|  |
| --- |
| **CS - 6110**  **OBJECT ORIENTED ANALYSIS AND DESIGN LABORATORY** |
| **HOSPITAL MANAGEMENT SYSTEM (HMS)** |
| **SOFTWARE REQUIREMENT SPECIFICATION DOCUMENT** |
|  |
| SUBMITTED BY |
| |  | | --- | | **BHAVANA B** | | **2022503305** | | **III rd YEAR & BATCH - 1** | | **COMPUTER TECHNOLOGY** | |

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **CONTENT** | **PAGE NO** |
| **1** | **ABSTRACT** | 3 |
| **2** | **INTRODUCTION**  3.1 PURPOSE  3.2 SCOPE  3.3 DEFINITION AND ABBREVIATIONS  3.4 OVERVIEW | 4 |
| **3** | **OVERALL DESCRITION**  3.1 PRODUCT PERSPECTIVE  3.2 PRODUCT FUNCTIONS  3.3 USER CLASSES AND CHARACTERISTICS  3.4 OPERATING ENVIRONMENT  3.5 DESIGN AND IMPLEMENTATION CONSTRAINTS  3.6 USER DOCUMENTATION | 5 |
| **4** | **SPECIFIC REQUIRMENTS**  4.1 FUNCTIONAL REQUIREMENTS  4.2 NON-FUNCTIONAL REQUIRMENTS | 7 |
| **5** | **SYSTEM FEATURES**  3.1 DOCTOR MANAGEMENT  3.2 APPOINTMENT MANAGEMENT  3.3 CONFIRM APPOINTMENT  3.4 APPOINTMENT STATUS  3.5 CANCEL APPOINTMENT  3.6 PRESCRIPTION MANAGEMENT  5.7 BILLING AND PAYMENT  5.8 USER INTERFACE AND ACCESSIBILITY | 11 |
| **6** | **OTHER REQUIREMENTS**  6.1 LEGAL AND REGULATORY COMPLIANCE  6.2 SAFETY REQUIREMENTS  6.3 ENVIRONMENTAL REQUIREMENTS | 15 |
| **7** | **USE CASE DIAGRAM** | 16 |
| **8** | **APPENDICES**  3.5 GLOSSARY  3.6 SUPPORTING INFORMATION | 17 |
| **9** | **CONCLUSION** | 18 |
| **10** | **REQUIREMENT TRACEABILITY MATRIX** | 19 |

1. **ABSTRACT**

The Hospital Management System (HMS) is designed to streamline and enhance hospital operations, improve patient care, and facilitate efficient management of medical and administrative processes. This system integrates various functionalities to support hospital staff, including doctors, nurses, administrators, and patients, in their daily activities. The HMS provides features such as Doctor Management, which allows administrators to add and update doctor profiles, ensuring accurate and up-to-date information. The Appointment Management module enables patients to schedule, confirm, view, and cancel appointments, while also keeping track of appointment statuses. Prescription Management allows doctors to add and manage patient prescriptions, ensuring seamless integration with patient records. In the domain of Billing and Payments, the HMS supports the generation of detailed bills, processing of payments, and management of refunds, thus simplifying financial transactions and ensuring accurate billing. The system also prioritizes User Interface and Accessibility, offering a user-friendly interface that facilitates smooth interaction for all users, including patients, healthcare providers, and administrative staff.

**2. INTRODUCTION**

**2.1 Purpose**

The purpose of this Software Requirements Specification (SRS) is to outline the requirements for a Hospital Management System (HMS) designed to streamline hospital operations, enhance patient care, and integrate with external entities like insurance companies.

**2.2 Scope**

The Health Care Management System will cover electronic health records (EHRs), personalized treatment plans, telemedicine, remote monitoring, appointment scheduling, billing and claims management, resource allocation, clinical decision support, patient engagement, and analytics and reporting.

**2.3 Definitions and Abbreviations**

* **EHR**: Electronic Health Record
* **HMS**: Hospital Management System

**2.4 References**

* HIPAA (Health Insurance Portability and Accountability Act)
* GDPR (General Data Protection Regulation)

**2.5 Overview**

The Health Care Management System will facilitate management of patient data, treatment plans, remote consultations, and billing processes, improving overall hospital efficiency and patient care.

