



HEALTHAI: Intelligent Healthcare Assistant using IBM Granite

1. INTRODUCTION

1.1 Project Overview

HEALTHAI: Intelligent Healthcare Assistant using IBM Granite is a generative AI-powered application designed to provide smart healthcare support to patients through an interactive and intuitive interface. The system leverages IBM's Granite language model to facilitate health-related conversations, predict diseases based on symptoms, suggest possible treatment plans, and display useful health analytics. Developed using Python and Streamlit, the application aims to simplify patient engagement and support early diagnosis and treatment planning through AI.

1.2 Purpose

The primary purpose of this project is to harness the power of Generative AI for delivering accessible, reliable, and intelligent healthcare support. HEALTHAI serves as a virtual health assistant that helps users:

- Get instant responses to general health queries.
- Predict diseases based on symptoms using AI.
- Receive relevant treatment suggestions.
- View simple, clear analytics on health trends.

This project also demonstrates the practical application of IBM Granite models in solving realworld healthcare problems, fulfilling academic and internship goals under the IBM Generative AI program.

IDEATION PHASE

1.3 Problem Statement

Date: 27 June 2025

Team ID: LTVIP2025TMID59552

Project Name: Health AI: Intelligent Healthcare Assistant Using IBM Granite.

Marks: 4 Marks

Customer Problem Statement Template

Create a problem statement to understand your customer's point & view. The Customer Problem Statement helps you focus on what mat-ters to create experiences people will love.

A well-articulated customer problem stament allows your team and your users to find the ideal solution your business faces. Throughout the process, you'll also be able to empathize with your customergur you better understand your

Template: <https://miro.com/templeplates/customerproblem-statement/>

Example:

| Problem Statement (PS) | (i am) | I'm trying to | But | Which makes me feel |
|------------------------|-----------|------------------------------|---|--|
| PS-1 | a patient | manage my health effectively | I face difficulty | frustrated and anxious about my well-being |
| I'm | | manage my health effectively | I face continued and lacks proacessing and medicaias' the current healthcare system is fragmented and lacks proactive support | |

1.4 Empathy Map Canvas

Date: 27 June 2025

Team ID: LTVIP2025TMID59552

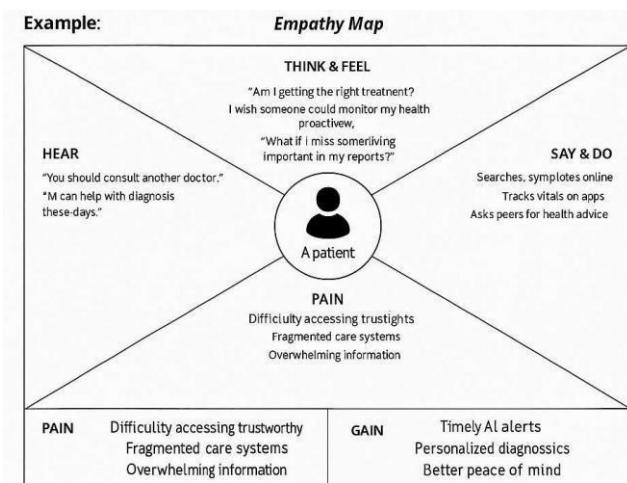
Project Name: Health AI: Intelligent Healthcare Assistant Using IBM Granite Maximum Marks: 2 Marks

Empathy Map Canvas

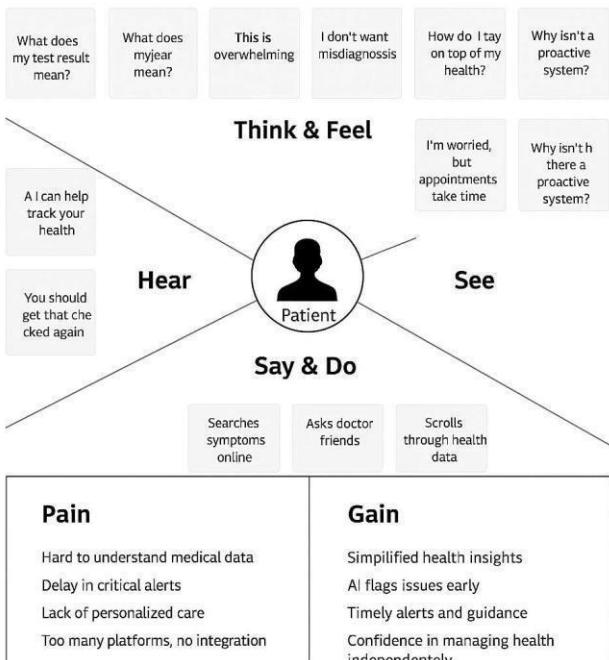
An empathy map a simple, easy-to-digest visual that captures knowledge abou a user's behaviors and attitudes.

It is a useful to helping teams teams understand their users.

