Hospital Appointment Scheduling Management System

Document ID: ASP-CMS-SRS-02.doc

Author(s): Janani D P

Version: 0.1

Date: March 27, 2024

1. Abstract

The Hospital Appointment Scheduling System is designed to simplify the process of scheduling and managing appointments for patients in an orthopaedic hospital environment. This system enables users to easily book appointments, view available time slots, and receive reminders for scheduled appointments. Administrators have access to features such as managing time slots, viewing appointments, and sending reminders to patients. Users can login in the system and book appointments in the available time slots. The system ensures smooth coordination between healthcare providers and patients, leading to an efficient scheduling and communication.

2. Objective and Scope

The objective and the scope of the project is to build a Hospital Management System to improve and simplify the administrative procedures in a hospital setting for patients. The scope of the product includes the following basic features:

- Enable users to securely access the Hospital Patient Management System with personalized credentials.
- Facilitate the booking of patient appointments and time slot management.
- Users can view available time slots, select desired appointments, and receive confirmation messages upon successful booking.
- Simplify appointment allocation to healthcare providers within the HPMS.
- Reduce missed appointments by sending timely reminders to patients.

3. Functional requirements

3.1 Login to the system

The system validates the entered information, ensuring that all fields are filled correctly and that the username and email are unique. Then, the system creates a new user account and stores the provided information in the database. Each and every user should be authenticated

with a User Name and Password to login into the system. If the credentials are correct, the user is authenticated and granted access to their account/dashboard.

The admin enters their admin username and password on the login page. The system authenticates the credentials and grants access to the admin dashboard upon successful verification.

3.2Add/Update time slots

For administrators, the system enables easy management of appointment time slots, allowing for the addition, viewing, and updating of available time slots to accommodate patient needs.

3.3View time slots

On the user side, patients can conveniently view and select from updated time slots, providing flexibility in scheduling appointments within the hospital system.

3.4Appointment Booking

Patients select a preferred appointment date and time from the available slots and provide their contact information.

Once submitted, the system confirms the appointment and sends a confirmation message to the patient.

3.5Add/View/Delete Appointments

Admins can log into the system and view a list of existing appointments.

They can add new appointments by specifying the patient's details, appointment date, and time.

Users can also cancel their appointments after booking.

3.6Appointment Reminders

Admins can set up the reminder system by configuring the timing and method of reminders.

4. Non-Functional Requirements

The system should support a large number of time slots without slowing down. The user interface for viewing time slots should work well on different devices. Patient information should be encrypted and stored securely. Canceling appointments should not lead to data loss. Users should confirm before deleting appointments to avoid mistakes. Access to appointment details should be restricted to authorized users.

5. Design

5.1High Level Design

For appointment reminders, a background task scheduler will send reminders based on scheduled appointment times, with a notification service handling the actual delivery via email, SMS, or other channels according to user preferences. For the high-level design, a client-server architecture is established. The client-side will consist of a web application accessible to both admins and users. Admins will have additional privileges to manage time slots and appointments. The database will be relational, with tables for users, time slots, appointments, and reminder settings. Additionally, a background task scheduler to handle appointment reminders is incorporated. The system will follow RESTful principles for API design, ensuring scalability and maintainability. Security measures such as encryption, authentication, and authorization will be implemented to protect sensitive data. Finally, the system will be deployed on a cloud platform like AWS or Azure for scalability and availability.

