**PROJECT REPORT**

**Title: HealthAI: Intelligent Healthcare Assistant Using IBM Granite**

### **1. INTRODUCTION**

#### **1.1 Project Overview**

HealthAI is an intelligent healthcare assistant powered by IBM Watson Machine Learning and Generative AI. It aims to enhance accessibility to medical insights by offering users accurate, personalized, and data-driven guidance.

#### **1.2 Purpose**

The purpose of HealthAI is to serve as a virtual healthcare companion, helping users understand symptoms, receive predictive diagnoses, and access evidence-based treatment suggestions through a user-friendly interface.

### **2. IDEATION PHASE**

#### **2.1 Problem Statement**

In today’s world, people often turn to the internet for health-related queries but struggle with reliability and accuracy. HealthAI addresses this by providing trusted, AI-powered medical insights.

#### **2.2 Empathy Map Canvas**

#### The empathy map helps understand the target users' mindset and expectations when interacting with a virtual health assistant like HealthAI.

|  |  |
| --- | --- |
| Category | Description |
| Says | "I want quick answers to my health questions.""Is this symptom serious?" |
| Thinks | "Can I trust this information?""Will this help me avoid a hospital visit?" |
| Does | Searches symptoms onlineAsks friends or family for medical advice |
| Feels | Anxious about symptomsUncertain about next steps |
| Pains | Conflicting online informationLack of access to immediate healthcare |
| Gains | Reliable AI suggestionsQuick advice at homeHealth tracking insights |

#### **2.3 Brainstorming**

The team explored various ideas such as mental health bots, COVID symptom trackers, and AI nutritionists before settling on an all-in-one intelligent assistant with chat, prediction, and analytics powered by IBM Watson.

### **3. REQUIREMENT ANALYSIS**

#### **3.1 Customer Journey Map**

#### The customer journey in the HealthAI application follows a streamlined, user-friendly process:

#### **Start Application** – The user opens the HealthAI web app (built using Streamlit).

#### **Input Health Query** – The user either types a symptom-related question or fills out a symptom form.

#### **Receive Diagnosis** – The AI model (IBM Granite) responds with likely conditions and guidance.

#### **Review Treatment Plan** – The app displays a structured, AI-generated treatment recommendation.

#### **View Health Analytics** – Users can visualize vital stats like heart rate, BP, glucose, and receive trend-based insights.

#### **End/Next Action** – The user can reset the session, consult a doctor, or continue exploring the app.

#### **3.2 Session Requirements**

* Real-time symptom input via chat
* Prediction based on user profile
* Personalized treatment plans

Visualization of health metrics

#### **3.3 Data Flow Diagram**



#### **3.4 Technology Stack**

* **Frontend:** Streamlit
* **Backend:** Python
* **AI Service:** IBM Watson ML (Granite 13B)
* **Visualization:** Plotly
* **Environment Management:** virtualenv

### **4. PROJECT DESIGN**

#### **4.1 Problem-Solution Fit**

People need quick, understandable, and trustworthy health information. HealthAI fulfills this by using medical LLMs for better accuracy.

#### **4.2 Proposed Solution**

A layered app with UI (Streamlit), core logic (Python functions), and AI service (IBM Granite). It guides users from symptom input to personalized plans.

#### **4.3 Solution Architecture**

* **UI Layer:** Chat, forms, and charts (Streamlit)
* **Application Logic:** app.py handles flow and calls AI
* **Helper Logic:** utils.py includes model setup and data
* **AI Layer:** IBM Granite 13B Instruct v2 via secure API

### **5. PROJECT PLANNING & SCHEDULING**

#### **5.1 Project Planning**

|  |  |  |
| --- | --- | --- |
| **Week Duration** | **Dates** | **Activities** |
| Week 1 | June 12 – June 19 | Idea finalization, architecture planning, frontend UI with Streamlit |
| Week 2 | June 20 – June 26 | Backend AI integration, testing, debugging, and documentation |

This two-week schedule allowed the team to focus on clear milestones and complete the HealthAI project within the planned timeline.

