

# HOSPITAL MANAGEMENT SYSTEM

## TEAM MEMBERS

ANIKET AGGARWAL  
PAVAN KARTIK  
DIVYA KULKARNI  
ANIMESH SINGH

ANSH MAHAJAN  
PRATHMESH NAGPURE  
MANASH SANGAM  
ANUSHKA DIXIT

## ABOUT THE PROJECT

The proposed Hospital Management System (HMS) is a web-based solution aimed at enhancing the efficiency of hospital operations by digitalizing core processes. This system enables hospital staff to register patients, book appointments, and manage schedules seamlessly. Admins are provided with robust tools to oversee doctor schedules, update records, and generate detailed reports, ensuring smooth hospital management. Doctors can maintain their schedules, manage appointments, and recommend tests and medications directly through the system. By integrating these functionalities into a single platform, the HMS streamlines communication and coordination within the hospital, ultimately improving patient care and administrative efficiency.

# APPLICATION FEATURES

# APPLICATION FEATURES

## PATIENT AND APPOINTMENT MANAGEMENT

- Patient Registration: Users (hospital staff) can register new patients and maintain accurate patient records.
- Appointment Booking: Users can book appointments with doctors on behalf of patients, ensuring the availability of doctor schedules during the booking process.
- View and Cancel Appointments: Doctors and Users can view scheduled appointments and cancel them if necessary.

## DOCTOR AND SCHEDULE MANAGEMENT

- Doctor Schedule Setup and Updates: Doctors are required to set up their schedules every three days, with the ability to update availability for specific days.
- Appointment Management: Doctors can view and cancel appointments directly through the system.
- Suggest Tests and Medications: Doctors can recommend future medical tests and prescribe medications based on patient consultations.

## ADMINISTRATIVE FEATURES AND CONTROLS

- Pre-Created Admin Account: Admins have a pre-created account with full control over system management.
- Doctor Registration and Management: Admins can register doctors, update their details, and manage their schedules, including canceling appointments if a doctor becomes unavailable.
- Import Doctors: Admins can import doctor data from external JSON/XML files, ensuring no duplicate entries in the system.
- View Patient Details: Admins can view detailed patient information, allowing them to monitor patient records and manage hospital operations effectively.

# BACKEND FEATURES

## LOGGING AND LOGGER FILE MAINTENANCE:

- Tracks system activities, errors, and user actions.
- Maintains organized logger files for monitoring, debugging, and audit trails.

## USER AUTHENTICATION

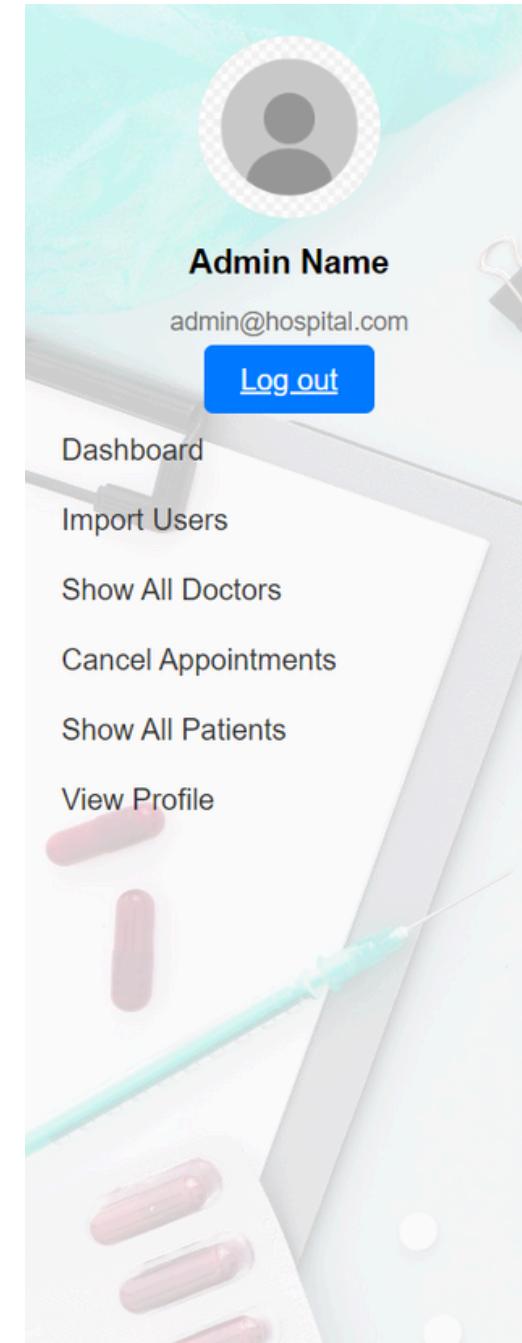
- Secure login system for all users.
- Ensures role-based access to functionalities (Admin, Doctor, Hospital Staff).

## ROLE-BASED FUNCTIONALITY

- Admin: Manage doctor registration, schedules, patient details, and appointments.
- Doctor: Set up and update schedules, view/cancel appointments, and suggest tests/medications.
- Hospital Staff (Users): Register patients, book appointments, and access relevant doctor schedules.

# UI SNAPSHOTS

# ADMIN DASHBOARD



## Admin Dashboard

Today's Date  
2024-08-28

### Welcome, Admin

Manage the hospital's operations from here.

