**RESOLVE NOW**

**(YOUR PLATFORM FOR ONLINE COMPLAINTS)**

**ADITYA COLLEGE OF ENGINEERING AND TECHNOLOGY**

**SUBMITTED BY**

|  |  |
| --- | --- |
| PAMARTHI NAGA DURGA | 24P31F0064 |
|  |  |

**INDEX**

|  |  |  |
| --- | --- | --- |
| S.NO | CONTENT | PAGE.NO |
| 1 | INTRODUCTION | 1 |
| 2 | TECHNICAL ARCHITECTURE | 2-3 |
| 3 | FEATURES OF RESOLVENOW | 4-5 |
| 4 | TECHNOLOGY STCAK | 6 |
| 5 | PROJECT STRUCTURE | 7-8 |
| 6 | BACKEND | 8-9 |
| 7 | FRONTEND | 9-10 |
| 8 | PROJECT IMPLEMENTATION | 11-16 |
| 9 | SUMMARY | 17 |

**Introducing Resolve Now: Your Platform for Online Complaints**

Welcome to Resolve Now, your ultimate destination for addressing concerns and finding solutions! Our platform is designed to meet all your complaint resolution needs, whether you're seeking quick support, detailed assistance, or simply looking to voice your concerns. With an intuitive interface and a wide range of tools, it’s never been easier to submit, track, and resolve complaints. Explore helpful resources, connect with dedicated support teams, and manage your issues seamlessly from anywhere. Dive into Resolve Now, where every concern finds a solution, and every voice is heard!

**Resolve Now: Your Platform for Online Complaints**

Resolve Now was developed using the MERN Stack, a powerful blend of modern technologies (MongoDB, Express.js, React, and Node.js). This platform transforms the way users address complaints, combining the need for quick resolutions with the convenience of technology.

Designed with today’s users in mind, Resolve Now is robust, user-friendly, and tailored to meet your needs. Whether you're filing a new complaint or tracking an existing one, our platform delivers a seamless and personalized experience to suit your preferences.

MongoDB works behind the scenes to securely store and organize a vast database of complaints, ensuring quick access to your information. React provides a sleek, intuitive interface that looks great and performs flawlessly on any screen, while Express.js and Node.js keep the platform running smoothly and efficiently. Compatible with desktops, tablets, and phones, Resolve Now allows you to manage complaints anytime, anywhere. With its responsive, clean, and simple design, submitting, tracking, and resolving complaints is just a few clicks away**.**

**FEATURES OF BOOKNEST**

**User Registration and Authentication**

Securely register accounts, log in, and authenticate identities to access the complaint resolution platform.

**Query Listings**

Display a comprehensive list of user-submitted queries with details such as query type, category, description, status, and submission date.

**Query Selection**

Enable users to filter and select queries based on criteria like category, status, priority, or date submitted.

**Submission Process**

Allow users to submit queries, attach relevant files, and track progress. Upon submission, a query is generated, and the system updates its status accordingly.

**Query Confirmation**

Provide users with a confirmation page or notification containing query details, including reference ID, category, and estimated resolution time.

**Query History**

Enable users to view past and current queries, with options to track resolution progress, review responses, and provide feedback on the resolution experience.

**Support Agent Dashboard**

Offer support agents an interface to manage assigned queries, respond to users, update statuses, and collaborate on resolutions.

**Create Query**

Support agents can create, edit, and update query records, ensuring accurate tracking and resolution management.

**Admin Dashboard**

Provide administrators with tools to oversee query listings, user accounts, support agent activities, and platform operations, including user and agent management.

**Reporting and Analytics**

Generate reports and analytics on query volumes, resolution times, user satisfaction, and other metrics to gain insights into platform performance**.**

**Integration with External APIs**

Integrate with third-party APIs for services like secure file storage, notification systems, and user support tools to enhance platform functionality and user experience.

**TECHNOLOGY STACK**

**Frontend:**

1. React
2. Tailwind CSS
3. Bootstrap

**Backend:**

1. Node.js
2. Express.js

**Database:**

1. MongoDB

**Other:**

1. Axios
2. Multer – for image uploading
3. Recharts – for analytics
4. React Router DOM

**PROJECT STRUCTURE**

RESOLVE NOW/

│

├── Backend/

│ │ └── config.js

│ ├── server.js

│ ├── package. Json

│ └── package-lock. Json

│ ├── schema.js

├── frontend/

│ ├── node\_modules/

│ ├── public/

│ ├── src/

│ │ ├── Admin/

│ │ ├── Agent/

│ │ ├── Common/

│ │ ├── User/

│ │ ├── Images/

│ │ ├── App.js

│ │ ├── App.css

│ │ └── index.js

│ ├── index.html

│ ├── vite.config.js

│ ├── package. json

│ └── package-lock. Json

**BACKEND**

The key files used in the Backend development are

* server.js: Main file to initialize the server, connect to MongoDB, and manage routes.
* config.js: Contains the MongoDB URI and connection logic.
* db/Support & db/Users: Holds Mongoose schemas and route handlers for user and support agent operations.
* uploads/: Stores files (e.g., evidence or attachments) uploaded by users**.**

