

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

Теплоенергетичний факультет

Кафедра автоматизації проектування енергетичних процесів і систем

"На правах рукопису"  
УДК 004.855.5

«До захисту допущено»  
Завідувач кафедри  
                    О.В. Коваль  
(підпис)                      (ініціали, прізвище)  
“    ”                      2018р.

## Магістерська дисертація

зі спеціальності - 122 Комп'ютерні науки та інформаційні технології  
за спеціалізацією - Геометричне моделювання в інформаційних системах  
на тему: Моделювання процесів визначення положення об'єктів на  
динамічному наборі зображень

Виконав: студент   6   курсу, групи   ТР-71мп    
Байда Дмитро Володимирович  
(прізвище, ім'я, по батькові)

\_\_\_\_\_ (підпис)

Науковий керівник: к.т.н. Шалденко О.В.  
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

\_\_\_\_\_ (підпис)

Консультант \_\_\_\_\_  
(назва розділу)                      (вчені ступінь та звання, прізвище, ініціали)

\_\_\_\_\_ (підпис)

Рецензент \_\_\_\_\_  
(посада, вчене звання, науковий ступінь, прізвище та ініціали)

\_\_\_\_\_ (підпис)

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

Студент \_\_\_\_\_  
(підпис)

Київ - 2018

**Національний технічний університет України  
“Київський політехнічний інститут імені Ігоря Сікорського”**

Факультет теплоенергетичний

Кафедра автоматизації проектування енергетичних процесів і систем

Рівень вищої освіти другий рівень

Спеціальність “Комп’ютерні науки та інформаційні технології”

ЗАТВЕРДЖУЮ

\_\_\_\_\_  
(прізвище, ініціали)                      (підпис)  
«\_\_\_\_\_» \_\_\_\_\_ 2018р.

ЗАТВЕРДЖУЮ

Завідувач кафедри  
Коваль О.В. \_\_\_\_\_  
(прізвище, ініціали)                      (підпис)  
«\_\_\_\_\_» \_\_\_\_\_ 2018р.

**З А В Д А Н Н Я  
НА МАГІСТЕРСЬКУ ДИСЕРТАЦІЮ СТУДЕНТУ**

Байді Дмитру Володимировичу

(прізвище, ім’я, по батькові)

1. Тема дисертації Моделювання процесів визначення положення об’єктів на динамічному наборі зображень

Науковий керівник Шалденко Олексій Вікторович, к.т.н.

(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від 15 березня 2018 року №1151-с

2. Строк подання студентом дисертації \_\_\_\_\_

3. Об’єкт дослідження комп’ютерні інформаційні системи і технології

4. Предмет дослідження використання нейронних мереж для виявлення змін положення об’єкту на наборі зображень

5. Перелік питань, які потрібно розробити аналіз існуючих методів побудови систем моделювання процесів визначення положення об’єктів на динамічному наборі зображень; вибір найбільш відповідної апаратної бази; вибір засобів розробки програмного забезпечення і створення людино-машинного інтерфейсу;

6. Перелік ілюстративного матеріалу схема розпізнавання пози людини; схема роботи користувача з системою; модель бази даних; архітектура програмного застосування; архітектура згорткової нейронної мережі; патерн MVVM; графічний інтерфейс програмного застосування;

7. Перелік публікацій тези доповіді XVI міжнародної науково-практичної конференції аспірантів, магістрантів, студентів «Сучасні проблеми наукового забезпечення енергетики»; тези доповіді V науково-практичної дистанційної конференції молодих вчених і фахівців з розробки програмного забезпечення «СУЧАСНІ АСПЕКТИ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»

8. Дата видачі завдання «11» вересня 2017 р.

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Аналіз проблеми моделювання процесів визначення положення об'єктів на динамічному наборі зображень	11.09.2017-12.11.2017	
2	Аналіз існуючих реалізацій та шляхів вирішення проблеми визначення положення об'єктів на динамічному наборі зображень	13.11.2017-15.01.2018	
3	Аналіз існуючих механізмів класифікації пози людини	16.01.2018-05.03.2018	
4	Аналіз алгоритмів машинного навчання для розпізнавання положення об'єктів	06.03.2018-27.06.2018	
5	Аналіз вимог до програмного застосунку	28.06.2018-18.08.2018	
6	Моделювання схеми роботи програми	19.08.2018-30.08.2018	
7	Розробка архітектури програмного забезпечення	01.09.2018-20.09.2018	
8	Розробка дизайну графічного інтерфейсу	21.09.2018-10.10.2018	
9	Розробка програмного застосунку	11.10.2018-30.11.2018	
10	Оформлення документації	01.12.2018-07.12.2018	

Студент

\_\_\_\_\_

( підпис )

Байда Д.В.

\_\_\_\_\_

(прізвище та ініціали)

Науковий керівник

\_\_\_\_\_

( підпис )

Шалденко О.В.

\_\_\_\_\_

(прізвище та ініціали)

# РЕФЕРАТ

## **Структура й обсяг дипломної роботи.**

Магістерська дисертація складається зі вступу, шести розділів, висновку, переліку посилань з 41 найменувань, 2 додатки, і містить 21 рисунки, 25 таблиць. Повний обсяг магістерської дисертації складає 78 сторінок, з яких перелік посилань займає 4 сторінок, додатки – 10 сторінок.

**Актуальність теми.** Все більше і більше сфер людської діяльності автоматизується за допомогою машинного навчання. Комп'ютер може вирішувати задачі, які колись були могли бути вирішити тільки людьми. Також в наш час, для забезпечення збирання цінної статистики на спортивних подіях, також можна автоматизувати людську роботу і замінити її на спеціальні програми, які працюють на техноллогіях, заснованих на методах та алгоритмах машинного навчання та зору.

**Мета дослідження** полягає у дослідженні моделювання процесів визначення положення об'єктів на динамічному наборі зображень, а також створенні програмного забезпечення для визначення пози людини на відео та сповіщення користувача про задані пози засобами електронної пошти.

Для досягнення поставленої задачі були сформульовані наступні **завдання дослідження**, що визначили логіку дослідження та його структуру:

— забезпечення користувача персональним кабінетом, в якому можна додавати власні камери спостереження;

— визначення пози людини на відеопотоці і виведення скелету на екран користувача;

— класифікація пози людини і визначення негативних для людини поз тіла;

— сповіщення користувача засобами електронної пошти про пози які було розпізнано на відеопотоці;

— надати можливість розпізнавати пози з відеопотоку вебкамери;

— забезпечити користувача зручним графічним інтерфейсом, який може працювати як на комп'ютері, так і на мобільному телефоні.

**Об'єктом дослідження** є алгоритми та методи машинного навчання та зору для визначення положення об'єктів на динамічному наборі зображень.

**Предметом дослідження** є програмне забезпечення для розпізнавання пози людини на відео.

**Методи дослідження.** Розв'язання поставлених задач виконувались засобами комп'ютерного програмування та вивчення напрацювань світової спільноти програмістів, зокрема були вивчені методи розробки:

- створення програмних систем з використанням триланкової архітектури;
- розпізнавання поз людини та подальше розпізнавання з застосуванням сучасних методів машинного навчання.

**Наукова новизна одержаних результатів.** Найбільш суттєвими науковими результатами магістерської дисертації є:

- удосконалено спосіб комбінації різних методів машинного навчання для розпізнавання та класифікації пози людини на відео;
- набуло подальшого розвитку використання триланкової архітектури для створення складних програмних систем, які легко розширювати.

**Практичне значення одержаних результатів** роботи полягає в розробці програмного продукту, який дозволить автоматизувати роботу з розпізнавання поз людини на відео та подальшим сповіщення користувачів засобами електронної пошти.

**Ключові слова.** *МАШИННЕ НАВЧАННЯ, КЛАСИФІКАЦІЯ, РОЗПІЗНАВАННЯ, .NET CORE, БРАУЗЕР, ORM.*

# ABSTRACT

## **The structure and volume of the thesis.**

Master's thesis consists of an introduction, five chapters, conclusion, list of references with 41 titles, 2 annexes, and contains 21 figures, 25 tables. The full range of master's thesis is 78 pages with a list of links takes 4 pages, apps –10 pages.

**Topicality of the theme.** More and more spheres of human activity are automated with the help of machine learning. A computer can solve tasks that once could only be decided by humans. Also, in today's time, to secure the collection of valuable statistics on sports events, it is also possible to automate human work and replace it with special programs that work on technologies based on methods and algorithms of machine learning and vision.

**The purpose of the research** is to study the simulation of processes for determining the position of objects in a dynamic image set, as well as the creation of software for determining the posture of a person on the video and the user's alert of the specified poses by e-mail.

To accomplish the task, the following **research objectives** were formulated, which determined the logic of the research and its structure:

- providing the user with a personal cabinet where he can add own cameras;
- recognizing of human pose on a video stream;
- classification of human pose and the determination of the negative for the human body;
- email notification of user about the poses that were identified on the video stream;
- provide the ability to recognize poses from webcam;
- provide a user-friendly graphical interface that can work both on a computer and on a mobile phone.

**The object of research** are algorithms and methods of machine learning and vision to determine the position of objects on a dynamic image set.

**The subject of research** is software for recognizing a person's pose on a video.

**Scientific novelty of the results.** The most significant scientific results of the master thesis are:

— improved method of combining different methods of machine learning to recognize and classify a person's posture in a video;

— further developing of using of three-tier architecture to create complex software systems that are easy to expand.

**The practical significance of the results of the work** is to develop a software product that will automate the work of recognizing human poses on video and notifying users about it via e-mail

**Keywords.** *MACHINE LEARNING, CLASSIFICATION, RECOGNITION, .NET CORE, BROWSER, ORM.*

# ЗМІСТ

Вступ.....	10
1. Визначення положення об'єктів на динамічному наборі зображень .....	12
2. Огляд існуючих програмних рішень визначення положення об'єктів на динамічному наборі зображень.....	14
2.1 Аналіз існуючих програмних засобів .....	14
2.2 Опис організації розробки системи моделювання процесів визначення положення об'єктів на динамічному наборі зображень.....	15
3. Засоби реалізації програмної системи.....	17
3.1 Вибір архітектури програмного комплексу.....	17
3.2 Опис архітектури серверу.....	18
3.3 Опис архітектури клієнтського застосунку .....	20
3.4 Опис інструментів розробки.....	21
3.5 Обґрунтування вибору програмної реалізації .....	24
3.6 Обґрунтування вибору згорткової нейронної мережі для вирішення моделювання процесів визначення положення об'єктів на динамічному наборі зображень.....	25
3.7 Архітектура згорткової нейронної мережі MobileNet .....	28
3.8 Обґрунтування вибору лінійної регресії для вирішення моделювання процесів визначення положення об'єктів на динамічному наборі зображень.....	30
4. Опис програмної реалізації .....	34
4.1 Опис функціональності системи .....	34
4.2 Концептуальна модель бази даних .....	35
4.3 Опис таблиць бази даних.....	36
4.4 Етапи роботи програми для розпізнавання положення об'єкту.....	38
4.5 Розробка кабінету користувача .....	39
4.5.1 Модуль керування наборами камер відеоспостереження.....	39



4.5.2 Модуль розпізнавання поз людини на відеопотоці.....	40
4.5.3 Модуль сповіщення про надзвичайні пози .....	40
5. Методика роботи користувача з програмною системою .....	42
5.1 Інсталяція та системні вимоги.....	42
5.2 Інструкція з використання програмного продукту.....	42
6. Стартап проект.....	46
6.1 Основні ідеї проекту.....	46
6.2 Технологічний аудит ідеї проекту .....	48
6.3 Аналіз ринкових можливостей запуску стартап-проекту.....	49
6.4 Розроблення ринкової стратегії проекту .....	55
6.5 Розроблення маркетингової програми проекту .....	58
Висновки.....	62
Список використаних джерел .....	64
Додаток А .....	69
Додаток Б.....	77

## ВСТУП

Останні десятиріччя набувають все більшого поширення різноманітні алгоритми та прийоми машинного навчання. Сфера їх застосування розростається з кожним днем. Від класифікації зображень до імітації мозку примітивних тварин – все це є реальністю, яка оточує нас. Ніхто не може точно спрогнозувати, що нас чекає в найближчі роки, але скоріше за все нас чекають прориви в сфері штучного інтелекту, які піднімуть рівень комп'ютерних технологій на новий рівень.

Наявні алгоритми та архітектури нейронних мереж дають небачені раніше можливості для сучасних технологій. Вже створено тисячі різноманітних бібліотек і фреймворків, які дають змогу застосовувати ці технології для власних цілей та проектів.

Нас оточує величезна кількість інформації і більша частина з неї – неструктурована, але може містити дуже велику кількість знань, які могли б принести користь для людей. Одним із таких джерел є відео з камер спостереження. Більша частина із цієї інформації, на перший погляд, є несуттєвою, але з неї можна отримати більше користі ніж здається.

