Вінницький національний технічний університет
Факультет менеджменту та інформаційної безпеки
Кафедра менеджменту та безпеки інформаційних систем

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

на тему: Підвищення захищеності системи контролю доступу до приміщення з використанням face - ідентифікації користувача та віддаленого керування

Виконав: ст. 2-го курсу, групи УБ-20м
спеціальності 125– Кібербезпека
Освітня програма – Управління інформаційною безпекою_____
(шифр і назва напряму підготовки, спеціальності)
Кравчик В.В._____
(прізвище та ініціали)
Керівник: Голова секції УБ кафедри МБІС, професор, д.т.н.
Яремчук Ю.Є._____
(прізвище та ініціали)
«____» _____ 2021 р.
Опонент: к.т.н., доц., доцент каф. ОТ
Савицька Л.А._____
(прізвище та ініціали)
«____» _____ 2021 р.

Допущено до захисту

Голова секції УБ кафедри МБІС

д.т.н., проф. Яремчук Ю.Є.

“_____”_____2021 р.

Вінниця ВНТУ - 2021 рік

Вінницький національний технічний університет

Факультет менеджменту та інформаційної безпеки

Кафедра менеджменту та безпеки інформаційних систем

Освітньо-кваліфікаційний рівень магістр

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека

Освітньо-професійна програма - Управління інформаційною безпекою

ЗАТВЕРДЖУЮ
Голова секції УБ, кафедра МБІС

_____д.т.н., проф. Яремчук Ю.Є.
“_____”_____2021 р.

З А В Д А Н Н Я
на магістерську кваліфікаційну роботу студенту

_____
(прізвище, ім`я, по-батькові)

1. Тема роботи _____Підвищення захищеності системи контролю доступу до приміщення з використанням face - ідентифікації користувача та віддаленого керування_____
Керівник роботи _д.т.н., проф. Яремчук Юрій Євгенович_____
(прізвище, ім`я, по-батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “_24_” _вересня 2021 року
№__277__
2. Строк подання студентом роботи_____

3. Вихідні дані до роботи:_т е х н і ч н і с т а т т і п о т е м і , і н т е р н е т р е с у р с и , с т а н д а р т и , і с н у ю ч е п р о г р а м н е з а б е з п е ч е н н я_____

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): в першому розділі проаналізувати існуючі методи та засоби контролю доступу, а також розглянути представленні аналоги. В другому розділі здійснити розробку структурної схеми системи та блок схеми.

Здійснити вибір та обгрунтування елементної бази системи контролю та управління доступом також розробити принципову схему. В третьому розділі описати реалізацію та емуляцію схеми приладу. Реалізувати програмну частина, що відповідає за керування системою в цілому.Здійснити випробування системи контролю доступом з використанням розпізнава обличчя. В четвертому розділі здійснити розрахунок економічної доцільност.

5. Перелік графічного матеріалу (з точним зазначенням обов`язкових креслень)

у першому розділі наведено 7 рис.; у другому розділі наведено 7 рис.; у третьому розділі наведено 5 рис..

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Основна частина	Яремчук Ю.Є.		
Економічна частина	Адлер О.О.		

7.Дата видачі завдання 24 вересня 2021 р.

КАЛЕНДАРНИЙ ПЛАН

№	Назва та зміст етапу	Термін виконання		Примітка
		початок	закінчення	
1	Визначення напрямку магістерської кваліфікаційної роботи, формулювання теми			
2	Аналіз предметної області обраної теми			
3	Розробка алгоритму роботи			
4	Написання магістерської кваліфікаційної роботи на основі розробленої теми			
5	Передзахист магістерської кваліфікаційної роботи			
6	Виправлення, уточнення, корегування магістерської кваліфікаційної роботи			
7	Захист магістерської кваліфікаційної роботи			

Студент _____ _____Кравчик В.В._____

(підпис) (прізвище та ініціали)

Керівник магістерської кваліфікаційної роботи _____ ___Яремчук Ю.Є.____

(підпис) (прізвище та ін

АНОТАЦІЯ

В даній магістерській кваліфікаційній роботі «Підвищення захищеності системи контролю доступу до приміщення з використанням face - ідентифікації користувача та віддаленого керування» було розглянуто існуючі методи та засоби контролю доступу, здійснено огляд різних алгоритмів для розпізнавання обличчя представлених на даний момент та визначені основні їх недоліки. В ході виконання роботи, розроблена система контролю та управління доступом з використанням вдосконаленого алгоритму розпізнавання обличчя та здійснено розробку структурної та принципової схеми. Дана система контролю доступом була реалізована на практиці з можливістю віддаленого керування за допомогою месенджера Telegram і бота. Було здійснено тестування готової системи та отримано позитивний результат, щодо її роботи.

В результаті досліджень було підтверджено, що вдосконалений алгоритм та система на його основі краща за аналоги. Про що свідчать краща якість зображення, та більш точніше зіставляються відбитки обличчя з базою даних.

Ключові слова: контроль доступу, розпізнавання, камера, обличчя, ESP32.

SUMMARY

This master's thesis "Improving the security of access control to the premises using face - user identification and remote control" considered existing methods and tools for access control, reviewed various algorithms for face recognition presented at the moment and identified their main shortcomings. In the course of the work, an access control and management system was developed using an advanced face recognition algorithm and a structural and schematic diagram was developed. This access control system has been implemented in practice with the possibility of remote control using Telegram messenger and bot. The finished system was tested and a positive result was obtained regarding its operation.

As a result of research it was confirmed that the improved algorithm and the system based on it are better than analogues. As evidenced by better image quality, and more accurately compare facial prints with the database.

Key words: access control, recognition, camera, face, ESP32.

ЗМІСТ

ВСТУП 60
1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ТА ЗАСОБІВ КОНТРОЛЮ ДОСТУПУ 63
1.1Аналіз методів контрою доступу 63
1.2 Аналіз методів контролю доступу за допомогою розпізнавання обличчя 67
1.3 Аналіз існуючих засобів контролю доступу за допомогою розпізнавання обличчя. 71
1.4 Аналіз ефективності 2D технології розпізнавання облич 76
1.5 Висновки та постановка задачі 78
2. РОЗРОБКА ТА ПРОЕКТУВАННЯ ПРИСТРОЮ КОНТРОЛЮ ДОСТУПУ 80
2.1 Розробка структурної схеми 80
2.2 Вибір та обгрунтування елементної бази 86
2.3 Розробка принципової схеми 91
2.4 Висновки 94
3. РЕАЛІЗАЦІЯ ОХОРОННОЇ СИСИТЕМИ З ЕЛЕМЕНТОМ КОНТРОЛЮ ДОСТУПУ 94
3.1 Реалізація та емуляція схеми приладу 95
3.2 Реалізація програмної частини 99
3.3 Рекомендації з використання та налаштування приладу 104
3.4 Випробування прототипу приладу для контролю доступу до приміщення з використанням розпізнавання обличчя 105
3.5 Висновки 53
4. ЕКОНОМІЧНА ДОЦІЛЬНІСТЬ СТВОРЕННЯ ПРИЛАДУ ДЛЯ КОНТРОЛЮ ДОСТУПУ З ВИКОРИСТАННЯМ FACE-ІДЕНТИФІКАЦІЇ КОРИСТУВАЧА 54
4.1 Проведення наукового аудиту науково-дослідної роботи……...54
4.2 Проведення комерційного та технологічного аудиту науково- технічної розробки…… 57
4.3 Розрахунок витрат на здійснення науково-дослідної роботи 60

4.4 Розрахунок ефективності вкладених інвестицій та період їх окупності 68
4.5 Висновки до розділу..... 72
ВИСНОВКИ 73
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ..... 74
Додаток А ТЕХНІЧНЕ ЗАВДАННЯ..... 80
Додаток Б Перелік використаних елементів 84
Додаток В Лістинг програми 86
Додаток Г Схема електрична принципова 118
Додаток Д Ілюстраційний матеріал 120
Додаток Е Протокол перевірки навчальної (кваліфікаційної) робот……..121

ВСТУП

Останнім часом, проблеми безпеки все більше постають у різних сферах життя. Системи контролю і управління доступом (СКУД) міцно зайняли своє місце в переліку технічних систем безпеки. СКУД, це сукупність програмно-технічних засобів і чітко сформованої системи управління всім пересуванням персоналу. Захист будь-якого об'єкта включає кілька рубежів, число яких залежить від рівня режимності об'єкта. При цьому у всіх випадках важливим рубежем буде СКУД.

Добре організована з використанням сучасних технічних засобів система, дозволить вирішувати цілий ряд завдань. До числа найбільш важливих можна віднести наступні:

- протидія промисловому шпигунству;
- протидія розкраданню;
- протидія навмисному пошкодженню матеріальних цінностей;
- облік робочого часу;
- контроль своєчасності приходу і відходу співробітників;
- захист конфіденційності інформації;
- регулювання потоку відвідувачів;
- контроль в'їзду та виїзду транспорту.

При реалізації конкретних СКУД використовують різні способи і пристрої для ідентифікації і аутентифікації особистості. Наприклад:

– "FaceVACS" компанії "Cognitec Systems";
– "NEC's Face Recognition" компанії "NEC";
– "LUNA SDK" компанії "VisionLabs";
– "VeriLook SDK" компанії "Neurotechnology".

Актуальність представленої теми, полягає в тому, що в даний час, без системи контролю доступом вже не може обійтися практично ні одна організація. Існуть різні методи та системи контролю доступу, але вони мають певні недоліки, такі як:

– Тривалий час навчання. Алгоритму необхідно проаналізувати велику кількість тестових зображень;
– При виявленні, на положення обличчя є обмеження;
– Складна процедура запровадження нових шаблонів у базу даних;
– Низька швидкість спрацьовування;
– Точність роботи методу;
– Висока обчислювальна складність.

Дані недоліки потрібно вирішити, оскільки вони відчутно впливають на швидкодію систем та точність зіставлення відбитків обличчя.

Система, що розробляється повинна володіти максимальною функціональністю і мінімальною вартістю.

Метою даної роботи є покращення швидкодії системи порівняно з аналогами, а також вирішення недоліків, які присутні в аналогічних системах контролю доступу з використанням face-ідентифікації.

Для досягнення даної мети, потрібно виконати наступні завдання:

1. здійснити аналіз можливих варіантів проникнення зловмисника до приміщення;
2. здійснити аналіз захисту приміщення від проникнення зловмисника;
3. визначити недоліки існуючих методів і способів захисту приміщення від несанкціонованого проникнення;

4. розробити метод і спосіб підвищення захисту приміщення від несанкціонованого проникнення;
5. спроектувати систему контролю доступу з використанням face-ідентифікації користувача для підвищення захищеності приміщення.
6. реалізувати систему контролю доступу з використанням face-ідентифікації користувача для підвищення захищеності приміщення;
7. дослідити якість розробленої системи контролю доступом;
8. провести випробування системи контролю доступом.

Об'єкт дослідження – метод підвищення захищеності системи контролю доступом.

Предмет дослідження – розробка модуля підвищення захищеності системи контролю доступом до приміщення з використанням face-ідентифікації користувача.

Теоретичне значення результатів роботи – результати даної роботи можуть бути застосовані для контролю пересування працівників і сторонніх осіб до приміщення і повідомляти власника про сторонніх осіб які намагаються потрапити до приміщення.

Новизна – заключається в можливості роботи системи в оффлайн режимі та віддаленого керування за допомогою web-інтерфейсу, реалізація системи оповіщення та отримання медіаданих з камери за допомогою месенджера Telegram. Також є можливість підключення до «live режиму» камери для моніторингу ситуації з входу до приміщення.

Практична цінність роботи – полягає у тому, що вирішуються та аналізуються завдання контролю доступу з використанням біометричних даних, а саме обличчя. Дана розробка дає можливість здійснювати контроль доступу до приміщення за допомогою face-ідентифікації користувача і за допомогою віддаленого керування доступом. Також система повідомляє власника про сторонніх осіб за допомогою месенджера Telegram.

Однією з практичних функцій на відмінну від вже існуючих аналогів є віддалене керування приладом за допомого месенджера Telegram, що дозволяє

дізнатися про сторонніх осіб заранння. Прилад з даним функціоналом, який включає в себе одразу контроль доступу з використанням біометричних даних, швидке оповіщення про стороніх осіб і можливістю віддаленого керування є дуже практичним і дешевим варіантом для покриття завдань: контролю доступом.

1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ТА ЗАСОБІВ КОНТРОЛЮ ДОСТУПУ

1.1 Аналіз методів контрою доступу

Система контролю та управління доступом, СКУД (англ. Access Control System, ACS) — сукупність програмних та апаратних засобів, які забезпечують захист об'єкта від несанкціонованого проникнення, які також формують реєстрацію входу-виходу людей або транспорту через задані точки, наприклад: двері , ворота, турнікети, шлагбауми та інше[1].

Система контролю та управління доступом відіграє важливу роль у комплексній системі охорони підприємства, забезпечує і збереження майна споруд та працівників[2].

Для організації прохідної на підприємстві необхідно використовувати турнікети наприклад. Даний вид запобіжних пристроїв дозволить якщо не повністю виключити попадання сторонньої особи на об'єкт, то принаймні дуже ускладнить їх подолання[3].

Невід'ємною особливістю сучасних СКУД є можливість здійснення обліку робочого часу працівників. Облік робочого дня - це програмно-апаратний комплекс. Програмне забезпечення дозволяє керувати

відвідуваністю, проводити онлайн-моніторинг та будувати різну звітність про відвідуваність. Для фіксації часу входу та виходу співробітник повинен мати ідентифікатор. Це може бути безконтактна карта, відбиток пальця, венозний малюнок долоні, геометрія обличчя та інше. Тому для вибору системи обліку робочого часу необхідно визначитися за яким ідентифікатором буде фіксуватися факт входу та виходу користувача[4].

Правильно спроектована та установлена СКУД дозволяє знизити витрати на охорону. Крім цього, сучасні системи контролю доступу містять великий потенціал для скорочення постійних витрат за рахунок автоматизації процесів[5].

Наприклад, сучасні системи реєстрації відвідувачів можуть замінити цілий відділ видачі перепусток, а це як мінімум 2 особи. А інтеграція системи обліку робочого часу із системами з розрахунку заробітної плати дозволять скоротити час на її розрахунок[6].

Поєднання з функціями обліку робочого часу системи контролю доступу перетворюється на справжню систему аналітики бізнесу. І облік робочого часу – аж ніяк не єдиний інструмент для цього. Широкі можливості інтеграції з різними системами (документообігу, кадрового обліку, бухгалтерські та CRM) виводять автоматизоване отримання даних та їхню обробку на новий рівень[6].

Зазвичай система контролю та управління доступом складається з цілого ряду компонентів, починаючи з тих, які ідентифікують співробітника, і закінчуючи тими, що приймають рішення про надання доступу[7].

Найбільшу популярність на нашому ринку отримали ідентифікатори з безконтактною передачею даних RFID. А найпоширенішими форм-факторами буде, відповідно, пластикова карта та брелок. Хоча є, звичайно, ще й браслети, а також гігантський асортимент міток різних стандартів[8].

Типи ідентифікаторів:

– безконтактні картки (EM-Marine або Mifare);
– безконтактні брелоки (EM-Marine або Mifare);
– безконтактні мітки (UHF).

Рисунок 1.1 – Зовнішній вигляд RFID міток

В якості ідентифікатора може виступати не тільки карта або брелок, а й відбиток пальця, венозний малюнок (пальця, долоні), геометрія (форма) обличчя, та інше[9].

Зчитувач – невід'ємна складова СКУД. Цей електронний пристрій зчитує інформацію про об'єкт (наприклад, біометричні дані) та передає їх контролеру або терміналу для подальшого рішення про надання доступу. Існують різні моделі зчитувачів, які відрізняються за способом ідентифікації, за стандартами читання даних, що підтримуються, за типом підключення[10].

Типи зчитувачів:

– зчитувачі стандарту EM-Marine;
– зчитувачі стандарту Mifare;
– зчитувачі дальньої дії (UHF);
– зчитувач відбитка пальців;
– зчитувачі венозного малюнка пальця;
– зчитувачі долоні;
– зчитувач з геометрії обличчя.

Рисунок 1.2 – Зовнішній вигляд моделей зчитувачів для СКУД

Варіанти виконання пристроїв залежать від типу ідентифікатора, що застосовується. Так, для зчитування безконтактних карт обладнання виконується найчастіше з ABS пластику, стійкого до ультрафіолету, з електронною платою та антеною всередині корпусу. Обладнання, що монтується зовні приміщення, виконується з матеріалів, здатних витримувати температурні перепади та вплив опадів. Також є зчитувачі далекого радіусу дії, призначені для ідентифікації об'єктів на відстані до 12 метрів. Подібні пристрої зручні у використанні зі шлагбаумами для парковок та автопроїздів тощо[10].

Для біометричних ідентифікаторів (відбиток пальця, венозний малюнок, геометрія обличчя) потрібні біометричні зчитувачі. Біометричні технології ґрунтуються на «зчитуванні» певних тілесних характеристик користувача. Дані перетворюються на цифровий код, який потім надходить у контролер або термінал, і базу даних програмного забезпечення[11].

Варто звернути увагу, що біометричний ідентифікатор, наприклад відбиток пальця, зберігається в базі як унікальний цифровий код, з якого не

можна відновити папілярний малюнок відбитка пальця. Ця технологія дозволить дотримуватися закону про захист персональних даних[12].

Використовуючи біометричні технології в системах контролю та управління доступом, ви отримуєте ряд переваг перед традиційними системами. Наприклад, RFID ключ може бути втрачений співробітником і ключем може скористатися зловмисник, щоб проникнути на об'єкт. Плюсом використання таких систем є швидкість додавання нового співробітника до бази даних та здешевлення адміністрування такої системи, оскільки немає необхідності відновлювати втрачені карти та заводити ключі для нових користувачів[13].

Дані пристрої забезпечують фізичну перешкоду, яка обмежує доступ до об'єкта, що контролюється. Усі пристрої можна розділити на дві групи за принципом дії: електромеханічні та електромагнітні[14].

Електрозамки - мають різний клас захищеності і різний тип установки. Вони можуть використовуватися як усередині приміщення, так і зовні, можуть бути врізними та накладними[15].

1.2 Аналіз методів контролю доступу за допомогою розпізнавання обличчя

Розпізнавання облич (Face Recognition – англ.) – це одні з найбільш перспективних методів біометричної безконтактної ідентифікації людини по обличчю[17].

Перші системи розпізнавання облич були реалізовані як програми, що встановлюються на комп'ютер. У наш час технологія розпізнавання облич найчастіше використовується в системах відеоспостереження, контролю доступу, на різноманітних мобільних та хмарних платформах. Журнал Массачусетського технологічного інституту – MIT Technology Review включив технологію розпізнавання облич до списку 10 проривних технологій 2017 року[16].

Система розпізнавання облич може бути описана як процес зіставлення осіб, що потрапили в об'єктив камери з базою даних раніше збережених та ідентифікованих зображень облич еталонів. По структурній реалізації системи розпізнавання облич можна назвати три поширені схеми[17]:

1. Аналіз відеопотоку на сервері:

Найбільш поширена схема реалізації - IP-камера передає відеопотік на сервер, на сервері спеціалізоване програмне забезпечення виконує аналіз відеопотоку та порівняння отриманих з відеопотоку зображень осіб, з базою осіб еталонів[18].

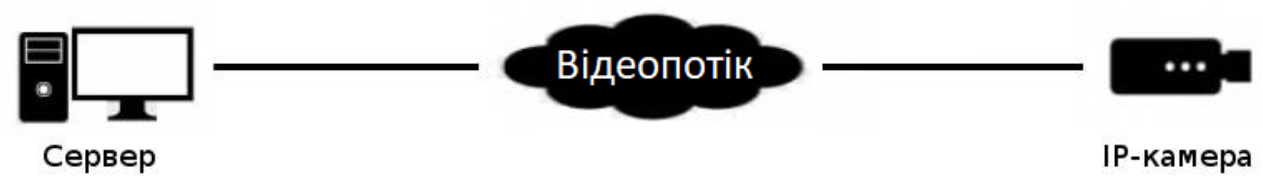

Рисунок 1.3 – Схематичний вигляд аналізу відеопотоку на сервері

Недоліками такої схеми є висока навантаження на мережу, висока вартість сервера, навіть до найпотужнішого сервера можна підключити обмежену кількість IP-камер, тобто чим більше система, тим більше серверів. Перевагою є можливість використовувати вже існуючу систему відеоспостереження[19].

2. Аналіз метаданих на IP-камері:

В даному випадку аналіз зображення буде проводитися на самій камері, а на сервер передаватимуться оброблені метадані[20].

Рисунок 1.4 – Схематичний вигляд аналізу метаданих на IP-камері

Недоліки — потрібні спеціальні камери, вибір яких невеликий, вартість камер вища за звичайні. Також у системах різних виробників буде по-різному вирішуватись питання зберігання та розміру бази даних розпізнаних облич еталонів, а також питань взаємодії програмного забезпечення камери з сервером[21].

Переваги - підключення практично необмеженої кількості камер одного сервера[22].

3. Аналіз метаданих на пристрої контролю доступу:

На відміну від перших двох схем де використовуються IP-камери, в даному випадку камера вбудована в пристрій контролю доступу, який, крім розпізнавання обличчя, що природно відбувається на пристрої, виконує функції управління доступом як правило через турнікет або електрозамок встановлений на двері. База даних осіб стандартів зберігається на пристрої або в хмарі, і зазвичай вже не у вигляді фотозображень[23].

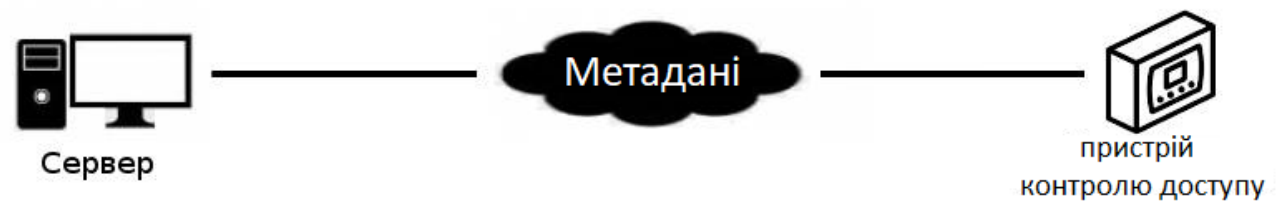

Рисунок 1.5 – Схематичний вигляд аналіз метаданих на пристрої контролю доступу

Недоліки - зазвичай всі такі пристрої випускаються для використання в приміщеннях.

Переваги — низька вартість систем у порівнянні із системами відеоспостереження, які використовуються для розпізнавання осіб[24].

У будь-якому разі успіх реалізації проектів із розпізнавання осіб залежить від трьох важливих факторів[25]:

– алгоритм розпізнавання;

– бази даних розпізнаних осіб (еталонів);

– швидкодія алгоритму.

Як правило система складається з камери відеоспостереження та програмного забезпечення, яке виконує аналіз зображень. Програмне забезпечення для розпізнавання осіб засноване на обробці зображень та обчислення складних математичних алгоритмів, які вимагають більш потужний сервер, ніж зазвичай потрібно для систем відеоспостереження[26].

Існує декілька технологій розпізнавання обличчя, наприклад двовимірна (2D), тривимірна (3D), по текстурі шкіри обличчя, тепловізорному зображенні.

Найпопулярнішою технологією являється 2D технологія[27]. В основі технології 2D розпізнавання облич, лежать плоскі двовимірні зображення. Алгоритми розпізнавання облич використовують: антропометричні параметри особи, графи - моделі облич або еластичні 2D-моделі облич, а також зображення з особами представлені деяким набором фізичних чи математичних ознак[28].

Величезною перевагою 2D розпізнавання облич є наявність готових баз даних облич еталонів, і готової інфраструктури[29].

Недоліками даної технології являються вищі коефіцієнти помилок FAR (False Acceptance Rate – ймовірність того, що помилково ідентифікує незареєстрованого користувача) та FRR (False Reject Rate – ймовірність того, що система не ідентифікує зареєстрованого користувача) порівняно з 3D розпізнаванням облич[30].

3D розпізнавання (Three-dimensional face recognition – англ.) проводиться як правило по реконструйованим тривимірним образам. Технологія 3D розпізнавання облич має більш високі якісні характеристики. Хоча, звісно і вона не є ідеальною[31].

Таблиця 1.1 – Коефіцієнти помилок FAR і FRR в технологіях 2D та 3D розпізнавання обличь, джерело: security.com Software quality 2020

Метод біометричної ідентифікації	Коефіцієнт хибного пропуску, FAR	Коефіцієнт хибного пропуску, FRR
Розпізнавання обличчя 2D	0,1%	2,5%

Розпізнавання обличчя 3D	0,0005%	0,1%

Але дана технологія досить просто підлягає взлому. Навіть Face ID незважаючи на всю крутість, був зламаний в'єтнамською компанією Bkav відразу після надходження у продаж. Маска була створена за допомогою 3D-принтера. Собівартість створення маски лише $150. Створення маски досить складно для звичайної людини, але для професіоналів це досить просто[32].

