Health AI – Project Report

1. Project Overview

Health AI is an AI-powered medical assistant designed to provide symptom analysis and general treatment suggestions.

It uses the IBM Granite 3.2 2B Instruct model (a large language model) to generate natural language responses.

2. Features - Disease Prediction: Enter symptoms and receive possible conditions with recommendations.

- Treatment Plan Generation: Provides personalized treatment suggestions based on condition, age, gender, and medical history.

- User-Friendly Interface built using Gradio.

3. Tech Stack

- Python

- Transformers (IBM Granite model)

- PyTorch

- Gradio

4. System Architecture

1. Input Layer – User enters symptoms or patient details.

2. Processing Layer – Granite LLM processes text input and generates response.

3. Output Layer – Display results in Gradio UI.

5. Installation & Setup

Prerequisites

- Python 3.8+

- pip or conda

- GPU recommended for faster inference

Install Dependencies

pip install torch transformers gradio

Run the App

python health_ai.py

6. Usage

Disease Prediction

- Enter symptoms (e.g., fever, cough, headache).
- Click 'Analyze Symptoms'.
- Get possible conditions and general advice.

Treatment Plan

- Enter condition, age, gender, medical history.
- Click 'Generate Treatment Plan'.
- Receive a general treatment plan with home remedies.

7. Example Outputs

Example 1 – Disease Prediction

Input: Symptoms: fever, cough, fatigue

Output: Possible Conditions: Common Cold, Flu, COVID-19, or other viral infections.

Recommendations: Rest, hydration, fever reducers. IMPORTANT: Consult a doctor.

Example 2 – Treatment Plan

Input: Condition: Hypertension, Age: 45, Gender: Male, History: Diabetes

Output: Lifestyle changes: reduce salt, exercise, manage stress.

Possible medication: antihypertensives (consult doctor).

8. Limitations

- Not a substitute for professional medical advice.
- May generate inaccurate or incomplete results.
- Limited to text-based suggestions.

9. Future Enhancements

- Integration with verified medical APIs.
- Multi-language support.
- Improved UI/UX with structured outputs.
- Patient history tracking.