

📄 DocSpot:SeamlessAppointmentBookingforHealth

Full Stack MERN Project Documentation

📄 1.Introduction

Project Title:

DocSpot:SeamlessAppointmentBookingforHealth

TeamID:LTVIP2025TMID53714

BoyinaVijayaSimhaReddy.

Boyina Upendra.

Buddi Divya.

Burla Bindu.

Overview:

DocSpot is a full-stack healthcare web application that streamlines the process of booking medical appointments.Itprovides auser-friendlyinterfaceforpatientstobookappointments online,doctors to manage their schedules, and administrators to oversee system operations. The app includes securerole-basedlogins andintegrated online paymentsfor a complete appointmentexperience.

📄 2.ProjectOverviewPurpose:

Toprovideacentralized,convenient,andsecureplatformforhealthcareappointmentmanagement accessible by patients, doctors, and administrators.

KeyFeatures:

- User(Patient)registration,login,andappointmentbooking
- Doctorloginandappointment management dashboard
- Adminpanel tomanageusersandappointments
- Razorpayintegrationforonlinepayments
- SecureauthenticationusingJWT

📄 3.SystemArchitecture

Frontend (React.js)

- BuiltwithReactandReactRouter
- UIcomponentsfromMaterialUI,Bootstrap,AntDesign

- AxiosforHTTPrequests

- Responsivedesignformobileanddesktop

Backend(Node.js+Express.js)

- RESTfulAPIarchitecture
- Role-basedaccesscontrol
- RazorpaypaymentAPIintegration
- JWTauthenticationandroute protection

Database(MongoDB)

- Collections:
 - users(patients)
 - doctors
 - appointments
 - admins
 - payments
- Mongooseforschema definitionsanddatamodeling

📖 4.SetupInstructions

Prerequisites:

- Node.js(v18+)
- MongoDB(LocalorAtlas)
- npm
- Git

Installation&Setup:

bash

Copycode

#Clonerepository

gitclonehttps://github.com/your-username/docspot.git

#Frontendsetup cd

client

npminstall

Frontendterminal:

```
C:\Users\chinnu\Desktop\Prescripto>cd prescripto-full-stack/frontend
C:\Users\chinnu\Desktop\Prescripto\prescripto-full-stack\frontend>npm run dev
> frontend@0.0.0 dev
> vite

VITE v5.3.3 ready in 211 ms
VITE v5.3.3 ready in 211 ms
VITE v5.3.3 ready in 211 ms
VITE v5.3.3 ready in 211 ms
VITE v5.3.3 ready in 211 ms
VITE v5.3.3 ready in 211 ms
→ Local: http://localhost:5173/
```

#Backendsetup

cd ../servernpm

install

Backendterminal:

```
Microsoft Windows [Version 10.0.22631.5039]
(c) Microsoft Corporation. All rights reserved.

C:\Users\chinnu\Desktop\Prescripto>cd prescripto-full-satack/backend
The system cannot find the path specified.

C:\Users\chinnu\Desktop\Prescripto>cd prescripto-full-stack/backend

C:\Users\chinnu\Desktop\Prescripto\prescripto-full-stack\backend>npm start

> backend@1.0.0 start
> node server.js

Server started on PORT:4000
Database Connected
```

.env file in /server/

MONGO_URI=your_mongo_uri

JWT_SECRET=your_jwt_secret

RAZORPAY_KEY=your_razorpay_key

RAZORPAY_SECRET=your_razorpay_secret

PORT=5000

5.FolderStructure Client:

pgsqlCopy

code

client/

└─public/

└─src/

| └─components/

| └─pages/

| └─context/

| └─App.js,index.js

Server:

pgsqlCopy

code

server/

└─controllers/

└─routes/

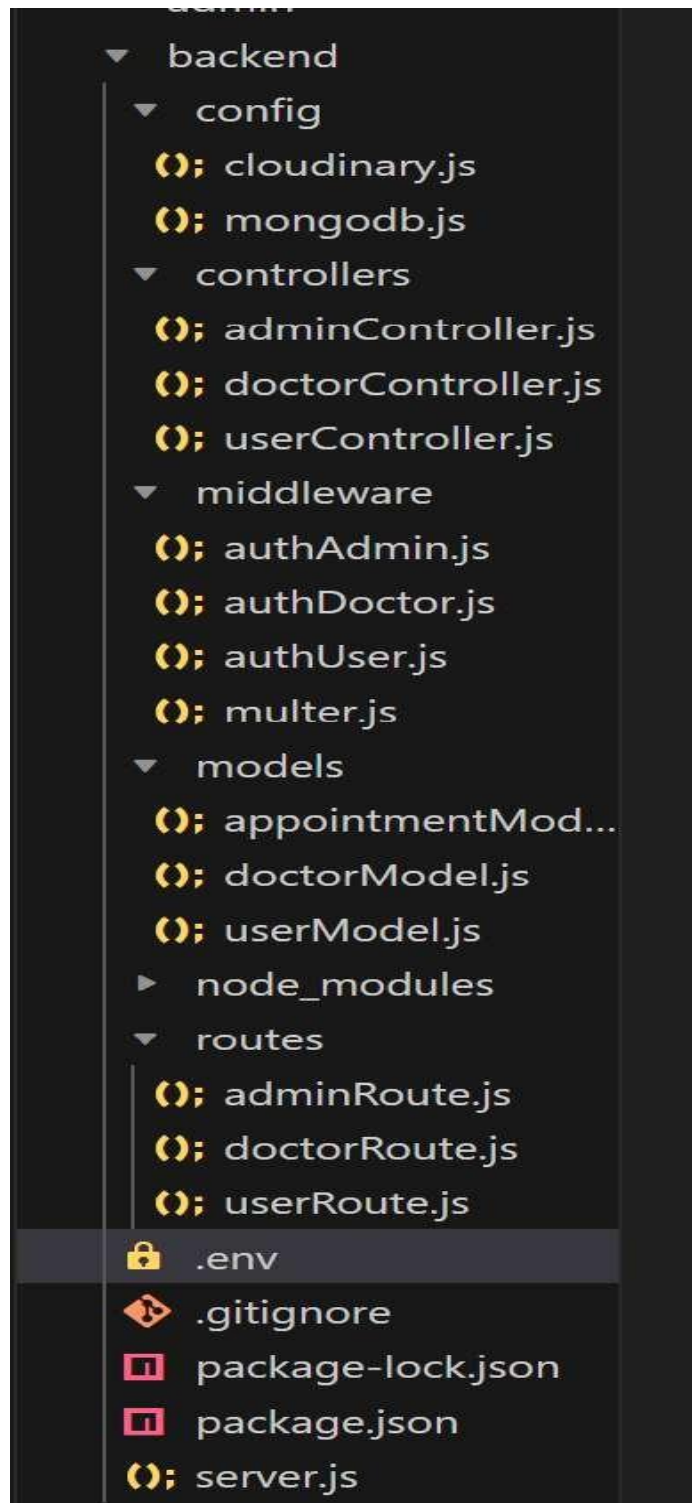
└─models/

└─config/