Також 3D розпізнавання вимагає спеціальних камер для сканування, які в кілька разів дорожчі за звичайні камери, які використовуються в 2D розпізнаванні[32].

1.3 Аналіз існуючих засобів контролю доступу за допомогою розпізнавання обличчя.

Серцем будь-якої системи контролю та управління доступом є її контролер. Ці пристрої вирішують такі завдання:

– управління виконуючими пристроями;

– зберігання бази даних користувачів;

– зберігання журналу подій.

Без перебільшення можна сказати, що контролер - це мозок будь-якої системи доступу[33].

Системи розпізнавання обличчя використовують комп'ютерні алгоритми для обробки та аналізу відмінних деталей обличчя людини. Деталі, такі як відстань між очима, чи форма підборіддя, перетворюються на математичне зображення, потім воно порівнюються з даними в базі даних розпізнавання обличчя. Дані про певне обличчя називають шаблоном обличчя і відрізняються від фотографії, оскільки вони розроблені так, щоб включати лише певні деталі, які можна використовувати для відрізнення обличчя від інших[34].

Розпізнавання обличчя на даний момент вже використовується в багатьох країнах і навіть на спортивних змаганнях у Сполучених Штатах та інших країнах[35].

Рисунок 1.6 – Застосування технології розпізнавання обличчя в банках, джерело: privatbank.ua

Банки вже використовують системи розпізнавання облич під час розгляду кредитних заявок. Банки запобігають шахрайським угодам на круглі суми за допомогою системи розпізнавання облич, а деякі економлять десятки мільйонів за місяць[36].

Банки впроваджують систему ідентифікації по облич у відділеннях банку та запускають можливість переказу грошей з ідентифікацією по фотографії через мобільний додаток[37].

У банківському секторі розпізнавання облич використовується в:

– у системах контролю доступу, для організації фізичного доступу до найбільш охоронюваних приміщень та приміщень у яких зберігаються цінності та кошти;

– для ідентифікації, верифікації, аутентифікації як фізично, так і програмних програм;

– для підтвердження особи пред'явника паспорта та фото на паспорті.

Система розпізнавання облич є ефективним інструментом для запобігання проникненню «небажаних» осіб на територію, що охороняється[38].

До систем розпізнавання облич на таких об'єктах можуть бути підключені як системи контролю доступу, які працюють у режимі верифікації,

так і системи відеоспостереження, які можуть у режимі моніторингу інформувати про потрапляння в об'єктив камери людей їхнього «чорного списку».

Існує кілька важливих показників для оцінки якості програмного забезпечення.

Найбільш важливі з них є FRR і FAR[39].

Як розраховується FRR (False Reject — Петро, не розпізнаний як Петро):

$$\text{FRR} = \frac{FR}{Nt} \times 100\%, \quad (1.1)$$

де Nt — кількість еталонів зображень у базі даних;

FR — кількість помилкових нерозпізнань.

Як розраховується FAR (False Acceptation — Петро розпозначений як Іван):

$$\text{FAR} = \frac{FA}{Nt} \times 100\%, \quad (1.2)$$

де Nt — кількість еталонів зображень у базі даних;

FA — кількість помилкових розпізнань.

Перше і саме важливе, що потрібно знати про ці два показника, це те, що вони не абсолютні, а відносні, тобто вони можуть мінятися в залежності від налаштувань алгоритму розпізнавання облич[40].

Друге це то, що ці показники взаємозв'язані - чим менше FAR тим більше FRR[39].

Орієнтовані значення FRR і FAR для системи розпізнавання облич та їх взаємозв'язок представлені в таблиці[41]:

Таблиця 1.2 – Взаємозв’язок коефіцієнтів помилок FAR і FRR, джерело: security.com Software quality 2020

FAR	FRR
0,1%	2,5%
0,01%	7%
0,001%	10%

Оскільки виходить такий цікавий взяємозв’язок коефіцієнтів з таблиці 1.2, можна порівняти FAR і FRR різних методів біометричної ідентифікації та порівняти їх порівняно один від одного:

Таблиця 1.3 – Коефіцієнти помилок FAR і FRR в різних методах розпізнавання обличь, джерело: security.com Software quality 2020

Метод біометричної ідентифікації	Коефіцієнт хибного пропуску, FAR	Коефіцієнт хибного пропуску, FRR
Відбиток пальця	0,001%	0,6%
Розпізнавання обличчя 2D	0,1%	2,5%
Розпізнавання обличчя 3D	0,0005%	0,1%
Райдужна оболонка ока	0,00001%	0,016%
Сітківка ока	0,0001%	0,4%
Малюнок вен	0,0008%	0,01%

Системи розпізнавання осіб можуть використовуватись у системах контролю доступу у двох режимах[43]:

– режим ідентифікації – рішення про допуск приймається на основі лише даних від системи розпізнавання осіб. Наприклад, база даних з співробітників 100 осіб, і завдання системи розпізнавання порівняти обличчя поточної людини з базою даних у 100 осіб. Тобто, порівняння відбувається 100:1. Якщо людина буде ідентифікована як співробітник, то їй буде надано доступ[44].

Цей режим найефективніше використовувати в завданнях виявлення сторонніх на контрольованій території. Як правило є сенс використовувати в зонах підприємства, що особливо охороняються, куди доступ дозволено обмеженому колу осіб. До системи розпізнавання підключаються всі камери встановлені на даній території у разі виявлення будь-якої особи, яка не міститься у базі даних, відбувається інформування служби безпеки[45].

– режим верифікації - ідентифікація в даному випадку проводиться за допомогою іншої технології, наприклад RFID, або може використовуватися мобільні ідентифікатори, або відбитки пальця або венозний малюнок руки або пальця[46].

Людина підносить карту до зчитувача система його ідентифікує, тобто встановлює що це Тарас, і Тарасу дозволено доступ зараз. Система розпізнавання облич у даний момент вже знає, що це Тарас, і використовуючи

лише фото Тараса з бази даних порівнює, пред'явника RFID карти з фотографією Тараса у базі даних. Тобто, порівняння відбувається 1:1[47].

У режимі верифікації працює взагалі ідеально, оскільки завдання верифікації дуже просте навіть для середніх за якістю систем розпізнавання облич. Даний режим доцільно використовувати на будь-яких прохідних – бізнес-центрах, виробничих підприємствах, інститутах, школа[48].

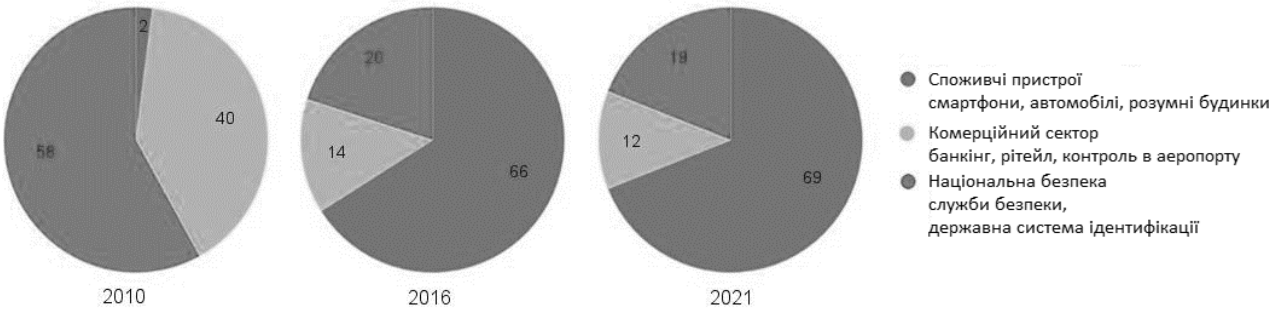

Рисунок 1.7 – Зміна ринку біометричного обладнання (частка ринку, %), джерело: security.com Software quality 2020

Галузі, в яких сконцентровані найбільші можливості для впровадження систем розпізнавання облич з кожним роком збільшуються в мірі використання. Що призводить до збільшення попиту на біометричне обладнання, як результатом цього є виробництво та вдосконалення даного обладнання та алгоритмів для обробки біометричних даних[49].

Не мала кількість негативу йде з боку критиків на рахунок технології розпізнавання облич. Наприклад, американський союз захисту громадянських свобод (ACLU) вирішив «знищити» Amazon, продемонструвавши «жахливу» неефективність їхньої системи розпізнавання осіб. І видавши досить помітну статтю, яку розповсюдили ЗМІ без будь-якого осмисленням викладених фактів.

Суть притензії така — система розпізнавання облич Amazon визначила 28 конгресменів США як злочинців[50]. Якщо дізнатися подробиці даного експеременту, то фотографії 535 американських конгресменів порівнювалися з базою з 25 000 фотографій раніше затриманих осіб. Незважаючи на заздалегідь відому відсутність парламентаріїв на 25 000 фотографій, технологія ідентифікувала 28 конгресменів як злочинців[50]. Більшість ЗМІ приводять цю

новину як доказ низької ефективності, а найвідчайдушніші взагалі як доказ повної не працездатності.

А тепер якщо порахувати кількість порівнянь, які мала зробити система, то для цього ми 535 фотографій конгресменів помножимо на 25000 фотографій які містилися в базі даних, вийде 1337500. З 1337500 порівнянь 28 виявилися помилковими. Тобто коректними виявилися 1337472. Тепер переведемо у відсотки 1337472 розділимо на 1337500, і отримаємо ефективність системи 99,997%[49].

Випливає, що система працює чудово. ACLU помилялися в тому, що розпізнавання обличь є неефективним і непрацездатним. Оскільки дослідження і розрахунки похибок говорять про протилежне. Але все одно багато хто виявляє побоювання щодо використання біометричних даних державою так і ще більші побоювання з приводу використання цих даних комерційними структурами[44].

1.4 Аналіз ефективності 2D технології розпізнавання облич

Технологічно системи іноді можуть сильно відрізнятися щодо розпізнавання осіб, але вони мають приблизно загальні принципи роботи.

Розпізнавання обличчя можна озділити на декілька короків:

Крок 1: Виявлення особи

Для початку камера виявить обличчя людини, чи вона одна або перебуваючи в натовпі. Особа найкраще виявляється в той момент, коли людина дивиться прямо в камеру, проте сучасні технологічні досягнення дозволяють також виявляти обличчя і в тих ситуаціях, коли людина не дивиться прямо в камеру (звичайно, у певних межах).

Крок 2: Аналіз особи

Потім знімається фотографія обличчя та починається його аналіз. Більшість рішень для розпізнавання облич використовує 2D-зображення

замість об'ємних 3D-зображень, оскільки вони можуть більш просто зіставляти 2D-фото із загальнодоступними фотографіями або фотографіями, що є в базі даних. Кожна особа складається з помітних орієнтирів або вузлових точок. Кожна людина має 80 вузлових точок. Програми для розпізнавання осіб аналізують вузлові точки, такі як відстань між вашими очима або форма ваших вилиць.

Крок 3: Конвертація зображення у дані

Після цього аналіз особи перетворюється на математичну формулу. Ваші риси особи стають числовим кодом. Такий числовий код називається відбитком особи (faceprint). Подібно до унікальної структури відбитка великого пальця, кожна людина має свій власний відбиток обличчя.

Крок 4: Пошук збігів

Далі код порівнюється з базою даних відбитків облич. Потім технологія визначає відповідність ваших точних даних тому, що представлено в базі даних. У цій базі даних є фотографії з ідентифікаторами, які можна порівняти.

Системи розпізнавання осіб які завжди здатні точно зіставити відбитки облич із базою даних. Але як правило, помилки виникають внаслідок поганої якості зображень або нестачі інформації в базі даних. Погане освітлення або низька якість зображення можуть ускладнити точного аналізу вузлових точок людини. Наприклад, на дані може вплинути затемнення ряду рис особи. Це створює помилку у відбитку особи, внаслідок чого неможливо буде порівняти його з правильними даними в базі даних.

Крім основних етапів виявлення та розпізнавання існують проміжні етапи обробки найденних облич, які можуть вирішити дані недоліки: фільтри які після виявлення облич допомагають знизити вплив шумів, а також маска значущих областей, що дозволяє прибрати вплив кутових областей зображення, що містять задній план.

З метою усунення шумів на зображеннях обличчя можна застосувати фільтр Гауса. Фільтр Гауса – це фільтр розмиття зображення, який

використовує нормальний розподіл (також зване Гаусовим розподілом) для обчислення перетворення, що застосовується до кожного пікселя зображення.

Розмиття по Гаусу дозволяє позбутися небажаних шумів на зображеннях і зводить до мінімуму їх вплив при класифікації облич.

1.5 Висновки та постановка задачі

В даному розділі було проведено аналіз існуючих СКУД. Проведено аналіз різних видів обробки біометричних даних, проведений аналіз існуючих засобів обробки біометричних даних. А також теоретично розглянуто місця і сфери застосування приладів контролю і управління доступом за допомогою біометричних даних.

Метою даної роботи є покращення швидкодії системи порівняно з аналогами, а також вирішення недоліків, які присутні в аналогічних системах контролю доступу з використанням face-ідентифікації.

Для досягнення даної мети, потрібно виконати наступні завдання:

1. здійснити аналіз можливих варіантів проникнення зловмисника до приміщення;
2. здійснити аналіз захисту приміщення від проникнення зловмисника;
3. визначити недоліки існуючих методів і способів захисту приміщення від несанкціонованого проникнення;
4. розробити метод і спосіб підвищення захисту приміщення від несанкціонованого проникнення;
5. спроектувати систему контролю доступу з використанням face-ідентифікації користувача для підвищення захищеності приміщення.
6. реалізувати систему контролю доступу з використанням face-ідентифікації користувача для підвищення захищеності приміщення;
7. дослідити якість розробленої системи контролю доступом;
8. провести випробування системи контролю доступом.

2. РОЗРОБКА ТА ПРОЕКТУВАННЯ ПРИСТРОЮ КОНТРОЛЮ ДОСТУПУ

Система контролю і управління доступом – це комплекс технічних та програмних засобів безпеки, що здійснює регулювання входу/виходу та переміщень людей чи транспортних об’єктів на територіях, які знаходяться під охороною, для адміністративного моніторингу та попереджень несанкціонованого проникнення [26].

2.1 Розробка структурної схеми

Першим кроком потрібно визначити функціонал приладу, який буде реалізуватися. Основною функцією даного приладу є контроль доступу до певної території з обмеженим доступом. Оскільки територія є обмеженою, стороні особи не повинні потрапити туди ніяким чином без дозволу. Оскільки дана особа може бути зловмисником і може заволодіти деякою інформацію. Захист такої території відбувається різними способами і засобами. Одним з таких способів є контроль доступу[27].

Контроль доступу реалізується різними способами. Наприклад прохідна на підприємстві, в якості контролю виступає охоронець, який контролює потік людей, які проходять через дану прохідну. Але в даному випадку є багато різних факторів, які можуть посприяти проникненню сторонніх осіб[25].

Для покращення такої системи встановлюються турнікети з RFID зчитувачами. Які зменшують шанс потрапити на територію без відома охоронця. Також дана система дає можливість контролювати працівників даного підприємства (коли приходять або йдуть з роботи, скільки часу витрачають на перерви і тому подібне)[28].

Але є і мінуси даної системи, а саме[30]:

– працездатність мітки втрачається при частковому механічному пошкодженні;

– вартість системи перевищує номінальну вартість системи обліку, заснованої на штрих-кодах;

– складність самостійного виготовлення. Штрих-код можна надрукувати на будь-якому принтері.

– схильність перешкод у вигляді електромагнітних полів.

– недовіра користувачів, можливості використання її для збору інформації про людей.

– встановлена \u200b\u200bтехнічна база для зчитування штрих-кодів істотно перевершує за обсягом рішення на основі RFID.

– недостатня відкритість вироблених стандартів.

Якщо зловмисник заволодіє міткою, система не може розпізнати невідому особу, як зловмисника. Оскільки він використовує мітку працівника підприємства і система розпізнає зловмисника, як співробітника. Таку ситуацію може покрити система відеонагляду. І охорона може затримати зловмисника, який проник на територію[32].

Гарним рішенням цих проблем є система контролю доступу з використанням розпізнавання обличчя. В даному випадку унеможливлюється втрата картки для пропуску, оскільки її немає взагалі. Розпізнавання особи відбувається по рисам обличчя. Система сама розпізнає особу і вирішує, чи пропускати дану людину на територію[32].

Даний прилад має багато переваг над простими системами контролю доступу. Як згадувалося раніше, тут відсутні картки для пропуску, оскільки вони не потрібні, система поєднує в собі контроль доступу і відеонагляд. Оскільки можна підключитися до системи і переглядати відеопотік з камери[34].

Керування даною системи можна виконувати віддалено, достатньо підключення до мережі WiFi. Якщо якимось доступу до інтернету немає, то можна підключитися до системи локально, використовуючи офлайн веб інтерфейс в якому доступні всі основні функції системи[18].

Також дана система відправлятиме повідомлення про невідомих, за допомогою месенджера Telegram та бота. Керування системою також здійснюється за допомогою бота[41].

Досить вагомою перевагою є ціна приладу і простота використання. При низьких затратах для створення даної системи, вона покриває велику кількість задач поставлених на неї, з яким система контролю доступу справляється чудово. Дану систему з легкістю можна інтегрувати до вже існуючих систем контроля доступу для їх модернізації. Оскільки в даній системі використовується контролер, який гнучкий для додаткового функціоналу і подальшій модернізації в цілому[46].

Першим кроком до створення приладу є створення структурної моделі. Щоб знати з чого буде складатися прилад і які зв'язки будуть між ними. Наступним кроком є вибір елементної бази, тобто сам контролер, камера, другорядні радіодеталі і т.д. З основних кроків перед написанням логіки роботи приладу залишається створення принципової схеми приладу. Схема буде показувати зв'язки між радіоелементами і яким чином модулі будуть пов'язані між собою[48].

Дана схема дає уявлення про прилад в цілому. Який функціонал зможе покриватися для контролю доступу, його основнф блоки, вузли, частин та головних зв'язків між ними. [45].

Після даного етапу виконання даного етапу стає зрозуміло, навіщо потрібний даний пристрій і як він працює в основних режимах роботи, як взаємодіють його частини [32].

Модулем для оповіщення про невідому особу, було вибрано месенджер Telegram. Оскільки в даному месенджері можна легко створити бота для оповіщення і керування вистемою віддалено. За допомогою бота можна буде відправляти певні команди модулю, які будуть після виконання відправляти відповідь назад боту. Наприклад включити освітлення, щоб зробити фото з входу до території на даний момент. Або наприклад відкрити ворота, чи двері довіреній особі і т.д.[49].

Другим кроком потрібно зробити структурну схему приладу контроля доступу (рис. 2.1). На схемі показано основні блоки приладу, та зв'язки між ними, а саме між контролером, камерою, реле, електромагнітом, блоком живленя та кнопкою для виходу, яка встановлюєсять з внутрішньої сторони приміщення або території. Як видно з структурної схеми осовним модулем є мікроконтролер ESP32, до якого підкючаються всі інші модулі та блок живлення. Керування віддалено відбувається за допомогою модуля WiFi, який вмонтований в модуль разом з ESP32.

Обрана структурна схема має ряд переваг, такі як: простота та використання мінімальної кількість додаткових модулів для реалізації контроля доступу. На рисунку 2.1 відображено, що модулі підключені максимально логічно, оскільки ESP32 отримує дані від камери та зчитує сигнали від кнопки, за допомогою отриманих даних відбувається керуавання модулем реле, яке в свою чергу керує електромагнітом. Всю систему живить блок живлення, а саме ESP32 і електромагніт, так як реле і камера отримує живлення від самого ESP32.

Керуючись наведеними аргументами вище, було обрано для реалізації саме таку структуру схему.

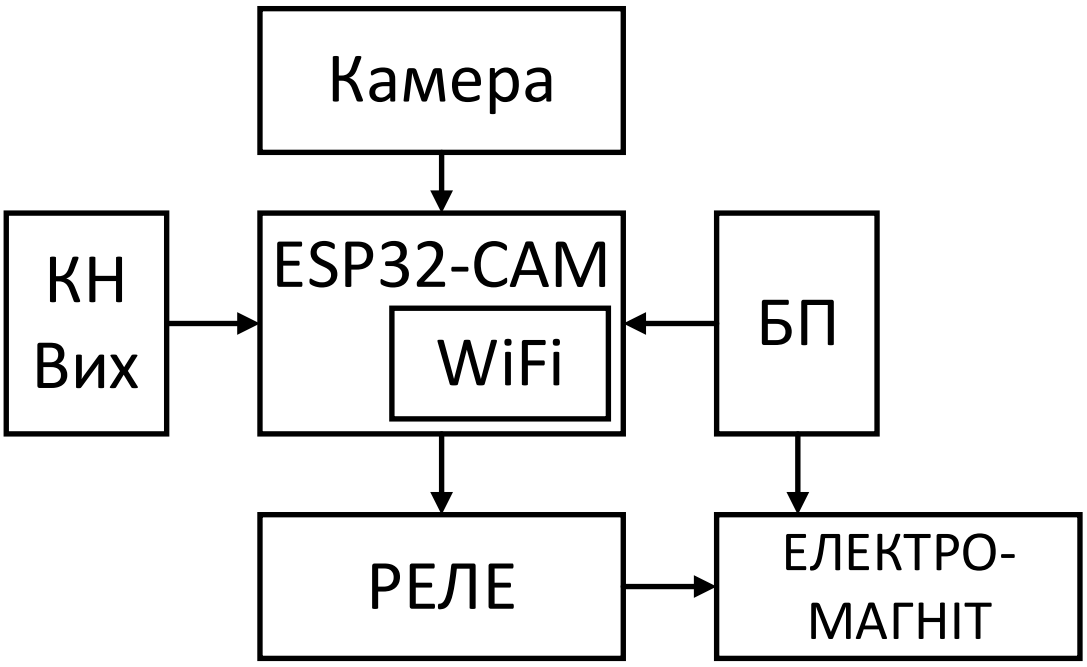

Рисунок 2.1 – Структурна схема приладу

Слідуючим кроком є розроблення блок схеми роботи системи (рис.2.2). З блок схеми можна зробити висновок, що система постійно знаходиться в режимі нагляду. Якщо розглянути більш детально логіку роботи системи, то вона заключається в слудуючому: Після старту приладу одазу йде підключення до мережі WiFi, якщо з якихо причин дана мережа відсутня або доступ до інтернету відсутній, то додатково є офлайн web-інтерфейс, з якого доступні майже всі функції (окрім тих, що потребують підключення до мережі інтернет); Після підключення до WiFi прилад розпочинає моніторити територію за допомогою камери, якщо певна особа наближається до входу в приміщення або територію, камера починає передавати дані до контролера; в свою чергу контролер за допомогою алгоритму розпізнає обличчя особи в цифровому вигляді набору певних значень. Контролер опрацьовує ці дані і звіряє їх з еталоним фото. Еталоне фото також проходить дану обробку за допомогою алгоритму; порівняння поточного обличчя і еталоного відбувається також за допомогою певних алгоритмів для розпізнавання;

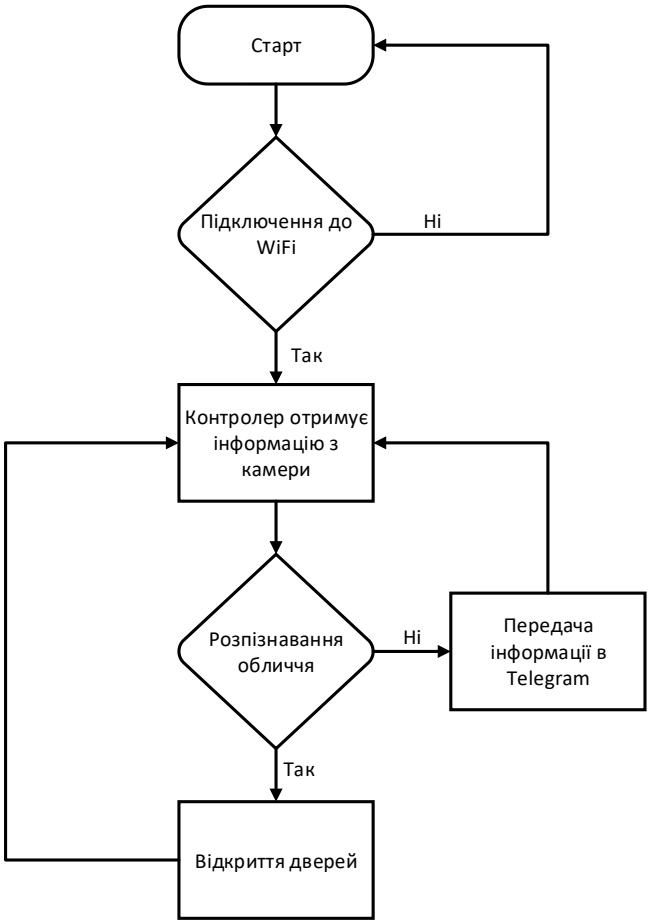

Рисунок 2.2 – Блок схема роботи приладу

Якщо обличчя було не розпізнано, то контролер відправляє дане фото особи в чат Telegram за допомогою бота і повідомляє, що замічено невідому особу. Якщо обличчя було розпізнано, то двері відкриються і доступ буде дозволений.

