

МІНІСТЕРСТВО ВНУТРІШНІХ СПРАВ УКРАЇНИ
ЛЬВІВСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ ВНУТРІШНІХ СПРАВ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ УПРАВЛІННЯ, ПСИХОЛОГІЇ
ТА БЕЗПЕКИ

Кафедра інформаційних технологій

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ДИСТАНЦІЙНОГО
ВІДЕОСПОСТЕРЕЖЕННЯ ТА ІДЕНТИФІКАЦІЇ РУХУ

кваліфікаційна робота
здобувача вищої освіти
4 курсу денної форми навчання
Олега Васильовича ФЕЦЯКА

Науковий керівник:
старший викладач кафедри
інформаційних систем та технологій
Наталія ГАЛАЙКО

Рецензент:

вчене звання, науковий ступінь

(Ім'я ПРИЗВИЩЕ рецензента)

Кваліфікаційна робота допущена до захисту

« ____ » _____ 2025 р., протокол № ____

Завідувач кафедри інформаційних технологій

Ігор ФЕДЧАК

(підпис)

Львів
2025

АНОТАЦІЯ

Фецяк О. Розробка інформаційної системи дистанційного відеоспостереження та ідентифікації руху. – Рукопис.

У кваліфікаційній роботі здійснено розробку інформаційної системи для дистанційного відеоспостереження та ідентифікації руху. Проведено аналіз сучасних методів відеоспостереження, технологій обробки відеопотоків та алгоритмів детекції руху. Визначено основні вимоги до системи, що включають автоматизацію процесу моніторингу, високу швидкість обробки даних, інтеграцію з іншими засобами безпеки та забезпечення стабільної роботи навіть в умовах обмеженого інтернет-з'єднання. Особлива увага приділена питанням безпеки та збереження відеоданих, що є критично важливими для правоохоронних органів.

Проведено порівняльний аналіз існуючих програмних рішень у сфері відеоспостереження, що дозволило визначити їхні переваги та недоліки. У результаті створено інформаційну систему, що забезпечує централізований моніторинг відеопотоків, автоматичне виявлення руху та швидке реагування на підозрілі події.

Ключові слова: інформаційна система, відеоспостереження, детекція руху, автоматизація моніторингу, правоохоронні органи.

ABSTRACT

Fetsiak O. Development of an Information System for Remote Video Surveillance and Motion Detection. Manuscript.

The qualification work focuses on the development of an information system for remote video surveillance and motion detection. An analysis of modern video surveillance methods, video stream processing technologies, and motion detection algorithms has been conducted. The main system requirements were identified, including the automation of the monitoring process, high-speed data processing, integration with other security measures, and ensuring stable operation even with limited internet connectivity. Special attention was given to security and data preservation issues, which are critical for law enforcement agencies.

A comparative analysis of existing software solutions in the field of video surveillance was carried out, allowing the identification of their advantages and disadvantages. As a result, an information system was created, providing centralized monitoring of video streams, automatic motion detection, and quick response to suspicious events.

Keywords: information system, video surveillance, motion detection, monitoring automation, law enforcement.

ЗМІСТ

ВСТУП	5
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ	8
1.1. Основні засади дослідження	8
1.2. Стан та перспективи дослідження в Україні	10
1.3. Аналіз відомих програмних засобів	12
Висновки до першого розділу	27
РОЗДІЛ 2. СИСТЕМНИЙ АНАЛІЗ ТА ПРОЄКТУВАННЯ СИСТЕМИ	28
2.1. Побудова дерева цілей	28
2.2. Проєктування системи	32
2.3. Вибір та обґрунтування методів вирішення задачі	37
2.4. Вибір засобів розроблення	41
Висновки до другого розділу	44
РОЗДІЛ 3. ПРАКТИЧНА РЕАЛІЗАЦІЯ	45
3.1. Структура програмного забезпечення	45
3.2. Реалізація програмної системи	46
3.3. Аналіз контрольного прикладу	51
Висновки до третього розділу	53
ВИСНОВКИ	54
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	56
ДОДАТКИ	59

ВСТУП

Актуальність теми дослідження. В умовах стрімкого зростання рівня урбанізації, глобалізації та поширення технологій безпеки особливої уваги набуває питання автоматизації відеоспостереження для правоохоронних органів. Традиційні системи моніторингу, що базуються на безперервному перегляді відеопотоків операторами, вимагають значних людських ресурсів та не завжди забезпечують оперативність реагування на інциденти [1]. Одним із ключових викликів є суб'єктивний фактор – втома, неуважність або перевантаження оператора може призвести до пропуску критично важливої події. Сучасні технології комп'ютерного зору та штучного інтелекту дозволяють значно підвищити ефективність відеоспостереження, оскільки можуть працювати в режимі реального часу, аналізуючи великі обсяги відеоданих і автоматично виявляючи підозрілі події, об'єкти чи аномальну поведінку. Особливо актуальними ці технології стають для забезпечення правопорядку в громадських місцях, на стратегічно важливих об'єктах та транспортних вузлах.

Розробка інформаційної системи дистанційного відеоспостереження, яка включатиме можливості автоматичної ідентифікації руху, сприятиме підвищенню ефективності роботи правоохоронних органів. Така система дозволить не лише фіксувати правопорушення, а й прогнозувати потенційні загрози, що значно покращить загальний рівень громадської безпеки.

Аналіз останніх досліджень. Сучасні дослідження у сфері автоматизованого відеоспостереження активно розвиваються, зокрема у напрямках комп'ютерного зору, глибокого навчання та аналізу поведінки об'єктів. Дослідження Huu-Thanh [2] присвячене використанню методів глибокого навчання для детекції руху та ідентифікації аномальної поведінки у відеопотоках. Автори пропонують алгоритми, які дозволяють автоматично виявляти нетипову поведінку осіб, що може вказувати на потенційні

правопорушення. Redmon & Farhadі запропонували YOLO (You Only Look Once) [3] – одну з найефективніших моделей обробки відеопотоків у реальному часі, яка дозволяє швидко ідентифікувати об'єкти та їхнє переміщення у просторі. Дана технологія широко застосовується у системах відеоспостереження для виявлення підозрілих дій та аналізу руху. Ren et al. (2017) у своїх дослідженнях представили Faster R-CNN – модель, яка забезпечує високу точність розпізнавання об'єктів у складних відеосценах [4]. Liu et al. (2016) розробили SSD (Single Shot MultiBox Detector) [5], яка ефективно виявляє об'єкти на основі попередньо навчених нейронних мереж, що дозволяє підвищити продуктивність систем відеоспостереження. Значна увага приділяється також методам автоматичного аналізу аномальної поведінки осіб у відеопотоці, що може бути корисним для превентивного виявлення злочинної діяльності. Таким чином, дослідження у сфері інтелектуального відеоспостереження активно розвиваються, проте залишається відкритим питання ефективної інтеграції цих технологій у правоохоронну діяльність з урахуванням специфічних потреб служб безпеки.

Мета дослідження. Метою дослідження є розробка інформаційної системи дистанційного відеоспостереження та ідентифікації руху, яка дозволить автоматизувати процеси моніторингу, підвищити точність виявлення потенційних загроз та зменшити навантаження на операторів правоохоронних органів.

Завдання дослідження:

- провести аналіз сучасних методів детекції руху та ідентифікації об'єктів у відеопотоці;
- розробити структуру інформаційної системи, що дозволяє автоматизувати процес моніторингу;
- здійснити проектування системи з використанням об'єктно-орієнтованого підходу;
- здійснити вибір технологій, засобів розроблення та реалізувати систему.

Об’єкт дослідження: процес дистанційного відеоспостереження та ідентифікації руху.

Предмет дослідження: методи та засоби дистанційного відеоспостереження та ідентифікації руху.

Методи дослідження. Методи дослідження включають аналіз наукових джерел для оцінки сучасного стану технологій відеоспостереження та методів обробки відеопотоків. Використовуються методи машинного навчання та глибокого навчання для ідентифікації руху та аналізу поведінки. Для перевірки працездатності системи застосовувались методи комп’ютерного моделювання, що дозволили створити прототип та протестувати його у віртуальному середовищі.

Практичне значення роботи. Практична цінність роботи полягає в можливості застосування розробленої інформаційної системи відеоспостереження для підвищення ефективності роботи правоохоронних органів та інших структур безпеки. Автоматичний аналіз відео значно зменшить навантаження на операторів, дозволяючи оперативно виявляти підозрілі дії без необхідності постійного перегляду відеопотоків. Ідентифікація руху та аналіз поведінки в режимі реального часу сприятимуть швидкому прийняттю рішень і запобіганню злочинам. Таким чином, запропоноване рішення сприятиме підвищенню рівня громадської безпеки та ефективності роботи правоохоронних органів.

Структура роботи. Робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків. Повний обсяг основного тексту складає 51 сторінка, кількість ілюстрацій – 26, таблиць – 1, додатків – 1, кількість літературних джерел – 32. Загальний обсяг роботи – 63 сторінки.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ

1.1. Основні засади дослідження

У сучасному світі питання забезпечення громадської безпеки та правопорядку є одними з найважливіших завдань для правоохоронних органів. Стрімке зростання рівня злочинності, терористичних загроз та порушень громадського порядку вимагає впровадження новітніх технологій, здатних забезпечити ефективний контроль за ситуацією у режимі реального часу. Одним із найперспективніших напрямків у цій сфері є використання систем дистанційного відеоспостереження, які поєднують класичні методи відеофіксації із сучасними алгоритмами штучного інтелекту та комп'ютерного зору [6].

Використання таких технологій дозволяє не тільки отримувати відеоматеріали у режимі реального часу, а й здійснювати їхній автоматичний аналіз. Системи, що використовують методи машинного навчання, можуть самостійно виявляти аномальну поведінку, фіксувати несанкціоновані дії, ідентифікувати осіб та розпізнавати рух об'єктів на основі попередньо визначених сценаріїв. Це значно підвищує ефективність роботи правоохоронців, зменшує вплив людського фактору та дозволяє швидко реагувати на потенційні загрози.

Розпізнавання об'єктів стало важливим елементом сучасних технологій. Кілька років тому такі системи, як розпізнавання обличчя чи інших об'єктів, здавались фантастикою. Нині багато сфер, зокрема безпека аеропортів, роботизовані системи, автопілоти та сканери, активно використовують ці технології. Наприклад, у виробництві за допомогою роботів можна автоматично виявляти та видаляти дефектні деталі на конвеєрі для подальшої переробки. Технології розпізнавання обличчя також допомагають у прогнозуванні та запобіганні небезпечним ситуаціям у

людних місцях, а комп'ютерний зір дозволяє створювати автопілоти для допомоги водіям у критичних ситуаціях. Розпізнавання образів є однією з основних задач у розвитку інтелектуальних систем і має значний практичний вплив. Терміни «розпізнавання» і «класифікація» часто використовуються як синоніми, але вони не завжди є взаємозамінними, оскільки мають різні сфери застосування та підходи [7].

Математичні моделі та алгоритми є основою для досліджень у багатьох сферах, починаючи від розрахунку балістичних траєкторій і закінчуючи вибухами атомних бомб. Однак деякі методи, які базуються на відмові від створення повних математичних моделей, надають кращі результати для специфічних завдань, таких як прогнозування врожайності, рівня річок або розпізнавання образів. В цих випадках важливіше отримати просту відповідь, наприклад, класифікацію об'єкта за попередньо визначеними категоріями.

У ряді випадків розпізнавання образів також включає виявлення рухомих об'єктів. Техніки виявлення руху в комп'ютерному зорі дозволяють відстежувати та аналізувати рух об'єктів, що є важливим у відеоспостереженні, моніторингу доріг, безпеці аеропортів, охороні морських кордонів та інших сферах. Завдання виявлення і відстежування рухомих об'єктів є важливими для систем безпеки, контролю дорожнього руху та інших інтелектуальних пристроїв спостереження [8].

Актуальність розробки інформаційної системи дистанційного відеоспостереження та ідентифікації руху пояснюється тим, що сучасні технології дозволяють значно вдосконалити процес забезпечення безпеки у громадських місцях, на транспортних вузлах, у стратегічно важливих об'єктах та в інших критичних зонах. Традиційні методи відеоспостереження мають низку недоліків, серед яких основними є:

- Необхідність постійного моніторингу оператором, що збільшує ймовірність людської помилки та підвищує навантаження на персонал.

- Відсутність автоматичного аналізу відео – більшість існуючих систем не здатні самостійно розпізнавати загрози та вимагати від оператора вручного перегляду великого обсягу даних.

- Обмежені можливості інтеграції – застарілі системи відеоспостереження не завжди мають змогу взаємодіяти з базами даних, що ускладнює ідентифікацію осіб та транспортних засобів.

Сучасні підходи, які базуються на технологіях глибокого навчання та комп'ютерного зору, дозволяють вирішити ці проблеми [9]. Інформаційна система, яка здатна не лише здійснювати відеоспостереження, а й автоматично аналізувати отримані дані, може значно підвищити рівень безпеки, покращити розкриття злочинів та пришвидшити реакцію на потенційні загрози.

1.2. Стан та перспективи дослідження в Україні

В Україні спостерігається активний розвиток технологій відеоспостереження, зокрема в рамках проекту "Безпечне місто", що активно реалізується у великих містах країни. Цей проект передбачає впровадження систем відеоспостереження на стратегічних об'єктах, в громадських місцях, а також на важливих транспортних вузлах. Основною метою є забезпечення контролю за безпекою громадян, моніторинг стану навколишнього середовища та запобігання злочинам. Однак, більшість існуючих систем працюють в режимі пасивного спостереження, що означає, що операторам доводиться вручну переглядати відеоматеріали та ухвалювати рішення про необхідність втручання. Такий підхід суттєво обмежує ефективність відеоспостереження, адже людський фактор може призводити до затримок у реагуванні або навіть втрати важливих даних [10].

Незважаючи на це, в Україні спостерігається стрімкий розвиток автоматизованих систем відеоспостереження, здатних здійснювати не лише запис відео, а й виконувати складніші функції. Відеоспостереження вже активно використовується для зорового контролю, безперервного запису та

оперативного аналізу ситуацій, що допомагає знижувати ризик кримінальних інцидентів. Одним із важливих елементів, що підвищують ефективність відеоспостереження, є інтеграція систем з іншими технологіями, такими як охоронні сигналізації або автоматичні детектори руху, що дозволяють оперативно реагувати на тривожні ситуації.

З технологічного погляду, розвиток цифрових систем відеоспостереження в Україні відбувається через перехід від аналогових до цифрових систем, зокрема, через використання ІР-камер, які забезпечують високу якість зображення, а також можливість віддаленого перегляду через Інтернет. У цифрових системах відбувається передача відеосигналу через мережу, що значно підвищує гнучкість та ефективність роботи відеоспостереження. Завдяки використанню таких технологій, оператори можуть здійснювати моніторинг відео з декількох камер одночасно, при цьому зберігаючи можливість запису у різних форматах та з різною роздільною здатністю.

Проте, незважаючи на позитивні досягнення в цій галузі, однією з головних проблем залишається необхідність забезпечення надійної передачі даних та запобігання помилковим спрацюванням системи. Наприклад, необхідно уникати фальшивих тривог, спричинених рухом незначних об'єктів або змінами освітлення. У цьому контексті, важливою є розробка нових методів обробки відеоданих, що дозволить автоматично відрізняти корисні сигнали від шуму, а також забезпечити високу точність ідентифікації.

Прогнозується, що з впровадженням сучасних технологій штучного інтелекту, відеоспостереження в Україні набуде нових можливостей. Однією з ключових перспектив є автоматична ідентифікація осіб та об'єктів за допомогою нейромереж, що дозволить оперативно виявляти підозрілі особи та небезпечні предмети. Такі технології, як розпізнавання обличчя та номерних знаків, дозволять не лише підвищити рівень безпеки, а й значно спростити процес ідентифікації осіб, які можуть становити загрозу [11].

Також важливим напрямком є розвиток технологій для аналізу поведінки людей у реальному часі. Системи, здатні виявляти підозрілу активність, таку як агресивна поведінка, залишені без нагляду предмети або нетипові маршрути пересування, стануть важливим інструментом для забезпечення безпеки в громадських місцях та на великих об'єктах.

