|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Software Requirement 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 | 15/05/2025 | |

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** | **Sign-off Date** |
| 1.0.0 | Supervisor Nida Munawar | 15/05/2025 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Table of Contents**

[1.](#_30j0zll) Introduction **7**

[1.1.](#_3znysh7) Purpose of Document 7

[1.2.](#_2et92p0) Intended Audience 7  
1.3 Abbreviations ………………………………………………………………………………………...7

[1.4.](#_26in1rg) Document Convention 7

[2.](#_35nkun2) Overall System Description **8**

[2.1.](#_1ksv4uv) Project Background 8

[2.2.](#_44sinio) Project Scope 8

[2.3.](#_4i7ojhp) Not In Scope 9

[2.4.](#_2xcytpi) Project Objectives 9

[2.5.](#_1ci93xb) Stakeholders 9

[2.6.](#_2p2csry) Operating Environment 9

[2.7.](#_3o7alnk) System Constraints 9

[2.8.](#_23ckvvd) Assumptions & Dependencies 10

[3.](#_ihv636) External Interface Requirements **11**

[3.1.](#_32hioqz) Hardware Interfaces 11

[3.2.](#_1hmsyys) Software Interfaces 11

[3.3.](#_1v1yuxt) Communications Interfaces 12

[4.](#_4f1mdlm) Functional Requirements **13**

[4.1.](#_2u6wntf) Functional Hierarchy 13

[4.1.1.](#_37m2jsg) Use Case Diagram 15

[5.](#_1mrcu09) Non-functional Requirements **28**

[5.1.](#_46r0co2) Performance Requirements **28**

[5.2.](#_111kx3o) Safety Requirements 28

[5.3.](#_3l18frh) Security Requirements 28

[5.4.](#_206ipza) User Documentation 29

[6.](#_2zbgiuw) References 30

[7.](#_1egqt2p) Appendices 31

1. **Introduction**

* 1. **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 serves as 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
  1. **Intended Audience**
* Jury
* Supervisor (Nida Munawar)
* Students of Fast NU
* Potential Users of this product

**1.3 Abbreviations**

* HOD = Head Of Department
* KPI = Key Performance Indicators
  1. **Document Convention**
* Font Family = Arial
* Font Size = 12 for headings, 10 for the rest of the content

1. **Overall System Description**
   1. **Project Background**

In the evolving healthcare landscape, ensuring high-quality, efficient, and accessible telehealth services is critical. Traditional telehealth platforms often fall short in maintaining professionalism, enhancing diagnostic accuracy, and providing comprehensive patient-doctor interaction records. These gaps result in limited accountability, inefficiencies, and unmet patient needs.

CureBridge aims to address these challenges by integrating cutting-edge artificial intelligence, real-time transcription, and centralized data management into a single, robust telehealth platform. By leveraging advanced technologies, the project aspires to streamline medical consultations, improve diagnostic accuracy, and uphold professional standards in patient-doctor interactions.

### Problem Statement:

1. **Lack of Accountability:** Existing telehealth platforms do not adequately monitor doctor-patient conversations, leading to potential unprofessional behavior.
2. **Limited Diagnostic Support:** Traditional systems lack AI-driven tools to assist doctors in extracting symptoms and predicting diseases during consultations.
3. **Fragmented Data Systems:** Patient records are often scattered across multiple platforms, making it difficult for doctors and patients to access historical consultation data.
4. **Resource Inefficiency:** Manual tracking of consultation quality and patient history increases workload and is prone to errors.
   1. **Project Scope**

**CureBridge** is a comprehensive telehealth platform designed to address key gaps in traditional telemedicine services. It leverages AI-driven tools, real-time transcription, and centralized data management to enhance the quality, efficiency, and accessibility of healthcare.

