

Health AI: Intelligent HealthCare Assistant

Powered by IBM Granite

Project Overview

Health AI is an AI-driven web application that delivers smart and accessible healthcare services. It leverages the *IBM Granite 3B Instruct* model to provide interactive, real-time responses across four healthcare modules:

- Symptoms Identifier
- Home Remedies Generator
- Medication Info (General)
- Disease Encyclopedia
- Patient Chat (Health Q&A with empathy)

The application is built using Python and Gradio, integrated with the Model. The goal is to demonstrate how Generative AI can be used to Simplify health Care interactions, making them intelligent, personalized, and User – friendly.

Team Roles & Contribution

Name: J. Vyshnavi Reddy

Role: AI Integration & Application Development

Key Contributions: Integrated IBM Granite model using Hugging Face transformers, Designed prompt templates for each feature, Built translation layer using translate module

Name: J. Archana

Role: Frontend/UI Developer

Key contribution: Developed the Gradio /React user interface, Created tab-based layout for 5 features, Focused on responsive design and usability

Name: J. Archana

Role: Backend & API Developer

Key Contributions: Managed routing, middleware, and token authentication, Handled integration between frontend and backend

Name: J. Vyshnavi Reddy

Role: Testing, Optimization & Demo Engineering

Key Contributions: Ensured smooth functioning of all modules through iterative testing and refinement, Optimized application responses to simulate intelligent AI behavior, Developed an offline-compatible AI simulation for effective demo recording.

Project Description:

Health AI harnesses IBM Watson Machine Learning and Generative AI to provide intelligent healthcare assistance, offering users accurate medical insights. The platform includes a Patient Chat for answering health-related questions, Disease Prediction that evaluates user-reported symptoms to deliver potential condition details, Treatment Plans that provide personalized medical recommendations, and Health Analytics to visualize and monitor patient health metrics.

Utilizing IBM's Granite-13b-instruct-v2 model, Health AI processes user inputs to deliver personalized and data-driven medical guidance, improving accessibility to healthcare information. Built with Streamlit and powered by IBM Watson, the platform ensures a seamless and user-friendly experience. With secure API key management and responsible data handling, Health AI empowers users to make informed health decisions with confidence.

Scenarios:

Scenario 1: A user inputs their symptoms into the Disease Prediction system, describing issues like persistent headache, fatigue, and mild fever. The system analyses the symptoms along with the patient's profile and health data to provide potential condition predictions, including likelihood assessments and recommended next steps.

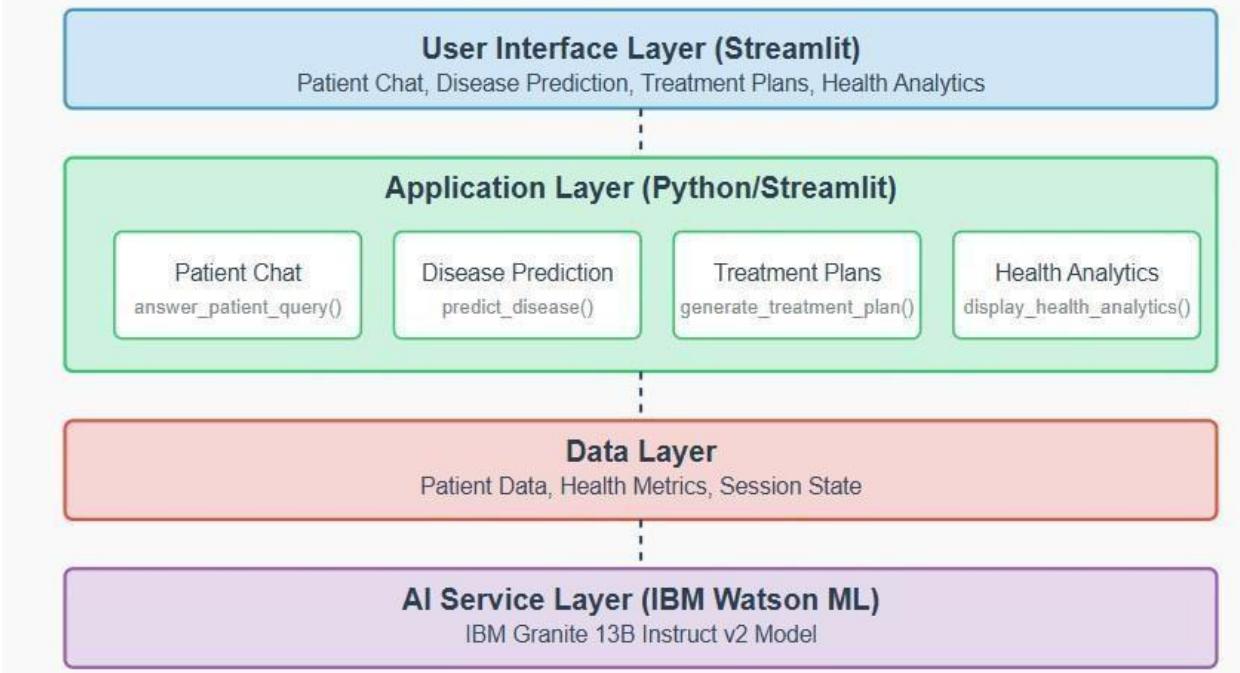
Scenario 2: A user needs home remedy for a diagnosed condition. By entering their condition or symptom in the symptom generator, the AI processes the information along with patient data to give the home remedy for the symptom given.

