

# DOCSPOT: SEAMLESS APPOINTMENT BOOKING FOR HEALTH

## SUBTITLE:

A SMART HEALTHCARE APPOINTMENT SOLUTION

### **Team Members:**

Team Leader : SK Gouse Basha

Team Member : Gonuguntla Chandana

Team Member : Veeramachineni Tirumala

Team Member : V Triveni

Team Member : Jonnalagada Haritha

# PROJECT OVERVIEW:

- **Purpose:**
- DocSpot aims to simplify the healthcare appointment process by providing users with an intuitive platform to search for doctors, view availability, and book appointments online. It enhances patient-doctor interaction and minimizes appointment management overhead for healthcare providers.
- **Features:**
- User registration and login
- Doctor profile browsing and filtering
- Real-time appointment scheduling
- Notifications via email
- Admin dashboard for managing users and doctors
- Secure authentication and role-based access control

# ARCHITECTURE:

## **Frontend (React):**

Built with ReactJS, utilizing functional components and React Router for navigation. State is managed via Redux, and API calls are handled using Axios. Responsive design implemented using Tailwind CSS.

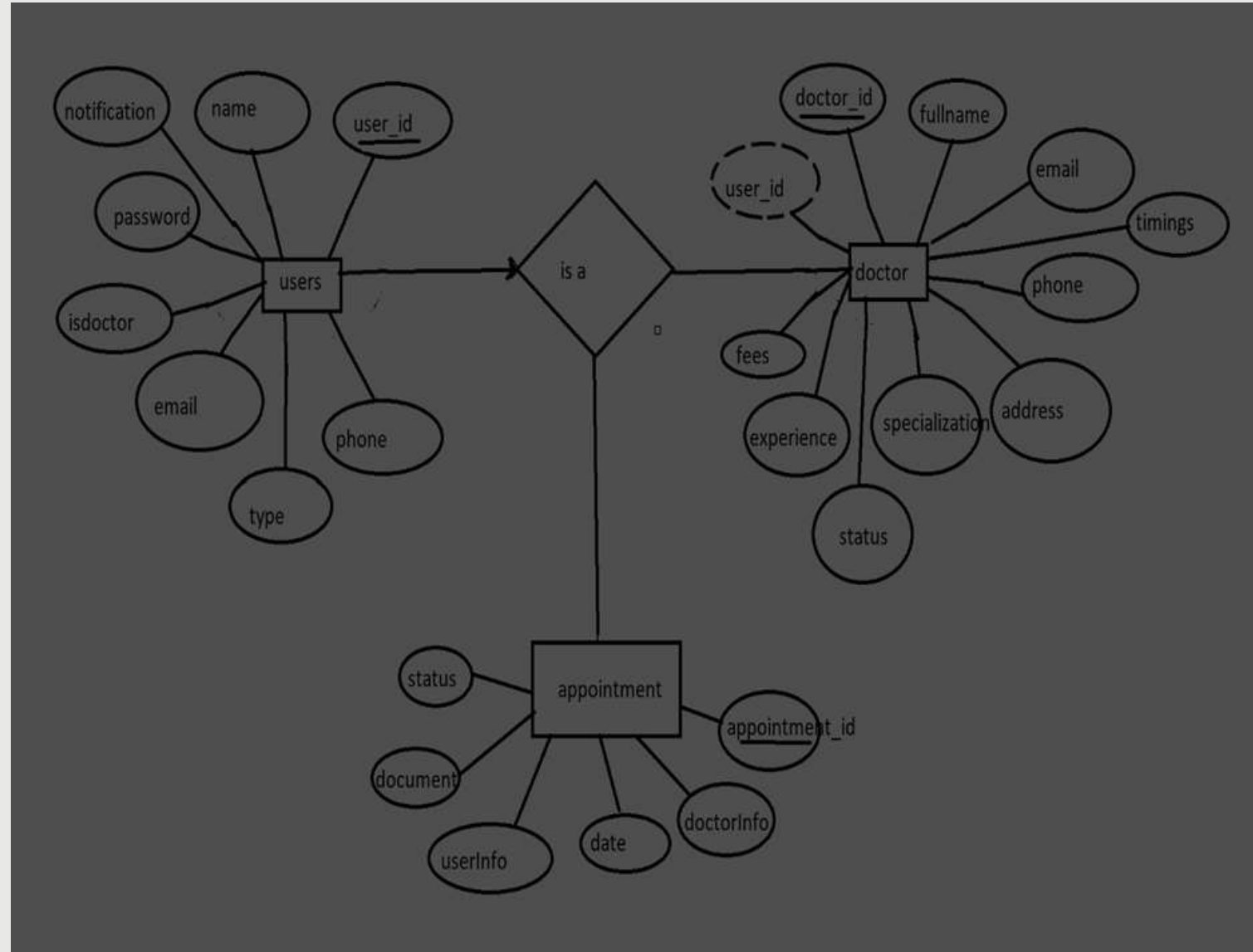
## **Backend (Node.js + Express.js):**

RESTful API built with Express.js. Controllers handle business logic; middleware manages authentication and error handling. Role-based access for users (patient, doctor, admin).