#### Included Functionalities

* **Real-Time Transcription**
  + Capture and display patient-doctor conversations in real time using transcription services.
  + Store transcriptions securely for future reference.
* **AI Diagnostic Assistance**
  + Assist doctors in extracting symptoms from conversations.
  + Provide real-time disease predictions based on extracted symptoms.
* **Sentiment Analysis**
  + Detect toxic behavior, harassment, or unprofessional interactions during consultations.
  + Flag inappropriate consultations for administrator review.
* **Centralized Patient Records**
  + Store consultation histories, diagnostic reports, and patient data in a unified repository.
  + Provide secure, easy access to medical records for patients and doctors.
* **Booking and Consultation Management**
  + Allow patients to book and manage appointments through a user-friendly portal.
  + Enable doctors to view, confirm, and manage appointments.
* **Administrative Tools**
  + Provide administrators with tools for quality control, including reviewing flagged interactions and managing user access.

#### Excluded Functionalities

* Manual verification of uploaded or existing medical records.
* Offline access to consultation records or AI features.
* Real-time diagnostics for medical emergencies requiring physical equipment or interventions.
  1. **Not In Scope**
* Development of custom AI models from scratch for all use cases. Pre-trained models will be adapted.
* Handling or integration of non-healthcare-related services (e.g., billing systems).
* Offline functionality for areas with limited internet access.

* 1. **Project Objectives**
* Enhance diagnostic accuracy through AI tools supporting symptom extraction and disease prediction.
* Centralize consultation records for continuity of care and easy reference.
* Improve doctor-patient communication with real-time transcription and user-friendly interfaces.
* Ensure high standards of professionalism through sentiment analysis tools.
* Provide a scalable platform addressing accessibility challenges in underserved areas.
  1. **Stakeholders**
* **Patients:** Seamless appointment scheduling, access to medical records, and clear consultation summaries.
* **Doctors:** Diagnostic assistance, transcription tools, and efficient workflow management.
* **Administrators:** Oversight tools for monitoring consultation quality and ensuring compliance.
* **Supervisors:** Project reviewers and guides ensuring the delivery aligns with goals.
  1. **Operating Environment**.
* Cloud-based deployment with secure data storage and real-time AI processing.
* Video consultations are supported by API for secure video/audio communication.
* Real-time transcription integrated via Deepgram.
* User access via web browsers with stable internet connectivity
  1. **System Constraints**
* AI models require access to pre-trained resources and sufficient computational power.
* The system relies on MySQL for centralized database management.
* Continuous internet connectivity is necessary for real-time functionalities such as transcription and AI processing.
* Blockchain-based smart contracts for certain features depend on Multichain node availability.
  1. **Assumptions**
* Users will provide accurate and truthful data during consultations and research paper submissions.
* Healthcare professionals will follow standard operating procedures when using AI diagnostic tools.
* All stakeholders have basic proficiency with digital platforms and web-based applications.
  1. **Dependencies**
* **Third-Party APIs:** API for video calling, Deepgram for transcription, and CrossRef for research paper verification.
* **AI Models:** Pre-trained models like BERT, Random Forest, and LSTM for specific use cases.
* **Database:** MySQL is the backend for storing records and user data.
* **Frameworks:** Flask and Spring Boot for backend logic, Angular/React for frontend development.

1. **External Interface Requirements**
   1. **Hardware Interfaces**

**Desktop Devices:**

* Minimum specifications:
  + 4GB RAM
  + Dual-core processor
  + Operating system: Windows 10, macOS 10.13, or Linux (kernel 4.x or later)
* Devices must support a web browser capable of running the platform and installing Python dependencies locally.

**Audio/Video Support:**

* A microphone and camera are required for video consultations through Calling API.

**Server Requirements:**

* Cloud or local servers must support the deployment of AI models and MySQL database management systems.
  1. **Software Interfaces**

**Database Management:**

* **Azure MySQL Database:**
  + Central repository for storing user information, consultation records, and AI-generated data.

**AI Frameworks:**

* TensorFlow or PyTorch for deploying pre-trained models (e.g., BERT, Random Forest, LSTM).
* Python libraries for implementing AI functionalities.

