Full Stack Development with MERN

Frontend Development Report

Date	21/Jul/2024
Team ID	SWTID1720103759
Project Name	Resolve Radar
Maximum Marks	

Project Title: Complaint Management System

Date: 21/Jul/2024

Prepared by: Resolve Radar

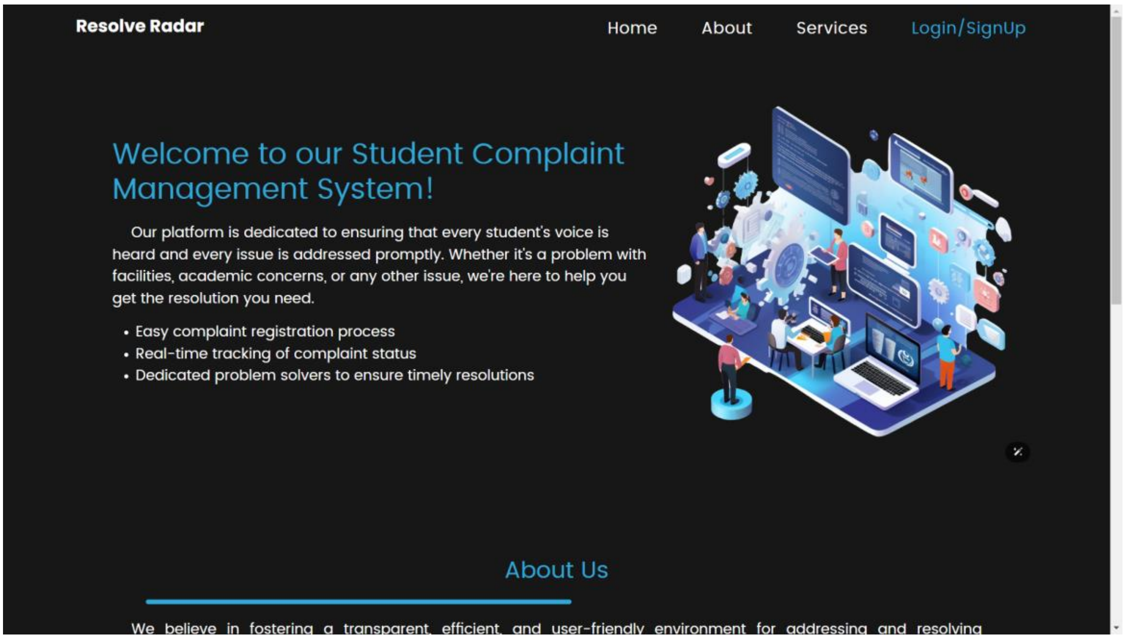
Objective

The objective of this report is to document the frontend development progress and key aspects of the user interface implementation for the Complaint Management System project.

Technologies Used

- **Frontend Framework:** React.js
- **State Management:**
- **UI Framework/Libraries:** Bootstrap
- **API Libraries:** Bcrypt, Axios, Mongoose and Express