**3. OVERALL DESCRIPTION**

**3.1 Product Perspective**

The HMS will be a web-based application accessible by various actors including patients, doctors, nurses, administrators, and insurance companies. It will integrate with existing hospital systems and external insurance platforms.

**3.2 Product Functions**

The HMS will provide:

* Management of EHRs
* Creation of personalized treatment plans
* Telemedicine capabilities
* Remote monitoring of patients
* Scheduling of appointments
* Billing and claims management
* Allocation of hospital resources
* Clinical decision support
* Patient engagement tools
* Analytics and reporting

**3.3 User Classes and Characteristics**

* **Patient**:

Individuals receiving care, needing access to their health records, appointments, and telemedicine services.

* **Doctor**:

Medical professionals who need access to patient records, treatment plans, and clinical decision support tools.

* **Nurse**:

Healthcare providers who will manage patient care, assist in appointments, and support remote monitoring.

* **Administrator**:

Hospital staff responsible for overall system management, scheduling, and resource allocation.

* **Insurance Company**:

External entities needing access to billing and claims information for processing insurance claims.

**3.4 Operating Environment**

The HMS will operate on a cloud-based platform with a responsive web interface compatible with major browsers and mobile devices.

**3.5 Design and Implementation Constraints**

* Compliance with HIPAA and GDPR for data protection.
* Integration with existing hospital systems and insurance company platforms.

**3.6 User Documentation**

User manuals, online help, and training guides will be provided for each user class.

**4. SPECIFIC REQUIREMENTS**

**4.1 Functional Requirements**

1. **Add Doctor**
   * **Description**: The system must allow an administrator to enter new doctor information into the system.
   * **Requirements**:
     + The system must provide a form to enter doctor details (e.g., name, specialization, contact information).
     + The system must validate the entered data (e.g., check for required fields, format).
     + The system must store the doctor’s information in the database.
     + The system must confirm successful addition of the doctor.
2. **Update Details**
   * **Description**: The system must allow updates to existing records (e.g., doctor details, patient information).
   * **Requirements**:
     + The system must provide an interface to search for and select the record to be updated.
     + The system must allow modifications to selected records.
     + The system must validate the updated data.
     + The system must save changes to the database and confirm successful updates.
3. **Confirm Appointment**
   * **Description**: Allows a user to confirm an appointment.
   * **Requirements**:
     + The system must provide an interface to view appointment details and confirm the appointment.
     + The system must update the appointment status to confirmed in the database.
     + The system must notify the patient and doctor of the confirmation.
4. **Appointment Status**
   * **Description**: Allows users to view the status of appointments (e.g., scheduled, confirmed, completed).
   * **Requirements**:
     + The system must provide an interface to view appointment statuses.
     + The system must display the status of each appointment based on its current state.
5. **Cancel Appointment**
   * **Description**: Allows users to cancel an existing appointment.
   * **Requirements**:
     + The system must provide an interface to select and cancel an appointment.
     + The system must update the appointment status to canceled in the database.
     + The system must notify the patient and doctor of the cancellation.
6. **Add Prescription**
   * **Description**: Allows a doctor to add a prescription to a patient’s record.
   * **Requirements**:
     + The system must provide a form for entering prescription details (e.g., medication, dosage).
     + The system must associate the prescription with the correct patient record.
     + The system must store the prescription details in the database.
7. **Payment**
   * **Description**: Facilitates the processing of payments for services rendered.
   * **Requirements**:
     + The system must provide a payment interface for patients to make payments.
     + The system must integrate with payment gateways for processing transactions.
     + The system must record payment details and update patient billing information.
8. **Refund Payment**
   * **Description**: Allows the processing of refunds for canceled or overpaid services.
   * **Requirements**:
     + The system must provide an interface to initiate and process refunds.
     + The system must update the payment records to reflect the refund.
     + The system must notify the patient of the refund.
9. **Generate Bill**
   * **Description**: Generates a bill for the services provided.
   * **Requirements**:
     + The system must compile all charges related to a patient’s services.
     + The system must generate a bill with itemized charges and totals.
     + The system must provide an interface to view and print the bill.
10. **Make Appointment**
    * **Description**: Allows patients to schedule new appointments with doctors.
    * **Requirements**:
      + The system must provide an interface for patients to select a doctor, date, and time for the appointment.
      + The system must check the availability of the doctor and time slot.
      + The system must create and save the appointment in the database.
      + The system must notify the patient and doctor of the new appointment.

