

HOSPITAL APPOINTMENT BOOKING SYSTEM

TABLE OF CONTENTS

1. Abstract
2. Introduction
3. Problem Statement
4. Objectives of the Project
5. Scope of the Project
6. Literature Survey
7. Existing System
8. Proposed System
9. System Architecture
10. System Flow Diagram
11. ER Diagram
12. Database Design
13. Module Description
14. Technology Stack
15. API Design
16. User Interface Design
17. Testing
18. Advantages of the System
19. Limitations
20. Future Enhancements
21. Conclusion
22. References

1. ABSTRACT

The Hospital Appointment Booking System is a comprehensive web-based application designed to automate and streamline the process of scheduling medical appointments in hospitals and healthcare centers. In many traditional hospital environments, appointment management is still handled through manual registers or partially digital systems. These approaches often result in long patient waiting times, appointment overlaps, poor resource utilization, and administrative inefficiencies.

This project aims to overcome these challenges by providing a centralized, user-friendly platform that efficiently connects patients, doctors, and hospital administrators. The system allows patients to register securely, log in, and book appointments with doctors by selecting the appropriate department, available doctors, and preferred time slots. By offering real-time availability information, the system minimizes scheduling conflicts and improves patient convenience.

Doctors are provided with a dedicated dashboard where they can view their scheduled appointments, manage availability, update consultation status, and access patient appointment details. This helps doctors plan their schedules effectively and ensures better patient flow management within the hospital.

Administrators play a key role in maintaining the system. Through the admin module, hospital staff can manage doctor profiles, patient records, department information, and appointment schedules. Administrators can monitor overall system activity, generate reports, and ensure smooth coordination between patients and doctors.

From a technical perspective, the backend of the system is developed using Java Spring Boot, which provides robust support for RESTful APIs, security, scalability, and transaction management. Spring Boot ensures efficient handling of concurrent users and smooth integration with the database. The frontend is developed using Angular, enabling a responsive and dynamic user interface with seamless navigation and real-time data updates.

Overall, the Hospital Appointment Booking System enhances operational efficiency, reduces manual effort, improves patient experience, and ensures better healthcare service delivery. The system is scalable, secure, and can be extended in the future to include features such as online payments, notifications, electronic medical records, and analytics, making it suitable for modern healthcare institutions.

2. INTRODUCTION

Healthcare institutions such as hospitals and clinics require efficient and reliable systems to manage patient appointments and ensure smooth patient flow. In many traditional hospital environments, appointment scheduling is still handled using manual registers or basic semi-digital systems. These methods often result in long waiting times, appointment overlaps, data inconsistencies, and inefficient utilization of medical resources, which negatively impact both patients and healthcare providers.

Manual appointment systems make it difficult to maintain accurate records, especially when handling many patients daily. Changes such as appointment cancellations, rescheduling, or doctor unavailability are not updated in real time, leading to confusion, overcrowding, and poor patient experience. Additionally, administrative staff spend a significant amount of time managing appointments manually, increasing the chances of human error and operational delays.

The Hospital Appointment Booking System addresses these challenges by introducing a fully digital, web-based solution that automates the appointment scheduling process. The system enables patients to book appointments online by selecting departments, doctors, and available time slots, ensuring transparency and convenience. Real-time appointment management helps prevent scheduling conflicts and reduces patient waiting time within the hospital.

From the hospital's perspective, the system improves workflow efficiency by providing doctors with clear schedules and allowing administrators to monitor appointments, patient records, and doctor availability through a centralized platform. By reducing manual intervention, the system enhances data accuracy, improves operational efficiency, and supports better decision-making.

Overall, the Hospital Appointment Booking System improves the quality of healthcare services by offering a reliable, scalable, and user-friendly appointment management solution. It ensures better coordination between patients, doctors, and hospital administrators, making it highly suitable for modern healthcare institutions and aligned with the objectives of digital transformation in healthcare.

3. PROBLEM STATEMENT

In many hospitals, appointment booking is still managed through manual or semi-digital processes. Patients are often required to physically visit the hospital to register and book appointments, which is inconvenient and time-consuming, particularly for elderly patients and those residing in remote areas. This manual approach results in long queues, increased waiting times, and overall dissatisfaction among patients.

Doctors also face scheduling conflicts due to improper appointment tracking and the absence of real-time updates. Without a centralized appointment management system, overlapping appointments and inefficient time utilization are common. Furthermore, reliance on manual paperwork increases the risk of data inconsistency, record loss, and human errors, making it difficult to maintain accurate patient and appointment records.

Therefore, there is a pressing need for a reliable, automated system that can efficiently manage hospital appointments, reduce manual workload, and improve the overall quality of healthcare services.

4. OBJECTIVES OF THE PROJECT

The primary objective of the **Hospital Appointment Booking System** is to automate and simplify the appointment scheduling process in healthcare institutions. The specific objectives:

- To automate the hospital appointment booking process using a web-based system.
- To reduce patient waiting time by enabling online appointment scheduling.
- To improve hospital workflow efficiency through systematic appointment management.
- To provide secure, accurate, and reliable data storage for patient and appointment information.
- To eliminate manual paperwork and minimize human errors in appointment management.