**Web Frameworks:**

* Backend: Flask (Python) or Spring Boot (Java) for API handling and logic processing.
* Frontend: Angular or React for responsive user interface development.

**APIs:**

* **Room API:** Facilitates secure video and audio communication for consultations.
* **Deepgram API:** Provides real-time transcription services during consultations.

.

**Data Encryption:**

* AES-256 is used to secure sensitive data like user credentials and consultation records.
  1. **Communications Interfaces**

**Data Exchange Format:**

* JSON format for seamless data exchange between frontend and backend systems.

**Real-Time Communication:**

* **WebSocket Protocols:** Used for real-time data transmission (e.g., transcription updates during live consultations).
* **API Video Service:** Manages secure video/audio communication streams.

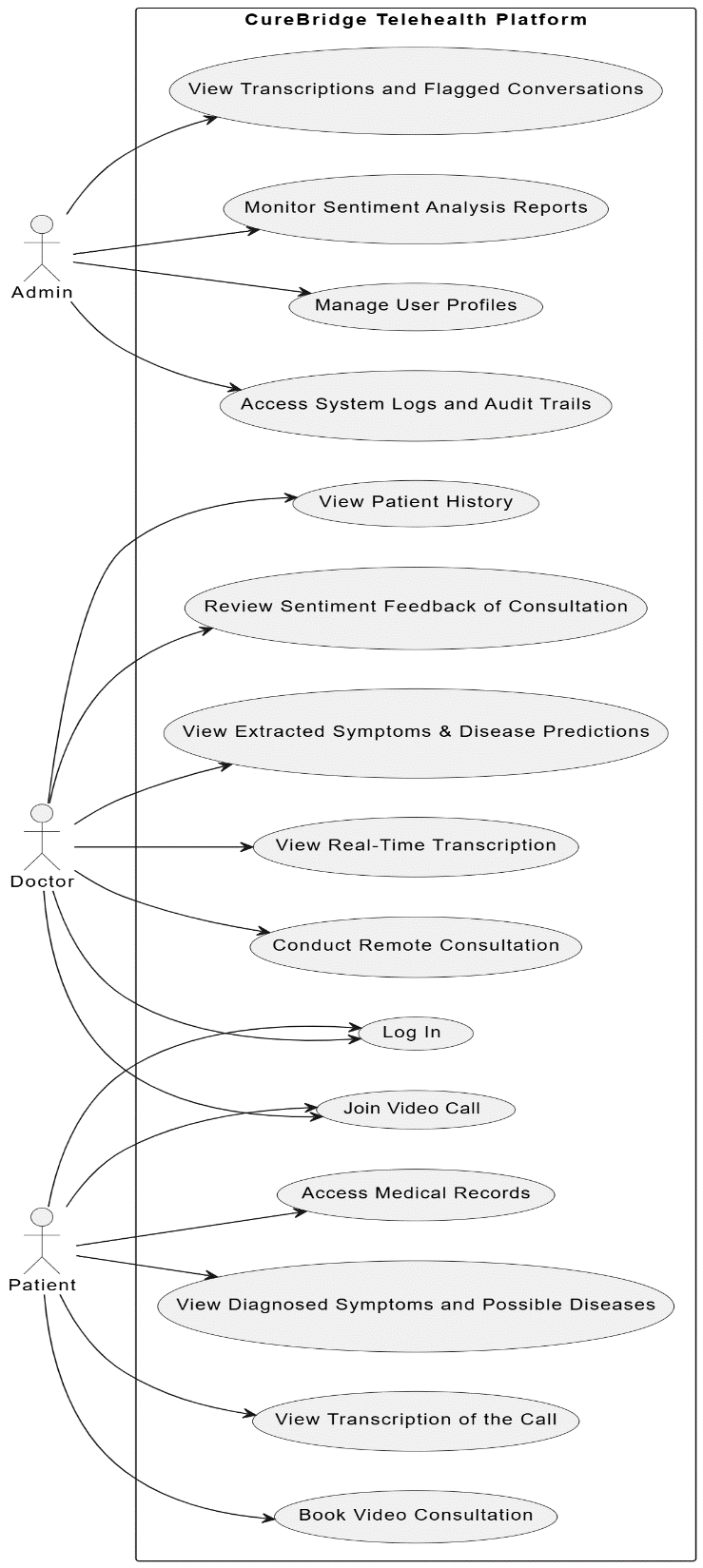
**Blockchain Interaction:**

* Communicate with Multichain blockchain to implement transparent smart contracts related to promotion criteria and other administrative tasks.

**Network Requirements:**

* A stable internet connection ensures uninterrupted video calls, transcription processing, and database synchronization.