Creating an effective solution requires understanding their the person who is experiencing it, it. Exele participants consider how participants consider uset highs, lows, goals, and challenges



Reference: <https://www.mural.co/templates/empathy-map-canxas>



1.5 Brainstorming

Date: 27 June 2025

Team ID: LTVIP2025TMID59552

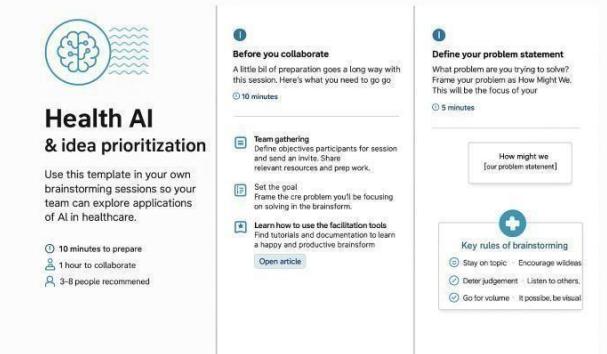
Project Name: Health AI: Intelligent Healthcare Assistant Using IBM Granite Maximum Marks: 4 Marks

Brainstorm & Idea Prioritization in Health AI

Brainstorming in Health AI promotes free, creative thinking to generate innovative solutions for healthcare challenges using artificial intelligence. To collect a wide range of ideas from diverse team members, then prioritize based on impact, feasibility, and urgency. Encourage maximum idea generation, regardless of practicality at first.

Cross-functional team members (AI developers, clinicians, analysts) co-create ideas. Ideal for distributed teams using tools like Miro or Mural. AI-driven symptom checking, disease prediction, treatment plans, and patient engagement tools. Impact – Patient outcomes and healthcare system improvement. Feasibility – Technical readiness with health regulations. **Reference:** Brainstorm and idea prioritization template | Mural

Step-1: Team Gathering, Collaboration and Select the Problem Statement



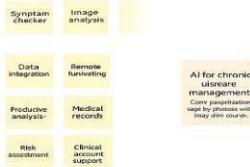
Step-2: Brainstorm, Idea Listing and Grouping



Health AI & idea prioritization

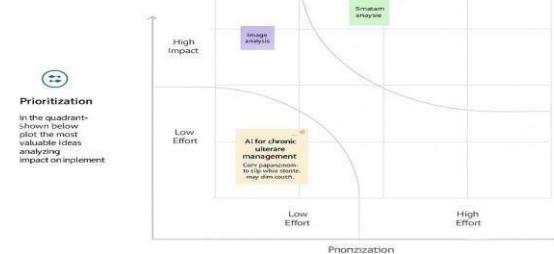
In a brainstorming session, list and group ideas for addressing your problem statement:

Health AI



Step-3: Idea Prioritization

In the quadrants shown below, plot the most valuable ideas, analyzing Impact on the problem vs. effort to implement



2. REQUIREMENT ANALYSIS

2.1 Customer Journey Map

Health AI



3.2 Solution Requirement

Solution Requirements (Functional & Non-functional)

| | |
|--------------|---|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID59552 |
| Project Name | HealthAI-Intelligent Healthcare Assistant Using IBM Granite |

| | |
|---------------|---------|
| Maximum Marks | 4 Marks |
|---------------|---------|

Functional Requirements:

Following are the functional requirements of the proposed solution.

| FRNo. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
|-------|-------------------------------|--|
| FR-1 | User Registration | Registration through Form Registration through Gmail Registration through LinkedIn |
| FR-2 | User Confirmation | Confirmation via Email Confirmation via OTP |
| FR-3 | Disease Prediction | Symptom-based prediction Model scoring using IBM Granite AI |
| FR-4 | Health Assistant Chat | Natural language query handling Context-aware health response |
| FR-5 | Treatment Plan Recommendation | Display treatments based on disease Explain predicted outcome |
| FR-6 | Health Analytics | View past predictions Graphs for health trends |

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

| Functional Requirement | Sprint | Story ID | User Story / Task | Story Points | Priority |
|------------------------|----------|----------|---|--------------|----------|
| Registration | Sprint 1 | US#4 | As a user, I can register for the application (US3) | 5 | High |
| | | US#2 | As a user, oral responses can be analyzed using speech-to-text (US2) | 8 | High |
| Login | Sprint 1 | US#3 | As a user, health data can be input into system | 7 | High |
| | | US#1 | As a user, I can log in to the application | 2 | High |
| Dashboard | Sprint 2 | US#1 | As a user, I can view health data visualizations on the central dashboard (US5) | 2 | Medium |

3.3 Data Flow Diagram

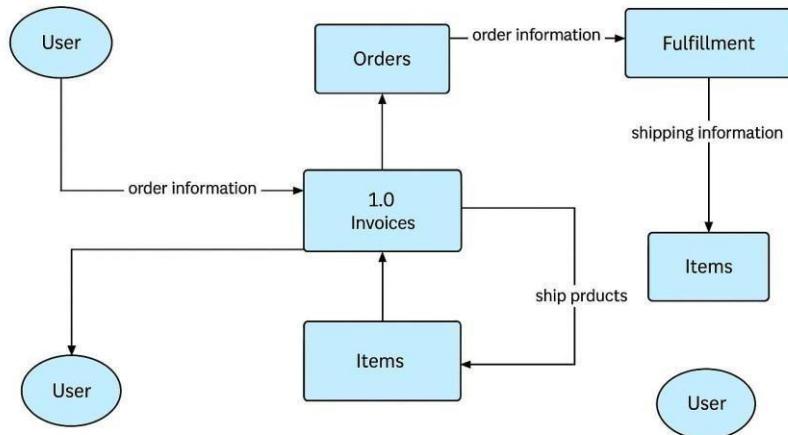
Data Flow Diagram & User Stories

| | |
|---------------|---|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID59552 |
| Project Name | HealthAI-Intelligent Healthcare Assistant Using IBM Granite |
| Maximum Marks | 2 Marks |