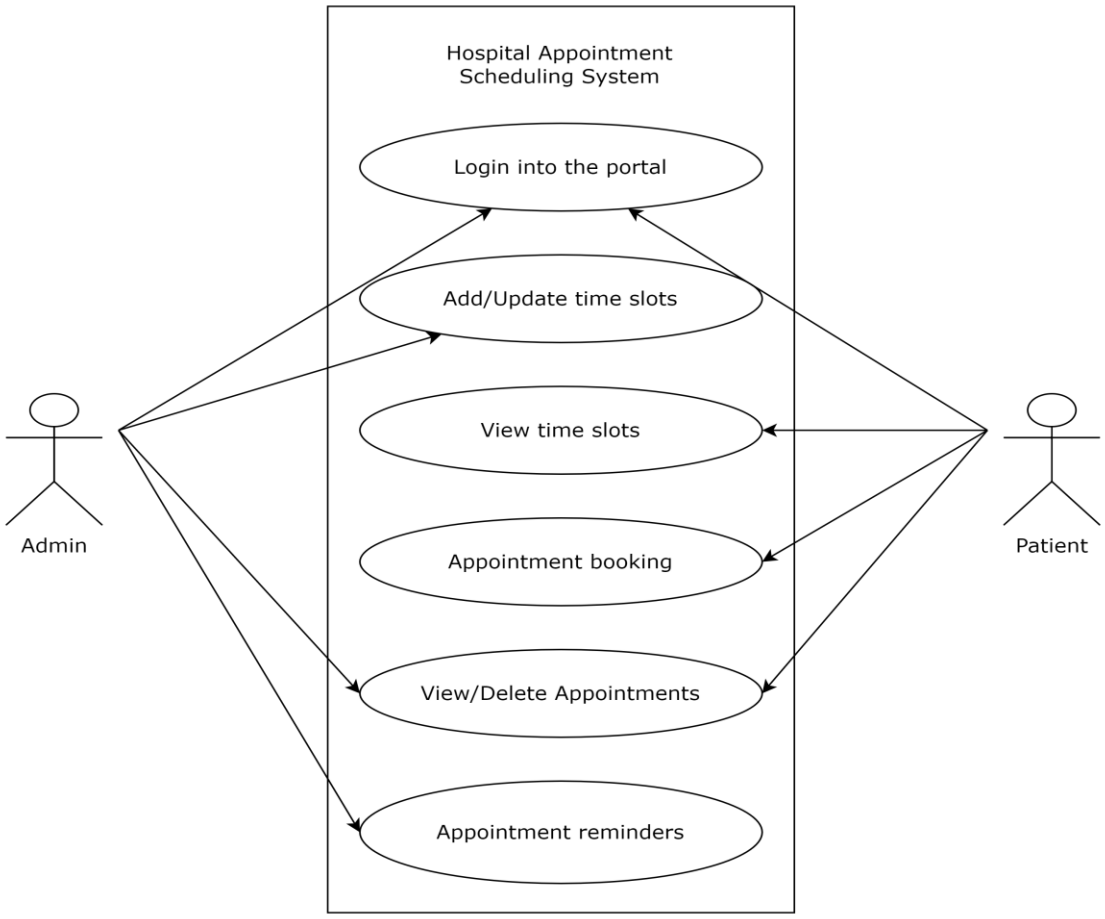
5.2Low Level Design

For the low-level design of the mentioned features, a database schema is established with a table for time slots, including fields for date, start time, end time, and availability. Backend logic will consist of API endpoints to handle operations such as adding, updating, and viewing time slots. These endpoints will validate inputs and interact with the database accordingly. The frontend interface will present available time slots to users in a user-friendly format. For

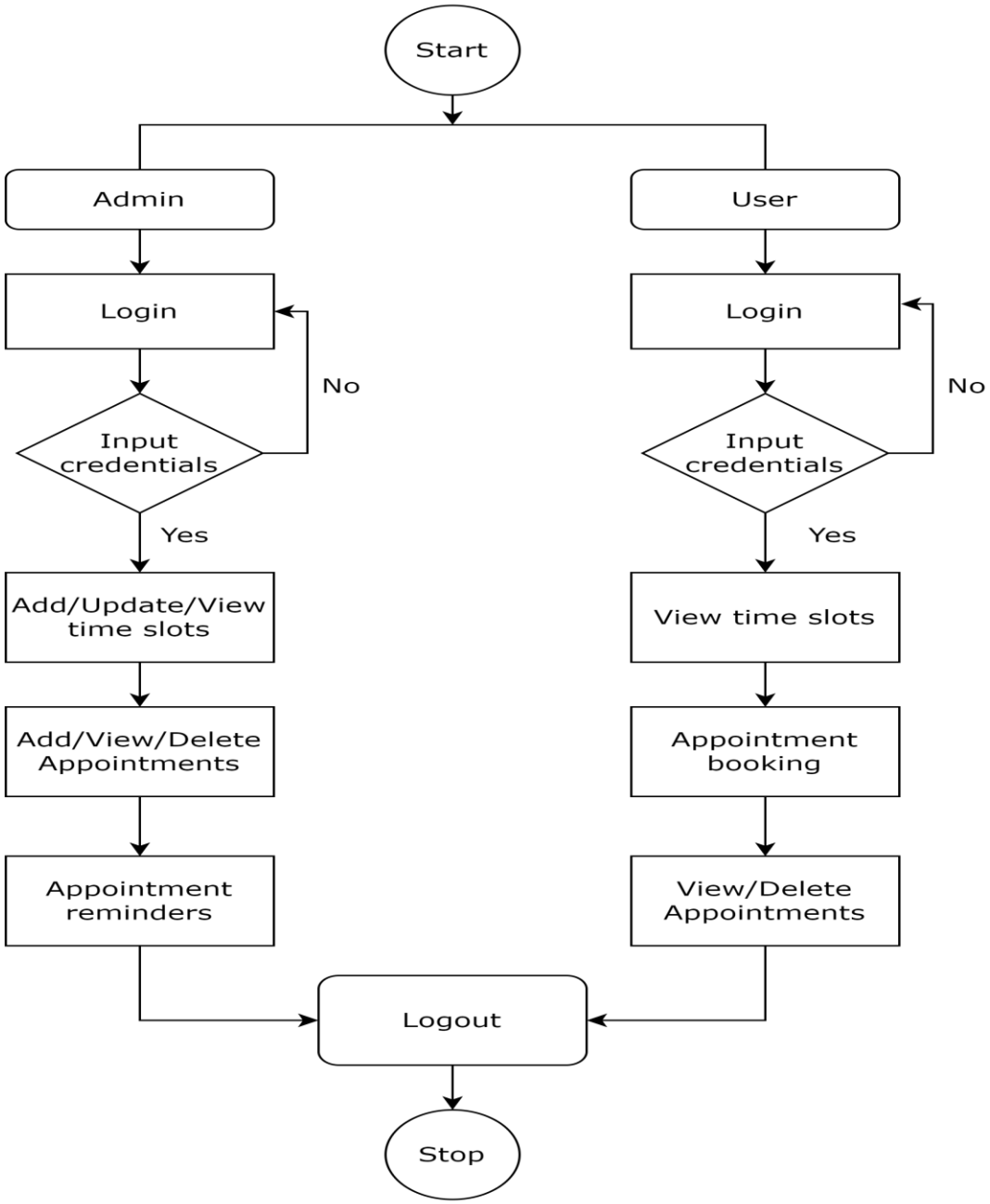
appointment booking, another API endpoint will validate inputs, check availability, and store booking details in the database. The frontend will provide a booking interface for users to select time slots and provide necessary information. To view appointments, a backend endpoint will retrieve booked appointments for a specific user, while the frontend will display them in a readable format. Deleting appointments will involve backend logic to cancel appointments and remove corresponding records from the database, with the frontend providing a cancellation interface with a confirmation prompt.

