки, визначена у SQL - файлі `main.sql`.

3.4. Результати реалізації інформаційної підсистеми

У результаті реалізації інформаційної системи з визначення оптимальних параметрів чотиритактного двигуна мотоцикла було створено

систему, яка об'єднує апаратні та програмні компоненти в єдиний комплекс. Розроблене рішення дозволяє здійснювати збір, збереження, обробку, візуалізацію даних, отриманих від сенсорів двигуна.

Користувач має можливість запускати розрахунок ефективності двигуна на основі актуальних параметрів (оберти, температура, тиск, рівень мастила), збереження результатів та перегляд звітів. Для більшої наочності реалізовано відображення історії оптимізацій у вигляді лінійного графіка.

Система підтримує мультикористувацький доступ як у межах локальної Wi - Fi мережі, так і через VPN (ZeroTier), що забезпечує безпечний віддалений доступ до функціоналу навіть поза локальною мережею. Дані з сенсорів надходять від мікроконтролера Arduino UNO, а передача до Raspberry Pi здійснюється через Wi - Fi або USB.

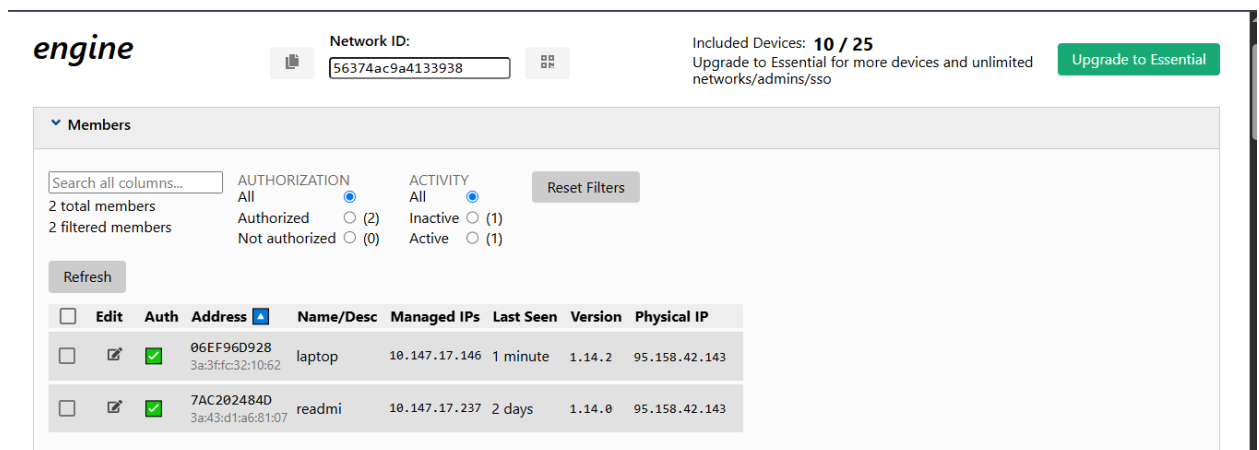


Рисунок 3.12 – Інтерфейс ZeroTier

Як результат було реалізовано мобільний застосунок на базі Flutter, що забезпечує інтерфейс користувача для трьох категорій: адміністратора, інженера та звичайного користувача. Програма містить окремі сторінки для авторизації, перегляду поточних сенсорних даних, запуску оптимізації, аналізу результатів у вигляді таблиць і графіків, а також профілю користувача. Програмна частина взаємодіє з REST API, розгорнутим на Raspberry Pi 5, що обробляє запити та працює з базою даних MariaDB.

В першу чергу було реалізовано сторінки і логіку роботи авторизації та реєстрації користувача в системі. На рисунку 3.13 відображене головне вікно входу з авторизацією користувача.

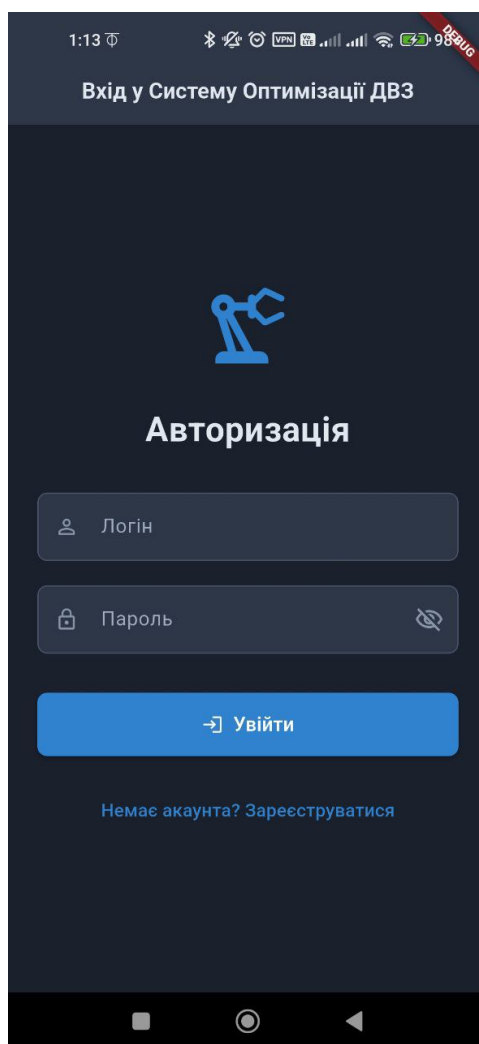


Рисунок 3.13 – Головне вікно логіну користувача

Також на екрані присутній перехід на сторінку реєстрації, якщо користувач ще не зареєстрований (рис. 3.14):

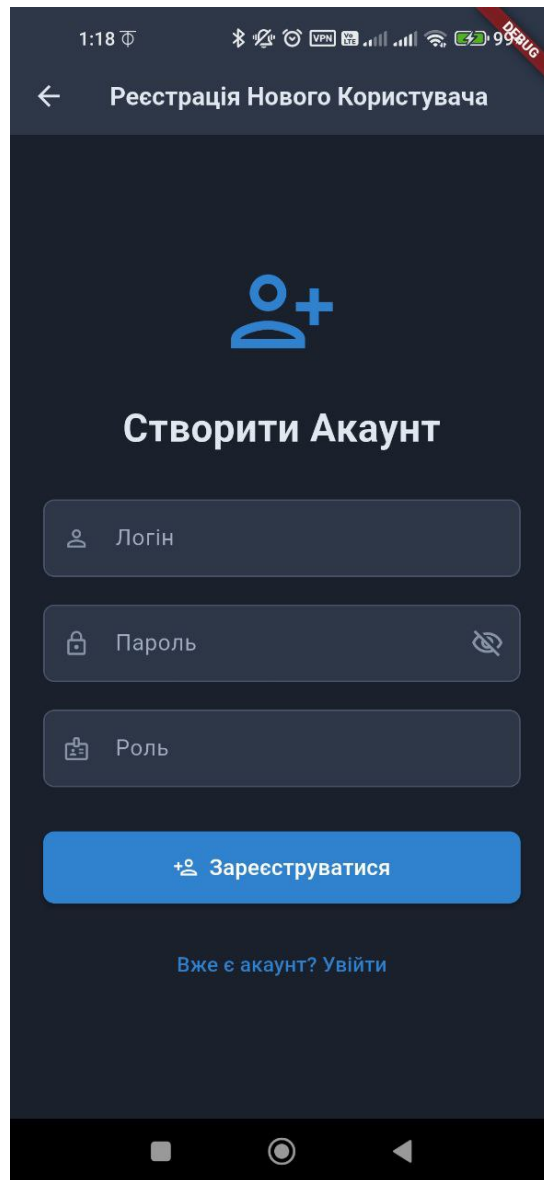


Рисунок 3.14 – Головне вікно реєстрації користувача

На сторінці реєстрації також присутня логіка з обмеженнями та правилами, а саме для логіну користувача потрібно вписати, щонайменше трьох символів. Для створення паролю присутнє правило, яке визначає, що для його верифікації новий користувач повинен внести не менше шести символів і далі вписати свою роль (рис. 3.15).

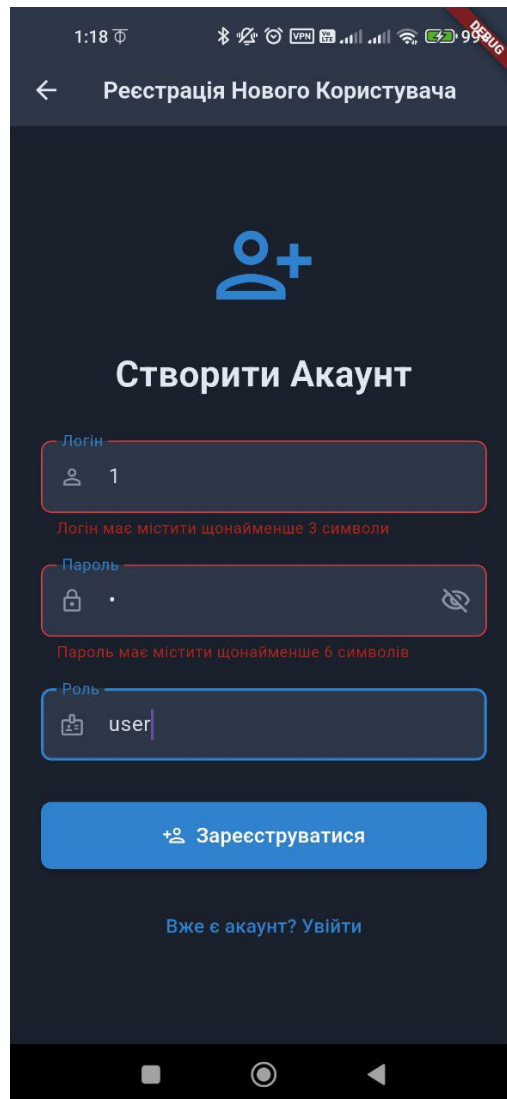


Рисунок 3.15 – Головне вікно реєстрації з обмеженнями

Після успішного логіну або реєстрації користувач потрапляє на головну сторінку програми де у верхньому меню справа був реалізований профіль користувача з логіном користувача і його роль (рис. 3.16). Також система охоплює чотири підсторінки (оптимізація параметрів, звіт оптимізації, аналіз параметрів, графік ефективності) для всіх типів користувачів (рис 3.17). Було реалізована кнопка «Вийти з акаунту».



