Intelligent Health Prediction System

Final Documentation

Course: Studio Projektowe 1
Project Type: University Project

Authors (Group 1):

Patryk Chamera pchamera@student.agh.edu.pl
Karol Bystrek karbystrek@student.agh.edu.pl
Mateusz Bielówka mbielowka@student.agh.edu.pl
Berenike Banek berenike@student.agh.edu.pl
Maksim Dziatkou mdziatkou@student.agh.edu.pl

January 26, 2026

Contents

Executive Summary

The **Intelligent Health Prediction System** is an AI-powered, distributed application designed to assess personalized health risks. By leveraging custom-trained machine learning models and generative AI (Google Gemini), the system analyzes user-submitted medical questionnaires to provide immediate risk assessments and actionable health recommendations.

The primary objective of this project is to demonstrate the integration of modern full-stack technologies with advanced data science pipelines. The system is capable of predicting the onset probability of critical conditions—specifically Diabetes, Heart Attack, and Stroke—while also offering a holistic "Habits Assessment" to gauge general wellness.

The application adheres to a microservices architecture, ensuring scalability and modularity. It features a React-based frontend, a Spring Boot backend acting as the API gateway and logic core, and a dedicated FastAPI Python service for high-performance model inference. All components are containerized using Docker, facilitating consistent deployment across environments.

This report details the system architecture, the technical implementation of the subsystems, and provides a comprehensive analysis of the machine learning methodologies employed, including data preprocessing, model selection, and performance evaluation.

System Architecture

The system implements a containerized, distributed microservices architecture, adhering to the "Separation of Concerns" principle. This design ensures modularity, scalability, and ease of maintenance. The application is composed of three primary services communicating via HTTP/REST protocols within a secure Docker network.

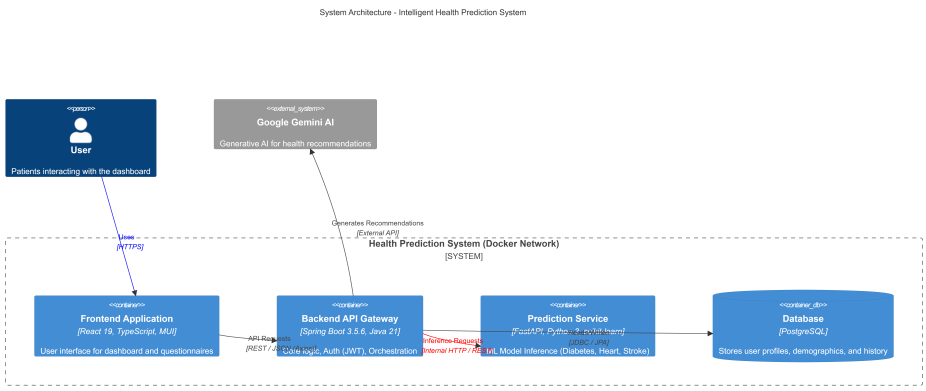


Figure 2.1: High-Level System Architecture Diagram

2.1 Frontend Application

The client-side application is built using **React 19** and **TypeScript**, leveraging the **Vite** build tool for optimized performance. It serves as the primary interface for users to interact with the prediction services.

Technical Stack & Features

- **UI Framework:** Material-UI (MUI v7) provides a consistent, accessible, and responsive design system.
- **State Management:** The application utilizes **TanStack Query** for efficient server-state management (caching, synchronization) and React Context for global application state (user sessions, themes).
- **Security:** Protected routes ensure that sensitive pages (History, Account, Questionnaires) are accessible only to authenticated users via JWT validation.

- **Visualization:** MUI X Charts are used to render dynamic visual representations of health trends and risk probability distributions.

2.2 Backend Service (API Gateway)

The core orchestration layer is a **Spring Boot 3.5.6** REST API running on **Java 21**. It acts as the central hub for the system, managing data flow between the client, the database, and the AI services.

Core Responsibilities

- **API Gateway:** Routes requests to the appropriate internal services (e.g., forwarding inference requests to the Python service).
- **Data Persistence:** Manages relational data using **Spring Data JPA** and **PostgreSQL**. Schema evolution is strictly controlled via **Flyway** migrations.
- **Generative AI Integration:** The backend holds the API keys and logic for communicating with the Google Gemini API, ensuring that sensitive credentials are never exposed to the client.

Authentication & Role-Based Access Control (RBAC)

Security is implemented using **Spring Security** and **JSON Web Tokens (JWT)**. The system supports multiple user roles to enforce access policies:

- **User Role:** Standard access. Users can manage their own profile, submit questionnaires, and view their own history.
- **Admin Role:** Elevated privileges. Administrators have the authority to manage user accounts and view system-wide statistics.

2.3 Python Prediction Service

A dedicated inference engine built with **FastAPI** and **Python 3**. This service is designed to be stateless and lightweight, focusing solely on loading pre-trained machine learning models and serving predictions.

Technical Specifications

- **Inference Engine:** Uses `scikit-learn` and `joblib` to load serialized models (`.pkl`) into memory at startup.
- **Validation:** `Pydantic` models ensure rigorous data validation for all incoming requests, preventing malformed data from reaching the models.
- **Performance:** The service runs on an ASGI server (Uvicorn), capable of handling asynchronous requests for high throughput.

Artificial Intelligence & Machine Learning

The intelligence of the system is derived from a hybrid approach: traditional supervised learning models for specific disease risk prediction, and Large Language Models (LLMs) for holistic health assessment and recommendation generation.

3.1 Supervised Learning Models

Three distinct models were developed to predict critical health conditions. Each model underwent a rigorous pipeline of data cleaning, feature engineering, and hyperparameter tuning [2].

3.1.1 1. Diabetes Prediction Model

Dataset: Trained on the "Diabetes Prediction Dataset" (Kaggle), comprising 100,000 patient records.

Data Pipeline & Preprocessing

- **Cleaning:** Initial analysis revealed no missing values, but 3,854 duplicate entries were identified and removed to ensure data integrity.
- **Encoding:** Categorical variables such as `gender` and `smoking_history` were transformed using Label Encoding.
- **Imbalance Handling:** The dataset exhibited a significant class imbalance (only 8.5% positive cases). SMOTE (Synthetic Minority Over-sampling Technique) was applied to the training set to prevent majority class bias.
- **Feature Selection:** Based on Random Forest feature importance analysis, the top 5 features were selected: `HbA1c_level`, `blood_glucose_level`, `bmi`, `age`, and `smoking_history`. Recent studies confirm the strong correlation between BMI and glycemic control [3].

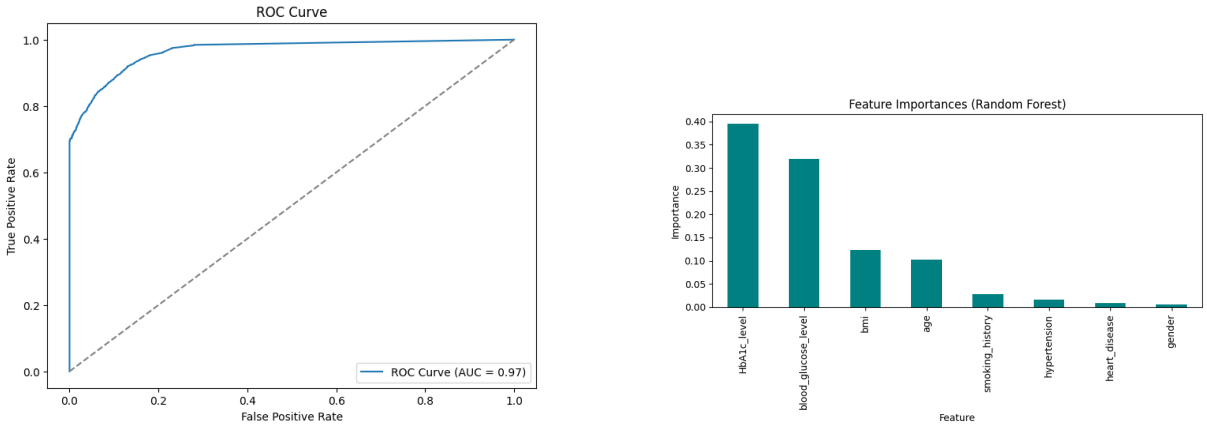
Model Architecture & Performance

A **Random Forest Classifier** was selected due to its robustness against overfitting. The model was optimized using class weighting and hyperparameter tuning.