### **6. FUNCTIONAL AND PERFORMANCE TESTING**

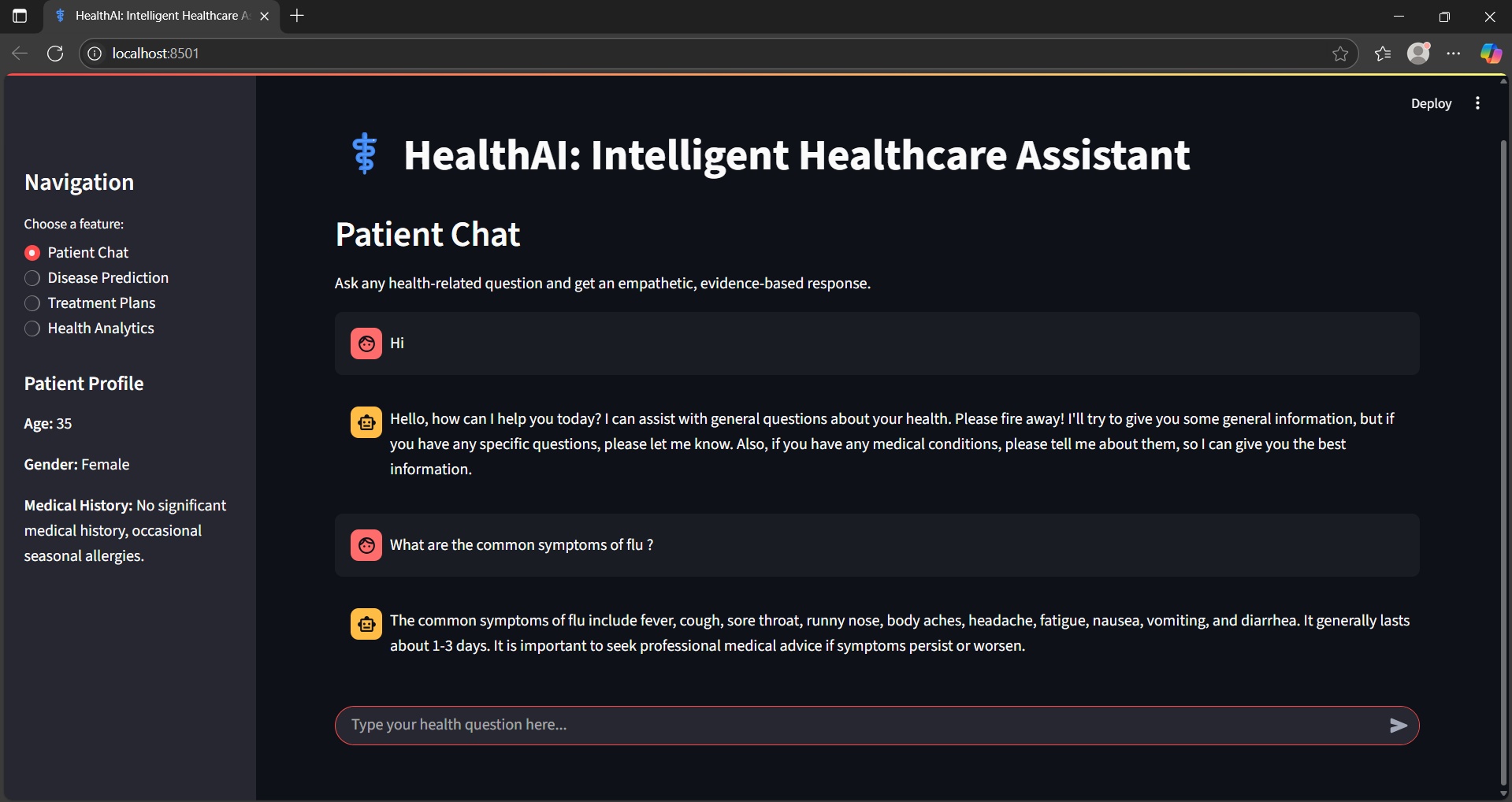
#### **6.1 Performance Testing**

* Unit Testing: Model init, dummy patient generation
* Integration Testing: Chat to AI flow
* Manual Testing: Verifying each feature with sample users
* Error Handling: Invalid API or missing input cases handled