## **Database (MongoDB):**

Schema includes Users, Doctors, Appointments, and Availability. MongoDB is accessed using Mongoose for schema validation and easy query handling.

# ER DIAGRAM



# SETUP INSTRUCTIONS

## Prerequisites:

- Node.js (v16+)
- MongoDB (local or Atlas)
- Git

## Installation:

```
# Clone the repository
git clone https://github.com/your-org/docspot.git
cd docspot
# Install frontend dependencies
cd client
npm install
# Install backend dependencies
cd ../server
npm install
```

## Environment Variables:

```
PORT=5000
MONGO_URI=your_mongo_connection_string
JWT_SECRET=your_jwt_secret
```

# Running the Application

## Frontend:

```
cd client  
npm start
```

## Backend:

```
cd server  
npm start
```

# API DOCUMENTATION

Endpoint	Method	Description	Request Body	Response
/api/auth/register	POST	Register new user	{ name, email, password }	User object + token
/api/auth/login	POST	Login user	{ email, password }	User object + token
/api/doctors	GET	List all doctors	—	Array of doctors
/api/appointments	POST	Book appointment	{ userId, doctorId, date }	Appointment confirmation

# Authentication

Authentication is handled using JWT (JSON Web Tokens). Upon login or registration, a token is issued and stored in localStorage. Protected routes use middleware to verify the token and allow/deny access accordingly.

- User roles:** patient, doctor, admin
- Token expiry:** 1 hour
- Refresh mechanism:** To be added in future releases

## User Interface

**Key UI Screens:**

- Login/Register
- Doctor Search
- Appointment Booking
- User Dashboard
- Admin Panel

## Testing

**Tools Used:**

- Jest & React Testing Library for frontend
- Mocha & Chai for backend
- Postman for manual API testing

**Strategy:**

- Unit tests for utility functions
- Integration tests for API endpoints
- E2E tests with Cypress (planned)

## BOOK A DOCTOR

Login

Register



### Effortlessly schedule your doctor

appointments with just a few clicks,  
putting your health in your hands.

Book your Doctor



Book a Doctor

localhost:3000/register

MyLowWallpapers... MS Guides - Using... New tab - P... Post An Wallpaper... Gmail YouTube Maps ANOMALY DETECT... Untitled document... ASUS Software Port...

Home Login Register

## Sign up to your account

Full name

John

Email

j@gmail.com

Password

\*\*\*\*


Phone

+04567890

☐ Admin ☒ User

Register

Have an account? [Login here](#)



Book a Doctor

localhost:3000/login

MyLowWallpapers... MS Guides - Using... New tab - P... Post An Wallpaper... Gmail YouTube Maps ANOMALY DETECT... Untitled document... ASUS Software Port...

Home Login Register

## Sign in to your account

Email


j@gmail.com

Password

\*\*\*\*

Let's Enter

Don't have an account? [Register here](#)



MediCareBook

Hi..Admin

All Appointments for Admin Panel

Appointment ID	User Name	Doctor Name	Date	Status
672f7fa9b4c8952618190c57b	User	Koushick	2024-11-09 20:36	approved
672f7ce54c8952b18190cba5	User	Koushick	2024-11-09 20:46	approved
67303fb33ae507476ffb12d7	User	Koushick	2024-11-10 10:37	approved
6730423aaa10078f304ccaffe	User	Koushick	Sun Nov 10 2024 10:48:00 GMT+0530 (India Standard Time)	approved
6730a3e26adc4e5cc1d89d4e	User	Koushick	Sun Nov 10 2024 17:45:00 GMT+0530 (India Standard Time)	approved
6730a3e36adc4e5cc1d89d52	User	Koushick	Sun Nov 10 2024 17:45:00 GMT+0530 (India Standard Time)	approved
6730a73a8adc4e5cc1d89db3	User	Koushick	Sun Nov 10 2024 17:59:00 GMT+0530 (India Standard Time)	approved

© 2023 Copyright: MediCareBook

Book A Doctor

User

Doctor Registration request sent successfully

Apply for Doctor

Personal Details:

Full Name: Sr@VA

Phone: 91755584121

Email: user@gmail.com

Address: Chennai

Professional Details:

Specialization: Blood

Experience: 2


Fees: 5001


Timings: 06:00 - 12:00


Submit

© 2023 Copyright: MediCareBook

## Book A Doctor

 Appointments

 Apply doctor

 Logout

 User

## Home

### Dr. Koushick

Phone: **09176478438**

Address: **chennai**

Specialization: **ENT**

Experience: **5 Yrs**

Fees: **1000**

Timing: **07:00 : 12:00**

[Book Now](#)

### Dr. Karthick

Phone: **09176478438**

Address: **Kerala**

Specialization: **Bone**

Experience: **10 Yrs**

Fees: **500**

Timing: :

[Book Now](#)

# Known Issues

- Doctor availability is not synced with calendar in real-time
- No SMS notification integration yet
- Token refresh system not implemented

# Future Enhancements

Add video consultation feature

Enable SMS reminders for appointments

Admin analytics dashboard

Calendar sync with Google/Outlook



**THANK YOU**