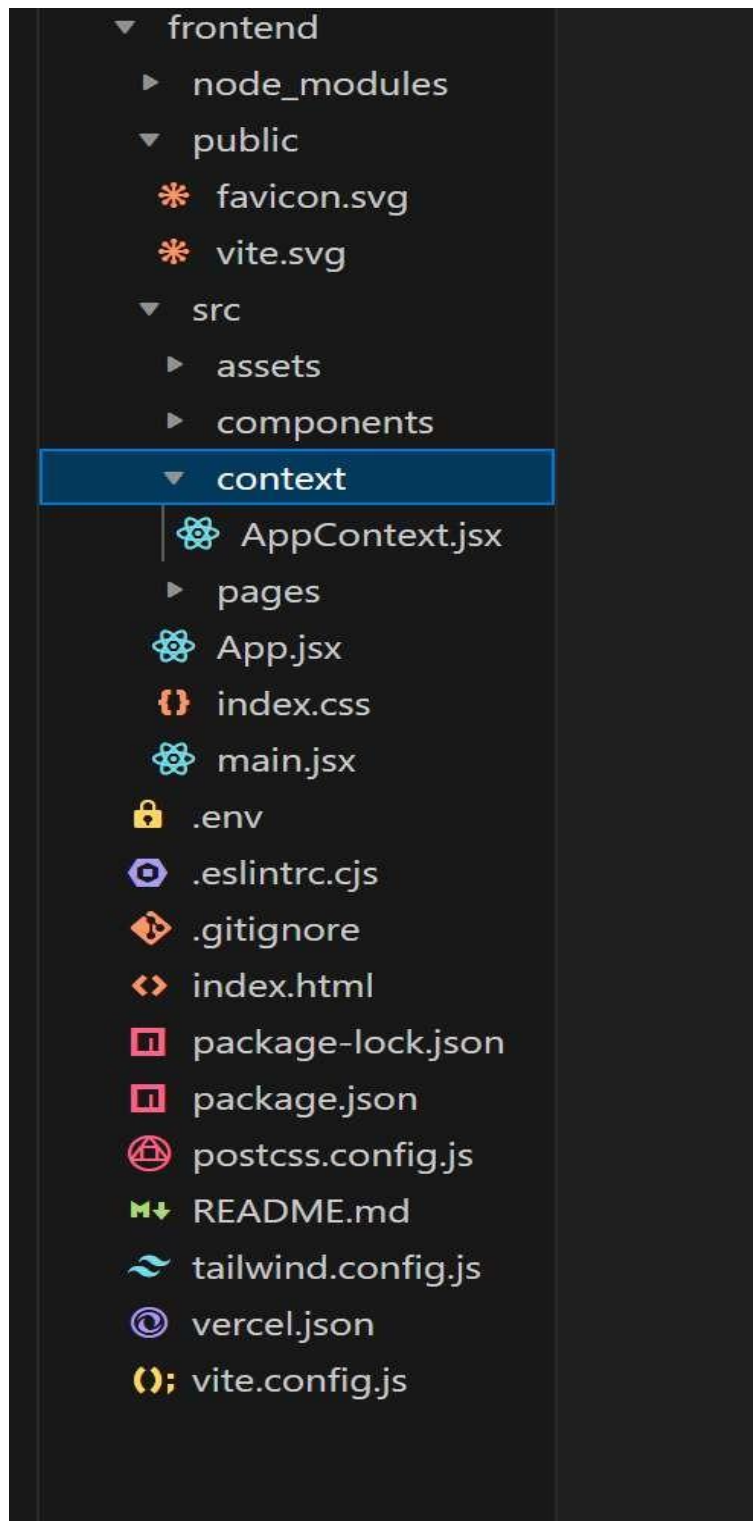
└─middlewares/

└─server.js

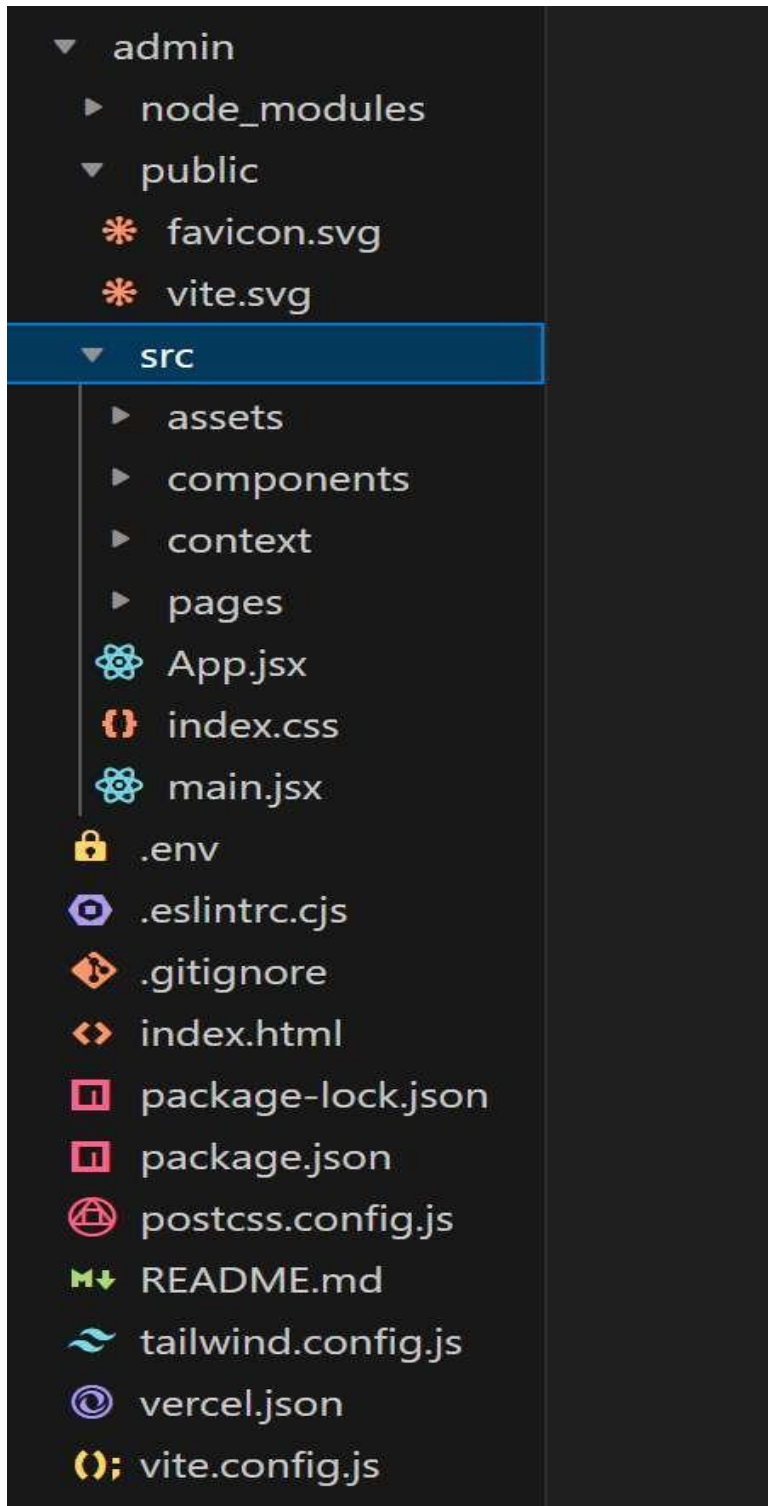
BackendFolderStructure:



FrontendFolderStructure:



AdminFolderSetup:



❓6. Running the Application

bash

Copycode

#Frontend

cd client

npm start

#Backend

cd server

npmstart

❓7. API Documentation

Method	Endpoint	Description
POST	/api/users/register	Register a new user
POST	/api/users/login	Login as user
POST	/api/doctors/login	Doctor login
POST	/api/admin/login	Admin login
POST	/api/appointments/book	Book an appointment
GET	/api/doctors/:id/appointments	Doctor views appointments
GET	/api/admin/appointments	Admin views all appointments
POST	/api/payments/order	Initiate Razorpay order
POST	/api/payments/verify	Verify payment

8. Authentication & Authorization

- **JWT Tokens** used for secure authentication
- **Protected Routes:** Based on roles (User, Doctor, Admin)
- **Token Flow:**
 - On login, token is generated and stored in local storage
 - Every protected request includes this token in the headers

9. User Interface Screenshots

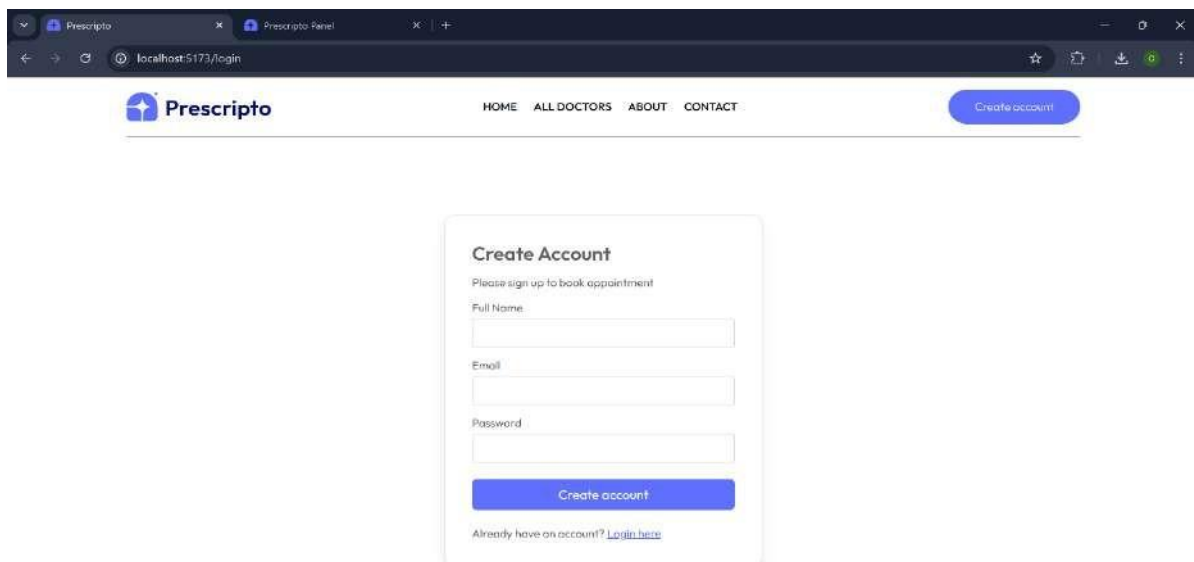
Please upload the following screenshots with the specified names:

Page	Screenshot Name
User Login Page	user-login.png
Doctor Login Page	doctor-login.png
Admin Login Page	admin-login.png
User Dashboard (after login)	user-dashboard.png
Appointment Booking Page	book-appointment.png
Payment Checkout Page	payment-page.png
Doctor Dashboard	doctor-dashboard.png
Admin Dashboard	admin-dashboard.png
Success Message after Bookings	success-booking.png
Payment Success Page	payment-success.png