10

Doctors

100

Patients

5

New Appointments

2

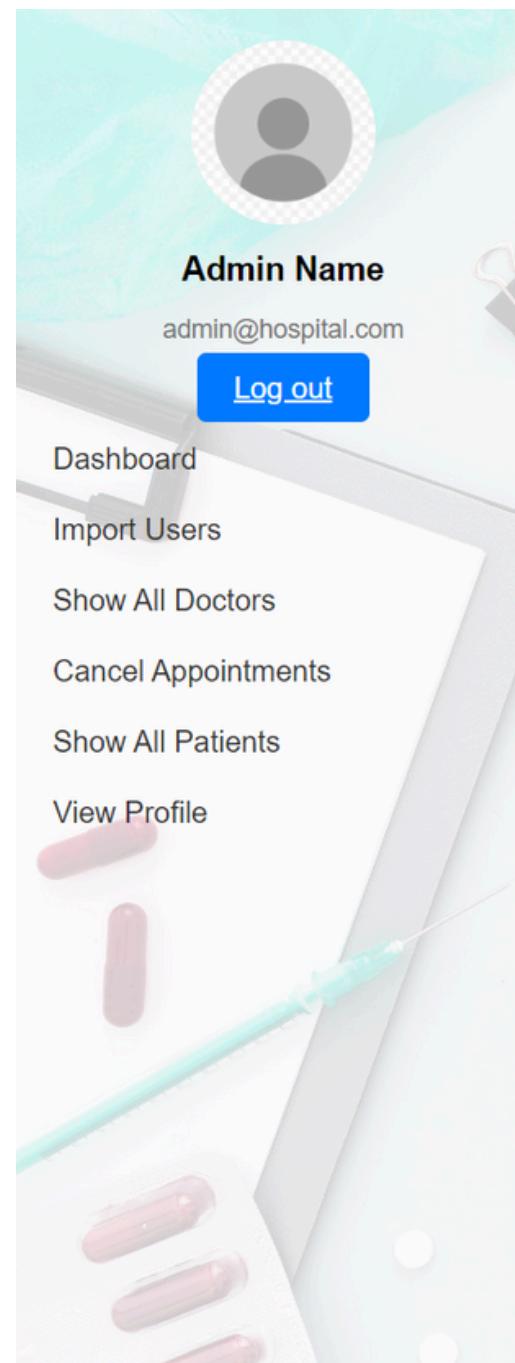
Canceled Appointments

### Manage Users

User ID	Name	Role	Edit	Delete
1	John Doe	Admin	<button>Edit</button>	<button>Delete</button>
2	Jane Smith	Doctor	<button>Edit</button>	<button>Delete</button>
3	Emily White	Staff	<button>Edit</button>	<button>Delete</button>

This is the Admin Dashboard of Admin Login after login page with all its functionalities on the side bar and basic details in center main content

# ADMIN - IMPORT USERS



## Import Users

Today's Date  
2024-08-28

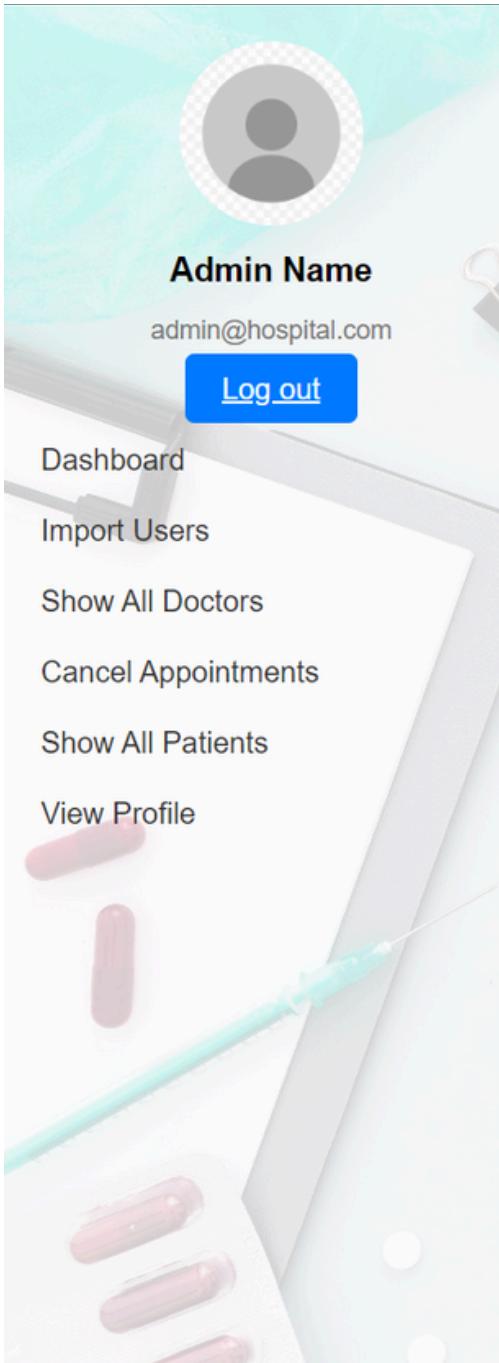
### Upload User Data

Upload File (JSON/XML):

No file chosen

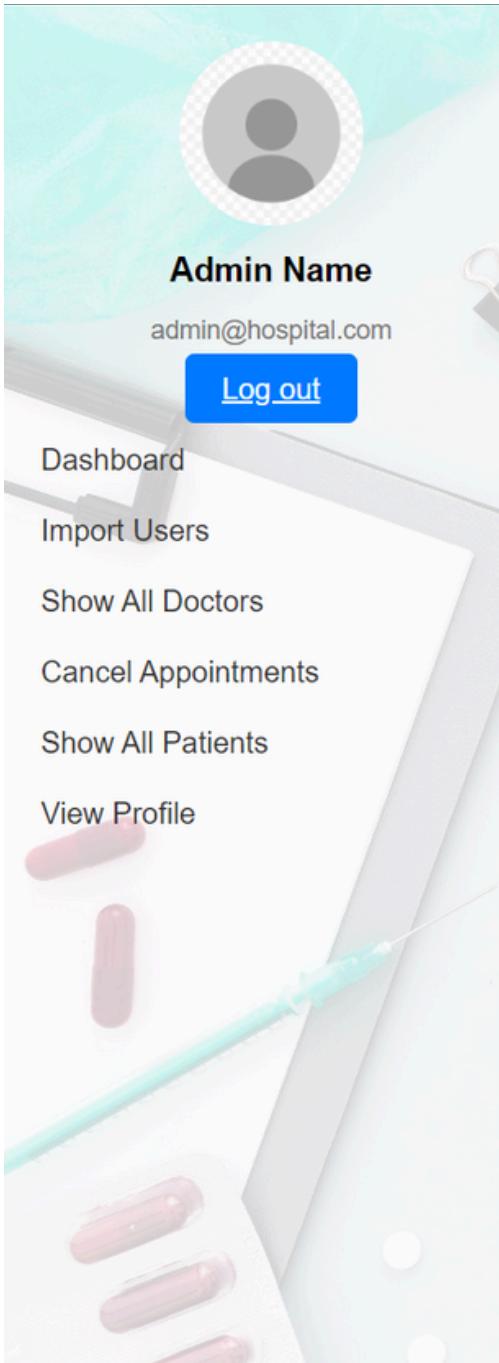
This is the Import User page for admin user, here after uploading an XML file of new user can be registered in the Database.