Рисунок 3.16 – Верхнє меню програми

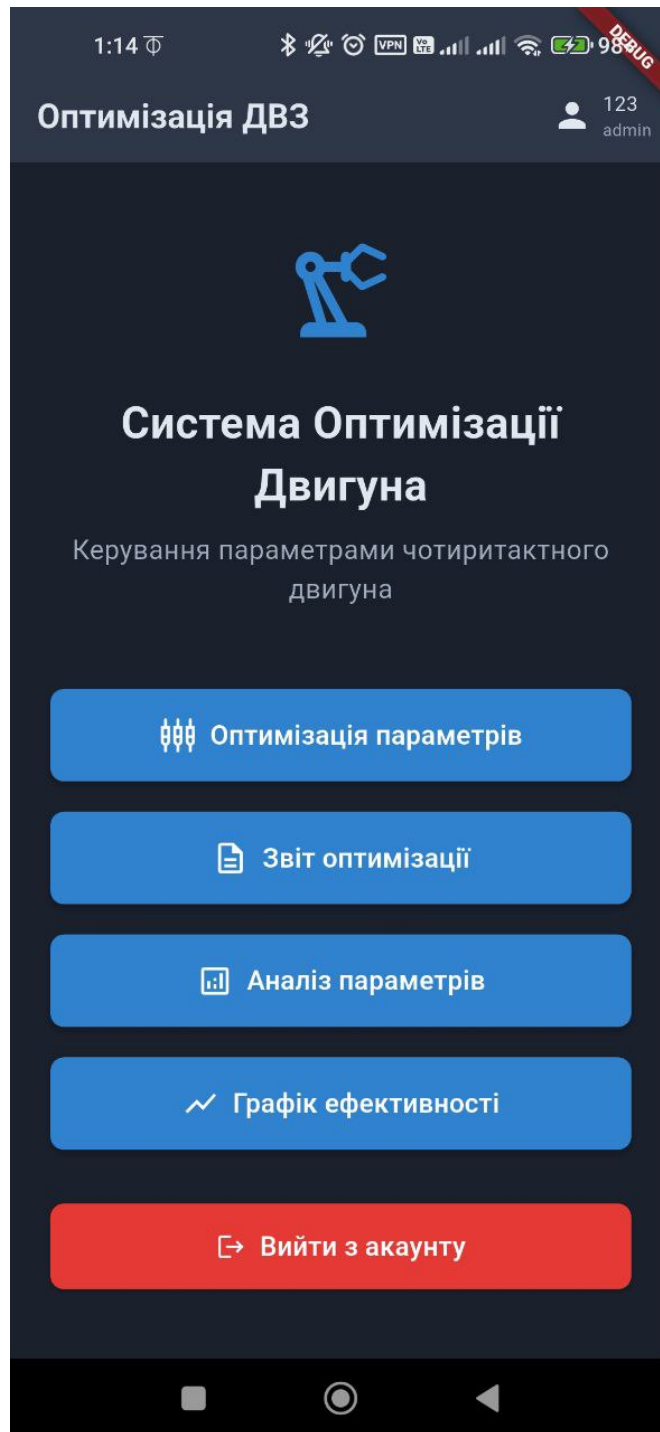


Рисунок 3.17 – Головне сторінка програми

В програмі була реалізована логіка різних опції і можливостей для користувачів. При натисканні на логін користувача у верхньому меню в залежності від його ролі буде відображена різна інформація. Наприклад, якщо адміністратор системи заходить в меню буде відображатись стилізована таблицка усіх користувачів системи з їх роллю та можливістю змінити для них пароль або видалити з системи (рис. 3.18 – 3.20).

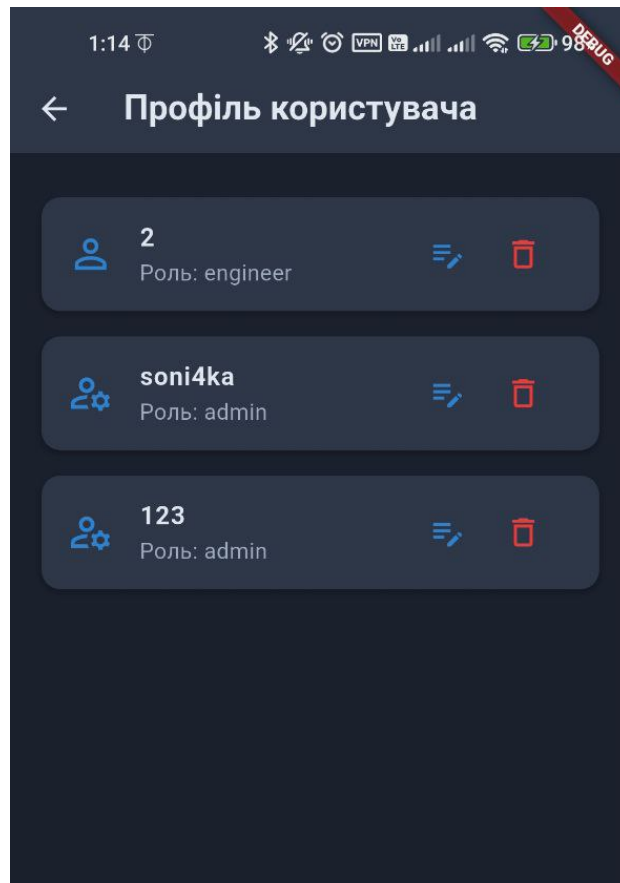


Рисунок 3.18 – Меню адміністратора системи

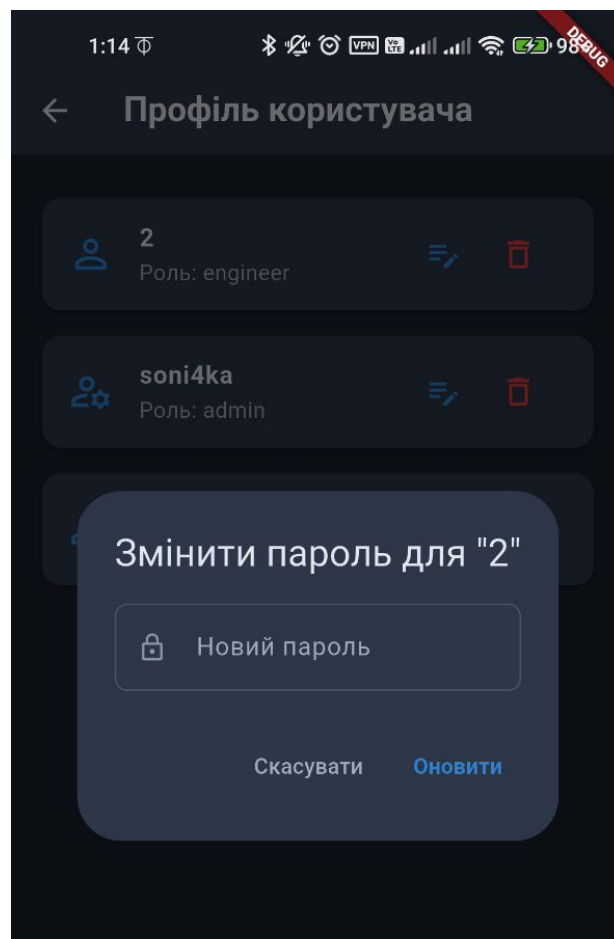


Рисунок 3.19 – Вікно зміни паролю

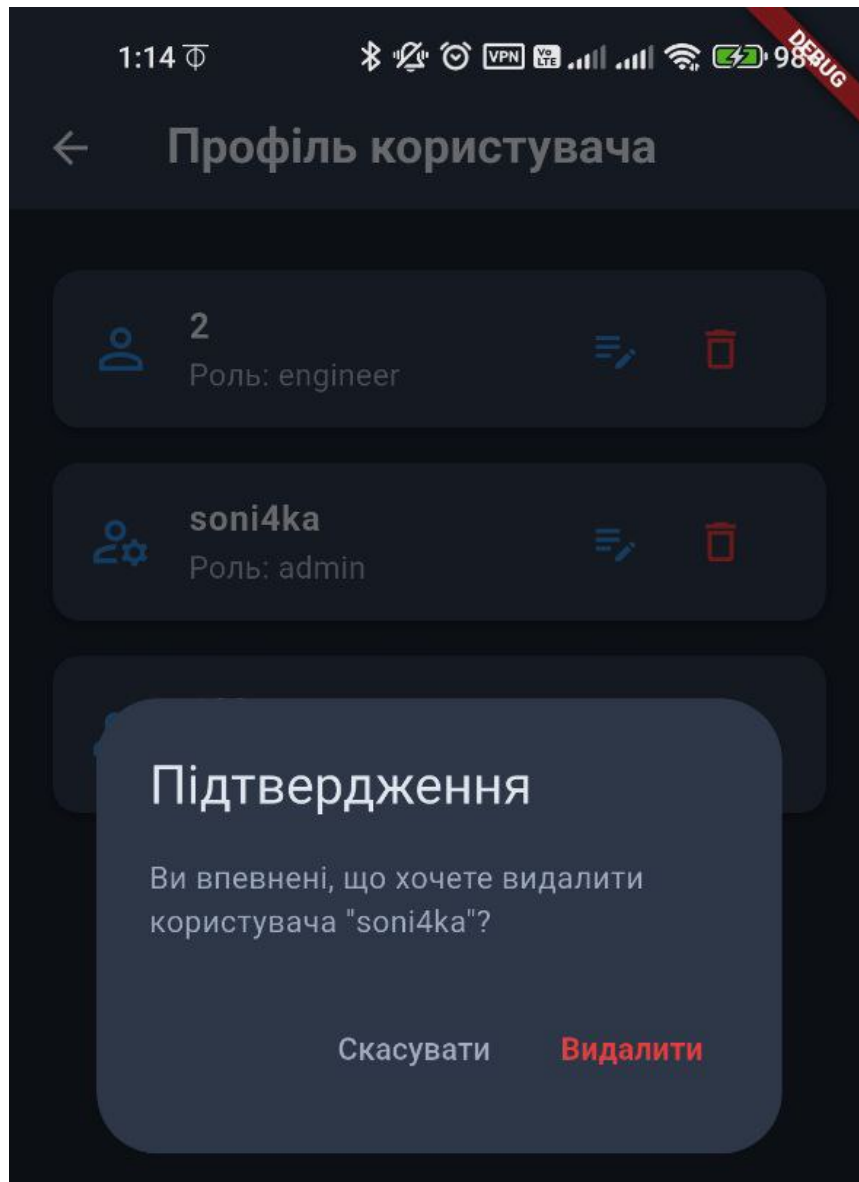


Рисунок 3.20 – Вікно видалення користувача

Для користувачів з типом ролі «інженер» або звичайний користувач відсутня можливість видалення інших користувачів та перегляд їх наявності в цілому. Але для клієнтів системи збережена функція зміна паролю і можливість переглянути свою роль. Додатково для користувачів з цими типами ролей було розроблено поля для введення особистих даних, таких як «ПІБ» та «Телефон» (рис. 3.21).

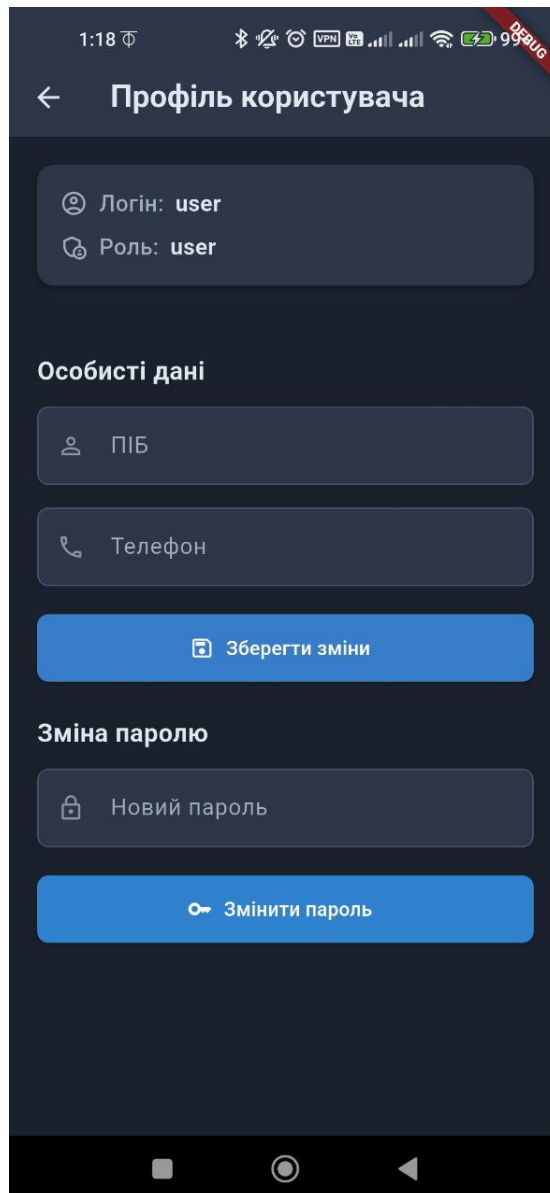


Рисунок 3.21 – Меню користувача системи

Сторінка «Оптимізація параметрів» мобільного застосунку призначена для введення користувачем поточних показників чотиритактного двигуна з метою їх подальшого аналізу та розрахунку оптимальних значень.

