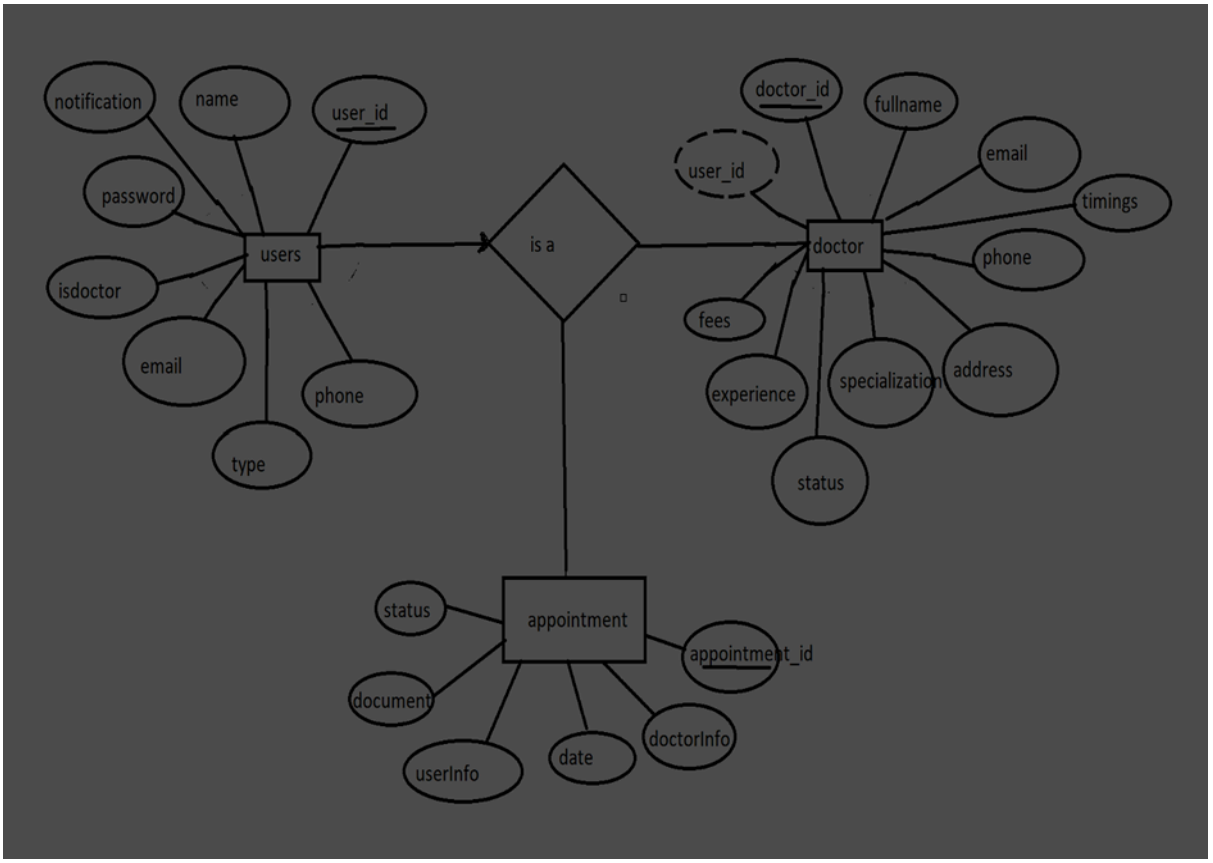
ER Diagram

DocSpot: Seamless Appointment Booking for Health



Here there is 3 collections namely users, doctors, and appointments which have their own fields in

Users:

1. **_id: (MongoDB creates by unique default)**
2. **name**
3. **email**
4. **notification**
5. **password**
6. **isdoctor**
7. **type**
8. **phone**

Doctor:

1. **userID: (can be act as foreign key)**
2. **_id: (MongoDB creates by unique default)**
3. **fullname**
4. **email**
5. **timings**
6. **phone**

7. **address**
8. **specialization**
9. **status**
10. **experience**
11. **fees**

Appointment

1. **_id: (MongoDB creates by unique default)**
2. **doctorInfo**
3. **date**
4. **userInfo**
5. **document**
6. **status**

Prerequisites

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:**

  Momentjs 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/

- **React.js:**

  React.js is a popular JavaScript library for building user interfaces. It enables developers to create interactive and reusable UI components, making it easier to build dynamic and responsive web applications.

  Install React.js, a JavaScript library for building user interfaces.

  Follow the installation guide: https://reactjs.org/docs/create-a-new-react-app.html

- **Antd:**

  Ant Design is a React. js UI library that contains easy-to-use components that are useful for building interactive user interfaces. It is very easy to use as well as integrate. It is one of the smart options to design web applications using react.

