

Department of Computer Science and Engineering,  
Chitkara University

Project Report on  
“Healthcare Appointment System”



Submitted by :

Name - Yashika  
Roll No - 2110991585  
Group - G7 (5th sem)

[Batch 2021-25]

## Problem Statement

Build a system that facilitates appointment scheduling for healthcare services. Include features like user profiles, appointment reminders, and a calendar for healthcare professionals.

## Introduction

This project presents a full stack functional “Doctor Application” developed using the MERN stack (MongoDB, Express.js, React.js, and Node.js). The application facilitates user registration (authentication), doctor appointment, using backend APIs which are implemented from scratch.

## Technologies used

Frontend - React.js, CSS, with Bootstrap and Ant Design for a user friendly and intuitive interface.

Backend - Express.js for API development and Node.js for server-side operations.

Database - MongoDB for storing user data.



# Features

**1. User Authentication with Extended Session Duration:** The application prioritizes user convenience by implementing a token-based authentication system that maintains user sessions for 24 hours. This feature significantly reduces the frequency of logins, enhancing user experience and engagement.

## **2. Appointment Scheduling:**

Admin (Doctor): Doctor will be able to see the appointment notification that user has booked the appointment and is able to respond either "Approved" to give message to user(patient) that doctor is available, and "Reject" to give message to user(patient) that doctor is not available at given time.

## **3. Appointment Reminders:**

User(patient) will receive a reminder in the Notification section about the approval of the appointment.

## **4. Appointment Reminders:**

User(patient) will receive a reminder in the Notification section about either the approval or the rejection of the appointment.

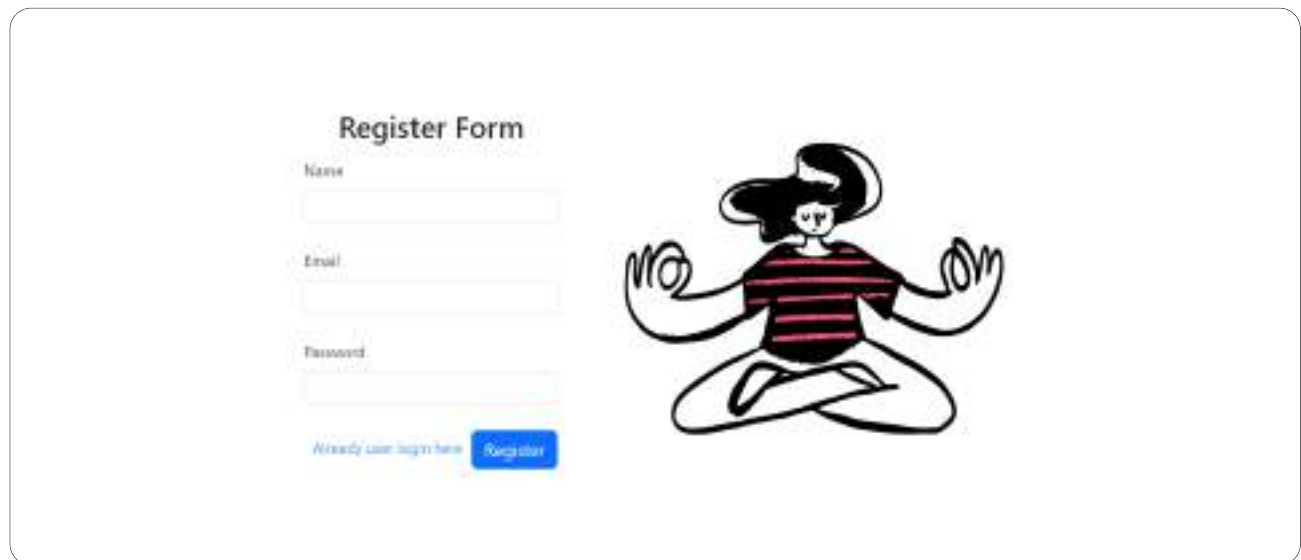
## **5. Waiting List Management**

Doctor can maintain a waiting list for appointments in case of unavailability. Patients will be notified when slots become available.

## On Boarding Screens

### Register Form Page

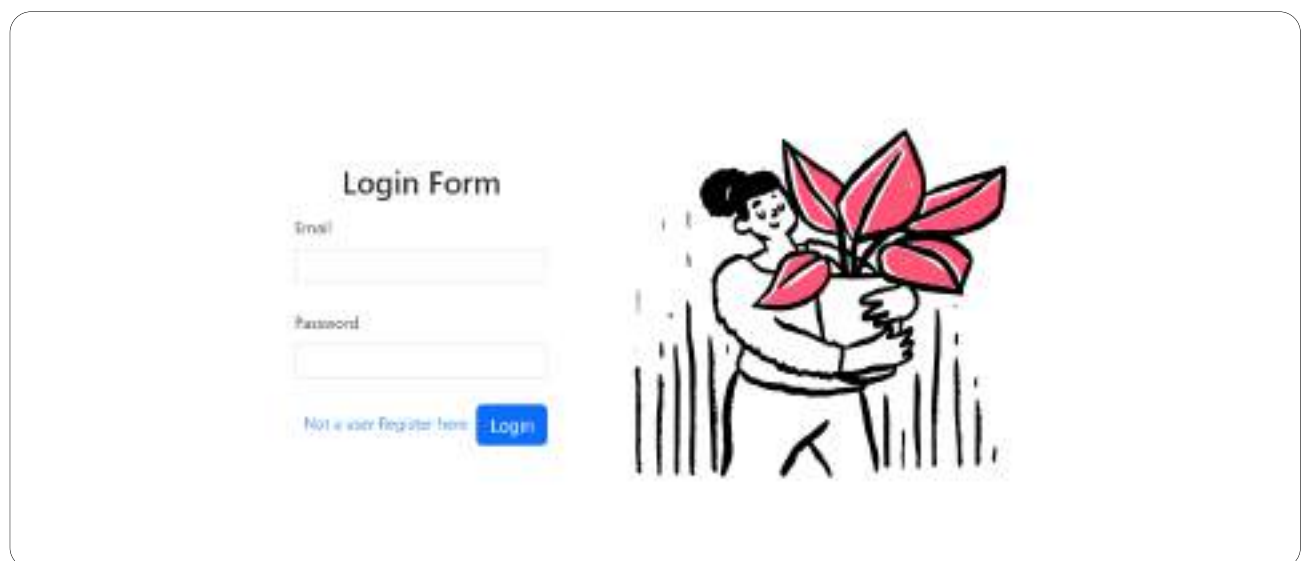
If the user is new, the user needs to register by filling Name, Email and Password



The image shows a UI mockup for a registration form. On the left, the form is titled "Register Form" and contains three input fields labeled "Name", "Email", and "Password". Below the "Password" field, there is a link that says "Already user login here" and a blue button labeled "Register". To the right of the form is a cartoon illustration of a person with dark hair, wearing a red and black striped shirt, sitting in a meditative lotus position with their hands in a mudra.

### Log-In Form Page

If the user has previously created the account, the user needs to login by filling Email and Password



The image shows a UI mockup for a login form. On the left, the form is titled "Login Form" and contains two input fields labeled "Email" and "Password". Below the "Password" field, there is a link that says "Not a user Register here" and a blue button labeled "Login". To the right of the form is a cartoon illustration of a person with dark hair, wearing a white shirt, holding a large potted plant with large red leaves.

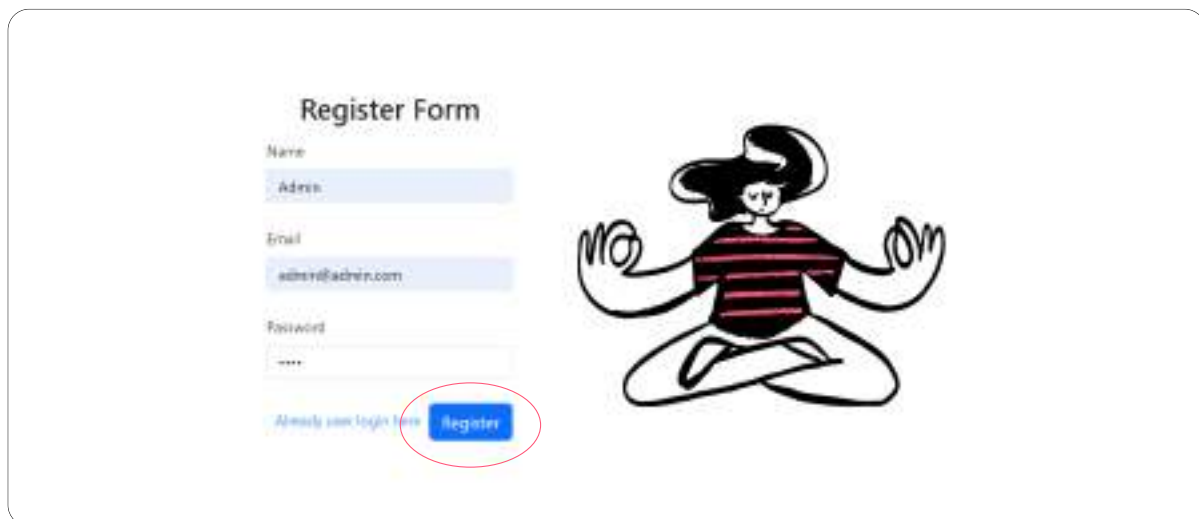
## Create Admin Account

The “Admin” is responsible for approving the doctor request. If any user has applied for the “Apply Doctor” by submitting valid proofs, the admin can verify and approve or reject the application.

### Register Form Page

For Instance,

Name = Admin, Email = admin@admin.com, Password = 12345



The image shows a web form titled "Register Form". It contains input fields for Name, Address, Email, and Password. The Email field is pre-filled with "admin@admin.com". Below the Password field, there is a link "Already have login here" and a blue "Register" button, which is circled in red. To the right of the form is a cartoon illustration of a person with multiple arms in a meditative pose.