Project Structure



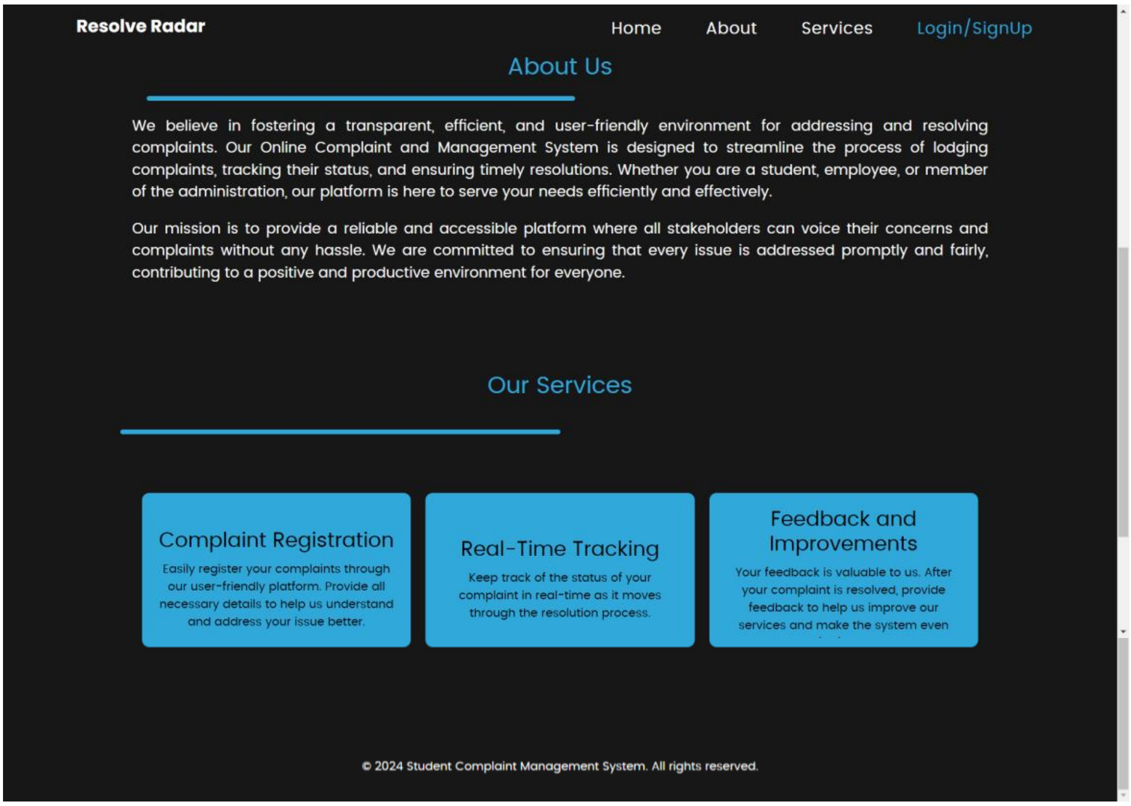
Welcome to our Student Complaint Management System!

Our platform is dedicated to ensuring that every student's voice is heard and every issue is addressed promptly. Whether it's a problem with facilities, academic concerns, or any other issue, we're here to help you get the resolution you need.

- Easy complaint registration process
- Real-time tracking of complaint status
- Dedicated problem solvers to ensure timely resolutions

About Us

We believe in fostering a transparent, efficient, and user-friendly environment for addressing and resolving



About Us

We believe in fostering a transparent, efficient, and user-friendly environment for addressing and resolving complaints. Our Online Complaint and Management System is designed to streamline the process of lodging complaints, tracking their status, and ensuring timely resolutions. Whether you are a student, employee, or member of the administration, our platform is here to serve your needs efficiently and effectively.

Our mission is to provide a reliable and accessible platform where all stakeholders can voice their concerns and complaints without any hassle. We are committed to ensuring that every issue is addressed promptly and fairly, contributing to a positive and productive environment for everyone.

Our Services

Complaint Registration

Easily register your complaints through our user-friendly platform. Provide all necessary details to help us understand and address your issue better.

Real-Time Tracking

Keep track of the status of your complaint in real-time as it moves through the resolution process.