Ще однією перспективою є інтеграція систем відеоспостереження з аналітичними платформами правоохоронних органів, що дозволить швидше отримувати необхідні дані, а також покращить взаємодію між різними державними структурами. Завдяки такій інтеграції, правоохоронці зможуть більш оперативно реагувати на надзвичайні ситуації та координувати свої дії на основі об'єктивних відеоданих.

Використання цих технологій дозволить значно підвищити ефективність роботи силових структур, автоматизувати процеси відеоспостереження та скоротити час реагування на правопорушення. Впровадження таких інновацій також знизить навантаження на персонал і дозволить зосередитись на більш складних завданнях, що потребують людської уваги та оцінки. У результаті, рівень громадської безпеки в Україні значно підвищиться, а технології дистанційного відеоспостереження та ідентифікації руху стануть важливим елементом на шляху до розвитку "розумних" міст та безпечного суспільства.

1.3. Аналіз відомих програмних засобів

На ринку, існує множина систем аналогів. Розглянемо найбільш популярні.

Blue Iris – це потужна програма для відеоспостереження, яка дозволяє здійснювати моніторинг, запис та аналіз відео з численних камер у реальному часі. Вона підтримує широкий спектр камер, включаючи IP-камери, веб-камери, а також камери, підключені через локальну мережу. Ця система є однією з найпопулярніших та універсальних у своїй категорії, що

дозволяє створювати ефективні рішення для контролю різних об'єктів, таких як житлові будинки, офіси, магазини та підприємства [12].

Blue Iris дозволяє працювати з різними типами камер, не обмежуючись лише IP-камери, а й підтримує аналогові камери, підключені через відеореєстратори. Програма також оснащена вбудованою технологією для виявлення руху, що дає змогу автоматично починати запис відео або надсилати сповіщення, якщо на відеопотоці відбуваються зміни. Для запису відео система дозволяє використовувати як безперервний режим, так і активацію запису на основі подій, таких як виявлення руху.

Однією з ключових можливостей Blue Iris є мобільний доступ. Через спеціальний мобільний додаток користувачі можуть отримати доступ до відео та налаштувань програми, що дозволяє здійснювати моніторинг з будь-якої точки світу. Крім того, програма може інтегруватися з іншими системами безпеки, такими як датчики руху або інші системи охорони. Blue Iris також пропонує різноманітні режими перегляду відео, зокрема для детального аналізу відеоархівів. Головне вікно представлене на рис.1.1.



Рис.1.1. Головне вікно сервісу Blue Iris

Переваги Blue Iris:

- *Гнучкість і масштабованість.* Програма дозволяє підключати велику кількість камер, що дає змогу створювати різноманітні системи відеоспостереження для малих і великих об'єктів.

- *Широкий вибір підтримуваних камер.* Blue Iris підтримує різні типи камер, що дозволяє користувачеві вибирати техніку на свій розсуд, знижуючи витрати на апаратне забезпечення.
- *Детекція руху та автоматизація.* Програма ефективно використовує алгоритми для виявлення руху, що дозволяє заощаджувати місце на носіях та збільшувати ефективність системи.
- *Мобільний доступ.* Завдяки мобільному додатку користувачі можуть отримувати доступ до відео та налаштувань навіть коли вони перебувають поза домом чи офісом.
- *Підтримка віддаленого перегляду.* Всі відео з камер можна переглядати з будь-якого пристрою, підключеного до Інтернету.

Недоліки Blue Iris:

- *Висока вимогливість до ресурсів системи.* Для оптимальної роботи Blue Iris вимагає потужного комп'ютера, особливо якщо йдеться про великі системи з численними камерами та високою роздільною здатністю.
- *Інтерфейс програми може бути складним для новачків,* особливо коли йдеться про налаштування детекції руху, запису відео або інтеграцію з іншими системами.
- *Проблеми з сумісністю деяких камер.* Можуть виникати проблеми з підключенням деяких моделей камер.
- *Платна ліцензія.* Програма є платною, що може бути мінусом для деяких користувачів, які шукають безкоштовні рішення.

iSpy – це потужне програмне забезпечення для відеоспостереження, яке забезпечує підтримку широкого спектра камер, включаючи IP-камери, USB-камери та інші типи пристроїв. Програма дозволяє здійснювати моніторинг в реальному часі, запис відео, а також інтегрувати різноманітні системи для автоматичного виявлення руху, що робить її корисною для контролю об'єктів, таких як домівки, офіси, склади, підприємства тощо. iSpy надає можливість для інтеграції з іншими охоронними системами, такими як

датчики руху та тривожні сигналізації, а також надає віддалений доступ до відео через мобільні додатки або веб-браузер [13].

Програма має можливість запису відео з автоматичним активуванням при виявленні руху. Крім того, iSpy дозволяє налаштувати сповіщення про події, що з'являються, та здійснювати відеоаналітику для детальнішого перегляду записів. Це дає змогу значно зменшити час на пошук важливих моментів у архіві. Окрім цього, iSpy має потужні функції для керування та збереження відеофайлів, а також для налаштування параметрів камери, що дозволяє адаптувати систему під конкретні потреби користувача (рис.1.2).

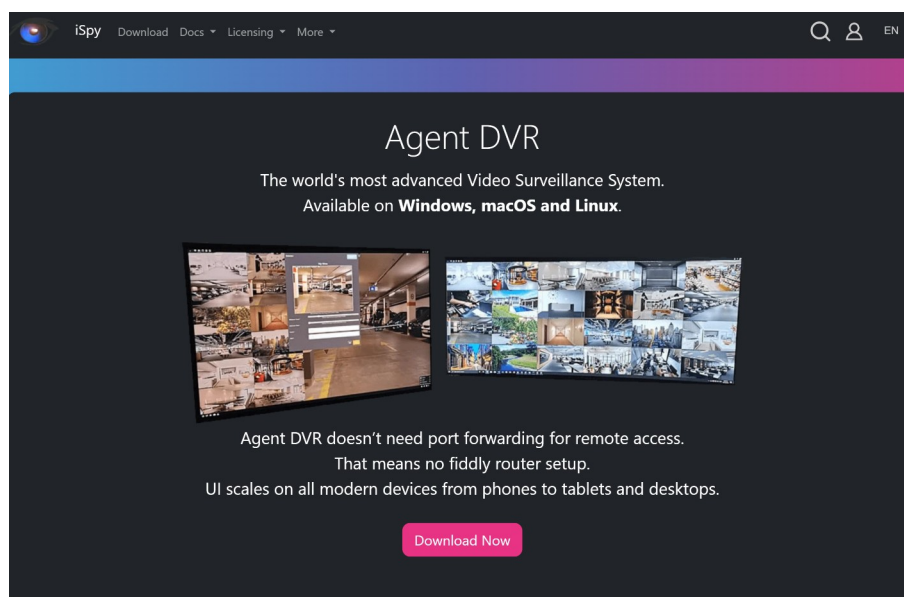


Рис.1.2. Головне вікно системи iSpy

Переваги iSpy:

- *Широка підтримка камер.* Підтримка IP-камер, USB-камер та відеообладнання через відеореєстратори.
- *Детекція руху.* Автоматичне активування запису або сповіщень при виявленні руху.
- *Мобільний доступ.* Можливість віддаленого перегляду через мобільний додаток або веб-браузер.
- *Безкоштовна базова версія.* Доступ до основних функцій безкоштовно.

- *Гнучкість в налаштуваннях.* Можливість налаштувати систему під конкретні потреби, включаючи частоту запису, методи зберігання і відеоаналітику.

- *Інтеграція з іншими системами.* Можливість підключення додаткових охоронних датчиків та сигналізацій.

Недоліки iSpy:

- *Складність налаштування для новачків.* Для повноцінного використання може знадобитися певний рівень технічних знань.

- *Обмеження безкоштовної версії.* Деякі розширені функції, такі як розпізнавання облич або інтеграція з нейромережами, доступні лише у платній версії.

- *Проблеми з сумісністю камер.* Не всі камери можна без труднощів інтегрувати, особливо якщо вони мають специфічні формати або налаштування.

- *Потрібна постійна стабільність Інтернет-з'єднання.* Віддалений доступ через Інтернет потребує надійного з'єднання.

- *Обмежені можливості для аналізу відео в реальному часі.* Для глибшої аналітики може бути необхідна додаткова настройка або використання платних модулів.

Milestone XProtect – це високопродуктивне програмне забезпечення для відеоспостереження, яке дозволяє створювати масштабовані, надійні та потужні системи для моніторингу та запису відео з великої кількості камер. XProtect підтримує інтеграцію з різноманітними типами камер, включаючи IP-камери, PTZ-камери, камери з високою роздільною здатністю та інші сучасні пристрої відеоспостереження. Це рішення здебільшого використовується в корпоративних або великих інфраструктурах для забезпечення високої безпеки, таких як підприємства, магазини, торгові центри, аеропорти або інші об'єкти, де необхідно постійно моніторити великі площі [14].

XProtect надає гнучку архітектуру для зберігання та обробки відео, включаючи можливості для інтеграції з системами управління доступом, аналітикою, датчиками руху та іншими рішеннями безпеки. Програмне забезпечення має потужні інструменти для детекції руху, що дозволяє автоматично реагувати на події, а також для інтеграції з аналітичними платформами для глибокого аналізу відео. Крім того, XProtect надає можливість віддаленого доступу до відео через мобільні додатки та веб-браузери, що дозволяє операторам здійснювати моніторинг з будь-якої точки світу. Головне вікно представлено на рис.1.3.

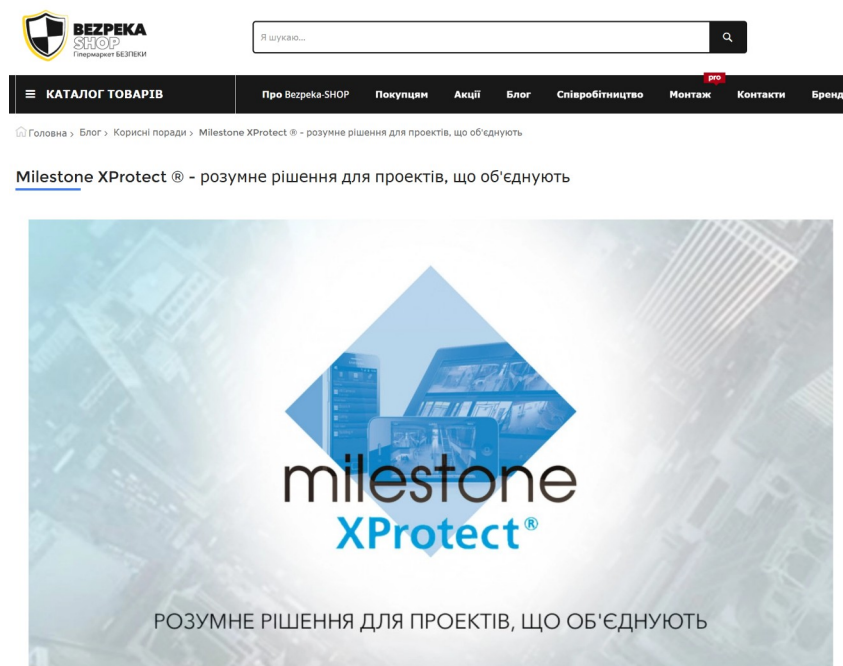


Рис.1.3. Головне вікно сервісу Milestone XProtect

Переваги Milestone XProtect:

- *Масштабованість.* Підходить як для малих, так і для великих установок з великою кількістю камер.
- *Сумісність.* Підтримує широкий спектр камер та периферійних пристроїв, що дозволяє легко інтегрувати нові технології.
- *Розширена аналітика.* Вбудовані функції детекції руху, розпізнавання номерних знаків, а також інші аналітичні можливості.
- *Висока надійність і безпека.* Підтримка алгоритмів для захисту даних та відновлення відеоархіву у разі збою.

- *Гнучкість в налаштуваннях.* Можливість кастомізації для адаптації до різних типів об'єктів та сценаріїв використання.
- *Інтеграція з іншими системами.* Підтримка інтеграції з іншими безпековими рішеннями, такими як системи управління доступом або охоронні сигналізації.
- *Віддалений доступ.* Можливість доступу до відео та налаштувань через мобільні додатки та веб-інтерфейси.
- *Підтримка великої кількості користувачів.* Можливість надання доступу різним користувачам з різними рівнями прав.

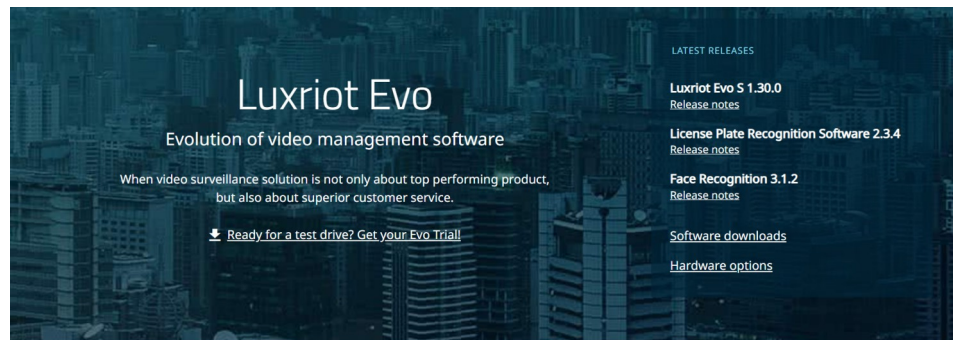
Недоліки Milestone XProtect:

- *Висока вартість.* Milestone XProtect – це преміум-рішення, яке потребує значних інвестицій для початкової установки та ліцензування.
- *Складність налаштування.* Налаштування системи може бути складним і потребує досвіду в роботі з відеоспостереженням та ІТ-інфраструктурою.
- *Високі вимоги до апаратного забезпечення.* Для ефективної роботи необхідні потужні сервери та інфраструктура для зберігання та обробки великої кількості відео.
- *Потреба в спеціалізованому обслуговуванні.* Для правильного функціонування системи часто потрібна підтримка.

Luxriot VMS – це високопродуктивне програмне забезпечення для відеоспостереження, яке дозволяє інтегрувати та керувати великою кількістю камер в рамках єдиної платформи. Luxriot VMS (Video Management Software) використовується для моніторингу, запису та управління відео в реальному часі, пропонуючи користувачам простоту, гнучкість та високу ефективність. Програмне забезпечення підтримує різні типи камер, включаючи IP-камери, PTZ-камери, а також камери з підтримкою високої роздільної здатності [15].

Luxriot VMS надає потужні інструменти для аналізу відео, що дозволяє ефективно використовувати систему для ідентифікації руху, розпізнавання об'єктів та виявлення підозрілих ситуацій. Вбудовані можливості для

детекції руху дозволяють автоматично запускати запис або надсилати сповіщення при виявленні змін у відео з камер. Програмне забезпечення також підтримує інтеграцію з іншими системами безпеки та забезпечує віддалений доступ до відеоматеріалів через мобільні додатки або веб-інтерфейс.



Video Management Software Luxriot EVO Editions

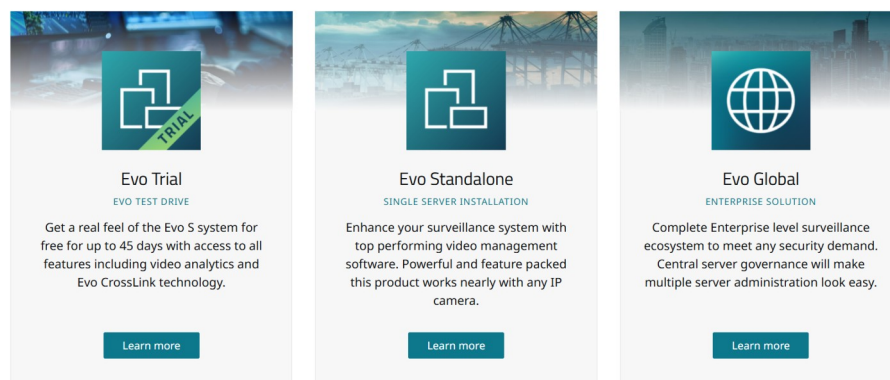


Рис.1.4. Головне вікно системи Luxriot VMS

Переваги Luxriot VMS:

- *Гнучкість у налаштуваннях.* Програмне забезпечення надає широкі можливості для кастомізації, що дозволяє налаштовувати систему відповідно до потреб користувача.
- *Масштабованість.* Luxriot VMS підходить як для малих, так і для великих корпорацій з численними камерами.
- *Інтеграція з іншими системами.* Luxriot VMS може інтегруватися з іншими системами безпеки, такими як системи охорони, управління доступом, а також з аналітичними платформами.

