**Міністерство освіти і науки України**
**Національний технічний університет**
**«Дніпровська політехніка»**

Навчально-науковий

інститут електроенергетики
(інститут)
Факультет інформаційних технологій
(факультет)
Кафедра інформаційних технологій та комп'ютерної інженерії
(повна назва)

# ПОЯСНЮВАЛЬНА ЗАПИСКА
**кваліфікаційної роботи ступеня** ____магістра___
(бакалавра, спеціаліста, магістра)

**студента** Соболевського Івана Олексійовича
(ПІБ)
**академічної групи** 123м-21-1
(шифр)
**спеціальності** 123 «Комп'ютерна інженерія»
(код і назва спеціальності)
**за освітньо-професійною програмою** 123 Комп'ютерна інженерія
(офіційна назва)
на тему: «Програмно-технічна реалізація комп'ютерної системи IP-відео-нагляду комплексу «Золоті ключі» з опрацюванням передачі відео інформації на базі Raspberry Pi»
(назва за наказом ректора)

| **Керівники** | **Прізвище, ініціали** | **Оцінка за шкалою** | | **Підпис** |
|---|---|---|---|---|
| | | **рейтинговою** | **інституційною** | |
| кваліфікаційної роботи | проф. Цвіркун Л.І. | | | |
| розділів: | | | | |
| теоретичний розділ | проф. Цвіркун Л.І. | | | |
| синтез системи | доц. Бешта Д.О. | | | |
| розроблення програмного забезпечення | ас. Панферова Я.В. | | | |

| | | | | |
|---|---|---|---|---|
| **Рецензент** | | | | |

| | | | | |
|---|---|---|---|---|
| **Нормоконтролер** | проф. Цвіркун Л.І. | | | |

**Дніпро**
**2022**

**ЗАТВЕРДЖЕНО:**

завідувач кафедри
інформаційних технологій
та комп'ютерної інженерії
(повна назва)

_____ Гнатушенко В.В.
(підпис) (прізвище, ініціали)
«___»_____ 2022 року

**ЗАВДАННЯ**
**на кваліфікаційну роботу**
**ступеня** магістра
(бакалавра, спеціаліста, магістра)

**студенту** Соболевського І.О. **академічної групи** 123м-21-1
(прізвище та ініціали) (шифр)

**спеціальності** 123 «Комп'ютерна інженерія»
(код і назва спеціальності)

**за освітньо-професійною програмою** 123 Комп'ютерна інженерія
(офіційна назва)

на тему: ««Програмно-технічна реалізація комп'ютерної системи IP-відео-нагляду комплексу «Золоті ключі» з опрацюванням передачі відео інформації на базі Raspberry Pi»
(назва за наказом ректора)

затвержена наказом ректора НТУ "Дніпровська політехніка" від 31 жовтня 2022 р. № 1200

| Розділ | Зміст | Термін виконання |
|---|---|---|
| Стан питання та постановка завдання | На основі матеріалів виробничих практик, інших науково-технічних джерел сформулювати наукове завдання, конкретизувати предмет та мету досліджень | 10.10.2022 |
| Теоретичний | Обґрунтувати теоретичну базу розв'язання наукового завдання, якому присвячено роботу | 24.10.2022 |
| Синтез системи | Розробка комп'ютерної системи | 14.11.2022 |
| Експериментальний розділ | Проведення і обробка результатів експериментів | 05.12.2022 |
| Графічна частина | Графічні результати роботи подати у вигляді рисунків схем таблиць на 10 арк. формату А4. | 10.12.2022 |

**Завдання видано** _____ проф. Цвіркун Л.І.
(підпис керівника) (прізвище та ініціали)

**Дата видачі** 10 жовтня 2022 р.

**Дата подання до екзаменаційної комісії** 15.12.2022 р.

**Прийнято до виконання** _____ Соболевський І.О.
(підпис студента) (прізвище та ініціали)

# РЕФЕРАТ

Пояснювальна записка 84 с., 30 рис., 6 табл, 1 дод., 31 джерел.

ВІДЕОСПОСТЕРЕЖЕННЯ, АНАЛІЗ ВІДЕОПОТОКУ, НЕЙРОННІ МЕРЕЖІ, TENSORFLOW, РОЗПІЗНАВАННЯ ОБ'ЄКТІВ, DOCKER

Об'єкт розробки: комп'ютерна система відеоспостереження комплексу "Золоті ключі".

Мета роботи: розробити комп'ютерну систему відеоспостереження аналітичним модулем для комплексу "Золоті ключі". Обґрунтувати використання власної системи відеоспостереження поміж існуючих рішень.

Пояснювальна записка має аналіз існуючих систем відеоспостереження та систем аналізу відеопотоку, описує недоліки та переваги кожної з них.

За допомогою цих даних було сформульовано завдання дослідження.

У теоретичному розділі вирішено наукове завдання побудувавши модель системи відеоспостереження з аналітичним модулем.

У розділі «Синтез системи» сформульовані технічні вимоги до створюваної системи, побудована структурна схема системи відеоспостереження.

У розділі «Розроблення програмного забезпечення» проведена розробка програмного забезпечення на основі побудованих схем алгоритмів, описаний зв'язок між програмами та їх функціональні можливості.

В експериментальному розділі поставлена задача експерименту і проведено експеримент створеної системи відеоспостереження з аналітичним модулем.

**ЗМІСТ**

# ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАК, ОДИНИЦЬ І ТЕРМІНІВ

БД – база даних;

КС – комп’ютерна система;

ДБЖ – джерело безперебійного живлення;

ПЗ – програмне забезпечення;

АН – агентство нерухомості;

SSH – Secure SHell;

JWT – JSON WEB TOKEN;

RAID – Redundant Array of Independent Disks;

PAL – Phase Alternating Line;

API – Application Programming Interface;

AI – Artificial intelligence;

RAM – Random Access Memory;

WebHook – функція, яка призначена для того, щоб надати веб-адресату інформацію після виконання певних умов;

AWS – Amazon Web Services;

LTS – long term support;

SQL – Structured query language;

CPU – Central processing unit;

GPU – Graphics processing unit.

# ВСТУП

Системи відеоспостереження є невід'ємною частиною безпеки у сучасному світі. Воно потрібно для забезпечення протидії злочинності.

Системи відеоспостереження варіюються від однієї камери для спостереження за клієнтами на касі до загальноміських систем, які контролюють тисячі камер. Основною проблемою рішень на ринку є відсутність централізованої системи з аналітичними можливостями для розпізнавання та інформування користувачів про помічені об'єкти.

**Мета і завдання дослідження.** *Метою роботи* є розробка і дослідження власної структури програмно-технічних засобів комп'ютерної системи відеоспостереження комплексу "Золоті ключі".

Для досягнення поставленої мети необхідно вирішити такі завдання:

– дослідити та проаналізувати існуючі системи відеоспостереження;

– на основі дослідження існуючих систем відеоспостереження, обробити їх характеристики, виділити переваги та недоліки;

– розробити модель системи відеоспостереження;

– розробити специфікацію обладнання системи;

– розробити структурну схему системи відеоспостереження;

– розробити схеми алгоритмів для системи відеоспостереження;

– програмно реалізувати отриманні моделі, методи та алгоритми і побудувати дослідницький прототип системи відеоспостереження;

– проведення експерименту системи відеоспостереження

*Об'єкт дослідження* – процес розробки комп'ютерної системи відеоспостереження комплексу "Золоті ключі".

*Предмет дослідження* – методи аналітичної обробки відеозображення з камери.

*Методи дослідження.* Для досягнення поставленої мети використано методи аналізу та синтезу, порівняння, а також метод математичного моделювання.

*Наукові положення:*

1. Розроблені та обґрунтовані вимоги щодо моделювання системи відеоспостереження.

2. Описаний зв’язок між компонентами програмного забезпечення для інтеграції функціонально розмеженних модулів до системи.

*Наукові результати:*

1. Запропонований метод аналізу відеозображення, який відрізняється тим, що здатен працювати з відеопотоком камер спостереження, здатен забезпечувати аналіз в реальному часі та сповіщати користувачів про помічений об’єкт.

2. Обґрунтовано застосування власної системи відеоспотереження з модулем аналізу відеопотоку поміж конкурентів завдяки виконанню порівняльних експериментальних досліджень.

**Обґрунтованість і достовірність наукових положень, висновків і рекомендацій** підтверджуються тим, що в роботі використані: сучасні методи для створення систем аналізу та розпізнавання образів, з використанням нейронних мереж, сучасне обладнання для побудови системи, ефективні методи розробки програмного забезпечення, такі як мікросервісна архітектура та функціональне програмування, експериментальні підтвердження результатів теоретичних досліджень.

**Практичне значення отриманих результатів полягає** в розробці методу аналізу відеозображення, який дозволяє аналізувати та розпізнавати об’єкти у реальному часі та інтеграції його до системи відеоспостереження.

# 1 СТАН ПИТАННЯ ТА ПОСТАНОВКА ЗАВДАННЯ

## 1.1 Стан питання

На даний момент котеджні містечка користуються шаленою популярністю. Це майже райські куточки, ізольовані від галасливої цивілізації. У такому котеджному містечку, як «Золоті ключі», люди можуть придбати як готовий будинок, так і земельну ділянку, щоб побудувати будинок їхньої мрії, відповідно до фантазії архітектора, смаку власника та бюджету майбутнього власника.

У цих місцях гостро стоїть питання безпеки, вона має бути завжди на першому плані.

Навіть за наявності охорони не слід нехтувати безпекою від внутрішніх загроз, таких як сусіди, гості, співробітники, туристи, люди, які тимчасово орендують житло.

Кожен будинок має бути захищений системою відеоспостереження, щоб запобігти злочинам з найменш очікуваних сторін.

Відеонагляд повинен виконувати такі функції:

а) відеозапис на певний проміжок часу – відеозапис завжди має бути доступним як доказ для правоохоронних служб;

б) відображення камер в режимі реального часу – відеоспостереження має показувати реальне зображення в режимі реального часу є важливою функцією, яка може бути корисною для:

1) регулювання, для того щоб знайти найкращий кут огляду;

2) переглянути поточну ситуацію, наприклад, отримати автоматичне сповіщення про тривогу;

3) аналіз роботоспроможності камер;

в) режими нагляду та дозволу:

1) нагляд – розпізнавання та сповіщення про загрози:

— користувача;

— охорону;

— правозахисні органи;

2) дозвіл – охоронна сигналізація відключається, тому що, наприклад, в будинку є уповноважені особи і сигналізація не має сенсу.

Такі системи відеонагляду зазвичай виконують свою функцію вже постфактум, допомагають з ходом розслідування вже після скоєння злочину. Для того, щоб система відеонагляду допомагала запобігати скоєнню злочинів, потрібно використати сучасні технології, такі як нейронні мережі та методи розпізнавання об’єктів та обличь, це дозволить швидко реагувати на загрози і вчасно інформувати охорону та поліцію.

### 1.2 Аналіз існуючих систем

Системи відеоспостереження – це програмно-технічні засоби, які являються важливим елементом безпеки, вони встановлюються в будинках, офісах, дворах та інших місцях для протидії злочинам. Система може бути простою або складною. Проста система відеоспостереження може складатися з однієї камери, яка може передавати відеосигнал на монітор. Що стосовно комплексної системи, вона може працювати за протоколом IP і транслювати відео на кілька підключених комп'ютерів одночасно. Системи відеоспостереження поділяються на аналогові та цифрові. [1]

#### 1.2.1 Аналогові системи відеоспостереження

Аналогові системи відеоспостереження створюються за допомогою коаксіального кабелю, кабелю з витою парою або оптоволоконної системи. Стандартним способом передачі відеосигналу в аналогових системах є використання телевізійного кабелю PAL. Ця система вважається застарілою та використовується в місцях, де необхідно встановити недороге відеоспостереження.

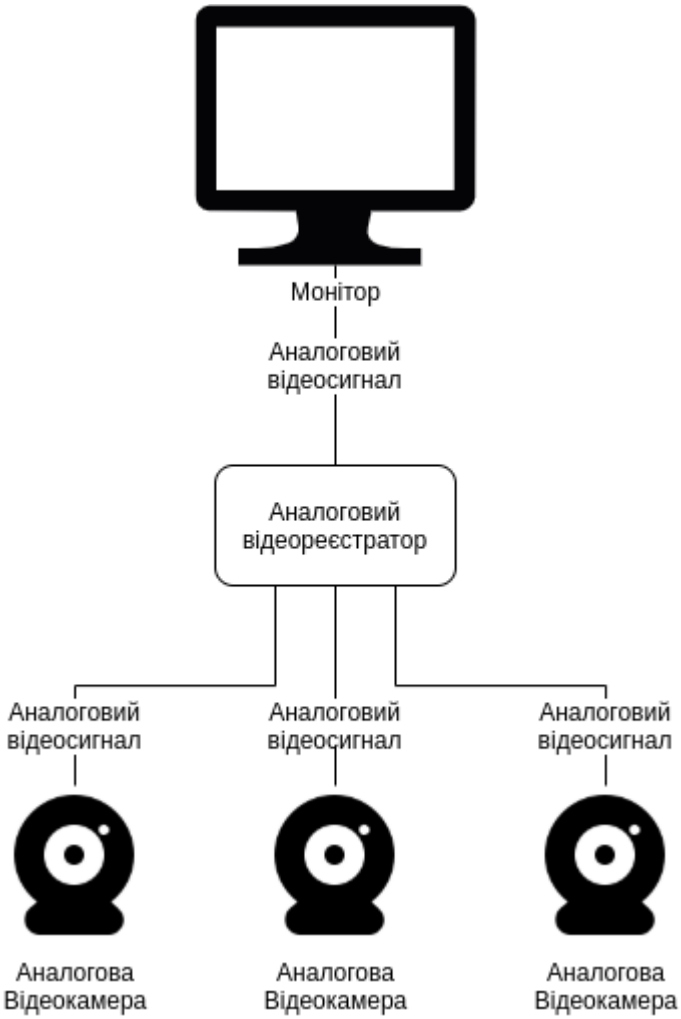


Рисунок 1.1 – Приклад аналогової системи відеоспостереження

Переваги:

— недорога ціна;

— висока надійність;

— простота підключення;

— легке налаштування;

— простота використання.

Недоліки:

— необхідність постійного обслуговування;

— немає цифрового кодування;

— системою можна керувати тільки з місця її встановлення;

— максимально, кількість встановлених камер залежить від вбудованих роз’ємів;

— відсутність можливості кодування відеосигналу;
— відсутність можливості доступу з мережі;
— неможливо програмно аналізувати події в кадрі;
— неможливість керувати масштабуванням або рухом.

### 1.2.2 Цифрові системи відеоспостереження

Цифрові системи відеоспостереження мають переваги перед аналоговими системами і є їх сучасними нащадками. Вони доступні в декількох варіаціях.

Перший варіант (гібридний) — Аналогові камери підключаються до комп'ютера, який оцифровує сигнал для передачі сигналу з камер у мережу або запису відеопотоку в цифровому форматі. Такі системи зазвичай розміщують там, де вже розташовані аналогові камери, і оцифрувати їх буде дешевше, ніж повністю замінити.

Другий варіант — це технологія HD-SDI (послідовний цифровий інтерфейс високої чіткості), передача відеосигналу через послідовний цифровий інтерфейс для передачі зображень високої роздільної здатності. HD-SDI камери є проміжною ланкою між аналоговими та IP системами. З телебачення цей стандарт перейшов на відеоспостереження.

HD-SDI створюється з аналогового композитного сигналу, сигнали спочатку розбиваються на компоненти: яскравість Y і колірні сигнали U (або Cr) і V (або Cb). Потім кожен компонент оцифровується та передається до кодера, де збереження даних відбувається в послідовності, яка відповідає структурі SDI.

HD-SDI використовують стандартизоване співвідношення сторін 16:9. Залежно від роздільної здатності зображення камери високої роздільної здатності діляться на два типи:

1) HDTV 720p, підтримка роздільної здатності 1280x720 пікселів, висока точність кольору, співвідношення сторін 16:9, використання послідовного сканування.

2) HDTV 1080p, підтримка роздільної здатності 1280x720 пікселів, висока точність кольору, співвідношення сторін 16:9 [2].

Переваги:

— хороша якість зображення, підключення через цифровий інтерфейс, не псується зображення при дистанційній передачі;

— відсутність затримки;

— швидке та ефективне налаштування.

— Недоліки:

— відстань передачі (максимум 150 метрів через коаксіальний кабель);

— вартість пристрою вища, ніж у аналогових систем.

### 1.2.3 IP системи відео спостереження

Інтернет-протокол (IP) — це базовий набір (або комунікаційний протокол) форматів цифрових повідомлень і правил для обміну повідомленнями між комп'ютерами в одній мережі або кількох взаємопов'язаних мережах за допомогою набору протоколів Інтернету (часто називаються TCP/IP). Обмін повідомленнями здійснюється у формі дейтаграм, також відомих як пакети даних або просто пакети. Набір комунікаційних протоколів, що складається з чотирьох рівнів абстракції: комунікаційного (найнижчого) рівня, Інтернет-рівня, транспортного рівня та прикладного рівня (найвищого) [3].

IP відеоспостереження в основному передбачає використання IP камер і може працювати без відеореєстратора. IP-камера записує відео та передає дані через локальну мережу, як зображено на рисунку 1.2. Камери мають датчики руху, мікрофони, сигналізацію, нічник, аудіо вхід і вихід, веб-сервер. IP камери налаштовуються через браузер. Багато IP-камери також мають можливість підключення через WiFi, що дозволяє встановлювати їх у важкодоступних місцях. Дані можуть зберігатися на жорстких дисках або за допомогою хмарних технологій.