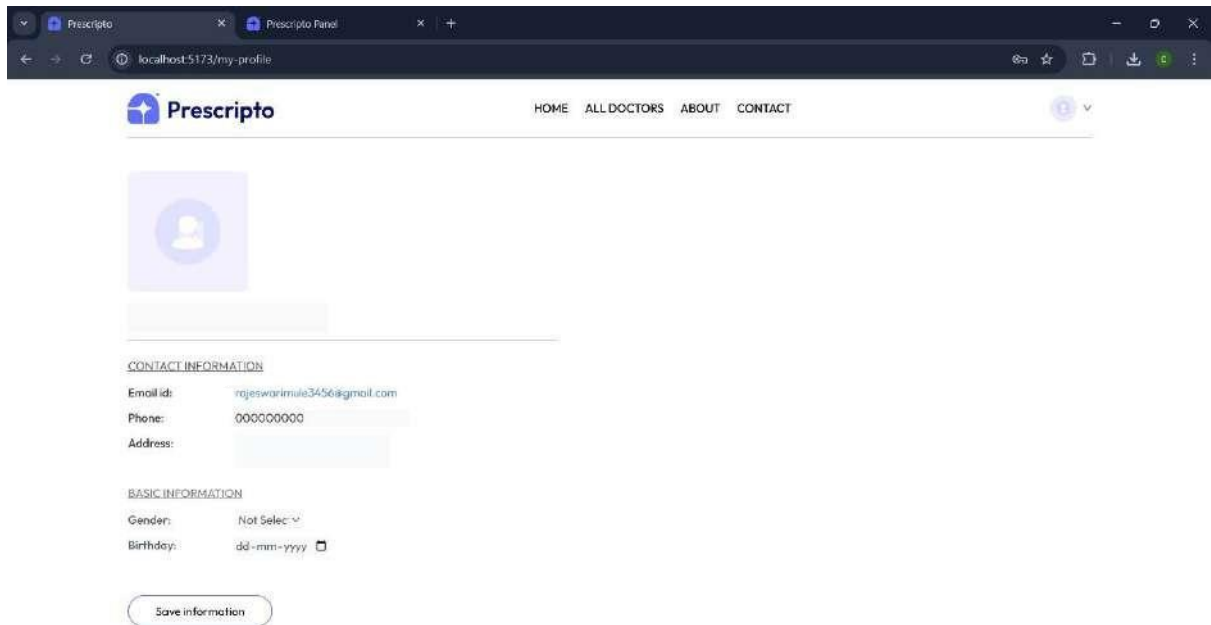
Homepage(website):



UserLogin:



UserDashboard:



The screenshot shows a web browser window with two tabs: 'Prescripto' and 'Prescripto Panel'. The address bar shows 'localhost:5173/my-profile'. The page has a header with the 'Prescripto' logo and navigation links: 'HOME', 'ALL DOCTORS', 'ABOUT', and 'CONTACT'. A user profile section on the left includes a placeholder for a profile picture and a name field. Below this, there are two sections: 'CONTACT INFORMATION' and 'BASIC INFORMATION'. The 'CONTACT INFORMATION' section has fields for 'Email id:' (with value 'rajeswarimule3456@gmail.com'), 'Phone:' (with value '000000000'), and 'Address:'. The 'BASIC INFORMATION' section has fields for 'Gender:' (with value 'Not Select') and 'Birthday:' (with value 'dd-mm-yyyy'). A 'Save information' button is located at the bottom of the form.

Prescripto

HOME ALL DOCTORS ABOUT CONTACT

CONTACT INFORMATION

Email id: rajeswarimule3456@gmail.com

Phone: 000000000

Address:

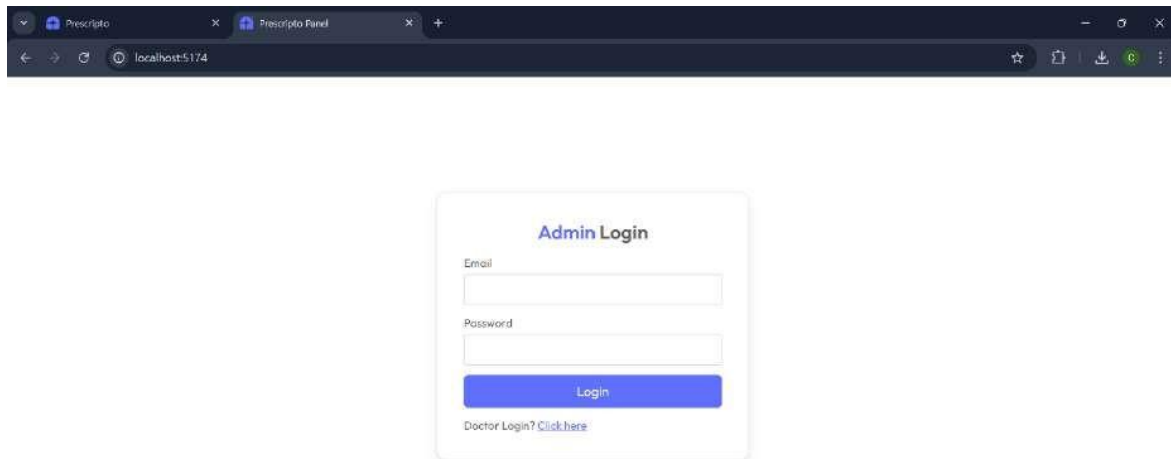
BASIC INFORMATION

Gender: Not Select

Birthday: dd-mm-yyyy

Save information

AdminLoginPage:



The screenshot shows a web browser window with two tabs: 'Prescripto' and 'Prescripto Panel'. The address bar shows 'localhost:5174'. The page displays a login form titled 'Admin Login'. The form has two input fields: 'Email' and 'Password'. Below the password field is a blue 'Login' button. At the bottom of the form, there is a link that says 'Doctor Login? Click here'.

Admin Login

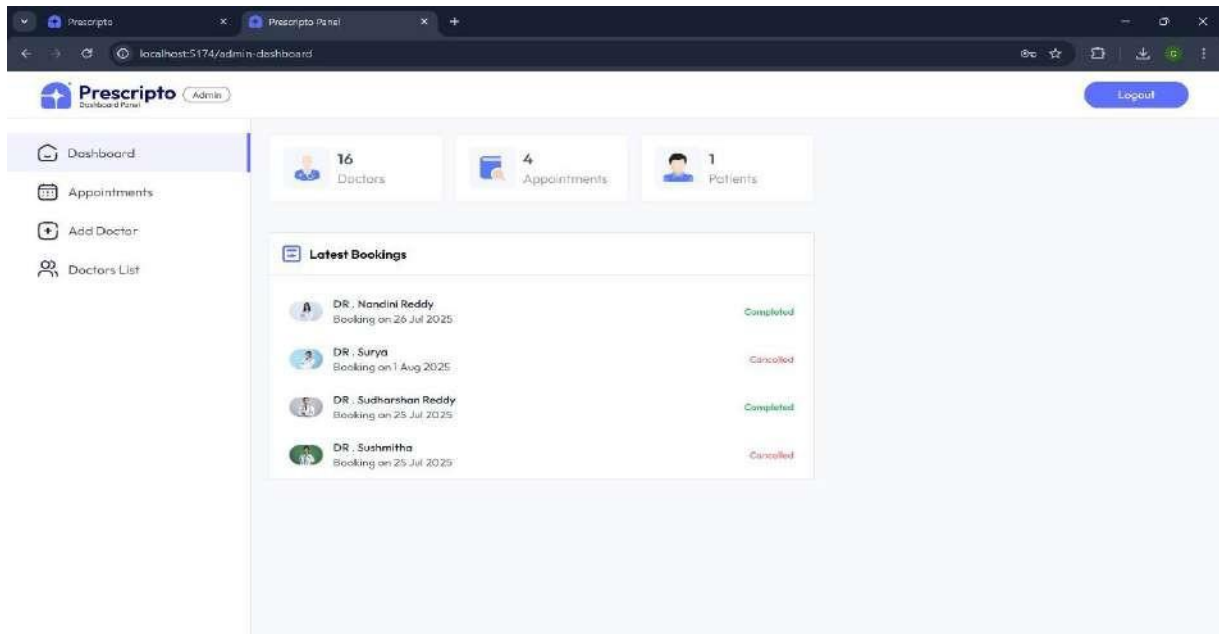
Email

Password

Login

Doctor Login? [Click here](#)

AdminDashboardPage:



DoctorLoginPage:



Doctor Login

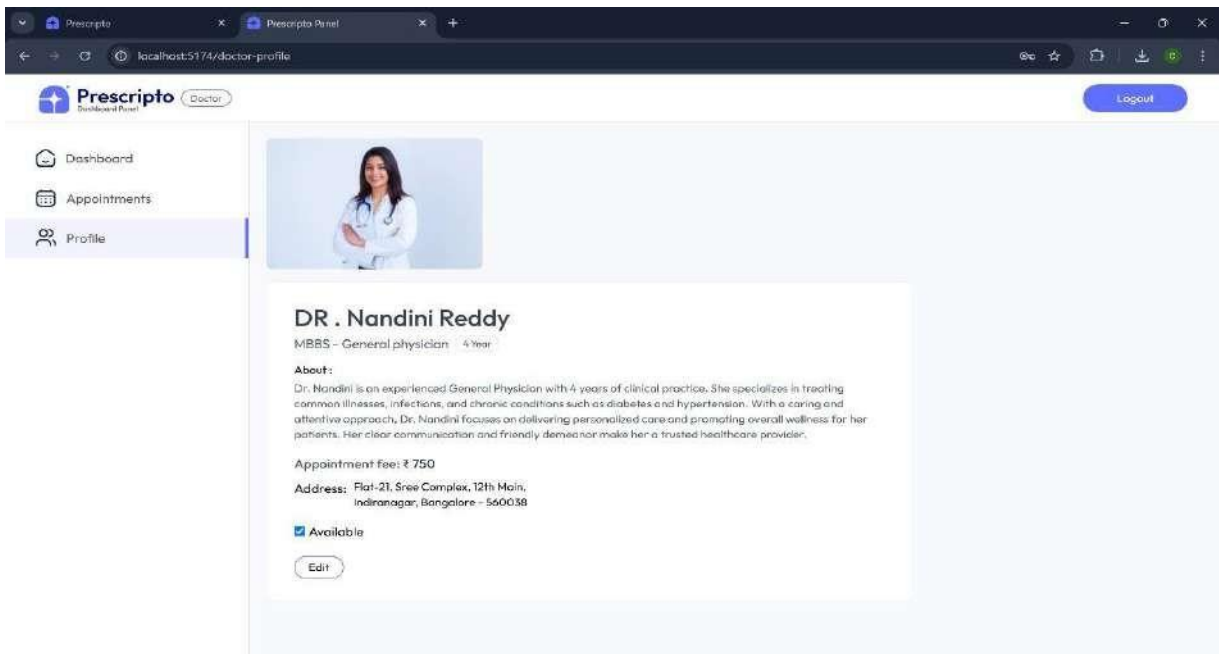
Email

Password

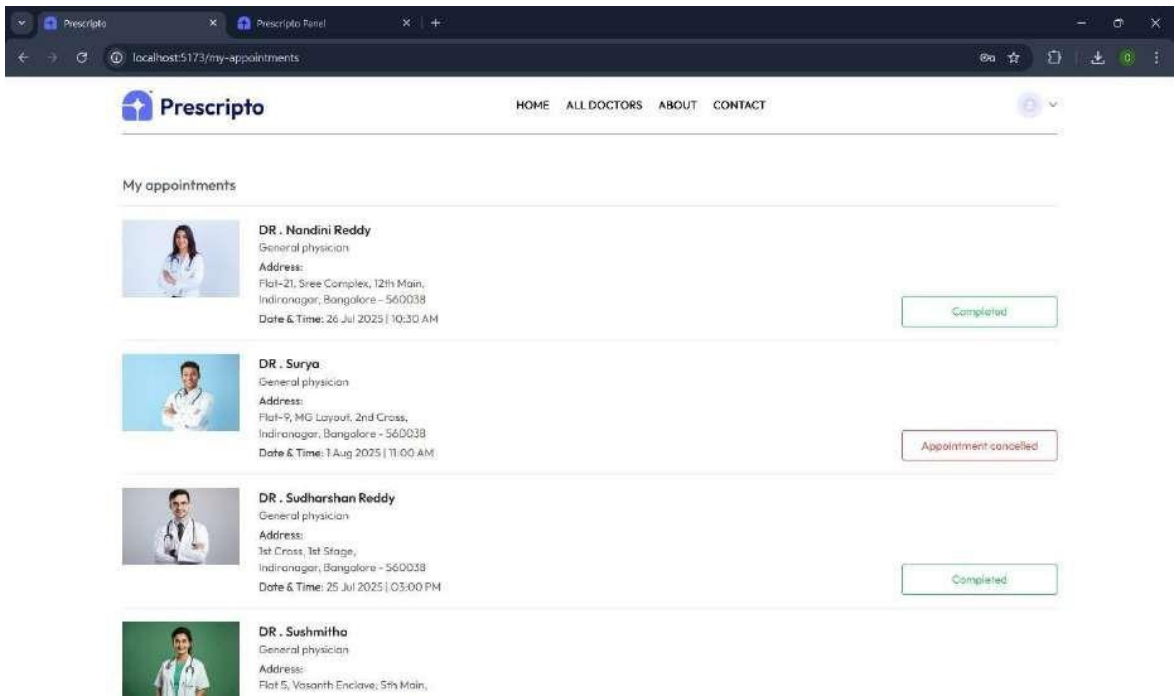
Login

Admin Login? [Click here](#)