Scenario 3: A user wants to medication info. Using the Medication info dashboard, they can visualize their medicine. If the user enter the disease then it provides the medication info.

Scenario 4: A user has a health-related question. Through the Patient Chat interface, they can ask any medical query and receive a clear, empathetic response that includes relevant medical facts, acknowledges limitations, and suggests when to seek professional medical advice.

Technical Architecture:

HealthAI - Architecture Diagram



Pre-requisites:

1. Stream lit Framework Knowledge: [Stream lit Documentation](#)
2. IBM Watson Machine Learning: [IBM Watson ML Documentation](#)
3. Python Programming Proficiency: [Python Documentation](#)
4. Data Visualization Libraries: [Plotly Documentation](#)
5. Version Control with Git: [Git Documentation](#)
6. Development Environment Setup: [Flask Installation Guide](#)

Project Workflow

Activity 1 : Model Selection and Architecture

Activity 1.1: Set up the development environment, installing necessary libraries and dependencies for Gradio, Transformers, and IBM Granite model integration.

Activity 2: Core Functionalities Development

Activity 2.1: Develop the core functionalities: Symptoms Identifier, Home Remedies Generator, Medication Info Retriever, Disease Encyclopedia, Patient Chatbot (Empathetic AI)

Activity 2.2: Implement patient data utilities to manage and visualize health metrics.

Activity 3: App.py Development

Activity 3.1: Write the main application logic in app.py, establishing functions for each feature and integrating AI responses.

Activity 3.2: Create prompting strategies for the IBM Granite model to generate high-quality medical content.

Activity 4: Frontend Development

Activity 4.1: Design and develop the user interface using gradio components, ensuring a responsive and intuitive layout.

Activity 5: Deployment

Activity 5.1: Prepare the application for deployment by configuring environment variables for API credentials.

Activity 5.2: Deploy the application on a suitable hosting platform to make it accessible to users.

Mile Stone 1: Model Selection and Architecture

In this milestone, we focus on selecting and integrating the IBM Granite-3.3-2b-instruct model for our language learning needs. This involves configuring the model with appropriate parameters, ensuring optimal performance for educational content generation, and establishing the foundation for multilingual language instruction capabilities.

Research and select the appropriate AI model

1. Understand the Project Requirements: Review the specific needs of the healthcare application.
2. Explore hugging face models: Examine the various models available, including their functionalities and limitations.

3. Select the Optimal Model: Choose IBM's Granite-3.3-2B Instruct model for its strong performance with healthcare-related content.