Performance Metrics:

- **Accuracy:** 97.3%
- **ROC-AUC Score:** 0.97
- **Precision (Positive Class):** 1.00
- **Recall (Positive Class):** 0.69
- **F1-Score (Positive Class):** 0.82

The decision threshold was adjusted to `0.89` to prioritize the F1-score, balancing the trade-off between identifying all cases and minimizing false alarms.



(a) ROC Curve (AUC 0.97)

(b) Feature Importance

Figure 3.1: Performance Visualizations for Diabetes Model

3.1.2 2. Heart Attack Prediction Model

Dataset: A compilation of four databases (Cleveland, Hungary, Switzerland, Long Beach VA) sourced from the UCI Machine Learning Repository.

Data Pipeline & Preprocessing

- **Feature Engineering:** Features with excessive missing values (`slope`, `ca`, `thal`) were dropped. Remaining missing values were imputed using KNNImputer ($k = 5$).
- **Key Predictors:** The model identified Cholesterol (`chol`) and ST depression induced by exercise (`oldpeak`) as the strongest predictors.

Model Architecture & Performance

An **AdaBoost Classifier** was chosen after comparative testing against Logistic Regression and SVM. AdaBoost provided the highest cross-validation accuracy (0.8139 ± 0.0205).

Performance Metrics:

- **Accuracy:** 81.52%
- **ROC-AUC Score:** 0.9027
- **F1-Score (Disease Class):** 0.83

- **Generalization Gap:** 0.0014 (Difference between CV and Test accuracy)

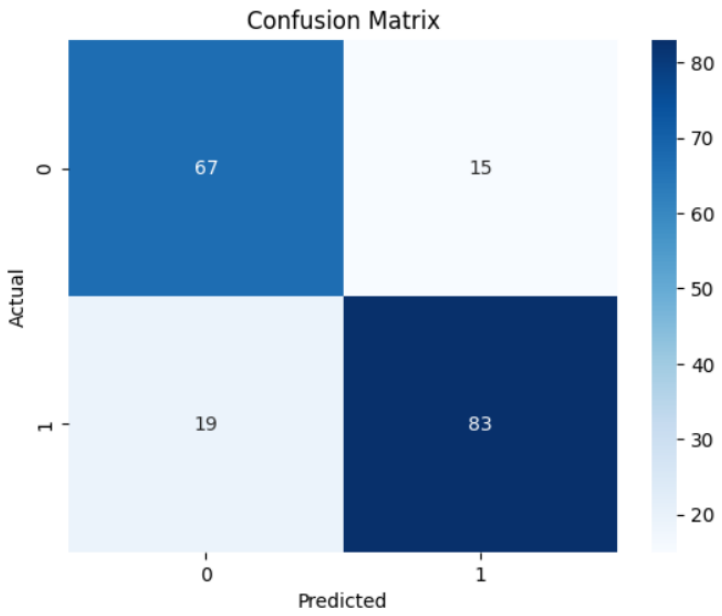


Figure 3.2: Confusion Matrix for Heart Attack Prediction

3.1.3 3. Stroke Prediction Model

Dataset: The "Stroke Prediction Dataset" by fedesoriano.

Data Pipeline & Preprocessing

- **Imputation:** Missing BMI values were filled using KNNImputation ($k = 5$).
- **Feature Selection:** The feature `smoking_status` was dropped due to excessive missing values (30.22%).
- **Sampling:** Due to the rarity of stroke events, the dataset is highly imbalanced. SMOTE was utilized to oversample the minority class during training.

Model Architecture & Performance

A **Support Vector Machine (SVM)** was selected. Given the high stakes of missing a stroke prediction, the primary optimization metric was **Recall**. SVM outperformed Random Forest and Gradient Boosting in identifying positive cases.

Performance Metrics:

- **Accuracy:** 83.56%
- **Recall (Macro Avg):** 0.80
- **ROC-AUC:** 0.7998
- **Recall (Disease Class):** 0.76

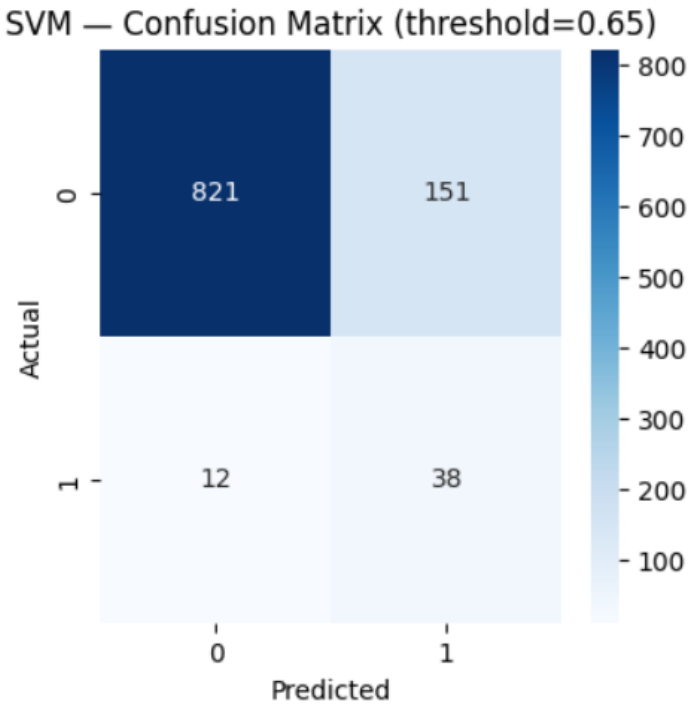


Figure 3.3: Confusion Matrix for Stroke Prediction

While the precision for the positive class is lower (0.20), the high recall ensures that the system functions effectively as a screening tool, flagging potential risks for further medical review.

3.2 Generative AI Integration (Google Gemini)

To provide users with understandable context for their results, the system integrates the Google Gemini API.

3.2.1 Recommendation Engine

For every prediction (Diabetes, Heart Attack, Stroke), the backend constructs a structured prompt containing the user's input data and the model's calculated probability. This prompt is sent to Gemini, which returns a personalized, empathetic explanation of the risk factors and actionable advice (e.g., "Your HbA1c level of 6.6 suggests pre-diabetes; consider increasing fiber intake...").

3.2.2 Habits Assessment Module

Unlike the disease models, the **Habits Assessment** does not use a pre-trained classifier. Instead, it relies entirely on the reasoning capabilities of the LLM.

- **Workflow:** The user submits subjective data regarding sleep patterns, diet quality, physical activity, and stress management.
- **Processing:** The backend forwards this unstructured data to Gemini with instructions to evaluate the lifestyle comprehensively.

- **Output:** The AI generates a quantitative "Wellness Score" (0-100) and a qualitative "Wellness Plan," providing a holistic view of the user's health that complements the specific medical predictions.