Піля чого все відбувається по тому самому принципу дій. Система процює достатньо просто, оскільки в даному варіанті висвітлено основну функцію роботи даного приладу.

Блок-схеми даного типу показують рух інформаційного потоку по конкретній системі, а також інші доступні варіанти в залежності від напряму потоку. Щоб схематично представити переміщення користувача на своїй системі та спланувати максимально зручний інтерфейс досить просто.

Отже вибір даної блок схеми можна обгрунтувати тим, що простота роботи бок схеми забезпечує стабільну та швидку роботу приладу.

2.2 Вибір та обгрунтування елементної бази

В якості мікроконтролера було обрано ESP32-CAM - модуль Wi-Fi (рис.2.3), на базі популярного чіпу ESP32, з камерою OV2640 640*480. Модуль також має слот для карт microSD. З модулем працюють приклади з бібліотеки, що встановлюються з espressif SDK. Модуль є досить популярним рішенням багатоьх проектів і різного роду функціоналу. Основною функцією даного модуля є саме отримання відео-потоку з камери через Wi-Fi підключення.

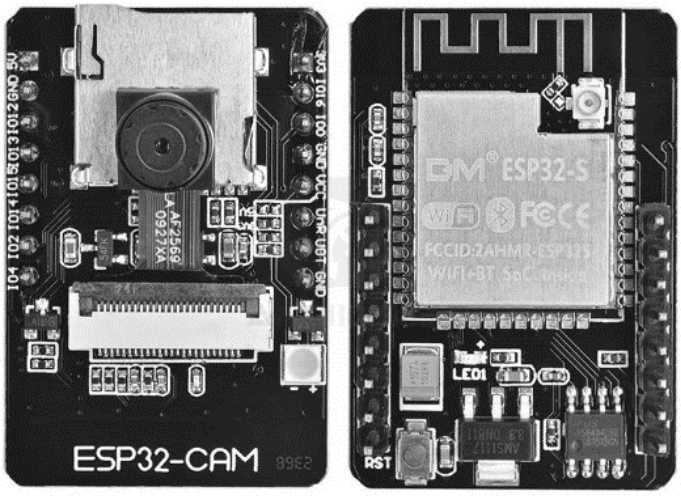

Рисунок 2.3 – Зовнійній вигляд ESP32-CAM

На платі присутні:

– бездротовий модуль ESP32 з інтегрованим Wi-Fi та Bluetooth контролерами;

– камера OV2640 йде окремо, але на платі передбачений роз'єм для її підключення;

– гніздо для карт пам'яті micro-SD;

– світлодіод, який, якщо дотримуватися логічних міркувань, мабуть призначений відігравати роль спалаху.

І оскільки ESP32 не просто реалізує зв'язок по бездротових інтерфейсах, але і є повноцінним контролером сам по собі, то на платі його сигнали виведені на штирьові роз'єми для подальшого використання.

Висока продуктивність дозволяє використовувати ESP32 у системах обробки зображень та мовлення у реальному часі. Домашня автоматика, розумний будинок, контроль здоров'я, сільське господарство, промисловість, робототехніка, іграшки… Усього не перерахувати. ESP32 має:

– двоядерний (або одноядерний) 32-розрядний процесор з тактовою частотою 160 або 240 МГц;

– Wi-Fi: 802.11 b/g/N;

– Bluetooth: v4.2 BR/EDR and BLE;

– велика кількість периферійних модулів, у тому числі:

- SPI x 4
- ADC
- DAC x 2
- UART x 3
- CAN
- I2C x 2

– 520 КБ SRAM.

Мікроконтролери сімейства AVR, а потім платформа Arduino з'явилися задовго до ESP32. Як і у будь-якої плати, заснованої на ESP32, MH-ET LIVE ESP32 DevKit має досить великий набір мов програмування. По-перше, це Arduino C, по-друге, Lua, а по-третє і по-четверте - MicroPython і Espruino.

Однією з ключових особливостей Arduino є відносно низький поріг входження, що дозволяє практично будь-якій людині створити рішення швидко та легко. Платформа зробила важливий внесок у open source hardware співтовариство і дозволила долучитися до нього величезній кількості радіоаматорів. Середу розробки Arduino IDE можна вільно завантажити з офіційного сайту. Незважаючи на очевидні обмеження в порівнянні з професійним середовищем розробки, Arduino IDE покриває 90% того, що потрібно для проектів.

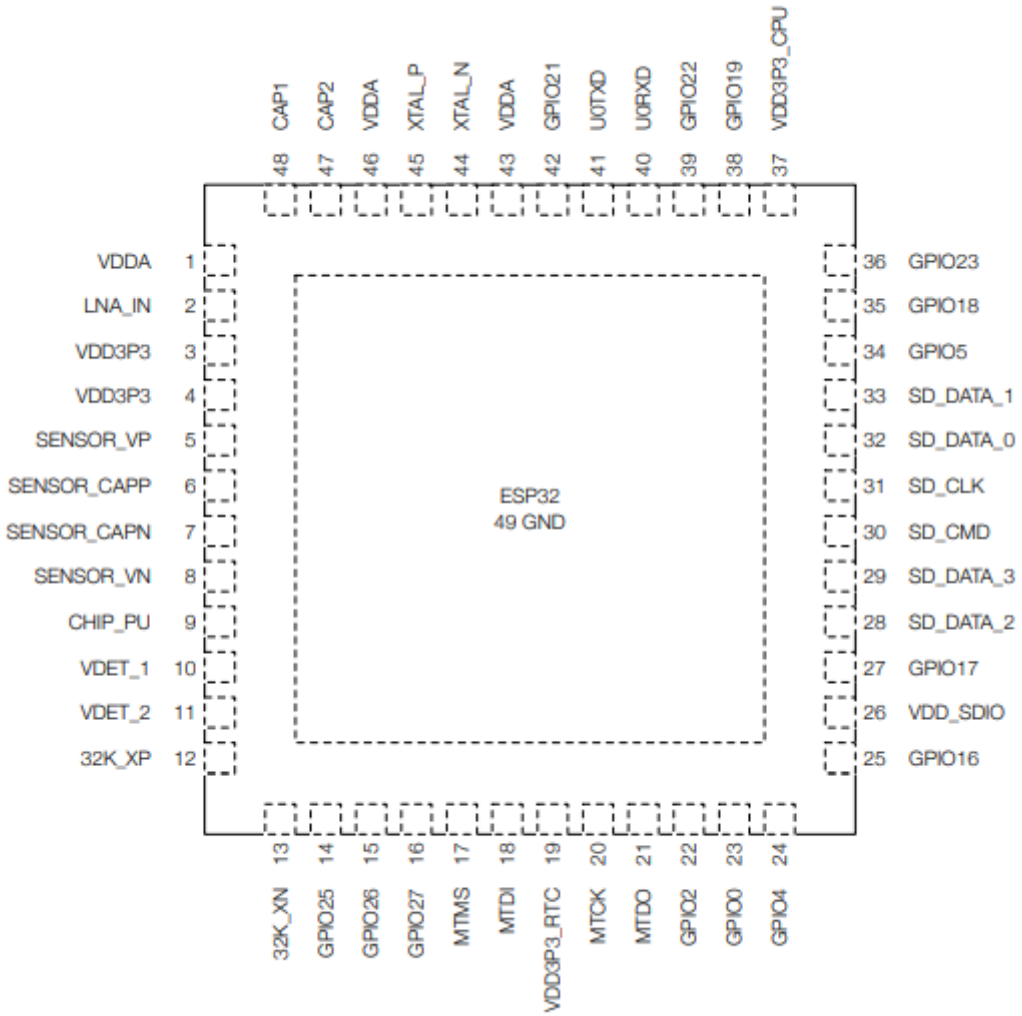

Рисунок 2.4 – Зображення виходів мікроконтролера ESP32

В порівнянні з іншимми контролерами, даний модуль включає в себе всі потрібні функції, також його характеристики задовольняють вимоги приладу в порівняні з його аналогами. Велика кількість різних інтерфейсів дає можливість для підключення різного роду модулів до контролера. Саме тому ESP32-CAM був обраний для реалізації приладу, оскільки він справляється з багатозадачність використовують свої два ядра і тому робота приладу залишатиметься на високоиу рівні.

Прошивається ESP32 по UART, тому для цих цілей USB-UART перехідник на базі CH340G або аналогічні йому. Для підключення використовуються виходи:

– GPIO1 - U0TXD - підключається до Rx UART.

– GPIO3 - U0RXD - підключається до Tx UART.

Крім того, для програмування модуля необхідно підтягнути виведення GPIO0 до землі. Відповідно, після закінчення прошивки підтяжку треба прибрати. І ось так у результаті виглядає повне підключення зображене на рисунку 2.4.

Причому при підключенні живлення через вхід 3.3В спостерігається багато різноманітних проблем — прошивається через раз, працювати модуль відмовлявся в принципі і т. д. І ці проблеми разом усунулися при подачі 5В живлення через відповідний вхід (рис.2.4).

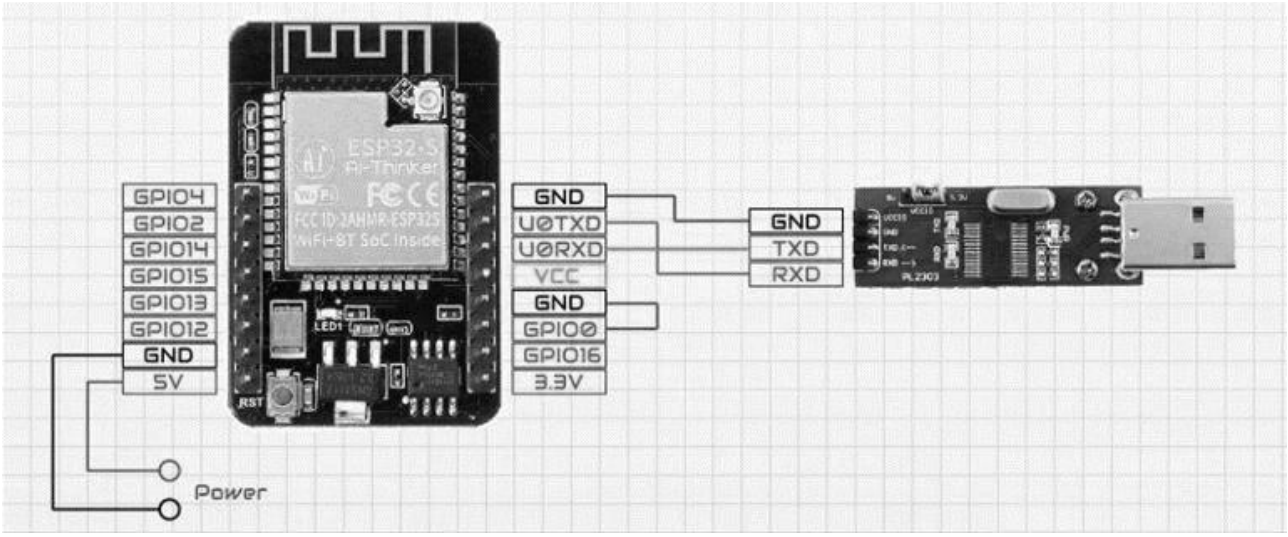

Рисунок 2.4 – Схематичне зображення підключення ESP32-CAM до USB-UART перехідника

В якості USB-UART перехідника було обрано конвертор на базі чіпа CH341A (рис.2.5) - мультифункціональний конвертер з USB 2.0 в UART, EPP, I2C та SPI.

У режимі UART підтримуються як RX/TX, а й інші сигнали управління, тому на чіпі можна створити USB-модем. У режимі паралельного порту реалізовано інтерфейс EPP та емуляцію принтера (що дозволяє підключати принтери з інтерфейсом LPT безпосередньо до USB без написання додаткових драйверів). У послідовному режимі підтримуються інтерфейси I2C та SPI.

Чіп вимагає мінімальної додаткової обв'язки (зовнішній кварц та кілька конденсаторів). Можливе використання зовнішньої EEPROM для зберігання користувацьких Vendor ID, Device ID та деяких налаштувань чіпа. Живлення від 5 В (через вбудований LDO-регулятор), або від 3.3 безпосередньо.

Схема програматора відрізняється від референсної з даташиту лише використанням зовнішнього LDO-регулятора замість вбудованого, мабуть, вбудований виявився не дуже надійним. Крім CH341A на платі присутні 2 світлодіоди (Power і Run), 2 резистори, 5 конденсаторів (2 для кварцу та 3 для LDO-регулятора), кварц на 12 МГц, LDO-регулятор AMS1117 на 1 та ZIF-сокет на два чіпи в корпусі DIP-8-300. З налаштувань є єдиний джампер, який перемикає програматор між I2C/SPI та USB-UART, при цьому у нього змінюється Device ID.

Рисунок 2.5 – Зовнішній вигляд USB-UART перехідника на базі чіпу CH341A

В порівняні з іншими програматорами, даний програматор може використовувати в якості джерела живлення для налагодження приладу. Оскільки на борту присутніє розділення живлення на 5В і 3.3В з використанням стабілізатора на 3.3В, а також необхідна обв’язка радіоелементів для програмування, як реалізована не на всіх аналогах програматорів.

Було обрано даний програматор, тому що він покриває всі потреби для налагодження приладу і цілому.

Для можливості керування електромагнітом, або електрозамком було обрано двоканальний модуль реле (рис.2.6), тому що для них потрібно лише живлення на 12В в основному, а з такими задачами чудово спрвляється реле.

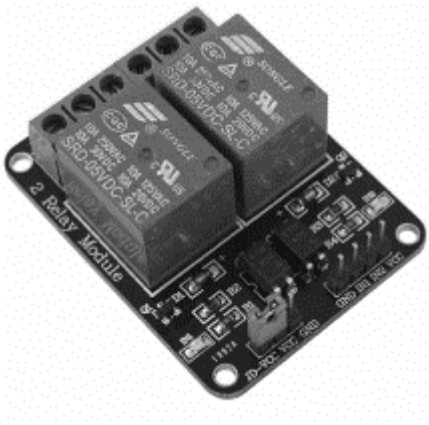

Рисунок 2.6 – Зовнішній вигляд двоканального модуля реле

Керування відбувається за допогою подачі високого або низького сигналу на сигнальнй вхід модуля, після чого перемикання реле відбувається за допомогою об’язки з використаням опторозв’язки. Вона потрібна для того, якщо контролер використовує 3.3В. Живлення даного модуля реле 5В.

Було обрано даний модуль, тому що контролер ESP32 використовує сигнал 3.3В, тому рішення з опторозв’язкою вирішує дане питання. Також порівняно з аналогами модулів реле, даний модуль виділяється наявністю оптороз’язки. Додатковою опцією в модулі присутній джемпер для розілення живлення логіки модуля, та керуванням реле, при необхідності їх можна розділи. Дана функція також відсутня у його аналогів.

2.3 Розробка принципової схеми

Так як було обрано готовий модуль для керування приладом на базі ESP32, а саме Espressif ESP32-CAM, інші модулі також було обрано вже готові (рис.2.7), оскільки готові модулі зі сторони бюджету вигідніше купувати, ніж їх збирати самостійно. Оскільки це не лише економія коштів, але й часу затраченого на їх виготовлення і тестування і доопраювання. Так як всі

модулі системи готові, можна використати з’єднувальні провода, швидкої і комфортної збірки прототипу приладу.

Якщо використовувати додаткові модулі і радіодеталі в системі, наприклад світлодіоди, герконові датчики, датчики руху і т.д. для візуалізації роботи приладу, та розширенню функціоналу. Потрібно буде додаткова обв’язка з пасивних компонентів для деяких з них.

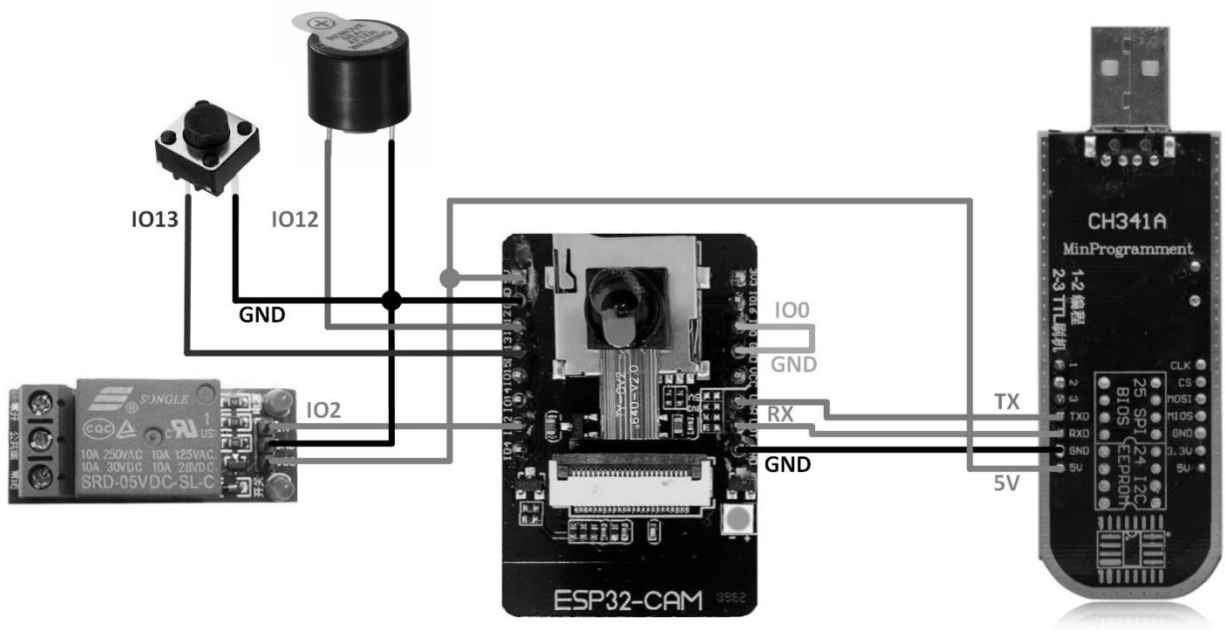

Рисунок 2.7 – Схематичне зображення підключення модулів до контролера

Основою частиною і мозком приладу є ESP32, як згадувалося вище. Він виконує майже всі сункції:

– отримання інформації від камери;

– обробка інформації;

– відправка і отримання GET/POST запитів;

– обробка другорядних задач (зчитування кнопок, датчиків, керування світлодіодами, бузерами і т.д.);

– підтримка локального web-інтерфейсу.

І багато інших функцій може виконувати контролер паралельно. Тому і було обрано даний модуль з поміж інших варіантів. Так як він включає в себе головну перевагу – вмонтований WiFi.

Якщо розглянути схему (рис.2.7), можна побачити мінімальний набір деталей для реалізації даного приладу. А саме:

– ESP32-CAM;
– USB-TLL програматор;
– модуль реле;
– тактова кнопка;
– зумер.

USB-TLL потрібний, для декількох невідємних речей в процесі розробки приладу, а конкретніше написанні коду програми для нього. Він знадобиться того, щоб запрграмувати контролер, також дає змогу дебажити код на протязі написання програми, та тестування приладу.

Це потрібно для того, що перевіряти написаний код на роботоздатність і правильне виконання заданих завдань. Також він в подальшому потрібний для того, щоб отримуватися IP-адрусу ESP32, яку їй в свою чергу видає роутер до якого контролер підключиться. Заданим IP-адресом можна буде перейти на web-інтерфейс приладу і виконати певні налаштування для його правильної роботи в подальшому.

Модуль реле потрібний для того, щоб відкривати двері, якщо використовуються електромагніти, електрозасови або електрозамки. Також можуть використовуватися і турнікети різних типів. В даному випадку розглядаєтсья ситуація з електромагнітом. Принцип реле дуже простий, в потрібний момент (коли йому дасть сигнал контролер), він повинен відкрити двері, а саме перестати подавати струм до електромагніту на визначений період часу. Він керується за допомогою низького та високо рівнів сигналу. При високому рівні – він вмикаєтсья, а при низькому – вимикається. Існують також два типи реле:

– реле високого сигналу;
– реле низького сигналу.

На даний момент потрібно сзертати увагу при написанні логіки роботи приладу, оскільки можна побудувати таку логіку, що система при успішній перевірці не буде пускати в средину ні стороніх, ні співробітників.

На схемі також присутні кнопка та зумер. Тактова кнопка слугує тут, як кнопка для виходу з території. Зумер виступає в роді звукової візуалізації. Якщо система пропустить, то він видасть короткий звуковий сигнал, що дасть працівнику знати, що доступ йому дозволений. Якщо сигнал довгий – це означає, що щось пішло не так і система не розпізнала працівника, або це просто стороня особа, яка хоче потрапити в середину.

Дану схему можна ще розбавити іншими візуалізаційними деталями, а також розшити її можливості, які будуть покривати більше потреб підприємства. На даному етапі система покриває лише контроль доступу і модуль оповіщення охорони або власника про невідомих осіб, які намагаються потрапити на територію або приміщення за до помогою месенджра Telegram.

2.4 Висновки

В даному розділі було описано, та розроблено структурну схему приладу контроля доступу до приміщення з використанням розпізнавання обличчя. Також створено блок-схему роботи приладу, та описано принцип роботи і спрацювання системи в різних ситуаціях. Описано та аргументовано перелік елементів та модулів для використання при розробленні прототипу приладу.

Створено схематичне зображення підключення основних модулів та пасивних елементів до мікроконтролера.

Спираючись на вибрані елементи та модулі було розроблено схему електричну принципову приладу контролю доступу з використанням розпізнавання бличчя.

3. РЕАЛІЗАЦІЯ ОХОРОННОЇ СИСИТЕМИ З ЕЛЕМЕНТОМ КОНТРОЛЮ ДОСТУПУ

У даному розділі буде земульовано та реалізовано прилад контролю доступу до приміщення з використанням розпізнаваня облччя.

3.1 Реалізація та емуляція схеми приладу

Так як всі основні деталі – є готовими модулями, в такому випадку прилад можна зібрати в онлайн конструкторі, та земулювати принцип його роботи, щоб переконатися в в його працездатності.

На даний момент існує багато різних онлайн конструкторів для таких задач, одним з найкращих є Autodesk Circuits. Даних конструктор дозволяє зібрати будь-яку схему на базі Arduino та ESP, а також її можна там і запрограмуватися. Таким чином можна вдостовіритися в правильній роботі приладу перед його реалізацією.

На рисунку 3.1 зображено земульовану схему приладу для контролю доступу до приміщення. В схемі можна побачити: ESP32-CAM, реле, USB-TLL програматор, батарейний відсік, та понижаючий стбілізатор.

За допомогою онлайн конструктора з'єднуємо модулі з контролером, після

чого безпосередньо приступаємо до програмування зібраної схеми.

Для початку підключимо потрібні бібліотеки для ESP32, однією з них буде станартна бібліотека WiFi.h, яка вбудована в Arduino IDE з коробки.

