DOCTORY AI

A Multi-Modal AI System for Preliminary Disease Diagnosis

Digital Egypt Pioneers Initiative (DEPI)

Graduation Project Documentation

Submitted to DEPI Program

December 6, 2025

Contents

1 Project Planning

1.1 Project Motivation

Access to fast, reliable, and understandable medical guidance is limited for a large number of people, especially in remote areas or regions with weak healthcare infrastructure.

Most AI-based diagnostic tools currently available provide raw technical outputs (such as probability scores or heatmaps) that are difficult for the average patient to interpret.

This project aims to bridge that gap by combining Computer Vision, Machine Learning, and Large Language Models (LLMs) to create an intelligent, human-centered medical assistant that acts as a bridge between complex medical data and the patient.

1.2 Project Scope

DOCTORY AI is a unified AI-driven medical diagnostic platform designed to assist users in identifying potential diseases through various input methods, including medical images, clinical measurements, and textual symptom descriptions.

The scope specifically covers four critical medical domains:

1. **Pneumonia Detection:** Using chest X-ray images to identify

lung opacity.

2. **Malaria Detection:** Analyzing blood smear microscopy images for parasitic infection.

3. **Diabetes Risk Prediction:** Using tabular diagnostic values (glucose, BMI, insulin levels).

4. **Heart Disease Risk Analysis:** Evaluating lifestyle and medical indicators (age, cholesterol, blood pressure).

It is important to note that the system output is always a **preliminary** result and is never intended to replace professional medical advice or doctors.

1.3 Project Objectives

The primary objectives of this project are:

- Build accurate AI diagnostic models for the four specified diseases.
- Provide an intuitive, accessible, and simple user interface for non-technical users.
- Integrate Large Language Models to translate technical predictions into empathetic, human-friendly medical explanations.

- Ensure data privacy by following a strictly stateless system design.

- Design a scalable backend architecture capable of supporting future cloud deployment.

1.4 Development Methodology

The project follows the **Agile Development** methodology due to its flexibility and the need for iterative validation of AI models:

- **Sprint 1:** Requirement analysis, dataset acquisition, and feasibility study.
- **Sprint 2:** Model development, training, validation, and hyperparameter optimization.
- **Sprint 3:** Backend API development using Flask and ONNX Runtime for efficient inference.
- **Sprint 4:** Frontend integration, UI design, and LLM API communication setup.
- **Sprint 5:** System testing, user evaluation, bug fixing, and final documentation.

1.5 Tools & Technologies

The following technologies were selected to ensure performance and scalability:

Category	Technologies Used
Programming	Python, HTML, CSS, JavaScript
ML Frameworks	TensorFlow, PyTorch, Scikit-learn
Computer Vision	Ultralytics YOLOv11, VGG16
Backend	Flask, Flask-CORS
LLM Integration	API-based LLM Service
Version Control	GitHub

Table 1: Technology Stack

1.6 Team Roles & Responsibilities

The development of DOCTORY AI is executed by a dedicated team. Each member holds specific responsibilities to ensure the successful delivery of the project components.

Team Member	**Key Responsibilities**
Jasmine	Data acquisition, Developing Diabetes Model & Backend
Basmala	UI/UX Implementation, Documentation, & Presentation
Arwa	Team Leadership, LLM Integration & Deployment
Aya	Developing Pneumonia Detection Model & Documentation
Esraa	Developing Malaria Detection Model & Documentation
Haneen	Developing Heart Disease Risk Model & Documentation

Table 2: Team Members and Assigned Roles

2 Stakeholder Analysis

Stakeholders are individuals, groups, or entities that can affect or are affected by the system's objectives. Understanding their specific interests ensures better system alignment and user satisfaction.

