Software Requirements Specification

–

Author(s): Poornima R Nair

Date: December 23, 2024

Contents

1. Abstract

Hospital Management System is a console-based application designed to streamline and automate key operations within a hospital. The system offers role-based access for two main user categories: Admin and Doctor. The Admin user is responsible for managing doctor and patient records, scheduling and modifying appointments, and overseeing the inventory of medical equipment. Doctors can view their assigned appointments and add or update patient treatment records.The application ensures efficient data management through secure authentication, input validation, and centralized data storage. With its service-oriented architecture and clear separation of responsibilities, the system is scalable and maintainable, providing a solution that supports modern healthcare environments. This software enhances hospital efficiency, improves communication between admin and doctors, and ensures secure management of hospital resources.

2. Modules

2.1 Authentication Module

The Authentication Module is the backbone of the application, ensuring secure access to the system. It includes a login system where users authenticate using a username and password. Role-based access control is implemented to distinguish between Admins and Doctors, ensuring users can only access functionalities relevant to their roles. The module also manages user sessions, maintaining secure and uninterrupted interactions during active use while logging out inactive sessions to prevent unauthorized access.

2.2 Admin Module

The Admin Module provides comprehensive tools for managing the hospital's key resources. Admins can perform CRUD (Create, Read, Update, Delete) operations on patient and doctor records, ensuring data is up-to-date and accurate. The appointment management system allows admins to add, view, update, and cancel appointments, providing flexibility in

scheduling. Additionally, the equipment management feature enables admins to view and manage hospital resources, ensuring operational efficiency and resource availability.

2.3 Doctor Module

The Doctor Module equips doctors with tools to manage their daily activities efficiently. In addition to viewing appointments and streamlining schedules, this module allows doctors to manage patient medical records effectively. Doctors can securely view, add, and update patient medical records, ensuring seamless collaboration with patients while maintaining the confidentiality and integrity of sensitive medical data.

2.4 Core Services

Core Services encompass the essential backend functionalities supporting the application. A Logger Service ensures real-time system monitoring and debugging, enhancing application reliability. The Authentication Service handles secure user login and role management. Admin and Doctor Services implement the respective role-specific business logic, ensuring smooth operation of their functionalities. These services follow a service-oriented architecture, enabling scalability and easy maintenance.

3. Functional Requirements

3.1 User Authentication

The User Authentication module ensures secure access to the application by validating users with a username and password. It supports two distinct user roles—Admin and Doctor—each with tailored access to role-specific functionalities. The system securely maintains active user sessions, preventing unauthorized access while enabling smooth navigation for authenticated users. Additionally, the application provides a secure logout mechanism to protect sensitive data and ensure compliance with best practices in user authentication.

3.2 Admin Functionalities

The Admin functionalities provide extensive tools to manage hospital operations efficiently. Admins can view, add, edit, and delete records for doctors, ensuring accurate and up-to-date information. They can also schedule new appointments, update existing ones, and cancel appointments when needed. A comprehensive view of all appointments allows Admins to manage schedules effectively. Moreover, Admins can oversee and manage the hospital's medical equipment inventory, ensuring resources are appropriately allocated and maintained for uninterrupted services.

3.3 Doctor Functionalities

Doctors have access to functionalities that streamline their daily activities and enhance patient care. They can view all their assigned appointments, helping them prepare for consultations and manage their time effectively. Additionally, doctors can securely view, add, and update patient medical records, enabling them to maintain comprehensive and accurate documentation. This module supports a structured workflow while ensuring the delivery of quality medical care and safeguarding sensitive patient information.

3.4 Logging Requirements

The Logging module tracks and records system activities to ensure transparency and accountability. It maintains a detailed log of errors and user actions, providing an audit trail for debugging and monitoring. By tracking system usage, the application can identify anomalies and optimize performance. This module enhances the overall reliability and maintainability of the application, ensuring that critical events are logged and retrievable when needed.