5. SCOPE OF THE PROJECT

The scope of the Hospital Appointment Booking System defines the functional boundaries and capabilities of the application. The system will include the following features:

- Online appointment booking for patients through a web interface.
- Role-based access control for Administrator, Doctor, and Patient.
- Real-time display of doctor availability and time slots.
- Appointment status tracking (booked, completed, cancelled).
- Secure user login and registration with authentication and authorization.

The system is designed to be scalable and can be enhanced in the future with additional features such as:

- Payment integration.
- Automated notifications and reminders.
- Integration with electronic medical records (EMR).

6. LITERATURE SURVEY

A review of existing hospital management systems reveals that most applications primarily focus on patient record management and billing rather than appointment scheduling. While some online appointment booking systems exist, they often lack flexibility, real-time updates, and proper role-based access control.

Many current solutions fail to efficiently handle appointment conflicts or provide a centralized platform for administrators, doctors, and patients. Security and data consistency are also major concerns in these systems.

The proposed Hospital Appointment Booking System addresses these limitations by offering:

- Real-time appointment management.
- Secure authentication and role-based access.
- Improved usability and system reliability.

7. EXISTING SYSTEM

The existing hospital appointment system relies heavily on manual processes. Appointments are recorded in physical registers or basic software tools, leading to several drawbacks:

- Manual register-based appointment booking.
- Long queues and increased patient waiting time.
- No online access for patients to book appointments.
- High possibility of appointment overlaps and scheduling conflicts.
- Increased dependency on paperwork and manual record maintenance.

These limitations negatively affect both patients and hospital staff, reducing overall efficiency and service quality.

8. PROPOSED SYSTEM

The proposed **Hospital Appointment Booking System** is a web-based, centralized solution designed to overcome the limitations of the existing system. It enables patients to book appointments online without physically visiting the hospital.

Key Features:

- Centralized web-based appointment management.
- Online appointment booking with department-wise doctor selection.
- Real-time time slot availability to avoid scheduling conflicts.
- Secure authentication and role-based access control.
- Efficient management of patients, doctor, and appointment data.

Benefits:

- Improved transparency in appointment scheduling.
- Reduced patient waiting time.
- Enhanced hospital workflow efficiency.
- Better healthcare service delivery through automation.

9. SYSTEM ARCHITECTURE

The System Architecture of the Hospital Appointment Booking System defines the overall structure of the application and the interaction between its components. The system adopts a three-tier architecture, which ensures modularity, scalability, security, and ease of maintenance.

The three main layers of architecture are:

- **Presentation Layer (Frontend)**
- **Application Layer (Backend)**
- **Database Layer**

9.1 Presentation Layer (Frontend)

The Presentation Layer is responsible for user interaction and interface design. It is developed using Angular, which provides a dynamic and responsive web interface.

Responsibilities:

- Secure user registration and login.
- Online appointment booking based on department and doctor availability.
- Display of appointment status (booked, completed, cancelled).
- Profile management for different roles (Administrator, Doctor, Patient).

Communication:

Angular communications with the backend through **RESTful APIs** using HTTP methods such as **GET, POST, PUT, and DELETE**. This ensures real-time updates and enhanced user experience.

9.2 Application Layer (Backend)

The Application Layer is the core of the system, developed using Java Spring Boot. It handles business logic, request processing, authentication, and authorization.

Responsibilities:

- Validating user requests.
- Managing appointment scheduling logic.
- Preventing appointment conflicts.
- Handling role-based access control.
- Securing APIs using authentication mechanisms.
- Communicating with the database layer.

Spring Boot exposes **REST APIs** that act as a bridge between the frontend and the database, ensuring smooth data flow and system reliability.

9.3 Database Layer

The Database Layer is responsible for storing and managing system data. It maintains records related to:

- Patients
- Doctors
- Administrators
- Appointments
- Departments
- Login credentials

A relational database such as MySQL is used to ensure data consistency, integrity, and efficient retrieval. The backend communicates with the database using Spring Data JPA, which simplifies database operations and reduces boilerplate code.