# ADMIN - SHOW ALL DOCTORS



Admin Name  
admin@hospital.com  
[Log out](#)

Dashboard  
Import Users  
Show All Doctors  
Cancel Appointments  
Show All Patients  
View Profile



## Show All Doctors

Today's Date  
2024-08-28

### Doctors List

Dr. Jolly Kapoor	ID: 1401	Specialization: Cardiologist	Schedule: 9:00 AM - 5:00 PM	<a href="#">View Schedule</a>	<a href="#">Cancel Appointments</a>
Dr. Jenny Verma	ID: 1402	Specialization: Pediatrician	Schedule: 10:00 AM - 4:00 PM	<a href="#">View Schedule</a>	<a href="#">Cancel Appointments</a>

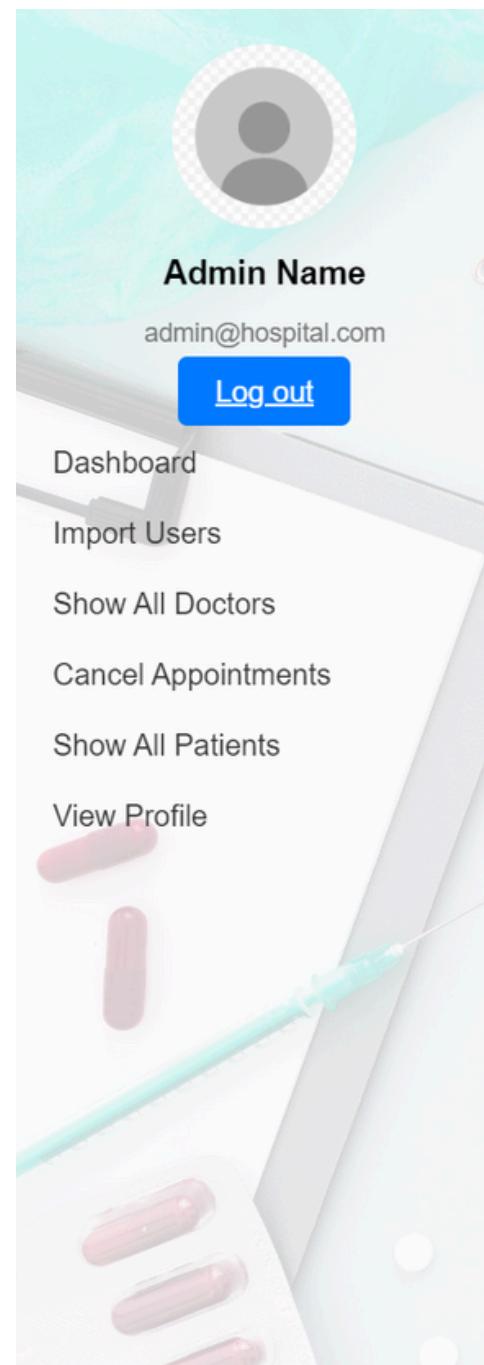
### Cancel Appointments

Are you sure you want to cancel all appointments for the selected doctor?

[Confirm](#)

This is the Show All Doctors page for admin users, here details of all doctors in database is shown. Admin can view there schedule and cancel them.

# ADMIN - CANCEL APPOINTMENTS



## Cancel Appointments

### Select Doctor & Date

Select Doctor:

Dr. Jolly Kappor

Select Date:

29 - 08 - 2024

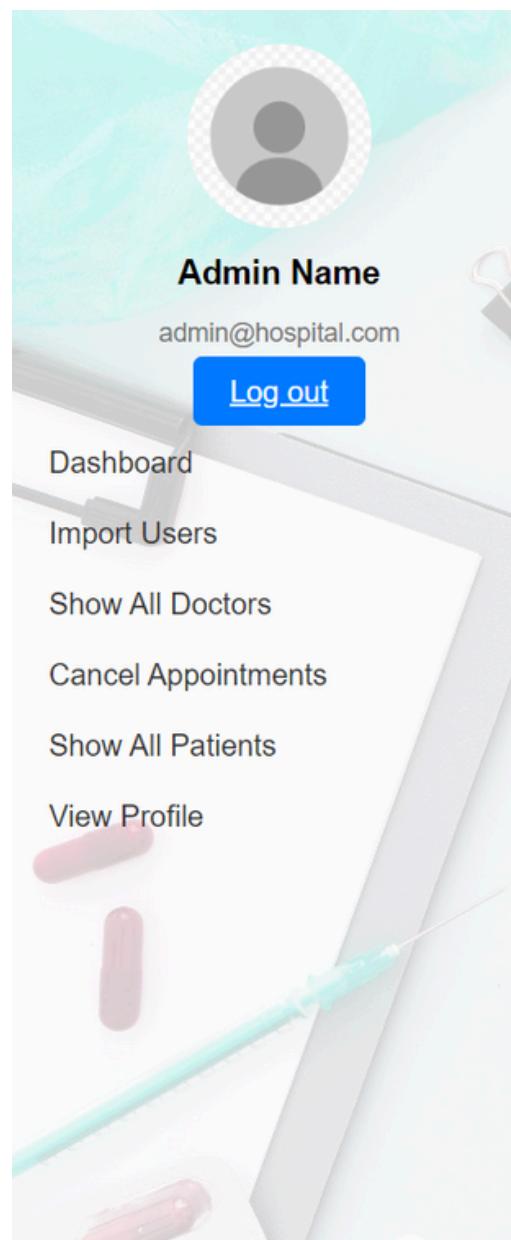


[Cancel Appointments](#)

Appointments for doctor ID 1 on 2024-08-29 have been cancelled.