Основна робоча область сторінки починається з інформаційного напису «Введіть поточні параметри двигуна:», що спрямовує дії користувача. Нижче розташований структурований блок, який складається з чотирьох полів для введення числових значень параметрів двигуна. Кожне поле візуально відокремлене.

- 1) Перше поле призначене для введення значення «Оберти (об/хв)».
- 2) Друге поле - для параметра «Температура (°C)».
- 3) Третє поле - для параметра «Тиск (бар)».

4) Четверте поле - для параметра «Рівень мастила (%)».

Після блоку введення даних розміщено дві функціональні кнопки. Перша, основна кнопка дії з написом «Оптимізувати», ініціює процес розрахунку та застосування оптимізаційних алгоритмів на основі введених користувачем даних. Друга кнопка, з написом «Обчислити з поточних даних», передбачає можливість автоматичного заповнення полів або розрахунку на основі даних, отриманих від сенсорної системи двигуна (рис. 3.22 – 3.24).

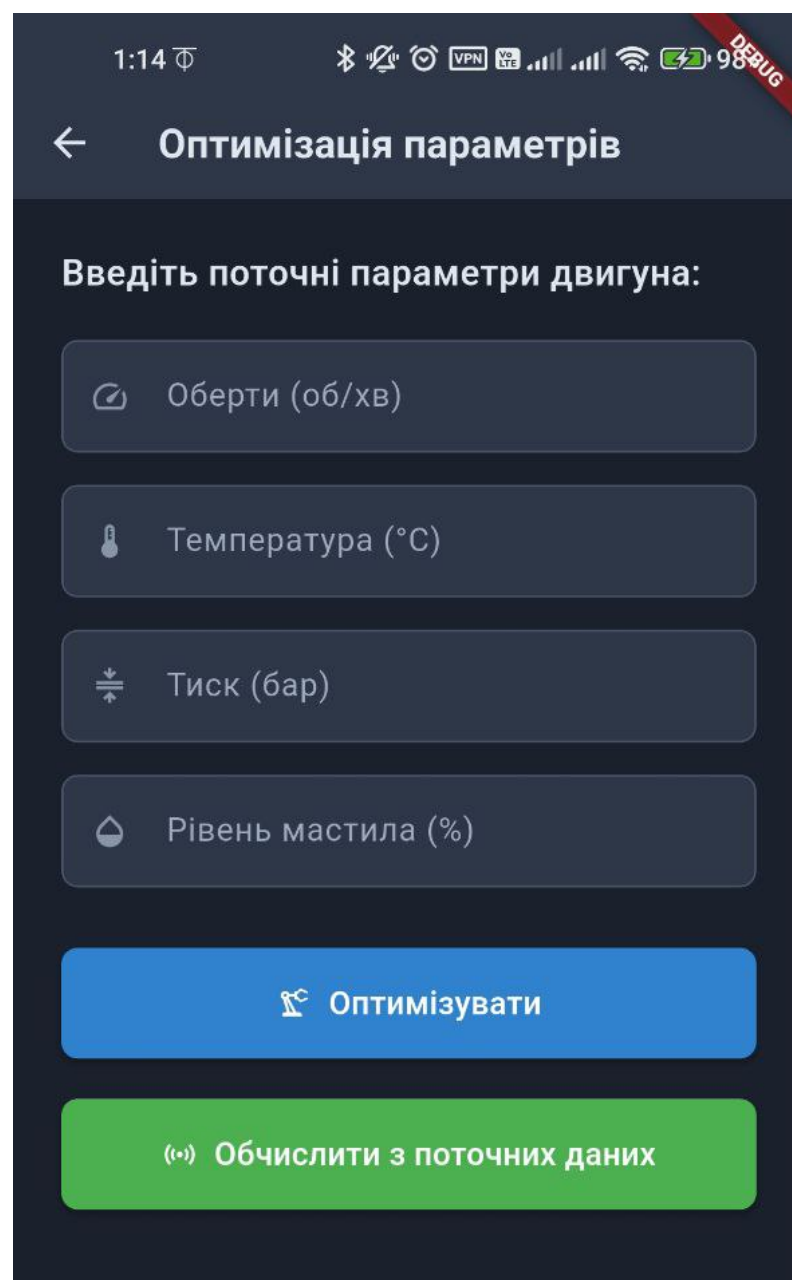


Рисунок 3.22 – Меню сторінки оптимізації параметрів

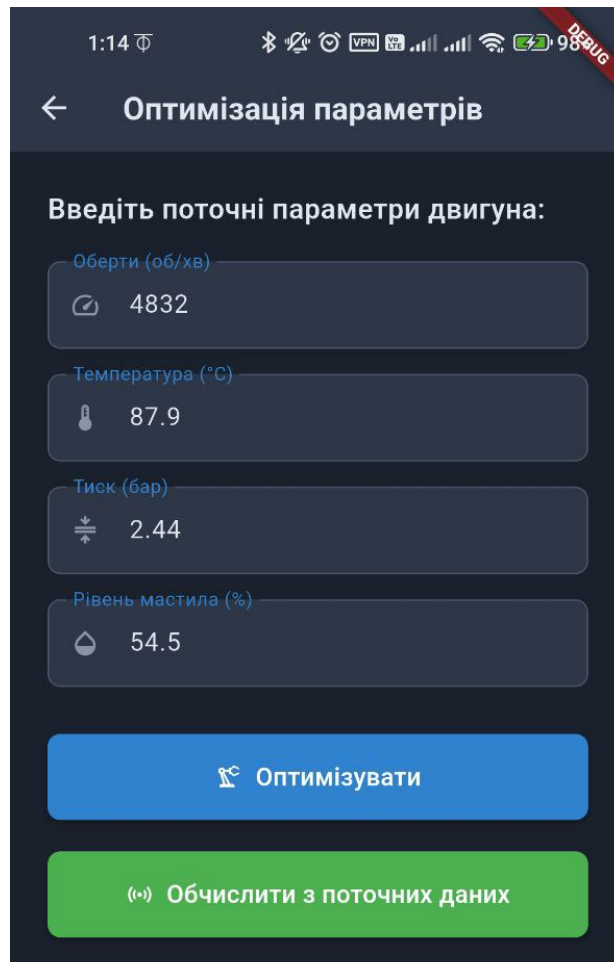


Рисунок 3.23 – Меню сторінки оптимізації параметрів з заповненими даними

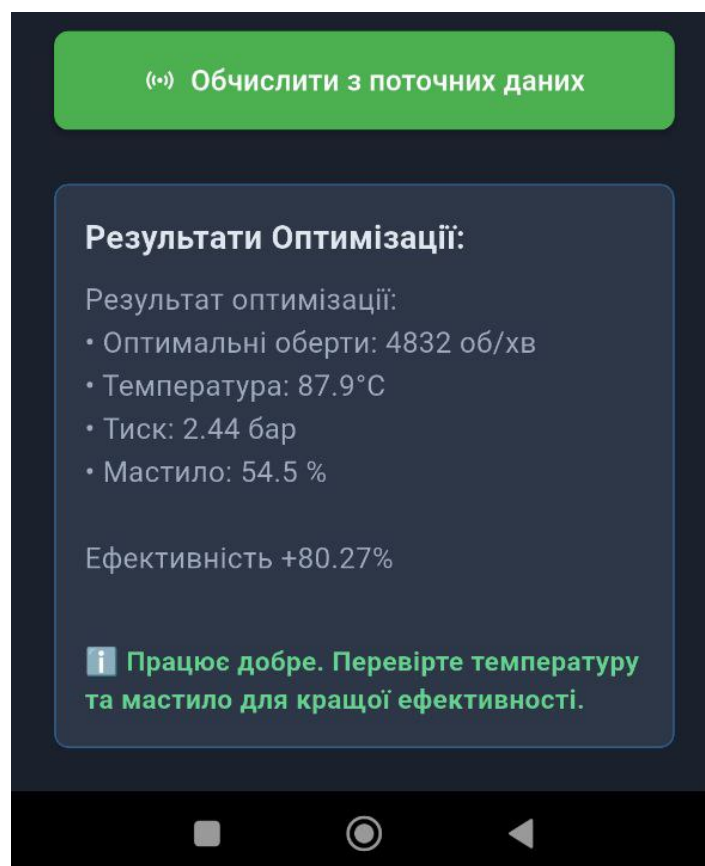


Рисунок 3.24 – Результат виконання оптимізації

Сторінка «Звіти Оптимізації» мобільного застосунку призначена для відображення списку згенерованих звітів про результати оптимізації параметрів чотиритактного двигуна, а також для надання користувачеві інструментів для управління цими звітами.

Основна робоча область сторінки представляє собою вертикально прокручуваний список звітів, де кожен елемент містить ідентифікаційну іконку, дату створення, ключову інформацію (зокрема, ефективність та параметри), а також набір функціональних іконок для швидких дій: експорту в PDF, редагування (тільки для адміністратора) та видалення звіту, що забезпечує інформативний перегляд та управління результатами оптимізації (рис. 3.25 – 3.28).