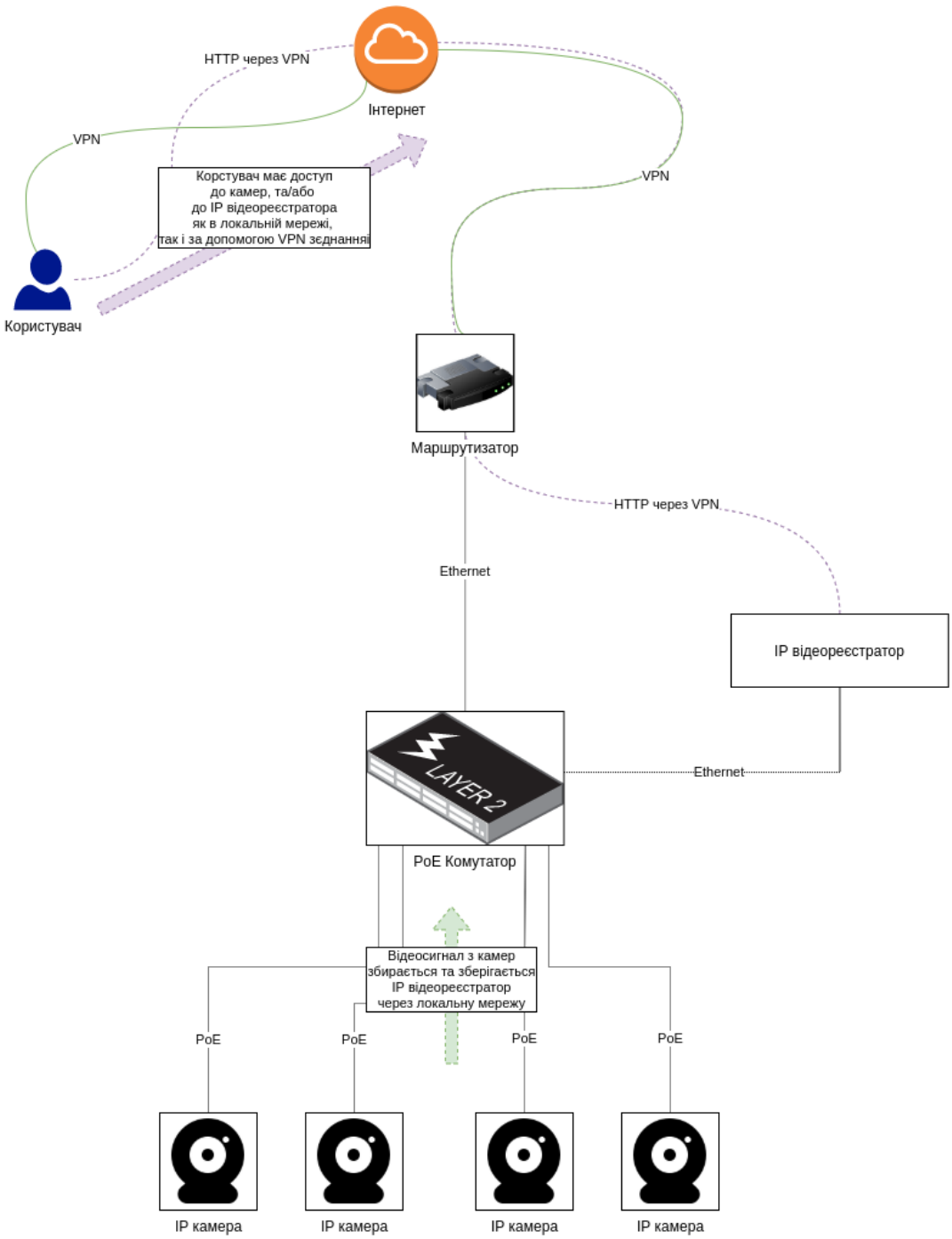


Рисунок 1.2 – Схема підключення цифрового відеоспостереження через мережевий відеореєстратор

Система IP-відеоспостереження пропонує значні переваги користувачам і є проривом у сфері безпеки та науки.

IP-камера – це невеликий комп’ютер, у якому є центральний процесор, який керує відеокамерою, мережевим інтерфейсом і модулями мікрофона або виходами сигналізації, процесор також контролює стиснення відеопотоку. Як

і будь-який комп’ютер, IP-камера має операційну систему, як правило, функціонально зменшений дистрибутив Linux із веб-сервером, розгорнутим для HTTP-з’єднання користувача.

RTSP (Real Time Streaming protocol) – це мережевий протокол прикладного рівня, призначений для мультиплексування та пакетування транспортних потоків мультимедіа (таких як інтерактивні медіа, відео та аудіо) через відповідний транспортний протокол. RTSP використовується в розважальних і комунікаційних системах для керування серверами потокового медіа. Протокол використовується для встановлення та контролю медіа-сеансів між кінцевими точками. Клієнти медіа-серверів видають такі команди, як відтворення, запис і пауза, щоб полегшити керування потоковим передаванням медіа від сервера до клієнта (відео на вимогу) або від клієнта до сервера (запис голосу) у реальному часі [4].

HTTP (HyperText Transfer Protocol) – цей протокол використовував старі моделі відеокамер, де відеопотік розбивався на зображення у форматі JPEG і відправлявся на сервер відеокамери, а клієнт отримував їх з певною періодичністю для відновлення зображення. . В даний час HTTP в основному обслуговує веб-сторінки, які відображають параметри відео або камери, які повертаються користувачеві [5].

RTP (транспортний протокол реального часу) – це мережевий протокол для передачі аудіо та відео через IP-мережі. RTP використовується в комунікаційних і розважальних системах, які включають потокове медіа, наприклад телефонію, додатки для відеотелеконференцій, включаючи WebRTC, телевізійні служби та веб-функції push-to-talk [6].

На додаток до звичайних камер відеоспостереження можна використовувати тепловізійні IP-камери, які можна використовувати разом зі звичайними камерами.

Стиснення відбувається в самій IP-камері, щоб зменшити кількість трафіку в мережі, оскільки ширина каналу в IP-мережі не завжди максимальна,

потрібно стискати трафік, щоб не заважати каналу мережі. Існує два основних підходи до стиснення: внутрішньокадрове та міжкадрове:

— внутрішньокадрове стиснення виконується лише всередині кадру, а не між кадрами (MJPEG);
— стиснення між кадрами здійснюється всередині кадрів і в окремих кадрах MPEG-4, H.264, H.265.

Переваги:

— віддалений доступ до камер в онлайн режимі;
— використовуйте камери з різних пристроїв одночасно;
— є можливість налаштувати програмний відеоаналіз, наприклад, зчитування номерних знаків, виявлення рухомих об’єктів або аналіз настрою людей;
— можливість збільшення кількості пристроїв системи;
— можливість підключення по wifi;
— шифрування відео трафіку.

Недоліки:

— вартість IP відеоспостереження дорожче за інші аналоги, та потребує покупку та налаштування мережі для передачі даних;
— якість зображення залежить від пропускної здатності мережі;
— справність системи відеоспостереження залежить від помилок каналу там стану мережі.

### 1.3 Проблеми сучасних систем

Згідно з проаналізованих матеріалів, ми побачили, що найчастішими проблемами сучасних систем відеоспостереження є:

— обмежений функціонал;
— недостатня централізація системи;
— складність налаштувань;
— складні інтерфейси користувачів;

— недостача методів аналізу кадру з розпізнаванням об’єктів, зокрема людей;
— безпека відеоінформації.

### 1.4 Постановка завдання дослідження

Розробка та синтез ПЗ для створення централізованої системи відеоспостереження, транслювання в реальному часі з відеокамер, стирання запису та доступ до відеоматеріалів, за термін визначений адміністратором системи. Надати користувачеві зручний доступ до трансляції та оптимізуйте систему для безперебійної роботи. Забезпечити систему відеоспостереження методами аналізу небезпек, та розпізнаванням об’єктів в кадрі, зокрема людей. Проаналізувати джерела інформації, для покращення продуктивності та роботоспроможності програмного забезпечення.

Задачі які необхідно вирішити:

— вивчити переваги та недоліки сучасних систем відеоспостереження;
— обґрунтувати доцільність використання власної системи відеоспостереження;
— визначення принципу роботи нової системи;
— визначити інструменти для розробки власної системи відеоспостереження для АН Золоті Ключи.

# 2 ТЕОРЕТИЧНИЙ РОЗДІЛ

## 2.1 Загальна характеристика комп'ютерної системи

Об'єктом дослідження є котеджне містечко "Золоті ключі", яке знаходиться в селі Слобожанське (Дніпропетровська область).

Комп'ютерна мережа котеджне містечка "Золоті ключі" складається з:

— 128 робочих станцій;

— 13 комутаторів;

— 8 маршрутизатору;

— 1 веб-сервер;

— 1 FTP-серверу.

## 2.2 Структура об'єкту дослідження

Котеджне містечко "Золоті ключі" має наступні складові комп'ютерної системи:

— маршрутизатори;

— комутатори;

— персональні комп'ютери;

— веб-сервер;

— FTP-сервер;

Ці елементи системи забезпечують безперебійну роботу котеджного містечка "Золоті ключі".

## 2.3 Обґрунтування і вибір методів дослідження

Під час вибору компонентів для побудови покращеної системи спостереження аналізувалися різні методи. Після вивчення кожного з них було вирішено вибрати наступні методи:

— порівняльний метод;

— метод аналізу та синтезу;

— метод математичного моделювання.

### 2.3.1 Методи аналізу та синтезу

Наявні системи відеоспостереження мають певні недоліки, в залежності від обраного типу та марки системи. До найпоширеніших проблем відносяться:

— недостатня централізація системи – система відеоспостереження повинна бути достатньо централізованою, аби забезпечити масштабування та гнучкість використовування декількома групами користувачів;

— обмежений функціонал – функціонал адміністратора повинен забезпечувати можливість підлаштовувати систему для різних сценаріїв використання, наприклад розмежувати права та доступи користувачів, цього бракує в сучасних системах відеоспостереження;

— складність налаштувань - через перенасиченість деяких сучасних систем відеоспостереження дуже важко зробити певні налаштування при її встановленні. А це може зайняти багато часу або призвести до зайвих помилок;

— складні інтерфейси користувачів – новим користувачам важко розібратись з сучасними системами відеоспостереження, система відеоспостереження з боку користувача повинна бути максимально зручною, основні важкі налаштування повинні припасти на адміністратора системи;

— безпека відеоінформації – це один з найважливіших пунктів системи відеоспостереження, якщо інформацію можна поцупити, то це може призвести до того, що інформація попаде до злочинців, які можуть дізнатись про вразливі міста домівок, або підприємства, які ця система повинна захищати.

— відсутність аналізу та розпізнавання об’єктів в кадрі – в сучасних системах відеонагляду відсутнє, або недостатньо інтегроване для використання з масштабною системою відеонагляду, застосування сучасної технології розпізнавання об’єктів в кадрі, такі методи аналізу кадру підвищать спроможність системи забезпечувати безпеку та ушвидшить реакцію на злочинні дії.

### 2.3.2 Порівняльний аналіз існуючих систем відеоспостереження

#### 2.3.2.1 Система відеонагляду Hikvision

Система Hikvision є найбільшим виробником продуктів відеоспостереження на ринку. Ця система розроблена китайською компанією Hikvision Digital Technology Co.

Переваги IP системи Hikvision полягають в тому що вона забезпечує безпечне та надійне з'єднання камери з відеореєстратором та володіє елементами штучного інтелекту з аналітичним програмним забезпеченням. Також перевагою цієї системи можна виділити якість зображення та поширеність системи на ринку [7].

Недоліками системи Hikvision являється:

— висока ціна;

— система немає API, для того, щоб інтегрувати її з іншими сервісами;

— ця система проприєтарна, що несе за собою неможливість доробки програмного коду під нужди системи;

— неможливо підключити камери іншого вендору до системи, що позбавляє систему гнучкості використовуємого обладнання.

#### 2.3.2.2 Система відеонагляду Dahua

Система Dahua являється найближчім конкурентом Hikvision на ринку. Система відеонагляду Dahua також є продуктом китайського ринку.

Перевагами системи Dahua є:

— низька вартість;

— надійність та технологічність системи, що позитивно відображається на процесі експлуатації;

— обладнання цієї системи застосовує матриці з високою роздільною здатністю, що забезпечує деталізація картинки з камери відеонагляду на високому рівні;

— система має простий для конфігурації інтерфейс;

— передбачається живлення с застосуванням технології PoE, що дає можливість заживити камеру та передавати інформацію через один кабель;

— обладнання має мінімальний рівень захисту системи, що гарантує високу надійність системи [8].

Недоліки системи Dahua:

— пропрієтарність системи;

— обмежений функціонал системи;

— немає аналітичного модулю та здібності системи до розпізнавання об’єктів в кадрі.

### 2.3.2.3 Система відеонагляду 360 Vision Technology

Системи відеонагляду 360 Vision Technology є продуктом британської компанії 360 Vision Technology Ltd.

Перевагою системи 360 Vision Technology є:

— використання сучасних технологій аналізу з використанням штучного інтелекту;

— теплові та радіолокатори;

— камери з кутом огляду в 360 градусів;

— є система оповіщення користувача;

— активна техпідтримка з боку розробників системи.

Недоліками системи є:

— дуже висока ціна;

— проприєтарність системи;

— систему складно дістати на ринку України;

— складний інтерфейс користувача [9].

### 2.3.2.4 Власна система відеонагляду

Переваги власної системи відеонагляду:

— система не буде прив’язана до конкретного вендору та постачальника обладнання;

— можливість вживити систему сповіщень користувачів;

— гнучкість системи;

— економія на обладнані;

— можливість вживити власний аналітичний модуль, яким можна гнучко керувати під потреби системи та користувача;

— можливість використати власний сервер для зберігання данних з камер;

— мінімалістичний інтерфейс користувача;

— легкість налаштування камер та дозволів.

Недоліки власної системи відеонагляду:

— потрібен час на розробку нової системи;

— необхідна підтримка системи з боку розробника.

#### 2.3.2.5 Висновки порівняльного аналізу систем відеоспостереження

Провівши аналіз систем відеоспостереження на ринку, було зроблено висновок, що кожна з них не задовольняє поставлених потреб. З цього слідує, що розробка власної системи відеоспостереження доцільна, так як дозволить розробити систему з усім необхідним функціоналом. Розробка власної системи дозволить мати гнучкий функціонал, не мати прив’язки до конкретного вендору обладнання, водночас мати централізований характер керування та застосовувати налаштування доступів під конкретного користувача системи.

### 2.3.3 Порівняльний аналіз існуючих систем аналізу відеопотоку

Для розробки власної системи відеоспостереження з модулем аналізу відеопотоку потрібно проаналізувати існуючі системи аналізу відеопотоку, для того, щоб обрати найбільш підходящий варіант. Аналіз відеопотоку відбувається за допомогою нейронних мереж, які були попередньо налаштовані та навчені, за допомогою різного роду зображень та відеофайлів, для того, щоб аналіз інформації був більш точним та швидким.

#### 2.3.3.1 Система аналізу відеопотоку Hikvision AI

Ця система поширено використовується для аналізу відеоіформації в системі Hikvision і є одною з небагатьох подібних систем, які в даний момент присутні та інтегровані в обладнання для відеонагляду.

Переваги системи Hikvision AI:

— система є професійною та спеціально заточеною під відеонагляд;

— швидкий аналіз вхідного сигналу;

— аналіз працює за умовами недостатнього освітлення;

— автоматичний рівень контрастності зображення.

Недоліки системи Hikvision AI:

— система працює за принципом чорної коробки, це унеможливлює гнучке налаштування системи, немає прямого зв’язку між вхідними та вихідними даними;

— система є пропрієтарною, та дає можливість застосовуватись тільки з обладнанням Hikvision [10].

#### 2.3.3.2 Система аналізу відеопотоку DeepVision AI

Система аналізу відеопотоку DeepVision AI являється рішенням SCaaS (Smart Communities as-a-Service). Це сервіс, який працює на базі хмарних технологій, являється сервісом аналізу відеопотоку з камер та забезпечує сповіщення користувачів системи для протидії злочинам.

Переваги системи DeepVision AI:

— не потребує власного серверу, всі данні обчислюються завдяки обробці в хмарі;

— система розпізнає не тільки об’єкти, а ще й аналізує злочинні дії в кадрі;

— працює з будь-яким вендором обладнання;

— система розроблена під працездатність в умовах великих населених пунктів та міст з великою кількістю об’єктів в кадрі.

Недоліки системи DeepVision AI:

— система не може бути розгорнутою локально;

— система потребує придбання підписки на сервіс з визначеним терміном, і оплатою з періодичністю в цей термін надалі;

— висока ціна системи [11].

#### 2.3.3.3 Система аналізу відеопотоку з застосуванням бібліотеки для створення нейронної мережі TensorFlow

TensorFlow це бібліотека для створення та навчання нейронних мереж від компанії Google. За допомогою цієї бібліотеки можна не тільки створити власну систему аналізу, а також можна використати вже існуючі нейронні мережі, які вже були натреновані для вирішення цієї задачі.

Переваги системи аналізу відеопотоку з застосуванням TensorFlow:

— гнучкість налаштувань;

— використання локально або як окремий сервіс;

— відкритий код;

— можливість застосувати вже готові рішення;

— немає прив’язки до вендору обладнання;

— велика кількість бібліотек з інтегрованою TensorFlow.

Недоліки системи аналізу відеопотоку з застосуванням TensorFlow:

— високі вимоги до ресурсів серверу, де буде розташована нейрона мережа;

— для забезпечення високого рівня продуктивності сервер повинен мати окремий модуль GPU з вбудованими тензорними ядрами;

— нові нейронні мережі повинні пройти навчання на заготовленій базі файлів [12].

#### 2.3.3.4 Висновки порівняльного аналізу систем аналізу відеопотоку

Проаналізувавши наявні системи аналізу відеопотоку та врахувавши усі переваги та недоліки, було зроблено висновок, що під задачу розробки власної системи відеоспостереження з модулем аналізу відеопотоку доцільно

використати локальну нейронну мережу-аналізатор на базі TensorFlow, що дозволить розгорнути модуль аналізу локально, відмовившись від плати за сторонні сервіси. Також це надасть можливість застосувати систему з будь-яким вендором обладнання та гнучко підлаштовувати нейронну мережу під задачу та під специфіку роботи відео обладнання.

### 2.3.3 Загальна модель системи відеоспостереження

Сучасні системи відеоспостереження повинні забезпечувати розпізнавання зловмисників та сповіщення про їх присутність.

Математична модель – це наближений опис довільного класу явищ зовнішнього світу, поданий за допомогою математичної символіки [13]. Метою створення математичної моделі є розпізнавання зловмисників з використанням відеоспостереження та інформування користувача через телеграм бот. Об’єктом моделювання є відеокамера на даху будинка.

Перед тим як почати синтез математичної моделі системи відеоспостереження, необхідно ознайомитись з процесами системи. Процес відеоспостереження складається з двох частин. Перша частина – апаратна зйомка з камери відеоспостереження, друга – програмна обробка відеосигналу, та аналіз отриманої інформації. Для апаратної зйомки використовується камера, яка знімає та передає відеопотік. Після цього сервер аналізує отриману інформацію для прийняття рішень, чи помічен злодій, чи треба оповістити про це користувача.