1. **Functional Requirements**
   1. **Functional Hierarchy**
2. **Patient Functionality** 2.1 Book Appointments.  
    2.2 View Personal Records:  
    2.2.1 Consultation History.  
    2.2.2 Real-Time Transcriptions.  
    2.2.3 Diagnoses and Follow-Up Plans.  
    2.3 Access Consultation Transcriptions.
3. **Doctor Functionality** 3.1 View Patient Records:  
    3.1.1 Patient Consultation History.  
    3.1.2 Transcription Records.  
    3.1.3 Past Diagnoses and Treatment Plans.  
    3.2 Use AI Assistance:  
    3.2.1 Extract Symptoms from Real-Time Transcription.  
    3.2.2 Generate Possible Diagnoses.  
    3.3 Monitor Consultation Sentiments.  
    3.4 Review Flagged Consultations.
4. **Admin Functionality** 4.1 Manage Users:  
    4.1.1 Add, Edit, or Delete Users.  
    4.1.2 Assign Roles and Permissions.  
    4.2 Oversee Flagged Consultations:  
    4.2.1 Review Sentiment Analysis Reports.  
    4.2.2 Take Corrective Actions (Revoking Access or Issuing Warnings).
5. **AI Processing** 5.1 Symptom Extraction from Real-Time Transcriptions.  
    5.2 Disease Prediction Based on Extracted Symptoms.  
    5.3 Sentiment Analysis of Consultations:  
    5.3.1 Detect Harassment or Toxic Behavior.  
    5.3.2 Flag Consultations for Admin Review.
6. **Real-Time Transcription Management** 6.1 Capture Conversations During Video Calls.  
    6.2 Store Transcriptions Securely in the Database.  
    6.3 Display Transcriptions in Real-Time for Both Parties.
7. **Appointment and Consultation Management** 7.1 Patient Portal:  
    7.1.1 View Available Doctors.  
    7.1.2 Schedule and Manage Appointments.  
    7.2 Doctor Portal:  
    7.2.1 View Scheduled Consultations.  
    7.2.2 Access Patient Details Before Appointments.  
    7.3 Admin Portal:  
    7.3.1 Oversee All Appointments and Cancellations.
8. **Sentiment Analysis** 8.1 Analyze Transcription Sentiments Post-Consultation.  
    8.2 Flag Negative Sentiments for Admin Review.  
    8.3 Provide Detailed Sentiment Reports to Admins.
9. **Data and Record Management** 9.1 Centralized Storage of Consultation Records.  
    9.2 Secure Access to Historical Transcriptions and Diagnoses.  
    9.3 Ensure GDPR and HIPAA Compliance.
10. **System Integration and Dependencies** 10.1 Calling API for Secure Video and Audio Communication.  
     10.2 Deepgram for Real-Time Transcription.  
     10.3 Database Management with Azure MySQL.  
     10.4 AI Model Implementation using TensorFlow or PyTorch.  
    * 1. **Use Case Diagram**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-PAT-01>** | | | | |
| **Use case ID:** | | UC-PAT-01 | | |
| **Actors:**  Patient: Initiates the process | | | | |
| **Feature:** Login (Role-Based Access) | | | | |
| **Pre-condition:** | | * The patient must have a valid account in the system. * The patient must log in with a valid username and password. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The patient selects "Login" on the system login page. | | | Displays the login form. |
| **2.** | The patient enters a username and password. | | | The system verifies the credentials. If valid, grant access. |
| **3.** | The patient clicks the "Submit" button. | | | The system redirects to the patient's dashboard. |
| **Alternate Scenarios:** Patient login to his/her account. | | | | |
| **1a: If invalid credentials are entered, the system displays an error message and prompts re-entry.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1.** | The patient is logged in and can access the role-specific features on the dashboard. | | | |
| **Use Case Cross-referenced** | | | UC-PAT-02: Book Appointment | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-PAT-02>** | | | | |
| **Use case ID:** | | UC-PAT-02 | | |
| **Actors:**  Patient: Initiates the process | | | | |
| **Feature:** Book Appointment | | | | |
| **Pre-condition:** | | The patient must be logged into the system.  Patients must have access to available doctor schedules. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The patient selects "Book Appointment" on the dashboard. | | | The system displays available doctor slots and times. |
| **2.** | The patient selects a doctor and time slot. | | | The system confirms the appointment and updates the patient's records. |
