

Final Project Submission Instructions

Each project will be carefully evaluated based on the Judging Criteria outlined below. The submission is complete only when all elements in the checklist are submitted in the requested formats. Please follow the instructions carefully; projects with missing or partial elements will be automatically disqualified. Any submissions received after the deadline (Oct 12th, 2025 @ 12:00PM) will be automatically disqualified from judging.

**Judging Criteria:**

* **Innovation & Creativity:** How unique and original is the project, how has the team gone beyond the task, how have they made the project their own?
* **Technical Implementation:** Quality of code, cloud architecture, and data integration.
* **Impact**: Relevance to healthcare and the potential real-world impact, is their an minimal viable prototype available, how well does it address the use case.
* **Presentation**: Clarity of the project demo and documentation.
* **Feasibility & Scalability:** Can this project be scaled up and deployed in a real healthcare setting?

**Code Submission:** All code must be uploaded and submitted via a version control platform (e.g., GitHub) by the deadline, Oct 12th 12:00 PM. Any updates or versions submitted after the deadline timestamp will be automatically disqualified.

**Presentation/Demo:** Teams must submit a brief project presentation or demo in video format (maximum of 4 minutes) explaining the project concept, technology stack, and healthcare impact. Upload the video to YouTube and ensure that appropriate permissions are set for access by the RUHealthHack team.

**Working Prototype:** A functional prototype/software or proof of concept is encouraged. It should demonstrate how cloud technologies and healthcare data are being utilized.

**Documentation:** Please complete the following document with all the requested information and send submit by 12:00 PM on Sunday, October 12, 2025.

**Team #: 32**

**Team Name: Wetrophia**

**Team Lead Name: Zhipeng Jiang**

**Team Lead Phone Number: 937-823-5986**

**Back Up Team Lead Name: Hanzhang Fang**

**Back Up Team Lead Phone Number: 214-606-3273**

**Complete List of Team Members and Phone Numbers**

**Qi Cong 862-280-7468**

**Siqi Jiang 201-469-1910**

**Xuan Zhang 762-763-5601**

**Yisha Zhang 646-460-1168**

**Naidan Zhang 973-979-4265**

**Natalie Kiwanian 551-258-9866**

**Track: Womens**

**Project**:

* Code Submitted
  + Github repository Link:

<https://github.com/MineSelf2016/women_health_agent>

* + Overview of the solution:

The AI OB/GYN Smart Triage System developed by Team Wetrophia is designed to address a critical gap in women’s healthcare - the frequent mis-routing of patients to the wrong specialists, leading to dangerous delays in diagnosis and treatment. Acting as a “GPS for healthcare,” the system intelligently guides each patient to the right provider in real time while maintaining transparency and auditability.

The platform enables patients to describe their symptoms through an AI-powered chat or phone interface. Using natural language understanding (OpenAI GPT-4) and severity assessment algorithms, the system determines the urgency of each case and matches it to one of six OB/GYN subspecialties, including general OB/GYN, maternal-fetal medicine, urogynecology, reproductive endocrinology, gynecologic oncology, and complex/minimally invasive surgery. If life-threatening conditions are detected, it immediately triggers an emergency alert (“Call 911”) response.

Under the hood, the solution integrates three functional layers:

1. Red-Flag Detection: Identifies 11 high-risk symptoms (e.g., bleeding, seizure, chest pain) with 95% accuracy.
2. Specialty Matching Engine: Evaluates symptom severity and confidence level to recommend the most appropriate specialist and next available appointment.
3. Audit Trail & Oversight: Records every AI decision, enabling physicians to review, override, and justify modifications, ensuring human oversight and continual system learning.

The system’s architecture is HIPAA-compliant, featuring end-to-end encryption, secure data transmission, and full audit logging for accountability. Future development aims to expand training datasets with diverse patient populations, integrate with Epic EMR, and extend functionality to other medical specialties and telehealth platforms, ultimately transforming how hospitals triage and route patients.

* + Details on cloud architecture and tools used

Twilio Voice API for patient calls

Multi-AI-Agent-based triage reasoning Chatbot (OpenAI GPT 4o-mini)

FastAPI for deployment and integration

Steamlit Web App Setup

* + Steps for installation or deployment

**System Architecture**

Twilio (voice input/output)

Streamlit (text chat UI)

TriageAgent (core logic)

├── Slot extraction & next-question logic

├── RAG retrieval from obgyn\_index/

├── LLM summarization & confirmation

└── Reference display (page numbers + snippets)

**Project Structure**

projects/

├── obgyn\_index/ # FAISS vector store for textbook embeddings

│ ├── index.faiss

│ └── index.pkl

├── app\_chat.py # Streamlit chat interface

├── main.py # Twilio voice interface (FastAPI + WebSocket)

├── schedule\_loader.py # Utility to read doctor schedule Excel

├── triage\_agent.py # Core multi-agent logic (slots + RAG + LLM)

├── .env # Contains OPENAI\_API\_KEY

└── README.md

**Quick Start**

**Install dependencies**

pip install streamlit openai langchain faiss-cpu python-dotenv twilio

**Prepare your .env**

Create a file named .env in the root folder:

OPENAI\_API\_KEY=sk-xxxxxxxxxxxxxxxx

**Launch the Streamlit App**

streamlit run app\_chat.py

Then open the displayed local URL (e.g., http://localhost:8501).

* Demo Presentation uploaded to YouTube with appropriate settings
  + YouTube Link: https://youtu.be/GsCxi1k11Tg?si=tD9cfDzOB9qjJmv-