Щоб вирішити цю задачу було наведено формулу 2.1, яка містить в собі наступні зміні:

— Сзг – повна множина отриманих даних з камери відеоспостереження;

— Снт – підмножина, значення якої не дають спрацювати тригеру;

— Ст – підмножина, значення якої дають спрацювати тригеру.

$$Сзг = \{Снт , Ст\} \qquad (2.1)$$

Під час зйомки, події в кадрі випадкові, та процес має непередбачуваний характер, для моделювання системи має сенс використати теорію. масового обслуговування та теорію ймовірності [14].

Щоб вирішити цю задачу за допомогою теоретичного інструментарію теорії ймовірності, можна використати ланцюг Маркова – це концепція, розроблена в рамках теорії ймовірності та статистики, яка встановлює сильну залежність між подією та іншою попередньою подією. Основна його корисність – аналіз поведінки стохастичних процесів. Ланцюг можна уявити в вигляді графа. Вершини цього графу ілюструють стани. Ребра – інтенсивність переходу між станами. За допомогою ланцюга Маркова можливо порахувати ймовірність кожного стану [15].

Вхідними даними для моделювання будуть стани досліджуваного кадру з відеокамери, які наведені на рисунку 2.2 у вигляді графу.

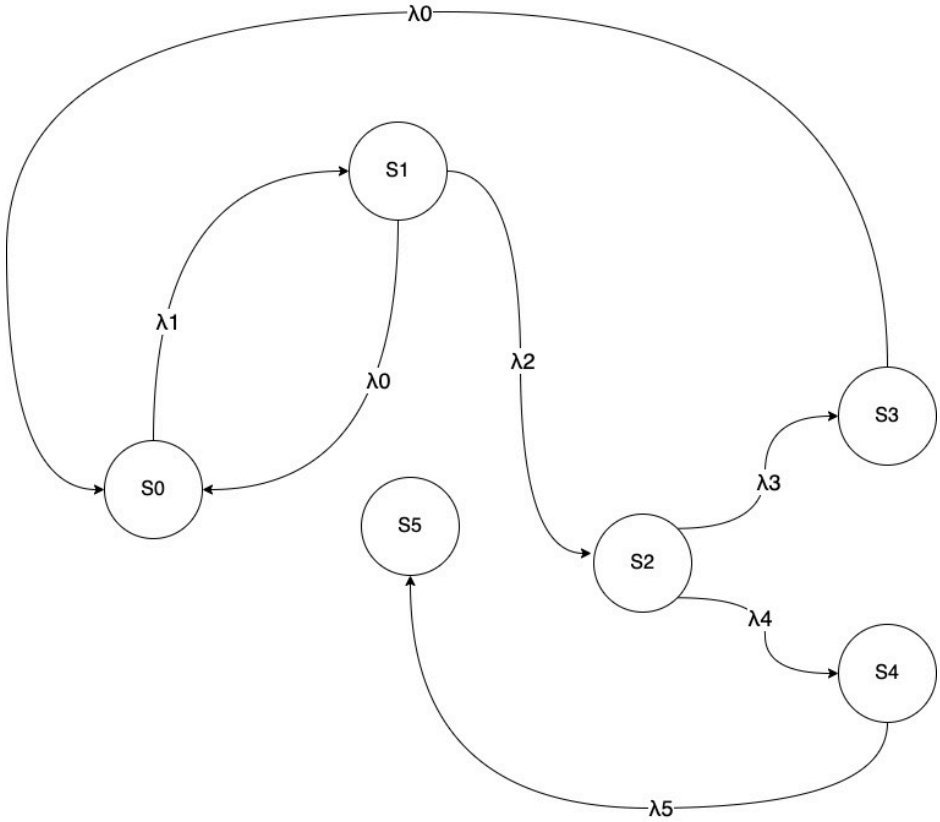


Рисунок 2.2 – Граф подій під час зйомки камери

Нижче наведені стани кадра, який досліджуємо:

— S0 – відсутність руху;

— S1 – помічен рух;

— S2 – помічен об’єкт;

— S3 – об’єкт не людина;

— S4 – об’єкт людина;

— S5 – сповіщення користувача.

Щоб скласти математичну модель треба розглянути кожну подію в кадрі як випадковий процес, у якого може бути тільки обмежена кількість станів. Кожна подія в математичній моделі відбувається лише один раз, з чого слідує, що модель одинарна.

Ймовірність *i*-го стану визначається, як ймовірність знаходження у стані $S_i$. Інтенсивність буде залежати від частоти кадрів з камери та від налаштувань програми, враховуючи інтервал, через який проміжок кадрів буде пропускати аналітичний модуль системи. Інтенсивності запиту визначає змінна $\lambda_i$, яка показує інтенсивність перевірки *i*-го досліджуваного кадру камери.

Система аналізу буде проходити по графу, починаючи з стану $S_0$, що є станом відсутності руху. При виявленні руху або об’єкту відбувається перехід у стан $S_i$ враховуючи інтенсивність запиту $\lambda_i$. Після того, як система потрапила до стану $S_5$ користувач отримає повідомлення про виявлений злочин. Можемо скласти математичну модель системи аналізу об’єктів. Математична модель зображена в формулі 2.2.

$$\begin{cases} \frac{dp_0}{dt} = -\ p_1 *\ \lambda_1 * p_1 * \lambda_0 *\ p_3 *\ \lambda_0, \\ \frac{dp_1}{dt} = p_0 * \lambda_1 -\ p_0 * \lambda_0 -\ p_2 *\ \lambda_2, \\ \frac{dp_2}{dt} = p_1 * \lambda_2 -\ p_3 *\ \lambda_3 -\ p_4 *\ \lambda_4, \\ \frac{dp_3}{dt} = p_2 * \lambda_3 +\ p_0 *\ \lambda_0, \\ \frac{dp_4}{dt} = p_2 * \lambda_4 -\ p_6 *\ \lambda_6, \\ \frac{dp_5}{dt} = p_4 * \lambda_6. \end{cases} \quad (2.2)$$

### 2.3.4 Створення моделі КС

Перед розробкою ПЗ та проведенням експериментів для системи розпізнавання зловмисників з використанням відеоспостереження необхідно розробити модель розпізнавання образів в КС, з клієнт серверною архітектурою.

Постановка задачі для моделювання: спираючись на попередньо отриману інформацію про існуючі системи моніторингу, необхідно створити модель моніторингу КС.

При плануванні моделі КС було виділено наступні елементи:

— вхід системи – початок аналізу системою;

— інтервал перевірки – перевірка виконується в проміжок кадрів в залежності від налаштувань системи;

— розпізнавання образів – перевірка виконується за алгоритмом, який зазначений на графі, зображеному на рисунку 2.2;

— сповіщення користувача – якщо перевірка знайшла об’єкт виконується функція сповіщення користувача;

— вихід з системи – кінець аналізу системою.

Модель комп’ютерної системи позначено на рисунку 2.3.

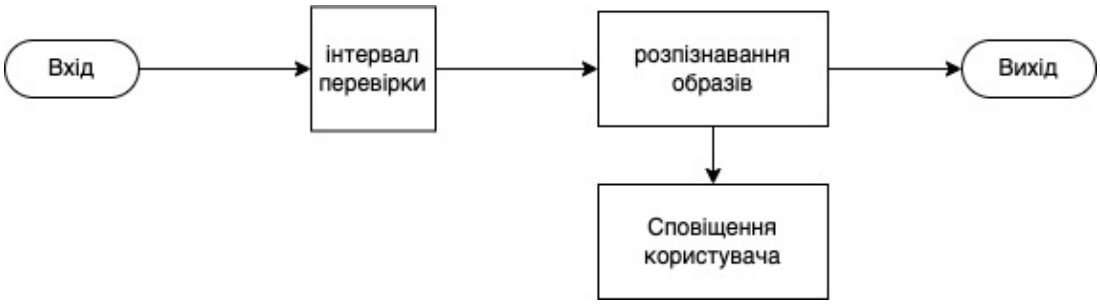


Рисунок 2.3 – модель розпізнавання образів в КС

### 2.4 Обґрунтування і вибір методів експериментальних дослідження

Задачею експериментатора буде виміряти час, за який система встигла розпізнати об’єкт, експериментатор буде вимірювати час між появою об’єкту в кадрі та реакцію системи на об’єкт. Похибка дослідження буде залежати як

від експериментатора, так як виміряний час може залежати від швидкості реакції людського ока, так і від системи аналізу, що може видавати нестабільну швидкість розпізнавання, через різну завантаженість системи, різну швидкість виконання програмного коду та може залежати від швидкості передачі даних у мережі. Враховуючі це, для проведення експериментальних досліджень було обрано метод середнього арифметичного для досягнення більш точних результатів.

### 2.5 Висновки теоретичної частини

У цьому розділі проаналізовано структуру об’єкту дослідження, та знайдено сам об’єкт дослідження, що дало можливість виявити найбільш придатні методи дослідження для побудови системи відеоспостереження. Найбільш придатними виявилися методи аналізу та синтезу, порівняння та математичного моделювання. Проведено аналіз та порівняння систем відеоспостереження та систем аналізу відеопотоку, які присутні на ринку, вони були оцінені за різними критеріями для виявлення переваг, недоліків та синтезу рішення під задачу зі збалансованими та найбільш актуальними характеристиками. Це дало можливість засотувати метод математичного моделювання та розробити модель комп’ютерної системи, для подальшої розробки власної системи відеоспостереження. Було обрано метод проведення експериментальних досліджень.

# 3 СИНТЕЗ СИСТЕМИ

## 3.1 Цілі впровадження системи

Перед впровадженням системи відеонагляду задано наступні цілі:

— забезпечення відеонагляду за об’єктом;

— надання користувачу доступу до перегляду камер;

— сповіщення користувача при виявлені камерою небезпеки;

— інтеграція модулю аналізу до системи відеонагляду.

## 3.2 Формулювання технічних вимог до системи відеоспостереження

### 3.2.1. Вимоги до реалізації системи

Система повинна складатись з таких частин:

— панель адміністратора;

— інтерфейс користувача;

— програмна частина камери;

— сервіс обробки відеозображення з впровадженим аналітичним модулем та модулем сповіщенням користувача.

Серверна частина системи має бути побудована на мікросервісній архитектурі програмного забеспечення, з використанням Docker контейнерів, завдяки чому сервера програми можна буде масштабувати, при збільшенні попиту користувачів, також це надасть можливість перенести систему в хмару, таку як AWS або Azure.

Серверна частина повинна бути написана мовою JavaScript з застосуванням платформи NodeJS, для ефективного використання ресурсів серверу, завдяки своїй асинхронній структурі обробки запитів.

Клієнтська частина повинна бути написана з використанням бібліотеки React, що покращить користувацький досвід та дозволить писати та тестувати окремі компоненти користувацького інтерфейсу.

Модуль аналізу об’єктів в кадрі повинен бути написаний з використанням бібліотеки TensorFlow для побудови та тренування нейромереж.

Модуль сповіщень повинен використовувати API Telegram для сповіщення користувача про небезпеку.

### 3.2.2 Вимоги до функцій виконуваних системою

Система відеоспостереження має виконувати такі функції:

— мати мінімалістичний графічний інтерфейс користувача, який дозволяє авторизованому користувачу переглядати трансляцію з камер;
— адміністратор повинен мати можливість додавати нові камери;
— забезпечувати можливістю розмежувати доступи до інформації для різних користувачей системи;
— забезпечувати новітні можливості аналізу об’єктів в кадрі;
— відеопоток повинен доходити від камери до сервера, та від сервера до користувача в шифрованому вигляді.

### 3.2.3 Вимоги до видів забезпечення

Щоб реалізувати систему необхідно встановити таке програмне забезпечення як Raspbian для Raspberry Pi, Docker та Docker-compose для серверної частини, яка може базуватись на будь-якій UNIX-подібній OS, таких як Debian, Ubuntu, CentOS і так далі. За допомогою Docker система може бути запущена на таких операційних системах як Windows, Linux та MacOS, які підтримують Docker або запущено в хмарах, таких як AWS та Azure.

Сервер системи програмного забезпечення повинен відповідати таким вимогам:

- 8-ми ядерний процесор;
- об’єм диска 1T;
- 64 ГБ RAM.

Сервер повинен мати встановлену одну з підтримуваних UNIX-подібних ОС з наступними версіями:

— CentOS: 7, 8;

— Debian: 10, 11;

— Ubuntu Server: 20.04 LTS, 21.04.

### 3.2.4 Вимоги до захисту інформації

Захист іформації в КС повинен базуватись на використанні публічного та приватного ключа для передачі відеопотоку. Ключ повинен базуватись на алгоритмі rsa. Довжина rsa-ключа має бути, не меншою ніж 4096 бітів, як рекомендовано National Security Agency.

Для користувачів системи повина використовуватись корпоративна пошта та випадково згенеровані паролі завдовжки 12 символів.

Пароль адміністратора системи повинен бути завдовжки 15 символів, та повинен бути згенерований випадково.

Для забезпечення максимального ступеню захисту, порт ssh серверу повинен бути перенесений з 22 на 2543, для того, щоб мережевим сканерам було важче знайти порт ssh та зрозуміти його призначення. Для захисту серверу вхідні порти, окрім портів 80, 443, 2543, 554 TCP та 554 UDP повинні бути зачинені. Замість паролю ssh повинен використовуватись ключ шифрування аналогічний ключу шифрування відеопотокую.

Сервер відеоспостереження повинен щоденно бекапитись для забеспечення швидкого відновлення при збої системи. Ці резервні повинні зберігатися протягом 1 місяця. Для системи бекапу було обрано ПЗ VEEM Backup. Доступ до входу в систему бекапу має тільки суперкористувач операційної системи. Резервні данні повинні бути захищенні від сторонніх особ.

### 3.2.5 Вимоги до ергономіки системи

Сервер повинен знаходитися в спеціальному приміщенні в будівлі, де знаходиться IT-відділ. Тільки системні адміністратори повинні мати доступ до цієї кімнати. Крім того, ці компоненти повинні знаходитись місцях зі зручним доступом для проведення планового обслуговування.

### 3.2.6 Розробка схеми функціональної структури

Архітектурним принципом побудови системи було обрано мікросервісну архітектуру, що дозволить розробляти та тестувати програмне забезпечення**,** як окремі програми, які мають вхідні та вихідні данні. Це дозволить декільком командам розробників займатись розробкою цього програмного забезпечення, також такий підхід гарно підходить під сучасні реалії побудови програмного забезпечення, так як дає гнучкість в виборі програмних компонентів системи, дозволяє не прив’язуватись до конкретної мови програмування та дозволяє швидко переписувати модулі на більш сучасні технології з плином часу, розвитком технологічного процесу та збільшенням обсягів обробки даних. Також, як перевагу можна виділити ізоляцію компонентної бази коду, що позитивно впливає на можливість тестування програмного коду та забезпечує меншу кількість багів в програмному забезпеченні [16].

Компоненти застосунку (рисунок 2.1):

1) база даних – компонент, який є основним сховищем для даних користувача та камери;

2) панель адміністратора – цей компонент являє собою адміністраторську панель, де адміністратор може редагувати налаштування, доступи та користувачів системи. Панель адміністратора також виконує роль центрального API сервера, який виконує роль прошарки між базою даних та іншими компонентами системи;

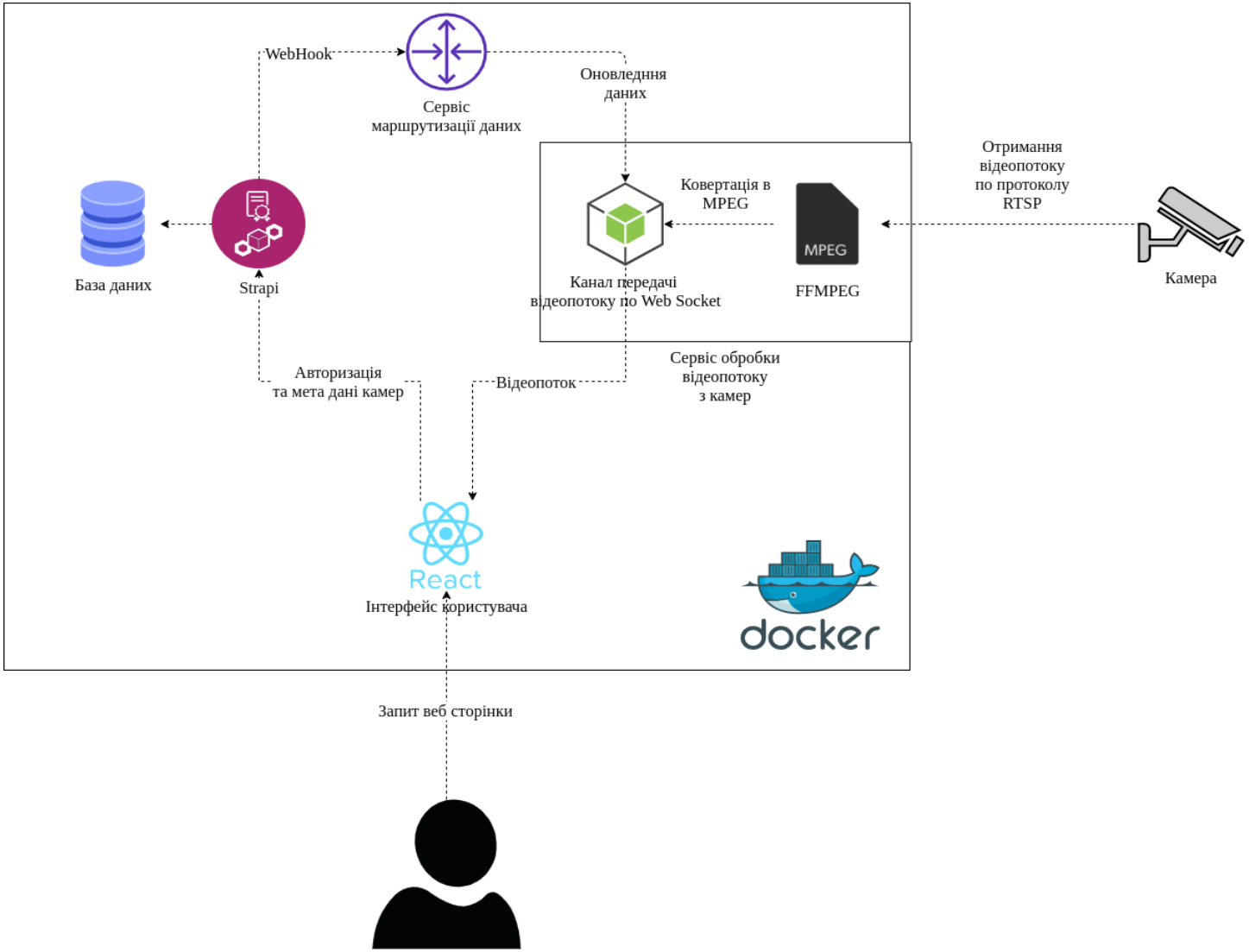


Рисунок 3.1 – Схема функціональної структури

3) обробник відеопотоку з камер – ця служба під'єднується до відеокамер та обробляє відеопоток для користувача, задача цього сервісу програмна обробка зображення камери, тут відбувається аналіз та реєстрація об'єктів в кадрі. При виявленні небезпеки цей сервіс оповіщає користувача за допомогою телеграм бота, надсилаючи час події та кадр на якому була помічена загроза;

4) інтерфейс користувача – ця служба призначена для того, щоб відтворювати інтерфейс для користувача системи, головна мета цього сервісу авторизація, та перегляд відеопотоку з камер, для перевірки камер, та слідкування за об'єктом в кадрі. Цей сервіс зв'язується з сервісами панелі адміністратора та обробником відеопотоку для отримання необхідної інформації, такої як інформація про користувача, дозволені для перегляду камери та відеопоток з цих камер.

