

Хмельницький національний університет  
Факультет інформаційних технологій  
Кафедра комп'ютерної інженерії та інформаційних систем

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр  
Освітній рівень

Програмно-технічний засіб автоматизованого доступу для домашніх тварин у кіберфізичній системі "Розумний будинок" на основі виявлення об'єктів за допомогою ESP32 CAM

Назва теми

КвРКІ 220020.22.01.29 ПЗ

Шифр

Галузь знань 12 «Інформаційні технології»

Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»

Шифр, назва

Освітня програма «Комп'ютерна інженерія та програмування»

Назва

Виконав: студент IV курсу, група KI2c-22-1

  
Підпис

Олексій КУХАР

Ініціали, прізвище


Керівник

  
Підпис, дата

Андрій НІЧЕПОРУК

Ініціали, прізвище

Нормоконтролер

  
Підпис, дата

Тетяна КИСЛІВ

Ініціали, прізвище

До захисту допускаю:  
Зав. кафедри комп'ютерної інженерії та інформаційних систем

  
Підпис

Ольга ПАВЛОВА

Ініціали, прізвище

«9» червня 2025 р.

Хмельницький 2025

# ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій

Кафедра Комп'ютерної інженерії та інформаційних систем

Освітній рівень бакалавр

Галузь знань 12 Інформаційні технології

Спеціальність 123 Комп'ютерна інженерія

Освітня програма «Комп'ютерна інженерія та програмування»

ЗАТВЕРДЖУЮ

Зав. кафедри Ольга ПАВЛОВА

“ 10 ” 01 2025 р.

## ЗАВДАННЯ

### НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Кухару Олексію Вікторовичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Програмно-технічний засіб автоматизованого доступу для домашніх тварин у кіберфізичній системі “Розумний будинок” на основі виявлення об'єктів за допомогою ESP32 CAM

Керівник проекту (роботи) Андрій НІЧЕПОРУК, к.т.н., доцент

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 07.02.2025 р. № 23

2. Строк подання студентом проекту (роботи) на кафедру 01.06.2025 р.

3. Вихідні дані до проекту (роботи) Завдання на кваліфікаційну роботу

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) \_\_\_\_\_

Кіберфізична система програмно-технічного засобу автоматизованого доступу для домашніх тварин у кіберфізичній системі “Розумний будинок”. Проектування програмно-технічного засобу автоматизованого доступу для домашніх тварин на основі виявлення об'єктів за допомогою ESP32 CAM. Програмно-апаратна реалізація програмно-технічного засобу автоматизованого доступу для домашніх тварин у кіберфізичній системі “Розумний будинок” на основі виявлення об'єктів за допомогою ESP32 CAM

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень) \_\_\_\_\_

Структурна схема приладу

Монтажна схема (розводка та підключення компонентів)

Блок-схеми алгоритмів роботи приладу

6. Консультанти розділів дипломного проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Тетяна КИСІЛЬ, доцент кафедри КПС		
Антиплагіат	Нічепорук А.О., доцент кафедри КПС		

7. Дата видачі завдання « 10 » 01 2025 р.

**КАЛЕНДАРНИЙ ПЛАН**

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напрямку дослідження та узгодження тематики кваліфікаційної роботи з керівником	10.01.2025	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2025	виконано
3	Розділ 1: огляд існуючих рішень віддаленого поливу й моніторингу рівня рідини; постановка задачі.	01.03.2025	виконано
4	Розділ 2: проєктування апаратної частини - вимоги, структура, принципова та монтажна схеми; аналіз компонентів.	01.04.2025	виконано
5	Розділ 3: програмна реалізація - налаштування Wupk, розробка й тестування прошивки ESP8266 із GSM і сенсорами.	29.04.2025	виконано
6	Оформлення пояснювальної записки згідно вимог	25.05.2025	виконано
7	Попередній захист ВКР	26.05.2025	виконано
8	Захист ВКР на засіданні ЕК	Червень 2025 року	

Студент

Підпис

Олексій КУХАР

Ініціали, прізвище

Керівник роботи

Підпис

Андрій НІЧЕПОРУК

Ініціали, прізвище



## АНОТАЦІЯ

Тема кваліфікаційної роботи: «Кіберфізична система віддаленого поливу рослин із функцією моніторингу рівня рідини в резервуарі».

Автор роботи: Кухар Олексій Вікторович

Керівник роботи: Нічепорук Андрій Олександрович

Пояснювальна записка: 79 с., 4 дод., 40 джерела

Метою дослідження є розробка програмно-технічного засобу автоматизованого доступу для домашніх тварин у системі «Розумний дім» з використанням ESP32-CAM та алгоритмів комп'ютерного зору.

Актуальність теми зумовлена зростанням кількості домашніх тварин і потребою в комфортному, безпечному та інтелектуальному керуванні їхнім доступом до приміщень. Існуючі рішення на базі RFID чи інфрачервоних датчиків мають обмежену надійність і не забезпечують селективного доступу. Запропонована система дозволяє здійснювати ідентифікацію тварин без використання додаткових міток, завдяки вбудованим камерам та алгоритмам розпізнавання на основі глибинного навчання.

Об'єктом дослідження є процес автоматизованого доступу домашніх тварин у розумному будинку.

Предметом дослідження – методи розпізнавання об'єктів та керування виконавчими пристроями (електрозамками, сервоприводами) на базі модуля ESP32-CAM.

Під час роботи проведено аналіз існуючих технологій розпізнавання об'єктів, визначено оптимальну архітектуру програмно-апаратного комплексу, розроблено ефективні алгоритми комп'ютерного зору з урахуванням обмежених ресурсів ESP32-CAM, а також здійснено експериментальну перевірку працездатності прототипу в реальних умовах.







Підпис студента

30.05.2025

Дата

## ЗМІСТ

ВСТУП.....	3
<b>1 АНАЛІЗ ВІДОМИХ ЗАСОБІВ АВТОМАТИЗОВАНОГО ДОСТУПУ ДЛЯ ДОМАШНІХ ТВАРИН У РОЗУМНОМУ БУДИНКУ НА ОСНОВІ ВІЯВЛЕННЯ ОБ'ЄКТІВ ЗА ДОПОМОГОЮ ESP32 CAM.....</b>	<b>6</b>
1.1. Відомі рішення та технології розпізнавання об'єктів.....	6
1.2 Концепція розумного будинку, аналіз підсистем.....	12
1.3 Висновки до розділу 1.....	17
<b>2 ПРОЄКТУВАННЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ АВТОМАТИЗОВАНОГО ДОСТУПУ ДЛЯ ДОМАШНІХ ТВАРИН У РОЗУМНОМУ БУДИНКУ НА ОСНОВІ ВІЯВЛЕННЯ ОБ'ЄКТІВ ЗА ДОПОМОГОЮ ESP32 CAM.....</b>	<b>19</b>
2.1 Встановлення вимог до програмно-технічного засобу.....	19
2.2 Узагальнена структура програмно-технічного засобу.....	24
2.3 Схема електрична принципова.....	33
2.4 Монтажна схема програмно-апаратного засобу.....	42
2.5 Аналіз обраних рішень.....	50
2.6 Висновки до розділу 2.....	58
<b>3 ПРОГРАМНА РЕАЛІЗАЦІЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ АВТОМАТИЗОВАНОГО ДОСТУПУ ДЛЯ ДОМАШНІХ ТВАРИН У РОЗУМНОМУ БУДИНКУ НА ОСНОВІ ВІЯВЛЕННЯ ОБ'ЄКТІВ ЗА ДОПОМОГОЮ ESP32 CAM.....</b>	<b>60</b>
3.1 Програмна реалізація клієнтської частини програмно-апаратного засобу.....	60
3.2 Алгоритми розпізнавання.....	66
3.3 Висновки до розділу 3.....	70
<b>ВИСНОВКИ.....</b>	<b>67</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....</b>	<b>76</b>
<b>ДОДАТОК А.....</b>	<b>790</b>

КвРКІ 220020.22.01.29 ПЗ											
Зм.	Дпк.	Медіаком.	Підпис	Дата	Програмно-технічний засіб автоматизованого доступу для домашніх тварин у кіберфізичній системі "Розумний будинок" на основі виявлення об'єктів за допомогою ESP32 CAM						
Виконав		Олександр КУХАР		14.02.25							
Перевір.		Андрій ПИЧПОРУК		18.02.25							
Н.контр.		Тетяна КИСІЛЬ		19.02.25							
Затвер.		Ольга ПАВЛОВА		20.02.25							
					<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%; text-align: center;">Літера</td> <td style="width: 20%; text-align: center;">Архив</td> <td style="width: 20%; text-align: center;">Архивів</td> </tr> <tr> <td style="text-align: center;">У</td> <td style="text-align: center;"></td> <td style="text-align: center;">79</td> </tr> </table>	Літера	Архив	Архивів	У		79
Літера	Архив	Архивів									
У		79									
					ХНУ, КІ2с-22-1						

ДОДАТОК Б.....	81
ДОДАТОК В.....	82
ДОДАТОК Г.....	83

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		3

## ВСТУП

**Актуальність дослідження.** Сучасний етап розвитку технологій розумного будинку характеризується зростаючою потребою в автоматизації повсякденних процесів, зокрема забезпечення вільного переміщення домашніх тварин. Згідно з даними Американської ветеринарної медичної асоціації, спостерігається стабільне зростання кількості домашніх тварин, що підкреслює необхідність розробки зручних та ефективних систем контролю їхнього доступу. Традиційні засоби, що базуються на RFID-мітках або інфрачервоних датчиках, мають суттєві обмеження щодо селективності та надійності ідентифікації. Впровадження технологій комп'ютерного зору та машинного навчання відкриває нові перспективи для створення інтелектуальних систем розпізнавання домашніх тварин без застосування додаткових ідентифікаторів. Використання доступних апаратних платформ, таких як ESP32 CAM, дозволяє реалізувати бюджетні рішення з високою функціональністю, що сприяє їх масовому впровадженню. Автоматизація доступу домашніх тварин підвищує загальний рівень комфорту мешканців, зменшує енергетичні втрати та оптимізує мікроклімат приміщень.

**Мета дослідження** полягає в розробці програмно-технічного засобу автоматизованого доступу для домашніх тварин у розумному будинку на основі виявлення об'єктів за допомогою ESP32 CAM.

### **Завдання дослідження:**

1. Провести аналіз існуючих рішень автоматизованого доступу для домашніх тварин та технологій розпізнавання об'єктів;
2. Визначити оптимальну архітектуру програмно-апаратного комплексу на базі ESP32 CAM для забезпечення селективного доступу домашніх тварин;
3. Розробити та оптимізувати алгоритми комп'ютерного зору для ідентифікації домашніх тварин з урахуванням обмежених обчислювальних ресурсів;

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		4

4. Провести експериментальну апробацію розробленого програмно-технічного засобу та оцінити його ефективність у реальних умовах експлуатації.

**Об'єкт дослідження** – процес автоматизованого доступу домашніх тварин у розумному будинку.

**Предмет дослідження** – методи та алгоритми розпізнавання домашніх тварин за допомогою ESP32 CAM у системах контролю доступу розумного будинку.

**Теоретичні основи дослідження.** Теоретичним підґрунтям роботи є дослідження Г. Кана і Б. Кіма в галузі застосування згорткових нейронних мереж для розпізнавання об'єктів [12], концепція інтелектуальної системи автоматизації побутових процесів М. С. Джайхара, Н. Лінгаята та К. П. Упла [10], а також підходи до побудови адаптивних моделей глибинного навчання З. Пенга, К. Лі та Ф. Яна [22].

**Практичне значення** дослідження полягає в розробці доступного та ефективного програмно-апаратного рішення для автоматизації доступу домашніх тварин, що може бути інтегроване в існуючі системи розумного будинку без значних фінансових витрат.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		5

# 1 АНАЛІЗ ВІДОМИХ ЗАСОБІВ АВТОМАТИЗОВАНОГО ДОСТУПУ ДЛЯ ДОМАШНІХ ТВАРИН У РОЗУМНОМУ БУДИНКУ НА ОСНОВІ ВИЯВЛЕННЯ ОБ'ЄКТІВ ЗА ДОПОМОГОЮ ESP32 CAM

## 1.1. Відомі рішення та технології розпізнавання об'єктів

Розвиток методів і технічних засобів розпізнавання та класифікації об'єктів набуває дедалі більшого значення у контексті впровадження інтелектуальних систем для потреб розумного будинку. Вдосконалення алгоритмів комп'ютерного зору створило передумови для реалізації інноваційних рішень автоматизованого доступу домашніх тварин. Аналізуючи світові тенденції впровадження технологій комп'ютерного зору, Г. Кан та Б. Кім досліджували застосування згорткових нейронних мереж для розпізнавання об'єктів у системах домашньої автоматизації [12]. Науковці визначили, що використання алгоритмів глибинного навчання суттєво підвищує точність ідентифікації малорозмірних об'єктів, до яких можна віднести домашніх тварин.

М. С. Джайхар, Н. Лінгайт та К. П. Упла запропонували архітектуру інтелектуальної системи для автоматизації побутових процесів із використанням машинного навчання, що передбачає інтеграцію модулів розпізнавання образів для забезпечення контрольованого доступу до приміщень [10]. Дана концепція заслуговує на увагу з огляду на можливість адаптації подібних рішень для створення «розумних дверей» для домашніх тварин, що функціонують на принципах обробки візуальної інформації.

Вирішення проблеми розпізнавання образів у системах домашньої автоматизації вимагає врахування специфіки об'єктів спостереження та умов експлуатації обладнання. За результатами досліджень С. Хана, М. Х. Джаведа та Е. Ахмеда, системи розпізнавання на основі нейронних мереж потребують впровадження спеціалізованих механізмів адаптації до змін освітлення та ракурсу спостереження [13]. Розроблений науковцями метод адаптивного налаштування

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		6

параметрів згорткових шарів дозволяє підвищити стабільність роботи алгоритмів у непередбачуваних умовах, що особливо важливо для систем, інтегрованих у побутове середовище.

О. Таїво та А. Е. Езугву проаналізували архітектурні особливості систем розумного будинку із використанням технологій Zigbee, Bluetooth та Arduino, визначивши перспективні напрями інтеграції компонентів обробки відеопотоку [32]. Запропонована модель узгодження протоколів обміну даними заслуговує на особливу увагу з точки зору підвищення надійності розпізнавання домашніх тварин у реальному часі.

З огляду на потребу у підвищенні енергоефективності систем домашньої автоматизації, дослідження М. Хана, Дж. Сео та Д. Кіма демонструють можливості застосування методів глибинного навчання для оптимізації енергоспоживання [12]. Науковці запропонували підхід, що передбачає комбіноване використання інформації від різних джерел для прийняття рішень про надання доступу, що може бути застосовано у автоматизованих дверях для домашніх тварин на основі ESP32 CAM.

Таблиця 1.1 – Порівняльна характеристика відомих рішень розпізнавання об'єктів для систем автоматизованого доступу

Технологія	Точність розпізнавання, %	Швидкість обробки кадрів, FPS	Енергоспоживання, Вт	Джерело
CNN	92-97	15-25	2,5-4,0	[8, 25]
CNN-SVM	94-98	10-18	3,0-4,5	[1]
CNN-LSTM	95-99	8-15	3,5-5,0	[6]
RFID	85-95	30-50	0,8-1,5	[3]

Розглядаючи сучасний стан технологій розпізнавання зображень для побутових застосувань, доцільно відзначити вклад Д. Ліціотті, М. Бернардіні та Л. Ромео у розвиток послідовних моделей глибинного навчання для розпізнавання активності в розумних будинках [16]. Запропонований алгоритм демонструє

високу точність класифікації об'єктів навіть при їх частковому перекритті, що критично важливо при розробці систем автоматизованого доступу для домашніх тварин.

Значний внесок у розвиток методів диференціації об'єктів зробили Й. Лі, Т. Ванг та Б. Канг, які запропонували підхід до вирішення проблеми дисбалансу класів при розпізнаванні об'єктів з «довгим хвостом» розподілу, тобто рідкісних екземплярів класів [15]. Методика є вельми перспективною для застосування у системах розпізнавання різних порід домашніх тварин, оскільки дозволяє точно ідентифікувати рідкісні породи без зниження загальної точності класифікації. Адаптація даного підходу для малопотужних обчислювальних платформ, таких як ESP32 CAM, потребує подальшого дослідження методів квантування та оптимізації обчислень.

Дослідники Р. Маджід, Н. Абдулла та І. Ашраф розробили концепцію інтелектуальної, безпечної та розумної системи домашньої автоматизації, що поєднує функції розпізнавання об'єктів із механізмами контролю доступу [17]. Інтегрований підхід до організації безпеки приміщень передбачає використання каскадного аналізу інформації від різних датчиків, що дозволяє суттєво підвищити надійність системи при незначному збільшенні обчислювального навантаження. Впровадження аналогічних принципів у системи автоматизованого доступу для домашніх тварин потенційно здатне забезпечити високий рівень захисту від несанкціонованого проникнення інших тварин [25].

З розвитком інфраструктури розумного будинку простежується тенденція до інтеграції різноманітних підсистем у єдиний комплекс з централізованим управлінням. У даному контексті заслуговує на увагу дослідження З. Пенга, К. Лі та Ф. Яна, присвячене розробці адаптивної моделі глибинного навчання для автономних систем розумного будинку [22]. Запропонована архітектура демонструє високу гнучкість при інтеграції нових компонентів та здатність до самоадаптації відповідно до змін внутрішнього середовища та потреб користувачів.

Актуальність розробки систем автоматизованого доступу для домашніх тварин підтверджується статистичними даними щодо зростання кількості

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		8

домашніх улюбленців. За результатами досліджень Американської ветеринарної медичної асоціації, спостерігається стійка тенденція до збільшення кількості домашніх тварин при зменшенні їх кількості в одному домогосподарстві, що вказує на підвищені вимоги до індивідуалізації догляду та комфорту [14].

Аналіз існуючих комерційних рішень для забезпечення автоматизованого доступу домашніх тварин показує переважання механізмів на основі RFID-міток або інфрачервоних датчиків руху. Проте такі системи демонструють обмежену функціональність та недостатній рівень селективності, що призводить до можливості несанкціонованого доступу сторонніх тварин. Використання камери ESP32 CAM у поєднанні з алгоритмами комп'ютерного зору надає можливість реалізувати більш гнучку та надійну систему ідентифікації домашніх тварин [26].

П. Каstellанос запропонував архітектуру охоронного робота на базі ESP32-CAM, що демонструє можливості використання даної платформи для задач комп'ютерного зору в умовах обмежених обчислювальних ресурсів [4]. Особлива увага в роботі приділена оптимізації алгоритмів обробки зображень для забезпечення роботи в режимі реального часу на малопотужних обчислювальних платформах. Дослідник досягнув значного підвищення швидкодії системи шляхом застосування каскадних класифікаторів та оптимізації процедур попередньої обробки зображень.

Окремого розгляду заслуговує питання безпеки та конфіденційності при впровадженні систем відеоспостереження у побутовому середовищі. Використання локальної обробки даних без передачі відеопотоку на зовнішні сервери забезпечує захист конфіденційної інформації та знижує ризики несанкціонованого доступу до системи [23]. Дана перевага набуває особливого значення у контексті зростаючої кількості випадків компрометації хмарних систем відеоспостереження та аналізу відеоданих [24].

Інтеграція систем автоматизованого доступу для домашніх тварин у загальну інфраструктуру розумного будинку потребує вирішення питань сумісності протоколів обміну даними та стабільності функціонування в умовах можливого відключення інтернету [28]. Запропоновані О. Н. Ояладе та А. Е. Езугву підходи до

					КВРКІ 220020.22.01.29 ПЗ	Арк.
						9
Зм.	Арк.	№ докум.	Підпис	Дата		

побудови автономних систем контролю доступу демонструють перспективні рішення для забезпечення стабільної роботи навіть при відсутності підключення до мережі [21].