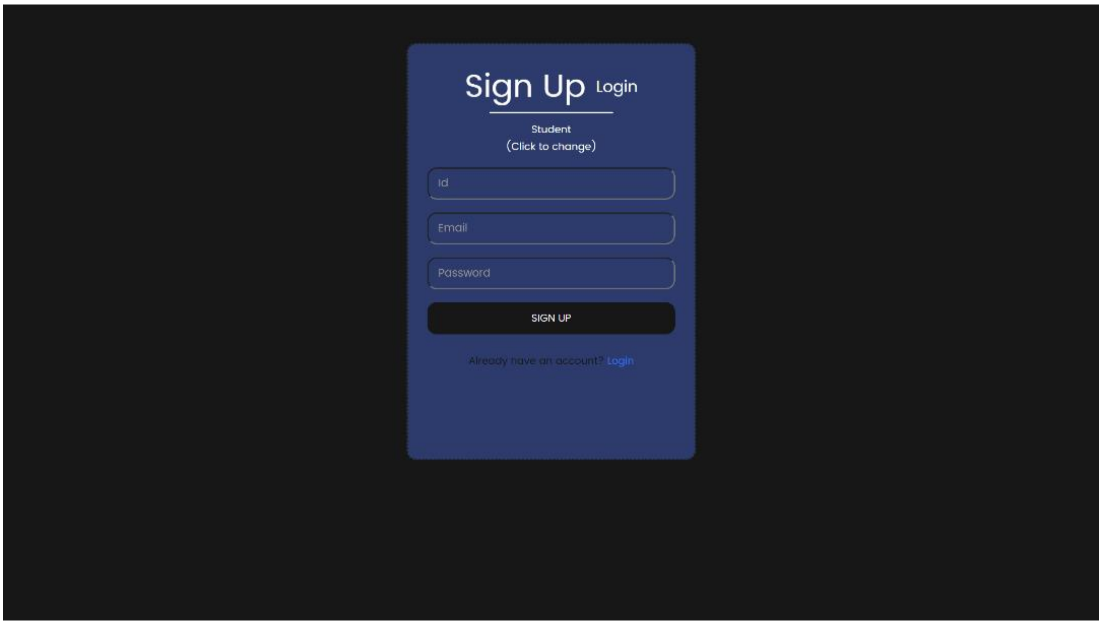
Feedback and Improvements

Your feedback is valuable to us. After your complaint is resolved, provide feedback to help us improve our services and make the system even

© 2024 Student Complaint Management System. All rights reserved.

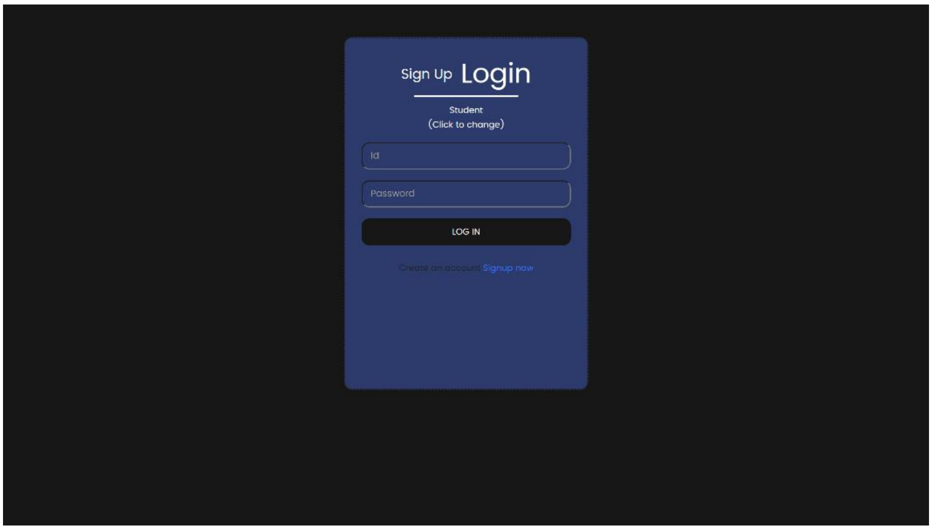
Landing Page:

The application's landing page is designed to provide a welcoming and intuitive entry point for users. Following a minimalistic design approach, the landing page focuses on essential elements to ensure a distraction-free user experience. It contains the ‘About Us’ and ‘Our Services’ information.



Sign Up:

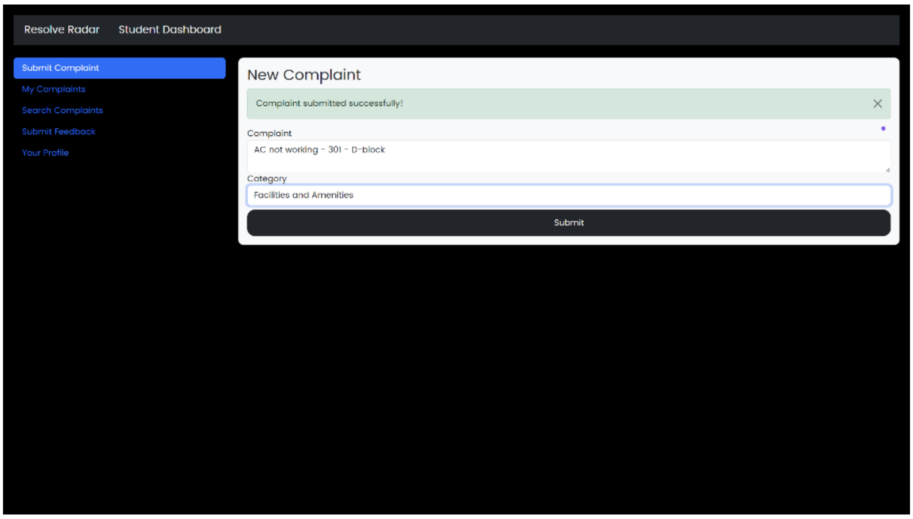
The users can sign up with their ID, email and password.