Структура БД зображена на рисунку 3.2.

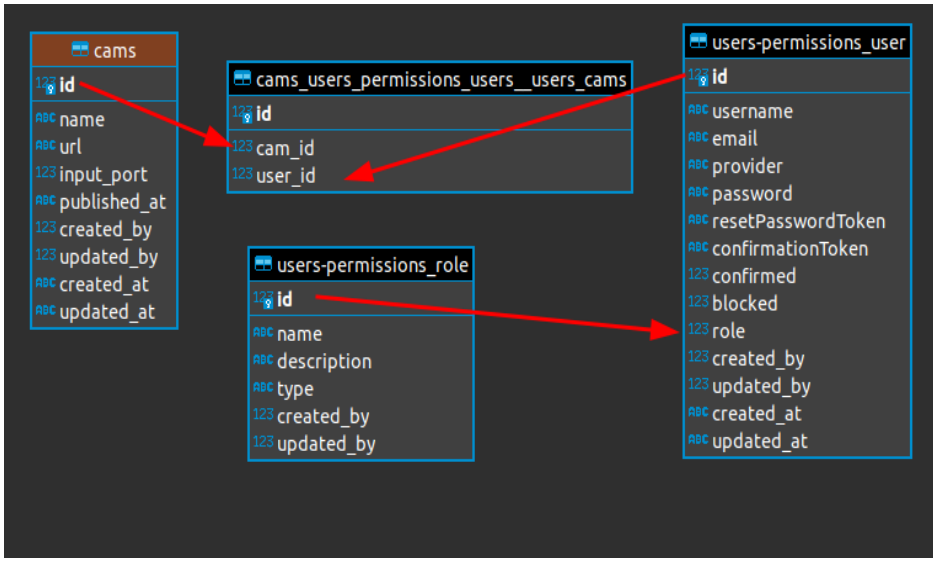


Рисунок 3.2 – Структура БД

Вона містить в собі такі таблиці:

1) користувачі – ця таблиця потрібна для зберігання даних авторизації користувача системи, а також ID користувача в телеграм боті для зв'язку з ним через Telegram API [17];

2) роль – користувачі системи мають за собою роль в системі для забезпечення поділу прав користувачів, такі як адміністратор та користувач без доступу до адмін панелі;

3) камери – в цій таблиці знаходяться камери, які можуть мати назву та посилання на трансляцію камери за обраним протоколом;

4) камери-користувачі – з'єднання користувачів і камер за принципом «багато-до-багатьох».

Оскільки архітектура бази не містить багато складних зв'язків, була обрана БД SQLite, яка забезпечить помірний вплив на ресурси сервера та достатню швидкість для обробки запитів SQL [18].

### 3.3 Вибір та обгрунтування застосування апаратних засобів

Як зазначено в вимогах до системи, для IP-відеоспостереження необхідно використовувати мікрокомп'ютер Raspberry Pi. Серед актуальних модельних рядів можна виділити моделі, наведені в таблиці 3.1.

Таблиця 3.1 – Характеристика модельного ряду Raspberry Pi

| Модель | Рік випуску | CPU | GPU | Тактова частота процесора | Оперативна пам'ять |
|---|---|---|---|---|---|
| Pi 3 A+ | 2018 | Broadcom BCM2837B0 | Video Core IV 400 МГц | 4x1.4 ГГц | 512 Мб |
| Pi 3 B | 2016 | Broadcom BCM2837 | Video Core IV 400 МГц | 4x1.2 ГГц | 1 Гб |
| Pi 3 B+ | 2018 | Broadcom BCM2837B0 | Video Core IV 400 МГц | 4x1.4 ГГц | 1 Гб |
| Pi 4 B | 2019 | Broadcom BCM2711 | Video Core VI | 4x1.5 ГГц | До 8 Гб |

Продовження таблиці 3.1

| Модель | Рік випуску | CPU | GPU | Тактова частота процесора | Оперативна пам’ять |
|---|---|---|---|---|---|
| Pi Zero W | 2017 | Broadcom BCM2835 | Video Core IV | 1x1 ГГц | 512 Мб |

Найслабшим з лінійки Raspberry Pi є Pi Zero, який має одноядерний процесор та 512 Мб оперативної пам’яті. Ця модель Raspberry Pi не підійде для передачі потового відеосигналу, тому що цей процес вимагає багато ресурсів з боку процесора і якщо чип не буде встигати з обробкою відеоінформації, то зображення, яке буде потрапляти на сервер буде не повним, або трансляція буде час від часу втрачатись.

Raspberry Pi лінійки 3 можна вважати ще актуальними, але компоненти системи, на яких вони побудовані вже застаріли, наприклад в моделях цій лінійки відсутній порт USB-C, який зараз являється стандартом, замість нього лінійка має порт Micro-USB. Лінійка має максимальну швидкість передачі даних у мережі 100 Мбіт/с. Також моделі лінійки Pi 3 мають обмеження у 1Гб оперативної пам’яті, що не дозволяє конкурувати з аналогічними моделями мікрокомп’ютерів.

Найновішою моделлю Raspberry Pi є модель 4 B, яка дозволяє гнучко налаштовувати конфігурацію оперативної пам’яті, має сучасний вхід під живлення USB-C, найновіший процесор та відеоядро, а також Ethernet порт зі швидкістю в 1Гбіт [19].

Для системи було обрано використовувати Raspberry Pi 4 B, з вбудованими 8GB оперативної пам’яті, зображений на рисунку 3.3, для забезпечення більшої швидкості передачі відеосигналу з камери.

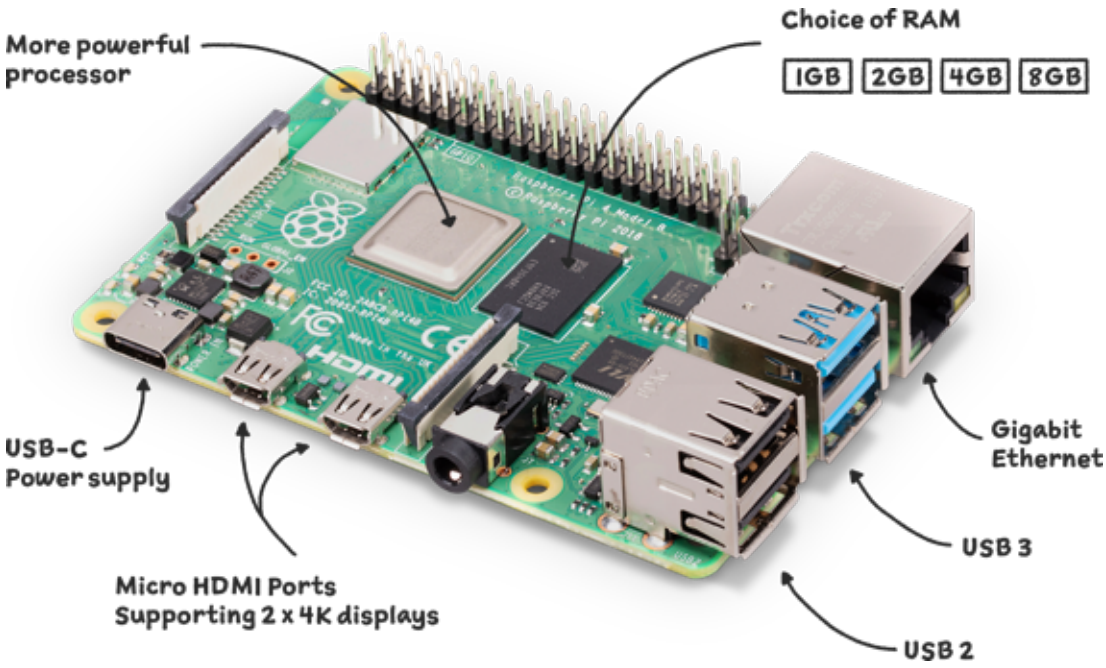


Рисунок 3.3 – Raspberry Pi 4 B

Для того, щоб система відеоспостереження функціонувала потрібно обрати камеру. Для Raspberry Pi існує багато модулів, які можна використати для зйомки потокового відео, але потрібна камера, яка буде знімати з високою роздільною здатністю, для того щоб знятий матеріал підлягав аналізу. Як модуль камери було обрано Raspberry Pi High Quality Camera, головною перевагою якого є висока роздільна здатність у 12,3 Мп. Також в комплектації цієї камери присутні об’єктиви 16мм та 6мм, що забезпечить правильну фокусировку кадру на різній відстані від об’єкту.

Характеристики камери:

— розмір зображення: 12.3MP, 4056 (Висота) × 3040 (Ширина);

— вихід: RAW12 / 10 / 8, COMP8;

— матриця: Sony IMX477R [20].

В якості джерела живлення для мікрокомп’ютера Raspberry Pi було обрано використовувати стандарт PoE 802.3af. Цей метод підключення користується широкою популярністю серед камер відеоспостереження, так як потребує всього один кабель для підключення інтернет каналу та каналу живлення. Для забезпечення підтримки PoE було обрано плату розширення Raspberry Pi Compute Module 4 PoE UPS Base Board [21].

Цей модуль розширення йде зі спеціальним корпусом для забезпечення захисту, має роз’єми під акумулятори, для забезпечення безперебійного живлення камери в ситуації, коли живлення відсутнє та має спеціальні виходи для підключення камери.

### 3.3.1 Вибір та характеристика мережевого обладнання

Для забезпечення роботи мережі та реалізації PoE живлення камер, було обрано маршрутизатор RB5009UPr+S+IN, який має 8 PoE портів, які можуть виконувати функції, як PoE виходів, так і входів, цей маршрутизатор буде в нагоді, для застосування передачі потокового відео з камер, так як має 7 гігабітних портів, та один 2.5ГБ порт. Також цей маршрутизатор має достатньо сильний процесор, що дозволить обробляти велику кількість запитів до мережі.

Характеристики RB5009UPr+S+IN:

— макс. пропускна спроможність: 12 Гбіт/с;

— RAM: 1Гб;

— Storage: 1Гб;

— Процесор: ARM 64 bit, 4 ядра, 350-1400 МГц;

— Ethernet: 7 шт 1 ГБ, 1 шт 2.5 ГБ;

— SFP: 1 шт 10 ГБ;

— PoE: 802.3af / at [22].

Для того, щоб забезпечити безперебійну роботу в умовах відсутності зв’язку з глобальною мережею інтернет через кабель, було вирішено використати термінал Starlink, це супутникова антена, яка повинна бути закріплена на столбі, або на даху буднику, для з’єднання з супутником, для отримання доступу до супутникового інтернету. Антена повинна бути закріплена в напрямку півночі для кращого рівня сигналу. Термінал Starlink можна використовувати як в режимі роздачі WiFi, так і для передачі даних через мережевий кабелі, але для другого режиму роботи потрібно придбати

спеціальний адаптер, так як термінал має пропрієтарний роз'єм, для того щоб з'єднати термінал з мережевим обладнанням через Ethernet. Термінал Starlink забезпечить системі резервний канал зв'язку до 100 МБ/с [23].

### 3.3.2 Вибір та характеристика серверного обладнання

Для того, щоб обробляти відеопоток з камер нагляду, тримати запущеними сервіси аналізу з використанням нейронних мереж та зберігати потокове відео, характеристики серверу повинні виконувати такі вимоги:

— об'єм диска 1 Тб;

— оперативна пам'ять 64 ГБ.

— тактова частота процесора: 2.4 ГГц, або більше;

— кількість ядер процесора: 4, або більше;

— відео ядро з вбудованими тензорними ядрами.

Щоб сервер міг працювати з окремими відеядром, потрібно підібрати серверне обладнання таким чином, щоб в материнській платі був роз'єм PCI express версії 4.0.

Як основу для сервера було обрано використати Dell Precision T5820, це корпус Midi-Tower з материнською платою Dell Precision T5820, яка маж роз'єм під процесор Intel Xeon W, тип пам'яті DDR4 з 8-ю слотами, наявні два роз'єми PCI express версії 4.0, мережева карта може забезпечити швидкість з'єднання до 1 Гбіт/с. В якості процесора, для конфігурації серверу було обрано Intel Xeon W-2175 SR3W2 з максимальною тактовою частотою у 4,3 ГГц. В якості оперативна пам'яті буде використовуватись Samsung DDR4-2400 дві планки по 32 Гб. Для забезпечення цілісності інформації на сервері було прийнято рішення поставити чотири жорсткі диски WD10PURZ об'ємом 1 ТБ, які будуть об'єднані контролером RAID Dell PERC H330+ у 5-й рейд масив [24].

Для того, щоб сервер міг забезпечувати потреби нейронної мережі, треба обрати відеочип з вбудованими тензорними ядрами, для цієї задачі ідеальним варіантом буде лінійка від компанії Nvidia, Nvidia A2 Tensor Core, цей відеочип має 1280 вбудованих ядер CUDA та 16 Гб GDDR6 пам'яті [25].

Для надійності та цілісності інформації, яка зберігається на сервері було прийнято рішення поставити ДБЖ, було обрано включити в необхідне обладнання APC Smart-UPS RM 3000VA 2U LCD, який забезпечить максимально 3000 Ват потужності. Цей ДМЖ забезпечить сервер додатковими 30 хвилинами при максимальному навантажені, для роботи за завершення процесів, збереження даних [26].

Таблиця 3.2 – Специфікація обладнання

| Позиція | Найменування і технічна характеристика | Тип, марка | Одиниці виміру | Кількість |
|---|---|---|---|---|
| 1 | Raspberry Pi 4 B | мікрокомп’ютер, raspberry pi | шт | 28 |
| 2 | Raspberry Pi High Quality Camera | камера, raspberry pi | шт | 28 |
| 3 | Raspberry Pi Compute Module 4 PoE UPS Base Board | плата розширення poe, raspberry pi | шт | 28 |
| 3 | маршрутизатор Mikrotik RB5009UPr+S+IN | маршрутизатор, mikrotik | шт | 4 |
| 6 | Dell Precision T5820 | сервер, dell | шт | 1 |
| 7 | Nvidia A2 Tensor Core | відеокарта, nvidia | шт | 1 |
| 8 | Intel Xeon W-2175 SR3W2 | процесор, intel xeon | шт | 1 |
| 9 | Samsung DDR4-2400 | оперативна пам’ять, samsung | шт | 2 |
| 10 | RAID Dell PERC H330+ | raid, dell | шт | 1 |
| 11 | WD10PURZ | жорсткий диск, WD | шт | 4 |

Продовження таблиці 3.2

| Позиція | Найменування і технічна характеристика | Тип, марка | Одиниці виміру | Кількість |
|---|---|---|---|---|
| 12 | APC Smart-UPS RM 3000VA 2U LCD | дбж, apc | шт | 2 |
| 13 | Термінал Starlink | супутникова антена, starlink | шт | 4 |
| 14 | Адаптер Starilink to ethernet | адаптер, starlink | шт | 4 |

### 3.4 Синтез структурної схеми системи за заданими показниками

Після аналізу інформації про об’єкт та підбірки апаратного обладнання, було розроблено результуючу структурну схему системи відеоспостереження, яка зображення на рисунку 3.11.

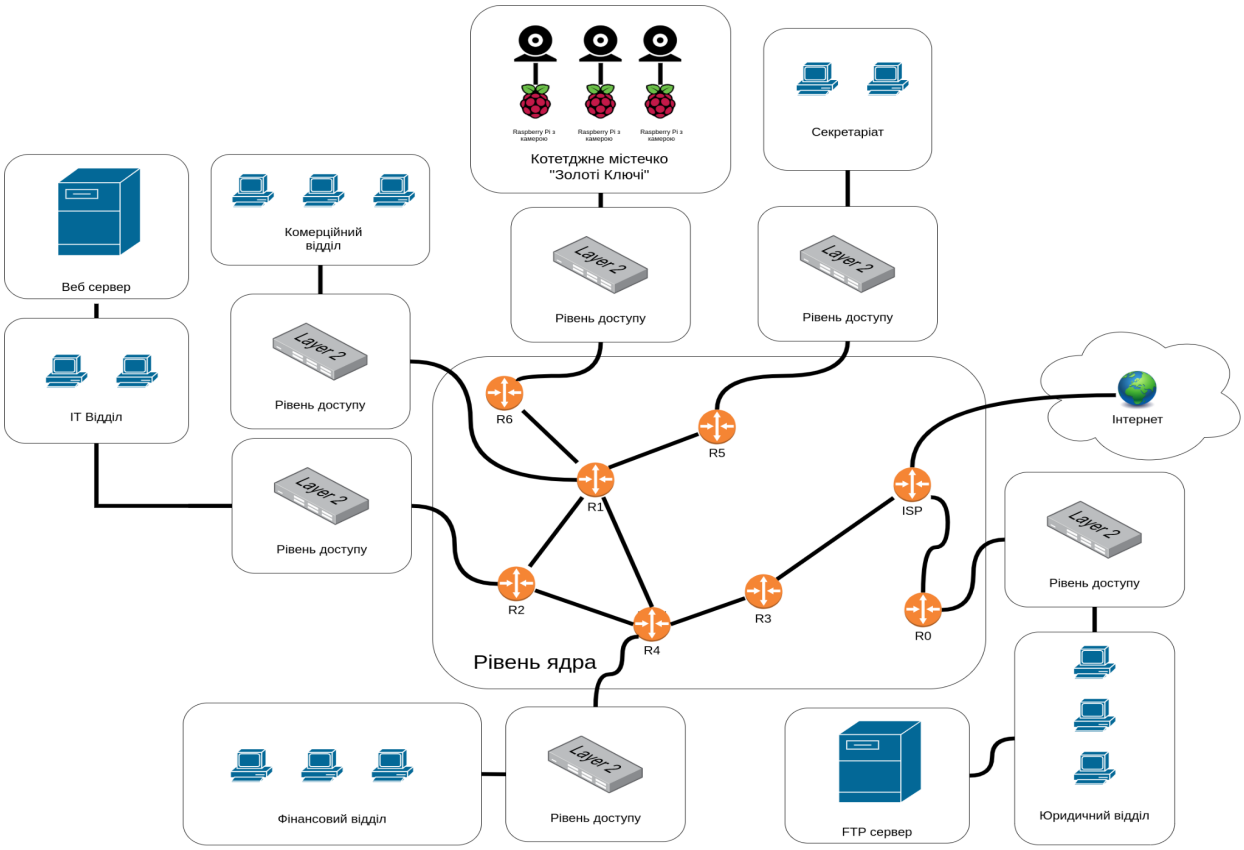


Рисунок 3.11– Схема функціональної структури “АН ЗОЛОТІ КЛЮЧІ”

### 3.5 Висновки до розділу

У цьому розділі розроблено вимоги до впровадження системи відеоспостереження, необхідні для надійної та стабільної роботи. Було розроблено схему функціональної структури, необхідної для побудови системи відеоспостереження, і структурну схему, що показує зв’язок між камерами та веб-сервером. Комп’ютерні компоненти системи відеоспостереження підібрані та включені до специфікації обладнання.