- *Вбудована аналітика.* Програма має вбудовані можливості для аналізу відео, включаючи детекцію руху, виявлення об'єктів, а також розпізнавання осіб і номерних знаків.
- *Віддалений доступ.* Luxriot VMS дозволяє здійснювати моніторинг і управління системою через мобільні додатки або веб-інтерфейс з будь-якої точки світу.
- *Надійність і безпека.* Програмне забезпечення забезпечує високу надійність збереження відеоархівів та захисту даних від несанкціонованого доступу.
- *Підтримка різних платформ.* Luxriot VMS підтримує роботу на різних операційних системах, таких як Windows, і має версії для серверів і клієнтських комп'ютерів.

Недоліки Luxriot VMS:

- *Висока вартість.* Програмне забезпечення може бути дорогим для малих компаній або організацій з обмеженим бюджетом.
- *Складність у налаштуванні.* Для налаштування та інтеграції системи можуть бути потрібні технічні знання та досвід.
- *Високі вимоги до апаратного забезпечення.* Для забезпечення високої продуктивності та стабільності роботи системи необхідне відповідне апаратне забезпечення.
- *Потребує постійного технічного обслуговування.* Для забезпечення стабільної роботи та оновлення програмного забезпечення може знадобитися підтримка професіоналів.

Vivotek VAST – це комплексне програмне забезпечення для управління відеоспостереженням, розроблене компанією Vivotek. VAST пропонує ефективні рішення для моніторингу, запису та управління відео з численних камер, що використовуються в різних сферах, від малого бізнесу до великих підприємств та охоронних систем. Це програмне забезпечення підтримує інтеграцію з камерами Vivotek та іншими брендами, надаючи користувачам

широкий спектр можливостей для безпечного управління відеоархівами, а також для ефективної ідентифікації руху та аналізу відео [16].

Однією з ключових особливостей Vivotek VAST є підтримка технології відеоаналітики, яка дозволяє здійснювати автоматичну детекцію руху, виявлення об'єктів і аналіз ситуацій на основі відео. Програмне забезпечення також пропонує гнучкість у налаштуваннях і може використовуватися як для малих систем відеоспостереження, так і для великих інтегрованих рішень, що складаються з сотень камер. Vivotek VAST підтримує масштабованість, дозволяючи збільшувати кількість камер і підключених пристроїв без втрати продуктивності. Воно також має потужний інтерфейс для дистанційного моніторингу, який дозволяє користувачам переглядати відео в реальному часі через мобільні додатки або веб-інтерфейс, що забезпечує зручний доступ з будь-якої точки світу. Головне вікно представлено на рис.1.5.

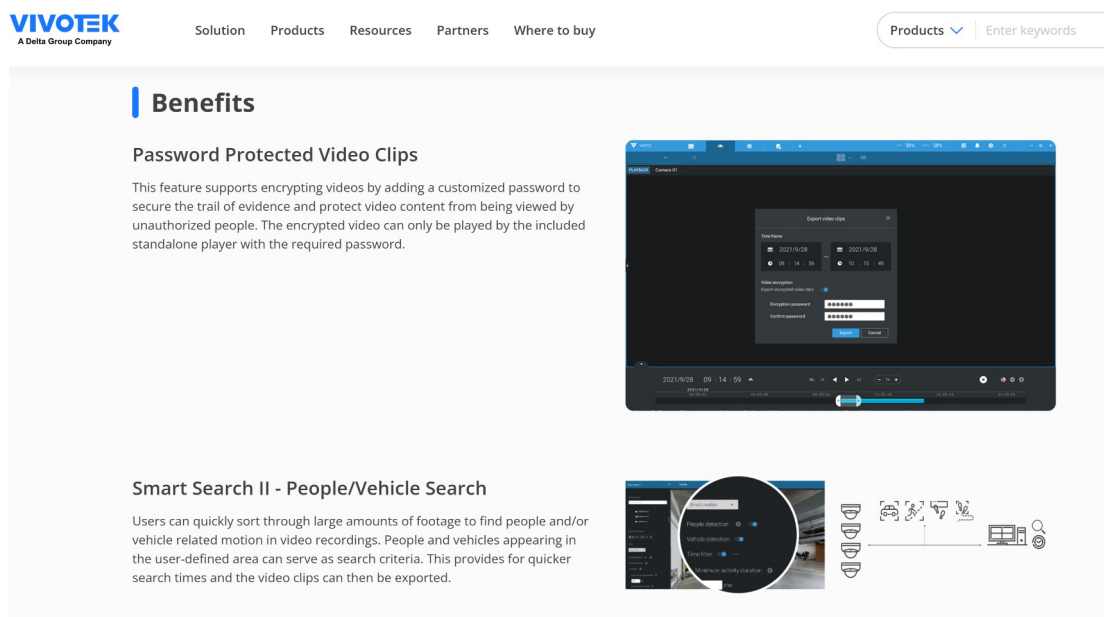


Рис.1.5. Головне вікно системи Vivotek VAST

Переваги Vivotek VAST:

- *Інтуїтивно зрозумілий інтерфейс.* Vivotek VAST має зручний інтерфейс, який дозволяє легко налаштовувати та управляти камерами, записами і відеоархівами.

- *Висока масштабованість.* Підтримка великих систем відеоспостереження з великою кількістю камер, що робить програму підходящою для великих підприємств та охоронних систем.
- *Потужні можливості аналітики.* Включає в себе вбудовану відеоаналітику для детекції руху, виявлення об'єктів, а також для аналізу подій у відео.
- *Віддалений доступ.* Можливість здійснювати моніторинг та керувати системою через мобільні додатки або веб-інтерфейс.
- *Підтримка інтеграції з іншими системами.* Vivotek VAST може інтегруватися з іншими рішеннями безпеки, такими як системи управління доступом та сигналізації.
- *Висока надійність.* Програмне забезпечення має вбудовані функції для забезпечення безперебійної роботи та захисту даних.

Недоліки Vivotek VAST:

- *Вартість.* Для малих підприємств чи домашніх користувачів вартість може бути досить високою.
- *Залежність від камер Vivotek.* Хоча система підтримує камери інших брендів, найкраща продуктивність досягається при використанні камер Vivotek.
- *Складність налаштування.* Для користувачів, які не мають досвіду в налаштуванні таких систем, можуть виникнути труднощі при інтеграції та налаштуванні програми.
- *Високі вимоги до апаратного забезпечення.* Програмне забезпечення може вимагати потужних комп'ютерних ресурсів для забезпечення стабільної роботи великих систем.

Camlytics – це програмне забезпечення для відеоаналітики, яке підтримує широкі можливості для відеоспостереження, виявлення руху та аналізу відео. Воно спеціалізується на інтеграції з різними камерами і забезпечує автоматичний аналіз відеопотоку для виявлення різноманітних подій, таких як рух, зміна сцени, а також для розпізнавання об'єктів.

Camlytics має зручний інтерфейс і пропонує потужні інструменти для створення автоматизованих систем відеоспостереження, що допомагають підвищити рівень безпеки і знизити навантаження на персонал [17].

Основною особливістю Camlytics є потужна система відеоаналітики, яка дозволяє проводити детекцію руху, визначати кількість людей або транспортних засобів у визначених зонах, а також використовувати алгоритми для моніторингу поведінки людей у реальному часі. Camlytics також підтримує аналіз подій на основі зображення і може відправляти сповіщення у разі виявлення будь-яких значущих змін у відеопотоці. Camlytics може бути інтегрований з існуючими системами відеоспостереження та використовувати їх для автоматичного збору даних. Він також підтримує роботу з різними камерами, включаючи IP-камери, CCTV і інші типи камер, що робить його універсальним рішенням для створення систем відеоспостереження. Система дозволяє здійснювати моніторинг через веб-браузери та мобільні додатки, що дає змогу керувати відеоархівами та переглядати відео з будь-якої точки світу. Головне вікно представлено на рис.1.6.

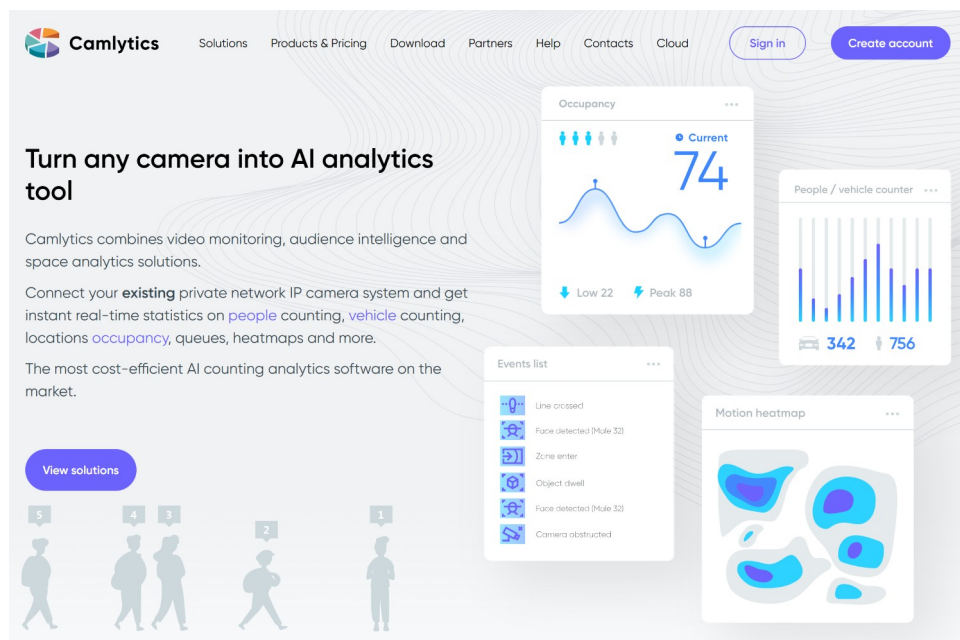


Рис.1.6. Головне вікно системи Camlytics

Переваги Camlytics:

- *Інтуїтивно зрозумілий інтерфейс.* Простий у використанні інтерфейс для налаштування та керування системою.
- *Сумісність з камерами.* Підтримка різних типів камер, таких як IP-камери, CCTV, а також інтеграція з існуючими системами відеоспостереження.
- *Відеоаналітика в реальному часі.* Потужна система відеоаналітики для виявлення руху, аналізу поведінки та розпізнавання об'єктів.
- *Можливість отримувати сповіщення.* Автоматичні сповіщення про події, що дозволяють оперативно реагувати на зміни.
- *Гнучкість налаштувань.* Можливість налаштування зони спостереження, рівня чутливості та інших параметрів для детектування подій.
- *Дистанційний доступ.* Підтримка віддаленого доступу через мобільні додатки та веб-інтерфейс.

Недоліки Camlytics:

- *Високі вимоги до апаратного забезпечення.* Для правильної роботи з великою кількістю камер може знадобитися потужне обладнання.
- *Ціна.* Camlytics може бути дорогим для невеликих підприємств або користувачів з обмеженим бюджетом.
- *Обмежена підтримка деяких виробників камер.* Хоча система підтримує більшість камер, деякі моделі можуть не працювати належним чином через специфічні вимоги.
- *Висока складність налаштування для новачків.* Налаштування відеоаналітики може вимагати певного досвіду, що може бути складним для новачків.

На основі проведеного аналізу здійснено порівняльну характеристику відомих засобів дистанційного відеоспостереження та ідентифікації руху. Результати наведені в табл.1.1.

Таблиця 1.1

Порівняльна характеристика програмних застосунків

Критерій	Blue Iris	iSp y	Milestone XProtect	Luxriot VMS	Vivotek VAST	Camlytics
Вартість	-	+	-	-	-	+
Підтримка складних систем розпізнавання об'єктів	-	-	+	+	+	-

Продовження таблиці 1.1

Потреба у потужному апаратному забезпеченні	-	-	+	+	+	-
Налаштування та управління	+	+	-	-	-	+
Масштабованість	+	+	+	+	+	+
Гнучкість інтеграції з іншими системами	+	+	+	+	+	+
Підтримка різних типів камер	+	+	+	+	+	+
Підтримка відеоаналітики	+	+	+	+	+	+
Мобільний доступ	+	+	+	+	+	+
Автоматичні сповіщення	+	+	+	+	+	+
Легкість у використанні	+	+	-	+	+	+
Залежність від інтернет-з'єднання	-	-	+	+	+	+

Джерело: складено автором за матеріалами [12-17]

Були застосовані такі критерії порівняння:

- *Вартість.* Програми, такі як iSpy та Camlytics, є доступними за ціною, що робить їх привабливими для малих підприємств та домашнього використання. Натомість Milestone XProtect мають високі початкові витрати, що робить їх менш доступними для обмежених бюджетів.

- *Складність налаштування та управління.* Програми, як Blue Iris та iSpy, є більш простими в налаштуванні та використанні, що робить їх оптимальними для користувачів з мінімальним досвідом у відеоспостереженні. Milestone XProtect потребує значних знань і досвіду для налаштування, що може бути бар'єром для менш досвідчених користувачів.

- *Потужність апаратного забезпечення.* Програми, які використовують складні алгоритми розпізнавання об'єктів та глибокого навчання, така як Milestone XProtect, вимагають потужного апаратного забезпечення для належної роботи, що може бути обмеженням для менш потужних систем.

- *Інтеграція та масштабованість.* Всі програми мають хорошу підтримку масштабованості та інтеграції з іншими системами безпеки, що робить їх придатними для різноманітних сценаріїв, від малих до великих установ.

- *Мобільний доступ та відеоаналітика.* Більшість програм підтримують мобільний доступ і пропонують різноманітні можливості відеоаналітики, що важливо для дистанційного моніторингу. Однак програма, як Luxriot VMS, забезпечує більш просунуті можливості в області розпізнавання об'єктів та подій.

- *Залежність від інтернет-з'єднання.* Програми, як Milestone XProtect, вимагають стабільного та швидкого інтернет-з'єднання для передачі та обробки даних в реальному часі, що може бути проблемою в умовах обмеженого інтернет-з'єднання.

Висновки до першого розділу

З огляду на проведений аналіз існуючих програмних засобів для дистанційного відеоспостереження та ідентифікації руху, можна стверджувати, що ця задача є надзвичайно актуальною. Сучасні системи відеоспостереження часто мають складні функції, зокрема виявлення руху, аналіз відео та інтеграцію з іншими системами безпеки. Однак, незважаючи на широкий вибір доступних рішень, багато з них потребують значних фінансових витрат або спеціалізованих технічних знань для налаштування. Тому існує нагальна потреба у розробці інформаційної системи, яка буде оптимально поєднувати доступність, простоту використання та високу ефективність у виявленні руху. Це дозволить забезпечити високий рівень безпеки на різних об'єктах без великих затрат та складної інтеграції. Розробка такої системи буде корисною як для правоохоронних органів, так і для комерційних структур, що мають потребу в ефективному та надійному контролі за своїми об'єктами.

РОЗДІЛ 2

СИСТЕМНИЙ АНАЛІЗ ТА ПРОЄКТУВАННЯ СИСТЕМИ

2.1. Побудова дерева цілей

Критично важливо сформулювати головну мету розроблюваної системи, що дозволить чітко окреслити її функціональні можливості. Головним завданням проєктованої системи дистанційного відеоспостереження є створення зручного інтерфейсу для користувачів, що забезпечить доступ до камер спостереження, встановлених на охоронюваних об'єктах. Для більш точної конкретизації цієї мети застосовується метод дерева цілей [18].

Дерево цілей – це ієрархічна схема, що відображає логічний ланцюжок "ціль – завдання – заходи – ресурси". Воно дозволяє формалізувати процес ухвалення рішень та структурно організувати розробку програмного забезпечення. Цей метод широко застосовується у програмно-цільовому плануванні та управлінні комплексними проєктами, зокрема й у даному випадку. Після ідентифікації основних потенційних проблем у процесі розробки програмного забезпечення визначаються відповідні цілі проєкту. Всі можливі цілі системи відображаються у вигляді дерева цілей. Таким чином, ключовою метою проєкту є створення високоякісного, стабільного та ефективного програмного продукту. Досягнення цього результату можливе лише за умови розробки добре продуманого функціоналу, створення зручного інтерфейсу та забезпечення надійності програмного забезпечення, що виключає помилки та збої у його роботі. Дерево цілей системи приведено на рис.2.1.