**4.2 Non-Functional Requirements**

1. **Performance**
   * The system must handle up to 10,000 concurrent users with a response time of less than 2 seconds for most operations.
   * The system must process payments and generate bills with minimal delay.
2. **Security**
   * The system must encrypt sensitive data, such as personal and payment information, during transmission and storage.
   * The system must implement role-based access controls to restrict access to sensitive functionalities and data.
3. **Reliability**
   * The system must have 99.9% uptime, with a robust backup and recovery plan in place.
   * The system must handle errors gracefully, providing meaningful error messages and maintaining data integrity.
4. **Usability**
   * The system’s interface must be user-friendly and intuitive for all user roles (patients, doctors, administrators).
   * The system must provide clear instructions and help documentation for performing key tasks.
5. **Scalability**
   * The system must be designed to accommodate future growth in users, transactions, and data without significant rework.
6. **Interoperability**
   * The system must integrate with external systems such as insurance company platforms and medical device interfaces.
7. **Maintainability**
   * The system must be designed with modularity to facilitate easy updates and maintenance.
   * The system must include logging and monitoring tools to assist with troubleshooting and performance monitoring.

**5. SYSTEM FEATURES**

* 1. **Doctor Management**

**5.1.1 Add doctor**

* **Feature Description**:
  + - * Allows administrators to add new doctor profiles to the system. This includes entering details such as name, specialty, contact information, and qualifications.
    - **Capabilities**:
      * Form to input doctor information.
      * Data validation to ensure accuracy and completeness.
      * Save and store doctor information in the database.
      * Confirmation notification upon successful addition.
    - **Update Details**
    - **Feature Description**:
      * Enables users to update existing records, including doctor profiles and patient details.
    - **Capabilities**:
      * Search and select records for updating.
      * Edit and modify record details.
      * Data validation for updated information.
      * Save and update records in the database.
      * Confirmation notification upon successful update.
  1. **Appointment Management**
     + **Make Appointment**
     + **Feature Description**: Facilitates the scheduling of new appointments by patients with doctors.
     + **Capabilities**:
       - * Interface for selecting doctor, date, and time.
         * Availability checking for doctors.
         * Creation and storage of appointment records.
         * Notifications sent to patients and doctors.
  2. **Confirm Appointment**
     + **Feature Description**: Allows for the confirmation of scheduled appointments.
     + **Capabilities**:
       - View appointment details.
       - Confirm appointment status.
       - Update appointment status in the database.
       - Notify both patient and doctor of confirmation.

**5.4 Appointment Status**

* + - **Feature Description**: Provides a view of the status for appointments.
    - **Capabilities**:
      * Interface to view appointment statuses (e.g., scheduled, confirmed, completed).
      * Real-time updates reflecting current appointment statuses.

**5.5 Cancel Appointment**

* + - **Feature Description**: Allows for the cancellation of existing appointments.
    - **Capabilities**:
      * Select and cancel an appointment.
      * Update appointment status to canceled in the database.
      * Notify patient and doctor of cancellation.

**5.6 Prescription Management**

* 1. **Add Prescription**
     + **Feature Description**:
       - Enables doctors to add new prescriptions to a patient’s record.
     + **Capabilities**:
       - Form to enter prescription details (e.g., medication, dosage).
       - Link prescription to the appropriate patient record.
       - Save and store prescription information in the database.
  2. **Billing and Payments**

**5.7.1 Generate Bill**

* + - **Feature Description**: Creates a bill for services provided to the patient.
    - **Capabilities**:
      * + Compile charges and generate a detailed bill.
        + Provide options to view and print the bill.

