Project Title

Health AI

1. Introduction

Team Leader: Praveen Kumar R
Team Members: Sanjay J, Narayanan, P K Subramani

2. Project Overview

Purpose:
The purpose of Health AI is to empower individuals and healthcare providers with intelligent digital assistance that enhances diagnosis, treatment, and wellness monitoring. By leveraging AI, real-time health data, and predictive analytics, the system supports doctors in decision-making, enables patients to track and manage health conditions, and provides personalized wellness guidance.

Features:
- Conversational Interface
- Symptom Checker & Policy Summarization
- Predictive Analytics (Health Forecasting)
- Personalized Health Tips
- Feedback Loop
- Anomaly Detection
- Multimodal Input Support
- User-Friendly Interface

3. Architecture

Frontend: Web-based UI (HTML, CSS, JavaScript)
Backend: Spring Boot (Java)
AI Integration: Python modules (IBM Watson Machine Learning & Generative AI)
Database: MySQL
Deployment: IBM Cloud CLI

4. Setup Instructions

Prerequisites:
- Java 17+ and Spring Boot
- Python 3.9+
- IBM Watson Machine Learning & Generative AI keys
- MySQL server
- Internet access for cloud APIs

Installation Process:
1. Clone the repository
2. Install backend dependencies (Maven/Gradle)
3. Install Python dependencies (requirements.txt)

4. Configure .env with IBM Watson credentials and MySQL DB details
5. Run backend Spring Boot server
6. Launch frontend (HTML/JS)
7. Interact with AI modules and database

5. Folder Structure

- backend/ – Spring Boot backend
- ai/ – Python AI integration
- frontend/ – HTML, CSS, JavaScript
- db/ – MySQL schema and scripts

6. Running the Application

1. Start Spring Boot backend
2. Run AI modules via Python scripts
3. Open frontend UI in browser
4. Upload health data, interact with AI chatbot, and view reports

7. API Documentation

- POST /chat/ask – User query → AI response
- POST /upload-report – Upload health document/data
- GET /get-tips – Personalized health recommendations
- GET /predict-health – Predictive analytics
- POST /feedback – Store user feedback

8. Authentication

- JWT-based authentication
- Role-based access (Doctor, Patient, Admin)
- IBM Cloud OAuth2 integration

9. User Interface

- Sidebar navigation
- Symptom checker and chatbot
- Health data visualization (charts, KPIs)
- Downloadable PDF health reports

10. Testing

- Unit Testing
- API Testing
- Manual Testing
- Edge Case Handling

11. Future Enhancements

- Mobile app integration
- Wearable health device connectivity
- Advanced disease detection with deep learning
- Multi-language support