

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій**

«На правах рукопису»
УДК 004.45

До захисту допущено:
Завідувач кафедри
_____ Олександр РОЛІК
«__» _____ 2024 р.

Магістерська дисертація

на здобуття ступеня магістра

за освітньо-професійною програмою

«Інформаційне забезпечення робототехнічних систем»

зі спеціальності 126 «Інформаційні системи та технології»

**на тему: «Система стеження за повітряними та наземними цілями мобільного
робота на основі комп'ютерного зору»**

Виконав:
студент 2 курсу, групи ІК-32мп
Кравчук Максим Олексійович

Керівник:
доцент каф. ІСТ, к.т.н., доц.
Ткач Михайло Мартинович

Рецензент:
доцент каф. ІП, к.т.н., доц.
Лісовиченко Олег Іванович

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____

Київ – 2024 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Рівень вищої освіти – другий (магістерський)

Спеціальність – 126 «Інформаційні системи та технології»

Освітньо-професійна програма «Інформаційне забезпечення робототехнічних систем»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Олександр РОЛІК

«___» _____ 2024 р.

ЗАВДАННЯ
на магістерську дисертацію студенту
Кравчуку Максиму Олексійовичу

1. Тема дисертації «Система стеження за повітряними та наземними цілями мобільного робота на основі комп'ютерного зору», науковий керівник дисертації Ткач Михайло Мартинович, к.т.н., доц., затверджені наказом по університету від «08» 11 2024 р. № 5016-с
2. Термін подання студентом дисертації «09» 12 2024 р.
3. Об'єкт дослідження: система стеження за статичними та динамічними цілями.
4. Вихідні дані: застосунок системи стеження за повітряними та наземними цілями мобільного робота на основі комп'ютерного зору, вісім додатків з графічним матеріалом
5. Перелік завдань, які потрібно розробити: оглянути проблему та існуючі рішення, сформулювати функціональні та нефункціональні вимоги до системи, обрати та обґрунтувати елементи та технології, спроектувати структурну схему системи, базу даних, реалізувати бізнес-логіку системи та інтерфейси користувача та протестувати систему.
6. Орієнтовний перелік графічного (ілюстративного) матеріалу: діаграма прецедентів, BPMN діаграма процесу стеження, блок-схема алгоритму класифікації об'єкту стеження, діаграма послідовності процесу стеження, структурна схема системи, ER-діаграма бази даних, схематична карта сторінок інтерфейсу, схема мобільного робота.

7. Дата видачі завдання 02.09.2024 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1.	Формування мети та задач дослідження	08.09	
2.	Аналіз предметної області, огляд існуючих аналогів	13.09	
3.	Розробка математичної моделі	13.09	
4.	Аналіз та розробка інформаційного забезпечення	20.09	
5.	Опис інформаційного забезпечення	27.09	
6.	Розробка програмного забезпечення	04.10	
7.	Опис програмного та технічного забезпечення	18.10	
8.	Аналіз результатів	25.10	
9	Оформлення щоденника, звіту і складання заліку з практики	01.11	
10	Подання дисертації до захисту	09.12	

Студент

Максим КРАВЧУК

Науковий керівник

Михайло ТКАЧ

РЕФЕРАТ

Система стеження за повітряними та наземними цілями мобільного робота на основі комп'ютерного зору: 108 с., 31 рис., 47 табл., 7 формул, 9 дод., 34 джерел.

МОБІЛЬНІ РОБОТИ, КОМП'ЮТЕРНИЙ ЗІР, ШТУЧНИЙ ІНТЕЛЕКТ, АЛГОРИТМИ СТЕЖЕННЯ, АВТОНОМНА НАВІГАЦІЯ, НЕЙРОННІ МЕРЕЖІ, ОБРОБКА ЗОБРАЖЕНЬ, PYTHON, LUA.

Об'єкт дослідження: система стеження за статичними та динамічними цілями.

Предмет дослідження: алгоритми стеження за цілями в різних умовах та сценаріях застосування мобільних роботів.

Актуальність магістерської дисертації зумовлена зростанням ролі мобільних роботів, які, завдяки інтеграції з технологіями штучного інтелекту та комп'ютерного зору, забезпечують ефективне стеження за цілями в динамічних умовах. Це відкриває нові можливості для автономних систем у промисловості, медицині, сільському господарстві та військовій сфері, підвищуючи рівень автоматизації, точності та безпеки.

Метою роботи є підвищення точності визначення місцезнаходження цілі для надійної навігації робота під час стеження використовуючи алгоритм комп'ютерного.

Для досягнення поставленої мети були поставлені та виконані наступні задачі:

- оглянути проблему та існуючі рішення;
- сформулювати функціональні та нефункціональні вимоги до системи;
- обрати та обґрунтувати елементи та технології;
- спроектувати структурну схеми системи, базу даних;
- реалізувати бізнес-логіку системи та інтерфейси користувача;
- протестувати систему.

ABSTRACT

Tracking system for air and ground targets of a mobile robot based on computer vision: 108 pages, 31 figures, 47 tables, 7 formulas, 9 appendices, and 34 references.

MOBILE ROBOTS, COMPUTER VISION, ARTIFICIAL INTELLIGENCE, TRACKING ALGORITHMS, AUTONOMOUS NAVIGATION, NEURAL NETWORKS, IMAGE PROCESSING, PYTHON, LUA.

Object of research: a tracking system for static and dynamic targets.

Subject of research: tracking algorithms for targets under various conditions and scenarios of mobile robot applications.

The relevance of the Master's thesis is determined by the growing role of mobile robots, which, through integration with artificial intelligence and computer vision technologies, ensure effective target tracking in dynamic conditions. This opens new opportunities for autonomous systems in industry, medicine, agriculture, and the military, enhancing automation, precision, and safety.

The aim of the thesis is to improve the accuracy of target localization for reliable robot navigation during tracking using computer vision algorithms.

To achieve this aim, the following tasks were set and accomplished:

- review the problem and existing solutions;
- define functional and non-functional system requirements;
- select and justify the elements and technologies;
- design the system's structural scheme and database;
- implement the business logic and user interfaces of the system;
- test the system.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ.....	9
ВСТУП.....	10
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ	12
1.1 Загальний огляд проблеми	12
1.2 Огляд та аналіз існуючих рішень систем для стеження за цілями мобільних роботів	13
1.2.1 DJI Drones.....	13
1.2.2 FLIR Black Hornet.....	15
1.2.3 Роботи KEENON	16
1.3.1 Алгоритм «Kernelized Correlation Filters».....	20
1.3.2 Формування мети та задач роботи.....	22
Висновки до розділу.....	24
2 ФОРМУВАННЯ ВИМОГ ДО СИСТЕМИ.....	25
2.1 Формування функціональних вимог	25
2.1.1 Взаємодія системи з користувачами певних ролей	26
2.1.2 Взаємодія системи з користувачами ролі Адміністратор	31
2.1.3 Взаємодія системи з користувачами ролі Пілот мобільного боту	37
2.1.4 Взаємодія інтерфейсу системи з системою Мобільного робота	39
2.2 Формування нефункціональних вимог	45
Висновки до розділу.....	46
3 РОЗРОБКА СИСТЕМИ	48
3.1 Вибір та обґрунтування елементів та технологій	48
3.1.1 Вибір алгоритму комп'ютерного зору для стеження за об'єктом	48

3.1.2 Вибір середовища для симуляції системи	50
3.1.4 Вибір технологій та розробка інтерфейсу симуляції.....	52
3.1.5 Вибір, проектування та інтеграція сховища даних	54
3.1.6 Вибір нейронної мережі для виявлення та класифікації об'єктів	54
3.1.7 Програмне забезпечення для розробки застосунку	56
3.2 Структурна схема системи	57
3.3 Проектування бази даних	58
3.4 Реалізація бізнес–логіки системи	61
3.5 Опис інтерфейсу користувача та інструкція використання застосунку	63
Висновки до розділу.....	70
4 ТЕСТУВАННЯ СИСТЕМИ.....	72
4.1 Мануальне тестування	72
4.1.1 Загальні тест–кейси системи	72
4.1.2 Тест–кейси ролі «Адміністратор»	74
4.1.3 Тест–кейси ролі «Пілот мобільного робота».....	77
4.1.4 Тестування функціоналу системи симуляції мобільного робота.....	80
Висновок до розділу.....	83
5 СТАРТАП ПРОЕКТ	84
5.1 Опис ідеї проекту	84
5.2 Технологічний аудит ідеї проекту	85
5.3 Аналіз ринкових можливостей запуску стартап–проекту	87
5.4 Розроблення ринкової стратегії проекту.....	96
5.5 Розроблення маркетингової програми стартап–проекту	99
Висновки до розділу.....	103
ВИСНОВОК.....	105

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	107
ДОДАТОК А	110
ДОДАТОК Б	111
ДОДАТОК В	112
ДОДАТОК Г	113
ДОДАТОК Д	114
ДОДАТОК Е	115
ДОДАТОК Ж	116
ДОДАТОК И	117
ДОДАТОК К	118

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ

КЗ – Комп'ютерний зір

ІІІ – Штучний інтелект

CNN – Convolutional Neural Network

DWA – Dynamic Window Approach

Faster R-CNN – Faster Region-based Convolutional Neural Network

HSV – Hue, Saturation, Value

KCF – Kernelized Correlation Filters

LiDAR – Light Detection and Ranging

LRR – Linear Ridge Regression

SLAM – Simultaneous Localization and Mapping

SWOT – Strengths, Weaknesses, Opportunities, Threats

UGC – User-Generated Content

UGV – Unmanned Ground Vehicle

YOLO – You Only Look Once

ВСТУП

Ще на початку 20-го століття, роботи та штучний інтелект були літературними засобами вираження своїх надії та страхів щодо технологій майбутнього. Як результат таких переживань з'явилося багато кінокартин, що поклали початок співіснування людей та роботів.

Загальновідомо, що розвиток мобільних роботів та розвиток їх симбіозу з технологіями штучного інтелекту є найважливішими секторами досліджень нашого часу. Серед усіх галузей розвитку роботів можна виділити декілька найактивніших ніш, а саме: індустрія, агрикультура, медицина та армія.

Аналізуючи розвиток роботів в індустрії можна підкреслити інтеграцію систем ШІ та поєднання їх із системами Інтернету речей. Дані інтеграції забезпечують підвищення продуктивності таких робіт, як зварювання, складання та перевірка продукції, аналізуючи великі обсяги даних на всіх етапах виробництва та повідомляючи своєчасно про брак. Схоже використання сучасних технологій можна побачити у сфері агрикультури. Використовуючи машинне навчання на великих обсягах даних, зібраних за допомогою датчиків або іншого записуючого обладнання, мобільні роботи можуть допомагати фермерам приймати більш обґрунтовані рішення, що призведе до підвищення продуктивності та оптимізації їхніх ресурсів.

Інноваційним трендом використання ШІ та комп'ютерного зору у медицині – є виявлення онкологічних захворювань. Такі результати є достатньо точними, щоб допомагати хірургам своєчасно приймати рішення і в результаті економити секунди часу, що можуть врятувати пацієнта.

В армії використання мобільних, стаціонарних роботів є критично важливою ланкою досліджень, бо у першу чергу важливо зберігати якомога більше особового складу. Однією з сфер, де автономні роботи показали величезні перспективи, є безпілотні літальні апарати, або дрони. Ці безпілотники можуть бути оснащені передовими датчиками та камерами, що дозволяє їм збирати розвідувальну та розвідувальну інформацію з повітря. Потім цю інформацію можна використовувати для прийняття військових рішень, не наражаючи людей–пілотів на небезпеку. Іншим

прикладом автономного робота для ведення війни є безпілотний наземний транспортний засіб або UGV. Ці машини призначені для роботи на землі та можуть використовуватися для різноманітних завдань, від транспортування припасів і обладнання до патрулювання небезпечних зон.

Отже проаналізувавши інноваційні прориви у сферах інтеграції мобільних або стаціонарних роботів, поєднання них із системами штучного інтелекту, можна виділи, що в більшості секторів лідирує комбінації роботів із технологіями розпізнавання на базі комп'ютерного зору.

Об'єкт дослідження: система стеження за статичними та динамічними цілями.

Предмет дослідження: алгоритми стеження за цілями в різних умовах та сценаріях застосування мобільних роботів.

Метою роботи є підвищення точності визначення місцезнаходження цілі для надійної навігації робота під час стеження використовуючи алгоритм комп'ютерного зору імплементований у середовище симуляції і розробка зручного інтерфейсу комунікації користувача із системою.

Магістерська дисертація складається з наступних розділів:

- аналіз предметної області та існуючих рішень;
- формування вимог до системи;
- розробка системи;
- тестування системи;
- стартап–проект.

Також вона включає 31 рисунок, 47 таблиць, 7 формул, 9 додатків та 34 джерела.

Загальний обсяг магістерської дисертації 108 сторінок.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ

1.1 Загальний огляд проблеми

У сучасному світі мобільні роботи стали невід'ємною частиною багатьох аспектів людського життя, активно впливаючи на промисловість, медицину, сільське господарство та навіть військові сфери. Розвиток технологій штучного інтелекту та комп'ютерного зору відкрив нові горизонти для застосування робототехніки, підвищуючи ефективність та безпеку виконання різноманітних завдань.

До переваг використання мобільних роботів відносять зростання швидкості та точності виконання процесів, що раніше вимагали значних людських ресурсів. В поєднанні із ШІ та машинним навчанням роботи виконують поставлені задачі навіть краще, бо можна сказати, що системи комп'ютерного зору для роботи, як очі для людей.

Різноманітність мобільних роботів вражає своєю широтою та функціональністю. Можна виділити такі види роботів:

- наземні роботи, які пересуваються на колесах або гусеницях, призначені для виконання завдань на рівній поверхні чи в складному рельєфі;
- підводні роботи використовуються для дослідження морських глибин, проведення ремонтних робіт на глибоководних об'єктах та моніторингу екологічного стану водних екосистем;
- безпілотні літальні апарати, які завдяки своїй мобільності та здатності досягати важкодоступних місць стають незамінними у багатьох галузях.

Кожен з таких видів може виконувати задачу стеження, навіть якщо середовище роботи роботів може сильно відрізнятися від робота до робота. Це досягається індивідуальним підходом проектування та розробки симбіотичної системи робота з системою штучного інтелекту, яка займає дуже важливу роль у виконанні поставлених задач.

Розуміючи, що варіативність роботів безмежна і кожен робот унікальний, наступними розділами буде проаналізовано існуючі системи для стеження за цілями мобільних роботів та підходи комп'ютерного зору для мобільних роботів.

1.2 Огляд та аналіз існуючих рішень систем для стеження за цілями мобільних роботів

1.2.1 DJI Drones

DJI, є провідним світовим виробником цивільних дронів і технологій аерофотозйомки[1].

Їхні дрони відомі своїми передовими функціями, надійністю та простотою використання, що робить їх популярними як серед професіоналів, так і серед любителів.

На рисунку 1.1 зображено дрон DJI Air 2s.



Рисунок 1.1 – Дрон DJI Air 2s

З рисунка видно, що дрон оснащений камерами, ультразвуковими та інфрачервоними датчиками, дані з яких обробляються та аналізуються алгоритмами комп'ютерного зору для виявлення та відстеження вибраного об'єкта. Потім система керування польотом використовує цю інформацію для коригування положення та орієнтації дрона, забезпечуючи плавне відстеження та уникнення перешкод. Зв'язок

між алгоритмами КЗ і системами керування дроном здійснюється внутрішньо, що дозволяє реагувати в режимі реального часу. Така система має назву ActiveTrack.

ActiveTrack була вперше представлена DJI у 2016 році під час випуску дрона Phantom 4, що зображений на рисунку 1.2. Це ознаменувало значний прогрес у споживчих технологіях дронів, надавши можливості професійного автономного відстеження та відеозйомки ширшій аудиторії.



Рисунок 1.2 – Дрон DJI Phantom 4

Незважаючи на те, що компанія DJI публічно не оприлюднила алгоритми, які використовуються в їхніх дронах, система ActiveTrack, ймовірно, включає в себе добре відомі алгоритми та моделі машинного навчання, а саме:

- згорткові нейронні мережі, наприклад, Faster R-CNN, YOLO. Дані алгоритми використовуються для виявлення та класифікації об'єктів у полі зору камери;
- трекери на основі фільтрів, наприклад, Kernelized Correlation Filter. Використовується для ефективного відстеження положення об'єкта в кадрі;

– алгоритми одночасної локалізація і картографування для побудови і оновлення мапи невідомого оточення з одночасним відстежуванням місцеположення рухаючись по ньому.

Для управління дроном користувач може використовувати пульт керування DJI, або власний смартфон із встановленим застосунком від DJI. Користувачі можуть переглядати телеметричні дані в реальному часі, включаючи висоту, швидкість і відстань до перешкод.

1.2.2 FLIR Black Hornet

FLIR Black Hornet – це серія безпілотних літальних апаратів, класу мікро, розроблених для військових і правоохоронних органів компанією FLIR Systems[2].

Класифікація дронів відбувається за такими показниками як маса, дальність та висота польоту, а також тривалість роботи. З рисунку 1.3 видно, що даний дрон якраз підходить під клас мікро.



Рисунок 1.3 – FLIR Black Hornet 4

Дрон оснащений бортовими камерами, які можуть передавати інформацію з них через захищене бездротове з'єднання на контролер оператора. Дані з камер обробляються в режимі реального часу, що дозволяє використовувати дрон для

виявлення об'єктів, відстеження та допомога в навігації особового складу. Аналогічно до дронів компанії DJI, дані дрони Black Hornet використовують симбіоз технологій машинного навчання для покращення таких алгоритмів як:

- SLAM, для побудови і оновлення мапи невідомого оточення;
- згорткові нейронні мережі, для сегментації зображення на регіони, щоб зрозуміти навколишнє середовище на піксельному рівні для точної навігації;
- алгоритмами планування шляху, для створення найкоротшого маршруту між точками доставки;
- алгоритмами уникнення перешкод, наприклад, DWA, VFH. Це дозволяє здійснювати навігацію в режимі реального часу навколо неочікуваних перешкод.

Можна впевнено сказати, що дрони такого класу є безумовним проривом у сфері військової розвідки. Також, це підтверджується тим, що дрони використовуються понад 20 військовими угрупованнями та понад 30 країнами.

1.2.3 Роботи KEENON

Keenon Robotics, є провідною технологією компанії, заснованої в Китаї, що спеціалізується на розробці роботів для різних галузей промисловості[3].

Їх роботи, що зображені на рисунку 1.4, для автоматизації доставки їжі в ресторанах, готелях та інших закладах, підвищуючи ефективність роботи та покращуючи досвід обслуговування клієнтів.

Роботи оснащені камерами та різними датчиками, як-от LiDAR, ультразвукові та інфрачервоні датчики, які збирають дані про навколишнє середовище. Ці дані обробляються бортовими комп'ютерами, що працюють з алгоритмами комп'ютерного зору, що дозволяє роботу розуміти навколишнє середовище та взаємодіяти з ним.

Аналогічно для попередніх роботів, системи Keenon не публікують точні алгоритми, що вони використовують. Проте, можна сказати точно, що у даних систем присутні реалізації алгоритмів SLAM, DWA, VFH. Також, роботи вміють стежити за

їжею та відвідувачами, що означає використання нейронних мереж, наприклад, Faster R-CNN, або YOLO.



Рисунок 1.4 – Приклад роботів Keenon для доставки їжі

Користувачі можуть взаємодіяти з роботими використовуючи інтерфейс сенсорного екрана, мобільний додаток та голосові команди.

1.3 Огляд та аналіз існуючих підходів використання комп'ютерного зору для стеження за об'єктами

Задача стеження за об'єктами може бути охарактеризована як математично, так і з точки зору алгоритмів комп'ютерного зору.

Математично, стеження за об'єктом можна описати як процес знаходження певної характеристики множини даних про локацію об'єкта в визначений проміжок часу. Розуміючи цю концепцію можна виділити математичні методи, що можуть задовольнити поставлену задачу.

Було розглянуто такі математичні засоби, як: Евклідова відстань, коефіцієнт кореляції Метью, коефіцієнт Бгаттачар'я та розходження Кульбака – Лейблера[4]. Поверхнево проаналізувавши їх, відразу можна визначити схожість усіх цих метрик,

а саме – порівняння пар. У випадку Евклідової відстані – це порівняння координат точок, коефіцієнт кореляції Метью використовується для оцінки якості бінарної класифікації, коефіцієнт Бгаттачар'я – оцінка схожості або перекриття між двома ймовірнісними розподілами, дивергенція Кульбака–Лейблера у свою чергу характеризує наскільки один ймовірнісний розподіл відрізняється від іншого розподілу.

Розглянуто найпростішу метрику – Евклідову відстань. Аналізується ситуація, де рух об'єкта представлений двома точками з координатами $(a1; a2)$ і $(b1; b2)$ у просторі в моменти часу $t1$ та $t2$. Поставлено задачу визначити відстань, на яку необхідно пересунути об'єкт камери. Для цього використовується формула Евклідової відстані, що задається як:

$$d = \sqrt{(a1 - b1)^2 + (a2 - b2)^2}. \quad (1.1)$$

Отже, обчисливши відстань d , дані відправляються до модулю, що керує рухом камери для корекції її положення.