This is the Cancel Appointments page for admin users, The admin can select a doctor and cancel appointments of them.

# ADMIN - SHOW ALL PATIENTS



## Show All Patients

Today's Date  
2024-08-28

### Patients List

Patient ID	Name	Age	Gender	Contact	Actions
101	Anjali Sharma	30	Female	98765-23456	<button>View Details</button> <button>Remove</button>
102	Bishwas Saproo	45	Male	90902-34234	<button>View Details</button> <button>Remove</button>

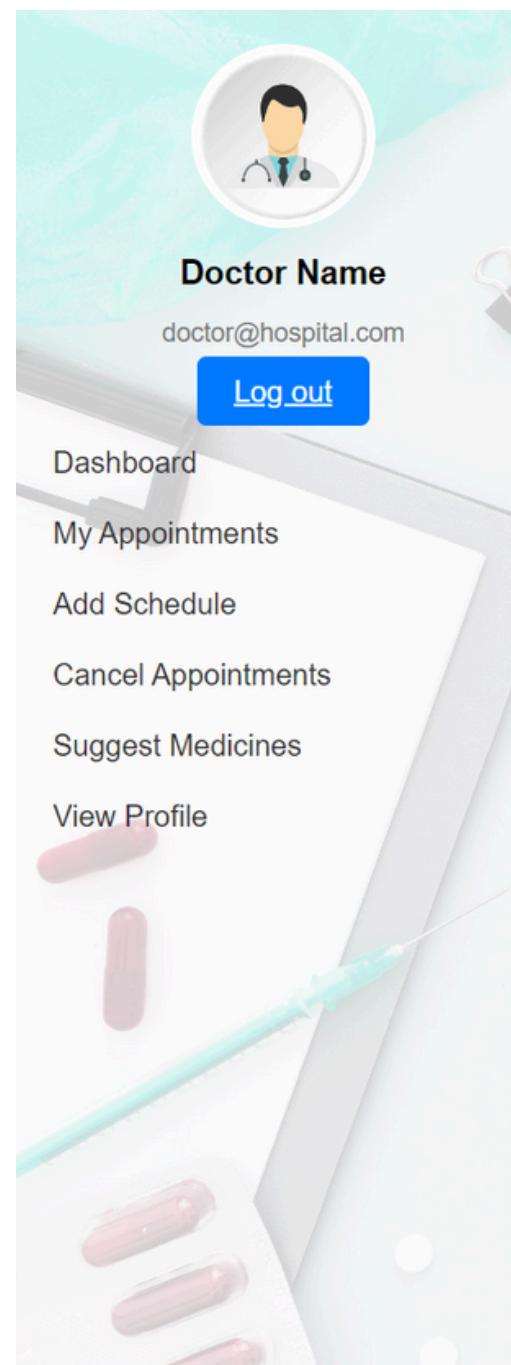
### Patient Details for Anjali Sharma

ID	101
Age	30
Gender	Female
Contact	98765-23456
Last Appointment Date	24-07-2024

© 2024 XYZ Hospital Management System. All rights reserved.

This is the Show All Patients page for admin users, The admin can see all the details of registered patients in the hospital and remove them.

# DOCTOR DASHBOARD



## Doctor Dashboard

Today's Date  
2024-08-28

### Welcome, Doctor

Manage your appointments, schedules, and patient care.

15

Today's Appointments

5

Canceled Appointments

10

Upcoming Sessions

3

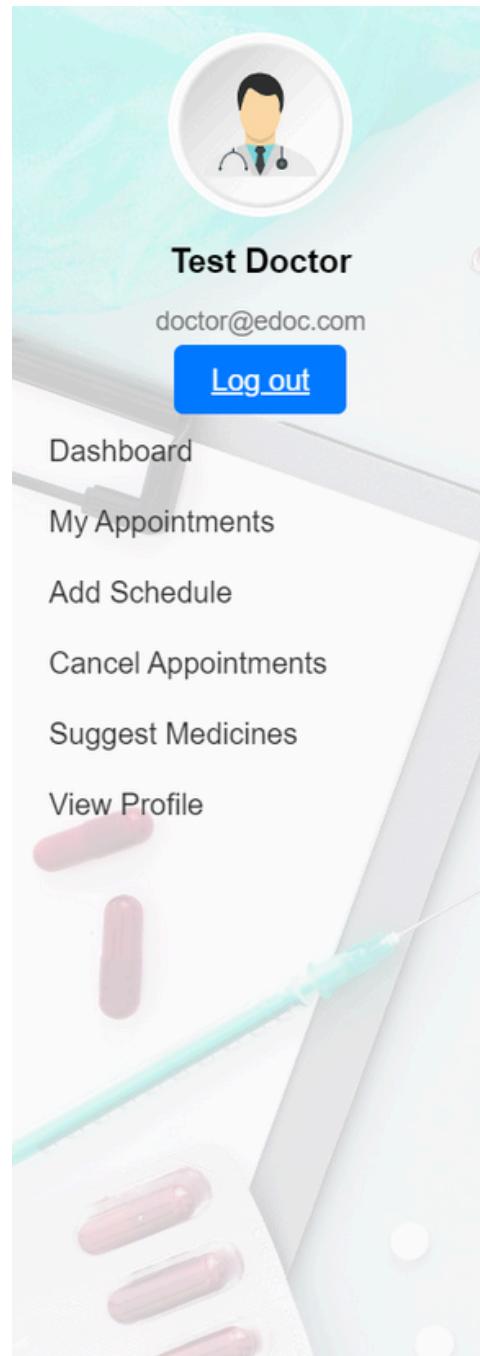
Medications Suggested

### Your Upcoming Sessions

Date	Time	Patient Name	Reason for Visit
01-09-2024	10:00 AM	Ansh Kapoor	Regular Checkup
01-09-2024	11:30 AM	Anant Garg	Follow-up
02-09-2024	15:30 PM	Jiya Bansal	Surgery
03-09-2024	10:30 AM	Lakshay Mishra	Blood Test
03-09-2024	13:00 PM	Mokshit Verma	Therapy