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

Example: DFD Level 0 (Industry Standard)



Health AI

| User Type | Functional Requirement | User Story/Task | Acceptance criteria | Priority | Rele |
|---------------------------|------------------------|---|---|----------|--------|
| Customer (Mobile user) | Registration | As a user, I can register by providing an email and password. | Email and password can be used to log in | High | Spri 1 |
| | USS1 | As a user, I will receive confirmation email | Confirmation email received | High | Spri 1 |
| | USS2 | As a user, I can enable systemwide speech-to-text | Speech-to-text is active throughout the app | Low | Spri 2 |
| Tester | USS3 | As a tester, I can analyze speech responses | Speech responses are analyzed correctly | Medium | Spri 1 |
| | USS4 | As an admin, I can view health data visualizations | | Sprint 1 | Spri 1 |
| Administrator | US4 | As a tester, I can analyze speech responses | Speech responses are analyzed correctly | Medium | Spri 1 |
| | US5 | As a tester, I can analyze speech responses | Health data visualizations are available | High | Spri 1 |
| Visualizaation | | | | | |

3.4 Technology Stack

Technology Stack (Architecture & Stack)

| | |
|---------------|---|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID59552 |
| Project Name | HealthAI-Intelligent Healthcare Assistant Using IBM Granite |
| Maximum Marks | 4 Marks |

Technical Architecture – HealthAI

HealthAI's technical architecture is designed to provide intelligent, personalized, and accessible healthcare assistance using IBM's AI capabilities. The architecture bridges the gap between healthcare

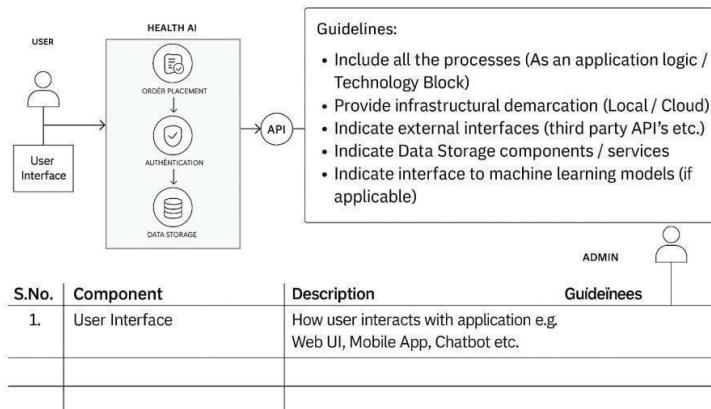
user needs and AI-driven digital solutions by clearly defining modules, workflows, and technology integrations.

It follows principles of modular design, AI integration, secure backend logic, and interactive frontend experiences.

References – Adapted for HealthAI

1. C4 Model – Software Architecture Visualization Used as the base modeling approach to define different levels of HealthAI's architecture (context, container, component). ↗ <https://c4model.com/>
2. IBM Order Processing System (Pandemic Reference) Inspired HealthAI's backend design by using modular components and AI-powered services similar to order-processing use cases. ↗ <https://developer.ibm.com/patterns/online-order-processing-system-during-pandemic/>
3. IBM Cloud Architecture Center Provided best practices and patterns for integrating AI models and deploying cloud-based healthcare applications. ↗ <https://www.ibm.com/cloud/architecture>
4. AWS Architecture Best Practices Used as a comparative reference to validate HealthAI's scalability, resilience, and service-based integration approach. ↗ <https://aws.amazon.com/architecture>
5. How to Draw Useful Technical Architecture Diagrams Guided the creation of simplified, functional diagrams for HealthAI's backend and AI data flow. ↗ <https://medium.com/theinternalstartup/howto-draw-useful-technical-architecture-diagrams-2d20c9fda90d>

Health AI



Health AI Technology Stack

- Application Logic-1: Python / Java
Patient intake and triage processing IBM Watson STT
- Application Logic-2: IBM Watson STT
Voice transcription for patient interactions IBM Watson Assistant
- Database MySQL / MongoDB
- Cloud Database IBM DB2 / IBM Cloudant
- File Storage: Medical imaging and document IBM Block Storage / Local Filesystem
- External API-1 IBM Weather API
Real-time environmental health tracking Aadhaar API
- External API-2 Aadhaar API
- Machine Learning Model Custom Object
Medical image classification Recognition Model
- Infrastructure Cloud Foundry /
Scalable deployment for Kubernetes / Local Server
clinical environments

3. PROJECT DESIGN

3.1 Problem Solution Fit

Problem – Solution Fit Template :

| | |
|---------------|---|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID59552 |
| Project Name | HealthAI-Intelligent Healthcare Assistant Using IBM Granite |
| Maximum Marks | 2 Marks |

Problem – Solution Fit Template : HealthAI solves a frequent and urgent problem: lack of easy access to valid healthcare information and insights. It taps into the existing behavior of users searching for medical information online and replaces it with a credible, AI-powered platform.