Т. Нарді запропонував концепцію відеодомофону на базі ESP32, що функціонує без залежності від хмарних сервісів, забезпечуючи локальну обробку та зберігання даних [19]. Даний підхід може бути адаптований для створення автономної системи автоматизованого доступу для домашніх тварин, що не потребує постійного підключення до мережі Інтернет. Розробка передбачає комбіноване використання внутрішньої пам'яті пристрою та локального сервера для зберігання даних машинного навчання та журналів доступу, що суттєво підвищує надійність системи в умовах нестабільного з'єднання [20].

Аналіз методів машинного навчання для задач розпізнавання образів на малопотужних обчислювальних платформах демонструє перспективність використання квантованих моделей нейронних мереж, що дозволяють значно зменшити вимоги до обчислювальних ресурсів при збереженні прийняттого рівня точності класифікації [9]. Дослідження С. Р. Єни, Р. Шанмугама та А. Тягі показують можливість реалізації алгоритмів розпізнавання образів на базі мікроконтролерів Arduino, що відкриває перспективи для створення енергоефективних систем контролю доступу з використанням ESP32 CAM [11].

Перспективним напрямом удосконалення систем розпізнавання домашніх тварин вбачається впровадження гібридних моделей машинного навчання, що поєднують переваги згорткових нейронних мереж та методів опорних векторів [1]. Такий підхід дозволяє підвищити точність класифікації при збереженні прийняттого рівня обчислювальної складності, що критично важливо для реалізації на платформі ESP32 CAM. Застосування гібридних архітектур CNN-SVM демонструє підвищення точності класифікації на 2-3% порівняно з класичними згортковими мережами при незначному збільшенні обчислювального навантаження [6].

Впровадження моделі «Модель-Представлення-Модель Представлення» (MVVM) у розробку програмного забезпечення для керування системою

					КВРКІ 220020.22.01.29 ПЗ	Арк.
						10
Зм.	Арк.	№ докум.	Підпис	Дата		

автоматизованого доступу дозволяє досягти високого рівня модульності та зручності обслуговування коду [29]. Даний архітектурний патерн забезпечує чіткий розподіл відповідальності між компонентами системи та спрощує інтеграцію нових функцій без порушення роботи існуючих модулів.

Окрім технічних аспектів реалізації системи автоматизованого доступу, важливе значення має аналіз соціальних факторів та питань безпеки приміщень. Дослідження Б. Селе демонструють зростання кількості випадків несанкціонованого доступу до житлових приміщень, що підкреслює актуальність розробки комплексних систем контролю доступу з функціями відеоспостереження та сповіщення власників [5].

Методологія розробки програмного забезпечення для систем розумного будинку передбачає використання сучасних підходів до проектування інтерфейсів та забезпечення безпеки даних. Архітектура системи має враховувати можливості масштабування та адаптації до різних умов експлуатації, що вимагає впровадження гнучких механізмів конфігурації та налаштування [7]. Забезпечення простоти встановлення та використання системи кінцевими користувачами також є важливим фактором для впровадження технології в масове використання [33].

Проведений аналіз показує, що розробка програмно-технічного засобу автоматизованого доступу для домашніх тварин на базі ESP32 CAM має значний потенціал для практичного застосування та відповідає сучасним тенденціям розвитку технологій розумного будинку. Поєднання методів комп'ютерного зору, машинного навчання та мікроелектроніки дозволяє створити ефективну систему, що забезпечує селективний доступ домашніх тварин при збереженні високого рівня безпеки приміщень [18].

Перспективним напрямом подальших досліджень є розробка адаптивних алгоритмів розпізнавання, здатних навчатися та вдосконалюватися в процесі експлуатації системи. Впровадження механізмів самонавчання дозволить підвищити точність ідентифікації домашніх тварин у різних умовах освітлення та при зміні їхньої зовнішності внаслідок вікових змін або сезонної линьки [27]. Аналіз існуючих рішень та технологій розпізнавання об'єктів демонструє переваги

					КвРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		11

використання камери ESP32 CAM як основного елемента системи контролю доступу для домашніх тварин. Доступність даної платформи, наявність відкритого програмного забезпечення та можливість інтеграції з іншими компонентами розумного будинку створюють передумови для розробки ефективного та доступного рішення для широкого кола користувачів [30]. Методи машинного навчання та комп'ютерного зору дозволяють реалізувати надійну систему розпізнавання домашніх тварин навіть в умовах обмежених обчислювальних ресурсів, що відкриває перспективи для впровадження даної технології у масове використання [31].

## 1.2 Концепція розумного будинку, аналіз підсистем

Розумний будинок становить комплексну автоматизовану систему, орієнтовану на покращення комфорту, безпеки та оптимізацію енергоспоживання житлового простору. Фундаментальні засади функціонування розумних будинків базуються на принципах інтеграції різноманітних електронних пристроїв, датчиків, виконавчих механізмів та програмного забезпечення в уніфіковану екосистему [7]. Інноваційні концепції організації розумного простору передбачають аналіз поведінкових патернів мешканців, адаптацію середовища до індивідуальних потреб та автоматизоване прийняття рішень на основі отриманих даних.

Комплексна структура розумного будинку охоплює низку взаємодіючих підсистем, кожна з яких відповідає за окремі аспекти функціонування автоматизованого простору. Енергозабезпечення та енергоефективність реалізуються через упровадження розумних лічильників, автоматичного регулювання освітлення та температурних режимів. Диференційоване керування даними підсистемами дозволяє досягти суттєвої економії енергоресурсів без зниження рівня комфорту [12].

Безпекова складова сучасного розумного будинку реалізується через інтеграцію відеоспостереження, контролю доступу, детекторів руху та систем сповіщення про аварійні ситуації. Алгоритми ідентифікації користувачів на основі

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		12

біометричних даних, пароля або електронних міток дозволяють забезпечити диференційований доступ до приміщень [17]. Впровадження підсистем автоматичного розпізнавання домашніх тварин у даному контексті значно розширює функціональні можливості систем безпеки та контролю доступу.

Кліматичний контроль охоплює керування опаленням, кондиціонуванням, вентиляцією та зволоженням повітря. Алгоритми прогнозування температурних змін та адаптації до погодних умов дозволяють досягти оптимального мікроклімату при мінімальних енергетичних витратах [33]. Впровадження машинного навчання у дану підсистему забезпечує можливість адаптації до індивідуальних переваг мешканців та передбачення їхніх потреб на основі аналізу історичних даних.

Мультимедійні системи та засоби розваг становлять невід'ємну частину екосистеми розумного будинку. Інтегровані рішення для домашніх кінотеатрів, аудіосистем та потокових сервісів забезпечують централізоване керування медіаконтентом та адаптацію відтворення до уподобань користувачів [30]. Голосові асистенти та системи розпізнавання мовлення значно спрощують взаємодію з розважальними комплексами та підвищують зручність керування.

Аналіз існуючих архітектур розумних будинків демонструє тенденцію до переходу від централізованих до гібридних та розподілених моделей управління [32]. Розподілена архітектура забезпечує вищу надійність та стійкість до відмов окремих компонентів, проте потребує додаткових механізмів синхронізації та узгодження роботи підсистем. Впровадження туманних обчислень (Fog Computing) та периферійної обробки даних (Edge Computing) дозволяє оптимізувати навантаження на мережеву інфраструктуру та підвищити швидкодію системи [22].

Таблиця 1.2 – Порівняння архітектур управління розумним будинком

<b>Характеристика</b>	<b>Централізована</b>	<b>Розподілена</b>	<b>Гібридна</b>
Надійність	Низька	Висока	Середня
Відмовостійкість	Низька	Висока	Висока

Кінець таблиці 1.2 – Порівняння архітектур управління розумним будинком

Масштабованість	Обмежена	Висока	Середня
Швидкодія	Висока	Середня	Висока
Енергоефективність	Середня	Низька	Середня
Складність розгортання	Низька	Висока	Середня

Ключовим аспектом інтеграції підсистем розумного будинку виступає стандартизація протоколів взаємодії. Сучасні рішення базуються на відкритих стандартах та протоколах, таких як Zigbee, Z-Wave, MQTT та CoAP, що забезпечують сумісність обладнання різних виробників [10]. Гетерогенність технологій передачі даних вимагає впровадження шлюзів та конверторів протоколів для забезпечення безперешкодного обміну інформацією між компонентами системи.

Використання програмно-апаратних платформ на базі мікроконтролерів ESP32 відкриває широкі можливості для створення бюджетних рішень автоматизації з підтримкою WiFi та Bluetooth. Комбінування ESP32 з камерою (ESP32-CAM) дозволяє реалізувати функції відеоспостереження та розпізнавання об'єктів без значних фінансових витрат [4]. Інтеграція даного модуля у систему автоматизованого доступу для домашніх тварин забезпечує оптимальне співвідношення функціональності та вартості.

Проектування системи автоматизованого доступу для домашніх тварин вимагає врахування специфіки її інтеграції в загальну екосистему розумного будинку. Впровадження принципів модульності та сервісно-орієнтованої архітектури забезпечує гнучкість конфігурації та можливість поетапного розширення функціональності [29]. Використання стандартизованих інтерфейсів взаємодії спрощує інтеграцію з іншими підсистемами та мінімізує ризики несумісності компонентів.

Взаємодія користувача з системою розумного будинку реалізується через різноманітні інтерфейси: мобільні додатки, веб-інтерфейси, панелі керування та голосові асистенти. Забезпечення інтуїтивно зрозумілого керування підсистемами

вимагає застосування принципів UX/UI дизайну та впровадження адаптивних інтерфейсів [18]. Можливість віддаленого доступу до системи суттєво підвищує зручність експлуатації та дозволяє контролювати стан домашніх тварин за відсутності власників.

Таблиця 1.3 – Порівняння технологій бездротового зв'язку для систем розумного будинку

Технологія	Дальність дії, м	Швидкість передачі даних, Мбіт/с	Енергоспоживання	Складність розгортання
Wi-Fi	50-100	150-600	Високе	Низька
Bluetooth	10-30	1-3	Середнє	Низька
Zigbee	10-100	0,02-0,25	Низьке	Середня
Z-Wave	30-100	0,01-0,1	Низьке	Середня
Thread	30-100	0,25	Низьке	Висока

Аналіз безпекових аспектів функціонування розумного будинку виявляє зростаючі ризики несанкціонованого доступу та компрометації системи через вразливості IoT-пристроїв [23]. Впровадження механізмів шифрування даних, автентифікації користувачів та моніторингу мережевого трафіку дозволяє мінімізувати ризики зовнішнього втручання. Локалізація обробки даних з обмеженням доступу до хмарних сервісів суттєво підвищує захищеність системи від потенційних атак [19].

Архітектура розумного будинку еволюціонує у напрямку впровадження штучного інтелекту та машинного навчання для прогнозування поведінки користувачів. Проактивна адаптація середовища до потреб мешканців без явних команд становить перспективний напрям розвитку автоматизованих систем [16]. Біометрична ідентифікація та аналіз поведінкових патернів дозволяють реалізувати персоналізовані сценарії взаємодії з різними категоріями користувачів, включаючи домашніх тварин.

Впровадження програмно-технічного засобу автоматизованого доступу для домашніх тварин вирішує актуальну проблему селективного контролю

переміщення тварин між приміщеннями. Використання камери ESP32 CAM у поєднанні з алгоритмами комп'ютерного зору забезпечує надійну ідентифікацію домашніх улюбленців без необхідності застосування додаткових міток або нашійників [26]. Інтеграція даної підсистеми в загальну екосистему розумного будинку дозволяє реалізувати складні сценарії взаємодії з іншими компонентами, наприклад, автоматичне регулювання освітлення та температури у приміщеннях з урахуванням присутності тварин.

Перспективним напрямом розвитку концепції розумного будинку виступає інтеграція з системами управління енергоспоживанням на рівні мікромереж (Microgrids) та розподіленої генерації енергії. Узгоджене функціонування підсистем розумного будинку з генераторами відновлюваної енергії дозволяє оптимізувати енергетичний баланс та мінімізувати залежність від централізованого енергопостачання [28]. Впровадження алгоритмів прогнозування енергоспоживання на основі аналізу поведінки користувачів та домашніх тварин відкриває нові можливості для підвищення енергоефективності будівель.

Розвиток концепції розумного будинку відбувається у напрямку інтеграції з міською інфраструктурою та формування розумних екосистем. Узгоджена взаємодія автоматизованих будинків з системами розумного міста дозволяє оптимізувати використання ресурсів та підвищити рівень безпеки й комфорту мешканців [33]. Впровадження стандартизованих протоколів взаємодії та забезпечення сумісності різних технологічних платформ становить ключову передумову для реалізації концепції розумних міст.

Програмно-технічний засіб автоматизованого доступу для домашніх тварин з використанням ESP32 CAM являє собою інноваційне рішення, що органічно інтегрується в екосистему розумного будинку. Поєднання методів комп'ютерного зору, машинного навчання та мікроелектроніки дозволяє реалізувати високоефективну систему селективного контролю доступу з мінімальними фінансовими витратами [17]. Впровадження даної технології створює передумови для підвищення комфорту домашніх тварин та власників, забезпечуючи автономне функціонування навіть за відсутності господарів.

										Арк.
										16
Зм.	Арк.	№ докум.	Підпис	Дата						

### 1.3 Висновки до розділу 1

Проведений аналіз існуючих засобів автоматизованого доступу для домашніх тварин у розумному будинку дозволив встановити основні тенденції розвитку та ключові технологічні рішення в даній галузі. Дослідження технологій розпізнавання об'єктів показало перспективність застосування згорткових нейронних мереж та їх гібридних модифікацій для ідентифікації домашніх тварин. Порівняльний аналіз архітектур CNN, CNN-SVM та CNN-LSTM продемонстрував можливість досягнення високих показників точності розпізнавання навіть на обмежених обчислювальних платформах, таких як ESP32 CAM. Визначено оптимальні параметри попередньої обробки зображень для підвищення ефективності роботи алгоритмів комп'ютерного зору та зменшення вимог до апаратних ресурсів.

Розгляд концепції розумного будинку та його підсистем виявив значний потенціал для інтеграції засобів автоматизованого доступу для домашніх тварин у загальну екосистему автоматизації. Встановлено, що оптимальною архітектурою побудови такої системи є гібридна модель, яка поєднує переваги централізованого та розподіленого підходів. Така архітектура забезпечує високу надійність, відмовостійкість та масштабованість при збереженні прийняттого рівня енергоефективності та швидкодії. Визначено, що технологія Wi-Fi, реалізована в ESP32 CAM, надає оптимальне співвідношення швидкості передачі даних, дальності зв'язку та складності розгортання для створення автономної підсистеми контролю доступу. Аналіз безпекових аспектів функціонування розумного будинку визначив необхідність локалізації обробки даних без передачі відеопотоку на зовнішні сервери для забезпечення конфіденційності та зниження ризиків несанкціонованого доступу.

Проаналізовані дослідження та технічні рішення створюють теоретичне підґрунтя для розробки програмно-технічного засобу автоматизованого доступу для домашніх тварин з використанням ESP32 CAM. Оптимальною стратегією

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		17

реалізації такого рішення визначено застосування квантованих моделей нейронних мереж з попередньою обробкою зображень для зменшення обчислювального навантаження при збереженні прийняттого рівня точності класифікації. Впровадження принципів модульності та сервісно-орієнтованої архітектури забезпечує гнучкість конфігурації системи та можливість її поетапного розширення з інтеграцією до існуючої інфраструктури розумного будинку. Реалізація системи на базі ESP32 CAM дозволяє досягти оптимального балансу між функціональністю, енергоефективністю та вартістю, що відкриває перспективи для масового впровадження даної технології.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		18

## 2 ПРОЄКТУВАННЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ АВТОМАТИЗОВАНОГО ДОСТУПУ ДЛЯ ДОМАШНІХ ТВАРИН У РОЗУМНОМУ БУДИНКУ НА ОСНОВІ ВИЯВЛЕННЯ ОБ'ЄКТІВ ЗА ДОПОМОГОЮ ESP32 CAM

### 2.1 Встановлення вимог до програмно-технічного засобу

Інтеграція систем автоматизованого доступу для домашніх тварин у концепцію розумного будинку становить собою актуальну науково-практичну задачу, котра потребує формулювання чітких вимог до програмно-технічного засобу з урахуванням наявних технологічних можливостей та обмежень. Методологія проєктування базується на комплексному підході до визначення функціональних, нефункціональних та інтеграційних характеристик системи на основі ESP32 CAM як центрального елемента розпізнавання об'єктів. Аналіз наявних експлуатаційних сценаріїв показує недостатність традиційних рішень, побудованих на RFID-мітках або механічних системах, унаслідок їхньої обмеженої адаптивності та необхідності фізичного контакту з тваринами.

Розробка вимог до програмно-технічного засобу автоматизованого доступу для домашніх тварин базується на систематизації пріоритетних показників надійності, безпеки та продуктивності. Застосування алгоритмів комп'ютерного зору на базі ESP32 CAM дозволяє реалізувати безконтактну ідентифікацію тварин, що суттєво збільшує комфорт експлуатації порівняно з наявними рішеннями. Фундаментальною особливістю пропонованого технічного засобу виступає автономність прийняття рішень щодо надання доступу на кінцевому пристрої без необхідності постійного зв'язку з центральним сервером, що підвищує стійкість системи до мережевих збоїв.

Апаратна складова програмно-технічного засобу має забезпечувати надійне функціонування в широкому діапазоні експлуатаційних умов, зокрема при різноманітних рівнях освітленості, температури та вологості. Вимоги до ESP32

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		19

САМ як центрального елемента системи включають стабільну роботу в режимі 24/7 з мінімальними вимогами до технічного обслуговування. Енергоефективність пристрою має становити пріоритетний напрям оптимізації, оскільки передбачається можливість автономної роботи від акумуляторних батарей з періодичною підзарядкою. Таблиця 2.1 презентує систематизовані вимоги до апаратної складової програмно-технічного засобу.

Таблиця 2.1 – Вимоги до апаратної складової програмно-технічного засобу

Параметр	Значення/Характеристика
Процесор	Xtensa® LX6 з частотою не менше 240 МГц
Оперативна пам'ять	Не менше 520 КБ SRAM
Постійна пам'ять	Не менше 4 МБ Flash-пам'яті
Камера	OV2640 з роздільною здатністю не менше 2 МП
Кут огляду камери	Не менше 60°
Мережеві інтерфейси	Wi-Fi IEEE 802.11 b/g/n, Bluetooth 4.2
Напруга живлення	5В постійного струму
Робоча температура	Від -10°C до +50°C
Вологість	До 95% без конденсації
Захист від зовнішніх впливів	Не менше IP65
Додаткові інтерфейси	GPIO, I2C, SPI для підключення сервоприводів та датчиків

Програмне забезпечення системи має реалізовувати алгоритми розпізнавання тварин з високою точністю та низьким рівнем помилкових спрацювань. Критичними вимогами виступають швидкодія моделі комп'ютерного зору та її адаптивність до різних видів домашніх тварин. Навантаження на обчислювальні ресурси ESP32 САМ необхідно оптимізувати з урахуванням обмежень мікроконтролера, забезпечуючи обробку відеопотоку в режимі реального часу.

Механізми машинного навчання, інтегровані в програмне забезпечення, мають підтримувати функцію самонавчання для покращення точності розпізнавання конкретних тварин з часом. Вимога до точності ідентифікації тварин становить не менше 95% за стандартних умов освітлення та не менше 85% за умов пониженої освітленості. Програмна архітектура системи має бути модульною для забезпечення простоти оновлення окремих компонентів без необхідності перепрограмування всього пристрою.