4. Non – Functional Requirements

4.1 Performance Requirements

The console-based Hospital Management Application must execute operations swiftly and efficiently, ensuring that users experience minimal delays. Actions such as adding, retrieving, or updating records should complete within 1 second, even under moderate loads. The

system should be capable of managing up to 1,000 records per module (patients, doctors, appointments, etc.) without any noticeable performance degradation.

4.2 Usability Requirements

The application should provide a simple and intuitive interface suitable for users with varying technical expertise. Role-specific menu options must be clearly presented and easy to navigate. The app should offer helpful feedback for successful actions, such as confirmation messages, and meaningful error messages like "Invalid input: Please use DD/MM/YYYY format" to guide users in correcting mistakes.

4.3 Reliability and Availability

The application must function consistently and remain available for users at all times. It should be designed to handle unexpected inputs or actions without crashing. To ensure data is not lost during unforeseen interruptions, the system must save changes to a file or database immediately after critical operations like adding or updating records

4.4 Efficiency

The application should utilize system resources efficiently, ensuring it remains lightweight and suitable for low-powered environments. It must avoid resource-intensive operations such as excessive file I/O or complex loops, and it should execute commands with minimal CPU and memory usage.

4.5 Scalability

Though initially designed for small-scale operations, the application must support future growth. It should accommodate additional modules like Lab Management or Pharmacy without requiring significant code restructuring. Moreover, it should handle an increased volume of data while maintaining consistent performance.

4.6 Maintainability and Modularity

The system should follow a modular design, separating functionalities like authentication, patient management, and appointment booking into distinct classes and methods. This modularity, combined with comprehensive inline comments and documentation, ensures that future developers can easily update, debug, or extend the application as needed.

5. High-Level Design

Here's a clear and precise structure for the Hospital Management System's main interface:

- Welcome Screen: Users are authenticated based on their credentials, and their roles are determined dynamically. Roles such as Admin or Doctor are returned upon successful authentication.
- Role-Based Main Menu:
  - Admin Menu:
    - Manage Patients
    - Manage Doctors
    - Manage Appointments
    - Manage Equipment
    - Logout
  - Doctor Menu:
    - View Appointments
    - View Patient Medical Records
    - Add Patient Medical Records
    - Update Patient Medical Records
    - Logout
- Action Submenus: Each menu option leads to specific tasks, e.g., "Manage Patients" lets the admin
    - Add Patient
    - Edit Patient
    - Delete Patient
    - View Patients

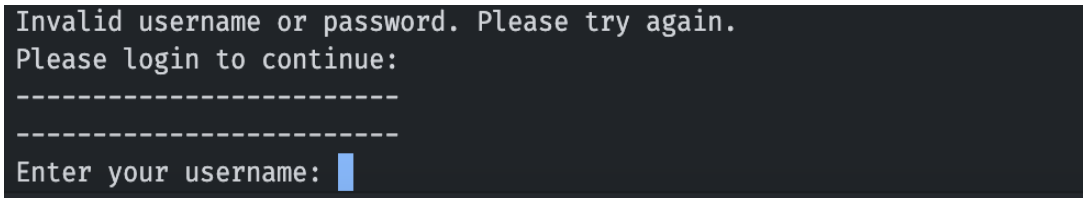
5.1 Login screen

```
----------------------------------------
Welcome to the Hospital Management System!
----------------------------------------
Please login to continue:
------------------------
------------------------
Enter your username:
```

5.2 Admin menu

```
Login successful! Welcome, Admin!
----------------------------------------
Admin Menu
----------------------------------------
1. View All Patients
2. Add Patient
3. Edit Patient
4. Delete Patient
5. View All Doctors
6. Add Doctor
7. Edit Doctor
8. Delete Doctor
9. Add Appointment
10. View All Appointments
11. Update Appointments
12. Cancel Appointments
13. View All Equipments
14. Add Equipments
15. Delete Equipment
0. Logout
----------------------------------------
Enter your choice:
```

5.3 Error screen



6. Low-Level Design

- User Authentication: The system authenticates users by validating their credentials (username and password). Upon successful login, the system manages user sessions and provides role-based access to specific functionalities, ensuring proper access control. Users can also securely log out when done.