Application Walkthrough

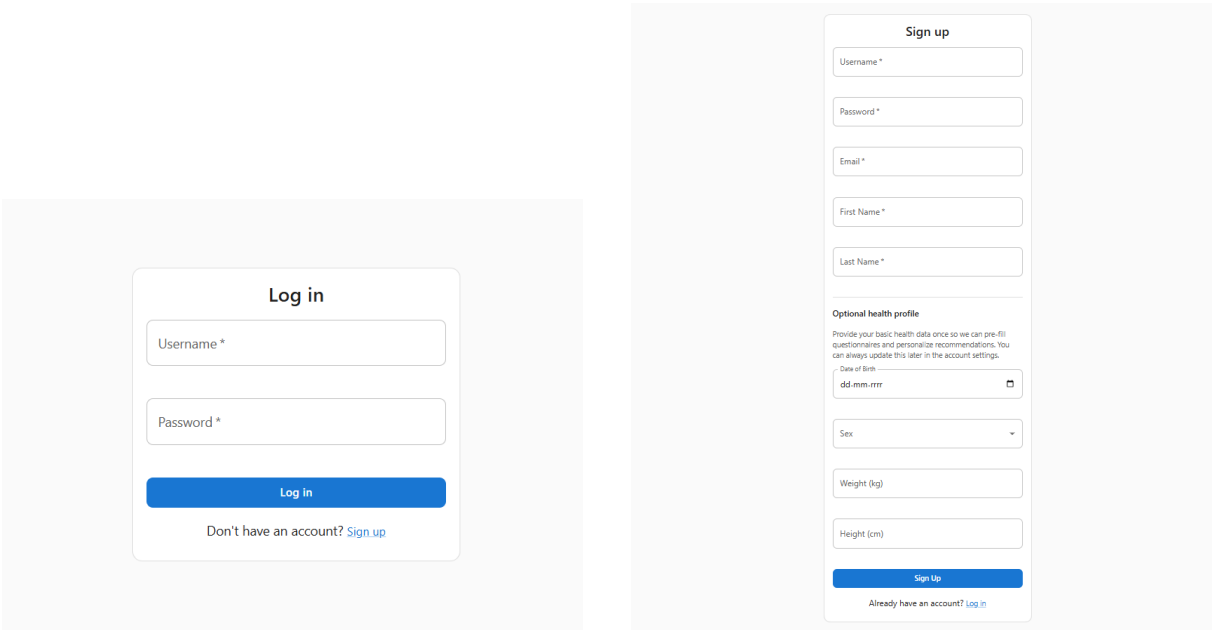
This section provides a detailed visual guide to the application's core functionality, demonstrating the user flow from authentication to complex health analysis.

4.1 Authentication

Access to the system is secured via JWT-based authentication. The interface provides separate workflows for returning users and new registrations.

4.1.1 Login Signup

The **Login Page** validates user credentials against the backend database and issues a secure token for session management. The **Signup Page** collects essential identity information including username, email, and password to create a new account.



(a) Login Interface (b) Registration Interface

Figure 4.1: Authentication Modules

4.2 Home Dashboard

The **Home Dashboard** acts as the central hub for the user experience. It provides an immediate overview of the user's health status, including:

- A personalized welcome message.
- Quick access cards to all health questionnaires.
- Visualizations of recent health trends and statistics.
- Summaries of recent risk assessments.

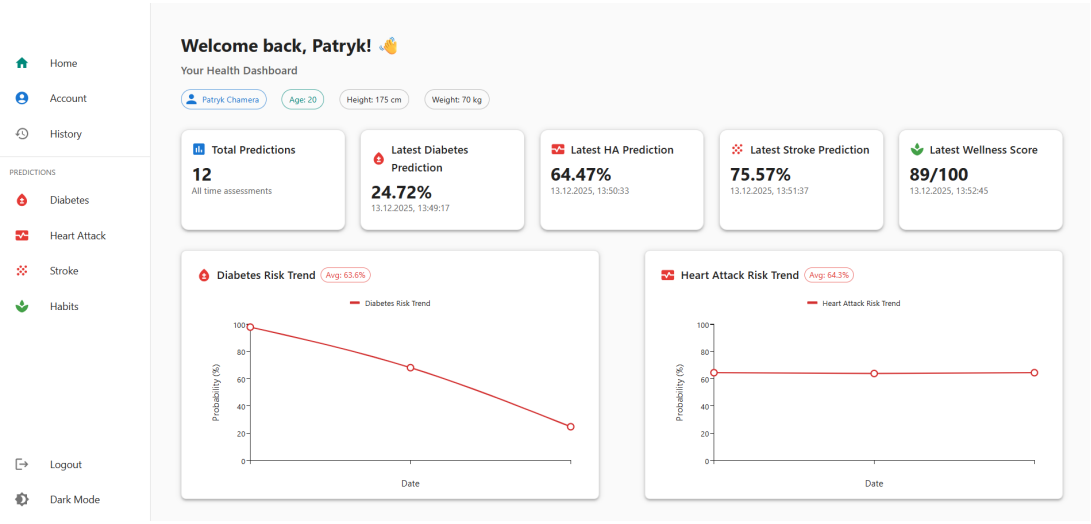


Figure 4.2: Dashboard: Top View with Welcome Message and Quick Actions

Figure 4.3: Dashboard: Health Trends and Statistical Overview

4.3 Health Questionnaires Predictions

The core functionality of the system is divided into disease-specific prediction modules. Each module consists of a data collection form and a detailed results view.

4.3.1 Diabetes Assessment

The Diabetes Questionnaire collects critical metabolic indicators including HbA1c level (average blood sugar over 2-3 months), current blood glucose, BMI, age, and smoking history.