Now after filling the form, the user will click “Register”

### MongoDB Compass

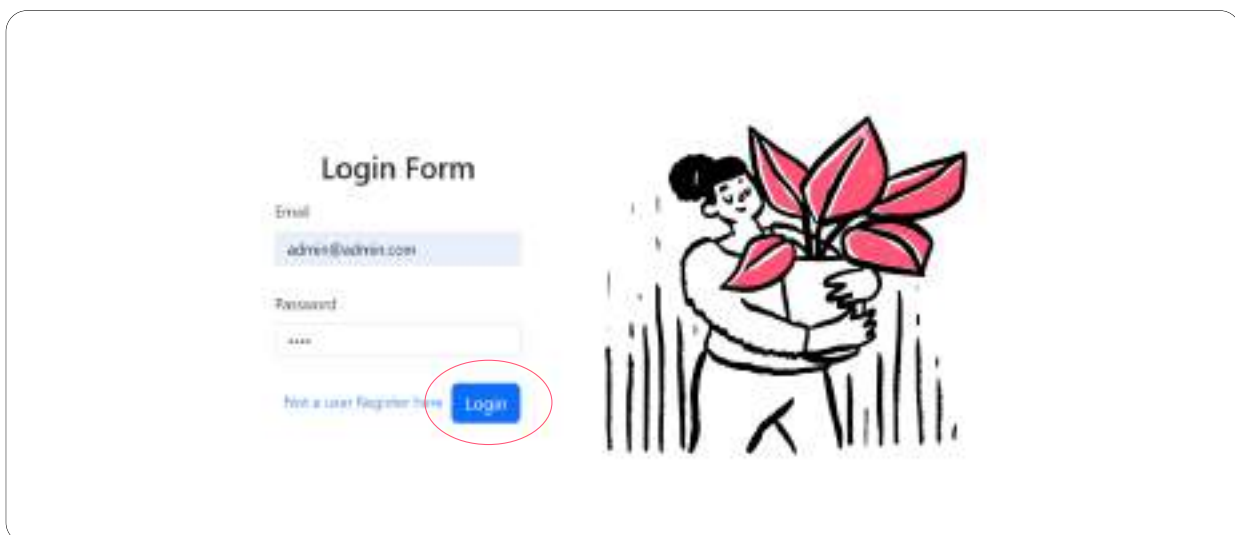


To make user, admin, isAdmin value will be changed to “true”

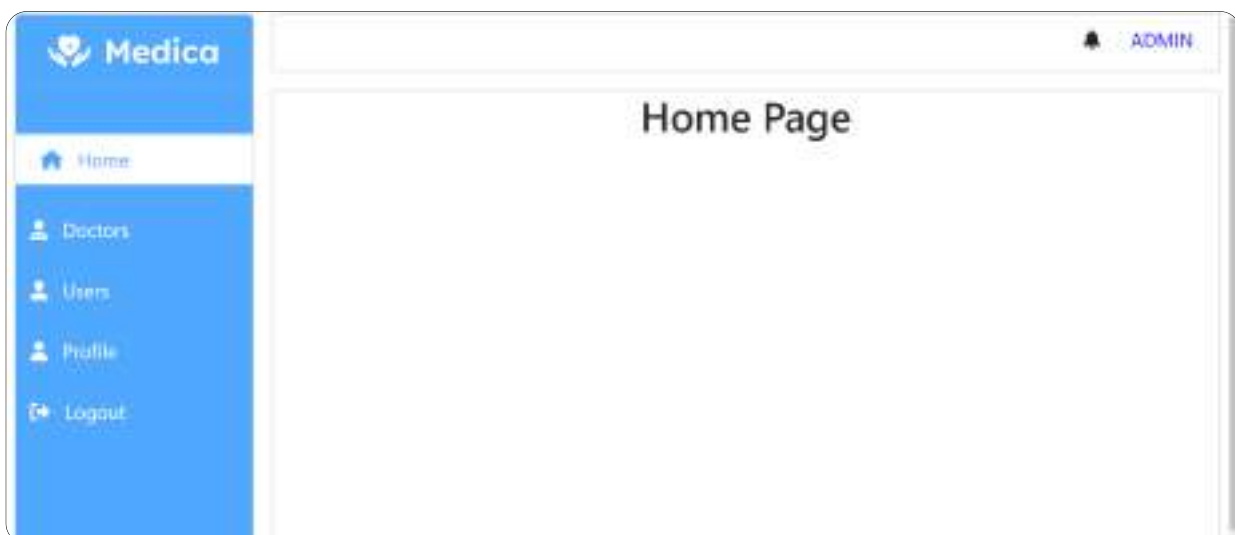


Now, the user is "Admin".

## Login Form Page



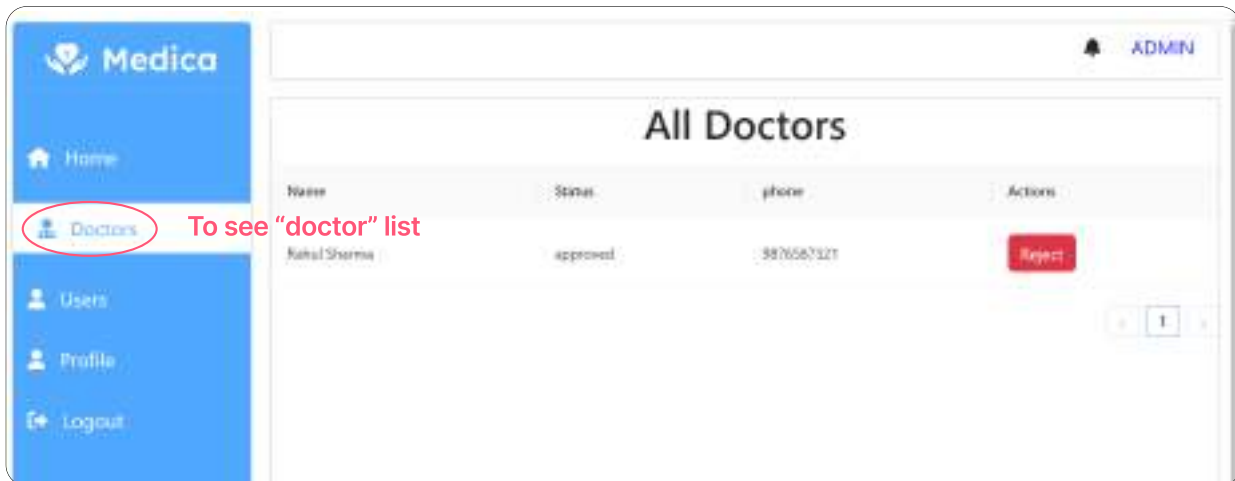
Now after filling the form, the user will click “Login” button



Admin account is successfully created.

## Home Page

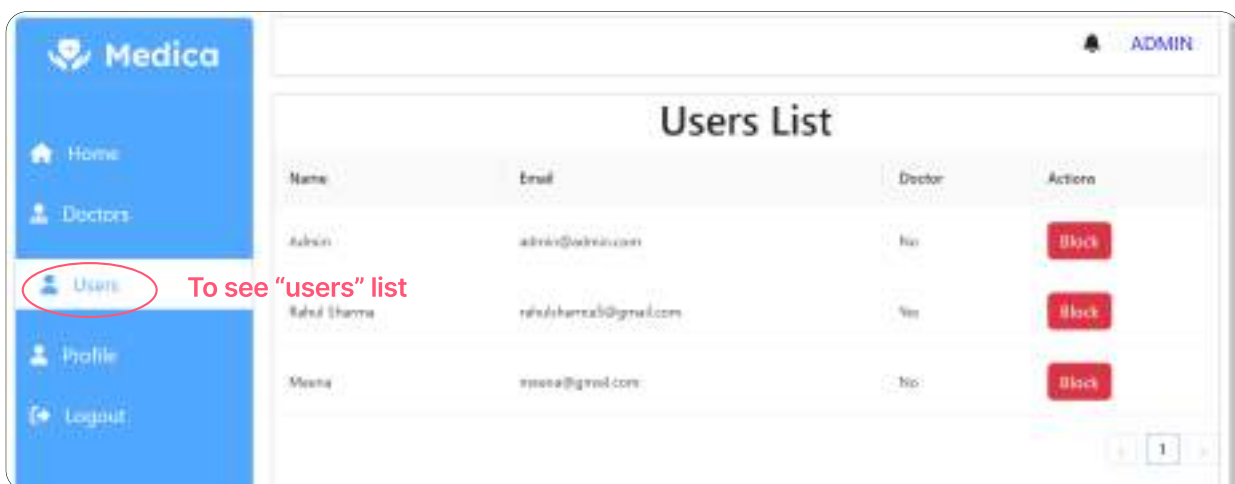
We will now direct to the Home page.