- Admin Module: The Admin module allows for complete management of the hospital's data. This includes adding, updating, and deleting patient and doctor records, scheduling and managing appointments, and overseeing medical equipment. The Admin has full control over the system.

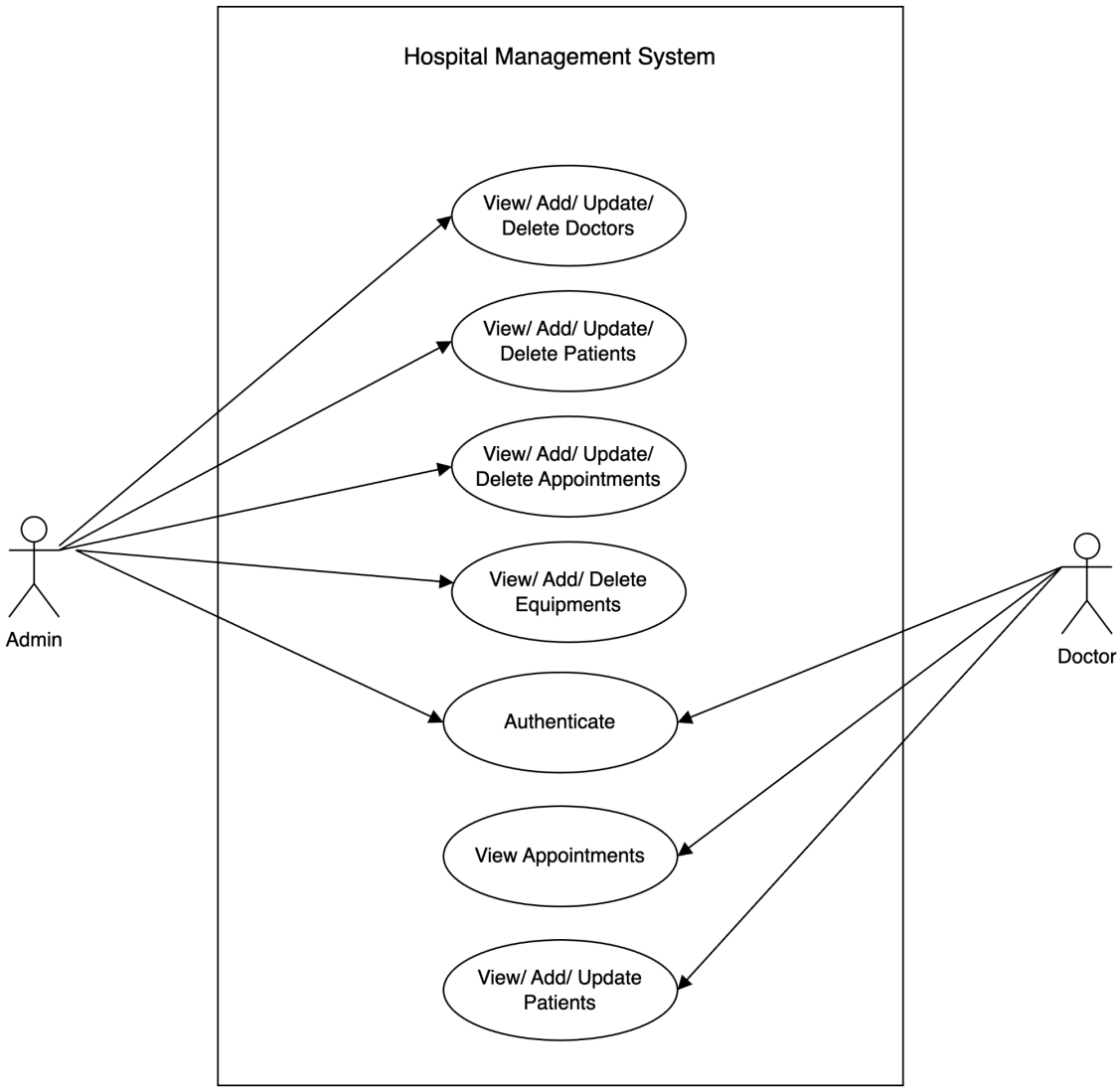
- Doctor Module: The Doctor module provides access to a doctor's assigned appointments, enabling them to view and manage their schedule effectively. It also allows doctors to securely add, view, and update treatment records for their patients, ensuring accurate and comprehensive medical documentation.