Створення системи для дистанційного відеоспостереження та ідентифікації руху є кінцевою та основною метою проєкту. Однак для її досягнення необхідно розбити цю мету на більш дрібні складові завдання, які формують багаторівневу структуру, організовану у вигляді ієрархічного

дерева цілей. Ця методика дозволяє послідовно та логічно спланувати реалізацію проекту, визначити ключові кроки та розподілити їх за рівнями складності та важливості. Відповідно до цього підходу, виконання підцілей нижнього рівня є обов'язковим для досягнення цілей вищого рівня. Таким чином, реалізація кожного наступного етапу проекту будується на основі вже виконаних завдань.

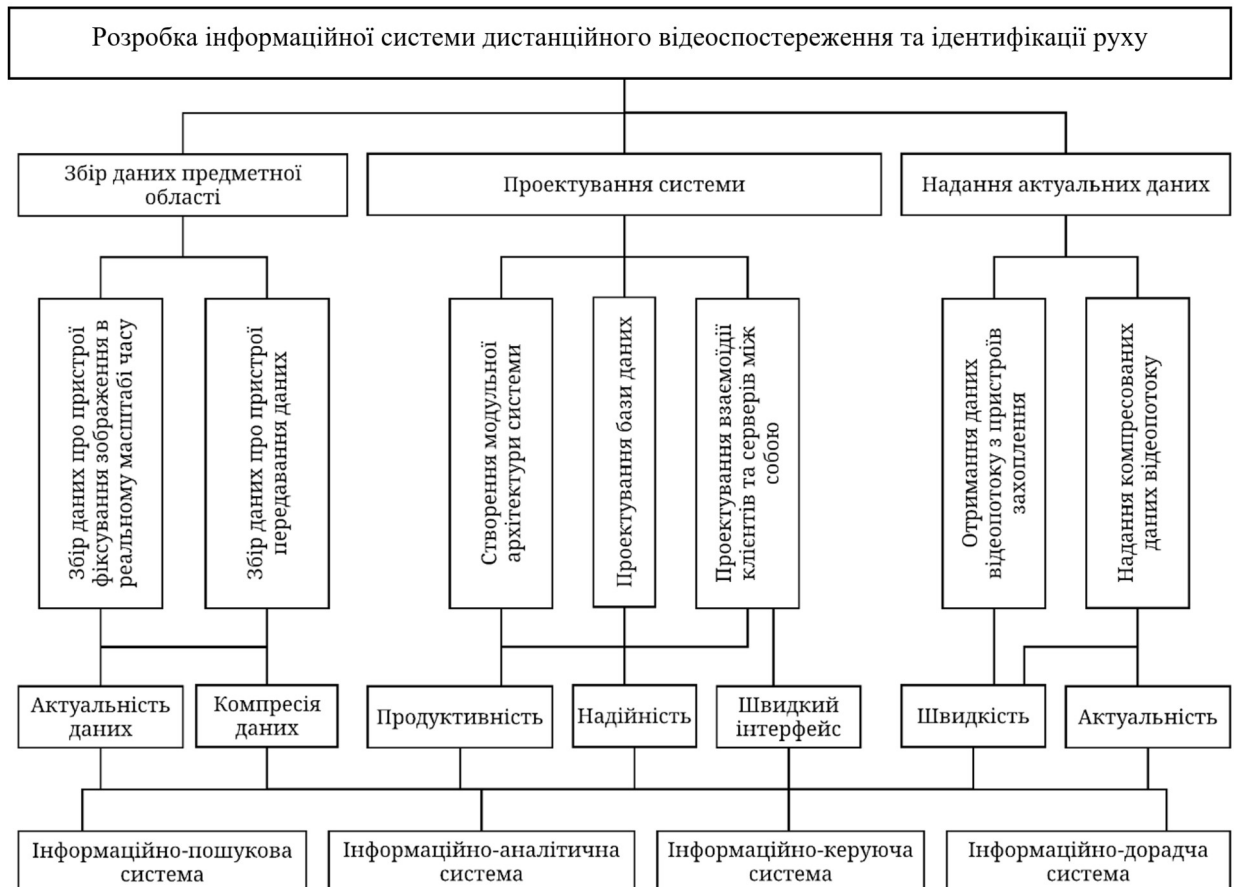


Рис.2.1. Дерево цілей системи

На першому рівні ієрархії можна виділити три ключові підцілі, які безпосередньо впливають на досягнення головної мети.

Перша підціль: «Збір даних предметної області». Початковим етапом розробки будь-якої складної інформаційної системи є ретельний аналіз предметної області, в якій вона буде функціонувати. У нашому випадку цей процес передбачає збір даних щодо існуючих систем відеоспостереження, їхньої архітектури, методів передачі відеопотоків, типів використовуваного обладнання та особливостей його функціонування. Головна мета цього

етапу – створення інформаційної бази, яка стане основою для подальшого проектування. Вивчаються можливості різних технічних рішень, способи їх інтеграції та підключення до програмного забезпечення. Зокрема, необхідно розглянути характеристики камер спостереження, принципи їхнього з'єднання з сервером, методи компресії та збереження відеопотоків. Також на цьому етапі оцінюються існуючі стандарти безпеки та вимоги до збереження даних, що є критично важливими для роботи у сфері правоохоронної діяльності. Адже інформація, що надходить із камер, повинна бути захищеною від несанкціонованого доступу та підробки.

Друга підціль: «Проектування системи». Після того, як зібрано достатньо інформації про предметну область, наступним етапом є проектування майбутньої системи. На цьому етапі визначається загальна архітектура програмного продукту, обираються відповідні технології, середовища розробки та програмні інструменти. Процес проектування включає кілька важливих аспектів:

- Архітектура системи – вибір клієнт-серверної або децентралізованої моделі, способів комунікації між компонентами, підходів до обробки та зберігання відеопотоків.
- База даних – розробка структури зберігання мета-даних про відеозаписи, логування подій та обліку користувачів.
- Користувацький інтерфейс – створення зручного та інтуїтивно зрозумілого інтерфейсу для взаємодії з системою.
- Протоколи безпеки – впровадження заходів для захисту інформації та уникнення несанкціонованого доступу.

Третя підціль: «Надання актуальних даних». Однією з головних функцій системи є забезпечення користувачів актуальною інформацією про стан об'єктів, що охороняються. Це означає, що дані з камер відеоспостереження повинні передаватися без значних затримок, незалежно від якості інтернет-з'єднання користувача.

Надання користувачам своєчасної та достовірної інформації є критично важливим, адже в умовах затримки передачі відео можливий ризик пропуску загрозливої ситуації на об'єкті. Наприклад, якщо система видаватиме застарілі кадри, зловмисники можуть встигнути завдати шкоди, а оператор не зможе вчасно зреагувати.

Визначення типу інформаційної системи. Побудоване дерево цілей дозволяє визначити, який саме тип інтелектуальної інформаційної системи доцільно реалізовувати для вирішення поставлених завдань. Розглядалися такі варіанти:

- Інформаційно-пошукова система – орієнтована на зберігання та швидкий пошук необхідної інформації.
- Інформаційно-аналітична система – здійснює аналіз великих масивів даних та надає звіти.
- Інформаційно-керуюча система – дозволяє автоматично керувати процесами на основі отриманих даних.
- Інформаційно-дорадча система – надає рекомендації та повідомлення про зміни у ситуації.

Аналізуючи особливості проектованої системи, було вирішено зупинитися на варіанті інформаційно-дорадчої системи. Така система дозволяє оперативно повідомляти користувача про зміни в обстановці на об'єкті охорони, використовуючи дані з камер відеоспостереження. Водночас саме користувач приймає рішення про необхідність реагування на отримані повідомлення. Таким чином, побудова інформаційно-дорадчої системи найкраще відповідає вимогам та задачам дистанційного спостереження. Вона забезпечує ефективну роботу користувача, дозволяючи йому отримувати актуальну інформацію про стан об'єкта у зручному форматі та з мінімальними затримками.

2.2. Проєктування системи

Схематичне зображення різних складових розроблюваної системи значно полегшує розуміння її структури та функціональності. Для цього використовується концептуальна модель, яка дозволяє відобразити ключові аспекти її розробки та функціонування. Процес створення такої моделі передбачає кілька важливих етапів, що включають постановку задачі створення моделей, формування гіпотез, визначення припущень, а також детальну формалізацію вимог до вхідних даних, параметрів та аргументів системи. Крім того, концептуальна модель передбачає пояснення вимог до системи, методів їх реалізації та змістовний опис її роботи. Для реалізації цього підходу використовується побудова UML-діаграм різних типів [19], які відображають структуру та поведінку системи, забезпечуючи ефективне проєктування інформаційної системи дистанційного відеоспостереження та ідентифікації руху.

Об'єктний підхід є ефективним для опису призначення системи дистанційного відеоспостереження за об'єктами охорони. Використання *діаграми варіантів використання* дозволяє чітко представити основні сутності, які беруть участь у роботі системи, а також дії, що можуть бути виконані за їх допомогою (рис.2.2).

Основними користувачами системи виступають адміністратор, який налаштовує систему відповідно до потреб та фактичного стану об'єктів кінцевого користувача, а також користувач – кінцевий користувач системи дистанційного відеоспостереження. Діаграма варіантів використання містить ключові функціональні можливості, серед яких налаштування IP сервера даних, реєстрація та вхід у систему, введення і оновлення особистих даних, додавання нових відеокамер, перегляд списку камер і відеопотоків. Крім того, діаграма варіантів використання дозволяє визначити, як взаємодіють користувачі з системою, які сценарії її використання можливі та як розподіляються обов'язки між адміністраторами та кінцевими користувачами.

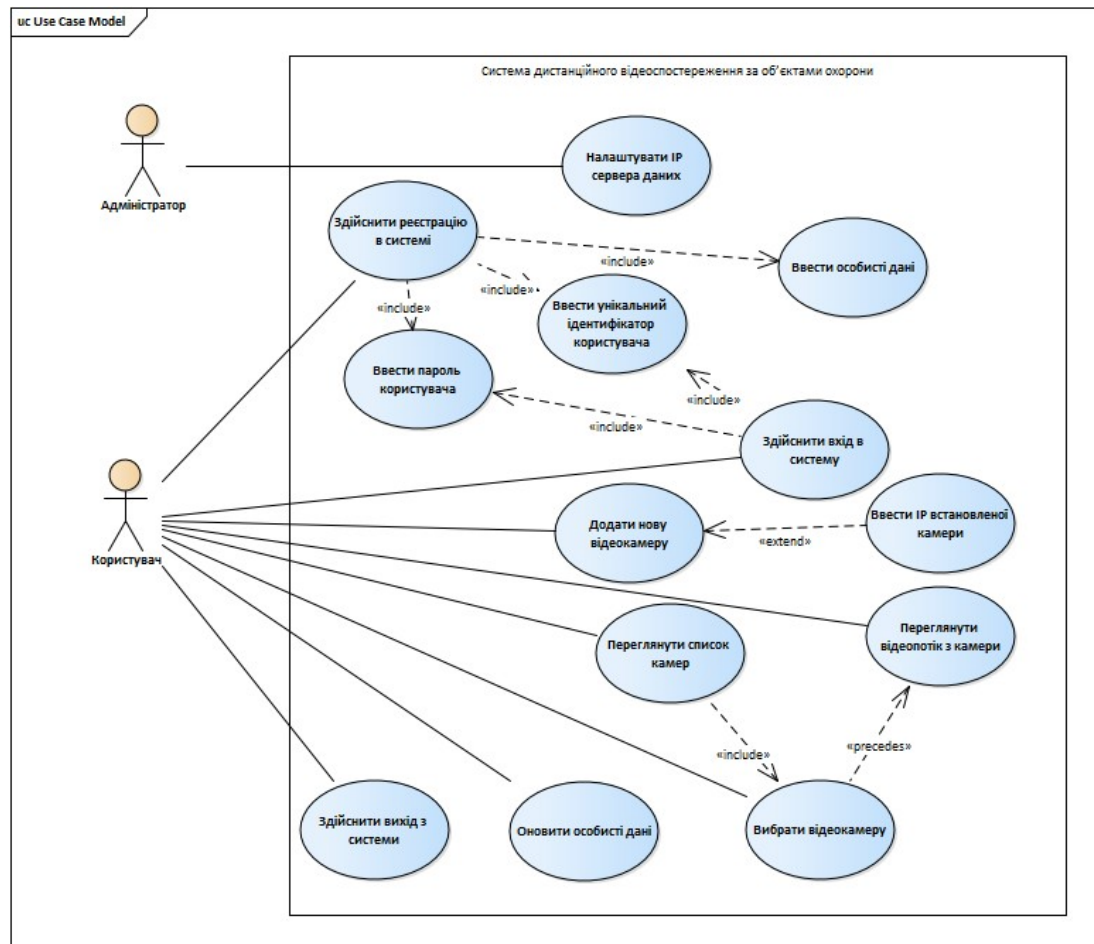


Рис. 2.2. Діаграма варіантів використання

Діаграма послідовності використовується для графічного відображення взаємодії об'єктів у часі. Вона дозволяє впорядковано представити всі процеси взаємодії між сутностями, які присутні в системі. Діаграма послідовності, створена для системи дистанційного відеоспостереження, деталізує набір послідовних дій користувача та системи, починаючи від авторизації в системі й закінчуючи переглядом відеопотоку з вибраної камери (рис.2.3).

Окремо побудована діаграма послідовності для прецеденту «Переглянути зображення з обраної відеокамери», у якій беруть участь такі об'єкти, як користувач, адміністратор, система та база даних. У цій діаграмі чітко відображено процеси обробки запитів, перевірки прав доступу, отримання даних із бази та передача результатів користувачеві.

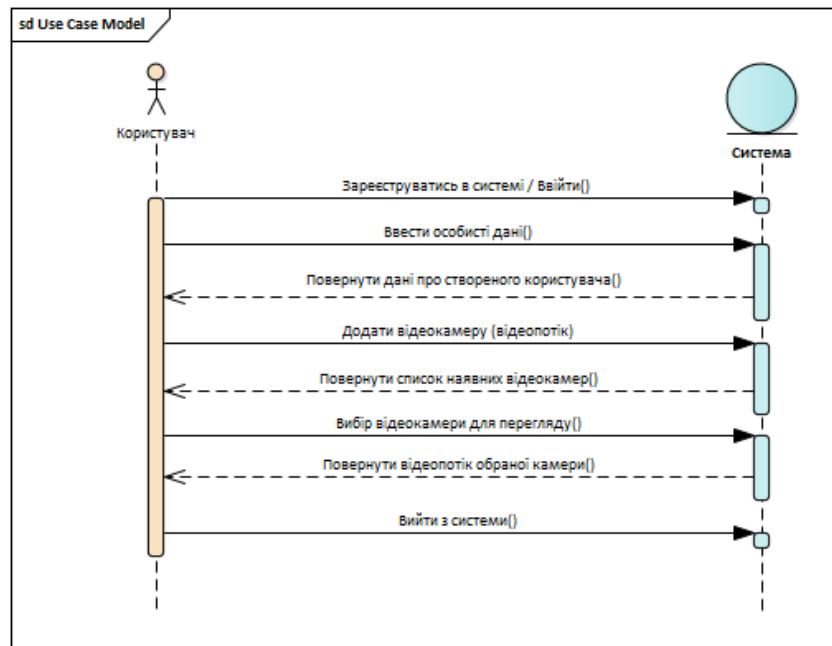


Рис. 2.3. Лінія життя актора «Користувач» в системі

Додатково, діаграма послідовності дозволяє аналізувати потенційні затримки у передачі даних, оптимізувати процеси обробки відеопотоків і передбачати можливі ситуації, пов'язані з втратою кадрів або порушенням зв'язку з сервером. Дану діаграму зображено на рисунку 2.4.