The screenshot shows the Medica Admin Dashboard. The sidebar on the left has a blue background with white text and icons. The 'Doctors' menu item is highlighted with a red circle. The main content area is titled 'All Doctors' and contains a table with the following data:

Name	Status	phone	Actions
Rahul Sharma	approved	9976567121	<a href="#">Reject</a>

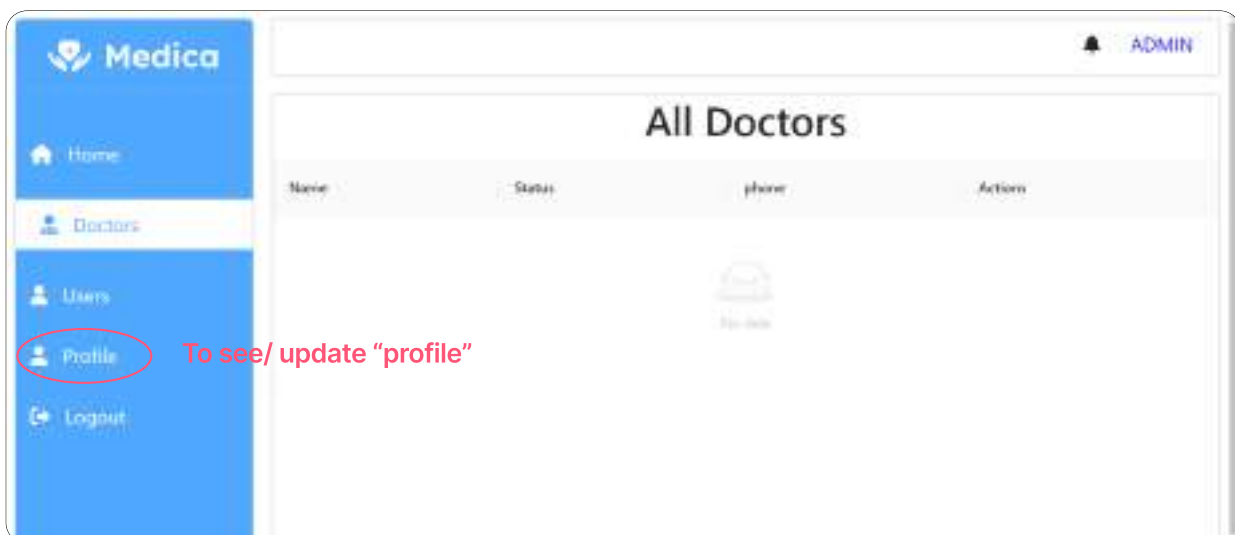
At the bottom right of the table, there is a pagination control showing '1'.




The screenshot shows the Medica Admin Dashboard. The sidebar on the left has a blue background with white text and icons. The 'Users' menu item is highlighted with a red circle. The main content area is titled 'Users List' and contains a table with the following data:

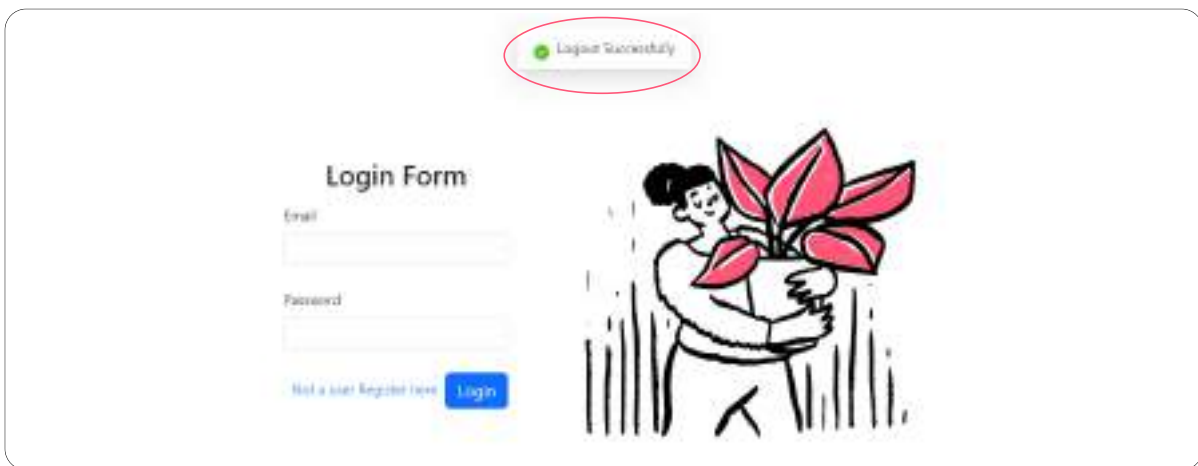
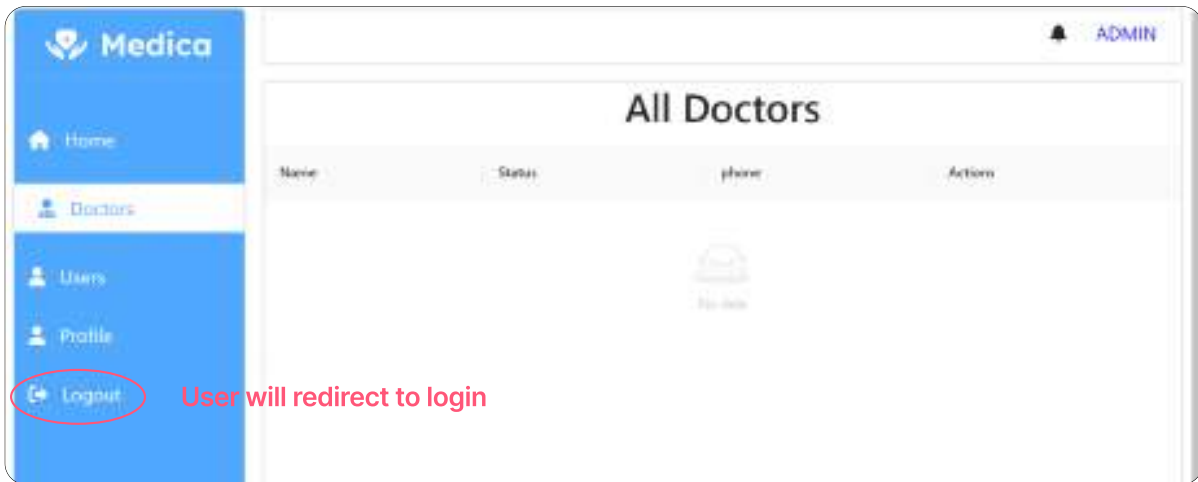
Name	Email	Doctor	Actions
Admin	admin@admin.com	No	<a href="#">Block</a>
Rahul Sharma	rahulsharma@gmail.com	No	<a href="#">Block</a>
Meena	meena@gmail.com	No	<a href="#">Block</a>

At the bottom right of the table, there is a pagination control showing '1'.

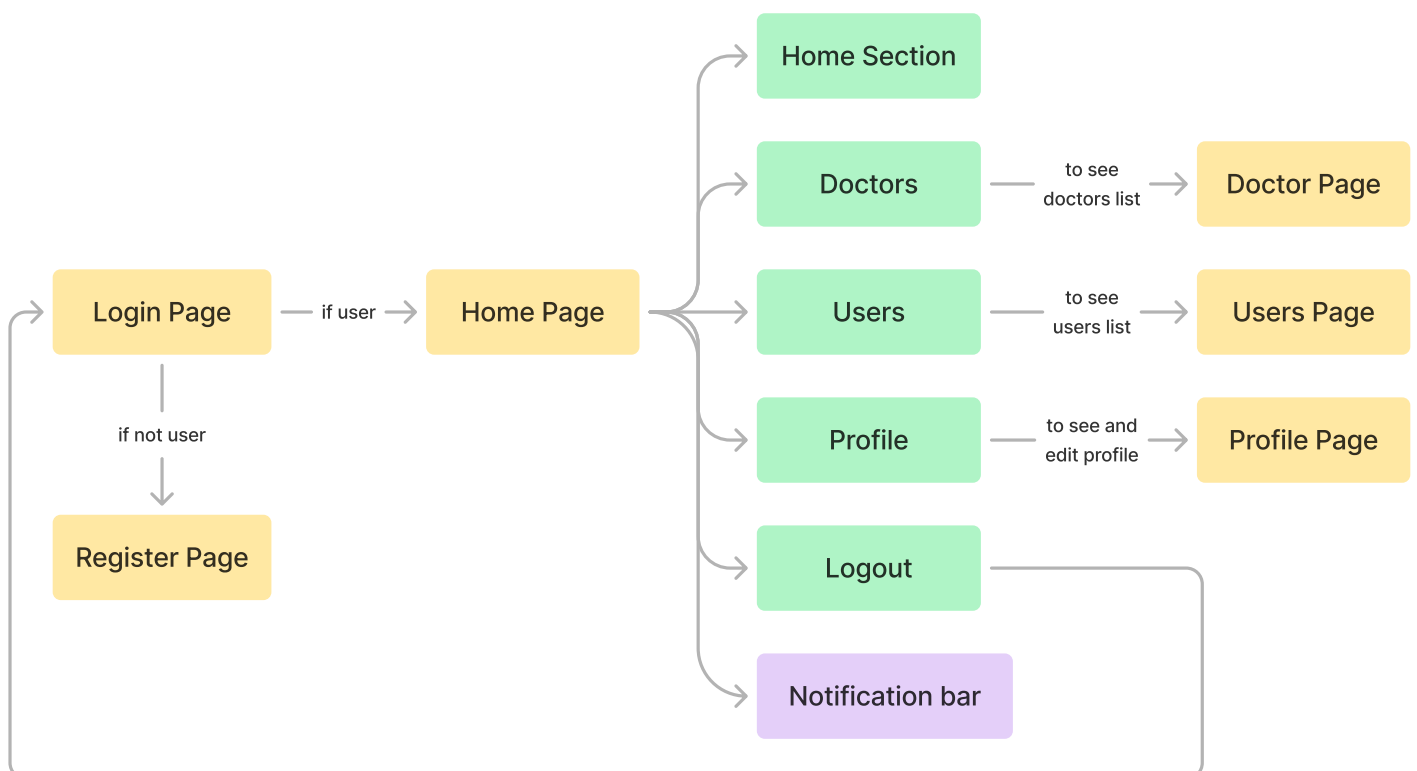


The screenshot shows the Medica Admin Dashboard. The sidebar on the left has a blue background with white text and icons. The 'Profile' menu item is highlighted with a red circle. The main content area is titled 'All Doctors' and contains a table with the following data:

Name	Status	phone	Actions
			



## Flow Chart





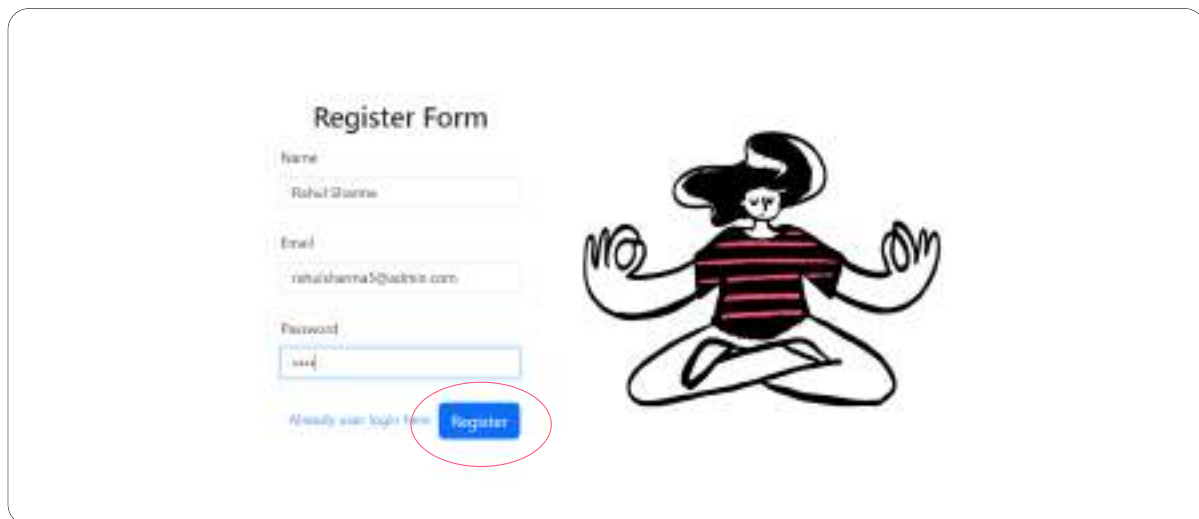
## Create Doctor Account

The “Doctor” first needs to create an account and apply for ‘Doctors’. Then, the admin will receive a notification and can either approve the request or reject it. If admin approves it doctor can receive the notifications for appointment and will update the user if he is available or not.

### Register Form Page

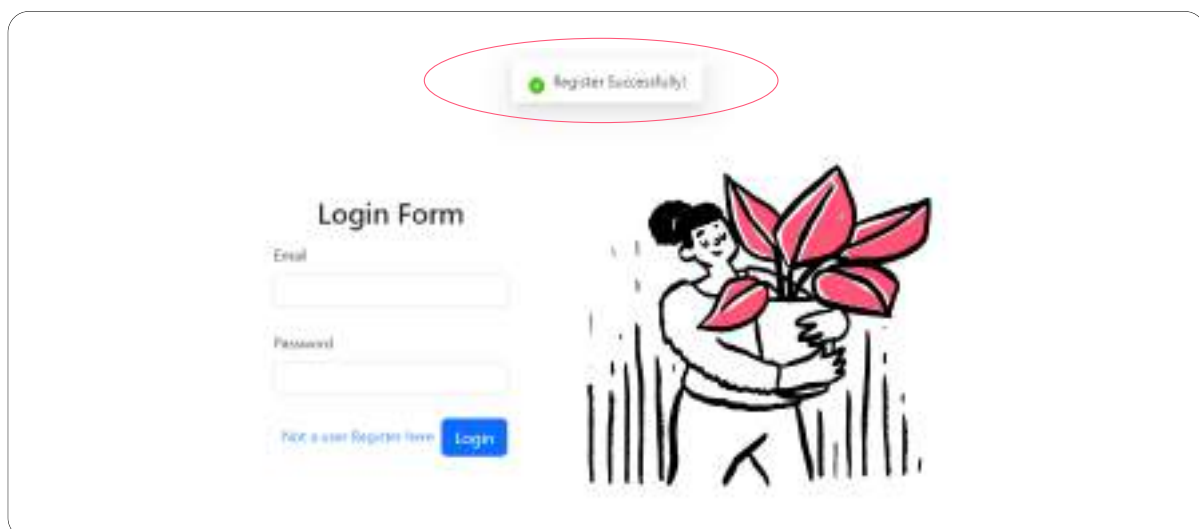
For Instance,

Name = Rahul Sharma, Email = rahulsharma5@gamil.com, Password = !12a



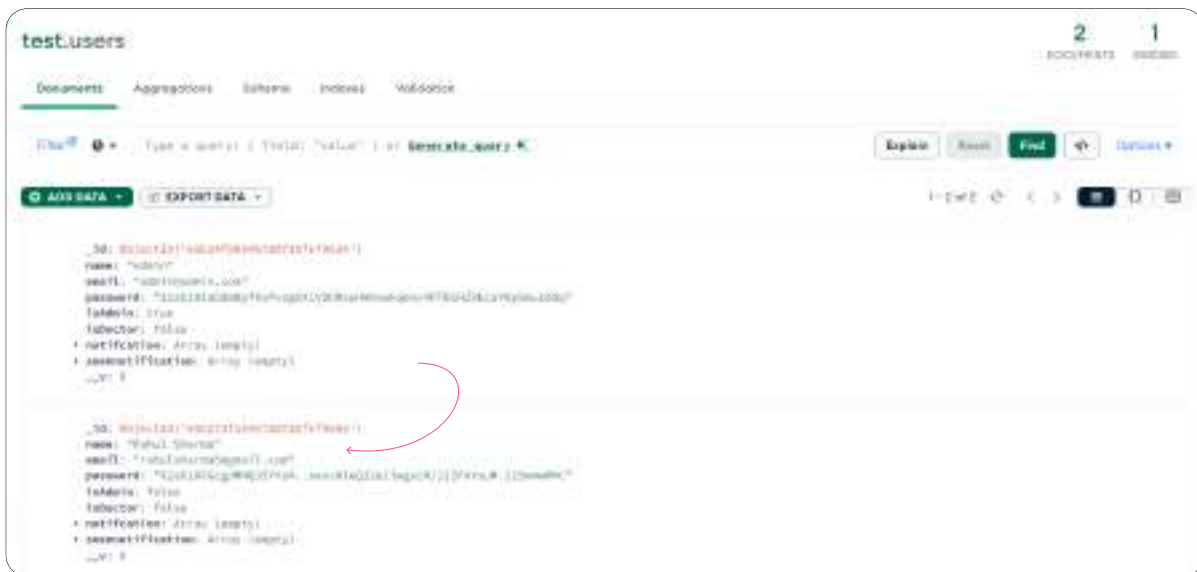
The image shows a web page titled "Register Form". It contains three input fields: "Name" with the value "Rahul Sharma", "Email" with the value "rahulsharma5@gamil.com", and "Password" with the value "!12a". Below the password field is a link that says "Already have login form?". To the right of the form is a cartoon illustration of a person with multiple arms in a yoga pose. A red circle highlights the "Register" button at the bottom of the form.

Now after filling the form, the user will click “Register”

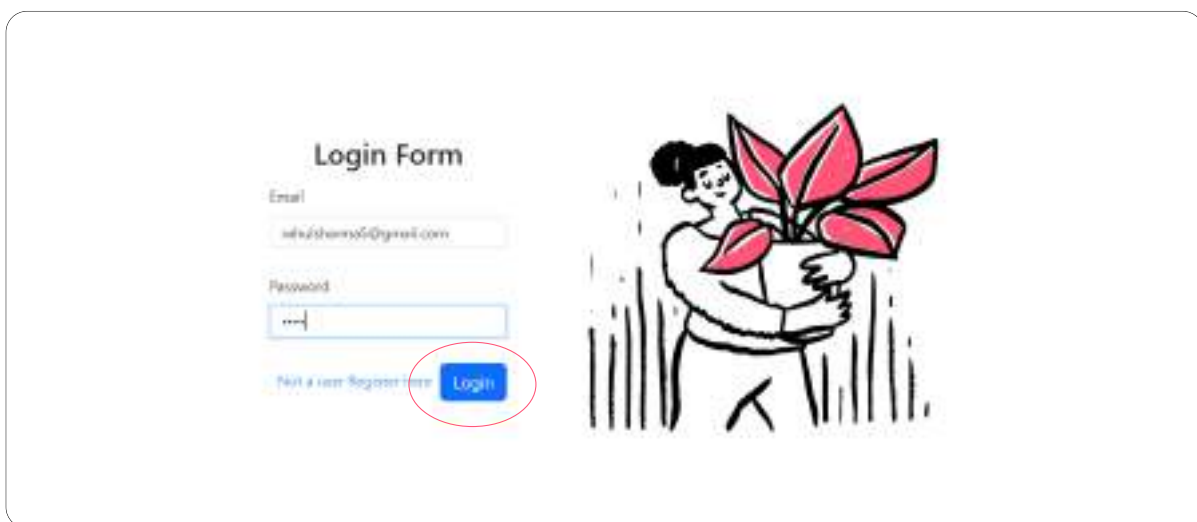


The image shows a web page titled "Login Form". It contains two input fields: "Email" and "Password". Below the password field is a link that says "Not a user Register here". To the right of the form is a cartoon illustration of a person holding a large potted plant. A red circle highlights a message box at the top of the page that says "Register Successfully!".

## MongoDB Compass



## Login Form Page



Now after filling the form, the user will click “Login” button



## Home Page



User account is successfully created. Now user will apply as 'Doctor'. First the user will go to "Apply Doctor" section.