  Follow the installation guide: https://ant.design/docs/react/introduce

- **HTML, CSS, and JavaScript: Basic knowledge of HTML for creating the structure of your app, CSS for styling, and JavaScript for client-side interactivity is essential.**
- **Database Connectivity: Use a MongoDB driver or an Object-Document Mapping (ODM) library like Mongoose to connect your Node.js server with the MongoDB database and perform CRUD (Create, Read, Update, Delete) operations. To Connect the Database with Node JS go through the below provided link:**

https://www.section.io/engineering-education/nodejs- mongoosejs-mongodb/

- **Front-end Framework: Utilize Reactjs to build the user-facing part of the application, including entering booking room, status of the booking, and user interfaces for the admin dashboard.**

For making better UI we have also used some libraries like material UI and boostrap.

Install Dependencies:

• Navigate into the cloned repository directory:

cd book-a-doctor

• Install the required dependencies by running the following commands:

cd frontend

npm install

cd ../backend

npm install

Start the Development Server:

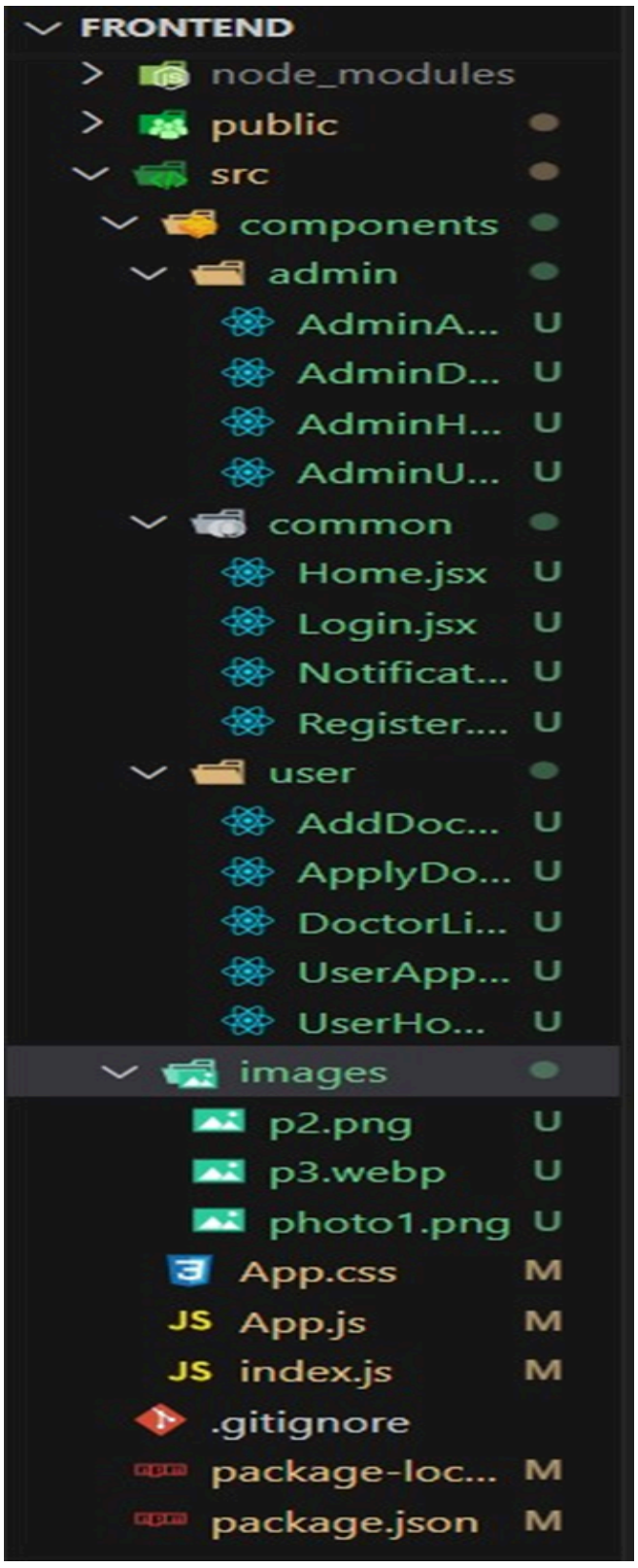
• To start the development server, execute the following command:

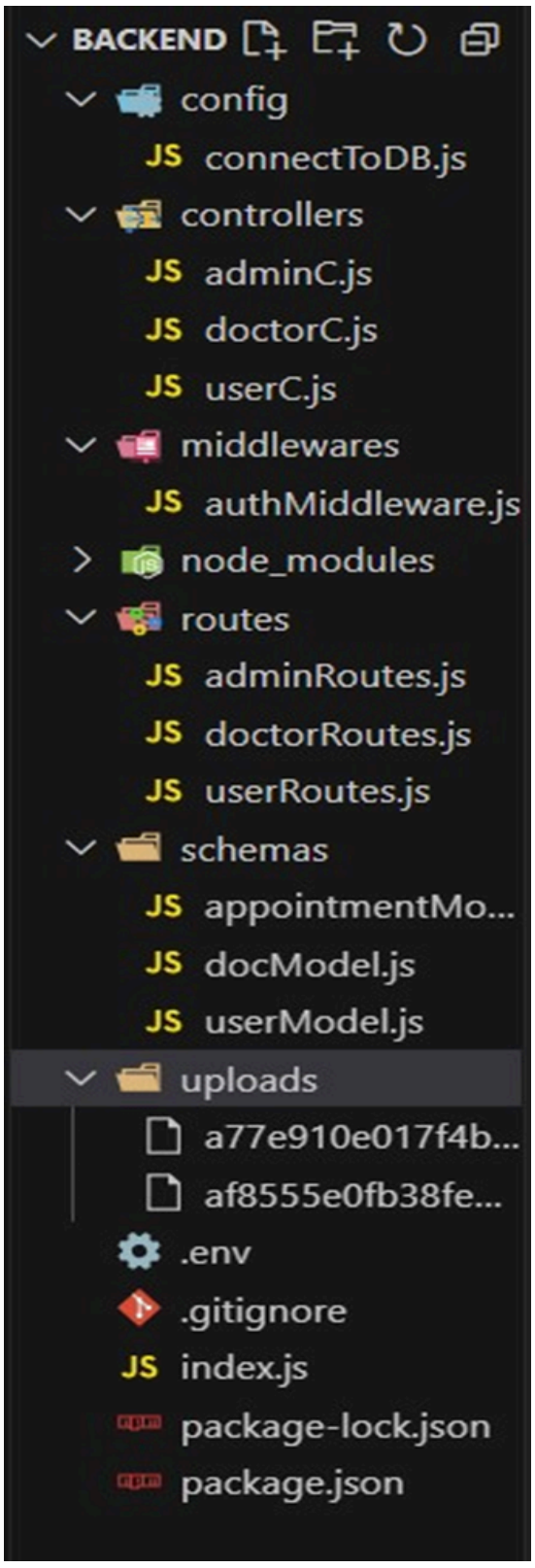
npm start

• The book a doctor app will be accessible at http://localhost:3000

You have successfully installed and set up the online complaint registration and management app on your local machine. You can now proceed with further customization, development, and testing.

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
App.js M
index.js M
.gitignore
package-loc... M
package.json M



The first image is of frontend part which is showing all the files and folders that have been used in UI development

The second image is of Backend part which is showing all the files and folders that have been used in backend development

Application Flow: The project has 2 type of user – Customer and Doctor and other will be Admin which takes care to all the user. The roles and responsibilities of these two types of users can be inferred from the API endpoints defined in the code. Here is a summary:

Customer/Ordinary:

1. **Create an account and log in to the system using their email and password.**
2. **They will be shown automatically all the doctors in their dashboard.**
3. **After clicking on the Book Now, a form will generate in which date of appointment and documents need to send.**
4. **They can sees the status of their appointment and can get a notification if the appointment is schedule or not.**
5. **The user can also cancel it`s booking in booking history page and can change the status of booking.**

Admin:

1. **Manage and monitor the overall operation of the appointment and the type of users and doctors to the application.**
2. **He monitors the applicant of all doctors and approve them and then doctors are registered in the app.**
3. **Implement and enforce platform policies, terms of service, and privacy regulations.**

Doctor:

1. **Gets the approval from the admin for his doctor account.**
2. **Manages all the appointments that are getting from the users**

SETUP AND CONFIGURATIONFolder setup

Setup & configuration

Let’s start with the project development with the help of the given activities

Folder setup

Duration: 1 Hrs

Skill Tags:

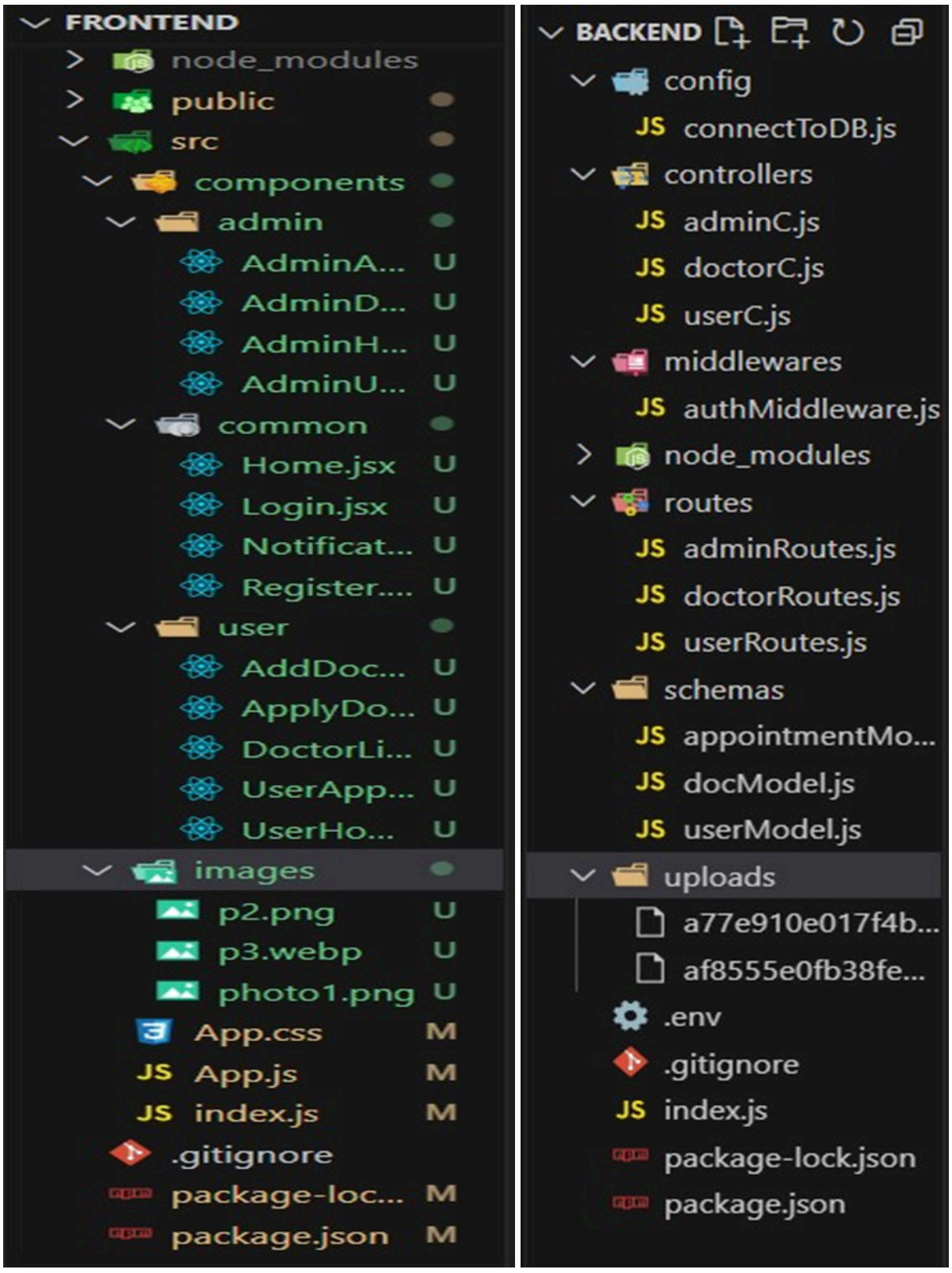
- **Folder setup:**

1. **Create frontend and**
2. **Backend folders**

. Open the backend folder to install necessary tools

For backend, we use:

- **cors**
- **bcryptjs**
- **express**
- **dotenv**
- **mongoose**
- **Multer**
- **Nodemon**
- **jsonwebtoken**


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

BACKEND DEVELOPMENT Setup express server

Backend Development

In this milestone explains about Backend Development

Duration: 1 Hrs

Skill Tags:

- **Setup express server**

1. **Create index.js file in the server (backend folder).**
2. **define port number, mongodb connection string and JWT key in env file to access it.**
3. **Configure the server by adding cors, body-parser.**

- **Add authentication: for this,**

1. **You need to make a middleware folder and in that make an authMiddleware.js file for the authentication of the projects and can use in.**

Ref: link

DATABASE Configure MongoDB

Database

In This Milestone Explains about Database

Duration: 1 Hrs

Skill Tags:

Configure MongoDB

-

1. **Import mongoose.**
2. **Add database connection from config.js file present in config folder**
3. **Create a model folder to store all the DB schemas like renter, owner and booking, properties schemas.**
   ref: link

FRONTEND DEVELOPMENT
Installation of tools

Frontend Development

In this milestone explains about Frontend Development

Installation of required tools

Duration: 1 Hrs

Skill Tags:

- **Installation of required tools:**

- **For frontend, we use:**

1. **React**
2. **Bootstrap**
3. **Material UI**
4. **Axios**
5. **Antd**
6. **mdb-react-ui-kit**

