|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Software Design Specifications**  ***CureBridge***  **Version: 01.00**   |  |  | | --- | --- | | Project Code | F24-92 | | Supervisor | *Nida Munawar* | | Co-Supervisor | *Dr Nauman Durrani* | | Project Team | *Kuljeet Khatri(21K-3362)*  *Syed Muhammad Zaid(21K-3348)*  *Hassan Abbas(21K-3286)* | | Submission Date | 17/12/2024 | |     **Document History**   |  |  |  |  | | --- | --- | --- | --- | | Version | Name of Person | Date | Description of change | | - | - | - | - | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  |     **Distribution List**   |  |  | | --- | --- | | **Name** | **Role** | | *Nida Munawar* | Supervisor | | *Dr Nauman Durrani* | Co-Supervisor | |  |  |     **Document Sign-Off**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Version** | **Sign-off Authority** | **Project Role** | **Signature** | **Sign-off Date** | | 1.0.0 | Mam Nida Munawar | Supervisor |  | 15/05/2025 | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |  |  |  |  |  | |

**Document Information**

| **Category** | **Information** |
| --- | --- |
| Customer | FAST-NU |
| Project | CureBridge |
| Document | Software Design Specification |
| Document Version | 1.0 |
| Status | Draft |
| Author(s) | Kuljeet Khatri  Syed Muhammad Zaid  Hassan Abbas |
| Approver(s) | Nida Munawar |
| Issue Date | 15/05/2025 |
| Document Location | - |
| Distribution | Advisor  Project Coordinator’s Office (through Advisor) |

**Definition of Terms, Acronyms and Abbreviations**

| **Term** | **Description** |
| --- | --- |
| ASP | Active Server Pages |
| DD | Design Specification |

|  |  |
| --- | --- |
| HOD | Head of Department |

|  |  |
| --- | --- |
| AI | Artificial Intelligence |
|  |  |
|  |  |

**Table of Contents**

[**1**](#_37m2jsg) **Introduction *6***

[*1.1*](#_4d34og8) *Purpose of Document 6*

[*1.2*](#_1mrcu09) *Intended Audience 6*

[*1.3*](#_46r0co2) *Document Convention 6*

[*1.4*](#_2lwamvv) *Project Overview 6*

[*1.5*](#_111kx3o) *Scope 7*

[**2**](#_3l18frh) **Design Considerations *8***

[*2.1*](#_206ipza) *Assumptions and Dependencies 8*

[*2.2*](#_4k668n3) *Risks and Volatile Areas 8*

[**3**](#_qsh70q) **System Architecture *9***

[*3.1*](#_2zbgiuw) *System Level Architecture 9*

[*3.2*](#_1pxezwc) *Software Architecture 10*

[**4**](#_1egqt2p) **Design Strategy *11***

[**5**](#_3ygebqi) **Detailed System Design 1*3***

[*5.1*](#_2dlolyb) *Database Design 13*

[5.1.1](#_147n2zr) Scheme Diagram *13*

[5.1.2](#_sqyw64) Data Dictionary *14*

[5.1.2.1](#_3cqmetx) Data 1 *14*

[5.1.2.2](#_1rvwp1q) Data 2 *15*

[5.1.2.3](#_3o7alnk) Data 3 *16*

[5.1.2.4](#_3o7alnk) Data 4 *17*

[*5.2*](#_23ckvvd) *Application Design* ***18***

[5.2.1](#_ihv636) Sequence Diagram *18*

[5.2.1.1](#_32hioqz) <Sequence Diagram 1> *18*

[5.2.1.2](#_1hmsyys) <Sequence Diagram 2> *19*

[5.2.1.3](#_2grqrue) <Sequence Diagram 3> *20*

[5.2.2](#_vx1227) State Diagram ***21***

[5.2.2.1](#_3fwokq0) <State Diagram 1> *21*

[5.2.2.2](#_4f1mdlm) <State Diagram 2> *22*

[5.2.2.3](#_4bvk7pj) <State Diagram 3> *23*

[**6**](#_3tbugp1) **References *24***

[**7**](#_nmf14n) **Appendices *25***

# Introduction

## Purpose of Document

*This Software Requirements Specification (SRS) document provides a detailed and structured outline of the performance management system's functional and non-functional requirements. It is a comprehensive guide for all stakeholders, including business users, developers, and technical teams, to ensure a shared understanding of the system’s goals, functionalities, and constraints.*

* *To provide a clear communication framework among stakeholders*
* *To act as a reference document for developers during the system’s design, implementation, and testing phases.*
* *To ensure that the system’s design and implementation align with the business objectives*
* *To establish a foundation for requirement traceability throughout the project lifecycle*

## Intended Audience

* *Fast NU*
* *Jury*
* *Supervisor (Nida Munawar)*
* *Students of Fast NU*
* *Potential Users of this product*
* *Our Team*

## Document Convention

* *Font Family = Arial*
* *Font Size = 12 for headings, 10 for the rest of the content*

## Project Overview

*CureBridge is a healthcare application designed to bridge critical gaps in telehealth services and traditional medical care using AI and advanced web technologies. The system aims to enhance healthcare delivery efficiency, accuracy, and professionalism through innovative features like real-time transcription, AI-driven diagnostics, and sentiment analysis. It provides a centralized platform for patients, doctors, and administrators to streamline medical consultations and improve the overall quality of care.*