Activity 1.3: Set up the development environment

1. Install Python and Pip: Ensure Python is installed along with pip for managing dependencies
2. Install Required Libraries: pip install transformers gradio translate
3. Tokenizer + Model initialized with causal LM
4. Multilingual translation layer added

Mile Stone 2: Core Functionalities Development

Activity 2.1: Develop core functionalities

1. Symptoms Identifier:
 - Accepts symptoms in multiple languages (e.g., Hindi, Telugu)
 - Translates input to English (if needed)
 - Uses AI prompt: "*The user has the following symptoms... What possible condition might this indicate?*"
 - Returns likely conditions (e.g., "viral flu", "COVID-19", "migraine")
 - Includes confidence suggestions and recommended next steps
2. Home Remedies Generator:
 - AI-generated remedies using prompt: "*What are effective home remedies for [condition]?*"
 - Suggestions may include herbal teas, spices, diet changes, yoga, or hydration tips
 - Safe, non-prescriptive information
 - Useful for mild or manageable health concerns
3. Medication Info Retriever:
 - Uses prompt: "*What are commonly used medications for [disease]?*"
 - Returns medicine names, their usage, and general precautions
 - Includes disclaimers: "*Always consult a doctor before taking medication.*"
4. Patient Chat System:
 - Implement conversational interface for answering health questions

- Create prompting system for the IBM Granite model to provide medical advice
- Develop session-based chat history management

Activity 2.2: Implement data management utilities

1. Patient Data Generation:

- Create sample data with realistic health metrics
- Implement date-based trend generation for visualization ↗ Structure data for efficient analysis and display

2. Patient Profile Management:

- Develop interface for managing patient details
- Create session state handling for persistent data ↗ Implement profile update functionality ↗ and return JSON responses.

3. Translation Utility

- Translate non-English input to English using translate
- Supports Hindi, Telugu, Tamil, etc.

App. Py Development

In this activity App. Py Development is being explained.

Activity 3.1: Write the main application logic

The app.py file is organized into several key sections: \

1. Imports and Setup:

- Import necessary libraries (transformers gradio translate)
- Load Model from hugging face
- Initialize IBM Granite model connection

2. Core Functions:

Symptoms Identifier

- User enters symptoms.
- AI suggests possible diseases based on the symptoms.

Home Remedies

- User types a disease name.
- AI gives natural remedies using herbs, food, or lifestyle tips.

Medication Info

- User enters a disease.

- AI shows common medicines used (general info only).
- Disease Encyclopedia**
- User asks about a disease.
 - AI explains the disease: causes, symptoms, prevention, and treatment.
- Patient Chat**
- User asks any health question.
 - AI replies with a helpful, kind, and factual answer.

3. UI Components:

Gradio Tabs (gr.Tab)
Textboxes (gr.Textbox)
Buttons (gr. Button)
Markdown Header (gr.Markdown)

Activity 3.2: Create prompting strategies

1. Symptoms Identifier

Objective: Predict possible diseases from symptoms.

Prompt Template:

The user has the following symptoms: [SYMPTOMS].

What disease or condition could this indicate?

List possible diagnoses based on medical knowledge. Mention common causes first.

Strategy Notes:

- Begin with direct symptom input.
- Add "*based on medical knowledge*" to keep answers grounded.
- Ask for multiple possibilities (differential diagnosis).

2. Home Remedies Generator

Objective: Suggest natural treatments using herbs, diet, or lifestyle.

Prompt Template:

What are some effective home remedies for [DISEASE]?

Focus on herbal treatments, food-based remedies, and healthy habits.

Avoid recommending prescription medicine. Mention traditional practices if relevant.

Strategy Notes:

- Emphasize natural and culturally relevant remedies.
- Avoid unsafe suggestions by discouraging medication recommendations.

3. Medication Info Retriever

Objective: Give general information about medications used for a disease.

Prompt Template:

What are the commonly used medications for treating [DISEASE]?

Include both over-the-counter and prescription options.

