

DocSpot- Seamless Appointment Booking For Health

INTRODUCTION:

Team ID : LTVIP2025TMID51265

Team Size : 4

Team Leader : Patil Rekha

Team member :C Usha Rani

Team member : Yarra Kadirappagari Swetha

Team member : Junturu Haritha

ONLINE COMPLAINT REGISTRATION AND MANAGEMENT SYSTEM

INTRODUCTION:

DocSpot is a full-stack appointment booking platform designed to simplify healthcare access. Built using React.js, Node.js, and MongoDB, it offers a seamless experience for patients, doctors, and admins. Users can easily browse doctors, book appointments, and manage schedules in real time. The platform ensures secure data handling and efficient communication through a robust client-server architecture.

DESCRIPTION

Booking a doctor's appointment has never been easier. With our convenient online platform, you can quickly and effortlessly schedule your appointments from the comfort of your own home. No more waiting on hold or playing phone tag with busy receptionists. Our user-friendly interface allows you to browse through a wide range of doctors and healthcare providers, making it simple to find the perfect match for your needs.

With our advanced booking system, you can say goodbye to the hassle of traditional appointment booking. Our platform offers real-time availability, allowing you to choose from a range of open slots that fit your schedule. Whether you prefer early morning, evening, or weekend appointments, we have options to accommodate your needs.

Scenario-based Case Study:

Scenario: Booking an Appointment with a Doctor

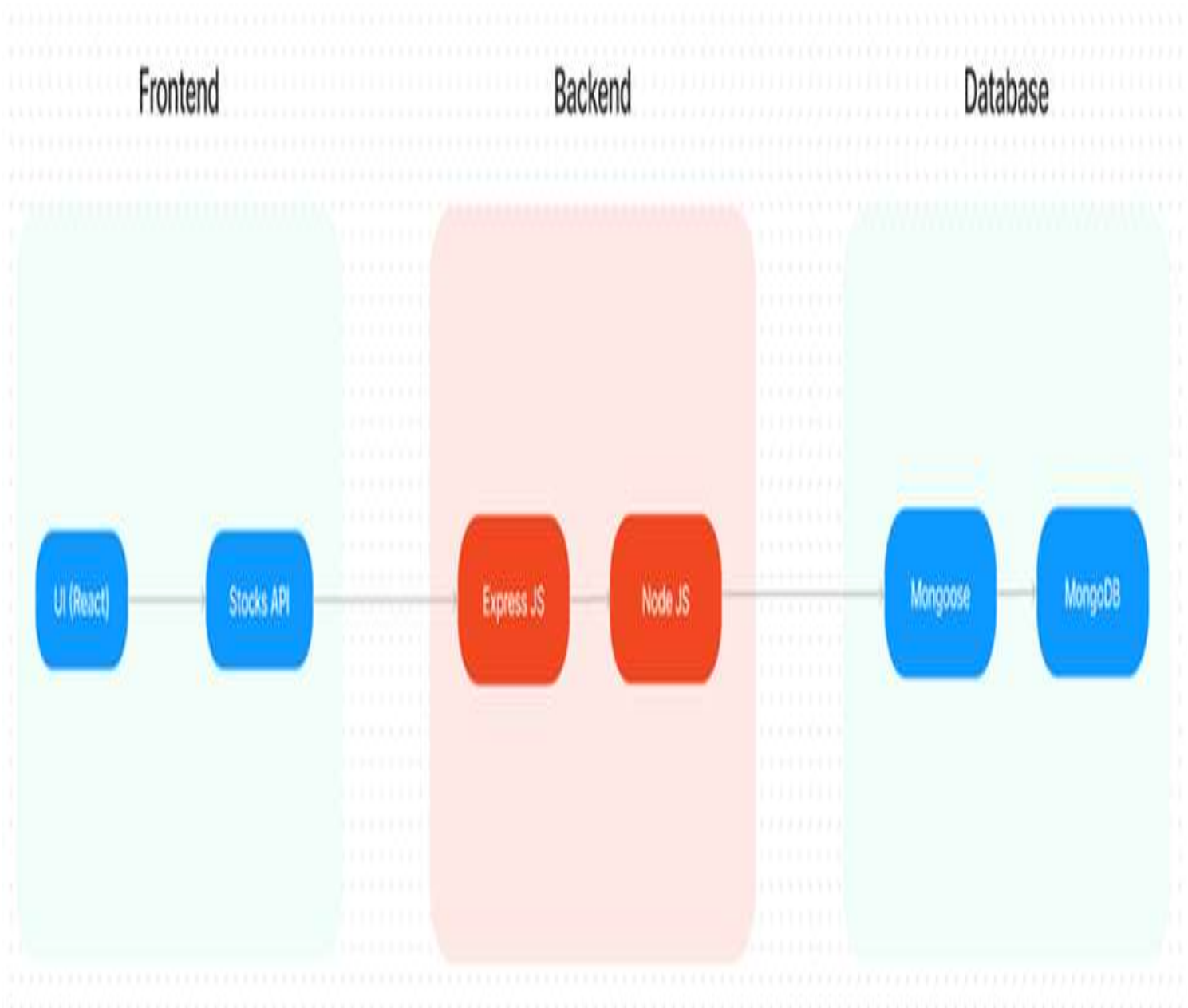
User Registration: John, who needs to see a doctor for a routine check-up, visits the Book a Doctor app and signs up as a Customer. He provides his email and creates a password.