This is the Doctor Dashboard of Doctor Login after the login page with all its functionalities on the side bar and basic details in center main content

# DOCTOR - VIEW APPOINTMENTS



## View Appointments

Today's Date  
2024-08-28

Select Date:

29 - 08 - 2024

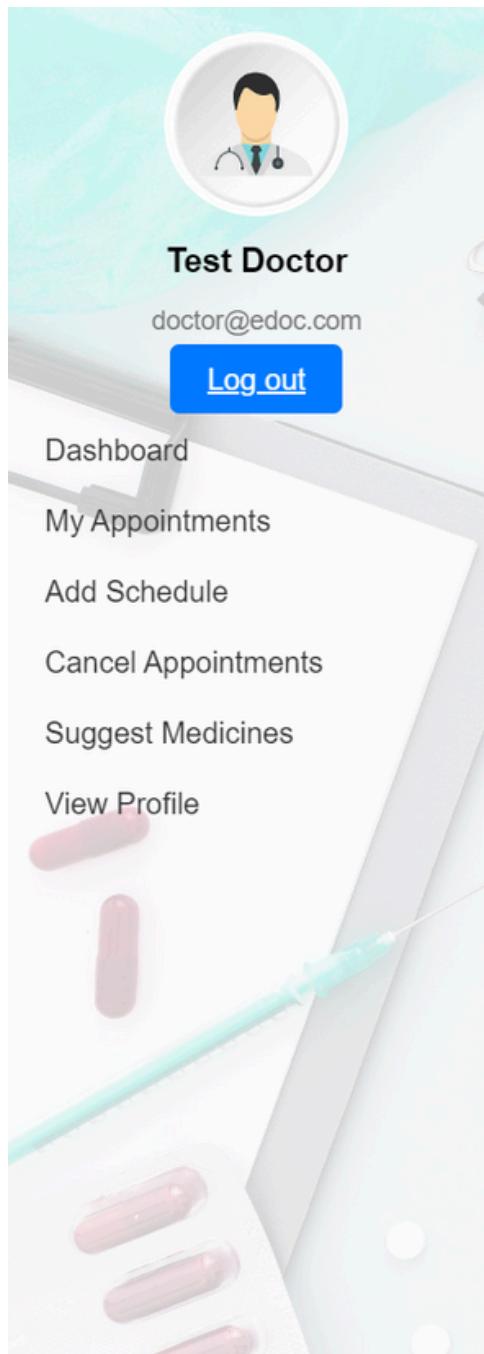
[View Appointments](#)

Patient Name	Time	Actions
ANjali Sharma	10:00 AM	<a href="#">Cancel</a>
Bishwas Saproo	11:00 AM	<a href="#">Cancel</a>

Appointment for ANjali Sharma has been cancelled.

This is the View Appointments page for doctors. The doctors can see the details of appointments assigned to them.

# DOCTOR - ADD SCHEDULES



## Add Schedule

Today's Date  
2024-08-28

**Date:**  
30 - 08 - 2024

**Start Time:**  
09 : 00

**End Time:**  
15 : 00

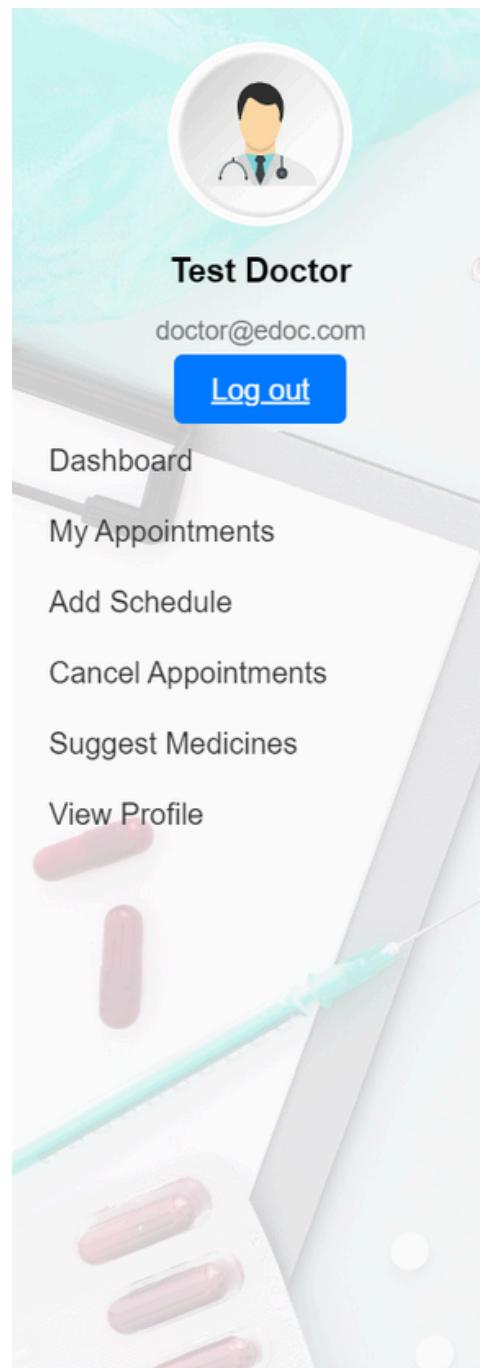
**Availability Status:**

**Add Schedule**

Schedule added for 2024-08-30 from 09:00 to 15:00. Status: available

This is the Add Schedule page for doctors. The doctors can add schedules for their upcoming 3 days here with multiple time slots as status of available or not available.