The screenshot shows the 'Apply Doctor' form in the Medica application. The sidebar is the same as the Home page, but the 'Apply Doctor' link is highlighted. The form is titled 'Apply Doctor' and is divided into two main sections: 'Personal Details' and 'Professional Details'. The 'Personal Details' section includes fields for First Name, Last Name, Phone No, Email, Address, and a Submit button. The 'Professional Details' section includes fields for Specialization, Experience, Fees Per Consultation, and a Submit button. The form is currently empty, with placeholder text in the input fields.

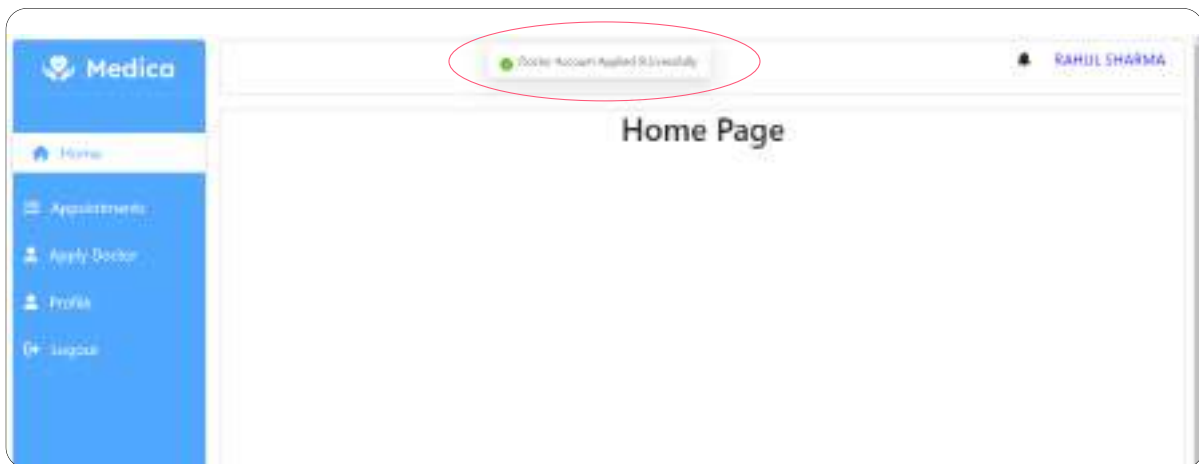
User will now fill the form details, and click "Submit" button.

The screenshot shows the 'Apply Doctor' form with the following details filled in:

Personal Details :		
First Name	Last Name	Phone No
Rahul	Sharma	9876543210
Email	Address	Pincode
rahulsharma@gmail.com	123, 456 Street	123456

Professional Details :		
Specialization	Experience	Fees Per Consultation
MBBS, MD	5	200
Timings	Submit	
0900 - 1200		

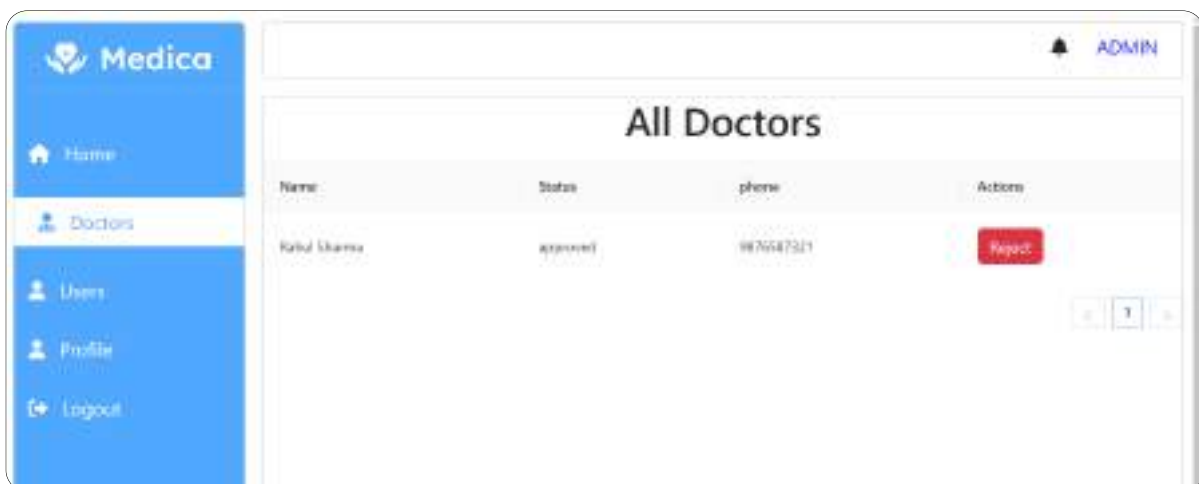


## Admin Account

Now admin will receive the update if user has applied for the 'doctor'. The admin will verify the details and will either approve or keep it in the waiting list.

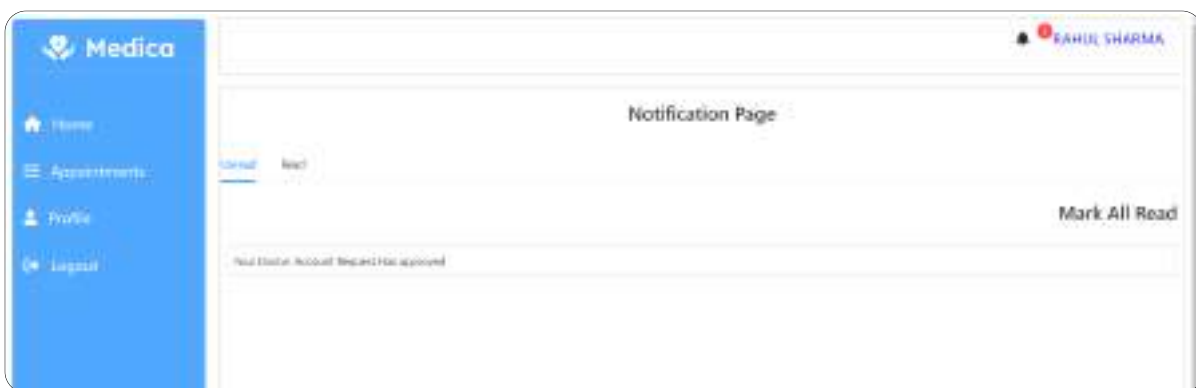


After approving the screen will be as follows-

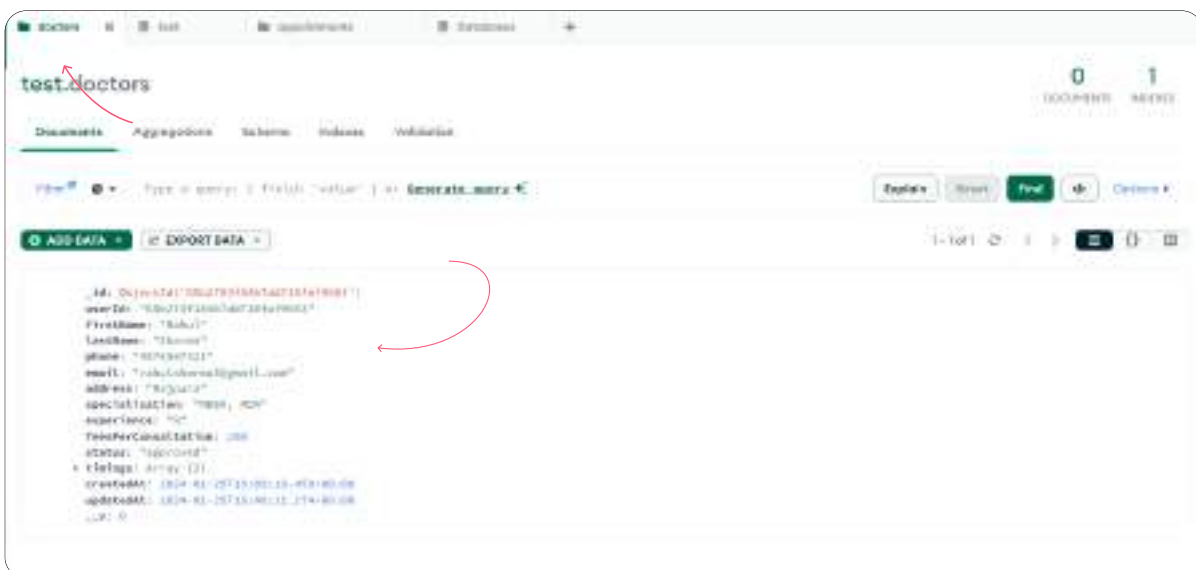


## User Account

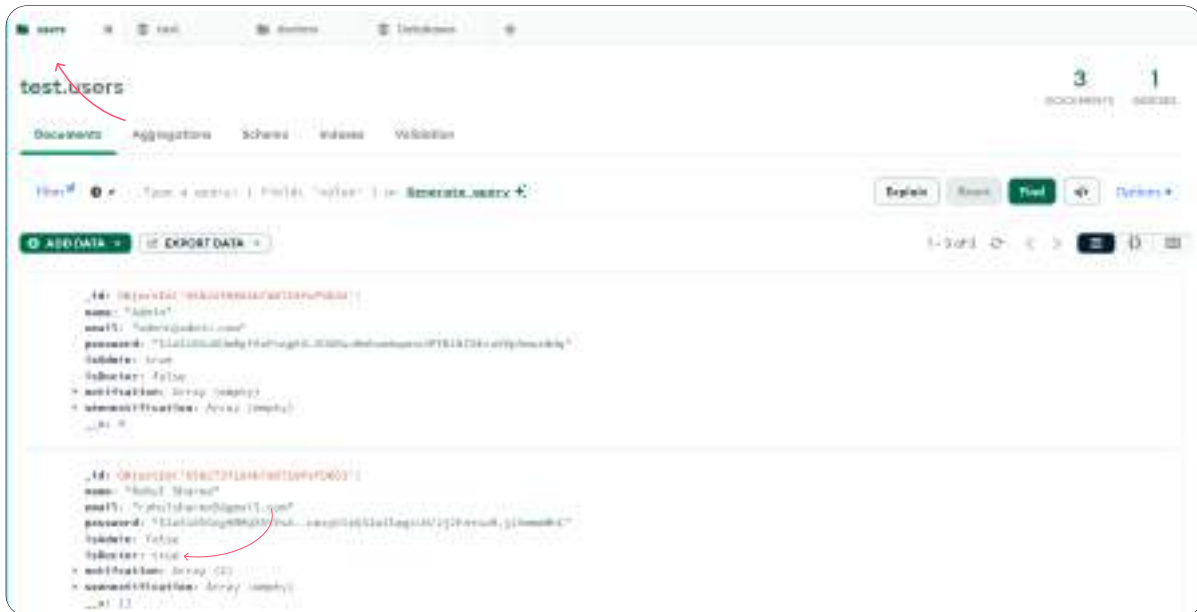
User receives the notification and in Home Page section, user can see his doctor section. Also, the “Apply Doctor” section will be dismissed.