9.4 Architecture Workflow

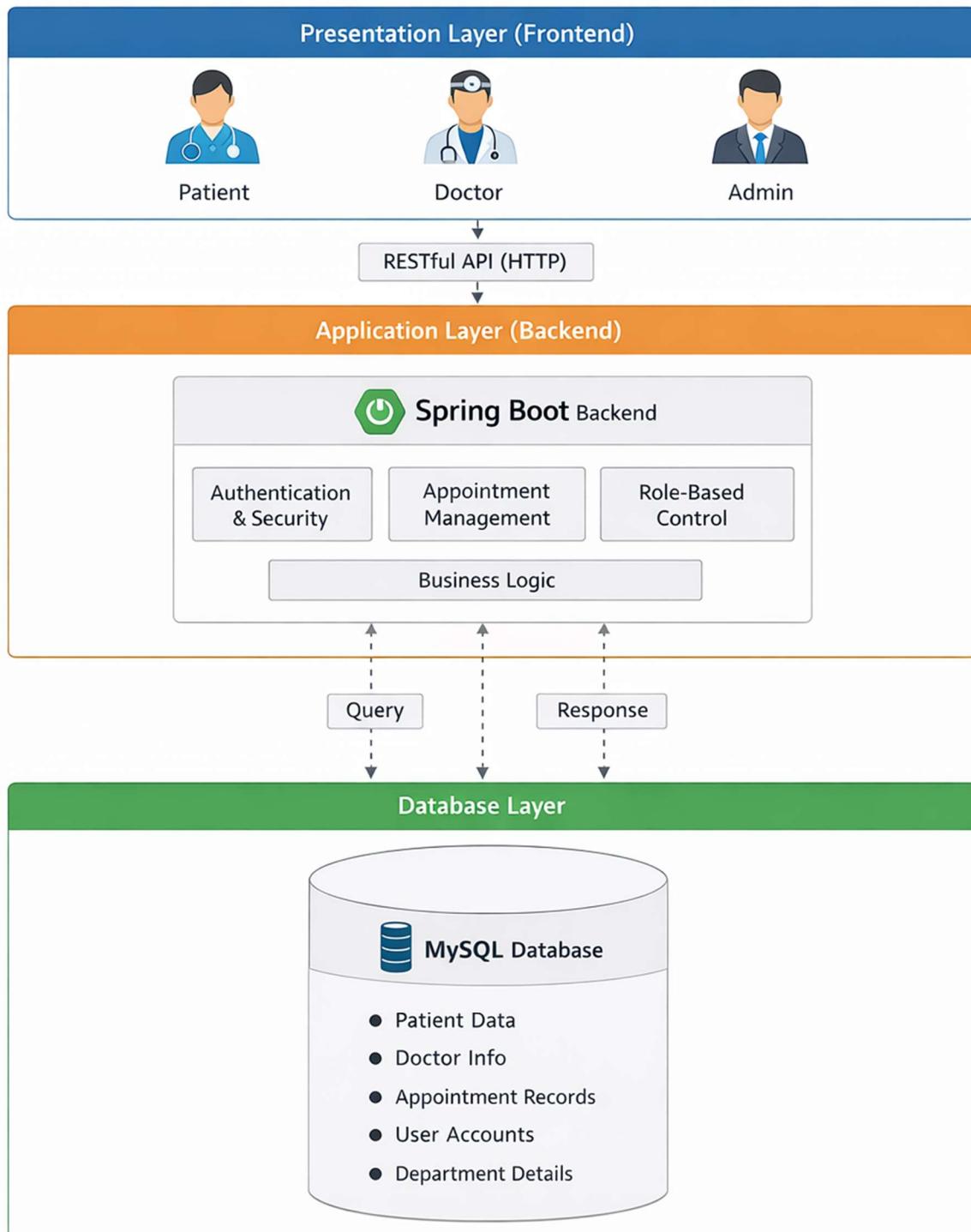
The workflow of the system can be summarized as follows:

1. The user accesses the application through a web browser.
2. Requests are sent from the Angular frontend to the Spring Boot backend via REST APIs.
3. The backend processes the request, applies business rules, and interacts with the database.
4. The database returns the requested data to the backend.
5. The backend sends the response back to the frontend.
6. The front end displays the result to the user in real time.

9.5 Advantages of Architecture

- **Separation of concerns** improves maintainability.
- **Scalable and flexible design** supports future enhancements.
- **Secure data handling** ensures patient confidentiality and system reliability.
- **Easy integration** with additional modules such as payment gateways or notifications.
- **Multi-user support** allows simultaneous access by administrators, doctors, and patients.