Створимо змінні логін і пароль точки доступу, коди модуль буде підключатися:

```
const char* ssid = "login";
const char* password = "password";
```

Пілся чого кажем модулю в якому режмі він повинен працювати, нам потрібно, щоб модуль працював в режимі чи клієнта, тому задаємо режим STA:

```
WiFi.mode(WIFI_STA);
```

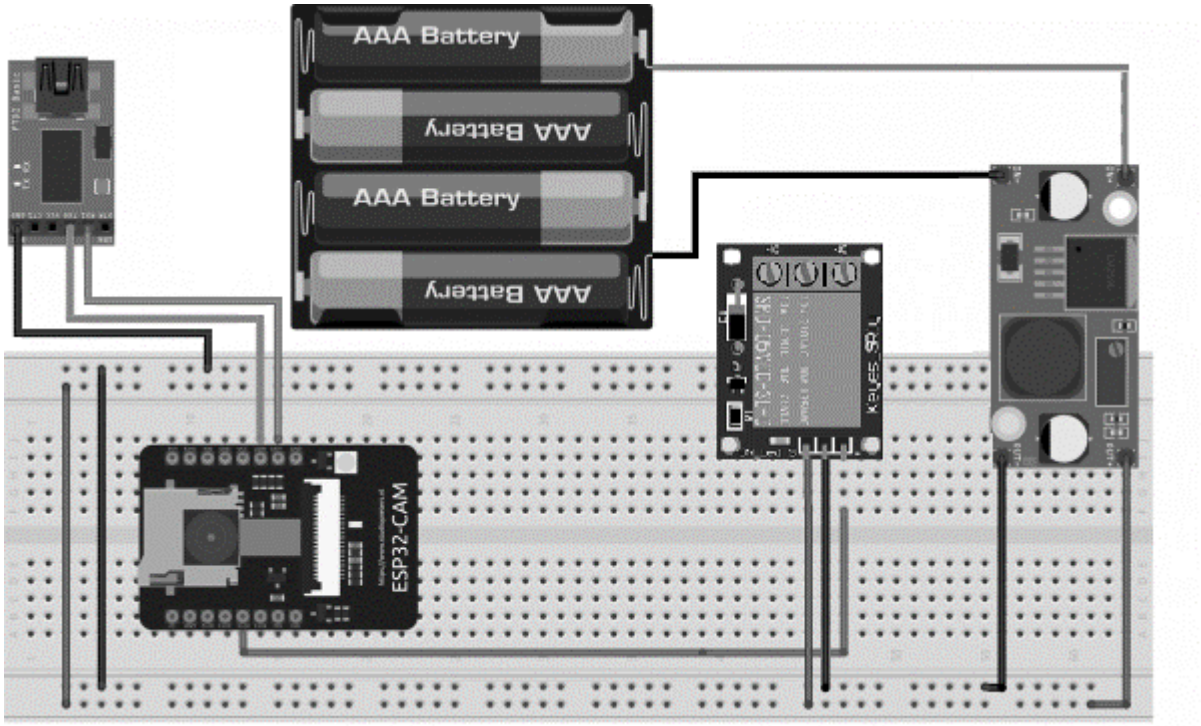

Рисунок 3.1 – Схема приладу для контроля доступу до приміщення з використанням розпізнавання обличчя

Після чого потрібно підключитися до точки доступу. Використаємо цикл для цього, якщо не вийде підключитися з першого разу, він пробіжить знову і так до тих пір поки не підключиться до WiFi мережі:

```
for (int i=0;i<2;i++) {
   WiFi.begin(ssid, password);
   delay(1000);
   Serial.println("");
   Serial.print("Connecting to ");
   Serial.println(ssid);
long int StartTime=millis();
   while (WiFi.status() != WL_CONNECTED) {
      delay(500);
      if ((StartTime+5000) < millis()) break;
 }
   if (WiFi.status() == WL_CONNECTED) {
     Serial.println("");
```

```
    Serial.println("STAIP address: ");
    Serial.println(WiFi.localIP());
    Serial.println("");
  }
 }
```

Якщо вийшло підключитися і статус мережі – підключено, тоді контролер скаже про це, вивівши в серійний порт отриману IP-адресу по якій можна підключитися до нього.

Наступний кусок коду показує, що клієн постійно слухає канал, якщо там щось прийшло, то виконує функцію getCommand().

```
while (client.connected()) {
    if (client.available()) {
      char c = client.read();
getCommand(c);
 }
}
```

Якщо прийшла команда stop, то контролер припиняє дану сесію:

```
if ((currentLine.indexOf("/?")!=-1)&&(currentLine.indexOf(" HTTP")!=-1)) {
        if (Command.indexOf("stop")!=-1) {
          client.println();
          client.println();
          client.stop();
        }
}
```

Дивлячись на дані результати емуляції в яких ESP32 без проблем підключилася до мережі WiFi, та отримала і опрацювала всі відправлені їй команди. Можна сказати, що контролер справився зі своєю поставленою задачею. Отже можна дійти до висновку, що схема приладу працює належним чином.

Далі можна приступа до основного і найбільшого етапу – програмна реалізація приладу для контрлю доступу. Після чого буде збірка тестового прототипу для перевірки його в реальних умовах.

3.2 Реалізація програмної частини

Програмна частина приладу хоча і складається з різних частин, але з’єдна глобальними змінними для обміну данити для функціонування один з одним.

Грубо кажучи код можна поділити на такі частини:

– web-інтерфейс з можливістю налаштування камери;

– відправка повідомлень в Telegram та опрацювання відповідей;

– обробка та порівняння обличчя з еталоним.

На початок розглянемо частину з web-інтерфейсом. Так як мережі WiFi якмось чином може не бути, або вона продаде ів непідходячий моменм, потрібно щоб було доступно майже весь функціолнал прилад. Оскільки даний модуль може працювати в режиі і клієнта і точки доступу одночасно – це чудове рішення поставленої задачі.

Для цього нам спершу потрібно створити зміні з логіном і паролем від вже існуючої точки доступу, а також придумати логін пароль для нової точки доступу, катра підніметься на самі ESP32:

```
const char* ssid = "STA_login";
const char* password = "STA_password";

const char* apssid = "AP_login";
const char* appassword = "AP_password";
```

Для роботи WiFi потрібно підключити деякі бібліотеки, а саме WiFi.h, WiFiClientSecure.h. На самому початку потрібно задати швидкість обміну данимим в моніторі порту та режим дебагу, для перевірки того, що туди приходить і навпаки:

```
Serial.begin(115200);
 Serial.setDebugOutput(true);
 Serial.println();
```

Звичайно і про камеру не забули, для її роботи також потрібні деякі бібліотеки підключити: esp_camera.h; soc/soc.h; soc/rtc_cntl_reg.h.

Післятого, як задали швидкість серійному порту, топрібно ініціалізувати камеру:

```
esp_err_t err = esp_camera_init(&config);
 if (err != ESP_OK) {
  Serial.printf("Camera init failed with error 0x%x", err);
  delay(1000);
  ESP.restart();
 }
```

Як бачимо, якщо по якихось причинах камера не запустилася, контролер перезапуститься і все почне заново. Тому моменту, щось камера не прцює, не буде. Оскільки це основна задача даного модуля.

Як було сказано на початку, що даний модуль працює в двох режимах обосно (AP і STA), тому пробріно при старті задати команду йому:

```
WiFi.mode(WIFI_AP_STA);
```

При цьому потрібно буде запустити ці два режими за допомгою ось цих команд:

```
WiFi.begin(ssid, password);
WiFi.softAP((WiFi.localIP().toString()+"_"+(String)apssid).c_str(), appassword);
```

Після старта модуля в серійний порт монітору, він видасть дві IP-адреси. Одна зниз це та, що модулю видала точка доступу, а друга – це локальна. По якій можна підлючатися, якщо точка доступу не доступна по деяким причинам.

Для подальшої роботи, потрібно створити Telegram бота. Його створення не займає багато часу, чи навичків для чього. Для початку потрібно скачати Telegram на телефон, якщо він не встановлений.

– Далі відкрийте Telegram і виконайте наступні кроки, щоб створити бота Telegram. Спочатку знайдіть «botfather» та натисніть BotFather, як показано нижче. Або відкрийте це посилання t.me/botfather у своєму смартфоні.

– Повинне відкритися наступне вікно, і вам буде запропоновано натиснути кнопку /start.

– Потім введіть /newbot та дотримуйтесь інструкцій, щоб створити свого бота. Дайте йому ім’я та ім’я користувача.

– Якщо ваш бот успішно створено, ви отримаєте повідомлення з посиланням для доступу до бота та маркера бота. Збережіть маркер бота, оскільки він вам знадобиться, щоб ESP32 міг взаємодіяти з ботом.

Створення бота на цьому закінчується, залишається отримати ID бота, щоб повідомлення приходили тільки собі і більше нікому. Для цього потрібно:

– У своєму обліковому записі Telegram знайдіть «IDBot» або відкрийте це посилання t.me/myidbot у своєму смартфоні.

– Почніть розмову з цим ботом і введіть /getid. Ви отримаєте відповідь зі своїм ідентифікатором користувача. Збережіть цей ідентифікатор користувача, оскільки він вам знадобиться пізніше.

Після створення бота таотримання ID чату, потрібно створити зміні для бота:

```
String myToken = "botToken";
String myChatId = "chatID";
```

В даному модулі є роз’єм для SD флешки, на якій можна роботи фото і туди зберіза, можна зарання тути перенести фото і з них робити порівняння обличь. Але це напевно не безпечно, оскільки кожен хто проходитеме біля даного модуля, може нашколити і змінити якісь днані на ній.

Тому було вирішено скористатися онлайн сервісом, на якому зберігаються всі фото користувачів (еталоні фото). Було обрано сервіс github.com.

На даному сервісі було створено новий репозиторій, в якому знаходиться дві особо і в кожній з них є по 2 еталоних фото. Ось так виглядає весь шлях до фото:

```
const faceImagesPath = 'https://SecurityAdmin.github.io/esp32-cam/face-control/facelist/';
const faceLabels = [User1, User2];
```

```
faceImagesCount = 2 ;
```

Тут видно, що вказано тільки два користувача User1 і User, в який є по два фото в папці, про що повідомляє зміна faceImagesCount.

Після всі цих маніпуляцій, потріно прокинути алгоритм розпізнавання оличчя контолеру. Схема обличь лежить в корені репозиторія:

```
const modelPath = 'https:// SecurityAdmin.github.io/esp32-cam/face-control/';
```

Також для роботи розпізнавання модулю потріно скачати деякі файли, які так само знаходяться в корені репозиторія:

```
Promise.all([
        faceapi.nets.faceLandmark68Net.load(modelPath),
        faceapi.nets.faceRecognitionNet.load(modelPath),
        faceapi.nets.ssdMobilenetv1.load(modelPath)
      ])
```

Таким чином завантажуються і самі еталоні фото:

```
if (!labeledFaceDescriptors) {
          message.innerHTML = "Loading face images...";
          labeledFaceDescriptors = await loadLabeledImages();
          message.innerHTML = "";
        }
```

Застосовуємо фільтр Гауса на зашумлене одноканального зображення:

```
double** getGaussian(int height, int width, double sigma) {
   double** filter = new double * [width];
   for (size_t i = 0; i < width; ++i) {
      filter[i] = new double[height];
   }
   double sum = 0;
   for (int i = 0; i < height; i++) {
      for (int j = 0; j < width; j++) {
         filter[i][j] = exp(-(i * i + j * j) / (2 * sigma * sigma)) / (2 * 3.14 * sigma * sigma);
         sum += filter[i][j];
      }
   }
```

```
    return filter;
}
```

Функція getGaussian створює масив заданого розміру (height, width), у якому зберігаються значення фільтра Гауса. Формула для розрахунку значень така -g(x, y) = 1/(2*pi*sigma^2)*exp(-(x^2 + y^2)/(2*sigma^2)), де сігма - стандартне відхилення, а g(x,y) - значення у фільтрі. Тип double вибраний для того, щоб при множенні на unsigned char (тип даних у вхідному зображенні) не виходили нулі. Image – це така структура:

```
struct Image {
    unsigned char** data = nullptr;
    size_t rows = 0;
    size_t cols = 0;
};
```

Виклик функції відбувається так:

```
double sigma = FindSigma(image);
double** filter = getGaussian(5, 5, sigma);
Image gauss = GaussianBlur(image, filter, 5, 5);
```

Виходить ось такий принцип роботи програми, що при старті приладу, він підключається до мережі WiFi, та піднімає свою локальну точку доступу, яка Частвоко вріза по функціоналу.

Після чого прилад скачує деякі схеми та алгоритми розпізнаваня обличчя і самі еталоні фото працівників, потім починає працювати в штатному режимі.

Якщо працівник підійшов до камери, вона робить знімок і порівнює його з еталоним і вирішує пускати дану людину чи ні.

Якщо система не розпізнала когось, тоді відправляється повідомлення в Telegram в якому знаходиться саме фото людини яку система не розпізнала і текстове повідомлення, що ця людини не знайома взагалі.

Якщо система озпізнала а особі працівника, тоді вона пропускає далі і відправляє в Tekegram повідомлення з іменем тієї людини, котрій система дозволила пройти.

3.3 Рекомендації з використання та налаштування приладу

В даному підрозділі описано основний функції приладу, а також рекомендації для його встановлення та використання.

Дана версія приладу призначена для внутрінього встановлення для забезпечення контролю і управлінням доступом до приміщення.

Після встановлення приладу і підключення до нього електромагніту, можна також налаштувати його за допомогою веб-інтерфейсу, а конкретніше, налаштувати камеру.

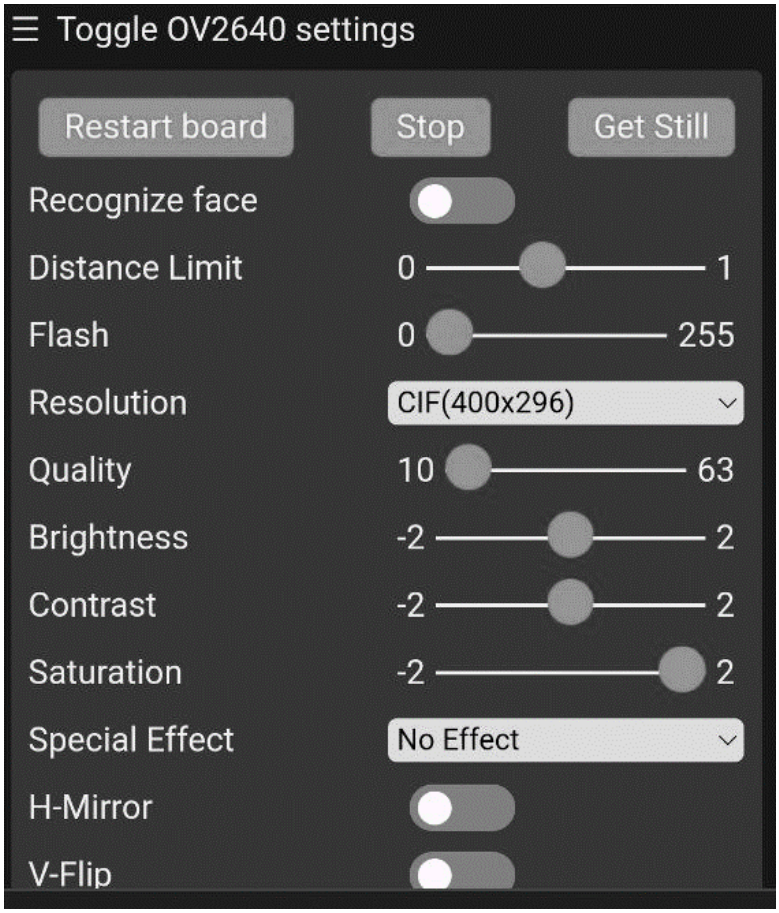

Рисунок 3.2 – Зовнійшій вигляд web-інтерфейсу

В даних налаштуваннях можна відрегклюватися чіткість, насиченість зображення, можна поміняти розширення стріму, перезавантадити пристрій, призупинити все, відзеркалити зображення в двох площинах. Також можна вибрати одни з декількох ефектів.

В даному меню налаштовується по факту тільки камера одна. Ще ж можливітсь ввімкнути світлодіод, який слугує тут, як вспишка або фонарь, щоб камера чткіше бачила лице людини. Також тут можна змінити пін реле, яке підключене до пристроя.

Додатковим функціоналом є те, що можна підключити ще сервопривід в майбутньому, який в теорії зможе автоматизувати відеритття та закриття дверей.

Після налаштувань камери пристрій в загальному, стає готовий до експлуатації. За винятком, що потрібно заповнити базу еталонів фотокартками співробітників.

Якщо брати в загальному, то в кожного підприємства, або групи людей завжди буде потрібно програмувати мікроконтролер, аде всі ці зміни на надий момент вшиваються в контролер.

В подальшому доопрацюванні даного приладу планується удосконалити програмно його, щоб узагальнити для всіх підприємст і груп людей даний прилад. Як то кажуть, щоб все було одразу з «коробки» готове для використання при мінімальному налаштуванні.

3.4 Випробування прототипу приладу для контролю доступу до приміщення з використанням розпізнавання обличчя

В даному розділі проведемо перевірку працездатності приладу для конролю доступа з використанням розпізнавання облччя.

Для перевірки працездатності і правильної роботи приладу знадобиться: джерело живлення 12В, та 5В, або можна використати понижаючий модуль

щоб з 12В зробити 5В для живлення контролера. 12В потрібні лише для живлення електромагніту, або електрозамка.

На рисунку 3.3 зображе всі елементи, які потрібні для збору системи контролю доступу з використанням розпізнавання обличчя.

Щоб зібрати такий прилад потрібно придбати лише п’ять елементів.

Під цифрою «1» знаходиться програматор CH341A, він слугує не тільки для прграмування в даному випадку, а й для дебагу і отримання IP-адресу, яку отримав прилад від точки доступа.

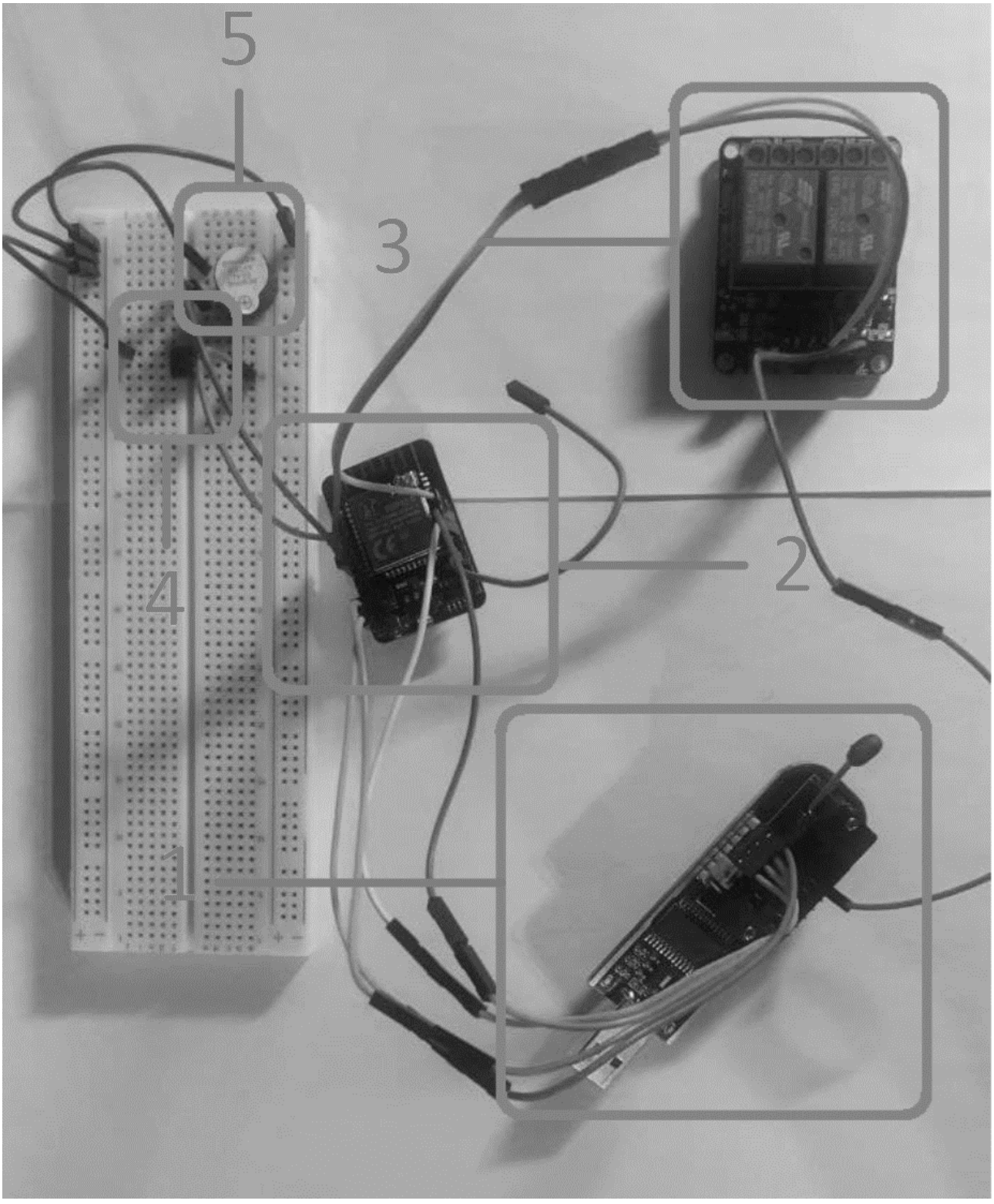

Рисунок 3.3 – Зовнійшій всіх осноаних елементів зібаних на макетній платі

Під цифрою «2» знаходиться «серце» приладу – ESP32-CAM. Даний модуль сумістив в собі весь функціонал, як контроль доступу, так і система оповіщення і модуль розпізнавання обличь.

Під цифрою «3» знаходиться модуль реле, який не потребує свого представлення. Так як його робота просто і не велеки, хоча він відіграє важливу роль в даній систему контроля домтупу.

Під цифрою «4» знаходиться тактова кнопка, за допомогою якої можна вийти з території. Кнопка відправляє сигнал до контролера, а контролер в свою чергу відправляє його до реле, щоб воно відкрило двері.

Підцифрою «5» знаходиться зумер, який слугує чисто для звукової візуалізаці при розпізнаванні людини звучить короткий сигнал, при невідомій людині звучить довгий звуковий сигнал.

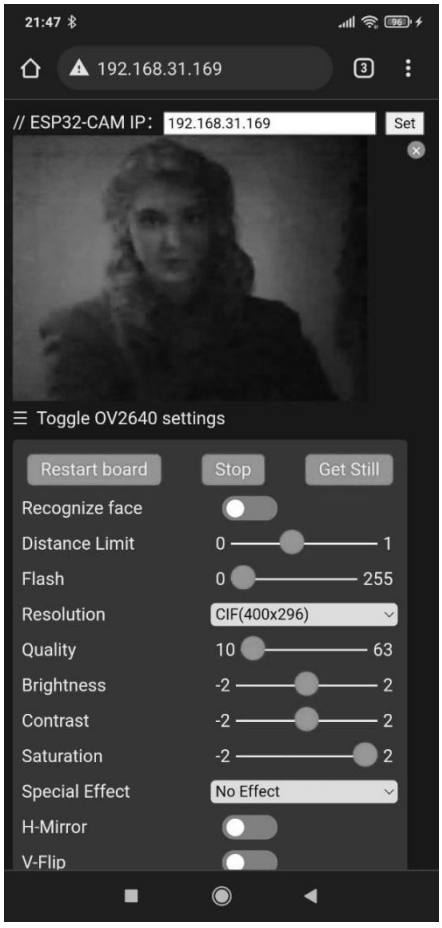

Рисунок 3.4 – Зовнішній web-інтерфейсу при розпізнаванні обличчя

Після налаштувань камери через web-інтерфейс – прилад готовий до експлуатації. Коли працівник підійде до камери, прилад одразу розпочне його порівнювати з еталонами, які є в його базі даних (рис.3.4).

Після того, як прилад розпізнав правцівника, він відправляє сигнал до реле, яке в свою чергу відмикає двері. І правник без проблем може топрапити до приміщення.

Адже RFID мітки забирають багато часу щоб їх просканувати. Оскіль мітки можуть загубитися, працівник міг забути її дома і т.д. Система з розпізнаванням обличчя максимально комфортна, як для працівників, так і для власників. Адже економить велику кількість часу і коштів.