***Included Functionalities:***

* ***Patient and Doctor Access to Data:*** *Seamless access for patients and doctors to consultation records, real-time transcriptions, and patient medical histories.*
* ***AI-Driven Diagnosis Assistance:*** *Real-time extraction of symptoms and disease prediction based on live conversations.*
* ***Real-Time Transcription:*** *Automatic transcription of doctor-patient consultations stored securely for future reference.*
* ***Sentiment Analysis:*** *Detection of toxic behavior or unprofessional interactions during consultations to ensure high-quality care.*
* ***Centralized Record Management:*** *Consolidate patient consultation records, including transcriptions, diagnoses, and treatment histories.*
* ***Admin Oversight and Quality Control:*** *Tools for administrators to audit flagged consultations and manage user permissions for a professional environment.*

***Excluded Functionalities:***

* *Support for physical diagnostic tests or manual analysis of reports.*

## Scope

***Included Functionalities:***

* ***Patient and Doctor Access to Data:*** *Both patients and doctors can access consultation records, medical histories, and real-time transcriptions.*
* ***AI-Driven Diagnosis Assistance:*** *Real-time extraction of symptoms and disease prediction to support doctors in making informed decisions.*
* ***Real-Time Transcription:*** *Automatic transcription of doctor-patient conversations, securely stored for easy retrieval.*
* ***Sentiment Analysis:*** *Monitoring consultations for toxic or unprofessional interactions to maintain service quality and professionalism.*
* ***Centralized Record Management:*** *A single repository for consultation records, including transcripts and diagnoses, ensuring continuity of care.*
* ***Admin Oversight and Quality Control:*** *Admins can audit flagged consultations, manage permissions, and ensure adherence to professional standards.*

***Excluded Functionalities:***

* *Integration of physical medical tests or manual verification of patient-reported symptoms.*

# Design Considerations

*The design of CureBridge involves addressing several critical issues to ensure the system meets its objectives and functions seamlessly. These considerations lay the foundation for a robust and scalable solution:*

## Assumptions and Dependencies

* ***Technology Stack:*** *The platform assumes the availability of stable and compatible APIs for video calls and Deepgram for transcription services.*
* ***Data Availability:*** *It relies on structured and unstructured medical data for training and implementing AI models for symptom extraction and disease prediction.*
* ***User Base:*** *The system is designed assuming users have basic technical literacy to interact with the application.*

## Risks and Volatile Areas

# *Data Privacy and Security: Ensuring patient data confidentiality and adherence to healthcare compliance regulations (e.g., HIPAA).*

# *Real-Time Performance: Managing the computational load of real-time transcription, sentiment analysis, and AI-powered diagnostics.*

# *AI Model Generalization: Ensuring AI models generalize well across diverse medical datasets and scenarios.*

# *System Integration: Smooth integration of APIs and backend AI models with the web application interface.*

# System Architecture

*[This section should provide a high-level overview of how the functionality and responsibilities of the system are partitioned and then assigned to subsystems or components. The main purpose is to understand how the system is decomposed and how the individual parts work together to provide the desired functionality].*

## System Level Architecture

*CureBridge is designed as a modular system where functionalities are partitioned into distinct subsystems to ensure clarity, scalability, and maintainability. The top-level architecture includes:*

#### *System Decomposition into Elements*

1. ***User Subsystem:***
   * *Comprises patient, doctor, and admin portals for interaction.*
   * *Provides tailored functionalities such as appointment booking, consultation management, and audit tools.*
2. ***AI Processing Subsystem:***
   * *Handles real-time transcription, symptom extraction, disease prediction, and sentiment analysis.*
   * *Includes pre-trained models like BERT and Random Forest for processing tasks.*
3. ***Communication Subsystem:***
   * *Facilitates video and audio communication using Calling API.*
   * *Ensures secure and real-time communication during consultations.*
4. ***Data Management Subsystem:***
   * *Manages centralized storage of transcriptions, consultation records, and user data in an Azure MySQL database.*
5. ***Integration Layer:***
   * *Acts as a bridge between the frontend and backend, facilitating seamless communication between user actions and the AI models or database.*

#### *Relationships Between Elements*

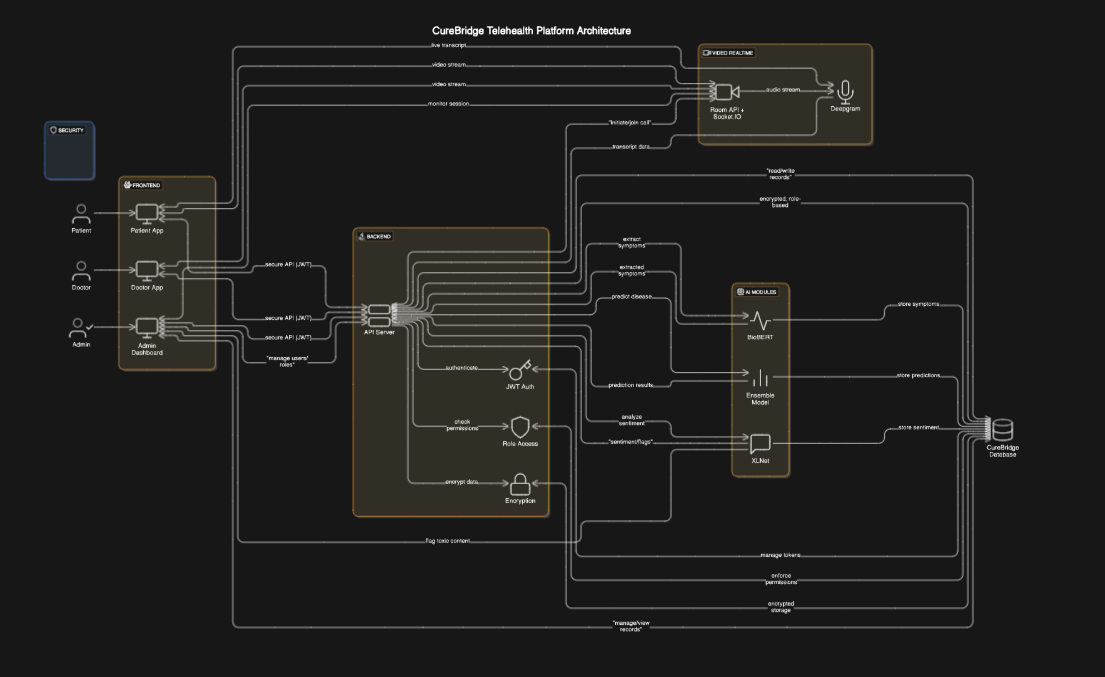
* *The* ***user subsystem*** *communicates with the* ***AI processing subsystem*** *to fetch real-time insights during consultations.*
* *The* ***data management subsystem*** *interacts with the* ***communication subsystem*** *to securely store conversation transcriptions.*
* *The* ***integration layer*** *ensures smooth data flow across all components.*