Hospital Appointment Booking System



10. SYSTEM FLOW DIAGRAM

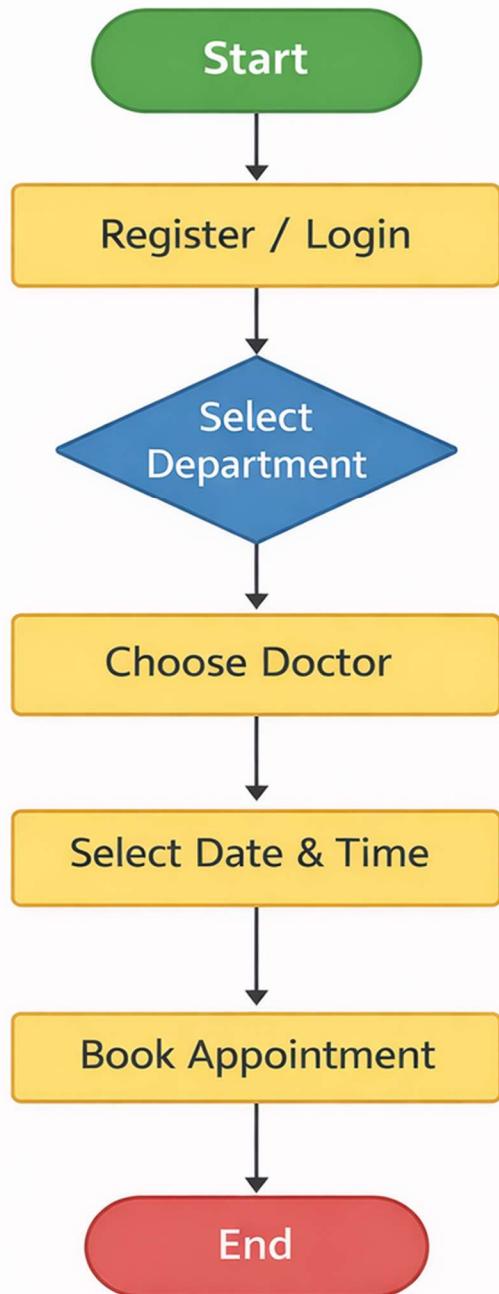
The **System Flow Diagram** represents the logical flow of activities performed by different users in the Hospital Appointment Booking System. It explains how data and control move from one step to another during the appointment booking process. This diagram helps in understanding the working of the system in a clear and structured manner.

10.1 Patient Flow

The Patient Flow describes the sequence of actions performed by a patient while using the system:

1. **Register / Login**
 - The patient registers by providing personal details such as name, email, contact number, and password.
 - If you have already registered, the patient logs in using valid credentials.
2. **Select Department**
 - After successful login, the patient selects the required medical department (e.g., Cardiology, Orthopedics, General Medicine).
3. **Choose Doctor**
 - Based on the selected department, the system displays a list of available doctors.
 - The patient selects a preferred doctor.
4. **Select Date and Time Slot**
 - The system displays available dates and time slots for the selected doctor.
 - The patient chooses a convenient slot.
5. **Book Appointment**
 - Once the slot is selected, the appointment is confirmed and stored in the database.
 - The patient can view the appointment status.

SYSTEM FLOW DIAGRAM:



11. ENTITY RELATIONSHIP (ER) DIAGRAM

The Entity Relationship (ER) Diagram represents the data model of the Hospital Appointment Booking System. It shows the entities involved in the system and the relationships between them. The ER diagram helps in understanding how data is stored, related, and managed in the database.

11.1 Entities

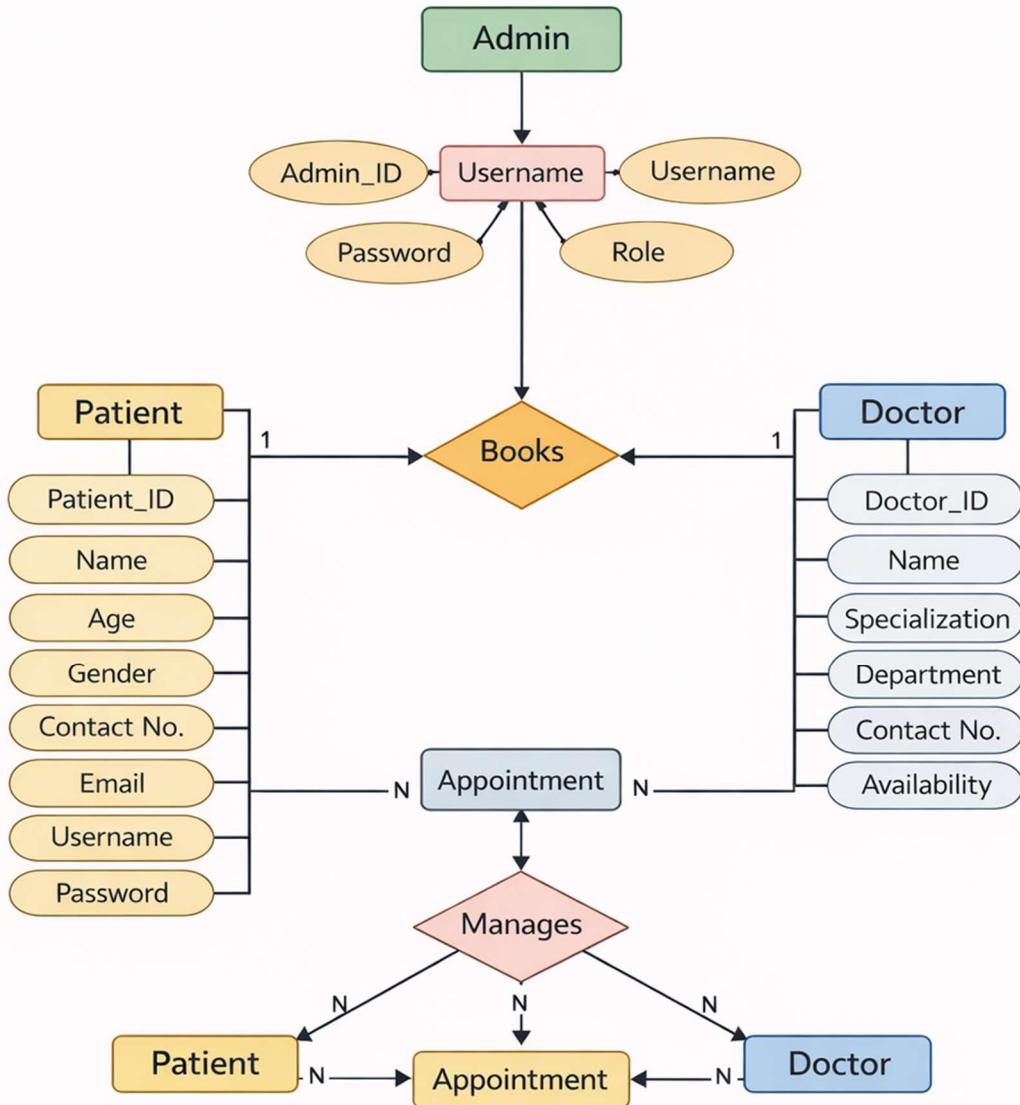
Patient <ul style="list-style-type: none">• Patient_ID (Primary Key)• Name• Age• Gender• Contact Number• Email• Username• Password	Doctor <ul style="list-style-type: none">• Doctor_ID (Primary Key)• Name• Specialization• Department• Contact Number• Availability
Appointment <ul style="list-style-type: none">• Appointment_ID (Primary Key)• Appointment Date• Time Slot• Status• Patient_ID (Foreign Key)• Doctor_ID (Foreign Key)	Admin <ul style="list-style-type: none">• Admin_ID (Primary Key)• Username• Password• Role