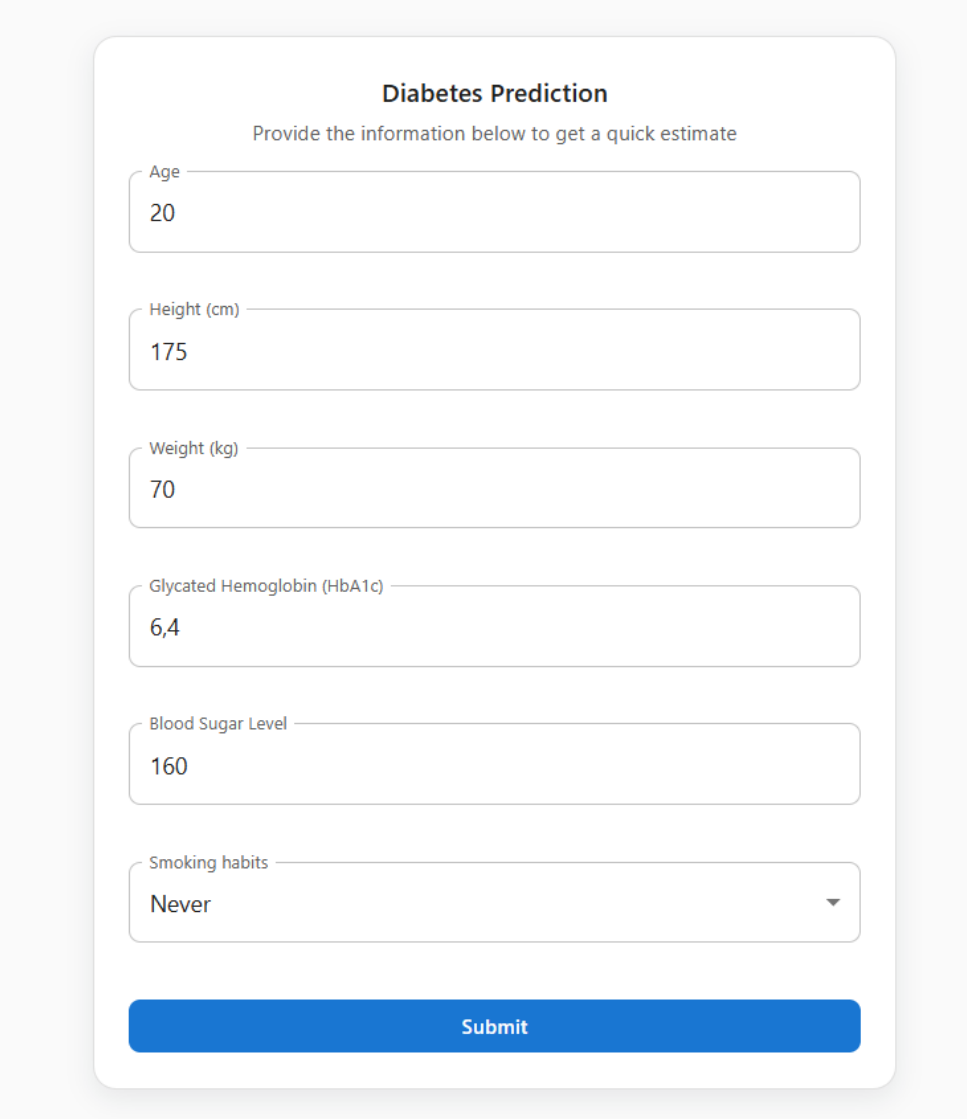


Figure 4.4: Diabetes Risk Assessment Form

Prediction Results

Upon submission, the system presents the **Prediction Probability** (a percentage indicating risk) and **AI-Generated Recommendations**. The Gemini LLM analyzes input values (e.g., specific HbA1c levels) to provide contextual medical insights and lifestyle advice.

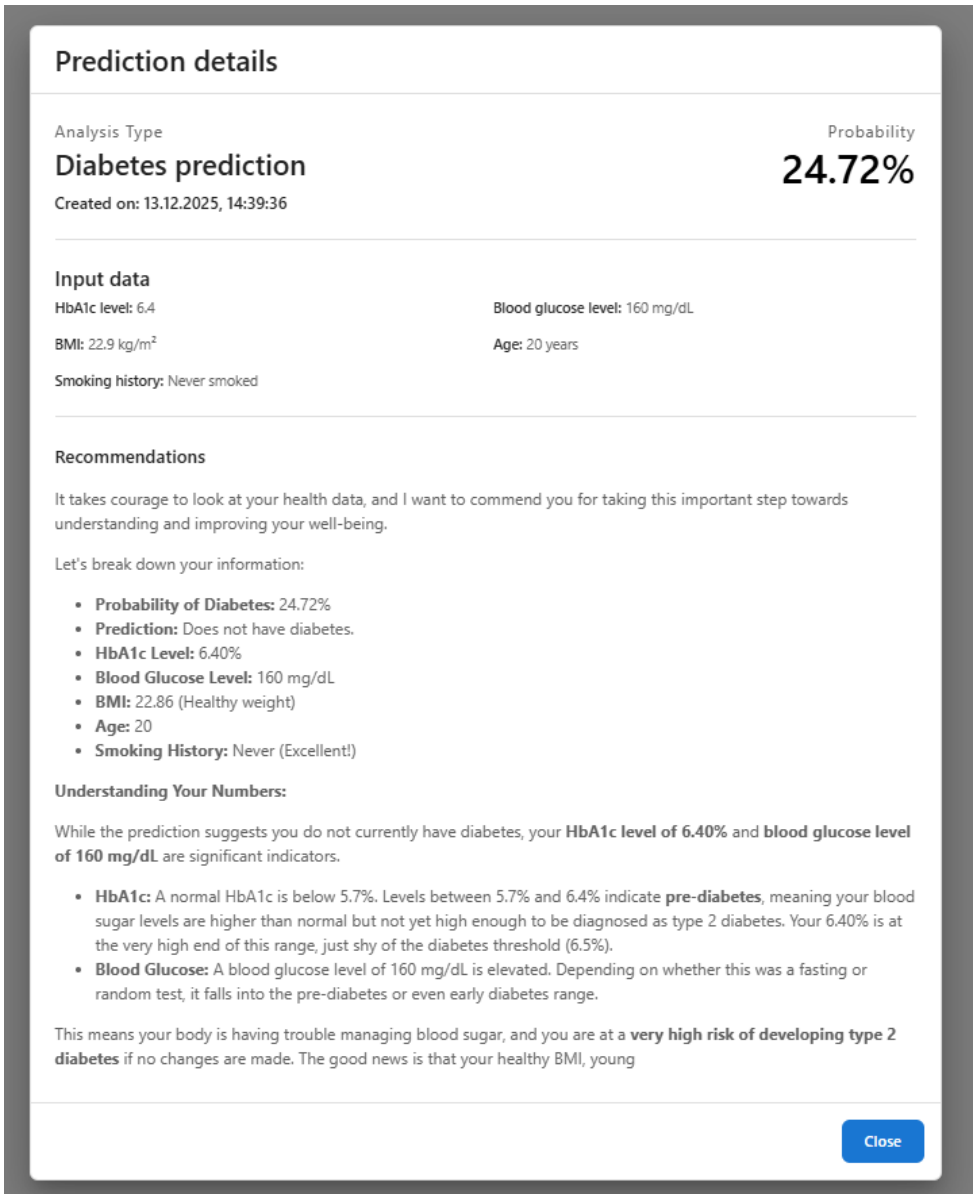


Figure 4.5: Diabetes Prediction Results with AI Insights

4.3.2 Heart Attack Assessment

This module gathers clinical cardiovascular parameters: chest pain type, resting blood pressure, serum cholesterol, maximum heart rate, ST depression (oldpeak), and exercise-induced angina.