# DOCTOR - SUGGEST MEDICINES



## Suggest Medicines

Patient ID:

101

Medicines:

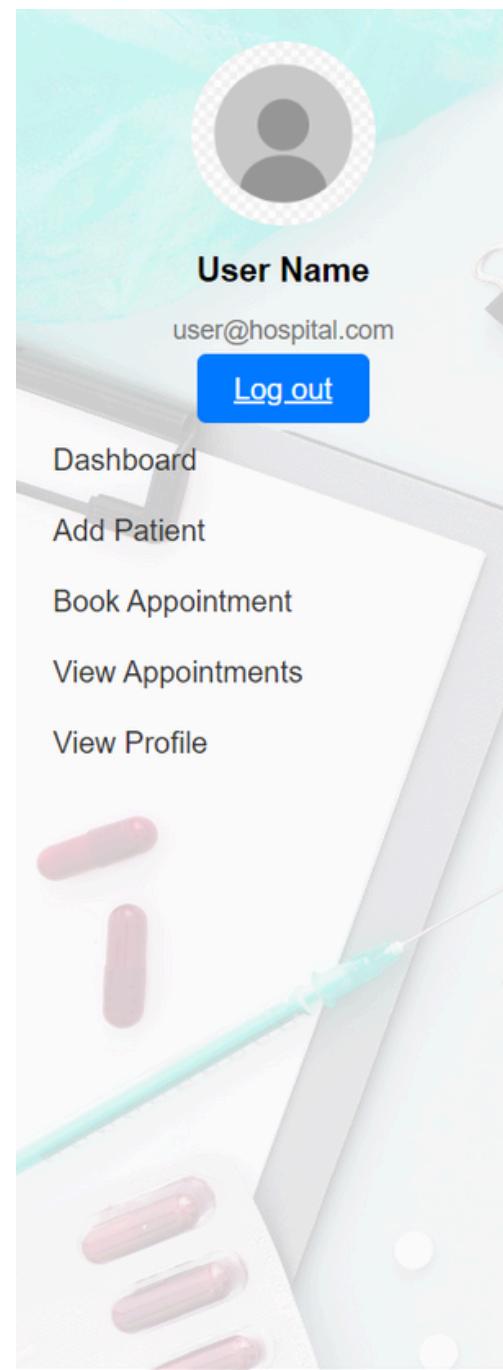
Corocene, cetirizine , Cough Syrup

**Suggest Medicines**

Medicines suggested for Patient ID 101: Corocene, cetirizine , Cough Syrup

This is the Suggest Medicines page for doctors. The doctors can Suggest the medicines to a specific patients from this page.

# USER DASHBOARD



## User Dashboard

Today's Date  
2024-08-28

### Welcome, User

Manage patients and appointments efficiently.

50

Patients

10

Appointments

2

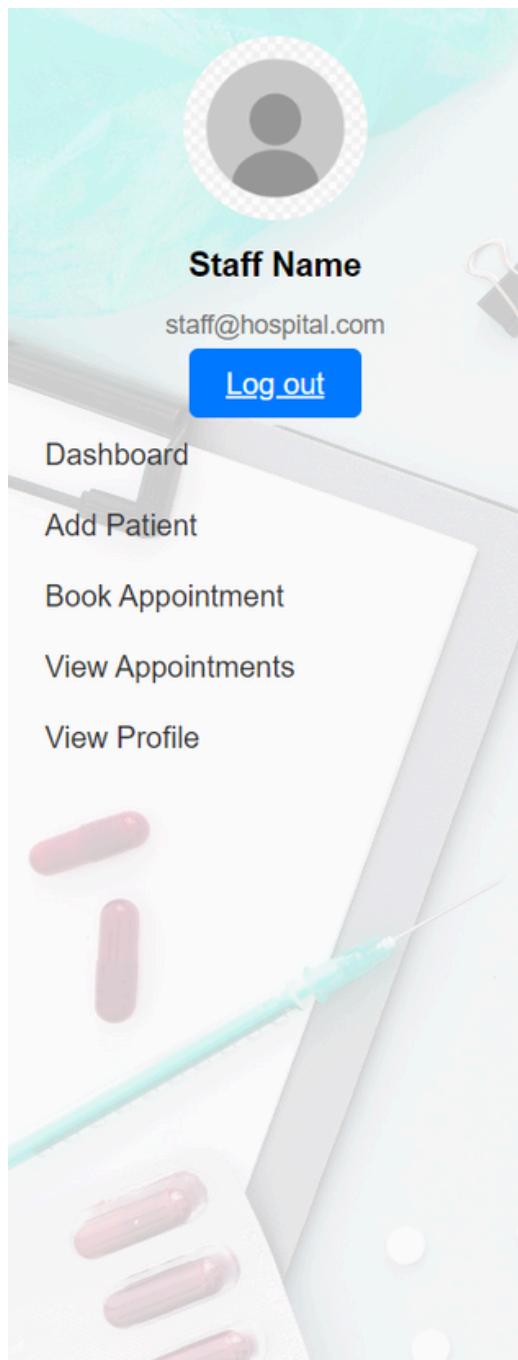
New Appointments

### Doctor Schedules

Doctor Name	Specialty	Day	Time
Dr. John Doe	Cardiologist	Monday	9:00 AM - 12:00 PM
Dr. Jane Smith	Dermatologist	Tuesday	1:00 PM - 4:00 PM
Dr. Emily Davis	Neurologist	Wednesday	10:00 AM - 2:00 PM

This is the User Dashboard of User Login after the login page with all its functionalities on the side bar and basic details in center main content

# USER - ADD PATIENT



## Hospital Staff Dashboard

Today's Date  
2024-08-28

### Add Patient

Name:

Anant Mehta

Age:

21

Gender:

Male

Contact Number:

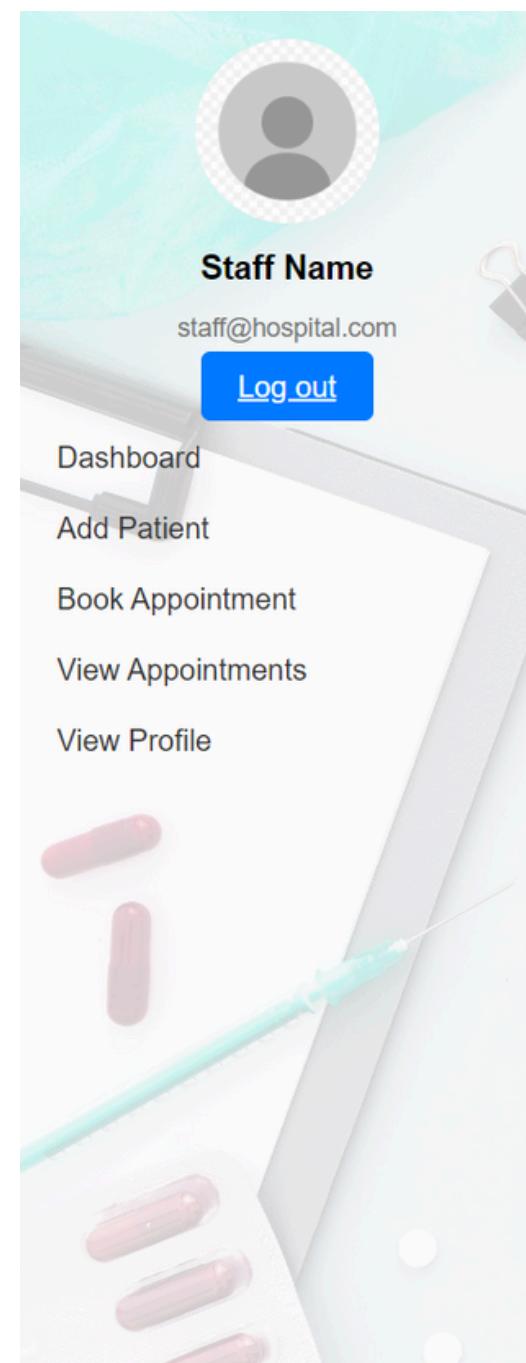
3482783932

Add Patient

Patient Anant Mehta added successfully.

This is the add patients page for the staff. The staff can add new patients in the database as registration.

# USER - BOOK APPOINTMENT



## Hospital Staff Dashboard

Today's Date  
2024-08-28

### Book Appointment

Patient ID:

101

Doctor ID:

1402

Date:

01-09-2024



Time:

10:00

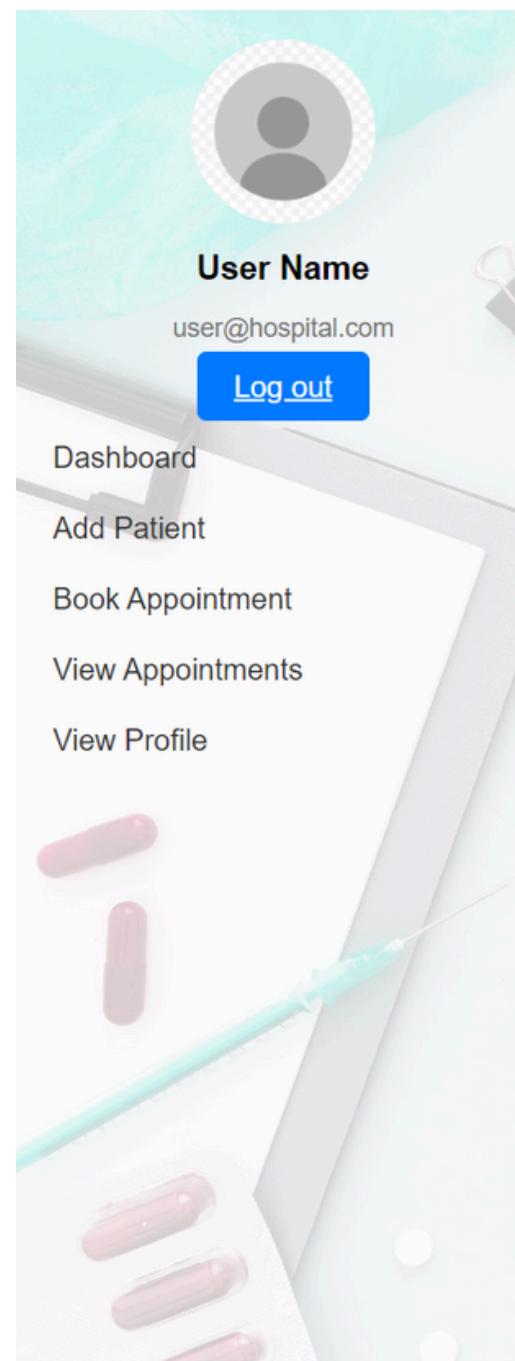


**Book Appointment**

Appointment booked for Patient ID 101 with Doctor ID 1402 on 2024-09-01 at 10:00.

This is the book appointments page by the staff. The staff can add a new appointment in the database according to the doctor availability.

# USER - VIEW APPOINTMENTS



## Hospital Staff Dashboard

Today's Date  
2024-08-28

### View Appointments

Doctor ID:

1402

Date:

28-08-2024



[View Appointments](#)

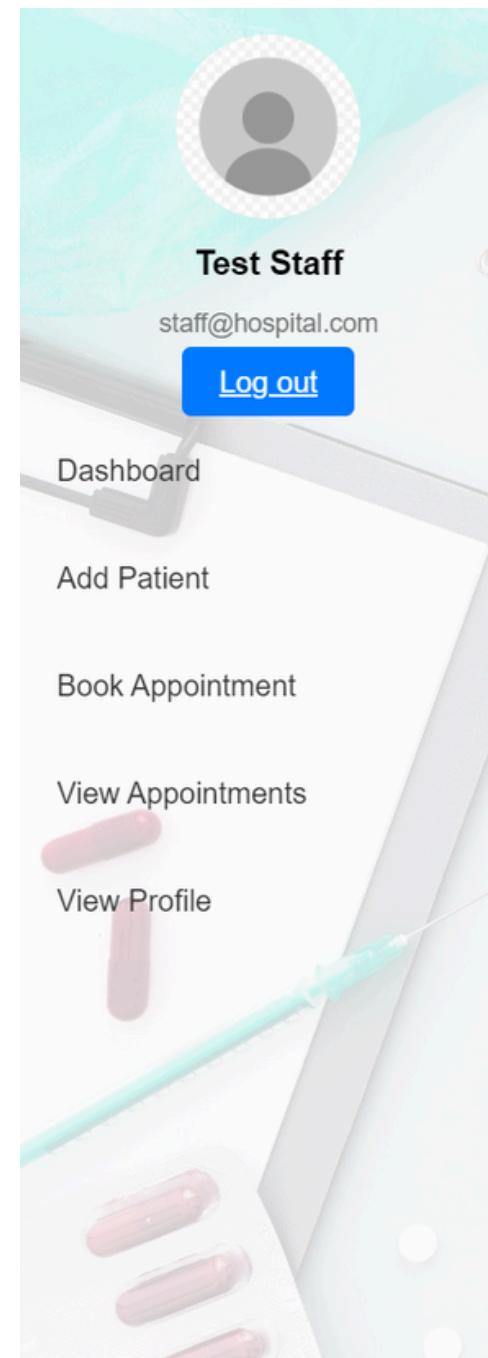
#### Appointments for Doctor ID 1402 on 2024-08-28:

Patient ID: P123, Time: 10:00 AM

Patient ID: P124, Time: 11:00 AM

This is the view appointments page for the staff. The staff can see the existing appointments for upcoming days for each doctor.