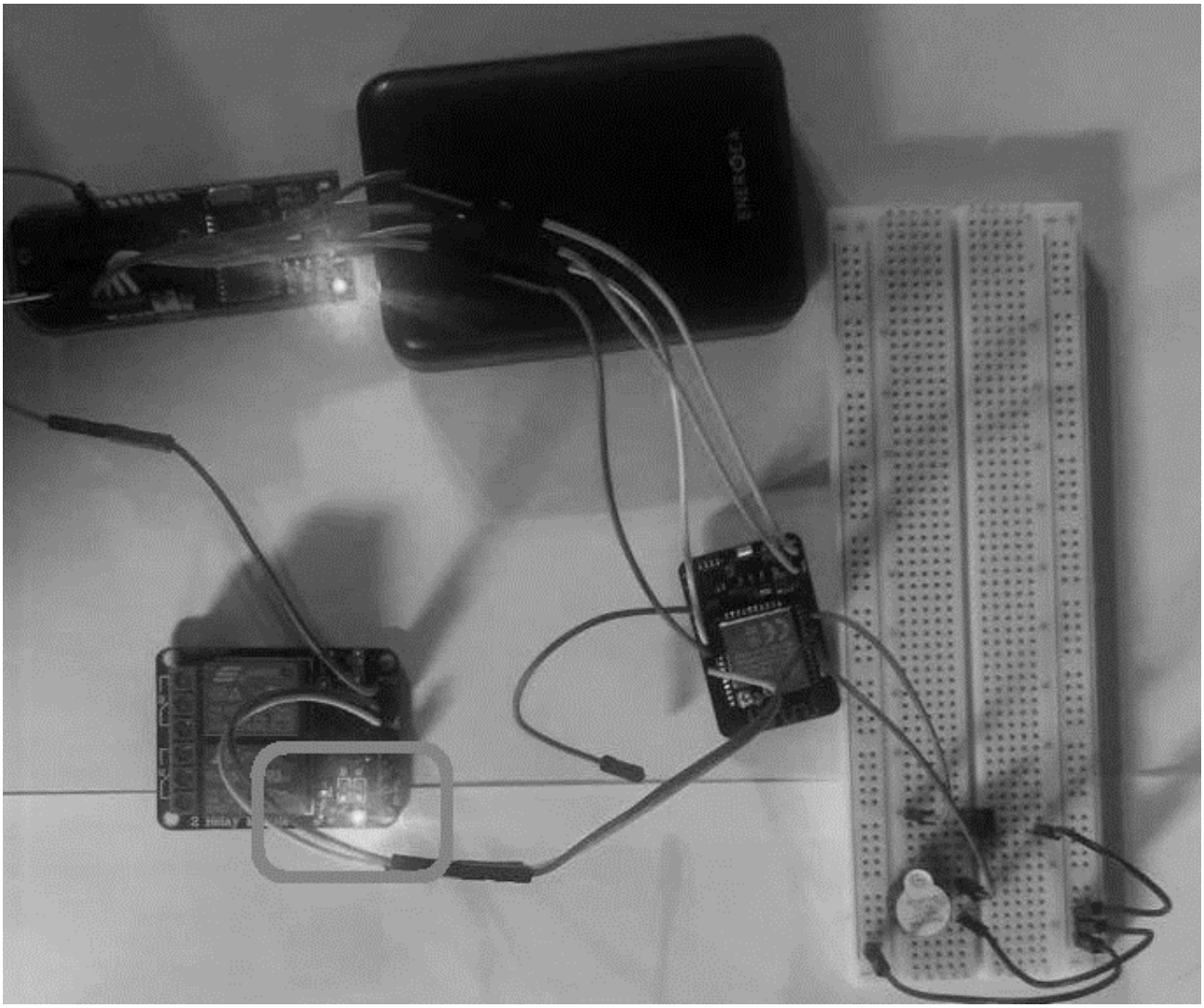

Рисунок 3.5 – Зовнішній вигляд спрацювання реле при розпізнаванні обличчя

Якщо система не розпізнала працівника, або це був зловмисник, вона одразу відправляє фото невідомого в Telegram (рис.3.5) і повідомляє, що це невідома їй людина. За допомого даного бота, можна віддалено керувати приладом. На фото можна замінити чотири кнопки (open, still, ledon, ledoff).

З двома останіми кнопками все просто, ledon вмикає слітлодіод, а ledoff його вимикає. Кнопка open відкриває двері примусово, якщо потрібно впустити певну людину на територію.

Кнопка still запрошу в контролера фото в ланий момент. Контролер робить фото і відправляє його в Telegram. Таким чином можна дізнатися, що коїться на вході в любий момент.

Також можна підключитися до web-інтерфейсу і дивитися відеопотік з камери в любий комфортний час, головне бути в одній мережі з приладом контролю доступа.

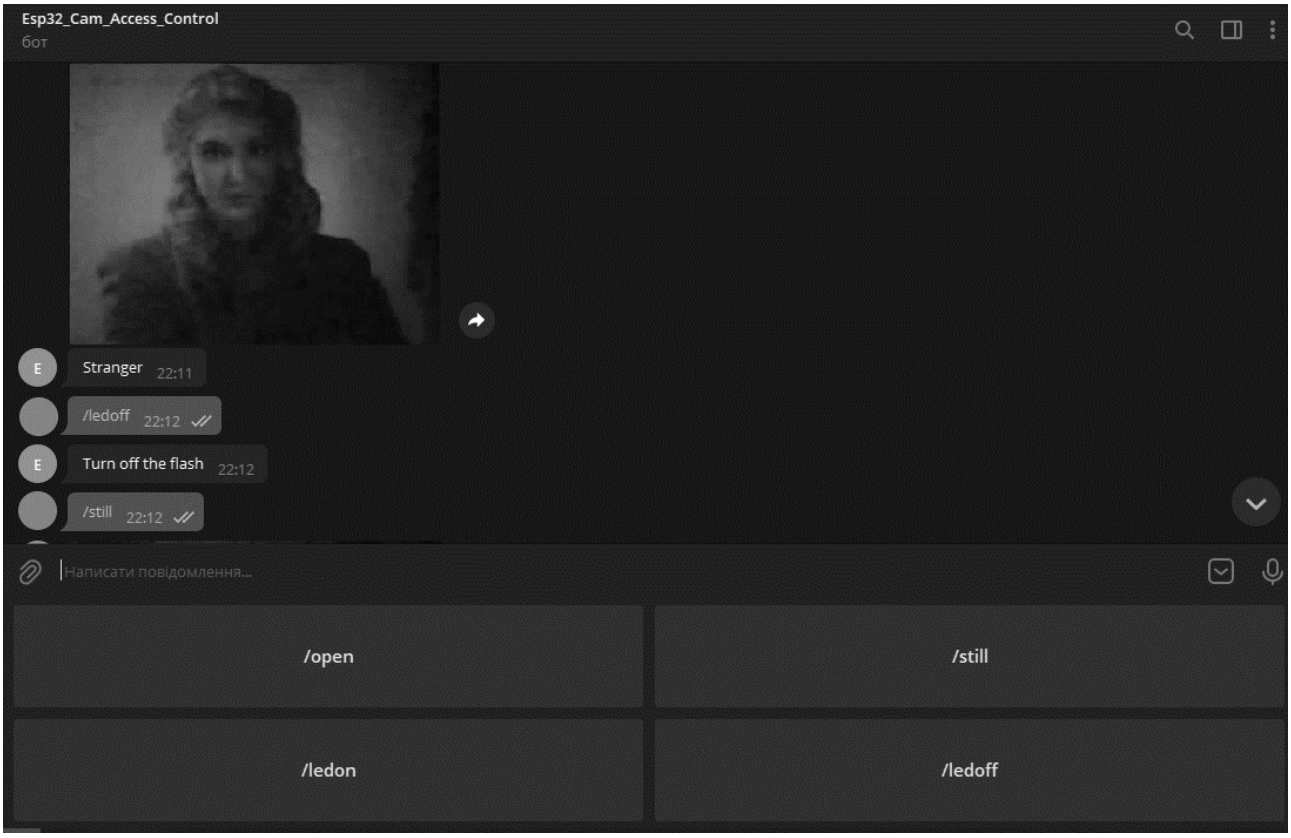

Рисунок 3.5 – Зовнішній вигляд Telegram бота

Фінальний варіант реалізації приладу отримав модуль Wi-Fi ESP32-CAM, який оснащений:

– Контролер: ESP32, 2 ядра, 32-біт;

– Робоча частота процесора: 240 МГц / 600 DMIPS;

– Оперативна пам'ять: Вбудована: 520 КБ / Зовнішня: 4 МБайт;

– Підтримувані інтерфейси: UART/SPI/iec/PWM/ADC/DAC;

– Вбудований Lwip і FreeRTOS;

– Підтримка STA/AP/STA + AP Робочий режим;

– Підтримка Smart Config/AirKiss розподіленої мережі;

– Напруга живлення: 5В;

– Модуль камери: OV2640;

– Кількість пікселів: 2 Мпіксел;

– Кут огляду: 66 градусів;

– Максимальна роздільна здатність матриці (відео): UXGA 1600x1200 (15 fps) / SVGA 800x600 (30 fps);

– Підтримка форматів відеозахвату: 8/10-bit Raw RGB, YUV(422/420), RGB565/555;

– Підтримка камер: OV2640 і OV7670;

– Зберігання даних: micro-SD карта пам'яті.

3.5 Висновки

Вданому розділі було земульовано, та реалізовано прилад контролю доступу до приміщення з використанням розпізнавання обличчя. Були проведені тести в онлайн конструкторі «Autodesk Circuits», котрі показали ефективність та працездатність приладу. Після проведених тестів в програмі, було реалізовано та запрограмовано прилад в цілому для контролю доступу.

Також було розглянуто процес програмування контролера для певних дій та функціоналу в цілому. Було описано детальне налаштування приладу для кінцевого використання, а також розписано детальну інструкцію з експлуатації приладу в тих чи інших випадках. Тестування приладу в реальних

умовах показало, що прилад відповідає встановленим умовам і виконує всі поставлені на нього функції.

1 4 ЕКОНОМІЧНА ДОЦІЛЬНІСТЬ СТВОРЕННЯ ПРИЛАДУ ДЛЯ КОНТРОЛЮ ДОСТУПУ З ВИКОРИСТАННЯМ FACE-ІДЕНТИФІКАЦІЇ КОРИСТУВАЧА

4.1 Проведення наукового аудиту науково-дослідної роботи

Суб'єктами виконання науково-дослідних робіт і розробок є наукові організації, науково-дослідні центри при закладах вищої освіти, науково- дослідні, проектно-конструкторські організації, експериментальні підприємства, а також науково-виробничі об'єднання.

Метою фундаментальних і частково пошукових досліджень не є одержання продукту, виробу або послуги, що можуть стати товаром і оформитися у вигляді певного комерційного інвестиційного проекту. Однак на їхній основі здійснюється генерація ідей, які можуть трансформуватися в проекти науково-дослідних і дослідно-конструкторських робіт. Тому пошукові роботи можуть мати деяку комерційну цінність.

Наукова і науково-технічна результативність НДР не може бути оцінена з використанням методу дисконтування грошових потоків, за винятком випадків, коли дослідження мають вартісні характеристики результату НДР як наукової інформації, тому у деяких випадках результати дослідження можуть мати вартісні характеристики результату науково-дослідної роботи як наукової інформації, яку купує замовник. Тобто у такому випадку може виникнути фактична ефективність науково-дослідної роботи.

Для наукових і пошукових науково-дослідних робіт зазвичай здійснюють оцінювання наукового ефекту.

Основними ознаками наукового ефекту науково-дослідної роботи є новизна роботи, рівень її теоретичного опрацювання, перспективність, рівень розповсюдження результатів, можливість реалізації. Науковий ефект НДР можна охарактеризувати двома показниками: ступенем наукової новизнита рівнем теоретичного опрацювання.

Значення показників ступеня новизни і рівня теоретичного опрацювання науково-дослідної роботи в балах наведено в табл. 4.1 та 4.2.

Таблиця 4.1 – Показники ступеня новизни науково-дослідної роботи

Ступінь новизни	Характеристика ступеня новизни	Значення показника ступеня новизни, бали
1	*2*	*3*
Принципово нова	Робота якісно нова за постановкою задачі і ґрунтується на застосуванні оригінальних методів дослідження. Результати дослідження відкривають новий напрям в цій галузі науки і техніки. Отримано принципово нові факти, закономірності; розроблено нову теорію. Створено принципово новий пристрій, спосіб, метод	60…100
Нова	Отримано нову інформацію, яка суттєво зменшує невизначеність наявних значень (по-новому або вперше пояснено відомі факти, закономірності, впроваджено нові поняття, розкрито структуру змісту). Проведено суттєве вдосконалення, доповнення і уточнення раніше досягнутих результатів	40…60
Відносно нова	Робота має елементи новизни в постановці задачі і методах дослідження. Результати дослідження систематизують і узагальнюють наявну інформацію, визначають шляхи подальших досліджень; вперше знайдено зв’язок (або знайдено новий зв’язок) між явищами. В принципі, відомі положення поширено на велику кількість об’єктів, в результаті чого знайдено ефективне рішення. Розроблено більш прості способи для досягнення відомих результатів. Проведено часткову раціональну модифікацію (з ознаками новизни)	10…40
Традиційна	Робота виконана за традиційною методикою. Результати дослідження мають інформаційний характер. Підтверджено або поставлено під сумнів відомі факти та твердження, які потребують перевірки. Знайдено новий варіант рішення, який не дає суттєвих переваг порівняно з існуючим	2…10
Не нова	Отримано результат, який раніше зафіксований в інформаційному полі та не був відомий авторам	1…2

Таблиця 4.2 – Показники рівня теоретичного опрацювання науково-дослідної роботи

Характеристика рівня теоретичного опрацювання	Значення показника рівня теоретичного опрацювання, бали
1	2
Відкриття закону, розробка теорії	80…100

Продовження таблиці 4.2

1	2
Глибоке опрацювання проблеми: багатоаспектний аналіз зв'язків, взаємозалежності між фактами з наявністю пояснень, наукової систематизації з побудовою евристичної моделі або комплексного прогнозу	60…80
Розробка способу (алгоритму, програми), пристрою, отримання нової речовини	20…60
Елементарний аналіз зв'язків між фактами та наявною гіпотезою, класифікація, практичні рекомендації для окремого випадку тощо	6…20
Опис окремих елементарних фактів, викладення досвіду, результатів спостережень, вимірювань тощо	1…5

Показник, який характеризує науковий ефект, визначається за формулою:

$$\mathrm{E}_{\text{нау}} = 0{,}6 \cdot k_{\text{нов}} + 0{,}4 \cdot k_{\text{теор}},,,, \qquad (4.1)$$

де $k_{нов}$, $k_{теор}$ – показники ступенів новизни та рівня теоретичного опрацювання науково-дослідницької роботи, бали;

0,6 та 0,4 – питома вага (значимість) показників ступеня новизни та рівня теоретичного опрацювання науково-дослідної роботи

$$\mathrm{E}_{\text{нау}} = 0{,}6 \cdot 50 + 0{,}4 \cdot 60 = 54,,,$$

Визначення характеристики показника $E_{нау}$ проводиться на основі висновків експертів, виходячи з граничних значень, які наведено в табл. 2.3.

Таблиця 4.3 – Граничні значення показника наукового ефекту

Досягнутий рівень показника	Кількість балів
Високий	70…100
Середній	50…69
Достатній	15…49
Низький (помилкові дослідження)	1…14

Відповідно до таблиці 4.3 значення показника наукового ефекту має середній рівень і становить 54 бали, адже для розробки приладу було проведено суттєве вдосконалення, доповнення і уточнення раніше досягнутих результатів.

4.2 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту є оцінювання науково-технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності, тобто під час виконання магістерської кваліфікаційної роботи.

Для проведення комерційного та технологічного аудиту залучаємо 3-х незалежних експертів, які є провідними викладачами випускової кафедри.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу здійснімо із застосуванням п’ятибальної системи оцінювання за 12-ма критеріями, наведеними в табл. 4.4.

Таблиця 4.4 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
№	0	1	2	3	4
1	2	3	4	5	6
Технічна здійсненність концепції					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку

3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів

Продовження таблиці 4.4

1	2	3	4	5	6
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної Динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна Конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витрачати значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої	Необхідно отримання великої кількості дозвільних	Процедура отримання дозвільних документів для	Необхідно тільки повідомлення відповідним органам про	Відсутні будь-які регламентні обмеження на виробництво та

	кількості дозвільних документів на виробництво та реалізацію продукту	документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	виробництва та реалізації продукту вимагає незначних коштів та часу	виробництво та реалізацію продукту	реалізацію продукту

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки потрібно зведемо до таблиці 4.5.

Таблиця 4.5 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки

Критерії	Експерт		
	1	2	3
	Бали:		
1. Технічна здійсненність концепції	4	2	3
2. Ринкові переваги (наявність аналогів)	3	3	4
3. Ринкові переваги (ціна продукту)	4	2	3
4. Ринкові переваги (технічні властивості)	3	4	2
5. Ринкові переваги (експлуатаційні витрати)	2	3	2
6. Ринкові перспективи (розмір ринку)	3	2	2
7. Ринкові перспективи (конкуренція)	3	2	1
8. Практична здійсненність (наявність фахівців)	2	3	3
9. Практична здійсненність (наявність фінансів)	3	4	2
10. Практична здійсненність (необхідність нових матеріалів)	3	2	3
11. Практична здійсненність (термін реалізації)	2	1	3
12. Практична здійсненність (розробка документів)	3	3	4
Сума балів	$СБ_1=35$	$СБ_2=31$	$СБ_3=32$
Середньоарифметична сума балів $СБ_с$	$СБ_с = \frac{\sum_{i=1}^{3} СБ_1}{3}=32,67$		

За результатами розрахунків, наведених в таблиці 4.5, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в табл. 4.6.

Таблиця 4.6 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів СБ, розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41…48	Високий
31…40	Вищий середнього
21…30	Середній
11…20	Нижчий середнього
0…10	Низький

Згідно проведених досліджень рівень комерційного потенціалу розробки становить 32,67 бали, що, згідно таблиці 4.5, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки вище середнього. Данна розробка дає можливість здійснювати контроль доступу до приміщення за допомогою face-ідентифікації користувача і за допомогою віддаленого керування доступом. Також система повідомляє власника про сторонніх осіб за допомогою месенджера Telegram. Однією з практичних функцій на відмінну від вже існуючих аналогів є віддалене керування приладом за допомого месенджера Telegram, що дозволяє дізнатися про сторонніх осіб заранння. Прилад з даним функціоналом, який включає в себе одразу контроль доступу з використанням біометричних даних, швидке оповіщення про стороніх осіб і можливістю віддаленого керування є дуже практичним і дешевим варіантом для покриття завдань: контролю доступом.

4.3 Розрахунок витрат на здійснення науково-дослідної роботи

Витрати, пов’язані з проведенням науково-дослідної, дослідно-конструкторської, конструкторсько-технологічної роботи, створенням дослідного зразка і здійсненням виробничих випробувань, під час планування,

обліку і калькулювання собівартості науково-дослідної роботи групуються за такими статтями:

- витрати на оплату праці;
- відрахування на соціальні заходи;
- паливо та енергія для науково-виробничих цілей;
- витрати на службові відрядження;
- спецустаткування для наукових (експериментальних) робіт;
- програмне забезпечення для наукових (експериментальних) робіт;
- витрати на роботи, які виконують сторонні підприємства, установи і організації;
- інші витрати;
- накладні (загальновиробничі) витрати.

До статті «Витрати на оплату праці» належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці, також будь-які види грошових і матеріальних доплат, які належать до елемента «Витрати на оплату праці».

Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників (Зо) розраховують відповідно до посадових окладів працівників, за формулою:

$$З_о = \sum_{i=1}^{k} \frac{М_{ni} \cdot t_i}{Т_р}, \quad (4.2)$$

де *k* – кількість посад дослідників, залучених до процесу досліджень;

$М_{ni}$ – місячний посадовий оклад конкретного дослідника, грн;

t_i – кількість днів роботи конкретного дослідника, дн.;

T_p – середня кількість робочих днів в місяці, T_p=*21…23* дні.Проведені розрахунки бажано звести до таблиці.

Таблиця 4.7 – Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Кількість днів роботи	Витрати на заробітну плату, грн
Керівник проекту	10500	457	30	13 710
Науковий співробітник	8500	370	30	11 100
Лаборант	6500	282	30	8 460
Всього				33 270

Витрати на основну заробітну плату робітників ($З_р$) за відповідними найменуваннями робіт розраховують за формулою:

$$З_р = \sum_{i=1} C_i \cdot t_i, \quad (4.3)$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за вико-нану відповідну роботу, грн/год;

t_i – час роботи робітника на виконання певної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C_i можнавизначити за формулою:

$$С_і = \frac{М_м \cdot К_і \cdot К_с}{Т_р \cdot Т_{зм}}, \quad (4.4)$$

де $М_М$ – розмір прожиткового мінімуму працездатної особи або мінімальної місячної заробітної плати (залежно від діючого законодавства), грн;

K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду;

K_c – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

T_p – середня кількість робочих днів в місяці, приблизно $T_p = 21…23$ *дні*;

$t_{зм}$ – тривалість зміни, год.

$$С_п = \frac{6500 \cdot 1{,}8 \cdot 2}{23 \cdot 8} = 23\ 400 \text{ грн}$$

Таблиця 4.8 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
Програміст мікроконтролерів	184	6	2	125,12	23 400
Схемотехнік	184	5	1,7	108,09	19 890
Монтажник	184	4	1,5	95,38	17 550
Всього					60 840

Додаткова заробітна плата розраховується як 10 … 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$З_{дод} = З_о + З_р \cdot \frac{Н_{дод}}{100\%}, \quad (4.5)$$

де $Н_{дод}$ – норма нарахування додаткової заробітної плати.

$$З_{дод} = (60840 + 33270) \cdot 0{,}1 = 9\ 411 \text{ грн.}$$

До статті «Відрахування на соціальні заходи» належать відрахування внеску на загальнообов'язкове державне соціальне страхування та для здійснення заходів щодо соціального захисту населення (ЄСВ – єдиний соціальний внесок).

Нарахування на заробітну плату дослідників та робітників розраховується як 22% від суми основної та додаткової заробітної плати дослідниківі робітників за формуло:

$$З_н = (З_о + З_р + З_{дод}) \cdot \frac{Н_{зп}}{100\%}, \quad (4.6)$$

де $Н_{зп}$ – норма нарахування на заробітну плату.

$$З_{дод} = 60840 + 33270 + 9411) \cdot 0{,}22 = 22\,774{,}62 \text{ грн.}$$

Витрати на комплектуючі вироби *($К_в$)*, які використовують при дослідженні нового технічного рішення, розраховуються, згідно з їхньою номенклатурою, за формулою:

$$К_в = \sum_{=1}^{n} Н_j \cdot Ц_j \cdot К_{jі} \quad (4.7)$$

де $Н_j$ – кількість комплектуючих *j*-го виду,шт.;

$Ц_j$ – покупна ціна комплектуючих *j*-го виду, грн;

$К_j$ – коефіцієнт транспортних витрат, ($К_j$ *= 1,1 … 1,15*). Проведені розрахунки зведемо до таблиці 4.9.

Таблиця 4.9 – Витрати на комплектуючі

Найменування комплектуючих	Кількість, шт.	Ціна за штуку, грн	Сума, грн
- SP32-CAM	1	219	219
Модуль реле 5В 10А з опторозв'язкою	1	38	38
Програматор EEPROM на CH341A	1	99	99
Тактова кнопка	1	5	5
Активний динамік	1	17	17
Комплект перемичок	1	69	69
DC-DC понижуючий перетворювач MP1584 3A	1	25	25
Блок живлення	1	370	370
Всього			842

Витрати на комплектуючі становитимуть:

$К_в$= 842·1,1 = 926,2 грн.

Спецустаткування для наукових (експериментальних) робіт

Вартість спецустаткування визначається за прейскурантом гуртових цін або за даними базових підприємств за відпускними і договірними ці- нами. До балансової вартості устаткування окрім прейскурантної вартості входять витрати на його транспортування і монтаж, тому ці витрати беруться додатково в розмірі 10…12% від вартості устаткування.

Балансову вартість спецустаткування розраховують за формулою:

$$B_{спец}$$

де $Ц_i$ – ціна придбання одиниці спецустаткування даного виду, марки, грн;

Спр - кількість одиниць устаткування відповідного найменування, які придбані для проведення досліджень, шт.

$К_i$ - коефіцієнт, що враховує доставку, монтаж, налагодження устаткування тощо, ($К_i$ =1,10 ….1,12)

k = кількість найменувань устаткування

Таблиця 4.10– Витрати на придбання спецустаткування по кожному виду

Найменування устаткування	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Комп'ютер	2	20 000	40 000
Принтер	1	1 600	1 600
Всього			41600

Витрати на придбання спецустаткування становитимуть:

$В_{спец}$= 41600·1,1=45 760грн.

Програмне забезпечення для наукових робіт

До статті «Програмне забезпечення для наукових робіт» належать витрати на розробку та придбання спеціальних програмних засобів і

програмного забезпечення, (програм, алгоритмів, баз даних) необхідних для проведення досліджень, також витрати на їх проектування, формування та встановлення.

До балансової вартості програмного забезпечення входять витрати на його інсталяцію, тому ці витрати беруться додатково в розмірі 10…12% відвартості програмного забезпечення.

Балансову вартість програмного забезпечення розраховують за формулою:

$$В_{прг} = \sum_{i=1}^{к} Ц_{iпрг} \cdot С_{прг} \cdot К_{i} \quad (4.9)$$

де $Ц_{iпрг}$ – ціна придбання одиниці програмного засобу цього виду, грн;

$С_{прг}$ - кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

$К_i$ – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($К_i$ = *1,10…1,12);*

k – кількість найменувань програмних засобів. Отримані результати зведемо до таблиці 4.11

Таблиця 4.11 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Операційна система Windows	2	3900	7800
Visual Studio Professional	2	2600	5200
Всього			13000

Витрати на придбання програмних засобів становитимуть :

$$В_{прог}=13000 \cdot 1,1=14\ 300 \text{ грн.}$$

Амортизація обладнання, комп’ютерів та приміщень А, які використовувалися під час (чи для) виконання даного етапу роботи розраховуються за формулою:

$$А = \frac{Ц}{Тв} \cdot \frac{tk}{12}, \quad (4.10)$$

де Ц – загальна балансова вартість всього обладнання, комп'ютерів, приміщень тощо, що використовувались для виконання даного етапу роботи, грн.;

$Т_в$ – термін корисного використання, роки;

t_k – термін використання обладнання, приміщень тощо, місяці.