Browsing Doctors: Upon logging in, John is presented with a dashboard displaying a list of available doctors and healthcare providers.

He filters the list based on his preferences, such as specialty, location, or availability.

Admin Approval (Background Process): In the background, the admin reviews new doctor registrations and approves legitimate applicants. Approved doctors are then registered in the app and can start managing their appointments.

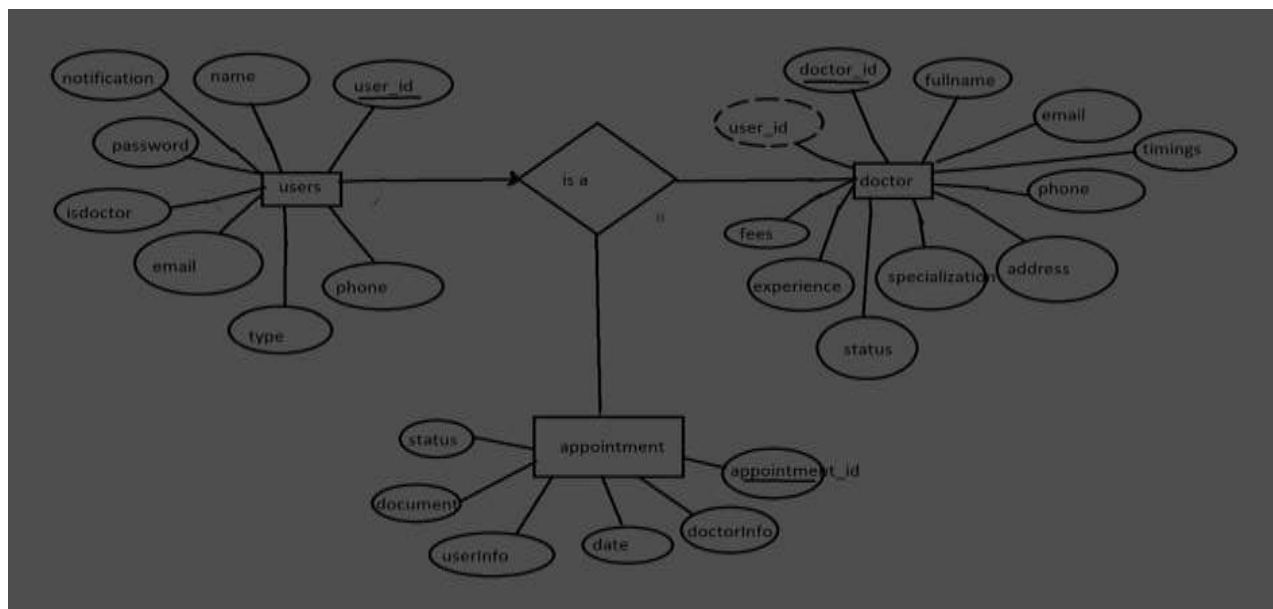
TECHNICAL ARCHITECTURE



The technical architecture of our Book a Doctor app follows a client-server model, where the front end serves as the client and the back end acts as the server. The front end encompasses not only the user interface and presentation but also incorporates the Axios library to connect with the backend easily by using RESTful Apis.

The front end utilizes the bootstrap and material UI library to establish a real-time and better UI experience for any user whether it is an admin, doctor, or ordinary user working on it. On the backend side, we employ Express.js frameworks to handle the server-side logic and communication.

For data storage and retrieval, our backend relies on MongoDB. MongoDB allows for efficient and scalable storage of user data, including user profiles, for booking rooms, adding rooms, etc. It ensures reliable and quick access to the necessary information. Together, the frontend and backend components, along with Moment, Express.js, and MongoDB, form a comprehensive technical architecture for our Book a Doctor app. This architecture enables real-time communication, efficient data exchange, and seamless integration, ensuring a smooth and immersive booking of an appointment and many more experiences for all users.



PRE-REQUISITES

PRE-REQUISITES:

Here are the key prerequisites for developing a full-stack application using Node.js, Express.js, MongoDB, and React.js:

•Node.js and npm:

Node.js is a powerful JavaScript runtime environment that allows you to run JavaScript code on the server side. It provides a scalable and efficient platform for building network applications.

Install Node.js and npm on your development machine, as they are required to run JavaScript on the server side.

Download: <https://nodejs.org/en/download/>