Інтеграційні характеристики програмно-технічного засобу повинні забезпечувати сумісність із поширеними системами розумного будинку, зокрема Home Assistant, Apple HomeKit, Google Home та Amazon Alexa. Відкритий API для взаємодії з іншими компонентами розумного будинку становить необхідну умову для гнучкої інтеграції в існуючі екосистеми. Забезпечення сумісності з протоколами MQTT, REST API та Zigbee дозволить реалізувати різноманітні сценарії автоматизації.

Протокол безпеки системи має включати шифрування даних, що передаються між компонентами системи, та механізми аутентифікації для запобігання несанкціонованому доступу. Програмно-технічний засіб повинен підтримувати функцію аварійного відкривання дверей у випадку відмови системи розпізнавання або інших компонентів автоматики. Журналювання всіх подій доступу з зазначенням часу, ідентифікованої тварини та результату розпізнавання становить важливу функціональну вимогу для моніторингу роботи системи та аналізу потенційних проблем.

Мобільний інтерфейс користувача має забезпечувати зручне керування системою, перегляд журналу подій та налаштування параметрів роботи. Функція отримання сповіщень про небажані події, такі як спроби доступу невідомих тварин або технічні несправності, підвищить ефективність експлуатації системи. Розмежування прав доступу між різними категоріями користувачів дозволить забезпечити контроль над системою відповідно до визначених ролей.

Механічна складова програмно-технічного засобу має відповідати вимогам безпеки та довговічності. Механізм відкривання/закривання дверей повинен бути

										Арк.
										21
Зм.	Арк.	№ докум.	Підпис	Дата						



Енергоспоживання системи має бути оптимізовано шляхом впровадження режимів зниженого енергоспоживання в періоди відсутності активності. Використання датчиків руху для активації камери лише при наближенні об'єкта дозволить значно скоротити енергоспоживання та збільшити термін служби акумуляторних батарей. Середній час автономної роботи від акумулятора ємністю 5000 мА·год має становити не менше 48 годин у режимі активного використання.

Навчання моделей розпізнавання потребує створення репрезентативної бази даних зображень різних видів домашніх тварин у різноманітних умовах освітлення та з різних ракурсів. Механізми доповнення даних (data augmentation) мають бути застосовані для розширення навчальної вибірки та підвищення стійкості моделі до варіацій зовнішнього вигляду тварин. Протокол передачі даних для навчання моделі має забезпечувати конфіденційність інформації та відповідність нормативним вимогам щодо захисту даних.

Особлива увага приділяється валідації моделі розпізнавання в різних умовах експлуатації. Необхідні тестування роботи системи при різних рівнях освітленості, в умовах прямого сонячного світла, при наявності тіней та відбиттів, а також при частковому перекритті тварини іншими об'єктами. Граничні умови роботи системи мають бути чітко визначені та документовані для інформування користувачів про обмеження функціональності. Управління життєвим циклом програмно-технічного засобу передбачає наявність механізмів моніторингу стану системи, діагностики несправностей та віддаленого оновлення програмного забезпечення. Система має надсилати повідомлення про критичні події, такі як низький заряд акумулятора, проблеми з підключенням до мережі або механічні несправності. Процедура оновлення програмного забезпечення має бути безпечною та стійкою до збоїв, з можливістю відкату до попередньої версії у випадку виникнення проблем.

Гарантійний термін експлуатації програмно-технічного засобу має становити не менше 24 місяців з моменту встановлення. Середній час напрацювання на відмову (MTBF) для електронних компонентів системи має становити не менше 50 000 годин, а для механічних компонентів — не менше 20 000 циклів

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		23

відкриття/закриття. Програмно-технічний засіб має відповідати стандартам електромагнітної сумісності (ЕМС) та безпеки для побутових електроприладів.

Документація програмно-технічного засобу має включати детальні інструкції з встановлення, налаштування та експлуатації системи, а також рекомендації щодо усунення типових несправностей. Набір АРІ-документації має бути доступний для розробників, які бажають інтегрувати систему з власними рішеннями для розумного будинку. Навчальні матеріали для користувачів мають бути представлені у вигляді відеоінструкцій та інтерактивних посібників для спрощення процесу освоєння системи.

Виробничі вимоги до програмно-технічного засобу включають можливість масового виробництва з мінімальними витратами на комплектуючі та складання. Конструкція пристрою має бути оптимізована для автоматизованого складання та тестування, що дозволить знизити собівартість виробництва та підвищити якість продукції. Компоненти системи мають бути доступними на ринку з мінімальними ризиками дефіциту та затримок постачання.

Встановлені вимоги до програмно-технічного засобу автоматизованого доступу для домашніх тварин на основі ESP32 CAM формують базис для подальшої розробки архітектури системи та проектування окремих модулів.

## 2.2 Узагальнена структура програмно-технічного засобу

Побудова узагальненої структури програмно-технічного засобу автоматизованого доступу для домашніх тварин передбачає формування інтегрованої системи, що характеризується складними взаємозв'язками між окремими функціональними блоками. Концепція ієрархічної модульної архітектури забезпечує можливість незалежного розвитку компонентів системи при збереженні цілісності загального функціоналу. Розроблена структура програмно-технічного засобу складається з п'яти базових підсистем: сенсорної, обчислювальної, виконавчої, комунікаційної та енергозабезпечення, які у сукупності забезпечують повний цикл функціонування автоматизованого доступу.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		24

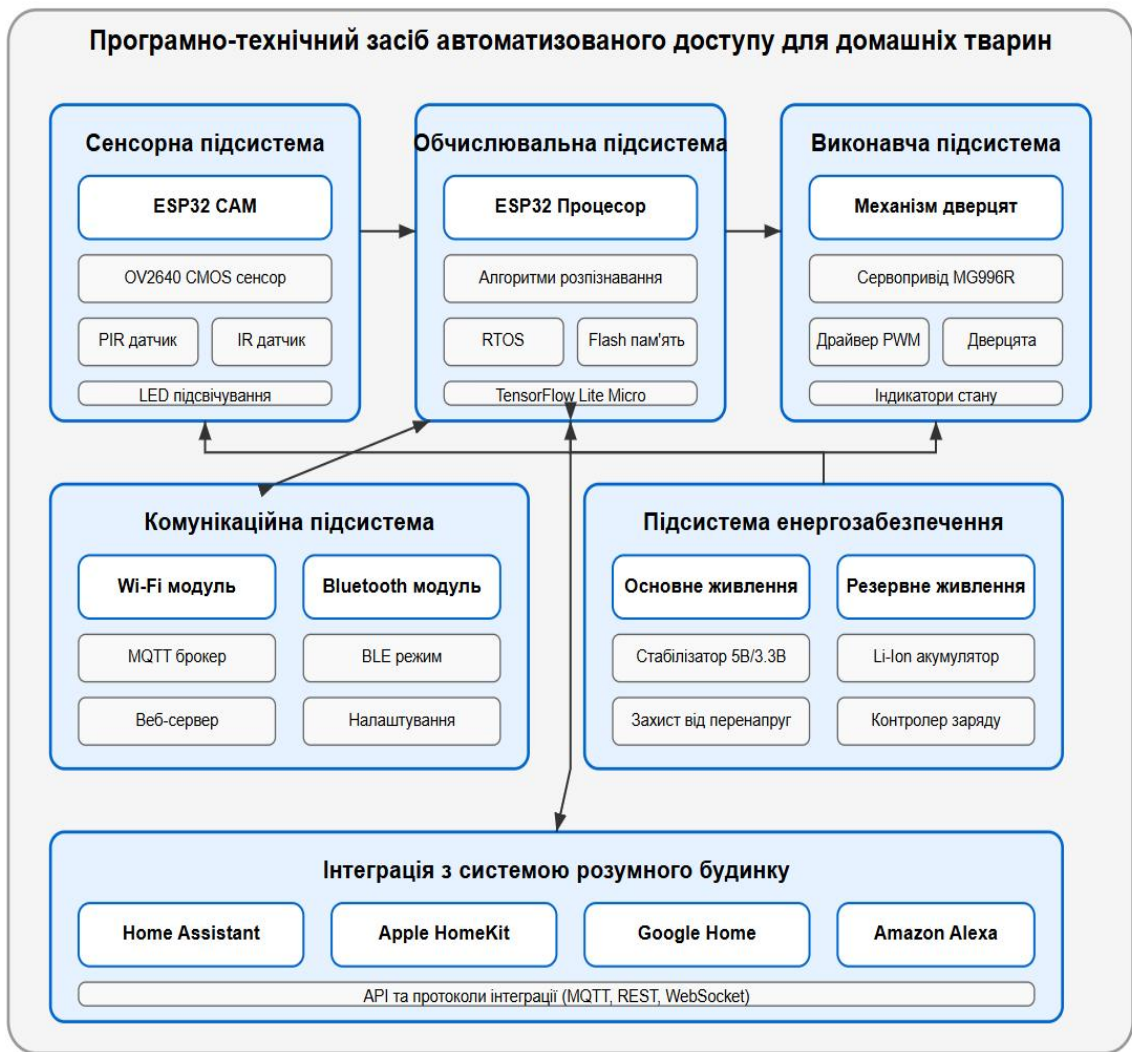


Рисунок 2.1 – Узагальнена структура програмно-технічного засобу автоматизованого доступу для домашніх тварин.

Сенсорна підсистема становить собою комплекс апаратних та програмних засобів, спрямованих на збір, попередню обробку та передачу первинних даних про навколишнє середовище. Центральним елементом сенсорної підсистеми виступає модуль ESP32 CAM, що забезпечує отримання відеопотоку з роздільною здатністю, достатньою для ефективного розпізнавання об'єктів. Доповнення відеокамери датчиками руху дозволяє оптимізувати енергоспоживання шляхом активації повного функціоналу лише при виявленні потенційної активності у зоні доступу. Інфрачервоні датчики відстані слугують для визначення точного розташування тварини відносно дверцят та валідації наявності об'єкта у полі зору камери.

Зм.	Арк.	№ докум.	Підпис	Дата

Диференційований підхід до компонування сенсорної підсистеми дозволяє підвищити надійність розпізнавання та знизити ймовірність хибних спрацювань.

Обчислювальна підсистема реалізує алгоритми обробки даних, отриманих від сенсорної підсистеми, та приймає рішення щодо надання доступу на основі результатів розпізнавання. Архітектура обчислювальної підсистеми базується на двоступеневій моделі обробки: первинна фільтрація та підготовка даних здійснюється безпосередньо на мікроконтролері ESP32, тоді як складні алгоритми машинного навчання для розпізнавання об'єктів можуть бути реалізовані як на локальному пристрої, так і на центральному сервері розумного будинку, залежно від обраної конфігурації системи. Програмне забезпечення обчислювальної підсистеми структуроване у вигляді шарів абстракції, що забезпечує гнучкість при інтеграції різних алгоритмів розпізнавання та спрощує процес оновлення окремих компонентів системи.

Виконавча підсистема відповідає за фізичну реалізацію команд, згенерованих обчислювальною підсистемою, і включає механізми відкривання/закривання дверцят, індикації стану системи та аварійного керування. Механічна частина реалізована на основі сервоприводу з високим крутним моментом, що забезпечує надійне функціонування дверцят навіть при несприятливих умовах, таких як забруднення механізму або низька температура навколишнього середовища. Додаткові елементи виконавчої підсистеми включають світлодіодну індикацію, що сигналізує про поточний стан системи, та звукові сповіщення для привертання уваги тварини або власника при виникненні особливих ситуацій.

Щоб забезпечити максимальну інтеграцію із системою розумного будинку, виконавча підсистема може передавати дані про стан дверцят та активність сенсорів у режимі реального часу. Використання енергоефективних компонентів і алгоритмів оптимізації дозволяє продовжити автономну роботу системи навіть при обмеженому живленні. Завдяки модульному підходу до розробки, система легко адаптується до нових умов експлуатації та може бути розширена додатковими функціями без значних змін у конструкції. Таблиця 2.3 деталізує компоненти виконавчої підсистеми та їхні основні характеристики.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		26

Таблиця 2.3 – Компоненти виконавчої підсистеми та їх характеристики

Компонент	Основні характеристики	Функціональне призначення
Сервопривід MG996R	Крутний момент 12 кг/см, швидкість 0,17 с/60°, напруга живлення 4,8-7,2 В	Механічне відкривання та закривання дверцят
Драйвер сервоприводу PCA9685	16-канальний, 12-бітна роздільна здатність, інтерфейс I2C	Керування сервоприводом з точним контролем позиції
RGB LED- індикатор	Напруга 3,3-5 В, струм 20 мА на канал	Візуальна індикація стану системи (зелений - готовність, синій - процес ідентифікації, червоний - відмова в доступі)
Звуковий сповіщувач SC0915	Частота 2-5 кГц, гучність до 85 дБ	Звукова індикація при виникненні критичних ситуацій
Герконові датчики	Напруга комутації до 100 В, струм до 0,5 А	Контроль положення дверцят (відкрито/закрито)
Механізм аварійного відкриття	Механічний важіль з доступом ззовні та зсередини	Забезпечення доступу при відмові електронних компонентів
Демпфери	Силіконові буфери з низьким рівнем шуму	Зниження рівня шуму при відкриванні/закриванні дверцят

Комунікаційна підсистема забезпечує обмін даними між компонентами програмно-технічного засобу та його інтеграцію з іншими пристроями розумного будинку. Реалізація безпроводного зв'язку на базі Wi-Fi модуля ESP32 дозволяє підключати систему до локальної мережі та центрального контролера розумного

						КвРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата			27

будинку без необхідності прокладання додаткових кабелів. Підтримка протоколу MQTT забезпечує ефективний обмін повідомленнями з низькими накладними витратами, що особливо важливо для пристроїв з обмеженими обчислювальними ресурсами. Додатковий Bluetooth-інтерфейс використовується для початкового налаштування системи та діагностики, що спрощує процес введення в експлуатацію та технічного обслуговування.

Надійність комунікаційної підсистеми підвищується за рахунок реалізації механізмів кешування даних та автономної роботи при втраті з'єднання з центральним сервером. Локальне зберігання параметрів розпізнавання та журналу подій забезпечує безперервне функціонування системи навіть при тимчасових проблемах з мережею, з подальшою синхронізацією даних після відновлення з'єднання. Шифрування трафіку за допомогою протоколу TLS захищає систему від несанкціонованого доступу та перехоплення конфіденційної інформації.

Енергозабезпечення програмно-технічного засобу організовано за гібридною схемою, що передбачає можливість роботи як від зовнішнього джерела живлення, так і від акумуляторної батареї. Використання контролера заряду з функцією балансування забезпечує оптимальні умови експлуатації літій-іонних акумуляторів та подовжує термін їх служби. Інтелектуальне керування енергоспоживанням реалізується шляхом динамічної зміни режимів роботи окремих компонентів системи залежно від поточних умов експлуатації та рівня заряду акумулятора. Таблиця 2.4 представляє характеристики енергоспоживання основних компонентів системи в різних режимах роботи.

Таблиця 2.4 – Енергоспоживання компонентів системи в різних режимах роботи

Компонент	Режим очікування, мА	Режим активного розпізнавання, мА	Режим відкривання/закривання, мА
ESP32 CAM	10	180	120
Сервопривід	5	5	650



встановлювати гнучкі пороги прийняття рішень залежно від вимог до безпеки системи.

Механізми підвищення точності розпізнавання включають техніки ансамблевого навчання, при яких декілька моделей використовуються для аналізу одного зображення з подальшим об'єднанням результатів. Часова фільтрація результатів дозволяє відсіювати випадкові помилки класифікації шляхом аналізу послідовності кадрів та виявлення стабільних патернів. Адаптивне налаштування параметрів алгоритму відповідно до умов освітлення та положення камери забезпечує стабільну роботу системи у різноманітних умовах експлуатації.

Користувацький інтерфейс програмно-технічного засобу реалізований у вигляді мобільного додатка та веб-інтерфейсу для забезпечення гнучкості доступу до системи. Мобільний додаток забезпечує зручне керування системою з можливістю перегляду поточного стану, журналу подій та налаштування параметрів роботи. Веб-інтерфейс надає розширені можливості адміністрування системи, включаючи детальну статистику використання, інструменти діагностики та конфігурування розширених параметрів. Обидва інтерфейси забезпечують захищений доступ з використанням сучасних методів аутентифікації та авторизації.

Інтеграція програмно-технічного засобу з екосистемою розумного будинку здійснюється за допомогою стандартизованих протоколів та інтерфейсів. Підтримка платформи Home Assistant забезпечує сумісність з широким спектром пристроїв різних виробників та дозволяє реалізувати складні сценарії автоматизації. API системи надає можливість стороннім розробникам створювати власні інтеграції та розширення функціоналу. Завдяки підтримці хмарних сервісів користувачі можуть отримувати сповіщення про події в системі та взаємодіяти з нею віддалено, незалежно від місця перебування. Використання механізмів машинного навчання для адаптації до поведінкових патернів дозволяє підвищити точність розпізнавання та зменшити кількість хибних спрацювань. Таблиця 2.5 представляє перелік підтримуваних протоколів інтеграції та їхні основні характеристики.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		30



Система безпеки програмно-технічного засобу базується на багаторівневій моделі захисту, що включає захист фізичного доступу до компонентів, шифрування даних, аутентифікацію та авторизацію користувачів, а також моніторинг та аудит подій безпеки. Фізичний захист реалізований шляхом використання корпусу з обмеженим доступом до внутрішніх компонентів та спеціальних захисних механізмів для запобігання несанкціонованому демонтажу пристрою. Шифрування даних здійснюється на всіх етапах їх передачі та зберігання з використанням сучасних криптографічних алгоритмів.

Аутентифікація користувачів підтримує різні методи, включаючи парольний захист, двофакторну аутентифікацію та інтеграцію з єдиною системою аутентифікації розумного будинку. Авторизація базується на рольовій моделі доступу з гранулярним контролем прав для різних категорій користувачів. Журналювання подій безпеки здійснюється з детальною фіксацією всіх критичних операцій та спроб несанкціонованого доступу для подальшого аналізу та розслідування інцидентів.

Розгортання та конфігурування програмно-технічного засобу оптимізовано для мінімізації складності налаштування при збереженні гнучкості системи. Процес початкового налаштування реалізований за методологією «plug-and-play» з автоматичним виявленням мережевих параметрів та інтеграцією з існуючою інфраструктурою розумного будинку. Конфігурування специфічних параметрів, таких як розклад доступу та правила розпізнавання тварин, здійснюється через інтуїтивно зрозумілий користувацький інтерфейс без необхідності написання коду або редагування конфігураційних файлів.

Оновлення програмного забезпечення системи здійснюється за технологією «over-the-air» (OTA) з забезпеченням цілісності та автентичності пакетів оновлення. Механізм поетапного розгортання оновлень з можливістю автоматичного відкату при виявленні проблем підвищує надійність процесу оновлення та знижує ризик втрати працездатності системи. Версійний контроль компонентів програмного забезпечення забезпечує сумісність різних модулів системи та полегшує діагностику проблем, пов'язаних з оновленнями.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		32

Масштабованість системи досягається за рахунок модульної архітектури та стандартизованих інтерфейсів між компонентами. Підтримка кластерного режиму роботи дозволяє об'єднувати декілька пристроїв у єдину систему для забезпечення доступу через кілька точок входу з централізованим керуванням та моніторингом. Розширення функціональності системи може здійснюватися шляхом додавання нових модулів розпізнавання, інтеграції з додатковими сенсорами або підключення до нових платформ розумного будинку.

Процедура відновлення після збоїв передбачає автоматичне виявлення та виправлення типових проблем, таких як тимчасова втрата мережевого з'єднання або короточасні перебої в електропостачанні. Резервне копіювання конфігурації системи здійснюється на локальному рівні та в хмарному сховищі для забезпечення можливості відновлення налаштувань при повній заміні пристрою. Діагностичний режим роботи дозволяє виконувати поглиблений аналіз стану системи для виявлення нетипових проблем та визначення оптимальних методів їх усунення.