- Appointment Management: The system supports the scheduling, updating, and cancellation of appointments. Admin and patients can both book appointments, while doctors and admins can modify or cancel appointments as needed to accommodate changes in the hospital's schedule.

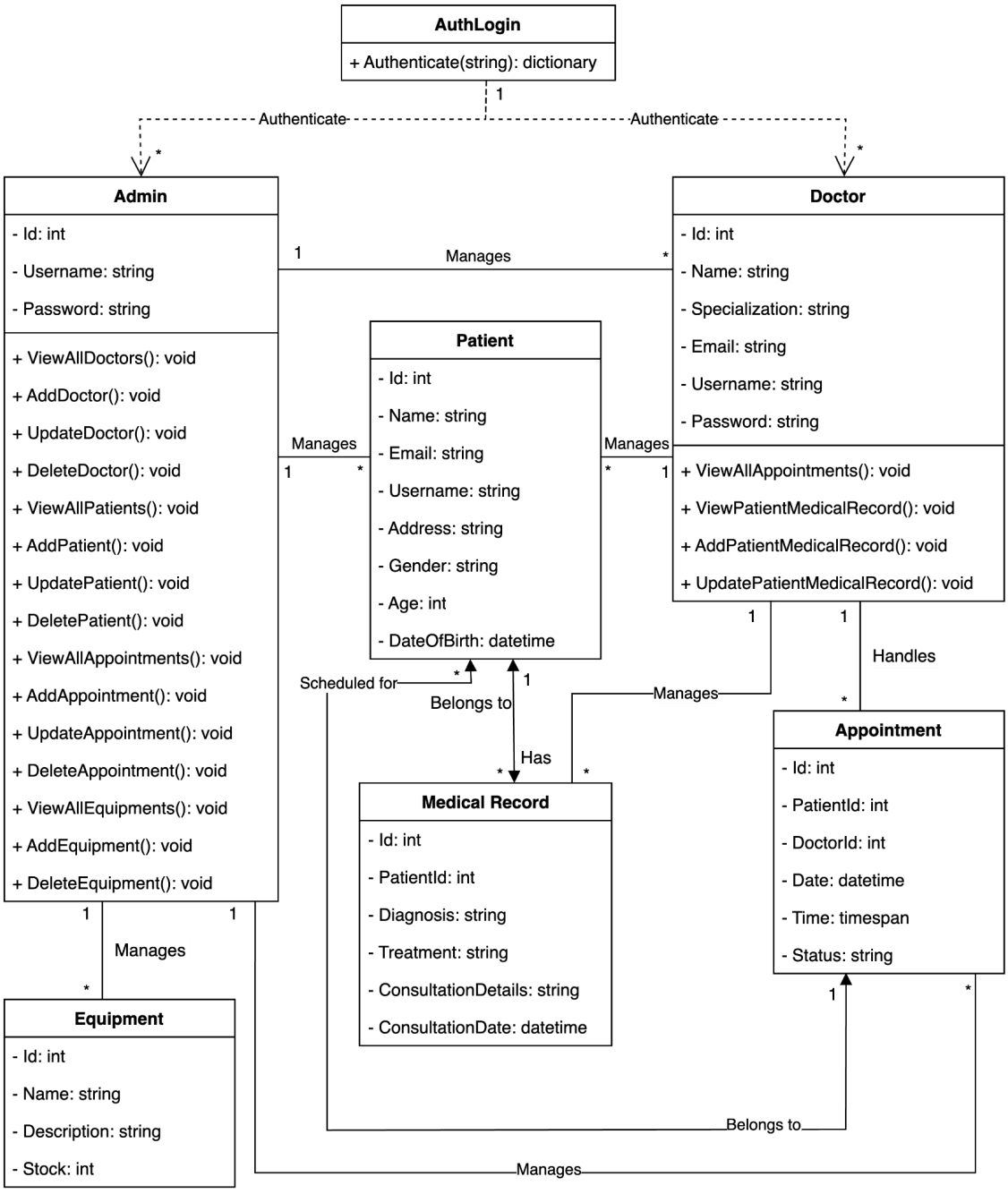
- Equipment Management: This module allows for the management of the hospital's medical equipment. Admins can view the equipment inventory, add new items, and track the usage and condition of the equipment to ensure proper maintenance and availability.