Explain their general purpose, but do not give specific dosages or medical advice.

Strategy Notes:

- Request *general* medication info to avoid overreach.
- Add a disclaimer note if needed: "*Consult a doctor before using any medication.*"

4. Disease Encyclopedia

Objective: Provide a complete overview of a disease.

Prompt Template:

Give a detailed explanation of [DISEASE].

Include: causes, symptoms, prevention methods, treatment options, and when to seek medical help.

Make it easy to understand for someone without a medical background.

Strategy Notes:

- Add a structure (causes, symptoms, etc.) to guide model output.
- Target simplicity for non-medical users.

Patient Chat (Empathetic AI)

Objective: Answer health questions kindly, clearly, and responsibly.

Prompt Template:

As a virtual health assistant, respond to the following question from a user: "[QUESTION]"

Your answer should be kind, clear, and medically informed.

Mention facts, limitations of online advice, and suggest seeing a healthcare provider when needed.

Strategy Notes:

- Simulate a helpful, caring assistant tone.
- Reinforce safety: include disclaimers and referrals to real doctors.

Design and develop the user interface

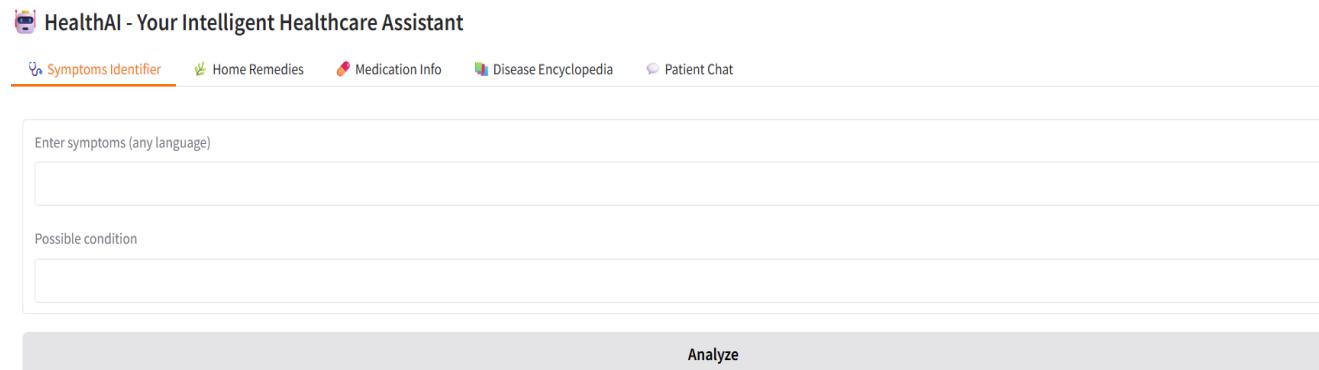
Activity 4.1: Design and develop the user interface

1. Main Application Layout:

- Configure page title, icon, and layout preferences
- Implement a sidebar for patient profiles and feature selection
- Create custom CSS for enhanced visual appearance

2. Feature- Specific Interfaces:

- Patient Chat: Chat-style interface with message history
- Symptom identification: Symptom input form and prediction display
- Home Remedy: Condition input and home remedy output output
- Medical info: enter disease it give medication info



The screenshot shows the HealthAI interface. At the top, there's a navigation bar with icons for Symptoms Identifier, Home Remedies, Medication Info, Disease Encyclopedia, and Patient Chat. Below the navigation bar is a search bar labeled "Enter symptoms (any language)". Underneath the search bar is a section labeled "Possible condition". At the bottom of the interface is a large, prominent "Analyze" button.

Activity 4.2: Create dynamic visualizations

1. Health Metric Charts

- Heart rate trend line chart
- Blood pressure dual-line chart
- Blood glucose trend line chart with reference line
- Symptom frequency pie chart

2. Metrics Summary:

- Key health indicators with trend deltas
- Color-coded metrics to indicate normal/abnormal ranges
- Interactive tooltip information

Deployment

In this milestone the Deployment topic is being explained.