11.2 Relationships

- Patient → Appointment (One-to-Many) One patient can book multiple appointments, but each appointment belongs to only one patient.

- Doctor → Appointment (One-to-Many) One doctor can handle multiple appointments, but each appointment is assigned to only one doctor.
- Admin → System Management Admin manages doctors, patients, departments, and appointments (administrative control).

11.3 ER Diagram



ER Diagram: Hospital Appointment Booking System

12. DATABASE DESIGN

The database design plays a crucial role in ensuring efficient data storage, retrieval, and management in the Hospital Appointment Booking System. The system uses a relational database model, where data is organized into tables with proper relationships using primary keys and foreign keys. This design ensures data integrity, consistency, and avoids redundancy.

12.1 Patient Table

The Patient table stores all patient-related information required for authentication and appointment booking.

Column Name	Data Type	Description
id	Long	Primary key, uniquely identifies a patient
first_name	String	Patient's first name
last_name	String	Patient's last name
username	String	Unique username for login
password	String	Encrypted password
phone	String	Contact number

12.2 Doctor Table

The Doctor table contains details of doctors available in the hospital.

Column Name	Data Type	Description
id	Long	Primary key, uniquely identifies a doctor
name	String	Doctor's name
department	Enum	Medical department (e.g., Cardiology)
phone	String	Contact number

12.3 Appointment Table

The Appointment table stores appointment-related information and acts as a linking table between Patient and Doctor.

Column Name	Data Type	Description
id	Long	Primary key
patient_id	FK	Foreign key referencing Patient table
doctor_id	FK	Foreign key referencing Doctor table
date	String	Appointment date
time	String	Appointment time slot
status	Enum	Appointment status (Booked, Completed, Cancelled)

13. MODULE DESCRIPTION

The system is divided into multiple modules based on user roles. Each module performs specific operations to ensure smooth functioning of the system.

13.1 Admin Module

The Admin module provides complete control over the system.

Functions:

- Admin Login
- View of registered doctors
- View of registered patients
- View all appointments
- Monitor system activities

This module ensures proper system supervision and management.

13.2 Patient Module

The Patient module allows patients to interact with the system and manage their appointments.

Functions:

- Patient registration and login
- Book appointments based on doctor availability
- View appointment history
- Cancel appointments

This module improves patient convenience by enabling online appointment booking.

13.3 Doctor Module

The Doctor module helps doctors manage their schedules efficiently.

Functions:

- Doctor registration and login
- View scheduled appointments
- Update appointment status (Completed / Cancelled)

This module helps doctors manage patient flow and consultation timing.

14. TECHNOLOGY STACK

The Hospital Appointment Booking System is developed using modern web technologies to ensure scalability, security, and performance.

Layer	Technology
Frontend	Angular, Bootstrap
Backend	Java, Spring Boot
Database	H2 / MySQL
Tools	Visual Studio Code, Postman

15. API DESIGN (SAMPLE)

The system follows a RESTful API architecture, allowing smooth communication between frontend and backend.