Також, цікавою метрикою є відстань Бгаттачар'я. Даний метод широко застосовується в областях розпізнавання образів, теорії інформації та машинного навчання для вимірювання подібності між різними наборами даних або моделями. Поставлено задачу визначити коефіцієнт Бхаттачар'я для ймовірнісних розподілів P і Q . Формули 1.2 та 1.3, що наведені далі, визначають коефіцієнт для неперервних та дискретних розподілів.

$$BC = \int \sqrt{P(x) * Q(x)} dx, \quad (1.2)$$

$$\sum \sqrt{P(i) * Q(i)}. \quad (1.3)$$

З точки зору алгоритмів комп'ютерного зору, стеження за об'єктами є однією із найскладніших операцій, бо головною характеристикою опису об'єкта становиться

множина кадрів. Беручи до уваги не ідеальність камер, оточення, з'являються такі труднощі, як: зміни освітлення, кольору, текстури об'єкту. Навіть коли об'єкт починає повертатися у кадрі, алгоритми можуть сприймати це як шум. Саме через такі труднощі у реалізації та покращенні алгоритмів, сьогодні задачі стеження за об'єктом у секторі комп'ютерного зору є дуже активною та важливою нішою серед вчених та ентузіастів.

В цілому, процес стеження, що зображений на рисунку 1.5, є циклічним і складається з 5 етапів, а саме:

- імпортування вхідних даних;
- обробка даних алгоритмом комп'ютерного зору;
- аналіз результатів роботи алгоритму комп'ютерного зору;
- візуалізація результату;
- оновлення вхідних даних.

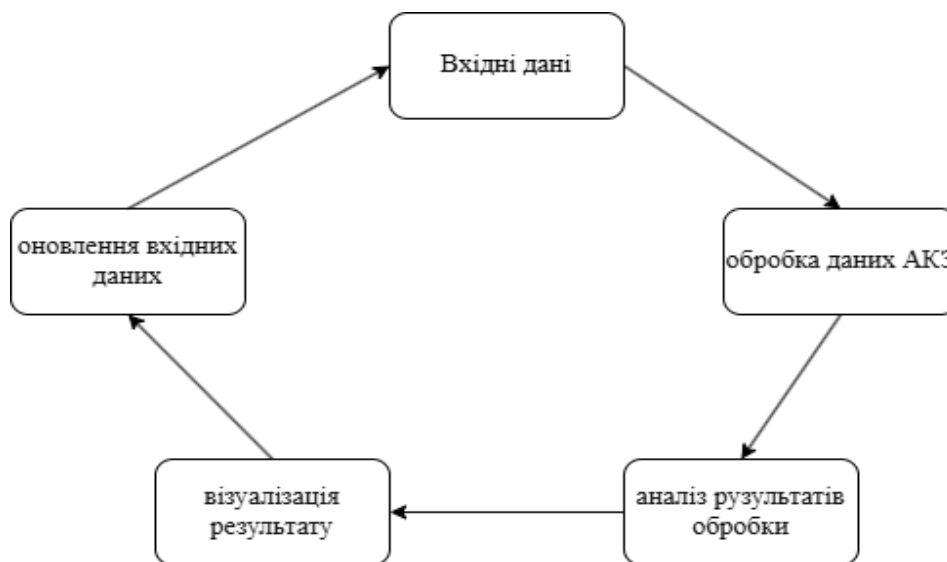


Рисунок 1.5 – Діаграма процесу стеження

Серед усіх відомих алгоритмів стеження на ціллю можна виділити дві великі категорії алгоритмів, а саме: генеративні та розрізнявальні. Головна відмінність яких полягає у використанні візуальних даних для виконання поставлених завдань. Виділяючи найголовніше, генеративні алгоритми вивчають основний розподіл візуальних функцій для кожного класу, що дозволяє їм створювати або

реконструювати зображення. Вони спрямовані на те, як об'єкти або сцени візуально представлені в різних умовах. Проте розрізнявальні алгоритми зосереджуються на визначенні меж або відмінностей між візуальними класами без моделювання того, як генеруються самі дані. Використовуючи такі алгоритми для виявлення, класифікація та відстеження об'єктів, вони зосереджуються на шаблонах, які відрізняють цільові об'єкти від фону чи інших класів.

Проаналізувавши обидві категорії, я б хотів більш детально оглянути такий алгоритм, як: Kernelized Correlation Filters.

1.3.1 Алгоритм «Kernelized Correlation Filters»

Методи стеження на базі ядрових кореляційних фільтрів виділяються тим, що використовують великі набори даних, що зазвичай складаються з усіх можливих варіантів вікна області інтересу. Цей датасет формується на етапі обчислення маючи заданий розмір вікна. Далі алгоритм KCF навчається на цих даних, а отже аналізує і запам'ятовує ключові деталі на кожному із зображень, що допомагають у визначенні місцезнаходження об'єкта в кожному наступному кадрі.

Зберігаючи цю інформацію, алгоритм дозволяє зручно відтворювати положення об'єкта з часом, більш того, даний датасет може бути використаний іншими моделями для тренування, але в випадку поставлення задачі стеження на об'єктом у реальному часі, некоректний вибір алгоритму може призвести до зниження ефективності системи стеження.

Відомо, що традиційні алгоритми прагнуть використовувати більшу кількість позитивних даних. Позитивними називають дані, що містять об'єкт пошуку. Проте, негативні дані також містять важливу інформацію, яку розрізнявальні алгоритми можуть використовувати для покращення точності.

Алгоритм KCF, як раз, втілює такі характеристики, як: швидкість обробки зображень та низька затратність ресурсів під час аналізу. Такі фантастичні показники досягаються тим, що алгоритм використовує два важливих підходи, а саме: ядровий

триюк (англ. kernel trick) у двовимірному просторі та властивості підвиду матриць Тепліца у перетвореннях Фур'є.

КСР розглядає задачу відстеження об'єкта як задачу класифікації, де необхідно відрізнити цільовий об'єкт від фону. Це формулюється через задачу лінійної регресії з регуляризацією (англ. Linear Ridge Regression), мета якої – знайти такі вагові коефіцієнти, що забезпечують найменшу похибку класифікації між прогнозованими і реальними значеннями. Формулою 1.4 визначають формулу мінімізації вагового вектору w , при відомих векторі ознак x та векторі справжніх значень y . λ відіграє роль параметра регуляризації, що запобігає перенавчанню моделі.

$$\min_w \sum_{i=1}^n (w^T x_i - y_i)^2 + \lambda \|w\|^2. \quad (1.4)$$

У поєднанні з LRR, КСР використовує циркулярні матриці, приклад зображено на рисунку 1.6, що зберігають усі можливі циклічні зсуви зразка зображення, забезпечуючи більшу стійкість до змін положення об'єкта.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}$$

Рисунок 1.6 – Приклад циркулярної матриці

Застосовуючи ядровий триюк, алгоритм обробляє нелінійні залежності. В цьому підході обчислення проводяться у більш вимірному просторі, де дані стають лінійно роздільними. Для цього використовується функція ядра, що має формулу (1.5), яка обчислює скалярний добуток у новому просторі без необхідності явного переходу.

$$k(x, x') = \langle \varphi(x), \varphi(x') \rangle_v. \quad (1.5)$$

Для досягнення високої швидкості обробки даних, вагові коефіцієнти обчислюються у частотному просторі, де множення елементів виконується швидше. Замість обробки всіх зразків у вихідному просторі, алгоритм обчислює вагові коефіцієнти у просторі Фур'є. Останнім для оцінки положення об'єкта у новому кадрі є обчислення суми вагових коефіцієнтів, помножених на значення ядра між поточним зразком та всіма попередніми, що були використані.

1.3.2 Формування мети та задач роботи

У цьому розділі буде розглянуто задачі даної системи.

1.3.2.1 Збереження особового складу

Задача збереження особового складу полягає у мінімізації ризиків для життя і здоров'я людей, що беруть участь у завданнях стеження та моніторингу об'єктів у небезпечних умовах. Система повинна враховувати потреби максимальної автоматизації, яка дозволяє зменшити пряму участь людей у процесах, що можуть бути небезпечними.

Завдяки використанню автономних мобільних роботів, оснащених сенсорами та камерами, що дозволяють оператору стежити за навколишнім середовищем і керувати загальними діями, система здатна мінімізувати кількість операцій к яких потрібні люди. У поєднанні із сучасними алгоритмами комп'ютерного зору система здатна самостійно виявляти, ідентифікувати та відстежувати об'єкти без постійного контролю оператора, що також знижує ризик для людей, оскільки роботи можуть виконувати більшість завдань незалежно, швидко реагуючи на зміну умов у навколишньому середовищі.

1.3.2.2 Оптимізація часу на пошуки

Задача оптимізації часу на пошуки полягає у підвищенні швидкості виявлення та ідентифікації цілей під час пошукових операцій. Зважаючи на те, що модуль стеження повинен бути автономним і не залежати від типу мобільного робота, це дозволяє використовувати різні мобільні роботи, в тому числі і дрони. Також, сам модуль стеження підтримує різні типи камер, що дозволяє більш тонко налаштовувати модуль до середовища роботи.

Знов–таки, Використання передових алгоритмів комп'ютерного зору та аналізу в режимі реального часу дозволяє роботам швидко адаптуватися до змін у навколишньому середовищі, виявляючи цілі та інші важливі об'єкти без затримок. А отже, це суттєво скорочує час, необхідний для реагування на критичні ситуації, що підвищує загальну ефективність операцій.

1.3.2.3 Автономне забезпечення безпеки об'єкта

Поставлена задача передбачає розміщення систем стеження на периметрі об'єкта, який потребує захисту, що дозволяє здійснювати повний контроль над оточенням. Система повинна передбачати її використання для постійного моніторингу і швидкого реагування на будь–які загрози чи порушення периметру, забезпечуючи цілодобову безпеку. Завдяки розміщенню таких модулів стеження на ключових точках навколо об'єкта система отримує всебічне уявлення про його межі, що дозволяє миттєво фіксувати наближення сторонніх об'єктів та оперативно інформувати операторів. Такий підхід забезпечує не тільки раннє виявлення потенційних загроз, але й автономне управління реагуванням, яке включає сигналізацію або активацію механізмів захисту.

Висновки до розділу

У розділі було розглянуто сучасні технології стеження за з акцентом на використання комп'ютерного зору. Було визначено, що розвиток таких систем базується на інтеграції штучного інтелекту та машинного навчання, що дозволяє досягти високої точності та надійності у відстеженні цілей у різних середовищах.

Розглянуто існуючі рішення, включаючи дрони DJI та FLIR Black Hornet, які є провідними прикладами в області мобільних роботів. Дрони DJI, оснащені камерами та сенсорами, використовують технологію ActiveTrack, що дозволяє їм автоматично виявляти та відстежувати об'єкти, коригуючи положення у режимі реального часу завдяки алгоритмам, таким як згорткові нейронні та трекарам на основі фільтрів. FLIR Black Hornet застосовує подібний підхід для забезпечення навігації і стеження в умовах реального часу, завдяки використанню технологій SLAM, згорткових нейронних мереж та алгоритмів уникнення перешкод.

Також у розділі було детально проаналізовано алгоритми стеження, включаючи генеративні та розрізнявальні методи. Зокрема, розглянуто алгоритм KCF, який забезпечує ефективне відстеження об'єктів за рахунок обробки зображень із низькими витратами ресурсів, використовуючи ядерний трюк та циркулярні матриці.

Таким чином, аналіз предметної області та існуючих рішень показав, що поєднання мобільних роботів з системами комп'ютерного зору має значний потенціал для застосування у таких сферах, як промисловість, агрикультура, медицина, військова розвідка та інші. Існуючі розробки підтверджують, що роботи, здатні ефективно відстежувати об'єкти в різноманітних умовах, можуть суттєво підвищити ефективність виконання поставлених завдань, знижуючи ризики для людини та збільшуючи продуктивність.

2 ФОРМУВАННЯ ВИМОГ ДО СИСТЕМИ

2.1 Формування функціональних вимог

Проаналізувавши аналоги систем та алгоритмів, можна сформувати загальний вигляд майбутньої системи. Загальний вигляд повинен бути сформований чітко і зрозуміло, як розробникам, так і користувачам системи.

Для виконання такої задачі існують функціональні вимоги системи. Написання функціональних вимог полягає у створення документу, що складається із функцій та поведінки, де функції – це те, що виконує система, а поведінка – це як система це робить[5].

Даний розділ буде містити сформовані функції та їх поведінку залежно до типу акторів, які працюють з системою, а саме: Адміністратор, Пілот мобільного роботу, та Мобільний робот. В таблиці 2.1 можна знайти детальний опис кожного з акторів.

Таблиця 2.1 – Детальний опис акторів

Назва	Опис
Адміністратор	Адміністратор системи має право керувати профілями інших користувачів, модифікаціями модуля стеження. Також він може переглядати усі можливі результати та логи використання системи.
Пілот мобільного роботу	Пілот мобільного роботу має доступ до систем керування роботом та камер, що допомагають йому орієнтуватися у просторі. Також він може: передивлятися результати використання системи своєї бригади та профілі побратимів, і завантажувати результати використання системи.
Мобільний робот	Мобільний робот представляє собою логічну одиницю, якою буде керувати користувач з роллю Пілот мобільного роботу. Мобільний робот повинен виконувати усі надіслані команди від пілоту, а також надавати інформацію з бортових приладів

На рисунку 2.1 можна побачити діаграму ієрархії ролей системи.

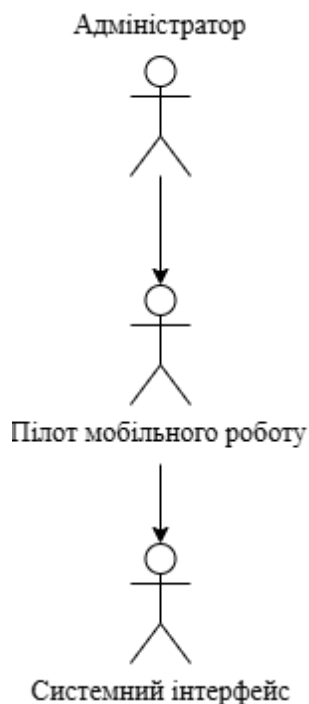


Рисунок 2.1 – Діаграма ієрархії ролей системи

Загальна діаграма прецедентів наведена в додатку К.

У наступних підрозділах будуть наведені та описані функціональні вимоги, що є спільними для кожної з ролей, а також є окремими для кожного з акторів.

2.1.1 Взаємодія системи з користувачами певних ролей

2.1.1.1 Автентифікація

Функціонал вимоги «Автентифікація» повинна бути доступна для всіх типів користувачів системи без обмежень. Даний функціонал має найважливішу задачу у системі, а саме перевірка та підтвердження того, ким є користувач на вході і який функціонал користувач може використовувати[6].

На рисунку 2.2 можна побачити діаграми прецедентів для функціональної вимоги «Автентифікація».

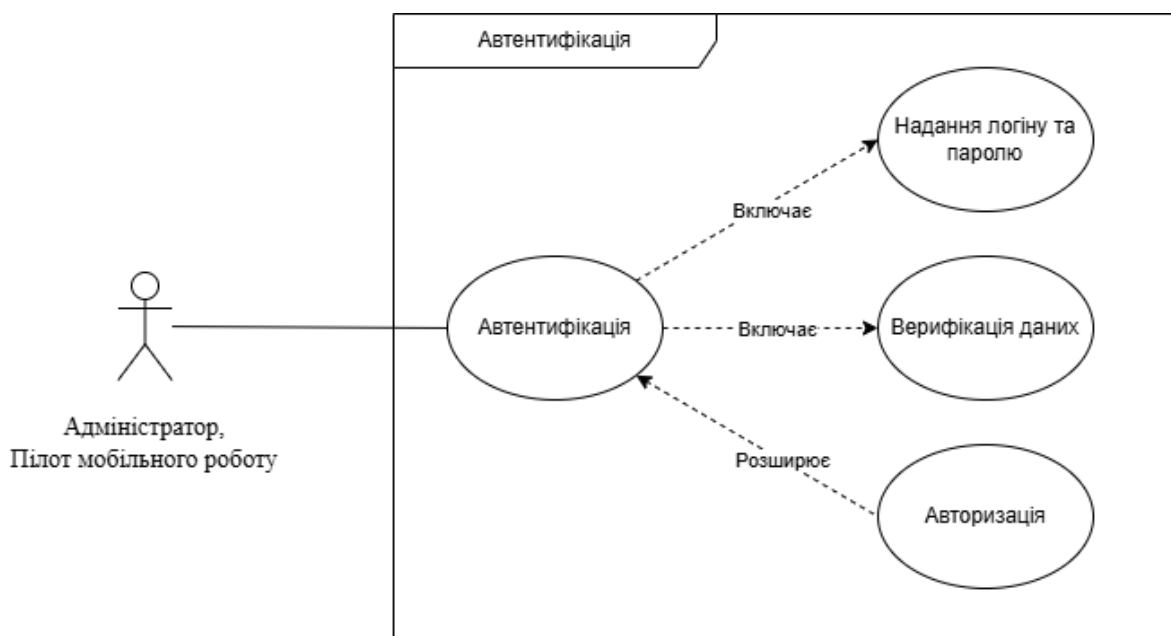


Рисунок 2.2 – Діаграма прецедентів до функціональної вимоги автентифікації

В таблиці 2.2 можна побачити детальний опис діаграми прецедентів до функціональної вимоги автентифікації

Таблиця 2.2 – Детальний опис діаграми прецедентів до функціональної вимоги автентифікації

Прецедент	Опис
Автентифікація	Даний прецедент позначає початковий стан функціоналу. На даному етапі Адміністратор та Пілот мобільного роботу заgonу отримують доступ до системи в залежності до їх ролі. Прецедент «Автентифікація» розширюється прецедентом «Авторизація» та включає такі прецеденти: «Надання логіну та паролю» і «Авторизація».
Надання логіну та паролю	Даний прецедент позначає реалізацію введення логіна та паролю користувачами у спеціальну форму системи.
Верифікація даних	Функціонал даного прецеденту полягає у порівнянні захешованого паролю, що ввів користувач із захешованим паролем, що зберігається у базі даних.

Прецедент	Опис
Авторизація	Даний прецедент позначає реалізацію механізму обмежень використання функціоналу системи залежно до ролі користувача.

2.1.1.2 Перегляд профілів членів загону

Функціонал перегляду профілів членів загону, аналогічно до попередньої функціональної вимоги, доступний усім користувачам системи. Проте, користувачі з ролями Пілот мобільного роботу можуть шукати профілі тільки членів власного загону. Для ролі Адміністратор загону обмежень немає.

В таблиці 2.3 можна побачити детальний опис діаграма прецедентів до функціональної вимоги перегляд профілів членів загону

Таблиця 2.3 – Детальний опис діаграма прецедентів до функціональної вимоги перегляд профілів членів загону

Прецедент	Опис
Перегляд профілів членів загону	Даний функціональна вимога описує можливість перегляду інформації про користувачів системи. Реалізації повинна враховувати роль користувача, що використовує механізм пошуку та перегляду. Прецедент деталізується трьома претендентами: «Пошук користувачів за певними критеріями», «Відображення знайдених профілів на сторінці пошуку», «Відображення інформації користувача на сторінці профілю».
Пошук користувачів за певними критеріями	Даний прецедент позначає реалізацію сторінки пошуку профілів за критеріями: позивний, номер загону та інші. Також, користувачі повинні мати можливість відвідувати та використовувати дану сторінку.

Прецедент	Опис
Відображення знайдених профілів на сторінці пошуку	Функціонал даного прецеденту полягає у відображенні короткої інформації про користувача та можливість отримати більше повну інформацію в залежності від ролі користувача, що здійснює пошук.
Відображення інформації користувача на сторінці профілю	Для реалізації прецеденту «Відображення інформації користувача на сторінці профілю» система повинна надати максимально можливу інформацію про користувача залежно до ролі користувача, що здійснює пошук.

Діаграма прецедентів функціоналу перегляду профілів членів загону зображена на рисунку 2.3.

