HealthAI — Intelligent Healthcare Assistant

Project Documentation

1.Introduction

- Project tile: HealthAI - Intelligent Healthcare Assistant
- Team members :S. Muthupandi, V.Manoj, G. Madesh, C.Krishna, K.Keerthyvasan

2.project overview

Propose:

The purpose of HealthAI is to provide a safe and intelligent healthcare assistant that helps users with:

- Disease Prediction
- Treatment Plans
- Patient Support Chat
- Health Analytics Dashboard

HealthAI leverages Hugging Face Transformers and PyTorch inside Google Colab to give safe, general healthcare insights.

Future :

Disease Prediction:

Key Point: Early awareness of possible conditions

Functionality:

Accepts patient details (name, age, gender, history, allergies, medications) and current symptoms

Uses an LLM to predict common conditions

Returns structured JSON with:

- Possible conditions
- Red flag symptoms
- General recommendations

Treatment Plan:

Key Point: General care suggestions for common conditions

Functionality:

User enters a medical condition (e.g., Diabetes, Migraine)

AI generates:

- Treatment overview
- Self-care tips
- Lifestyle recommendations
- Signs when to seek urgent care

24/7 Patient Chat Support:

Key Point: Conversational assistance

Functionality:

- Chatbot answers health-related queries in 3–6 short bullet points
- Friendly and safe responses
- Guardrails block unsafe advice (e.g., overdose, unprescribed medicine)
- Always ends with a medical disclaimer

Analytics Dashboard:

Key Point: Health monitoring through data

Functionality:

- Users can upload CSV/Excel files with health metrics (Heart Rate, Blood Pressure, Glucose, etc.)
- Data is visualized using Plotly interactive charts
- Default dataset available for demo
- Helps track weekly health trends

Safety Guardrails:

Key Point: Responsible AI usage

Functionality:

- Detects unsafe or harmful requests
- Blocks responses containing dangerous advice
- Adds disclaimers reminding users to consult healthcare professionals

3. Architecture

Frontend (Gradio UI):

- Interactive web interface built with Gradio
- Tabs for prediction, treatment, chat, and analytics

Backend (Python in Google Colab):

- Hugging Face Transformers for LLM responses
- PyTorch for model inference
- Pandas & Plotly for health data visualization

Model Integration:

- Default Model: ibm-granite/granite-3.2-2b-instruct
- Fallbacks: zephyr-7b-beta, Mistral-7B-Instruct

Guardrails:

- Detects unsafe prompts (e.g., overdose, self-surgery)
- Automatically adds disclaimers

4. Setup Instructions

Prerequisites:

- Python 3.9+
- Google Colab account
- Hugging Face access

Installation (in Colab):

!pip install transformers torch gradio pandas plotly -q

Run the app:

Demo.queue()

Demo.launch(share=True)

5. Folder / Notebook Structure

Healthai.ipynb

│

├— Model Loader (with fallback)

├— Guardrails

├— Disease Prediction

├— Treatment Plan

├— Chatbot

├— Analytics Dashboard

└— Gradio UI

6. Running the Application

1. Open Google Colab.
2. Upload and run healthai.ipynb.
3. Install dependencies.
4. Launch the Gradio interface.
5. Use tabs to test disease prediction, treatment plans, chatbot, and analytics.

7.API Documentation

Backend APIs available include:

POST /prediction/analyze

- Accepts patient profile and symptoms
- Returns possible conditions, red flags, and general recommendations in JSON

POST /treatment/plan

- Accepts patient profile and a medical condition
- Responds with a general treatment overview, lifestyle tips, and when to seek care

POST /chat/ask

- Accepts a user health-related query
- Responds with an AI-generated safe and friendly answer in bullet points

POST /analytics/upload

- Accepts CSV or Excel files containing health vitals (Heart Rate, BP, Glucose, etc.)
- Returns an interactive Plotly chart showing weekly health trends

GET /disclaimer

- Returns the medical disclaimer used in all responses

8. Authentication

Currently, the HealthAI project runs in open mode in Google Colab (any user with the shared Gradio link can access it).