Purpose :

- ① Solve complex health-related problems using intelligent and accessible AI assistance
- ② Increase solution adoption by reflecting how users already seek medical information online
- ③ Improve communication using conversational chat and visual analytics
- ④ Build user trust with consistent, evidence-based responses



Health AI Problem-Solution-Fit Template

| | | | | |
|---------------------------|---|-----------|---|---|
| CS ONSTOWER | CUSTOMER SEGMENTS (CS) Who are your target users? e.g. elderly individuals with chronic conditions, rural patients with limited access to healthcare, or caregivers of Alzheimer's patients. | CS | JOBS-TO-BE-DONE / PROBLEMS (J&P) What specific health challenges do they face? e.g. medication adherence, early diagnosis, appointment management, lifestyle monitoring. | J&P |
| TR TRIGGERS | TRIGGERS (TR) What drives users to seek a solution? e.g. worsening symptoms, hospital readmission, advice from a physician or caretaker. | TR | EMOTIONS: BEFORE / AFTER (EM) How do they feel before the solution vs. after? e.g. digital literacy, cost, lack of smartphones, unreliable internet. | EM |
| CC CONSTRAINTS | CUSTOMER CONSTRAINTS (CC) What obstacles might prevent them from accessing help? e.g. digital literacy, cost, lack of smartphones, unreliable internet. | CC | BEHAVIOUR (BE) What do users do to manage their health problems? e.g. rely on memory for meds, ask family members for help. | CHANNLES OF BEHAVIOUR (CH) 8.1 Online: <ul style="list-style-type: none">Health forums,YouTube health advice, telemedicine apps. 8.2 Offline: <ul style="list-style-type: none">primary care visits,local health camps, support groups. |
| RC | PROBLEM ROOT CAUSE (RC) What's the deeper reason this problem exists? e.g. lack of awareness, systemic inefficiencies. | RC | YOUR SOLUTION (SL) Describe your Health AI product or concept. | |

References :

1. <https://www.ideahackers.network/problem-solution-fit-canvas/>
2. <https://medium.com/@epicantus/problem-solution-fit-canvas-aa3dd59cb4fe>

3.2 Proposed Solution

| | |
|---------------|---|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID59552 |
| Project Name | HealthAI-Intelligent Healthcare Assistant Using IBM Granite |
| Maximum Marks | 2 Marks |

Proposed Solution Template:

Project team shall fill following information in the proposed solution template.

| S.No. | Parameter | Description |
|-------|--|---|
| 1. | Problem Statement (Problem to be solved) | Identify a pressing issue in healthcare your AI aims to address |
| 2. | Idea / Solution description | Summarize your Health AI solution and how it works |
| 3. | Novelty / Uniqueness | What makes your idea different from existing healthcare technologies? |
| 4. | Social Impact / Customer Satisfaction | How will it improve lives, patient outcomes, or user experience? |
| 5. | Business Model (Revenue Model) | How will your solution generate revenue or remain sustainable? |

3.3 Solution Architectur

| | |
|------|--------------|
| Date | 27 June 2025 |
|------|--------------|

| | |
|---------------|---|
| Team ID | LTVIP2025TMID59552 |
| Project Name | HealthAI-Intelligent Healthcare Assistant Using IBM Granite |
| Maximum Marks | 2 Marks |

Solution Architecture – HealthAI

Solution architecture in HealthAI serves as the bridge between real-world healthcare challenges and advanced AI-driven technology. It outlines how HealthAI is built to deliver accurate, personalized, and responsive medical support.

Goals of HealthAI's Solution Architecture:

1. Identify the most effective AI-driven technology to solve the problem of inaccessible or unreliable healthcare information.
2. Design the complete structure — from user input (like symptoms or questions) to backend AI processing using IBM Granite and secure API handling.
3. Define key features and development phases, including modules like:
 - o Patient Chat
 - o Disease Prediction
 - o Treatment Plan Generation
 - o Health Analytics

Key Characteristics of the HealthAI Architecture:

- Modular and Scalable Design: Each core functionality is independently built using Python and Streamlit.
- AI Integration: IBM Granite (13B Instruct v2) is used to process all medical queries and generate accurate, natural-language responses.
- . User Interface: Streamlit provides an intuitive frontend with form-based inputs, chatbot interfaces, and dynamic visualizations using Plotly.
- Data Flow: User inputs are sent to the AI model via a central shared function (`shared_model.py`), processed securely, and returned in structured output.
- Security: Environment variables (`.env`) are used for API key management to protect sensitive credentials.