# 4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ ВІДЕОСПОСТЕРЕЖЕННЯ АН ЗОЛОТІ КЛЮЧІ

## 4.1 Призначення й область застосування програмного забезпечення

Призначеня ПЗ – забезпечення АН Золоті Ключі системою відеонагляду та системою аналізу та сповіщень користучів та мешканців про небезпеку.

Серверне ПЗ повинно встановлюватись на веб-сервер, ця частина програмного комплексу відповідає за збір даних з камер та за надання інтерфейсів користувачам за HTTPS запитами.

Програмне забеспеченя камер повинно встановлюватись на мікрокомп’ютер Raspberry Pi, та повинно виконувати функції зйомки та обробки запитів від сервера, з наданням відеопотоку.

Область застосування – АН “Золоті ключі”.

## 4.2 Обґрунтування технічних характеристик програм

Панель адміністратора та інтерфейс користувача повинні бути реалізовані як два окремі сервіси та запускатись за допомогою інструментарія для управління ізольованими Linux-контейнерами Docker.

Для реалізації додатку, як для серверної, так і для клієнтської частини, була обрана мова програмування JavaScript з використанням TypeScript та компілятора Babel. Мова програмування JavaScript може виконуватися як у браузері користувача, що ідеально підходить для веб-додатків, так і на сервері, який обробляє запити клієнтів клієнта.

Клієнтська частина реалізована з використанням бібліотеки React. React постає як сучасна бібліотека для розробки адаптивних веб-додатків, ця бібліотека дозволяє реалізувати реактивний підхід розробки програмного забезпечення для веб інтерфейсу.

Компонент серверу буде реалізован за допомогою NodeJS. NodeJS це платформа яка виконує код JavaScript на сервері та є дуже ефективним способом

реалізації мережевих сервісів, які мають збалансований характер поміж використанням ресурсів та швидкістю обробки запитів клієнта [27].

Для реалізації панелі адміністратора застосується Strapi — це CMS система, яка застосовує запити API для доставки вмісту та створює веб-портал для адміністратора, на основі заданої структури бази даних [28].

Сервіс обробки відеопотоку з камер буде реалізована за допомогою локального сервісу ffmpeg. Задачею цього сервісу є обробка відеопотоку за допомогою протоколу RTSP і перетворення його у формат mpeg, для того щоб його можна було відтворювати на веб-сторінці користувача, за допомогою декодера JsMpeg і відображати в елементі сторінки canvas [29].

Для впровадження аналітичних можливостей розпізнавання об’єктів, буде використана бібліотека face-recognition, яка використовує Tensorflow API як модуль машинного навчання. Tensorflow це відкрита бібліотека, яка розроблена компанією Google, яка має широкий функціональний спектр для побудови та тренування нейронних мереж і має достатню основу для побудови систем з розпізнавання образів та обличь.

## 4.3 Опис розробленої програми

### 4.3.1 Загальні відомості

Увесь додаток можна описати як клієнт-серверне програмне забеспечення. Обидві частини були написані на JavaScript, але використовувалися додаткові зовнішні бібліотеки. Клієнтська частина веб-додатку була створена за допомогою бібліотеки React, сервер – за допомогою NodeJS [30].

Мета програми – трансляція та передача відеозображення користувачу. Аналіз отриманих кадрів, виявлення загроз та сповіщення користувачів про них.

### 4.3.2 Функціональне призначення

Функції, які виконує додаток:

— додавати користувачів до системи;

— додати в систему камери;

— надати користувачам доступ до камер;

— вхід користувача по логіну та паролю

— надання доступу до трансляції камер користувачу;

— зберегання на FTP сервері записів з камер;

— можливість запису камери;

— розпізнавання об’єктів, загроз;

— сповіщення користувача про загрозу;

### 4.3.3 Опис логічної структури програми

На рисунках 4.1, 4.2 та 4.3 зображена схема функціонування серверної частини обробки запитів клієнта, та передача відеопотоку.


Рисунок 4.1 – Схема алгоритму обробки запитів клієнта (частина перша)

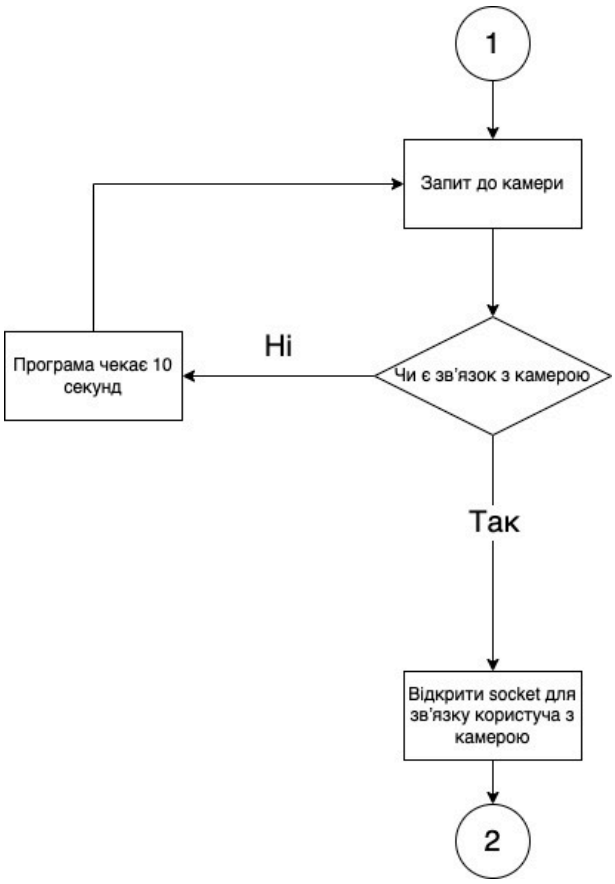


Рисунок 4.2 – Схема алгоритму обробки запитів клієнта (частина друга)

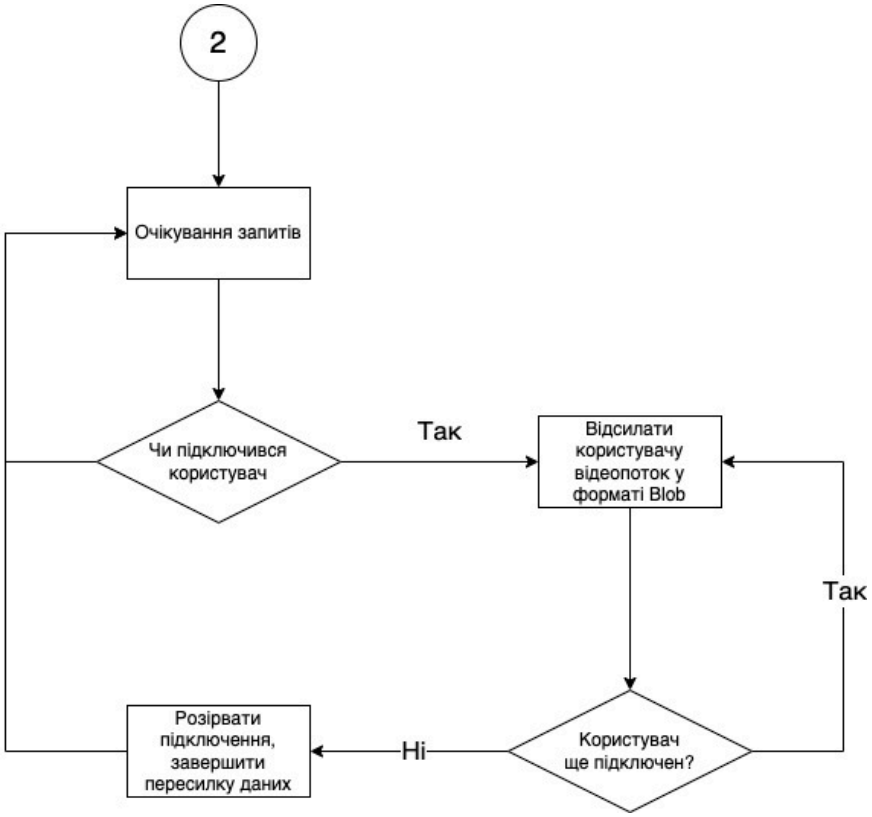


Рисунок 4.3 – Схема алгоритму обробки запитів клієнта (частина третя)

На рисунку 4.3 зображено схему функціонування аналітичного модулю для обробки та аналізу зображень з подальшим повідомленням користувача про загрозу.

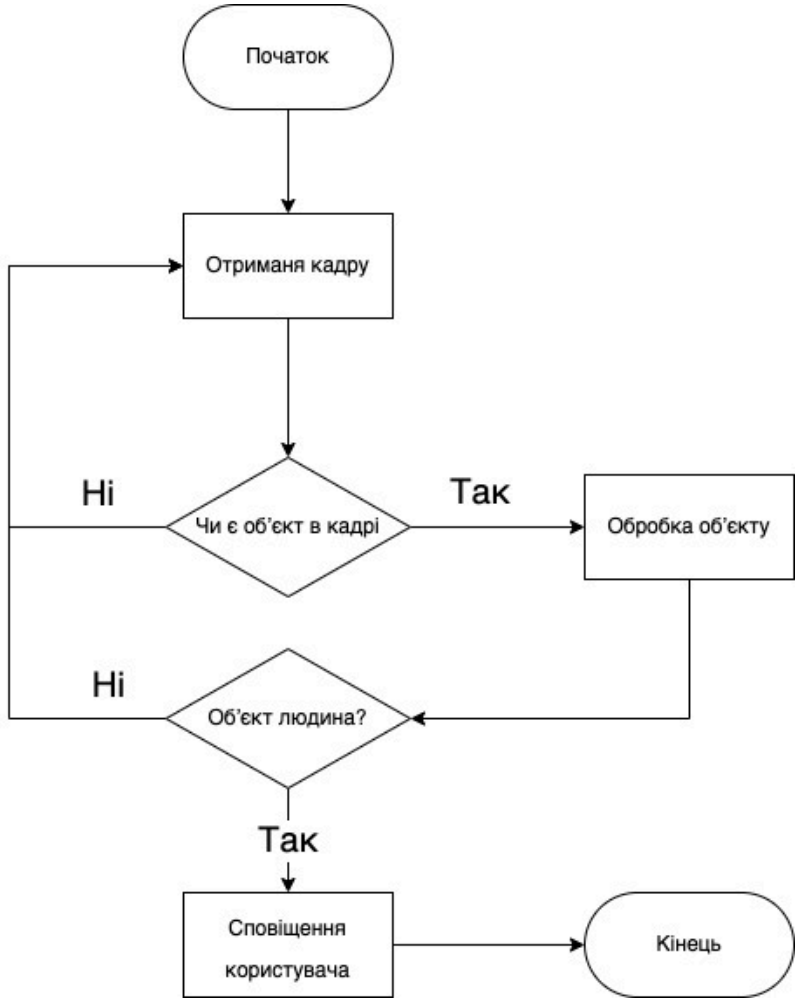


Рисунок 4.4 – Схема алгоритму аналітичного модулю

#### 4.3.3.1 Виклик і завантаження

Для запуску програмного забезпечення необхідно встановити зовнішні пакети залежностей. Для проекту JavaScript необхідно виконати команду “npm install” для встановлення необхідних залежностей, які зформовані в файлі package.json.

Для запуску програмного забезпечення використовується Docker. Запуск ПЗ відбувається за допомогою команди «docker-compose up». При

першому запуску ПЗ команда встановить усі необхідні образи Docker контейнерів та запустить усі компоненти програмного коду.

### 4.3.4 Використані технічні засоби

#### 4.3.4.1 Технічні засоби серверної частини

При ініціалізації програми спочатку завантажується служба панелі адміністрування, а також база даних. Після цього запускається служба обробки відеопотоку камер, яка використовує API для доступу до панелі адміністрування активних камер. Після отримання списку камер до кожної камери підключається служба обробки, завдяки програмі FFMPEG. Служба розгалужує програму FFMPEG і починає читати stdout.

При підключенні до камери FFMPEG намагається вивести дані в стандартний вихід, тоді як служба потокового відео зчитує дані та записує їх у файли, доступні через SFTP на сервері. Також сервіс ініціалізує канал передачі даних - WebSocket, який очікує підключення клієнтів.

Під час підключення до WebSocket клієнт надсилає JWT ключ, отриманий під час авторизації. Сервіс перевіряє його, якщо токен дійсний - починається передача відеопотоку.

Коли дані оновлюються в панелі адміністратора, активується WebHook, який надсилає нову інформацію до служби маршрутизації даних. Якщо камери оновлено, служба маршрутизації даних сповіщає службу обробки відеопотоків, яка порівнює та оновлює камери – включає нові відеопотоки, виключає видалені.

В сервіс з обробки відеопотоку встороєнний аналітичний модуль з підтримкою API Tensorflow, бібліотеки для роботи з нейронною мережою. Аналітичний модуль програми аналізує трансляцію з підключених камер та виконує перевірку кожного 4-го кадру. Спочатку аналізується рух об’єктів в кадрі, якщо рух помічен програма переходить в стан аналізу людей в кадрі, після того як в кадрі було помічено людину модуль відсилає зображення з камери користувачу, який був підкріплений до камери через Telegram бота.

#### 4.3.4.2 Технічні засоби користувацького інтерфейсу

Користувацький інтерфейс містить в собі три основних частини:

1) панель адміністратора;
2) інтерфейс перегляду камер;
3) телеграм бот.

Панель Адміністратора – набір сторінок, який динамічно генерується в залежності від структури бази даних, де знаходиться форми для налаштування системи та списки налаштувань для перегляду їх адміністратором. Панель адміністратора захищена за допомогою авторизації. Адміністратор системи може назначати певні повноваження для користувачів, в тому числі право на авторизацію та використання адмін панелі. (рисунок 3.1).

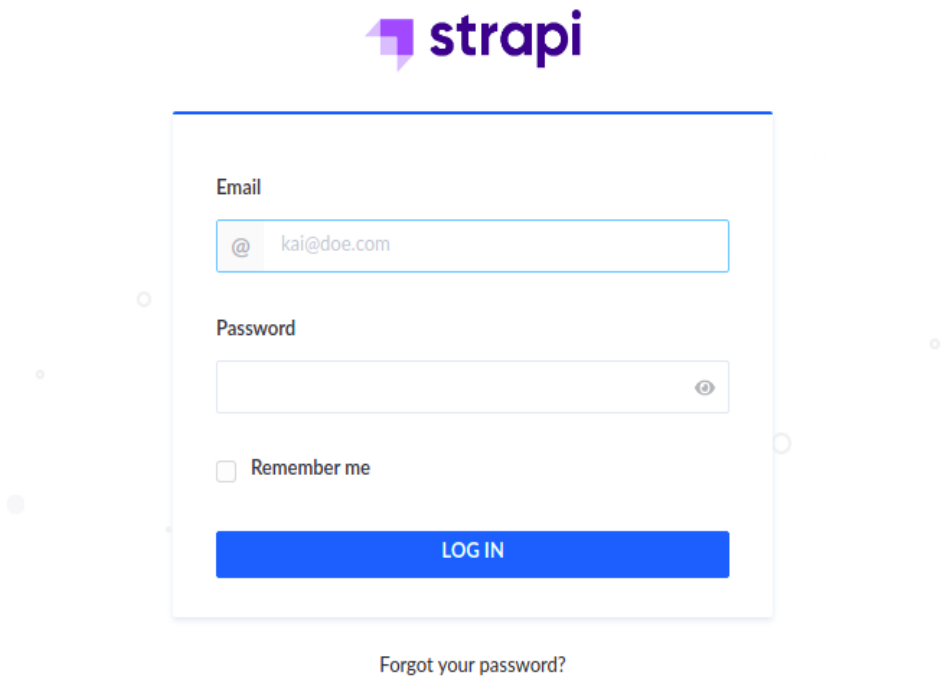


Рисунок 4.1 – Вхід в панель адміністратора

Після авторизації в системі, для керування та переходу між сторінками з налаштуваннями використовується бокова панель (рисунок 4.2).

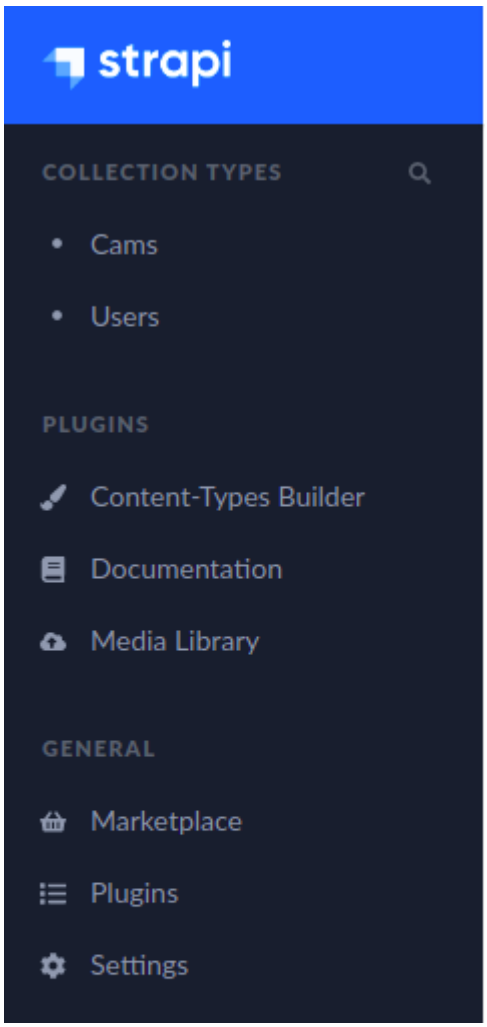


Рисунок 4.2 – Бокова панель

У боковій панелі адміністратора є вибір між двома сторінками, для керування системою:

1) Камери – сторінка адміністрування з можливістю перегляду (рис. 4.3), створення (рис. 4.5) і редагування (рис. 4.4) камер.