- Main Application Flow: After login, the system displays a role-specific menu based on the user's credentials. Each role (Admin and Doctor) has a tailored menu to access the relevant functionalities, ensuring a seamless and efficient user experience.

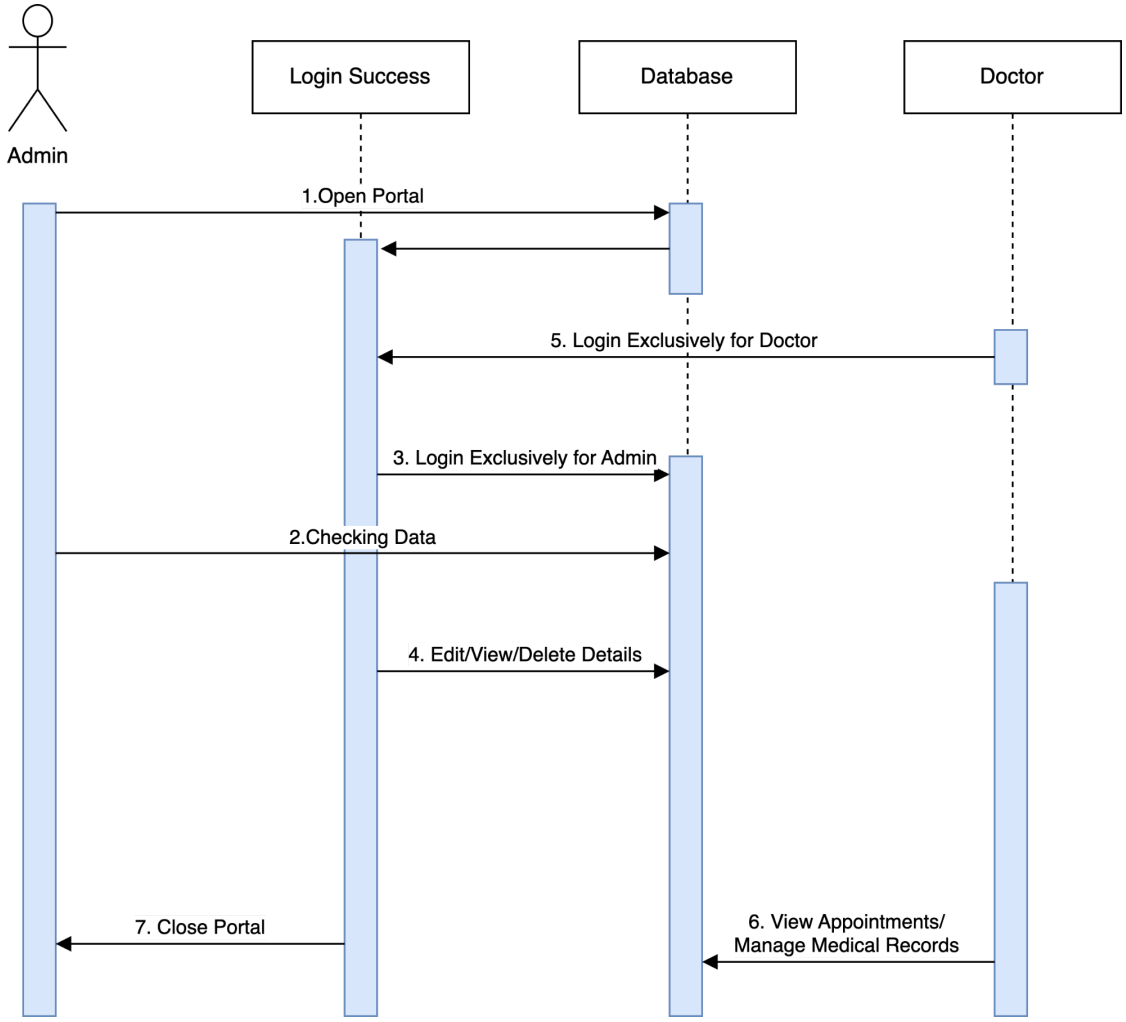
7. Use Case Diagram



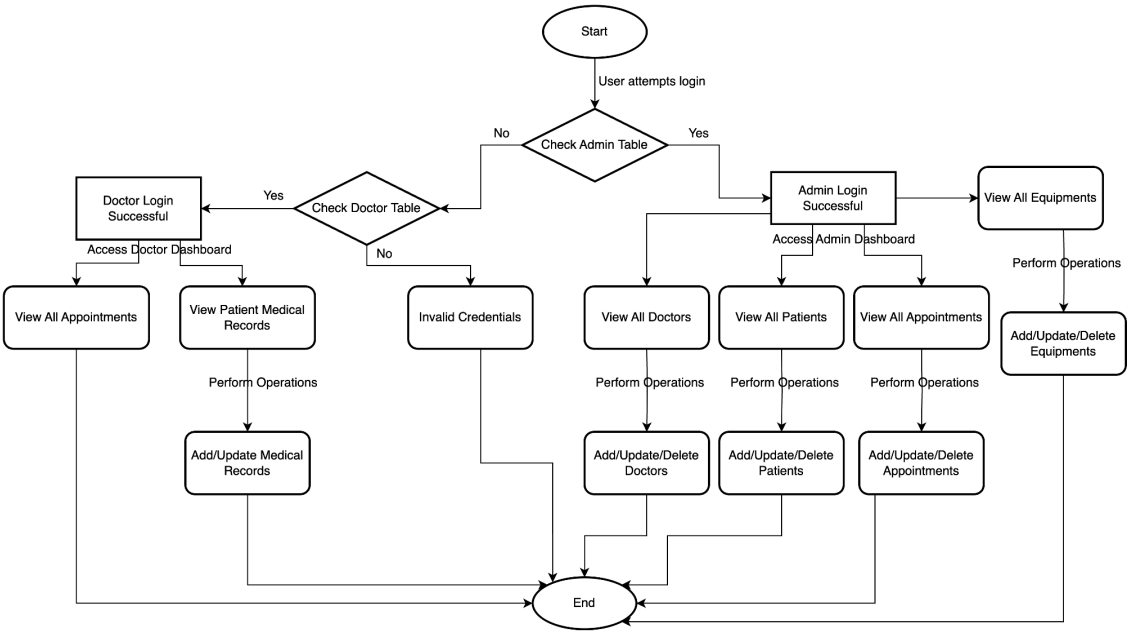
8. Class Diagram



9. Sequence Diagram



Flow Chart


Start
User attempts login
Check Admin Table
No
Yes
Check Doctor Table
Yes
No
Doctor Login Successful
Access Doctor Dashboard
View All Appointments
View Patient Medical Records
Invalid Credentials
Perform Operations
Add/Update Medical Records
Admin Login Successful
Access Admin Dashboard
View All Doctors
View All Patients
View All Appointments
View All Equipments
Perform Operations
Add/Update/Delete Equipments
Perform Operations
Perform Operations
Perform Operations
Add/Update/Delete Doctors
Add/Update/Delete Patients
Add/Update/Delete Appointments
End

10. ER Diagram