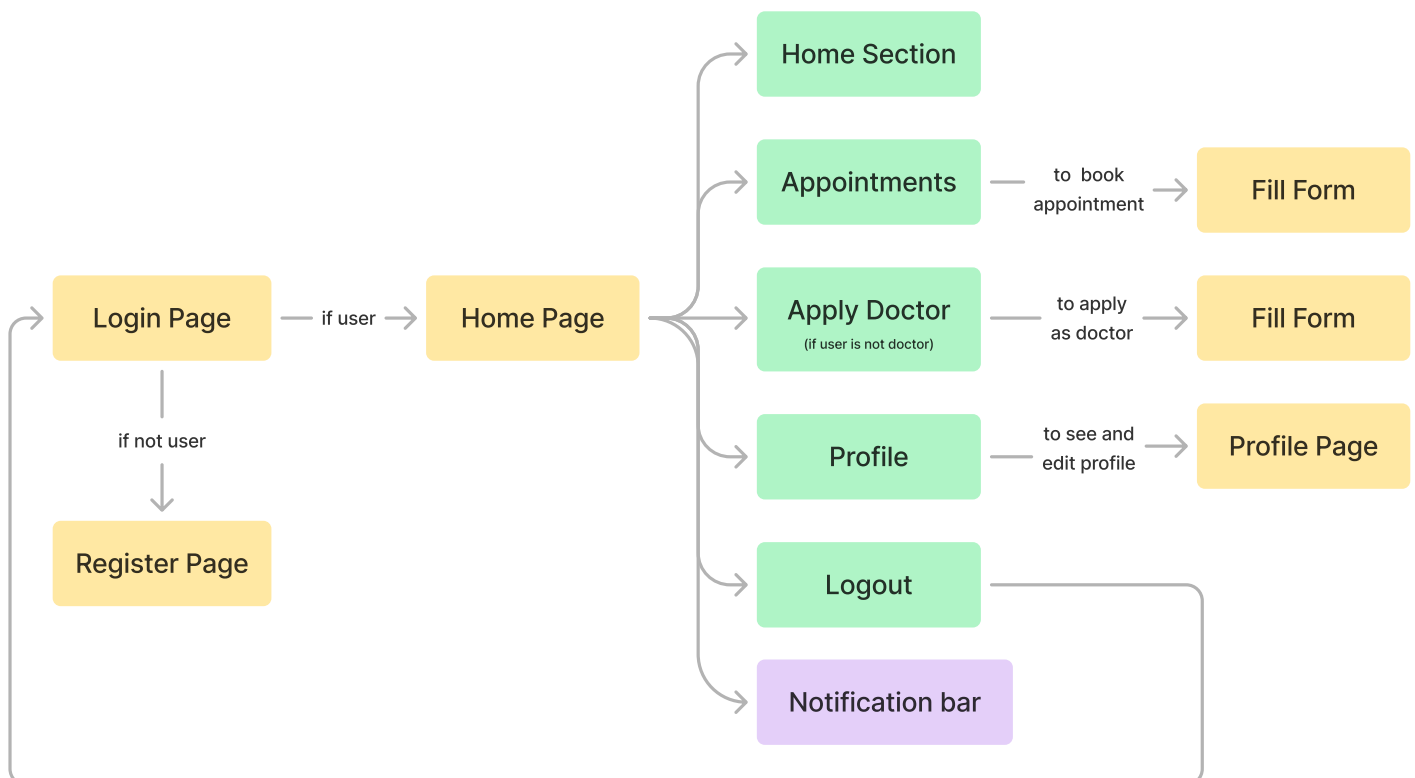
## MongoDB Compass



## MongoDB Compass



## Flow Chart



## Updating Profile

If doctor wants to change any information, doctor can go to the profile section and update required the details.

**Medica**

Home Appointments Profile Logout

**Manage Profile**

**Personal Details :**

First Name: Rahul Last Name: Sharma Phone No: 9876543210

Email: rahulsharma@gmail.com Address: 123 Main St, New York, NY 10001

**Professional Details :**

Specialization: MD, MS Experience: 5 Fees Per Consultation: 200

Save Update

**Medica**

Home Appointments Apply Doctor Profile Logout

**Apply Doctor**

**Personal Details :**

First Name: Rahul Last Name: Sharma Phone No: 9876543210

Email: rahulsharma@gmail.com Address: 123 Main St, New York, NY 10001

**Professional Details :**

Specialization: MD, MS Experience: 5 Fees Per Consultation: 200

Save Submit

Click "Submit" button.

**Medica**

Home Appointments Profile Logout

**Home Page**

Doctor Profile Updated

Dr. Rahul Sharma

Specialization: MD, MS

Experience: 5

Fees Per Consultation: 200

Timings: 09:00 - 20:00

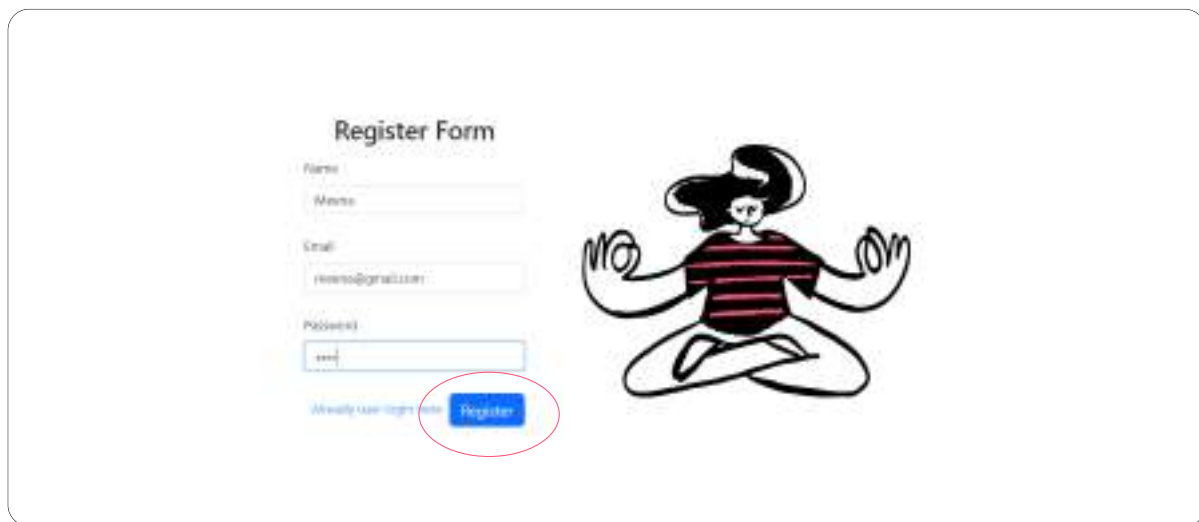
## Create User Account

The “User” first needs to create an account and is able to check the doctor availability and can book the appointment. The doctor will get update about the appointment will update the user if he is available or not.

### Register Form Page

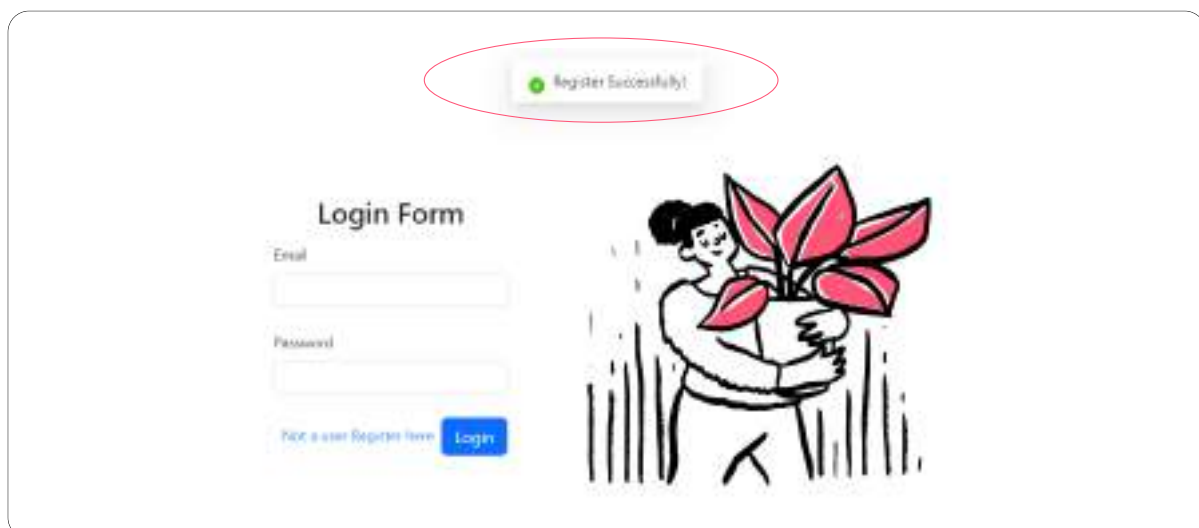
For Instance,

Name = Meena, Email = meena@gamil.com, Password = !ab3



The image shows a web form titled "Register Form". It contains four input fields: "Name" (with "Meena" entered), "Email" (with "meena@gamil.com" entered), "Password" (with "!ab3" entered), and a "Confirm Password" field (which is empty). Below the fields are two links: "Already user login here" and a blue "Register" button. The "Register" button is circled in red. To the right of the form is a cartoon illustration of a person with long dark hair, wearing a red and black striped shirt, sitting in a meditative pose with hands in a mudra.

