CRISIS RESPONSE AND REPORTS

Overview:

The Crisis Response and Reports serves as a comprehensive solution for individuals and organization to address and resolve various incidents and issues effectively. With its user-friendly interface and robust features, users can easily submit detailed reports, including incident descriptions, locations and timestamps, ensuring clarity and accuracy in communication.

Moreover, the platform fosters community engagement by enabling users to interact with each other's reports, offering support, suggestions, and solutions where needed. Through collaborative efforts and shared insights, the Reporting System empowers users to address challenges efficiently and foster a culture of problem-solving within their respective communities.

Furthermore, the Crisis Response and Reports prioritizes user experience and satisfaction by providing intuitive tools for managing profile information and preferences. Users have full control over their profiles, allowing them to customize their personal details, profile pictures, and bio to accurately represent themselves within the community. This emphasis on user empowerment and customization not only enhances user engagement but also cultivates a sense of ownership and belonging within the platform. By streamlining the reporting process and offering robust profile management features, the Crisis Response and Reports aims to promote transparency, accountability, and collaboration in addressing incidents and problems at both individual and organizational levels.

Features:

User Authentication and Authorization:

The system implements robust user authentication and authorization mechanisms to ensure that only authenticated users can access certain features and that each user has appropriate permissions based on their role.

Adding Posts:

Users can create new posts containing detailed information about incidents or problems they encounter. They can provide descriptions. locations, timestamps, and any other relevant data to help others understand the situation.

Liking Posts:

Users have the ability to express their support or agreement with a particular post by liking it. This features helps to surface the most popular or

important posts within the system and provides positive feedback to users who submit valuable content.

Commenting on Posts:

Users can engage in discussions by leaving comments on posts. They can ask questions, provide suggestions, share relevant information, or offer solutions to the reported incidents or problems. Comments foster collaboration and community engagement within the platform.

Viewing User Profile Information:

Each user has a profile page where they can view their personal information, including their username, total likes and total comments. The profile page also displays statistics such as the total numbers of posts created by the user and the total number of comments they have made.

Deleting Posts:

Post owners have the ability to delete their own posts if they no longer wish to have them displayed on the platform. This feature gives users control over the content they have published and allows them to manage their contributions effectively.

Technologies Used:

- Node.js
- Express.js
- MongoDB
- Mongoose
- Passport.js
- Bcrypt.js
- EJS (Embedded JavaScript templating)
- Bootstrap.

Architecture:

The Crisis Response and Reports follows a client-server architecture, with the following components:

- **Client-side:** HTML, CSS, and JavaScript files served to users' web browsers to render the user interface and handle client-side interactions.
- **Server-side:** Node.js application built with Express.js that handles HTTP requests, interacts with the database, performs authentication and authorization, and serves dynamic content using EJS templates.
- **Database:** MongoDB database used to store user data, posts, comments, and other application-related information.

- **Authentication:** Passport.js middleware integrated into the Express.js application to handle user authentication and authorization using local username and password credentials.
- **Frontend Templating:** EJS templating engine used to generate dynamic HTML content on the server-side, allowing for the creation of reusable templated and the inclusion of dynamic data.
- **Styling:** Bootstrap framework utilized for CSS styling, providing pre-designed components and responsive layout utilities for enhanced user interface design

Setup Instructions:

1. **Install Dependencies:** npm install express mongoose passport bcrypt body-parser express-session
2. **MongoDB:** Set up a mongoDB database and provide the connection URL in 'index.js'.
3. Run the application using 'node index.js'.

Usage:

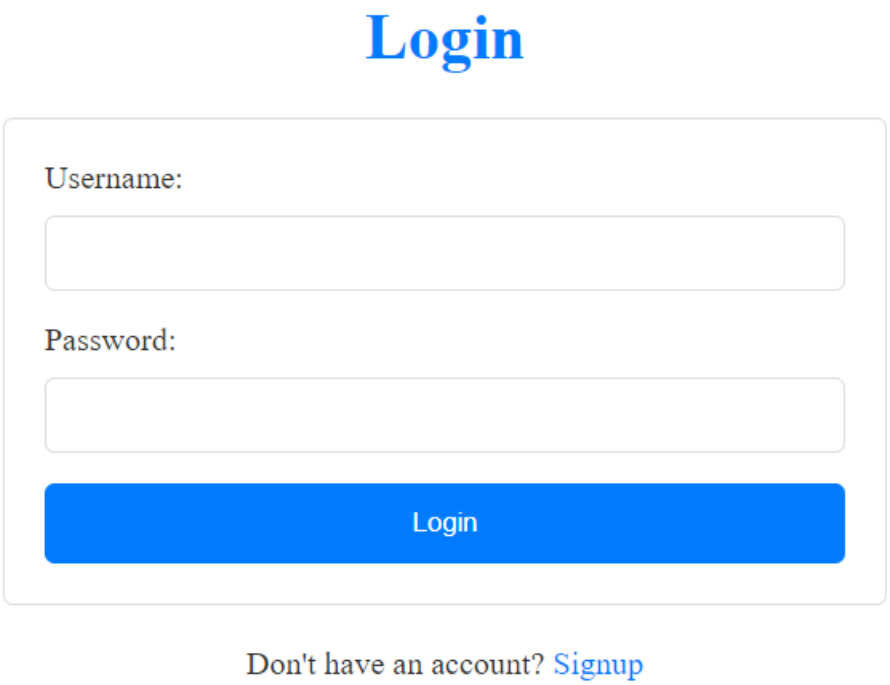
1. **Login:** Navigate to the login page and enter your credentials.
2. **Add post:** Go to the "Add post" page, fill in the incident and problem details, and submit the form.
3. **View posts:** Access the home page to view all posts, including likes and comments.
4. **List/Comment:** Interact with posts by liking them or adding comments.
5. **Profile:** Visit the profile page to view your personal information, total posts, and total comments.
6. **Delete posts:** If you are the owner of a post, you can delete it from the home page.