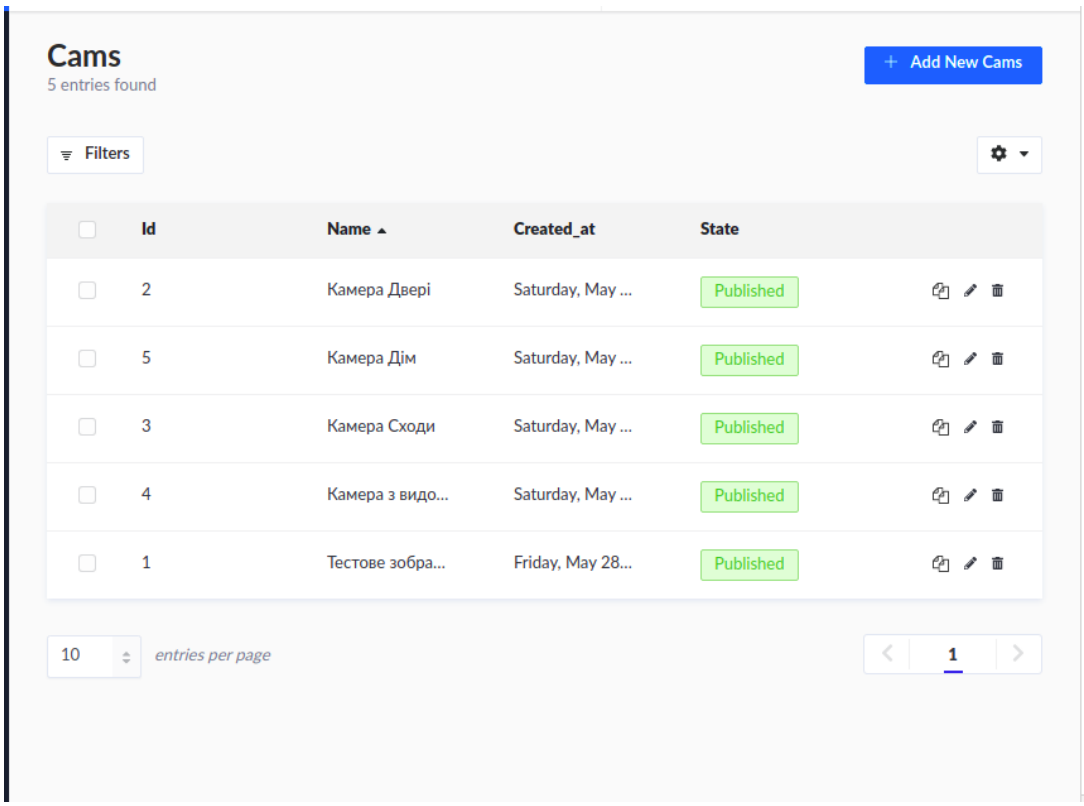


Рисунок 4.3 – Список створених камер

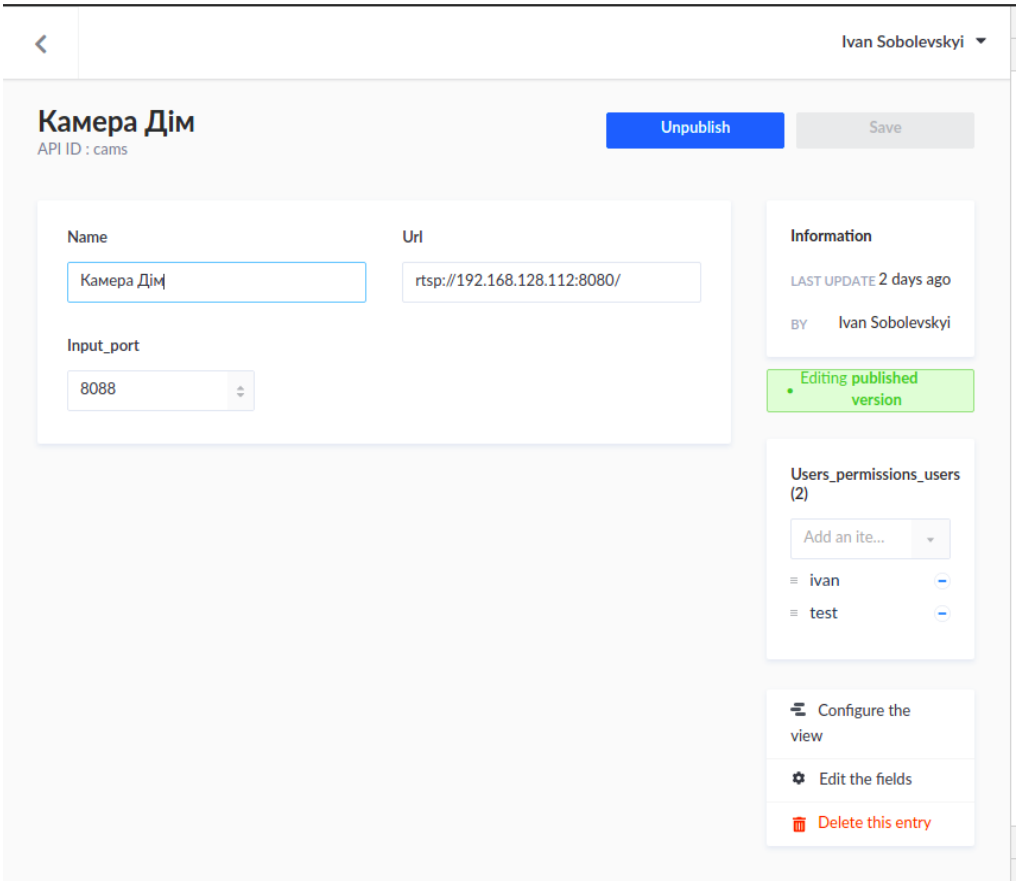


Рисунок 4.4 – Редагування камери

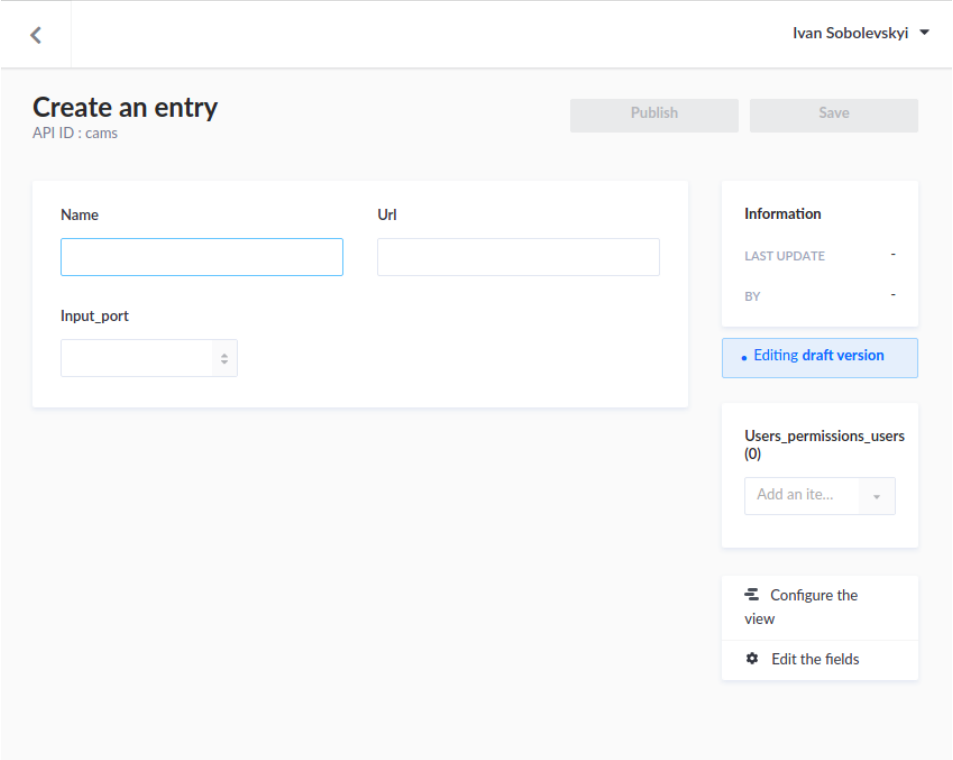


Рисунок 4.5 – Нова камера

Крім того, користувач панелі адміністратора може додавати або редагувати користувачів з доступом до певної камери (рисунок 4.6).

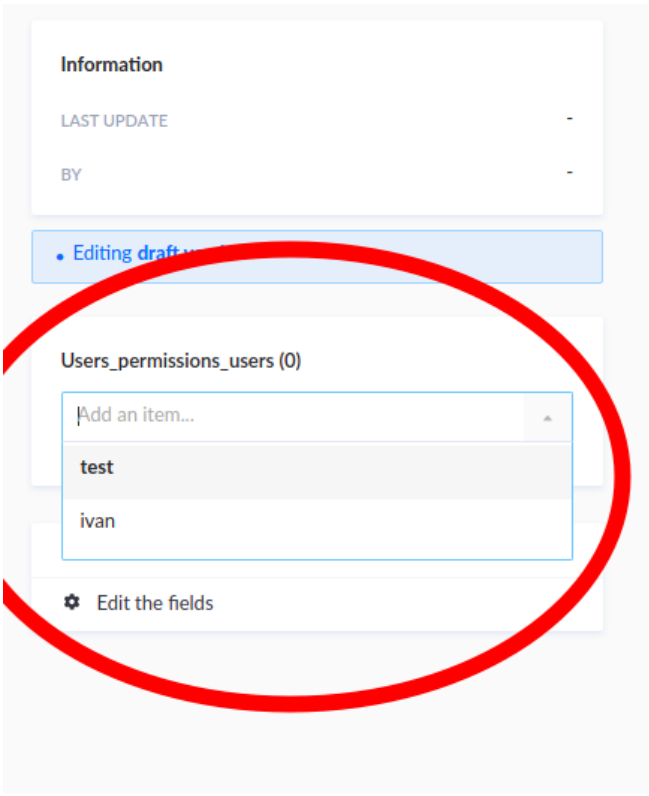


Рисунок 4.6 – Редагування прав користувача аміністратором

2) Користувачі – сторінка адміністрування з можливістю перегляду (рисунок 4.7), створювання (рисунок 4.9) і редагування (рисунок 4.8) користувачів.

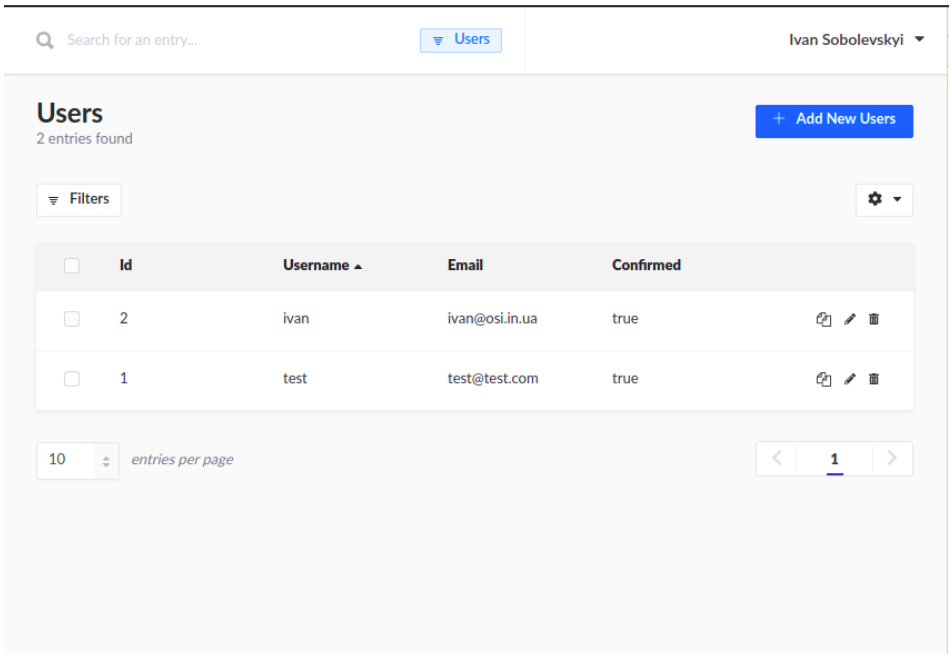


Рисунок 4.7 – Список користувачів

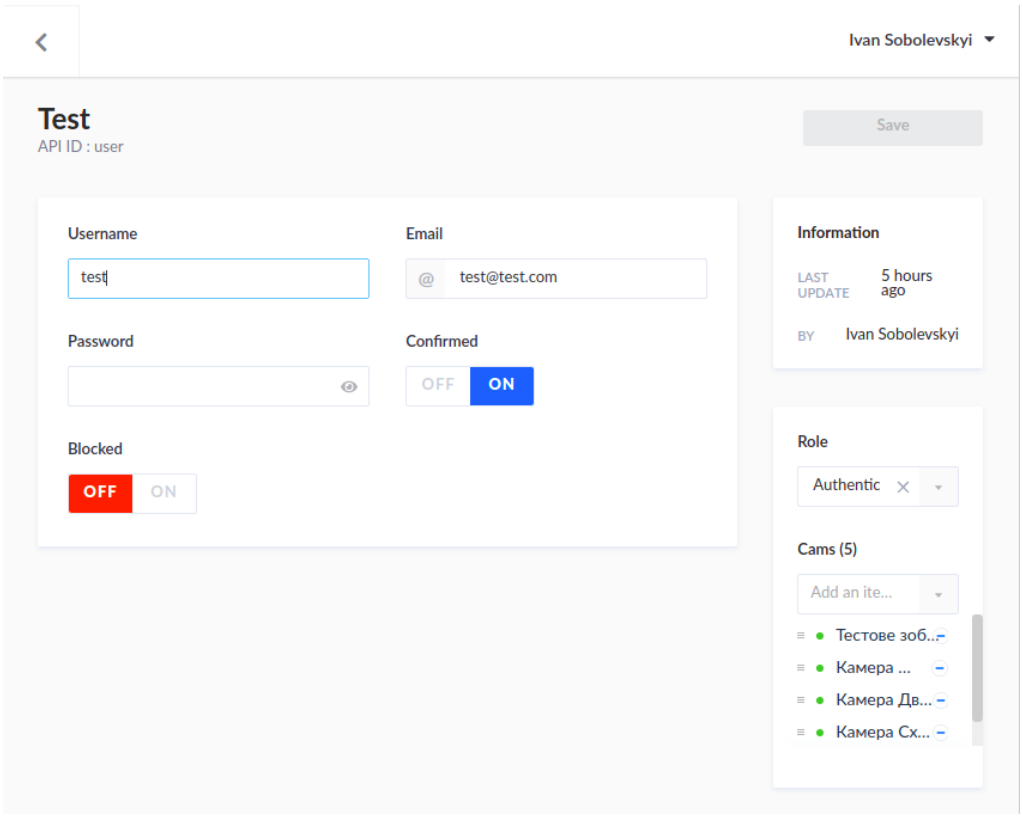


Рисунок 4.8 – Редагування користувача

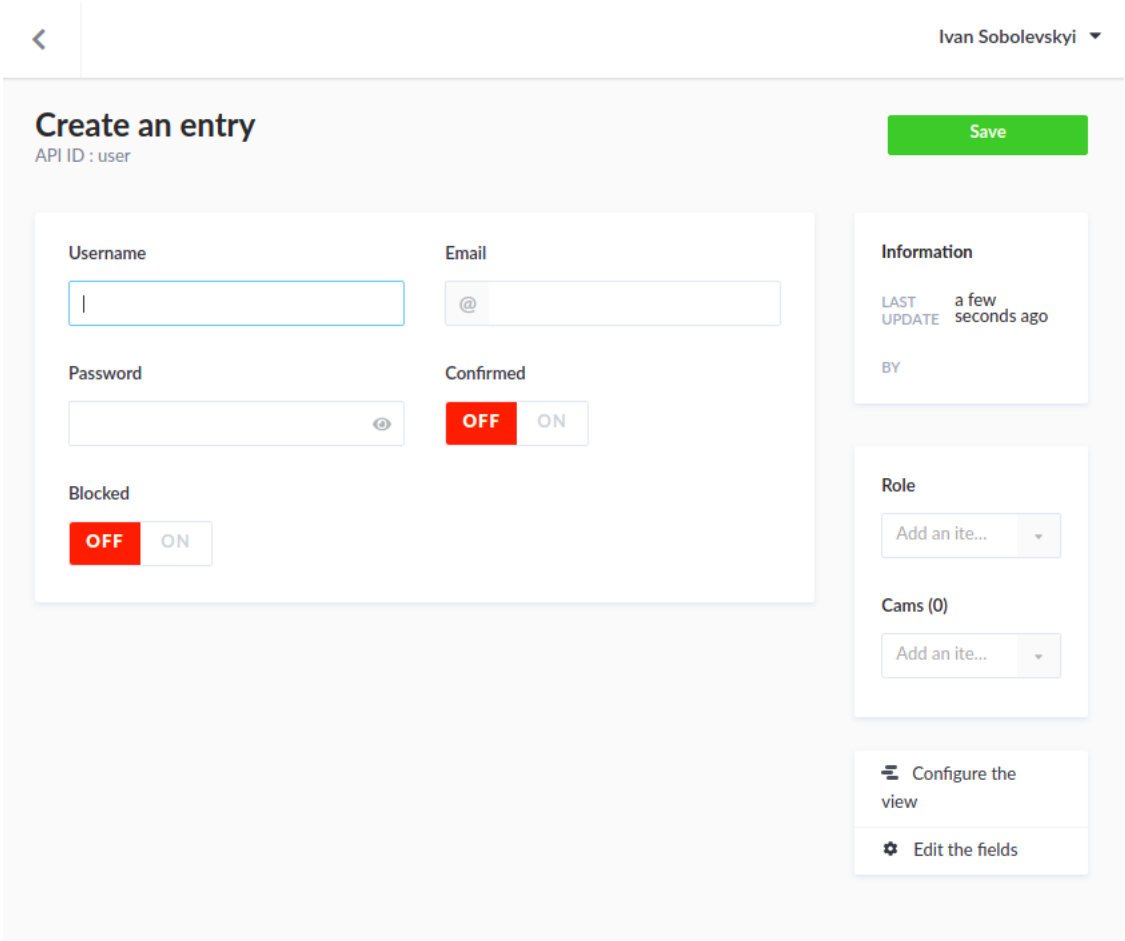


Рисунок 4.9 – Сторінка створення

Адміністратор може редагувати камери, які доступні для користувача системи, для надання доступу перегляду цієї камери (Рисунок 4.10).