Installation instructions: <https://nodejs.org/en/download/package-manager/>

npm init

•Express.js:

Express.js is a fast and minimalist web application framework for Node.js. It simplifies the process of creating robust APIs and web applications, offering features like routing, middleware support, and modular architecture.

Install Express.js, a web application framework for Node.js, which handles server-side routing, middleware, and API development.

Installation: Open your command prompt or terminal and run the following command:

npm install express

•MongoDB:

MongoDB is a flexible and scalable NoSQL database that stores data in a JSON-like format. It provides high performance, horizontal scalability, and seamless integration with Node.js, making it ideal for handling large amounts of structured and unstructured data.

Set up a MongoDB database to store your application's data.

Download: <https://www.mongodb.com/try/download/community>

Installation instructions: <https://docs.mongodb.com/manual/installation/>

•Moment.js:

Moment.js is a JavaScript package that makes it simple to parse, validate, manipulate, and display date/time in JavaScript. Moment.js allows you to display dates in a human-readable format based on your location. Install React.js, a JavaScript library for building user interfaces.

Follow the installation guide: <https://momentjs.com/>

PROJECT STRUCTURE

▼ FRONTEND

> node_modules

> public

▼ src

▼ components

▼ admin

AdminA... U

AdminD... U

AdminH... U

AdminU... U

▼ common

Home.jsx U

Login.jsx U

Notificat... U

Register.... U

▼ user

AddDoc... U

ApplyDo... U

DoctorLi... U

UserApp... U

UserHo... U

▼ images

p2.png U

p3.webp U

photo1.png U

App.css M





JS App.js M














JS index.js M

.gitignore

package-loc... M

package.json M

✓ **BACKEND**    

- ✓  config
 - JS connectToDB.js
- ✓  controllers
 - JS adminC.js
 - JS doctorC.js
 - JS userC.js
- ✓  middlewares
 - JS authMiddleware.js
- >  node_modules
- ✓  routes
 - JS adminRoutes.js
 - JS doctorRoutes.js
 - JS userRoutes.js
- ✓  schemas
 - JS appointmentMo...
 - JS docModel.js
 - JS userModel.js
- ✓  uploads
 -  a77e910e017f4b...
 -  af8555e0fb38fe...
 -  .env
 -  .gitignore
 - JS index.js
 -  package-lock.json
 -  package.json