Великою проблемою, яку на даний момент вирішують за допомогою людського нагляду, є слідкування за станом пацієнтів у лікарнях. У таких місцях потрібна негайна реакція на різноманітні надзвичайні події або падіння людей, для того щоб попередити шкідливі наслідки для людського здоров'я. Ще одним з прикладів застосування даних методів може бути забезпечення виокремлення цінної статистики на спортивних подіях для подальшого аналізу та удосконалення техніки гри спортсменів. Також можна автоматизувати інші аспекти людської сфери діяльності спеціальним програмним забезпеченням, яке працює на технологіях, заснованих на методах та алгоритмах машинного навчання та зору.

Людська увага не завжди може забезпечити високий рівень реакції та концентрації на поставлені задачі. Також людина не може обробити одночасно

велику кількість інформації. Тому використання комп'ютерних технологій є дуже актуальною задачею, для вирішення подібних задач.

Останнім часом спостерігається тенденція автоматизації все більшої кількості сфер людської діяльності. В наслідок цього підвищується якість аналізу інформації, зменшується вплив людського фактор та собівартість виконання робіт. Таким чином, було поставлена задача дослідити можливість застосування алгоритмів та технік машинного навчання для автоматизації процесів виявлення та оповіщення про несприятливі для людського здоров'я зміни положення тіла, наприклад, падіння внаслідок втрати свідомості.

# 1. ВИЗНАЧЕННЯ ПОЛОЖЕННЯ ОБ'ЄКТІВ НА ДИНАМІЧНОМУ НАБОРІ ЗОБРАЖЕНЬ

У сучасному світі в останні десятиріччя широкого поширення набули проблеми систематизації інформації та виділення серед неї корисних для людей знань. Існує багато сфер людської діяльності де їхня праця застосовується тільки тому, що її важко автоматизувати. Але в наш час все більше набувають розповсюдження алгоритми машинного навчання, які імітують та моделюють діяльність людського мозку. Це надало нові величезні можливості для застосування комп'ютерів замість людей в професіях, де це дає змогу автоматизувати діяльність з меншими помилками і більшою ефективністю роботи [4].

Серед таких сфер діяльності можна виділити роботу оператора камер відеоспостереження. Велике значення в цій сфері діяльності має людський фактор, людина може запросто відволіктися або просто не звернути увагу на важливі події, які відбуваються на відеопотоці. Також людина не може одночасно оброблювати велику кількість інформації, яка поступає з різних джерел, тому часто таку роботу виконують декілька людей, щоб зменшити кількість помилок та збільшити загальну увагу над подіями, які відбуваються на відео.

Важливою проблемою, яку вирішують за допомогою спостереження, є контроль та реакція над шкідливими для людського змінами стану тіла, тобто падіння людини тощо [14]. В таких випадках необхідна швидка реакція та допомога з боку інших людей, щоб зменшити негативні наслідки для здоров'я.

Тому було запропоновано дослідити методи моделювання процесів визначення положення об'єктів та можливість їх застосування для вирішення поставленої задачі.

Основні завдання розробки системи:

- аналіз існуючих методів побудови систем моделювання процесів визначення положення об'єктів на динамічному наборі зображень;
- вибір найбільш відповідної апаратної бази;

— вибір засобів розробки програмного забезпечення і створення людино-машинного інтерфейсу.

Основними функціями системи є:

— забезпечення користувача персональним кабінетом, в якому можна додавати власні камери спостереження;

— визначення пози людини на відеопотоці і виведення скелету на екран користувача;

— класифікація пози людини і визначення негативних для людини поз тіла;

— сповіщення користувача засобами електронної пошти про пози які було розпізнано на відеопотоці;

— надати можливість розпізнавати пози з відеопотоку вебкамери;

— забезпечити користувача зручним графічним інтерфейсом, який може працювати як на комп'ютері, так і на мобільному телефоні.

## **Висновки до розділу 1**

Задача моделювання процесів визначення положення об'єктів на динамічному наборі зображень полягає в створенні програмного продукту, який дозволить розпізнавати та класифікувати пози людини на динамічному наборі зображень, тобто відео. Дані можливості надають змогу автоматизувати сфери людської діяльності, де потрібен людський контроль та реакція на події, які зображені на відео з систем спостереження.

## **2. ОГЛЯД ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ ВИЗНАЧЕННЯ ПОЛОЖЕННЯ ОБ'ЄКТІВ НА ДИНАМІЧНОМУ НАБОРІ ЗОБРАЖЕНЬ**

З кожним роком набувають все більшого поширення різноманітні алгоритми машинного навчання. На їх основі вже побудовано багато програмного забезпечення [8], яке вирішує дуже широкий спектр прикладних задач. Серед них можна виділити і задачу визначення положення об'єктів на динамічних наборах зображень.

З метою автоматизації визначення та оповіщення про ситуації в яких людина може приймати незвичну позу на відеопотоці було запропоновано розробити програмну систему, яка допоможе розпізнавати скелет та пози людини та сповіщувати засобами електронної пошти користувача про надзвичайні ситуації.

### **2.1 Аналіз існуючих програмних засобів**

У процесі пошуку інформації, та аналізу існуючих рішень, було виявлено, що на даний момент такі системи є досить складними у використанні, або реалізують поставлену задачу не в повному обсязі:

— “Realtime Multi-Person Pose Estimation” — це комп'ютерна система, яка реалізує модель розпізнавання пози людини OpenPose [34]. Недоліком даної системи для поставленої задачі можна вважати, що вона не дозволяє класифікувати пози людини та оповіщати через електронну пошту про надзвичайні ситуації;

— “Paper Architecture” — це реалізація архітектури “глибоких нейронних мереж”. Дане програмне забезпечення також не реалізує поставлену задачу в повному обсязі а лише дозволяє виділяти скелет людини на відео.

Також недоліком цих систем є їх громіздкість. Для початківців ці програми можуть здатися переповненими функціоналом та дуже специфічними в плані налаштування [23].

Так як автоматизація людської діяльності за допомогою алгоритмів машинного навчання набуває широкого розповсюдження, то розробка зручної, інтуїтивно зрозумілої, гнучкої та невимогливої до апаратного забезпечення системи для розпізнавання пози людини на відеопотоці є актуальною задачею.

Одними із варіантів, який зможе працювати на будь-яких сучасних пристроях, є розробка програмної системи, яка заснована на веб-технологіях.

## **2.2 Опис організації розробки системи моделювання процесів визначення положення об'єктів на динамічному наборі зображень**

Задача моделювання процесів визначення положення об'єктів на динамічному наборі зображень є досить широкою та має безліч нюансів, які розробник повинен дослідити та узгодити із замовником. У результаті дослідження аналогічних систем, а також вивчення предметної області, було вирішено створити систему, основними функціями якої є розпізнавання положення об'єктів на зображенні, розпізнавання точок скелету, класифікація пози людини на зображенні, а також повідомлення користувача про настання заданих поз [17].

Для виконання поставленої задачі було обрано набір засобів реалізації, було спроектовано архітектуру системи, а також її компонентів. Була спроектована схема таблиць бази даних для зберігання даних системи. Також був розроблений дизайн для майбутньої системи. Усі вимоги були обговорені з замовником і в результаті розробки необхідно отримати програмний продукт, який їх задовольняє.

## **Висновки до розділу 2**

У результаті огляду існуючих програмних систем було проаналізовано основні недоліки та можливості кожної з систем. Як наслідок, було вирішено створити аналогічний програмний продукт, який дозволить вирішувати поставлені задачі краще ніж у аналогів, також буде надавати ширший набір функцій. Також було обрано засоби реалізації та спроектовано основні частини майбутнього програмного продукту.



### 3. ЗАСОБИ РЕАЛІЗАЦІЇ ПРОГРАМНОЇ СИСТЕМИ

Аналізуючи поставлену задачу та методи її розв'язку, було вирішено розроблювати програмний комплекс на основі веб-технологій. Головною перевагою веб-застосунку перед іншими варіантами є його універсальність і можливість використання на будь-яких пристроях без портування на цільову операційну систему (браузер і його віртуальна машина виступає як цільова універсальна операційна система і комп'ютер) [35].

#### 3.1 Вибір архітектури програмного комплексу

Для реалізації поставленої задачі було вирішено використовувати триланкову архітектуру, яка складається з таких компонентів: сервер, база даних і клієнт. Схема даної архітектури зображена на рисунку 3.1.

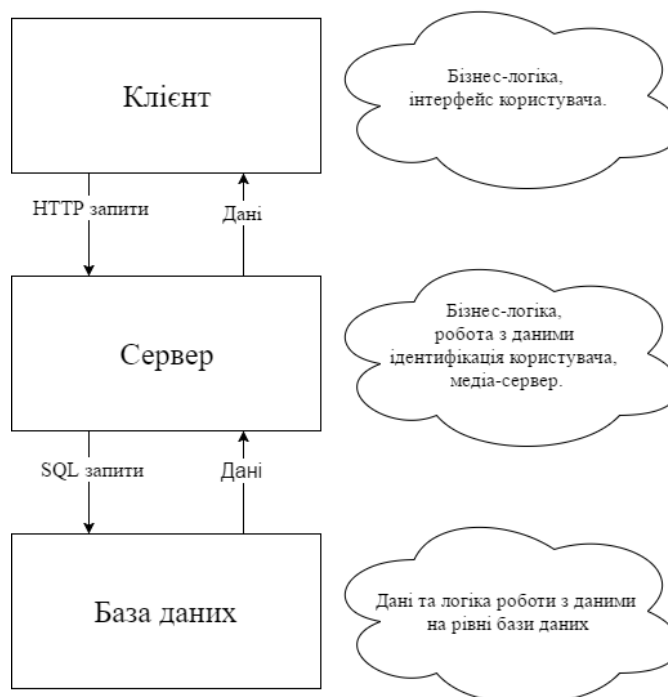


Рисунок 3.1 – Триланкова архітектура програмного комплексу

Головним центром програмного комплексу є сервер. У ньому зосереджена основна бізнес-логіка та логіка доступу до бази даних. За допомогою серверу відбувається ідентифікація користувача для надання індивідуального доступу до програмного застосунку. Сервер є єдиним зв'язком між користувачем та базою даних, щоб унеможливити пошкодження даних та їх використання не за призначенням. Для того, щоб користуватися програмою, потрібно бути авторизованим у системі, тому дана логіка реалізуються на рівні серверу, тому що на рівні користувача можлива підміна прав доступу та інші методи неконтрольованого доступу до даних [41].

Під час користування програмою, користувач взаємодіє з клієнтським додатком, яким є веб-сайт в даному випадку. На рівні користувача реалізований інтерфейс, за допомогою якого відбувається налаштування програми та перегляд результатів роботи. Також на користувацькому рівні відбувається попередня обробка даних перед відправленням на сервер і також опрацювання результатів від сервера. Ще на цьому рівні відбувається перший етап аутентифікації користувача для обмеження неконтрольованого доступу до програми.

Важливою задачею рівня бази даних є забезпечення збереження даних, які сервер зберігає для подальшого використання [37]. Також забезпечується цілісність даних за допомогою зовнішніх зв'язків та ключів. На рівні бази даних також можна реалізовувати деяку бізнес-логіку, яка не потребує використання зовнішніх джерел даних окрім самої бази даних та її таблиць.

## **3.2 Опис архітектури серверу**

Шаблон проектування — це архітектура, рішення яке описує яким чином вирішуються задачі, які часто зустрічаються при розробці програмних систем чи додатків [25].

Для реалізації серверу було використано фреймворк, який є реалізацією шаблону проектування MVC [1].

MVC (Model-View-Controller) — це шаблон проектування, головною ідеєю якого є відділити логіку застосунку від представлення (рисунок 3.2). Принципом MVC є розділення програмної реалізації системи на три головні компоненти: М — Model (Модель), V — View (Представлення), С — Controller (Контролер), таким чином, що редагування будь-якого компонента може відбуватися незалежно [15].

Модель — містить знання про предметну область, дані та правила доступу до цих даних, але нічого не знає про контролери та представлення. У моделі відбувається бізнес-логіка роботи застосунку [26]. Модель надає контролеру дані які запрошує користувач, або інша система.

Представлення — надає можливість по-різному відображати дані отримані будь-яким способом від моделі, може містити в собі логіку. Представлення — це кінцевий інтерфейс, з яким взаємодіє користувач. Користувач може передавати дані через представлення.

Контролер — реалізує взаємодію між моделлю і представленням. У функції контролера входить відстеження визначених подій, що виникають в результаті дій користувача. Контролер дозволяє структурувати код шляхом групування пов'язаних дій в окремий клас.