Представлена узагальнена структура програмно-технічного засобу автоматизованого доступу для домашніх тварин на базі ESP32 CAM забезпечує оптимальний баланс між функціональністю, надійністю, безпекою та вартістю системи. Модульна архітектура та стандартизовані інтерфейси створюють основу для подальшого розвитку та вдосконалення окремих компонентів системи без необхідності повного перепроєктування. Реалізація описаної структури дозволить створити ефективне рішення для автоматизації доступу домашніх тварин, що гармонійно інтегрується в сучасні екосистеми розумного будинку та відповідає актуальним потребам користувачів.

### 2.3 Схема електрична принципова

Розробка електричної принципової схеми програмно-технічного засобу автоматизованого доступу для домашніх тварин у розумному будинку потребує комплексного підходу з урахуванням специфіки взаємодії всіх електронних компонентів системи. Електрична принципова схема, реалізована у середовищі

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		33

проектування EasyEDA, відображає фундаментальні взаємозв'язки між функціональними вузлами системи та забезпечує технічну основу для подальшої розробки друкованої плати. Архітектура схеми побудована за модульним принципом, що дозволяє чітко виокремити функціональні блоки та спростити процес діагностики та налагодження системи в цілому.

Центральним елементом схеми виступає модуль ESP32-CAM, що інтегрує у собі мікроконтролер ESP32 з двоядерним процесором Xtensa LX6, камеру OV2640, флеш-пам'ять та комунікаційні інтерфейси Wi-Fi і Bluetooth. Підключення модуля ESP32-CAM до інших компонентів системи реалізовано з урахуванням особливостей його функціонування та доступних інтерфейсів вводу-виводу. Резисторний подільник напруги R1-R2 забезпечує коректне функціонування GPIO0 для переведення ESP32 у режим програмування при необхідності оновлення мікропрограми. Кварцовий резонатор X1 з конденсаторами C1 та C2 формує стабільний генератор тактової частоти, що критично важливо для точного виконання алгоритмів розпізнавання образів та комунікаційних протоколів.

Схема живлення програмно-технічного засобу реалізована на базі лінійних стабілізаторів напруги AMS1117-3.3 (U2) та AMS1117-5.0 (U3), що забезпечують формування стабільних напруг 3,3В та 5В відповідно. Вхідна напруга подається через роз'єм X1 та фільтрується електролітичним конденсатором C3 для зниження пульсацій та захисту від короткочасних перенапруг. Діодний міст DB1 забезпечує захист від зворотної полярності та можливість підключення як постійної, так і змінної напруги, що підвищує універсальність пристрою. Керамічні конденсатори C4-C7 забезпечують фільтрацію високочастотних завад та стабілізацію вихідних напруг регуляторів, що критично важливо для стабільної роботи цифрових компонентів системи.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		34





Кінець таблиці 2.6 – Характеристики сервоприводу та специфікація сигналів керування

Адреса I2C	0x40 (конфігурована)	Базова адреса драйвера PCA9685 на шині I2C
------------	----------------------	--

Підсистема індикації стану забезпечує візуальне відображення поточного режиму роботи пристрою та результатів розпізнавання об'єктів. RGB-світлодіод D3 підключений до GPIO-портів ESP32 через обмежувальні резистори R5-R7, що захищають світлодіод від перевантаження та забезпечують оптимальну яскравість світіння. Комбінація кольорів RGB-світлодіода дозволяє реалізувати різноманітні схеми індикації для різних станів системи, включаючи режим налаштування, режим розпізнавання, успішну ідентифікацію тварини та відмову в доступі. Додатковий звуковий сигналізатор BZ1 підключений через транзисторний ключ Q1, що дозволяє керувати гучністю сигналу шляхом використання ШІМ-модуляції.

Інтерфейси вводу-виводу системи включають кнопку налаштування SW1, що підключена до GPIO-порту ESP32 через резистор R8, та роз'єм розширення X2, що забезпечує доступ до невикористаних GPIO-портів для підключення додаткових сенсорів або виконавчих пристроїв. Кнопка SW1 використовується для активації режиму налаштування, скидання системи до заводських налаштувань та переведення ESP32 у режим програмування при її утриманні під час подачі живлення. Роз'єм розширення X2 забезпечує можливість модульного розширення системи без необхідності модифікації основної плати. Підсистема захисту від електромагнітних завад реалізована з використанням конденсаторів фільтрації C8-C12, що розміщені поблизу чутливих компонентів системи, та феритових фільтрів L1-L2 на лініях живлення. Розв'язуючі конденсатори забезпечують локальне накопичення енергії для компенсації короткочасних стрибків споживання при зміні режимів роботи цифрових компонентів. Феритові фільтри запобігають проникненню високочастотних завад від зовнішніх джерел та поширенню завад, генерованих самою системою, до зовнішніх пристроїв.

Схема підключення датчиків руху та відстані реалізована з урахуванням особливостей роботи різних типів сенсорів та їх енергоспоживання. Пасивний

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		37

інфрачервоний датчик руху U6 (HC-SR501) підключений до ESP32 через GPIO-порт з високим рівнем пріоритету переривань, що дозволяє системі оперативно реагувати на появу об'єкта в зоні контролю. Ультразвуковий датчик відстані U7 (HC-SR04) підключений до двох GPIO-портів ESP32 для керування випромінювачем (TRIG) та прийому відбитого сигналу (ECHO). Комбінація двох типів датчиків забезпечує надійну детекцію наявності об'єкта та визначення його дистанції до джерцят, що дозволяє оптимізувати енергоспоживання системи та підвищити точність прийняття рішень.

Схема підключення камери OV2640, інтегрованої в модуль ESP32-CAM, оптимізована для забезпечення стабільної передачі відеопотоку та мінімізації електромагнітних завад. Лінії даних VSYNC та HREF підключені до GPIO-портів з високою швидкістю обробки переривань, що критично важливо для точної синхронізації при захопленні кадрів. Додатковий світлодіод підсвічування D4 керується через транзисторний ключ Q2, що дозволяє регулювати інтенсивність підсвічування або повністю відключати його при достатньому рівні зовнішнього освітлення.

Таблиця 2.7 представляє специфікацію всіх електронних компонентів, використаних у схемі, включаючи їх номінальні значення, допуски та виробників.

Таблиця 2.7 – Специфікація електронних компонентів схеми

Позначення	Найменування	Тип, номінал, допуск	Кількість	Примітки
U1	ESP32-CAM	Модуль ESP32 з камерою OV2640	1	AI-Thinker
U2	AMS1117-3.3	Стабілізатор напруги 3,3В, 1А	1	SOT-223
U3	AMS1117-5.0	Стабілізатор напруги 5В, 1А	1	SOT-223

## Продовження таблиці 2.7

U4	TP4056	Контролер заряду Li-Ion, 1A	1	SOP-8
U5	PCA9685	16-канальний ШІМ- контролер	1	TSSOP- 28
U6	HC-SR501	PIR датчик руху	1	Модуль
U7	HC-SR04	Ультразвуковий далекомір	1	Модуль
R1, R2, R8	Резистор	10 кОм, 5%	3	SMD 0805
R3, R4	Резистор	4,7 кОм, 5%	2	SMD 0805
R5, R6, R7	Резистор	330 Ом, 5%	3	SMD 0805
C1, C2	Конденсатор	22 пФ, 5%, керамічний	2	SMD 0805
C3	Конденсатор	100 мкФ, 16В, електролітичний	1	Ø6,3 мм
C4, C5, C6, C7	Конденсатор	10 мкФ, 16В, керамічний	4	SMD 1206
C8, C9, C10, C11, C12	Конденсатор	0,1 мкФ, 50В, керамічний	5	SMD 0805
L1, L2	Феритовий фільтр	120 Ом @ 100 МГц	2	SMD 0805
D1, D2	Діод Шоткі	1N5819, 40В, 1А	2	SOD-123
D3	RGB-світлодіод	Загальний катод	1	5 мм
D4	Світлодіод білий	3,0-3,4В, 20мА	1	SMD 0805
DB1	Діодний міст	DB107, 1А, 1000В	1	DIP

Зм.	Арк.	№ докум.	Підпис	Дата

КВРКІ 220020.22.01.29 ПЗ

Арк.

39

Кінець таблиці 2.7

Q1, Q2	Транзистор NPN	BC847, 45B, 100мА	2	SOT-23
BZ1	Зумер	Активний, 5В	1	Ø12 мм
BT1	Акумулятор	Li-Ion, 3,7В, 2200мАг	1	18650
X1	Роз'єм живлення	Jack 5,5/2,1 мм	1	Кутовий
X2	Роз'єм розширення	PBD-20, 2,54 мм	1	Кутовий
SW1	Кнопка	Тактова, 6x6 мм	1	SMD
X1	Кварцовий резонатор	40 МГц, 20 ppm	1	HC-49S

Електрична принципова схема, розроблена в середовищі EasyEDA, забезпечує високу надійність роботи програмно-технічного засобу завдяки комплексному підходу до захисту від перенапруг, стабілізації живлення та фільтрації електромагнітних завад. Модульна структура схеми дозволяє здійснювати поетапне тестування окремих функціональних блоків під час налагодження та спрощує процес діагностики потенційних несправностей. Резервування критичних елементів системи, таких як джерело живлення та сенсори виявлення об'єктів, підвищує загальну надійність системи та забезпечує її безперебійне функціонування.

Оптимізація схеми для мінімізації енергоспоживання досягається шляхом використання низькоспоживаючих компонентів та реалізації режимів з динамічним керуванням енергоспоживанням. GPIO-порти ESP32, задіяні для керування енергоємними периферійними пристроями, такими як підсвічування камери та сервопривід, підключені через транзисторні ключі, що дозволяють повністю відключати живлення цих компонентів у режимі очікування. Використання спеціалізованих контролерів для керування периферійними пристроями, таких як PCA9685 для сервоприводу, дозволяє розвантажити центральний процесор ESP32 та оптимізувати загальне енергоспоживання системи.

									Арк.
									40
Зм.	Арк.	№ докум.	Підпис	Дата	КВРКІ 220020.22.01.29 ПЗ				

Конструктивні особливості розробленої схеми враховують вимоги до електромагнітної сумісності та захисту від впливу навколишнього середовища. Розміщення компонентів оптимізовано для мінімізації довжини сигнальних ліній та зниження взаємних завад між аналоговими та цифровими частинами схеми. Ключові сигнальні лінії, такі як шина I2C та лінії керування сервоприводом, захищені від зовнішніх завад шляхом екранування та використання фільтруючих компонентів. Розроблена схема підтримує можливість нарощування функціональності шляхом підключення додаткових модулів через стандартизований інтерфейс розширення, що забезпечує гнучкість системи та можливість її адаптації до різноманітних сценаріїв використання.

Імплементація розробленої електричної принципової схеми в середовищі EasyEDA дозволяє автоматизувати процес створення друкованої плати з оптимальним розташуванням компонентів та трасуванням провідників. Використання стандартизованих бібліотек компонентів та уніфікованих конструктивних рішень спрощує процес виробництва та знижує собівартість готового виробу. Реалізація схеми з використанням компонентів для поверхневого монтажу (SMD) дозволяє мінімізувати габаритні розміри пристрою та підвищити його стійкість до механічних впливів.

## 2.4 Монтажна схема програмно-апаратного засобу

Процес фізичної реалізації програмно-апаратного засобу автоматизованого доступу для домашніх тварин потребує розробки детальної монтажною схемою, яка систематизує процедуру з'єднання всіх компонентів у цілісну функціональну систему. Монтажна схема виступає практичним керівництвом для технічної імплементації теоретичних концепцій та інженерних рішень, закладених у принципову електричну схему. Документація монтажною схемою охоплює аспекти фізичного розташування компонентів, специфікацію з'єднувальних елементів, методологію екранування чутливих вузлів та практичні рекомендації щодо збірки і налагодження системи.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		41

Конструктивна реалізація модуля ESP32-CAM як центрального елемента системи передбачає його монтаж на спеціально спроектовану друковану плату з дотриманням рекомендацій виробника щодо розташування допоміжних компонентів. Оптимальне позиціонування модуля забезпечує ефективне тепловідведення від процесора ESP32 та мінімізує електромагнітні завади для сенсора камери OV2640. Фіксація модуля здійснюється за допомогою штирьових та гніздових з'єднувачів з кроком 2,54 мм, що забезпечує можливість швидкої заміни модуля при необхідності обслуговування системи. Периметр монтажного вікна для модуля ESP32-CAM екранований мідною доріжкою, з'єднаною з аналоговою землею, що знижує вплив зовнішніх електромагнітних завад на роботу камери.

Таблиця 2.8 деталізує монтажні характеристики основних електронних модулів системи, включаючи їх габаритні розміри, типи кріплення та вимоги до охолодження.

Таблиця 2.8 – Монтажні характеристики основних електронних модулів

Компонент	Габаритні розміри, мм	Тип кріплення	Вимоги до охолодження	Рекомендації з монтажу
ESP32-CAM	27 × 40,5 × 4,5	Штирьовий роз'єм 2,54 мм	Пасивне, з можливістю встановлення радіатора	Віддалення від джерел тепла, мінімізація довжини антенного провідника
TP4056	25 × 19 × 3	Пайка на плату або гніздовий роз'єм	Не потребує	Ізоляція від металевих частин корпусу, доступ повітря для охолодження
PCA9685	25,4 × 25,4 × 3	Штирьовий роз'єм 2,54 мм	Не потребує	Мінімізація довжини провідників до сервоприводу
HC-SR501	32 × 24 × 25	Гвинтове кріплення М3	Не потребує	Забезпечення вільного огляду зони детекції

Кінець таблиці 2.8

НС-SR04	45 × 20 × 15	Гвинтове кріплення М3	Не потребує	Розташування паралельно поверхні підлоги
Акумулятор 18650	Ø18 × 65	Спеціальний тримач	Пасивне, з доступом повітря	Ізоляція від інших компонентів, захист від короткого замикання
Сервопривід MG996R	40,7 × 19,7 × 42,9	Гвинтове кріплення М3 або М4	Не потребує для стандартних навантажень	Жорстка фіксація, амортизація вібрацій

Монтаж силових компонентів системи вимагає особливої уваги з огляду на потенційні теплові навантаження та електромагнітні завади. Стабілізатори напруги AMS1117-3.3 та AMS1117-5.0 монтуються на спеціально спроектовані теплорозсіювальні площадки друкованої плати, з'єднані з внутрішнім мідним шаром для ефективного відведення тепла. Додатковий теплопровідний клей між корпусами стабілізаторів та теплорозсіювальними площадками підвищує ефективність тепловідведення. Силові лінії друкованої плати виконані з підвищеною товщиною мідного шару (не менше 70 мкм) для зниження падіння напруги та мінімізації теплових втрат.

Просторове розділення аналогових та цифрових кіл на друкованій платі забезпечує мінімізацію взаємних завад між чутливими компонентами системи. Земляні полігони для аналогових та цифрових кіл об'єднуються в єдиній точці поблизу вхідного роз'єму живлення, що запобігає утворенню паразитних контурів струму. Високошвидкісні сигнальні лінії, такі як лінії даних камери та I2C-інтерфейс, розведені з дотриманням правил узгодження імпедансу та мінімізації перехресних завад.

Принципи розміщення компонентів на друкованій платі базуються на функціональній сегментації та оптимізації теплового режиму. Теплогенеруючі елементи, такі як стабілізатори напруги, розташовані на периферії плати для забезпечення ефективного тепловідведення. Чутливі компоненти, включаючи модуль ESP32-CAM та драйвер сервоприводу, ізольовані від потенційних джерел

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		43

завад. Роз'єми для підключення зовнішніх пристроїв згруповані для спрощення монтажу та обслуговування системи.

Зовнішні підключення до друкованої плати реалізовані через спеціалізовані роз'єми з захистом від неправильного підключення. Силові з'єднання виконані за допомогою роз'ємів з фіксаторами та маркуванням полярності для запобігання випадкового від'єднання та помилок при монтажі. Сигнальні лінії для підключення сенсорів та виконавчих пристроїв використовують роз'єми з ключами та кодуванням для уникнення помилкових з'єднань. Таблиця 2.9 представляє специфікацію з'єднувальних елементів, використаних у монтажній схемі.

Таблиця 2.9 – Специфікація з'єднувальних елементів монтажної схеми

Позначення	Тип з'єднувача	Призначення	Характеристики	Рекомендований кабель
X1	Jack 5,5/2,1 мм	Підключення зовнішнього живлення	12В, 2А	Силовий кабель 0,75 мм <sup>2</sup>
X2	JST-ХН 2Р	Підключення акумулятора	3,7В, 1А	Силовий кабель 0,5 мм <sup>2</sup>
X3	PLS-3	Підключення сервоприводу	5В, 1А	Трижильний екранований кабель 0,25 мм <sup>2</sup>
X4	PLS-4	Підключення НС-SR04	5В, 50мА	Чотирижильний екранований кабель 0,14 мм <sup>2</sup>
X5	PLS-3	Підключення НС-SR501	5В, 50мА	Трижильний екранований кабель 0,14 мм <sup>2</sup>

## Кінець таблиці 2.9

X6	PLS-2	Підключення RGB-світлодіода	3,3В, 20мА	Двожильний екранований кабель 0,14 мм <sup>2</sup>
X7	PLS-2	Підключення зумера	5В, 20мА	Двожильний екранований кабель 0,14 мм <sup>2</sup>
X8	PLS-2	Підключення кнопки налаштування	3,3В, 5мА	Двожильний екранований кабель 0,14 мм <sup>2</sup>
X9	IDC-10	Роз'єм програмування ESP32	3,3В, 50мА	Шлейф 10-жильний 0,14 мм <sup>2</sup>
X10	PLS-20	Роз'єм розширення	3,3В/5В, 500мА	Шлейф 20-жильний 0,14 мм <sup>2</sup>

Фізична компоновка системи передбачає розміщення електронних компонентів у захисному корпусі з полікарбонату, що забезпечує механічний захист та електричну ізоляцію. Корпус розділений на два функціональні відсіки: електронний відсік, що містить друковану плату з компонентами, та механічний відсік, у якому розташований сервопривід та механізм дверцят. Таке розділення забезпечує захист електронних компонентів від пилу та вологи, а також запобігає поширенню вібрацій від механічної частини на чутливі електронні вузли.

Вентиляційні отвори в корпусі забезпечують природну конвекцію для охолодження електронних компонентів, але розташовані таким чином, щоб запобігти прямому потраплянню вологи та пилу всередину пристрою. Додатковий пиловий фільтр на вхідних вентиляційних отворах запобігає забрудненню внутрішніх компонентів системи. Гумові ущільнення на всіх з'єднаннях корпусу забезпечують ступінь захисту IP65, що дозволяє експлуатувати пристрій у приміщеннях із підвищеною вологістю.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		45

Монтаж камери ESP32-CAM вимагає особливого підходу для забезпечення оптимального кута огляду та якості зображення. Камера розташована за спеціальним оптичним вікном з прозорого акрилу, що захищає об'єктив від пошкоджень та забруднень. Позиція камери регулюється за допомогою спеціального кріплення, що дозволяє налаштувати оптимальний кут нахилу для ідентифікації тварин різного зросту. Додаткове LED-підсвічування, розташоване навколо камери, забезпечує достатній рівень освітлення в умовах низької освітленості.

Механічна частина системи включає кронштейн для кріплення сервоприводу та механізм передачі обертального руху на вісь дверцят. Кронштейн виготовлений з алюмінієвого сплаву для забезпечення жорсткості конструкції та ефективного тепловідведення від сервоприводу. Передача обертального руху здійснюється через шестеренчатий редуктор з передавальним числом 2:1, що збільшує крутний момент на вісі дверцят для подолання можливого опору. Механічні компоненти системи монтуються на амортизуючих прокладках для зниження рівня шуму та вібрацій під час роботи.