Installation of required tools:

For frontend, we use:

React

Bootstrap

Material UI

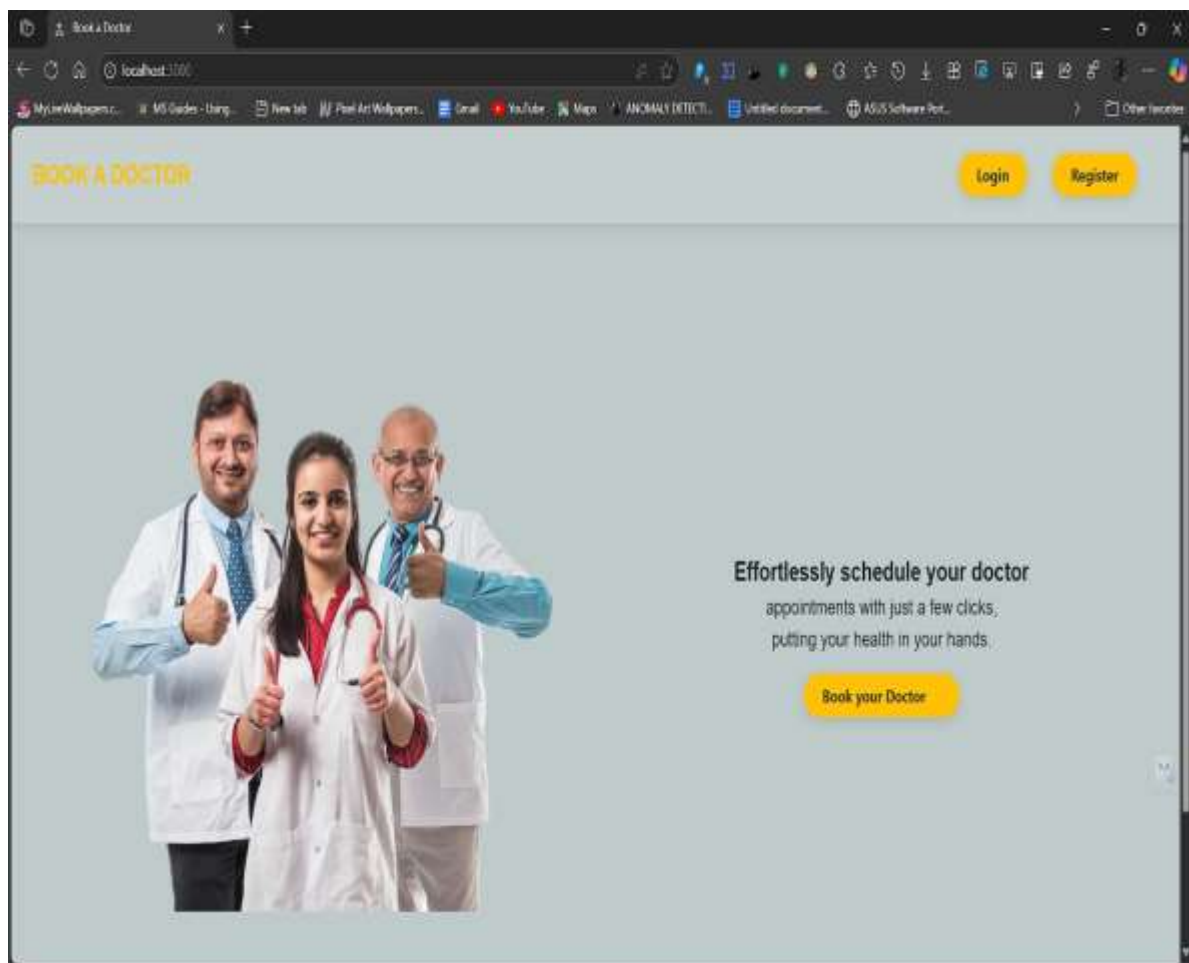
Axios

Antd

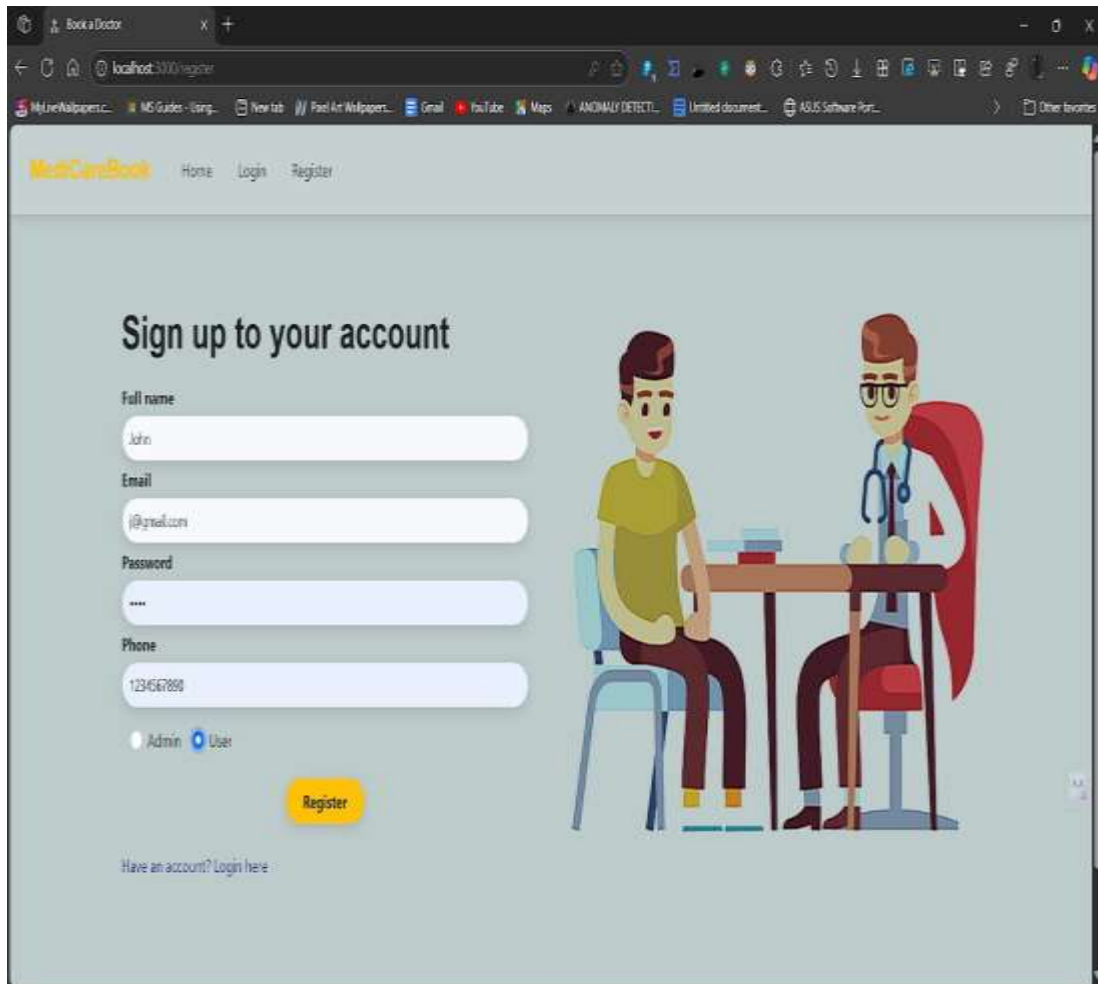
mdb-react-ui-kit

react-bootstrap

On completing the development part, we then run the application one last time to verify all the functionalities and look for any bugs in it. The application's user interface looks a bit like the one provided below.