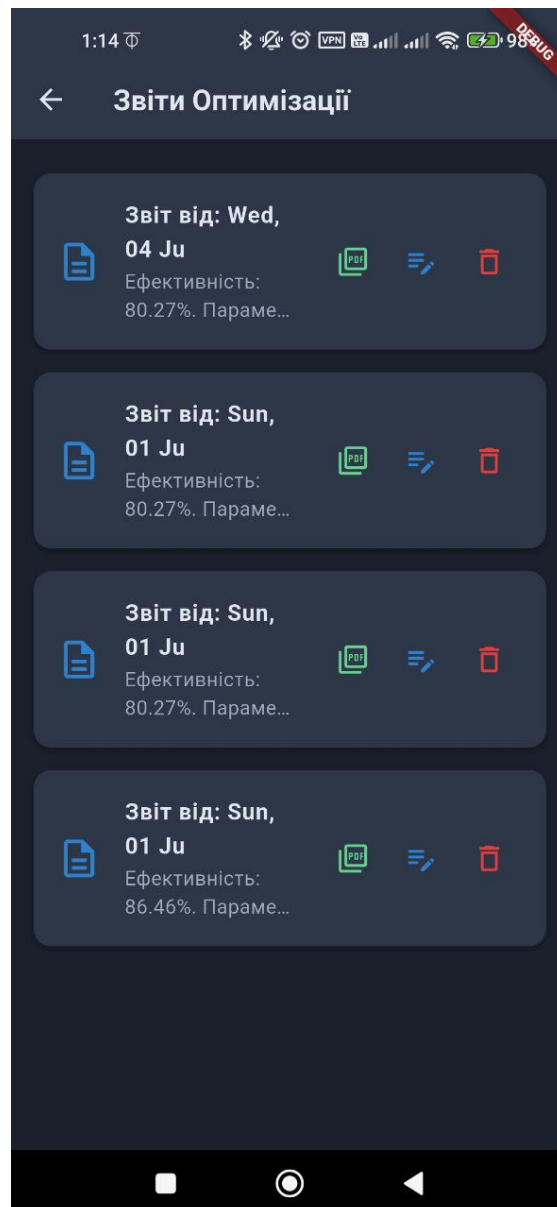


Рисунок 3.25 – Сторінка звіту оптимізації

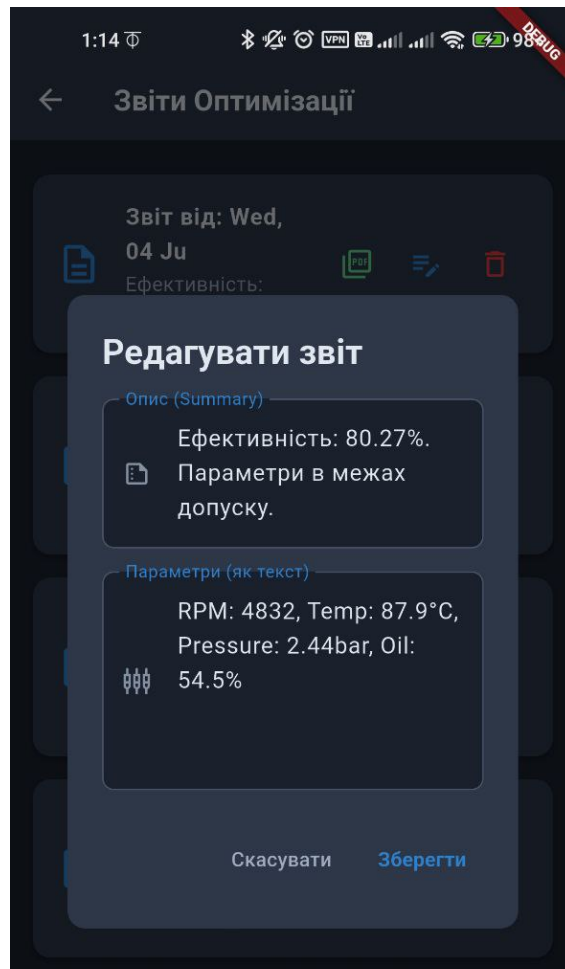


Рисунок 3.26 – Вікно редагування звіту (тільки для адміністратора)

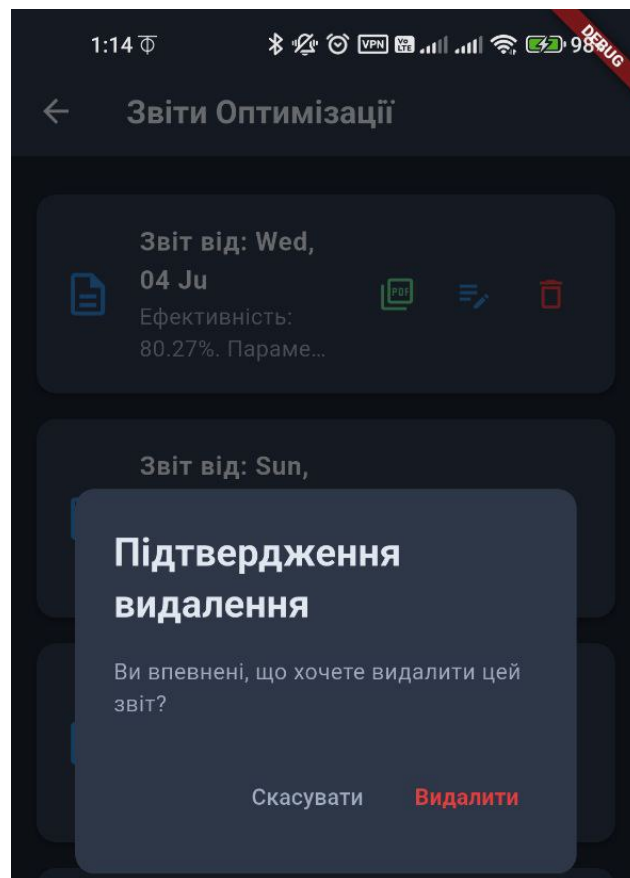


Рисунок 3.27 – Вікно видалення звіту

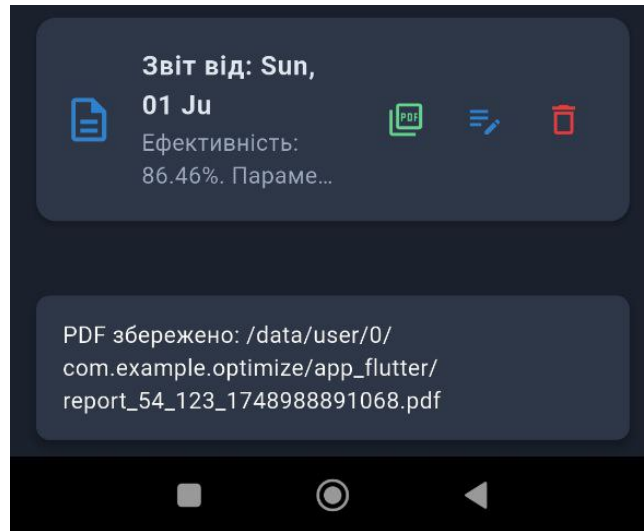


Рисунок 3.28 – Вікно збереження файлу в PDF

Сторінка «Аналіз Параметрів Двигуна» мобільного застосунку призначена для відображення в реальному часі або останніх зчитаних значень ключових експлуатаційних параметрів чотиритактного двигуна. Під переліком параметрів розташована функціональна кнопка «Зчитати Нові Дані». Кнопка призначена для ініціювання запиту на отримання актуальних значень параметрів від сенсорної системи двигуна (рис. 3.29).

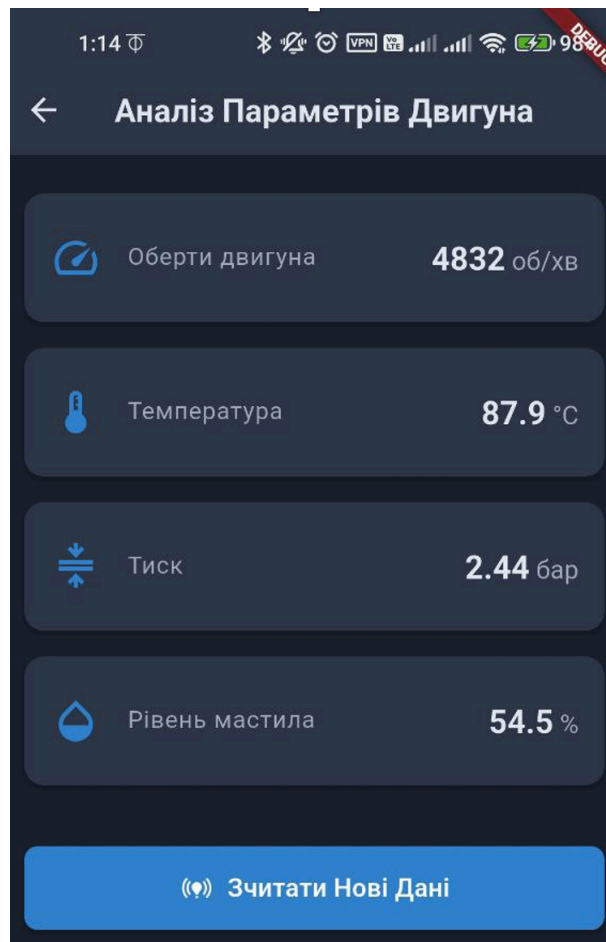


Рисунок 3.29 – Сторінка аналізу параметрів двигуна

Сторінка «Діаграма Ефективності» мобільного застосунку призначена для графічної візуалізації показників ефективності роботи чотиритактного двигуна за різні періоди, представлених у вигляді стовпчастої діаграми (рис. 3.30).

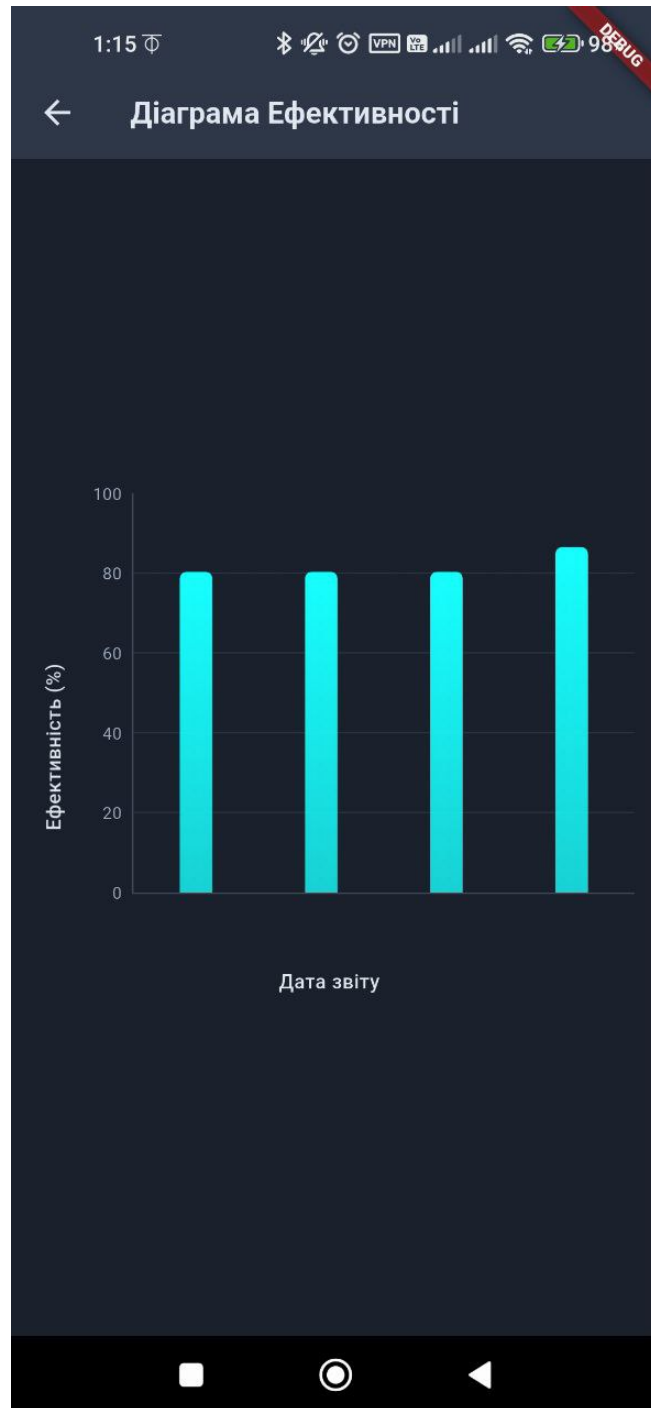


Рисунок 3.30 – Сторінка діаграми ефективності

Також була розроблена логіка і дизайн сторінки коли користувач не виконав жодної оптимізації (рис. 3.31):