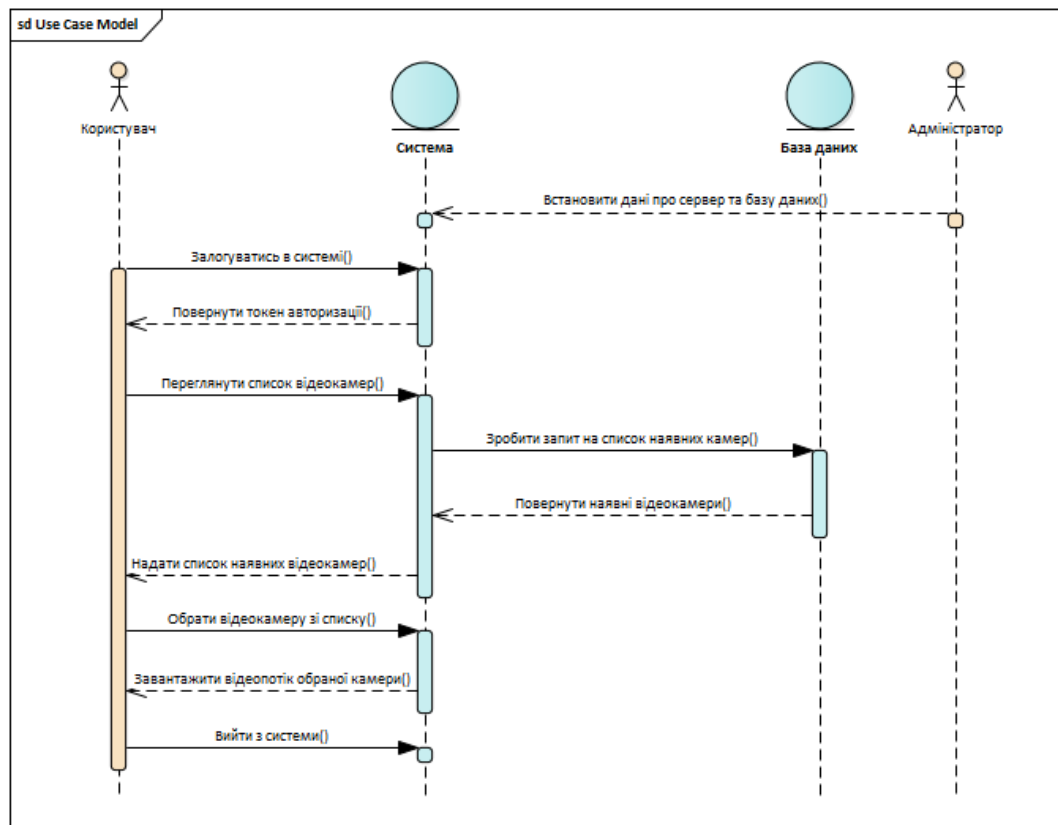


Рис. 2.4. Діаграма послідовності для прецеденту «Переглянути зображення»

На даній більш деталізованій діаграмі можна бачити зображення чотирьох об'єктів, як от Користувач, Адміністратор, Система та База даних. Кожен з цих об'єктів виконує свою роль у прецеденті «Переглянути зображення з обраної відеокамери», до прикладу База даних відповідає за своєчасне прийняття запиту на список наявних відеокамер, за його обробку та повернення результатів системі.

Для відображення ключових робочих процесів у системі використовується *діаграма діяльності* (рис.2.5).

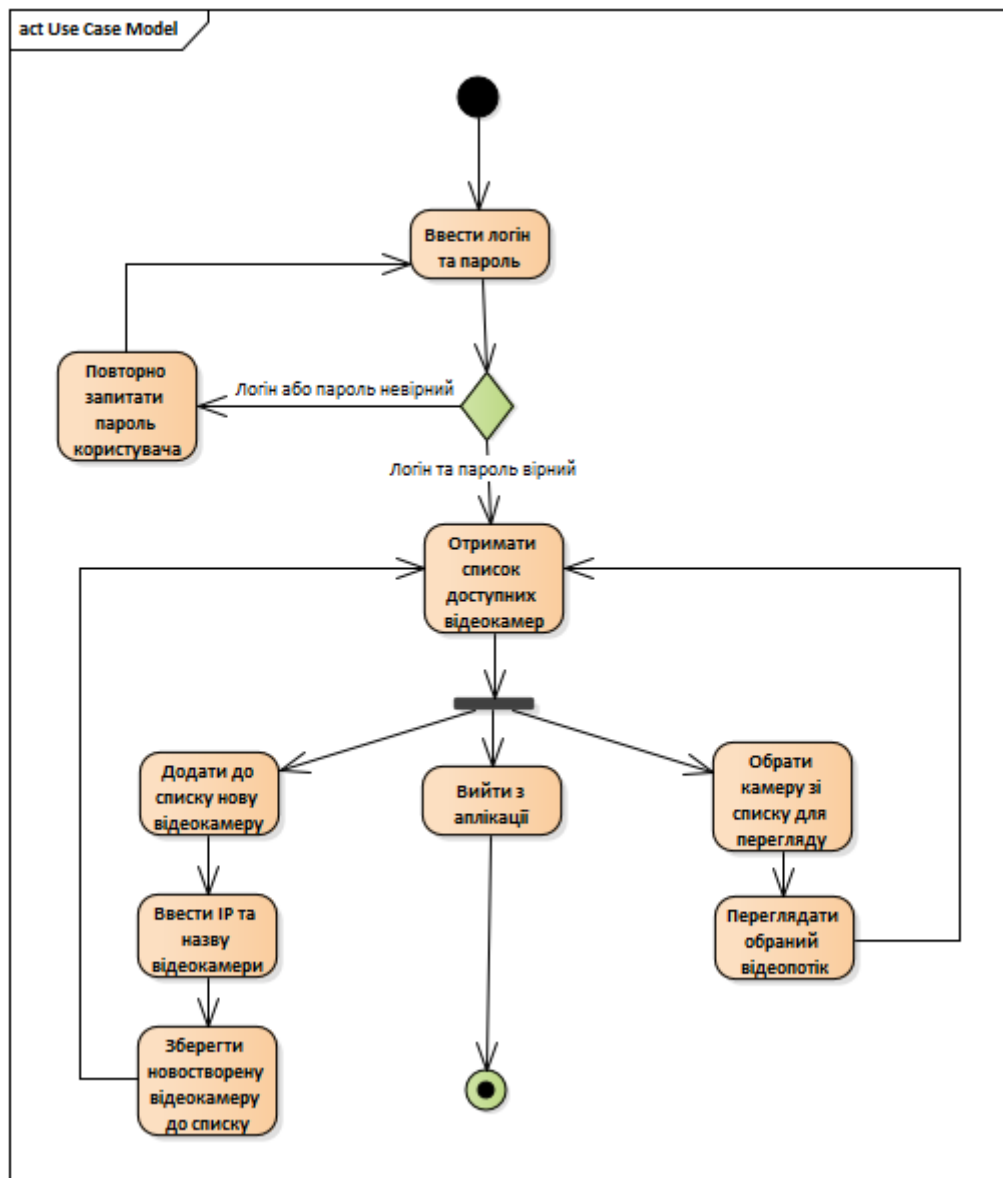


Рис. 2.5 Діаграма діяльності.

Цей вид діаграми дозволяє представити процеси в динаміці, показуючи переходи між станами системи у вигляді дій, які виконуються при зміні цих

станів. Вона дозволяє чітко простежити, які процеси відбуваються під час авторизації користувача, додавання відеокамер, перегляду відеопотоку та інших операцій. Крім того, діаграма діяльності дає змогу оцінити ефективність алгоритмів ідентифікації руху, що є ключовим аспектом розробки системи. У процесі аналізу цієї діаграми можна визначити вузькі місця у роботі системи та запропонувати варіанти оптимізації для підвищення швидкодії та точності аналізу відеопотоків. Перший етап роботи системи передбачає очікування введення логіна та паролю. Після введення користувачем своїх облікових даних відбувається перевірка коректності введених значень.

Якщо система виявляє невідповідність даних, вона повертається до початкового стану очікування введення логіна і пароля. У разі успішної авторизації відбувається завантаження списку доступних відеокамер, після чого користувач може переглянути зображення з вибраних пристроїв. У рамках ідентифікації руху система аналізує отримані відеопотоки в реальному часі, порівнює з еталонними моделями та може повідомляти адміністратора про підозрілі дії чи аномальні ситуації. Такий підхід значно підвищує ефективність відеоспостереження, дозволяючи оперативно реагувати на можливі загрози.

Діаграма компонентів. Для відображення системи з точки зору залежностей між її компонентами використовується UML-діаграма компонентів. У розробленій моделі системи дистанційного відеоспостереження визначено такі ключові елементи, як клієнтська частина програмного забезпечення (представлення), серверна частина, що відповідає за обробку запитів, а також бази даних клієнта і сервера (рис.2.6). Крім того, система передбачає наявність інтерфейсу API сервера, через який здійснюється взаємодія між клієнтською та серверною частинами. На діаграмі відображено залежності між цими компонентами, а також взаємозв'язки з такими артефактами, як файли конфігурації `web.xml` та програмні модулі `RemoteViewActivity.java`.

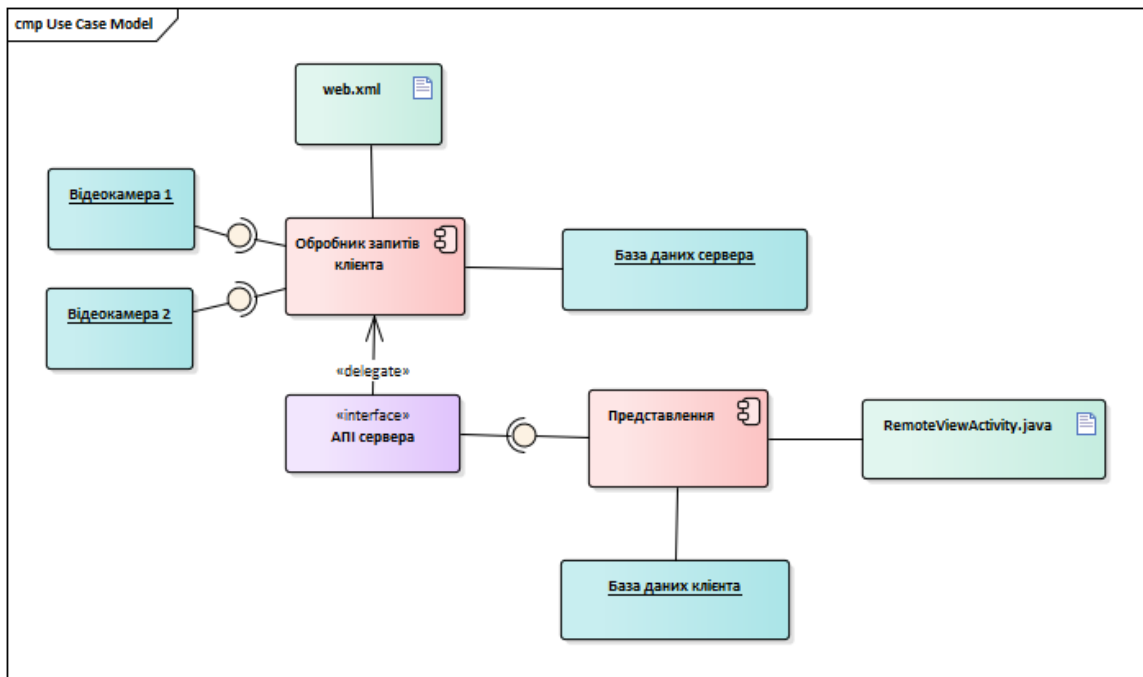


Рис. 2.6. Діаграма компонентів

Важливо зазначити, що структура системи розроблена з урахуванням можливості її масштабування, що дозволяє легко інтегрувати додаткові модулі, розширювати функціонал і забезпечувати підтримку нових алгоритмів обробки відеопотоків та ідентифікації руху.

2.3. Вибір та обґрунтування методів вирішення задачі

Розробка інформаційної системи дистанційного відеоспостереження та ідентифікації руху передбачає використання сучасних методів комп'ютерного зору та машинного навчання для аналізу відеопотоку в режимі реального часу. Одним із ключових аспектів такої системи є ефективне розпізнавання руху та ідентифікація об'єктів, що вимагає застосування відповідних алгоритмів та моделей.

Одним із найбільш ефективних методів для розпізнавання об'єктів у відеопотоці є використання дескриптора ознак гістограми орієнтованих градієнтів (HOG). Цей метод дозволяє аналізувати контури та форми об'єктів, що дає змогу системі визначати їхнє положення та ідентифікувати тип руху [25]. HOG широко використовується в комп'ютерному зорі та реалізований у таких бібліотеках, як OpenCV, що забезпечує зручну

інтеграцію з іншими модулями системи. Алгоритм HOG працює шляхом розбивки зображення на невеликі регіони, у межах яких обчислюються градієнти яскравості та їхні орієнтації. Ці градієнти формують гістограми, що дозволяють описати структуру об'єкта. Основні етапи обчислення HOG включають попередню обробку зображення, розрахунок градієнтів, визначення величини та орієнтації кожного пікселя, а також побудову гістограм. У результаті система отримує числове представлення об'єкта, яке можна використовувати для подальшої класифікації. Алгоритм складається з трьох основних кроків: попередня обробка даних, розрахунок градієнтів, обчислення величини та орієнтації [26].

Попередня обробка даних є ключовим етапом у процесі машинного навчання. Перш ніж розпочати аналіз, необхідно підготувати зображення, зокрема змінити його пропорції до співвідношення 1:2. Оптимальним є розмір 64x128 пікселів, що дозволяє розділити зображення на блоки розміром 8x8 або 16x16 пікселів. На наступному етапі розраховуються градієнти для кожного пікселя зображення. Градієнт характеризує зміни яскравості в горизонтальному (x) та вертикальному (y) напрямках. Для його обчислення аналізується невелика ділянка зображення:

121	10	78	96	125
48	152	68	125	111
145	78	85	89	65
154	214	56	200	66
214	87	45	102	45

Рис. 2.7. Піксельна матриця для фрагменту

Щоб визначити градієнт у напрямку x, обчислюється різниця між значенням пікселя справа і пікселем зліва від вибраної точки. Аналогічно, градієнт у напрямку y отримується як різниця між значеннями пікселів, розташованих вище та нижче від заданої точки. Для конкретного пікселя з інтенсивністю 85 результати обчислень будуть такими:

- Зміна в напрямку X (G_x) = 89 – 78 = 11
- Зміна напрямку Y (G_y) = 68 – 56 = 8

Ці значення формують дві матриці: одна містить градієнти по X, інша – по Y. Найбільші значення градієнтів зазвичай фіксуються по краях об'єктів на зображенні. Наступним кроком є визначення величини градієнта та його орієнтації за допомогою теореми Піфагора [27].

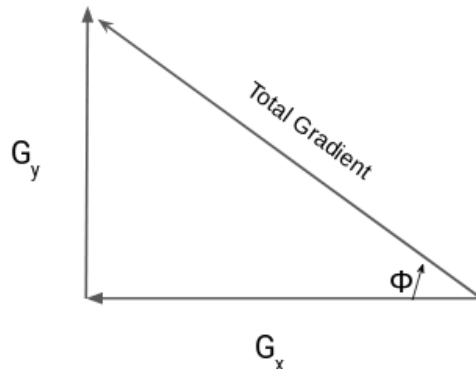


Рис. 2.8. Геометричний вигляд розрахунку градієнту

Враховуючи, що градієнти є перпендикулярними, загальна величина градієнта розраховується за формулою:

$$\text{Загальна величина градієнта} = \sqrt{G_x^2 + G_y^2} \quad (2.1)$$

Для обраного пікселя загальна величина градієнта = $\sqrt{121+64} = 13,6$

Щоб визначити орієнтацію (або напрямок) для того самого пікселя використаємо тангенс для кута ϕ як зображено на рис. 2.8:

$$\tan(\phi) = \frac{G_x}{G_y} \quad (2.2)$$

Отже, значення кута буде:

$$\phi = \text{atan}\left(\frac{G_x}{G_y}\right) \quad (2.3)$$

У нашому прикладі $\phi \approx 36$ градусів. Після обчислення всіх значень будується гістограма градієнтів, що описує контури зображення у вигляді набору числових характеристик. На наступному етапі гістограми всіх сегментів зображення об'єднуються у єдиний вектор. Наприклад, якщо розмір зображення 8×16 сегментів, а кожна гістограма містить 20 бінів, отримаємо масив із 2560 чисел. Отримані дані можуть бути використані в алгоритмах машинного навчання для прийняття рішень, наприклад, для розпізнавання об'єктів. Оскільки обсяг характеристик є значним, а кількість

навчальних зображень обмежена, доцільно використовувати машину опорних векторів (SVM).