Student Dashboard:

Login:

Users can log in using their ID and password



New Complaint

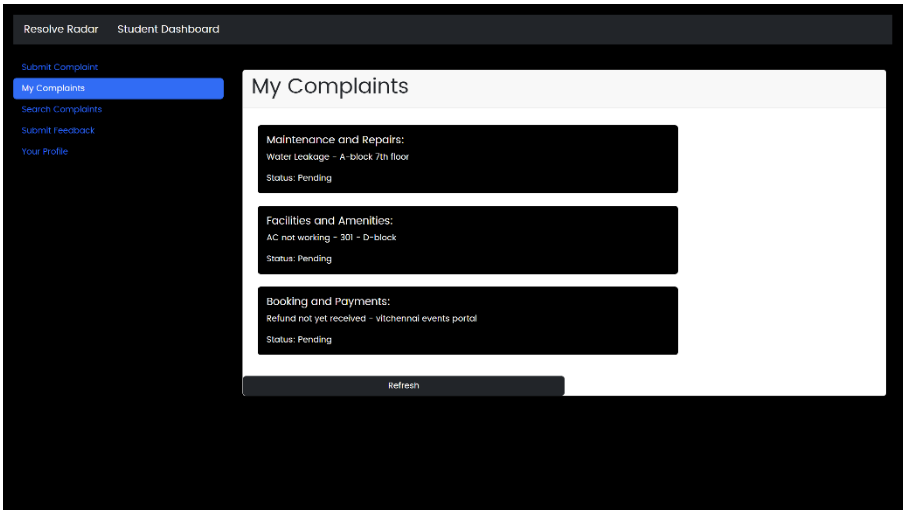
Complaint submitted successfully!

Complaint: AC not working - 301 - D-block

Category: Facilities and Amenities

Submit Complaint:

Users can create a new complaint, choose the category and submit the complaint.



My Complaints

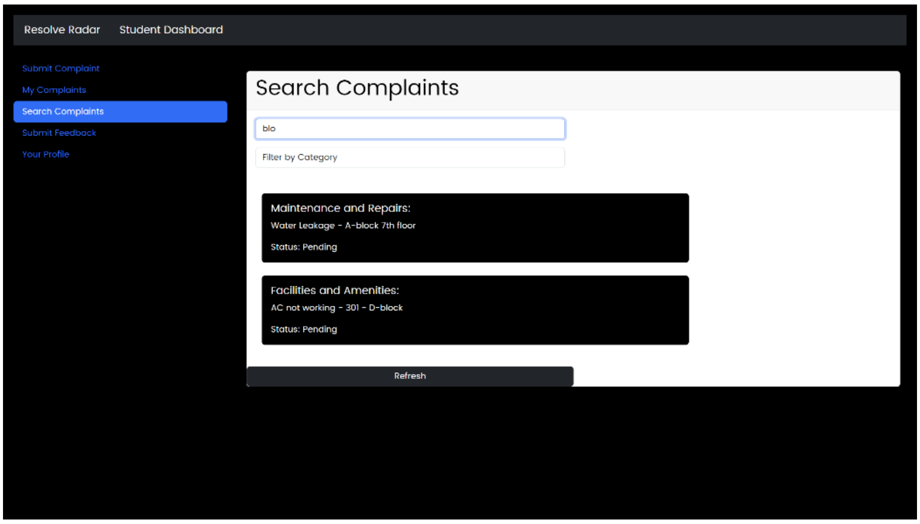
Maintenance and Repairs:
Water Leakage - A-block 7th floor
Status: Pending