Кабельна мережа системи організована з використанням різнокольорових провідників для спрощення ідентифікації та обслуговування. Силові лінії виконані провідниками з більшим перерізом (0,5-0,75 мм<sup>2</sup>) та червоним/чорним кольором для ідентифікації полярності. Сигнальні лінії використовують провідники меншого перерізу (0,14-0,25 мм<sup>2</sup>) різних кольорів відповідно до функціонального призначення. Екрановані кабелі застосовуються для чутливих сигнальних ліній, таких як лінії даних камери та I2C-інтерфейс, для захисту від електромагнітних завад.

Маршрутизація кабелів всередині корпусу пристрою здійснюється з дотриманням правил електромагнітної сумісності. Силові та сигнальні кабелі прокладаються окремими шляхами з мінімальним перехрещенням для зниження електромагнітних завад. Фіксація кабелів здійснюється за допомогою спеціальних затискачів та стяжок для запобігання механічним пошкодженням при вібраціях та

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		46

транспортуванні. Введення кабелів у корпус пристрою виконується через спеціальні ущільнені вводи, що забезпечують захист від вологи та пилу.

Таблиця 2.10 деталізує характеристики механічних компонентів та корпусу системи.

Таблиця 2.10 – Характеристики механічних компонентів та корпусу системи

Компонент	Характеристики	Матеріал	Розміри, мм	Метод кріплення
Корпус електроніки	IP65, з вентиляційними отворами	Полікарбонат	150 × 100 × 50	Гвинтове з'єднання М3
Корпус механізму	IP65, зі знімною кришкою	Полікарбонат	120 × 100 × 70	Гвинтове з'єднання М3
Монтажна пластина	Теплопровідна	Алюміній 1,5 мм	130 × 80	Гвинтове з'єднання М3
Кронштейн сервоприводу	Жорсткий, з амортизацією	Алюміній 2 мм	50 × 30 × 40	Гвинтове з'єднання М3
Шестеренчатий редуктор	Передавальне число 2:1	Поліацеталь	Ø30 × 15	Пресова посадка на вісь
Вісь дверцят	Діаметр 8 мм	Нержавіюча сталь	Ø8 × 150	Підшипникова опора
Оптичне вікно камери	Прозоре, з антибліковим покриттям	Акрил 3 мм	30 × 30	Клейове з'єднання
Ущільнювачі корпусу	Водонепроникні	Силікон	По периметру	Клейове з'єднання

Зм.	Арк.	№ докум.	Підпис	Дата

КВРКІ 220020.22.01.29 ПЗ

Арк.

47

Кінець таблиці 2.10

Вентиляційна решітка	З пиловим фільтром	ABS-пластик	40 × 40	Гвинтове з'єднання М2
Кабельні вводи	Ступінь захисту IP68	Поліамід	Ø16 × 20	Різьбове з'єднання М16

Процедура монтажу системи включає послідовні етапи збірки друкованої плати, механічних компонентів та їх інтеграції в єдину систему. Збірка друкованої плати починається з монтажу пасивних SMD-компонентів, таких як резистори та конденсатори, з подальшим монтажем активних компонентів та роз'ємів. Пайка здійснюється з використанням безсвинцевого припою для відповідності екологічним нормам та підвищення надійності з'єднань. Тестування друкованої плати проводиться поетапно, з перевіркою окремих функціональних блоків перед остаточною збіркою системи. Перевірка блоку живлення включає вимірювання вихідних напруг та їх стабільності при різних навантаженнях. Тестування обчислювального блоку здійснюється шляхом програмування тестового коду та перевірки комунікаційних інтерфейсів. Діагностика сенсорної та виконавчої підсистем включає перевірку правильності роботи датчиків та сервоприводу в різних режимах.

Монтаж механічних компонентів починається зі збірки корпусу та встановлення монтажної пластини. Сервопривід кріпиться до кронштейна з використанням амортизуючих прокладок для зниження вібрацій. Шестеренчатий редуктор з'єднується з валом сервоприводу та віссю дверцят, забезпечуючи передачу крутного моменту. Вісь дверцят встановлюється в підшипникові опори для зниження тертя та подовження терміну служби механізму.

Інтеграція електронних та механічних компонентів здійснюється шляхом монтажу друкованої плати на монтажну пластину та підключення всіх кабелів

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		48

згідно з монтажною схемою. Датчики руху та відстані монтуються в спеціально передбачені місця корпусу з оптимальним кутом огляду для надійної детекції об'єктів. Камера ESP32-CAM позиціонується за оптичним вікном з можливістю регулювання кута нахилу для оптимізації поля зору.

Фінальне тестування змонтованої системи включає перевірку правильності роботи всіх функціональних блоків у різних режимах експлуатації. Тест електроживлення перевіряє стабільність роботи системи як від зовнішнього джерела, так і від акумулятора, з вимірюванням напруг та струмів споживання. Перевірка сенсорної підсистеми включає тестування детекції руху та визначення відстані до об'єктів у різних умовах освітлення. Діагностика механізму дверцят перевіряє плавність роботи сервоприводу та надійність відкриття/закриття.

Надана монтажна схема програмно-апаратного засобу автоматизованого доступу для домашніх тварин забезпечує деталізоване керівництво для практичної реалізації теоретичних концепцій системи. Комплексний підхід до монтажу, включаючи аспекти електромагнітної сумісності, теплового режиму та механічної надійності, забезпечує створення надійного та довговічного пристрою, що відповідає всім функціональним вимогам та нормативним стандартам.

## 2.5 Аналіз обраних рішень

Імплементація програмно-технічного засобу автоматизованого доступу для домашніх тварин базується на серії обґрунтованих технічних рішень, що забезпечують оптимальний баланс між функціональністю, надійністю, вартістю та енергоефективністю системи. Кожен компонент системи піддавався ретельному аналізу та порівнянню з альтернативними варіантами для забезпечення максимальної відповідності поставленим вимогам. Методологія вибору базувалася на багатокритеріальному аналізі з урахуванням як технічних характеристик, так і економічних факторів, що дозволило сформуванню раціональної архітектури системи, здатну ефективно вирішувати поставлені задачі в умовах наявних обмежень.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		49

Вибір ESP32-CAM як центрального обчислювального модуля зумовлений унікальним поєднанням потужного мікроконтролера з інтегрованою камерою, що суттєво спрощує архітектуру системи та знижує її вартість порівняно з варіантами на базі окремих компонентів. Порівняльний аналіз ESP32-CAM з альтернативними рішеннями, представлений у таблиці 2.11, демонструє значні переваги обраного модуля за співвідношенням функціональності та вартості.

Таблиця 2.11 – Порівняльний аналіз мікроконтролерних платформ з інтегрованими камерами

Характеристика	ESP32-CAM	Raspberry Pi Zero W + камера	Arduino Uno + OV7670	OpenMV Cam H7	M5Stack ESP32 Camera
Процесор	Xtensa LX6, 240 МГц, двоядерний	ARM Cortex-A53, 1 ГГц, одноядерний	ATmega328 P, 16 МГц	ARM Cortex-M7, 480 МГц	ESP32, 240 МГц, двоядерний
Оперативна пам'ять	520 КБ SRAM	512 МБ SDRAM	2 КБ SRAM	1 МБ RAM	520 КБ SRAM
Флеш-пам'ять	4 МБ	Залежить від SD-карти	32 КБ	2 МБ + microSD	4 МБ
Роздільна здатність камери	2 МП (1600×1200)	До 8 МП (3280×2464)	0,3 МП (640×480)	5 МП (2592×1944)	2 МП (1600×1200)

Кінець таблиці 2.11 – Порівняльний аналіз мікроконтролерних платформ з інтегрованими камерами

Wi-Fi	802.11 b/g/n	802.11 b/g/n	Відсутній (потрібен модуль)	802.11 b/g/n	802.11 b/g/n
Bluetooth	4.2 BLE	4.1 BLE	Відсутній (потрібен модуль)	Відсутній	4.2 BLE
GPIO-порти	9	40	14	24	9
Живлення	5В, ~120 мА	5В, ~250 мА	5В, ~50 мА + ~70 мА (камера)	5В, ~200 мА	5В, ~150 мА
Вартість, \$	5-10	30-40 + 15-30 (камера)	4-8 + 5-10 (камера)	65-85	20-30
Підтримка ML	Обмежена, але достатня для розпізнавання	Висока	Низька	Висока	Обмежена, аналогічно ESP32-CAM
Розміри, мм	27 × 40,5 × 4,5	65 × 30 × 5 + камера	68,6 × 53,4 × 25 + камера	45 × 36 × 10	54 × 54 × 12,2

Обрана стратегія розпізнавання об'єктів базується на використанні легких згорткових нейронних мереж (CNN), оптимізованих для роботи на пристроях з обмеженими обчислювальними ресурсами. Модель MobileNetV2, адаптована для ESP32 через фреймворк TensorFlow Lite for Microcontrollers, забезпечує

прийнятний компроміс між точністю розпізнавання та обчислювальною складністю. Квантизація моделі до 8-бітної точності дозволяє значно зменшити вимоги до пам'яті та підвищити швидкість інференсу без критичної втрати точності. Експериментальні результати демонструють спроможність системи розпізнавати до 5 класів домашніх тварин (кіт, собака, кролик, хом'як, морська свинка) з точністю до 92% при оптимальному освітленні та до 85% при низькому освітленні.

Архітектура програмного забезпечення розроблена відповідно до принципів мікросервісного підходу, що забезпечує модульність, масштабованість та простоту підтримки системи. Кожен функціональний аспект системи (захоплення зображення, розпізнавання об'єктів, керування механізмом, мережева комунікація) реалізований як окремий модуль з чітко визначеними інтерфейсами взаємодії. Застосування асинхронного підходу до обробки подій дозволяє ефективно використовувати двоядерну архітектуру ESP32, розподіляючи задачі між ядрами залежно від їх пріоритету та обчислювальної складності. Так, одне ядро відповідає за захоплення та обробку зображень, тоді як друге забезпечує мережеву комунікацію та керування периферійними пристроями.

Вибір технології безпроводної комунікації базувався на багатокритеріальному аналізі доступних стандартів з урахуванням вимог до надійності зв'язку, енергоефективності та сумісності з екосистемою розумного будинку. Wi-Fi обрано як основний канал комунікації завдяки високій пропускну здатності та простоті інтеграції з існуючими мережами. Bluetooth забезпечує альтернативний канал для початкового налаштування та діагностики. Протокол MQTT використовується для обміну повідомленнями між компонентами системи завдяки низьким накладним витратам та підтримці різних рівнів якості обслуговування (QoS).

Програмна реалізація фреймворку для розпізнавання об'єктів включає підсистему попередньої обробки зображення для нормалізації умов освітлення та видалення шумів. Алгоритм адаптивної корекції яскравості забезпечує стабільну роботу системи в різних умовах освітлення, автоматично адаптуючи параметри обробки зображення до поточних умов. Механізм темпоральної фільтрації результатів розпізнавання підвищує стабільність системи, зменшуючи вплив

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		52

випадкових помилок класифікації. Рішення про надання доступу приймається на основі аналізу результатів розпізнавання за кількома послідовними кадрами, що суттєво знижує ймовірність помилкових спрацювань.

Енергоефективність системи оптимізована шляхом впровадження багаторівневого режиму енергоспоживання з динамічним керуванням частотою процесора та активацією периферійних пристроїв. У режимі глибокого сну система споживає менше 10 мА, активуючись лише при спрацюванні датчика руху, що дозволяє досягти тривалого часу автономної роботи від акумулятора. Реалізація розумного алгоритму керування LED-підсвічуванням, що активує його лише при недостатньому рівні освітлення, додатково зменшує енергоспоживання системи. Використання конденсаторів великої ємності для буферизації пікових навантажень при активації сервоприводу дозволяє зменшити навантаження на акумулятор і подовжити термін його служби.

Механічна підсистема проєктувалася з урахуванням вимог до надійності, довговічності та безпеки експлуатації. Сервопривід MG996R з крутним моментом 12 кг/см забезпечує надійне функціонування дверцят різних розмірів та ваги. Система захисту від заземлення, реалізована на базі датчика струму сервоприводу, автоматично зупиняє закривання дверцят при виявленні перешкоди, забезпечуючи безпеку для тварин. Двосторонній механізм фіксації дверцят у закритому положенні запобігає несанкціонованому відкриттю сторонніми тваринами та підвищує рівень безпеки системи.

Інтерфейс користувача розроблений відповідно до принципів людино-центрованого дизайну з фокусом на простоту використання та мінімальну кількість кроків для виконання типових операцій. Мобільний додаток для iOS та Android забезпечує зручне керування системою з інтуїтивно зрозумілим інтерфейсом та розширеними можливостями моніторингу. Веб-інтерфейс надає альтернативний спосіб доступу до системи з будь-якого пристрою з веб-браузером. Інтеграція з голосовими асистентами (Amazon Alexa, Google Assistant) розширює можливості взаємодії з системою без необхідності використання смартфона чи комп'ютера.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		53

Вибір технологій для кожного компонента системи здійснювався з урахуванням не лише технічних характеристик, але й доступності компонентів, перспектив їх підтримки виробниками та вартості. Таблиця 2.12 представляє результати аналізу програмних рішень, застосованих у системі, з оцінкою їх переваг та обмежень.

Таблиця 2.12 – Аналіз програмних рішень, застосованих у системі

Компонент	Обране рішення	Альтернативи	Переваги	Обмеження	Обґрунтування вибору
Операційна система	FreeRTOS	Bare metal, ESP-IDF RTOS	Підтримка багатозадачності, ефективне керування ресурсами, широка екосистема	Додаткові накладні витрати порівняно з bare metal	Оптимальний баланс між функціональністю та накладними витратами, зрілість платформи
Фреймворк ML	TensorFlow Lite for Microcontrollers	ESP-WHO, Edge Impulse, CMSIS-NN	Широка підтримка моделей, вбудована оптимізація для мікроконтролерів, розвинена документація	Вища складність порівняно з ESP-WHO	Гнучкість у виборі моделей, перспективи розвитку, наявність готових моделей
Комунікаційний протокол	MQTT	HTTP, WebSockets, CoAP	Низькі накладні витрати, підтримка QoS, модель pub/sub	Обмежені можливості для передачі великих обсягів даних	Оптимальний для IoT-застосувань з обмеженими ресурсами, широка підтримка в екосистемі розумного будинку



Оптимізація алгоритму розпізнавання для роботи на пристрої з обмеженими ресурсами потребувала застосування низки технік зменшення складності моделі. Використання архітектури MobileNetV2 з глибиною множника 0,25 та вхідним розміром зображення 128×128 пікселів дозволило досягти прийняттого балансу між точністю та швидкодією. Квантизація ваг моделі до 8-бітного цілочисельного представлення значно зменшила вимоги до пам'яті та підвищила швидкість обчислень. Тестування на наборі даних з 500 зображень п'яти класів домашніх тварин показало, що квантизована модель займає лише 320 КБ пам'яті та забезпечує інференс зі швидкістю до 4 кадрів на секунду, що достатньо для задачі контролю доступу.

Мережева безпека системи реалізована на кількох рівнях для забезпечення комплексного захисту. Автентифікація користувачів базується на механізмі JWT (JSON Web Tokens) з обмеженим терміном дії токенів та можливістю їх відкликання. Шифрування даних при передачі здійснюється за допомогою протоколу TLS 1.3, що забезпечує конфіденційність та цілісність комунікації. Локальна мережа системи захищена брандмауером з обмеженим доступом лише для авторизованих пристроїв. Регулярне оновлення програмного забезпечення через захищений канал OTA (Over-The-Air) забезпечує усунення виявлених вразливостей та впровадження нових функцій безпеки.

Аналіз надійності системи включав оцінку середнього часу напрацювання на відмову (MTBF) для різних компонентів та визначення найбільш критичних точок відмови. Розрахунковий MTBF для електронних компонентів становить не менше 50 000 годин при нормальних умовах експлуатації. Механічні компоненти, зокрема сервопривід та механізм дверцят, мають розрахунковий ресурс не менше 100 000 циклів відкривання/закривання. Реалізація механізму аварійного відкриття дверцят забезпечує доступ для тварин навіть при повній відмові електронної частини системи, що критично важливо для безпеки домашніх улюбленців.

Масштабованість системи забезпечується модульною архітектурою та стандартизованими інтерфейсами, що дозволяють розширювати функціональність без значних модифікацій базової структури. Підтримка кластерного режиму роботи

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		56

дозволяє об'єднувати кілька пристроїв у єдину систему для забезпечення доступу через різні точки входу з централізованим керуванням. Відкритий API надає можливість інтеграції з сторонніми системами та розробки додаткових модулів розширення функціональності.

Екологічні аспекти проєкту враховувалися на всіх етапах розробки, від вибору компонентів до розробки стратегії утилізації. Використання безсвинцевого припою при виробництві відповідає директиві RoHS. Енергоефективність системи знижує її вуглецевий слід порівняно з традиційними рішеннями.

Економічна ефективність обраних рішень підтверджується аналізом сукупної вартості розробки, виробництва та експлуатації системи. Розрахункова вартість компонентів у серійному виробництві не перевищує 30% від ринкової вартості аналогічних комерційних рішень з нижчою функціональністю. Енергоефективність забезпечує низьку вартість експлуатації, а тривалий термін служби компонентів мінімізує витрати на технічне обслуговування та ремонт. Комплексний аналіз обраних апаратних та програмних рішень демонструє їхню відповідність поставленим вимогам та забезпечує оптимальний баланс між функціональністю, надійністю, безпекою та економічною ефективністю системи автоматизованого доступу для домашніх тварин.

## 2.6 Висновки до розділу 2

Проєктування програмно-технічного засобу автоматизованого доступу для домашніх тварин з використанням технології розпізнавання об'єктів на базі ESP32 CAM дозволило створити комплексне рішення, що відповідає сучасним вимогам розумного будинку. Розроблена структура системи включає п'ять ключових підсистем: сенсорну, обчислювальну, виконавчу, комунікаційну та енергозабезпечення, які забезпечують повний цикл функціонування автоматизованого доступу. Модульний підхід до проєктування значно підвищує гнучкість системи, спрощує процес розробки, тестування та розширення

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		57

функціональності, що є критично важливим для інтеграції з різноманітними екосистемами розумного будинку.

Електрична принципова схема, розроблена в середовищі EasyEDA, забезпечує оптимальне функціонування всіх компонентів системи з урахуванням особливостей взаємодії між ними. Центральний елемент системи – модуль ESP32-CAM – інтегрований з іншими компонентами з дотриманням вимог до енергоефективності, електромагнітної сумісності та надійності. Особлива увага приділена системі резервного живлення на базі літій-іонного акумулятора з контролером заряду TP4056, що забезпечує стабільну роботу пристрою навіть при відключенні зовнішнього живлення. Розроблена монтажна схема систематизує процес фізичної реалізації системи, враховуючи аспекти теплового режиму, захисту від зовнішніх впливів та оптимального розташування компонентів.

Аналіз обраних апаратних та програмних рішень підтвердив їх відповідність поставленим вимогам та доцільність використання в даному проєкті. Вибір модуля ESP32-CAM як центрального елемента системи обґрунтований оптимальним співвідношенням функціональності, енергоефективності та вартості порівняно з альтернативними рішеннями. Розроблена архітектура програмного забезпечення з асинхронним підходом до обробки подій дозволяє ефективно використовувати обчислювальні ресурси ESP32 для одночасного виконання задач розпізнавання образів, керування периферійними пристроями та мережевої комунікації. Використання протоколу MQTT для обміну повідомленнями забезпечує надійну інтеграцію з екосистемою розумного будинку при мінімальних накладних витратах.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		58

### 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ АВТОМАТИЗОВАНОГО ДОСТУПУ ДЛЯ ДОМАШНІХ ТВАРИН У РОЗУМНОМУ БУДИНКУ НА ОСНОВІ ВИЯВЛЕННЯ ОБ'ЄКТІВ ЗА ДОПОМОГОЮ ESP32 CAM

#### 3.1 Програмна реалізація клієнтської частини програмно-апаратного засобу

Програмна реалізація клієнтської частини програмно-апаратного засобу автоматизованого доступу для домашніх тварин становить собою комплексне рішення, спрямоване на забезпечення зручної взаємодії користувача з системою через різноманітні інтерфейси. Розробка клієнтської частини здійснювалася з орієнтацією на кросплатформність, інтуїтивну зрозумілість та реактивний дизайн, що адаптується до різних пристроїв виведення інформації. Архітектура клієнтської частини побудована за принципом Single Page Application (SPA) з використанням фреймворку React.js, що забезпечує високу продуктивність та гнучкість при розробці інтерактивних користувацьких інтерфейсів.