**Technologies Used:**

* **Node.js** – Runtime for server-side JavaScript execution
* **Express.js** – Framework for building RESTful APIs
* **MongoDB** – NoSQL database for storing query and user data
* **Multer** – Middleware for handling file uploads

**The backend handles:**

* API endpoints for queries, users, and support agents
* File uploads for query-related attachments
* Database operations for CRUD (Create, Read, Update, Delete)
* Authentication and role-based access control

**FRONTEND**

**1. Technologies Used:**

* React.js – For building dynamic user interfaces
* Vite – Fast build tool for React
* Tailwind CSS / Bootstrap – For responsive design and styling
* Axios – For handling API requests
* React Router DOM – For navigation between pages

**2. Key Folders:**

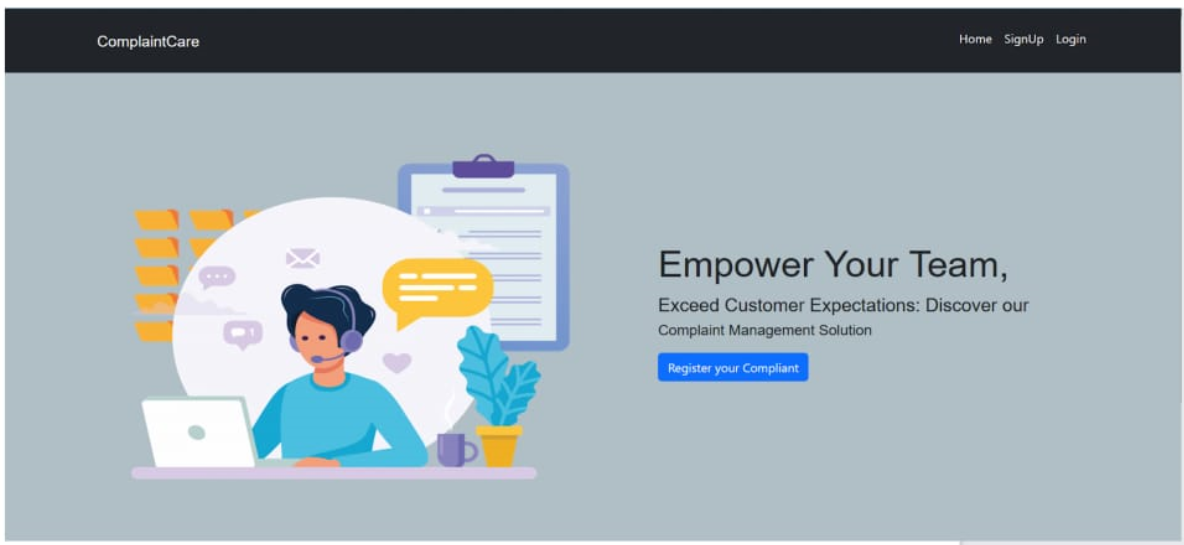
* src/Admin/ – Components for admin tasks (e.g., manage users, queries)
* src/Support/ – Allows support agents to view, respond to, and resolve queries
* src/User/ – Interfaces for submitting queries, tracking complaints, and managing profiles
* src/Components/ – Common components like Navbar, Footer, QueryCards

**3. Main Files:**

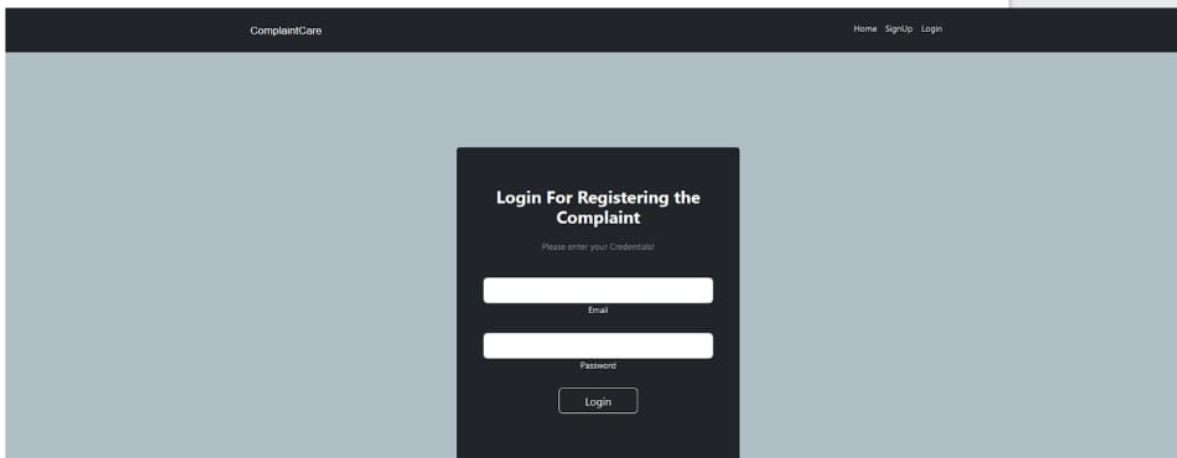
* App.jsx: Main component defining the route structure
* main.jsx: React entry point that renders <App /> into the DOM
* App.css / index.css: Global and default styling
* vite.config.js: Configuration for the Vite build process
* index.html: Template used by Vite to inject React content