SVM (Support Vector Machine) – це модель машинного навчання, що використовується для класифікації даних. Вона будує гіперплощину, яка розділяє різні класи даних. Для коректної роботи алгоритму необхідні навчальні вибірки, що містять позитивні та негативні приклади. У нашому випадку позитивні приклади – це зображення з людьми, а негативні – без них. Процес навчання включає три основні етапи: екстракція ознак, навчання моделі та виявлення об'єкта. Під час навчання модель отримує здатність класифікувати цільовий об'єкт, а на етапі виявлення застосовується алгоритм, який аналізує нові зображення та розпізнає потрібні об'єкти [28].

Для багатьох застосувань можна використати готові навчальні моделі, що доступні у відкритих бібліотеках. Це особливо корисно, коли немає потреби адаптувати модель під вузькоспеціалізовані завдання.

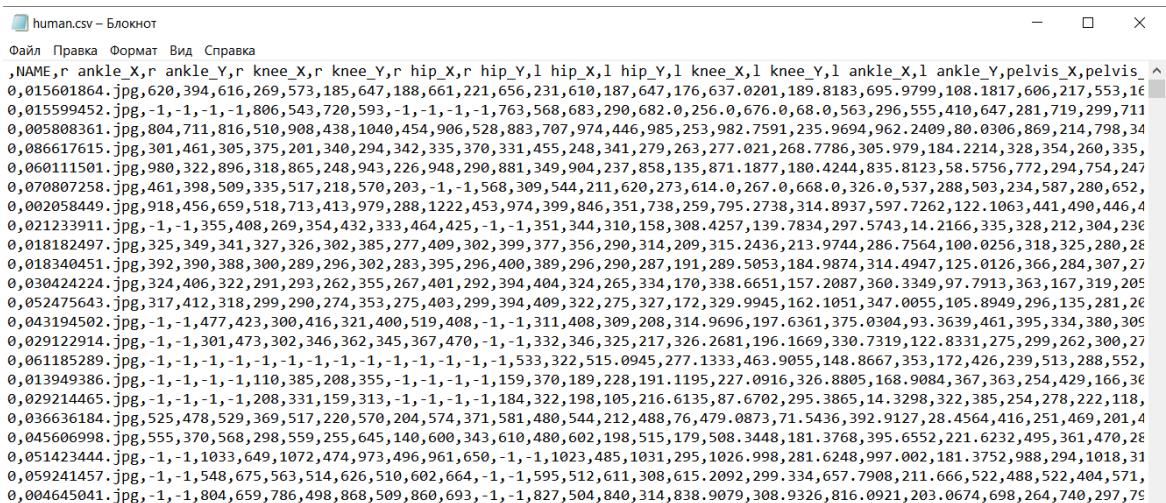


Рис. 2.9. Вигляд файлу з навчальною моделлю людини

Таким чином, поєднання HOG, SVM та сучасних методів глибокого навчання дозволяє створити ефективну інформаційну систему дистанційного відеоспостереження, здатну розпізнавати об'єкти, аналізувати їхній рух та забезпечувати високий рівень автоматизації процесів моніторингу.

2.4. Вибір засобів розроблення

Для створення інформаційної системи дистанційного відеоспостереження та ідентифікації руху необхідно вибрати засоби, які забезпечать повну реалізацію функціоналу, зручність у розробці та подальшій експлуатації, а також мінімізують технічні труднощі. З огляду на потребу в мобільності, клієнтську частину системи вирішено реалізувати як додаток для однієї з популярних мобільних операційних систем. Сьогодні лідерами ринку є Android та iOS, які разом охоплюють переважну більшість користувачів, тоді як інші платформи, за даними досліджень, займають лише 0,3% ринку мобільних пристроїв. Тому аналіз зосереджено саме на цих двох системах.

Android – це операційна система, розроблена для смартфонів, планшетів і нетбуків, яка базується на модифікованому ядрі Linux. У 2005 році Google придбав компанію Android Inc., а згодом створив Android Open Source Project (AOSP) для розвитку платформи. Перший пристрій із цією системою, T-Mobile G1 від HTC, з'явився у 2008 році. Android вирізняється відкритим API, що спрощує доступ до базових функцій через стандартні виклики, можливістю заміни системних додатків сторонніми рішеннями та зручним SDK із емулятором і інструментами для швидкої розробки [29].

iOS – це операційна система від Apple, створена для iPhone, iPad та iPod Touch. Представлена у 2007 році для iPhone, вона згодом розширилася на інші пристрої компанії. У 2017 році App Store налічував понад 2 мільйони додатків, із яких значна частина адаптована для iPad, а загальна кількість завантажень перевищила 130 мільярдів. iOS використовує інтерфейс із підтримкою багатоконтактних жестів (перетягування, дотики, затискання), а також акселерометри для реакції на струс чи обертання пристрою. Система має розширені функції доступності для людей із вадами зору чи слуху. Остання версія, iOS 17 (на березень 2025 року), підтримує сучасні вимоги до продуктивності та безпеки.

Для розробки під ці платформи розглянуто мови програмування Java та Kotlin. Java, створена Sun Microsystems у 1995 році Джеймсом Гослінгом, стала основою платформи Java SE. Її переваги включають об'єктно-орієнтований підхід, незалежність від платформи завдяки JVM, простоту освоєння, безпеку через шифрування, портативність, багатопотоковість і високу продуктивність із JIT-компілятором. Kotlin – сучасна мова від JetBrains, представлена як офіційна для Android на Google I/O у 2017 році. Вона сумісна з Java, лаконічна, безпечна, забезпечує швидшу компіляцію та скорочує код на 40% порівняно з аналогами.

Серед інтегрованих середовищ розробки (IDE) для мобільних додатків обрано Android Studio, офіційне середовище для Android на базі IntelliJ IDEA від JetBrains. Воно замінило застарілий Eclipse із плагіном ADT, який Google більше не підтримує. Android Studio, остання стабільна версія якого – 2024.1, пропонує емулятор, розширені інструменти налагодження та підтримку Kotlin і Java. Альтернатива, Android Developer Tools (ADT) для Eclipse, поступається за актуальністю та функціоналом.

Для автоматизації збірки проектів розглянуто Apache Ant, Maven і Gradle. Apache Ant, хоч і гнучкий, застарів і не підходить для сучасних вимог. Maven, орієнтований на Java-проекти, використовує XML для опису залежностей і порядку збірки, але поступається за гнучкістю. Gradle, із мовою на базі Groovy, став основним вибором для Android Studio завдяки підтримці великих багатомодульних проектів, інкрементальній збірці та високій швидкості. Його версія 8.5 забезпечує оптимальну продуктивність.

Для хостингу системи проаналізовано хмарні платформи Heroku та Microsoft Azure [30]. Heroku підтримує широкий спектр мов (Java, Python, Node.js тощо), але має обмеження у безкоштовному доступі. Microsoft Azure, із глобальною мережею дата-центрів, пропонує моделі PaaS та IaaS, гнучке масштабування й оплату за спожиті ресурси. На 2025 рік Azure вирізняється безкоштовними опціями для студентів і розробників, що робить його привабливим вибором.

Системи керування базами даних обрано з врахуванням архітектури системи. Для серверної частини підходить MySQL – реляційна СКБД від Oracle із версією 8.0 (2025 рік), що підтримує транзакції, реплікацію та повнотекстовий пошук. Для клієнтської частини на Android оптимальною є SQLite – вбудована, легка база даних із підтримкою SQL, яка інтегрована в ОС і доступна у версії 3.45 [31].

Після аналізу обрано Android як платформу через її поширеність, доступність пристроїв і відсутність потреби в macOS, необхідної для iOS-розробки через Xcode. Java стала основною мовою програмування завдяки універсальності, надійності та підтримці як клієнтської, так і серверної частин. Android Studio забезпечить розробку клієнтського додатка, а IntelliJ IDEA – серверного модуля. Gradle використано для клієнтської збірки, а Maven – для серверної через їхню адаптацію до відповідних задач. Хостинг на Microsoft Azure обрано за надійність і економічність, тоді як MySQL і SQLite оптимізують роботу з даними на сервері та клієнті відповідно.

Технічні характеристики обраних засобів

- Android версія 14 із багатовіконністю, покращеним режимом енергозбереження, групуванням сповіщень і JIT-компілятором для швидкості.
- Android Studio версія 2024.1 із емулятором, підтримкою Java 22, Kotlin 1.9, профілюванням продуктивності та адаптивним інтерфейсом.
- MySQL версія 8.0 із трикратним прискоренням, підтримкою JSON і багатоядерною реплікацією.
- SQLite версія 3.45, легка й надійна, із 100% покриттям тестами.
- Gradle версія 8.5, швидка й гнучка для Android-проектів.
- Jetty версія 11 як серверний контейнер із підтримкою Java 22, асинхронним HTTP і малим споживанням пам'яті.

Апаратні вимоги:

- Процесор: MediaTek Dimensity 6020 (8 ядер, 2.2 ГГц).
- Оперативна пам'ять: 4 ГБ.

- ОС: Android 14 або вище.

Такий вибір засобів забезпечує ефективну розробку системи дистанційного відеоспостереження та ідентифікації руху з урахуванням сучасних вимог і доступності для широкої аудиторії.

Висновок до другого розділу

В даному розділі було проведено детальне дослідження задачі, визначено її сильні та слабкі сторони, що дозволило покращити продукт ще на етапі проектування. В ході роботи було побудовано дерево цілей проекту, яке чітко визначає основну мету. Додатково система включає модулі ідентифікації руху, що дозволяють автоматично аналізувати відеопотоки та виявляти нестандартні ситуації, які можуть свідчити про потенційну загрозу. Такий підхід підвищує рівень автоматизації процесів та зменшує навантаження на операторів спостереження. Таким чином, здійснене проектування стане надійним фундаментом для подальшої розробки програмного продукту, забезпечуючи його високу ефективність та надійність у використанні.

РОЗДІЛ 3

ПРАКТИЧНА РЕАЛІЗАЦІЯ

3.1. Структура програмного забезпечення

Інформаційна система дистанційного відеоспостереження та ідентифікації руху розроблена на основі мови програмування Java, що забезпечує кросплатформенність та гнучкість у використанні. Для коректної роботи програми необхідна версія Java не нижче 6, проте рекомендовано використовувати версію 7 та вище із останніми оновленнями для підвищення продуктивності та безпеки. Система побудована по модульній структурі, що представлена у п'ять основних пакетів, що відповідають за різні функціональні аспекти: `dialog`, `entity`, `main`, `util` та `worker`.

Оскільки система призначена для ефективного віддаленого відеоспостереження, використання повноцінної бази даних не є обов'язковим. Замість цього для зберігання параметрів підключених пристроїв застосовується формат JSON, що забезпечує швидке зчитування та простоту редагування налаштувань.

При запуску система автоматично перевіряє наявність файлу конфігурації `device.json`. Якщо файл існує, програма зчитує параметри та здійснює спробу підключення до всіх зареєстрованих пристроїв. В середньому цей процес займає 5-7 секунд, після чого система готова до роботи. Крім основних налаштувань пристроїв, система також підтримує збереження історії відеозаписів. Відеоархів завантажується безпосередньо з відеореєстраторів, що дозволяє здійснювати ретроспективний аналіз подій та ідентифікацію об'єктів у кадрі.

З метою підвищення ефективності відеоспостереження та аналізу даних система може бути доповнена алгоритмами ідентифікації руху. Це дозволяє автоматично визначати зміну обстановки в кадрі, що критично важливо для правоохоронних органів.

Таким чином, розроблена інформаційна система не лише дозволяє віддалено керувати відеореєстраторами, а й надає можливості для інтелектуального аналізу відеоданих, що значно підвищує її цінність для правоохоронних структур.

3.2. Реалізація програмної системи

Програма розроблена для роботи на основних операційних системах – Windows, Linux та OS X, що успішно реалізовано. Рекомендується Java версії 7, а для OS X підтримується лише Apple Java 6 останньої версії – 1.6.0_65. Для функціонування необхідно встановити VLC-плеєр версії не нижче 2.1.4. Версії Java та VLC повинні відповідати архітектурі системи (32-бітна або 64-бітна), інакше програма не запуститься. Для перегляду відео через NAT на Windows потрібен Cygwin. Розроблена система призначена для моніторингу та управління відеореєстратором “MIKRO-KOD” [32]. Вона дозволяє налаштовувати пристрій, переглядати відео в різних режимах і якості, відстежувати аварійні події, переглядати архів, а також працювати з віддаленим відеореєстратором без статичної IP-адреси. Максимальна кількість камер – 16, кількість відеореєстраторів не обмежена. Головне вікно серверної частини програми показано на рис. 3.1.

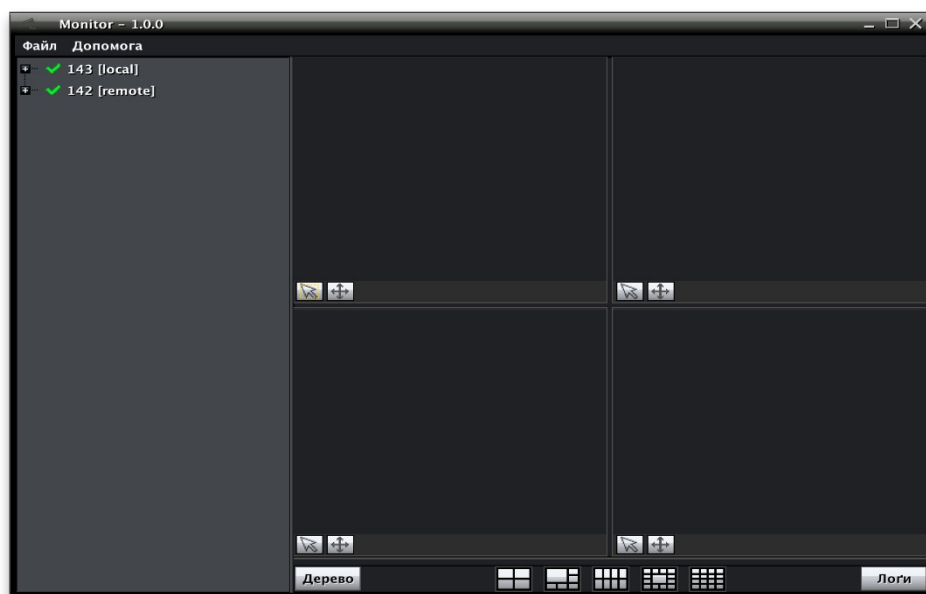


Рис. 3.1. Головне вікно серверної частини програми

При першому запуску необхідно додати відеореєстратори через меню “Файл” → “Додати відеореєстратор”. У вікні введіть назву, адресу, SSH-порт, порт камер, імена користувачів і паролі для пристрою та камер, а також укажіть керуючу камеру (рис. 3.2). Після збереження програма підключиться до пристрою: зелена галочка в дереві об’єктів позначить успіх, червоний хрестик – помилку. Для видалення пристрою клацніть на нього правою кнопкою миші, виберіть “Видалити пристрій”, підтвердіть дію – система видалить його та зупинить усі камери.

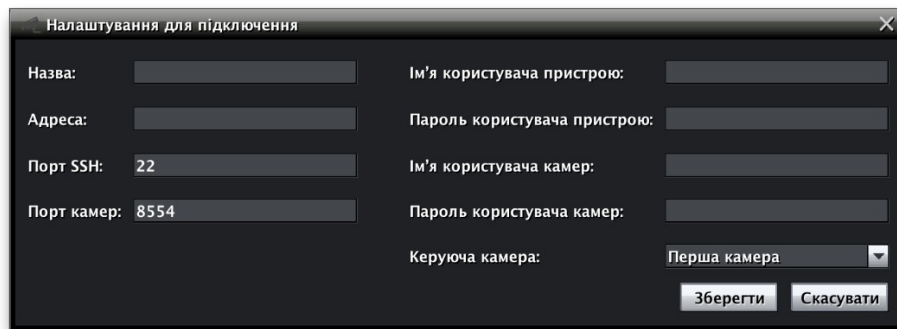


Рис. 3.2. Вікно додавання нового пристрою

Щоб переглянути відео, виберіть відеореєстратор, клацніть правою кнопкою миші та оберіть режим трансляції, наприклад, відео й звук у високій якості, лише відео низької якості чи лише звук високої якості. Для економії ресурсів система не дозволяє відкривати одну камеру в різних якостях одночасно (рис. 3.3). У разі втрати зв’язку програма автоматично перепідключається, повідомляючи користувача.