#### *Interfaces to External Systems*

* ***API:*** *For video/audio communication.*
* ***Deepgram API:*** *For real-time transcription.*
* ***Cloud Storage (Azure):*** *For secure data management and accessibility.*

#### *Global Design Strategies*

* ***Error Handling:*** *Centralized error logging and reporting mechanisms to capture API failures, transcription errors, or model processing issues.*
* ***Security:*** *Adherence to healthcare data privacy regulations with encrypted storage and secure API calls.*

****

## Software Architecture

*The software architecture divides the system into layers to streamline functionality and interactions:*

#### *Layers of Architecture*

1. ***User Interface Layer:***
   * *Built with Angular or React for a responsive, user-friendly front end.*
   * *Provides distinct portals for patients, doctors, and admins, each with role-specific features.*
2. ***Middle Tier (Application Layer):***
   * *Handles business logic and API integrations.*
   * *Includes services for communication, transcription, AI processing, and auditing.*
   * *Developed using Spring Boot for backend logic.*
3. ***Data Access Layer:***
   * *Manages interaction with the database (Azure MySQL) for storing and retrieving user and consultation data.*
   * *Ensures secure access to sensitive information.*

#### *Interactions Between Layers*

* *The* ***User Interface Layer*** *sends user inputs to the* ***Middle Tier****, which processes the requests and interacts with the* ***Data Access Layer*** *for data retrieval or updates.*
* *AI models within the* ***Middle Tier*** *process real-time transcription and provide outputs to the* ***User Interface Layer*** *for display.*

# Design Strategy

*The design strategy for CureBridge is focused on modularity, scalability, and maintainability, ensuring the platform's architecture can accommodate current requirements while being flexible for future enhancements. Key design considerations include the following:*

### *1. Future System Extension or Enhancement*

* ***Modular Architecture:*** *The system is designed with distinct modules for AI processing, user management, and data handling, enabling seamless addition or modification of features without disrupting existing functionality.*
* ***Scalability:*** *Cloud-based storage (Azure) and APIs ensure the system can scale with increasing user demands and additional functionalities, such as integration with new APIs for advanced diagnostics.*
* ***Adaptability:*** *The use of flexible AI models (e.g., BERT and Random Forest) allows for quick updates or model replacements as new techniques emerge.*

### *2. System Reuse*

* ***Reusability of Components:***
  + *The middleware and APIs for communication and transcription can be reused across other telehealth platforms or healthcare applications.*
  + *The AI modules (e.g., sentiment analysis, and symptom extraction) are designed as standalone components, making them reusable for various healthcare-related projects.*
* ***Service-Oriented Approach:*** *The design follows a service-oriented approach, where individual services can be reused or integrated into other systems.*

### *3. User Interface Paradigms*

* ***Role-Based Dashboards:*** *Intuitive and role-specific dashboards are designed for patients, doctors, and administrators, enhancing user experience and minimizing cognitive load.*
* ***Responsive Design:*** *Built using Angular/React frameworks, the interface adapts to different devices, including desktops, tablets, and mobile devices.*
* ***Accessibility:*** *Includes features like live transcription and clear visual layouts to accommodate users with disabilities or limited technical literacy.*

### *4. Data Management (Storage, Distribution, Persistence)*

* ***Centralized Data Storage:*** *Azure MySQL serves as the primary database, storing consultation records, transcription data, and user information securely.*
* ***Data Distribution:*** *Secure APIs ensure consistent data access across the patient, doctor, and admin portals.*
* ***Persistence:*** *Redundancy mechanisms are in place to avoid data loss, with regular backups and fail-safe recovery procedures.*

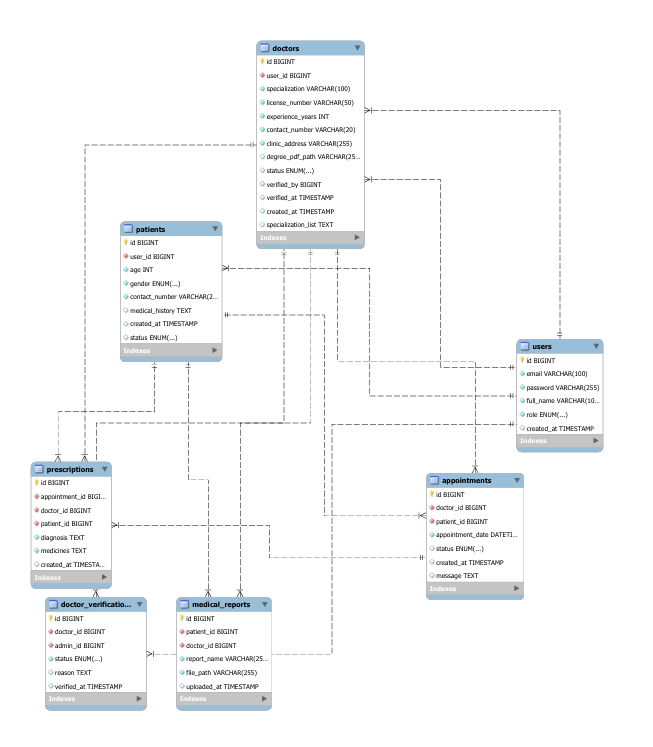
### *5. Concurrency and Synchronization*

