Full Stack Development with MERN
Project Documentation

1. Introduction

Project Title: Online Complaint Registration System

Team Members - Akash M
Ahamed Abdul kadhar J
Pozhilan R
Ponmoorthi M

2. Project Overview

Purpose:
The Online Complaint Registration System enables users to lodge complaints online, track their status, and ensure timely resolution. It simplifies the traditional complaint process by digitizing submissions and updates.

Features:

- User Registration and Login.
- File complaints with supporting details.
- Admin dashboard for complaint management.
- Status tracking for users.
-

3. Architecture

Frontend:

Built using React, the frontend offers an intuitive interface for users and admins, utilizing React Router for navigation and Context API for state management.

Backend:
Powered by Node.js and Express.js, the backend manages API requests, data validation, and business logic.

Database:

MongoDB stores user details, complaint information, and logs, structured with collections for users, complaints, and admin.

4. Setup Instructions

Prerequisites:

- Node.js (v14 or higher)
- MongoDB (local or cloud instance)
- Git

Installation

I. **Clone the repository:**

```
git clone https://github.com/Akashmurugan123/Online-complaint-registration.git
```

II. **Navigate to the project directory:**cd Online-complaint-registration

III. **Install dependencies for both client and server:**

```
cd client && npm install
cd ../server && npm install
```

Set up environment variables in .env files in the root of both directories.

5. Folder Structure

Client(frontend)

/src: Contains React components, pages, and assets.
/src/context: Holds state management logic.

Server:

/routes: Contains route handlers for API endpoints.
/models: Defines MongoDB schemas.

6. Running the Application

Frontend:

code-cd frontend
npm start

Backend:

After Node installation
code- npm start or
node index.js

7. API Documentation

Endpoint: /api/auth/login
Method: POST
Description: User login.

Request:
Json format -
{"Name":"MyName", "email": "user@example.com", "password": "password123","Mobile No": "0123456789" }

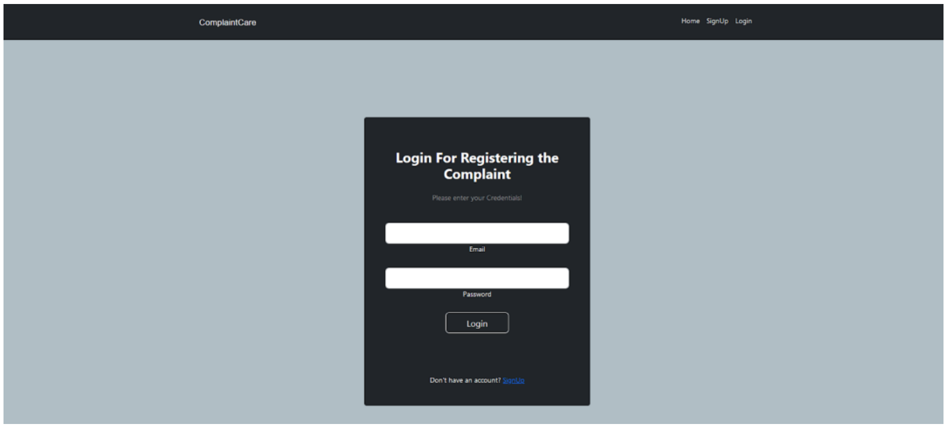
Response:
json code-
{"token": "JWT_TOKEN", "user": { "id": "123", "name": "John Doe" ,"email": "john@example.com", "password": "Hello@123","Mobile No": "9876543210" } }

8. Authentication

- Uses **JWT tokens** for secure authentication.

Tokens are generated upon login and stored in local storage for session management.

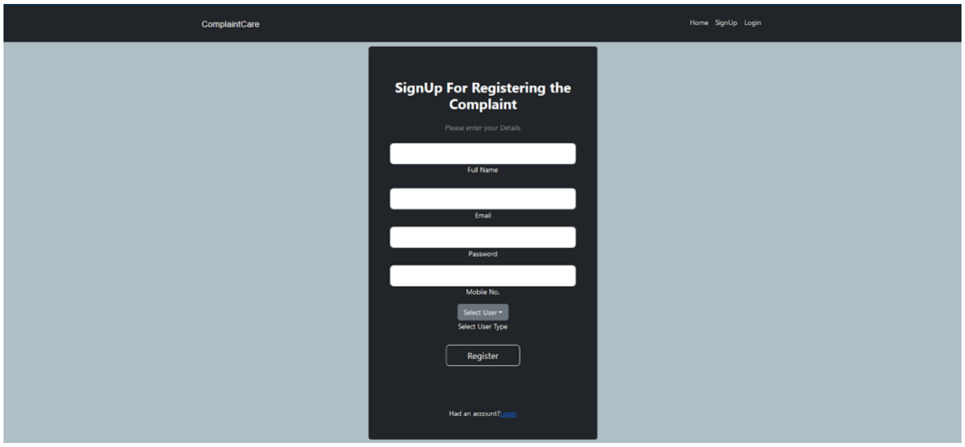
9. User Interface



Login For Registering the Complaint

Please enter your Credentials!