Facilities and Amenities:
AC not working - 301 - D-block
Status: Pending

Booking and Payments:
Refund not yet received - vitchennai events portal
Status: Pending

My Complaint:

On this page, the students can view their complaints



Search Complaints

blo

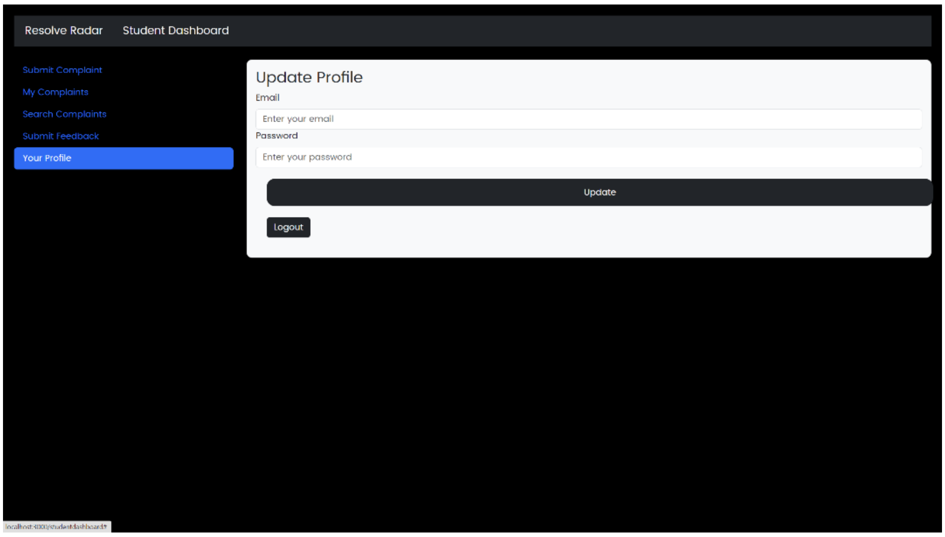
Filter by Category

Maintenance and Repairs:
Water Leakage - A-block 7th floor
Status: Pending

Facilities and Amenities:
AC not working - 301 - D-block
Status: Pending

Search Complaints

A search bar is employed where the students can search for complaints.