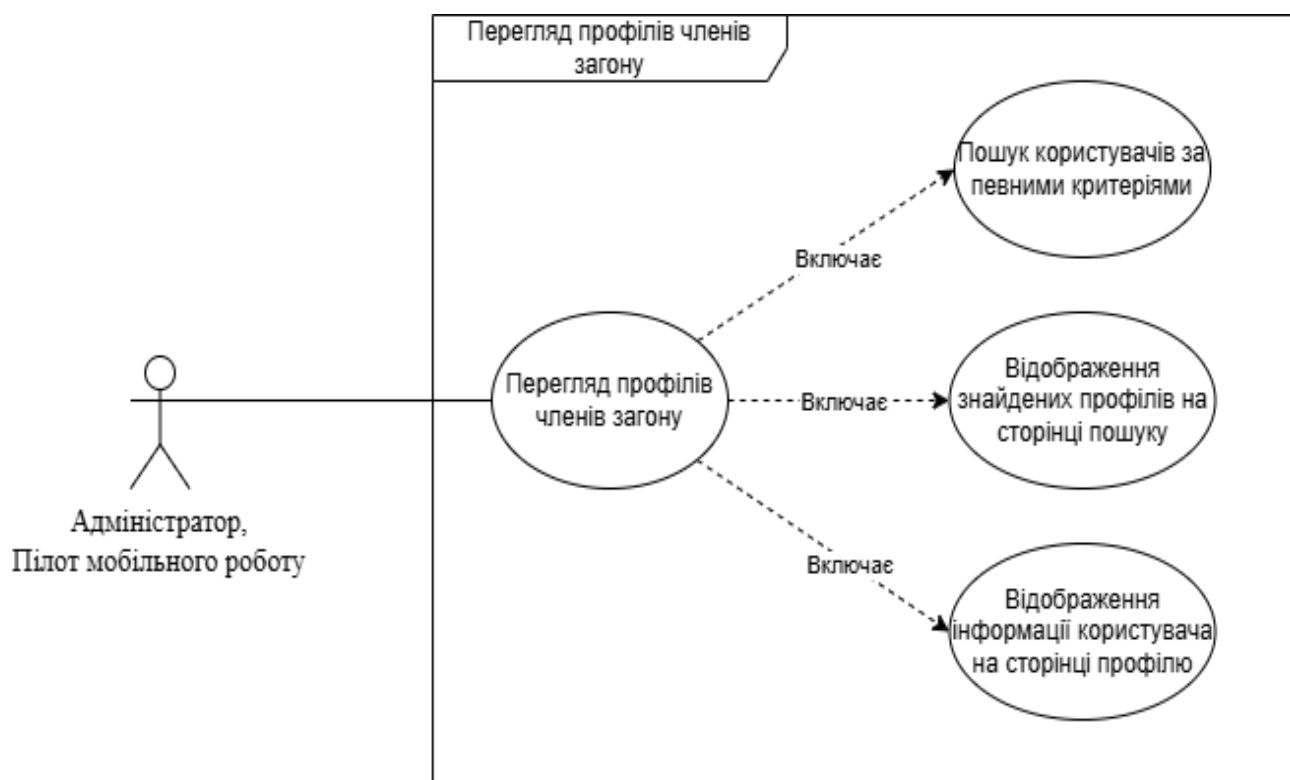


Рисунок 2.3 – Діаграма прецедентів до функціональної вимоги перегляд профілів членів загону

2.1.1.3 Перегляд доступних модифікацій модулю спостереження

Функція перегляду доступних модифікацій модулю спостереження доступна усім користувачам системи і полягає у відображенні налаштувань модулю стеження за цілями мобільного роботу серед яких є: вибір палітри кольорів цілі стеження та моделі штучного інтелекту для класифікації об'єктів.

Слід зауважити, правильна палітра кольорів може значно покращити здатність виявляти та відстежувати цілі, а модель може забезпечувати більш високу точність розпізнавання залежно до оточення мобільного роботу. Загалом, ця функція надає користувачам гнучкість у налаштуванні системи відповідно до їхніх конкретних потреб та умов експлуатації. Вона сприяє підвищенню продуктивності мобільного робота та дозволяє ефективніше виконувати поставлені завдання.

Діаграма прецедентів функціоналу перегляду доступних модифікацій модулю спостереження зображена на рисунку 2.4.

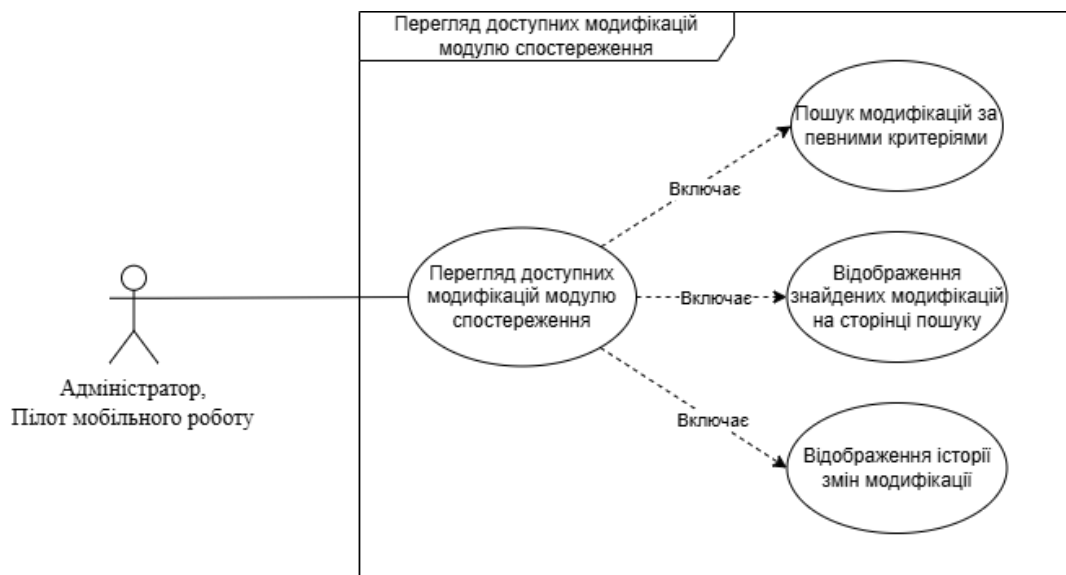


Рисунок 2.4 – Діаграма прецедентів до функціональної вимоги перегляд доступних модифікацій модулю спостереження

В таблиці 2.4 можна побачити детальний опис діаграма прецедентів до функціональної вимоги перегляд профілів членів загону.

Таблиця 2.4 – Детальний опис діаграма прецедентів до функціональної вимоги перегляд доступних модифікацій модулю спостереження.

Прецедент	Опис
Перегляд доступних модифікацій модулю спостереження	Дана функціональна вимога описує можливість перегляду доступних модифікацій модулю спостереження. Реалізації повинна враховувати роль користувача, що використовує механізм пошуку та перегляду. Прецедент деталізується трьома претендентами: «Пошук модифікацій за певними критеріями», «Відображення знайдених модифікацій на сторінці пошуку», «Відображення історії змін модифікації».
Пошук модифікацій за певними критеріями	Даний прецедент позначає реалізацію сторінки пошуку модифікацій за критеріями: дата створення, дата модифікації, назва модифікації. Також, користувачі повинні мати можливість відвідувати та використовувати дану сторінку.
Відображення знайдених модифікацій на сторінці пошуку	Функціонал даного прецеденту полягає у відображенні інформації про модифікацію в залежності від ролі користувача, що здійснює пошук.
Відображення історії змін модифікації	Для реалізації даного функціоналу система повинна відображати історію змін модифікації на окремій сторінці. Даний функціонал доступний тільки для Адміністратора.

2.1.2 Взаємодія системи з користувачами ролі Адміністратор

Адміністратор має найширші можливості керування системою. До цих можливостей можна віднести:

- керувати усіма профілями користувачів системи;
- керувати усіма модифікаціями модуля спостереження мобільного роботу;

– переглядати розширену інформацію про зміни у системі.

2.1.2.1 Керування профілями користувачів

Адміністратор має можливість створювати, редагувати та видаляти будь-які профілі користувачів системи, щоб підтримувати систему оновленою.

Діаграма прецедентів функціоналу керування профілями користувачів зображена на рисунку 2.5.

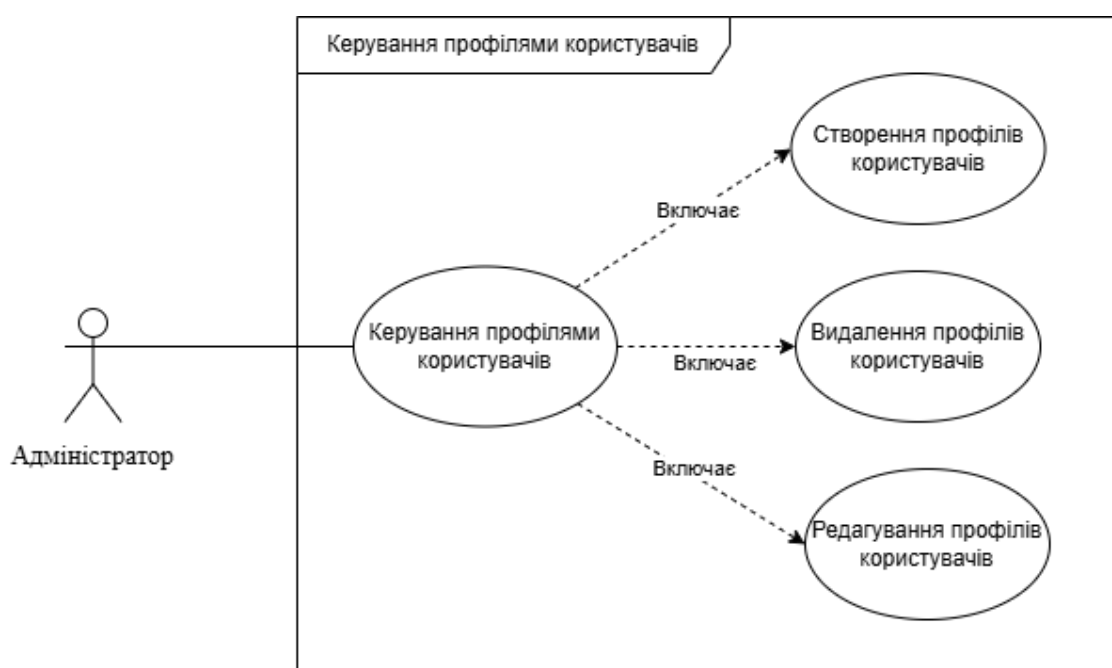


Рисунок 2.5 – Діаграма прецедентів до функціональної вимоги керування профілями користувачів

В таблиці 2.5 можна побачити детальний опис діаграми прецедентів до функціональної вимоги перегляд результатів операцій мобільного робота.

Таблиця 2.5 – Детальний опис діаграми прецедентів до функціональної вимоги перегляд результатів операцій мобільного робота.

Прецедент	Опис
Керування	Дана функціональна вимога описує можливість підтримувати

Прецедент	Опис
профілями користувачів	систему оновленою у секторі користувачів, що також покращує безпеку, бо тільки актуальні користувачі будуть мати доступ до системи. Прецедент деталізується такими прецедентами: «Створення профілів користувачів», «Видалення профілів користувачів» та «Редагування профілів користувачів»
Створення профілів користувачів	Даний прецедент описує наявність форми створення нових користувачів, в якій потрібно буде вказувати такі дані, як: позивний, спеціалізація за замовчуванням, номер бригади та група крові. Система повинна мати валідацію для всіх полів форми.
Видалення профілів користувачів	Функціонал даного прецеденту полягає у видаленні профілів користувачів, які не повинні мати доступ до системи. Система повинна наслідувати принцип м'якого видалення для збереження статистичних даних. Під час видалення система повинна валідувати, що видалення ініційовано не випадково.
Редагування профілів користувачів	Функціонал даного прецеденту полягає у редагуванні профілів користувачів у випадку некоректних даних, зміні даних чи інших варіантах. Система повинна наслідувати принцип збереження історії зміни інформації про користувача для збереження статистичних даних. Під час редагування система повинна валідувати введені дані.

2.1.2.2 Керування модифікаціями модуля спостереження мобільного роботу

Тільки адміністратор повинен володіти механізмом керування модифікаціями модуля спостереження мобільного роботу, щоб підтримувати систему оновленою та

протидіяти діям, що направлені на погіршення роботи системи. Це дозволяє забезпечити безпеку та стабільність функціонування робота в різних умовах. Адміністратор може контролювати доступ до налаштувань і оновлень, запобігаючи несанкціонованим змінам.

Діаграма прецедентів функціоналу керування модифікаціями модуля спостереження мобільного роботу зображена на рисунку 2.6.

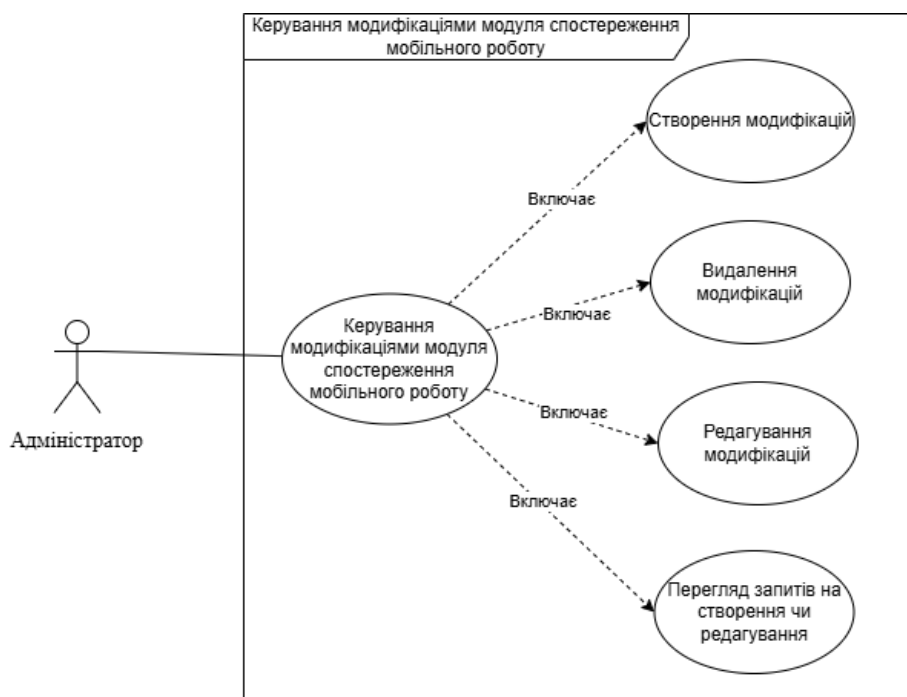


Рисунок 2.6 – Керування модифікаціями модуля спостереження мобільного роботу зображена

В таблиці 2.6 можна побачити детальний опис діаграми прецедентів до функціональної вимоги керування модифікаціями модуля спостереження мобільного роботу.

Таблиця 2.6 – Детальний опис діаграми прецедентів до функціональної вимоги керування модифікаціями модуля спостереження мобільного роботу зображена.

Прецедент	Опис
Керування модифікаціями	Дана функціональна вимога описує можливість підтримувати модуль спостереження оновленим, що

Прецедент	Опис
модуля спостереження мобільного роботу	покращує ефективність стеження за цілями. Прецедент деталізується такими прецедентами: «Створення профілів користувачів», «Видалення профілів користувачів» та «Редагування профілів користувачів»
Створення модифікацій	Даний прецедент описує наявність форми створення нових модифікацій, в якій потрібно буде вказувати такі дані, як: колір цілі, тип цілі, назву модифікації та модель класифікації цілей. Система повинна мати валідацію для всіх полів форми.
Редагування модифікацій	Функціонал даного прецеденту полягає у редагуванні модифікацій некоректних даних, зміні даних чи інших варіантах. Система повинна наслідувати принцип збереження історії зміни інформації про модифікації для збереження статистичних даних. Під час редагування система повинна валідувати введені дані.
Видалення модифікацій	Даний прецедент описує логіку видалення модифікацій. Система повинна наслідувати принцип м'якого видалення для збереження статистичних даних. Під час видалення система повинна валідувати, що видалення ініційовано не випадково.

2.1.2.3 Переглядати розширену інформацію про зміни у системі

Для Адміністратор важливо розуміти, що відбувається із системою в будь-який час. Тож, система повинна мати механізм перегляд розширеної інформації про зміни у системі. Система повинна мати механізм завантаження цієї інформації, що доступний тільки Адміністратору.

Діаграма прецедентів функціональної вимоги перегляду розширеної інформації про зміни у системі зображена на рисунку 2.7. А в таблиці 2.7 можна побачити детальний опис діаграма прецедентів до функціональної вимоги перегляду розширеної інформації про зміни у системі.



Рисунок 2.7 – Діаграма прецедентів функціональної вимоги перегляду та завантаження розширених результатів роботи системи

Таблиця 2.7 – Детальний опис діаграми прецедентів до функціональної вимоги перегляду та завантаження розширених результатів роботи системи.

Прецедент	Опис
Перегляд розширеної інформації про зміни у системі	Дана функціональна вимога описує можливість слідкувати за змінами у системі. Прецедент деталізується такими прецедентами: «Завантаження інформації» та «Перегляд інформації на відповідних сторінках»
Завантаження інформації	Система повинна надавати Адміністратору функціонал для завантаження інформації про використання системи у вигляді логів для подальшого їх аналізу. Це забезпечить підвищену стабільність системи.

Прецедент	Опис
Перегляд інформації на відповідних сторінках	У ролі Адміністратора, система повинна відображати інформацію про останні зміни на сторінках де ці дані критично важливі

2.1.3 Взаємодія системи з користувачами ролі Пілот мобільного боту

Пілот мобільного дрону займає важливе місце у системі, бо його обов'язки містять контролювання мобільного дрону, аналіз результатів роботи дрону, забезпечення безпечної та надійної роботи дрону. Також, пілот мобільного дрону повинен стежити за актуальністю даних модуля стеження.

В таблиці 2.8 можна побачити детальний опис діаграми прецедентів взаємодії системи з користувачами ролі Пілот мобільного боту.

Таблиця 2.8 – Детальний опис діаграми прецедентів взаємодії системи з користувачами ролі Пілот мобільного боту.

Прецедент	Опис
Управління мобільним роботом	Даний прецедент описує логіку взаємодії Пілоту мобільного робота з Роботом через систему. Система повинна надавати зручний інтерфейс в якому Пілот мобільного робота зможе безпроблемно контролювати Робот. Доповнюється прецедентом «Перемикання режимів пілотування»
Перемикання режимів пілотування	Даний прецедент описує логіку інтерфейсу систему, який буде у зручній формі дозволяти перемикати режими пілотування з статичного на динамічний. Також система повинна графічно відображати зміну режимів
Аналіз результатів	Даний прецедент описує логіку інтерфейсу систему, який буде

Прецедент	Опис
роботи дрону	у зручній формі дозволяти аналізувати дані, що Робот надсилає в систему. Аналіз даних повинен відбуватися за рахунок використання штучного інтелекту. Система повинна візуалізувати результати аналізу.
Забезпечення надійної роботи дрону	Даний прецедент описує логіку інтерфейсу систему, який буде у зручній формі сповіщати Пілота мобільного дрону про можливі зіткнення з перешкодами за допомогою інформації з Роботу

Діаграма прецедентів взаємодії системи з користувачами ролі Пілот мобільного боту на рисунку 2.8.

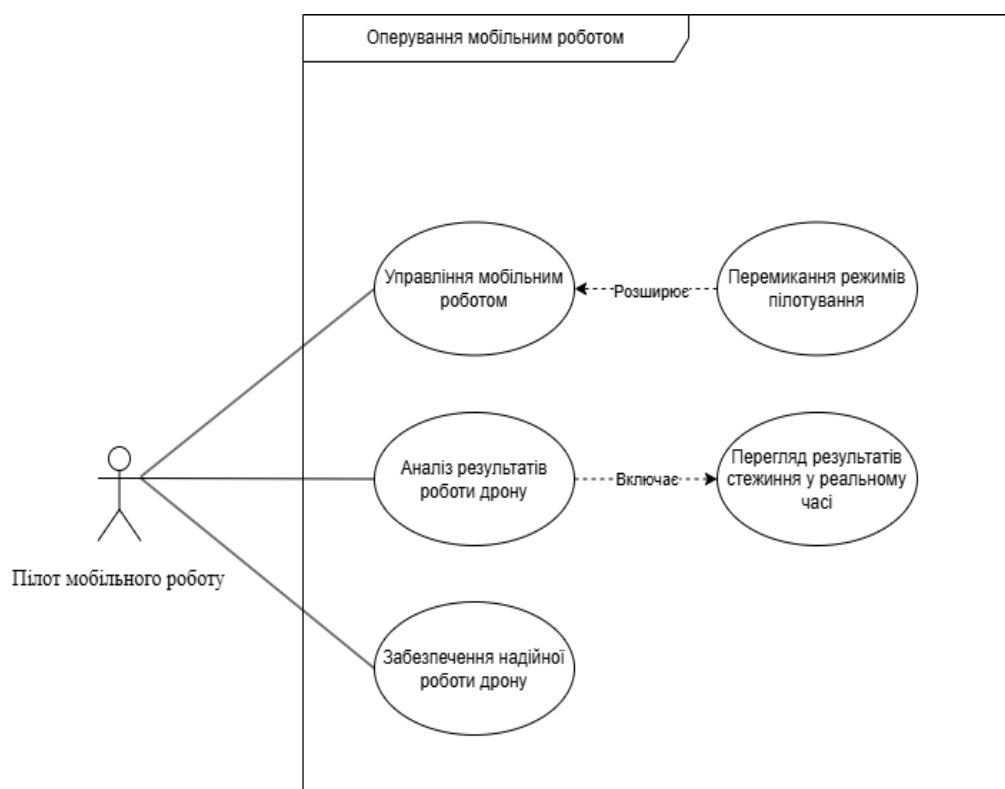


Рисунок 2.8 – Діаграма прецедентів взаємодії системи з користувачами ролі Пілот мобільного боту

2.1.4 Взаємодія інтерфейсу системи з системою Мобільного робота

Функціональні вимоги взаємодії інтерфейсу системи з системою Мобільного робота є вкрай важливим, бо вони визначають найважливішу ланку проєкту. Без стабільного та прописаного зв'язку між цими двома компонентами, не можна гарантувати, що система буде працювати стабільно, не можна буде написати тест-кейси для покриття edge-кейсів системи[7].

Тож у наступних підрозділах буде розглянуто діаграми прецедентів взаємодії інтерфейсу системи з системою Мобільного робота.