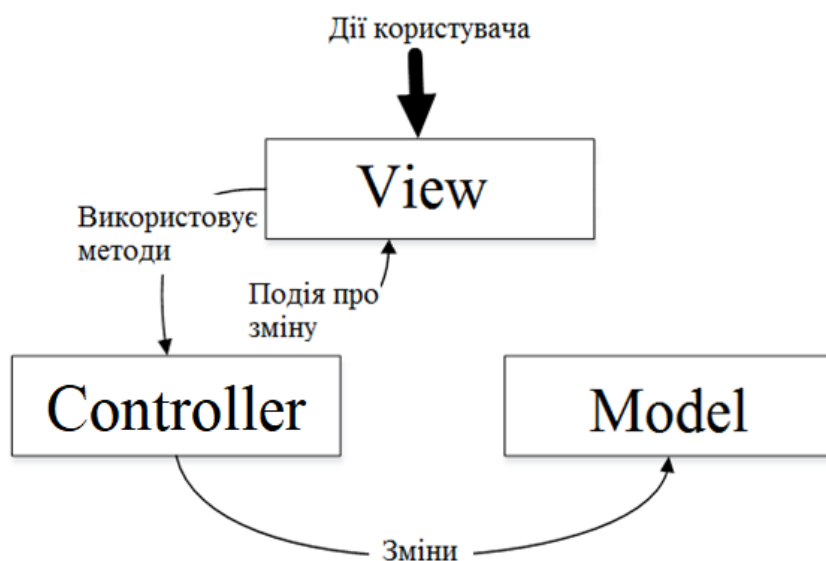


Рисунок 3.2 – Схема роботи MVC шаблону

Основна мета застосування цієї концепції полягає в відділенні бізнес-логіки (моделі) від її візуалізації (представлення) [12]. За рахунок такого поділу підвищується можливість повторного використання та гнучкість системи. Також для великих проєктів, застосування шаблону MVC або його похідних дозволяє застосувати тестування різноманітних компонентів програмної системи.

### 3.3 Опис архітектури клієнтського застосунку

Для реалізації клієнтського застосунку було використано фреймворк, в основу якого було покладено шаблон проектування MVVM (рисунок 3.3).

MVVM спрощує розділення розробки графічного інтерфейсу користувача від розробки бізнес-логіки, відомої як модель [31]. MVVM зручно використовувати замість класичного MVC та йому подібних у тих випадках, коли на платформі, де ведеться розробка, присутнє “зв'язування даних”.



Рисунок 3.3 – Схема роботи MVVM шаблону

Шаблон MVVM ділиться на три частини:

1. Модель (Model), як і в класичному шаблоні MVC, Модель являє собою фундаментальні дані, що необхідні для роботи застосунку;
2. Представлення (View), представлення — це графічний інтерфейс, тобто вікно, кнопки тощо;

3. Модель представлення (ViewModel, що означає “Model of View”) з одного боку є абстракцією представлення, а з іншого надає обгортку даних з моделі, які мають зв'язуватись.

### 3.4 Опис інструментів розробки

Програмний комплекс побудований за принципами триланкової архітектури побудови програм. Кожен рівень цієї архітектури реалізовано з використанням різних технологій і головною ціллю було створення мультиплатформного рішення з використанням відкритих технологій.

Для клієнтського рівня було використано такий набір технологій: мова програмування TypeScript, фреймворк для побудови веб-застосунків Vue.js, бібліотеку для машинного навчання TensorFlow.js, фреймворк для створення графічних інтерфейсів Bootstrap.

TypeScript — мова програмування, представлена Microsoft восени 2012; позиціонується як засіб розробки веб-застосунків, що розширює можливості JavaScript. TypeScript є зворотно сумісним з JavaScript [32]. Фактично, після компіляції програму на TypeScript можна виконувати в будь-якому сучасному браузері або використовувати спільно з серверною платформою Node.js.

Переваги над JavaScript:

- можливість явного визначення типів (статична типізація);
- підтримка використання повноцінних класів;
- підтримка підключення модулів.

Vue.js — JavaScript-фреймворк що використовує шаблон MVVM для створення інтерфейсів користувача на основі моделей даних, через реактивне зв'язування даних. Vue використовує синтаксис шаблонів на основі HTML, що дозволяє декларативно зв'язувати рендеринг DOM з основними екземплярами даних в Vue [40]. Одна із

найвиразніших особливостей Vue — це ненав'язлива реактивна система. Моделі це прості JavaScript об'єкти. Це робить керування станами дуже простим та інтуїтивним [11].

TensorFlow.js — відкрита програмна бібліотека для машинного навчання цілій низці задач, розроблена компанією Google для задоволення її потреб у системах, здатних будувати та тренувати нейронні мережі для виявлення та розшифровування образів та кореляцій, аналогічно до навчання й розуміння, які застосовують люди [6].

Bootstrap — це набір інструментів з відкритим кодом, призначений для створення веб-сайтів та веб-додатків, який містить шаблони CSS та HTML для типографіки, форм, кнопок, навігації та інших компонентів інтерфейсу, а також додаткові розширення JavaScript. Він спрощує розробку динамічних веб-сайтів і веб-додатків.

Для серверного рівня було використано такі технології: мова програмування C# та платформа .NET Core, фреймворк для створення RESTful сервісів ASP.NET Core MVC, ORM для доступу до бази даних Entity Framework Core, фреймворк для машинного навчання Accord.NET [24].

C# — об'єктно-орієнтована мова програмування з безпечною системою типізації для платформи .NET та .NET Core. Синтаксис C# близький до C++ і Java. Мова має строгу статичну типізацію, підтримує поліморфізм, перевантаження операторів, вказівники на функції-члени класів, атрибути, події, властивості, винятки, коментарі у форматі XML. Нововведенням C# стала можливість легшої взаємодії, порівняно з мовами-попередниками, з кодом програм, написаних на інших мовах, що є важливим при створенні великих проектів. Якщо програми на різних мовах виконуються на платформі .NET, .NET бере на себе клопіт щодо сумісності програм. На даний момент C# визначено флагманською мовою корпорації Microsoft, бо вона найповніше використовує нові можливості .NET [7].

.NET Core — безкоштовний крос-платформний фреймворк з керованим кодом підтримуваний на Windows, Linux і Mac OSX. На відміну від .NET Framework вихідний код .NET Core є повністю відкритим. Він містить CoreCLR — повністю крос-платформну реалізацію CLR, віртуальну машину, яка керує виконанням програм

в .NET середовищі. CoreCLR поставляється з оптимізованим JIT компілятором RyuJIT. .NET Core також включає в себе CoreFX, яка представляє собою часткове відгалуження FCL. Реалізації усіх класів також відкриті. У той час як .NET Core розділяє підмножину API .NET Framework, він містить також власний API, який не є частиною .NET Framework [36].

ASP.NET Core MVC — це фреймворк для створення веб-додатків та API-інтерфейсів за допомогою застосування шаблону проектування MVC. ASP.NET Core MVC надає заснований на моделях спосіб побудови динамічних веб-сайтів, що дозволяє точно розділити рівні логіки програмного застосунку. Він надає повний контроль над розміткою, підтримує принципи TDD та використовує найновіші веб-стандарти.

Entity Framework Core — це легковісний, легкий для розширення крос-платформний фреймворк для забезпечення уніфікованого доступу до різноманітних баз даних.

Accord.NET — це фреймворк для виконання наукових розрахунків на платформі .NET. Фреймворк складається з набору бібліотек, які мають відкритий вихідний код [28]. Головні сфери використання включають в себе лінійну алгебру, методи оптимізації, статистику, машинне навчання, штучні нейронні мережі, обробку зображень.

Для рівня бази даних було обрано PostgreSQL.

PostgreSQL — об'єктно-реляційна система керування базами даних. Порівняно з іншими проектами з відкритим кодом, такими як Apache, FreeBSD або MySQL, PostgreSQL не контролюється якоюсь однією компанією, її розробка можлива завдяки співпраці багатьох людей та компаній, які хочуть використовувати цю СКБД та впроваджувати в неї найновіші досягнення [24].

### 3.5 Обґрунтування вибору програмної реалізації

При проектуванні системи було вивчено та проаналізовано предметну область та вимоги замовника. Після ретельного аналізу було вирішено розроблювати програмний продукт, який заснований на веб-технологіях для використання за допомогою веб-браузера.

Технології, які використовуються на сервері, були обрані за принципом зручності у використанні, відкритості вихідних кодів, актуальності в наш час та можливістю виконання на будь-якій операційній системі. Платформа .NET Core надає змогу створювати програми на мовах сімейства .NET, зокрема мова С#, на будь-якій операційній системі. Фреймворк ASP.NET Core MVC надає величезні можливості для створення веб-серверів будь-якої складності, забезпечуючи при цьому велику швидкодію та надійність. Також перевагою використання цих технологій є те, що їх можна запуснути у Docker контейнерах та побудувати на цьому швидке та надійне розгортання серверів з подальшим масштабуванням без створення при цьому спеціальної архітектури програмного забезпечення. Для доступу до бази даних було використано фреймворк Entity Framework Core через його зручність та універсальність. За допомогою цього фреймворку можливо абстрагуватися від конкретної реалізації бази даних і створити відповідні моделі даних. Це надає змогу зосередити всю увагу на створенні якісної та протестованої бізнес-логіки [34]. Для реалізації класифікації поз людини було обрано фреймворк Accord.NET через його багату колекцію різноманітних алгоритмів машинного навчання та статистики.

На клієнтському рівні було обрано технології, які задовольняють такі ж умови як і серверні, але з поправкою на виконання в браузері. Фреймворк Vue.js було обрано через те, що за допомогою нього можливо створювати складні графічні інтерфейси, які легко модифікувати, тестувати та розширювати в подальших циклах розробки програмного забезпечення. Мова TypeScript була обрана через те, що вона забезпечує типізацію та підтримку модулів. Також ця мова програмування надає безліч



різноманітного “синтаксичного цукру”, який дозволяє скоротити кількість зайвого коду. Також важливою властивістю мови є, що її можна скомпілювати в Javascript попередніх стандартів, які підтримуються в застарілих браузерах. Для забезпечення розпізнавання скелету людини було використано бібліотеку машинного навчання Tensorflow.js та модель OpenPose, яка побудована на архітектурі нейронної мережі MobileNet. Це дало змогу в режимі реального часу розпізнавати скелет людини, який потім передається на сервер для подальшої класифікації пози людини. Для того щоб стилізувати клієнтський додаток було вирішено використати фреймворк Bootstrap через його багату бібліотеку готових інтерфейсів та вбудовану підтримку та масштабування до розмірів мобільного телефону [40].

Базою даних було обрано PostgreSQL через те, що це проект з відкритим вихідним кодом і надзвичайно великою підтримкою з боку розробників. Також важливою властивістю є велика швидкість роботи, надійність та кількість вбудованих можливостей. Цю базу даних можливо розгорнути на будь-якій сучасній операційній системі або в Docker контейнері.

Дані технології в сукупності дають змогу збудувати якісний та надійний продукт, який захищений від патентних позовів з боку розробників, бо всі ці технології покриті ліцензіями, які виключають таку можливість і надають доступ до вихідних кодів даних проектів.

### **3.6 Обґрунтування вибору згорткової нейронної мережі для вирішення моделювання процесів визначення положення об'єктів на динамічному наборі зображень**

Згорткова нейронна мережа це клас штучних нейронних мереж, які використовують згорткові шари для фільтрації входів для виділення корисної інформації. Операція згортки включає комбінування вхідної інформації (карта

особливостей) з ядром згортки для формування трансформованої карти особливостей. Фільтри в згорткових шарах модифікуються на основі навчання параметрів, щоб виділити найбільш корисну інформацію для конкретної задачі. Згорткові мережі налаштовуються автоматично для знаходження найкращих особливостей, базованих на задачі [17].

Застосування згорткових мереж включає в себе різноманітні системи обробки зображень, голосу.

Згорткові нейронні мережі складаються з вхідного шару, вихідного шару і одного або багатьох прихованих шарів. Згорткова мережа відрізняється від звичайної нейронної мережі тим, що нейрони в цих шарах впорядковані в трьох вимірах (ширина, висота та глибина). Це дозволяє мережі трансформувати вхід на три виміри на виході. Приховані шари це комбінація згорткових шарів, шарів об'єднання, нормалізації та повнооб'єднаних шарів. Згорткова нейронна мережа використовує багато шарів для перетворення вхідних даних до більших рівнів абстракції [21].

Згорткова нейронна мережа покращує свою можливість розпізнавання для незвичайно розміщених об'єктів за допомогою використання шарів об'єднання для обмеженої інваріантності переміщення та обертання. Об'єднання також дозволяє використання більшої кількості згорткових шарів завдяки зменшенню споживання пам'яті. Шари нормалізації використовуються щоб нормалізувати більше локальних вхідних регіонів через переміщення всіх входів на шар середнього значення. Інші техніки регуляризації такі як пакетна нормалізація, де нормалізуються активації всього пакету, або виключення, де ігноруються випадково обрані нейрони протягом всього процесу навчання, також можна використовувати. Повністю-об'єднані шари мають нейрони, які функціонально схожі на шари згортки, але вони відрізняються тим, що вони з'єднані з усіма активаціями попереднього шару.

Сучасні згорткові нейронні мережі використовують 1x1 згорткові ядра, щоб зменшити споживання пам'яті щоб, в подальшому, дозволити більш ефективну обробку. Це робить згорткову нейронну мережу придатною для багатьох сфер застосування машинного навчання.



### 3.7 Архітектура згорткової нейронної мережі MobileNet

Архітектура MobileNet базована на глибинних відділених згортках, які є формою факторизованих згортках, які розділені стандартною згорткою в глибинну згортку і називається точковою згорткою. Для MobileNet глибинна згортка відноситься як єдиний фільтр для кожного вхідного каналу. Точкова згортка потім застосовується як згортка розміру  $1 \times 1$ , щоб комбінувати виходи глибинної нейронної мережі. Стандартна згортка фільтрує та комбінує входи в новий набір виходів в один крок. Глибинна відділена згортка розділяє це в два шари, відділений шар для фільтрування та окремий шар для комбінації. Ця факторизація має ефект великого зменшення розрахунків та розміру моделі [17]. На рисунках (3.5, 3.6, 3.7) зображено як стандартна згортка факторизована в глибинну та  $1 \times 1$  згортки.