Register page:



The screenshot shows a web browser window with the address bar displaying 'localhost:3000/register'. The browser's tab is titled 'Book a Doctor'. The page features a navigation bar with 'Home', 'Login', and 'Register' links. The main heading is 'Sign up to your account'. Below this, there are four input fields: 'Full name' (containing 'John'), 'Email' (containing 'j@gmail.com'), 'Password' (containing four dots), and 'Phone' (containing '1234567890'). At the bottom of the form, there are two radio buttons: 'Admin' (unselected) and 'User' (selected). A yellow 'Register' button is positioned below the radio buttons. To the right of the form is an illustration of a doctor in a white coat and stethoscope sitting at a desk with a patient. At the bottom left, there is a link that says 'Have an account? Login here'.

Book a Doctor

localhost:3000/register

Home Login Register

Sign up to your account

Full name

John

Email

j@gmail.com

Password

....

Phone

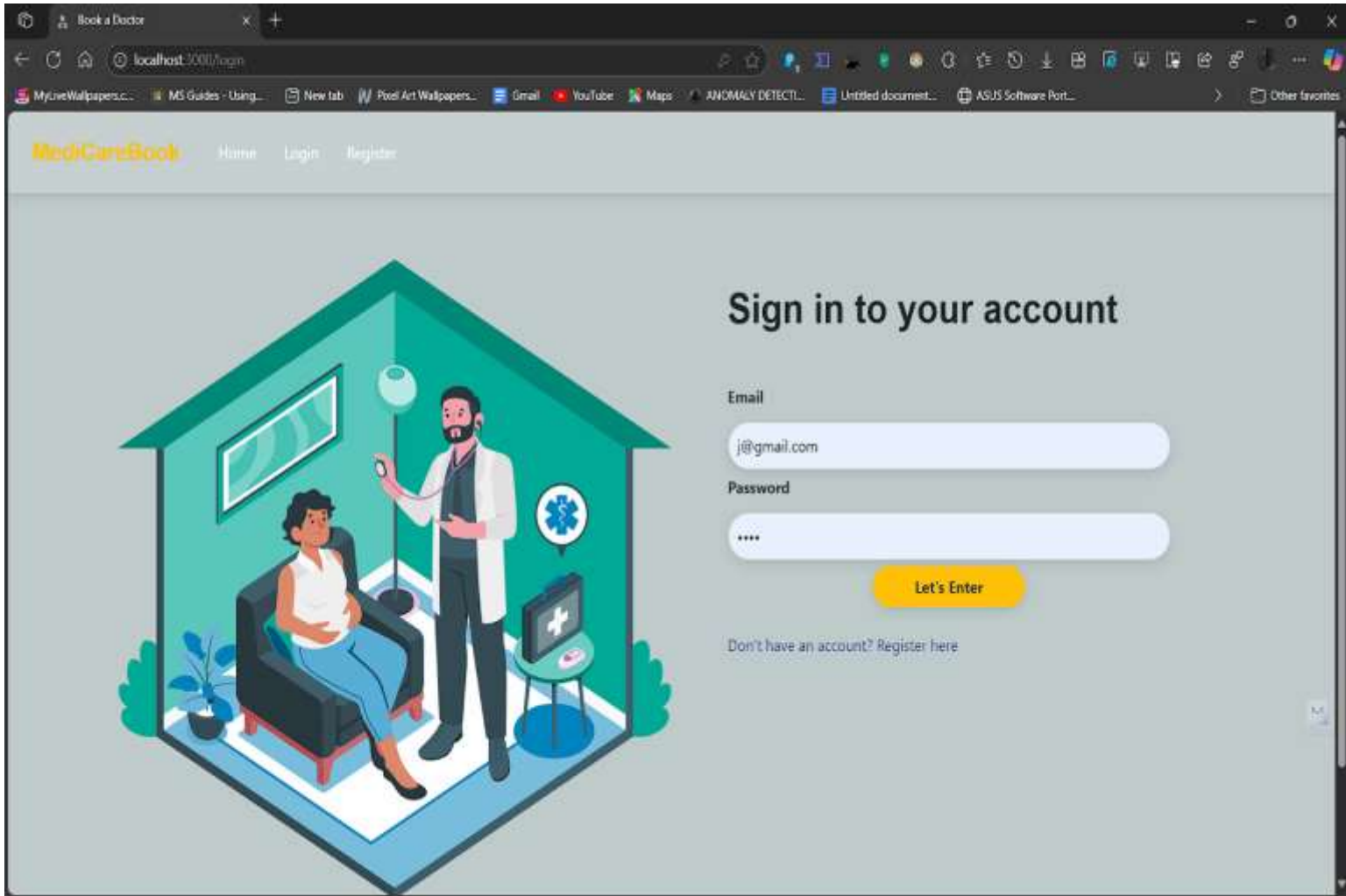
1234567890

☐ Admin ☒ User

Register

Have an account? [Login here](#)

Login page:



Admin Dashboard:

MediCareBook

📅 Users

👨 Doctor

Logout

🔔 Hi..Admin

All Appointments for Admin Panel

Appointment ID	User Name	Doctor Name	Date	Status
672f7a9b4c8952b18190cb7b	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
6730423aaa10078f304cce6e	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
6730a73a6adc4e5cc1d89db3	User	Koushick	Sun Nov 10 2024 17:59:00 GMT+0530 (India Standard Time)	approved

Doctor dashboard:

Book A Doctor

📅 Appointments

👤 Apply doctor

🚪 Logout

🔔 User

🟢 Doctor Registration request sent successfully

Apply for Doctor

Personal Details:

* Full Name:

* Phone:

* Email:

* Address:

Professional Details:

* Specialization:

* Experience:

* Fees:

* Timings: →

© 2023 Copyright: MediCareBook

Admin approve doctor:

MediCareBook

Users

Doctor

Logout

Hi..Admin

Successfully updated approve status of the doctor!

All Doctors

Key	Name	Email	Phone	Action
672f7a384c8952b18190cb90	Koushick	ki@gmail.com	09176478438	Reject
67307e6892c4412ed47f29f1	Karthick	Ka@gmail.com	09176478438	Reject
6730f7955b6839a24169d7a3	SHIVA	user@gamil.com	91755584121	Approve
6730f7955b6839a24169d7a7	SHIVA	user@gamil.com	91755584121	Approve

© 2023 Copyright: MediCareBook

BOOK DOCTOR :

Book A Doctor

📅 Appointments

👤 Apply doctor

🚪 Logout

🔔 Mouli

Dr. Koushick

Phone: 09176478438

Address: chennai

Specialization: ENT

Experience: 5 Yrs

Fees: 1000

Timing: 07:00 : 12:00

Book Now

Dr. SHIVA

Phone: 91755584121

Address: chennai

Specialization: Blood

Experience: 2 Yrs

Fees: 5001

Timing: 06:00 : 12:00

Book Now

Booking appointment

Doctor Details:

Name: SHIVA

Specialization: Blood

Appointment Date and Time:

10-11-2024 11:44 PM

Documents

Choose File bc.png

Close

Book

© 2023 Copyright: MediCareBook

ALL HISTORY :

<div>MediCareBook</div> <div><div></div><div>Users</div></div> <div><div></div><div>Doctor</div></div> <div><div></div><div>Logout</div></div>	Hi..Admin				
	All Appointments for Admin Panel				
	Appointment ID	User Name	Doctor Name	Date	Status
	672f7a9b4c8952b18190cb7b	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
	6730423aaa10078f304cce6e	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
	6730a73a6adc4e5cc1d89db3	User	Koushick	Sun Nov 10 2024 17:59:00 GMT+0530 (India Standard Time)	approved