AI-Powered Triage System
System Design Document (SDD)

Group 17
Esmeralda Amado
Jessica Parker
Jonathan Su
Nneoma Ajawara
Yvan Michel Kemsseu Yobeu

December 2025

Contents

Version	Description	Date Added
0.1	Initial SDD created for Start Snapshot and planned system architecture.	11/30/2025
0.2	Added UI design and database overview.	12/01/2025
1.0	Final version for Start Snapshot submission.	12/05/2025

Version Description

1 Introduction

1.1 Purpose

The purpose of this Software Design Document (SDD) is to outline the technical design and system structure for the AI-Powered Triage System. This version covers the Start Snapshot, representing the foundation of the system.

1.2 Intended Audience

This document is intended for:

- The CS 3338 course instructor and graders.
- Developers responsible for implementing system components.
- Testers working with TestRail and Jira.
- Future maintainers/contributors of the system, in later snapshots.

1.3 System Overview

The AI-Powered Triage System is a web-based healthcare tool that allows authorized medical staff to:

- Log in securely.
- Access patient demographic and historical visit information.
- Receive AI-generated summaries and triage recommendations.
- Complete medical visit documentation assisted by AI.

This SDD describes the structure and behavior of the sytem and all major components.

2 System Architecture

2.1 Workflow

The workflow of the AI Triage System is as follows:

1. The user logs in through the secure authentication portal.
2. The front-end client requests patient data from the back-end API.
3. Data is retrieved from the database and displayed to the user.
4. When the user interacts with the AI, the back-end sends relevant patient information to the AI model.
5. The AI returns a summary or triage recommendation.
6. The user updates the patient's visit history or completes forms.
7. The system stores updated visit documentation in the database.

2.2 Component Breakdown

The AI Triage System will consist of the following components:

- **Client-Side (Front-End)** A web interface built using HTML/CSS/JavaScript or a front-end framework like React.
- **Server-Side (Back-End)** A REST API built using Python Flask, FastAPI, or Node.js for:
  - User authentication
  - Database interactions
  - Passing data to/from the AI model
- **Database Layer** Will store patient demographic information and historic visit notes using a relational database (MySQL).
- **AI Service Layer** Connects to an AI model such as OpenAI GPT or a local machine learning model trained on healthcare text.

page

3 User Interface

3.1 How to Use

Users will be able to interact with the system, using the following components:

- **Main UI Pages**: Verified healthcare staff can authenticate themselves through the Login Page. They can also view patients' demographics and visit history through the Patient Profile Page.

- **Dashboard**: To display list of patients, recent activity, and quick links.
- **AI Chat Interface**: Allows staff to ask triage-related questions.
- **Visit Documentation Form**: A structured form for healthcare staff to complete patient encouter records.

3.2 Database Design Overview

Database will contain the following tables:

- **Staff** Columns: staff_id, name, email, role, password_hash
- **Patients** Columns: patient_id, name, age, gender, date_of_birth
- **Visits** Columns: visit_id, patient_id, visit_date, condition, treatment, outcome

3.3 UI Screenshots

Screenshots of tentative UI design may be added in later versions.

4 Glossary

AI - Artificial Intelligence

API - Application Programming Interface

DB - Database

SDD - System Design Document

UI - User Interface

5 References

Current references:

- CS 3338 Lab 14 Instructions (Canvas).
- CS 3338 Final Project Instructions (Canvas).
- Example Final Projects From Past Group (Github).
- Overleaf Documentation: `https://www.overleaf.com/learn`
- HIPAA Overview: `https://www.hhs.gov/hipaa/`