Heart Attack Prediction

Специфікація вимог до програмного забезпечення

1.0

16.06.2025

Сандін Олексій

1. **Історія версій**

|  |  |  |  |
| --- | --- | --- | --- |
| **Дата** | **Опис** | **Автор** | **Коментар** |
| 16.06.2025 | Версія 1.0 | Сандін Олексій | Створення документу |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**ЗМІСТ**

[1. Вступ 4](#_Toc201277542)

[1.1 Призначення 4](#_Toc201277543)

[1.2 Контекст та бізнес-цілі 4](#_Toc201277544)

[1.3 Визначення та скорочення 5](#_Toc201277545)

[1.4. Огляд документа 6](#_Toc201277546)

[2. Загальний опис 6](#_Toc201277547)

[2.1. Опис продукту 6](#_Toc201277548)

[2.2. Функції продукту 7](#_Toc201277549)

[2.3. Характеристики користувачів 8](#_Toc201277550)

[2.4. Загальні обмеження 8](#_Toc201277551)

[3. Вимоги до системи 9](#_Toc201277552)

[3.1. Функціональні вимоги 9](#_Toc201277553)

[3.2. Нефункціональні вимоги 10](#_Toc201277554)

[3.3. Вимоги до інтерфейсу 11](#_Toc201277555)

[3.4. Вимоги до даних 12](#_Toc201277556)

1. Вступ

1.1 Призначення

Цей документ визначає функціональні та нефункціональні вимоги до програмного продукту, що розробляється в рамках магістерської роботи на тему «Дослідження методів машинного навчання для прогнозування серцевого нападу». Система призначена для автоматичного аналізу медичних даних пацієнтів з метою оцінки ризику виникнення серцевого нападу за допомогою алгоритмів машинного навчання. Основною метою є створення інструменту, який дозволяє лікарям і пацієнтам здійснювати раннє виявлення небезпечних станів та своєчасно вживати профілактичні заходи.

Програмний продукт реалізує механізми завантаження, обробки та аналізу медичних записів, побудови прогнозних моделей, візуалізації результатів та надання рекомендацій. Цей документ слугує технічною основою для розробників, аналітиків, тестувальників, науковців та інших учасників проєкту, що залучені до процесу розробки, впровадження і верифікації створеної системи.

1.2 Контекст та бізнес-цілі

Система прогнозування серцевого нападу із застосуванням методів машинного навчання розробляється як інструмент підтримки прийняття рішень для лікарів та засіб ранньої діагностики для пацієнтів. У контексті глобального зростання захворюваності на серцево-судинні недуги та потреби в превентивних заходах, запропонована система дозволяє виявляти ризик серцевого нападу на ранніх етапах шляхом аналізу медичних показників за допомогою алгоритмів машинного навчання.

Бізнес-цілями системи є:

* підвищення точності прогнозування серцевих нападів за рахунок використання сучасних ML-моделей;
* скорочення часу на аналіз даних пацієнта завдяки автоматизації оцінки ризику;
* створення адаптивної та масштабованої платформи, що може бути інтегрована в інформаційні системи медичних закладів;
* зниження кількості критичних випадків шляхом надання інструменту раннього попередження;
* сприяння впровадженню цифрової медицини через доступну систему з відкритим кодом або модульною структурою для подальших досліджень і розвитку.

1.3 Визначення та скорочення

ML – Machine Learning (Машинне навчання) — підгалузь штучного інтелекту, яка займається розробкою алгоритмів, що навчаються на основі даних для прийняття рішень або прогнозування.

AI – Artificial Intelligence (Штучний інтелект) — загальний термін для опису комп’ютерних систем, які можуть виконувати завдання, що зазвичай потребують людського інтелекту.

LR – Logistic Regression (Логістична регресія) — алгоритм машинного навчання для бінарної класифікації.

RF – Random Forest (Випадковий ліс) — ансамблевий метод класифікації та регресії, що базується на колекції дерев рішень.

KNN – K-Nearest Neighbors (Метод k найближчих сусідів) — класифікаційний алгоритм, що визначає клас об'єкта на основі найближчих до нього прикладів із навчальної вибірки.

SVM – Support Vector Machine (Метод опорних векторів) — алгоритм для класифікації, що шукає оптимальну гіперплощину для розділення класів.

AUC-ROC – Area Under the Receiver Operating Characteristic Curve — метрика, що характеризує здатність моделі розрізняти класи.

SMOTE – Synthetic Minority Over-sampling Technique — метод збалансування даних шляхом генерації синтетичних прикладів для менш представленого класу.

TPR / FPR – True Positive Rate / False Positive Rate — показники оцінки класифікаційної моделі.

API – Application Programming Interface — інтерфейс прикладного програмування, який дозволяє взаємодію між програмними компонентами.