For secure deployments, the following authentication methods can be added:

Token-based Authentication (JWT / API Keys)

- Assigns each user an access token.
- Ensures only authorized users can run predictions or access the chatbot.

OAuth2 / Hugging Face Login

- Uses Hugging Face or Google login for authentication.
- Ideal for cloud deployments (Hugging Face Spaces, Streamlit Cloud, etc.).

Role-Based Access Control (RBAC)

- Different permissions for Patients, Doctors, and Researchers.
- Example: Only doctors can access advanced treatment recommendations.

Session Management

- Tracks user sessions for better monitoring.
- Maintains patient profile data during the session.

Planned Enhancements

- Encrypted storage of uploaded health files.

- Audit logs for chatbot interactions.
- Integration with IBM Cloud Identity for enterprise deployment.

9. User Interface

Sidebar with patient details & model settings

Tabs for:

- 🧪 Disease Prediction
- 💊 Treatment Plan
- 💬 Patient Chat
- 📊 Analytics Dashboard

10. Testing

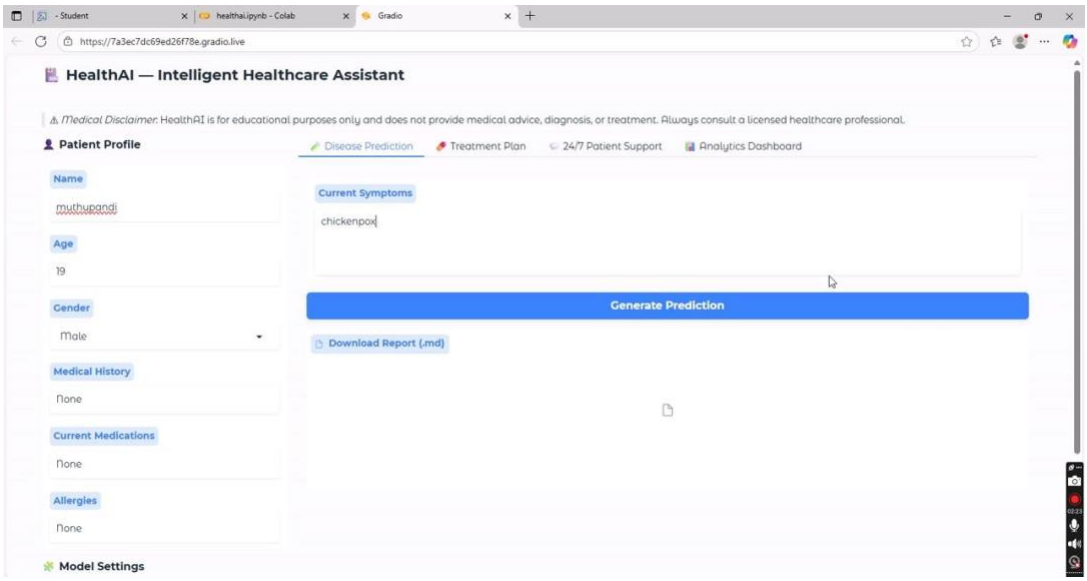
Unit Testing → Prompt builders, JSON parsing