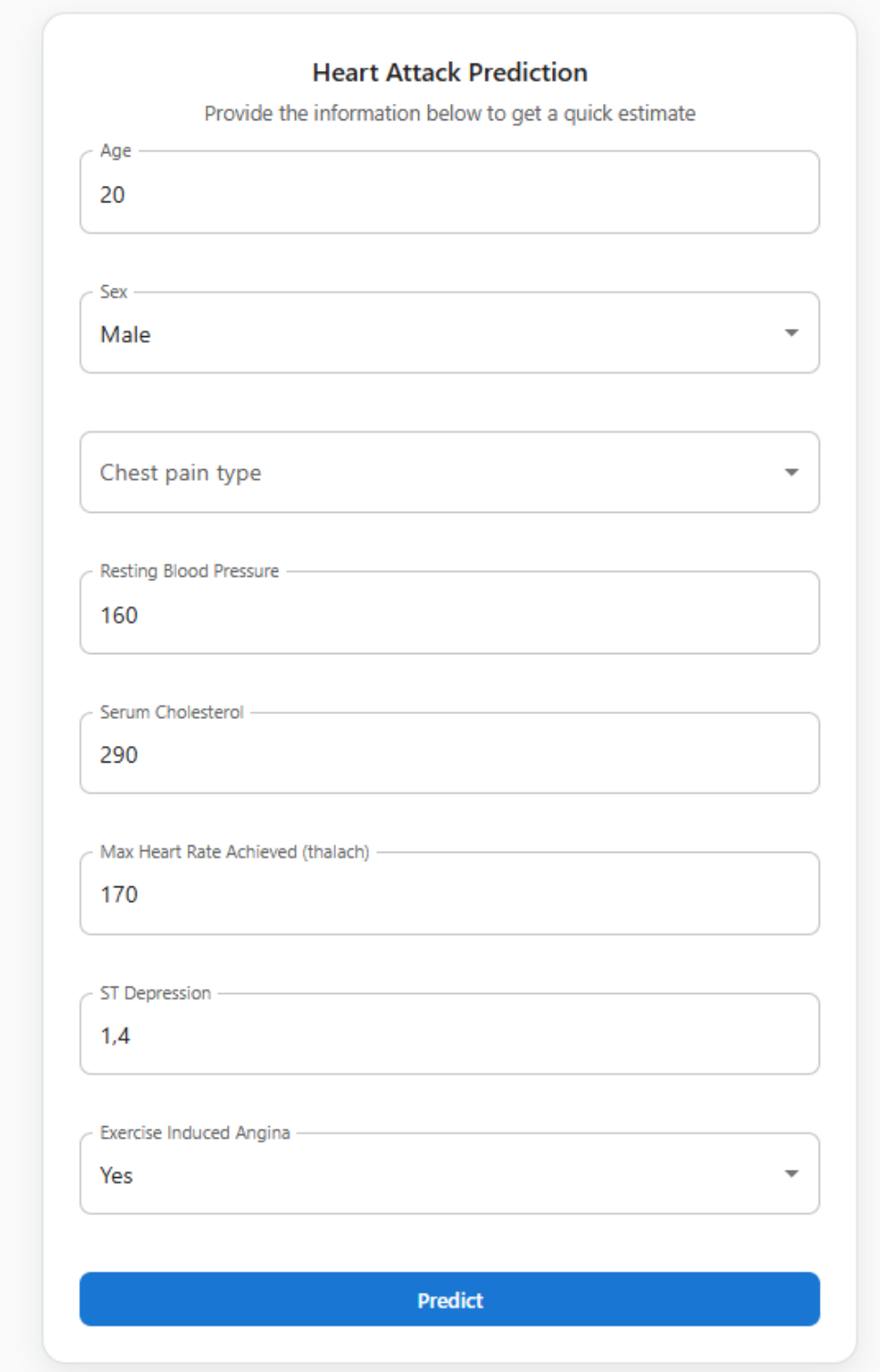


Figure 4.6: Heart Attack Risk Assessment Form

Prediction Results

The results page interprets the clinical data, offering probability scores and evidence-based cardiovascular advice regarding blood pressure management and cholesterol levels.

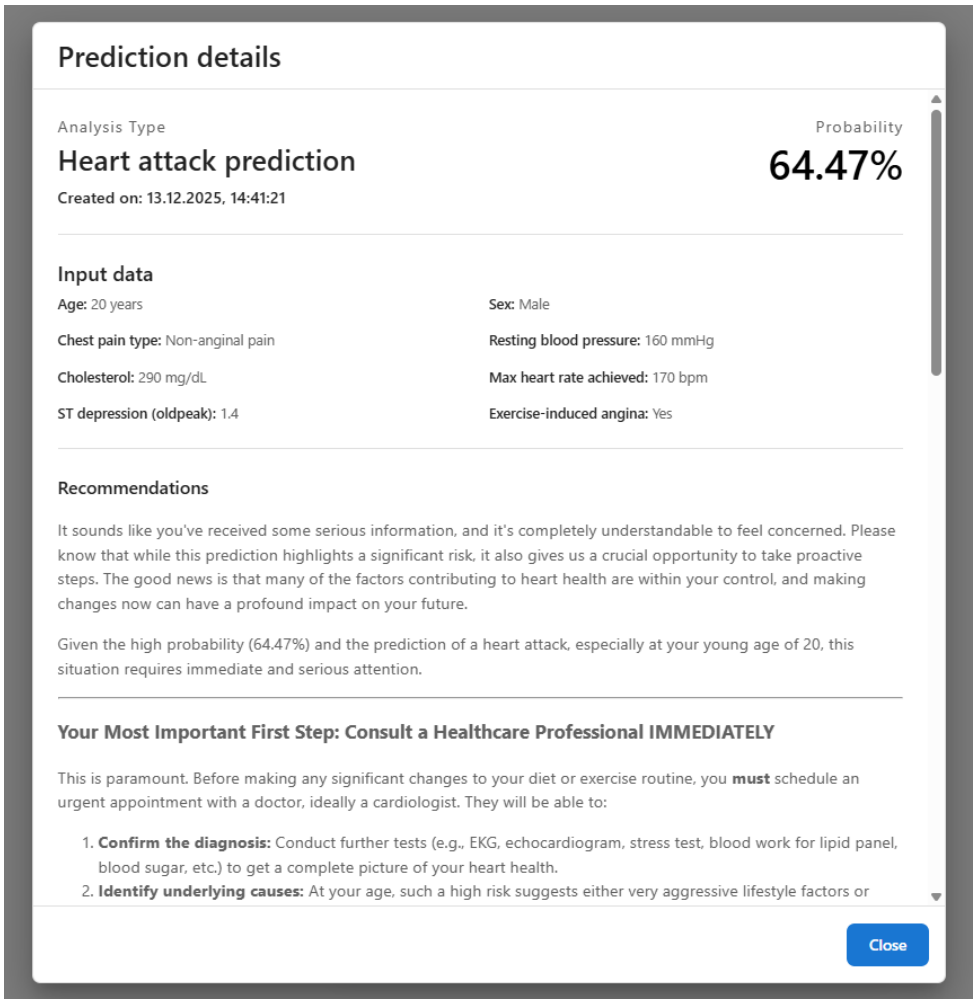


Figure 4.7: Heart Attack Risk Analysis and Recommendations

4.3.3 Stroke Assessment

The Stroke Questionnaire captures demographic and medical history factors associated with stroke risk, including hypertension, heart disease, work type, average glucose level, and BMI.