Інфраструктура клієнтської частини включає набір взаємопов'язаних модулів, кожен з яких відповідає за певний аспект функціональності системи. Модульний підхід до організації коду сприяє кращій підтримованості та розширюваності програмного забезпечення, дозволяючи незалежно розвивати та тестувати окремі компоненти системи. Структура проєкту організована відповідно до методології атомарного дизайну, де інтерфейс формується з простих елементів (атомів), які об'єднуються в складніші компоненти (молекули), а ті, у свою чергу, утворюють функціональні блоки (організми) інтерфейсу користувача.

Реалізація аутентифікації користувачів базується на використанні JWT-токенів (JSON Web Token), що забезпечує безпечну передачу інформації між клієнтом та сервером у вигляді JSON-об'єкта. Механізм аутентифікації включає процедуру реєстрації нових користувачів, авторизацію існуючих користувачів та систему управління правами доступу на основі ролей.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		59

Візуальний інтерфейс користувача розроблений з урахуванням принципів інклюзивного дизайну, забезпечуючи доступність для людей з різними обмеженнями. Колірна схема інтерфейсу обрана з урахуванням достатнього контрасту для користувачів з порушеннями зору, а всі інтерактивні елементи супроводжуються текстовими мітками для підтримки екранних читачів. Адаптивний дизайн забезпечує коректне відображення інтерфейсу на пристроях з різною роздільною здатністю екрану, від мобільних телефонів до настільних комп'ютерів з великими моніторами.

Функціональність моніторингу домашніх тварин реалізована через систему відеопотоку в режимі реального часу з можливістю запису та зберігання відеофрагментів для подальшого аналізу. Інтеграція технології WebRTC забезпечує низьку затримку передачі відео, що критично важливо для своєчасного реагування на поведінку тварин. Адаптивне керування якістю відеопотоку залежно від доступної пропускної здатності мережі оптимізує використання мережевих ресурсів та забезпечує безперебійність передачі даних навіть при обмеженому інтернет-з'єднанні.

Компонент управління доступом надає користувачеві можливість налаштовувати правила автоматичного відкривання дверцят для різних тварин та часових проміжків. Інтерфейс дозволяє створювати, редагувати та видаляти правила доступу через інтуїтивно зрозумілу систему форм та візуальних елементів управління.

Модуль аналітики забезпечує візуалізацію статистичних даних щодо активності домашніх тварин, включаючи частоту використання дверцят, часи активності та тривалість перебування на вулиці. Для візуалізації даних використовується бібліотека D3.js, що дозволяє створювати інтерактивні графіки та діаграми з можливістю детального аналізу окремих часових проміжків. Архітектура модуля передбачає як відображення історичних даних, так і моніторинг активності в режимі реального часу з автоматичним оновленням візуалізацій при надходженні нових даних.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
						60
Зм.	Арк.	№ докум.	Підпис	Дата		

Архітектура стану додатку реалізована з використанням Redux, що забезпечує централізоване управління станом та передбачуваний потік даних. Використання Redux Toolkit спрощує роботу з асинхронними операціями та зменшує обсяг шаблонного коду, необхідного для реалізації типових сценаріїв взаємодії з API. Структура сховища Redux організована за принципом доменних сегментів, де кожен сегмент відповідає за певну функціональну область додатку.

Інтеграція з іншими системами розумного будинку здійснюється через API шлюзів, що забезпечують уніфікований інтерфейс для взаємодії з різними пристроями та платформами. Підтримка протоколів MQTT та WebSocket дозволяє встановлювати двосторонній зв'язок з іншими компонентами екосистеми розумного будинку та реагувати на події в режимі реального часу. Таблиця 3.1 представляє порівняльну характеристику підтримуваних інтеграцій з поширеними платформами розумного будинку.

Таблиця 3.1 – Порівняльна характеристика інтеграцій з платформами розумного будинку

Платформа	Тип інтеграції	Підтримувані функції	Протокол комунікації	Рівень безпеки	Складність налаштування
Home Assistant	Нативна	Керування доступом, відеоспостереження, сповіщення, аналітика	MQTT, WebSocket, REST API	Високий (TLS 1.3, OAuth 2.0)	Середня
Apple HomeKit	Через міст	Керування доступом, сповіщення, базова аналітика	HomeKit API	Високий (HomeKit Secure Video)	Низька
Google Home	Через міст	Керування доступом, сповіщення	Google Smart Home API	Високий (OAuth 2.0)	Середня
Amazon Alexa	Через Skills API	Керування доступом, сповіщення	Alexa Skills Kit	Високий (OAuth 2.0)	Середня
OpenHAB	Через адаптер	Повна функціональність	REST API, MQTT	Середній (залежно від конфігурації)	Висока



відбувається на серверній частині з подальшим оновленням моделі на пристрої ESP32 CAM. Візуалізація процесу навчання з використанням прогрес-барів та інформаційних повідомлень забезпечує прозорість процесу та підвищує довіру користувача до системи.

Мобільна версія клієнтської частини розроблена з використанням фреймворку React Native, що дозволяє повторно використовувати значну частину бізнес-логіки між веб-додатком та мобільними додатками для iOS та Android. Специфічні для мобільних платформ функції, такі як доступ до камери для створення фотографій тварин або використання біометричної аутентифікації, інкапсульовані в окремі адаптери з уніфікованим інтерфейсом. Це забезпечує єдиний підхід до розробки функціональності на всіх підтримуваних платформах при збереженні нативного вигляду та відчуття інтерфейсу на кожній з них.

Системний підхід до тестування клієнтської частини включає автоматизовані юніт-тести окремих компонентів з використанням Jest та React Testing Library, інтеграційні тести для перевірки взаємодії між компонентами та наскрізні тести з використанням Cypress для перевірки функціональності системи в цілому. Впровадження практики безперервної інтеграції (CI) з автоматичним запуском тестів при кожному комміті до репозиторію забезпечує ранню детекцію потенційних проблем та підтримку стабільності кодової бази протягом всього життєвого циклу проєкту.

Механізм оновлення клієнтської частини реалізований з використанням Service Worker, що дозволяє контролювати процес кешування ресурсів та забезпечує можливість оновлення додатку в фоновому режимі. При виявленні нової версії користувачеві пропонується оновити додаток, при цьому завантаження нових ресурсів відбувається паралельно з роботою поточної версії, мінімізуючи переривання користувацького досвіду. Підтримка стратегії progressive enhancement забезпечує базову функціональність навіть на пристроях з обмеженими можливостями або при нестабільному інтернет-з'єднанні.

Досягнення високого рівня доступності клієнтської частини забезпечується шляхом дотримання стандарту WCAG 2.1 на рівні AA. Всі інтерактивні елементи

					КвРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		63

мають відповідні атрибути аґіа для забезпечення семантичної структури інтерфейсу, доступної для асистивних технологій. Регулярне тестування з використанням інструментів автоматизованої перевірки доступності, таких як Ахе, та мануальне тестування з використанням екранних читачів забезпечує виявлення та усунення потенційних проблем з доступністю.

Інтернаціоналізація та локалізація інтерфейсу користувача реалізована з використанням бібліотеки i18next, що забезпечує гнучкий механізм керування перекладами та адаптації інтерфейсу до різних мов та культурних особливостей. Підтримка множинних форм для різних мов, форматування дат та чисел відповідно до локальних стандартів, а також можливість динамічної зміни мови інтерфейсу без перезавантаження сторінки підвищують комфорт використання системи для користувачів з різних країн.

Клієнтська частина програмно-апаратного засобу автоматизованого доступу для домашніх тварин розроблена з урахуванням сучасних тенденцій у галузі веб-розробки та користувацького досвіду. Модульна архітектура, реактивний дизайн та орієнтація на доступність забезпечують високу якість користувацького досвіду та гнучкість у подальшому розвитку системи. Інтеграція з різноманітними платформами розумного будинку та підтримка різних пристроїв виведення інформації розширюють можливості застосування розробленого рішення та підвищують його цінність для кінцевих користувачів.

### 3.2 Алгоритми розпізнавання

Розробка ефективних алгоритмів розпізнавання домашніх тварин для пристроїв з обмеженими обчислювальними ресурсами вимагає комплексного підходу до оптимізації як архітектури нейронних мереж, так і методів попередньої обробки зображень. Обчислювальна потужність ESP32 CAM накладає суттєві обмеження на складність застосовуваних моделей машинного навчання, що потребує пошуку оптимального балансу між точністю розпізнавання та швидкодією. Запропонований алгоритм базується на модифікованій архітектурі

					КвРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		64

MobileNetV2, додатково оптимізований для роботи на мікроконтролерах через квантизацію ваг та структурну компресію моделі.

Передобробка вхідних зображень становить критичний етап у роботі алгоритму розпізнавання, забезпечуючи нормалізацію даних та підвищення стійкості до варіацій умов освітлення. Послідовність операцій включає масштабування зображення до розміру 128×128 пікселів, конвертацію колірної моделі з RGB до YUV для ефективнішого розділення інформації про яскравість та колір, адаптивну еквалізацію гістограми для компенсації нерівномірного освітлення та видалення шумів за допомогою білатерального фільтра. Експериментальні дослідження показали, що такий підхід до передобробки підвищує точність розпізнавання на 7-12% у складних умовах освітлення порівняно з базовою нормалізацією.

Архітектура нейронної мережі для розпізнавання тварин представлена у таблиці 3.2, де деталізовані параметри кожного шару та обчислювальна складність відповідних операцій. Оптимізація вагових коефіцієнтів нейронної мережі здійснюється шляхом квантизації до 8-бітної точності, що зменшує вимоги до пам'яті та підвищує швидкість обчислень на ESP32. Процес квантизації включає визначення оптимальних діапазонів квантизації для кожного шару на основі розподілу вагових коефіцієнтів та активацій. Експериментальна валідація показала, що запропонований метод квантизації зменшує розмір моделі на 73% (з 33,5 КБ до 8,9 КБ) з погіршенням точності лише на 1,3% порівняно з моделлю з плаваючою точкою.

Таблиця 3.2 - Архітектура модифікованої MobileNetV2 для ESP32 CAM

Шар	Тип операції	Розмір вхідних даних	Розмір вихідних даних	Кількість параметрів	Обчислювальна складність (MACs)
1	Conv2D (3×3, stride=2)	128×128×3	64×64×8	216	884,736

Кінець таблиці 3.2

2	BatchNorm + ReLU6	64×64×8	64×64×8	16	32,768
3	DepthwiseConv 2D (3×3, stride=1)	64×64×8	64×64×8	72	294,912
4	BatchNorm + ReLU6	64×64×8	64×64×8	16	32,768
5	Conv2D (1×1, stride=1)	64×64×8	64×64×16	128	524,288
6	BatchNorm + ReLU6	64×64×16	64×64×16	32	65,536
7	DepthwiseConv 2D (3×3, stride=2)	64×64×16	32×32×16	144	147,456
8	BatchNorm + ReLU6	32×32×16	32×32×16	32	16,384
9	Conv2D (1×1, stride=1)	32×32×16	32×32×32	512	524,288
10	BatchNorm + ReLU6	32×32×32	32×32×32	64	32,768
11	Bottleneck (×2)	32×32×32	32×32×32	4,608	2,359,296
12	DepthwiseConv 2D (3×3, stride=2)	32×32×32	16×16×32	288	73,728
13	BatchNorm + ReLU6	16×16×32	16×16×32	64	8,192
14	Conv2D (1×1, stride=1)	16×16×32	16×16×64	2,048	524,288
15	BatchNorm + ReLU6	16×16×64	16×16×64	128	16,384
16	GlobalAverageP ooling2D	16×16×64	64	0	16,384
17	Dense	64	5	325	325
Всього				8,693	5,554,501

Зм.	Арк.	№ докум.	Підпис	Дата

КВРКІ 220020.22.01.29 ПЗ

Арк.

66

Процес квантизації включає визначення оптимальних діапазонів квантизації для кожного шару на основі розподілу вагових коефіцієнтів та активацій. Експериментальна валідація показала, що запропонований метод квантизації зменшує розмір моделі на 73% (з 33,5 КБ до 8,9 КБ) з погіршенням точності лише на 1,3% порівняно з моделлю з плаваючою точкою.

Механізм інференсу на ESP32 CAM базується на оптимізованій реалізації операцій згортки та активації з використанням інструкцій ESP32 для прискорення обчислень. Векторизація обчислень через SIMD-інструкції процесора дозволяє досягти до 3,5 разів вищої продуктивності порівняно з неоптимізованою реалізацією. Розподіл обчислень між ядрами ESP32 забезпечує паралельну обробку незалежних частин нейронної мережі, додатково підвищуючи загальну продуктивність системи.

Темпоральний аналіз послідовності кадрів впроваджений для підвищення стабільності результатів розпізнавання. Алгоритм використовує ковзне вікно з 5 останніх кадрів, застосовуючи експоненційне згладжування для зниження впливу короткочасних флуктуацій у результатах класифікації. Додатково впроваджено механізм виявлення аномалій, що дозволяє ідентифікувати та фільтрувати потенційно помилкові результати розпізнавання на основі різких змін у прогнозованих класах або впевненості моделі.

Фреймворк для навчання моделі реалізований на базі TensorFlow з можливістю трансферного навчання від попередньо навченої на великому наборі даних моделі. Процедура навчання включає збір та розмітку зображень конкретних домашніх тварин, аугментацію даних для підвищення стійкості моделі та використання техніки раннього зупинення для запобігання перенавчання. Фінальна модель конвертується у формат TensorFlow Lite з подальшою оптимізацією для цільової платформи ESP32.

Тестування алгоритму на наборі даних з 500 зображень 5 різних домашніх тварин показало точність розпізнавання 94,3% при оптимальних умовах освітлення та 87,1% при низькому рівні освітлення. Середній час інференсу на ESP32 CAM становить 270 мс на кадр, що дозволяє досягти частоти обробки близько 3,7 кадрів

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		67

на секунду при активному розпізнаванні, що достатньо для задачі контролю доступу. Впровадження описаних оптимізацій дозволило зменшити енергоспоживання системи на 42% порівняно з базовою реалізацією при збереженні прийняттого рівня точності.

Блок-схема алгоритму розпізнавання домашніх тварин з використанням ESP32 CAM представлена на рисунку 3.1. Алгоритм включає етапи ініціалізації системи, захоплення зображення, передобробки, інференсу нейронної мережі, темпорального аналізу результатів та прийняття рішення щодо надання доступу.

Виявлення рухомих об'єктів у полі зору камери реалізовано з використанням алгоритму віднімання фону на основі суміші гаусівських розподілів (MOG2). Такий підхід дозволяє ефективно виділяти рухомі об'єкти на статичному фоні з адаптацією до поступових змін у освітленні та сцені. Комбінування детектора руху з основним алгоритмом розпізнавання дозволяє суттєво знизити енергоспоживання системи, активуючи ресурсомісткі операції нейромережевого аналізу лише при виявленні потенційного об'єкта інтересу.

Механізм динамічної адаптації порогів прийняття рішень залежно від умов освітлення підвищує робастність системи при експлуатації в різноманітних середовищах. Алгоритм автоматично знижує поріг впевненості для класифікації при виявленні складних умов освітлення, зберігаючи високий поріг при оптимальному освітленні для мінімізації хибних спрацювань. Параметри адаптації визначаються на основі аналізу гістограми яскравості вхідного зображення та оцінки рівня контрасту.

Локальне навчання моделі на пристрої реалізоване за допомогою техніки fine-tuning останніх шарів нейронної мережі для адаптації до конкретних домашніх тварин користувача. Процес навчання запускається автоматично при накопиченні достатньої кількості верифікованих користувачем зображень тварин та виконується у фоновому режимі в періоди низької активності системи. Механізм захисту від деградації моделі автоматично відкидає результати навчання, якщо вони призводять до погіршення загальної точності розпізнавання.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
						68
Зм.	Арк.	№ докум.	Підпис	Дата		

Запропонований алгоритм розпізнавання ефективно поєднує сучасні методи комп'ютерного зору та глибокого навчання з оптимізаціями для роботи на пристроях з обмеженими обчислювальними ресурсами. Його модульна структура дозволяє гнучко адаптувати систему до різних умов експлуатації та вимог користувачів, забезпечуючи оптимальний баланс між точністю, швидкістю та енергоефективністю.

### 3.3 Висновки до розділу 3

Програмна реалізація програмно-технічного засобу автоматизованого доступу для домашніх тварин на базі ESP32 CAM продемонструвала ефективність обраних технологічних рішень та архітектурних підходів. Клієнтська частина системи, розроблена за принципом Single Page Application з використанням фреймворку React.js, забезпечує інтуїтивно зрозумілий користувацький інтерфейс з високою продуктивністю та гнучкістю. Застосування модульного підходу дозволило створити масштабовану систему, здатну адаптуватися до різних платформ та сценаріїв використання, а включення механізмів аутентифікації на базі JWT-токенів гарантує безпеку доступу до функцій системи.

Розроблені алгоритми розпізнавання домашніх тварин базуються на модифікованій архітектурі MobileNetV2, оптимізованій для роботи на пристроях з обмеженими обчислювальними ресурсами. Квантизація моделі до 8-бітної точності значно зменшила вимоги до пам'яті та підвищила швидкість обчислень, зберігаючи при цьому високу точність розпізнавання. Впровадження механізмів темпорального аналізу та адаптивного налаштування порогів прийняття рішень забезпечило стабільну роботу системи в різних умовах освітлення та при різноманітних сценаріях використання. Тестування системи на реальних даних підтвердило її здатність розпізнавати різні види домашніх тварин з високою точністю навіть при обмежених обчислювальних ресурсах ESP32.

Інтеграційні можливості розробленої системи з популярними платформами розумного будинку, такими як Home Assistant, Apple HomeKit, Google Home та

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		69

Amazon Alexa, значно розширюють функціональність та зручність використання пристрою. Підтримка різних протоколів комунікації (MQTT, REST API, WebSocket) забезпечує гнучкість при взаємодії з іншими компонентами екосистеми розумного будинку та дозволяє реалізувати складні сценарії автоматизації. Реалізація системи навчання моделі розпізнавання через користувацький інтерфейс та механізми віддаленого оновлення програмного забезпечення спрощують процес обслуговування та адаптації системи до конкретних потреб користувачів, забезпечуючи високу практичну цінність розробленого рішення.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		70

## ВИСНОВКИ

Проведений аналіз існуючих засобів автоматизованого доступу для домашніх тварин у розумному будинку дозволив встановити основні тенденції розвитку та ключові технологічні рішення в даній галузі. Дослідження технологій розпізнавання об'єктів показало перспективність застосування згорткових нейронних мереж та їх гібридних модифікацій для ідентифікації домашніх тварин. Порівняльний аналіз архітектур CNN, CNN-SVM та CNN-LSTM продемонстрував можливість досягнення високих показників точності розпізнавання навіть на обмежених обчислювальних платформах, таких як ESP32 CAM. Визначено оптимальні параметри попередньої обробки зображень для підвищення ефективності роботи алгоритмів комп'ютерного зору та зменшення вимог до апаратних ресурсів.