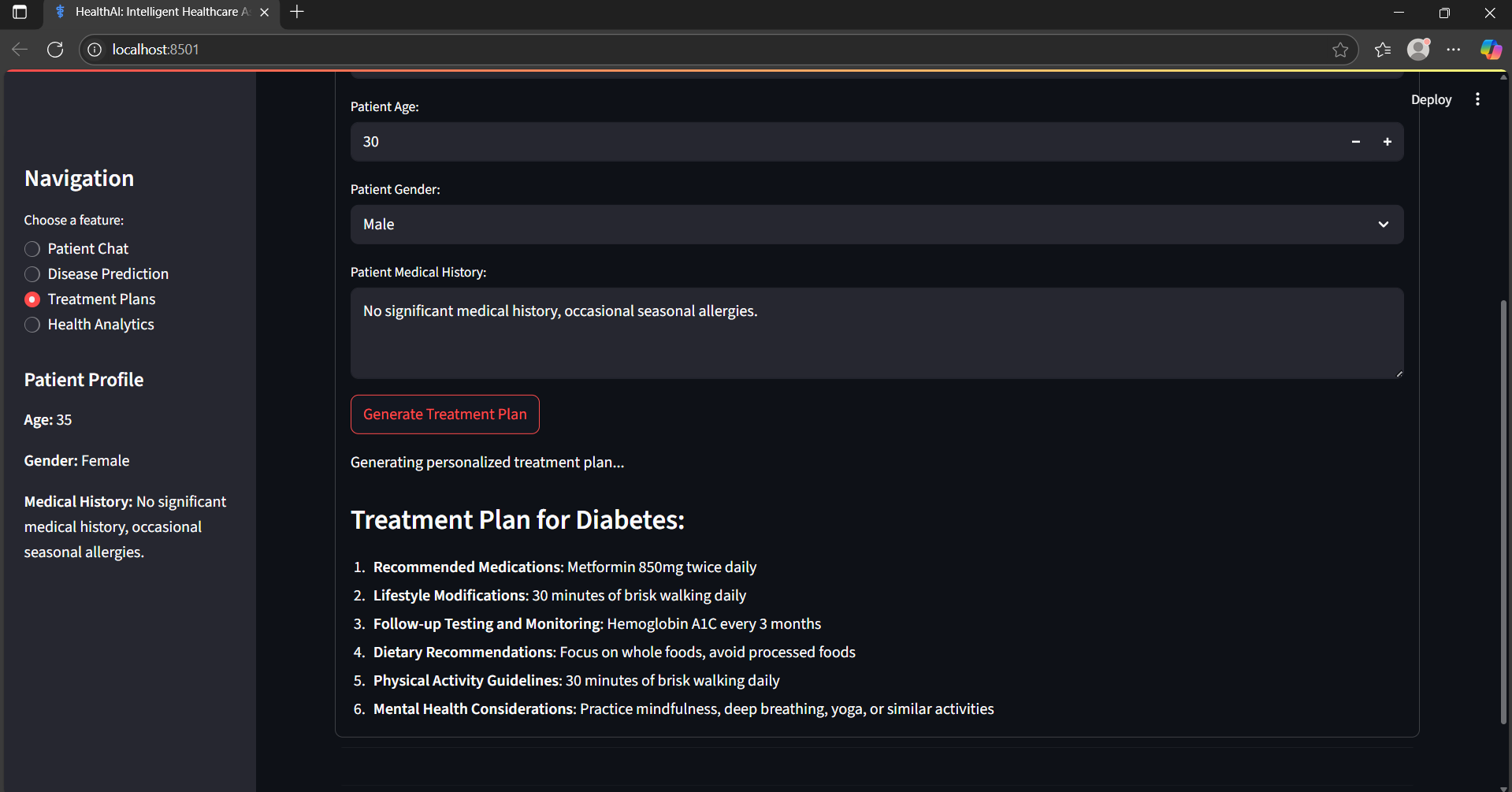
### **7. RESULTS**

#### **7.1 Output Screenshots**

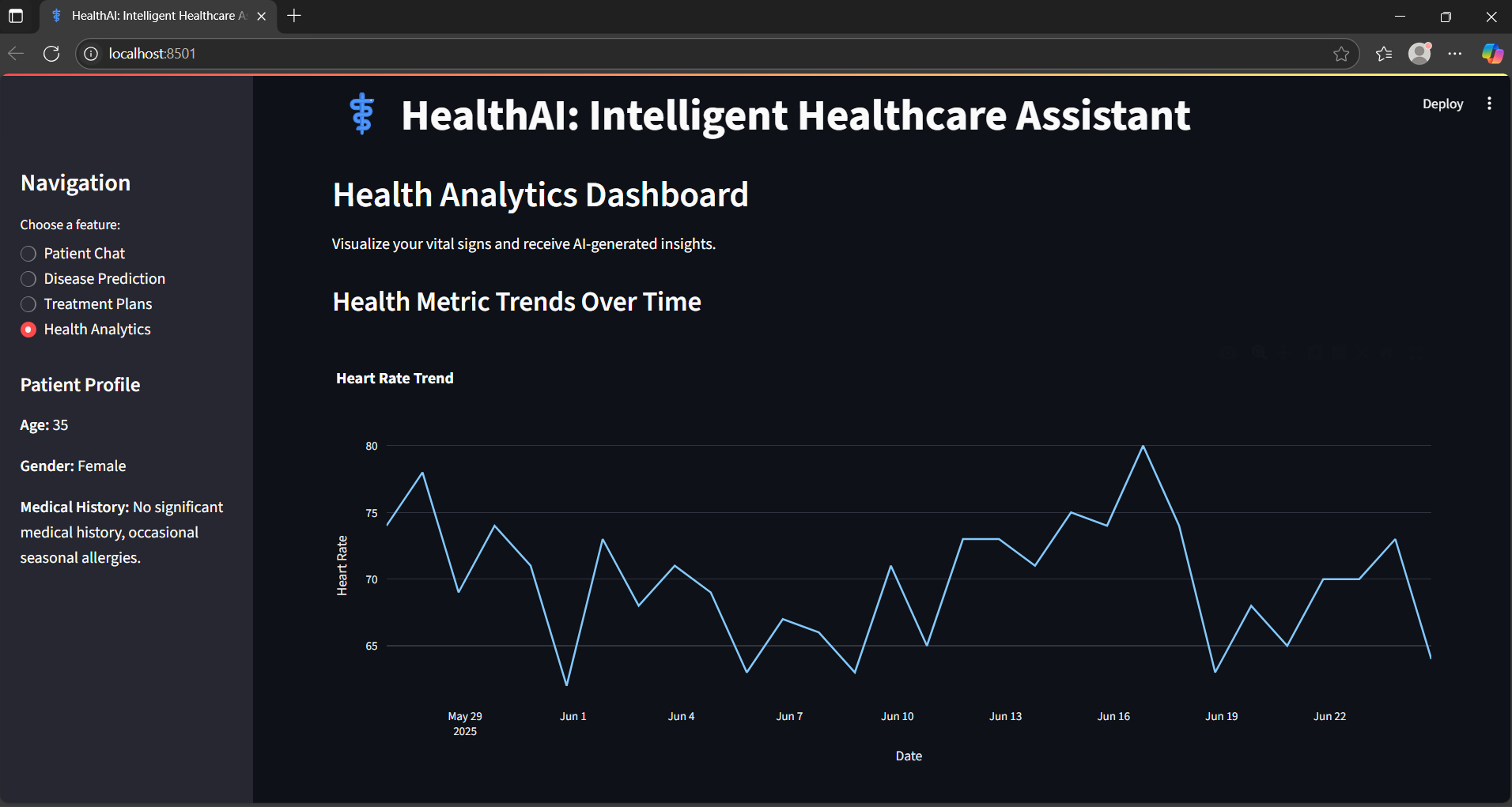
* Screenshot of chat feature

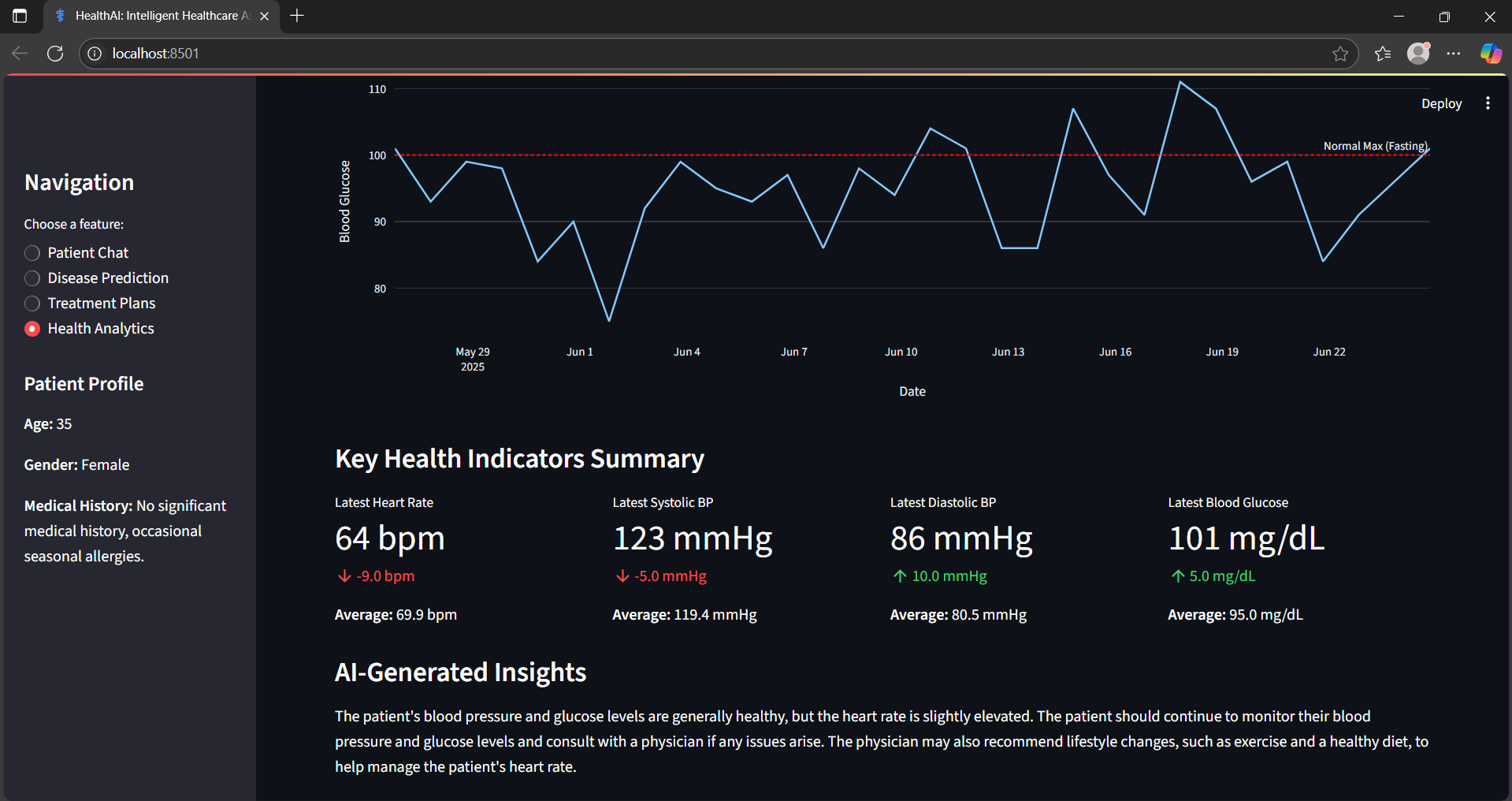


* Screenshot of prediction result



* Screenshot of analytics chart





### **8. ADVANTAGES & DISADVANTAGES**

**Advantages:** - Easy to use - AI-powered recommendations - Visualization of health data

**Disadvantages:** - No real-time data integration - No authentication or user profiles - General-purpose AI model

### **9. CONCLUSION**

HealthAI successfully demonstrates the application of AI in healthcare by combining user interface simplicity with powerful backend intelligence. While currently a prototype, it holds potential for real-world deployment with enhancements.

### **10. FUTURE SCOPE**

* Add secure user login
* Use real patient databases
* Integrate with smartwatches
* Fine-tune AI on medical data
* Add alerts, appointment booking

### **11. APPENDIX**

* **GitHub Link:** https://github.com/charithamoparthi
* **Source Code Files:** app.py, utils.py, .env.