PROJECT DOCUMENTATION

PROJECT TITTLE: HEALTH AI

PROJECT ID: NM2025TMID02783

TEAM LEADER:

NAME: DHANUSH KUMAR V

MAIL ID: dhanush170306@gmail.com

TEAM MEMBERS

NAME: GNANASIMEYON G

MAILID: gnanasimeyon1603@gmail.com

NAME: DHEEPAN SENTHILNATHAN

MAIL ID: vlndheepan@gmail.com

NAME: SARAN V

MAIL ID: Saranvelmurugansaranvelmurugan@gmail.com

FEATURES

1. Disease Prediction

Input: User symptoms (e.g., "fever, headache, cough").

Output: Possible medical conditions and general recommendations.

2. Treatment Plans

Input: Condition, Age, Gender, and Medical History.

Output: Personalized treatment suggestions (home remedies + general medication).

3. Gradio Interface

User-friendly tabbed interface with two sections:

Disease Prediction

Treatment Plans

Installation

1. Clone the Repository

git clone https://github.com/your-username/health-ai-assistant.git

cd health-ai-assistant

2. Create a Virtual Environment (Optional but Recommended)

python -m venv venv

source venv/bin/activate # On Linux/Mac

venv\Scripts\activate # On Windows

3. Install Dependencies

pip install -r requirements.txt

requirements.txt should include:

gradio

torch

transformers

▶️ Usage

Run the app with:

python app.py

If using Google Colab or Jupyter Notebook, just run all cells.

The app will start and generate a public Gradio link (if share=True is enabled).

---

Project Structure

health-ai-assistant

│

├── app.py # Main application file

├── requirements.txt # Dependencies

├── README.md # Documentation

Code Overview

1. Model Loading

model\_name = "ibm-granite/granite-3.2-2b-instruct"

tokenizer = AutoTokenizer.from\_pretrained(model\_name)

model = AutoModelForCausalLM.from\_pretrained(model\_name)

2. Text Generation Function

Handles prompt → model → response.

def generate\_response(prompt, max\_length=1024):

...

return response

3. Disease Prediction Function

def disease\_prediction(symptoms):

prompt = f"... Symptoms: {symptoms} ..."

return generate\_response(prompt, max\_length=1200)

4. Treatment Plan Function

def treatment\_plan(condition, age, gender, medical\_history):

prompt = f"... Condition: {condition}, Age: {age} ..."

return generate\_response(prompt, max\_length=1200)

5. Gradio Interface

Two Tabs: Disease Prediction & Treatment Plans.

Each tab has input widgets + output display.