Розрахунки амортизаційних витрат для даного програмного забезпечення зведені в табл. 4.12.

Таблиця 4.12 – Розрахунок амортизаційних відрахувань

Найменування програмного забезпечення	Кількість	Балансова вартість, грн.	Термін корисного використання, роки	Термін використання, міс.	Величина амортизаційних відрахувань, грн.
Комп'ютер	2	20 000	3	2	2 222
Принтер	1	1600	3	2	89
Операційна система Windows	2	3900	3	2	434
Visual Studio Professional	2	2600	3	2	255
	Всього:				3 000

Витрати на силову електроенергію (B_e) розраховують за формулою:

$$В_е = В \cdot П \cdot Ф \cdot К_п, \quad (4.11)$$

де *В* – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії);

П– встановлена потужність обладнання, кВт;

Ф – фактична кількість годин роботи обладнання, годин.

$К_п$ – коефіцієнт, що враховує використання потужності, $К_{впі} < 1$;

$$В_е = 2,28 \cdot 0,84 \cdot 6 \cdot 0,7 = 8,04 \text{ грн.}$$

Проведені розрахунки занесемо до таблиці 4.13.

Таблиця 4.13 – Витрати на електроенергію

Найменування операції	Найменування обладнання	Встановлена потужність, кВт	Тривалість операції, год	Кількість днів	Сума, грн
Консолідація даних	Комп'ютер	0,84	6	30	241,2
Програмування	Комп'ютер	0,84	6	30	241,2
			Всього		482,4

Інші витрати

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуються як 50…100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_в = (З_о + З_р) \cdot \frac{Н_{зп}}{100\%}, \qquad (4.12)$$

де $Н_{ів}$ – норма нарахування за статтею «Інші витрати».

Інші витрати складатимуть:

$I_в$ = (60840+33270) · 0,5 = 47 055 грн.

Витрати на проведення науково-дослідної роботи розраховуються яксума всіх попередніх статей витрат за формулою:

$$В_{заг} = З_О + З_р + З_{дод} + З_Н + К_в + В_{спец} + В_{прг} + А_{обл} + В_е + І_в . \qquad (4.13)$$

$В_{заг}$= 60840+33270+9411+22774,62+926,2+45760+14300+3000+482,4+47055= 237819,22

Загальні витрати *ЗВ* на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховуються за формулою:

$$ЗВ = \frac{В_{заг}}{\mu}, \qquad (4.14)$$

де η – коефіцієнт, який характеризує етап (стадію) виконання науково- дослідної роботи. Так, якщо науково-технічна розробка знаходиться на стадії: науково-дослідних робіт, то μ =0,1; технічного проектування, то μ =0,2; розробки конструкторської документації, то μ =0,3; розробки технологій, то μ=0,4; розробки дослідного зразка, то μ =0,5; розробки промислового зразка, то μ=0,7; впровадження, то μ =0,9.

ЗВ= 237819,22/0,9= 264243,58 грн.

1.1 4.4 Розрахунок ефективності вкладених інвестицій та період їх окупності

У даному підрозділі кількісно спрогнозуємо, яку вигоду можна отримати у майбутньому від впровадження результатів виконаної наукової роботи. Розрахуємо збільшення чистого прибутку підприємства $\Delta П_i$, для кожного із років, протягом яких очікується отримання позитивних результатів від впровадження розробки, за формулою

$$\Delta П_i = \sum_1^n (\Delta Ц_0 \cdot N + Ц_0 \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot (1 - \frac{\upsilon}{100}), \quad (4.15)$$

де $\Delta Ц_0$ – покращення основного оціночного показника від впровадження результатів розробки у даному році;

N – основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

ΔN – покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки;

$Ц_o$ – основний оціночний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

n – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки:

λ – коефіцієнт, який враховує сплату податку на додану вартість.

Ставка податку на додану вартість дорівнює 20%, а коефіцієнт λ = 0,8333;

p – коефіцієнт, який враховує рентабельність продукту, *p* = 0,2;

υ – ставка податку на прибуток. У 2021 році — 18%.

Припустимо, що при прогнозованій ціні 10000 грн. за прилад для контролю доступу з використанням face-ідентифікації користувача та віддаленого керування, термін збільшення прибутку складе 3 роки. Після завершення розробки і її вдосконалення, можна буде підняти його ціну на 500 грн. Кількість наданих послуг також збільшиться: протягом першого року — на 200 шт., протягом другого року — на 100 шт., протягом третього року на 50 шт. До моменту впровадження результатів наукової розробки реалізації продукту не було:

$\Delta П_1$ = (0*500+ (10000 + 500)*200* 0,8333* 0,2) * (1 - 0,18) = 286 988,52 грн.

$\Delta П_2$ = (0*500+ (10000 + 500)*(200+100)* 0,8333* 0,2) * (1 - 0,18) = 430 482,78 грн.

$\Delta П_3$ = (0*500+ (10000 + 500)*(200+100+50)* 0,8333* 0,2) * (1 - 0,18) = 502 229,91 грн.

Отже, комерційний ефект від реалізації результатів розробки за три роки складе 1 219 701,21 грн.

Далі розраховують приведену вартість збільшення всіх чистих прибутків *ПП*, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_{1}^{T} \frac{\Delta П_i}{(1+\tau)^t}, \qquad (4.16)$$

де $\Delta П_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої МКР, грн;

Т – період часу, протягом якого виявляються результати впровадженої МКР, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні; для України цей показник знаходиться на рівні 0,15;

t – період часу (в роках).

$$ПП = \frac{286\,988{,}52}{(1+0{,}15)^1} + \frac{430\,482{,}78}{(1+0{,}15)^2} + \frac{502\,229{,}91}{(1+0{,}17)^3} = 282747{,}3 + 325506{,}8 + 330224{,}3 = \\ = 938\,478{,}4 \text{ грн.}$$

Далі розраховують величину початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково- технічної розробки. Для цього можна використати формулу:

$$PV = k_{\text{інв}} \cdot \text{ЗВ} \quad (4.17)$$

де $k_{інв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково- технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо; зазвичай $k_{інв}$ =2...5, але може бути і більшим;

ЗВ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн.

$$PV = 2 \cdot 264243{,}58 = 528487{,}16 \text{ грн}$$

Тоді абсолютний економічний ефект $E_{абс}$ або чистий приведений дохід(NPV, Net Present Value) для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = ПП - PV, \quad (4.18)$$

де ПП – приведена вартість всіх чистих прибутків, що їх отримає підприємство (організація) від реалізації результатів наукової розробки, грн.

$E_{абс}$= 938478,4 - 528487,16= 409991,24 грн.

Оскільки $E_{абс} > 0$, то вкладання коштів на виконання та впровадження результатів роботи може бути доцільним.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій $E_в$ за формулою:

$$E_в = \sqrt[T_ж]{1 + \frac{E_{абс}}{PV}} - 1, \quad (4.19)$$

де $E_{абс}$ – абсолютна ефективність вкладених інвестицій, грн.;

PV – теперішня вартість інвестицій PV = ЗВ, грн.;

$T_ж$ – життєвий цикл наукової розробки, роки.

$$E_в = \sqrt[3]{1 + \frac{409991,24}{528487,16}} - 1 = 1,21 - 1 = 0,21$$

Далі слід порівняти розраховану величину $E_в$ з мінімальною (бар'єрною) ставкою дисконтування τ_{min}, яка визначає ту мінімальну дохідність, нижче за яку інвестиції вкладатися не будуть. У загальному вигляді мінімальна (бар'єрна) ставка дисконтування визначається за формулою:

$$\tau_{min} = d + f, \quad (4.20)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2020 році в Україні d = (0,9...0,12);

f – показник, що характеризує ризикованість вкладень; зазвичай, величина f= (0,05...0,5), але може бути і значно більше.

$$\tau_{min} = 0,12 + 0,05 = 0,17.$$

Оскільки $E_в > \tau_{min}$, то інвестор буде зацікавлений у фінансуванні приладу для контролю доступу з використанням face-ідентифікації користувача та віддаленого керування.

Термін окупності вкладених у реалізацію наукового проекту інвестицій $T_{ок}$ можна розрахувати за формулою:

$$T_{ок} = \frac{1}{E_в}, \quad (4.21)$$

Для приладу контролю доступу з використанням face-ідентифікації користувача та віддаленого керування термін окупності складе:

$$T_{ок} = 1/0,21 = 4,76 \text{ р.}$$

Оскільки $T_{ок}$ > 3-х років, можна зробити висновок, що фінансування даної наукової розробки не буде доцільним.

1.2 4.5 Висновки до розділу

У даному розділі було проаналізовано економічну доцільність розробки приладу для контролю доступу з використанням face-ідентифікації користувача та віддаленого керування. Опитування експертів показало, що запропонована розробка має високий рівень комерційного потенціалу. Вкладання коштів на виконання та впровадження результатів МКР є доцільним. Дана розробка має високий рівень щорічної ефективності та повністю окупиться по закінченню першого року після впровадження.

ВИСНОВКИ

В даний час, розробка і впровадження систем контролю та управління доступом є однією з найактуальніших і важливих завдань в сфері забезпечення охоронних заходів на об'єктах.

У ході виконання роботи було здійснено аналіз можливих варіантів несанкціонованого проникнення до приміщення, також було досліджено вразливі місця в охоронній системі, власне підміна RFID карток користувачів. Створено модель контролю доступу, наведено критерії її оцінки та методи перевірки. Запропоновано способи та методи захисту приміщення від витоку інформації по іншим каналам, яких не охоплює дана система, а також критерії оцінки об’єкту захисту.

Отже використання данної системи доцільне для установ та підприємств, де потрібно обмежити доступ та проводити аудит переміщення персоналу на об’єкті. Наступними кроками є: подальша модернізація системи для покращення її технічних характеристик, а також розширення функціональності продукту для різних потреб споживача.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Царьов Р.Ю. Біометричні технології: навч. посіб. для вищих навчальних закладів / Р.Ю. Царьов, Т. М. Лемеха. – Одеса: ОНАЗ ім. О.С. Попова, 2016. – 140 с.: іл.

2. Ворона В.А., Тихонов В.А. Системы контроля и управления доступом. М.: Горячая линия Телеком, 2015. 272 с

3. Сабынин В.Н. Организация пропускного режима первый шаг к обеспечению безопасности и конфиденциальности информации // Информост - радиоэлектроники и телекоммуникации, 2001. № 3

4. Тихонов О.О., Малышева А.С., Шаповалов А.В., Гамбург А.Е., Стасенко Л.А, Курилин А.С. Функции универсальных СКУД: что нужно потребителю. //Системы безопасности 2011 № 4. С. 108-119.

5. Сорокин К. Применение биометрических технологий в обеспечении информационной безопасности бизнеса. //СКУД. Антитеррористм-2013 2013 С. 46-47.

6. Портал «Scanberry». - «Принцип работы RFID и ее применения» / Электронные текстовые данные, рис., табл. URL: https://scanberry.ru/news/perenosnye-rfid-schityvateli-i-skanery/

7. Reynolds M. «Microwave RFID: Passive Scattering and Active Transponders», MIT, 2002.

8. Портал «RTLService» - «Принцип работы RFID и ее применение» / Электронные текстовые данные, рис., табл. URL:https://rtlservice.com/ru/company/blog/princip_raboty_ tehnologii_rfid_i_ee_primenenie/свободный.

9. Быков О. Что надо знать, заказывая карты Mifare для СКД / О. Быков // Алгоритм безопасности. – 2011. – № 4. – С. 40–41.

10. Корнеев С.В., Рунге А.В. «К вопросу об управлении эффективной поверхностью рассеяния диполей в технологии радиочастотной

идентификации». В кн.: Антенны / Под ред. Бахраха Л.Д. Вып. 6. – М.: Радио и связь, 2002.

11. Сучасні системи безпеки бізнесу [Електронний ресурс]. – Режим доступу: https://ssbb.com.ua/uk/sistemy-kontrolya-dostupa/sistema-kontrolyudostupu/sistema-kontrolya-i-upravleniya-dostupom/

12. Кухарев Г. А. Биометрические системы: Методы и средства идентификации личности человека. – СПб.: Политехника, 2001. – 240 с.

13. RFID Radio Frequency Identification Technology Tutorial [Электронный ресурс] – Режим доступа: http://www.radio-electronics.com/

14. Разработка системы контроля и управления доступом к охраняемым объектам [Электронный ресурс] – Режим доступа: http://studbooks.net/...

15. Валерий Дшхунян, Владимир Шаньгин. Электронная идентификация. Бесконтактные электронные идентификаторы и смарт-карты – : АСТ, НТ Пресс, 2004

16. Зегжда Д.П. Основи безпеки інформаційних систем / Д.П. Зегжда, А.М. Івашко. - М.: Гаряча лінія - телеком, 2000. - 452 с., мул.

17. Комп'ютерна злочинність і інформаційна безпека / А.П. Леонов [і др.]; під общ. Ред. А.П. Леонова. - Мінськ: АРІЛ, 2000. - 552 с.

18. Финкенцеллер Клаус: RFID-технологии. Справочное пособие –: Додека XXI век, 2010

19. Тихонов В.А., Райх В.В. Информационная безопасность: концептуальные, правовые, организационные и технические аспекты: Уч. пособие. М.: Гелиос АРВ, 2006.

20. Быков О. Перевод СКУД с карт EM MARIN на карты MIFARE® : полезные сведения для владельца объекта СКД / О. Быков // Avtoritet.Net. — [Б. м.], 2014. Режим доступу: – URL: http://avtoritet.net/library/press/245/8158/articles/8745

21. Ярочкин В.И. Інформаційна безпека. Підручник для студентів вузів / 3_е изд. - М.: Академічний проект: Трікста, 2005. - 544 з.

22. RFID-технологии в производстве // Журнал «ИСУП», Горбунов А.О. - 2005. -№4. - стр. 8-9.

23. Карты и RFID технология, интернет-страница компании «Credit-card», [Електронний ресурс]. – Режим доступу: http://credit-card.ru/articles/technology/rfid-cards.php.

24. Информационная безопасность [Электронний ресурс]: новости, актуальные вопросы законодательства: электронный журнал. – Режим доступу: http://www.itsec.ru/

25. RFID технологии, интернет-страница компании «IQSKLAD» [Електронний ресурс]. – Режим доступу: http://www.iqsklad.ru/rfid.

26. Выбор и применение систем контроля и управления доступом [Электронний ресурс] – Режим доступу: http://allmedia.ru/...

27. М. Федоров, Технология RFID. Опыт использования и перспективные направления, «Компоненты и технологии» №9, 2005 г.

28. Шарфельд Т. «Системы RFID низкой стоимости» с Приложениями Девиля И., Дамура Ж., Чаркани Н., Корнеева С. и Гуларии А. Перевод с английского и научная редакция Корнеева С. Москва – 2006.

29. Из чего состоит система RFID, интернет-страница компании «ПОСМагазин» [Електронний ресурс]. – Режим доступу: http://posmagazin.ru/articles/18/120/

30. Барсуков В.С. Интегральная защита информации // Системы безопасности, 2002. №5, 6.

31. Горлицин И. Контроль и управление доступом просто и надежно КТЦ «Охранные системы», 2002.

32. Near Field Communication (NFC) Technology and Measurements [Электронный ресурс] – Режим доступа: https://cdn.rohde-schwarz.com/...

33. Система контроля и управления доступом. Принцип действия [Электронный ресурс]. – Режим доступа: http://www.intersyst.ru/solutions/165/46

34. А.В.Петраков. Защита и охрана личности, собственности, информации: Справное пособие. - М.: Радио и связь, 1997.

35. Т. Шарфельд (с Приложениями И. Девиля, Ж. Дамура, Н. Чаркани, С. Корнеева и А. Гуларии) Системы RFID низкой стоимости / С. Корнеев. — Москва: 2006.

36. Татарченко И.В., Соловьев Д.С. Концепция интеграции унифицированных систем безопасности // Системы безопасности. № 1 (73). С. 8689.

37. Системи контролю доступу. [Електронний ресурс]. – Режим доступу: https://leater.com/ua/services/sistema-kontrolyu-dostupu.html

38. RFID технология. [Електронний ресурс]. – Режим доступу: https://www.cleverence.ru/articles/rfid/rfid-tekhnologiya-chto-eto-takoe-kak-rabotaet-sistema-opisanie-i-primenenie/

39. Не типичные функции СКУД. [Електронний ресурс]. – Режим доступу: http://www.secuteck.ru/articles2/sys_ogr_dost/netipichnye-funktsii-skud/

40. Обзор возможностей СКУД [Электронный ресурс]. – Режим доступу: http://www.sistema-dostupa.ru/i03.htm/

41. Кмпоненты RFID-метки. [Електронний ресурс]. – Режим доступу: https://www.rst-invent.ru/about/technology/

42. Robotshop. [Електронний ресурс]. – Режим доступу: https://www.robotshop.com/media/files/pdf/arduinomega2560datasheet.pdf. – Arduino Mega 2560 Datasheet.

43. Microchip. [Електронний ресурс]. – Режим доступу: https://ww1.microchip.com/downloads/en/devicedoc/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf. – Atmel ATmega640/V-1280/V-1281/V-2560/V-2561/V Datasheet.

44. Filipeflop. [Електронний ресурс]. – Режим доступу: https://img.filipeflop.com/files/download/Datasheet_SIM800L.pdf. – SIM800L Hardware Design V1.00 Datasheet.

45. Nxp. [Електронний ресурс]. – Режим доступу: https://www.nxp.com/docs/en/data-sheet/MFRC522.pdf. – MFRC522 Datasheet.

46. Mpja. [Електронний ресурс]. – Режим доступу: https://www.mpja.com/download/31227sc.pdf. – HC-SR501 PIR MOTION DETECTOR Datasheet.

47. Robotchip. [Електронний ресурс]. – Режим доступу: https://robotchip.ru/obzor-modulya-rele-2-x-kanalny/. – 2-х канальне реле.

48. Haoyuelectronics. [Електронний ресурс]. – Режим доступу: http://www.haoyuelectronics.com/Attachment/MP1584/MP1584.pdf. – MP1584 Datasheet.

49. dlnmh9ip6v2uc. [Електронний ресурс]. – Режим доступу: https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Prototyping/TP4056.pdf. – TP4056 1A Standalone Linear Li-lon Battery Charger with Thermal Datasheet.

50. Onsemi. [Електронний ресурс]. – Режим доступу: https://www.onsemi.com/pub/Collateral/P2N2222A-D.PDF. – Amplifier Transistors P2N2222A Datasheet

ДОДАТКИ

Додаток А

Вінницький національний технічний університет

Факультет менеджменту та інформаційної безпеки

Кафедра менеджменту та безпеки інформаційних систем

ЗАТВЕРДЖУЮ

Голова секції “Управління інформаційною
безпекою” кафедри МБІС
д.т.н., професор

_____Ю.Є.Яремчук
“____”_____2021 р.

ТЕХНІЧНЕ ЗАВДАННЯ

до магістерської кваліфікаційної роботи на тему:

Підвищення захищеності системи контролю доступу до приміщення з використанням face - ідентифікації користувача та віддаленого керування

08-42.МКР.005.00.000.ТЗ

Керівник магістерської кваліфікаційної роботи

к.т.н., доц., доцент каф. МБІС

_Яремчук Ю.Є._____.

Вінниця – 2021 р.

1. Найменування та область застосування

Підвищення захищеності системи контролю доступу до приміщення з використанням face - ідентифікації користувача та віддаленого керування. Область застосування: контроль пересування працівників і сторонніх осіб до приміщення в різних сферах діяльності.

2.Підстава для розробки

Розробка виконується на основі наказу ректора ВНТУ № від 2021р.

3. Мета та призначення розробки

3.1 Мета розробки: теоретичний і практичний розгляд способів і засобів контролю доступу з використанням біометричних даних, а також підвищеня захищеності системи контролю доступу з використанням face-ідентифікації.

3.2 Призначення: контроль пересування працівників і сторонніх осіб до приміщення і повідомлення власника про сторонніх осіб які намагаються потрапити до приміщення.

4. Джерела розробки

4.1. Царьов Р.Ю. Біометричні технології: навч. посіб. для вищих навчальних закладів / Р.Ю. Царьов, Т. М. Лемеха. – Одеса: ОНАЗ ім. О.С. Попова, 2016. – 140 с.: іл.

4.2. Ворона В.А., Тихонов В.А. Системы контроля и управления доступом. М.: Горячая линия Телеком, 2015. 272 с.

4.3. Espressif. [Електронний ресурс]. – Режим доступу: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf – ESP32 Series Datasheet.

4.4. Robotchip. [Електронний ресурс]. – Режим доступу: https://robotchip.ru/obzor-modulya-rele-2-x-kanalny/. – 2-х канальне реле.

5. Вимоги до приладу

5.1 Вимоги до функціональних характеристик:

5.1.1 Прилад повинен мати зручний, легкий у використанні web-інтерфейс для користувача;

5.1.2 Прилад повинен мати можливість інтегруватися в вже встановлені системи контролю доступу;

5.1.3 Прилад повинен виконувати процес розпізнання користувачів у системі контролю доступом за допомогою face-ідентифікації.

5.2 Вимоги до надійності:

5.2.1 Прилад повинен працювати без збоїв, у випадку виникнення критичних ситуацій, необхідно передбачити виведення відповідних повідомлень;

5.2.2 Бази даних фото користувачів повинні знаходитися на захищеному від витоку інформації сервісі;

5.2.3 Прилад повинен виконувати свої функції.

5.3 Вимоги до складу і параметрів технічних засобів:

– мікроконтролер – ESP32;

– відеокамера – OV2640 або OV7670;

– середовище функціонування – операційна система сімейство Windows;

– вимоги до техніки безпеки при роботі з приладом повинні відповідати існуючим вимогам та стандартам з техніки безпеки при користуванні електроприладами.

6. Вимоги до документації

6.1 Інструкція з використання та налаштування для користувачів, наведена у пункті 3.3.

7. Вимоги до технічного захисту

7.1 Неможливість отримання доступу до приладу користувачам без відповідних прав для цього.

8. Техніко-економічні показники

8.1 Дохід від реалізації даного проекту повинен перевищувати витрати.

8.2 Пристрій має бути конкурентноспроможним на ринку.

8.3 Можливість комерціалізації пристрою потенційним інвесторам.

9. Стадії та етапи розробки

№ з/п	Назва етапів магістерської кваліфікаційної роботи	Початок	Закінчення
1	Визначення напрямку магістерської роботи, формулювання теми		
2	Аналіз предметної області обраної теми		
3	Розробка алгоритму роботи		
4	Написання магістерської роботи на основі розробленої теми		
5	Розробка економічної частини		
6	Передзахист магістерської кваліфікаційної роботи		
7	Виправлення, уточнення, корегування магістерської кваліфікаційної роботи		
8	Захист магістерської кваліфікаційної роботи		

10. Порядок контролю та прийому

10.1 До приймання магістерської кваліфікаційної роботи надається:

– ПЗ до магістерської кваліфікаційної роботи;

– прилад;

– презентація;

– відзив керівника роботи;

– відзив рецензента

Технічне завдання до виконання прийняв _____Кравчик В.В.

Додаток Б

Перелік використаних елементів

Поз.	Найменування	Кіл.	Примітка
	Мікроконтролери		
M1	ESP32	1	
	Релейні модулі		
REL1	SRD-05VDC-SL-C	1	
	DC-DC модулі		
U1	DC-DC MP2307	1	
	Кнопки		
S1	Тактова кнопка	1	
	П’єзоелементи		
LS1	КРХ-1205В	1	
	Програматори		
U2	EEPROM 24xx і 25xx на CH341A	1	
	Відеокамери		
CAM1	OV2640	1	

					08-42.МКР.005.00.000 ПЕ	
Змн.	Лист	№ докум.	Підпис	Дата		
Розроб.		Кравчик В.В.			Система контролю доступу до приміщення з використанням face-ідентифікації Перелік елементів	Літ. / Арк. 85 / Аркушів 74
Перевір.		Яремчук Ю.Є				
Реценз.						ВНТУ, гр. УБ-16б
Н. Контр.						
Затверд.						