**5.7.2 Payment**

* + - **Feature Description**: Handles processing of payments for services rendered.
    - **Capabilities**:
      * Interface for patients to make payments.
      * Integration with payment gateways for transactions.
      * Record payment details and update patient billing information.

**5.7.3 Refund Payment**

* + - * **Feature Description**: Manages the processing of refunds for cancelled or overpaid services.
      * **Capabilities**:
        + Interface to initiate and process refunds.
        + Update payment records to reflect refunds.
        + Notify patient of the refund.
  1. **User Interface and Accessibility**
     1. **User-Friendly Interface**
        + **Feature Description**: Provides an intuitive and accessible interface for all users, including patients, doctors, and administrators.
        + **Capabilities**:
          - Easy navigation and user interaction.
          - Responsive design for various devices.
          - Help and support documentation.

**6. OTHER REQUIREMENTS**

**6.1 Legal and Regulatory Compliance**

The system must comply with relevant healthcare regulations and standards.

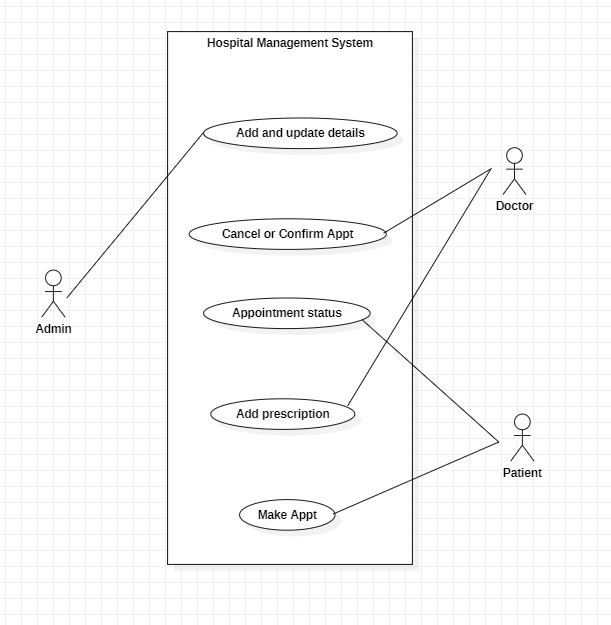
**6.2 Safety Requirements**

The system should include safeguards to ensure patient safety and data security.

**6.3 Environmental Requirements**

The system should be designed to operate in typical hospital IT environments, with considerations for hardware and network configurations.

**7. USE CASE DIAGRAM**

****

**8. APPENDICES**

* 1. **Glossary**
* Diagrams of system architecture, data flow diagrams, and wireframes for key interfaces.
  1. **Supporting Information**
* Diagrams of system architecture, data flow diagrams, and wireframes for key interfaces.

**9. CONCLUSION**

This SRS document details the requirements for the Hospital Management System (HMS), which aims to enhance hospital operations and patient care through features such as EHRs, personalized treatment plans, and telemedicine. Accurate requirements are crucial for successful development and implementation. The next steps involve reviewing and approving the SRS, followed by design, development, testing, and deployment of the HMS. The document will be updated as needed to reflect changes and ensure alignment with stakeholder needs. We appreciate the contributions of all stakeholders in shaping these requirements.

**10. REQUIREMENT TRACEABILITY MATRIX**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Req. ID** | **Requirement Description** | **Category** | **Source** | **Design Specification** | **Test Cases** | **Status** |
| HMS-001 | Patient registration and management | Functional Requirement | Stakeholders | Design Doc-1 | TC-001 | Implemented |
| HMS-002 | Doctor information management | Functional Requirement | Stakeholders | Design Doc-2 | TC-002 | Implemented |
| HMS-003 | Appointment scheduling system | Functional Requirement | Stakeholders | Design Doc-3 | TC-003 | In Progress |
| HMS-004 | Prescription and treatment management | Functional Requirement | Stakeholders | Design Doc-4 | TC-004 | In Progress |
| HMS-005 | Billing and payment system | Functional Requirement | Business Rules | Design Doc-5 | TC-005 | Pending |