# VIEW PROFILE



## View Profile

<b>Profile Picture:</b>	
<b>Name:</b>	Test Staff
<b>Email:</b>	staff@hospital.com
<b>Phone:</b>	+91 67890-76543
<b>Position:</b>	Staff Member
<b>Joined:</b>	February 15, 2022

© 2024 Bankops Hospital Management System. All rights reserved.

This is the view profile details for each user (admin, doctor & staff) to see their own details.

# BACKEND CONSOLE LOG OUTPUT

```
"C:\Program Files\Java\jdk-17\bin\java.exe" ...
[{"password":"322Ansh@","role":"admin","name":"ansh","mobileNo":"9682347116","email":"anshmahajan@gmail.com","username":"anshmahajan05"}, {"password":"322Ansh@","role":"doctor","name":"Aniket","mobileNo":"9682347116","email":"aniket@gmail.com","username":"aniket05"}, {"password":"946e37dd93db7e8c17739ed2b35f0ee0a1989a3e75589f170df82d6c","name":"pavan kartik","mobileNo":"9999999999","email":"pavankartik@gmail.com","username":"pavXn"}, {"password":"946e37dd93db7e8c17739ed2b35f0ee0a1989a3e75589f170df82d6c","name":"ansh","mobileNo":"9682347116","email":"anshmahajan@gmail.com","username":"anshmahajan05"}, {"password":"946e37dd93db7e8c17739ed2b35f0ee0a1989a3e75589f170df82d6c","name":"Aniket","mobileNo":"9682347116","email":"aniket@gmail.com","username":"aniket05"}]
Update Status-->true
Delete User: AuthUserTbl{UserId=2, name='pavan kartik', email='pavankartik@gmail.com', mobileNo='9999999999', username='pavXn', password='946e37dd93db7e8c17739ed2b35f0ee0a1989a3e75589f170df82d6c'}
Delete User: AuthUserTbl{UserId=1, name='ansh', email='anshmahajan@gmail.com', mobileNo='9682347116', username='anshmahajan05', password='946e37dd93db7e8c17739ed2b35f0ee0a1989a3e75589f170df82d6c'}
Delete User: AuthUserTbl{UserId=3, name='Aniket', email='aniket@gmail.com', mobileNo='9682347116', username='aniket05', password='946e37dd93db7e8c17739ed2b35f0ee0a1989a3e75589f170df82d6c'}

Process finished with exit code 0
```

# KEY LEARNINGS

# KEY LEARNINGS SYSTEM DESIGN AND DEVELOPMENT

- Clear Requirement Gathering: Understanding user roles (doctors, admins, patients) is essential for designing tailored features.
- Loose Coupling: Adopting loose coupling between system components improves maintainability and flexibility, making future updates easier.
- Database Design: A well-structured database schema ensures data integrity and efficient querying, especially in a complex system.
- Backend Development: Using Java's features like functional interfaces and the Stream API streamlines backend logic and enhances code efficiency.
- Scheduling Logic: Developing robust scheduling features involves managing complex logic, addressing edge cases, and ensuring usability.

# KEY LEARNINGS

## USER INTERFACE AND SECURITY

- Frontend Design: Creating role-specific, intuitive, and responsive user interfaces is crucial for a positive user experience.
- Integration Challenges: Ensuring smooth communication between the frontend and backend requires careful API design and thorough testing.
- User Authentication and Privacy: Implementing best practices in user authentication and privacy protection is critical for securing sensitive data.
- Testing and Debugging: Rigorous testing and debugging are vital for catching issues early and ensuring system stability and reliability.

# CONNECTIVITY

# OUR FRONTEDE (HTML,CSS,JS) TO ANY BACKEND

## **RESTful APIs:**

- Implement RESTful APIs on the backend to handle HTTP requests from the frontend.
- The frontend makes API calls to interact with backend services and retrieve or send data in JSON or XML format.

## **AJAX (Asynchronous JavaScript and XML):**

- Use AJAX to perform asynchronous data retrieval and updates from the backend without reloading the page.
- AJAX allows for a dynamic user experience by updating parts of the webpage based on server responses.

## **Fetch API:**

- Utilize the Fetch API in JavaScript for making HTTP requests to backend services.
- The Fetch API provides a modern, promise-based approach to handling network requests and responses.

## **WebSockets:**

- Establish real-time, bidirectional communication between the frontend and backend using WebSockets
- WebSockets enable live data updates and interactive features such as chat applications or live notifications.

# OUR BACKEND (JAVA,JDBC) TO ANY FRONTEND

## **Servlets and JSP (Java Server Pages):**

- Use Java Servlets to handle HTTP requests and generate dynamic HTML responses.
- JSP allows embedding Java code within HTML to create dynamic web content directly from the backend.

## **Spring Boot:**

- Employ Spring Boot to build and expose RESTful services and serve static HTML content.
- Spring Boot simplifies backend development with embedded servers and automatic configuration.

## **JSON/XML Responses:**

- Return data from the backend in JSON or XML format for consumption by the frontend.
- Java libraries like Gson or Jackson can be used to serialize Java objects into JSON or XML for communication with the frontend.

## **API Endpoints:**

- Define API endpoints in the backend to handle specific requests from the frontend.
- These endpoints manage interactions such as data retrieval, updates, or processing and send responses back to the frontend.

**THANK YOU**