Manual Testing → Chatbot answers, prediction/treatment accuracy

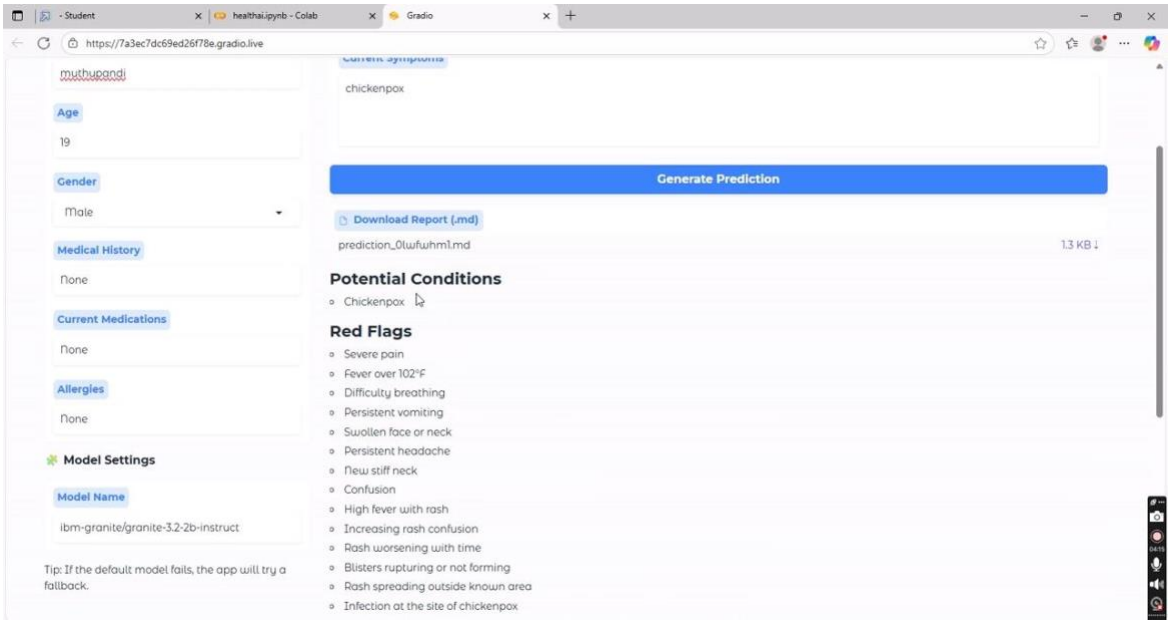
Edge Cases → Empty inputs, invalid CSV files

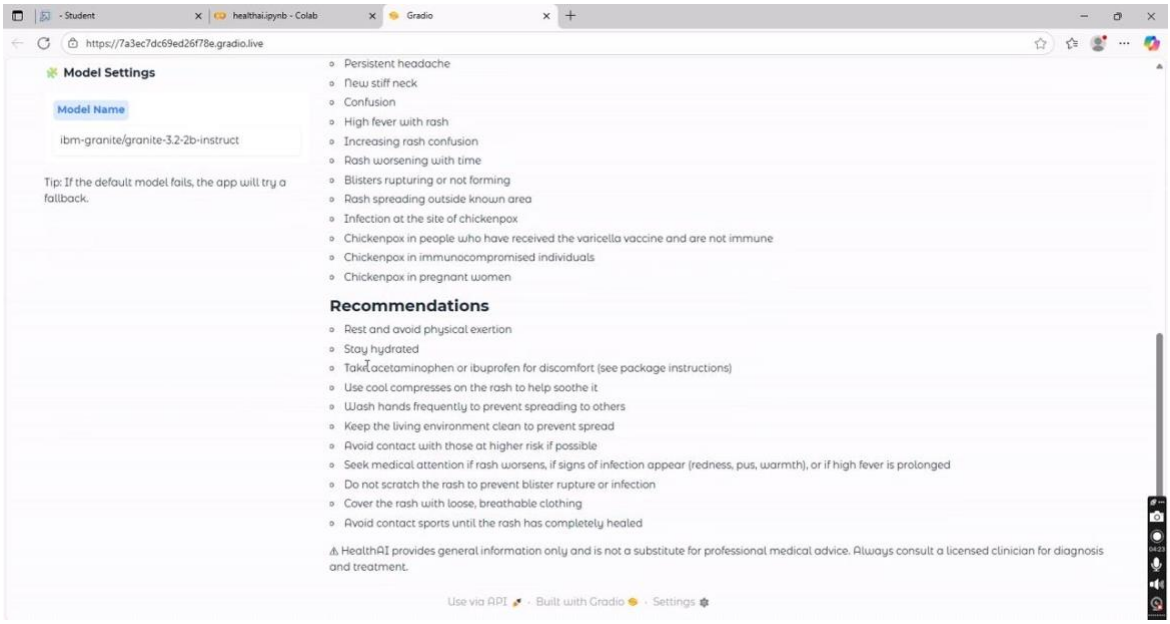
11. Screenshots

Disease Prediction:

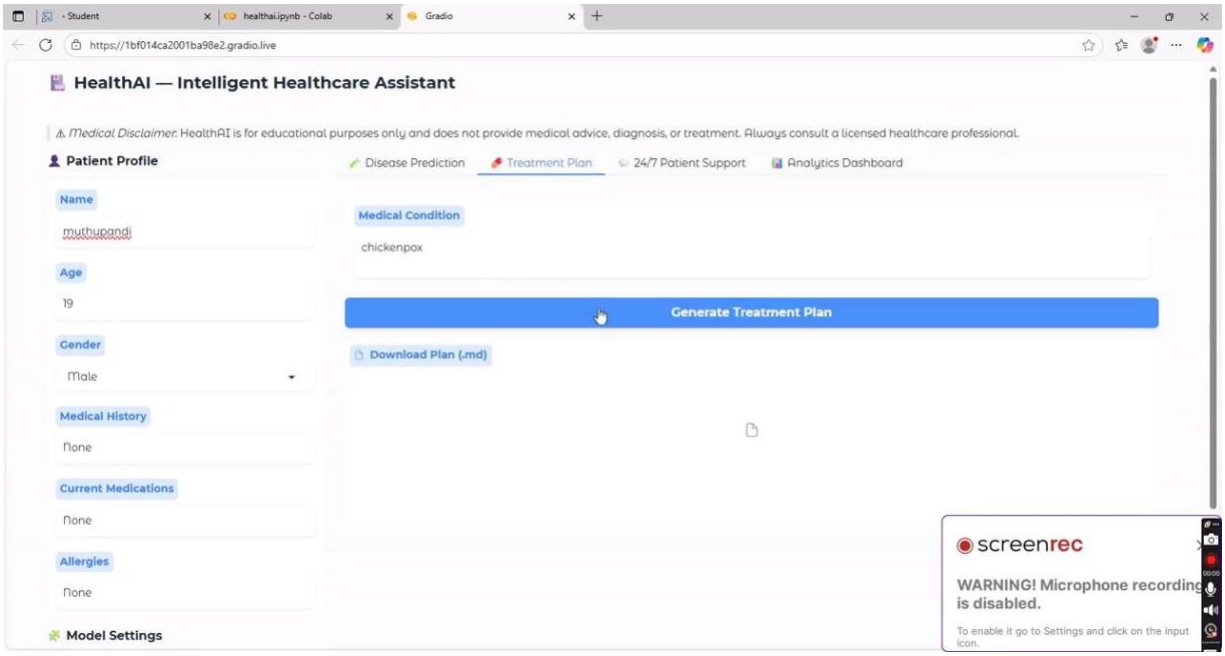


Output :