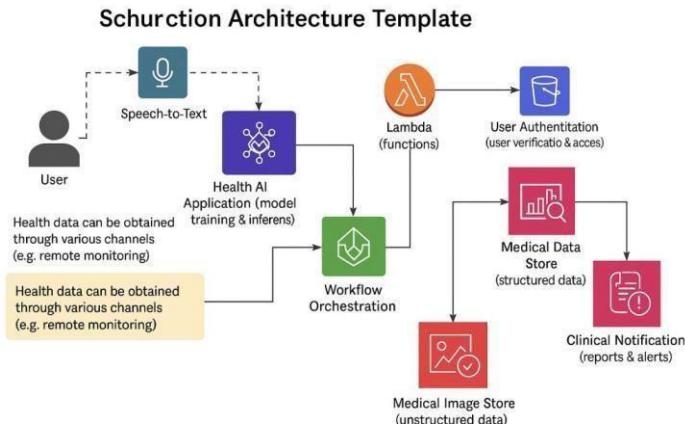


Figure 1: Architecture and data flow of the health AI system

4. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)

| | |
|---------------|---|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID59552 |
| Project Name | HealthAI-Intelligent Healthcare Assistant Using IBM Granite |
| Maximum Marks | 4 Marks |

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV)

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

| Functional Requirement | Sprint | User Story / Task | Story Points | Priority |
|------------------------|----------|---|--------------|----------|
| Registration | Sprint 1 | As a user, I can register for the application (US1) | 5 | High |
| Registration | Sprint 1 | As a user, real responses can be analyzed using speech-to-text (US2) | 8 | High |
| Login | Sprint 1 | As a user, health data can be input into system (US3) | 7 | High |
| Dashboard | Sprint 2 | As a user, I can log in to the application (US4) | 4 | Medium |
| Dashboard | Sprint 2 | As a user, I can view health data visualizations on the central dashboard (US5) | 2 | Medium |

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

| Functional Requirement | Sprint | Story ID | User Story / Task | Story Points | Priority |
|------------------------|----------|----------|---|--------------|----------|
| Registration | Sprint 1 | US#4 | As a user, I can register for the application (US3) | 5 | High |
| | | US#2 | As a user, oral responses can be analyzed using speech-to-text (US2) | 8 | High |
| Login | Sprint 1 | US#3 | As a user, health data can be input into system | 7 | High |
| | | US#1 | As a user, I can log in to the application | 2 | High |
| Dashboard | Sprint 2 | US#1 | As a user, I can view health data visualizations on the central dashboard (US5) | 2 | Medium |

5. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing Functional &

Performance Testing Template

| | |
|---------------|---|
| Date | 27 June 2025 |
| Team ID | LTVIP2025TMID59552 |
| Project Name | HealthAI-Intelligent Healthcare Assistant Using IBM Granite |
| Maximum Marks | |

Model Performance Test

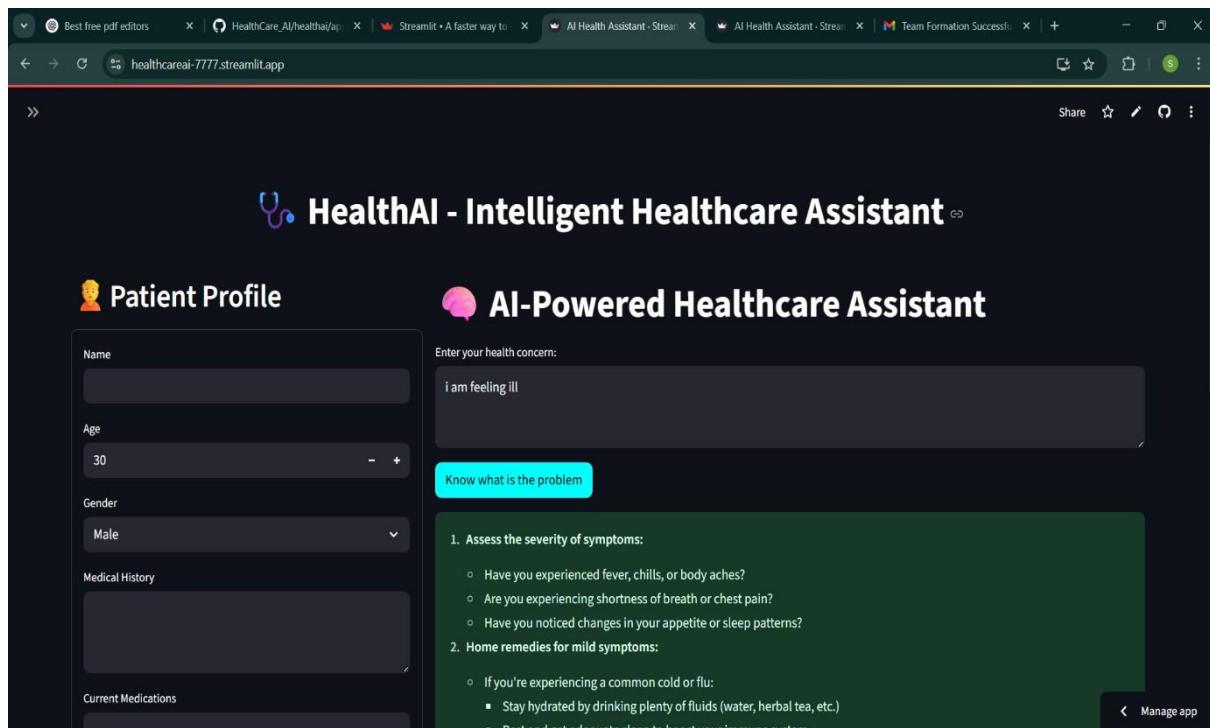
Test Scenarios & Result

Health AI Test Scenarios & Results

| Test Case | Scenario (What to test) | Expected Result | Result |
|-----------|-------------------------------------|-----------------------|------------------------|
| HT-A1 | Input Validation | Valid inputs accepted | Pass |
| HT-A2 | Name Input | Accepts alph, values | Accepts valid values |
| HT-A3 | Symptom Input | Logs correctly | Symptoms log correctly |
| HT-A4 | Content Generation | Created accurately | Generated accurately |
| HT-A5 | API Connection | API responds | API responds |
| HT-A6 | Response Time | Should be acceptable | Within an acceptable |
| HT-A7 | User submitty multiple inputs | Should not slow | Pass |
| HT-A8 | Upload transfer speed during migage | Should not lag | Should not lag |

