■ Medical AI Assistant Using IBM Granite LLM

1. Introduction

- Project Title: Medical AI Assistant Using IBM Granite LLM
- Team Member(s): [Add names here if applicable]

2. Project Overview

Purpose:
The Medical AI Assistant is an AI-powered application that provides health-related insights through two core functionalities:
1. Disease Prediction – Suggests possible medical conditions based on user-input symptoms.
2. Treatment Plan Generation – Offers general treatment suggestions, home remedies, and guidelines based on patient details.

Features:
1. Disease Prediction
2. Personalized Treatment Plan Suggestions
3. AI-Powered Medical Insights
4. User-Friendly Gradio Interface
5. Flexible Input Options (textboxes, dropdowns, numbers)
6. Accessible Deployment with Public Sharing

3. Architecture

Frontend (Gradio):
- Built with Gradio Blocks with two tabs (Disease Prediction, Treatment Plan).
- Collects inputs via textboxes, dropdowns, and numbers.

Backend (Python):
- Handles tokenization, prompt creation, response formatting.

LLM Integration:
- Uses ibm-granite/granite-3.2-2b-instruct from Hugging Face.
- GPU support if available, fallback to CPU.

Deployment Layer:
- Runs locally or publicly with share=True.

4. Setup Instructions

Prerequisites:
- Python 3.9+
- pip & virtual environment tools
- Hugging Face Transformers & Torch

Installation:
1. Clone repository
2. Install dependencies from requirements.txt
3. Run: python app.py
4. Access Gradio link in browser

5. Folder Structure

app/ – Main application folder
■■■ disease_prediction.py – Symptom-based analysis
■■■ treatment_plan.py – Treatment plan generation
■■■ utils.py – Helper functions
■■■ model_loader.py – Loads IBM Granite LLM
■■■ interface.py – Defines Gradio UI
main.py – Entry script
requirements.txt – Dependencies
README.md – Documentation

6. Running the Application

Run: python main.py
Gradio launches at: http://127.0.0.1:7860
Public link available with share=True

7. API Documentation (Future Integration)

- POST /analyze-symptoms → Returns conditions & recommendations
- POST /treatment-plan → Generates treatment plan

8. Authentication

Current demo runs open.
Secure options: JWT, OAuth2, Role-based access.

9. User Interface

Tabs:
1. Disease Prediction – Symptom input → Possible conditions
2. Treatment Plan – Patient details → Treatment plan

Disclaimer: Informational only, not a replacement for medical advice.

10. Testing

- Unit testing for prompts and responses.
- Manual Gradio testing.
- Edge case handling (empty inputs, long symptom lists).

11. Known Issues

- AI suggestions not 100% accurate.
- Long inputs truncated.
- Slow first load without GPU.
- No authentication in demo.

12. Future Enhancements

- Integration with medical databases (PubMed, WHO).
- Multi-language support.
- Secure authentication.
- Symptom severity scoring.
- Telemedicine integration.
- Mobile-friendly deployment.