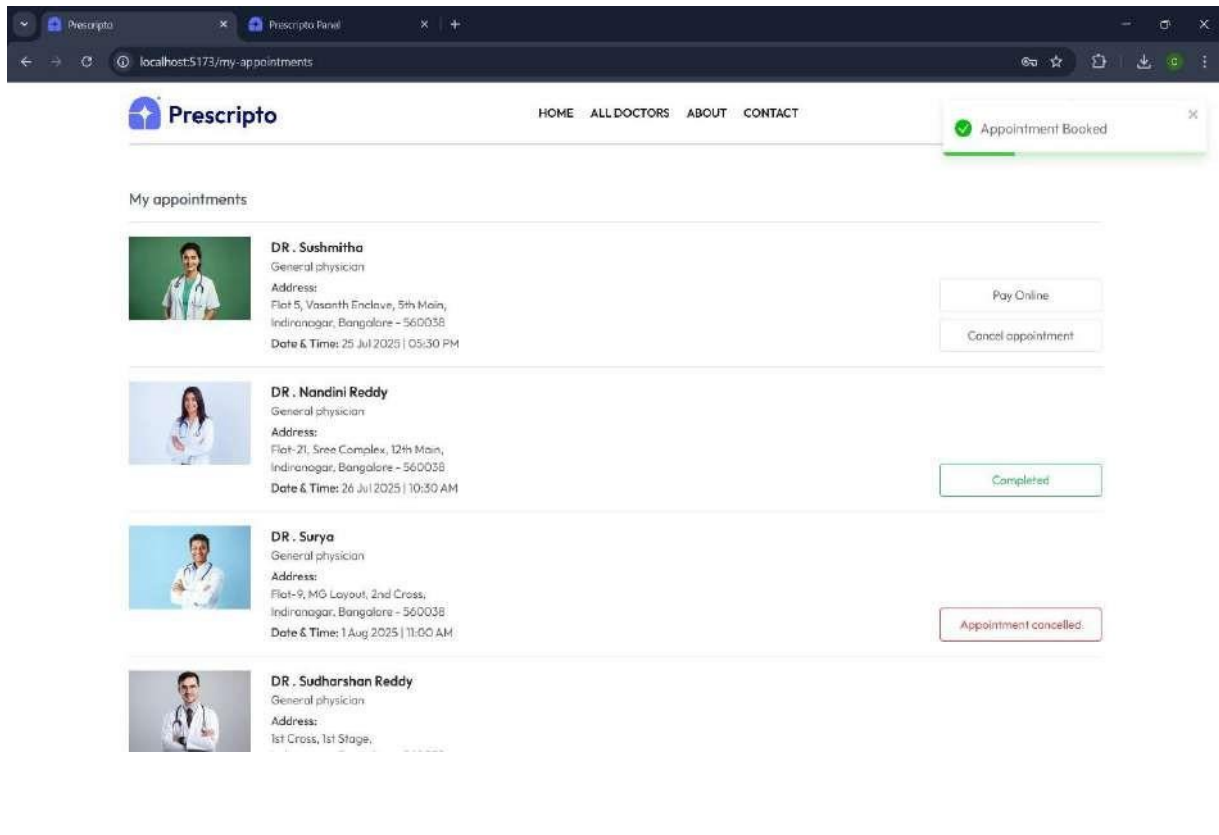
DoctorDashboard:



AppointmentBooking:



AppointmentBookedSuccessfully:



□ 10. Testing

Tools Used:

- Postman—Backend API testing
- Lighthouse—Frontend performance and accessibility
- Browser DevTools—UI and console validation

Testing Coverage:

- User login/logout and protected routes
- Doctor and admin access control
- Appointment conflict handling
- Payments success/failure scenarios

❓ 11. Demo or Showcase

- **Screenshots:** As listed above
- **Optional Demo Video Link:** *[Insert YouTube/Google Drive demo link]*

❓ 12. Known Issues

- No pagination for appointment list
- Doctor profile editing not implemented
- No email/SMS notifications yet

❓ 13. Future Enhancements

- Implement video consultations via WebRTC or Jitsi
- Add appointment filtering (by date/speciality)
- Notifications for appointment reminders
- Email confirmations for patients
- Real-time availability for doctors

❓ 14. Conclusion

DocSpots successfully integrates frontend and backend technologies to deliver a secure and user-friendly platform for medical appointment management. With online payments and role-based control, it lays the groundwork for scalable future enhancements in digital healthcare.