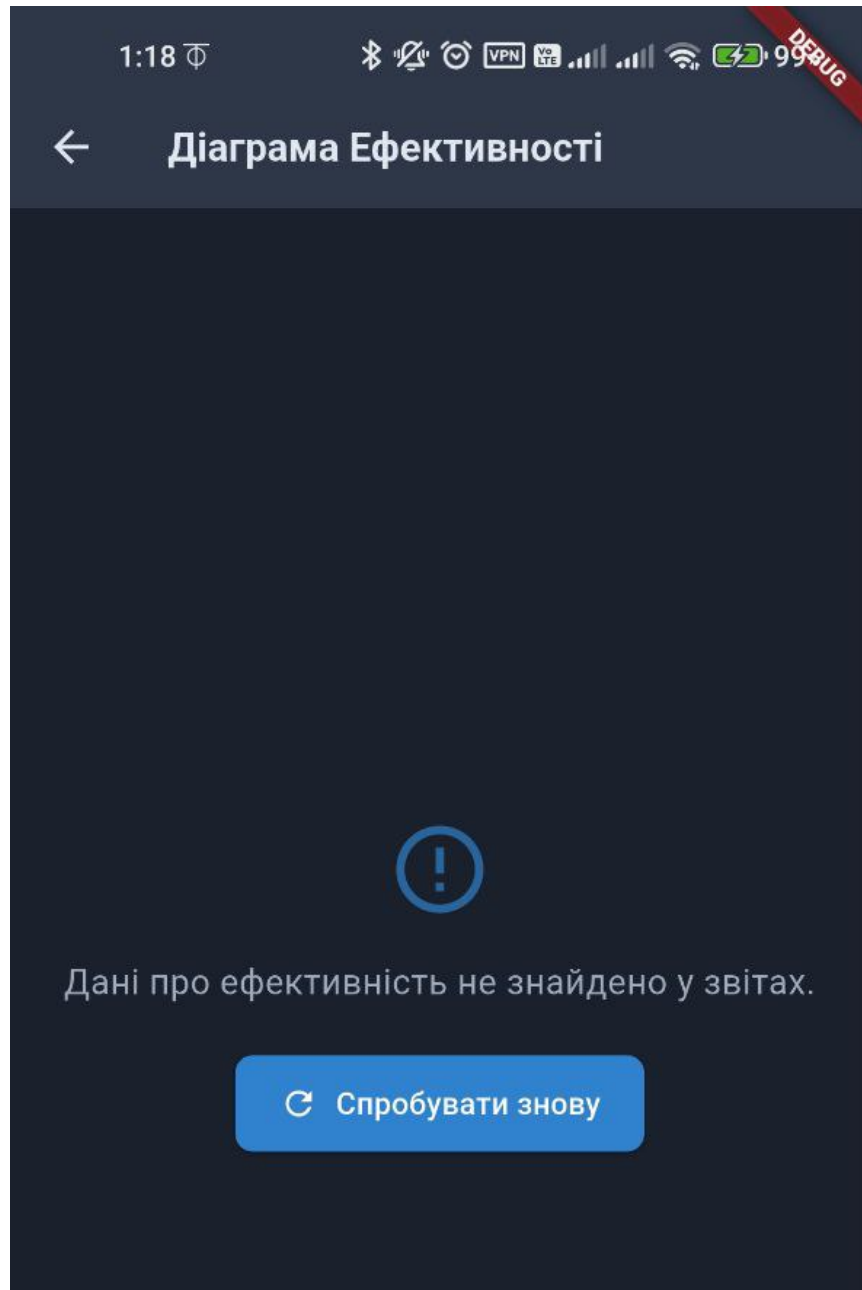


Рисунок 3.31 – Сторінка діаграми ефективності без графіків

Розроблений мобільний застосунок є комплексним інструментом для аналізу та оптимізації параметрів чотиритактного двигуна мотоцикла, забезпечуючи повний цикл роботи з даними. Він дозволяє користувачам реєструватися та безпечно входити в систему, вводити або автоматично зчитувати ключові параметри двигуна (оберти, температура, тиск, рівень мастила), ініціювати процес оптимізації, а також переглядати, експортувати в PDF та керувати звітами про результати. Крім того, користувачі можуть візуалізувати динаміку ефективності на діаграмах, аналізувати поточні

показники сенсорів та управляти своїми профілями з урахуванням рольової моделі доступу. Таким чином, застосунок надає інженерам, водіям та адміністраторам зручний мобільний інструментарій для моніторингу, аналізу та підвищення ефективності роботи двигунів.

ВИСНОВКИ

У бакалаврській кваліфікаційній роботі було виконано повний цикл проєктування, розроблення та часткової реалізації інформаційної системи для визначення оптимальних параметрів чотиритактного двигуна мотоцикла. Відповідно до поставленої мети було реалізовано комплексну систему, що об'єднує апаратні засоби (Arduino UNO, Raspberry Pi 5, сенсори), серверну частину на Flask API, базу даних MariaDB та мобільний застосунок, створений у середовищі Flutter.

У результаті виконання роботи було досягнуто низку вагомих результатів. По - перше, проведено глибокий аналіз предметної галузі технічної експлуатації чотиритактних двигунів та існуючих рішень для моніторингу їх параметрів, що обґрунтувало необхідність створення доступної мобільної системи. На основі цього аналізу сформульовано детальні функціональні та нефункціональні вимоги до інформаційної системи, побудовано її інформаційну модель та розроблено UML - діаграми, включаючи діаграми прецедентів, компонентів, класів та вимог, у середовищі Enterprise Architect. Важливим етапом стала розробка структури бази даних для ефективного збереження показників двигуна, результатів оптимізації та історичних звітів; для цього було побудовано інфологічну та даталогічну моделі та обґрунтовано вибір СКБД MariaDB, яка забезпечує високу продуктивність на одноплатному комп'ютері.

Подальша робота включала успішну реалізацію серверної логіки на основі мікрокомп'ютера Raspberry Pi 5 з використанням Flask (Python), що забезпечило надійну взаємодію з клієнтським застосунком та апаратною частиною, включно з обробкою вхідних даних, їх збереженням в базу даних та генерацією оптимізаційних рекомендацій. Паралельно було розроблено мобільний застосунок у Flutter, який дозволяє користувачам здійснювати моніторинг, запускати процес оптимізації, переглядати звіти, статистику та історію діагностики, при цьому забезпечено авторизацію користувачів,

адаптивний інтерфейс та зручну навігацію. Також було побудовано апаратну схему системи, яка включає сенсори температури, тиску та обертів, мікроконтролер Arduino для збору даних та Raspberry Pi для їх обробки, причому система підтримує роботу як у локальній Wi-Fi мережі, так і захищений VPN - доступ через ZeroTier. Крім того, розроблено програмну документацію, що містить необхідні специфікації, інструкції користувача, UML - діаграми та опис роботи модулів, а також описано вимоги до технічного та інформаційного забезпечення. На завершення, виконане тестування системи підтвердило її працездатність, відповідність поставленим вимогам та зручність для користувачів різних ролей, таких як інженери, власники мотоциклів та адміністратори.

Наукова новизна роботи полягає у розробці адаптивної інформаційної системи, яка об'єднує недорогі апаратні засоби з мобільним застосунком і надає реальні інструменти оптимізації роботи двигуна мотоцикла в режимі реального часу.

Практична цінність розробки полягає в можливості її використання у сервісних центрах технічного обслуговування мототехніки, навчальних закладах для демонстрації принципів роботи ДВЗ, а також для особистого контролю технічного стану мотоцикла.

Рекомендації щодо подальшого вдосконалення системи включають реалізацію прогнозування з використанням методів машинного навчання, підключення нових типів сенсорів, наприклад, вібраційних або газоаналізаторів, створення веб - версії інтерфейсу, а також розширення системи на інші типи двигунів і транспортних засобів.

Таким чином, результати, отримані в дипломному проєкті, повністю відповідають поставленій меті та завданням. Розроблена система продемонструвала ефективність, надійність і гнучкість, що дозволяє рекомендувати її до впровадження в реальні умови експлуатації.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. techtronic.com.ua. [Електронний ресурс]. Режим доступу до ресурсу: <https://techtronic.com.ua/princip-roboti-chotiritaktnogo-dviguna/> (дата звернення: 07.05.2025).
2. iits.org.ua. [Електронний ресурс]. Режим доступу до ресурсу: <https://iits.org.ua/> (дата звернення: 07.05.2025).
3. ukr.lpgtech.ua. [Електронний ресурс]. Режим доступу до ресурсу: https://ukr.lpgtech.ua/ugol-operezheniya-zazhiganiya_-detonaciya-v-dvigatele_-stepen-szhatiya_-vmt_-kompresiya (дата звернення: 07.05.2025).
4. sciencedirect.com. [Електронний ресурс]. Режим доступу до ресурсу: <https://www.sciencedirect.com/topics/engineering/crankshaft-speed> (дата звернення: 07.05.2025).
5. elprocus.com. [Електронний ресурс]. Режим доступу до ресурсу: <https://www.elprocus.com/intake-air-temperature-sensor/> (дата звернення: 07.05.2025).
6. tekniwiki.com. [Електронний ресурс]. Режим доступу до ресурсу: <https://www.tekniwiki.com/wiki/maf/> (дата звернення: 07.05.2025).
7. bosch-motorsport.com. [Електронний ресурс]. Режим доступу до ресурсу: <https://www.bosch-motorsport.com/content/downloads/Raceparts/en-GB/109918603.html> (дата звернення: 07.05.2025).
8. dynojet.com. [Електронний ресурс]. Режим доступу до ресурсу: <https://www.dynojet.com/power-vision/> (дата звернення: 07.05.2025).
9. holley.com. [Електронний ресурс]. Режим доступу до ресурсу: <https://www.holley.com/brands/racepak/?srsId=AfmBOorYhFNiZGFTnNSkBiRIJRyepjoucPirAuqiYUKWcys1AC9wmkpP> (дата звернення: 07.05.2025).

- 10.fastapi.tiangolo.com. [Электронный ресурс]. Режим доступа до ресурсу: <https://fastapi.tiangolo.com/uk/> (дата звернення: 07.05.2025).
- 11.sparxsystems.com. [Электронный ресурс]. Режим доступа до ресурсу: https://sparxsystems.com/enterprise_architect_user_guide/15.2/model_domains/example_business_modeling_diag.html (дата звернення: 07.05.2025).
- 12.responsive.io. [Электронный ресурс]. Режим доступа до ресурсу: <https://www.responsive.io/blog/write-business-requirements-document> (дата звернення: 07.05.2025).
- 13.altexsoft.com. [Электронный ресурс]. Режим доступа до ресурсу: <https://www.altexsoft.com/blog/functional-and-non-functional-requirements-specification-and-types/> (дата звернення: 07.05.2025).
- 14.geeksforgeeks.org. [Электронный ресурс]. Режим доступа до ресурсу: <https://www.geeksforgeeks.org/functional-vs-non-functional-requirements/> (дата звернення: 08.05.2025).
- 15.framework.scaledagile.com. [Электронный ресурс]. Режим доступа до ресурсу: <https://framework.scaledagile.com/nonfunctional-requirements> (дата звернення: 08.05.2025).
- 16.maxzosim.com. [Электронный ресурс]. Режим доступа до ресурсу: <https://www.maxzosim.com/sequence-diagrams/> (дата звернення: 08.05.2025).
- 17.aws.amazon.com. [Электронный ресурс]. Режим доступа до ресурсу: <https://aws.amazon.com/what-is/restful-api/> (дата звернення: 30.05.2025).
- 18.dbdiagram.io/d. [Электронный ресурс]. Режим доступа до ресурсу: <https://dbdiagram.io/d> (дата звернення: 31.05.2025).
- 19.arduino.ua. [Электронный ресурс]. Режим доступа до ресурсу: <https://arduino.ua/prod6990-plata-rozrobnika-esp32-c6-zero-mini-esp32-c6fh4-wifi-6-bluetooth-5> (дата звернення: 31.05.2025).
- 20.arduino.ua. [Электронный ресурс]. Режим доступа до ресурсу: <https://arduino.ua/ru/prod190-datchik-temperatyri-ds18b20->

[9f0f-b6db1f00d5cc?utm_source=chatgpt.com](https://doi.org/10.1017/9781017000000.000) (дата звернення:
02.05.2025).