6. RESULTS

6.1 Output Screenshots



The screenshot displays the HealthAI - Intelligent Healthcare Assistant application. On the left, there is a 'Patient Profile' section containing fields for Name, Age (set to 30), Gender (Male), Medical History, and Current Medications. On the right, there is an 'AI-Powered Healthcare Assistant' section. In this section, a user has entered the text 'I am feeling ill' into a text input field. Below this input field is a teal-colored button labeled 'Know what is the problem'. To the right of the input field, there is a large green sidebar. This sidebar contains two numbered sections: '1. Assess the severity of symptoms:' and '2. Home remedies for mild symptoms:'. Under '1. Assess the severity of symptoms:', there are three bullet points: 'Have you experienced fever, chills, or body aches?', 'Are you experiencing shortness of breath or chest pain?', and 'Have you noticed changes in your appetite or sleep patterns?'. Under '2. Home remedies for mild symptoms:', there are two bullet points: 'If you're experiencing a common cold or flu:' followed by 'Stay hydrated by drinking plenty of fluids (water, herbal tea, etc.)' and 'Get enough rest to help your immune system'.



Best free pdf editors | HealthCare_AI/healthai/ | Streamlit - A faster way to build | AI Health Assistant - Streamlit | AI Health Assistant - Streamlit | Team Formation Successful - Streamlit | +

healthcareai-7777.streamlit.app

Share | Manage app

HealthAI - Intelligent Healthcare Assistant

Patient Profile

Name: [Input Field]

Age: 30

Gender: Male

Medical History:

Current Medications:

AI-Powered Healthcare Assistant

Enter symptoms (comma-separated): headache, body pains, nose bleeding

Predict Disease

Top 3 likely diagnoses with estimated likelihoods:

1. Migraine: 70%
2. Tension headache: 20%
3. Sinusitis: 10%

Explanation: Migraine and tension headaches are common causes of headaches and body pains, often accompanied by nosebleeds due to increased blood vessel sensitivity. Sinusitis, while also causing headaches and body pains, is less likely given the absence of other typical symptoms like facial congestion or thick nasal discharge. However, the likelihood of sinusitis can increase if the patient has a history of recurrent sinus infections or if other symptoms suggest a sinus-related issue.

Best free pdf editors | HealthCare_AI/healthai/utils/ai/ | Streamlit - A faster way to build | AI Health Assistant - Streamlit | Team Formation Successful - Streamlit | +

healthcareai-7777.streamlit.app

Share | Manage app

Patient Profile

Name: [Input Field]

Age: 30

Gender: Male

Medical History:

Current Medications:

Allergies:

Save Profile

AI-Powered Healthcare Assistant

Enter diagnosed disease: migraine

Get Treatment Plan

disease: migraine

Treatment Plan:

1. First-line medication: Acute treatment with over-the-counter pain relievers (e.g., ibuprofen, naproxen) or prescription-strength triptans (e.g., sumatriptan, rizatriptan).
 - o For mild to moderate migraines, start with an over-the-counter pain reliever. If symptoms persist or worsen, proceed to triptans.
 - o Triptans should be taken as soon as possible after the onset of migraine symptoms.
2. Supportive care:
 - o Ensure adequate hydration and rest.
 - o Consider using non-pharmacological interventions such as:
 - Cold or warm compress on the head or neck.
 - Ginger or other natural remedies (consult a healthcare provider before use).
 - Relaxation techniques, deep breathing, or biofeedback.
3. When to escalate to a healthcare provider:
 - o If acute medication fails to provide relief within 2 hours.

Best free pdf editors | HealthCare_AI/healthai/utils/ai/ | Streamlit - A faster way to build | AI Health Assistant - Streamlit | Team Formation Successful - Streamlit | +

healthcareai-7777.streamlit.app

Share | Manage app

Health Analytics Dashboard

Raw Data

| Date | HeartRate | SystolicBP | DiastolicBP | BloodGlucose | Symptom |
|--------------|-----------|------------|-------------|--------------|-----------|
| 0 2024-01-01 | 72 | 120 | 80 | 95 | Headache |
| 1 2024-01-02 | 78 | 122 | 82 | 105 | Fatigue |
| 2 2024-01-03 | 85 | 130 | 88 | 110 | Dizziness |
| 3 2024-01-04 | 90 | 135 | 85 | 98 | Headache |
| 4 2024-01-05 | 76 | 118 | 76 | 92 | Nausea |
| 5 2024-01-06 | 70 | 115 | 75 | 89 | Fatigue |
| 6 2024-01-07 | 88 | 140 | 90 | 120 | Dizziness |