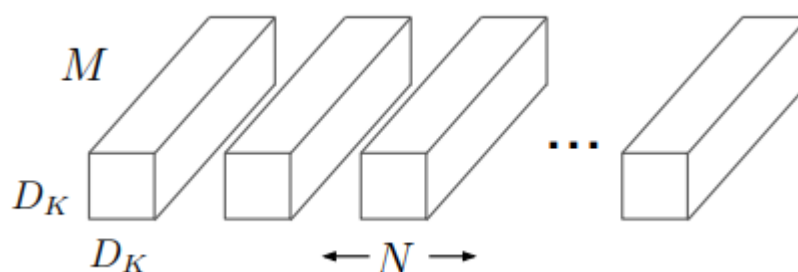


Рисунок 3.5 – Стандартні згорткові фільтри

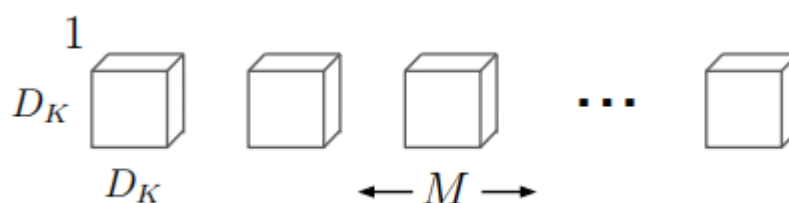


Рисунок 3.6 – Глибинні згорткові фільтри

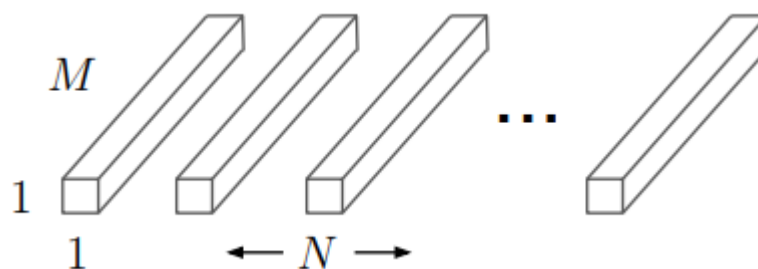


Рисунок 3.7 – Згорткові фільтри, які називаються точковими згортками в контексті глибинної відділеної згортки

Операція стандартної згортки має ефект фільтрації особливостей базованих на ядрах згортки і комбінує признаки в порядку, щоб створити нове представлення. Кроки фільтрації та комбінування можуть бути розділені на два кроки через використання факторизованих згорток, які називаються глибинними розділеними згортками для значного зменшення обчислювальної вартості. Глибинна розділена згортка складається з двох шарів: глибинних та точкових згорток [19]. Глибинні згортки використовуються для застосування єдиного фільтру для кожного вхідного каналу. Точкова згортка та проста  $1 \times 1$  згортка використовуються для створення лінійної комбінації виходів глибинного шару.

Глибинна згортка більш ефективна ніж стандартна згортка. Однак це тільки фільтрує вхідні канали, воно не комбінує їх для створення нових особливостей. Отже, додатковий шар, який комбінує лінійну комбінацію виходів глибинної згортки через  $1 \times 1$  згортку, необхідний для генерації цих нових особливостей.

Комбінація глибинної згортки та  $1 \times 1$  (точкової) називається глибинною відділеною згорткою.

Структура MobileNet побудована на глибинних відділених згортках за винятком першого шару, який є повною згорткою. На рисунку 3.8 зображено відмінність між шаром з звичайними згортками та згортками, які використовуються для архітектури MobileNet [39].

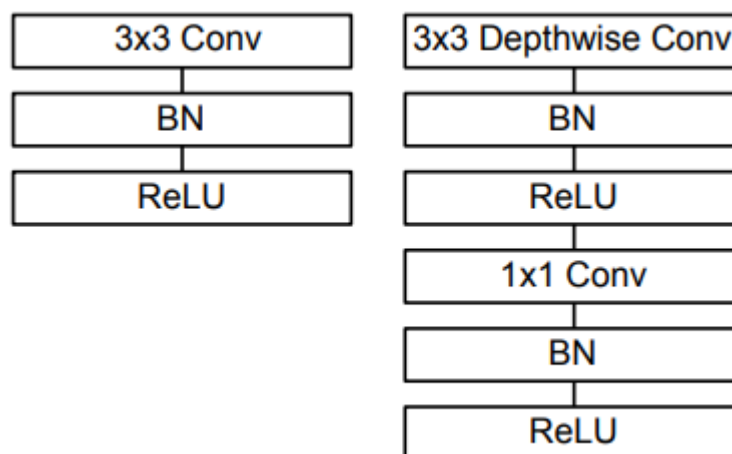


Рисунок 3.8 – Стандартний згортковий шар (зліва), глибинний відділений шар (праворуч)

Зменшення розмірності виконуються за допомогою крокового використання глибинних згорток так само, як і на першому шарі. Фінальне середнє об'єднання зменшує просторову роздільну здатність до 1 перед повно-об'єднаним шаром. В сумі MobileNet має 28 шарів.

MobileNet може використовуватися як ефективна база для сучасних систем розпізнавання об'єктів, для великомасштабної геолокації, розпізнавання рис обличчя.

### **3.8 Обґрунтування вибору лінійної регресії для вирішення моделювання процесів визначення положення об'єктів на динамічному наборі зображень**

Регресія — це метод моделювання цільового значення, який базований на незалежних предикторах. Цей метод найбільш використовується для прогнозування та знаходження причини та взаємозв'язку між змінними [28]. Техніки регресії в основному відрізняються кількістю незалежних змінних та типом взаємозв'язку між залежними та незалежними змінними.

Проста лінійна регресія — це тип регресійного аналізу де кількість незалежних змінних дорівнює одному, а також де є лінійний взаємозв'язок між незалежною та залежною змінними. На рисунку 3.9 зображений графік роботи лінійної регресії. Червоною лінією на зображеному графіку позначено найбільш точне приближення лінійної регресії. Ця лінія моделюється за допомогою лінійного рівняння прямої.

Головною задачею алгоритму лінійної регресії є знайти найкращі коефіцієнти рівняння прямої.

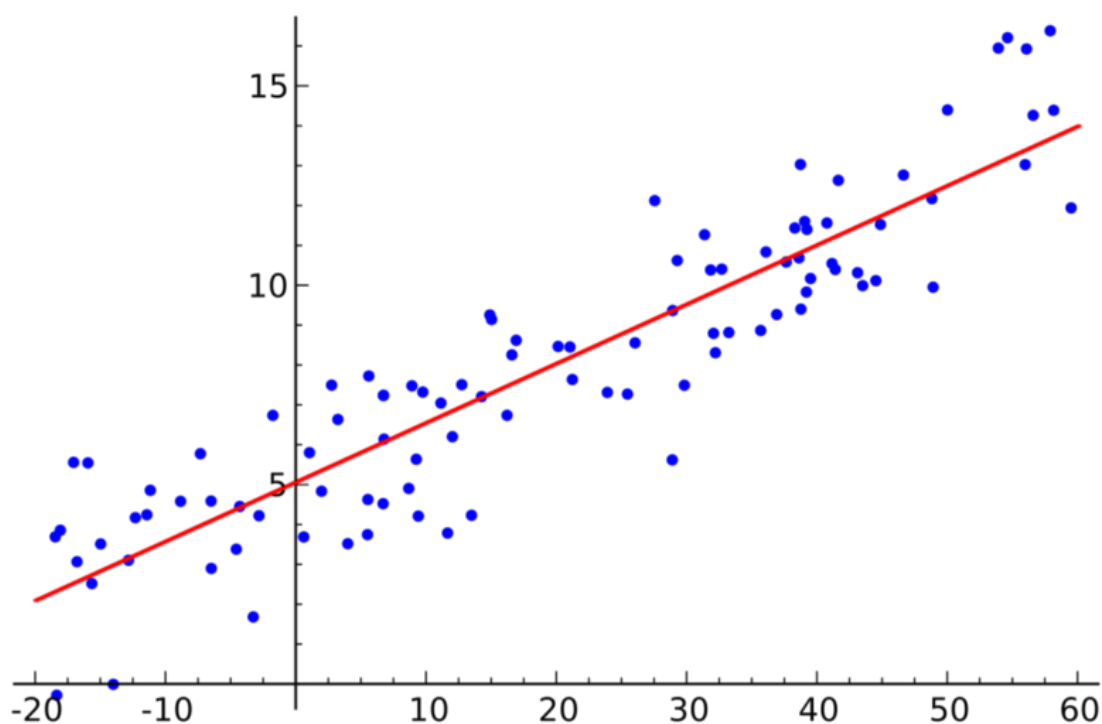


Рисунок 3.9 – Лінійна регресія

Функція вартості дозволяє знайти найбільш ймовірні значення для коефіцієнтів рівняння прямої, які дозволять забезпечити найкраще приближення для вхідних даних. Для того щоб знайти ці коефіцієнти потрібно застосувати мінімізацію похибки між правильним та прогнозованим значенням.

Різниця між прогнозованими значеннями та правильними значеннями — це похибка. Якщо піднести до квадрату, просумувати усі значення і розділити на загальну кількість усіх точок даних, можна отримати середньоквадратичну похибку на всіх вхідних даних [14].

Для того, щоб мінімізувати похибку та знайти коефіцієнти рівняння прямої, можна використати алгоритм градієнтного спуску. Ідея полягає в тому, що навчання починається з деяких значень коефіцієнтів і потім ці значення змінюються ітеративно, щоб зменшити похибку. На рисунку 3.10 зображена проблема вибору кроку для алгоритму градієнтного спуску.

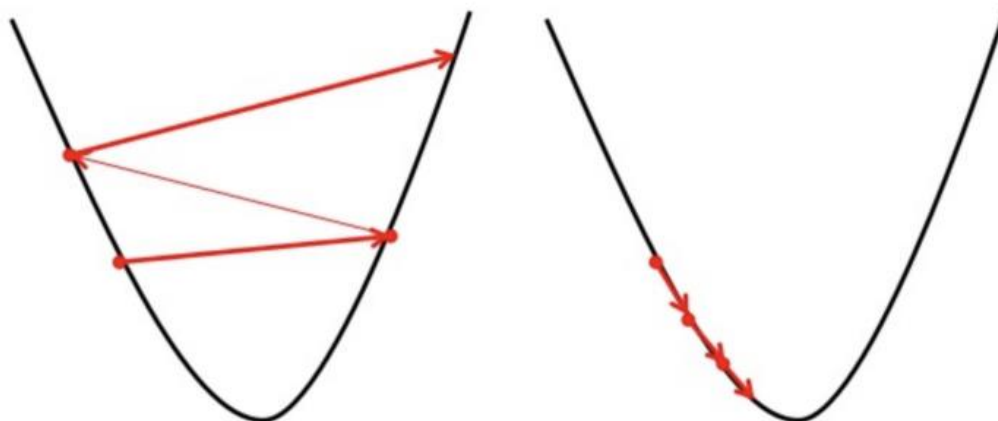


Рисунок 3.10 – Велика (зліва) та мала (праворуч) швидкість навчання

Якщо обрати занадто великий крок, то можна пропустити мінімум функції і результат навчання буде незадовільний. Якщо обрати занадто малий крок навчання, то за певну обмежену кількість кроків можна так і не потрапити в мінімум функції і також результат навчання буде незадовільний [23].

Також існує проблема випуклої та невивуклої функції мінімізації, яка зображена на рисунку 3.11.

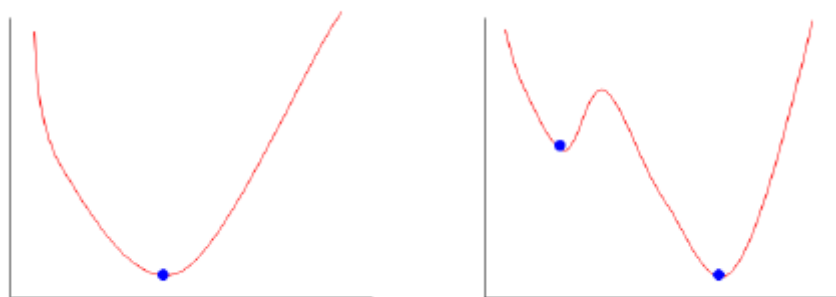


Рисунок 3.11 – Випукла (зліва) та невивукла (праворуч) функція мінімізації



Проблема полягає в тому, що градієнтний спуск може потрапити в локальний мінімум функції мінімізації та не знайти глобальний, в такому випадку результат навчання також може бути незадовільним.

### **Висновки до розділу 3**

В результаті аналізу предметної області та вимог, а також вимог, які були висунуті при проектуванні, було обрано та обгрунтовано засоби реалізації програмного продукту. Сформовано набір технологій, які будуть використовуватися для створення серверу, клієнту, бази даних, для розпізнавання положення об'єкта та виділення скелету людини, для класифікації пози людини та сповіщення користувача про обрані пози через електронну пошту.