PROJECT IMPLIMENTATION

Project Implementation

In this milestone explains about project implementation

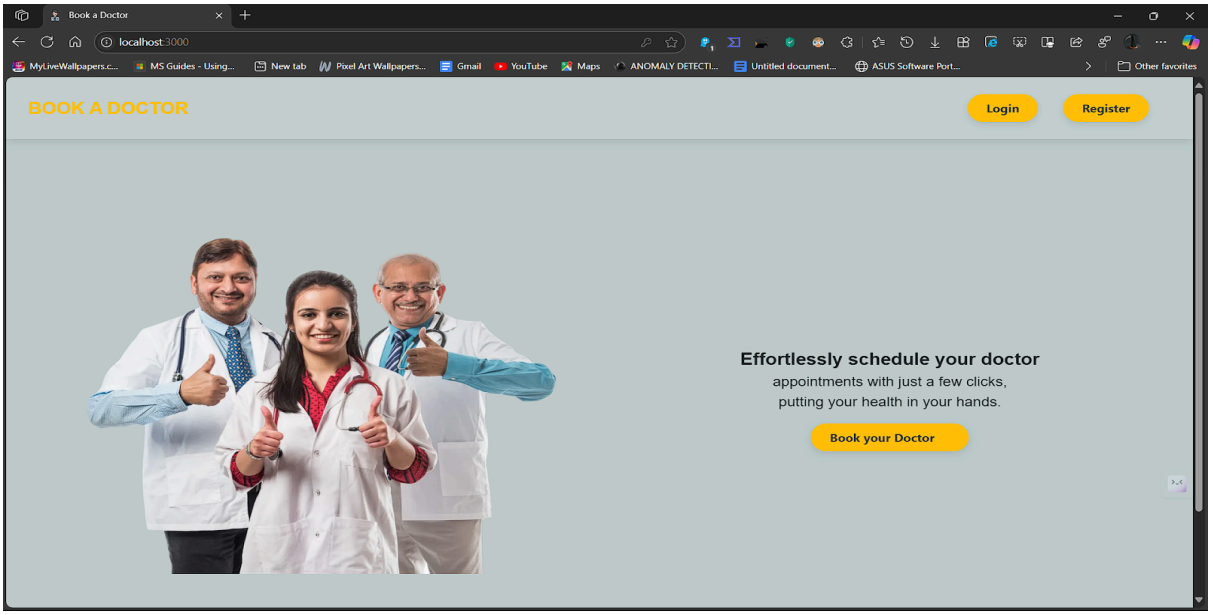
Landing page

Duration: 1 Hrs

Skill Tags:

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.

Landing page



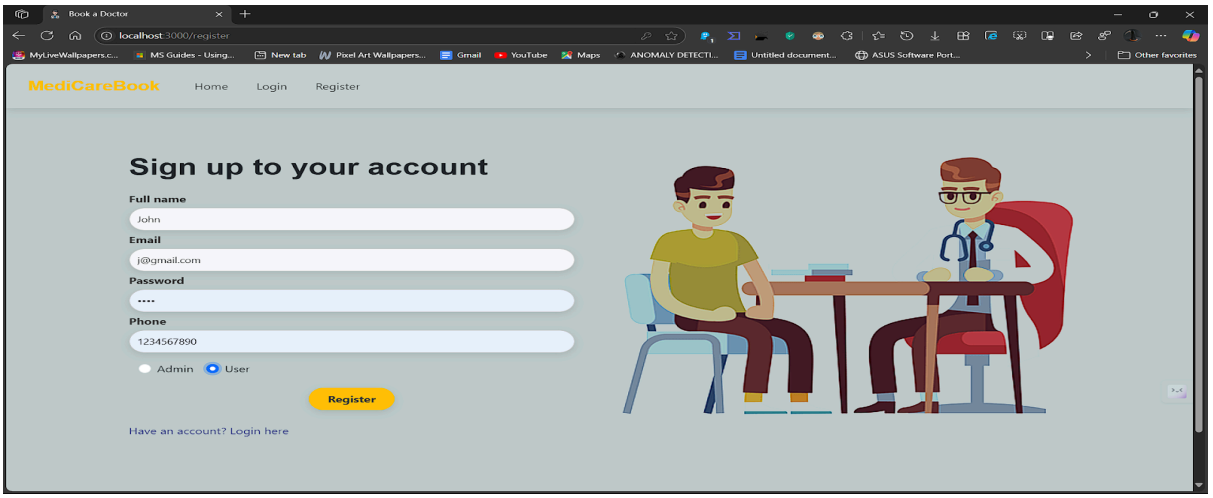
Register page

Register page

Duration: 1 Hrs

Skill Tags:

Register page:



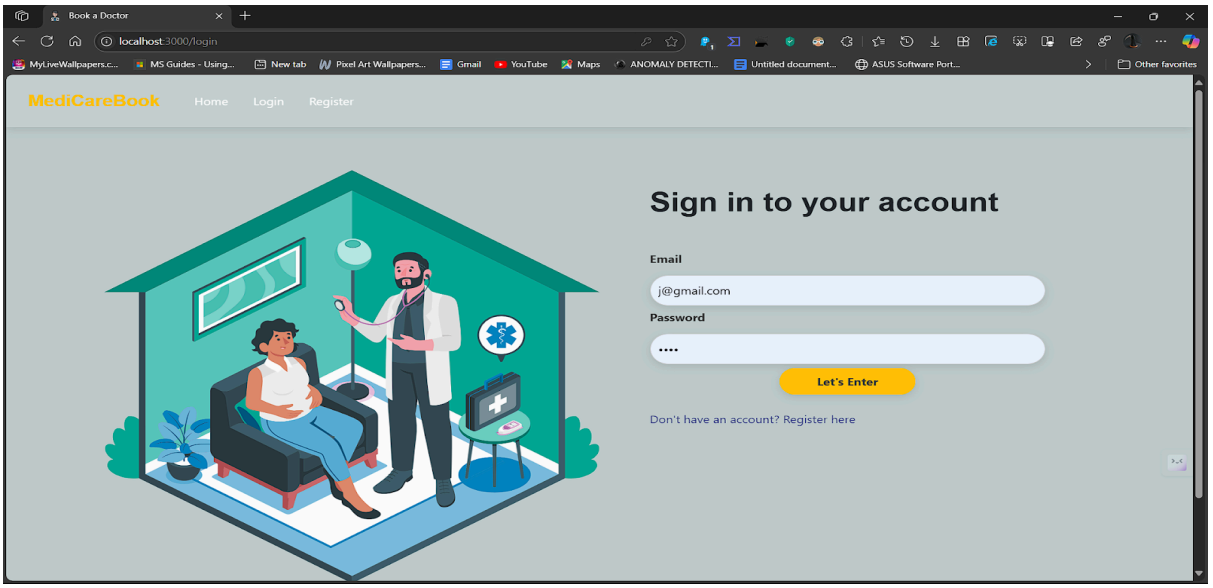
Login page

Login page

Duration: 1 Hrs

Skill Tags:

Login page:



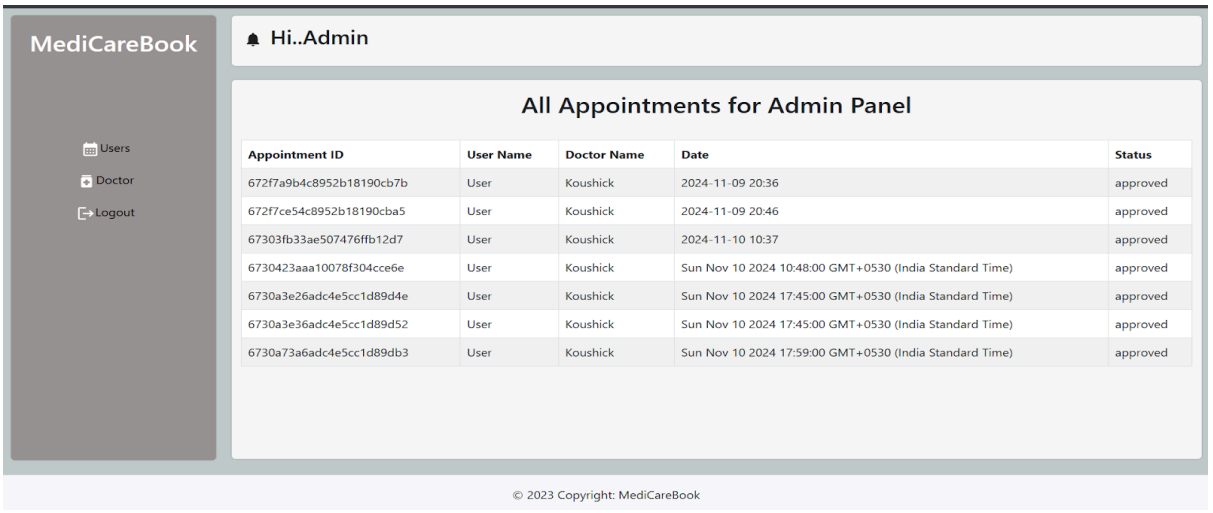
Admin Dashboard

Admin Dashboard

Duration: 1 Hrs

Skill Tags:

Admin Dashboard:



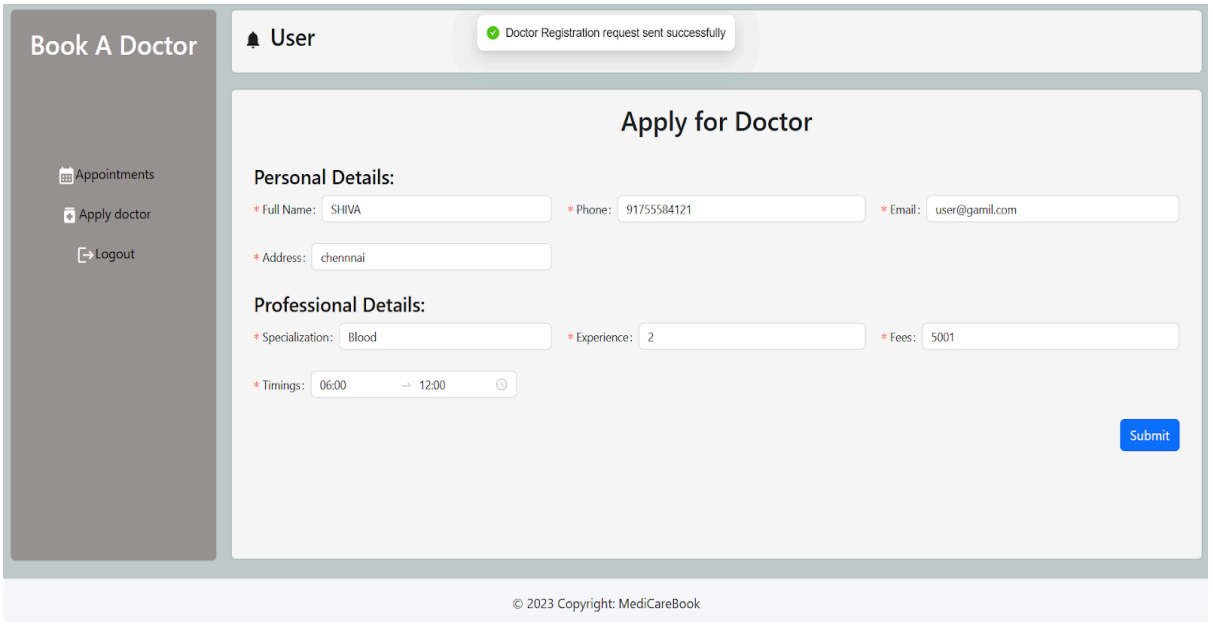
Doctor dashboard

Doctor dashboard

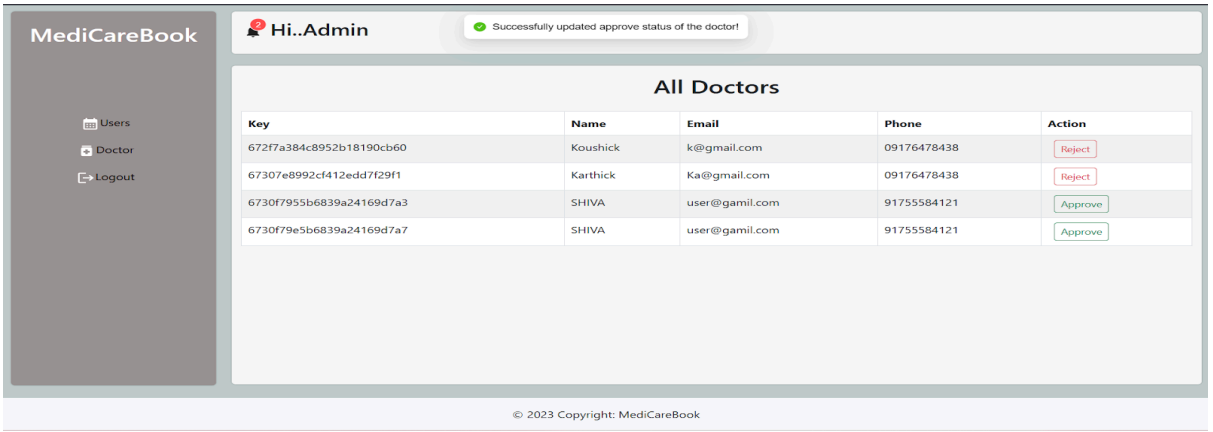
Duration: 1 Hrs

Skill Tags:

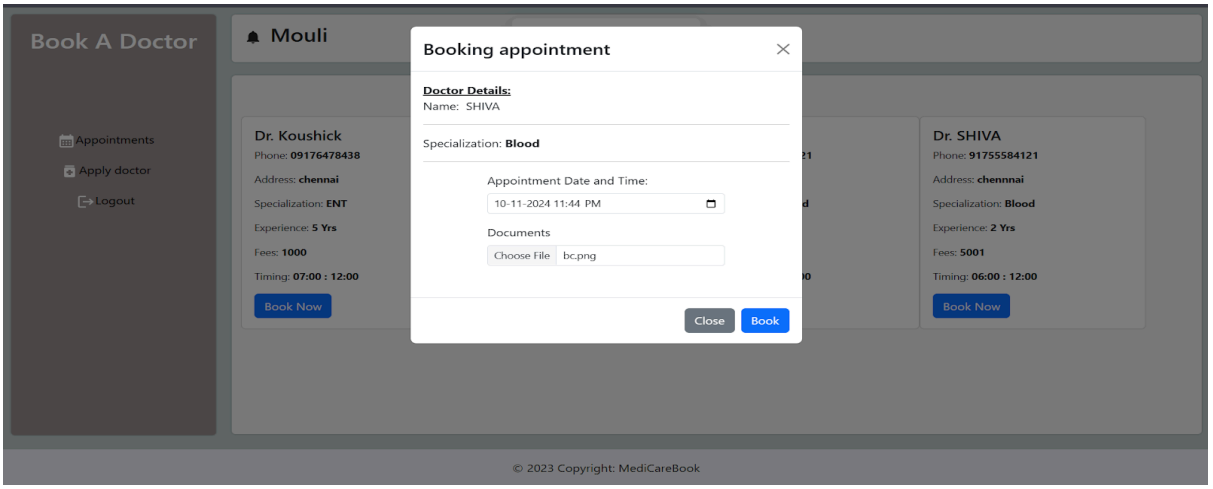
Doctor dashboard:



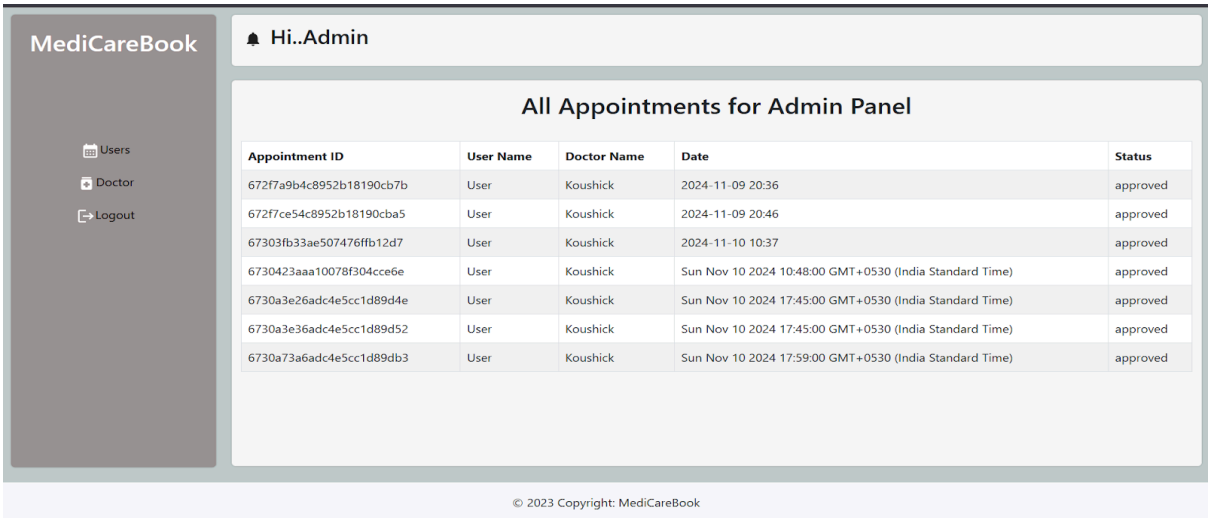
Admin approve doctor:



BOOK DOCTOR :


Book A Doctor
Mouli
Appointments
Apply doctor
Logout
Dr. Koushick
Phone: 09176478438
Address: chennai
Specialization: ENT
Experience: 5 Yrs
Fees: 1000
Timing: 07:00 : 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
Dr. SHIVA
Phone: 91755584121
Address: chennnai
Specialization: Blood
Experience: 2 Yrs
Fees: 5001
Timing: 06:00 : 12:00
Book Now
© 2023 Copyright: MediCareBook

ALL HISTORY :


MediCareBook
Hi..Admin
Users
Doctor
Logout
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
© 2023 Copyright: MediCareBook

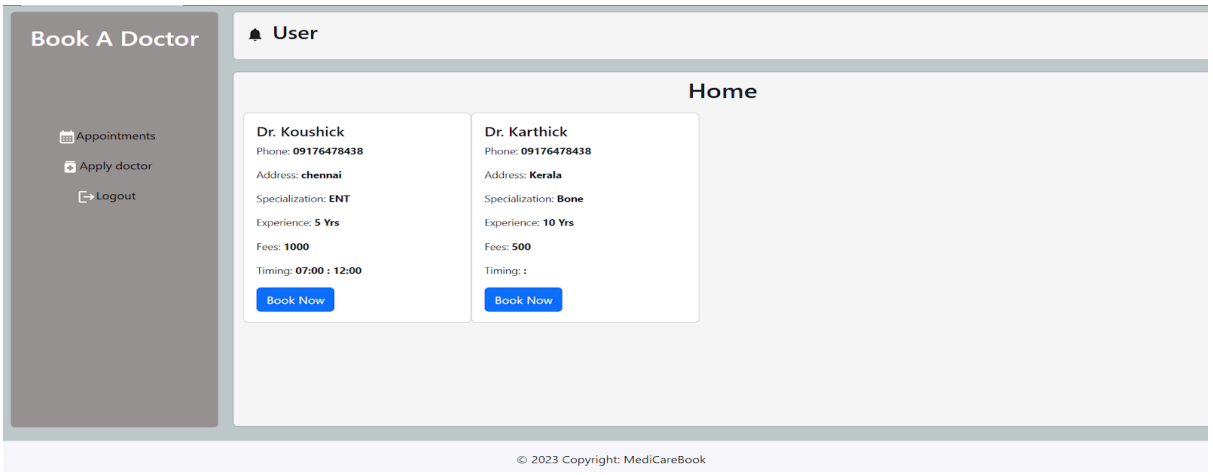
User dashboard

User dashboard

Duration: 1 Hrs

Skill Tags:

User dashboard:



Note: For code drive, click on link and demo link, click on link

Project FlowDemo Video and reference code link

Project FlowDemo Video and reference code link

reference video link:

- https://drive.google.com/drive/folders/1pteT8STdObONWwELNDHRK9biItLuiJ-1?usp=sharing

reference code link:

Doctor Appointment Booking Using MERN Source Code