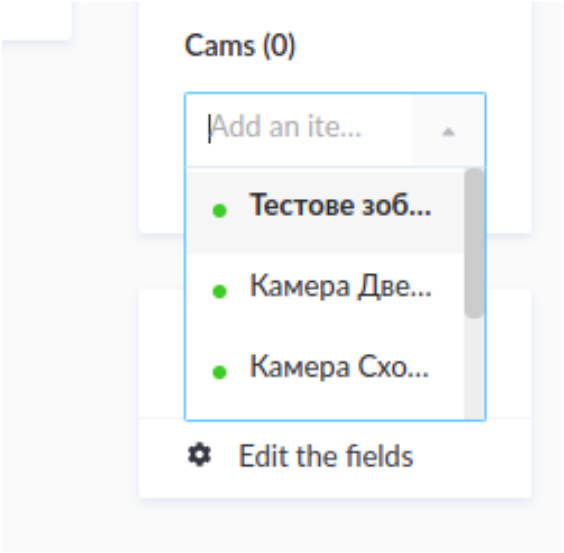


Рисунок 4.10 – Адміністратор надає доступи перегляду камери

Сторінка для перегляду камер користувачем. Спочатку для авторизації, користувач на сторінці входу повинен ввести надані адміністратором логін і пароль а потім натиснути кнопку «AUTH» (рис. 4.11).

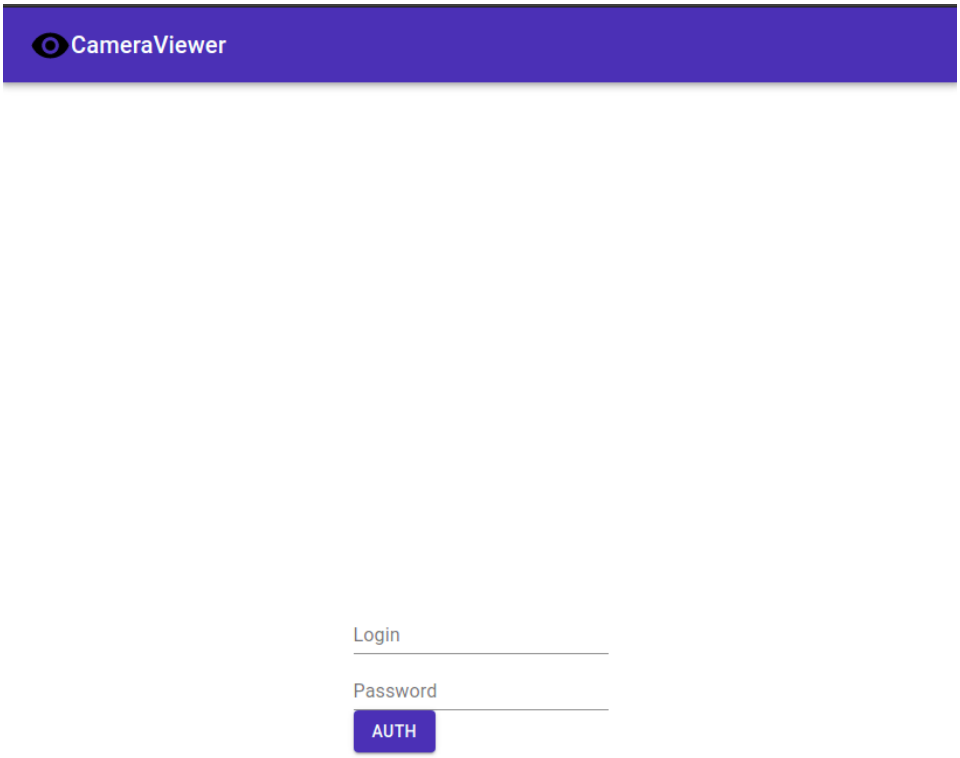


Рисунок 4.11– Вхід до користувацького інтерфейсу

Після успішної автентифікації користувача, він отримає маркер JWT і дані камери. Після цього сторінка перерисується, а користувач потрапить до інтерфейсу перегляду камер (рис. 4.12), який складається з таких елементів:

а) відео програвач – це елемент на сторінці, на який транслюється відеопоток з камери;
б) запливаючий список з камерами – це елемент сторінки, завдяки якому користувач може обрати потокову камеру (рис. 4.13);
в) кнопки для збереження відео – ці кнопки дозволяють користувачу керувати записом з камери відеоспостереження.

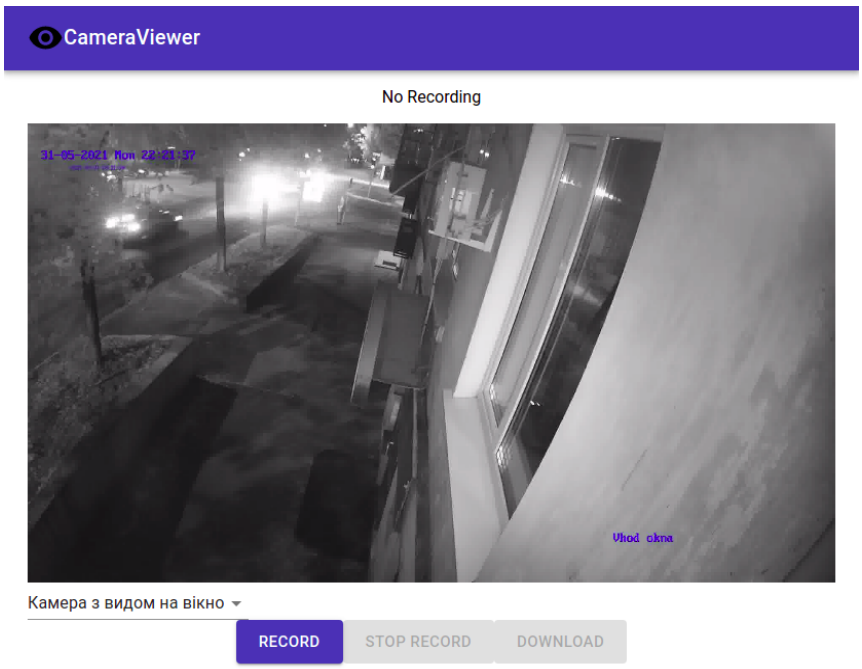


Рисунок 4.12 – Компонент перегляду трансляції з камер

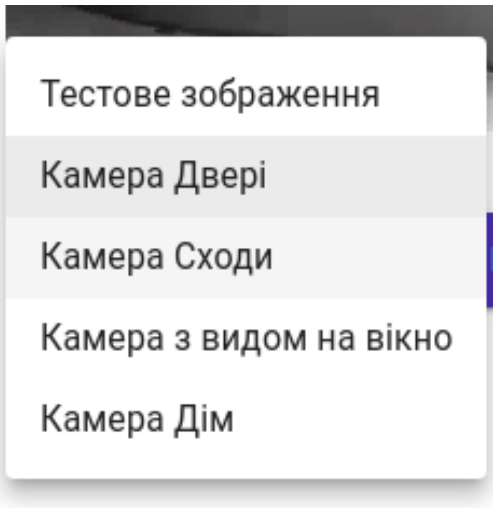


Рисунок 4.13 – Компонент вибору камери

Телеграм бот – це також частина користувацького інтерфейсу, яка виконує роль зовнішнього сервісу сповіщень користувача. При виявленні небезпеки, система висилає зображення з камери користувачу через телеграм бота. В кадрі зображений об’єкт (порушник), вказана точна дата та час поміченої загрози, приклад сповіщення від телеграм бота зазначено на рис. 4.17. Приклади помічених загроз зазначено на рис. 4.14, рис. 4.15 та рис. 4.16.


Рисунок 4.14 – Перший приклад поміченої загрози.


Рисунок 4.15 – Другий приклад поміченої загрози.


Рисунок 4.16 – Третій приклад поміченої загрози.

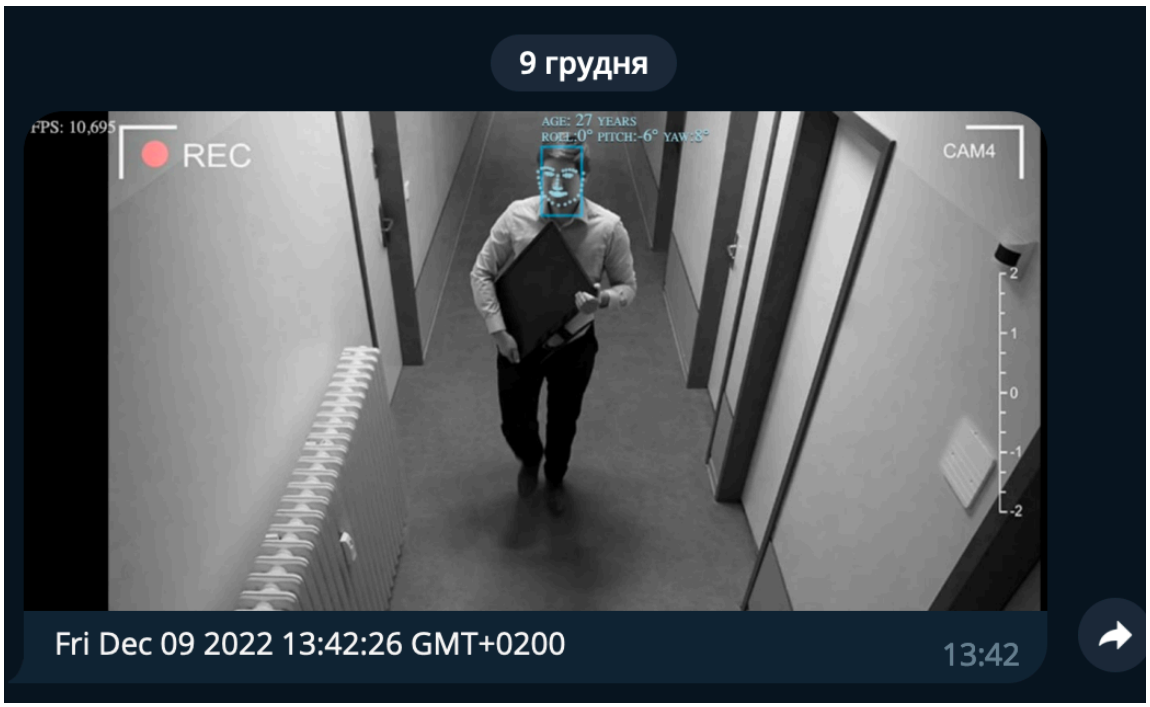


Рисунок 4.17 – приклад сповіщення про загрозу телеграм ботом

#### 4.3.4.3 Вхідні та вихідні дані

Вхід в админ панель:

а) точка API: http://localhost:1337;

б) метод авторизації: Basic Auth.

Вхід в інтерфейс користувача:

а) точка API: http://localhost:3000;

б) метод авторизації: Basic Auth.

Вхід для авторизації користувача:

а) точка API: http://localhost:1337/auth;

б) метод авторизації: Basic Auth.

Вхід для отримання інформації о камерах:

а) точка API: http://localhost:1337/cams;

б) метод авторизації: Bearer Token;

в) формат вихідних даних: Json;

г) вихідні дані:

— назва камери (string);

— адреса URL для доступу до камери (string);

— порт для підключення веб сокету (int).

Вхід для отримання відеопотоку з камер:

а) точка API: ws://localhost:80xx (для кожної камери відкривається окремий порт, заданий в адмін панелі);

б) метод авторизації: Bearer Token;

в) формат вихідних даних: Blob, MJPEG;

г) вихідні дані:

— бітова послідовність в формат MJPEG (bit).

Вихід сповіщення користувача:

а) точка API: https://api.telegram.org/bot{{TOKEN}}/sendMessage;

б) формат вихідних даних: json, jpg;

в) вихідні дані:

— проаналізоване зображення (jpg);

— точна дата та час, коли було.

### 4.4 Очікувані техніко-економічні показники

Розроблене програмне забезпечення дозволить економити на придбанні ліцензій та підписок на зовнішні сервіси, тому що усі компоненти системи розташовуються на локальному сервері та застосовуються бібліотеки з відкритим кодом та ліцензіями MIT, GPLv2, GPLv3, що робить їх повнітю безкоштовними. Система дозволить дешевше обирати апаратне забезпечення, так як система не прив’язана до конкретного вендору обладнання. Для покращення функціоналу системи не порібна заміна усього пакету ПЗ [31].

### 4.5 Висновки до розділу

Для того, щоб виконати поставлені задачі розробки системи відеоспостереження, було сформовано технічне завдання, задані призначення та сфери використання, та були обґрунтовані технічні

характеристики системи. Тому, для виконання системи були сплановані такі компоненти:

— панель користувача, використовуючи бібліотеку React та мову програмування JavaScript;

— панель адміністратора, використовуючи HRM Strapi, та мову програмування JavaScript;

— сервіс обробки відео потоку, з імплементованим аналізом об’єктів в кадрі та сповіщенням користувача про загрозу;

— сервіс маршрутизації даних для WebHook;

— написана конфігурація docker-compose мовою YAML.

# 5 ЕКСПЕРИМЕНТАЛЬНИЙ РОЗДІЛ

## 5.1 Мета і завдання експерименту

Мета експерименту: перевірка роботоспроможності системи відеоспостереження з імплементованим аналітичним модулем.

Задача експерименту: дослідним шляхом оцінити отримані показники, які ми отримуємо в результаті впровадження даної системи.

## 5.2 Методика експерименту

Методика проведення експерименту: Експеримент проводиться за допомогою відеофайлів, які взяті з камер відеоспостереження, що знаходять в різних умовах та ракурсах. Зображення підключаються до сервісу обробки відеопотоку, який описаний в розділі розробки програмного забезпечення. За допомогою віртуальної камери, яка встановлена на комп’ютері користувача, зображення з відеокамери передається до сервісу обробки відеопотоку за протоколом RTSP, що дозволить оцінити роботу модуля аналізу та модуля сповіщень користувача про загрозу. Результат буде порівняно з найпопулярнішою системою відеоспостереження Hikvision з застосуванням Hikvision AI.

## 5.3 Вимоги до експерименту

Експеримент проводиться для кожного відеофайлу окремо, як точка старту береться момент появи об’єкта в кадрі, як точка кінця береться момент, коли система аналізу розпізнала цей об’єкт, результат записується за допомогою запису екрану, а потім вимірюється час, який пройшов між точкою старту та кінця. Експеримент повинні окремо проводити три особи, для здобуття більш точного результату, серед результатів експериментаторів береться середнє отримане значення для кожного етапу. Похибка отриманих результатів буде залежати як від точності вимірів експериментатора, так і від самої системи аналізу.

Результат кожного етапу експерименту повинен бути записаний у відповідну колонку у таблиці результатів, ця таблиця має містити в собі номер відеоряду та результат аналізу цього відеоряду для кожної системи, над якою проводиться експеримент.

## 5.4 Результати експерименту

### 5.4.1 Сутність експерименту

Експеримент проводиться завдяки методу середнього арифметичного, аналізується 14 відеофайлів файлів у системах:

— Сервіс обробки відеопотоку;

— Hikvision AI.

Для кожного файлу розраховується час появи об’єкта в кадрі, який буде однаковим для кожного з трьох експериментаторів, цей час вважається точкою старту. Кожен експериментатор проганяє кожен файл у кожній системі аналізу, та записує час, який пройшов між точкою старту та моментом виявлення системою об’єкта. Після чого для кожного відповідного значення отриманого експериментаторами розраховується середнє арифметичне, яке буде вважатись результатом експерименту.

### 5.4.2 Результати експерименту в цифрах і фактах

Результати кожного дослідника наведені в таблиці 5.1, таблиці 5.2 та таблиці 5.3.

Таблиця 5.1 – Результати першого дослідника

| **№ відеоряду** | **Сервіс обробки відеопотоку, мс** | **Hikvision AI, мс** |
|---|---|---|
| 1 | 189 | 342 |
| 2 | 296 | 371 |
| 3 | 307 | 345 |
| 4 | 465 | 465 |

Продовження таблиці 5.1

| **№ відеоряду** | **Сервіс обробки відеопотоку, мс** | **Hikvision AI, мс** |
|---|---|---|
| 5 | 664 | 407 |
| 6 | 388 | 387 |
| 7 | 462 | 393 |
| 8 | 405 | 415 |
| 9 | 472 | 271 |
| 10 | 265 | 283 |
| 11 | 270 | 471 |
| 12 | 427 | 588 |
| 13 | 494 | 546 |
| 14 | 623 | 677 |

Таблиця 5.2 – Результати другого дослідника

| **№ відеоряду** | **Сервіс обробки відеопотоку, мс** | **Hikvision AI, мс** |
|---|---|---|
| 1 | 228 | 362 |
| 2 | 287 | 429 |
| 3 | 200 | 256 |
| 4 | 531 | 512 |
| 5 | 626 | 358 |
| 6 | 317 | 333 |
| 7 | 379 | 487 |
| 8 | 321 | 458 |
| 9 | 579 | 272 |
| 10 | 282 | 360 |
| 11 | 376 | 402 |
| 12 | 465 | 490 |

Продовження таблиці 5.2

| **№ відеоряду** | **Сервіс обробки відеопотоку, мс** | **Hikvision AI, мс** |
|---|---|---|
| 13 | 607 | 604 |
| 14 | 631 | 671 |

Таблиця 5.3 – Результати третього дослідника

| **№ відеоряду** | **Сервіс обробки відеопотоку, мс** | **Hikvision AI, мс** |
|---|---|---|
| 1 | 306 | 268 |
| 2 | 386 | 322 |
| 3 | 297 | 212 |
| 4 | 540 | 541 |
| 5 | 642 | 279 |
| 6 | 297 | 306 |
| 7 | 461 | 473 |
| 8 | 288 | 369 |
| 9 | 494 | 312 |
| 10 | 254 | 299 |
| 11 | 326 | 390 |
| 12 | 434 | 551 |
| 13 | 528 | 605 |
| 14 | 522 | 692 |

Зібравши данні, які отримали дослідники можна порахувати середню арифметичну для кожного етапу дослідження, середні показники зазначено в таблиці 5.4. На рисунку 5.1 зображено графік часу розпізнавання об’єкту для двох систем.