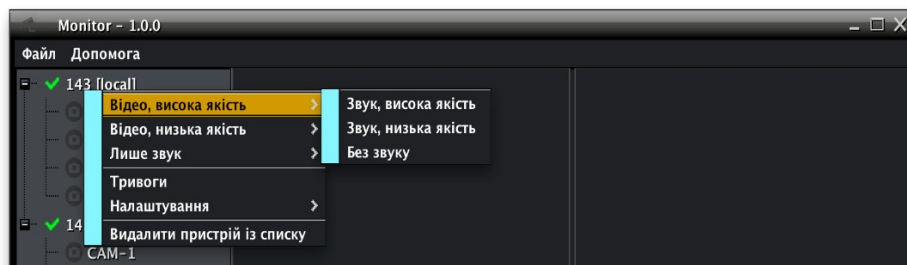


Рис. 3.3. Вибір режиму трансляції

3.2.1. Налаштування основних параметрів

Система підтримує гнучкі налаштування для роботи в повільних мережах або через 3G-модем, навіть без статичної IP-адреси. У налаштуваннях мережі задаються IP-адреса, маска підмережі та шлюз, а також активується трансляція через NAT (рис. 3.4). У режимі NAT для пристроїв із динамічною адресою (наприклад, через 3G) указується IP комп'ютера та UDP-порти, на які відеореєстратор передає потік. Якість відео регулюється пресетами “якість” (для запису) і “швидкість” (для перегляду), що оптимізує навантаження на мережу. Додаткові параметри відео доступні в меню. У налаштуваннях GSM задаються номери для SMS-сповіщень, наприклад, про аварійні ситуації.

The screenshot shows a software window titled "Налаштування камер і якостей" (Camera and Quality Settings). It contains several tabs: "Накладання дати" (Date Overlay), "Модем" (Modem), "Тривоги" (Alarms), "Пріоритети камер" (Camera Priorities), "Мережа" (Network), "Якість відео" (Video Quality), "GSM", and "Запис" (Recording). The "Мережа" tab is active, displaying the following configuration:

- IP: 192.168.56.143
- Маска: 255.255.255.0
- Шлюз: 0.0.0.0
- ☒ примусова відправка стріму з пристрою (відсутня постійна зовнішня IP адреса)
- Зовнішня IP адреса: [empty field]
- Тут можна вказати значення портів, які будуть пропонуватися за замовчуванням при перебуванні камери у режимі NAT.
- Перша камера: 11,111
- Друга камера: 22,222
- Третя камера: 33,333
- Четверта камера: 44,444

Рис. 3.4. Ввімкнення трансляції через NAT

Запис відео можливий у трьох режимах: постійний, за рухом або за розкладом. Налаштування застосовуються до кожної з чотирьох камер; запис лише відео чи аудіо вмикається або вимикається окремо (рис. 3.5). Розклад задається на тиждень із будь-якою кількістю інтервалів, де час завершення перевищує початок. Для запису за рухом обирається зона детекції

(прямокутник або квадрат) і чутливість. Збереження налаштувань кнопкою “Зберегти” зупиняє активні камери пристрою.

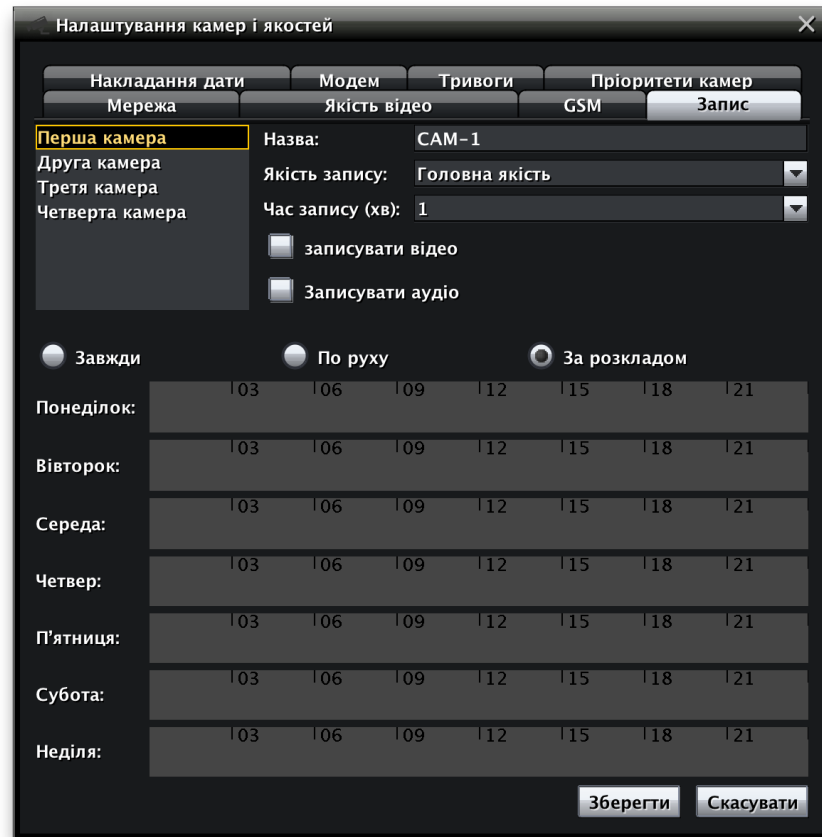


Рис. 3.5. Налаштування запису

3.2.2. Налаштування PPPoE

Для підключення через PPPoE необхідно клацнути правою кнопкою миші на пристрої, виберіть “Налаштування” → “Налаштування PPPoE” (рис. 3.6). Введіть логін і пароль; процес підключення видно у вкладці “Логи підключення”. Перевірка з’єднання доступна через вкладку “Пінг”. Для складних випадків увімкніть “Експертні налаштування” з додатковими параметрами.

Архівні записи доступні через “Налаштування” → “Історія” (правий клік на пристрої). У списку файлів оберіть потрібний, натисніть “Завантажити” (файл виділиться кольором), а подвійний клік відкриє його у VLC-плеєрі. Для видалення клацніть правою кнопкою миші та виберіть “Видалити локально” або “Видалити з пристрою”.

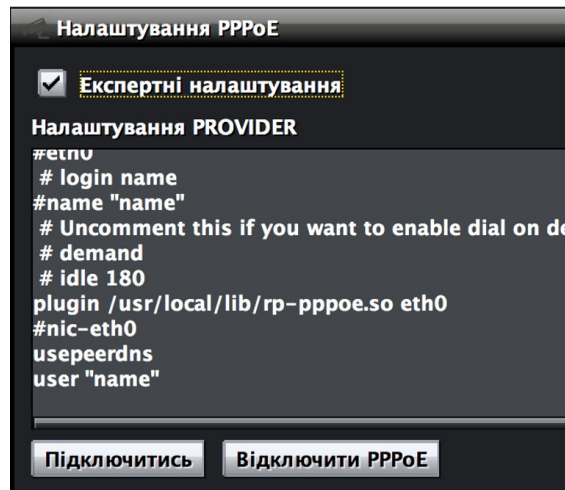


Рис. 3.6. Налаштування PPPoE

3.2.3. Додаткові параметри

Програма пропонує п'ять варіантів сітки для камер, автоматично адаптуючи розмір до їх кількості. Якщо сітка замала, система попереджає, і користувач закриває зайві камери. Вибір сітки – у нижній частині вікна, повноекранний режим – клавіша “F”. Журнал роботи відкривається кнопкою “Логи” (повторне натискання приховує), а кнопка “Дерево” активує компактний режим, розширюючи область камер. Інформація про камеру відображається кнопкою “I” у вікні над камерою. Налаштування живлення через “Налаштування” → “Налаштування живлення” → “Живлення камер” дозволяє відключати камери для економії (рис. 3.7). Зміна позицій камер виконується виділенням першої, потім другої камери – вони міняються місцями.

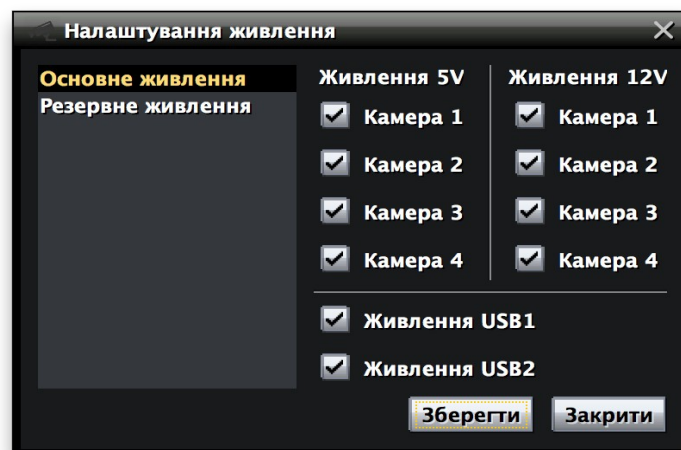


Рис. 3.7. Налаштування живлення камер

3.3. Аналіз контрольного прикладу

Крім основної програми був розроблений мобільний застосунок, функціональність якого наведемо на контрольному прикладі. Після запуску системи з'являється вікно входу, в яке необхідно ввести реєстраційні дані (рси.3.8).

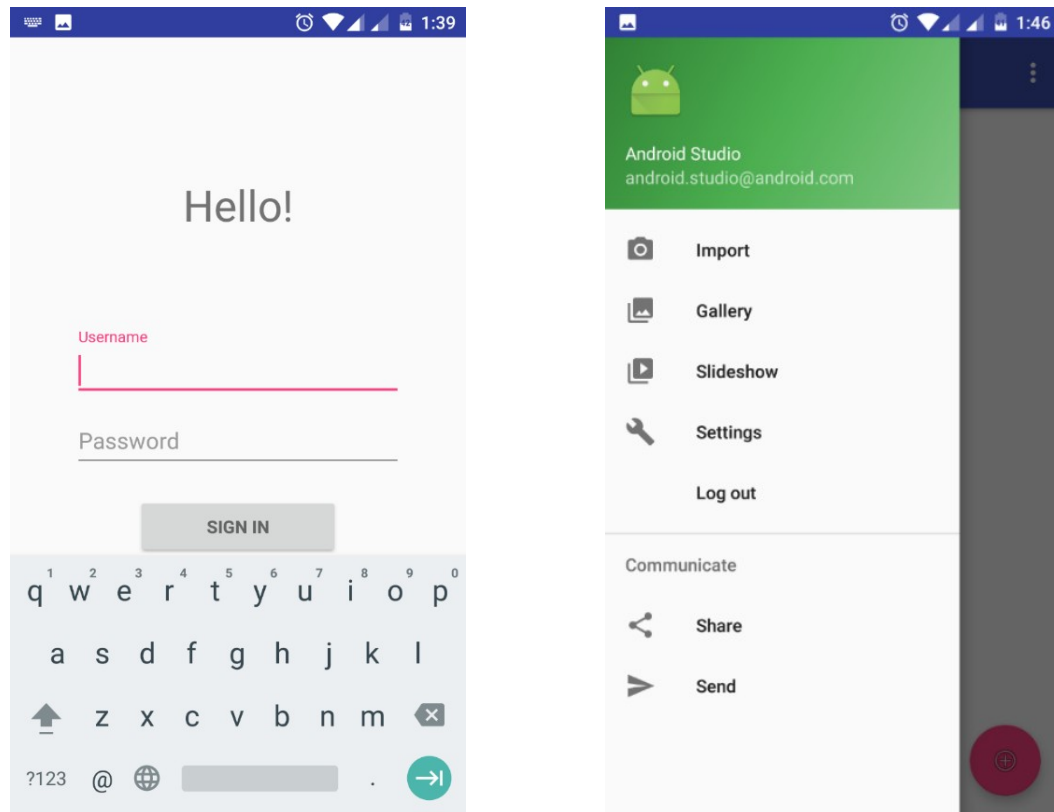


Рис. 3.8 Вікно авторизації та головного меню

На головному екрані розміщується основний контент системи – відеопотоки. Вони відображаються у вигляді невеликого прев'ю та назви відеопотоку. Це зроблено для зручної навігації між різними віддаленими камерами.

В головному меню виконується навігація між основними частинами програми: відеопотоками, шарінгом, налаштуваннями.

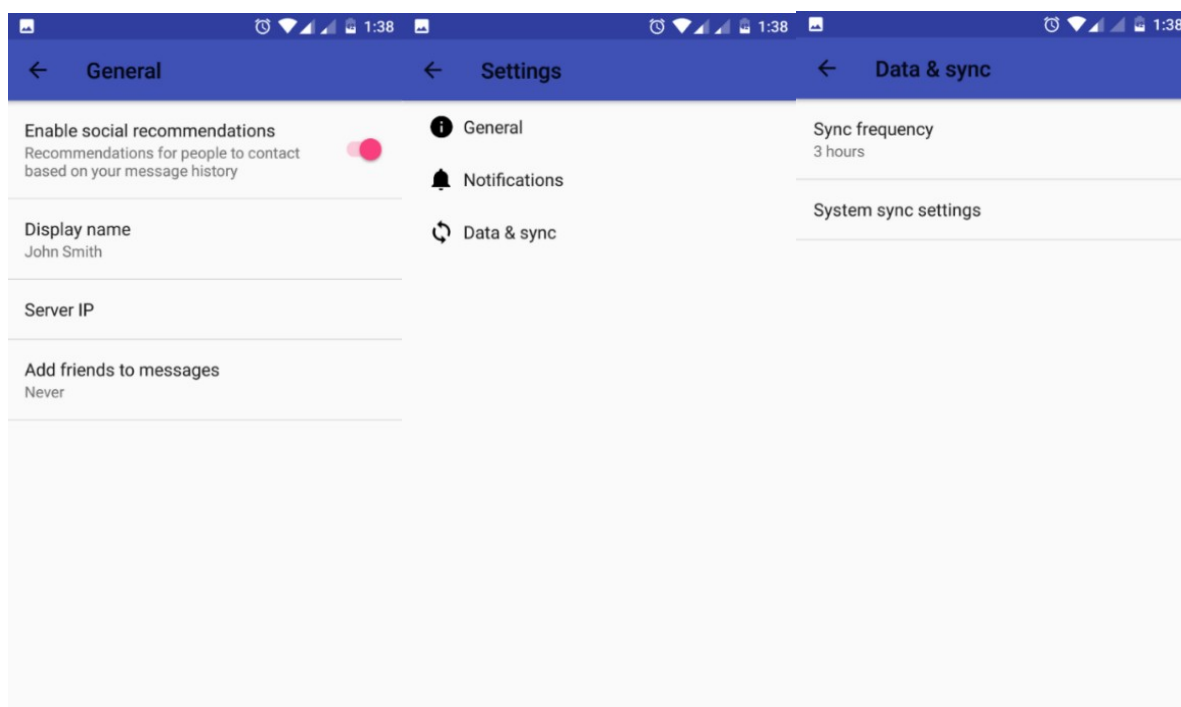


Рис. 3.9 Вікно налаштувань програми

Додати відеопотік можна з головного вікна програми. У вікні додавання потрібно заповнити два поля: назву віддаленої камери та посилання на потік даних, котрі представленні у вигляді відео-, або фотозображення.

Щоб переглянути відеопотік, потрібно вибрати потрібний у списку відеопотоків на головному екрані, при цьому автоматично запуститься модуль ідентифікації (рис.3.10).