6. Diagrams

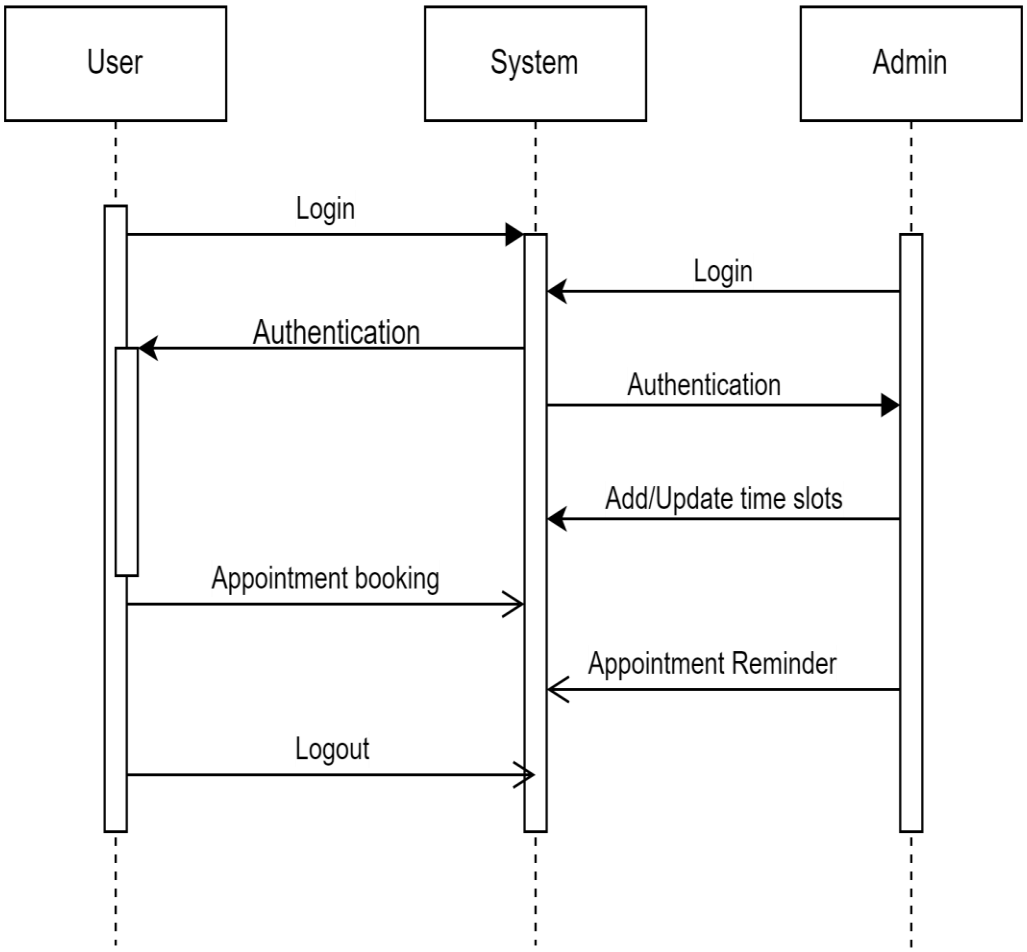
6.1 Use Case Diagram



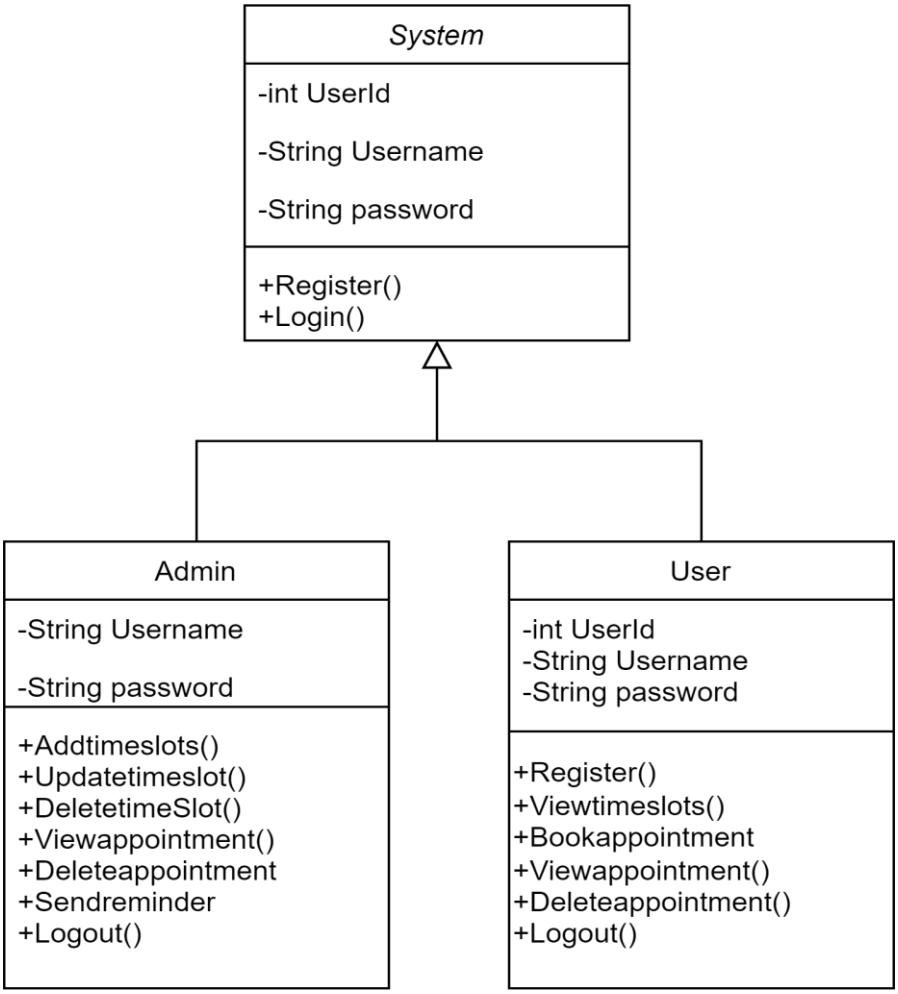
6.2Flow Diagram



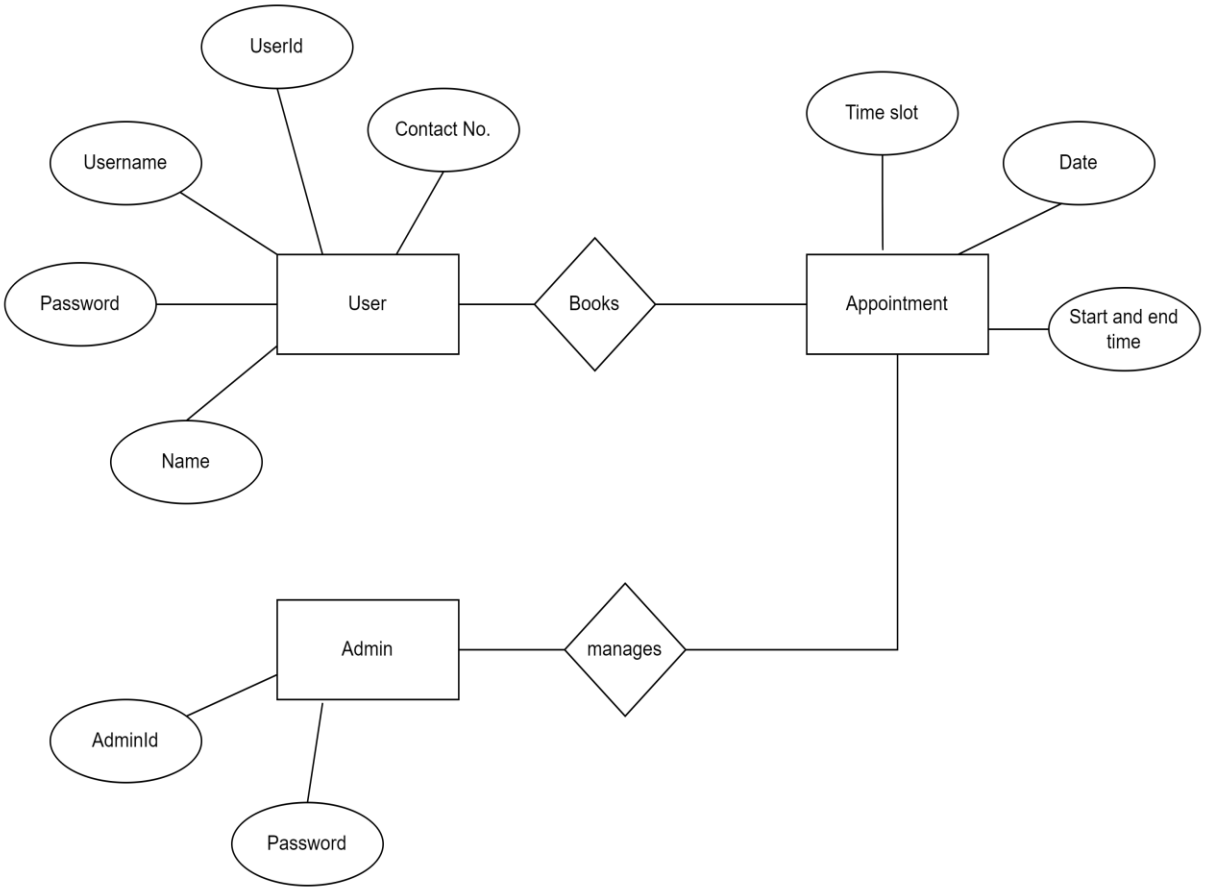
6.3Sequence Diagram



6.4Class Diagram



6.5Class Diagram



7. Test Cases

Test case	Test purpose	Test condition	Expected outcome	Actual Result
User Registration	Verify successful user registration	All required fields filled correctly	Successful registration with user data stored	User account is created is created and redirected to login page
User login	Confirm successful user login	Correct credentials entered	Successful login redirects user to dashboard	Authentication is successful and user is redirected to dashboard
Login error	Verify error message for incorrect login credentials	Incorrect username or password	Error message guides user to enter correct credentials	Authentication is successful
Availability of Time Slots	Ensure availability of time slots for booking	User accesses appointment booking page	Available time slots are displayed for selection	Time slots displayed for selection
Appointment Booking	Test successful booking of an appointment	User selects available time slot and confirms booking	Successful booking results in confirmation message	User successfully booked an appointment
View Appointments	Confirm visibility of upcoming appointments	User accesses appointment view page	Upcoming appointments are displayed in a list	User able to view upcoming appointments

Appointment Cancellation	Test cancellation of an existing appointment	User selects appointment to cancel	Successful cancellation removes appointment from list	User successfully cancelled an appointment, and it was removed from the appointment list
Appointment Reminders	Verify receipt of appointment reminders	User receives reminder before scheduled appointment	Reminder notification is sent according to user preference	User receives a reminder notification

8. Conclusion

In conclusion, the Hospital Appointment Scheduling Management System offers a streamlined solution for managing orthopaedic appointments, providing patients with a user-friendly experience. The system allows users to schedule appointments efficiently, view available time slots, and receive reminders for scheduled appointments. Additionally, administrators can access specialized features such as viewing their profile and managing booked time slots, enhancing the overall functionality of the system. With its robust features, the system aims to improve the efficiency of appointment management in a hospital environment, ultimately contributing to enhanced patient care and satisfaction.