| **3.** | The patient clicks "Confirm Appointment". | | | The system sends a confirmation email to the patient and doctor. |
| **Alternate Scenarios:** Patient make an appointment with doctor. | | | | |
| **1a: If the slot is unavailable, the system displays a message indicating no availability.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1.** | Appointment is successfully booked and reflected in both the patient and doctor's schedules. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross-referenced** | | | * **UC-PAT-01: Login (Role-Based Access)** | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-DOC-01>** | | | | |
| **Use case ID:** | | UC-DOC-01 | | |
| **Actors:**  **Doctor: Initiates the process** | | | | |
| **Feature:** Login (Role-Based Access) | | | | |
| **Pre-condition:** | | The doctor must have a valid account in the system.  The doctor must log in with a valid username and password. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The doctor selects "Login" on the system login page. | | | Displays the login form. |
| **2.** | The doctor enters the username and password. | | | The system verifies the credentials. If valid,grants access. |
| **3.** | The doctor clicks the "Submit" button. | | | The system redirects to the doctor's dashboard. |
| **Alternate Scenarios:** Doctor login to his/her account. | | | | |
| **1a: If invalid credentials are entered, the system displays an error message and prompts re-entry.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1.** | The doctor is logged in and can access role-specific functionalities. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross-referenced** | | | UC-DOC-02: Access Medical Records | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-DOC-02>** | | | | |
| **Use case ID:** | | UC-DOC-02 | | |
| **Actors:**  Doctor: Initiates the process | | | | |
| **Feature:** Access Medical Records | | | | |
| **Pre-condition:** | | * The doctor must be logged into the system. * Doctors must have access to patient records. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The doctor selects "Access Medical Records" from the dashboard. | | | Displays a list of patients and their records. |
| **2.** | The doctor selects a patient record to view. | | | The system displays the patient's medical history. |
| **3.** | The doctor reviews the medical record. | | | The doctor uses the information for the consultation |
| **Alternate Scenarios:** Doctor reviews the patient medical record. | | | | |
| **1a: If no records are found for the patient, the system displays a message indicating no records are available.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1.** | The doctor has accessed the required patient records. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross-referenced** | | | .  **UC-DOC-01: Login (Role-Based Access)** | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-ADM-01>** | | | | |
| **Use case ID:** | | UC-ADM-01. | | |
| **Actors:**  Administrator: Initiates the process | | | | |
| **Feature:** Login (Role-Based Access) | | | | |
| **Pre-condition:** | | * The administrator must have a valid account in the system. * The administrator must log in with a valid username and password. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The administrator selects "Login" on the system login page. | | | Displays the login form. |
| **2.** | The administrator enters username and password. | | | The system verifies the credentials. If valid, grant access. |
| **3.** | The administrator clicks the "Submit" button. | | | The system redirects to the administrator's dashboard. |
| **Alternate Scenarios:** Admin login to the account. | | | | |
| **1a: I: If invalid credentials are entered, the system displays an error message and prompts re-entry.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | The administrator is logged in and can access role-specific functionalities. | | | |
| **Use Case Cross-referenced** | | | **UC-ADM-02: Manage Users** | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-ADM-02>** | | | | |
| **Use case ID:** | | UC-ADM-02. | | |
| **Actors:**  **Administrator: Initiates the process** | | | | |
| **Feature:** Admin Full Action (All Admin Actions) | | | | |