30. bookauthority.org. [Електронний ресурс]. Режим доступу до ресурсу:
<https://bookauthority.org/profile/craig-smith> (дата звернення:
02.05.2025).

ДОДАТКИ

ДОДАТОК А

Специфікації вимог до системи

Item	Stereotype	Status	Difficulty	Priority
<input checked="" type="checkbox"/> ІС Оптимізації Параметрів Двигуна	Functional	Proposed	High	High
<input checked="" type="checkbox"/> 1 - Підсистема Аналізу Параметрів	Functional	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Аналіз Параметрів Двигуна	Functional	Proposed	High	High
<input checked="" type="checkbox"/> Виявлення аномалій даних	Functional	Proposed	Medium	Low
<input checked="" type="checkbox"/> Візуалізація вхідних даних	Functional	Proposed	Low	Low

Рисунок А.1 – Специфікація функціональних вимог (А)

<input checked="" type="checkbox"/> 2 - Підсистема Введення та Верифікації Даних	Functional	Proposed	High	High
<input checked="" type="checkbox"/> Верифікація Вхідних Даних	Functional	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Збереження сесій вводу	Functional	Proposed	High	Low
<input checked="" type="checkbox"/> Імпорт даних з файлу	Functional	Proposed	Low	Low
<input checked="" type="checkbox"/> Ручне введення параметрів	Functional	Proposed	Low	Low

Рисунок А.2 – Специфікація функціональних вимог (Б)

<input checked="" type="checkbox"/> 3 - Підсистема Звітуння	Functional	Proposed	High	High
<input checked="" type="checkbox"/> Експорт звітів (PDF, CSV)	Functional	Proposed	Medium	High
<input checked="" type="checkbox"/> Порівняння результатів оптимізації	Functional	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Формування зведеного звіту	Functional	Proposed	Low	High

Рисунок А.3 – Специфікація функціональних вимог (В)

<input checked="" type="checkbox"/> 4 - Підсистема Оптимізації	Functional	Proposed	High	High
<input checked="" type="checkbox"/> Алгоритм Оптимізації Параметрів	Functional	Proposed	High	Low
<input checked="" type="checkbox"/> Вибір стратегії оптимізації	Functional	Proposed	Medium	Low
<input checked="" type="checkbox"/> Забезпечення Зменшення Споживання Палива	Functional	Proposed	Low	Low
<input checked="" type="checkbox"/> Збереження профілів оптимізації	Functional	Proposed	Medium	Low
<input checked="" type="checkbox"/> Перевірка Відповідності Специфікаціям	Functional	Proposed	Medium	High
<input checked="" type="checkbox"/> Підтримка Тільки 4-тактних двигунів	Functional	Proposed	High	Medium
<input checked="" type="checkbox"/> Прогнозування результатів оптимізації	Functional	Proposed	Low	High

Рисунок А.4 – Специфікація функціональних вимог (Г)

Item	Stereotype	Status	Difficulty	Priority
<input checked="" type="checkbox"/> Інформаційна система з визначення оптимальних параметрів чотирьохтактного двигуна	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Безпека	Nonfunctional...	Proposed	High	Low
<input checked="" type="checkbox"/> Зберігання облікових даних у зашифрованому вигляді	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Передача даних через захищені протоколи (HTTPS, SSH)	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Рольовий доступ до системи (адміністратор, інженер)	Nonfunctional...	Proposed	Medium	Medium

Рисунок А.5 – Специфікація нефункціональних вимог (А)

<input checked="" type="checkbox"/> Зручність	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Інтуїтивно зрозумілий інтерфейс	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Мінімальні технічні вимоги до користувача для роботи з системою.	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Підтримка української мови	Nonfunctional...	Proposed	Medium	Medium

Рисунок А.6 – Специфікація нефункціональних вимог (Б)

<input checked="" type="checkbox"/> Масштабованість	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Можливість додавання нових модулів без зміни основного ядра	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Робота як на ПК, так і на Raspberry Pi 5 з сенсорним введенням	Nonfunctional...	Proposed	Medium	Medium

Рисунок А.7 – Специфікація нефункціональних вимог (В)

<input checked="" type="checkbox"/> Надійність	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Забезпечення 99% доступності протягом доби	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Збереження останнього стану сесії оптимізації після збою	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Щодобове автоматичне створення резервних копій	Nonfunctional...	Proposed	Medium	Medium

Рисунок А.8 – Специфікація нефункціональних вимог (Г)

<input checked="" type="checkbox"/> Продуктивність	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Інтерфейс має реагувати на дії користувача ≤ 0.5 сек у 90% випадків.	Nonfunctional...	Proposed	Medium	Medium
<input checked="" type="checkbox"/> Оптимізація параметрів має завершуватись ≤ 5 секунд	Nonfunctional...	Proposed	Medium	Medium

Рисунок А.9 – Специфікація нефункціональних вимог (Д)

ДОДАТОК Б

Діаграми послідовностей

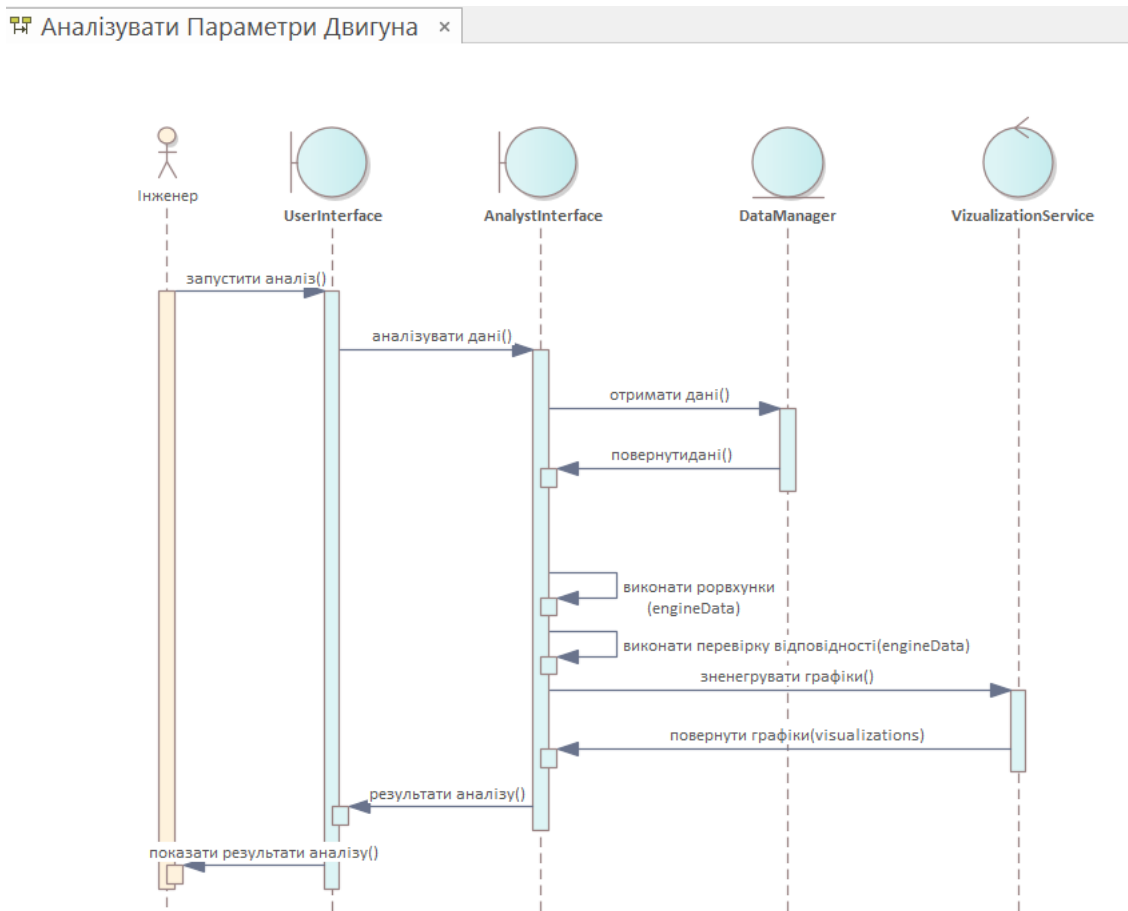


Рисунок Б.1 – Діаграма послідовності процесу «Аналізувати параметри двигуна»

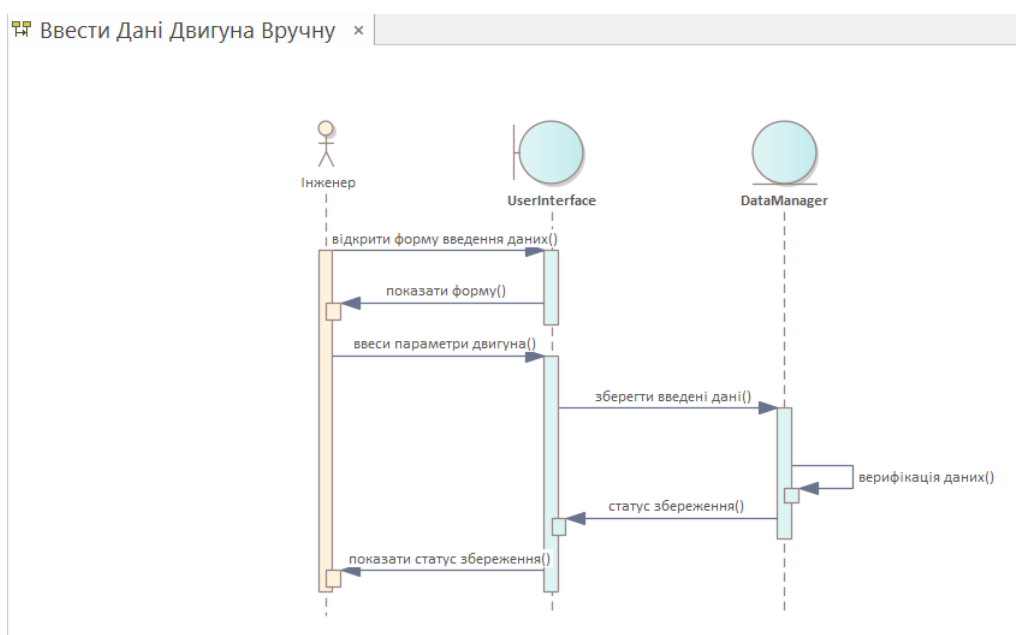


Рисунок Б.2 – Діаграма послідовності процесу «Ввести дані двигуна вручну»