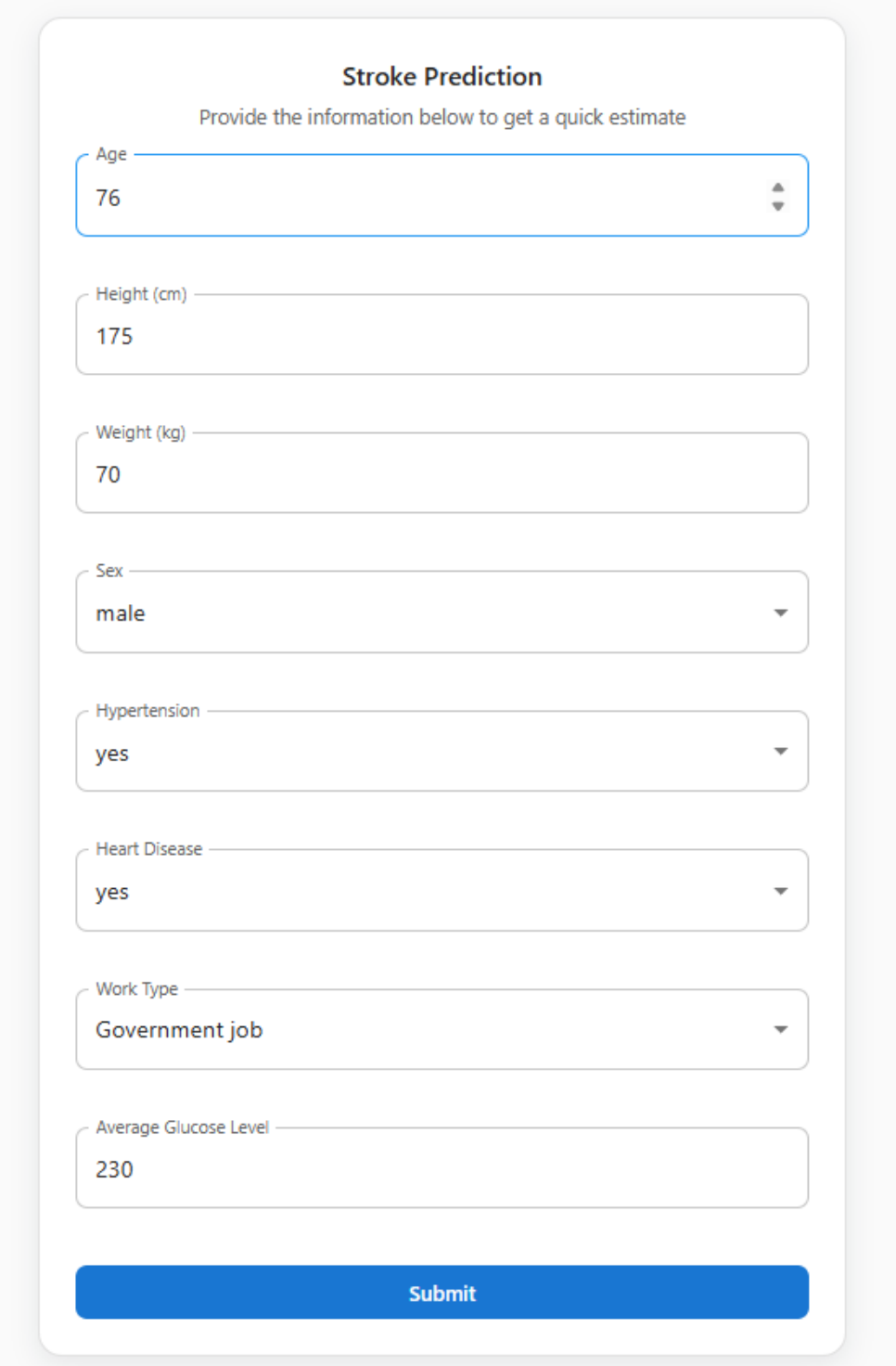


Figure 4.8: Stroke Risk Assessment Form

Prediction Results

The system evaluates the relationship between the user's medical history (e.g., hypertension) and stroke risk, providing actionable prevention strategies.

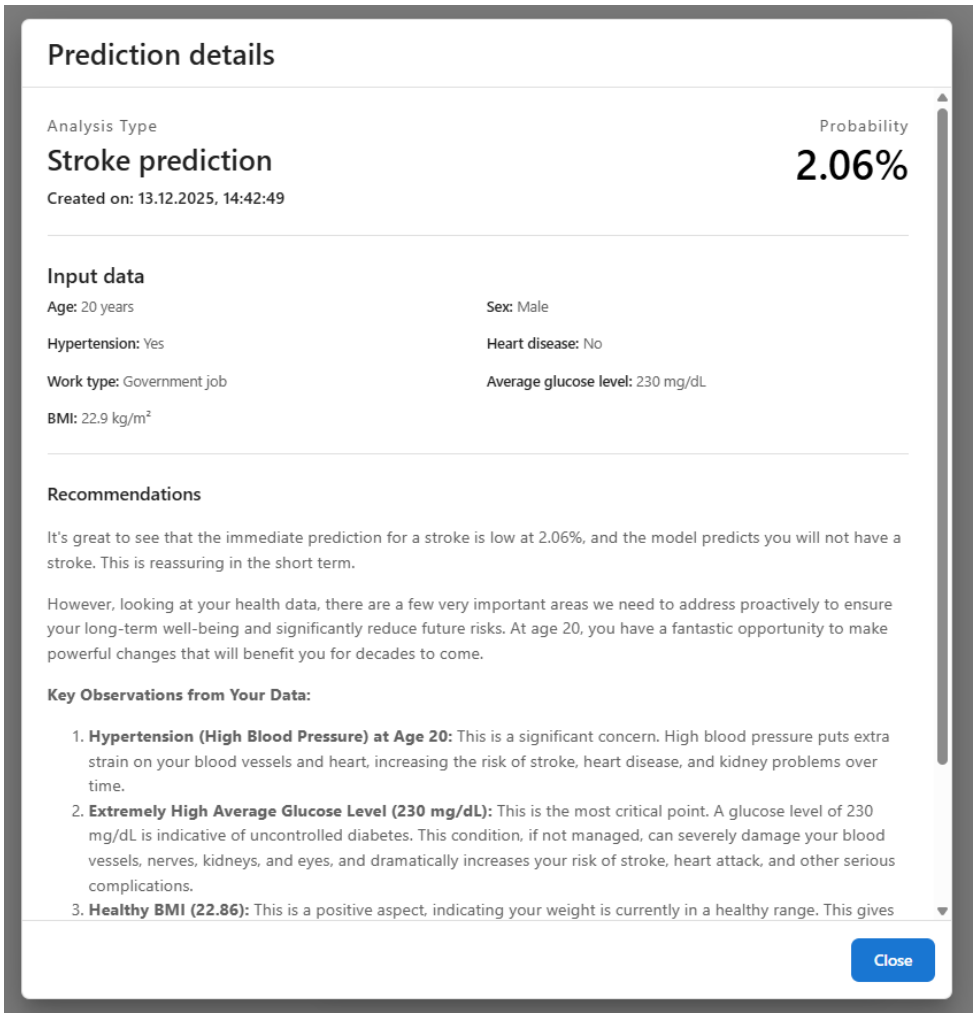


Figure 4.9: Stroke Prediction Outcome

4.4 Holistic Habits Assessment

Unlike the disease-specific models, the Habits Assessment focuses on general wellness. It evaluates unstructured lifestyle factors:

- **Physical Activity:** Frequency and intensity of exercise.
- **Dietary Patterns:** Nutritional habits.
- **Sleep Quality:** Duration and quality metrics.
- **Stress Management:** Coping mechanisms and stress levels.