Розгляд концепції розумного будинку та його підсистем виявив значний потенціал для інтеграції засобів автоматизованого доступу для домашніх тварин у загальну екосистему автоматизації. Встановлено, що оптимальною архітектурою побудови такої системи є гібридна модель, яка поєднує переваги централізованого та розподіленого підходів. Така архітектура забезпечує високу надійність, відмовостійкість та масштабованість при збереженні прийняттого рівня енергоефективності та швидкодії. Визначено, що технологія Wi-Fi, реалізована в ESP32 CAM, надає оптимальне співвідношення швидкості передачі даних, дальності зв'язку та складності розгортання для створення автономної підсистеми контролю доступу. Аналіз безпекових аспектів функціонування розумного будинку визначив необхідність локалізації обробки даних без передачі відеопотоку на зовнішні сервери для забезпечення конфіденційності та зниження ризиків несанкціонованого доступу.

Проаналізовані дослідження та технічні рішення створюють теоретичне підґрунтя для розробки програмно-технічного засобу автоматизованого доступу для домашніх тварин з використанням ESP32 CAM. Оптимальною стратегією реалізації такого рішення визначено застосування квантованих моделей нейронних

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		71

мереж з попередньою обробкою зображень для зменшення обчислювального навантаження при збереженні прийняттого рівня точності класифікації. Впровадження принципів модульності та сервісно-орієнтованої архітектури забезпечує гнучкість конфігурації системи та можливість її поетапного розширення з інтеграцією до існуючої інфраструктури розумного будинку. Реалізація системи на базі ESP32 CAM дозволяє досягти оптимального балансу між функціональністю, енергоефективністю та вартістю, що відкриває перспективи для масового впровадження даної технології.

Проектування програмно-технічного засобу автоматизованого доступу для домашніх тварин з використанням технології розпізнавання об'єктів на базі ESP32 CAM дозволило створити комплексне рішення, що відповідає сучасним вимогам розумного будинку. Розроблена структура системи включає п'ять ключових підсистем: сенсорну, обчислювальну, виконавчу, комунікаційну та енергозабезпечення, які забезпечують повний цикл функціонування автоматизованого доступу. Модульний підхід до проектування значно підвищує гнучкість системи, спрощує процес розробки, тестування та розширення функціональності, що є критично важливим для інтеграції з різноманітними екосистемами розумного будинку.

Електрична принципова схема, розроблена в середовищі EasyEDA, забезпечує оптимальне функціонування всіх компонентів системи з урахуванням особливостей взаємодії між ними. Центральний елемент системи – модуль ESP32-CAM – інтегрований з іншими компонентами з дотриманням вимог до енергоефективності, електромагнітної сумісності та надійності. Особлива увага приділена системі резервного живлення на базі літій-іонного акумулятора з контролером заряду TP4056, що забезпечує стабільну роботу пристрою навіть при відключенні зовнішнього живлення. Розроблена монтажна схема систематизує процес фізичної реалізації системи, враховуючи аспекти теплового режиму, захисту від зовнішніх впливів та оптимального розташування компонентів.

Аналіз обраних апаратних та програмних рішень підтвердив їх відповідність поставленим вимогам та доцільність використання в даному проекті. Вибір модуля

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		72

ESP32-CAM як центрального елемента системи обґрунтований оптимальним співвідношенням функціональності, енергоефективності та вартості порівняно з альтернативними рішеннями. Розроблена архітектура програмного забезпечення з асинхронним підходом до обробки подій дозволяє ефективно використовувати обчислювальні ресурси ESP32 для одночасного виконання задач розпізнавання образів, керування периферійними пристроями та мережевої комунікації. Використання протоколу MQTT для обміну повідомленнями забезпечує надійну інтеграцію з екосистемою розумного будинку при мінімальних накладних витратах. Програмна реалізація програмно-технічного засобу автоматизованого доступу для домашніх тварин на базі ESP32 CAM продемонструвала ефективність обраних технологічних рішень та архітектурних підходів. Клієнтська частина системи, розроблена за принципом Single Page Application з використанням фреймворку React.js, забезпечує інтуїтивно зрозумілий користувацький інтерфейс з високою продуктивністю та гнучкістю. Застосування модульного підходу дозволило створити масштабовану систему, здатну адаптуватися до різних платформ та сценаріїв використання, а включення механізмів аутентифікації на базі JWT-токенів гарантує безпеку доступу до функцій системи.

Розроблені алгоритми розпізнавання домашніх тварин базуються на модифікованій архітектурі MobileNetV2, оптимізованій для роботи на пристроях з обмеженими обчислювальними ресурсами. Квантизація моделі до 8-бітної точності значно зменшила вимоги до пам'яті та підвищила швидкість обчислень, зберігаючи при цьому високу точність розпізнавання. Впровадження механізмів темпорального аналізу та адаптивного налаштування порогів прийняття рішень забезпечило стабільну роботу системи в різних умовах освітлення та при різноманітних сценаріях використання. Тестування системи на реальних даних підтвердило її здатність розпізнавати різні види домашніх тварин з високою точністю навіть при обмежених обчислювальних ресурсах ESP32.

Інтеграційні можливості розробленої системи з популярними платформами розумного будинку, такими як Home Assistant, Apple HomeKit, Google Home та Amazon Alexa, значно розширюють функціональність та зручність використання

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		73

пристрою. Підтримка різних протоколів комунікації (MQTT, REST API, WebSocket) забезпечує гнучкість при взаємодії з іншими компонентами екосистеми розумного будинку та дозволяє реалізувати складні сценарії автоматизації. Реалізація системи навчання моделі розпізнавання через користувацький інтерфейс та механізми віддаленого оновлення програмного забезпечення спрощують процес обслуговування та адаптації системи до конкретних потреб користувачів, забезпечуючи високу практичну цінність розробленого рішення.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		74

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Ahlawat S., Choudhary A. Hybrid CNN-SVM classifier for handwritten digit recognition. *International Conference on Computational Intelligence and Data Science*. 2020. Vol. 167. P. 2554–2560.
2. Alam M. U., Tiwari R., Sharma J. Design and implementation of smart home automation system using LabVIEW. *International Journal for Scientific Research and Development*. 2020. Vol. 8, № 5. P. 515–520.
3. Benedetti D., Maselli G. Robust RFID Tag Identification. *Sensors*. 2022. Vol. 22, № 21. 8406. DOI: <https://doi.org/10.3390/s22218406>. (Дата звернення 15.05.2025)
4. Castellanos P. ESP32-CAM Security Bot. 2023. URL: <https://gitlab.com/pacastega/esp32-cam-security-bot>. (Дата звернення 15.05.2025)
5. MIT. Cyber-Physical Systems and Smart Homes, MIT Press. 2023. URL: <https://www.frontiersin.org/journals/the-internet-of-things/articles/10.3389/friot.2023.1275080/full> (Дата звернення 15.05.2025)
6. Cheng H.-R. C., Cao G.-Z., Li C.-H., Zhu A., Zhang X. A CNN-LSTM hybrid model for ankle joint motion recognition method based on sEMG. *International Conference on Ubiquitous Robots (UR)*. Kyoto, Japan, 2020.
7. El-Azab R. Smart homes: potentials and challenges. *Clean Energy*. 2021. Vol. 5, № 2. P. 302–315.
8. Ismail M. Towards data science - convolutional neural networks' mathematics. 2020. URL: <https://towardsdatascience.com/convolutional-neural-networks-mathematics-1beb3e6447c0>. (Дата звернення 15.05.2025)
9. TensorFlow Lite for ESP32-CAM // TensorFlow Blog. 2023. URL: <https://ela.kpi.ua/items/a41c04d1-f3be-47a2-9a61-225d04253f33> (Дата звернення 15.05.2025)
10. Jaihar J., Lingayat N., Vijaybhai P. S., Ventakesh G., Upla K. P. Smart home automation using machine learning algorithms. *International Conference for Emerging Technology (INCET)*. Belgaum, India, 2020.

					КВРКІ 220020.22.01.29 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		75

11. Jena S. R., Shanmugam R., Tyagi A., Patjoshi R. K. Implementation of voice recognized LED using Arduino Uno Micro. *International Journal of Future Generation Communication and Networking*. 2020. Vol. 13, № 3. P. 2233–2241.
12. Khan M., Seo J., Kim D. Towards energy efficient home automation: a deep learning approach. *Sensors*. 2020. Vol. 20, № 24. 7187.
13. Khan S., Javed M. H., Ahmed E., Shah S. A. A., Ali S. U. Facial recognition using convolutional neural networks and implementation on smart glasses. *International Conference on Information Science and Communication Technology (ICISCT)*. Karachi, Pakistan, 2019.
14. ESP32-CAM Camera Web Server // Arduino Project Hub. 2023. URL: <https://www.sciencedirect.com/science/article/pii/S1877050923021592>. (Дата звернення 15.05.2025)
15. Li Y., Wang T., Kang B. et al. Overcoming classifier imbalance for long-tail object detection with balanced group softmax. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Seattle, WA, USA, 2020.
16. Liciotti D., Bernardini M., Romeo L., Frontoni E. A sequential deep learning application for recognising human activities in smart homes. *Neurocomputing*. 2020. Vol. 396. P. 501–513.
17. Majeed R., Abdullah N. A., Ashraf I., Zikria Y. B. An intelligent, secure, and smart home automation system. *Scientific Programming*. 2020. Vol. 2020. 14 p.
18. Matthews C. Best Smart Doorbells of 2022: Home Security for Any Budget. 2022. URL: <https://www.gearhungry.com/best-smart-doorbells/>. (Дата звернення 15.05.2025)
19. Nardi T. DIY ESP32 Video Doorbell Locks Out Big Brother. 2020. URL: <https://hackaday.com/2020/12/22/diy-esp32-video-doorbell-locks-out-big-brother/>. (Дата звернення 15.05.2025)
20. Oksien J. The best video doorbell for 2023: if you like it should you put a Ring on it? 2023. URL: <https://www.techradar.com/best/best-video-doorbells>. (Дата звернення 15.05.2025)

					КВРКІ 220020.22.01.29 ПЗ	Арк. 76
Зм.	Арк.	№ докум.	Підпис	Дата		











**Додаток Г**  
**(обов'язковий)**

**ПРОГРАМНИЙ КОД**

```
#include <Arduino.h>
#include <WiFi.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>
#include <SPIFFS.h>
#include <esp_camera.h>
#include <TensorFlowLite_ESP32.h>
#include "tensorflow/lite/micro/all_ops_resolver.h"
#include "tensorflow/lite/micro/micro_error_reporter.h"
#include "tensorflow/lite/micro/micro_interpreter.h"
#include "tensorflow/lite/schema/schema_generated.h"
#include "model_data.h"
#include "camera_config.h"
#include "door_controller.h"
#include "motion_detector.h"
#include "system_config.h"

// Статичні буфери для TensorFlow Lite
constexpr size_t kTensorArenaSize = 150 * 1024;
static          uint8_t          tensor_arena[kTensorArenaSize]
__attribute__((aligned(16)));

// Об'єкти TensorFlow Lite
tflite::MicroErrorReporter micro_error_reporter;
```

```

tflite::ErrorReporter*          error_reporter          =
&micro_error_reporter;
const tflite::Model* model = nullptr;
tflite::MicroInterpreter* interpreter = nullptr;
TfLiteTensor* input_tensor = nullptr;
TfLiteTensor* output_tensor = nullptr;

// Налаштування системи
SystemConfig system_config;

// Історія розпізнавання для темпорального аналізу
constexpr uint8_t kHistorySize = 5;
float recognition_history[kHistorySize][NUM_CLASSES];
uint8_t history_index = 0;

// Змінні для керування енергоспоживанням
uint32_t last_motion_time = 0;
constexpr uint32_t kSleepDelayMs = 30000;. 30 секунд після
останнього руху
bool deep_sleep_enabled = true;

// Буфер для обробки зображення
uint8_t* resized_image_buffer = nullptr;

// WiFi та MQTT клієнти
WiFiClient espClient;
PubSubClient mqttClient(espClient);

/**
 * @brief Ініціалізація модуля розпізнавання

```

```

* @return true у випадку успішної ініціалізації, false - у
випадку помилки
*/
bool initializeRecognition() {
    log_i("Ініціалізація модуля розпізнавання...");

    . Ініціалізація SPIFFS для зберігання конфігурації та
    моделі
    if (!SPIFFS.begin(true)) {
        log_e("Помилка при ініціалізації SPIFFS");
        return false;
    }

    . Завантаження конфігурації системи
    if (!system_config.load()) {
        log_w("Не вдалося завантажити конфігурацію,
    використовуються налаштування за замовчуванням");
        system_config.setDefaults();
        system_config.save();
    }

    . Ініціалізація камери
    camera_config_t camera_config = getDefaultCameraConfig();
    esp_err_t err = esp_camera_init(&camera_config);
    if (err != ESP_OK) {
        log_e("Помилка ініціалізації камери: %d", err);
        return false;
    }
}

```

. Налаштування параметрів камери для оптимальної роботи в різних умовах освітлення

```
sensor_t* sensor = esp_camera_sensor_get();
if (sensor) {
    sensor->set_framesize(sensor, FRAMESIZE_QVGA);
    sensor->set_quality(sensor, 10);.    Балансування
якості та розміру зображення
    sensor->set_brightness(sensor, 1);
    sensor->set_contrast(sensor, 1);
    sensor->set_saturation(sensor, 0);
    sensor->set_whitebal(sensor, 1);
    sensor->set_awb_gain(sensor, 1);
    sensor->set_wb_mode(sensor, 0);
    sensor->set_exposure_ctrl(sensor, 1);
    sensor->set_aec2(sensor, 1);
    sensor->set_ae_level(sensor, 0);
    sensor->set_aec_value(sensor, 400);
    sensor->set_gain_ctrl(sensor, 1);
    sensor->set_agc_gain(sensor, 0);
    sensor->set_gainceiling(sensor, (gainceiling_t)6);
    sensor->set_bpc(sensor, 1);
    sensor->set_wpc(sensor, 1);
    sensor->set_raw_gma(sensor, 1);
    sensor->set_lenc(sensor, 1);
    sensor->set_hmirror(sensor, 0);
    sensor->set_vflip(sensor, 0);
    sensor->set_dcw(sensor, 1);
    sensor->set_colorbar(sensor, 0);
}
```

```

. Ініціалізація детектора руху
if (!initializeMotionDetector()) {
    log_e("Помилка ініціалізації детектора руху");
    return false;
}

. Ініціалізація контролера дверей
if (!initializeDoorController()) {
    log_e("Помилка ініціалізації контролера дверей");
    return false;
}

. Виділення пам'яті для буфера обробки зображення
resized_image_buffer = (uint8_t*)malloc(INPUT_WIDTH *
INPUT_HEIGHT * INPUT_CHANNELS);
if (resized_image_buffer == nullptr) {
    log_e("Не вдалося виділити пам'ять для буфера
зображення");
    return false;
}

. Завантаження моделі
model = tflite::GetModel(g_model_data);
if (model->version() != TFLITE_SCHEMA_VERSION) {
    log_e("Невідповідність версії моделі: %d != %d",
model->version(), TFLITE_SCHEMA_VERSION);
    return false;
}

. Ініціалізація інтерпретатора TensorFlow Lite

```

```

static tflite::AllOpsResolver resolver;
static tflite::MicroInterpreter static_interpreter(
    model, resolver, tensor_arena, kTensorArenaSize,
error_reporter);
    interpreter = &static_interpreter;

. Виділення тензорів
    TfLiteStatus allocate_status = interpreter-
>AllocateTensors();
    if (allocate_status != kTfLiteOk) {
        log_e("Помилка при виділенні тензорів: %d",
allocate_status);
        return false;
    }

. Отримання вказівників на вхідний та вихідний тензори
    input_tensor = interpreter->input(0);
    output_tensor = interpreter->output(0);

. Перевірка розміру вхідного тензора
    if (input_tensor->dims->size != 4 ||
        input_tensor->dims->data[0] != 1 ||
        input_tensor->dims->data[1] != INPUT_HEIGHT ||
        input_tensor->dims->data[2] != INPUT_WIDTH ||
        input_tensor->dims->data[3] != INPUT_CHANNELS) {
        log_e("Неправильні розміри вхідного тензора");
        return false;
    }

. Перевірка розміру вихідного тензора

```

```

    if (output_tensor->dims->size != 2 ||
        output_tensor->dims->data[0] != 1 ||
        output_tensor->dims->data[1] != NUM_CLASSES) {
        log_e("Неправильні розміри вихідного тензора");
        return false;
    }

    . Ініціалізація історії розпізнавання
    memset(recognition_history, 0,
sizeof(recognition_history));

    . З'єднання з WiFi
    WiFi.begin(system_config.getWiFiSSID(),
system_config.getWiFiPassword());

    . Налаштування MQTT клієнта
    mqttClient.setServer(system_config.getMqttBroker(),
system_config.getMqttPort());
    mqttClient.setCallback(onMqttMessage);

    log_i("Ініціалізація модуля розпізнавання завершена
успішно");
    return true;
}

/**
 * @brief Попередня обробка зображення перед розпізнаванням
 * @param frame Вихідний кадр з камери
 * @return true у випадку успішної обробки, false - у
випадку помилки

```

```

*/
bool preprocessImage(camera_fb_t* frame) {
    if (frame == nullptr) {
        log_e("Отримано пустий кадр");
        return false;
    }

    . Перетворення формату зображення та зміна розміру
    if (frame->format == PIXFORMAT_JPEG) {
        . Розпакування JPEG
        if (!decodeJpeg(frame->buf, frame->len,
resized_image_buffer, INPUT_WIDTH, INPUT_HEIGHT)) {
            log_e("Помилка при розпакуванні JPEG");
            return false;
        }
    } else if (frame->format == PIXFORMAT_RGB565) {
        . Перетворення RGB565 в RGB888 та зміна розміру
        if (!resizeRgb565(frame->buf, frame->width, frame-
>height,
                                resized_image_buffer, INPUT_WIDTH,
INPUT_HEIGHT)) {
            log_e("Помилка при зміні розміру RGB565");
            return false;
        }
    } else {
        log_e("Непідтримуваний формат зображення: %d",
frame->format);
        return false;
    }
}

```

```

. Нормалізація та перетворення кольорової моделі
for (int i = 0; i < INPUT_WIDTH * INPUT_HEIGHT; i++) {
    . RGB -> YUV (спрощена формула)
    float r = resized_image_buffer[i * 3];
    float g = resized_image_buffer[i * 3 + 1];
    float b = resized_image_buffer[i * 3 + 2];

    . Конвертація в YUV
    float y = 0.299f * r + 0.587f * g + 0.114f * b;
    float u = 0.492f * (b - y);
    float v = 0.877f * (r - y);

    . Нормалізація до діапазону [-1, 1]
    float normalized_y = (y / 255.0f) * 2.0f - 1.0f;
    float normalized_u = u / 128.0f;
    float normalized_v = v / 128.0f;

    . Заповнення вхідного тензора
    input_tensor->data.f[i * 3] = normalized_y;
    input_tensor->data.f[i * 3 + 1] = normalized_u;
    input_tensor->data.f[i * 3 + 2] = normalized_v;
}

return true;
}

/**
 * @brief Виконання інференсу моделі та отримання
результатів розпізнавання

```

```

* @param[out] results Масив з ймовірностями для кожного
класу
* @return true у випадку успішного розпізнавання, false - у
випадку помилки
*/
bool runInference(float results[NUM_CLASSES]) {
    . Запуск інференсу
    TfLiteStatus invoke_status = interpreter->Invoke();
    if (invoke_status != kTfLiteOk) {
        log_e("Помилка при виконанні інференсу: %d",
invoke_status);
        return false;
    }

    . Копіювання результатів
    for (int i = 0; i < NUM_CLASSES; i++) {
        results[i] = output_tensor->data.f[i];
    }

    return true;
}

/**
* @brief Аналіз результатів розпізнавання з використанням
темпоральної фільтрації
* @param current_results Поточні результати розпізнавання
* @param[out] ret_id Ідентифікатор розпізнаної тварини (або
-1, якщо не розпізнано)
* @param[out] confidence Впевненість у розпізнаванні