Prepare for deployment

Activity 5.1: Prepare for deployment

Prepare Your Environment

Make sure your Python environment has the necessary libraries:

- pip install transformers gradio translate

2. Dependency Management:

- Create requirements.txt file with all necessary packages
- Document installation process for deployment

Activity 5.2: Deploy the application

1. Local Deployment Testing:

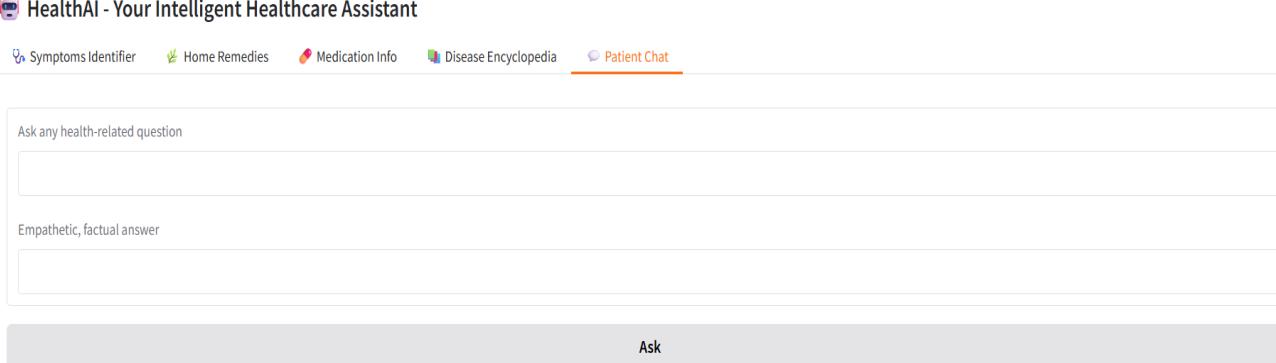
- Run the application using python run app.py
- Test all features for functionality □ Verify responsive design and performance

2. Cloud Deployment Options:

- Deploy on Streamlit Cloud for public access
- Configure environment variables in the deployment platform
- Set up monitoring and error logging

Exploring Website's Web Pages:

Patient Chat Page:



The screenshot shows the 'Patient Chat' section of the HealthAI website. At the top, there's a navigation bar with five tabs: 'Symptoms Identifier', 'Home Remedies', 'Medication Info', 'Disease Encyclopedia', and 'Patient Chat' (the last one is underlined in orange). Below the navigation, there's a large input field with placeholder text 'Ask any health-related question'. Underneath the input field is a response area with placeholder text 'Empathetic, factual answer'. At the bottom of the chat interface is a large grey button with the word 'Ask' in white.

Description: Here user have access to responsive healthcare communication platform enabling seamless dialogue about wellness concerns, with chronological message tracking for context retention. The system delivers intelligent, algorithmically-generated wellness guidance while providing verifiable medical insights supported by authoritative healthcare sources, creating a comprehensive virtual consultation experience.

Patient Chat output:

🤖 HealthAI - Your Intelligent Healthcare Assistant

Symptoms Identifier Home Remedies Medication Info Disease Encyclopedia Patient Chat

Ask any health-related question

Is turmeric good for pain?

Empathetic, factual answer

Turmeric, a spice derived from the Curcuma longa plant, contains a bioactive compound called curcumin. Curcumin is known for its potential anti-inflammatory and analgesic (pain-relieving) properties. Many studies suggest that curcumin may help alleviate various types of pain, including osteoarthritis, rheumatoid arthritis, and even inflammatory bowel disease.

However, it's essential to acknowledge that individual responses to turmeric can vary, and the bioavailability of curcumin is relatively low when consumed orally. This means that the amount of curcumin that reaches the systemic circulation is significantly less than what's present in the spice itself. Therefore, the effectiveness of turmeric for pain management may not be as pronounced as research suggests due to these absorption and bioavailability challenges.