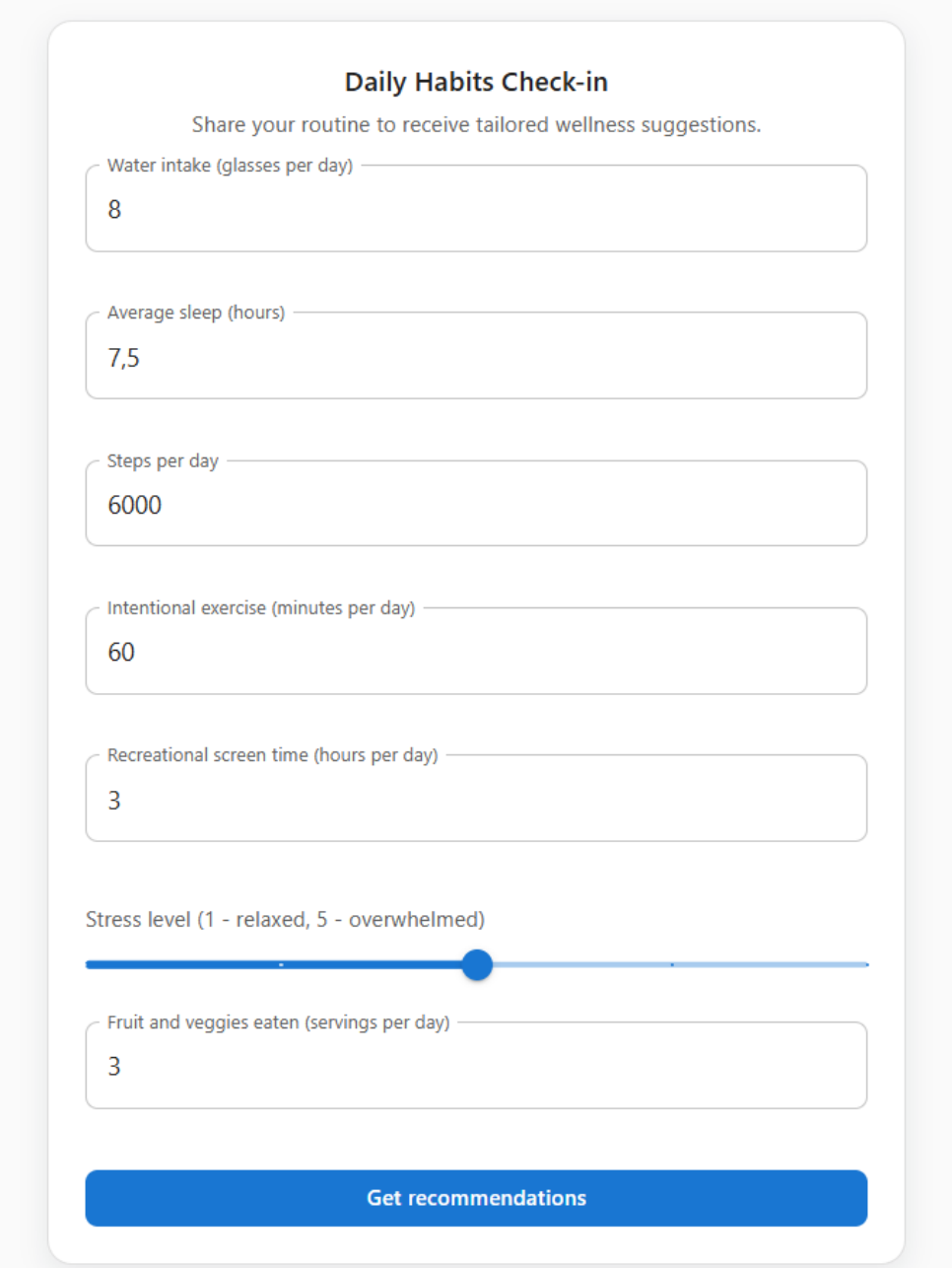


Figure 4.10: Lifestyle and Habits Questionnaire

Wellness Score & Plan

The result is a comprehensive **Wellness Score** accompanied by a holistic wellness plan. The AI generates sustainable recommendations for improving sleep, diet, and stress management to promote long-term well-being.

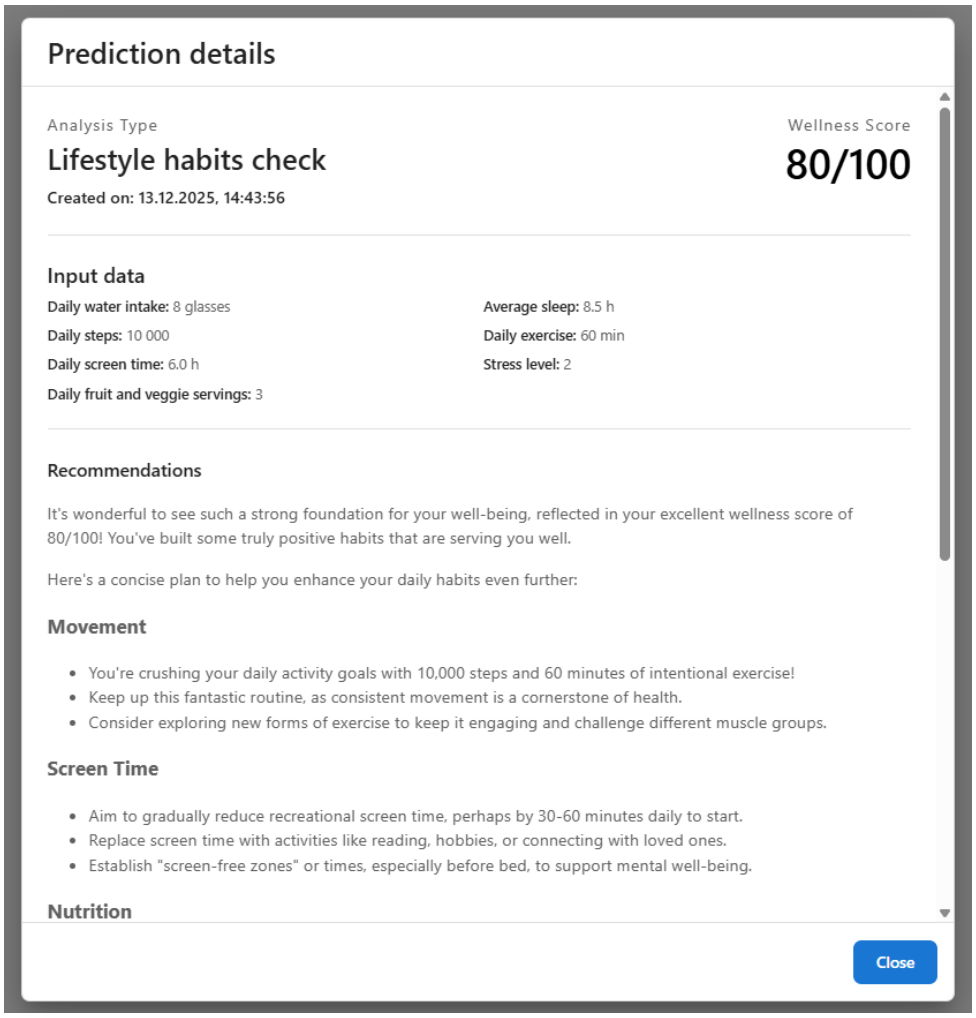


Figure 4.11: Wellness Score and Holistic Health Plan

4.5 User Profile & History

4.5.1 Prediction History

The History module serves as a longitudinal record of the user's health journey. It displays a chronological log of all assessments, including timestamps, risk scores, and saved AI recommendations, allowing users to track their progress over time.

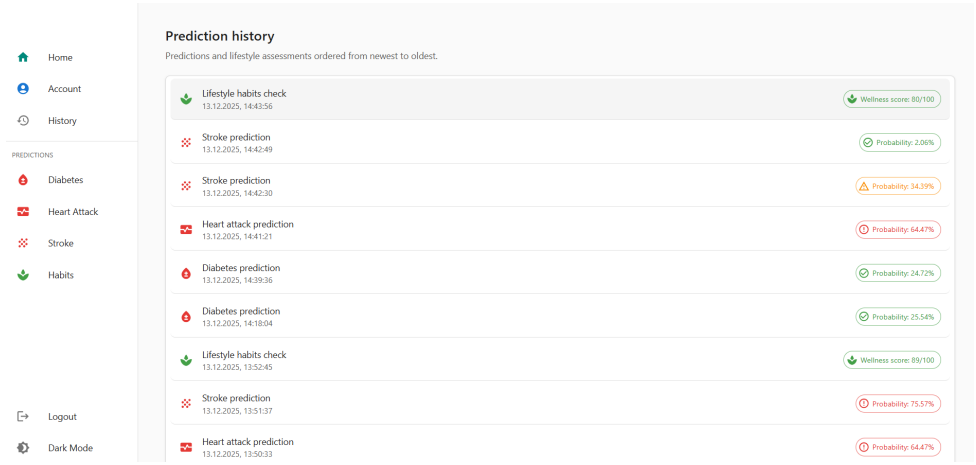


Figure 4.12: Comprehensive Prediction History Log

4.5.2 Account Settings

Users can manage their personal data via the Account Settings page. This includes updating profile information (name, email), modifying demographic data (height, weight) used in calculations, or permanently deleting their account.