2.1.4.1 Початок симуляції

Початок симуляції є одним із найважливіших етапів роботи системи. Під час цього етапу відбуваються попередні налаштування Робота та інтерфейсу системи.

Далі було розглянуто діаграму прецедентів взаємодії інтерфейсу системи з системою Мобільного робота, що зображена на рисунку 2.9, на етапі початку симуляції.

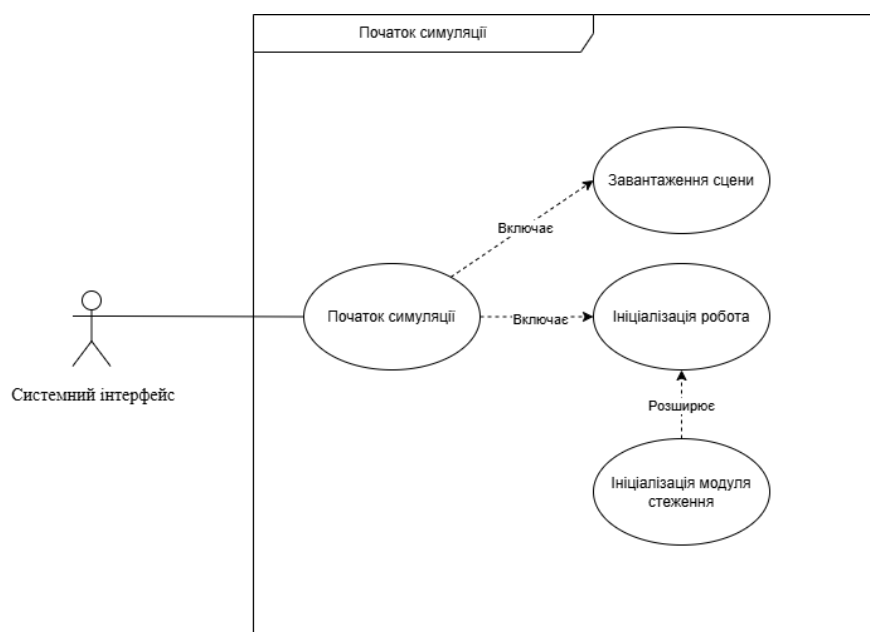


Рисунок 2.9 – Діаграма прецедентів взаємодії інтерфейсу системи з системою Мобільного робота на етапі початку симуляції

В таблиці 2.9 можна побачити детальний опис діаграми прецедентів взаємодії інтерфейсу системи з системою Мобільного робота на етапі початку симуляції.

Таблиця 2.9 – Детальний опис діаграми прецедентів взаємодії інтерфейсу системи з системою Мобільного робота на етапі початку симуляції.

Прецедент	Опис
Початок симуляції	Дана функціональна вимога описує можливість інтерфейсу системи починати симуляції системи стеження мобільного робота. Прецедент деталізується такими прецедентами: «Завантаження сцени», «Видалення профілів користувачів» та «Ініціалізація модуля стеження»
Завантаження сцени	Даний прецедент описує поведінку інтерфейсу системи після його запуску. Інтерфейсу системи повинен завантажити сцену та перевірити, щоб усі змінні інтерфейсу ініціалізовані коректно. Після перевірки, інтерфейс системи повинен створити новий файл логів.
Ініціалізація робота	Після успішної перевірки змінних, інтерфейси системи повинен надіслати дані модифікацій робота для їх подальшого використання системами Робота. Усі кроки повинні логуватися у створений файл. Прецедент деталізується прецедентом «Ініціалізація модуля стеження»
Ініціалізація модуля стеження	Останнім етапом початку симуляції є надсилання модифікації модуля стеження мобільного роботу. Дані модифікації використовуються системами модуля для коректного знаходження та стеження на ціллю. Усі кроки повинні логуватися у створений файл інтерфейсу системи

2.1.4.2 Симуляція мобільного робота з отриманням результатів

Закінчивши етап ініціалізації симуляції, починається сама симуляція мобільного робота на сцені. Даний етап містить такий ключовий аспект, як взаємодія робота з інтерфейсом системи у реальному часі. Під час симуляції робот виконує заплановані дії та реагує на динамічні зміни в середовищі. Це дозволяє користувачам оцінити ефективність алгоритмів управління та ввести необхідні корективи. Таким чином, симуляція забезпечує глибоке розуміння поведінки робота в різних ситуаціях.

Розглянуто діаграму, прецедентів взаємодії мобільного робота з інтерфейсом системи у реальному часі, що зображена на рисунку 2.10.

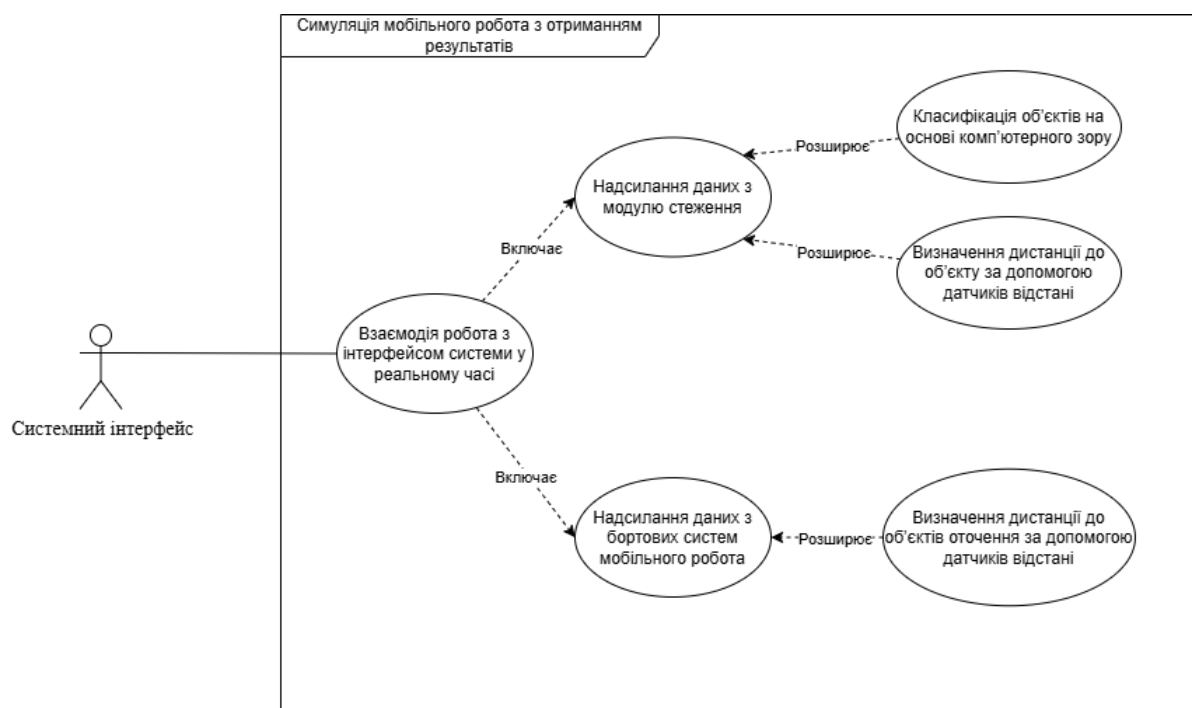


Рисунок 2.10 – Діаграма прецедентів взаємодії мобільного робота з інтерфейсом системи у реальному часі

В таблиці 2.10 можна побачити детальний опис діаграми прецедентів взаємодії мобільного робота з інтерфейсом системи у реальному часі.

Таблиця 2.10 – Детальний опис діаграми прецедентів взаємодії мобільного робота з інтерфейсом системи у реальному часі.

Прецедент	Опис
Взаємодія мобільного робота з інтерфейсом системи у реальному часі	Дана функціональна вимога описує можливість інтерфейсу системи надсилати та отримувати дані з модулю стеження мобільного робота та бортових датчиків Робота під час симуляції. Прецедент деталізується такими прецедентами: «Надсилання даних з модулю стеження», «Надсилання даних з бортових систем мобільного робота»
Надсилання даних з модулю стеження	Даний прецедент описує поведінку обміну даними між інтерфейсом системи та модулем стеження мобільного робота. Модуль стеження мобільного робота повинен надсилати три типи відео даних, а саме: необроблене відео для подальшого аналізу алгоритмами штучного інтелекту, відео типу HSV, та відео з накладанням Гауссового згладжування. Прецедент деталізується такими прецедентами: «Класифікація об'єктів на основі комп'ютерного зору», «Визначення дистанції до об'єкту за допомогою датчиків відстані»
Відстеження об'єктів на основі комп'ютерного зору	Після успішного отримання характеристик цілі, модуль стеження використовує алгоритми комп'ютерного зору для стеження за певною ціллю. Даний функціонал покращує сприймання інформації Пілотом мобільного роботу.
Визначення дистанції до об'єкту за допомогою датчиків відстані	Під час надсилання відео даних з камер модулю стеження, мобільний робот повинен надавати метричні дані відстанні до об'єкту стеження у реальному часі. Даний функціонал покращує сприймання інформації Пілотом мобільного роботу.

Прецедент	Опис
Надсилення даних з бортових систем мобільного робота	Даний прецедент описує поведінку обміну даними між інтерфейсом системи та бортових систем мобільного робота. Мобільний робот повинен мати налаштовані бортові сенсори відстані для надсилення цих даних на інтерфейс системи Пілоту мобільного робота. Даний функціонал допомагає Пілоту використовувати робота більш безпечно. Прецедент деталізується прецедентом: «Визначення дистанції до об'єктів оточення за допомогою датчиків відстані»
Визначення дистанції до об'єктів оточення за допомогою датчиків відстані	Під час динамічного режиму керування, мобільний робот повинен надавати метричні дані відстані до об'єкту оточення у реальному часі. Даний функціонал покращує сприймання інформації Пілотом мобільного роботу.

2.1.4.3 Взаємодія робота з інтерфейсом системи під час завершення симуляції

Аналогічно до початку симуляції, завершення цього процесу є критично важливим для забезпечення належного функціонування системи. Цей етап має ключове значення, оскільки під час нього виконується безпечне завершення роботи мобільного робота, що запобігає можливим збоям чи пошкодженням обладнання.

Діаграма прецедентів взаємодії робота з інтерфейсом системи під час завершення симуляції, що зображена на рисунку 2.11, ілюструє основні етапи цього процесу.

Завершення симуляції є завершальним і важливим етапом роботи системи, який гарантує її стабільність, точність та збереження важливих даних.

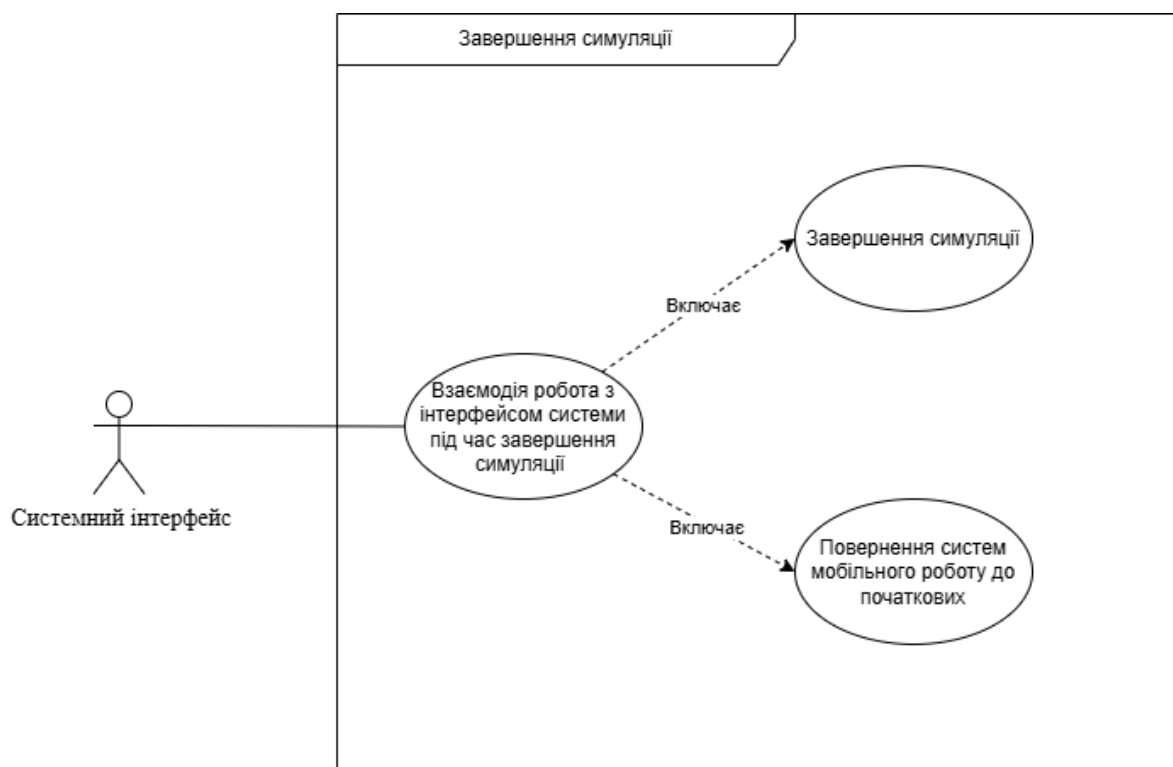


Рисунок 2.11 – Діаграма прецедентів взаємодії робота з інтерфейсом системи під час завершення симуляції

В таблиці 2.11 можна побачити детальний опис діаграми прецедентів взаємодії робота з інтерфейсом системи під час завершення симуляції.

Таблиця 2.11 – Детальний опис діаграми прецедентів взаємодії робота з інтерфейсом системи під час завершення симуляції.

Прецедент	Опис
Взаємодія робота з інтерфейсом системи під час завершення симуляції	Дана функціональна вимога описує взаємодії робота з інтерфейсом системи під час завершення симуляції. Інтерфейс системи повинний чітко і зрозуміло повідомляти Пілота мобільного дрона про завершення симуляції. Прецедент деталізується такими прецедентами: «Завершення симуляції», «Повернення систем мобільного роботу до початкових»
Завершення симуляції	Під час завершення симуляції інтерфейс системи повинен повідомити Пілота про те, що симуляція завершилася. Також

Прецедент	Опис
	інтерфейс системи повинен залогувати кінець симуляції і зберегти файл.
Повернення систем мобільного роботу до початкових значень	Після успішного завершення симуляції Робот повинен перевести свої системи у початковий стан і закінчити надсилання відео та метричних даних на інтерфейс системи.

2.2 Формування нефункціональних вимог

Нефункціональні вимогами називають набір стандартів, які використовуються для оцінки конкретної роботи системи[8].

У цьому розділі будуть сформовані наступні нефункціональні вимоги до інтерфейси системи та мобільного робота:

а) продуктивність;

1) час обробки зображень і відеопотоків повинен не перевищувати 50 мс на кожен кадр при роздільній здатності;

2) алгоритми обробки зображень повинні запускатися з частотою не менше 20 мс на кожен кадр;

3) реакція на команди оператора повинна відбуватися не довше, ніж за 100 мс після подачі команди;

б) надійність;

1) система повинна мати можливість працювати безперервно не менше 12 годин без перезавантаження, забезпечуючи стабільність в умовах довготривалого стеження;

в) безпека;

1) для всіх з'єднань, що передають дані між роботом і сервером, використовується шифрування на рівні AES–256 для захисту від несанкціонованого доступу;

2) логування дій адміністратора і користувачів із привілеями в захищений журнал для моніторингу підозрілої активності;

г) зручність використання;

1) оператор повинен мати змогу виконати основні дії у межах трьох кліків або менше;

2) кожне повідомлення про помилки або збої повинно містити коротку інструкцію для оператора щодо подальших дій;

г) сумісність;

1) Система повинна підтримувати операційні системи Windows 10/11 та Ubuntu 20.04+;

2) Модуль стеження повинен бути здатним інтегруватися з різними типами камер;

е) обслуговуваність;

1) всі дії та помилки повинні логуватися з деталізацією: дата, час, користувач, подія.

Висновки до розділу

У цьому розділі було визначено ключові функціональні та нефункціональні вимоги для системи стеження мобільного робота. Значна увага приділяється ролям користувачів системи: Адміністратор, Пілот мобільного робота та Мобільний робот. Кожен із цих акторів має специфічні функціональні обов'язки та рівень доступу, що обумовлює розподіл завдань і прав у системі.

Функціональні вимоги охоплюють такі аспекти, як автентифікація користувачів, можливість перегляду профілів, модифікацій модулю спостереження та результатів операцій. Наприклад, всі користувачі повинні проходити автентифікацію для отримання доступу до функціоналу системи, що гарантує захищеність від

несанкціонованого доступу. Адміністратори мають найширші повноваження, включаючи можливість створювати, редагувати й видаляти профілі користувачів, а також керувати модифікаціями модуля спостереження, забезпечуючи актуальність і стабільність його роботи.

Щодо нефункціональних вимог, основний акцент зроблено на продуктивності системи, зокрема на швидкості обробки зображень та відео. Так, обробка кадрів для забезпечення безперервного відстеження рухомих об'єктів має відбуватися за 50 мс на кожен кадр. Важливим є також забезпечення миттєвої реакції системи на команди оператора – не більше 100 мс. Крім того, система має відповідати високим стандартам надійності та безпеки, що включає стабільність з'єднання між інтерфейсом системи і мобільним роботом та захист даних.

Сформовані вимоги демонструють чітке розуміння функціональності системи та її обмежень, необхідних для забезпечення надійного та стабільного стеження за цілями мобільного робота за допомогою технологій комп'ютерного зору.

3 РОЗРОБКА СИСТЕМИ

3.1 Вибір та обґрунтування елементів та технологій

3.1.1 Вибір алгоритму комп'ютерного зору для стеження за об'єктом

Для розробки системи, цієї магістерської дисертації, було обрано алгоритм стеження, зображений в додатку Б, що використовує колір об'єктів. Порівняно з алгоритмом КСФ, точність даного алгоритму достатньо висока і для обмеженої продуктивності системи симуляції він не потребує складних обрахунків, що гарантує стабільність під час симуляції.

Отже, алгоритм стеження за об'єктами, що базується на кольорі об'єкта можна поділити на 6 головних етапів, що зображені на рисунку 3.1.



Рисунок 3.1 – Головні етапи алгоритму стеження за об'єктами, що базується на кольорі об'єкта

Етапи «Зчитування кадру» та «Результат» не являються головними етапами, тож не будуть описані.

Головною задачею етапу попередньої обробки зображення є застосування Гауссового згладжування з ядром певного розміру на вхідний кадр. Гауссове

згладжування – це накладання на обране зображення фільтру Гаусса, що формується формулою Гаусса (3.1). Даний фільтр є одним із найпопулярніших у сфері обробки зображень, бо він має форму функції Гаусового дзвінка і може бути легко збережений у 2D–матриці[9].

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3.1)$$

Результатом цього етапу є кадр відео в якому кожен піксель став середнім показником його сусідів, що в результаті прибрато шум і підготувало кадр до етапу «Створення та обробка кольорового порогу».

Задачею наступного етапу є створення чорно–білої маски, де білий колір буде відповідати кольору об'єкта стеження, а чорний – колір фону. Як було описано авторами Jaya P та Geethu Balakrishnan у їх роботі «Contour Based Object Tracking», кожен колір можна описати двома значеннями: нижній поріг та верхній поріг.[10] Далі алгоритм буде аналізувати кадр для пошуку усіх кольорів, що знаходяться у визначеній межі і змінювати дані пікселів на білий, якщо колір у межах, і чорний, якщо ні. Цю логіку можна описати функцією (3.2).

$$M(x, y) = \begin{cases} 1, & \text{якщо } L \leq H(x, y), S(x, y), V(x, y) \leq U, \\ 0, & \text{інакше} \end{cases} \quad (3.2)$$

Наступним етапом є «Морфологічні операції». Як було описано у роботі M. Michalko, J. Onuška, A. Lavrín, D. Cymbalák, O. Kainz «Tracking the object features in video based on OpenCV»[11]. Такі операції допомагають згладити краї об'єкта та закрити невеликі прогалини, які інакше могли б призвести до фрагментованих або неточних відображень, що стабілізують процес відстеження, дозволяючи алгоритму постійно фокусуватися на об'єкті стеження.

Наступним етапом є підготовка маски до візуалізації. На цьому етапі використовується функція `cvtColor` бібліотеки `openCV`. Цей метод застосовується для перетворення зображення з одного колірного простору в інший і налічує понад 150 кольорових просторів[12]. Функціонально даний етап використовується для тестування коректності роботи алгоритму. Оскільки оригінальна маска знаходиться в сірому кольоровому просторі, потрібно перетворити цю маску з сірого простору у RGB, щоб користувач мав можливість оцінити коректність роботи алгоритму, побачивши результат відділення цільового кольору від інших кольорів.

Зі свого боку, оригінальна матриця кадру використовується як вхідні дані до фінального етапу алгоритму, а саме «Виявлення контурів». Мета цього етапу є визначення точного розташування виявленого об'єкта в межах бінарної маски. Для досягнення мети використовується метод `findContours` бібліотеки `openCV`. Даний метод має достатньо широкий функціонал, що допомагає ігнорувати внутрішні межі, що корисно для виділення основного об'єкта[13]. Після виявлення контурів код вибирає найбільший контур за площею, припускаючи, що цей контур відповідає цільовому об'єкту. Визначивши найбільший контур, алгоритм розраховує його обмежувальний прямокутник, далі координати обмежувального прямокутника нормалізуються до діапазону від -1 до $+1$ та передаються у алгоритм керування модулю відстеження.