Update Profile

Email

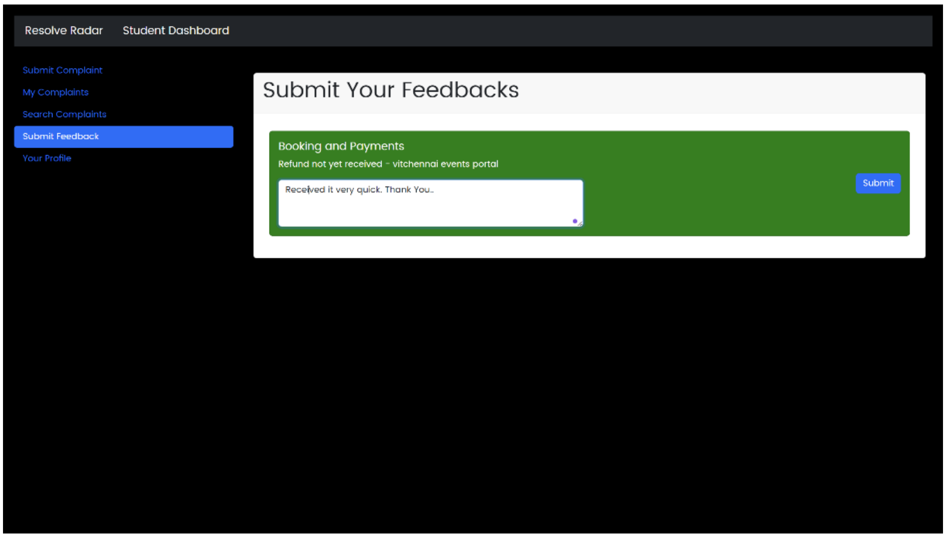
Enter your email

Password

Enter your password

Your Profile

The students can update their email and password on this page



Submit Your Feedbacks

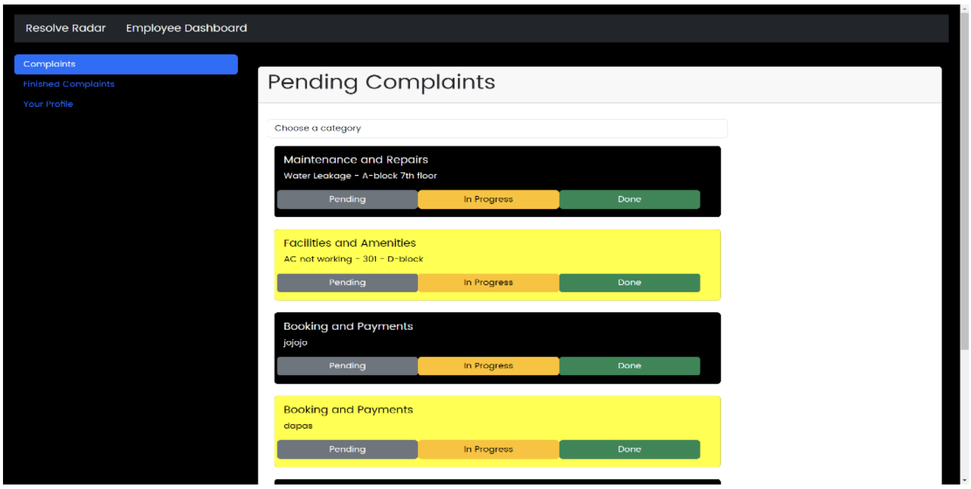
Booking and Payments
Refund not yet received - vitchennai events portal

Received it very quick. Thank You..

Submit Feedback

This option helps the students to submit their feedback on the complaints

Employee Dashboard:



Pending Complaints

Maintenance and Repairs
Water Leakage - A-block 7th floor

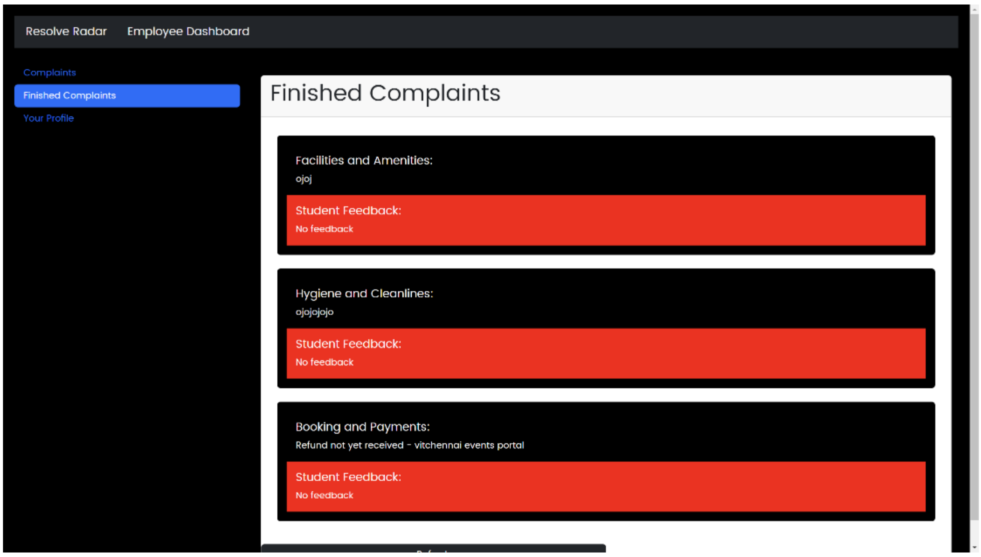
Facilities and Amenities
AC not working - 301 - D-block

Booking and Payments
jojojo

Booking and Payments
dapas

Pending Complaints:

The employee will be able to see the complaints listed as pending, in progress or done.