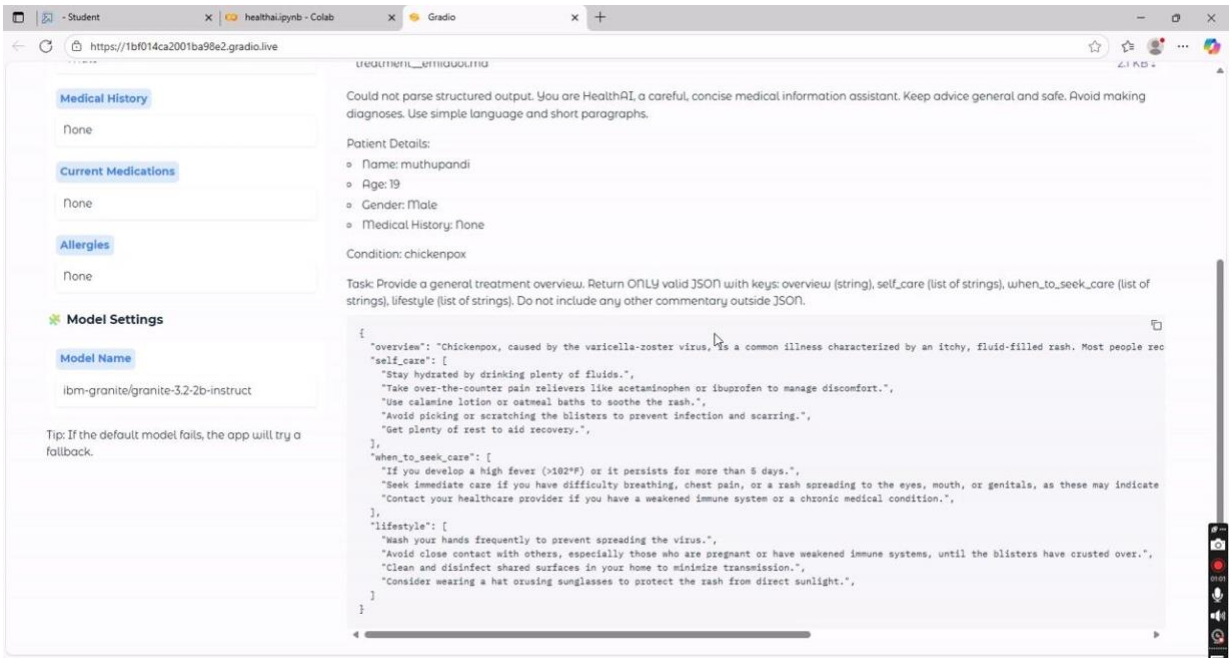

- Student
healthai.ipynb - Colab
Gradio
https://7a3ec7dc69ed26f78e.gradio.live
muthupandi
Age
19
Gender
Male
Medical History
None
Current Medications
None
Allergies
None
Model Settings
Model Name
ibm-granite/granite-3.2-2b-instruct
Tip: If the default model fails, the app will try a fallback.
chickenpox
Generate Prediction
Download Report (.md)
prediction_0lwfwhm1.md
1.3 KB
Potential Conditions
Chickenpox
Red Flags
Severe pain
Fever over 102°F
Difficulty breathing
Persistent vomiting
Swollen face or neck
Persistent headache
New stiff neck
Confusion
High fever with rash
Increasing rash confusion
Rash worsening with time
Blisters rupturing or not forming
Rash spreading outside known area
Infection at the site of chickenpox


- Student
healthai.ipynb - Colab
Gradio
https://7a3ec7dc69ed26f78e.gradio.live
Model Settings
Model Name
ibm-granite/granite-3.2-2b-instruct
Tip: If the default model fails, the app will try a fallback.
Persistent headache
New stiff neck
Confusion
High fever with rash
Increasing rash confusion
Rash worsening with time
Blisters rupturing or not forming
Rash spreading outside known area
Infection at the site of chickenpox
Chickenpox in people who have received the varicella vaccine and are not immune
Chickenpox in immunocompromised individuals
Chickenpox in pregnant women
Recommendations
Rest and avoid physical exertion
Stay hydrated
Take acetaminophen or ibuprofen for discomfort (see package instructions)
Use cool compresses on the rash to help soothe it
Wash hands frequently to prevent spreading to others
Keep the living environment clean to prevent spread
Avoid contact with those at higher risk if possible
Seek medical attention if rash worsens, if signs of infection appear (redness, pus, warmth), or if high fever is prolonged
Do not scratch the rash to prevent blister rupture or infection
Cover the rash with loose, breathable clothing
Avoid contact sports until the rash has completely healed
⚠ HealthAI provides general information only and is not a substitute for professional medical advice. Always consult a licensed clinician for diagnosis and treatment.
Use via API · Built with Gradio · Settings