EDA – Exploratory Data Analysis (Розвідувальний аналіз даних) — попереднє дослідження даних, що дозволяє виявити закономірності, аномалії та структуру.

1.4. Огляд документа

Цей документ містить специфікацію вимог до програмного забезпечення, розробленого в рамках кваліфікаційної роботи на тему «Дослідження методів машинного навчання для прогнозування серцевого нападу». Він складається з трьох основних частин:

* розділ 1 — вступна частина: містить загальне призначення, бізнес-контекст, термінологію та скорочення, що використовуються у документі;
* розділ 2 — загальний опис системи: включає основні функції продукту, характеристики цільових користувачів та існуючі технічні або організаційні обмеження;
* розділ 3 — специфікація вимог: функціональні та нефункціональні вимоги, вимоги до взаємодії з інтерфейсом користувача, базою даних та зовнішніми сервісами.

Документ орієнтований на команду розробки, тестування, дослідників у сфері цифрової медицини та інших учасників процесу впровадження. Він забезпечує чітке формалізоване бачення майбутнього програмного продукту та створює основу для його реалізації, перевірки та подальшого вдосконалення.

2. Загальний опис

2.1. Опис продукту

Програмний продукт є десктопною та веб-орієнтованою системою прогнозування серцевого нападу на основі методів машинного навчання. Метою системи є виявлення потенційного ризику серцевого нападу на ранніх етапах шляхом аналізу медичних показників пацієнта за допомогою інтелектуальних алгоритмів.

Система розробляється мовою Python із використанням бібліотек scikit-learn, pandas, NumPy, Matplotlib, Seaborn, Streamlit для реалізації веб-інтерфейсу. Продукт має потенціал інтеграції з медичними інформаційними системами (МІС) та підтримує режим локального або хмарного розгортання.

2.2. Функції продукту

Основні функції програмного забезпечення для прогнозування серцевого нападу включають:

* завантаження вхідних медичних даних з локального файлу (CSV, Excel);
* попередня обробка даних: виявлення та обробка пропущених значень, масштабування числових ознак, кодування категоріальних змінних;
* формування вибірки для навчання з урахуванням балансу класів,;
* навчання моделей машинного навчання: Logistic Regression, Random Forest, K-Nearest Neighbors;
* Оцінка ефективності моделей на тестовій вибірці за метриками точності (Accuracy), повноти (Recall), F1-міри та AUC-ROC;
* гіперпараметрична оптимізація моделей;
* порівняння продуктивності моделей у вигляді таблиць і графіків;
* прогнозування ризику серцевого нападу для нового пацієнта на основі введених медичних показників;
* виведення ймовірності серцевого нападу;

2.3. Характеристики користувачів

Система орієнтована на два основні категорії користувачів.

Лікар / Медичний фахівець

Основний користувач, який взаємодіє з системою для оцінки ризику серцевого нападу у пацієнтів. Не вимагається технічна підготовка — інтерфейс надає просту форму введення медичних даних пацієнта (вік, тиск, рівень холестерину тощо) та відображає результат прогнозу у зрозумілому форматі (графік ризику, категорія ризику, пояснення).

Дослідник / Аналітик з даних / Розробник

Використовує систему для тестування моделей, аналізу якості прогнозів, експериментів з алгоритмами машинного навчання. Має доступ до розширеного функціоналу: перегляд матриць помилок, графіків AUC, важливості ознак, логів навчання моделей.

Усі користувачі взаємодіють із системою через інтуїтивно зрозумілий інтерфейс.

2.4. Загальні обмеження

**Якість і повнота вхідних даних**

Точність прогнозу безпосередньо залежить від коректності медичних показників, що вводяться в систему. Наявність пропущених або некоректних значень (наприклад, нереальні значення тиску або віку) може суттєво знизити якість прогнозування або викликати помилку обчислень.

**Офлайн-режим та обмежена аналітика**

У базовій версії всі обчислення виконуються локально без використання хмарних ресурсів. Відсутність серверної обробки зменшує можливості масштабного аналізу або оновлення моделі в реальному часі.

**Обмеженість навчального датасету**

Прогнозна модель була навчена на загальнодоступному наборі даних (наприклад, Heart Disease UCI), який може не враховувати специфіку певних регіонів, расових або вікових груп. Це може обмежувати застосовність у клінічній практиці без додаткового перенавчання.

**Немедичний статус системи**

Система не є сертифікованим медичним приладом та не призначена для постановки діагнозів. Прогноз слугує лише як допоміжний інструмент для лікаря й не може бути використаний для самостійного прийняття рішень пацієнтом.

**Потреба в оновленнях моделей**

Оскільки алгоритми машинного навчання схильні до деградації з часом через зміну характеристик даних (concept drift), періодичне оновлення моделей є необхідним для підтримання актуальності.

