

HealthAI – Intelligent Healthcare Assistant

Project Documentation

1.Introduction

- Project title : HealthAI-Intelligent Healthcare Assistant
- Team Leader : Immanuvel Sagaya Raj A
- Team member : Prince G
- Team member : Nimogandhi G
- Team member :Poongabilian T

2. Project overview

The HealthAI application is a Gradio-based interactive assistant powered by the IBM Granite LLM.

It provides:

1. Disease Prediction from symptoms.
2. Treatment Plan suggestions personalized by age, gender, and history.
3. Health Analytics Dashboard with trends, metrics, and AI-generated health insights.

Disclaimer: This is for informational purposes only.

3.Technologies Used

- Python
- Transformers (HuggingFace) – for LLM (ibm-granite/granite-3.2-2b-instruct)
- Torch – model inference
- Gradio – web app interface
- Pandas & NumPy – health data generation & processing
- Altair – data visualization

4.Features

1. Disease Prediction – User inputs symptoms, AI suggests possible conditions & recommendations.
2. Treatment Plans – User provides condition, age, gender, history → AI generates personalized treatment plan.
3. Health Analytics Dashboard – Shows 90-day health data trends, symptom frequency, statistics, and AI insights.

5.Code Structure

1. Model Setup: Load IBM Granite model & tokenizer.
2. Core Functions:
 - generate_response(prompt, max_length)
 - disease_prediction(symptoms)
 - treatment_plan(condition, age, gender, history)
 - generate_health_insights(summary_string)
3. Data Generation & Processing:
 - Creates 90 days of sample data (heart rate, blood pressure, blood glucose, symptoms)
 - Computes rolling averages, symptom frequency, descriptive statistics
 - Prepares summary_string for AI insights
4. Visualization: Altair charts for vitals & symptoms, integrated with Gradio.
5. Gradio UI Tabs: Welcome, Disease Prediction, Treatment Plans, Health Analytics Dashboard.

6.Running the App

1. Install dependencies:
pip install torch transformers gradio pandas altair
2. Run script:
python health_ai.py
3. Access Gradio interface via local URL from terminal.