3.1.2 Вибір середовища для симуляції системи

Як середовище для симуляції систем робота було обрано `CoppeliaSim` через його багатфункціональність та здатність задовольнити специфічні вимоги системи.

Потрібно зазначити, що інтегроване середовище розробки з `CoppeliaSim` використовує такі мови програмування, як: Python та Lua. Бібліотеки цих мов, та простота їх використання надають можливість реалізувати детальну логіку управління критичними частинами симуляції безпосередньо всередині середовища, забезпечуючи високу точність та надійність результатів. Крім того, `CoppeliaSim`

пропонує обширну бібліотеку моделей та функцій, що спрощує процес налаштування та підвищує реалістичність симуляційних сценаріїв [14].

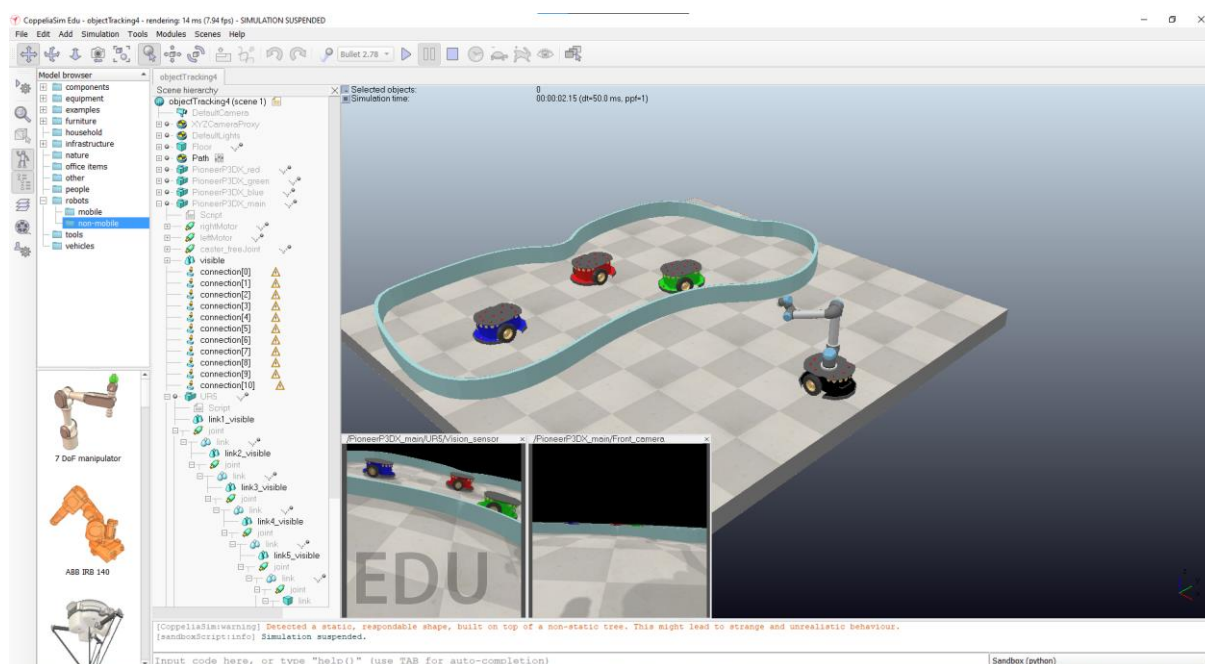


Рисунок 3.2 – Сцена симуляції CoppeliaSim

Вибір CoppeliaSim також обґрунтований можливостями розроблення зовнішнього інтерфейсу для керування симуляцією. У наступних розділах буде більш детально розписано процес створення такого інтерфейсу, проте, саме наявність таких механізмів комунікації сприяє більшій гнучкій інтеграції симуляції з іншими програмними компонентами або для реалізації складних алгоритмів управління, які можуть бути складними для впровадження лише всередині середовища симуляції [15].

Отже, маючи такий широкий функціонал самого середовища CoppeliaSim є доцільно обрати його для розробки системи стеження за повітряними та наземними цілями мобільного робота на основі комп'ютерного зору.

3.1.4 Вибір технологій та розробка інтерфейсу симуляції

Для розробки інтерфейсу управління симуляцією, було використано Python як основну мову програмування завдяки її високій популярності та широкій підтримці бібліотек для роботи з штучним інтелектом та комп'ютерним зором. Однією з основних бібліотек, яка використовувалася у проекті, є OpenCV, широко відома платформа для обробки зображень і відео. OpenCV містить численні функції для обробки зображень, такі як фільтрація, сегментація, розпізнавання об'єктів, а також трансформації, які дозволяють аналізувати та маніпулювати візуальними даними у реальному часі. Завдяки її потужним алгоритмам та зручному інтерфейсу, OpenCV стала ключовим компонентом для обробки даних із сенсорів та візуальної ідентифікації об'єктів у симуляції [16].

Для задач штучного інтелекту та детекції об'єктів у реальному часі було інтегровано модель YOLO, яка є однією з найбільш популярних та швидкодіючих моделей для розпізнавання об'єктів. Більш детально про вибір саме моделі YOLO буде йтися у наступних розділах. Проте можна сказати, що архітектура YOLO дозволяє застосовувати модель у задачах реального часу, що є ідеально підходить для системи.[17].

Ще однією важливою бібліотекою в цьому проекті є NumPy, яка забезпечує швидке обчислення числових операцій, таких як матричні перетворення, що необхідні для обробки вхідних даних моделі, масштабування та нормалізації зображень. NumPy дозволяє оптимізувати обчислення за рахунок векторизованих операцій, зменшуючи затримку при підготовці даних для YOLO та OpenCV. Завдяки таким характеристикам дана бібліотека є незамінним компонентом система, що націлено на швидку обробку та передачу даних у моделі машинного навчання [18].

Інтеграція з CoppeliaSim відбувалась за допомогою модуля RemoteAPIClient. RemoteAPIClient забезпечує зв'язок між Python та CoppeliaSim за допомогою протоколу ZeroMQ, який використовує серверно-клієнтську архітектуру, діаграма якої зображена на рисунку 3.3.

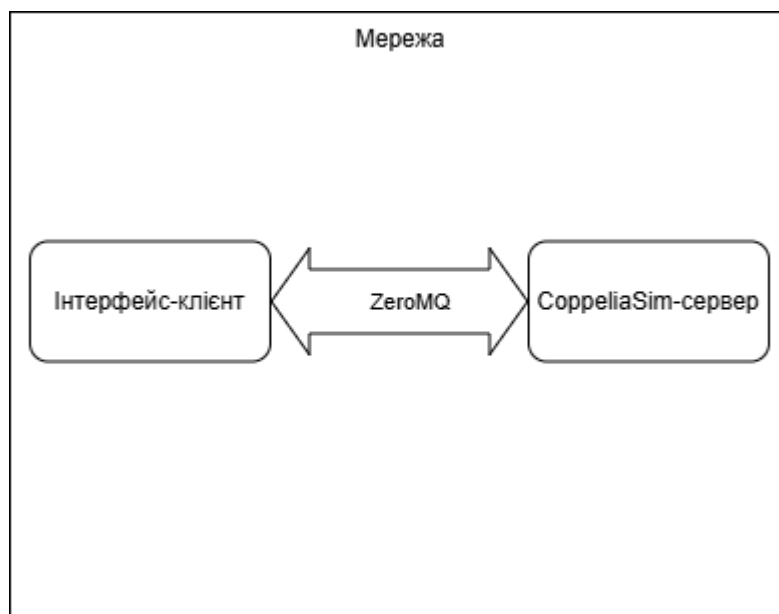


Рисунок 3.3 – Серверно-клієнтська архітектура

Python-скрипт працює як клієнт, що надсилає команди до сервера CoppeliaSim, і отримує дані від серверної сторони у симуляції. Це дозволяє взаємодіяти з об'єктами у CoppeliaSim, змінювати їх положення, орієнтацію, зчитувати параметри сенсорів, а також запускати й зупиняти симуляцію відповідно до вимог скрипта [19].

Для створення графічної частини системи було використано бібліотеку PyQt6, яка забезпечує багатофункціональний та зручний інтерфейс для розробки настільних додатків. PyQt6 є обгорткою для фреймворку Qt, відомого своєю високою продуктивністю та широкими можливостями для створення графічних інтерфейсів користувача[20].

Було створено вікна для відображення даних з камер симуляції, кнопки для запуску та зупинки, а також інструменти для налаштування параметрів середовища. Варто зазначити, що PyQt6 надає можливість роботи з сигналами та слотами, які забезпечують асинхронну обробку подій у програмі. Це дозволило реалізувати обмін даними, в режимі реального часу, між інтерфейсом користувача та симуляцією.[21]. У наступних розділах буде наведено фотографії клієнтського застосунка та розписано його можливості.

3.1.5 Вибір, проєктування та інтеграція сховища даних

Бази даних поділяються на кілька основних типів, залежно від їх структури та методу зберігання й обробки даних. Реляційні бази даних (RDBMS), такі як MySQL, PostgreSQL та SQLite, використовують таблиці для зберігання даних і працюють з ними на основі SQL–запитів.[22] Нереляційні або NoSQL бази, зокрема MongoDB, Redis та Cassandra, зберігають дані у форматах, що підходять для масштабованих та великих обсягів нереляційних даних.[23] Існують також об'єктно–орієнтовані та графові бази даних які застосовуються в специфічних випадках, зокрема для складних зв'язків між об'єктами.[24]

SQLite є оптимальним вибором для розроблюваної системи завдяки своїй легкості та відсутності потреби у серверній інфраструктурі, що дозволяє швидко налаштувати зберігання даних для локальних додатків або симуляційних середовищ. SQLite зберігає всю базу в одному файлі, що є ефективним для невеликих проєктів і розробки, коли обсяги даних не надто великі, а також для прототипування, де не потрібна висока пропускна здатність чи масштабування [25].

Для інтеграції SQLite з Python використовується SQLAlchemy – ORM–бібліотека, яка спрощує роботу з базами даних через об'єктно–реляційне відображення, дозволяючи зручно працювати з даними на рівні Python–коду.

У разі запуску цього стартап–проєкту на ринок і зростання обсягів даних або вимог до надійності й продуктивності, доцільно перейти до більш сучасних хмарних рішень, таких як Amazon RDS, Google Cloud SQL або Microsoft Azure SQL Database. Ці сервіси пропонують високий рівень масштабованості, резервування даних, а також інтеграцію з іншими хмарними інструментами, що сприяє стабільній роботі на високих навантаженнях.[26]

3.1.6 Вибір нейронної мережі для виявлення та класифікації об'єктів

Для вибору нейронної мережі для виявлення та класифікації об'єктів було проаналізовано кілька популярних моделей, серед яких YOLO11, Faster R–CNN та

SSD[27]. Кожна з них має свої особливості, переваги та недоліки, що детально розглянуто у таблиці 3.1.

Таблиця 3.1 – Порівняння моделей нейронної мережі

Характеристика	YOLO11	SSD	Faster R–CNN
Архітектура	Одноетапна, використовує CNN для одночасного виявлення та класифікації	Багатоступенева, спочатку генеруються регіони, потім класифікуються	Одноетапна, використовує багаторівневі предиктори
Швидкість	Висока, підходить для роботи в реальному часі	Висока, але поступається YOLO11	Низька, більше підходить для задач із високими вимогами до точності
Точність	Висока, особливо для великих об'єктів	Середня, можлива хибна класифікація	Дуже висока, але вимагає більше ресурсів
Використання ресурсів	Низьке, підходить для вбудованих систем	Середнє, оптимізоване для мобільних пристроїв	Високе, потребує потужного апаратного забезпечення
Підходить для реального часу	Так	Так	Ні

YOLO11 була обрана для цього проєкту завдяки високій швидкості, точності та ефективності, що забезпечується її одноетапною архітектурою [28]. Робота моделі схематично зображена в додатку В. Потрібно зазначити, що модель YOLO11 дозволяє

виявляти та класифікувати об'єкти за один прохід, що робить її ідеальною для роботи в реальному часі.

3.1.7 Програмне забезпечення для розробки застосунку

Для розробки системи було використано PyCharm та SQLiteStudio.

PyCharm – це інтегроване середовище розробки, спеціально створене для програмування на Python[29]. PyCharm надає розширений набір інструментів для написання, тестування та налагодження коду прямо з коробки, що дозволяє почати створювати програми без спеціальних налаштувань. Також, до переваг інтегрованого середовища розробки PyCharm для розробки системи стеження за повітряними та наземними цілями мобільного робота на базі комп'ютерного зору можна віднести:

- інтеграція з інструментами контролю версій, що надає зручний інтерфейс комунікації з репозиторіями Git для відстеження змін;
- підтримка віртуальних середовищ, що забезпечує легкий механізм створення та управління ізольованими середовищами Python;
- широка підтримка бібліотек та плагінів, дозволяє можливість інтеграції з популярними інструментами для полегшення розробки та покращення якості коду.

SQLite Studio – це інструмент для управління базами даних SQLite із графічним інтерфейсом[30]. Ця програма була обрана, бо:

- дозволяє зручно створювати, змінювати та аналізувати структуру бази даних;
- виконувати SQL–запити та переглядати результати у зручному форматі;
- не потребує встановлення;
- підтримує кросплатформенну роботу;
- надає прості у використанні функції для роботи з базами SQLite.

3.2 Структурна схема системи

У цьому розділі будуть наведені модулі системи, що були розроблені керуючись функціональними та нефункціональними вимогами. Схематичне зображення модулів наведено у додатку Д.

Розроблена система складається із двох головних модулів, а саме: модуль симуляції та модуль інтерфейсу симуляції.

Головною задачею модуля симуляції є отримання вхідних даних від модуля інтерфейсу, обробка та надсилання відповіді назад. Можна виділити наступні частини модулю симуляції:

- модуль керування мобільного робота. Цей модуль відповідає за навігацію мобільного робота системи, що є базою системи стеження. Модуль налічує логіку надсилання даних з бортових ультразвукових датчиків робота і його камери. Також, цей модуль керує двигунами робота, що забезпечує його мобільність;

- модуль камери та датчика відстані системи стеження. Робота цього модулю полягає у опрацюванні даних з камер для використання їх у алгоритмі стеження за об'єктом і передачі результатів роботи алгоритму у модуль керування роботичною рукою системи стеження. Також, даний модуль відповідає за надсилання відео з камер та даних з датчика відстані на модуль інтерфейсу системи;

- модуль керування роботичною рукою. Даний модуль використовує дані з модуля камери та датчика відстані системи стеження для управління складовими роботичної руки так, щоб об'єкт стеження постійно був у кадрі.

Зі свого боку модуль інтерфейсу системи можна поділити на такі складові:

- модуль авторизації користувачів. Даний модуль відповідає за отримання та оброблення даних із форми логіну для забезпечення відповідного функціоналу користувачам;

- модуль роботи із базою даних. Задачі цього модулю полягають у забезпеченні з'єднання із сховищем даних та обробці запитів від користувачів;

– модуль обробки даних із підмодуля камери та датчика відстані системи стеження. Цей модуль використовує відео дані із камер системи стеження для подальшої обробки їх нейронною мережею для класифікації об'єкта. Також, до задач можна віднести візуалізацію результатів класифікації та відображенні відстані до об'єкту стеження;

– модуль обробки даних із підмодуля керування мобільного робота. Даний модуль відповідає за обробку та візуалізацію даних із бортових сенсорів мобільного робота. Також, цей модуль займається надсиланням запитів керування мобільного робота із інтерфейсу системи до модуля симуляції.

3.3 Проектування бази даних

Для забезпечення надійного зберігання та обробку даних було спроектовано базу даних, схему якої наведено в додатку Е. Далі буде наведено детальний огляд кожної з таблиць.

В системі є такі таблиці:

- Users – зберігає дані про користувачів системи;
- Robots – інформація про мобільних роботів;
- Settings – конфігурації модулів роботів;
- Logs – журнал подій системи;
- Roles – зберігає дані про ролі.

Таблиця Users (таблиця 3.2) складається із таких даних: ідентифікатор користувача, який є унікальним; ім'я користувача; позивний; пароль у захешованому вигляді; роль користувача та дати створення і оновлення запису .

Ця таблиця має зовнішній ключ на таблицю Roles, що визначає доступні ролі користувачів.

На таблицю Users посилається таблиця: Logs.

Таблиця 3.2 – Users

Назва колонки	Тип даних	Атрибути
UserID	INT	PRIMARY KEY
Name	VARCHAR(255)	NOT NULL
Callsign	VARCHAR(255)	UNIQUE, NOT NULL
SquadID	INT	NOT NULL
PasswordHash	VARCHAR(255)	NOT NULL
Role	INT	FOREIGN KEY
CreatedAt	DATETIME	DEFAULT CURRENT_TIMESTAMP
UpdatedAt	DATETIME	ON UPDATE CURRENT_TIMESTAMP

Таблиця Robots (таблиця 3.3) містить інформацію про мобільних роботів, зокрема ідентифікатор робота, назву, тип робота, поточний статус, дату останнього обслуговування і дату створення запису.

Ця таблиця не має зовнішніх ключів, але на неї посилається таблиця Settings.

Таблиця 3.3 – Robots

Назва колонки	Тип даних	Атрибути
RobotID	INT	PRIMARY KEY, AUTO_INCREMENT
Name	VARCHAR(255)	NOT NULL
Type	VARCHAR(255)	NOT NULL

Назва колонки	Тип даних	Атрибути
Status	VARCHAR(255)	DEFAULT 'Active'
LastMaintenance	DATE	NULL
CreatedAt	DATETIME	DEFAULT CURRENT_TIMESTAMP

Таблиця Settings (таблиця 3.4) містить інформацію про налаштування, включаючи ідентифікатор налаштувань, ідентифікатор робота, тип модуля, параметри налаштувань у форматі JSON та дату застосування.

Ця таблиця має зовнішній ключ на таблицю Robots, що вказує, до якого робота належать налаштування.

Таблиця 3.4 – Settings

Назва колонки	Тип даних	Атрибути
SettingID	INT	PRIMARY KEY, AUTO_INCREMENT
RobotID	INT	FOREIGN KEY
ModuleType	VARCHAR(255)	NOT NULL
Parameters	JSON	NULL
AppliedAt	DATETIME	DEFAULT CURRENT_TIMESTAMP

Таблиця Logs (таблиця 3.5) містить журнал подій, включаючи ідентифікатор запису, опис події, ідентифікатор користувача, ідентифікатор робота, дату і час події

Ця таблиця має зовнішні ключі на таблиці Users та Robots.

Таблиця 3.5 – Logs

Назва колонки	Тип даних	Атрибути
LogID	INT	PRIMARY KEY, AUTO_INCREMENT
Event	VARCHAR(255)	NOT NULL
UserID	INT	FOREIGN KEY
RobotID	INT	FOREIGN KEY
Timestamp	DATETIME	DEFAULT CURRENT_TIMESTAMP

Таблиця Roles (таблиця 3.6) містить інформацію про ролі системи, включаючи ідентифікатор ролі та назву ролі.

На цю таблицю посилається таблиця Users. Зовнішніх ключів таблиця не має.

Таблиця 3.6 – Roles

Назва колонки	Тип даних	Атрибути
RoleID	INT	PRIMARY KEY, AUTO_INCREMENT
Name	VARCHAR(255)	NOT NULL

3.4 Реалізація бізнес–логіки системи

Система складається з трьох основних частин:

- модуль симуляції;
- модуль інтерфейсу пілота;
- модуль інтерфейсу системи керування даними.

Модуль симуляції відповідає за забезпечення та підтримку симуляційної частини створеної системи. Його задачі складаються у обробці фізичних дій з мобільним роботом, завантаження та відображення мобільного робота, модуля стеження мобільного робота та оточення мобільного робота. Також, модуль займається обробленням команд, що надійшли від пілота.

Як приклад команд від пілота можуть надходити запити на управління, а саме: їхати вперед, їхати назад, повертати ліворуч та повертати праворуч. Зі свого боку, симуляції після отримання команд повинна їх обробити та інтерпретувати у інструкції для двигунів мобільного робота.

Після цього модуль симуляції повинен надіслати результат виконання задачі. Для цього використовуються камери, за допомогою яких, пілот може бачити, що мобільний робот виконав задачу.

Наступною складовою є модуль інтерфейсу пілота. Задачі цього модуля є надсилання задач на модуль симуляції та отримання відповіді з їх подальшою візуалізацією.

Як приклад можна навести функціонал перемикання режимів мобільного робота між режимом стеження та режимом пересування. Пілот має перемикач для виконання зазначеної операції. При перемиканні модуль надсилає через мережу відповідну команду, а мобільний робот, зі свого боку, при отриманні повинен її виконати. Після виконання команди, пілот, використовуючи екран та показники з камер та датчиків робота може бачити результат виконання.

Модуль інтерфейсу системи керування даними відповідає за спілкування із сховищем даних для отримання, перегляду та редагування інформації. Доступ до модуля має користувачі із роллю «Адміністратор» та «Пілот мобільного робота». Це досягається використанням механізмів авторизації та автентифікації, що забезпечує надійний доступ до важливих даних.