* ***Real-Time Transcription and AI Processing:*** *Leveraging multi-threaded processing ensures that transcription and diagnostic assistance operate concurrently without lag.*
* ***Synchronization Across Portals:*** *Real-time updates to consultation records and AI outputs are synchronized between the doctor and patient portals to maintain consistency.*
* ***Load Balancing:*** *Ensures equitable resource allocation for multiple users during peak usage times, particularly for video consultations and real-time processing.*

# Detailed System Design

## Database Design

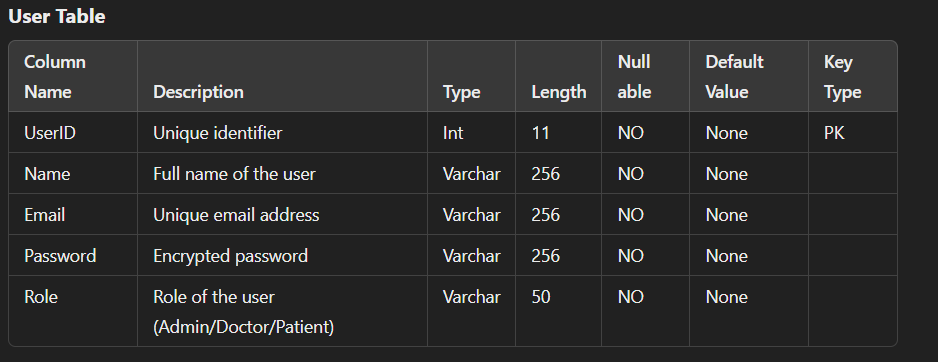
### Scheme Diagram



### Data Dictionary

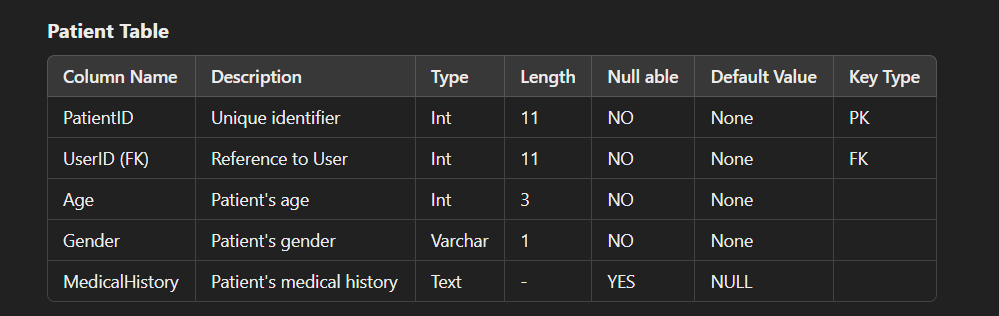
#### Data 1

|  |  |
| --- | --- |
| **User** | |
| **Name** | User |
| **Alias** | User |
| **Where-used/how-used** | All users register through the system. Their credentials allow role-based access (e.g., scheduling appointments, managing consultations, or auditing). |
| **Content description** | Represents a system user who can have different roles such as Patient, Doctor, or Admin. |



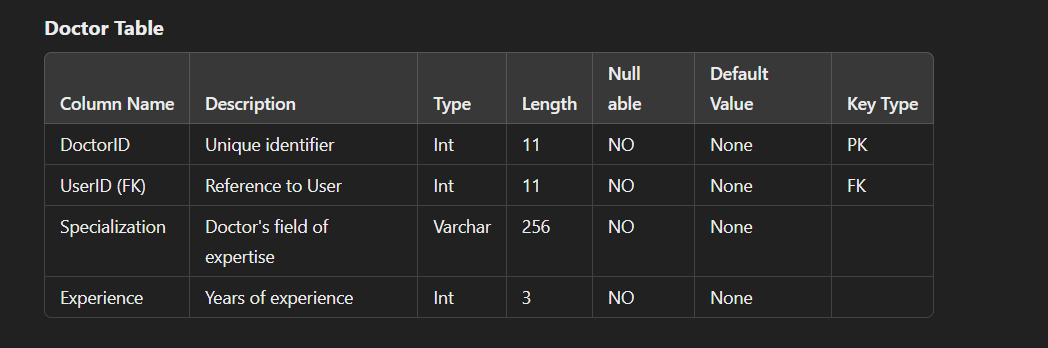
#### Data 2

|  |  |
| --- | --- |
| **Patient** | |
| **Name** | Patient |
| **Alias** | Patient |
| **Where-used/how-used** | Registers through the system. Schedules appointments with doctors and provides medical history for consultations. |

.