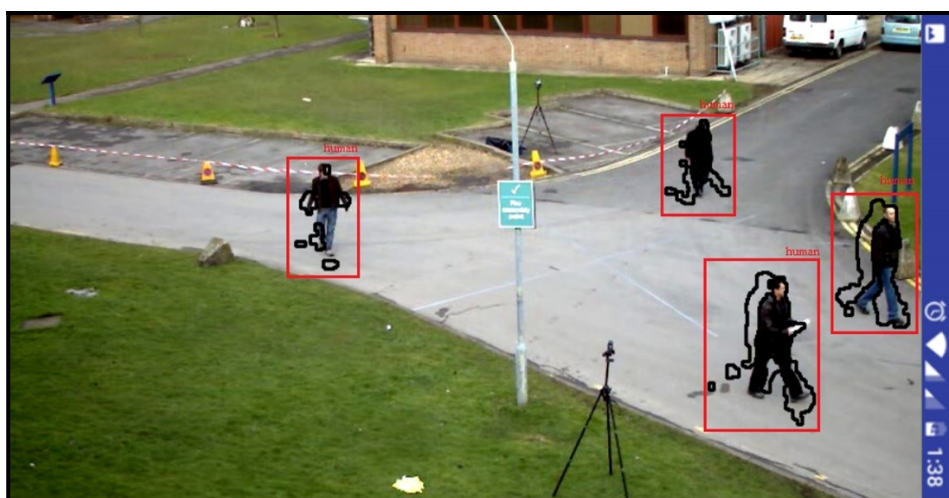


Рис. 3.10 Вікно перегляду відеопотоку в режимі реального часу

Відеопотік відображений на весь екран мобільного пристрою для детального спостереження за об'єктами. Зображення, котре отримується,

завжди якомога більш актуальне. Частота оновлення залежить від якості інтернет-з'єднання як пристрою, котрий отримує дані, так і пристрою, котрий ці дані генерує.

Після розпізнавання модель даних розширяється новими позитивними входженнями що пришвидшує та удосконалює подальші розпізнавання.

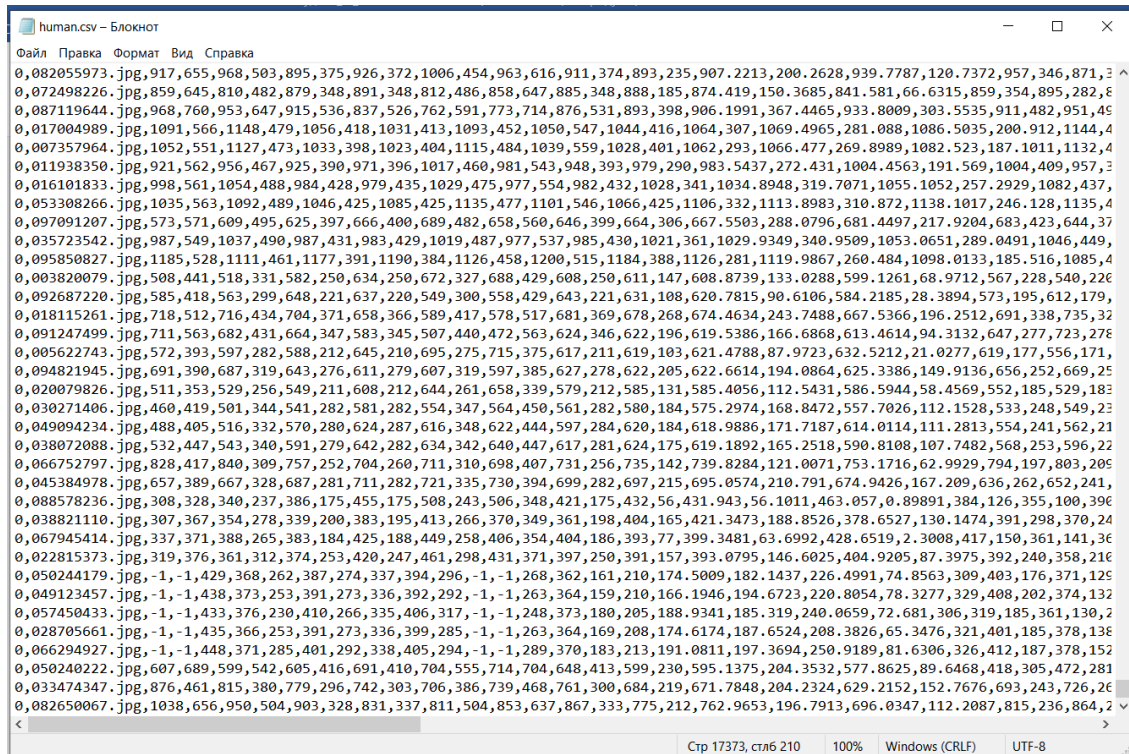


Рис. 3.11. Зміна моделі даних

Висновки до третього розділу

У цьому розділі представлено розробку інформаційної системи розпізнавання об'єктів у відеопотоці, яка базується на методах машинного навчання з використанням SVM (машини опорних векторів). Описано особливості її практичного застосування. Детально розглянуто призначення системи та структуру основних модулів.

Проведено аналіз контрольного прикладу, де продемонстровано роботу основного функціоналу програмної системи.

ВИСНОВКИ

У ході виконання бакалаврської кваліфікаційної роботи було розглянуто та проаналізовано питання розробки інформаційної системи дистанційного відеоспостереження та ідентифікації руху. Досліджено теоретичні основи технологій відеоспостереження, методів аналізу відеопотоків, алгоритмів детекції руху та систем автоматизованого моніторингу. Було здійснено огляд існуючих програмних рішень, таких як Milestone XProtect, Luxriot VMS, iSpy, Camlytics, та проведено їх порівняльний аналіз. Встановлено, що більшість з них вимагають потужного апаратного забезпечення, спеціалізованого налаштування та значних фінансових витрат, що ускладнює їх використання у правоохоронних органах та інших державних установах.

На основі проведеного аналізу було визначено основні вимоги до розроблюваної системи. Вона має забезпечувати ефективне відеоспостереження в режимі реального часу, підтримувати автоматичну ідентифікацію руху, бути зручною у використанні та адаптованою до потреб правоохоронних органів. Було застосовано методи системного аналізу для формування функціональних вимог, а також використано UML діаграми та моделювання бізнес-процесів для побудови логічної структури системи. Особливу увагу було приділено питанням безпеки та збереження даних, оскільки система повинна забезпечувати захист відеопотоків від несанкціонованого доступу та втручання.

Розроблена інформаційна система забезпечує централізований моніторинг відеопотоків, автоматичне виявлення руху та фіксацію підозрілих подій. Вона дозволяє оптимізувати роботу операторів, мінімізувати кількість хибних тривог та підвищити загальну ефективність відеоспостереження. Також передбачено можливість масштабування системи шляхом додавання

нових камер, підключення зовнішніх аналітичних модулів та інтеграції з базами даних правоохоронних органів.

Під час тестування було підтверджено відповідність системи поставленим вимогам. Вона демонструє високу точність ідентифікації руху та стабільність роботи навіть в умовах нестабільного інтернет-з'єднання. Практичне впровадження розробленого програмного комплексу дозволить значно покращити контроль за охоронюваними об'єктами, забезпечити швидке реагування на інциденти та зменшити навантаження на персонал. Крім того, система сприятиме зменшенню впливу людського фактора на процес відеоспостереження, що підвищить його ефективність і дозволить уникнути пропуску критично важливих подій.

Подальші дослідження можуть бути спрямовані на покращення алгоритмів обробки відео, розширення функціональних можливостей системи та її інтеграцію з нейромережевими технологіями для підвищення точності аналізу відеопотоків. Крім того, перспективним напрямком є застосування методів глибокого навчання для підвищення ефективності розпізнавання об'єктів та аномальної поведінки. Також можливе розширення можливостей системи за рахунок подальшої інтеграції з мобільними додатками.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Використання відеоаналітики у роботі Національної поліції. Методичні рекомендації. – Мирошніченко В.О, Кочеткова . І. Б.Махницький О.В. – Дніпропетровський державний університет внутрішніх справ. – Дніпро, 2020 – 34 с.
2. Huu-Thanh D., Tuan L., Hoang T. Deep Learning-Based Anomaly Detection in Video Surveillance: A Survey. Sensors. Vol. 23. 2020, doi: 10.3390/s23115024.
3. Redmon, J., & Farhadi, A. YOLOv3: An Incremental Improvement. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6578–6587.
4. Ren, S., He, K., & Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, 2017, pp. 1137–1149.
5. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., & Fu, C. SSD: Single Shot Multibox Detector. European Conference on Computer Vision (ECCV), 2018, pp. 21–37.
6. Батчаєв В.К., Шадська У.С. Відеоспостереження у публічних місцях: основи захисту персональних даних. К.Компринт, 2021. 98 с.
7. Sultani, W.; Chen, C.; Shah, M. Real-world anomaly detection in surveillance videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6479–6488.
8. Medel, J.R.; Savakis, A. Anomaly detection in video using predictive convolutional long short-term memory networks. arXiv2016, arXiv:1612.00390
9. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural

Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2022; pp. 1097–1105

10. Комарова О. Сучасні системи безпеки: чи змінюється служба в поліції? [Електронний ресурс] – Режим доступу: <https://www.radiosvoboda.org/a/svoboda-v-detalyakh-chy-zminyuyetsyasluzhba-v-politsiyi/29654855.html>
11. Спеціальна техніка Національної поліції України: навч. посіб. з дисц. «Тактико-спеціальна підготовка» / Ю. В. Гнусов, В. А. Світличний, Ю. М. Онищенко; Харк. нац. ун-т внутр. справ, факультет № 4, каф. Кібербезпеки. – Х.: ХНУВС, 2017. – 175 с.
12. Blue Iris by Perspective Software | Video Security and Webcam Software [Електронний ресурс] – Режим доступу: <https://blueirissoftware.com/>
13. iSpy - Best Video Surveillance and Security Camera Software for Motion detection [Електронний ресурс] – Режим доступу: <https://www.ispyconnect.com/>
14. Milestone Systems | Global leading video management software provider [Електронний ресурс] – Режим доступу: <https://www.milestonesys.com/>
15. luxriot VMS [Електронний ресурс] – Режим доступу: <https://www.luxriot.com/>
16. VAST 2. [Електронний ресурс] – Режим доступу: https://www.vivotek.com/en-US/products/software/vast_2
17. Camlytics [Електронний ресурс] – Режим доступу: <https://camlytics.com/>
18. Катренко А. В. Системний аналіз: підручник / А. В. Катренко.– Львів: Видавництво ПП "Новий світ - 2000", 2024.–395 с.
19. Rababah A. Assessing the Effectiveness of UML Models in Software System Development. International Journal of Applied Science and Research. 2024. Pp.13-24. <https://doi.org/10.56293/IJASR.2024.5703>.
25. Kumar, K.S.; Bhavani, R. Human activity recognition in egocentric video using HOG, GiST and color features. Multimed. ToolsAppl. 2020,79, 3543–3559.

26. Roy, P.K.; Om, H. Suspicious and violent activity detection of humans using HOG features and SVM classifier in surveillance videos. In *Advances in Soft Computing and Machine Learning in Image Processing*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 277–294
27. Jagadeesh, B.; Patil, C.M. Video based human activity detection, recognition and classification of actions using SVM. *Trans. Mach.Learn. Artif. Intell.* 2018,6, 22–22
28. Abidine, B.M.; Fergani, L.; Fergani, B.; Oussalah, M. The joint use of sequence features combination and modified weighted SVM for improving daily activity recognition. *Pattern Anal. Appl.* 2018,21, 119–138.
29. SDK Platform Tools release notes | Android Studio [Електронний ресурс] – Режим доступу: <https://developer.android.com/tools/releases/platform-tools>
30. Microsoft Azure – хмарна платформа для розгортання додатків. Офіційний сайт: <https://azure.microsoft.com/>
31. MySQL 8.0: Реляційна система керування базами даних. Офіційний сайт: <https://www.mysql.com/>
32. MikroKod [Електронний ресурс] – Режим доступу: www.mikrokod.com.ua/

ДОДАТКИ

Додаток 1

Логіка роботи сервера

```

public class Server {

    private static final String DATABASE_PATH = "./server/assets/database/1.0";

    private static final Logger LOG = LogManager.getLogger();

    private Tokens<SecureServerExchange> exchangeTokens;
    private Streams streams;
    private Profiles profiles;
    private Framers framers;
    private final Object lock = new Object();
    private final int port;

    private volatile Connection databaseConnection;
    private volatile Thread loopThread;

    public Server(int port) {
        this.port = port;
    }

    public Tokens<SecureServerExchange> getExchangeTokens() {
        checkReady();
        return exchangeTokens;
    }

    public Profiles getProfiles() {
        checkReady();
        return profiles;
    }

    public Streams getStreams() {
        checkReady();
        return streams;
    }

    /*public Framers getFramers() {
        checkReady();
        return framers;
    }*/

    public boolean isRunning() {
        return loopThread != null;
    }

    public boolean isReady() {
        return loopThread != null && databaseConnection != null && profiles != null;
    }
}

```

```

public void start() {
    synchronized (lock) {
        if (loopThread == null) {
            loopThread = new LoopThread();
            loopThread.start();
        }
    }
}

public void stop() {
    synchronized(lock) {
        if (loopThread != null) {
            loopThread.interrupt();
            loopThread = null;
            if(databaseConnection != null) {
                try {
                    databaseConnection.close();
                } catch (SQLException e) {
                    LOG.warn("Cannot close database: " + e);
                }
                databaseConnection = null;
            }
            profiles = null;
            streams = null;
            exchangeTokens = null;
            framers = null;
        }
    }
}

private void checkReady() {
    if(!isReady()) {
        throw new IllegalStateException("Server is not ready.");
    }
}

public static void main(String[] args) {
    System.out.println(System.getProperty("user.dir"));
    Server server = new Server(45675);
    server.start();
}

private class LoopThread extends Thread {

    @Override
    public void run() {
        try {
            LOG.debug("Open database.");
            org.h2.Driver.load();
            databaseConnection = DriverManager.getConnection(
                "jdbc:h2:" + DATABASE_PATH + ";create=true;TRACE_LEVEL_FILE=0;");

```

```

        profiles = new Profiles(databaseConnection);
        streams = new Streams(databaseConnection);
        exchangeTokens = new Tokens<>();
        framers = new Framers();
    } catch (SQLException e) {
        LOG.error("Cannot be initialize database: " + e);
        Server.this.stop();
        return;
    }

    ServerSocket serverSocket = null;
    while(true) {
        Socket socket = null;
        try {
            if(serverSocket == null) {
                LOG.debug("Create server socket.");
                serverSocket = new ServerSocket(port);
                System.out.println(serverSocket.getInetAddress());
            }
            socket = serverSocket.accept();
            LOG.debug("Accept client: " + socket);
            byte typeByte = readTypeByte(socket);
            try {
                Type type = Type.valueOfId(typeByte);
                ClientHost host = new ClientHost(socket);

                Exchange exchange;
                switch (type) {
                    case STREAMING: exchange = new StreamingServerExchange(Server.this,
host); break;
                    case SIGN_IN: exchange = new SignInServerExchange(Server.this, host);
break;
                    case SIGN_UP: exchange = new SignUpServerExchange(Server.this, host);
break;
                    case STREAM_REMOVE: exchange = new
StreamRemoveServerExchange(Server.this, host); break;
                    case STREAM_PREVIEW_LIST: exchange = new
StreamPreviewListServerExchange(Server.this, host); break;
                    case STREAM_LIST: exchange = new StreamListServerExchange(Server.this,
host); break;
                    case STREAM_APPEND: exchange = new
StreamAppendServerExchange(Server.this, host); break;
                    default:
                        LOG.error("Unsupported element type: " + type);
                        socket.close();
                        continue;
                }
                exchange.start();
            } catch (UnknownTypeException e) {
                LOG.error("Illegal client. Received type: " + typeByte);
                socket.close();
            }
        }
    }

```

```

    } catch (IOException | UnknownTypeException e) {
        LOG.warn("Connection problems: " + e);
        if(serverSocket != null) {
            try {
                if(socket != null) socket.close();
                serverSocket.close();
            } catch (IOException e1) {
                LOG.warn("Error when force close socket: " + e1);
            } finally {
                serverSocket = null;
            }
        }
        LOG.debug("Waiting 1 minute for try again establish connection.");
        try {
            Thread.sleep(1000 * 60);
        } catch (InterruptedException e1) {
            LOG.warn("Waiting 1 minute for try again establish connection is interrupted.");
        }
    }
    if(isInterrupted()) break;
}

private byte readTypeByte(Socket socket) {
    try {
        int read = socket.getInputStream().read();
        return (byte) read;
    } catch (IOException e) {
        LOG.warn("Reading element type failed: " + e);
        return -1;
    }
}
}
}
}

```