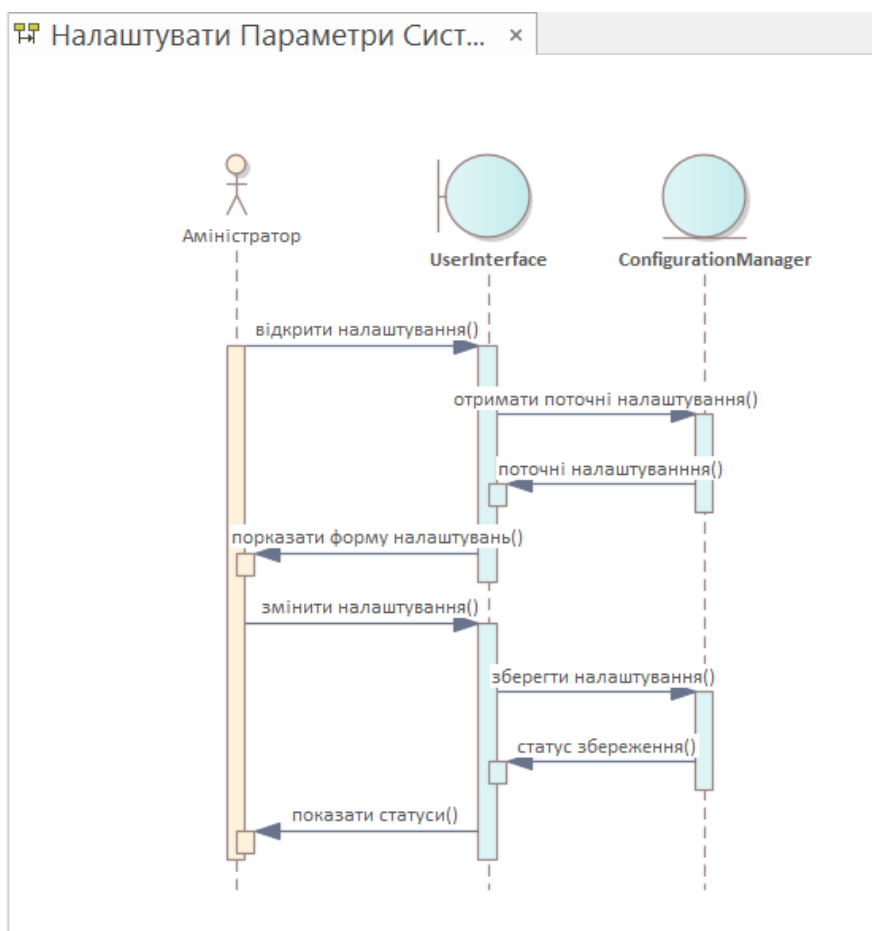


Рисунок Б.3 – Діаграма послідовності процесу «Налаштувати параметри»

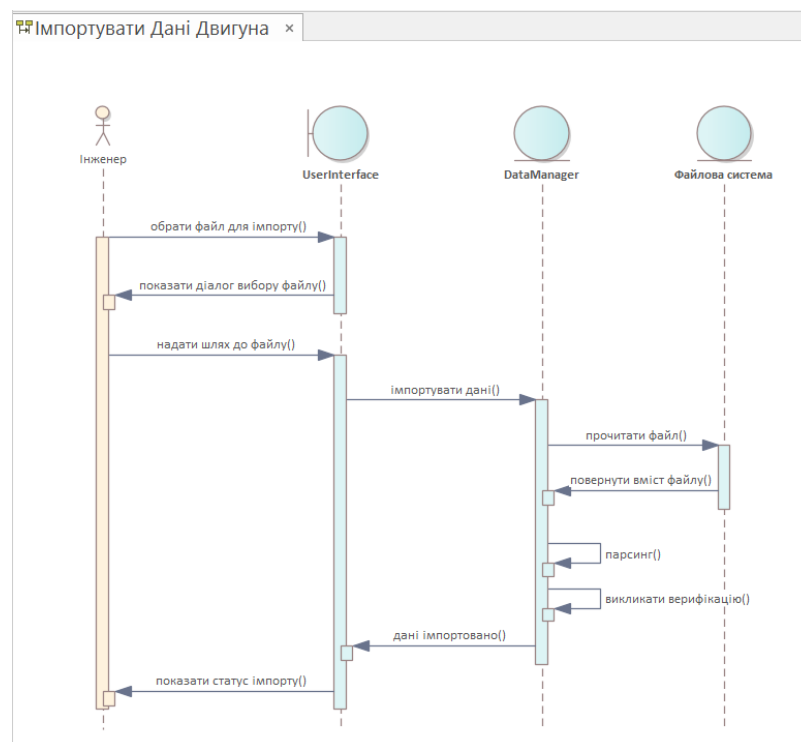


Рисунок Б.4 – Діаграма послідовності процесу «Імпортувати дані двигуна»

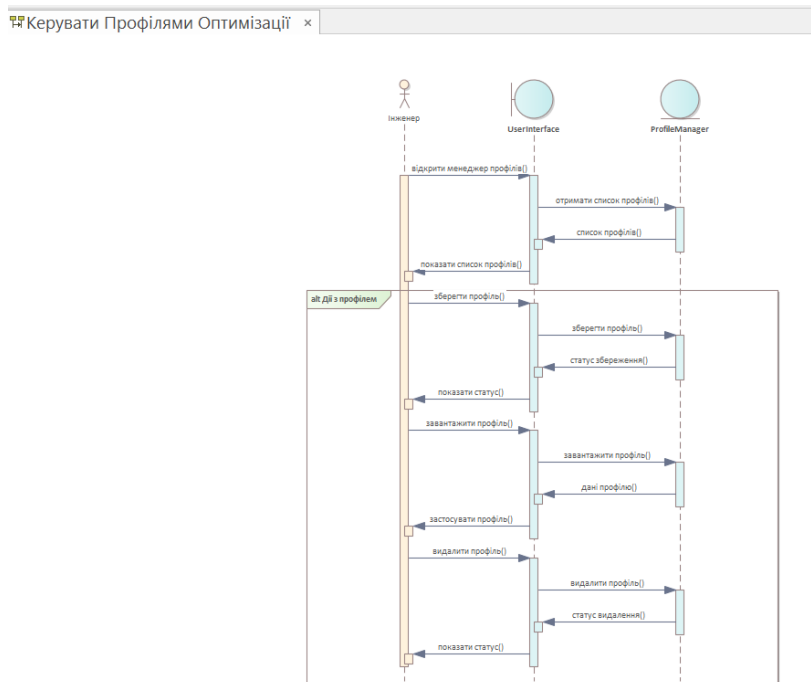


Рисунок Б.5 – Діаграма послідовності процесу «Керувати профілями оптимізації»

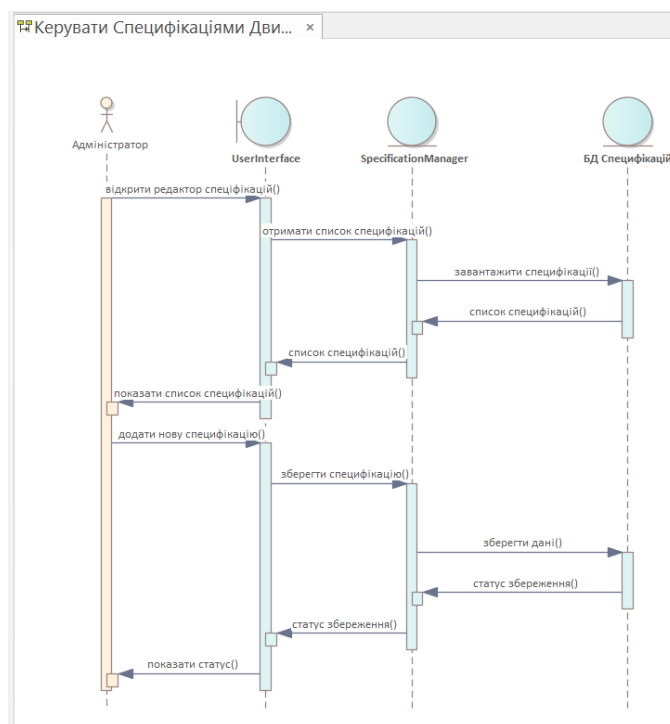


Рисунок Б.6 – Діаграма послідовності процесу «Керувати специфікаціями двигуна»

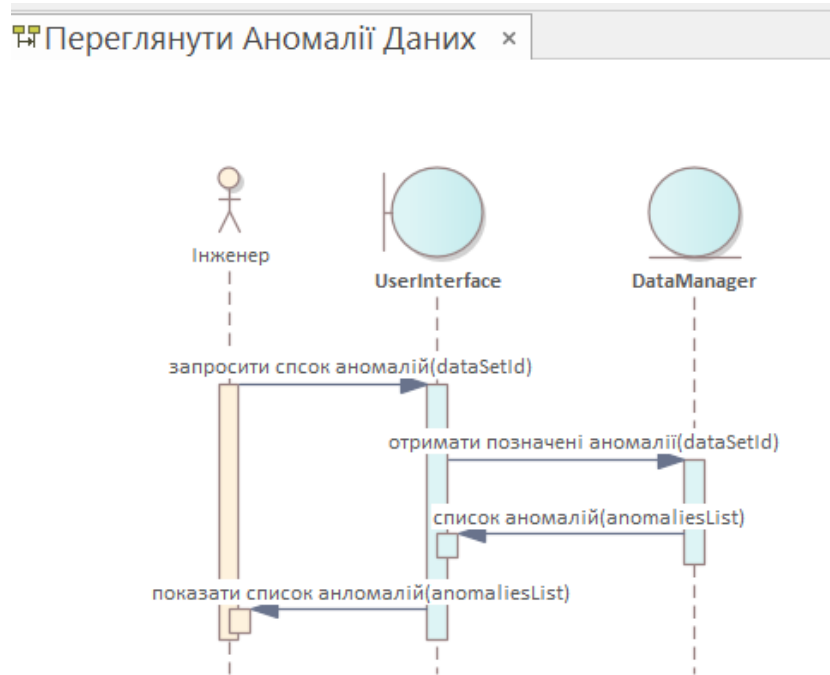


Рисунок Б.7 – Діаграма послідовності процесу «Переглянути аномалії даних»