Finished Complaints

Facilities and Amenities:
ojoj

Student Feedback:
No feedback

Hygiene and Cleanlines:
ojojojojo

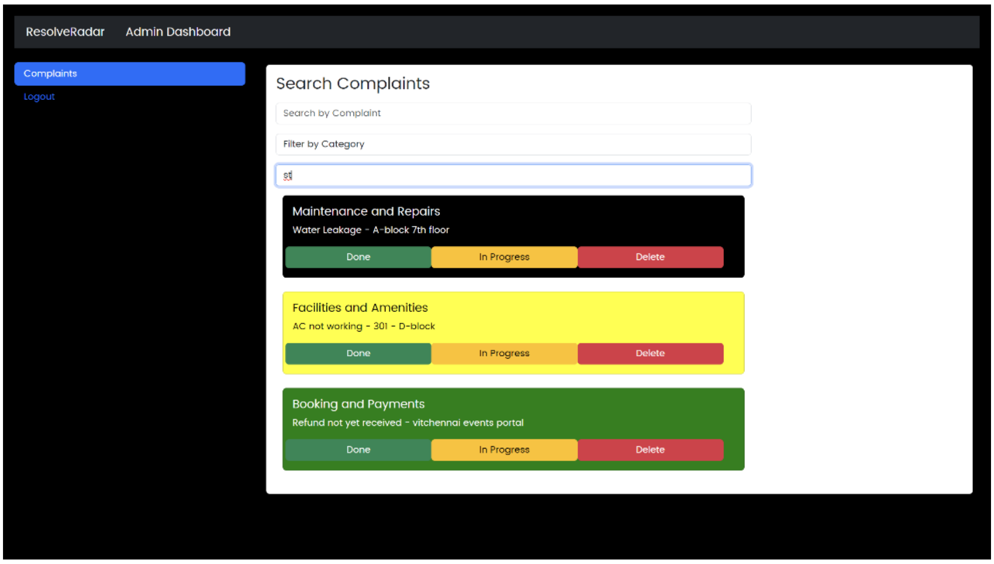
Student Feedback:
No feedback

Booking and Payments:
Refund not yet received - vitchennai events portal

Student Feedback:
No feedback

Finished Complaints:

The employee will be able to see the list of complaints already done along with the student's feedback.



Search Complaints

st

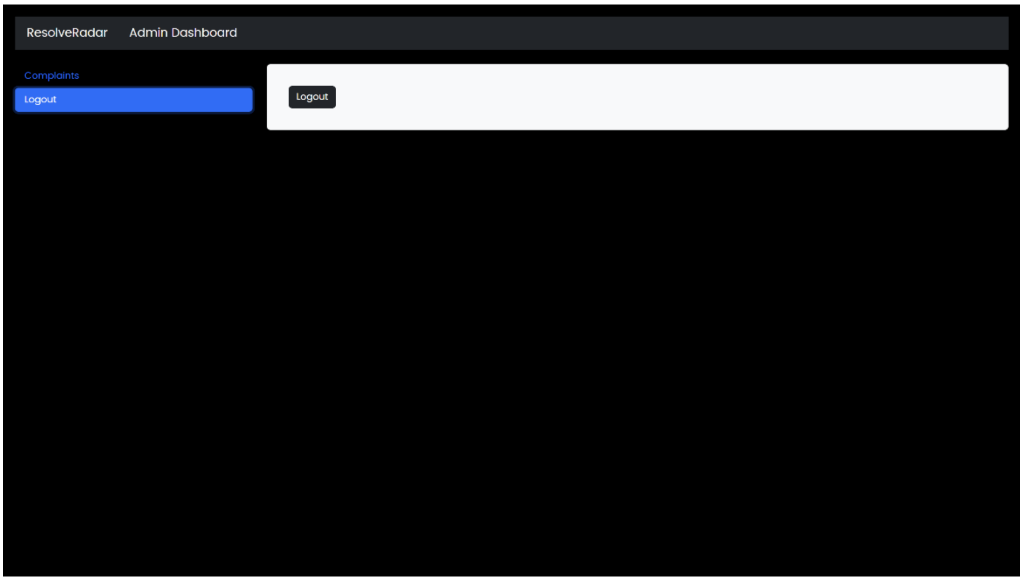
Maintenance and Repairs
Water Leakage - A-block 7th floor

Facilities and Amenities
AC not working - 301 - D-block

Booking and Payments
Refund not yet received - vitchennai events portal

Admin Dashboard

The admin can view the complaints and their progress



Logout

Admin can logout from the app using this page

Key Components

1. **App.js**
   - Responsible for routing and main application layout.
2. **/components**
   - Contains reusable UI components used across the application.