| **Pre-condition:** | | The administrator must be logged into the system.  Administrators must have permission to manage user accounts. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The administrator logs into the system. | | | Displays the admin dashboard with options like user management, consultation auditing, and sentiment analysis |
| **2.** | The administrator selects a user to update or modify. | | | The system displays user details and editable fields for roles/permissions. |
| **3.** | The administrator modifies user details and clicks "Save." | | | The system updates the user's role and permissions in the database. |
| **4.** | The administrator selects "Sentiment Analysis" to monitor the consultation tone. | | | The system analyzes ongoing consultations for sentiment and displays results. |
| **5.** | The administrator selects "Audit Consultations" to review consultations. | | | Displays a list of consultations for review, highlighting flagged interactions based on sentiment analysis. |
| **Alternate Scenarios:** Admin login to portal, to modify the user details or to review the sentiments of the past consultations. | | | | |
| **1a: If invalid credentials are entered, the system displays an error message and prompts re-entry.**  **2a: If no consultations are flagged, the system displays a message: "No flagged consultations available."** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | The administrator has successfully managed users, reviewed consultations, and accessed system settings. | | | |
| **Use Case Cross-referenced** | | | **UC-ADM-02: Manage Users**  **UC-ADM-03: Audit Consultations**  **UC-ADM-04: Sentiment Analysis** | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-PAT-03>** | | | | |
| **Use case ID:** | | UC-PAT-03 | | |
| **Actors:**  Patient: Initiates the process | | | | |
| **Feature:** Access Medical Records | | | | |
| **Pre-condition:** | | * The patient must be logged into the system. * Patients must have records available in the system. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The patient selects "Access Medical Records" from the dashboard. | | | Displays a list of the patient's available medical records. |
| **2.** | The patient selects a specific record to view. | | | The system displays the selected record, including medical history. |
| **3.** | The patient reviews the medical record details. | | | Patients can view detailed information about past consultations, treatments, etc. |
| **Alternate Scenarios:** Patient login to the account to view medical record. | | | | |
| **1a: If no records are found, the system displays a message indicating "No records found."** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | The patient successfully views their medical records. | | | |
| **Use Case Cross-referenced** | | | **UC-PAT-01: Login (Role-Based Access)** | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-PAT-04>** | | | | |
| **Use case ID:** | | UC-PAT-04 | | |
| **Actors:**  **Patient**: Initiates the process | | | | |
| **Feature:** **Real-Time Consultation** | | | | |
| **Pre-condition:** | | The patient must be logged into the system.  An appointment with a doctor must be confirmed. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The patient selects "Join Consultation" from the dashboard. | | | Displays a call window with the doctor’s details. |
| **2.** | The patient clicks "Start" to initiate the consultation. | | | The system establishes a secure call connection with the doctor. |
| **3.** | Patient participates in the consultation. | | | Real-time transcription and AI-powered diagnostics are available |
| **Alternate Scenarios:** Patient starts one to one consultation with doctor. | | | | |
| **1a: If there is a connection issue, the system prompts the patient to try again or contact support.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | The patient successfully completes the call consultation. | | | |
| **Use Case Cross-referenced** | | | UC-PAT-02: Book Appointment | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-PAT-05>** | | | | |
| **Use case ID:** | | UC-PAT-05 | | |
| **Actors:**  **Patient: Initiates the process** | | | | |
| **Feature:** Review Past Consultations | | | | |
| **Pre-condition:** | | The patient must be logged into the system.  Patients must have past consultations recorded. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The patient selects "Review Past Consultations" from the dashboard. | | | Displays a list of past consultations. |