#### Data 3

|  |  |
| --- | --- |
| **Doctor** | |
| **Name** | Doctor |
| **Alias** | Doctor |
| **Where-used/how-used** | Registers through the system. Accesses patient records, manages consultations, and provides diagnoses. |
| **Content description** | A user providing medical services, managing appointments, and consultations. |



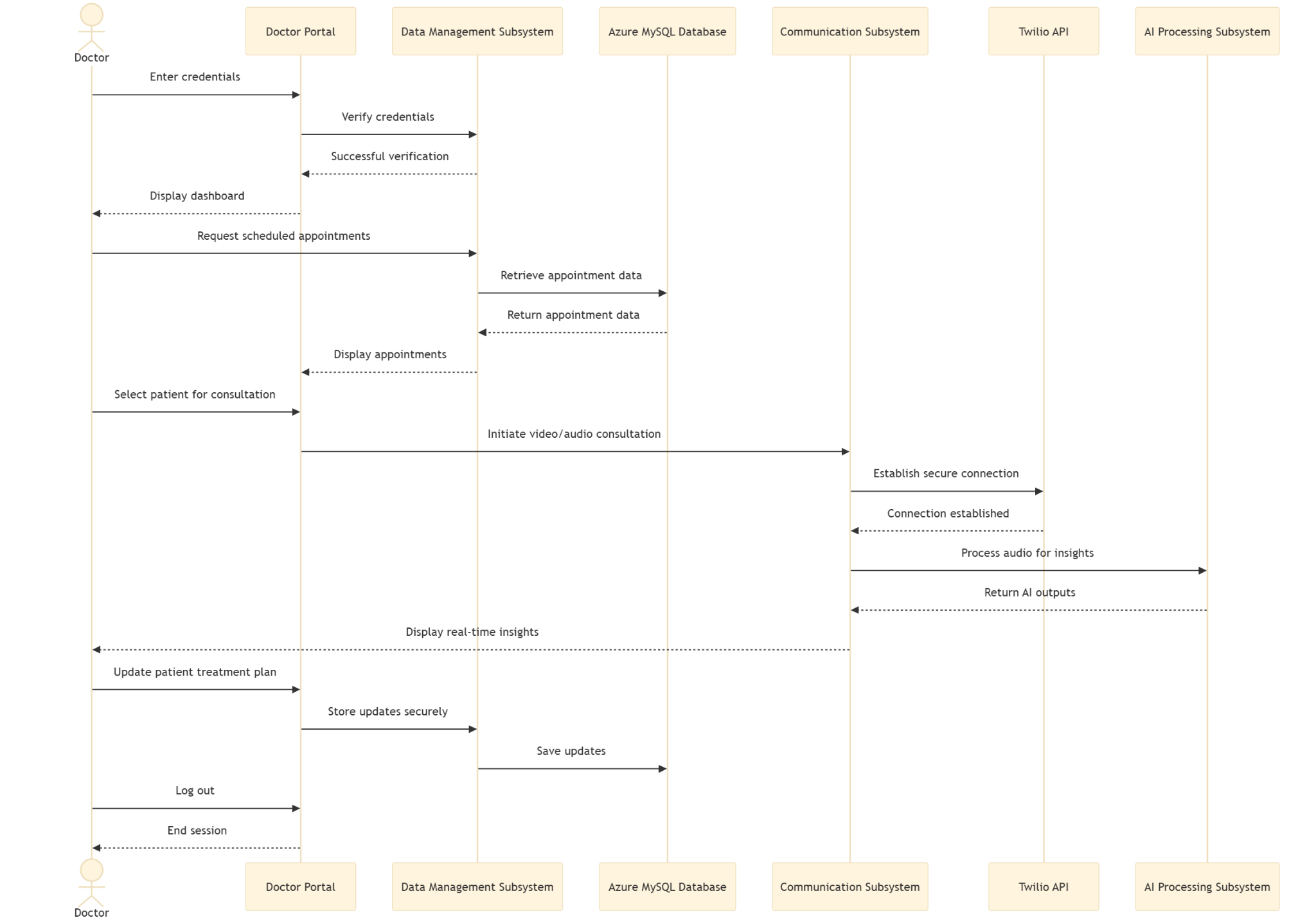
#### Data 4

|  |  |
| --- | --- |
| **Admin** | |
| **Name** | Admin or Manager |
| **Alias** | Admin |
| **Where-used/how-used** | Admins are either registered manually by higher-level authorities or are assigned system privileges. They are responsible for managing user accounts, including Patients, Doctors, and other Admins. Their duties include monitoring the **Audit Logs** to track system activity and ensure accountability. Additionally, Admins oversee the configuration and utilization of the AI Model, ensuring proper integration into consultations and evaluations. |
| **Content description** | The Admin represents a privileged user responsible for managing and maintaining the system's functionality and user roles. Admins play a critical role in overseeing operations, ensuring system security, and monitoring activities. |
|  | |