**Обмежена інтерпретованість складних моделей**

Моделі типу Random Forest або Neural Networks можуть мати обмежену прозорість для кінцевого користувача (наприклад, лікаря), що ускладнює пояснення, чому зроблено саме такий прогноз. В роботі частково компенсовано це використанням метрик важливості ознак.

3. Вимоги до системи

3.1. Функціональні вимоги

|  |  |
| --- | --- |
| **ID** | **Вимога** |
| FR-1 | Система повинна дозволяти користувачеві вводити медичні показники (вік, тиск, холестерин тощо). |
| FR-2 | Система повинна обробляти введені дані та передавати їх у модуль прогнозування. |
| FR-3 | Система повинна тренувати моделі на основі завантажених даних. |
| FR-4 | Система повинна використовувати попередньо навчену модель для класифікації ризику серцевого нападу. |
| FR-5 | |  | | --- | |  |  |  | | --- | | Система повинна виводити результат прогнозу у вигляді ймовірності настання серцевого нападу. | |
| FR-6 | Система повинна підтримувати завантаження CSV-файлів з масивами пацієнтських даних для пакетного прогнозування. |
| FR-7 | Система повинна працювати автономно, без обов’язкового підключення до Інтернету. |

3.2. Нефункціональні вимоги

|  |  |
| --- | --- |
| ID | Вимога |
| NFR-1 | **Продуктивність**: система повинна здійснювати прогноз за введеними даними не довше ніж за 10 секунду. |
| NFR-2 | **Надійність**: система повинна коректно обробляти некоректні або неповні вхідні дані без аварійного завершення. |
| NFR-3 | **Мобільність**: застосунок має підтримуватись на різних операційних системах(ОС Windows 10+, macOS 11+ та Linux Ubuntu 20.04+). |
| NFR-4 | **Зручність використання**: інтерфейс повинен бути інтуїтивно зрозумілим і не вимагати спеціальної підготовки медичного персоналу. |
| NFR-5 | **Масштабованість:** система має підтримувати пакетну обробку великої кількості записів (1000+ пацієнтів) без втрати стабільності. |
| NFR-6 | **Час запуску**: застосунок повинен бути готовий до роботи протягом 3 секунд після запуску. |

3.3. Вимоги до інтерфейсу

**GUI-1: Головний екран застосунку**

* Містить кнопку для завантаження датасету (у форматі CSV).
* Містить кнопку попереднього перегляду завантажених даних у табличному вигляді.
* Відображає коротку статистику датасету (кількість записів, кількість ознак, відсутні значення).

**GUI-2: Екран тренування моделей**

* Кнопка запуску тренування моделі.
* Відображає метрики оцінювання моделі: точність (accuracy), precision, recall, F1-score.
* Виводить матрицю плутанини та ROC-криву у вигляді графіків.

**GUI-3: Екран прогнозування**

* Містить форму для введення параметрів нового пацієнта.
* Виводить прогноз щодо ймовірності серцевого нападу.

**UX-вимоги**

* Інтерфейс має бути інтуїтивно зрозумілим, з логічною послідовністю дій.
* Усі елементи мають бути доступні з десктопа.
* Усі повідомлення про помилки або успішне завершення операцій мають бути чітко позначені кольором та текстом.
* Графіки мають бути інтерактивними (наприклад, за допомогою Plotly або Altair).
* Результати мають зберігатися у вигляді CSV або JSON при натисканні відповідної кнопки.

3.4. Вимоги до даних

**Типи даних**

Медичні параметри пацієнтів:

* + Вік, стать, артеріальний тиск, рівень холестерину, рівень цукру, частота серцебиття, електрокардіографічні показники, тощо.

Ознаки з результатів діагностики:

* + Дані ЕКГ, наявність стенокардії, результат навантажувального тесту.

Цільова змінна:

* + Індикатор серцевого нападу (0 — відсутній, 1 — імовірний).

Метадані про сесію:

* + Дата та час обробки, обрана модель, точність прогнозу.

Параметри тренування:

* + Назва моделі, гіперпараметри, вибраний метод пошуку (GridSearch, RandomizedSearch), крос-валідація.

**Формат збереження**

* **CSV** — для вхідних даних та результатів прогнозу (зручно для перегляду та експорту).
* **JSON** — для зберігання конфігурацій моделей, параметрів пошуку та логів тренування.

**Ідентифікація користувача**

* Прив’язка до користувача є необов’язковою; у MVP-версії авторизація не передбачена.
* Дані можуть бути знеособленими або симульованими (для дослідницьких цілей).

**Локальне збереження**

* Дані зберігаються локально на пристрої користувача (файлова система).
* У перспективі можлива інтеграція з хмарними рішеннями для синхронізації (опційно).

**Політика конфіденційності**

* Застосунок не передає персональні або медичні дані третім сторонам.
* Всі дані обробляються локально, відповідно до принципів GDPR.
* Застосунок підтримує опцію повного видалення даних користувача на вимогу.