| **2.** | The patient selects a consultation from the list. | | | The system displays a summary of the consultation, including transcription. |
| **3.** | The patient reviews the consultation summary and transcription. | | | Patients can download or print the records if needed. |
| **Alternate Scenarios:** Patient reviews the consultation transcription. | | | | |
| **1a: If no past consultations exist, the system displays "No past consultations available.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | The patient successfully view past consultations. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross-referenced** | | | **UC-PAT-01: Login (Role-Based Access)** | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-DOC-03>** | | | | |
| **Use case ID:** | | UC-DOC-03 | | |
| **Actors:**  **Doctor: Initiates the process** | | | | |
| **Feature:** Real-Time call Consultation | | | | |
| **Pre-condition:** | | The doctor must be logged into the system.  An appointment with a patient must be confirmed. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The doctor selects "Join call Consultation" from the dashboard.." | | | Displays a call window with patient details. |
| **2.** | The doctor clicks "Start" to initiate the consultation. | | | The system establishes a secure call connection with the patient. |
| **3.** | The doctor participates in the consultation. | | | Real-time transcription and AI-powered diagnostics are available. |
| **Alternate Scenarios:** Doctor provides consultation to patient. | | | | |
| **If there is a connection issue, the system prompts the doctor to try again or contact support.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | The doctor successfully completes the call consultation. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross-referenced** | | | **UC-DOC-02: Access Medical Records** | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-DOC-04>** | | | | |
| **Use case ID:** | | UC-DOC-04 | | |
| **Actors:**  **Doctor: Initiates the process** | | | | |
| **Feature:** AI-Powered Diagnostic Assistance | | | | |
| **Pre-condition:** | | The doctor must be logged into the system.  Patient consultation data must be available. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The doctor begins a consultation with a patient. | | | The system displays patient symptoms and medical history. |
| **2.** | Doctor inputs observed symptoms. | | | AI-powered diagnostics suggest possible conditions. |
| **3.** | The doctor reviews AI suggestions and makes a final diagnosis. | | | The system stores the diagnosis in the patient record. |
| **Alternate Scenarios:** Doctor inputs symptom or use symptom extraction AI. | | | | |
|  | | | | |
| **1a: If AI suggestions are inconclusive, the system prompts the doctor to input their diagnosis manually.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | Doctor's diagnosis is stored in the patient’s record. | | | |
| **Use Case Cross-referenced** | | | UC-DOC-02: Access Medical Records | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<UC-DOC-05>** | | | | |
| **Use case ID:** | | UC-DOC-05 | | |
| **Actors:**  **Doctor: Initiates the process** | | | | |
| **Feature:** Update and Store Patient Treatment Plans | | | | |
| **Pre-condition:** | | The doctor must be logged into the system.  Patient consultation and diagnosis must be available. | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | The doctor selects "Update Treatment Plan" from the dashboard. | | | Displays the patient's current treatment plan and diagnosis. |
| **2.** | The doctor updates the treatment plan based on consultation. | | | The system stores the updated treatment plan securely |
| **3.** | Doctor clicks "Save" to confirm the update. | | | The system confirms the update and stores it in the database. |
| **Alternate Scenarios:** Doctor updates patient treatment plan. | | | | |
| **The system displays an error message if the treatment plan cannot be updated.** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
|  | The patient’s treatment plan is updated and stored. | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross-referenced** | | | **UC-DOC-03: Real-Time Call Consultation** | |