**PROJECT IMPLEMENTATION**

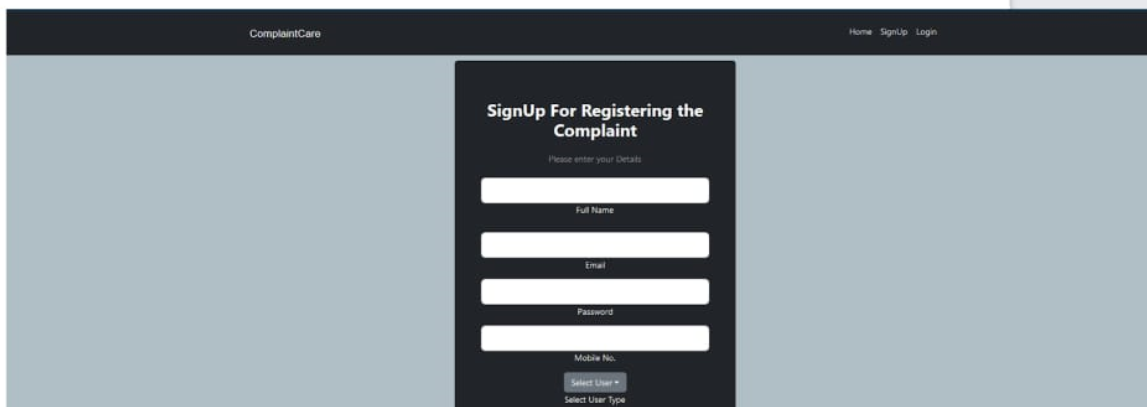
**Landing page: -**

****

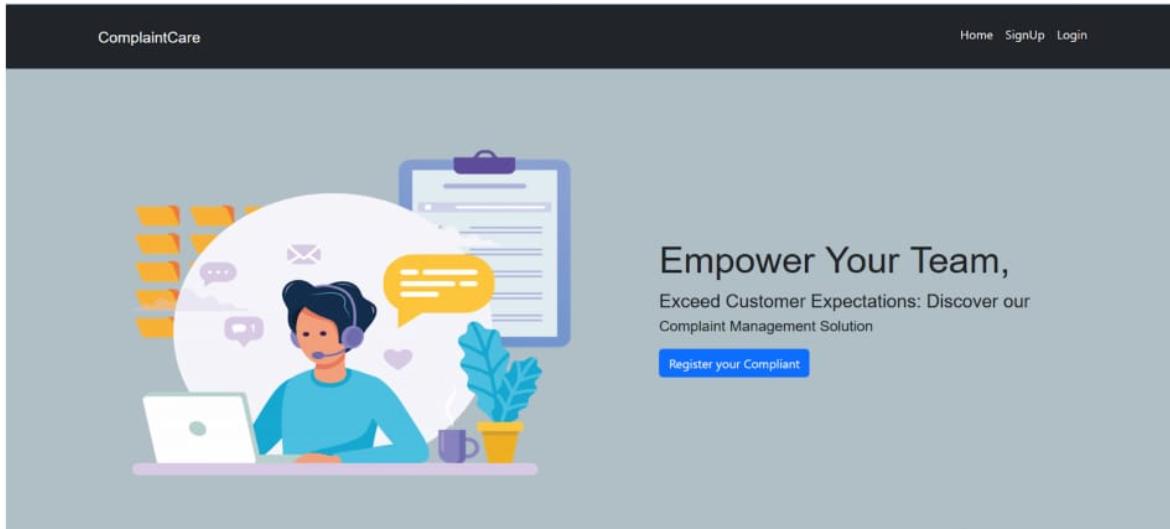
**Login Page: -**

****

**Registration Page: -**

****

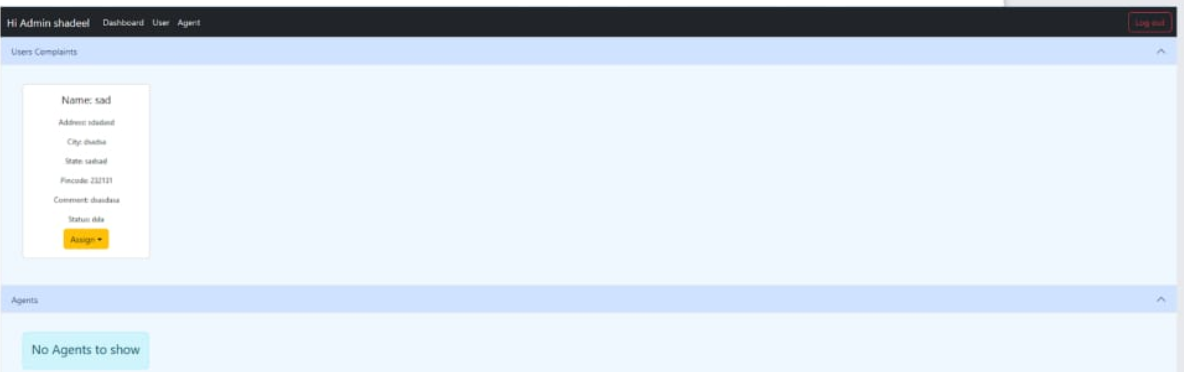
**Landing Page:**

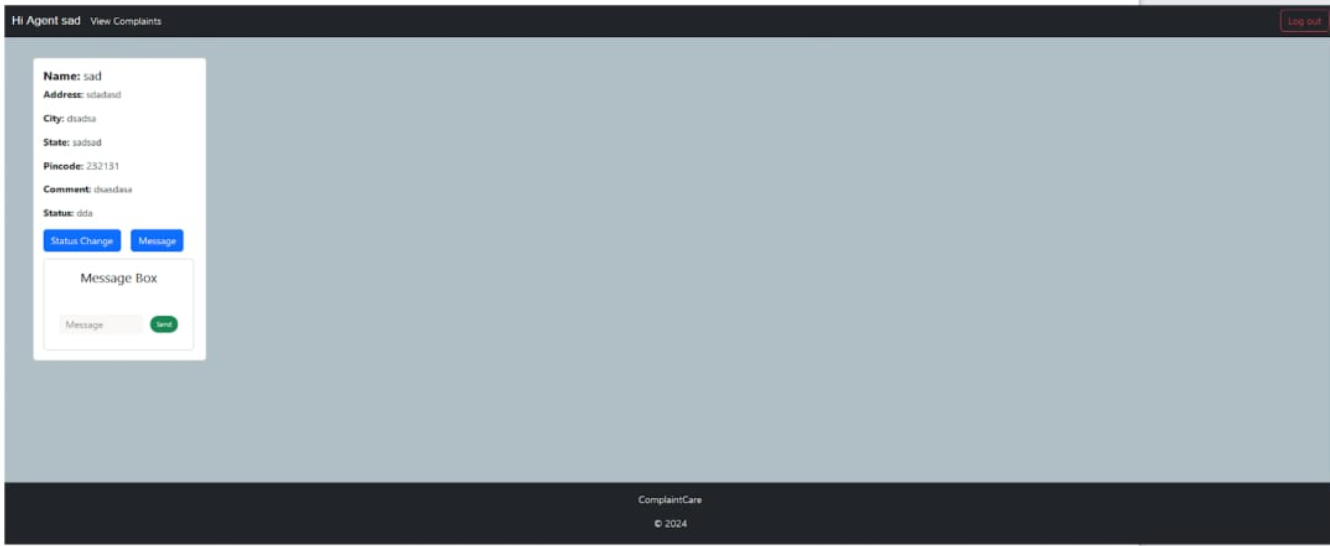
****

**Common Dashboard For Complaint:-**

****

**Admin Dashboard: -**

****

****

**Summary: Resolve Now, Your Online Query Solution**

Resolve Now is a full-stack online complaint resolution platform built using the MERN stack (MongoDB, Express.js, React, Node.js). It offers a seamless interface for users to submit, track, and resolve queries effortlessly. Key features include secure user registration and login, detailed query tracking with category and status filters, complaint history, and personalized query management. The responsive design ensures accessibility across desktops, tablets, and mobile devices, delivering a smooth, user-friendly experience.

The platform includes dedicated dashboards for support agents and administrators. Support agents can manage and respond to queries, while admins oversee users, agents, query listings, and platform analytics. Technologies like Axios streamline API requests, Multer supports file uploads for query evidence, and Recharts visualizes analytics data. With its intuitive UI and robust backend, Resolve Now provides a comprehensive and efficient solution for online complaint management.