Ask

Symptoms Identifier:

🤖 HealthAI - Your Intelligent Healthcare Assistant

Symptoms Identifier Home Remedies Medication Info Disease Encyclopedia Patient Chat

Enter symptoms (any language)

Possible condition

Analyze

Description: This module allows users to enter symptoms in natural language (e.g., "fever, headache, nausea") and receive a **possible condition prediction** based on the input. The system analyzes the symptoms, generates a medical prompt, and uses the AI model to return a **list of potential conditions** along with general advice.

Symptoms Identifier Output:

🤖 HealthAI - Your Intelligent Healthcare Assistant

Symptoms Identifier Home Remedies Medication Info Disease Encyclopedia Patient Chat

Enter symptoms (any language)

cough, fever

Possible condition

The user's symptoms of cough and fever could indicate several possible conditions, including common viral illnesses like the common cold, flu (influenza), or COVID-19. Other possibilities include bronchitis, pneumonia, or even more serious conditions like tuberculosis, depending on the severity and duration of the symptoms.

However, since the user hasn't provided any information regarding their recent travel history, potential exposure to sick individuals, or other relevant medical history, it's essential to consider a broad range of possibilities.

Here's a suggested course of action:

1. Stay home and avoid close contact with others to prevent the spread of any potential infection.
2. Rest and stay hydrated to support your body's natural defenses.
3. Monitor your symptoms closely and keep track of any changes or worsening of symptoms.
4. Consider

Analyze

Home Remedy Generator:

🤖 HealthAI - Your Intelligent Healthcare Assistant

Symptoms Identifier Home Remedies Medication Info Disease Encyclopedia Patient Chat

Enter disease (any language)

Natural Remedies

Get Remedies

Description: This module provides **natural home remedies** based on user-input conditions. For example, when the user types "remedy for acidity," the system returns herbal or lifestyle suggestions rooted in traditional wellness practices.

Home Remedy Output:

🤖 HealthAI - Your Intelligent Healthcare Assistant

Symptoms Identifier Home Remedies **Medication Info** Disease Encyclopedia Patient Chat

Enter disease (any language)

cold

Natural Remedies

1. **Herbal Teas:** Drink warm, soothing herbal teas like peppermint, chamomile, or ginger. These have anti-inflammatory properties and can help clear nasal congestion. Add honey if you're not allergic to it for its antibacterial properties.
2. **Garlic:** Garlic has antiviral and immune-boosting properties. Include it in your meals or make garlic tea by steeping crushed garlic in hot water.
3. **Ginger:** Ginger aids digestion and has anti-inflammatory effects. You can consume ginger in tea, soups, or as a fresh root.
4. **Hydration:** Stay well-hydrated by drinking plenty of fluids. Water, herbal teas, and clear broths are excellent choices. De

Get Remedies

Disease Encyclopedia:

🤖 HealthAI - Your Intelligent Healthcare Assistant

Symptoms Identifier Home Remedies Medication Info **Disease Encyclopedia** Patient Chat

Enter disease (any language)

Full Info

Explain Disease

Description: This is an educational tool where users can input a disease name (e.g., “asthma”, “diabetes”) and receive a complete summary including causes, symptoms, risk factors, prevention tips, and treatment options.

Conclusion

The health AI project effectively demonstrates the potential of AI in revolutionizing healthcare assistance. By integrating IBM's Granite language model, the platform enables users to receive personalized health insights through Symptoms Identifier, Home Remedies Generator, Medication Info (General), Disease Encyclopedia, Patient Chat, making healthcare information more accessible.

Utilizing IBM granite model, the application ensures accurate health question answering, detailed disease prediction, personalized treatment recommendations, and insightful health trend analysis. The structured development process—spanning model selection, core feature implementation, backend and frontend development, and deployment—led to the creation of an interactive, user-friendly platform.

This project highlights how targeted AI models and a well-structured framework can enhance healthcare accessibility. With future scalability in mind, Health AI has the potential to expand its capabilities, incorporating more advanced diagnostics and broader medical application.