Таблиця 5.4 – Середні показники часу у мілісекундах, за який система помітила та проаналізовала об’єкт, після його появи в кадрі.

| № відеоряду | Сервіс обробки відеопотоку, мс | Hikvision AI, мс |
|---|---|---|
| 1 | 241 | 324 |
| 2 | 323 | 374 |
| 3 | 268 | 271 |
| 4 | 512 | 506 |
| 5 | 644 | 348 |
| 6 | 334 | 342 |
| 7 | 434 | 451 |
| 8 | 338 | 414 |
| 9 | 515 | 285 |
| 10 | 267 | 314 |
| 11 | 324 | 421 |
| 12 | 442 | 543 |
| 13 | 543 | 585 |
| 14 | 592 | 680 |

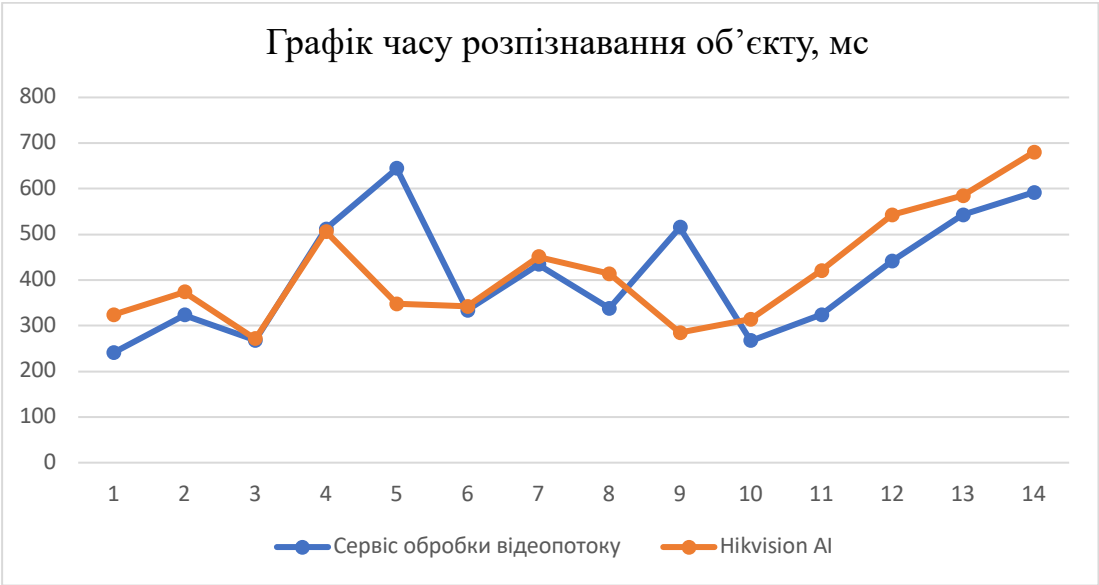


Рисунок 5.1 – Графік часу розпізнавання об’єкту

### 5.4.3 Аналіз відповідності досліджень

В підсумку, отримані результати показали, що система відеоспостереження імплементованним аналітичним модулем показала схожі результати з системою Hikvision AI, але при аналізі відеофайла під номером 9 система показала затримку на 230мс більшу ніж Hikvision AI, а у відеофайлі під номером 5 ця затримка сягнула 296мс. Це можна пояснити тим, що ці відеофайли були зняті вночі, та не мали підвищеної контрастності. Система Hikvision AI автоматично форматує кольори зображення для проведення аналізу кадру, тому вона має перевагу над розробленою системою відеоспостереження при таких умовах. Однак, якщо порахувати середню затримку власної системи – 412,64 мс та порівняти її з середньою затримкою Hikvision AI – 418,43 мс , то виходить що власна система реагує на 5,79 мс швидше. Максимальна затримка власної системи склала – 644 мс, а Hikvision AI – 680 мс.

### 5.4.4 Характеристика новизни результатів

Досліджена поведінка системи відеоспостереження, результати досліджень дозволили переконатись в тому, що використання системи відеоспостереження з аналітичним модулем є доцільним, так як результати показані розробленою системою є схожими і навіть кращими за аналогічну систему, яка присутня на ринку, що являється науково значущім результатом. Експеримент підтвердив теоретичні розрахунки, та несе практичну цінність, яка полягає в тому, що систему можна використовувати в цілях інтеграції з існуючими системами відеоспостереження, система може використовуватись з обладнаням, яке не прив’язано до конкретного вендору. Система відеоспостереження з вбудованим аналітичним модулем може служити для забезпечення вчасного реагування на злочини.

## 5.5 Висновки до розділу

В цьому розділі був проведений експеримент, який показав задовільні результати, перед проведенням експерименту було виявлено мету та поставлено задачу експерименту, також була поставлена методика, за якою проводився експеримент, роздані вказівки експериментаторам та зібрані дані. Після цього була проведена аналітика результатів експерименту, яка показала, що експеримент довів практичну та наукову значущість розробленої системи відеоспостереження.

# ВИСНОВКИ

Кваліфікаційна робота є завершеною науковою роботою, в якій вирішена науково-практична задача створення програмно-технічної реалізації комп'ютерної системи IP-відео-нагляду комплексу «Золоті ключі» з опрацюванням передачі відео інформації на базі Raspberry Pi.

Основні висновки і результати роботи полягають у наступному:

1. Досліджені існуючі системи відеоспостереження, було виявлено переваги та недоліки кожної з цих систем. Обґрунтована необхідність розробки власної системи відеоспостереження з метою удосконалення та вирішення недоліків сучасних систем;

2. Розроблено математичну модель комп’ютерної системи відеоспостереження з аналітичним модулем, яка була розглянута як система масового обслуговування, математична модель була побудована завдяки використанню ймовірнісної математичної моделі масового обслуговування. Це дозволило провести розрахунок станів аналітичного модулю, та залежність між вхідними та вихідними сигналами системи;

3. З метою забезпечення стабільності та надійності роботи системи відеоспостереження було проведено аналіз та сформовано специфікацію обладнання для роботи з системою. Також були сформовані вимоги до системи, і була створена структурна схема;

4. При розробці програмного забезпечення були використані результати теоретичних досліджень, та прийняті до виконання вимоги до системи. Було розроблено систему відеоспостереження з аналітичним модулем, за основу для програмного забезпечення було взято розроблену модель комп’ютерної системи;

5. Для побудови аналітичного модулю, була взята бібліотека face-api яка є обгорткою для бібліотеки TensorFlow. Це дало можливість використати заздалегідь натреновану нейронну мережу, для побудови системи

розпізнавання об’єктів, що дозволило зменшити час на розробку аналітичного модулю;

6. Перед проведенням експерименту було задано мету та завдання експерименту, була розроблена методика експерименту. Експеримент був проведений за участю трьох окремих дослідників, що дозволило отримати більш точні результати. Після обробки результатів був проведений аналіз. Отриманні результати експерименту дозволяють обґрунтувати доцільність використання розробленої системи відеоспостереження;

7. Результати експерименту показали, що використання розробленої системи відеоспостереження з аналітичним модулем є доцільним, так як вона показала схожі, а подекуди кращі результати ніж найбільш популярне альтернативне риночне рішення, при тому, що розроблена система дозволяє використовуватись з широким спектром обладнання, має відкритий код та може бути запущено як локально, так і в хмарі.

# ПЕРЕЛІК ПОСИЛАНЬ

1. Види систем відеоспостереження – вибір і огляд систем відеоспостереження URL: https://secur.ua/articles/ua_vidi-sistem-videosposterezhennja-vibir-i-ogljad-sistem-videosposterezhennja.html

2. Каталог відеокамер URL: https://secur.ua/videonablyudenie/kamery/vse-videokamery/

3. Комп'ютерні мережі / А. Г. Микитишин, М. М. Митник, П. Д. Стухляк, В. В. Пасічник. – Львів: Магнолія 2006, 2013. – 256 с.

4. Real Time Streaming Protocol (RTSP) URL: https://www.ietf.org/rfc/rfc2326.txt

5. Hypertext Transfer Protocol – HTTP/1.1 URL: https://www.w3.org/Protocols/rfc2616/rfc2616.html

6. RTP: A Transport Protocol for Real-Time Applications URL: https://www.ietf.org/rfc/rfc3550.txt

7. Всі переваги систем відеоспостереження Hikvision URL: https://chernigiv-city.com/ua/article/107314-vsi-perevagi-sistem-videosposterezhennya-hikvision

8. Основні переваги виробника Dahua URL: https://dostyp.com.ua/novini/u-chomu-osnovni-pierievaghi-virobnika-dahua-1

9. 360 Vision Technology URL: https://www.360visiontechnology.com/

10. Hikvision’s impressive AI analytics technologies URL: https://www.hikvision.com/en/core-technologies/ai-analytics/

11. Why DeepVision AI? Let's summarize URL: https://deepvisionai.in/

12. How TensorFlow Can Be Used for Video Analytics URL: https://www.opensourceforu.com/2020/01/how-tensorflow-can-be-used-for-video-analytics/

13. Основи біомедичного радіоелектронного апаратобудування : навчальний посібник / С. М. Злепко, С. В. Павлов, Л. Г. Коваль, І. С. Тимчик. – Вінниця : ВНТУ, 2011. – 134 с.

14. Литвинов А.Л. Теорія систем масового обслуговування : навч. посібник / А. Л. Литвинов ; Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків : ХНУМГ ім. О. М. Бекетова, 2018. – 141 с.

15. Сорока Л.І. Випадкові процеси: методичні рекомендації / Л. І. Сорока, І. В. Кальчук. – Луцьк: Східноєвропейський національний університет імені Лесі Українки, 2013. – 56 с.

16. Microservice architecture URL: https://microservices.io/

17. Telegram APIs URL: https://core.telegram.org/

18. What Is SQLite? URL: https://www.sqlite.org/index.html

19. Raspberrypi products URL: https://www.raspberrypi.com/products/

20. Raspberry Pi High Quality Camera URL: https://www.raspberrypi.com/products/raspberry-pi-high-quality-camera/

21. Raspberry Pi Compute Module 4 PoE UPS Base Board (22849) URL: https://evo.net.ua/raspberry-pi-compute-module-4-poe-ups-base-board-22849/?gclid=CjwKCAiA-dCcBhBQEiwAeWidtTh0iqJOEb2cq1e877GPNm_BqmGHzaiy0vco4BR0oScJmyniL_ERMxoCTj8QAvD_BwE

22. RB5009UPr+S+IN URL: https://mikrotik.com/product/rb5009upr_s_in

23. WORLD'S MOST ADVANCED BROADBAND SATELLITE INTERNET URL: https://www.starlink.com/technology

24. Precision 5820 Tower Workstation URL: https://www.dell.com/en-us/shop/desktop-computers/precision-5820-tower-workstation/spd/precision-5820-workstation/xctopt5820us

25. NVIDIA A2 Tensor Core GPU URL: https://www.nvidia.com/en-us/data-center/products/a2/

26. APC Smart-UPS, Line Interactive, 3kVA, Rackmount 2U, 230V, 8x IEC C13+1x IEC C19 outlets, Network Card, AVR, LCD URL: https://www.apc.com/shop/lv/en/products/APC-Smart-UPS-Line-Interactive-3kVA-Rackmount-2U-230V-8x-IEC-C13-1x-IEC-C19-outlets-Network-Card-AVR-LCD/P-SMT3000RMI2UNC

27. Node.js in Action / M.Cantelon, M. Harter, T. Holowaychuk, N. Rajlich. – Shelter Island: Manning Publications Co, 2013. – 416 p.

28. Strapi – Open source Node.js Headless CMS URL: https://strapi.io/features

29. A complete, cross-platform solution to record, convert and stream audio and video URL: https://ffmpeg.org/

30. Banks A. Learning React: Modern Patterns for Developing React Apps / A. Banks, E. Porcello. – Sebastopol, CA: O'Reilly Media, 2020. – 310 p.

31. Open source iniciative URL: https://opensource.org/licenses/MIT

# ДОДАТОК А

Текст програми веб відеонагляду

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
“ДНІПРОВСЬКА ПОЛІТЕХНІКА”

# ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
# ВЕБ ВІДЕОНАГЛЯД

Текст програми
804.02070743.22014-01 12 01
Листів 10

2022

# АНОТАЦІЯ

Дана програма містить в собі частину програмного коду для системи відеоспостереження з аналітичним модулем. Програма призначена для обробки відеотрансляції, виявлення об’єктів-порушників та сповіщення про них користувача системи.

Програма розроблена мовою JavaScript з використанням синтаксису ES6 та TypeScript.

# ЗМІСТ

# 1 ТЕКСТ ПРОГРАМИ

## 1.1 Сервіс збору відеопотоку

```
# Імпорт залежностей
const { getCams } = require('./api/cams/cams')
const Stream = require('./node-rtsp-stream-es6')
const Recorder = require('node-rtsp-recorder').Recorder

const onlineStreams = {}

# Функція створення запису з камери
function createRecord(cam) {
 console.log(cam.name.replace(/\s+/g, ''))
 var rec = new Recorder({
   url: cam.url,
   timeLimit: 5*60,
   folder: '/code/videos',
   name: cam.name.replace(/\s+/g, ''),
   fileNameFormat: 'hh:mm:ss'
 })
 rec.fileNameFormat =
 rec.startRecording();
}

# Функція створення трансляції відеопотоку
async function createStreams(cams) {
 for(const cam of cams) {
   stream = new Stream({
     name: cam.name,
```

```
    url: cam.url,
    port: cam.input_port
  })
  stream.start()

  createRecord(cam)
 }
}


async function main() {
 let data = undefined;
 while(data == undefined) {
  try {
# Завантажуємо список камер в системі з API
    data = await getCams();
  } catch {
    data = undefined
  }
 }
 await createStreams(data);
 console.log(data)
}

main()
```

### 1.2 Модуль обробки відеопотоку

```
# Імпортуємо модулі для розпізнавання об’єктів у кадрі
import * as faceapi from 'face-api.js';
```

```
import { canvas, faceDetectionNet, faceDetectionOptions, saveFile } from
'./commons';

async function run() {
# Завантаження файлів кешу
  await faceDetectionNet.loadFromDisk('../../weights')
  await faceapi.nets.faceLandmark68Net.loadFromDisk('../../weights')
  await faceapi.nets.faceRecognitionNet.loadFromDisk('../../weights')

# Завантаження кадру
  const referenceImage = await canvas.loadImage(REFERENCE_IMAGE)
  const queryImage = await canvas.loadImage(QUERY_IMAGE)

# Розпізнавання у кадрі
  const resultsRef = await faceapi.detectAllFaces(referenceImage,
faceDetectionOptions)
    .withFaceLandmarks()
    .withFaceDescriptors()

  const resultsQuery = await faceapi.detectAllFaces(queryImage,
faceDetectionOptions)
    .withFaceLandmarks()
    .withFaceDescriptors()

# Додавання додаткової інформації до кадру
  const faceMatcher = new faceapi.FaceMatcher(resultsRef)

  const labels = faceMatcher.labeledDescriptors
```

```
    .map(ld => ld.label)
  const refDrawBoxes = resultsRef
    .map(res => res.detection.box)
    .map((box, i) => new faceapi.draw.DrawBox(box, { label: labels[i] }))
  const outRef = faceapi.createCanvasFromMedia(referenceImage)
  refDrawBoxes.forEach(drawBox => drawBox.draw(outRef))
# Збереження кадру для відсилки у телеграм
  saveFile('referenceImage.jpg', (outRef as any).toBuffer('image/jpeg'))

  const queryDrawBoxes = resultsQuery.map(res => {
    const bestMatch = faceMatcher.findBestMatch(res.descriptor)
    return new faceapi.draw.DrawBox(res.detection.box, { label:
bestMatch.toString() })
  })
  const outQuery = faceapi.createCanvasFromMedia(queryImage)
  queryDrawBoxes.forEach(drawBox => drawBox.draw(outQuery))
  saveFile('queryImage.jpg', (outQuery as any).toBuffer('image/jpeg'))
  console.log('done, saved results to out/queryImage.jpg')
}

run()
```

### 1.3 Сервіс інтерфейсу користувача:

```
# Імпорт React та компонентів інтерфейсу
import React from 'react';
import store  from './store';
import TopBar from './components/TopBar';
import Stream from './components/Stream';
import { Auth } from './components/Auth';
```

```
import { Provider } from 'react-redux';

import Container from '@material-ui/core/Container';
import Select from '@material-ui/core/Select';
import MenuItem from '@material-ui/core/MenuItem';
import { useSelector } from 'react-redux'
import config from './config'

# Компонент підключення до трансляції камери
function MultipleStreamBar() {
 const cams = useSelector((state) => state.auth.cams)
 console.log(cams)
 const [port, setPort] = React.useState(cams[0].input_port);

 const jwt = useSelector((state) => state.auth.jwt)

# Зміна камери
 const handleChange = (event) => {
   console.log(event.target.value)
   setPort(event.target.value);
 };

# Відео програвач для перегляду трансляції з обраної камери
 return (
   <>
     <Stream     url={`ws://${config.backend.url}:${port}/?key=${jwt}`}     key='tab-
8083' id="8083">
# Елемент вибору поточної камери
      <Select
```

```
 labelId="demo-simple-select-error-label"
        id="demo-simple-select-error"
        value={port}
        onChange={handleChange}
       >
        {cams.map((cam) => <MenuItem key={cam.name}
value={cam.input_port}>{cam.name}</MenuItem>)}
       </Select>
     </Stream>
  </>
)
}
# Головний елемент додатку
function App() {
 const isAuth = useSelector((state) => state.auth.isAuth)
 console.log('token', isAuth)

 return (
  <div className="App">
   <header className="App-header">
    <TopBar
     name="CameraViewer"
    ></TopBar>
    <Container>
     {
      isAuth ?
       <MultipleStreamBar></MultipleStreamBar> :
       <Auth></Auth>
     }
```

```
      </Container>

    </header>
  </div>
);
}
# Підключення Redux
export function AppWrapper() {
 return (
  <Provider store={store}>
    <App />
  </Provider>
)
}
```