## 4. ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

Система розпізнавання поз людини на відеопотоці буде складатися з модулів, кожен з яких розбивається на різну кількість функціональних підблоків. Головним модулем є кабінет користувача, який одночасно є елементом компонування системи. Така структура, дозволяє легко та інтуїтивно користуватися системою за рахунок того, що модулі розташовуються у порядку, який зазвичай використовують для вирішення схожих задач.

На рисунку 4.1 наведена схема структури системи, на якій розташовані всі програмні модулі.

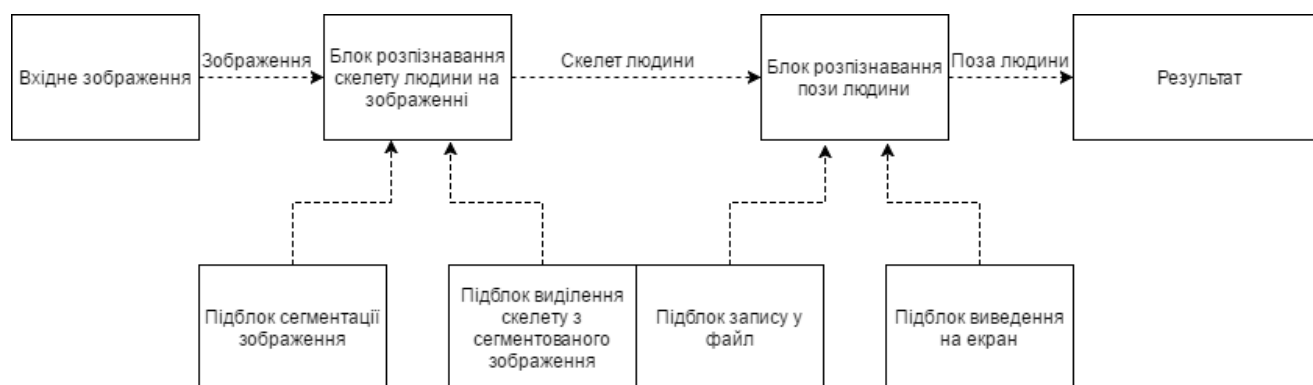


Рисунок 4.1 — Схема структури системи

### 4.1 Опис функціональності системи

Програмний застосунок для виділення скелету та розпізнавання пози людини на відеопотоці містить у собі одного головного актора – користувач системи;

На рисунку 4.2 представлена діаграма прецедентів, яка описує функції та дії актора у системі.

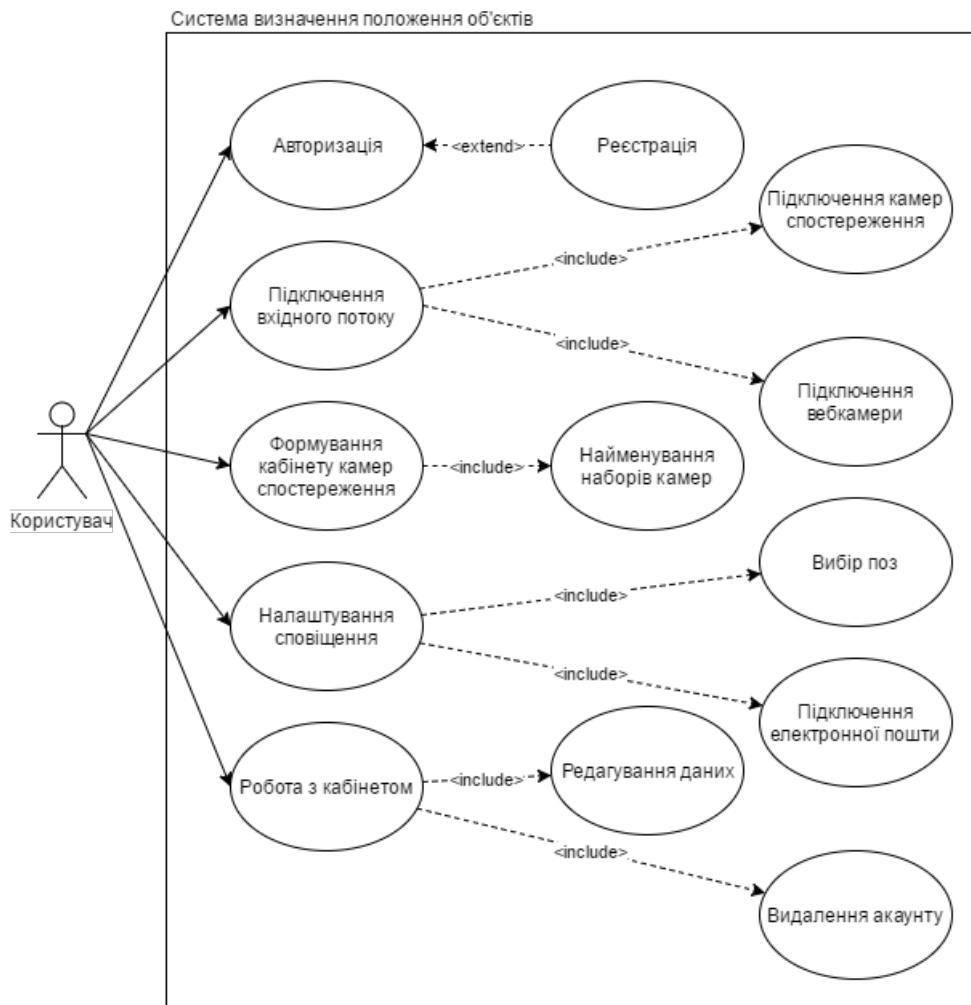


Рисунок 4.2 — Діаграма прецедентів системи

## 4.2 Концептуальна модель бази даних

База даних системи складається з трьох взаємопов'язаних таблиць реляційної бази даних, які створюють єдиний інформаційний простір для зберігання та отримання доступу до даних.

Головною таблицею є “Користувач”, яка містить у собі інформацію про користувача системи, основну інформацію про нього та індивідуальні налаштування.

Таблиця “Набір камер” містить основну інформацію про кожен конкретний набір камер відеоспостереження, його назву тощо.

Таблиця “Камера” містить інформацію про конкретну камеру відеоспостереження, посилання на ресурс, з якого буде поступати відеопоток в систему тощо.

Концептуальна модель бази даних приведена на рисунку 4.3.

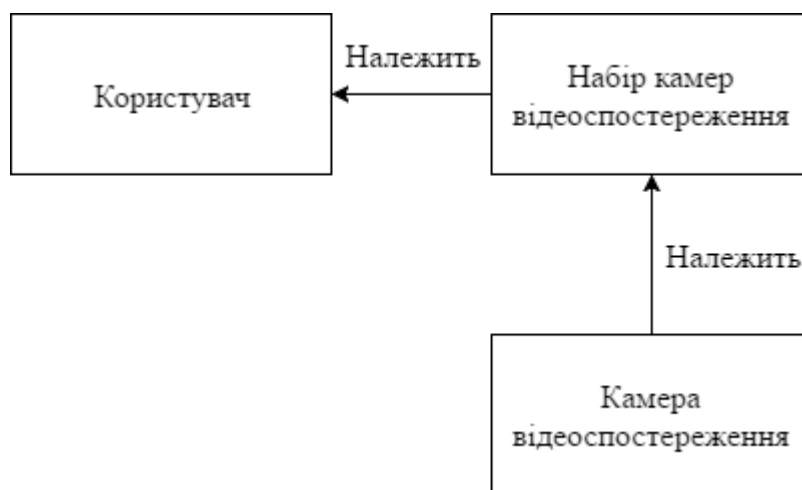


Рисунок 4.3 — Концептуальна модель БД

### 4.3 Опис таблиць бази даних

Для доступу до даних із бази даних для кожної таблиці створюється звичайний клас на мові С# з публічними властивостям, який є об'єктом відображенням таблиці в базі даних.

DbContext забезпечує об'єктно-орієнтований інтерфейс для доступу і маніпулювання даними, що зберігаються в базах даних. Клас DbSet відповідає таблиці в базі даних, поле моделі являє собою значення окремого стовпця рядка.

Також використання об'єктно-реляційної проекції дозволяє зручно отримувати данні з таблиць, які мають зв'язок “багато-до-багатьох” та “один-до-багатьох”.

База даних системи реалізована за допомогою ORM Microsoft Entity Framework Core.

Застосування даного фреймворку дозволяє абстрагуватися від конкретної реалізації бази даних і зосередитися на створенні правильної моделі даних та взаємозв'язку між компонентами даної моделі.

Розглянемо більш детально структури кожної із таблиць бази даних системи розпізнавання пози людини на відеопотоці.

Детальна інформація про їх структури (ім'я, тип і розмір поля, опис поля) приведена у таблицях 4.1 — 4.3.

Таблиця 4.1. Структура таблиці “Користувач”

Ім'я поля	Тип і розмір поля	Опис поля
Id	nvarchar(450)	Первинний ключ
UserName	nvarchar(100)	Логін користувача
Email	nvarchar(100)	Адреса електронної пошти
PasswordHash	nvarchar(10000)	Хеш паролю

Таблиця 4.2. Структура таблиці “Камера”

Ім'я поля	Тип і розмір поля	Опис поля
Id	nvarchar(450)	Первинний ключ
Url	nvarchar(255)	Адреса ресурсу з відеопотоком
CameraSetId	nvarchar(450)	Id набору камер

Таблиця 4.3. Структура таблиці “Набір камер”

Ім'я поля	Тип і розмір поля	Опис поля
Id	nvarchar(450)	Первинний ключ
Name	nvarchar(100)	Назва набору камер
UserId	nvarchar(450)	Id користувача

## 4.4 Етапи роботи програми для розпізнавання положення об'єкту

Для того, щоб розпізнати положення об'єкту на зображенні необхідно провести підготовку даних і застосувати набір алгоритмів машинного навчання для коректного розпізнавання. Процес обробки та розпізнавання положення зображень на рисунку 4.4.

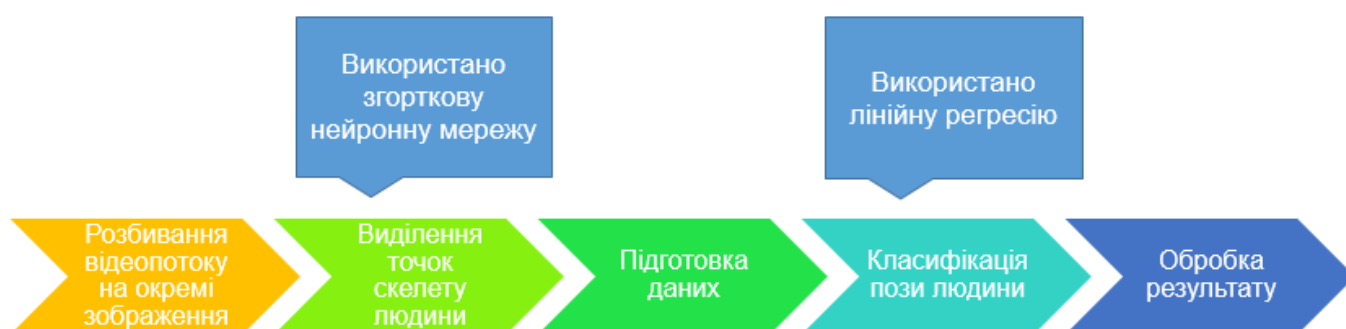


Рисунок 4.4 — Етапи розпізнавання положення об'єкту

На вхід програми подається відеопотік, який необхідно розбити на окремі кадри для подальшої обробки. Тому на першому етапі відбувається дискретизація вхідного відео. Після того, як відео було розбите на зображення окремі кадри передаються на сервер для подальшої обробки.

На другому етапі обробки відбувається виділення точок скелету людини на зображенні. Для цього етапу була обрана згорткова нейронна мережа, а також архітектура MobileNet. Після цього етапу з вхідного зображення отримується масив точок скелету людини, яка зображена на кадрі.

Після того, як отримано масив точок скелету, відбувається підготовка цих даних у формат, який придатний для класифікації пози людини. Дані формуються у спеціальний пакет даних, який являє собою ряд чисел, які представляють координати точок скелету.

На етапі класифікації пози людини використовується лінійна регресія, яка була попередньо навчена розпізнавати задані пози людини. У результаті розпізнавання на виході отримується поза людини.

На останньому етапі відбувається обробка результату класифікації пози. У даному випадку відбувається аналіз пози і попередження користувача засобами електронної пошти про настання обраної пози.

## **4.5 Розробка кабінету користувача**

Кабінет користувача – це основне робоче місце користувача в системі. Він включає в себе інші підмодулі, які виконують основні функції системи, та є елементом компонування в системі.

До цих модулів належать наступні:

- модуль керування наборами камер відеоспостереження;
- модуль розпізнавання поз людини на відеопотоці;
- модуль сповіщення про надзвичайні пози.