Sample APIs:

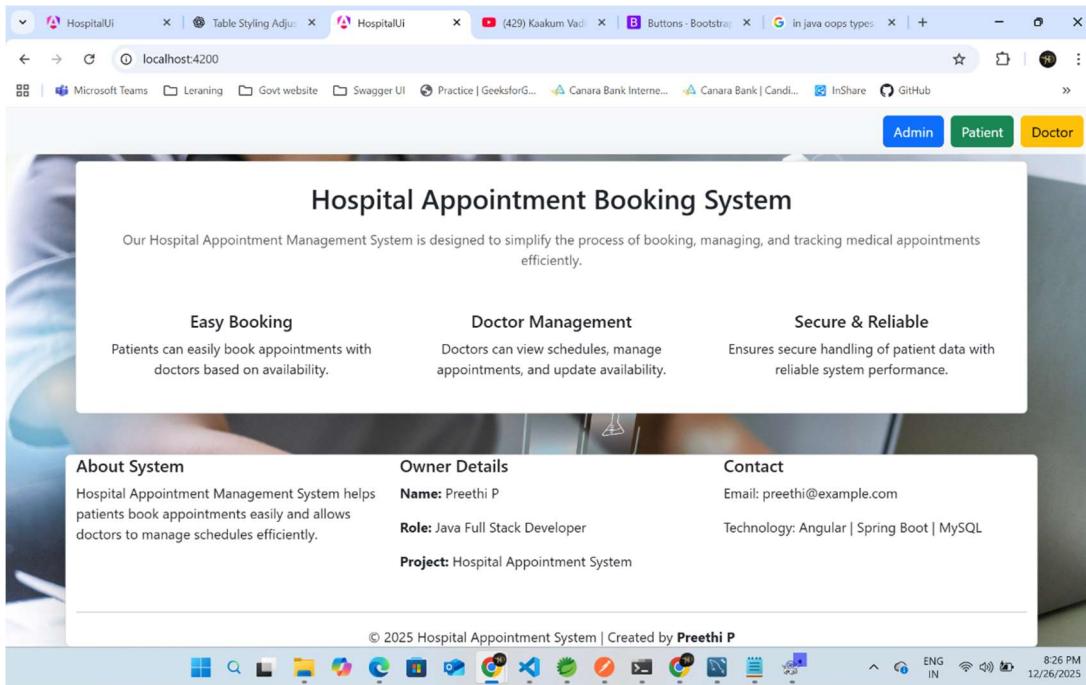
- **POST /hospital/appointments/book** → Used to book a new appointment
- **GET /hospital/appointments/slots** → Fetch available appointment slots
- **GET /hospital/doctors/by-department** → Retrieve doctors based on department

These APIs ensure modularity and easy integration.

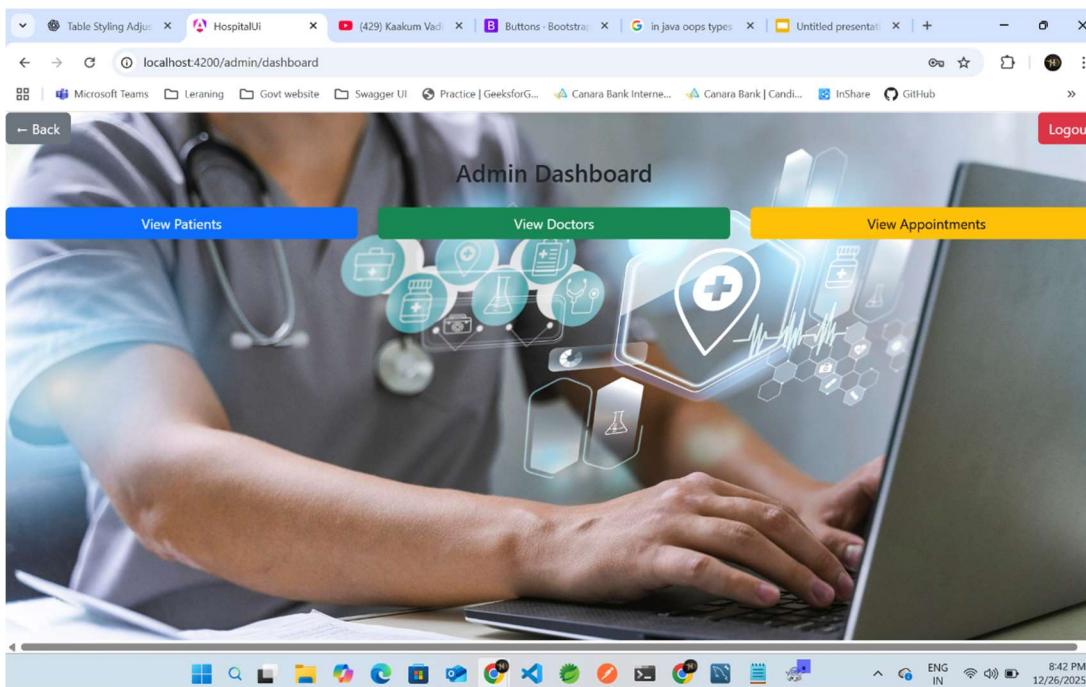
16. USER INTERFACE DESIGN

The user interface is designed to be simple, intuitive, and user-friendly. Angular and Bootstrap are used to create responsive and visually appealing pages.