Для прикладу, було розглянуто функціонал редагування профілів користувачів адміністратором. Для цього адміністратор повинен увійти у систему під власний логіном та паролем. Після цього система відобразить сторінку адміністратора на якій

адміністратор зможе, відкривши профіль користувача, змінити обраний профіль. Після зміни система зберігає нові дані.

Інший приклад – функціонал перегляду доступних модифікацій модулю стеження мобільного робота. Цього разу, пілот мобільного робота увійшов у систему під своїми даними. Система перевела його на сторінку, що доступна лише користувачам із роллю «Пілот мобільного робота». Користувач може побачити список доступних модифікацій, а при подвійному натисканні система відображає повну інформації про обрану модифікацію.

3.5 Опис інтерфейсу користувача та інструкція використання застосунку

Інтерфейс системи було розроблено за допомогою мови python та безкоштовної бібліотеки PyQt6. PyQt6 – це комплексний набір прив'язок Python для Qtv6. Він реалізований у вигляді більш ніж 35 модулів розширення та дозволяє використовувати Python[32] як мову для розробки крос-платформних додатків.

Інтерфейс системи стеження за повітряними та наземними цілями мобільного робота на основі комп'ютерного зору, як було зазначено, можна поділити на два головні модуля, а саме: модуль інтерфейсу пілота та модуль інтерфейсу системи керування даними.

Далі, буде розписано кожен сторінку застосунку та її функціонал. Схематичну карту сторінок застосунку можна знайти в додатку Ж.

Далі було розглянуто модуль інтерфейсу системи керування даними. При першому запуску система відображає сторінку авторизації (рисунк 3.4). Користувачу запропоновано ввести юзернейм та пароль, що було надано користувачем із роллю «Адміністратор», а також натиснути клавішу «Log in».

Після введення необхідних даних запускається процес перевірки введених значень. Система відправляє запит на сервер, де зберігається інформація про користувачів. Спочатку перевіряється наявність введеного юзернейму користувача у списку зареєстрованих облікових записів. Якщо такий запис існує, система порівнює введений пароль із відповідним значенням у базі даних. У випадку успішного збігу

користувач отримує доступ до основного інтерфейсу системи керування даними. При помилці у введенні логіна або пароля на екрані з'являється сповіщення з відповідним повідомленням про невдачу авторизацію.

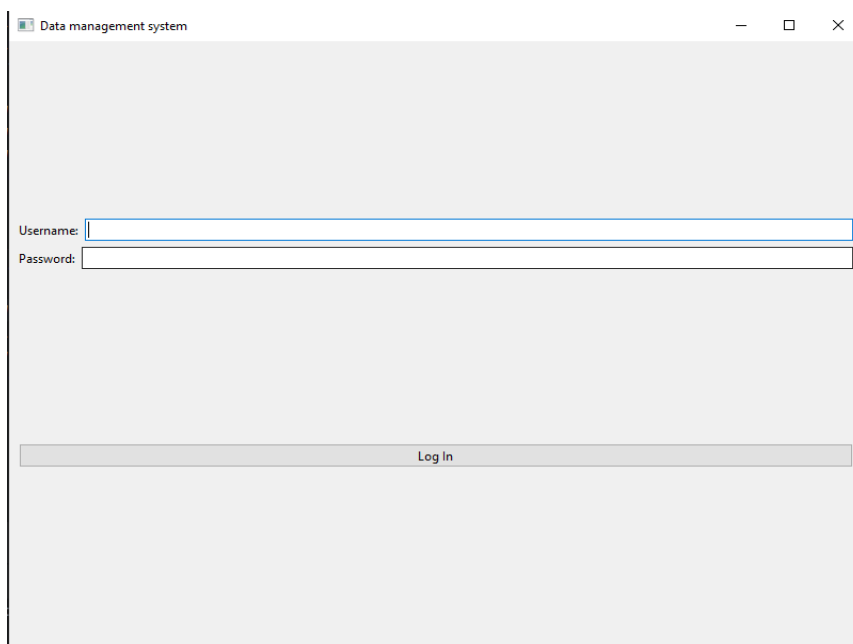


Рисунок 3.4 – Сторінка авторизації

Далі буде розглянуто сторінку користувача із роллю «Адміністратор».

Сторінка адміністратора (рисунок 3.5) є одним із ключових елементів системи, що забезпечує повний контроль над даними та користувачами. Вона містить функціонал для вибору діапазону дат, що дозволяє адміністратору фільтрувати дані за певний період часу для отримання розширених логів роботи системи. Поруч знаходиться кнопка «Get Logs», яка, використовується для завантаження логів за обраний інтервал часу. Також доступна кнопка «Log Out» що дозволяє адміністратору завершити роботу та повернутися до сторінки авторизації. Нижче розташований розділ «List of Users». У цьому розділі відображається список усіх користувачів, які зареєстровані в системі. Адміністратор має можливість переглядати облікові записи та фільтрувати їх за іменем, що значно спрощує пошук конкретного користувача. У нижній частині сторінки знаходиться розділ «List of Modifications», який відображає параметри, що стосуються модифікацій модулю стеження. Для полегшення навігації

й пошуку потрібної інформації у таблиці передбачене поле для фільтрації, яке дозволяє обмежити відображені дані за певними критеріями.

Data Management System

From: 11/18/2024 To: 11/18/2024 Get Logs

Log Out

List of Users

Filter users by name...

	Name	Callsign	DOF	Group ID
1	Alice12	AL1	2023-05-01	A
2	Bob	BO2	2022-11-12	B

List of Modifications

Filter modifications by name...

	Name	Colors	Model Path	Robots IDs
1	Main1	Red, Blue, Green	/path/to/model1.pt	1, 2, 3
2	Secondary	Yellow, Purple	/path/to/model2.pt	4, 5

Рисунок 3.5 – Сторінка адміністратора

Також адміністратор має можливість редагувати профілі користувачів та налаштовувати модифікації модулю стеження. На рисунках 3.6 та 3.7 наведено модальні вікна за допомогою яких відбуваються такі дії.

У процесі редагування профілю користувача адміністратор має можливість змінити основні дані, такі як ім'я користувача, позивний, дату приєднання до загону та ідентифікатор загону. Всі поля редагування організовані у зручному форматі, що дозволяє швидко оновити потрібну інформацію. Окрім основних параметрів профілю, адміністратору надається можливість працювати зі списком роботів, призначених конкретному користувачу. Цей список відображається у вигляді таблиці, що забезпечує зручний огляд усіх доступних роботів. Для роботи зі списком є кнопки

«Add Robot», для додавання нового робота, та «Remove Selected Robot», для видалення робота. У нижній частині знаходяться кнопки функціоналу збереження змін і їх відміни.

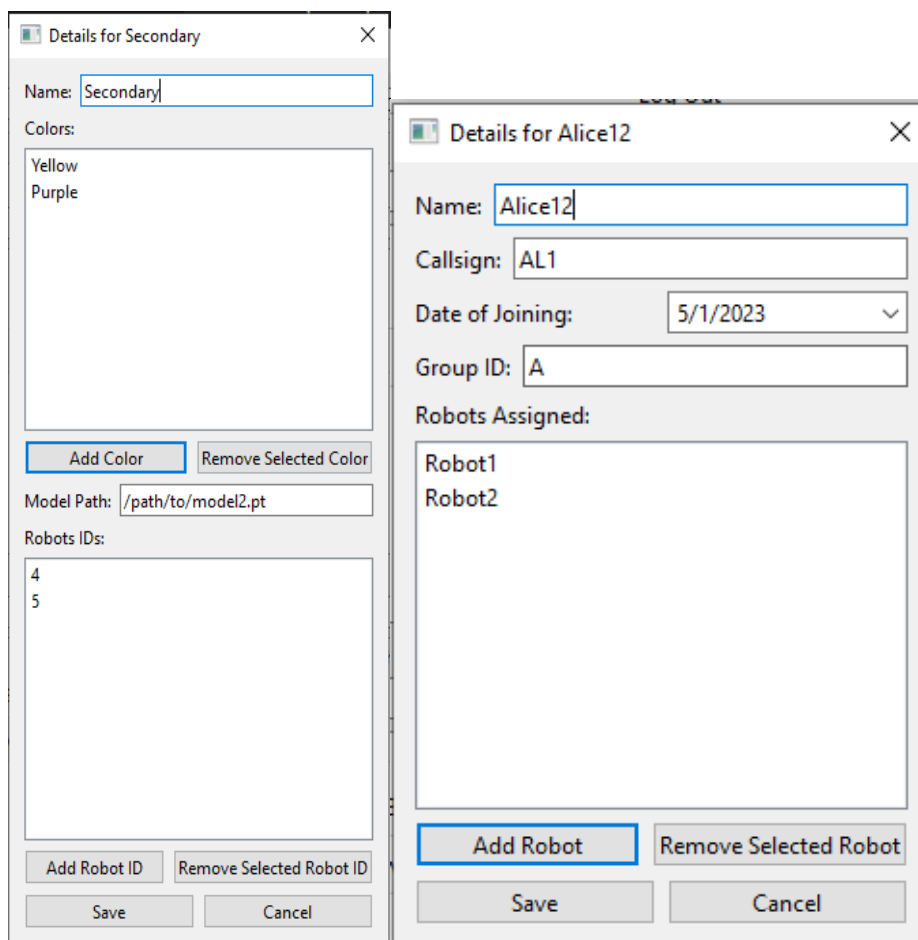


Рисунок 3.6, 3.7 – Модальне вікно редагування профілю користувача, Модальне вікно редагування модифікації

Модальне вікно для редагування інформації про модифікації надає адміністраторам можливість змінювати важливі параметри, пов'язані з конфігурацією системи. Основними елементами цього вікна є поля для оновлення назви модифікації, списку кольорів, шляху до моделі нейронної мережі та списку роботів, які застосовують цю конкретну модифікацію. Усі ці елементи організовані у структурованій формі, що забезпечує зручну навігацію та внесення змін. Для додавання та видалення кольорів та роботів присутні відповідні кнопки. Кнопки функціоналу збереження змін і їх відміни аналогічні з попереднім модальним вікном.

Логи зберігаються у текстовому файлі (рисунок 3.8). У логах відображаються ключові параметри кожної події: ідентифікаційний номер, юзернейм користувача, який ініціював дію, а також точний час спрацювання події. Така структура логів дозволяє адміністраторам детально аналізувати активність у системі, відслідковувати зміни та оперативно реагувати на потенційні проблеми.

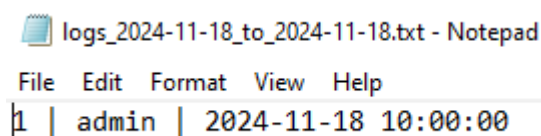


Рисунок 3.8 – Приклад лог-файлу

Наступною сторінкою є сторінка пілота (рисунок 3.9). На сторінці видно інтерфейс системи перегляду даних, який організований у два основних розділи: список користувачів загону пілота і список модифікацій, що присутні у його роботів. Розділ «Список користувачів» містить таблицю, у якій представлені дані про користувачів. Над таблицею є поле фільтрації за іменем. Далі можна побачити таблицю зі списком модифікацій, аналогічно до списку користувачів є поле фільтрації.

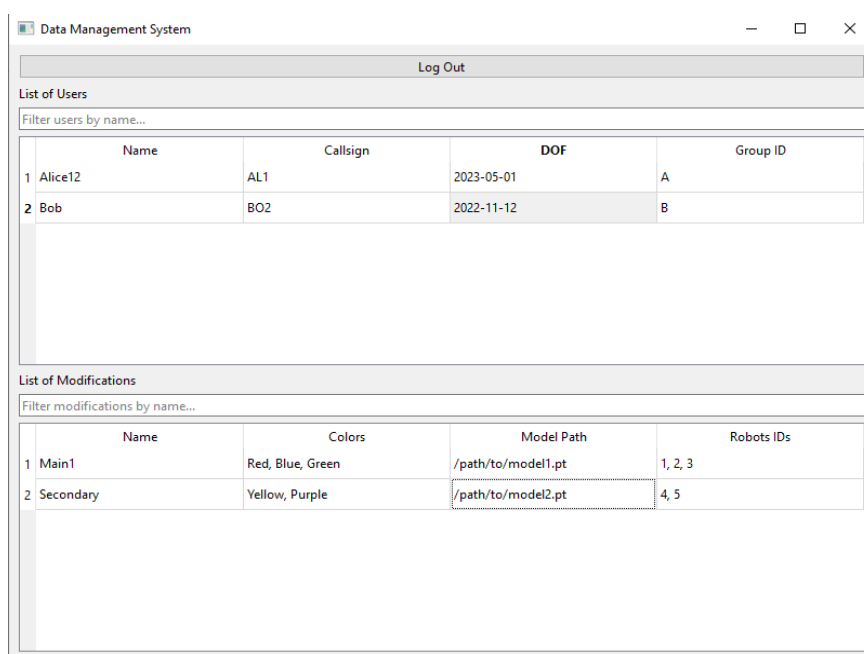


Рисунок 3.9 – Сторінка пілота мобільного робота

Також пілот має можливість переглядати профілі користувачів та налаштовувати модифікації модулю стеження. На рисунках 3.10 та 3.11 наведено модальні вікна за допомогою яких відбуваються такі дії.

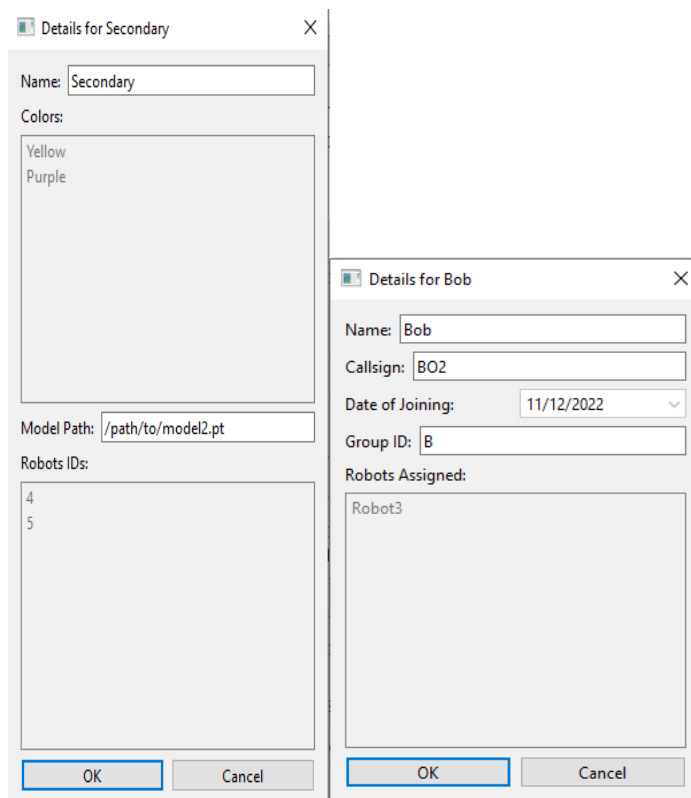


Рисунок 3.10, 3.11 – Модальне вікно перегляду профілю користувача, Модальне вікно перегляду модифікації

Модальне вікно призначене для перегляду інформації про користувача. У ньому присутні текстові поля імені користувача, позивного, дати приєднання до загону та ідентифікатора групи. Нижче знаходиться список призначених роботів. У нижній частині є дві кнопки: «OK» і «Cancel» які закривають модальне вікно.

Друге модальне вікно призначене для перегляду інформації про модифікацію. У верхній частині розташоване текстове поле для імені модифікації. Нижче знаходиться список кольорів, текстове поле для шляху до моделі нейронної мережі і список ідентифікаторів роботів. У нижній частині також є дві кнопки з аналогічним функціоналом.

Наступною складовою системи є модуль інтерфейсу пілота (рисунок 3.12). У верхній частині вікна розташовані кнопки «Start» і «Stop» для запуску та зупинки роботи системи симуляції, а також випадаюче меню для вибору кольору цілі стеження. У прикладі вибрано «Red», але доступні також варіанти «Green» і «Blue». Кольори, відповідають обраній модифікації для конкретного робота. Нижче розташоване ще одне випадаюче меню для вибору режиму роботи системи. Доступні два режими: відстеження (Tracking) і водіння (Driving).



Рисунок 3.12 – Сторінка модулю інтерфейсу пілота

У режимі відстеження центральна частина вікна займає відеопотік, що надходить із модуля стеження, який обробляється в реальному часі. На зображенні видно ціль червоного кольору, якого система визначила за допомогою алгоритмів комп'ютерного зору модуля стеження. Над роботом відображається прямокутник (рисунок 3.13), що є результатом аналізу зображення обраною у модифікації нейронною мережею.

У режимі водіння активується система парктроніків, яка представлена сірими прямокутниками (рисунок 3.14). Ці прямокутники змінюють колір залежно від

відстані до об'єктів: зелені для безпечної відстані, жовті для середньої, і червоні для критично близької відстані. У нижній частині розташоване текстове поле, яке використовується для виведення системної інформації та статусу роботи.

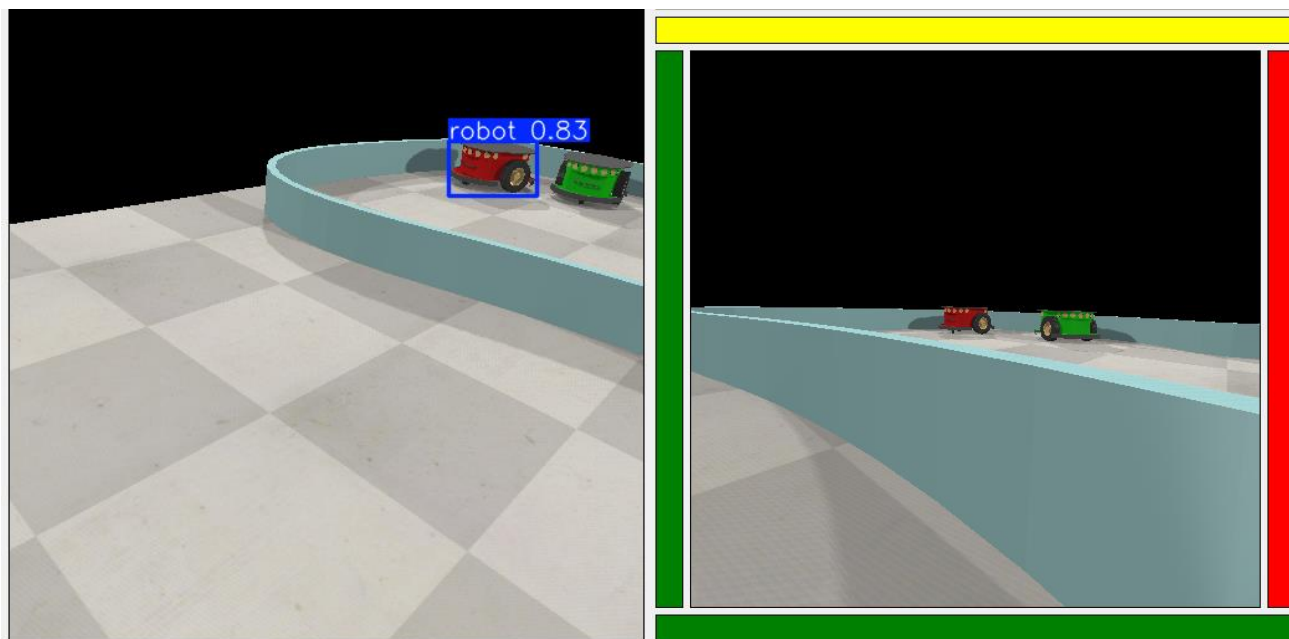


Рисунок 3.13, 3.14 – Візуалізація роботи нейронної мережі, Система парктроніків

Висновки до розділу

У цьому розділі обґрунтовано вибір елементів і технологій для створення системи стеження за повітряними та наземними цілями мобільного робота на основі комп'ютерного зору. Виявлення та стеження реалізовані за допомогою алгоритму, що базується на кольорі цілі, забезпечуючи високу швидкість, стабільність і точність роботи. Архітектурно система реалізована у вигляді клієнт-серверної моделі. Клієнтська частина представлена інтерфейсом пілота мобільного робота, що забезпечує інтуїтивно зрозумілий користувацький досвід. Серверна частина виконує функції обробки даних і симуляції в середовищі, створеному для перевірки роботи системи. Такий підхід дозволяє розділити задачі між компонентами, що підвищує масштабованість і продуктивність.

Для програмної реалізації системи обрано мову Python, яка завдяки бібліотекам OpenCV та NumPy дозволяє ефективно обробляти зображення та виконувати складні

математичні розрахунки. Для розробки інтерфейсу використано PyQt6, що забезпечує створення кросплатформеного графічного інтерфейсу з високою гнучкістю у налаштуванні. Нейронна мережа YOLO11 була інтегрована для класифікації цілей, що значно розширює можливості системи у визначенні об'єктів. База даних, побудована на SQLite, відповідає за збереження результатів операцій, параметрів модулів стеження та інших даних, необхідних для аналізу та управління.

Було створено структурну схему, яка відображає взаємодію між основними компонентами системи, що дозволило ретельно спроектувати її логіку та зв'язки. База даних також була детально спланована, з чітким описом кожної таблиці та її взаємодії з іншими елементами системи. Бізнес-логіка охопила всі ключові процеси, включаючи виявлення об'єктів, стеження за ними та управління мобільними роботами, забезпечуючи повну функціональність системи.