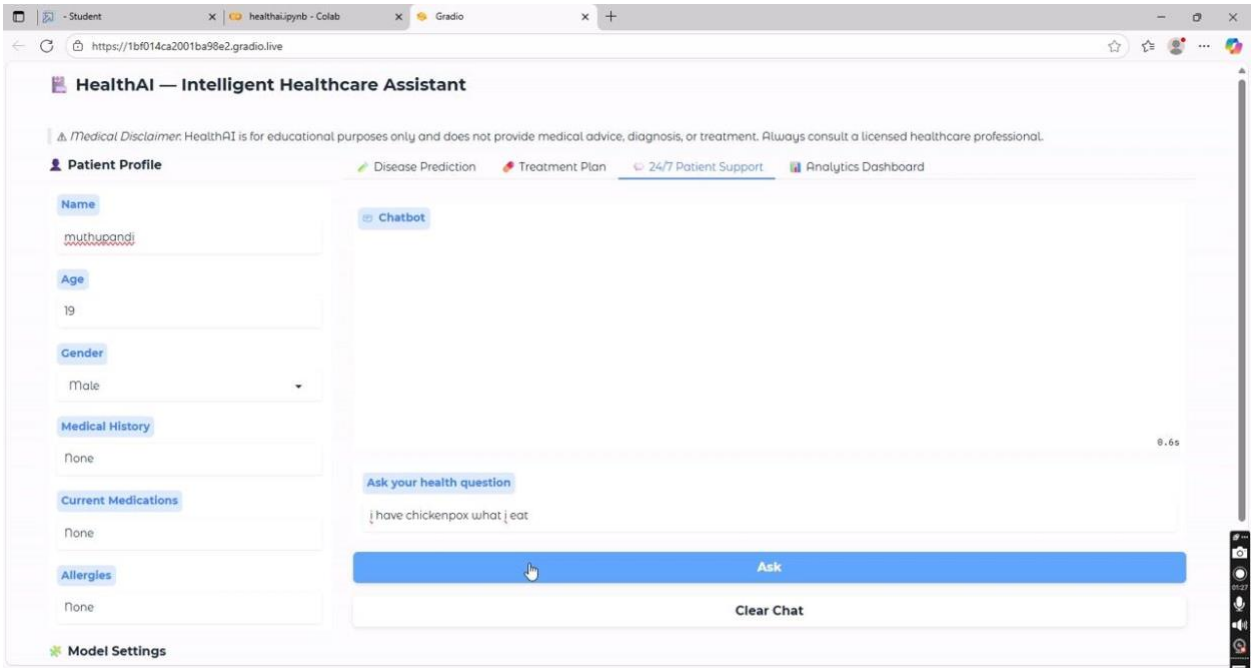
Treatment Plan:



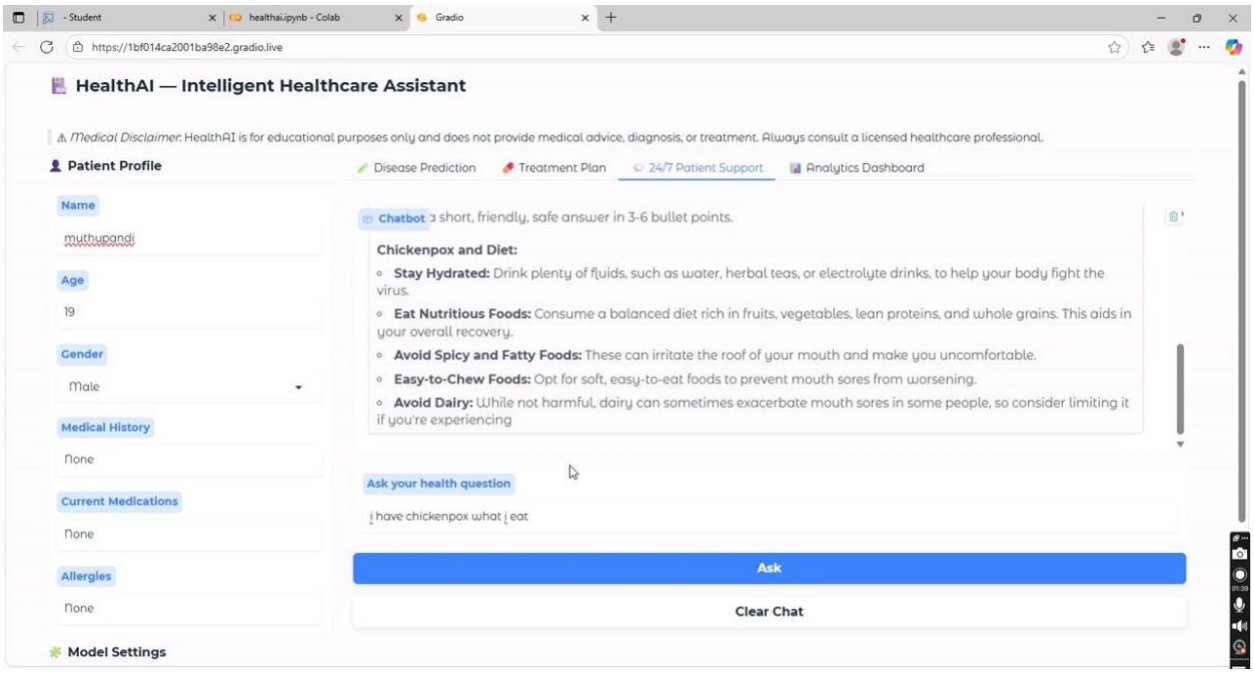
Output:



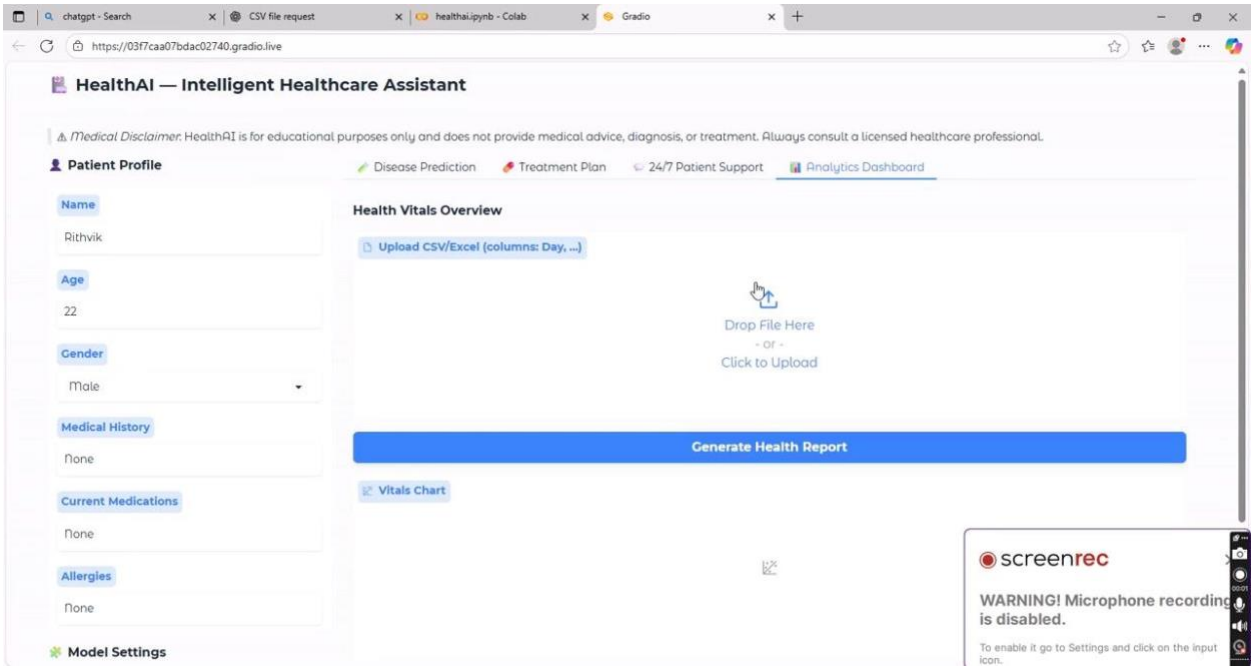
Patient Chat:



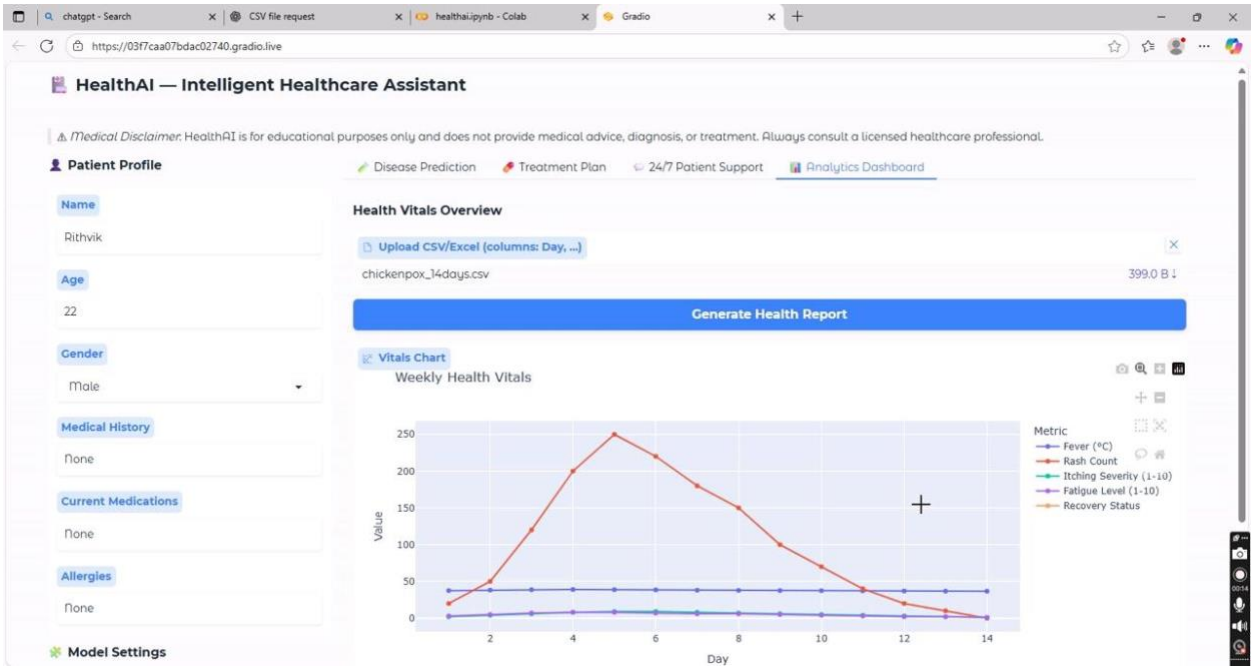
Output:



Analytics Dashboard:



Output:



12. Known Issues

- Model loading may fail if internet is unstable
- Heavy models may cause memory errors on small GPUs
- JSON parsing may fail if LLM outputs malformed text

13. Future Enhancements

- Multi-language support (Tamil, Hindi, etc.)
- Integration with wearable health device data
- Secure login & user history tracking
- Mobile app deployment