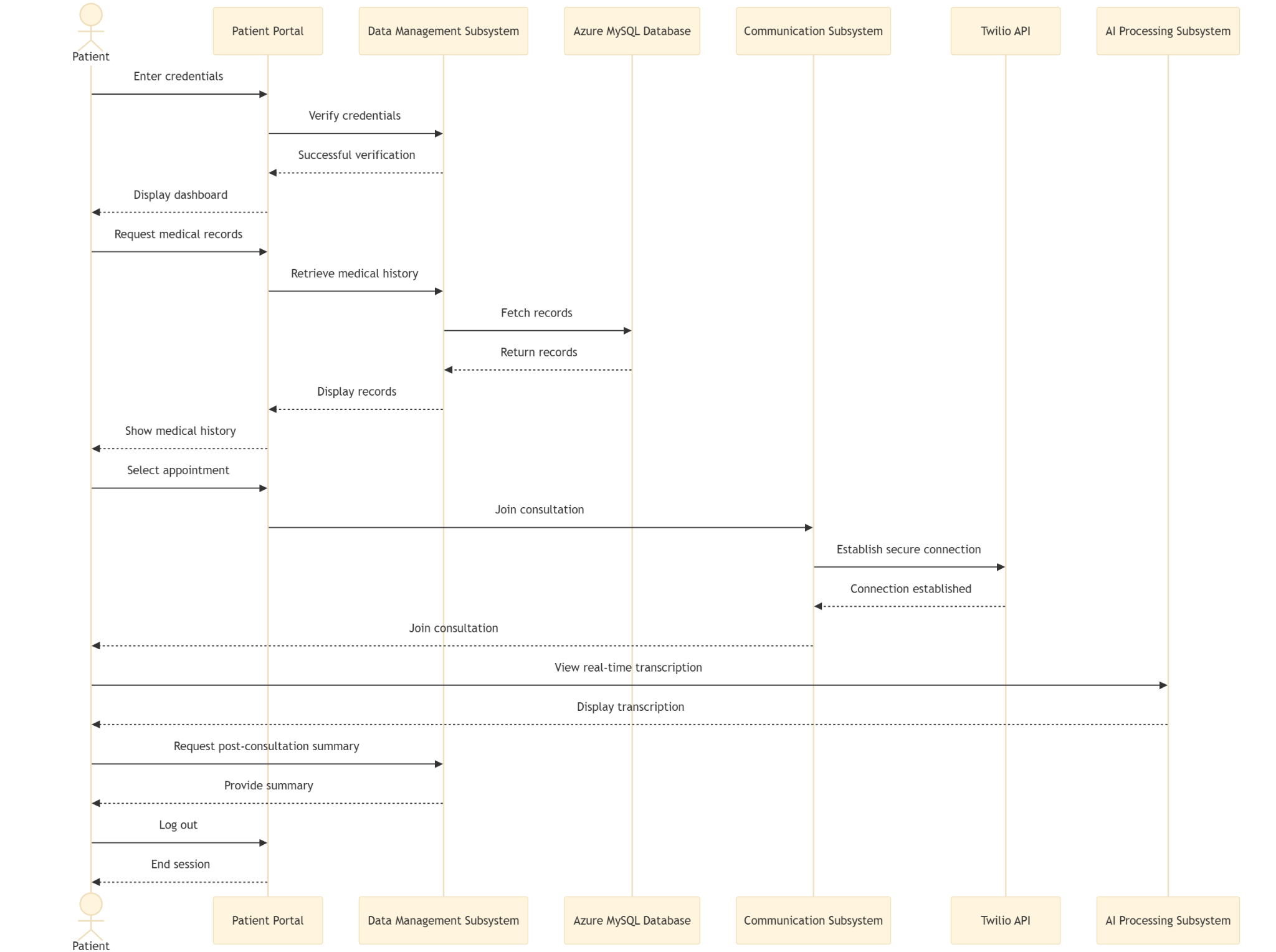
## Application Design

### Sequence Diagram

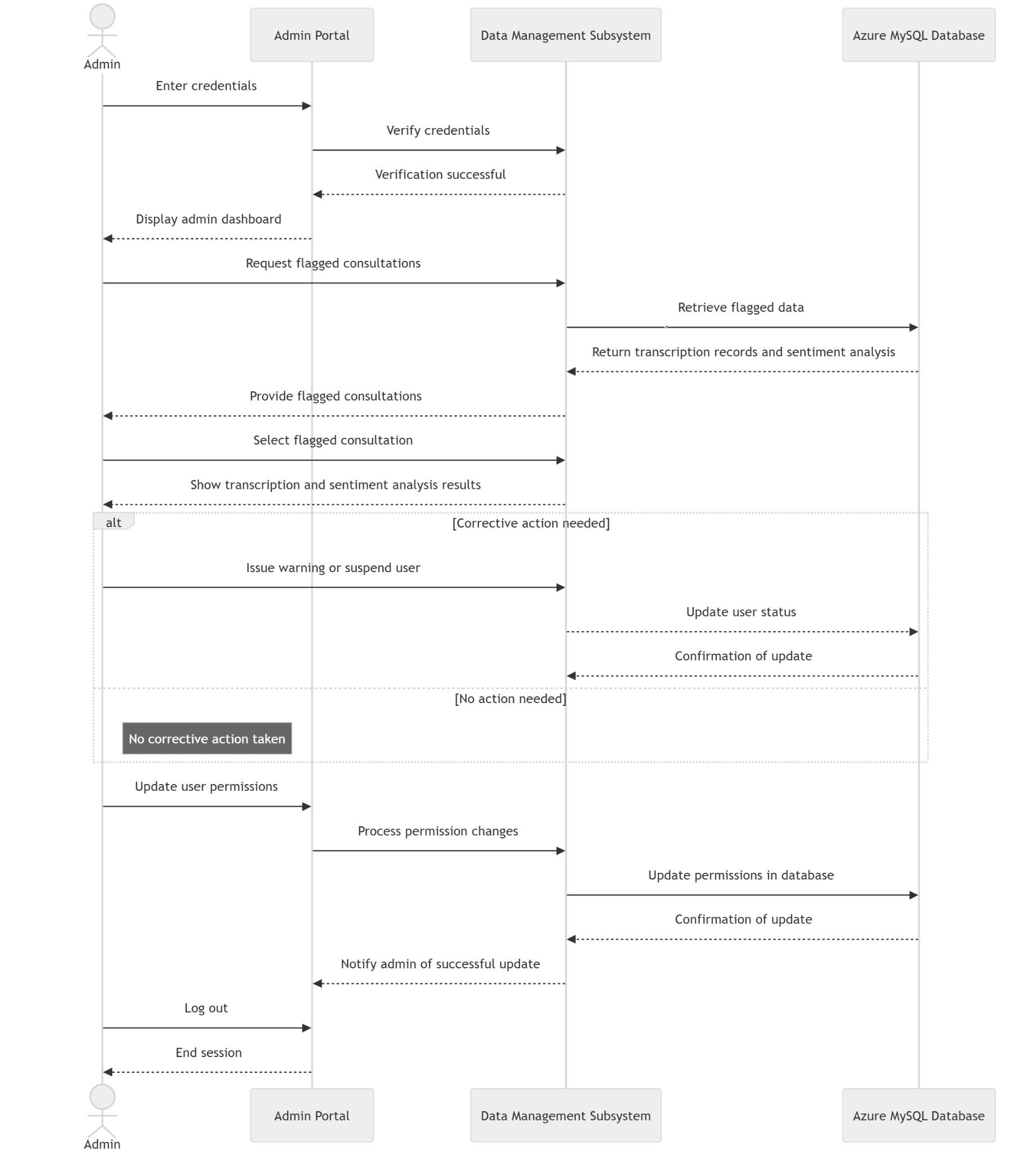
#### <Sequence Diagram 1>

**

#### <Sequence Diagram 2>

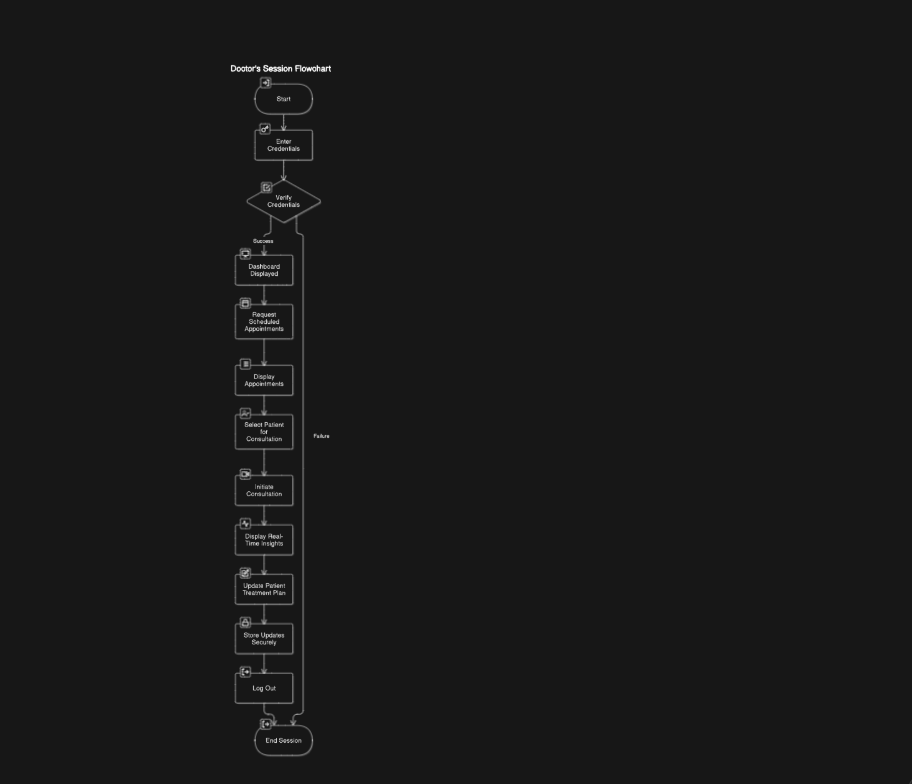
**

#### <Sequence Diagram 3>

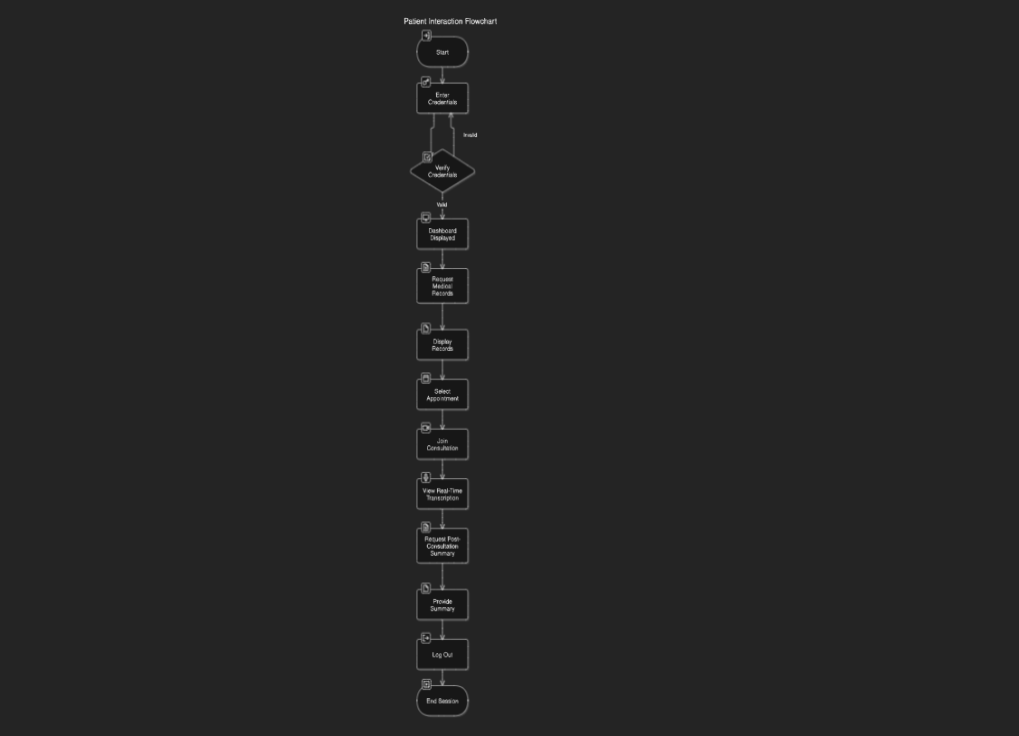
**

### State Diagram

#### <State Diagram 1 Doctor>

**

#### <State Diagram 2 Patient>



#### <State Diagram 3 Admin>

# References

* A. Haleem, M. Javaid, R.P. Singh, and R. Suman, “Telemedicine for healthcare: Capabilities, features, barriers, and applications,” *ScienceDirect*, Elsevier, India, 2021.
* X. Luo, P. Gandhi, S. Storey, and K. Huang, “A Deep Language Model for Symptom Extraction From Clinical Text and its Application to Extract COVID-19 Symptoms From Social Media,” *IEEE*, USA, April 2022.
* D. Guo, M. Li, Y. Yu, Y. Li, G. Duan, F. Wu, and J. Wang, “Disease Inference with Symptom Extraction and Bidirectional Recurrent Neural Network,” *IEEE*, Spain, 2018.
* A. Schiegl, “Disease-Symptom relation extraction from medical text corpora with BERT,” *reposiTUm*, 2021.
* S. Wada, R. Iida, K. Torisawa, T. Takeda, S. Manabe, and Y. Matsumura, “Extracting Symptom Names and Disease-Symptom Relationships from Web Texts Using a Multi-Column Convolutional Neural Network,” *IOS Press*, 2019.
* K.S. Kumar, M.S. Sathya, A. Nadeem, and S. Rajesh, “Diseases Prediction based on Symptoms using Database and GUI,” *IEEE*, India, 2022.
* P. Hema, N. Sunny, R.V. Naganjani, and A. Darbha, “Disease Prediction using Symptoms based on Machine Learning Algorithms,” *IEEE*, India, 2022.
* P. Hamsagayathri and S. Vigneshwaran, “Symptoms Based Disease Prediction Using Machine Learning Techniques,” *IEEE*, India, 2021.
* A.C.M.V. Srinivas, C. Satyanarayana, C. Divakar, and K.P. Sirisha, “Sentiment Analysis using Neural Network and LSTM,” *IOP Publishing*, India, 2021.