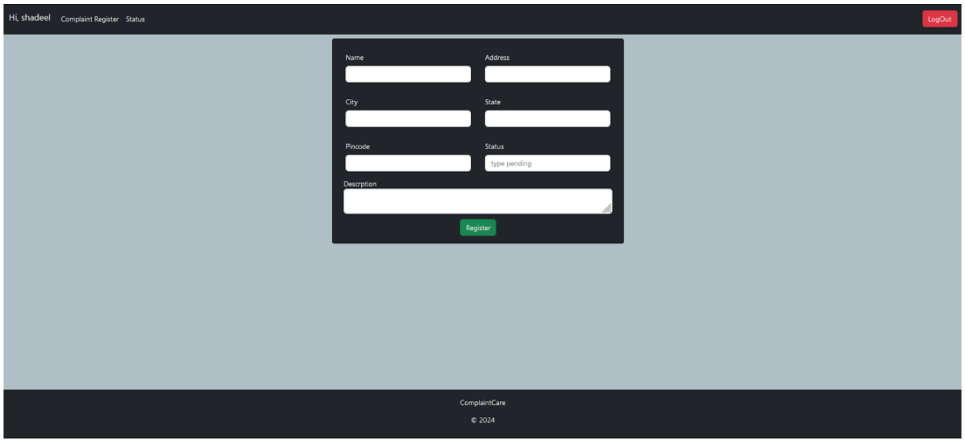
Don't have an account? SignUp



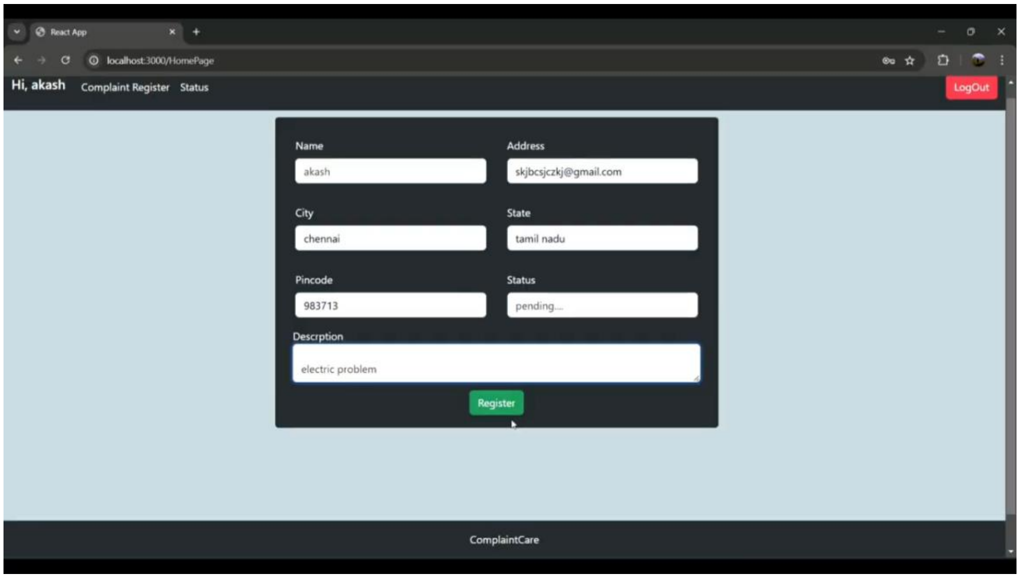
SignUp For Registering the Complaint

Please enter your Details

Had an account?Login



Hi, shadeel



Name: akash

Address: skjbcsjczkj@gmail.com

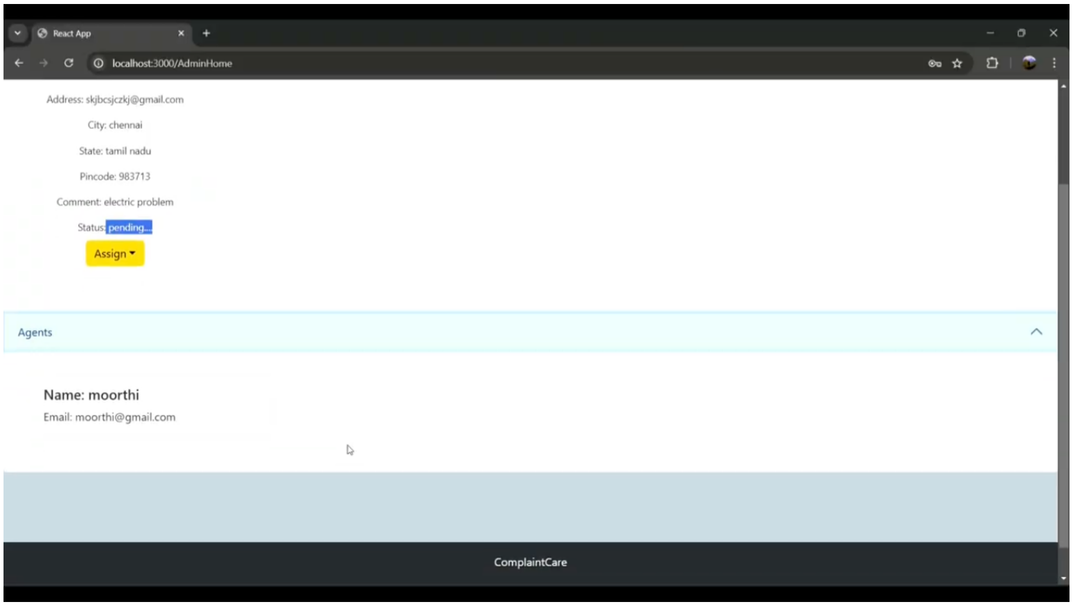
City: chennai

State: tamil nadu

Pincode: 983713

Status: pending...

Descrption: electric problem



Address: skjbcsjczkj@gmail.com

City: chennai

State: tamil nadu

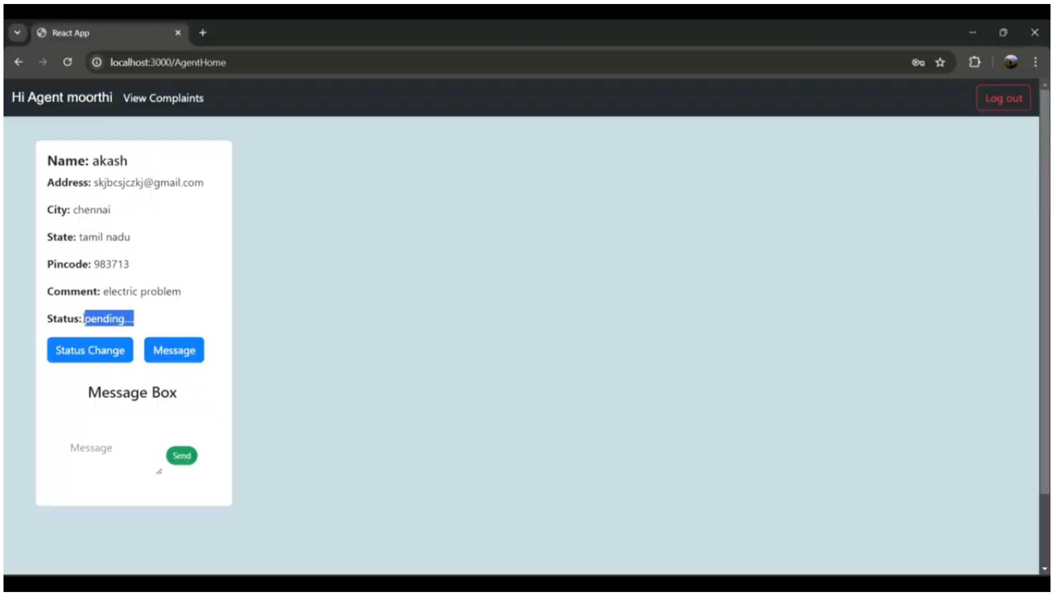
Pincode: 983713

Comment: electric problem

Status: pending...

Name: moorthi

Email: moorthi@gmail.com



Name: akash

Address: skjbcsjczkj@gmail.com

City: chennai

State: tamil nadu

Pincode: 983713

Comment: electric problem

Status: pending...

10. Testing

- **Tools Used:** Postman.

11. **Demo**

Demo(videolink-https://drive.google.com/file/d/14ES791txcsjzAh84SgLl5V5impsMYB-3/view?usp=drivesdk

12. Known Issues

- Database doesn’t allowed duplication in password and mobile number.

13. Future Enhancements

- ✦ Add real-time complaint notifications via WebSockets.
- ✦ Implement AI-based complaint categorization.
- ✦ Introduce multilingual support