Data Summary

Date HeartRate SystolicBP DiastolicBP BloodGlucose Symptom

Upload your health data (CSV or Excel)

Drag and drop file here Limit 200MB per file • CSV, XLSX

Browse files

health_data_with_symptoms.csv 291.08



7. ADVANTAGES & DISADVANTAGES

Advantages:

- **24/7 Accessibility:** Users can access healthcare assistance anytime without waiting for a doctor.
- **AI-Powered Responses:** Quick and intelligent answers using IBM Granite enhance user experience.
- **Early Disease Prediction:** Helps in identifying potential health issues at an early stage.
- **Modular System:** Divided into four independent modules for better organization and usability.
- **User-Friendly Interface:** Built using Streamlit, it provides a simple and intuitive experience.
- **Cost-Effective:** Reduces the need for continuous human supervision in basic healthcare queries.

Disadvantages:

- **Not a Replacement for Doctors:** Cannot replace actual medical consultation or diagnosis.
- **Depends on Internet Connection:** Requires stable internet to function effectively.
- **Limited to Pretrained Knowledge:** IBM Granite model may not always be updated with the latest medical information.
- **Security & Privacy:** Requires strict handling of user data for ethical and legal compliance.

8. CONCLUSION

The HEALTHAI project demonstrates how generative AI, specifically IBM Granite, can be effectively integrated into healthcare applications. By providing intelligent responses to user queries, disease prediction, treatment suggestions, and health analytics, this system can assist users in managing their health proactively. Though it is not a substitute for professional medical advice, it acts as a supportive tool that can bridge the gap between users and healthcare information in real time.

9. FUTURE SCOPE

- Integration with Real Medical Records:** In future, the system can be connected to Electronic Health Records (EHR) for more personalized responses.
- Mobile App Development:** A dedicated mobile version can improve accessibility on smartphones.
- More Advanced AI Models:** Upgrading to future IBM Granite versions or fine-tuning with medical datasets for better accuracy.
- Multi-Language Support:** Expanding to regional languages can make it more inclusive.
- Enhanced Security Measures:** Implementing data encryption and secure login to protect user privacy.
- Doctor Integration:** Providing live chat features with real doctors or teleconsultation options.

Source Code(if any)

```

File Edit Selection View Go Run Terminal Help < - > HealthCare_AI
D:\HEALTHAI\HealthCare_AI\src\main\java\com\smartbridge\healthcare\app\Main.java
1 package com.smartbridge.healthcare.app;
2
3 import java.io.BufferedReader;
4 import java.io.IOException;
5 import java.io.InputStreamReader;
6 import java.net.HttpURLConnection;
7 import java.net.URL;
8 import java.util.List;
9
10 import org.json.JSONObject;
11
12 import com.google.gson.Gson;
13
14 public class Main {
15
16     public static void main(String[] args) throws IOException {
17
18         String url = "https://api.us-east-1.aip.cloud.ibm.com/v1/token";
19         String tokenResponse = sendPostRequest(url);
20         System.out.println("Access token received: " + tokenResponse);
21
22         String predictResponse = predict(tokenResponse);
23         System.out.println("Predicted symptoms: " + predictResponse);
24
25     }
26
27     private static String sendPostRequest(String url) {
28
29         String grant_type = "client_credentials";
30         String client_id = "00000000-0000-0000-0000-000000000000";
31         String client_secret = "00000000-0000-0000-0000-000000000000";
32
33         Map<String, String> headers = new HashMap<>();
34         headers.put("Content-Type", "application/x-www-form-urlencoded");
35
36         String payload = "grant_type=" + grant_type + "&client_id=" + client_id + "&client_secret=" + client_secret;
37
38         URL obj = new URL(url);
39         HttpURLConnection con = (HttpURLConnection) obj.openConnection();
40
41         con.setRequestMethod("POST");
42         con.setDoOutput(true);
43         con.setDoInput(true);
44
45         DataOutputStream wr = new DataOutputStream(con.getOutputStream());
46         wr.writeBytes(payload);
47         wr.flush();
48         wr.close();
49
50         BufferedReader in = new BufferedReader(new InputStreamReader(con.getInputStream()));
51         String inputLine;
52         StringBuffer response = new StringBuffer();
53
54         while ((inputLine = in.readLine()) != null) {
55             response.append(inputLine);
56         }
57         in.close();
58
59         JSONObject json = new JSONObject(response.toString());
60         String access_token = json.getString("access_token");
61
62         return access_token;
63     }
64
65     private static String predict(String token) {
66
67         String url = "https://api.us-east-1.aip.cloud.ibm.com/v1/predict";
68
69         Map<String, String> headers = new HashMap<>();
70         headers.put("Authorization", "Bearer " + token);
71         headers.put("Content-Type", "application/json");
72
73         Gson gson = new Gson();
74
75         Map<String, String> symptoms = new HashMap<>();
76         symptoms.put("headache", "yes");
77         symptoms.put("fever", "no");
78         symptoms.put("cough", "yes");
79
80         String payload = gson.toJson(symptoms);
81
82         URL obj = new URL(url);
83         HttpURLConnection con = (HttpURLConnection) obj.openConnection();
84
85         con.setRequestMethod("POST");
86         con.setDoOutput(true);
87
88         DataOutputStream wr = new DataOutputStream(con.getOutputStream());
89         wr.writeBytes(payload);
90         wr.flush();
91         wr.close();
92
93         BufferedReader in = new BufferedReader(new InputStreamReader(con.getInputStream()));
94         String inputLine;
95         StringBuffer response = new StringBuffer();
96
97         while ((inputLine = in.readLine()) != null) {
98             response.append(inputLine);
99         }
99         in.close();
100
101         JSONObject json = new JSONObject(response.toString());
102         String[] predicted_diseases = json.getString("predicted_diseases").split(",");
103
104         String result = "The most likely diseases based on your symptoms are: ";
105
106         for (String disease : predicted_diseases) {
107             result += disease + ", ";
108         }
109
110         return result;
111     }
112
113     public static void main() {
114
115     }
116 }