* D. Li and J. Qian, “Text sentiment analysis based on long short-term memory,” *IEEE*, China, 2016.
* P.A. Naidu, K.D.K. Yadav, B. Meena, and Y.V.N. Meesala, “Sentiment Analysis By Using Modified RNN And A Tree LSTM,” *IEEE*, India, 2022.
* Karen Walker, “About The Future Of Telehealth And Technology,” *Forbes*, 2024.
* MARKETSANDMARKETS, “Telehealth & Telemedicine Market Size, Share & Trends - Global Forecast to 2028.”

# Appendices

#### Appendix A: Glossary of Terms

This section provides definitions of terms, acronyms, and abbreviations used in the document to ensure clear understanding.

|  |  |
| --- | --- |
| **Term** | **Description** |
| **AI** | Artificial Intelligence, enables machines to simulate human decision-making. |
| **KPI** | Key Performance Indicator, a measurable value that indicates success metrics. |
| **Tf** | TensorFlow, is an open-source framework for machine learning and AI applications. |
| **Calling API** | API facilitates secure communication via video and voice. |
| **DeepGram** | Real-time transcription API integrated for consultations. |

#### Appendix B: Assumptions and Dependencies

* **Assumptions**:  
  1. Availability of APIs (Room socket,Deepgram) for key functionalities like video calls and transcription.
  2. Users possess basic digital literacy to interact with the system efficiently.
  3. Doctors will adhere to ethical medical practices while using the platform.
* **Dependencies**:  
  1. Third-party APIs for transcription and communication.
  2. Availability of a stable and secure Azure MySQL database for storage.
  3. Pre-trained AI models (e.g., BERT, Random Forest) for sentiment analysis and diagnostic tools.

#### Appendix C: Diagrams

1. **ER Diagram**:  