### **4.5.1 Модуль керування наборами камер відеоспостереження**

Даний модуль дозволяє користувачу додатку керувати наборами камер відеоспостереження, а саме:

- створювати нові набори камер відеоспостереження;
- видаляти старі набори камер відеоспостереження
- редагувати вже створені набори камер відеоспостереження;
- додавати камери до наборів камер спостереження;

- видаляти камери з наборів камер спостереження;
- переглядати відеопотік в режимі реального часу;
- вмикати або вимикати сповіщення про пози людини.

#### **4.5.2 Модуль розпізнавання поз людини на відеопотоці**

Даний модуль виконує всю роботу пов'язану з розпізнаванням пози людини. Якщо користувач хоче розпізнавати пози з потоку відеокамери, то для коректної роботи даного модуля необхідний відповідний дозвіл в браузері. Основні функції даного модуля:

- розбивання вхідного відеопотоку на окремі кадри;
- розпізнавання скелету людини;
- підготовка пакету даних для класифікації;
- класифікація пози людини;
- візуалізація отриманих результатів.

#### **4.5.3 Модуль сповіщення про надзвичайні пози**

Даний модуль аналізує отриману з модуля розпізнавання інформацію та в залежності від заданих налаштувань сповіщає користувача про надзвичайні пози людини засобом електронної пошти. Також наявна можливість активації або вимкнення даного режиму, якщо користувача цікавить тільки результат розпізнавання поз людини на вхідному відеопотоці.



## **Висновки до розділу 4**

У результаті проектування та розробки програмного продукту, який дозволяє розпізнавати положення об'єкту, було надано користувачеві можливість через власний кабінет створювати, редагувати та видаляти власні набори відеокамер, спостерігати за процесом розпізнавання в режимі реального часу, а також налаштовувати засоби сповіщення через електронну пошту. Було створено базу даних, яка використовується для зберігання даних. Також створено клієнтський додаток, який надає графічний інтерфейс для користувача, а також серверний додаток, який виконує основну роботу, пов'язану з розпізнаванням, класифікацією поз людини та сповіщенням користувача через електронну пошту.

## **5. МЕТОДИКА РОБОТИ КОРИСТУВАЧА З ПРОГРАМНОЮ СИСТЕМОЮ**

Програмний комплекс розроблений з використанням веб-технологій і тому працює в браузерях, які підтримують актуальні веб-стандарти.

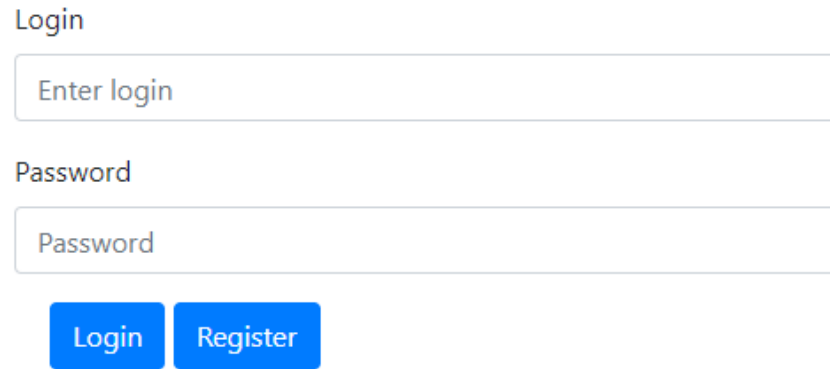
### **5.1 Інсталяція та системні вимоги**

Оскільки застосунок працює з використанням веб-технологій, тому не потребує встановлення на пристрій користувача. Проте, для використання необхідний веб-браузер, який підтримує актуальні веб-стандарти. Також необхідно мати стабільний доступ до інтернету з швидкістю не менше ніж 50 мбіт за секунду. Для використання можливостей програми з розпізнавання пози людини з вебкамери необхідно надати клієнтському додатку відповідний дозвіл.

### **5.2 Інструкція з використання програмного продукту**

При вході на клієнтський додаток, користувачеві необхідно авторизуватися в системі щоб почати працювати в системі. На рисунку 5.1 зображена форма авторизації.

Авторизація застосовується для обмеження прав незареєстрованого користувача, а також для забезпечення персонального кабінету користувача, в якому він може налаштовувати власні набори відеокамер.



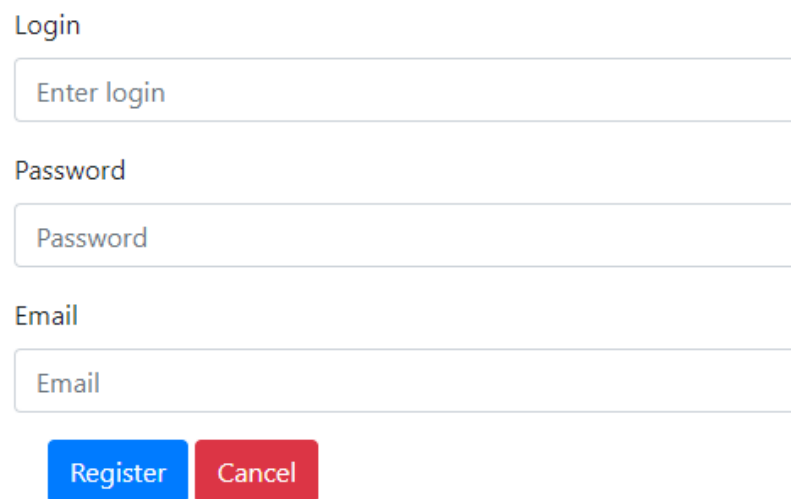
Login

Password

Login Register

Рисунок 5.1 — Форма авторизації

У випадку якщо користувач ще ніколи не користувався системою або хоче створити новий профіль, то йому необхідно зареєструватися в системі. На рисунку 5.2 зображена форма реєстрації.



Login

Password

Email

Register Cancel

Рисунок 5.2 — Форма реєстрації

Після того, як користувач авторизувався в системі, він отримує доступ до головного меню системи. За допомогою цього меню користувач має доступ до усіх сторінок системи, або має можливість вийти з свого профілю. Також він може перейти до свого основного робочого простору – кабінету користувача. На рисунку 5.3 зображене головне меню системи.

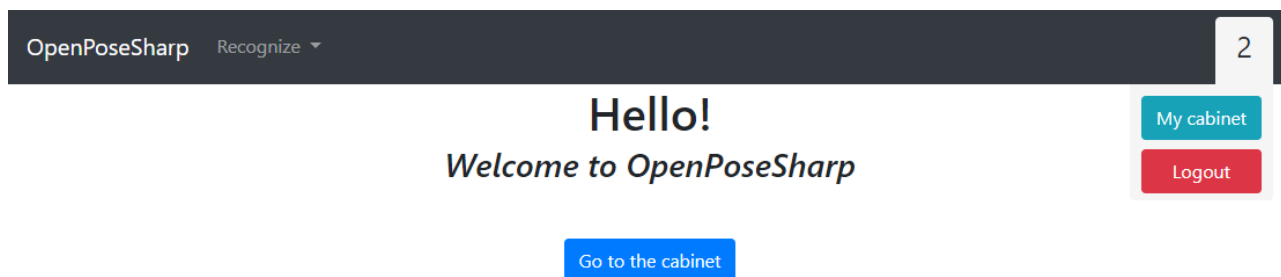


Рисунок 5.3 — Головне меню системи

Одним із головних компонентів кабінету користувача є меню керування наборами камер відеоспостереження. За його допомогою користувач має змогу створювати, редагувати та видаляти набори камер відеоспостереження. На рисунку 5.4 зображене меню керування наборами камер відеоспостереження.

### My cabinet

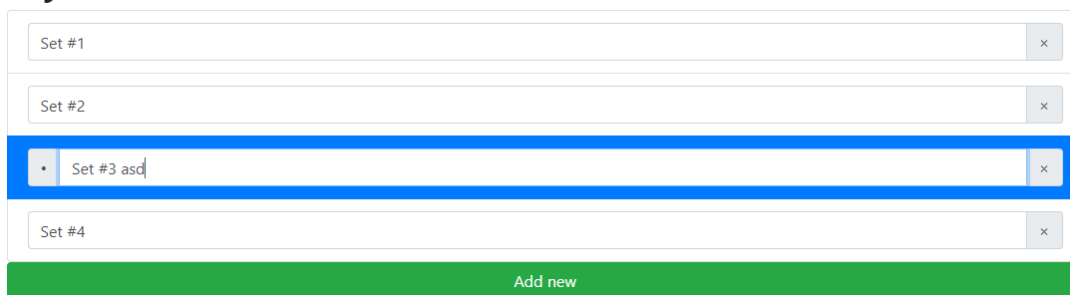


Рисунок 5.4 — Меню керування наборами камер відеоспостереження

Кожен набір може складатися з багатьох камер відеоспостереження. Головною функціональною одиницею системи є вікно перегляду та налаштування камери відеоспостереження. З його допомогою, користувача має змогу переглядати в режимі реального часу відеопотік і результати роботи модуля розпізнавання пози людини на відеопотоці. Також користувач має змогу вмикати модуль оповіщення про надзвичайні пози людини засобом електронної пошти. На рисунку 5.5 зображене головне вікно керування камерою відеоспостереження.

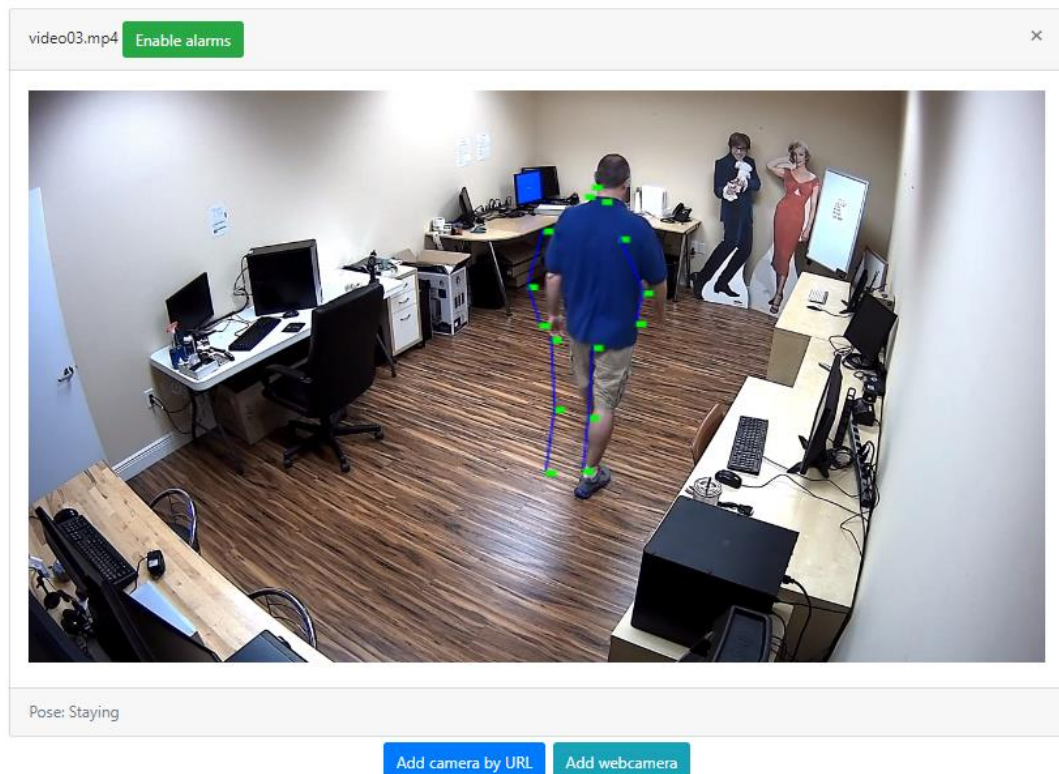


Рисунок 5.5 — Головне вікно керування камерою відеоспостереження

## Висновки до розділу 5

Після розробки було створено програмне забезпечення, яке дозволяє моделювати процеси визначення положення об'єктів на динамічному наборі зображень. Основними функціями є зручне управління наборами камер і налаштування засобів сповіщення користувача за допомогою зручного графічного інтерфейсу, який заснований на веб-технологіях. Користувач має змогу створити власний профіль, в якому буде зберігатися інформація про набори камер.

## 6. СТАРТАП ПРОЕКТ

Розділ має на меті проведення маркетингового аналізу стартап проекту задля визначення принципової можливості його ринкового впровадження та можливих напрямів реалізації цього впровадження [35]. Проведення маркетингового аналізу передбачає виконання нижченаведених кроків.

### 6.1 Основні ідеї проекту

В межах підпункту слід проаналізувати та подати у вигляді таблиць:

- зміст ідеї (що пропонується);
- можливі напрямки застосування;
- основні вигоди, що може отримати користувач товару;
- чим відрізняється від існуючих аналогів та замінників.

Перші три пункти подаються у вигляді таблиці (таблиця 6.1) і дають цілісне уявлення про зміст ідеї та можливі базові потенційні ринки.