Stakeholder	**Role & Interest**	**Influence**
End Users	Individuals seeking fast, confidential, and reliable preliminary medical information.	High
Development Team	The group responsible for system design, model training, and integration.	High
Academic Evaluators	Assessors judging the technical quality, documentation, and innovation level.	High
Healthcare Professionals	Doctors providing indirect validation and future clinical input or oversight.	Medium

Table 3: Stakeholder Matrix

3 System Architecture and Data Handling

3.1 Overall Architecture

The system follows a modern client–server architecture designed for modularity and scalability. The interaction between the Frontend (User Interface), the Flask Backend, the AI Inference Engine, and the External LLM Service is illustrated in Figure 1.

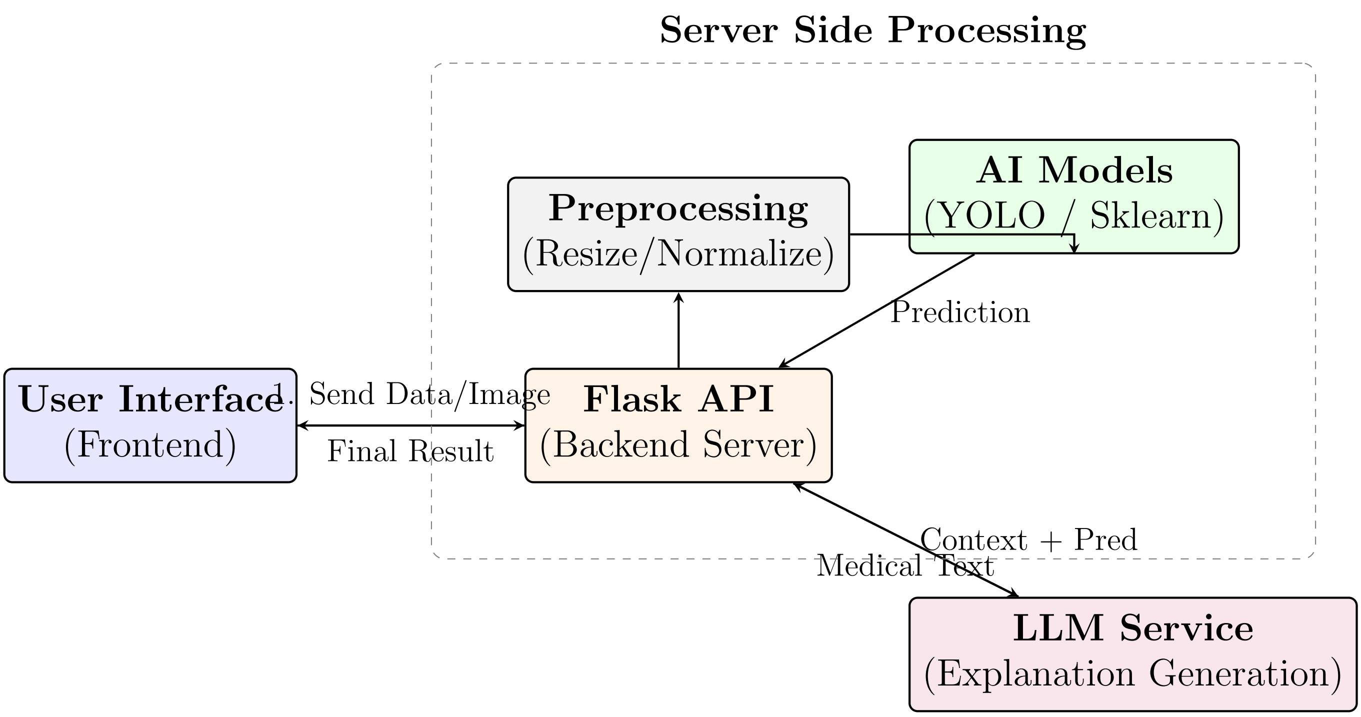

Figure 1: High-Level System Architecture Diagram showing the data flow between Client, Backend, AI Models, and LLM.

The architecture consists of three main layers:

- **Presentation Layer:** Handles user inputs and visualizes results.

- **Application Logic Layer:** A Flask-based server that routes requests and manages data preprocessing.