   * Visual representation of the database structure, showcasing entities such as User, Patient, Doctor, and Admin along with their relationships.
2. **Sequence Diagram**:  
   * Diagrams detailing interaction flows between users (patients, doctors, admins) and the system during critical operations like consultations and data retrieval.
3. **State Diagram**:  
   * Describes state transitions for key actors, such as:
     + **Doctor State Diagram**: From login to accessing records or managing consultations.
     + **Patient State Diagram**: From booking appointments to reviewing past consultations.

#### Appendix D: Testing Requirements

* **Functional Testing**:  
  1. Verification of login, role-based access, and dashboard functionalities.
  2. Testing transcription accuracy during live consultations.
* **Performance Testing**:  
  1. System load-handling capabilities for concurrent user sessions.
  2. Real-time AI performance in diagnosing conditions.
* **Security Testing**:  
  1. Role-based access control to ensure data confidentiality.
  2. Encryption mechanisms for data in transit and storage.

#### Appendix E: User Interface Mockups

* **Patient Dashboard**:  
  + Features include appointment scheduling, accessing past consultations, and real-time transcription visibility.
* **Doctor Dashboard**:  
  + Capabilities for viewing patient histories, AI-powered diagnostic tools, and sentiment analysis insights.
* **Admin Dashboard**:  
  + Functionalities for auditing flagged consultations, managing roles, and monitoring system performance.

#### Appendix F: Reference Materials

This appendix includes summaries of referenced works and their relevance to the project, such as the use of AI in healthcare, telemedicine challenges, and sentiment analysis using machine learning.