1. **Non-functional Requirements**

* 1. **Performance Requirements**

**Real-Time Response:**

* Video calls should be smooth and without noticeable lag.
* Real-time transcription must process and display text with minimal latency.

**Throughput:**

* The platform should handle multiple consultations simultaneously without performance bottlenecks.

**AI Model Efficiency:**

* Symptom extraction and disease prediction models should provide accurate results in real time.

**Load Handling:**

* The system must manage a high volume of users during peak hours without crashing or slowing down.

* 1. **Safety Requirements**
* **Patient Data Accuracy:** Ensure transcription and diagnosis suggestions are reliable to avoid miscommunication or errors in patient care.
* **System Reliability:** Avoid unexpected system crashes during live consultations, which could disrupt healthcare services.
* **Backup and Recovery:** Regular data backups and a recovery plan must be in place to prevent data loss.

* 1. **Security Requirements**
* **Data Encryption:** All communication, including video calls, transcriptions, and stored data, must be encrypted.
* **Role-Based Access:** Different access levels should ensure that users only have access to data and functionalities relevant to their role (e.g., patients cannot access admin tools).
* **Authentication:** Secure login mechanisms, such as two-factor authentication, should be implemented.
* **Data Privacy:** Ensure compliance with healthcare data regulations (e.g., HIPAA) to protect sensitive patient information.
* **Anomaly Detection:** Flag and log unauthorized access attempts for security review.

* 1. **User Documentation**
* **Patient Guide**: Instructions for booking consultations, accessing records, and viewing real-time transcriptions.
* **Doctor Guide:** Steps for managing appointments, accessing patient histories, using AI diagnostic tools, and reviewing transcripts.
* **Administrator Guide:** Guidelines for overseeing consultations, managing flagged interactions, and maintaining system operations.
* **Troubleshooting Guide:** Common issues and solutions for video calls, transcription errors, and accessing records.
* **Setup Guide:** Instructions for users to set up their accounts and configure the platform on their devices.

1. **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,” reposited, 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.”

1. **Appendices**

#### Appendix A: Glossary of Terms

* **AI (Artificial Intelligence)**: Technology enabling machines to simulate human intelligence for tasks like problem-solving and decision-making.
* **HIPAA**: Health Insurance Portability and Accountability Act ensuring patient data privacy.
* **GDPR**: General Data Protection Regulation, a legal framework for data protection and privacy in the EU.
* **Calling API**: An API for communication through video, voice, and messaging.
* **Deepgram**: A speech-to-text API used for real-time transcription.
* **Multichain**: A blockchain platform used for implementing decentralized smart contracts.

#### Appendix B: Assumptions and Limitations

* **Assumptions**:
  1. Users will have access to stable internet for real-time functionalities.
  2. Medical professionals will adhere to ethical guidelines while using diagnostic tools.
  3. Patients will input accurate personal and health information.
* **Limitations**:
  1. System features dependent on third-party APIs may experience outages beyond control.
  2. The platform will not operate offline.

#### Appendix C: Data Flow Diagrams

1. **Login Process**: A step-by-step depiction of how the system authenticates users.
2. **Real-Time Consultation**: Illustrates the process from initiating a call to storing transcription data securely.
3. **AI Diagnostic Workflow**: Showcases symptom input to disease prediction flow.

#### Appendix D: Testing Requirements

* **Unit Testing**: For individual components like login functionality, transcription modules.
* **Integration Testing**: Ensuring seamless interaction between Calling API, Deepgram, and backend services.
* **Performance Testing**: Stress testing the platform for high user volumes during peak hours.
* **Security Testing**: Testing role-based access, encryption protocols, and anomaly detection mechanisms.

#### Appendix E: User Interface Mockups

* **Patient Dashboard**: Mockups for appointment booking, accessing medical records, and viewing consultation history.
* **Doctor Dashboard**: Interface for viewing patient records, AI-powered suggestions, and treatment updates.
* **Admin Dashboard**: Tools for managing users, reviewing flagged consultations, and accessing reports.