Додаток В

Лістинг програми

```
//Введіть пароль облікового запису підключення WIFI
const char* ssid = "YourSSID";
const char* password = "YourPassword";

//Введіть пароль облікового запису підключення до точки доступу
const char* apssid = " YourAPSSID ";
const char* appassword = " YourAPPassword "; //Пароль точки доступу має містити щонайменше 8 символів

String myToken = "YourBotToken";
String myChatId = "YourChatID";

int pinDoor = 2;         //Контакт реле замка дверей IO2
long message_id_last = 0; //Початкове значення коду повідомлення Telegram
int timer = 0;           //Початкове значення очікування на команду повідомлення Telegram
int timerLimit = 10;     //Telegram чекає команди повідомлення, відеоекран призупиниться, доки час очікування не досягне 10 секунд

#include <WiFi.h>
#include <WiFiClientSecure.h>
#include "esp_camera.h"        //Відео
#include "soc/soc.h"           //Використовується для нестабільного живлення без перезавантаження
#include "soc/rtc_cntl_reg.h"   //
#include <ArduinoJson.h>       //Розбір функції формату json

String Feedback = ""; //Надіслати повідомлення клієнту
//Значення параметра інструкції
String Command = "", cmd = "", P1 = "", P2 = "", P3 = "", P4 = "", P5 = "", P6 = "", P7 = "", P8 = "", P9 = "";
//Значення стану розбирання команди
byte ReceiveState = 0, cmdState = 1, strState = 1, questionstate = 0, equalstate = 0, semicolonstate = 0;

//Налаштування контактів модуля Ancred ESP32-CAM
#define PWDN_GPIO_NUM     32
#define RESET_GPIO_NUM    -1
#define XCLK_GPIO_NUM      0
#define SIOD_GPIO_NUM     26
#define SIOC_GPIO_NUM     27

#define Y9_GPIO_NUM       35
#define Y8_GPIO_NUM       34
#define Y7_GPIO_NUM       39
#define Y6_GPIO_NUM       36
```

```
#define Y5_GPIO_NUM      21
#define Y4_GPIO_NUM      19
#define Y3_GPIO_NUM      18
#define Y2_GPIO_NUM       5
#define VSYNC_GPIO_NUM    25
#define HREF_GPIO_NUM     23
#define PCLK_GPIO_NUM     22

WiFiServer server(80);
WiFiClient client;

void setup() {
  WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0);  //Вимкніть живлення, коли живлення нестабільне, і перезапустіть налаштування

  Serial.begin(115200);
  Serial.setDebugOutput(true);  //Увімкніть діагностичний вихід
  Serial.println();

  //Параметри конфігурації відео
  camera_config_t config;
  config.ledc_channel = LEDC_CHANNEL_0;
  config.ledc_timer = LEDC_TIMER_0;
  config.pin_d0 = Y2_GPIO_NUM;
  config.pin_d1 = Y3_GPIO_NUM;
  config.pin_d2 = Y4_GPIO_NUM;
  config.pin_d3 = Y5_GPIO_NUM;
  config.pin_d4 = Y6_GPIO_NUM;
  config.pin_d5 = Y7_GPIO_NUM;
  config.pin_d6 = Y8_GPIO_NUM;
  config.pin_d7 = Y9_GPIO_NUM;
  config.pin_xclk = XCLK_GPIO_NUM;
  config.pin_pclk = PCLK_GPIO_NUM;
  config.pin_vsync = VSYNC_GPIO_NUM;
  config.pin_href = HREF_GPIO_NUM;
  config.pin_sscb_sda = SIOD_GPIO_NUM;
  config.pin_sscb_scl = SIOC_GPIO_NUM;
  config.pin_pwdn = PWDN_GPIO_NUM;
  config.pin_reset = RESET_GPIO_NUM;
  config.xclk_freq_hz = 20000000;
  config.pixel_format = PIXFORMAT_JPEG;
  //init with high specs to pre-allocate larger buffers
  if (psramFound()) {
    config.frame_size = FRAMESIZE_UXGA;
    config.jpeg_quality = 10;  //0-63 lower number means higher quality
    config.fb_count = 2;
  } else {
    config.frame_size = FRAMESIZE_SVGA;
    config.jpeg_quality = 12;  //0-63 lower number means higher quality
    config.fb_count = 1;
```

```
  }

  // camera init
  esp_err_t err = esp_camera_init(&config);
  if (err != ESP_OK) {
    Serial.printf("Camera init failed with error 0x%x", err);
    delay(1000);
    ESP.restart();
  }

  //drop down frame size for higher initial frame rate
  sensor_t * s = esp_camera_sensor_get();
  s->set_framesize(s, FRAMESIZE_QVGA);
//UXGA|SXGA|XGA|SVGA|VGA|CIF|QVGA|HQVGA|QQVGA  Встановіть початкову роздільну здатність зображення

  //світлодіод
  ledcAttachPin(4, 4);
  ledcSetup(4, 5000, 8);

  WiFi.mode(WIFI_AP_STA);  //Інші режими WiFi.mode(WIFI_AP); WiFi.mode(WIFI_STA);

  //Вкажіть статичний IP клієнта

  for (int i = 0; i < 2; i++) {
    WiFi.begin(ssid, password);    //Виконайте підключення до мережі

    delay(1000);
    Serial.println("");
    Serial.print("Connecting to ");
    Serial.println(ssid);

    long int StartTime = millis();
    while (WiFi.status() != WL_CONNECTED) {
      delay(500);
      if ((StartTime + 5000) < millis()) break;  //Зачекайте 10 секунд, щоб підключитися
    }

    if (WiFi.status() == WL_CONNECTED) {    //Якщо підключення вдалось
      WiFi.softAP((WiFi.localIP().toString() + "_" + (String)apssid).c_str(), appassword); //Встановіть SSID для відображення IP-адреси клієнта
      Serial.println("");
      Serial.println("STAIP address: ");
      Serial.println(WiFi.localIP());
      Serial.println("");

      for (int i = 0; i < 5; i++) { //Якщо підключено до WIFI, налаштуйте світлодіод на швидке моргання
        ledcWrite(4, 10);
        delay(200);
```

```
      ledcWrite(4, 0);
      delay(200);
     }
     break;
   }
  }

  if (WiFi.status() != WL_CONNECTED) {    //Якщо з’єднання не вдається
   WiFi.softAP((WiFi.softAPIP().toString() + "_" + (String)apssid).c_str(), appassword);

   for (int i = 0; i < 2; i++) { //Якщо не вдається підключитися до WIFI, налаштуйте світлодіод на повільне моргання
     ledcWrite(4, 10);
     delay(1000);
     ledcWrite(4, 0);
     delay(1000);
   }
  }

  //Вкажіть кінцеву IP-адресу точки доступу
  Serial.println("");
  Serial.println("APIP address: ");
  Serial.println(WiFi.softAPIP());
  Serial.println("");

  pinMode(4, OUTPUT);
  digitalWrite(4, LOW);

  server.begin();
}

void loop() {
  Feedback = ""; Command = ""; cmd = ""; P1 = ""; P2 = ""; P3 = ""; P4 = ""; P5 = ""; P6 = ""; P7 = ""; P8 = ""; P9 = "";
  ReceiveState = 0, cmdState = 1, strState = 1, questionstate = 0, equalstate = 0, semicolonstate = 0;

  client = server.available();

  if (client) {
   String currentLine = "";

   while (client.connected()) {
    if (client.available()) {
     char c = client.read();

     getCommand(c);   //Розберіть символи, отримані в буфері, до параметрів команди

     if (c == '\n') {
      if (currentLine.length() == 0) {
```

```
          if (cmd == "getstill") { //Зробіть скріншот відео
            getStill();
          }
          else if (cmd == "status") { //Отримати статус відео
            status();
          }
          else {  //Отримати домашню сторінку управління
            mainpage();
          }

          Feedback = "";
          break;
        } else {
          currentLine = "";
        }
      }
      else if (c != '\r') {
        currentLine += c;
      }

      if ((currentLine.indexOf("/?") != -1) && (currentLine.indexOf(" HTTP") != -1)) {
        if (Command.indexOf("stop") != -1) { //Якщо команда містить ключове слово stop, рядок
буде негайно відключено
          client.println();
          client.println();
          client.stop();
        }
        currentLine = "";
        Feedback = "";
        ExecuteCommand();
      }
    }
   }
   delay(1);
   client.stop();
  }
}

void ExecuteCommand() {
  //Serial.println("");
  //Serial.println("Command: "+Command);
  if (cmd != "getstill") {
    Serial.println("cmd= " + cmd + " ,P1= " + P1 + " ,P2= " + P2 + " ,P3= " + P3 + " ,P4= " + P4 + " ,P5=
" + P5 + " ,P6= " + P6 + " ,P7= " + P7 + " ,P8= " + P8 + " ,P9= " + P9);
    Serial.println("");
  }

  //Спеціальний командний блок
  if (cmd == "your cmd") {
```

```
} else if (cmd == "ip") { //Запит APIP, STAIP
  Feedback = "AP IP: " + WiFi.softAPIP().toString();
  Feedback += "<br>";
  Feedback += "STA IP: " + WiFi.localIP().toString();
} else if (cmd == "mac") { //Запит MAC-адреси
  Feedback = "STA MAC: " + WiFi.macAddress();
} else if (cmd == "restart") { //Скидання з'єднання WIFI
  ESP.restart();
} else if (cmd == "digitalwrite") { //Цифровий вихід
  ledcDetachPin(P1.toInt());
  pinMode(P1.toInt(), OUTPUT);
  digitalWrite(P1.toInt(), P2.toInt());
} else if (cmd == "digitalread") { //Цифровий вхід
  Feedback = String(digitalRead(P1.toInt()));
} else if (cmd == "analogwrite") { //Аналоговий вихід
  if (P1 == "4") {
    ledcAttachPin(4, 4);
    ledcSetup(4, 5000, 8);
    ledcWrite(4, P2.toInt());
  } else {
    ledcAttachPin(P1.toInt(), 9);
    ledcSetup(9, 5000, 8);
    ledcWrite(9, P2.toInt());
  }
}
else if (cmd == "analogread") { //Аналогове читання
  Feedback = String(analogRead(P1.toInt()));
} else if (cmd == "touchread") { //Торкніться, щоб прочитати
  Feedback = String(touchRead(P1.toInt()));
} else if (cmd == "restart") { //перезавантаження
  ESP.restart();
} else if (cmd == "flash") { //світлодіод
  ledcAttachPin(4, 4);
  ledcSetup(4, 5000, 8);
  int val = P1.toInt();
  ledcWrite(4, val);
} else if (cmd == "servo") { //Серводвигун
  ledcAttachPin(P1.toInt(), 3);
  ledcSetup(3, 50, 16);

  int val = 7864 - P2.toInt() * 34.59;
  if (val > 7864)
    val = 7864;
  else if (val < 1638)
    val = 1638;
  ledcWrite(3, val);
} else if (cmd == "relay") { //Реле
  pinMode(P1.toInt(), OUTPUT);
  digitalWrite(P1.toInt(), P2.toInt());
} else if (cmd == "uart") { //UART
```

```
   Serial.print(P1);

   //Telegram bot
   if (P1 == "unknown") { //незнайомець
    sendCapturedImage2Telegram(myToken, myChatId);
    String keyboard = "{\"keyboard\":[[{\"text\":\"/open\"},{\"text\":\"/still\"}],
[{\"text\":\"/ledon\"},{\"text\":\"/ledoff\"}]],\"one_time_keyboard\":false}";
    sendMessage2Telegram(myToken, myChatId, "Stranger", keyboard);
    getTelegramMessage(myToken);
   } else {  //Власник
    sendMessage2Telegram(myToken, myChatId, "Welcome back! " + P1, "");
    telegramCommand("/open");
   }
  } else if (cmd == "resetwifi") { //Скидання мережевого підключення
   for (int i = 0; i < 2; i++) {
    WiFi.begin(P1.c_str(), P2.c_str());
    Serial.print("Connecting to ");
    Serial.println(P1);
    long int StartTime = millis();
    while (WiFi.status() != WL_CONNECTED) {
     delay(500);
     if ((StartTime + 5000) < millis()) break;
    }
    Serial.println("");
    Serial.println("STAIP: " + WiFi.localIP().toString());
    Feedback = "STAIP: " + WiFi.localIP().toString();

    if (WiFi.status() == WL_CONNECTED) {
     WiFi.softAP((WiFi.localIP().toString() + "_" + P1).c_str(), P2.c_str());
     for (int i = 0; i < 2; i++) { //Якщо не вдається підключитися до WIFI, налаштуйте світлодіод
на повільне моргання
       ledcWrite(4, 10);
       delay(300);
       ledcWrite(4, 0);
       delay(300);
      }
      break;
    }
   }
  } else if (cmd == "framesize") {
   int val = P1.toInt();
   sensor_t * s = esp_camera_sensor_get();
   s->set_framesize(s, (framesize_t)val);
  } else if (cmd == "quality") { //Якість зображення
   sensor_t * s = esp_camera_sensor_get();
   s->set_quality(s, P1.toInt());
  } else if (cmd == "contrast") { //контраст
   sensor_t * s = esp_camera_sensor_get();
   s->set_contrast(s, P1.toInt());
  } else if (cmd == "brightness") { //яскравість
```

```
   sensor_t * s = esp_camera_sensor_get();
   s->set_brightness(s, P1.toInt());
  } else if (cmd == "saturation") { //насичення
   sensor_t * s = esp_camera_sensor_get();
   s->set_saturation(s, P1.toInt());
  } else if (cmd == "special_effect") { //Спецефекти
   sensor_t * s = esp_camera_sensor_get();
   s->set_special_effect(s, P1.toInt());
  } else if (cmd == "hmirror") { //Горизонтальне дзеркальне відображення
   sensor_t * s = esp_camera_sensor_get();
   s->set_hmirror(s, P1.toInt());
  } else if (cmd == "vflip") { //Поворот вертикально
   sensor_t * s = esp_camera_sensor_get();
   s->set_vflip(s, P1.toInt());
  } else {
   Feedback = "Command is not defined.";
  }
  if (Feedback == "") Feedback = Command;
}

//Розібрати командний рядок на змінну
void getCommand(char c)
{
  if (c == '?') ReceiveState = 1;
  if ((c == ' ') || (c == '\r') || (c == '\n')) ReceiveState = 0;

  if (ReceiveState == 1)
  {
   Command = Command + String(c);

   if (c == '=') cmdState = 0;
   if (c == ';') strState++;

   if ((cmdState == 1) && ((c != '?') || (questionstate == 1))) cmd = cmd + String(c);
   if ((cmdState == 0) && (strState == 1) && ((c != '=') || (equalstate == 1))) P1 = P1 + String(c);
   if ((cmdState == 0) && (strState == 2) && (c != ';')) P2 = P2 + String(c);
   if ((cmdState == 0) && (strState == 3) && (c != ';')) P3 = P3 + String(c);
   if ((cmdState == 0) && (strState == 4) && (c != ';')) P4 = P4 + String(c);
   if ((cmdState == 0) && (strState == 5) && (c != ';')) P5 = P5 + String(c);
   if ((cmdState == 0) && (strState == 6) && (c != ';')) P6 = P6 + String(c);
   if ((cmdState == 0) && (strState == 7) && (c != ';')) P7 = P7 + String(c);
   if ((cmdState == 0) && (strState == 8) && (c != ';')) P8 = P8 + String(c);
   if ((cmdState == 0) && (strState >= 9) && ((c != ';') || (semicolonstate == 1))) P9 = P9 + String(c);

   if (c == '?') questionstate = 1;
   if (c == '=') equalstate = 1;
   if ((strState >= 9) && (c == ';')) semicolonstate = 1;
  }
}
```

```
//Індивідуальний інтерфейс керування домашньою сторінкою
static const char PROGMEM INDEX_HTML[] = R"rawliteral(<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width,initial-scale=1">
    <title>ESP32 OV2460</title>
    <style>
      body {
        font-family: Arial,Helvetica,sans-serif;
        background: #181818;
        color: #EFEFEF;
        font-size: 16px
      }
      h2 {
        font-size: 18px
      }
      section.main {
        display: flex
      }
      #menu,section.main {
        flex-direction: column
      }
      #menu {
        display: none;
        flex-wrap: nowrap;
        min-width: 340px;
        background: #363636;
        padding: 8px;
        border-radius: 4px;
        margin-top: -10px;
        margin-right: 10px;
      }
      #content {
        display: flex;
        flex-wrap: wrap;
        align-items: stretch
      }
      figure {
        padding: 0px;
        margin: 0;
        -webkit-margin-before: 0;
        margin-block-start: 0;
        -webkit-margin-after: 0;
        margin-block-end: 0;
        -webkit-margin-start: 0;
        margin-inline-start: 0;
        -webkit-margin-end: 0;
        margin-inline-end: 0
      }
```

```
figure img {
   display: block;
   width: 100%;
   height: auto;
   border-radius: 4px;
   margin-top: 8px;
}
@media (min-width: 800px) and (orientation:landscape) {
   #content {
      display:flex;
      flex-wrap: nowrap;
      align-items: stretch
   }
   figure img {
      display: block;
      max-width: 100%;
      max-height: calc(100vh - 40px);
      width: auto;
      height: auto
   }
   figure {
      padding: 0 0 0 0px;
      margin: 0;
      -webkit-margin-before: 0;
      margin-block-start: 0;
      -webkit-margin-after: 0;
      margin-block-end: 0;
      -webkit-margin-start: 0;
      margin-inline-start: 0;
      -webkit-margin-end: 0;
      margin-inline-end: 0
   }
}
section#buttons {
   display: flex;
   flex-wrap: nowrap;
   justify-content: space-between
}
#nav-toggle {
   cursor: pointer;
   display: block
}
#nav-toggle-cb {
   outline: 0;
   opacity: 0;
   width: 0;
   height: 0
}
#nav-toggle-cb:checked+#menu {
   display: flex
```

```
}
.input-group {
   display: flex;
   flex-wrap: nowrap;
   line-height: 22px;
   margin: 5px 0
}
.input-group>label {
   display: inline-block;
   padding-right: 10px;
   min-width: 47%
}
.input-group input,.input-group select {
   flex-grow: 1
}
.range-max,.range-min {
   display: inline-block;
   padding: 0 5px
}
button {
   display: block;
   margin: 5px;
   padding: 0 12px;
   border: 0;
   line-height: 28px;
   cursor: pointer;
   color: #fff;
   background: #ff3034;
   border-radius: 5px;
   font-size: 16px;
   outline: 0
}
button:hover {
   background: #ff494d
}
button:active {
   background: #f21c21
}
button.disabled {
   cursor: default;
   background: #a0a0a0
}
input[type=range] {
   -webkit-appearance: none;
   width: 100%;
   height: 22px;
   background: #363636;
   cursor: pointer;
   margin: 0
}
```

```
input[type=range]:focus {
   outline: 0
}
input[type=range]::-webkit-slider-runnable-track {
   width: 100%;
   height: 2px;
   cursor: pointer;
   background: #EFEFEF;
   border-radius: 0;
   border: 0 solid #EFEFEF
}
input[type=range]::-webkit-slider-thumb {
   border: 1px solid rgba(0,0,30,0);
   height: 22px;
   width: 22px;
   border-radius: 50px;
   background: #ff3034;
   cursor: pointer;
   -webkit-appearance: none;
   margin-top: -11.5px
}
input[type=range]:focus::-webkit-slider-runnable-track {
   background: #EFEFEF
}
input[type=range]::-moz-range-track {
   width: 100%;
   height: 2px;
   cursor: pointer;
   background: #EFEFEF;
   border-radius: 0;
   border: 0 solid #EFEFEF
}
input[type=range]::-moz-range-thumb {
   border: 1px solid rgba(0,0,30,0);
   height: 22px;
   width: 22px;
   border-radius: 50px;
   background: #ff3034;
   cursor: pointer
}
input[type=range]::-ms-track {
   width: 100%;
   height: 2px;
   cursor: pointer;
   background: 0 0;
   border-color: transparent;
   color: transparent
}
input[type=range]::-ms-fill-lower {
   background: #EFEFEF;
```

```
    border: 0 solid #EFEFEF;
    border-radius: 0
}
input[type=range]::-ms-fill-upper {
    background: #EFEFEF;
    border: 0 solid #EFEFEF;
    border-radius: 0
}
input[type=range]::-ms-thumb {
    border: 1px solid rgba(0,0,30,0);
    height: 22px;
    width: 22px;
    border-radius: 50px;
    background: #ff3034;
    cursor: pointer;
    height: 2px
}
input[type=range]:focus::-ms-fill-lower {
    background: #EFEFEF
}
input[type=range]:focus::-ms-fill-upper {
    background: #363636
}
.switch {
    display: block;
    position: relative;
    line-height: 22px;
    font-size: 16px;
    height: 22px
}
.switch input {
    outline: 0;
    opacity: 0;
    width: 0;
    height: 0
}
.slider {
    width: 50px;
    height: 22px;
    border-radius: 22px;
    cursor: pointer;
    background-color: grey
}
.slider,.slider:before {
    display: inline-block;
    transition: .4s
}
.slider:before {
    position: relative;
    content: "";
```

```
            border-radius: 50%;
            height: 16px;
            width: 16px;
            left: 4px;
            top: 3px;
            background-color: #fff
        }
        input:checked+.slider {
            background-color: #ff3034
        }
        input:checked+.slider:before {
            -webkit-transform: translateX(26px);
            transform: translateX(26px)
        }
        select {
            border: 1px solid #363636;
            font-size: 14px;
            height: 22px;
            outline: 0;
            border-radius: 5px
        }
        .image-container {
            position: relative;
            min-width: 160px
        }
        .close {
            position: absolute;
            right: 5px;
            top: 5px;
            background: #ff3034;
            width: 16px;
            height: 16px;
            border-radius: 100px;
            color: #fff;
            text-align: center;
            line-height: 18px;
            cursor: pointer
        }
        .hidden {
            display: none
        }
    </style>
    <script src='https:\/\/ VolodumurVip.github.io/esp32-cam/face-control/face-
api.min.js'></script>
  </head>
  <body>
  ESP32-CAM IP：<input type="text" id="ip" size="20" value="192.168.">  <input
type="button" value="Set" onclick="start();">
    <figure>
```

```
        <div id="stream-container" class="image-container hidden">
          <div class="close" id="close-stream">x</div>
          <img id="stream" src="" style="display:none" >
          <canvas id="canvas" width="0" height="0"></canvas>
        </div>
      </figure>
      <section class="main">
        <div id="logo">
          <label for="nav-toggle-cb" id="nav-toggle">&#9776;  Toggle OV2640
settings</label>
        </div>
        <div id="content">
          <div id="sidebar">
            <input type="checkbox" id="nav-toggle-cb" checked="checked">
            <nav id="menu">
              <section id="buttons">
                <button id="restart">Restart board</button>
                <button id="stop-still">Stop</button>
                <button id="get-still">Get Still</button>
                <button id="toggle-stream" style="display:none">Start Stream</button>
              </section>
              <div class="input-group" id="uart-group">
                <label for="uart">Recognize face</label>
                <div class="switch">
                  <input id="uart" type="checkbox" class="default-action" checked="checked">
                  <label class="slider" for="uart"></label>
                </div>
              </div>
              <div class="input-group" id="distancelimit-group">
                <label for="distancelimit">Distance Limit</label>
                <div class="range-min">0</div>
                <input type="range" id="distancelimit" min="0" max="1" value="0.4" step="0.1"
class="default-action">
                <div class="range-max">1</div>
              </div>
              <div class="input-group" id="flash-group">
                <label for="flash">Flash</label>
                <div class="range-min">0</div>
                <input type="range" id="flash" min="0" max="255" value="0" class="default-
action">
                <div class="range-max">255</div>
              </div>
              <div class="input-group" id="framesize-group">
                <label for="framesize">Resolution</label>
                <select id="framesize" class="default-action">
                  <option value="10">UXGA(1600x1200)</option>
                  <option value="9">SXGA(1280x1024)</option>
                  <option value="8">XGA(1024x768)</option>
                  <option value="7">SVGA(800x600)</option>
                  <option value="6">VGA(640x480)</option>
```

```
                <option value="5">CIF(400x296)</option>
                <option value="4" selected="selected">QVGA(320x240)</option>
                <option value="3">HQVGA(240x176)</option>
                <option value="0">QQVGA(160x120)</option>
              </select>
            </div>
            <div class="input-group" id="quality-group">
              <label for="quality">Quality</label>
              <div class="range-min">10</div>
              <input type="range" id="quality" min="10" max="63" value="10" class="default-
action">
              <div class="range-max">63</div>
            </div>
            <div class="input-group" id="brightness-group">
              <label for="brightness">Brightness</label>
              <div class="range-min">-2</div>
              <input type="range" id="brightness" min="-2" max="2" value="0" class="default-
action">
              <div class="range-max">2</div>
            </div>
            <div class="input-group" id="contrast-group">
              <label for="contrast">Contrast</label>
              <div class="range-min">-2</div>
              <input type="range" id="contrast" min="-2" max="2" value="0" class="default-
action">
              <div class="range-max">2</div>
            </div>
            <div class="input-group" id="saturation-group">
              <label for="saturation">Saturation</label>
              <div class="range-min">-2</div>
              <input type="range" id="saturation" min="-2" max="2" value="0" class="default-
action">
              <div class="range-max">2</div>
            </div>
            <div class="input-group" id="special_effect-group">
              <label for="special_effect">Special Effect</label>
              <select id="special_effect" class="default-action">
                <option value="0" selected="selected">No Effect</option>
                <option value="1">Negative</option>
                <option value="2">Grayscale</option>
                <option value="3">Red Tint</option>
                <option value="4">Green Tint</option>
                <option value="5">Blue Tint</option>
                <option value="6">Sepia</option>
              </select>
            </div>
            <div class="input-group" id="hmirror-group">
              <label for="hmirror">H-Mirror</label>
              <div class="switch">
```

```
                    <input id="hmirror" type="checkbox" class="default-action"
checked="checked">
                    <label class="slider" for="hmirror"></label>
                  </div>
                </div>
                <div class="input-group" id="vflip-group">
                  <label for="vflip">V-Flip</label>
                  <div class="switch">
                    <input id="vflip" type="checkbox" class="default-action" checked="checked">
                    <label class="slider" for="vflip"></label>
                  </div>
                </div>