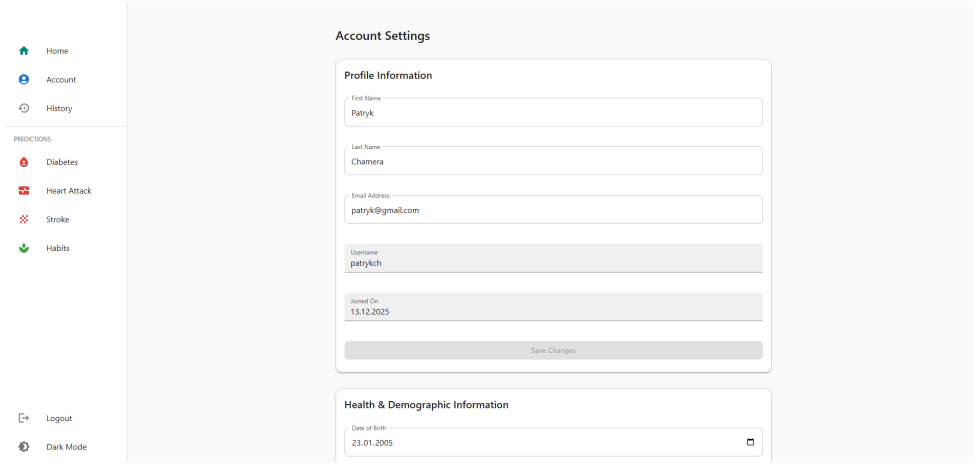


Figure 4.13: User Account and Demographic Settings

4.6 Admin Panel

The Admin Panel provides system administrators with comprehensive oversight and management capabilities for the health prediction system.

4.6.1 Admin Dashboard Overview

The main admin dashboard offers real-time system statistics and analytics, ensuring administrators can effectively monitor user activity and engagement. Key features include:

- **User Statistics:** Displays the total number of registered users in the system.

- **Prediction Analytics:** Provides a comprehensive breakdown of all health predictions, including total counts, daily volume, and categorized counts by prediction type (Diabetes, Heart Attack, Stroke, Habits Assessment).
- **Questionnaire Usage Visualization:** An interactive pie chart displaying the distribution of completed questionnaires by type.
- **Registration Trends:** A 30-day historical chart showing daily user registration activity.
- **Data Export:** The ability to export dashboard statistics as JSON for reporting and analysis.

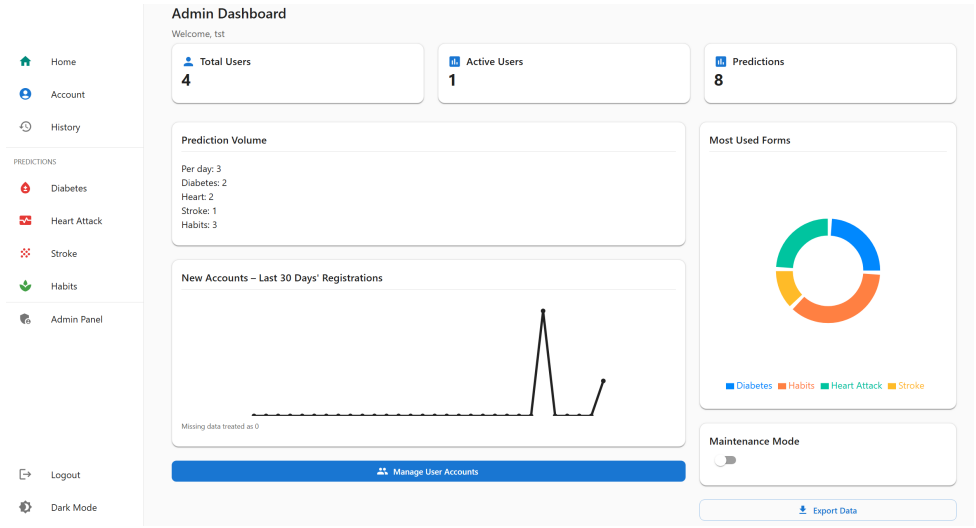


Figure 4.14: Admin Dashboard: Real-time system statistics and analytics overview.

4.6.2 User Account Management

Administrators have elevated permissions to manage all user accounts within the system. Capabilities include:

- **View All Users:** Access a paginated list of all registered users in the system.
- **Edit User Information:** Modify user details including first name, last name, and user roles (USER or ADMIN).
- **Delete User Accounts:** Remove user accounts from the system.
- **Administrative Access:** View prediction history for any user, access demographic data, and modify accounts regardless of ownership.

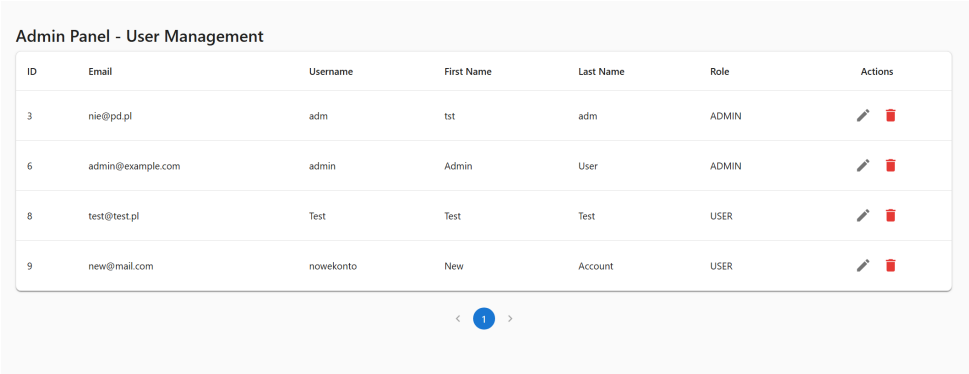


Figure 4.15: User Management: Interface for editing, deleting, and managing system users.

Setup and Deployment

5.1 Prerequisites

- **Docker Desktop** (installed and running)
- **Git** (for version control)
- **Google AI Studio API Key** (for Gemini integration)

5.2 Installation Guide

The system is designed for "One-Click Deployment" using Docker Compose.

5.2.1 1. Clone Repository

```
git clone <repository-url>
cd Intelligent-Health-Prediction-System
```

5.2.2 2. Environment Configuration

Configure the backend environment variables.

```
cp backend/.env.example backend/.env
```

Edit ‘backend/.env‘ to include your `JWT_SECRET` and `GEMINI_API_KEY`.

5.2.3 3. Start Services

```
docker-compose up --build
```

5.3 Service Endpoints

- **Frontend:** http://localhost:5173
- **Backend API:** http://localhost:8080
- **Swagger UI:** http://localhost:8080/swagger-ui/index.html
- **Python Service:** http://localhost:5000

Bibliography

[1] World Health Organization, "Diabetes: Key facts," 2023, accessed: 2025-10-22. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/diabetes

[2] A. Géron, K. Sawka, and Helion, *Uczenie maszynowe z użyciem Scikit-Learn, Keras i TensorFlow*. Helion, 2023.

[3] L. Deng, L. Jia, X.-L. Wu, and M. Cheng, "Association between body mass index and glycemic control in type 2 diabetes mellitus: A cross-sectional study," *Diabetes, Metabolic Syndrome and Obesity*, vol. 18, pp. 555–563, 2025. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC11853989/

[4] M. Mustafa, "Diabetes Prediction Dataset," Kaggle, 2023. [Online]. Available: https://www.kaggle.com/datasets/iammustafatz/diabetes-prediction-dataset/data

[5] A. Janosi, W. Steinbrunn, M. Pfisterer, and R. Detrano, "Heart Disease Dataset," UCI Machine Learning Repository, 1988. [Online]. Available: https://archive.ics.uci.edu/dataset/45/heart+disease

[6] Fedesoriano, "Stroke Prediction Dataset," Kaggle, 2021. [Online]. Available: https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset/data