```



SMARTBRIDGE

```
File Edit Selection View Go Run Terminal Help ↵ → ⚒ HealthCare_AI
```

HealthCare_AI

```
... app.py x app.py M dashboard
```

```
health > app.py
```

```
1 import streamlit as st
2 from util.ai import get_ai_response, predict_disease, generate_treatment_plan
3 from util.visualization import display_health_analytics
4
5 st.set_page_config(page_title="AI Health Assistant", layout="wide")
6
7 st.markdown("""
8         
28     """, unsafe_allow_html=True)
29
30 st.title("● AI-Powered Healthcare Assistant")
31
32 menu = st.sidebar.selectbox("Navigation", [
33     "Patient Chat",
34     "Disease Prediction",
35     "Treatment Plan",
36     "Health Analytics"
37 ])
38
39 if menu == "Patient Chat":
40     user_input = st.text_input("Enter your health concern:")
41     if st.button("Know what is the problem?"):
42         response = get_ai_response(user_input)
43         result = predict_disease(response)
44         st.success(result)
45
46 elif menu == "Disease Prediction":
47     symptoms = st.text_input("List symptoms (comma-separated):")
48     if st.button("Predict Disease"):
49         results = predict_disease(symptoms)
50         st.info(results)
51
52 elif menu == "Treatment Plan":
53     disease = st.text_input("Enter diagnosed disease:")
54     if st.button("Get Treatment Plan"):
55         plan = generate_treatment_plan(disease)
56         st.write(plan)
57
58 elif menu == "Health Analytics":
59     uploaded_file = st.file_uploader("Upload your health data (CSV or Excel)", type=['csv', 'xlsx'])
60
61     if uploaded_file is not None:
62         try:
```

```
File Edit Selection View Go Run Terminal Help ↵ → ⚒ HealthCare_AI
```

HealthCare_AI

```
... app.py x app.py M dashboard X
```

```
health > app.py x dashboard X display_health_analytics
```

```
1 import streamlit as st
2 import pandas as pd
3 import numpy as np
4 from util.ai import prompt_sample_summary
5
6 def display_health_analytics(df):
7     st.title("Health Analytics Dashboard")
8
9     if df is not None:
10         try:
11             st.subheader("Raw Data")
12             st.dataframe(df, use_container_width=True)
13
14             st.subheader("Data Summary")
15             st.write(df.describe(include='all'))
16
17             numeric_cols = df.select_dtypes(include=['float64', 'int64']).columns.tolist()
18             categorical_cols = df.select_dtypes(include=[object, 'category']).columns.tolist()
19
20             if 'Date' in df.columns and len(numeric_cols) > 0:
21                 st.subheader("Time Series Charts")
22                 for col in numeric_cols:
23                     fig = px.line(df['Date'], y=col, title=f'[col] Over Time')
24                     st.plotly_chart(fig, use_container_width=True)
25
26             if len(numeric_cols) > 0:
27                 st.subheader("Numeric Distributions")
28                 for col in numeric_cols:
29                     fig = px.histogram(df[col], bins=20, title=f'[col] Distribution')
30                     st.plotly_chart(fig, use_container_width=True)
31
32             if len(categorical_cols) > 0:
33                 st.subheader("Categorical Distributions")
34                 for col in categorical_cols:
35                     value_counts = df[col].value_counts().reset_index()
36                     value_counts.columns = [col, 'Count']
37
38                     if len(value_counts) < 10:
39                         fig = px.bar(value_counts, names=col, values='Count', title=f'[col] Distribution')
40                         fig = px.bar(value_counts, x=col, y='Count', title=f'[col] Distribution')
41                         st.plotly_chart(fig, use_container_width=True)
42
43             st.subheader("Single Metric Summary")
44             if st.button("Generate Sample Summary"):
45                 try:
46                     summary = prompt_sample_summary(df)
47                     st.success(summary)
48                 except Exception as e:
49                     st.error(f"Could not generate summary: {e}")
50
51             except Exception as e:
52                 st.error(f"Error displaying health analytics: {e}")
53         else:
54             st.info("Please upload a CSV or Excel file to begin.")
55
56     except Exception as e:
57         st.error(f"Error displaying health analytics: {e}")
```

Dataset Link GitHub & Project Demo Link

Both the dataset and the project demo video are uploaded to the GitHub repository and can be accessed via the following link: https://github.com/KESANI-SANTHOSH-KUMAR/HealthCare_AI