Таблиця 6.1. Опис ідеї стартап-проекту

<b>Зміст ідеї</b>	<b>Напрямки застосування</b>	<b>Вигоди для користувача</b>
Розпізнавання положення об'єктів на динамічному наборі зображень	Системи спостереження	Збільшення точності реакції на різноманітні події
	Спортивні події	Збирання статистики і реакція на події
	Медичні заклади та заклади догляду за людьми	Розпізнавання та реакція на надзвичайні події

Аналіз потенційних техніко-економічних переваг ідеї (чим відрізняється від існуючих аналогів та замінників) порівняно із пропозиціями конкурентів передбачає:

— визначення переліку техніко-економічних властивостей та характеристик ідеї [36];

— визначення попереднього кола конкурентів (проектів-конкурентів) або товарів-замінників чи товарів-аналогів, що вже існують на ринку, та проводиться збір інформації щодо значень техніко-економічних показників для ідеї власного проекту та проектів-конкурентів відповідно до визначеного вище переліку;

— проводиться порівняльний аналіз показників: для власної ідеї визначаються показники, що мають а) гірші значення (W, слабкі); б) аналогічні (N, нейтральні) значення; в) кращі значення (S, сильні) (таблиця 6.2).

Таблиця 6.2. Визначення сильних, слабких та нейтральних характеристик

№ п/п		(потенційні) товари/концепції конкурентів		
		Мій проект	Realtime Multi-Person Pose Estimation	Paper
1	W слабка сторона	Швидкодія	Відсутність класифікації поз	Відсутність класифікації поз
2		Високі вимоги до серверу	Високі апаратні вимоги	Відсутність графічного інтерфейсу
3	N нейтральна сторона	Мультиплатформність	Мультиплатформність	Мультиплатформність
4	S сильна сторона	Наявність графічного інтерфейсу	Наявність графічного інтерфейсу	Точність розпізнавання точок скелету
5		Можливість повідомлення користувача через електронну пошту	Швидкодія	Швидкодія

## 6.2 Технологічний аудит ідеї проекту

В межах даного підрозділу необхідно провести аудит технології, за допомогою якої можна реалізувати ідею проекту [13]. Визначення технологічної здійсненності ідеї проекту передбачає аналіз таких складових (таблиця 6.3):

- за якою технологією буде виготовлено товар згідно ідеї проекту;
- чи існують такі технології, чи їх потрібно розробити/доробити;
- чи доступні такі технології авторам проекту.

Таблиця 6.3. Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології реалізації	Наявність технологій	Доступність технологій
1	Веб-інтерфейс користувача	HTML + Vue.JS + CSS	Наявна	Доступна безкоштовно
2	Крос-платформений сервер	ASP.NET Core	Наявна	Доступна безкоштовно
3	Розпізнавання точок скелету людини	TensorFlow + MobileNet	Наявна	Доступна безкоштовно
4	Класифікація пози людини	Accord.NET	Наявна	Доступна безкоштовно
<p>Висновок: проект реалізувати можливо.          Обрана технологія реалізації ідеї проекту: Веб-інтерфейс користувача, крос-платформений сервер, розпізнавання точок скелету людини, класифікація пози людини</p>				

За результатами аналізу таблиці робиться висновок щодо можливості технологічної реалізації проекту: так чи ні, а також технологічного шляху, яким це доцільно зробити (з поміж названих технологій обираються такі, що доступні авторам проекту та є наявними на ринку).



### 6.3 Аналіз ринкових можливостей запуску стартап-проекту

Визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів [31].

Спочатку проводиться аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (таблиця 6.4).

Таблиця 6.4. Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	2
2	Загальний обсяг продаж, грн/ум.од	1000 грн
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Немає

Середня норма рентабельності в галузі (або по ринку) порівнюється із банківським відсотком на вкладення. За умови, що останній є вищим, можливо, має сенс вкласти кошти в інший проект.

За результатами аналізу таблиці робиться висновок щодо того, чи є ринок привабливим для входження за попереднім оцінюванням.

Надалі визначаються потенційні групи клієнтів, їх характеристики, та формується орієнтовний перелік вимог до товару для кожної групи (таблиця 6.5).

Таблиця 6.5. Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Розпізнавання положення об'єктів на динамічному наборі зображень	Медичні заклади, заклади догляду за людьми, спортивні організатори	Особливості купівлі: компанії заключають довготривалі договори, а стартапери віддають перевагу пробному терміну Використання: компанії вимагають точну роботу програмного продукту, а також широкий набір функцій	Стабільність роботи Невисока ціна Наявність пробного періоду Наявність документації Підтримка необхідних платформ Точність роботи

Після визначення потенційних груп клієнтів проводиться аналіз ринкового середовища: складаються таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають (таблиці 5.6-5.7).

Таблиця 6.6. Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Підходить для нових проектів	Для існуючих проектів виникне потреба переносити базу даних. Продукт більше підходить для нових проектів	Додавання можливості автоматизованого імпорту з різних типів сховищ
2	Обмеженість функцій	Інструмент обмежений наявними функціями і не має деяких функцій, які мають конкуренти	Додавання нових функцій за потреби

Таблиця 6.7. Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Популярність веб застосунків	Веб-індустрія постійно набирає обертів	Вихід на глобальний ринок
2	Потреба у використанні баз даних	SQL бази даних використовуються у більшості програмних застосунків	Використання ORM для баз даних
3	Відсутність повноцінних альтернатив	Існуючі альтернативи не надають повний набір можливостей	Розширення набору функцій

Надалі проводиться аналіз пропозиції: визначаються загальні риси конкуренції на ринку. Аналіз пропозиції необхідно виконати аналізуючи існуючі види конкуренцій [5].

Пропозиції повинні відповідати на питання “Як просувати продукт”.

Аналіз пропозицій зображено на таблиці 6.8.

Таблиця 6.8. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції - монополія/олігополія/ монополістична/чиста	чиста	Прямі договори з компаніями, презентація продукту на виставках
2. За рівнем конкурентної боротьби - локальний/національний /...	національний	Публікація статей на міжнародних сайтах

3. За галузевою ознакою - міжгалузева/ внутрішньогалузева	внутрішньогалузева	Розвивати напрямки, нерозвинуті конкурентами
4. Конкуренція за видами товарів: - товарно-родова - товарно-видова - між бажаннями	товарно-видова	Розповідати про свої переваги перед конкурентом у цій галузі
5. За характером конкурентних переваг - цінова / нецінова	нецінова	Надання функцій, які не надають конкуренти
6. За інтенсивністю - марочна/не марочна	марочна	Надання функцій, які не надають конкуренти

Таблиця 6.9. Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари- замінники
	- Realtime Multi-Person Pose Estimation - Paper	- ML.NET	Мінімізація витрат часу постачальників	Контроль якості	Лояльність споживачів
<b>Висновки</b> :	Визначити інтенсивність конкурентної боротьби з боку прямих конкурентів	Є можливості виходу на ринок, оскільки існуючі рішення не надають потрібних переваг	Постачальники підлаштовуютьс я під ринок	Клієнти диктують вимоги згідно з умовами експлуатації	Обмеженн я для роботи на ринку через товари замінники

На основі аналізу конкуренції, проведеного в п. 3.5 (таблиця 6.9), а також із урахуванням характеристик ідеї проекту (таблиця 6.2), вимог споживачів до товару (таблиця 6.5) та факторів маркетингового середовища (таблиця 6.6-6.7) визначається

та обґрунтовується перелік факторів конкурентоспроможності. Аналіз оформлюється за таблицею 10.

Таблиця 6.10. Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Сповіщення користувача засобами електронної пошти	Існуючі конкуренти не мають можливості сповіщення користувача засобами електронної пошти про настання обраних поз
2	Класифікація пози людини	Існуючі конкуренти не мають можливості класифікувати пози людини на динамічному наборі зображень

За визначеними факторами конкурентоспроможності (таблиця 6.10) проводиться аналіз сильних та слабких сторін стартап-проекту (таблиця 6.11).

Таблиця 6.11. Порівняльний аналіз сильних та слабких сторін

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з даним продуктом							
			-3	-2	-1	0	1	2	3	
1	Сповіщення користувача засобами електронної пошти	20	+							
2	Класифікація пози людини	10			+					

Фінальним етапом ринкового аналізу можливостей впровадження проекту є складання SWOT-аналізу (матриці аналізу сильних (Strength) та слабких (Weak) сторін, загроз (Troubles) та можливостей (Opportunities) (таблиця 6.12) на основі виділених ринкових загроз та можливостей, та сильних і слабких сторін (таблиця 6.11).

Перелік ринкових загроз та ринкових можливостей складається на основі аналізу факторів загроз та факторів можливостей маркетингового середовища. Ринкові загрози та ринкові можливості є наслідками (прогнозованими результатами) впливу факторів, і, на відміну від них, ще не є реалізованими на ринку та мають певну ймовірність здійснення.

Таблиця 6.12. SWOT-аналіз стартап-проекту

<b>Сильні сторони:</b> Сповіщення користувача Класифікація поз людини	<b>Слабкі сторони:</b> Швидкодія Високі вимоги до серверу
<b>Можливості:</b> Популярність веб платформи Відсутність повноцінних альтернатив	<b>Загрози:</b> Сумісність з старими проектами Обмеженість функцій

Перелік ринкових загроз та ринкових можливостей складається на основі аналізу факторів загроз та факторів можливостей маркетингового середовища. Ринкові загрози та ринкові можливості є наслідками (прогнозованими результатами) впливу факторів, і, на відміну від них, ще не є реалізованими на ринку та мають певну ймовірність здійснення.

Таблиця 6.13. Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Орієнтація поточної моделі на ринок медичних закладів	50 %	40 год
2	Орієнтація поточної моделі на ринок спортивних заходів	20 %	160 год
3	Орієнтація поточної моделі на ринок догляду за людьми	10 %	200 год

Альтернатива, де отримання ресурсів є більш простим та ймовірним – №4 "Переорієнтація на генерацію серверної частини", що становить 80 відсотків. Це значення перевищує інші альтернативи.

Альтернатива, де строки реалізації є більш стислими – №1 "Орієнтація поточної моделі на ринок стартаперів". Терміни реалізації в цьому разі становлять лише 40 годин.

## 6.4 Розроблення ринкової стратегії проекту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: опис цільових груп потенційних споживачів (таблиця 6.14).

За результатами аналізу потенційних груп споживачів (сегментів) автори ідеї обирають цільові групи, для яких вони пропонуватимуть свій товар, та визначають стратегію охоплення ринку [25].

Таблиця 6.14. Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Медичні заклади	Готові	Високий	Висока	Просто
2	Закладу догляду за людьми	Потребують недовгих переговорів	Середній	Середня	Складно
3	Спортивні організатори	Потребують переговорів	Низький	Низька	Дуже складно
Які цільові групи обрано: медичні заклади					

Для роботи в обраних сегментах ринку необхідно сформуванати базову стратегію розвитку (таблиця 6.15).

Таблиця 6.15. Визначення базової стратегії розвитку

№ п/ п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентос- проможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	Орієнтація поточної моделі на ринок медичних закладів	Стратегія концентрованого маркетингу	Медичні заклади потребують точності розпізнавання	Стратегія спеціалізації (спирається на диференціацію)
2	Орієнтація поточної моделі на ринок спортивних організаторів	Стратегія концентрованого маркетингу	Спортивні організатори потребують точності розпізнавання та широкий набір вбудованих в програмний продукт функцій	Стратегія спеціалізації (спирається на диференціацію)

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: опис цільових груп потенційних споживачів.

Перелік ринкових загроз та ринкових можливостей складається на основі аналізу факторів загроз та факторів можливостей маркетингового середовища. Після визначення потенційних груп клієнтів проводиться аналіз ринкового середовища: складаються таблиці факторів, що сприяють ринковому впровадженню проекту.

Наступним кроком є вибір стратегії конкурентної поведінки (таблиця 6.16).



Таблиця 6.16. Визначення базової стратегії конкурентної поведінки

No п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1	Ні	шукати нових споживачів	Так, потрібно збільшити швидкодію програмного продукту та збільшити набір вбудованих в програмний продукт функцій, потрібно зменшити кількість необхідних ресурсів для розміщення серверу, також треба збільшити підтримку мобільних пристроїв для охоплення нового ринку користувачів	Стратегія заняття конкурентної ніші

З обраних сегментів до постачальника (стартап-компанії) та до продукту розробляється стратегія позиціонування (таблиця 6.17), що полягає у формуванні ринкової позиції, за яким споживачі мають ідентифікувати торговельну марку/проект.

Таблиця 6.17. Визначення стратегії позиціонування

No п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкуренто-спроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувану комплексну позицію власного проекту (три ключових)
1	Стабільність роботи Невисока ціна Наявність документації Підтримка багатьох платформ	Стратегія спеціалізації (спирається на диференціацію)	Медичні заклади потребують швидкості розробки, яку надає підтримка багатьох платформ даним продуктом	пришвидшення розробки ПЗ підтримка декількох платформ.

## 6.5 Розроблення маркетингової програми проекту

Для цього у таблиці 6.18 потрібно підсумувати результати попереднього аналізу конкурентоспроможності товару.

Таблиця 6.18. Визначення ключових переваг концепції потенційного товару

No п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Пришвидшення розробки ПЗ	Підтримка декількох платформ	Більшість конкуренти підтримують лише одну платформу.
2	Використання ORM	Використовується ORM для доступу до БД	Конкурент Paper не використовує ORM

Надалі розробляється трирівнева маркетингова модель товару: уточнюється ідея продукту та/або послуги, його фізичні складові, особливості процесу його надання (таблиця 6.19).

Таблиця 6.19. Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
<b>I. Товар за задумом</b>	Розпізнавання положення об'єктів на динамічному наборі зображень		
<b>II. Товару реальному виконанні</b>	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	можливість оптимізації витрат часу	М	Тл
	можливість оптимізації витрат коштів	М	Вр
	відповідність актуальним технологіям	Нм	Тх
	Відповідає вимогам ДСТУ ISO/IEC 25030:2015 Програмна інженерія. Вимоги щодо якості та оцінювання програмного продукту (SQuaRE). Вимоги щодо якості		
	Пакування: готовий до використання інсталятор		
	Марка: Installer Generator		
<b>III. Товар із підкріпленням</b>	Потенційний користувач може ознайомитись з поточним товаром з наукових конференцій та публічних виступів, а також наукових вісників на яких була представлена інформація про даний продукт		
За рахунок чого потенційний товар буде захищено від копіювання: Назва і контент захищені ліцензією МІТ; захист інтелектуальної власності			

М/Нм – монотонні або немонотонні;

Вр/Тх/Тл/Е/Ор – вартісні, технічні, технологічні, ергономічні або органолептичні (останній – для продуктів харчування)

Після формування маркетингової моделі товару слід особливо відмітити – чим саме проект буде захищено від копіювання. Захист може бути організовано за рахунок захисту ідеї товару (захист інтелектуальної власності), або ноу-хау, чи комплексне поєднання властивостей і характеристик, закладене на другому та третьому рівнях товару [37].

Наступним кроком є визначення цінових меж, якими необхідно керуватись при встановленні ціни на потенційний товар (таблиця 6.20).

Таблиця 6.20. Визначення меж встановлення ціни

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	27...270 грн	135...270 грн	27000...98000 грн	27...135 грн

Наступним кроком є визначення оптимальної системи збуту, в межах якого приймається рішення (таблиця 6.21):

- проводити збут власними силами або залучати сторонніх посередників (власна або залучена система збуту);
- вибір та обґрунтування оптимальної глибини каналу збуту;
- вибір та обґрунтування виду посередників.

Таблиця 6.21. Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Клієнт повинен надаватися в режимах “тріал” та “повний”.	Легкість в встановленні, легкість в сплаті послуг	4: Розробник даного продукту - Веб-сайт - Користувач.	Проводити збут силами посередника.

Останньою складовою маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів (таблиця 6.22).

Таблиця 6.22. Концепція маркетингових комунікацій

No п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Купляють програми через авторизовану мережу	Веб-сайти	Підтримка декількох платформ Пришвидшення розробки ПЗ	Довести, що програмний продукт пришвидшить розробку ПЗ вразі	Розпізнавай, класифікуй, повідомлюй про пози людини

## Висновки до розділу 6

Розроблений програмний продукт має переваги над існуючими конкурентами та є конкурентноздатним на ринку. Програма має шляхи подальшого розвитку, визначені маркетингові стратегії та шляхи збуту. Основна цільова аудиторія — це медичні заклади, для яких важлива швидкість розробки та точність розпізнавання.

## ВИСНОВКИ

У результаті підготовки магістерської дисертації та під час її виконання було розв'язано багато різноманітних задач пов'язаних із застосуванням методів та алгоритмів машинного навчання для моделювання процесів визначення положення об'єкта на динамічному наборі зображень.

У ході аналізу існуючого програмного забезпечення розпізнавання поз людини на відеопотоці було досліджено системи, які слугують для розв'язання поставлених задач. Аналіз показав, що існуючі системи вирішують задачу не у повному обсязі, є громіздкими та мають високі апаратні вимоги до пристроїв користувачів.

У результаті роботи було створено програмний продукт, який надає такі функції:

- забезпечено користувача персональним кабінетом, в якому можна додавати власні камери спостереження;
- визначає пози людини на відеопотоці і виведення скелету на екран користувача;
- класифікує пози людини і визначення негативних для людини поз тіла;
- сповіщує користувача засобами електронної пошти про пози які було розпізнано на відеопотоці;
- надає можливість розпізнавати положення об'єкта з відеопотоку вебкамери;
- забезпечує користувача зручним графічним інтерфейсом, який може працювати як на комп'ютері, так і на мобільному телефоні.

Розроблений програмний продукт дозволяє автоматизувати розпізнавання поз людини на відеопотоці та допомагає сповіщувати про дані пози засобами електронної пошти.

Проведено огляд методів і засобів розробки програмної системи. Обґрунтовано вибір створення програмної системи, заснованої на веб-технологіях, а також

побудованої за триланковою архітектурою. Це дає змогу підвищити гнучкість та зручність системи, як у розробці та супроводі, так і у використанні.

За результатами виконання тестових завдань підтверджена коректність отриманих результатів, отже система відповідає поставленим вимогам.

Користувачами системи можуть бути різноманітні оператори, які здійснюють нагляд за іншими людьми через камери відеоспостереження з метою виявлення надзвичайних ситуацій, які виникають, а також з метою допомоги та швидкої реакції на дані ситуації. Програмне забезпечення може бути використано на будь-якій операційній системі, на якій встановлено браузер, який підтримує останні веб-стандарти, а також має постійний доступ до інтернету.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Anthony Gore. Full-Stack Vue.js 2 and Laravel 5 [Електронний ресурс]. — 2015. — Режим доступу: <https://bit.ly/2OEODzR>.
2. Aurélien Geron. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems [Електронний ресурс]. — 2015. — Режим доступу: <https://amzn.to/2R2rjZx>.
3. Balalaie, A. (2016-05-01). «Microservices Architecture Enables DevOps: Migration to a Cloud-Native Architecture». IEEE Software 33 (3): 42–52.
4. Kamill Gusmanov, Kevin Khanda, Dilshat Salikhov, Manuel Mazzara, Nikolaos Mavridis. "Jolie Good Buildings: Internet of things for smart building infrastructure supporting concurrent apps utilizing distributed microservices". Retrieved 9 Feb 2018.
5. Markus Egger. MVVM Survival Guide for Enterprise Architectures in Silverlight and WPF [Електронний ресурс]. — 2012. — Режим доступу: <https://www.packtpub.com/application-development/mvvm-survival-guide-enterprise-architectures-silverlight-and-wpf>.
6. Martin Fowler. GUI Architectures. Часть 1 [Електронний ресурс]. — 2009. — Режим доступу: <https://bit.ly/2CvCk1e>.
7. Martin Fowler. GUI Architectures. Часть 2 [Електронний ресурс]. — 2009 — Режим доступу: <https://habr.com/post/53536/>.
8. Matt Weisfeld. The Object-Oriented Thought Process. — Fourth Edition. — Addison-Wesley Professional, 2013. — 336 с.
9. Rodgers, Peter. "Service-Oriented Development on NetKernel- Patterns, Processes & Products to Reduce System Complexity". CloudComputingExpo. SYS-CON Media. Retrieved 19 August 2015.
10. Yuli Vasiliev. Beginning Database-Driven Application Development in .NET / Yuli Vasiliev. — Москва: ИЛ, 2008. — 400 с.



11. Адам Фримен. ASP.NET Core MVC с примерами на С# для профессионалов [Электронный ресурс]. — 2017. — Режим доступа: <https://bit.ly/2JbSgHe>.
12. Антони Синтес. Освой самостоятельно объектно-ориентированное программирование за 21 день = Sams Teach Yourself Object-Oriented Programming in 21 Days. — М.: «Вильямс», 2002. — С. 672.
13. Байда Д.В. Використання нейронних мереж для задач семантичної сегментації зображення / Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», 2018. — 2с.
14. Байда Д.В. Нейронні мережі як засіб для семантичної сегментації зображення / Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», 2018. — 3с.
15. Бакурова Анна Володимирівна. Нейронні мережі. / Запоріжжя: ЗНУ, 2007.- 29 с.
16. Бариленко В. И. Бизнес-анализ как важный вид консалтинговых услуг // РИСК: Ресурсы, Информация, Снабжение, Конкуренция. — № 4. — 2012. — С.202-207.
17. Блох Д.. CLR. Эффективное программирование / Джошуа Блох — М. Лори, 2014. — 310 с.
18. Болье А. Learning SQL [Электронный ресурс]. — 2005. — Режим доступа: <http://shop.oreilly.com/product/9780596007270.do>.
19. Гаврилова ТА., Хорошевский В.Ф., 2000. Базы знаний интеллектуальных систем / Учебник для вузов. – СПб, Изд-во “Питер”, 2000
20. Гаевский Александр Юлиевич. Самоучитель по созданию Web-страниц: / К.: А.С.К., 2012.- 472с.
21. Генри С. Уоррен. Алгоритмические трюки для программистов / Генри С. Уоррен. — М.: Вильямс, 2014. — 512 с.
22. Голощапов Алексей. Microsoft .NET. Программирование серверных приложений / Алексей Голощапов. — М.: БХВ-Петербург, 2012. — 448 с.
23. Гонсалвес Энтони. Изучаем .NET / Энтони Гонсалвес. — М.: Питер, 2016. — 640 с.

24. Гради Буч. Объектно-ориентированный анализ и проектирование с примерами приложений на С#. 2-е издание, Издательство: Бином, Невский Диалект, 2005, ISBN 0-8053-5340-2, ISBN 5-7989-0067-3, ISBN 5-7940-0017
25. Гупта Арун. С# 6.0 Основы / Арун Гупта. - М.: Вильямс, 2014.— 336 с.
26. Джеймс Чамберс. ASP.NET Core. Разработка приложений [Электронный ресурс]. — 2018. — Режим доступа: <https://bit.ly/2AliYL2>.
27. Зайченко Ю.П. Комп'ютерні мережі: Навчальний посібник. / К.: Слово, 2013.- 286с.- 20.00
28. Иан Грэхем. Объектно-ориентированные методы. Принципы и практика = Object-Oriented Methods: Principles & Practice. — 3-е изд. — М.: «Вильямс», 2004. — С. 880.
29. Квашнин А. Как продвигать проекты коммерциализации технологий: серия методических материалов «Практические руководства для центров коммерциализации технологий» / М. Катешова, А. Квашнин, под рук. П. Линдхольма, проект EuropeAid «Наука и коммерциализация технологий», 2006. – 52 с.
30. Квашнин А. Как управлять портфелем технологий и интеллектуальной собственностью : серия методических материалов «Практические руководства для центров коммерциализации технологий» / под рук. П. Линдхольма, проект EuropeAid «Наука и коммерциализация технологий», 2006. – 60 с.
31. Левин Александр..NET на десктопе / Левин Александр. — М.: Питер, 2013. — 970 с.
32. П. Дейтел. С# для программистов. Создаем приложения / П. Дейтел — М.: Питер, 2012. — 560 с.
33. Пашін В.П. Методичні вказівки до виконання економіко-організаційного розділу дипломних проектів (робіт) бакалаврів і спеціалістів для студентів Інституту прикладного системного аналізу з дисципліни “Економіка та організація виробництва” для студентів спеціальностей 7.080204 - “Соціальна інформатика”, 7.080203 - “Системний аналіз і управління”,

- 7.080404 – “Інтелектуальні системи прийняття рішень”, 7.080402 - “Інформаційні 75 технології проектування” / Пашін В.П., Романов В.В., Єгорова Н.В. – К. : НТУУ “КПІ”, 2011. – 118 с.
34. Рик Роджерс. Common Language Runtime. Разработка приложений / Роджерс Рик. — М.: Эком, 2010. — 130 с.
35. Сиерра К., Бейтс Б.. Изучаем С# / Кэти Сиерра, Берт Бейтс — М: Эксмо, 2016. — 720 с.
36. Тиль, П. От нуля к единице : как создать стартап, который изменит будущее / П. Тиль, Б. Мастерс; перевод с англ. – Москва : Альпина паблишер, 2015. – 188 с.
37. Филип Котлер, Роланд Бергер, Нильс Бикхофф. Стратегический менеджмент по Котлеру. Лучшие приемы и методы = The Quintessence of Strategic Management: What You Really Need to Know to Survive in Business. — М.: Альпина Паблишер, 2012. — 144 с.
38. Харниш, В. Правила прибыльных стартапов : как расти и зарабатывать деньги / В. Харниш ; пер. с англ. В. Хозинского. – Москва : Манн, Иванов и Фербер, 2012. – 279 с.
39. Хорстманн К. .NET. Вводный курс / Кей Хорстманн. — М:Вильямс, 2014. — 208 с.
40. Цехнер Марио. Программирование под Windows / Марио Цехнер. — М.: Питер, 2012. — 688 с.
41. Эванс Б.. С#. Новое поколение разработки/ Б. Эванс. — М.:Питер, 2014. — 400 с.

## ДОДАТОК А

Публікації

Моделювання процесів визначення положення об'єктів на динамічному наборі  
зображень

УКР.НТУУ"КПІ"\_ТЕФ\_АПЕПС\_ ТР31234\_18М

Аркушів 8

2018

## ДОДАТОК Б

Акт впровадження

Моделювання процесів визначення положення об'єктів на динамічному наборі  
зображень

УКР.НТУУ"КПІ"\_ТЕФ\_АПЕПС\_ТР31234\_18М

Аркушів 2

2018