- **Intelligence Layer:** Contains the deep learning models for diagnosis and the interface with the Large Language Model for explanations.

3.2 Stateless Design

DOCTORY AI is implemented using a strictly **stateless approach**. This means the system does not store any sensitive user data, medical images, or medical history in a persistent database. Each request is processed independently in real-time. This design choice maximizes user privacy and significantly reduces cybersecurity risks associated with data breaches.

3.3 API-Based Data Flow

The data lifecycle in the application follows these steps:

1. The user submits medical data or uploads an image via the frontend.

2. The backend receives the request and performs necessary preprocessing (resizing, normalization).

3. The specific AI model performs inference on the processed data.
4. The mathematical prediction is forwarded to the Large Language Model (LLM).
5. The LLM generates a readable, context-aware medical explanation.
6. The final result is sent back to the user interface.

3.4 Security and Privacy Considerations

To ensure user safety:

- **No Authentication:** No user login or registration is required to use the tool.
- **No Storage:** No permanent data storage of user inputs.
- **Encrypted Traffic:** All communication is secured via HTTPS protocols.

4 UI/UX Design

4.1 Design Philosophy

The interface follows a "Patient-First" philosophy. It prioritizes simplicity, high accessibility, and emotional comfort. Medical terminology is simplified for the general public without distorting the scientific meaning.

4.2 User Journey

The typical user flow is designed to be linear and friction-free:

1. User opens the web application.
2. User selects the desired diagnostic module (e.g., Pneumonia).
3. User uploads an image or enters the required medical data.
4. System processes data and returns an instant AI prediction.
5. User receives an empathetic explanation via the LLM.

4.3 Color Psychology

The color palette was chosen to evoke specific feelings:

- **Blue:** Represents trust, safety, and medical professionalism.
- **Green:** Indicates a healthy condition and positive feedback.
- **Red:** Used sparingly for alerts and high-risk warnings to alert the user without causing panic.

4.4 Conceptual Wireframes

Home Screen Layout

Structure Description:

- A welcoming header with the system name "DOCTORY AI".
- A brief medical disclaimer indicating that the system provides preliminary diagnosis only.
- **Four main diagnostic buttons:**
  - Pneumonia Detection
  - Malaria Detection
  - Diabetes Risk Prediction
  - Heart Disease Risk Prediction
- A navigation button to access the general medical chatbot.

The layout follows a clean grid design with clear icons for each diagnostic module to ensure ease of use for non-technical users.

Figure 2: Home Screen Layout Concept

Diagnostic Input Screen

Structure Description:

- An upload area for medical images (X-ray or blood smear) for image-based diagnosis.
- Input fields such as sliders, dropdowns, and numeric fields for tabular medical data.
- A central **"Analyze"** button to submit the data to the AI model.
- **Results Panel:**
  - The predicted medical condition.
  - The confidence level of the prediction.
  - A natural-language explanation generated by the LLM.

The screen is designed with minimal distractions to allow the user to focus only on the medical input and result.

Figure 3: Diagnostic Input Screen Layout Concept

5 Risk Analysis

Identifying potential risks is crucial for deployment:

- **Model Bias:** Potential for bias if the training dataset is unbalanced regarding age, gender, or ethnicity.
- **Over-reliance:** Users might treat the AI suggestion as a final medical verdict, ignoring professional advice.
- **Service Interruptions:** Dependency on third-party APIs (like LLM providers) could lead to downtime.
- **Ethical Risks:** The risk of false negatives (telling a sick patient they are healthy) is a critical concern.

6 Ethical Considerations

- **Transparency:** The system clearly states it is a preliminary diagnostic tool on every screen.
- **Privacy:** No personal data is stored, preserving user anonymity.
- **Explainability:** We prioritize providing reasons for the AI's decision rather than a "black box" output.
- **Medical Disclaimer:** Strong encouragement is provided to

consult professional doctors for confirmation.