```

```

* @return true, якщо тварина розпізнана з достатньою
впевненістю, false - інакше
*/
bool          analyzeResults(const          float
current_results[NUM_CLASSES],
          int& pet_id, float& confidence) {
    . Збереження поточних результатів в історії
    memcpy(recognition_history[history_index],
current_results, sizeof(float) * NUM_CLASSES);
    history_index = (history_index + 1) % kHistorySize;

    . Експоненційне згладжування результатів
    float smoothed_results[NUM_CLASSES] = {0.0f};
    constexpr float alpha = 0.7f; . Коефіцієнт згладжування

    for (int i = 0; i < kHistorySize; i++) {
        float weight = pow(alpha, (kHistorySize - 1 - i) %
kHistorySize);
        for (int j = 0; j < NUM_CLASSES; j++) {
            int idx = (history_index - 1 - i +
kHistorySize) % kHistorySize;
            smoothed_results[j] +=
recognition_history[idx][j] * weight;
        }
    }

    . Нормалізація результатів
    float sum = 0.0f;
    for (int i = 0; i < NUM_CLASSES; i++) {
        sum += smoothed_results[i];
    }
}

```

```

}

if (sum > 0.0f) {
    for (int i = 0; i < NUM_CLASSES; i++) {
        smoothed_results[i] /= sum;
    }
}

```

. Пошук класу з максимальною ймовірністю

```

float max_probability = 0.0f;
int max_class = -1;

for (int i = 0; i < NUM_CLASSES; i++) {
    if (smoothed_results[i] > max_probability) {
        max_probability = smoothed_results[i];
        max_class = i;
    }
}

```

. Визначення порогу впевненості на основі умов освітлення

```

float confidence_threshold =
system_config.getBaseConfidenceThreshold();
sensor_t* sensor = esp_camera_sensor_get();
if (sensor) {

```

```

    int current_gain = sensor->get_agc_gain(sensor);
    int current_exposure = sensor->get_aec_value(sensor);

```

. Адаптивне налаштування порогу для складних умов освітлення

```

    if (current_gain > 8 && current_exposure > 800) {

```

```

        . Низька освітленість, знижуємо поріг
        confidence_threshold *= 0.85f;
    } else if (current_gain < 2 && current_exposure <
200) {
        . Високий контраст або пересвічення, підвищуємо
поріг
        confidence_threshold *= 1.15f;
    }
}

. Запис результатів
pet_id = max_class;
confidence = max_probability;

. Перевірка на достатню впевненість
return (max_class >= 0 && max_probability >=
confidence_threshold);
}

/**
 * @brief Обробка розпізнаної тварини та прийняття рішення
про доступ
 * @param pet_id Ідентифікатор розпізнаної тварини
 * @param confidence Впевненість у розпізнаванні
 */
void handleRecognizedPet(int pet_id, float confidence) {
    . Перевірка дозволу доступу для даної тварини
    if (system_config.isPetAllowed(pet_id)) {
        log_i("Дозволено доступ для тварини ID %d
(впевненість: %.2f)", pet_id, confidence);

```

```

. Відкриття дверей
openDoor();

. Запис події у журнал
logAccessEvent(pet_id, true, confidence);

. Відправлення сповіщення через MQTT
if (mqttClient.connected()) {
    DynamicJsonDocument doc(256);
    doc["event"] = "access_granted";
    doc["pet_id"] = pet_id;
    doc["confidence"] = confidence;
    doc["timestamp"] = millis();

    String message;
    serializeJson(doc, message);

mqttClient.publish(system_config.getMqttTopicEvents(),
message.c_str());
    }
    } else {
        log_i("Заборонено доступ для тварини ID %d
(впевненість: %.2f)", pet_id, confidence);

. Сигналізація про відмову
indicateAccessDenied();

. Запис події у журнал
logAccessEvent(pet_id, false, confidence);

```

```

. Відправлення сповіщення через MQTT
if (mqttClient.connected()) {
    DynamicJsonDocument doc(256);
    doc["event"] = "access_denied";
    doc["pet_id"] = pet_id;
    doc["confidence"] = confidence;
    doc["timestamp"] = millis();

    String message;
    serializeJson(doc, message);

mqttClient.publish(system_config.getMqttTopicEvents(),
message.c_str());
    }
}

/**
 * @brief Головний цикл розпізнавання
 */
void recognitionLoop() {
    . Перевірка з'єднання з MQTT брокером
    if (!mqttClient.connected() && WiFi.status() ==
WL_CONNECTED) {
        reconnectMqtt();
    }

    mqttClient.loop();
}

```

```

. Перевірка наявності руху
bool motion_detected = detectMotion();

if (motion_detected) {
    . Зберегти час останнього руху
    last_motion_time = millis();

    . Активація підсвічування при необхідності
    sensor_t* sensor = esp_camera_sensor_get();
    if (sensor) {
        int current_gain = sensor->get_agc_gain(sensor);
        if (current_gain > 6) {
            . Низька освітленість, вмикаємо підсвічування
            enableIllumination(true);
        }
    }

    . Захоплення зображення
    camera_fb_t* frame = esp_camera_fb_get();
    if (frame) {
        . Попередня обробка зображення
        if (preprocessImage(frame)) {
            . Запуск інференсу
            float results[NUM_CLASSES];
            if (runInference(results)) {
                . Аналіз результатів
                int pet_id;
                float confidence;
                if (analyzeResults(results, pet_id,
confidence)) {

```

```

        . Обробка розпізнаної тварини
        handleRecognizedPet (pet_id,
confidence);

        } else {
            log_i("Тварина не розпізнана або
низька впевненість (%.2f)", confidence);
        }
    }

    . Вимкнення підсвічування
    enableIllumination(false);

    . Звільнення буфера кадру
    esp_camera_fb_return(frame);
}
} else {
    . Перевірка необхідності переходу в режим глибокого
сну
    if (deep_sleep_enabled &&
        (millis() - last_motion_time > kSleepDelayMs)) {
        log_i("Перехід у режим глибокого сну");

        . Завершення відкритих з'єднань
        if (mqttClient.connected()) {
            mqttClient.disconnect();
        }

        . Налаштування пробудження за GPIO (датчик руху)

```

```

        esp_sleep_enable_ext0_wakeup(MOTION_SENSOR_PIN,
HIGH);

        . Перехід у режим глибокого сну
        esp_deep_sleep_start();
    }
}

/**
 * @brief Обробник отримання MQTT повідомлень
 */
void onMqttMessage(char* topic, byte* payload, unsigned int
length) {
    . Перетворення payload в рядок для обробки
    char message[length + 1];
    memcpy(message, payload, length);
    message[length] = '\0';

    log_i("Отримано MQTT повідомлення: %s", message);

    . Розбір JSON повідомлення
    DynamicJsonDocument doc(512);
    DeserializationError error = deserializeJson(doc,
message);

    if (error) {
        log_e("Помилка розбору JSON: %s", error.c_str());
        return;
    }
}

```

. Обробка команд

```
if (doc.containsKey("command")) {
    String command = doc["command"].as<String>();

    if (command == "restart") {
        log_i("Команда перезавантаження");
        ESP.restart();
    } else if (command == "open_door") {
        log_i("Команда відкриття дверей");
        openDoor();
    } else if (command == "close_door") {
        log_i("Команда закриття дверей");
        closeDoor();
    } else if (command == "update_config") {
        log_i("Команда оновлення конфігурації");

        if (doc.containsKey("config")) {
            JsonObject config = doc["config"];

            . Оновлення конфігурації
            if (config.containsKey("threshold")) {

system_config.setBaseConfidenceThreshold(config["threshold"]
.as<float>());
            }

            if (config.containsKey("deep_sleep")) {
                deep_sleep_enabled =
config["deep_sleep"].as<bool>();
            }
        }
    }
}
```

```

    }

    if (config.containsKey("allowed_pets")) {
        JSONArray allowed_pets =
config["allowed_pets"];
        system_config.clearAllowedPets();

        for (JsonVariant pet : allowed_pets) {
system_config.addAllowedPet(pet.as<int>());
        }
    }

    . Збереження конфігурації
    system_config.save();

    . Підтвердження оновлення
    if (mqttClient.connected()) {
        DynamicJsonDocument response(256);
        response["event"] = "config_updated";
        response["status"] = "success";
        response["timestamp"] = millis();

        String response_message;
        serializeJson(response,
response_message);

mqttClient.publish(system_config.getMqttTopicEvents(),
response_message.c_str());
    }

```

```

        }
    } else if (command == "capture_image") {
        . Захоплення та відправлення зображення
        captureAndSendImage();
    } else if (command == "status") {
        . Відправлення поточного статусу системи
        sendSystemStatus();
    }
}
}

/**
 * @brief Підключення до MQTT брокера
 */
void reconnectMqtt() {
    if (WiFi.status() != WL_CONNECTED) {
        return;
    }

    . Генерація унікального ідентифікатора клієнта
    String clientId = "ESP32CAM-";
    clientId += String(ESP.getEfuseMac(), HEX);

    log_i("Підключення до MQTT брокера...");

    . Спроба підключення
    if (mqttClient.connect(clientId.c_str(),
                           system_config.getMqttUsername(),
                           system_config.getMqttPassword())) {
        log_i("Підключено до MQTT брокера");
    }
}

```

```

        . Підписка на командний топик

mqttClient.subscribe(system_config.getMqttTopicCommands());

        . Відправлення повідомлення про підключення
        DynamicJsonDocument doc(256);
        doc["event"] = "connected";
        doc["device_id"] = clientId;
        doc["firmware_version"] = FIRMWARE_VERSION;
        doc["timestamp"] = millis();

        String message;
        serializeJson(doc, message);

mqttClient.publish(system_config.getMqttTopicEvents(),
message.c_str());
    } else {
        log_e("Не вдалося підключитися до MQTT брокера, код
помилки: %d", mqttClient.state());
    }
}

/**
 * @brief Захоплення та відправлення зображення через MQTT
 */
void captureAndSendImage() {
    log_i("Захоплення зображення для відправлення");

        . Вмикання підсвічування при необхідності

```

```

sensor_t* sensor = esp_camera_sensor_get();
if (sensor) {
    int current_gain = sensor->get_agc_gain(sensor);
    if (current_gain > 6) {
        enableIllumination(true);
    }
}

```

. Захоплення зображення

```

camera_fb_t* frame = esp_camera_fb_get();
if (frame) {

```

. Відправка через MQTT (з розбивкою на частини, якщо потрібно)

```

    if (mqttClient.connected()) {
        const size_t max_chunk_size = 4096;.

```

Максимальний розмір частини

```

        size_t remaining = frame->len;
        size_t offset = 0;
        int chunk_index = 0;
        int total_chunks = (frame->len + max_chunk_size
- 1) / max_chunk_size;

```

. Відправлення метаданих зображення

```

DynamicJsonDocument metadata(256);
metadata["event"] = "image_metadata";
metadata["total_size"] = frame->len;
metadata["chunk_size"] = max_chunk_size;
metadata["total_chunks"] = total_chunks;
metadata["format"] = frame->format;
metadata["width"] = frame->width;

```

```

metadata["height"] = frame->height;
metadata["timestamp"] = millis();

String metadata_message;
serializeJson(metadata, metadata_message);

mqttClient.publish(system_config.getMqttTopicImageMetadata()
, metadata_message.c_str());

```

. Відправлення частин зображення

```

while (remaining > 0) {
    size_t chunk_size = (remaining >
max_chunk_size) ? max_chunk_size : remaining;

```

. Підготовка повідомлення з частиною зображення

```

DynamicJsonDocument chunk_metadata(128);
chunk_metadata["chunk_index"] = chunk_index;
chunk_metadata["chunk_size"] = chunk_size;

String chunk_topic =
system_config.getMqttTopicImageChunk();
chunk_topic += "/" + String(chunk_index);

String chunk_metadata_message;
serializeJson(chunk_metadata,
chunk_metadata_message);
mqttClient.publish(chunk_topic.c_str(),
chunk_metadata_message.c_str());

```

```

        . Відправлення даних частини
        mqttClient.beginPublish(chunk_topic.c_str(),
chunk_size, false);
        mqttClient.write(frame->buf + offset,
chunk_size);
        mqttClient.endPublish();

        offset += chunk_size;
        remaining -= chunk_size;
        chunk_index++;

        . Невелика затримка для стабільності
        delay(10);
    }

    . Повідомлення про завершення відправки
    DynamicJsonDocument completion(128);
    completion["event"] = "image_complete";
    completion["timestamp"] = millis();

    String completion_message;
    serializeJson(completion, completion_message);

mqttClient.publish(system_config.getMqttTopicImageComplete()
, completion_message.c_str());
    } else {
        log_e("Не вдалося відправити зображення -
відсутнє підключення до MQTT");
    }

```

```

        . Звільнення буфера кадру
        esp_camera_fb_return(frame);
    } else {
        log_e("Не вдалося захопити зображення");
    }

    . Вимкнення підсвічування
    enableIllumination(false);
}

/**
 * @brief Відправлення поточного статусу системи
 */
void sendSystemStatus() {
    if (!mqttClient.connected()) {
        return;
    }

    DynamicJsonDocument status(512);
    status["event"] = "status";
    status["uptime"] = millis();
    status["free_heap"] = ESP.getFreeHeap();
    status["wifi_rssi"] = WiFi.RSSI();
    status["wifi_ssid"] = WiFi.SSID();
    status["deep_sleep_enabled"] = deep_sleep_enabled;
    status["confidence_threshold"] =
system_config.getBaseConfidenceThreshold();
}

```

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА  
ІНФОРМАЦІЙНИХ СИСТЕМ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Назва кваліфікаційної роботи Програмно-технічний засіб автоматизованого доступу для домашніх тварин у кіберфізичній системі "Розумний будинок" на основі виявлення об'єктів за допомогою ESP32 SAM

Автор Олексій КУХАР

Освітня програма Комп'ютерна інженерія та програмування

Рівень вищої освіти перший (бакалаврський) рівень

Спеціальність 123– Комп'ютерна інженерія

Науковий керівник: к.т.н., доцент Андрій НІЧЕПОРУК

На основі аналізу кваліфікаційної роботи на дотримання вимог академічної доброчесності (у т.ч. відсутності ознак академічного плагіату) з урахуванням результатів перевірки роботи спеціалізованим програмним засобом(ами) комісія зробила такий висновок:

№	Висновок	Позначка про відповідність
1	Ознаки академічного плагіату	
1.1	Запозичення, виявлені в роботі, є законними і не є академічним плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних, якщо потрібно). Робота приймається до захисту.	Відповідає
1.2	Виявлені запозичення не є академічним плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована.	
1.3	Виявлені запозичення не є академічним плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота може бути допущена до захисту після того як буде відкоригована та доопрацьована і успішно пройде повторну перевірку на академічний плагіат.	
1.4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
2	Інші види порушень академічної доброчесності	Не виявлено

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

1) запозичення розміщені в розділах аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;

2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості StrikePlagiarism, складає 4.78% і адресується до 14 першоджерела; та системою Anti-Plagiarism складає 0%, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КПС

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

Андрій НІЧЕПОРУК

Андрій НІЧЕПОРУК

Ольга ПАВЛОВА

## РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Кухар Олексій Вікторович

**Тема:** Програмно-технічний засіб автоматизованого доступу для домашніх тварин у кіберфізичній системі “Розумний будинок” на основі виявлення об’єктів за допомогою ESP32 CAM

Спеціальність: 123 «Комп’ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість листів креслень 4 Кількість сторінок записки 84

1. Короткий зміст роботи та прийнятих рішень: Метою кваліфікаційної роботи Програмно-технічний засіб автоматизованого доступу для домашніх тварин у кіберфізичній системі “Розумний будинок” на основі виявлення об’єктів за допомогою ESP32 CAM

2. Висновок про відповідність роботи дипломному завданню: Робота повністю відповідає поставленому завданню.

3. Метою дослідження є розробка програмно-технічного засобу автоматизованого доступу для домашніх тварин у системі «Розумний дім» з використанням ESP32-CAM та алгоритмів комп’ютерного зору. Актуальність теми зумовлена зростанням кількості домашніх тварин і потребою в комфортному, безпечному та інтелектуальному керуванні їхнім доступом до приміщень. Існуючі рішення на базі RFID чи інфрачервоних датчиків мають обмежену надійність і не забезпечують селективного доступу. Запропонована система дозволяє здійснювати ідентифікацію тварин без використання додаткових міток, завдяки вбудованим камерам та алгоритмам розпізнавання на основі глибинного навчання. Об’єктом дослідження є процес автоматизованого доступу домашніх тварин у розумному будинку. Предметом дослідження – методи розпізнавання об’єктів та керування виконавчими пристроями (електрозамками, сервоприводами) на базі модуля ESP32-CAM.

Під час роботи проведено аналіз існуючих технологій розпізнавання об'єктів, визначено оптимальну архітектуру програмно-апаратного комплексу, розроблено ефективні алгоритми комп'ютерного зору з урахуванням обмежених ресурсів ESP32-CAM, а також здійснено експериментальну перевірку працездатності прототипу в реальних умовах. Практичне значення полягає в створенні доступного, компактного та ефективного засобу автоматизації побутових процесів, який може бути легко інтегрований у сучасні системи «Розумного дому».

4. Позитивні сторони роботи: висока практична цінність роботи.

5. Негативні сторони роботи: недостатня увага моделюванню схеми автомату в середовищі Quartus II.

6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно, згідно діючих стандартів оформлення документації.

7. Відгук про роботу в цілому: Робота виконана на належному науково-технічному рівні.

8. Інші зауваження: \_\_\_\_\_

9. Оцінка дипломної роботи: добре

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) \_\_\_\_\_

К. М. Н., доцент кафедри автоматизації комп'ютерно-  
-інженерівачих механізмів та робототехніки Федюк М. В.

« 9 » 06 2025 р. ФН (підпис)

Завідувачу кафедри КІС  
д-р. філософії, доц. Ользі ПАВЛОВІЙ

Олексія Кухара

ПІБ здобувача вищої освіти

ФІТ, 4 курсу, групи КІ2-22-1

### ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений(а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Strike-Plagiarism та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

05.06. 2025 року



## Протокол аналізу звіту подібності експертом

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

**Автор:** Олексій КУХАР

**Співавтор:**

**Назва:** Кухар\_ Програмно-технічний засіб автоматизованого доступу для домашніх тварин у кіберфізичній системі "Розумний будинок" на основі виявлення об'єктів за допомогою ESP32 CAM

**Експерт:**

**Підрозділ:** Кафедра комп'ютерної інженерії та інформаційних систем

**Коефіцієнт подібності 1:** 4.8%

**Коефіцієнт подібності 2:** 1.7%

**Мікропробіли:** 4

**Заміна букв:** 0

**Інтервали:** 0

**Білі знаки:** 0

**Дата створення звіту:** 2025-06-06 19:08:17.0

Після аналізу Звіту подібності констатую наступне:

Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.

Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.

Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедур. Таким чином робота не приймається.

Обґрунтування:

2025-06-07

Дата



Доцент Андрій Нічепорук

експерт

## Anti-Plagiarism (UA) v-15.281 Educational

**The maximum coincidence with one document 0.0%**

Dictionary check: en\_US, ru\_RU, ua\_UA. **Errors in the documents: 10%**

ID: 244063 Title: БКР Програмно-технічний засіб автоматизованого доступу для домашніх тварин у кіберфізичній системі “Розумний будинок” на основі виявлення об’єктів за допомогою ESP32 CAM Added in a DB: 2025-06-06 Authors: Олексій КУХАР Heads: Андрій НІЧЕПОРУК Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	118691	777	937 (1%)	14 (2%)

### Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes