**Vision and Scope Document**

**for**

**App software system for emotion detection**

**Version 1.0**

**approved**

**Prepared by**

**Yushchenko Artur**

**NURE**

## 16.06.2025

## 1. Постановка мети.

Ціллю роботи є розробка прототипу веб-застосунку для тестування та порівняння продуктивності сучасних архітектур нейронних мереж — CNN, RNN та ViT — у задачі автоматичного розпізнавання емоцій за мімікою людини. Система дозволяє здійснювати завантаження зображень або відео, запуск моделі на аналіз, візуалізацію результатів та оцінювання ключових метрик продуктивності.

Розроблений веб-застосунок імітує реальні сценарії інтеграції моделей комп’ютерного зору у веб-сервіси та дозволяє об’єктивно порівнювати CNN, RNN і ViT у єдиному середовищі. Кінцевою метою є отримання експериментальних даних для формулювання практичних рекомендацій щодо вибору оптимальної архітектури в залежності від специфіки проєкту та обмежень середовища.

Для реалізації застосунків використано сучасні технології:

* клієнтська частина. Інтерфейс веб-додатку розроблено з використанням React (TypeScript). Реалізовано інструменти завантаження зображень/відео, виклику моделей, виведення результатів класифікації;
* серверна частина. Сервер реалізовано на базі Python з використанням Flask. Він забезпечує обробку REST-запитів, інтеграцію з навчальними моделями та логіку аналізу результатів;
* база даних. Для збереження історії аналізів, результатів класифікації та конфігурацій моделей застосовано PostgreSQL. Структура даних охоплює сутності: “Model”, “Upload”, “Result”, “TestCase”.

Для реалізації програмної системи використано трирівневу архітектуру, яка передбачає чітке розділення на:

* клієнтський рівень відображення результатів, взаємодія з API, запуск аналізу моделей;
* серверний рівень запуск inference-процесу, взаємодія з моделями CNN, RNN та ViT, збереження результатів;
* рівень бази даних, структуроване збереження даних про аналіз, користувацький ввід та вивід метрик.

Усі компоненти системи об’єднано в єдиний програмний продукт із чіткою структурою проекту. Архітектура реалізована з підтримкою масштабованості, що дозволяє у майбутньому додавати нові моделі чи розширювати функціональність. Реалізація трьох моделей в одному середовищі дозволила забезпечити рівні умови для порівняння їх продуктивності за ключовими метриками.

## 2. Загальний опис системи.

Програмна система являє собою клієнт-серверний веб-застосунок, призначений для експериментального порівняння продуктивності та ефективності різних архітектур нейронних мереж — CNN, RNN та Vision Transformer — у задачі автоматичного розпізнавання емоцій за мімікою людини. Застосунок дозволяє здійснювати завантаження зображень або відео, виведення результатів класифікації, а також зберігання й перегляд історії аналізів.

Система виконує роль наглядного прикладу інтеграції, що забезпечує рівні умови для оцінювання трьох популярних підходів до побудови моделей глибинного навчання. Архітектура кожної реалізації ізольована, що дозволяє незалежно оцінити її.

Розроблене програмне рішення вирішує задачу об’єктивного вибору архітектури нейронної мережі для конкретних умов: від обмежених середовищ до високоточних застосунків.

Клієнтська частина реалізована з використанням React і дозволяє користувачу завантажити вхідні дані, обрати модель для аналізу, переглянути результати класифікації та порівняльні метрики. Кожна реалізація забезпечує незалежний запуск відповідної моделі (CNN, RNN або ViT) з єдиною структурою інтерфейсу. Дизайн побудовано з урахуванням адаптивності та інтуїтивної зручності, із застосуванням бібліотеки Tailwind.

Серверна частина реалізована з використанням Python і фреймворку Flask. Вона відповідає за обробку REST-запитів, запуск моделей глибинного навчання, обробку результатів, валідацію даних і запис до бази. Крім того, сервер підтримує базові механізми безпеки, обмеження доступу до API та структуру логування запитів. Архітектура передбачає можливість масштабування, розгортання моделей на GPU-серверах, а також інтеграцію з зовнішніми сервісами.

Для зберігання результатів аналізу та історії тестів використовується реляційна база даних PostgreSQL. Усі запити до БД ізолюються за допомогою ORM-рівня, що спрощує подальше розширення функціоналу, включаючи симуляцію більш складних сценаріїв, роботу з потоковим відео та оцінку продуктивності в умовах обмежених ресурсів.

## 3. Основний функціонал системи

У системі передбачено лише один тип користувача — User, який має повний доступ до всього функціоналу без обмежень. Це обумовлено експериментальним призначенням застосунку, де не реалізовано поділ на ролі або систему акаунтів. Основна увага зосереджена на можливості користувача проводити повноцінну взаємодію з інтерфейсом аналізу: завантажувати зображення або відео, запускати моделі аналізу емоцій (CNN, RNN, ViT), переглядати результати й історії тестувань (див. табл. 3.1).

Таблиця 3.1 – Функціонал програмного застосунку “emotion detection”

|  |  |  |
| --- | --- | --- |
| № | Функція | Опис |
| 1 | Завантаження зображення або відео | Дозволяє користувачу обрати локальний файл (зображення або відео) для аналізу. |
| 2 | Запуск аналізу | Ініціює обробку завантаженого файлу вибраною моделлю та отримання результату. |
| 3 | Виведення результатів | Візуальне представлення результатів аналізу (класифікація емоцій, ймовірність, графіки). |

Продовження таблиці 3.1

|  |  |  |
| --- | --- | --- |
| 5 | Перегляд історії аналізів | Дає змогу переглядати попередні сесії аналізу, включно з моделлю, даними та результатом. |
| 6 | Детальна інформація про аналіз | Дозволяє переглянути повну інформацію про конкретний аналіз: модель, дані, метрики. |

## 4. Загальні обмеження

Розроблений тестовий застосунок для порівняння архітектур нейронних мереж (CNN, RNN, ViT) у задачах розпізнавання емоцій має такі обмеження:

* застосунок потребує постійного доступу до інтернету, оскільки обробка даних виконується на серверній стороні за допомогою моделей, розгорнутих у бекенді;
* система не передбачає багатокористувацький режим чи розмежування ролей — весь функціонал доступний одному типу користувача;
* завантаження зображень або відео можливе лише локально — відсутня інтеграція з камерою в реальному часі або хмарними сховищами;
* не реалізовано розпізнавання декількох облич на одному зображенні, аналіз виконується для одного обличчя на кадр;
* додаток не підтримує офлайн-режим та не зберігає дані локально у браузері;
* інтерфейс орієнтовано на десктопні браузери — мобільна адаптація не реалізована;
* не передбачено можливості підвантаження кастомних моделей користувачем — використовуються лише заздалегідь навчені CNN, RNN та ViT.

## 5. Припущення та залежності

Під час розробки та тестування програмної системи прийнято такі припущення й визначено ключові залежності:

* користувач має стабільне інтернет-з’єднання для взаємодії з сервером, що обробляє моделі;
* взаємодія з додатком передбачається у сучасному браузері (Chrome, Firefox, Edge), який підтримує стандарти ES6+ і Web API;
* користувач працює з десктопної версії браузера, оскільки мобільна підтримка не реалізована;
* кожна модель (CNN, RNN, ViT) розгортається в окремому середовищі з єдиним інтерфейсом для уніфікованого тестування;
* серверна частина функціонує на Flask (Python), з доступом до моделей глибинного навчання, а база даних PostgreSQL в тестовому середовищі;
* порівняльний аналіз продуктивності моделей виконується за однакових умов: на одних і тих самих зображеннях, з однаковим інтерфейсом взаємодії та однаковим API.