                <div class="input-group" id="servo-group">
                  <label for="servo">Servo</label>
                  <div class="range-min">0</div>
                  <input type="range" id="servo" min="0" max="180" value="90" class="default-
action">
                  <div class="range-max">180</div>
                  <select id="pinServo" width="30"><option value="2"
selected>IO2</option><option value="12">IO12</option><option
value="13">IO13</option><option value="14">IO14</option><option
value="15">IO15</option></select>
                </div>
                <div class="input-group" id="relay-group">
                  <label for="relay">Relay</label>
                  <div class="switch">
                    <input id="relay" type="checkbox" class="default-action" checked="checked">
                    <label class="slider" for="relay"></label>
                  </div>
                  <select id="pinRelay" width="30"><option value="2">IO2</option><option
value="12">IO12</option><option value="13" selected>IO13</option><option
value="14">IO14</option><option value="15">IO15</option></select>
                </div>
              </nav>
            </div>
          </div>
        </section>
        Result：<input type="checkbox" id="chkResult" checked>
        <div id="message" style="color:red">Please wait for loading model.<div>

        <script>
        //Розпізнавання фото
        const aiView = document.getElementById('stream')
        const aiStill = document.getElementById('get-still')
        const canvas = document.getElementById('canvas')
        var context = canvas.getContext("2d");
        const message = document.getElementById('message');
        const uart = document.getElementById('uart');
        const chkResult = document.getElementById('chkResult');
```

```
    const distancelimit = document.getElementById('distancelimit')
    var res = "";

    const faceImagesPath = 'https:// VolodumurVip.github.io/esp32-cam/face-control/facelist/';
//Шлях до папки зі зразками фотографій
    const faceLabels = [User1, User2];     //Список папок з фото людей
    faceImagesCount = 2 ;                    //Кількість фотографій у папці користувачів,
іменовані файли JPG із серійними номерами 1.jpg, 2.jpg...

    const modelPath = 'https:// VolodumurVip.github.io/esp32-cam/face-control/'; //Шлях до
файлу моделі
    let displaySize = { width:320, height: 240 }
    let labeledFaceDescriptors;
    let faceMatcher;

    //Завантаження моделей
    Promise.all([
     faceapi.nets.faceLandmark68Net.load(modelPath),
     faceapi.nets.faceRecognitionNet.load(modelPath),
     faceapi.nets.ssdMobilenetv1.load(modelPath)
    ]).then(function(){
     message.innerHTML = "";
     aiStill.click();  //Отримати відеозображення
    })

    async function DetectImage() {  //Виконайте розпізнавання обличчя
     canvas.setAttribute("width", aiView.width);
     canvas.setAttribute("height", aiView.height);
     context.drawImage(aiView,0,0,canvas.width,canvas.height);
     if (!chkResult.checked) message.innerHTML = "";

     if (!labeledFaceDescriptors) {
      message.innerHTML = "Loading face images...";
      labeledFaceDescriptors = await loadLabeledImages();  //Прочитайте зразки фотографій
      message.innerHTML = "";
     }

     if (uart.checked) {
      let displaySize = { width:canvas.width, height: canvas.height }

      faceMatcher = new faceapi.FaceMatcher(labeledFaceDescriptors,
Number(distancelimit.value))  //Верхня межа, якщо людина перевищує це значення, буде
відображатися, як невідомо, інакше відобразиться ім'я людини

      const detections = await
faceapi.detectAllFaces(canvas).withFaceLandmarks().withFaceDescriptors();
      const resizedDetections = faceapi.resizeResults(detections, displaySize);

      const results = resizedDetections.map(d => faceMatcher.findBestMatch(d.descriptor));
```

```
        if (chkResult.checked) message.innerHTML = JSON.stringify(results);
        //console.log(JSON.stringify(detections));
        //console.log(JSON.stringify(resizedDetections));
        //console.log(JSON.stringify(results));

        res = "";
        results.forEach((result, i) => {
          if (uart.checked) {
           //Коли впізнають обличчя
           var query = document.location.origin+'?uart='+result.label;
           fetch(query)
            .then(response => {
             console.log(`request to ${query} finished, status: ${response.status}`)
            })
          }

          res+= i+","+result.label+","+result.distance+"<br>";

          const box = resizedDetections[i].detection.box
          var drawBox = new faceapi.draw.DrawBox(box, { label: result.toString()})
          drawBox.draw(canvas);
         })

         uart.checked = false;  //Оскільки кожне розпізнавання займає трохи пам’яті,
розпізнавання припиняється після розпізнавання.
         if (chkResult.checked) message.innerHTML = res;
       }
       aiStill.click();
     }

     function loadLabeledImages() {  //Прочитайте зразки фотографій
       return Promise.all(
        faceLabels.map(async label => {
         const descriptions = []
         for (let i=1;i<=faceImagesCount;i++) {
          const img = await faceapi.fetchImage(faceImagesPath+label+'/'+i+'.jpg')
          const detections = await
faceapi.detectSingleFace(img).withFaceLandmarks().withFaceDescriptor();
          descriptions.push(detections.descriptor)
         }
         return new faceapi.LabeledFaceDescriptors(label, descriptions)
        })
       )
     }

     aiView.onload = function (event) {
       try {
        document.createEvent("TouchEvent");
        setTimeout(function(){DetectImage();},250);
       } catch(e) {
```

```
        setTimeout(function(){DetectImage();},150);
       }
      }
 double** getGaussian(int height, int width, double sigma) {
   double** filter = new double * [width];
   for (size_t i = 0; i < width; ++i) {
      filter[i] = new double[height];
   }
   double sum = 0;
   for (int i = 0; i < height; i++) {
      for (int j = 0; j < width; j++) {
         filter[i][j] = exp(-(i * i + j * j) / (2 * sigma * sigma)) / (2 * 3.14 * sigma * sigma);
         sum += filter[i][j];
      }
   }

   for (int i = 0; i < height; i++) {
      for (int j = 0; j < width; j++) {
         filter[i][j] /= sum;
      }
   }
   return filter;
}

Image GaussianBlur(Image img, double** filter, int c, int r) {
   Image newimg;
   newimg.cols = img.cols - c + 1;;
   newimg.rows = img.rows - r + 1;;
   newimg.data = new unsigned char * [newimg.rows];
   for (size_t i = 0; i < newimg.rows; ++i) {
      newimg.data[i] = new uchar[newimg.cols];
   }

   for (size_t y = 0; y < newimg.rows; y++) {
      for (size_t x = 0; x < newimg.cols; x++) {
         int sum = 0;
         for (int h = y; h < y + r; h++) {
            for (int w = x; w < x + c; w++) {
               sum += filter[h - y][w - x] * img.data[h][w];
            }
         }
         newimg.data[y][x] = sum;
      }
   }
   for (size_t y = 0; y < c; ++y) {
      delete[] filter[y];
   }
   delete[] filter;

   return newimg;
```

```
}

double FindSigma(const Image& img) {
   double mean = 0;
   double sum = 0;
   double count = 0;
   for (size_t y = 0; y < img.rows; y++) {
      for (size_t x = 0; x < img.cols; x++) {
         count += 1;
         sum += img.data[y][x];
      }
   }
   mean = sum / count;
   double sigma = 0;
   for (size_t y = 0; y < img.rows; y++) {
      for (size_t x = 0; x < img.cols; x++) {
         sigma += (img.data[y][x] - mean) * (img.data[y][x] - mean);
      }
   }
   sigma /= count;
   return sigma;
}

      //Основна функція
      function start() {
        var baseHost = 'http://'+document.getElementById("ip").value;

        const hide = el => {
          el.classList.add('hidden')
        }

        const show = el => {
          el.classList.remove('hidden')
        }

        const disable = el => {
          el.classList.add('disabled')
          el.disabled = true
        }

        const enable = el => {
          el.classList.remove('disabled')
          el.disabled = false
        }

        const updateValue = (el, value, updateRemote) => {
          updateRemote = updateRemote == null ? true : updateRemote
          let initialValue
          if(!el) return;
          if (el.type === 'checkbox') {
```

```
    initialValue = el.checked
    value = !!value
    el.checked = value
  } else {
    initialValue = el.value
    el.value = value
  }

  if (updateRemote && initialValue !== value) {
    updateConfig(el);
  }
}

function updateConfig (el) {
  let value
  switch (el.type) {
    case 'checkbox':
      value = el.checked ? 1 : 0
      break
    case 'range':
    case 'select-one':
      value = el.value
      break
    case 'button':
    case 'submit':
      value = '1'
      break
    default:
      return
  }

  if (el.id =="flash") {  //Додати користувацьку команду flash
    var query = baseHost+"?flash=" + String(value);
  } else if (el.id =="servo") {  //Додайте користувацькі команди сервоприводу
    var query = baseHost+"?servo=" + pinServo.value + ";" + String(value);
  } else if (el.id =="relay") {  //Додайте спеціальні інструкції реле
    var query = baseHost+"?relay=" + pinRelay.value + ";" + Number(relay.checked);
  } else if (el.id =="uart") {  //Додати користувацьку команду uart
    return;
  } else if (el.id =="distancelimit") {  //Додана спеціальна команда distancelimit
    return;
  } else {
    var query = `${baseHost}/?${el.id}=${value}`
  }

  fetch(query)
    .then(response => {
      console.log(`request to ${query} finished, status: ${response.status}`)
    })
}
```

```
    document
     .querySelectorAll('.close')
     .forEach(el => {
      el.onclick = () => {
       hide(el.parentNode)
      }
     })

    const view = document.getElementById('stream')
    const viewContainer = document.getElementById('stream-container')
    const stillButton = document.getElementById('get-still')
    const enrollButton = document.getElementById('face_enroll')
    const closeButton = document.getElementById('close-stream')
    const stopButton = document.getElementById('stop-still')          //Додано змінну
stopButton
    const restartButton = document.getElementById('restart')          //Додайте змінну
перезапуску
    const flash = document.getElementById('flash')                  //Додати змінну flash
    const servo = document.getElementById('servo')                   //Додати змінну
сервоприводу
    const pinServo = document.getElementById('pinServo');            //Додати змінну pin servo
    const relay = document.getElementById('relay')                  //Додати змінну реле
    const pinRelay = document.getElementById('pinRelay');            //Додати змінну контакту
реле
    const uart = document.getElementById('uart')                    //Додайте змінну uart
    var myTimer;
    var restartCount=0;
    var streamState = false;

    stopButton.onclick = function (event) {
     window.stop();
     message.innerHTML = "";
    }

    // Attach actions to buttons
    stillButton.onclick = () => {
     view.src = `${baseHost}/?getstill=${Date.now()}`
     show(viewContainer);
    }

    closeButton.onclick = () => {
     hide(viewContainer)
    }

    //Додано подію натискання кнопки живлення перезавантаження
    restartButton.onclick = () => {
     fetch(baseHost+"/?restart");
    }
```

```
        // Attach default on change action
        document
          .querySelectorAll('.default-action')
          .forEach(el => {
            el.onchange = () => updateConfig(el)
          })

        framesize.onchange = () => {
          updateConfig(framesize)
        }

        // read initial values
        fetch(`${baseHost}/?status`)
        .then(function (response) {
          return response.json()
        })
        .then(function (state) {
          document
          .querySelectorAll('.default-action')
          .forEach(el => {
            if (el.id=="flash") {  //Додано значення параметра світлодіода за замовчуванням 0
              flash.value=0;
              var query = baseHost+"?flash=0";
              fetch(query)
                .then(response => {
                  console.log(`request to ${query} finished, status: ${response.status}`)
                })
            } else if (el.id=="servo") {  //Додайте сервопривод, щоб встановити значення за
замовчуванням 90 градусів
              servo.value=90;
              /*
              var query = baseHost+"?servo=" + pinServo.value + ";90";
              fetch(query)
                .then(response => {
                  console.log(`request to ${query} finished, status: ${response.status}`)
                })
              */
            } else if (el.id=="relay") {  //Додано значення параметра реле за замовчуванням 0
              relay.checked = false;
              /*
              var query = baseHost+"?relay=" + pinRelay.value + ";0";
              fetch(query)
                .then(response => {
                  console.log(`request to ${query} finished, status: ${response.status}`)
                })
              */
            } else if (el.id=="uart") {  //Додано значення параметра uart за замовчуванням 0
              uart.checked = false;
            } else if (el.id=="distancelimit") {  //Додано значення за замовчуванням для обмеження
відстані 0.4
```

```
            distancelimit.value = 0.4;
           } else {
            updateValue(el, state[el.id], false)
           }
          })
        })
      }

      var href=location.href;
      if (href.indexOf("?")!=-1) {
       ip.value = location.search.split("?")[1].replace(/http:\/\//g,"");
       start();
      }
      else if (href.indexOf("http")!=-1) {
       ip.value = location.host;
       start();
      }

   </script>
   </body>
</html>
)rawliteral";

//Встановіть початкове значення меню, щоб отримати формат json
void status(){
 //Повернути статус відео
 sensor_t * s = esp_camera_sensor_get();
 String json = "{";
 json += "\"framesize\":"+String(s->status.framesize)+",";
 json += "\"quality\":"+String(s->status.quality)+",";
 json += "\"brightness\":"+String(s->status.brightness)+",";
 json += "\"contrast\":"+String(s->status.contrast)+",";
 json += "\"saturation\":"+String(s->status.saturation)+",";
 json += "\"special_effect\":"+String(s->status.special_effect)+",";
 json += "\"vflip\":"+String(s->status.vflip)+",";
 json += "\"hmirror\":"+String(s->status.hmirror);
 json += "}";

 client.println("HTTP/1.1 200 OK");
 client.println("Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept");
 client.println("Access-Control-Allow-Methods: GET,POST,PUT,DELETE,OPTIONS");
 client.println("Content-Type: application/json; charset=utf-8");
 client.println("Access-Control-Allow-Origin: *");
 client.println("Connection: close");
 client.println();

 for (int Index = 0; Index < json.length(); Index = Index+1024) {
  client.print(json.substring(Index, Index+1024));
 }
}
```

```
//Повернути домашню сторінку HTML або вміст змінної зворотного зв’язку
void mainpage() {
 client.println("HTTP/1.1 200 OK");
 client.println("Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept");
 client.println("Access-Control-Allow-Methods: GET,POST,PUT,DELETE,OPTIONS");
 client.println("Content-Type: text/html; charset=utf-8");
 client.println("Access-Control-Allow-Origin: *");
 client.println("Connection: close");
 client.println();

 String Data="";
 if (cmd!="")
  Data = Feedback;
 else
  Data = String((const char *)INDEX_HTML);

 for (int Index = 0; Index < Data.length(); Index = Index+1024) {
  client.print(Data.substring(Index, Index+1024));
 }
}

//Повернути зображення у форматі JPEG
void getStill() {
 camera_fb_t * fb = NULL;
 fb = esp_camera_fb_get();
 if(!fb) {
  Serial.println("Camera capture failed");
  delay(1000);
  ESP.restart();
 }

 client.println("HTTP/1.1 200 OK");
 client.println("Access-Control-Allow-Origin: *");
 client.println("Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept");
 client.println("Access-Control-Allow-Methods: GET,POST,PUT,DELETE,OPTIONS");
 client.println("Content-Type: image/jpeg");
 client.println("Content-Disposition: form-data; name=\"imageFile\"; filename=\"picture.jpg\"");
 client.println("Content-Length: " + String(fb->len));
 client.println("Connection: close");
 client.println();

 uint8_t *fbBuf = fb->buf;
 size_t fbLen = fb->len;
 for (size_t n=0;n<fbLen;n=n+1024) {
  if (n+1024<fbLen) {
   client.write(fbBuf, 1024);
   fbBuf += 1024;
  }
  else if (fbLen%1024>0) {
```

```
    size_t remainder = fbLen%1024;
    client.write(fbBuf, remainder);
   }
  }
  esp_camera_fb_return(fb);

//  pinMode(4, OUTPUT);
//  digitalWrite(4, LOW);
}

//Отримуйте останні новини від Telegram
void getTelegramMessage(String token) {
  const char* myDomain = "api.telegram.org";
  String getAll="", getBody = "";
  JsonObject obj;
  DynamicJsonDocument doc(1024);
  String result;
  long update_id;
  String message;
  long message_id;
  String text;

  WiFiClientSecure client_tcp;
  client_tcp.setInsecure();   //run version 1.0.5 or above
  Serial.println("Connect to " + String(myDomain));
  if (client_tcp.connect(myDomain, 443)) {
   Serial.println("Connection successful");
   timer = 0;
   if (client_tcp.connected()) {
    while (timer<timerLimit) {  //Припиніть моніторинг повідомлень через timerLimit після
останнього отримання повідомлення
      getAll = "";
      getBody = "";

      String request = "limit=1&offset=-1&allowed_updates=message";
      client_tcp.println("POST /bot"+token+"/getUpdates HTTP/1.1");
      client_tcp.println("Host: " + String(myDomain));
      client_tcp.println("Content-Length: " + String(request.length()));
      client_tcp.println("Content-Type: application/x-www-form-urlencoded");
      client_tcp.println("Connection: keep-alive");
      client_tcp.println();
      client_tcp.print(request);

      int waitTime = 5000;   // timeout 5 seconds
      long startTime = millis();
      boolean state = false;

      while ((startTime + waitTime) > millis()) {
       //Serial.print(".");
       delay(100);
```

```
        while (client_tcp.available()) {
          char c = client_tcp.read();
          if (c == '\n') {
            if (getAll.length()==0) state=true;
            getAll = "";
          } else if (c != '\r')
            getAll += String(c);
          if (state==true) getBody += String(c);
          startTime = millis();
        }
        if (getBody.length()>0) break;
      }

      //Отримайте значення формату json останнього повідомлення
      deserializeJson(doc, getBody);
      obj = doc.as<JsonObject>();
      //result = obj["result"].as<String>();
      //update_id =  obj["result"][0]["update_id"].as<String>().toInt();
      //message = obj["result"][0]["message"].as<String>();
      message_id = obj["result"][0]["message"]["message_id"].as<String>().toInt();
      text = obj["result"][0]["message"]["text"].as<String>();

      if (message_id!=message_id_last&&message_id) {
        int id_last = message_id_last;
        message_id_last = message_id;
        if (id_last==0) {
          message_id = 0;
        } else {
          Serial.println(getBody);
          Serial.println();
        }

        if (text!="") {
          Serial.println("["+String(message_id)+"] "+text);
          telegramCommand(text);  //Виконайте команди
        }
      }
      delay(1000);
      timer++;
    }
   }
  }
}

//Надсилайте зображення в Telegram-бот
String sendCapturedImage2Telegram(String token, String chat_id) {
  const char* myDomain = "api.telegram.org";
  String getAll="", getBody = "";

  camera_fb_t * fb = NULL;
```

```
 fb = esp_camera_fb_get();
 if(!fb) {
  Serial.println("Camera capture failed");
  delay(1000);
  ESP.restart();
  return "Camera capture failed";
 }

 Serial.println("Connect to " + String(myDomain));

 WiFiClientSecure client_tcp;
 client_tcp.setInsecure();   //run version 1.0.5 or above

 if (client_tcp.connect(myDomain, 443)) {
  Serial.println("Connection successful");

  String head = "--Taiwan\r\nContent-Disposition: form-data; name=\"chat_id\"; \r\n\r\n" +
chat_id + "\r\n--Taiwan\r\nContent-Disposition: form-data; name=\"photo\"; filename=\"esp32-
cam.jpg\"\r\nContent-Type: image/jpeg\r\n\r\n";
  String tail = "\r\n--Taiwan--\r\n";

  uint16_t imageLen = fb->len;
  uint16_t extraLen = head.length() + tail.length();
  uint16_t totalLen = imageLen + extraLen;

  client_tcp.println("POST /bot"+token+"/sendPhoto HTTP/1.1");
  client_tcp.println("Host: " + String(myDomain));
  client_tcp.println("Content-Length: " + String(totalLen));
  client_tcp.println("Content-Type: multipart/form-data; boundary=Taiwan");
  client_tcp.println();
  client_tcp.print(head);

  uint8_t *fbBuf = fb->buf;
  size_t fbLen = fb->len;
  for (size_t n=0;n<fbLen;n=n+1024) {
   if (n+1024<fbLen) {
    client_tcp.write(fbBuf, 1024);
    fbBuf += 1024;
   } else if (fbLen%1024>0) {
    size_t remainder = fbLen%1024;
    client_tcp.write(fbBuf, remainder);
   }
  }

  client_tcp.print(tail);

  esp_camera_fb_return(fb);

  int waitTime = 10000;   // timeout 10 seconds
  long startTime = millis();
```

```
  boolean state = false;

  while ((startTime + waitTime) > millis()) {
   Serial.print(".");
   delay(100);
   while (client_tcp.available()) {
      char c = client_tcp.read();
      if (state==true) getBody += String(c);
      if (c == '\n') {
        if (getAll.length()==0) state=true;
        getAll = "";
      }
      else if (c != '\r')
        getAll += String(c);
      startTime = millis();
    }
    if (getBody.length()>0) break;
  }
  client_tcp.stop();
  Serial.println();
  Serial.println(getBody);
 } else {
  getBody="Connected to api.telegram.org failed.";
  Serial.println("Connected to api.telegram.org failed.");
 }

//  pinMode(4, OUTPUT);
//  digitalWrite(4, LOW);

 return getBody;
}

//Надсилайте текстові повідомлення Telegram і командні кнопки
String sendMessage2Telegram(String token, String chat_id, String text, String keyboard) {
 const char* myDomain = "api.telegram.org";
 String getAll="", getBody = "";

 String request = "parse_mode=HTML&chat_id="+chat_id+"&text="+text;
 if (keyboard!="") request += "&reply_markup="+keyboard;

 Serial.println("Connect to " + String(myDomain));

 WiFiClientSecure client_tcp;
 client_tcp.setInsecure();   //run version 1.0.5 or above

 if (client_tcp.connect(myDomain, 443)) {
  Serial.println("Connection successful");
  client_tcp.println("POST /bot"+token+"/sendMessage HTTP/1.1");
  client_tcp.println("Host: " + String(myDomain));
  client_tcp.println("Content-Length: " + String(request.length()));
```

```
   client_tcp.println("Content-Type: application/x-www-form-urlencoded");
   client_tcp.println("Connection: keep-alive");
   client_tcp.println();
   client_tcp.print(request);

   int waitTime = 5000;   // timeout 5 seconds
   long startTime = millis();
   boolean state = false;

   while ((startTime + waitTime) > millis()) {
    Serial.print(".");
    delay(100);
    while (client_tcp.available()) {
       char c = client_tcp.read();
       if (state==true) getBody += String(c);
       if (c == '\n') {
         if (getAll.length()==0) state=true;
         getAll = "";
       } else if (c != '\r')
         getAll += String(c);
       startTime = millis();
     }
     if (getBody.length()>0) break;
   }
   client_tcp.stop();
   Serial.println();
   Serial.println(getBody);
  } else {
   getBody="Connected to api.telegram.org failed.";
   Serial.println("Connected to api.telegram.org failed.");
  }

//  pinMode(4, OUTPUT);
//  digitalWrite(4, LOW);

  return getBody;
}

//Виконайте команди повідомлення Telegram
void telegramCommand(String text) {
 if (!text||text=="") return;
 timer = 0;
 // Індивідуальна інструкція
 if (text=="/still") {        //Зробіть скріншот відео
   sendCapturedImage2Telegram(myToken, myChatId);
 } else if (text=="/ledon") {  //Увімкніть світлодіод
   ledcDetachPin(4);
   pinMode(4 , OUTPUT);
   digitalWrite(4, HIGH);
   sendMessage2Telegram(myToken, myChatId, "Turn on the flash", "");
```

```
  } else if (text=="/ledoff") {  //Вимкніть світлодіод
    ledcDetachPin(4);
    pinMode(4 , OUTPUT);
    digitalWrite(4, LOW);
    sendMessage2Telegram(myToken, myChatId, "Turn off the flash", "");
  } else if (text=="/open") {  //Відкрийте дверний замок
    pinMode(pinDoor , OUTPUT);
    digitalWrite(pinDoor, HIGH);
    delay(2000);
    digitalWrite(pinDoor, LOW);
  }
}
```

Додаток Г

Схема електрична принципова

Додаток Д

Ілюстраційний матеріал

Додаток Е

Протокол перевірки навчальної (кваліфікаційної) роботи