Інтерфейс користувача був створений з урахуванням потреб користувачів різних ролей. Для адміністратора передбачено розширений функціонал управління даними та конфігурації системи, тоді як пілоту мобільного робота надано доступ до інструментів управління та моніторингу в реальному часі. Усі ці аспекти проілюстровані прикладами, що демонструють практичну реалізацію та можливості системи.

.

4 ТЕСТУВАННЯ СИСТЕМИ

4.1 Мануальне тестування

Один із фундаментальних принципів тестування свідчить: 100% автоматизація неможлива[32]. Відомо, що найпершим етапом тестування усіх систем є тестування ручне. Ручне тестування, або ж мануальне – це таке тестування при якому не використовуються ніякі засоби автоматизації і перевірка системи відбувається людиною. Для виконання тестування системи, зазвичай, тестувальник складає собі план тестування, що складається із тест-кейсів. У наступних розділах було сформовано план тестування для кожної ролі системи.

4.1.1 Загальні тест-кейси системи

Загальні тест-кейси системи наведено у таблиці 4.1.

Таблиця 4.1 – Загальні тест-кейси системи

Назва	Вхідні дані	Очікуваний результат
Автентифікація із коректними даними ролі «Адміністратор»	Введені коректні дані юзернейму та паролю для ролі «Адміністратор» та натиснута кнопка «Log in»	Успішний вхід у систему та переадресація на сторінку адміністратора.
Автентифікація із коректними даними ролі «Пілот мобільного робота»	Введені коректні дані юзернейму та паролю для ролі «Пілот мобільного робота» та натиснута кнопка «Log in»	Успішний вхід у систему та переадресація на сторінку профілю пілота.

Назва	Вхідні дані	Очікуваний результат
Автентифікація із некоректними даними	Введені некоректні дані юзернейму та пароллю	Система успішно реагує на некоректні дані та показує помилку
Перегляд профілів членів загону	Користувач увійшов у систему та знаходиться на сторінці адміністратора чи профілю пілота. Користувач бачить список профілів членів загону та подвійно натискає на певний профіль	Система успішно реагує на подвійне натискання на профіль та відображає модальне вікно профілю користувача.
Перегляд доступних модифікацій модулю спостереження	Користувач увійшов у систему та знаходиться на сторінці адміністратора чи профілю пілота. Користувач бачить список доступних модифікацій модулю спостереження та подвійно натискає на певну модифікацію	Система успішно реагує на подвійне натискання на модифікацію та відображає модальне вікно обраної модифікації.

Створивши план тестування було проведено мануальне тестування та занотовано результати у таблицю 4.2.

Таблиця 4.2 Результати тестування загальних тест-кейсів системи

Назва	Отриманий результат	Статус
Автентифікація із коректними даними ролі «Адміністратор»	Успішний вхід у систему та переадресація на сторінку адміністратора.	Успіх
Автентифікація із коректними даними ролі «Пілот мобільного робота»	Успішний вхід у систему та переадресація на сторінку профілю пілота.	Успіх
Автентифікація із некоректними даними	Система відобразила помилку	Успіх
Перегляд профілів членів загону	Система відобразила модальне вікно профілю користувача.	Успіх
Перегляд доступних модифікацій модулю спостереження	Система відобразила модальне вікно обраної модифікації.	Успіх

Можна зробити висновками, що система взаємодіє із користувачами різних ролей коректно.

4.1.2 Тест-кейси ролі «Адміністратор»

У цьому розділі описаний план тестування системи для ролі «Адміністратор» (таблиця 4.3).

Створивши план тестування було проведено мануальне тестування та занотовано результати у таблицю 4.4.

Можна зробити висновки, що система взаємодіє із адміністратором працює коректно.

Таблиця 4.3 Тест–кейси ролі «Адміністратор»

Назва	Вхідні дані	Очікуваний результат
Керування профілями користувачів. Відображення модального вікна.	Адміністратор знаходить на сторінці адміністратора. Адміністратор бачить список профілів інших користувачів системи. Адміністратор подвійно натискає на профіль користувача із списку	Система реєструє подвійне натискання на профіль користувача і відображає модальне вікно редагування інформації обраного користувача.
Керування профілями користувачів. Зміна та збереження даних.	Адміністратор бачить модальне вікно редагування. Адміністратор змінює дані обраного профілю. Адміністратор зберігає дані.	Система реєструє на зміну даних та оброблює подію збереження змін. Система оновлює список користувачів. Система оновлює дані користувача у сховищі.
Керування модифікаціями модуля спостереження. Відображення модального вікна.	Адміністратор знаходить на сторінці адміністратора. Адміністратор бачить список модифікацій. Адміністратор подвійно натискає на модифікацію із списку	Система реєструє подвійне натискання на модифікацію і відображає модальне вікно редагування.

Назва	Вхідні дані	Очікуваний результат
Керування модифікаціями модуля спостереження. Зміна та збереження даних.	Адміністратор бачить модальне вікно редагування. Адміністратор змінює дані модифікації Адміністратор зберігає дані.	Система реєструє на зміну даних та оброблює подію збереження змін. Система оновлює список модифікаціями модуля спостереження. Система оновлює дані модифікації у сховищі.
Перегляд розширеної інформації про зміни у системі	Адміністратор знаходиться на сторінці адміністратора. Адміністратор бачить функціонал завантаження розширеної інформації про зміни у системі. Адміністратор ініціює завантаження даних.	Система оброблює запит на завантаження розширеної інформації про зміни у системі за певний період часу та генерує лог-файл формати txt.

Таблиця 4.4 Результати тестування тест-кейси ролі «Адміністратор»

Назва	Отриманий результат	Статус
Керування профілями користувачів. Відображення модального вікна.	Система відображає модальне вікно редагування інформації обраного користувача.	Успіх
Керування профілями користувачів. Зміна та збереження даних.	Система оновила список користувачів. Система оновила дані користувача у сховищі	Успіх

Назва	Отриманий результат	Статус
Керування модифікаціями модуля спостереження. Відображення модального вікна.	Система відображає модальне вікно редагування модифікації із списку	Успіх
Керування модифікаціями модуля спостереження. Зміна та збереження даних.	Система оновила список модифікацій модуля спостереження. Система оновила дані модифікації у сховищі.	Успіх
Перегляд розширеної інформації про зміни у системі	Система згенерувала файл розширеної інформації про зміни у системі за певний період часу формату txt.	Успіх

4.1.3 Тест–кейси ролі «Пілот мобільного робота»

У цьому розділі описаний план тестування системи для ролі «Пілот мобільного робота» (таблиця 4.5) та у таблиці 4.6 наведено результати тестування.

Таблиця 4.5 – Тест–кейси ролі «Пілот мобільного робота»

Назва	Вхідні дані	Очікуваний результат
Перемикання режимів пілотування	Пілот знаходиться у системі управління мобільним роботом. Пілот бачить	Система оброблює запит на перемикання режимів пілотування. Система змінює вид камери та використовує логіку

Назва	Вхідні дані	Очікуваний результат
	функціонал перемикання режимів пілотування. Пілот ініціює перемикання режимів.	відповідного режиму пілотування.
Перегляд результатів стеження у реальному часі	Пілот знаходиться у системі управління мобільним роботом. Пілот бачить функціонал перегляду результатів стеження у реальному часі.	Система у режимі пілотування «Стеження» використовує дані із модуля стеження мобільного робота і транслює зображення після обробки нейронною мережею на екран пілота
Аналіз результатів роботи дрона	Пілот знаходиться у системі управління мобільним роботом. Пілот бачить функціонал перегляду результатів роботи дрона.	Система оброблює інформації з мобільного робота та транслює її у вигляді тексту у відповідне поле інтерфейсу системи управління мобільним роботом
Забезпечення надійної роботи дрону	Пілот знаходиться у системі управління мобільним роботом. Пілот бачить функціонал забезпечення надійної роботи дрону	Система у режимі пілотування «Пересування» використовує дані із бортових систем мобільного робота і візуалізує показники з цих систем у інтерфейсі системи управління мобільним роботом

Створивши план тестування було проведено мануальне тестування та занотовано результати у таблицю 4.6.

Таблиця 4.6 Результати тестування тест-кейси ролі «Пілот мобільного робота»

Назва	Отриманий результат	Статус
Перемикання режимів пілотування	Система змінила вид камери та використовує логіку відповідного режиму пілотування. Пілот має змогу виконати основні дії у межах трьох кліків.	Успіх
Аналіз результатів роботи дрона	Система транслює результати у вигляді тексту у відповідне поле інтерфейсу системи управління мобільним роботом	Успіх
Перегляд результатів стеження у реальному часі	Система у режимі пілотування «Стеження» транслює зображення після обробки нейронною мережею на екран пілота. Час обробки не перевищує 50 мс на кожен кадр. Частота алгоритму 24 мс на кожен кадр	Успіх
Забезпечення надійної роботи дрону	Система у режимі пілотування «Пересування» візуалізує показники з бортових систем у інтерфейсі системи управління	Успіх

Назва	Отриманий результат	Статус
	мобільним роботом. Система робота може працювати більше 12 годин.	

Можна зробити висновки, що система взаємодіє із пілотом працює коректно.

4.1.4 Тестування функціоналу системи симуляції мобільного робота

У цьому розділі описаний план тестування взаємодії інтерфейсу системи з системою симуляції мобільного робота (таблиця 4.7) та у таблиці 4.8 наведено результати тестування.

Таблиця 4.7 – План тестування взаємодії інтерфейсу системи

Назва	Вхідні дані	Очікуваний результат
Завантаження сцени симуляції	Система симуляції мобільного робота ініціює завантаження сцени системи симуляції	Сцена завантажується коректно та інтерфейс системи керування повідомляє про це
Ініціалізація робота	Система симуляції мобільного робота ініціалізує початкові значення для підмодулів мобільного робота.	Початкові значення підмодулів мобільного робота ініціалізуються коректно та інтерфейс системи керування повідомляє про це
Відстеження об'єктів на основі	Система симуляції мобільного робота	Система симуляції коректно отримує характеристики цілі та

Назва	Вхідні дані	Очікуваний результат
комп'ютерного зору	знаходиться у режимі симуляції та обрано режим «Стеження»	слідкує на нею
Визначення дистанції до об'єкту за допомогою датчиків відстані	Система симуляції мобільного робота знаходиться у режимі симуляції та обрано режим «Стеження»	Система симуляції коректно вимірює відстань до цілі стеження та надсилає дані на інтерфейс системи управління
Визначення дистанції до об'єктів оточення за допомогою датчиків відстані	Система симуляції мобільного робота знаходиться у режимі симуляції та обрано режим «Пересування»	Система симуляції коректно вимірює відстань до об'єктів оточення за допомогою датчиків відстані та надсилає дані на інтерфейс системи управління
Повернення систем мобільного робота до початкових значень	Симуляція мобільного робота закінчена	Симуляція закінчується коректно та інтерфейс системи керування повідомляє про це

Створивши план тестування було проведено мануальне тестування та занотовано результати у таблицю 4.8. Слід зауважити, що тестування тест-кейсу «Відстеження об'єктів на основі комп'ютерного зору» відбувалося покроково, згідно з, діаграмою послідовності процесу стеження в додатку Г.

Таблиця 4.8 – Результати тестування функціоналу системи симуляції мобільного робота

Назва	Отриманий результат	Статус
Завантаження сцени	Сцена завантажилася коректно	Успіх

Назва	Отриманий результат	Статус
симуляції	та інтерфейс системи керування повідомив про це	
Відстеження об'єктів на основі комп'ютерного зору	Система симуляції коректно отримує характеристики цілі та слідкує за нею.	Успіх
Ініціалізація робота	Початкові значення підмодулів мобільного робота ініціалізовані коректно та інтерфейс системи керування повідомляє про це	Успіх
Визначення дистанції до об'єкту за допомогою датчиків відстані	Система симуляції коректно вимірює відстань до цілі стеження та надсилає дані на інтерфейс системи управління. Реакція на команди 85 мс після подачі команди.	Успіх
Визначення дистанції до об'єктів оточення за допомогою датчиків відстані	Система симуляції коректно вимірює відстань до об'єктів оточення за допомогою датчиків відстані та надсилає дані на інтерфейс системи управління	Успіх
Повернення систем мобільного роботу до початкових значень	Симуляція мобільного робота закінчена	Успіх

Висновок до розділу

Результати тестування підтвердили високу ефективність, функціональність і надійність розробленої системи стеження. Система демонструє стабільну роботу при виконанні задач розпізнавання та стеження за об'єктами, включаючи як статичні, так і динамічні цілі, навіть за умов різноманітних перешкод і складної траєкторії руху. Алгоритми комп'ютерного зору забезпечили точне визначення об'єктів і стабільне оновлення позицій у реальному часі, що підтверджує їхню здатність до адаптації в умовах реальних викликів.

Завдяки оптимізованій архітектурі система зберігає високий рівень продуктивності, що гарантує мінімальну затримку та оперативну реакцію. Тестування також підтвердило відповідність системи нефункціональним вимогам у критичних сценаріях і зручність використання для оператора. Випробування показали, що система коректно реагує на команди користувачів, забезпечуючи точне виконання інструкцій без втрати даних чи збою алгоритмів.

Під час тестування модуль стеження довів свою здатність обробляти зображення з високою швидкістю, використовуючи ресурси економно. Водночас функціонал виявлення перешкод і визначення відстаней дозволив уникати потенційних зіткнень та інших ризиків, забезпечуючи безпеку експлуатації. Загалом, система виконала всі поставлені вимоги, довівши свою готовність до подальшої інтеграції у реальні сценарії використання та масштабування, що підтверджує її високу практичну цінність і відповідність сучасним технічним стандартам.

5 СТАРТАП ПРОЕКТ

5.1 Опис ідеї проекту

В рамках цього підрозділу будуть розглянуті наступні складові:

- зміст ідеї стартап–проекту;
- напрямки застосування стартап–проекту;
- вигоди для кінцевого користувача;
- відмінність від вже існуючих аналогів.

Перші три пункти з вищенаведеного списку наведено в таблиці 5.1.

Таблиця 5.1 – Опис ідеї стартап–проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Система стеження за цілями мобільного робота, з двома методами роботи, з модулем стеження у вигляді механізованої руки з сенсорами відстані, камерами, що використовує алгоритми машинного навчання для стеження та класифікації цілей.	Використання у якості модифікації військового дрону пасивного стеження	Мінімізація часу, що потрібний для знаходження та класифікації цілей. Збереження особового складу при виконанні задачі стеження за цілями.
	Використання у якості дрона–рятувальника для пошуку зниклих людей	Мінімізація часу на знаходження людей, що зникли.

Також, потрібно проаналізувати відмінність від вже існуючих аналогів. Для аналізу використовуються наступні характеристик: сильні (S), слабкі (W) та нейтральні (N), які наведено в таблиці 5.2. Аналіз передбачає визначення списку техніко–економічних властивостей ідеї стартап–проекту та порівняння показників запропонованого рішення з товарами–аналогами конкурентів, де оцінюється саме запропонований стартап–проект.

Таблиця 5.2 – Визначення сильних (S), слабких (W) та нейтральних (N) характеристик ідеї проекту стартапу

№ п/п	Техніко– економічні характерист ики ідеї	(потенційні) товари/концепції конкурентів				W (слабка сторон а)	N (нейтра льна сторон а)	S (сильна сторона)
		Мій проект	DJI Drones	FLIR Black Hornet	KEEN ON			
1.	Модульний блок системи стеження	Так	Ні	Ні	Ні	–	–	+
2.	Класифікаці я об’єктів стеження	Так	Ні	Так	Та	–	+	–
3.	Зручність у використан ні людиною	Так	Так	Так	Ні	+	–	–
4.	Вартість	Серед ня	Серед ня	Висок а	Висок а	–	+	–
5.	Наявність клієнтської бази	Так	Ні	Ні	Ні	–	–	+

Проаналізувавши аналоги та заповнивши таблицю 5.2, бачимо, що відносно потенційних аналогів/конкурентів власна ідея є унікальною, що є важливим для успішності та конкурентоспроможності стартап–проекту.

5.2 Технологічний аудит ідеї проекту

У цьому розділі буде проведено аналіз технологій, які мають бути використані про розробці ідеї стартап–проекту, їх наявність та доступність авторам проекту.

Технологічна здійсненність ідеї проекту наведена в таблиці 5.3.

Таблиця 5.3 – Технологічна здійсненність ідеї стартап-проекту

№ п/ п	Ідея проекту	Технології реалізації	Наявність технологій	Доступність технологій
1	СУБД	SQLite	Наявні, доброблювати не потрібно	Доступні
2	Сховище результатів роботи робота	Microsoft Azure Blob Storage	Наявні, доброблювати не потрібно	Доступні
3	Сховище логів роботи робота	MongoDB	Наявні, доброблювати не потрібно	Доступні
4	Обробка даних, отриманих з роботу	Python	Наявні, доброблювати не потрібно	Доступні
5	Мова програмування та фреймворк для ро зробки клієнтсько го інтерфейсу системи	Python, PyQt6	Наявні, доброблювати не потрібно	Доступні
6	Мова програмування для розробки систем робота	Python, Lua	Наявні, доброблювати не потрібно	Доступні
7	Мова програмування	Python, Lua	Наявні, доброблювати не потрібно	Доступні

№ п/ п	Ідея проекту	Технології реалізації	Наявність технологій	Доступність технологій
	для розробки систем робота			
8	Розпізнавання об'єктів на зображенні та визначення їх позиції на зображенні	YOLO11, Roboflow	Наявні, доброблювати не потрібно	Доступні
9	Допоміжні бібліотеки для обробки даних	OpenCV, numpy, math, та інші бібліотеки для роботи з ШІ	Наявні, доброблювати не потрібно	Доступні
Обрані технології реалізації ідеї проекту: мови програмування Python, Lua; модель розпізнавання об'єктів YOLO11; бібліотеки OpenCV, PyOt6				

Проаналізувавши таблицю 5.3, видно, що всі технології, які мають бути використані для реалізації поточного проекту не потребують доробки, є безкоштовними та наявні у відкритому доступі. Також, для більш якісного розуміння принципу застосування обраних технологій було спроектовано ієрархічну структуру систем МР, що зображена в додатку И.

5.3 Аналіз ринкових можливостей запуску стартап–проекту

Даний підрозділ націлений на аналіз ринкових можливостей.

Попередня характеристика потенційного ринку для стартап–ідеї наведена в таблиці 5.4.

Таблиця 5.4 – Попередня характеристика потенційного ринку стартап–проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	<4
2	Загальний обсяг продаж, грн/ум.од	227 млн грн
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу	Відсутні
5	Специфічні вимоги до стандартизації та сертифікації	AES–256 (захисту від несанкціонованого доступу), ISO 13482 (стандарт безпеки для мобільних роботів, що контактують з людьми), ISO 29988 (стандарт, який охоплює робототехнічні сенсори та системи, включаючи технології визначення оточення)
6	Середня норма рентабельності в галузі (або по ринку), %	23–30%

Проаналізувавши потенційний ринок стартап–проекту, можна зробити висновок, що ринкова ситуація є сприятливою для втілення та розвитку ідеї.

Наступним кроком було проаналізовано потенційних клієнтів системи.

Таблиця 5.5 – Характеристика потенційних клієнтів стартап–проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
-------	--------------------------	--	---	-----------------------------

1.	Потреба у надійній системі стеження за об'єктами	Виробники мілітарних та пошукових мобільних робіт.	Мають чіткі регламенти відбору систем та їх сертифікацію	Висока точність, висока надійність, простота використання
2.	Потреба у модульній системі стеження та класифікації цілей	Виробники охоронних систем.		
3.	Потреба у надійній системі класифікації цілей стеження	Виробники статичних систем спостереження	Мають чіткі регламенти відбору систем та їх сертифікацію	Висока точність, висока надійність, простота використання

Далі було проаналізовано фактори загроз та можливостей, що зображені в таблицях 5.6 та 5.7.

Таблиця 5.6 – Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1.	Безпека даних	Незахищені з'єднання можуть призвести до витоку даних або несанкціонованого доступу	Використання шифрування AES-256 для захисту передавання даних, логування всіх дій користувачів

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
2.	Сумісність із різними пристроями	Можливі проблеми із сумісністю з різними ОС або типами камер	Забезпечення підтримки для Windows 10/11, Ubuntu 20.04+ і розширеної інтеграції камер
3.	Збої у роботі мобільного робота	Несподівані збої або зіткнення можуть зашкодити роботі або поставити під загрозу дані	Впровадження систем сигналізації для оператора про можливі зіткнення, використання сенсорів для уникнення
4.	Точність алгоритмів розпізнавання	Помилки в класифікації об'єктів можуть призвести до некоректного моніторингу	Використання вдосконалених моделей, таких як YOLO, та регулярне навчання на актуальних наборах даних

Таблиця 5.7 – Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1.	Розширення ринку	Збільшення попиту на ринку або вихід на нові ринки	Вихід на нові ринки, адаптація продукту до потреб нового ринку
2.	Технологічні інновації	Впровадження нових технологій, які можуть підвищити ефективність виробництва	Інвестування в нові технології для оптимізації процесів та зниження витрат
3.	Зростання доходів споживачів	Збільшення купівельної спроможності клієнтів	Розширення асортименту преміальних продуктів, поліпшення якості обслуговування клієнтів

Отже, компанія стикається з ризиками в сфері безпеки, сумісності та ефективності роботи, але має можливості для зростання через інновації та розширення ринку. Реагуючи на ці фактори, вона може зміцнити свої позиції, підвищуючи якість продуктів і послуг.