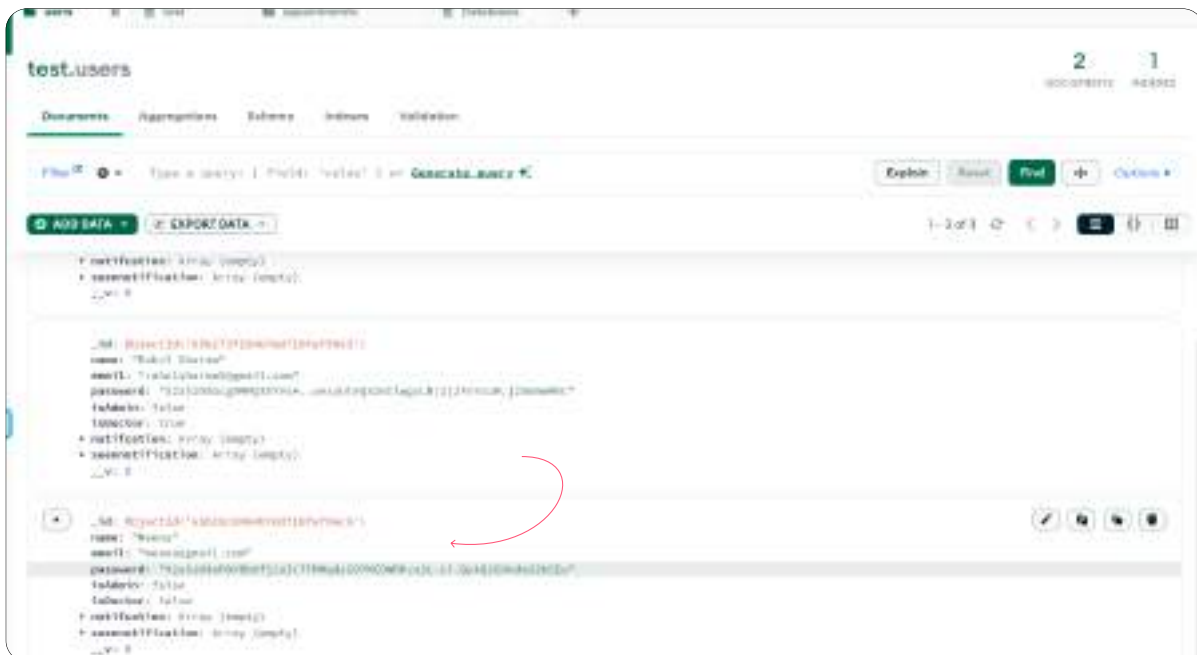
Now after filling the form, the user will click “Register”



The image shows a web form titled "Login Form". It contains two input fields: "Email" and "Password". Below the fields are two links: "Not a user Register here" and a blue "Login" button. Above the form, a green success message "Register Successfully!" is displayed in a rounded rectangle, which is circled in red. To the right of the form is a cartoon illustration of a person with short dark hair, wearing a white shirt, holding a large potted plant with red leaves.



## MongoDB Compass



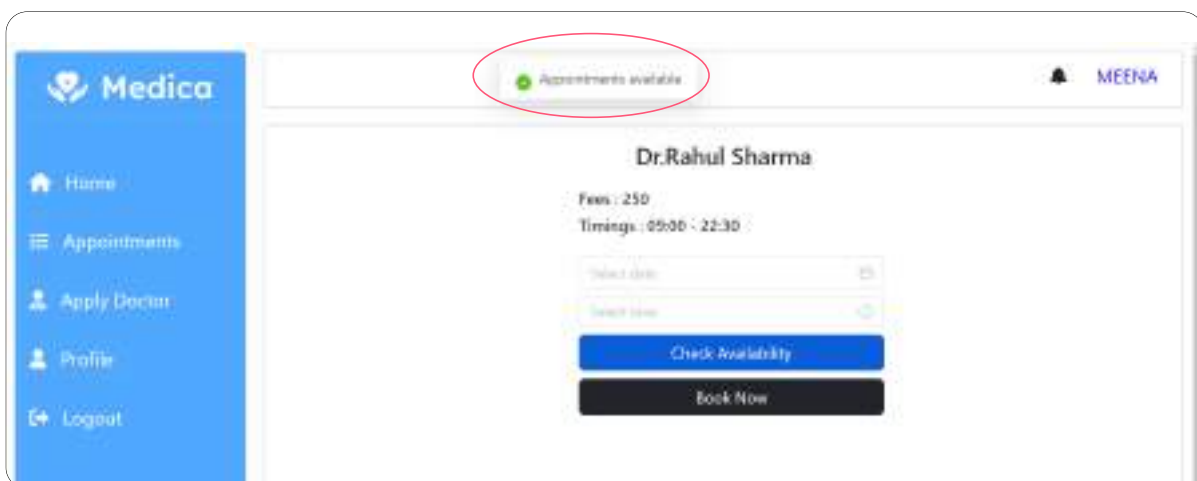
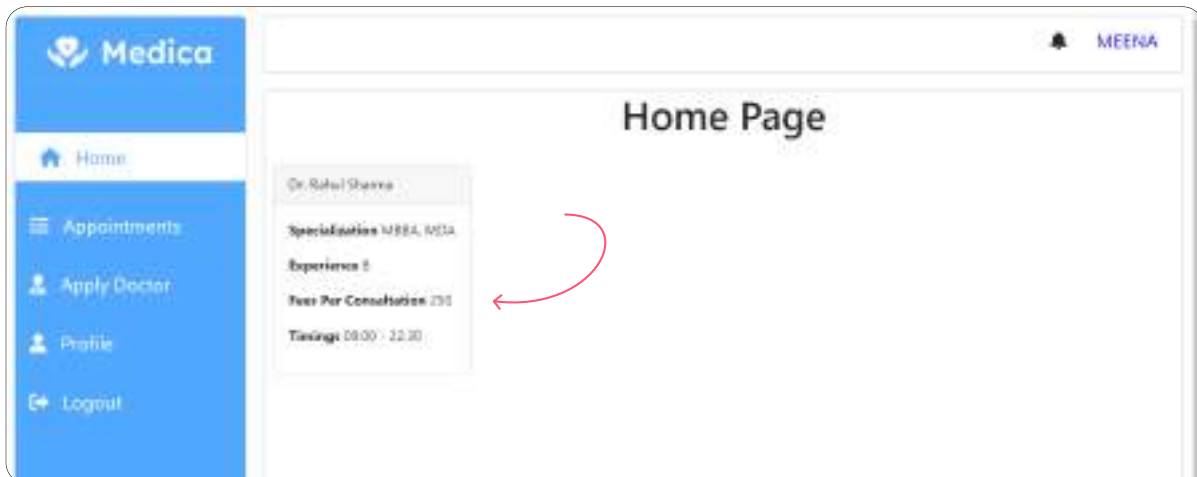
## Login Form Page



Now after filling the form, the user will click “Login” button

## Home Page

In home section user can see the doctors list & can decide to which doctor the user want to book the appointment. Here, user will select card of “Dr Rahul Sharma”



Since availability is available, user will now book the appointment

The screenshot shows the Medica app interface. On the left is a blue sidebar with navigation options: Home, Appointments, Apply Doctor, Profile, and Logout. The main content area displays the doctor's name 'Dr. Rahul Sharma', fees of 250, and timings from 09:00 to 22:30. Below this, there are input fields for 'Select date' (showing 17-01-2024) and 'Select time' (showing 14:00). At the bottom are two buttons: 'Check Availability' (blue) and 'Book Now' (black). A red arrow points to the 'Book Now' button.

This screenshot shows the same Medica app interface as the previous one, but with a green notification bubble at the top that says 'Appointment Book successfully'. The 'Book Now' button is now highlighted with a yellow border, indicating it has been clicked.

In appointment section,

The screenshot shows the 'Appoinmntnets Lists' section of the Medica app. The sidebar is the same as before. The main content area has a title 'Appoinmntnets Lists' and a table with three columns: ID, Date & Time, and Status. The table contains one row with the following data:

ID	Date & Time	Status
659295eb8467dd7164f972b	17-01-2024 14:00	pending

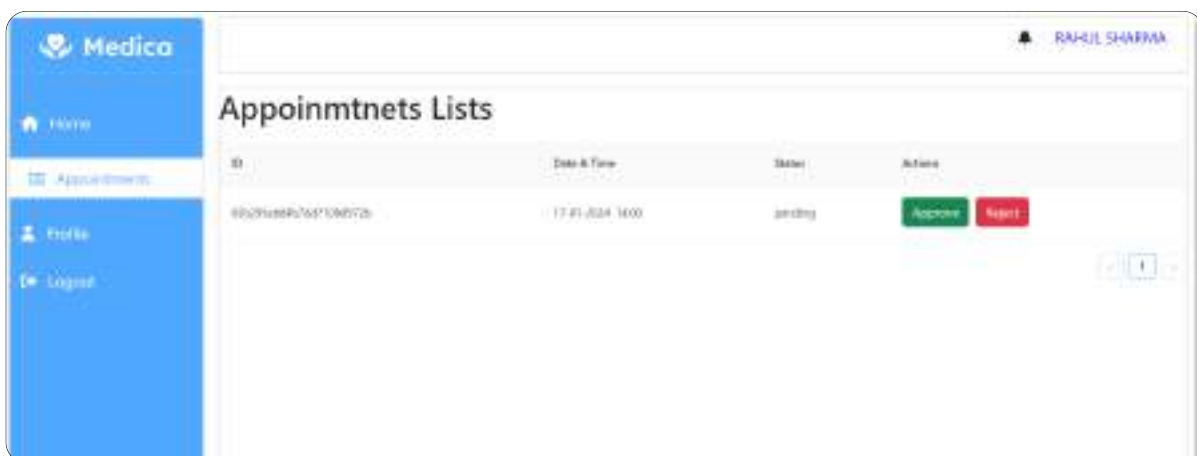
The 'pending' status is circled in red. At the bottom right of the table, there are pagination controls showing '1' of 1 items.