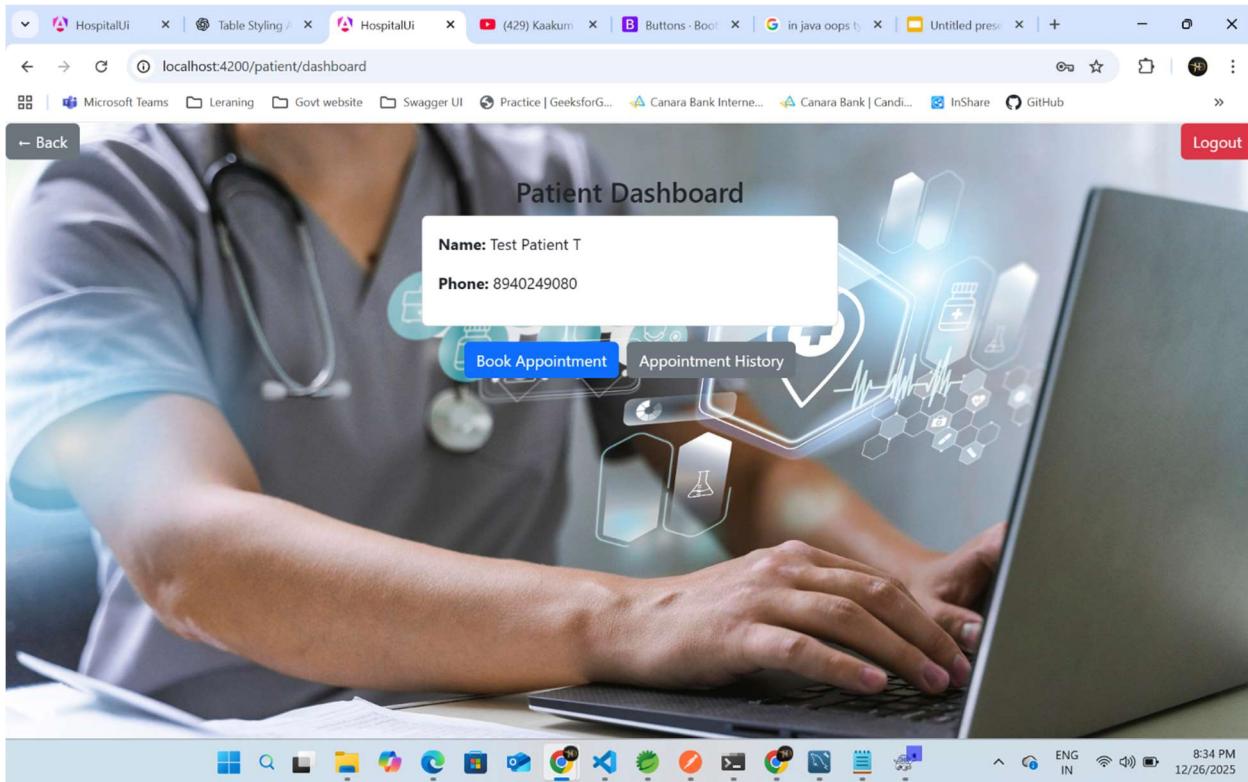
16.1 Home Page



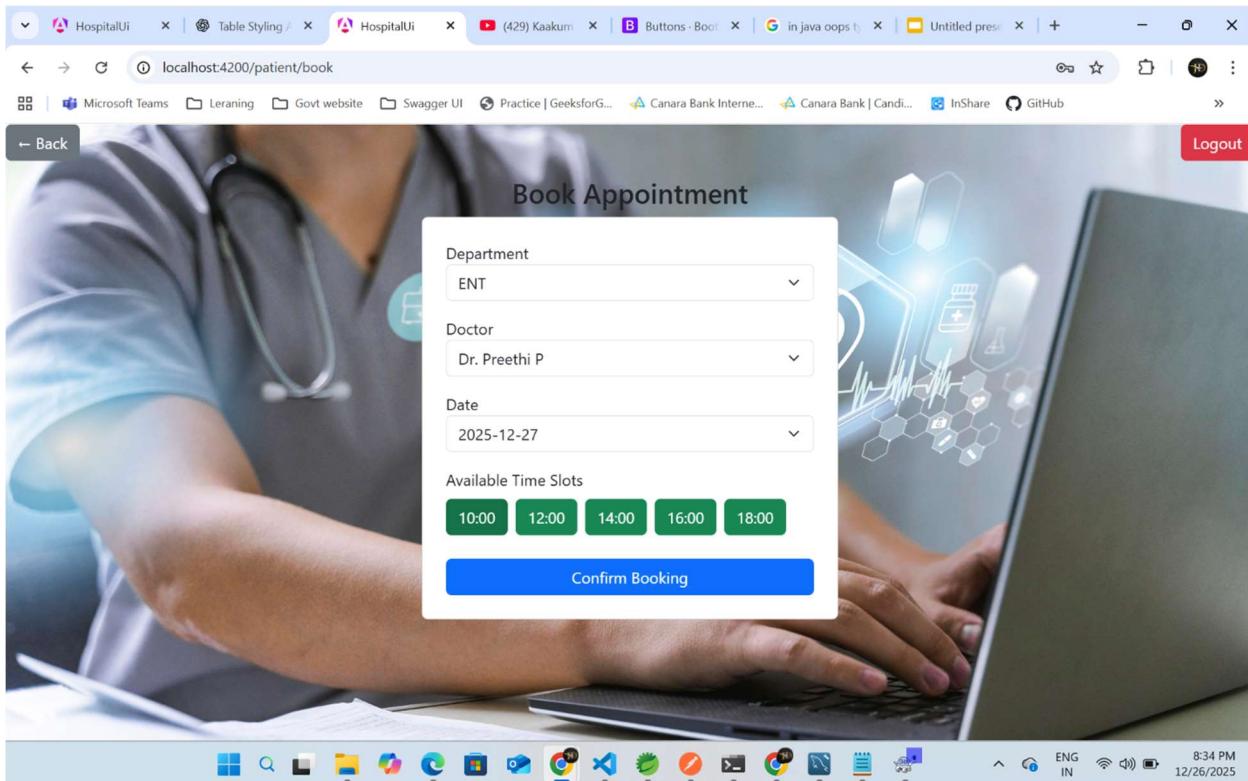
16.2 Admin Dashboard



16.3 Patient Dashboard



16.4 Patient Book Appointment



16.5 Patient Appointment History

A screenshot of a web browser window titled "localhost:4200/patient/history". The page displays a table titled "My Appointments" with the following data:

ID	Doctor	Department	Date	Time	Status	Action
2	Dr. Preethi P	ENT	2025-12-28	12:00	BOOKED	Cancel
3	Dr. Preethi P	ENT	2025-12-27	12:00	CANCELLED	

The background of the page features a doctor's hands typing on a laptop keyboard, overlaid with medical icons like a stethoscope, test tubes, and a heart monitor.

16.6 Doctor Dashboard

A screenshot of a web browser window titled "localhost:4200/doctor/dashboard". The page displays a table titled "Doctor Dashboard" with the following data:

ID	Patient	Patient Phone	Date	Time	Status	Action
1	Test Patient T	8940249080	2025-12-27	10:00	IN_PROGRESS	Solve Reject
3	jagan P	123456789	2025-12-27	12:00	CANCELLED	
2	jagan P	123456789	2025-12-28	12:00	BOOKED	In-Progress

The background of the page features a doctor's hands typing on a laptop keyboard, overlaid with medical icons like a stethoscope, test tubes, and a heart monitor.

17. TESTING

Testing is an essential phase in software development to ensure that the system functions correctly, reliably, and meets user requirements. The Hospital Appointment Booking System was tested using multiple techniques to identify and eliminate defects.

17.1 Unit Testing

- Performed on individual modules such as patient registration, doctor management, and appointment booking.
- Each function was tested independently to verify that it produces the expected output for valid inputs.

17.2 Integration Testing

- Conducted to ensure smooth interaction between different modules.
- Verified frontend-backend communication and database connectivity.
- Confirmed that integrated components work together without errors.

17.3 API Testing Using Postman

- RESTful APIs were tested using Postman.