Таблиця 5.8 – Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства
1. Тип конкуренції: олігополія	Декілька великих компаній домінують на ринку, створюючи бар'єри для нових учасників і контролюючи цінову політику	Встановлення конкурентних цін, інвестування в інновації та унікальні продукти для виділення серед конкурентів
2. Рівень конкурентної боротьби: національний	Конкуренція зосереджена в межах країни, з акцентом на відповідність локальним потребам та законодавству	Зосередження на локальних маркетингових кампаніях та підтримка репутації бренду в країні
3. Галузева ознака: внутрішньогалузева	Конкуренція відбувається між компаніями однієї галузі, що пропонують схожі продукти або послуги	Підвищення якості продукту та розвиток інновацій для збереження лідерства в галузі
4. Конкуренція за видами товарів: товарно-видова	Конкуренція між різними варіантами одного товару, де важлива якість і специфікація продукту	Інвестиції в розробку продуктів, які краще відповідають потребам споживачів

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства
5. Характер конкурентних переваг: нецінова	Конкуренція базується на унікальних якостях товару, сервісі або бренді, а не на ціні	Інвестиції в якість обслуговування клієнтів та розвиток іміджу бренду
6. Інтенсивність: марочна	Конкуренція базується на лояльності до бренду та впізнаваності	Підтримка позитивного іміджу, активні маркетингові кампанії, розвиток програми лояльності

Далі за допомогою методу М. Портера було проаналізовано умови конкуренції та заповнено таблицю 5.9.

Таблиця 5.9 – Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари замітники
	Відсутні	FLIR, DJI	Сила постачальників висока. Кількість постачальників висока.	Точність, зручність, надійність, безпека	Присутні окремі модулі системи.

Висновки	Прямі конкуренти в галузі відсутні. Інтенсивність конкурентної боротьби низька	Потенційні конкуренти наявні. Бар'єри виходу на ринок середні	Постачальники умови не диктують, бо є легка можливість змінити постачальника	Сила впливу клієнтів середня, та буде зменшуватися з часом. Для клієнта важливіше якість ніж ціна	Сила впливу середня, Стартап втілює симбіоз систем, що покращує зручність користування.
----------	--	---	--	---	---

Виконавши аналіз за М. Портером, видно, що вихід та робота на ринку є можливою. Потрібно зауважити, що для більш ефективного виходу на ринок, стартап–проект повинен мати залежність діапазону цін з модельним рядом системи.

У таблиці 5.10 було визначено фактори конкурентоспроможності стартап–проекту.

Таблиця 5.10 – Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування
1	Інноваційність	Використання технології комп'ютерного зору та машинного навчання для стеження і класифікації цілей, що робить проект унікальним на ринку
2	Гнучкість застосування	Проект може бути використаний як для військових, так і для рятувальних операцій, що розширює цільову аудиторію

№ п/п	Фактор конкурентоспроможності	Обґрунтування
3	Зручність використання	Легкість управління системою, зокрема через модульну конструкцію, що дозволяє швидко адаптувати її під різні умови
4	Висока надійність	Використання шифрування та захисту даних, що відповідає стандартам, необхідним для безпечної експлуатації у військових і рятувальних умовах

Було проведено аналіз сильних та слабких сторін стартап–проєкту опираючись на фактори конкурентоспроможності.

Таблиця 5.11 – Порівняльний аналіз сильних та слабких сторін проєкту

№ п/п	Фактор конкурентоспроможності	Бали 1–20	Рейтинг товарів–конкурентів						
			–3	–2	–1	0	+1	+2	+3
1	Інноваційність	5		+					
2	Гнучкість застосування	7		+					
3	Зручність використання	4				+			
4	Висока надійність	12				+			

Було проведено SWOT–аналіз стартап–проєкту, що полягає у розділенні чинників і явищ на чотири категорії: сильних і слабких сторін проєкту, можливостей та загроз[33]. Цей метод дозволить виявити потенційні переваги та недоліки, а також оцінити зовнішні фактори, що можуть вплинути на розвиток проєкту. Результати SWOT-аналізу стануть основою для прийняття обґрунтованих управлінських рішень та планування подальших дій.

Таблиця 5.12 – SWOT-аналіз стартап-проекту

Сильні сторони: Інноваційні технології на основі комп'ютерного зору та штучного інтелекту для точного відстеження і класифікації цілей. Висока гнучкість у застосуванні для військових і рятувальних операцій.	Слабкі сторони: Високі вимоги до безпеки даних і сумісності з іншими пристроями, що потребує додаткових ресурсів для інтеграції та захисту. Можливі технічні збої та складність у використанні системи без підготовки.
Можливості: Зростання попиту на робототехнічні рішення для моніторингу та стеження. Впровадження нових технологій для оптимізації та зниження витрат на виробництво.	Загрози: Ризики, пов'язані з витоком даних через ненадійні з'єднання або кібератаки

SWOT-аналіз показує, що стартап-проект має високий потенціал для успіху завдяки інноваційності та гнучкості, проте потребує уваги до безпеки та готовності адаптуватися до конкурентного тиску.

Було проаналізовано альтернативи ринкового впровадження системи та наведено їх у таблиці 5.13.

Таблиця 5.13 – Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Співпраця з великими партнерами для спільного виходу на ринок	Висока	Середні
2	Самостійний вихід на ринок із залученням інвесторів	Середня	Довгі

№ п/п	Альтернатива ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
3	Вихід через участь у державних програмах підтримки стартапів	Висока	Короткі

З обраних альтернатив доцільно вибрати третю, оскільки отримання ресурсів є більш простим і ймовірним, а строки реалізації – більш стислими

5.4 Розроблення ринкової стратегії проекту

Першим кроком розроблення ринкової стратегії стартап–проекту є аналіз цільових груп потенційних споживачів, що наведено у таблиці 5.14. Детальний аналіз цих груп дозволяє визначити специфічні потреби, інтенсивність конкуренції в сегменті та простота входу у сегмент[34]. У таблиці 5.14 представлено сегментацію споживачів за різними критеріями, такими як готовність споживачів сприйняти продукт, орієнтовний попит в межах цільової групи, психологічні фактори та соціально-економічний статус.

Цей аналіз є критичним для розуміння того, як найкраще позиціонувати продукт або послугу на ринку. Визначення цільових аудиторій допомагає спрямувати маркетингові зусилля більш ефективно, адаптуючи стратегії під конкретні сегменти ринку. Це дозволяє не лише задовольнити потреби клієнтів, але й створити конкурентні переваги, забезпечуючи успішний вихід стартапу на ринок.

Таблиця 5.14 – Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Виробники мілітарних та пошукових мобільних роботів	Висока	Середній	Низька	Висока
2	Виробники охоронних систем	Середня	Високий	Середня	Середня
3	Виробники статичних систем спостережен ня	Низька	Низький	Висока	Низька
Які цільові групи обрано: Виробники мілітарних та пошукових мобільних роботів, оскільки вони мають високу готовність сприйняти продукт і простий вхід у сегмент, що підвищує шанси на успіх					

Було розроблено базову стратегію розвитку, яка відображає ключові напрямки розвитку системи та орієнтири для подальшого охоплення ринку. Результати роботи над стратегією представлені у вигляді таблиці 5.15.

Таблиця 5.15 – Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні і позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1	Співпраця з великими партнерами для виходу на ринок	Широке охоплення через партнерство	Сильна мережа збуту, швидкий доступ до споживачів	Стратегія розширення первинного попиту

Визначено базову стратегію конкурентної поведінки стартап-проекту, таблиця 5.16.

Таблиця 5.16 – Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1	Так	Компанія планує залучати нових споживачів, орієнтуючись на незайняті сегменти	Ні, компанія не планує копіювати, а фокусуватиметься на інноваційних характеристиках	Стратегія лідера

На основі даних з таблиць 5.5, 5.15 та 5.16, було розроблено стратегію позиціонування. Це є важливим кроком, бо саме ця стратегія буде створювати торговельну марку, за якою споживачі будуть ідентифікувати систему.

Таблиця 5.17 – Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап–проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Висока точність, висока надійність, простота використання	Стратегія розширення первинного попиту	Інноваційність, гнучкість, зручність інтеграції	Технологічність, ефективність, безпека

5.5 Розроблення маркетингової програми стартап–проекту

Першим кроком розроблення маркетингової програми є визначення ключових переваг, які наведено в таблиці 5.18.

Таблиця 5.18 – Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами
1	Надійність системи стеження та точність	Висока надійність і точність у відстеженні цілей	Використання інноваційних технологій для точного моніторингу, що забезпечує перевагу у якості
2	Гнучкість і простота інтеграції в різні системи	Легкість інтеграції та адаптації під різні умови	Модульна конструкція, що дозволяє адаптувати продукт для різних сфер використання, випереджаючи конкурентів

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами
3	Висока ефективність і безпека	Забезпечення безпеки даних і надійність у роботі	Захист даних відповідно до високих стандартів, що підвищує довіру та лояльність споживачів

Таблиця 5.19 представляє трирівневу маркетингову модель стартап-проєкту, яка структуровано описує різні аспекти продукту та його цінність для користувача. Ця модель враховує ключові аспекти стратегічного планування маркетингової діяльності, що дозволяє забезпечити ефективне просування продукту та досягнення цільових аудиторій.

Таблиця 5.19 – Опис трьох рівнів моделі стартап-проєкту

Рівні товару	Сутність та складові		
I. Товар за задумом	Задоволення потреби у надійності та точності системи стеження, основна функціональна вигода – точний і безпечний моніторинг цілей.		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	1. Надійність	М	Тх
	2. Точність	М	Тх
	3. Модульність	М	Тх
	4. Легкість інтеграції	М	Тх
	Якість: відповідність стандартам безпеки, надійності, і сертифікація		
	Пакування: надійна упаковка для безпечного транспортування		
	Марка: SafeguardPro		
III. Товар із підкріпленням	До продажу: підтримка клієнтів під час вибору та консультації щодо інтеграції		

Рівні товару	Сутність та складові
	Після продажу: сервісна підтримка, оновлення програмного забезпечення, навчання персоналу
Захист від копіювання буде забезпечено через поєднання унікальних характеристик, високої якості виконання та після продажно́ї підтримки, що включає оновлення і сервісні послуги	

У таблицях 5.20 – 5.22 буде визначено межі встановлення ціни, сформовано систему збуту та визначено концепцію маркетингової комунікації.

Таблиця 5.20 – Визначення меж встановлення ціни

№ п/п	Рівень цін на товари– замінники	Рівень цін на товари– аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	Товари замінники відсутні	\$1000 – \$10000	Високий (\$2000+ місячний дохід)	Верхня межа: \$950; Нижня межа: \$600

Таблиця 5.21 – Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Виробники з потребою в надійних та високоточних системах	Забезпечення технічної підтримки, навчання користувачів	Прямий канал	Власна система збуту через прямий продаж
2	Клієнти, які потребують адаптивних рішень з	Інформаційна підтримка,	Короткий канал	Залучена система збуту через

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
	гнучкими характеристиками	консультації щодо інтеграції		авторизованих дистриб'юторів
3	Клієнти, орієнтовані на швидкий сервіс та післяпродажну підтримку	Післяпродажне обслуговування, оновлення обладнання	Короткий канал	Змішана система збуту: власний збут із підтримкою через партнерів

Таблиця 5.22 – Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Потреба у надійності, точності, простоті використання	Спеціалізовані онлайн-платформи, галузеві виставки	Надійність, інноваційність, простота інтеграції	Ознайомити цільову аудиторію з перевагами продукту його унікальними характеристиками	Підкреслити надійність і технологічність

№ п/п	Специфіка поведінки цільових клієнтів	Канали кому нікацій, якими корис туються ціль ові клієнти	Ключові позиції, обрані для позиціонуван ня	Завдання рекламного повідомлення	Концепція рекламного звернення
2	Потреба в оперативній підтримці та гнучкості у використан ні	Соціальні мережі, професійні форуми, вебінари	Гнучкість, зручність, швидка підтримка	Виявити конкурентні переваги та забезпечити легкість доступу до інформації та підтримки	Показати зручність і доступність продукту
	Орієнтація на довгострок ову співпрацю та сервіс	Прямі контакти, е- mail маркетинг, галузеві семінари	Довгостроко ве обслуговуван ня, технічна підтримка	Підкреслити до вгостроко у кор исть і технічну підтримку, яку пропонує проду кт	Акцент на підтримці та сервісі

Висновки до розділу

Проект має високу ймовірність комерційного успіху на ринку завдяки існуючому попиту на технологічні рішення для стеження та моніторингу. Динаміка ринку вказує на зростаючий інтерес до таких продуктів у різних галузях, включаючи військову, рятувальну та охоронну, що підкріплює рентабельність проекту. Здатність запропонувати надійний, точний та адаптивний продукт дозволяє проекту задовольняти широкий спектр потреб клієнтів і виділятися серед конкурентів.

Впровадження проекту є перспективним завдяки зацікавленості цільової аудиторії, яка включає виробників мілітарних і пошукових роботів, охоронних систем

і систем спостереження. Бар'єри входження на ринок є помірними, а ключовими конкурентами виступають менш інноваційні продукти, які поступаються проекту за характеристиками. Конкурентоспроможність проекту забезпечується його унікальними технічними властивостями, що відповідають високим вимогам ринку щодо надійності та точності.

Оптимальним варіантом ринкової реалізації є співпраця з великими партнерами для спільного виходу на ринок, що дозволяє швидше масштабувати продукт, отримати доступ до розгалужених каналів збуту та знизити витрати на маркетинг. Така стратегія сприяє швидкому досягненню цільової аудиторії, забезпеченню технічної підтримки клієнтів та зміцненню позицій проекту на ринку.

Зважаючи на аналіз ринкових перспектив, конкуренцію та обрану стратегію реалізації, подальша імплементація проекту є виправданою. Проект має реальні шанси на успіх завдяки унікальним характеристикам, орієнтованості на потреби клієнтів та наявності стратегії, що дозволяє швидко зайняти конкурентну позицію. Розробка плану підтримки та післяпродажного обслуговування сприятиме підвищенню лояльності клієнтів і забезпечить довгострокову стабільність проекту.

ВИСНОВОК

У межах магістерської роботи розроблена система стеження за повітряними та наземними цілями мобільного робота на основі комп'ютерного зору. Основні результати процесу розробки узагальнені нижче.

В результаті аналізу предметної області проаналізовано сучасний стан технологій стеження та алгоритмів комп'ютерного зору. Детально вивчено приклади використання дронів DJI та FLIR Black Hornet, а також алгоритмів Kernelized Correlation Filters і SLAM, які є основою сучасних систем стеження. Виявлено ключові проблеми, зокрема необхідність підвищення точності, адаптивності до змін середовища та зменшення впливу людського фактора. На основі цього сформульовано мету і завдання дослідження.

На основі аналізу сформовано функціональні та нефункціональні вимоги до системи, які враховують сценарії використання. Зокрема, визначено такі ролі користувачів, як Адміністратор і Пілот мобільного робота, розроблено вимоги до модулів стеження, включаючи можливість налаштування алгоритмів і палітр кольорів для ідентифікації об'єктів. Це забезпечило основу для створення гнучкої та масштабованої системи.

Було проведено аналіз технологій, серед яких було обрано мови програмування Python та Lua, бібліотеки OpenCV та PyQt6, SQLite для бази даних, і модель комп'ютерного зору YOLO для виявлення об'єктів. Розроблено структурну схему системи, включаючи базу даних для зберігання конфігурацій і логів. Реалізовано інтерфейс користувача, який дозволяє легко налаштовувати параметри системи та отримувати результати аналізу.

У ході тестування системи виконано мануальне тестування системи, включаючи перевірку коректності виконання основних функцій, таких як автентифікація, стеження за цілями, обробка відеоданих та оцінка відстаней до об'єктів. Система успішно показала високу точність стеження навіть у складних динамічних умовах. Тестування підтвердило, що система відповідає функціональним і нефункціональним вимогам.

У рамках стартап-проєкту проаналізовано перспективи комерціалізації розробленої системи в рамках стартап-проєкту. Виконано SWOT-аналіз, який виявив сильні сторони, як-от гнучкість і точність системи, а також розроблено маркетингову стратегію, що включає позиціонування продукту для промислових, медичних і військових застосувань.

Загалом, розроблена система повністю досягла поставленої мети. Її впровадження дозволяє значно підвищити ефективність автономних робіт у стеженні за цілями, забезпечуючи точність і надійність у виконанні завдань. Отримані результати свідчать про великий потенціал для подальшого розвитку автономних систем із використанням сучасних технологій комп'ютерного зору.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. DJI. URL: <https://www.dji.com>
2. TELEDYNE FLIR. URL: <https://www.flir.com/>
3. KEENON. URL: <https://www.keenon.com/en/>
4. S. Yadav and S. Payandeh, «Understanding Tracking Methodology of Kernelized Correlation Filter,» 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 2018, pp. 1330-1336. URL: <https://ieeexplore.ieee.org/document/8614990>
5. Функціональні вимоги. URL: <https://visuresolutions.com/uk/%D0%B1%D0%BB%D0%BE%D0%B3/%D1%84%D1%83%D0%BD%D0%BA%D1%86%D1%96%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%96-%D0%B2%D0%B8%D0%BC%D0%BE%D0%B3%D0%B8/>
6. Аутентифікація і авторизація: що це і в чому відмінність. URL: <https://qagroup.com.ua/publications/autentyfikatciia-i-avtoryzatciia/>
7. What Is an Edge Case? URL: <https://www.coursera.org/articles/edge-case>
8. Функціональні та нефункціональні вимоги. URL: <https://www.guru99.com/uk/functional-vs-non-functional-requirements.html#what-is-nonfunctional-requirement>
9. Основні операції із зображеннями. URL: <https://shorturl.at/dWKYI>
10. Jaya P and Contour Based Object Tracking (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (3) , 2014, 4128-4130 URL: <https://www.ijcsit.com/docs/Volume%205/vol5issue03/ijcsit20140503322.pdf>
11. M. Michalko, J. Onuška, A. Lavrín, D. Cymbalák and O. Kainz, «Tracking the object features in video based on OpenCV,» 2016 International Conference on Emerging eLearning Technologies and Applications (ICETA), Stary Smokovec, Slovakia, 2016, pp. 223-226. URL: <https://ieeexplore.ieee.org/document/7802081>
12. Python OpenCV | cv2.cvtColor() method. URL: <https://www.geeksforgeeks.org/python-opencv-cv2-cvtColor-method/>

13. Find and Draw Contours using OpenCV | Python. URL: <https://www.geeksforgeeks.org/find-and-draw-contours-using-opencv-python/>
14. CoppeliaSim User Manual. URL: <https://manual.coppeliarobotics.com/>
15. E. Rohmer, S. P. N. Singh and M. Freese, «V-REP: A versatile and scalable robot simulation framework,» 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 2013. URL: <https://ieeexplore.ieee.org/document/6696520>
16. OpenCV: Open Source Computer Vision Library. URL: <https://github.com/opencv/opencv>
17. Ultralytics YOLO Docs. URL: <https://docs.ultralytics.com/>
18. Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, Travis E. Oliphant, «Array Programming with NumPy». URL: <https://arxiv.org/abs/2006.10256>
19. Coppeliarobotics Remote API. URL: <https://manual.coppeliarobotics.com/en/remoteApiOverview.htm>
20. The complete PyQt6 tutorial — Create GUI applications with Python. URL: <https://www.pythonguis.com/pyqt6-tutorial/>
21. PyQt6 Reference Guide. URL: <https://www.riverbankcomputing.com/static/Docs/PyQt6/>
22. Relational database. URL: https://en.wikipedia.org/wiki/Relational_database
23. NoSQL. URL: <https://en.wikipedia.org/wiki/NoSQL>
24. Графова база даних. URL: <https://shorturl.at/bFEib>
25. What Is SQLite?. URL: <https://www.sqlite.org/>
26. Top 7 Cloud Databases. URL: <https://www.dbmaestro.com/blog/database-automation/top-7-cloud-databases>

27. Choosing the Right Classification Model. URL:

<https://www.mathworks.com/campaigns/offers/next/choosing-the-best-machine-learning-classification-model-and-avoiding-overfitting.html>

28. Ultralytics YOLO11. URL: <https://docs.ultralytics.com/models/yolo11/>

29. PyCharm. URL: <https://www.jetbrains.com/pycharm/>

30. SQLite Studio User Manual. URL:

https://github.com/pawelsalawa/sqlitestudio/wiki/User_Manual

31. PyQt6 6.7.1. URL: <https://pypi.org/project/PyQt6/>

32. Ручне та автоматизоване тестування. URL: <https://qalight.ua/baza-znaniy/ruchne-ta-avtomatizovane-testuvannya/>

33. SWOT-аналіз. URL: <https://uk.wikipedia.org/wiki/SWOT-%D0%B0%D0%BD%D0%B0%D0%BB%D1%96%D0%B7>

34. ОСОБЛИВОСТІ ПЛАНУВАННЯ РИНКОВОЇ СТРАТЕГІЇ ФІРМИ. URL:

<https://lib.chmnu.edu.ua/pdf/posibnuku/329/9.pdf>