Routing

Routing is managed using React Router. Here are the main routes:

- **'/'**- Landing page of the application.
- **'/users'** – For the users to sign up for the app.
- **'/studentdashboard'** - Student profile management.
- **'/employeedashboard'**- Employee profile management
- **'/admindashboard'-** Admin profile management

Integration with Backend

The frontend communicates with the backend APIs hosted on 'mongodb://localhost:27017/ResolveRadar'. Key endpoints include:

- **POST /Express/users/signup** - Handles user signup.
- **POST /Express/users/login** - Handles user authentication.
- **GET /Express/studentId** - Retrieves the Student data for display by student ID.
- **GET /Express/employeeId** - Retrieves the Employee data for display by employee ID.
- **POST/Express/studentId/addComplaint-**Add a complaint to a student

- **PUT/Express/users/updateProfile/studentId-**Update student profile(email/password)
- **PUT/Express/users/updateProfile/employeeId-**Update employee profile(email/password)
- **POST /Express/studentdashboard-** Create complaints.
- **GET /Express/studentdashboard** - Retrieves all the complaints data.
- **GET /Express/employeedashboard** - Retrieves all the Employee's complaints data.
- **GET /Express/studentdashboard/studentId-** Retrieves all the complaints data for display by ComplaintId.
- **GET /Express/employeedashboard/status-** Retrieves all the complaints data for display by complaint status.
- **PUT/Express/markAsDone/id -**Update the complaint status and mark it as done.
- **PUT/Express/markAsInProgress/id -**Update the complaint status and mark it as in progress.

User Interface (UI) Design

- The UI design follows an efficient design principle with the following features.

1. **Simplicity and Clarity**
   The design is minimalistic, focusing on essential elements to prevent user distraction. Each input field is clearly labeled with placeholder text and icons (user, email, and password), ensuring users quickly understand the required information. Buttons with straightforward labels like "Sign Up," "Login," and "Student/Employee" roles maintain clarity and simplicity. AdminDash and EmployeeDash interfaces use intuitive layouts to present navigation options and content areas clearly.

2. **Consistency**
   Consistency is maintained through uniform styling across all elements. The use of consistent button styles, input field designs, and iconography contributes to a cohesive look and feel. The consistent color scheme and typography enhance visual harmony, making the interface intuitive for users. This consistency extends to the AdminDash and EmployeeDash, where similar design patterns and component structures are used to create a familiar user experience.

3. **Navigation**
   The navigation is straightforward and accessible, with clear links to "Home," "About," "Services," and "Login/SignUp" on the landing page. AdminDash and EmployeeDash provide tabbed navigation for accessing different sections of the dashboard, such as complaints and profiles, ensuring users can easily navigate to different parts of the application.

4. **State management**
   State management in the provided code is handled using React's useState hook, which maintains local component states for handling user input and toggling between different roles and forms.

5. **Feedback and Visibility**
   The design provides immediate feedback to users through visual cues. For example, when users switch between "Sign Up" and "Login" forms, the active state is highlighted to show the current selection. Error handling is considered, with potential alerts or messages to guide users in case of input mistakes or

server errors. In AdminDash and EmployeeDash, tabs and navigation links clearly indicate the active section, ensuring users always know their current context.

6. **Accessibility**
   The design takes accessibility into account by ensuring that interactive elements like buttons and input fields are easily distinguishable and clickable. The use of icons alongside text labels improves comprehension for users with different needs, including those with limited literacy or language barriers. High contrast between text and background colors enhances readability.

7. **Responsiveness**
   The use of flexible containers and responsive design principles ensures that the interface is usable across different devices and screen sizes, enhancing the overall user experience. This responsiveness is crucial for both the SignUp/Login forms and the dashboards, allowing users to access the platform from various devices.

- Implemented using Bootstrap Library.