- Validated request handling, response correctness, and error management.
- Tested HTTP methods such as GET and POST to ensure accurate data exchange.

17.4 User Interface Testing

- UI testing checked usability, navigation, and responsiveness of the application.
- All screens were tested across different resolutions.
- Ensured a consistent and user-friendly experience.

18. ADVANTAGES

The Hospital Appointment Booking System offers several benefits to hospitals, doctors, and patients:

- Saves time by enabling online appointment booking.
- Reduces administrative workload and paperwork.
- Improves hospital workflow efficiency.
- Minimizes appointment conflicts.
- Provides an easy-to-use and user-friendly interface.

19. LIMITATIONS

Despite its advantages, the system has certain limitations:

- Requires a stable internet connection.
- Payment integration is not implemented.
- Basic authentication mechanisms are used.
- Limited to appointment management functionality.

These limitations can be addressed in future versions of the system.

20. FUTURE ENHANCEMENTS

The system is designed to be scalable and can be extended with advanced features such as:

- Implementation of JWT-based authentication for enhanced security.
- Email and SMS notifications for appointment confirmation and reminders.
- Integration of online payment gateway for consultation fees.
- Development of a mobile application for wider accessibility.

21. CONCLUSION

The Hospital Appointment Booking System successfully automates the appointment scheduling process and addresses the challenges of traditional manual systems.

- Improves hospital efficiency.
- Reduces patient waiting time.
- Ensure accurate data management.
- Provides scalable architecture, secure design, and user-friendly interface.

The system is well-suited for real-world healthcare environments and can be further enhanced to meet evolving requirements.

22. REFERENCES

- Spring Boot Documentation
- Angular Official Documentation
- Oracle Java Documentation
- Bootstrap Documentation