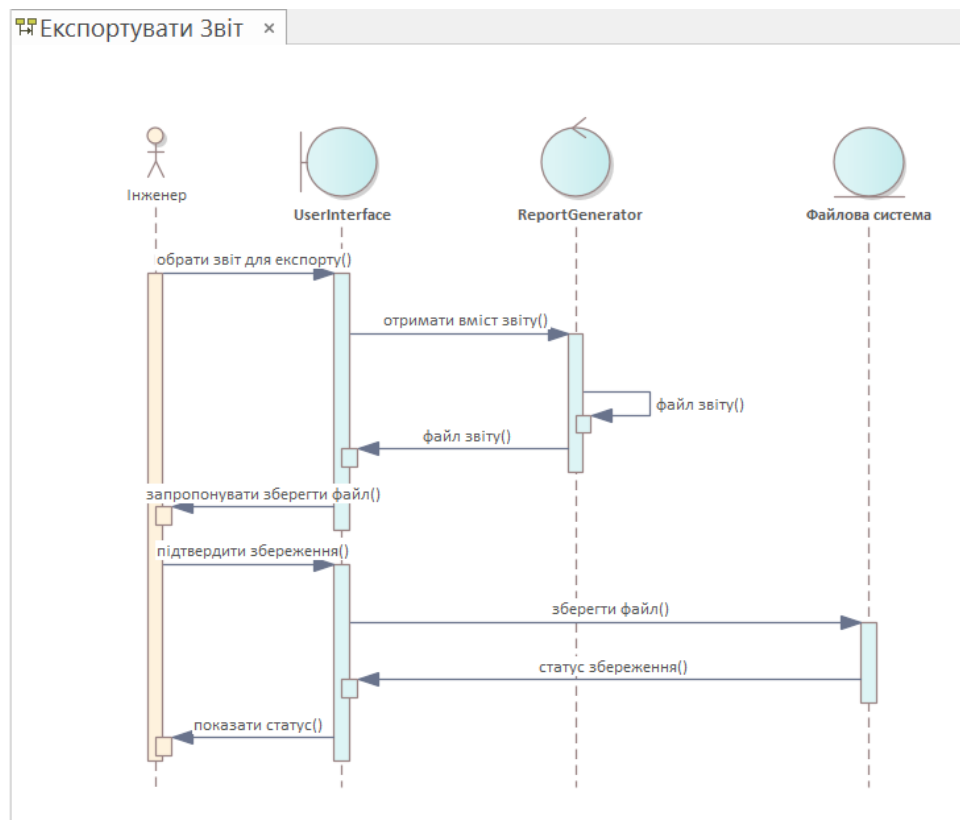


Рисунок Б.8 – Діаграма послідовності процесу «Експортувати звіт»

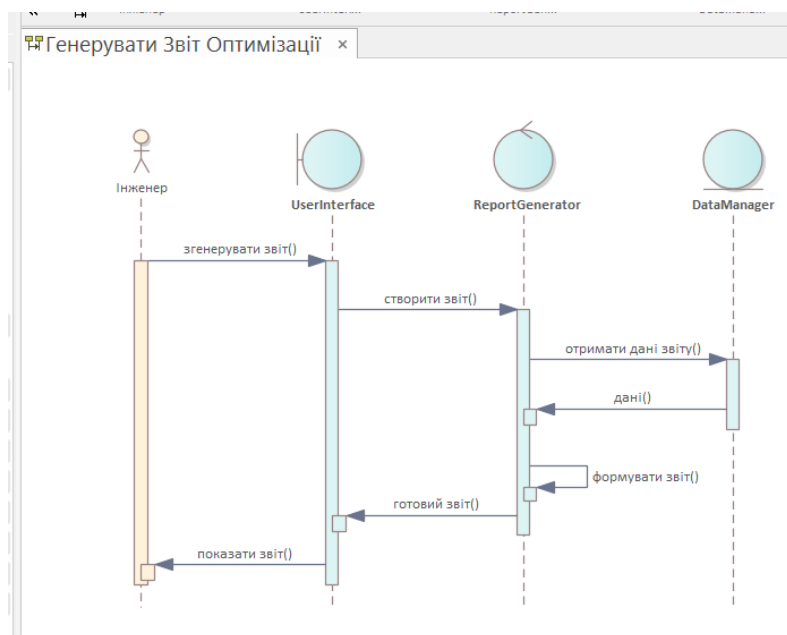


Рисунок Б.9 – Діаграма послідовності процесу «Генерувати звіт оптимізації»

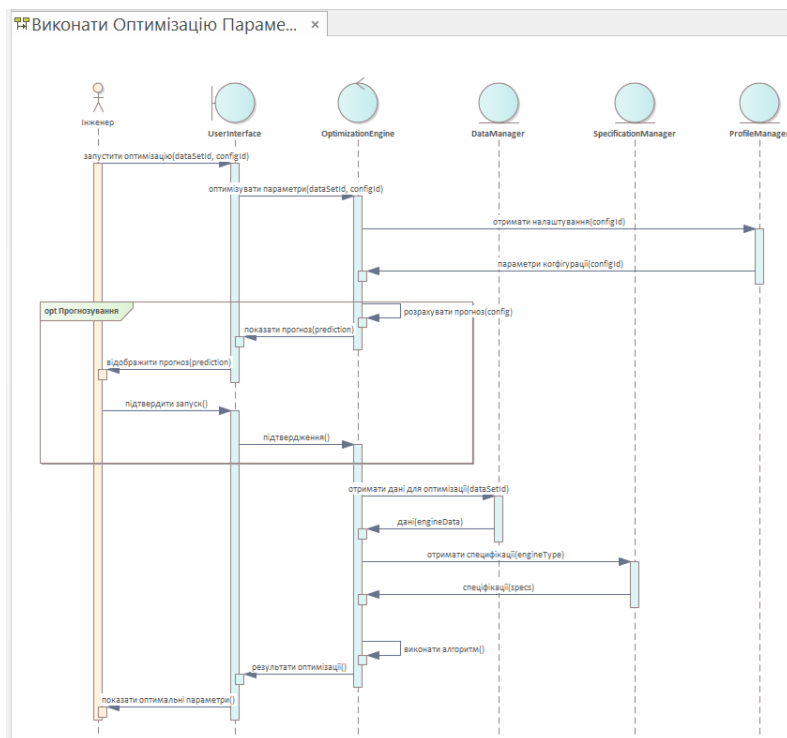


Рисунок Б.10 – Діаграма послідовності процесу

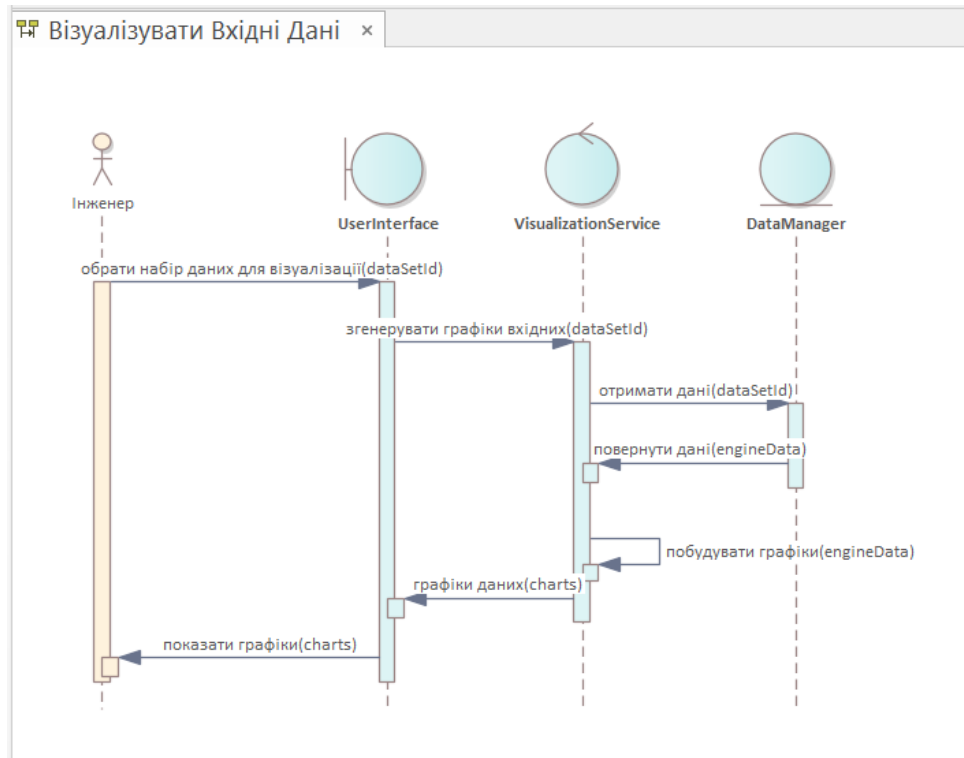


Рисунок Б.11 – Діаграма послідовності процесу «Візуалізувати вхідні дані»

ДОДАТОК В

Таблиці класифікаторів

Таблиця В.1 - Класифікатор користувачів

Ідентифікатор	Логін	ПІБ	Роль
USR_001	admin	Сидоренко А.В	ROLE_ADMIN
USR_002	engineer01	Коваль О.О.	ROLE_ENGINEER
USR_003	user2025	Іваненко В.В	ROLE_USER

Джерело: сформовано автором на основі виконаного дослідження

Таблиця В.2 - Класифікатор типів сенсорних даних

Код сенсора	Назва параметра	Одиниця вимірювання
SNS_RPM	Оберти двигуна	об/хв
SNS_TEMP	Температура двигуна	°C
SNS_PRS	Тиск масла	бар
SNS_OIL	Рівень мастила	%

Джерело: сформовано автором на основі виконаного дослідження

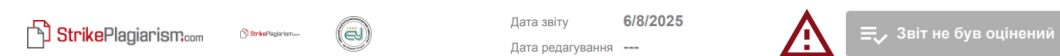
Таблиця В.3 - Класифікатор ролей користувачів

Код ролі	Назва ролі	Опис
ROLE_ADMIN	Адміністратор	Повний контроль над системою, управління користувачами
ROLE_ENGINEER	Інженер	Може переглядати дані, проводити оптимізацію
ROLE_USER	Користувач	Має обмежений доступ: перегляд своїх результатів

Джерело: сформовано автором на основі виконаного дослідження

ДОДАТОК Г

Звіт подібності



Звіт подібності

метадані

Назва організації

Kyiv National Economic University named after Vadym Hetman KNEU

Заголовок

Розроблення інформаційної системи з визначення оптимальних параметрів чотирьохтактного двигуна мотоцикла

Автор

Науковий керівник / Експерт

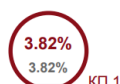
Борисенко Олександр Андрійович Тішков Б.О.

підрозділ

кафедра інформаційних систем в економіці

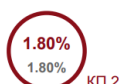
Обсяг знайдених подібностей

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.



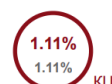
25

Довжина фрази для коефіцієнта подібності 2



12129

Кількість слів



95910

Кількість символів

Тривога

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про МОЖЛИВІ маніпуляції в тексті. Спотворення в тексті можуть мати навмисний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв		2
Інтервали		2
Мікропробіли		2
Білі знаки		0
Парафрази (SmartMarks)		26

Подібності за списком джерел

Нижче наведений список джерел. В цьому списку є джерела із різних баз даних. Копір тексту означає в якому джерелі він був знайдений. Ці джерела і значення Коефіцієнту Подібності не відображають прямого плагіату. Необхідно відкрити кожне джерело і проаналізувати зміст і правильність оформлення джерела.

10 найдовших фраз		Копір тексту
ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	https://ir.kneu.edu.ua/bitstreams/22acc4f6-0461-4d18-a9a4-ff2fd8c10ff/download	44 0.36 %
2	https://ir.kneu.edu.ua/bitstreams/f2d2da61-14eb-4ba9-b82f-2663cc9ae485/download	43 0.35 %
3	https://kneu.edu.ua/userfiles/Faculty_of_Information_Systems_and_Technology/kaf%20ise/Metod_rekom_en_6_050101.doc	28 0.23 %

Рисунок Г.1 – Звіт подібності