## MongoDB Compass

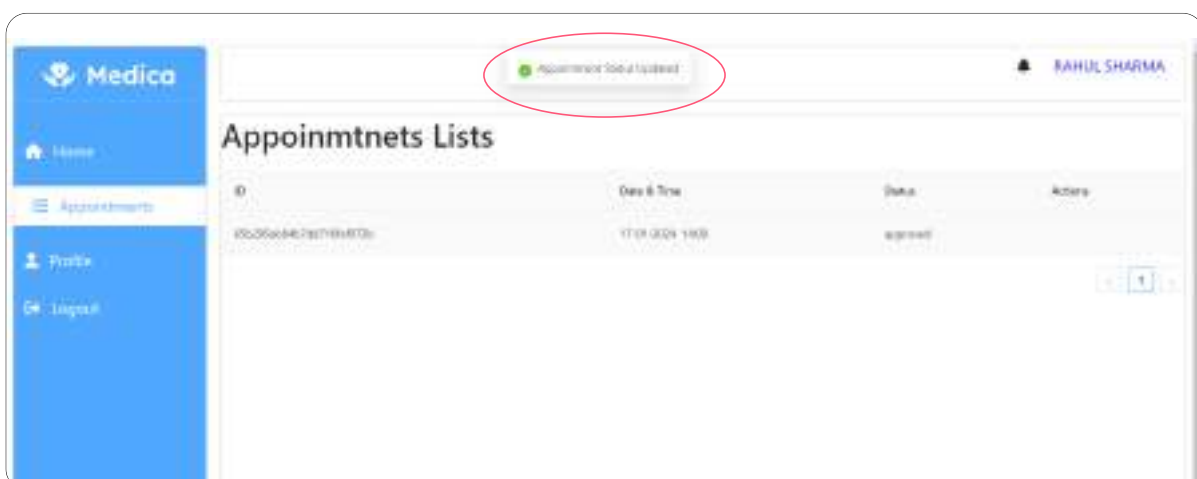


## Doctor Account

In appointment section, doctor is able to see the appointment, if doctor is available he will click “Approve”, if not then “Reject”

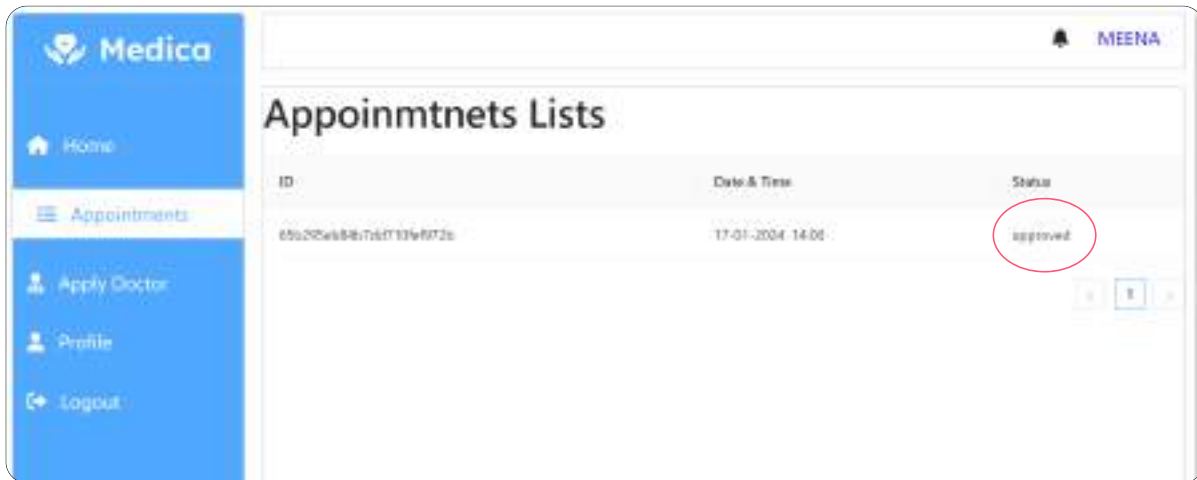


Clicking “Approve” we can see,

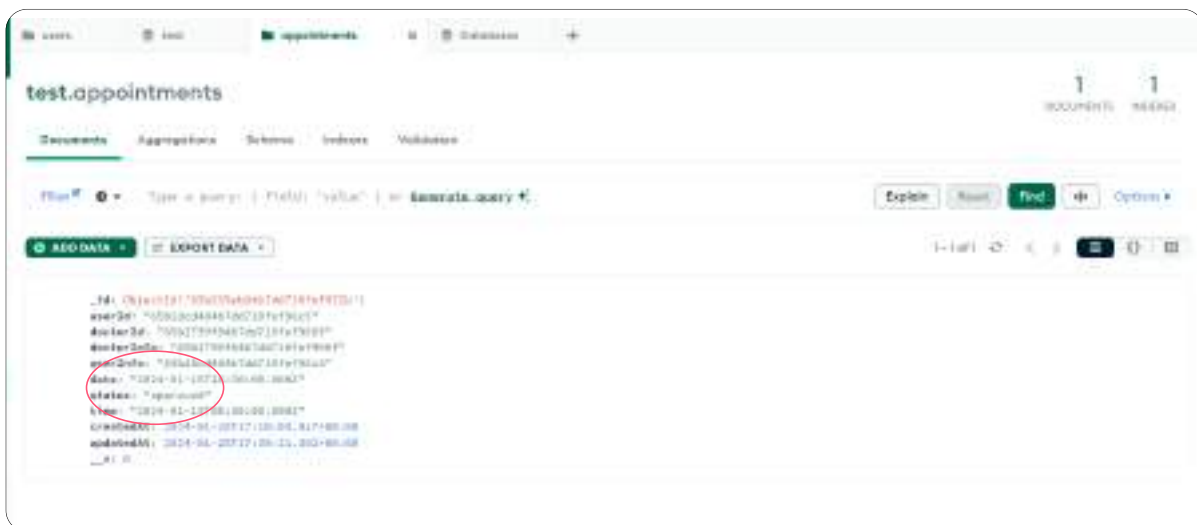


## Home Page

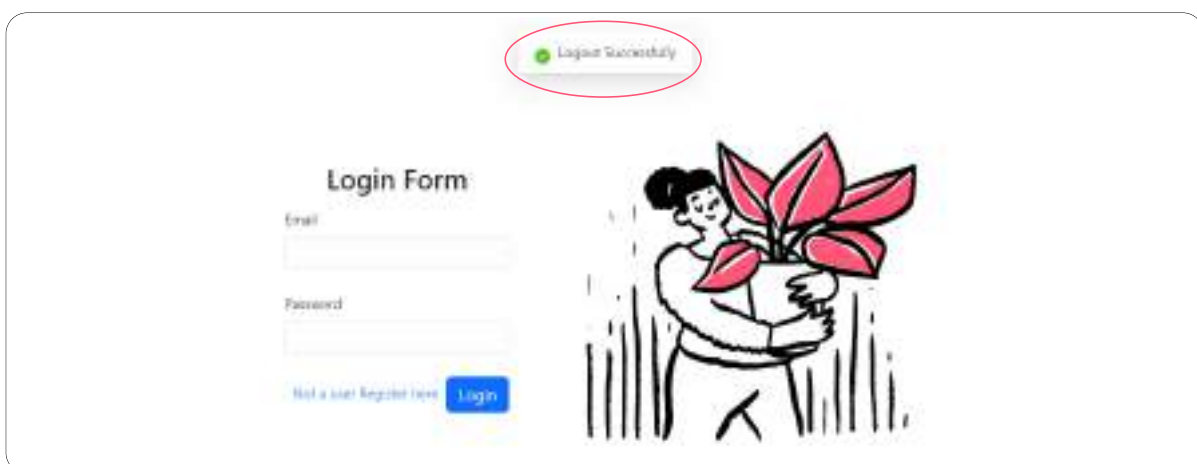
The user can also see the approved appointment.



## MongoDB Compass



## Logout



# Future Advancement

## 1. Doctor & Patient Communication

The live communication between Doctor and Patient can save a lot of time. The means of communication could be chat stream, voice or video call.

## 2. Medical Record Management

Patient to upload and manage their medical records within the system for healthcare professionals to access with their consent so that doctor can rectify the disease easily. This can also save time if by mistake patients forget to bring their records to the clinic

## 3. Reporting & Analysing

Generating reports on appointment data, user activity, and system performance for insights and optimization can be really beneficial for both doctors and patients as they can see the live track.

## 4. Multi-language system

Language can't be a barrier to anything, anyone can seek medical help

## 5. Feedback and Rating system

Feedback and Rating from patients can really help doctors, it can increase the trust for other users by seeing the rating and feedback that is provided by other in real.

**Thank You!**