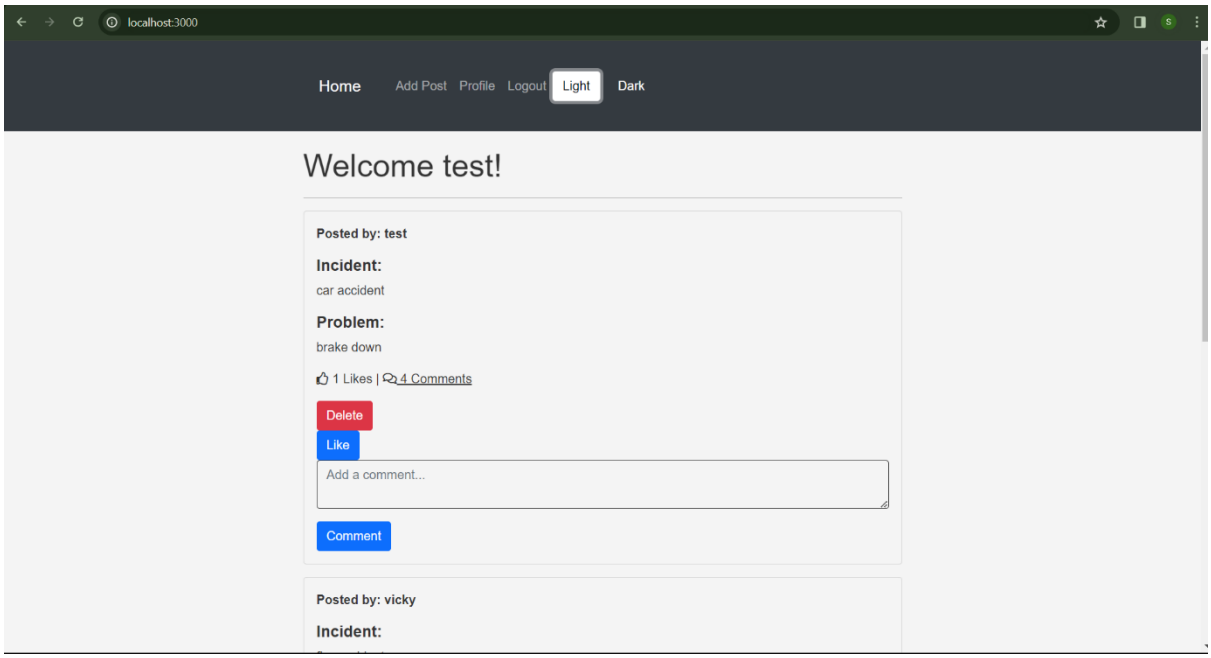
Future Improvements:

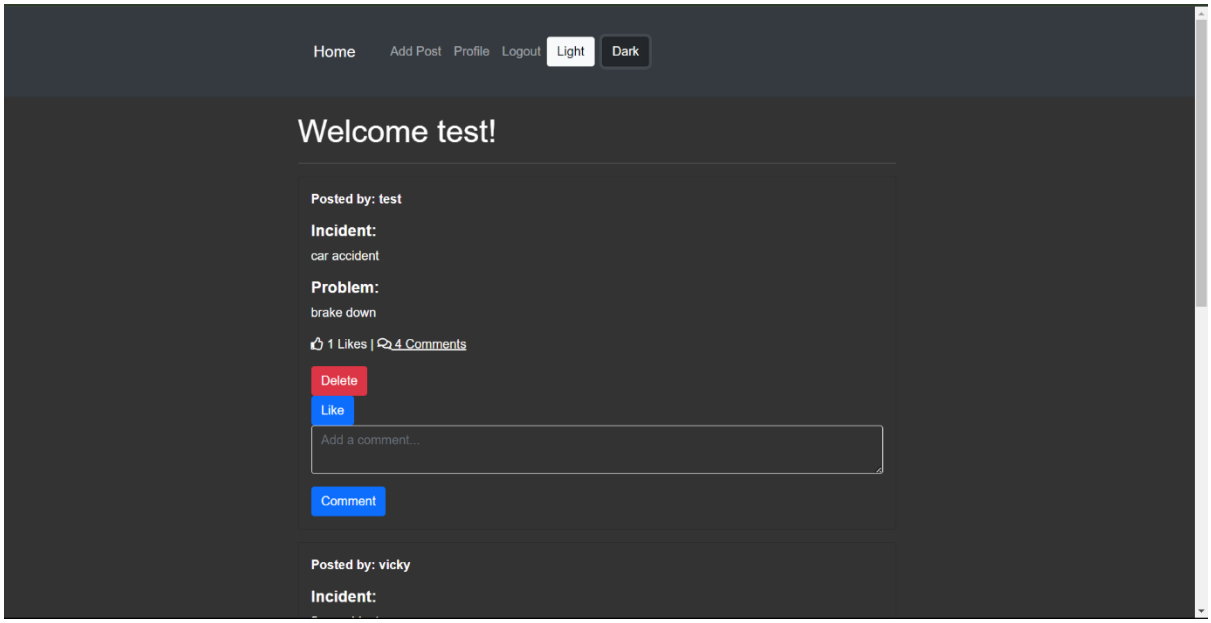
- Implement user roles and permissions to provide different levels of access and functionality based on user roles (e.g., admin, moderator, regular user).
- Enhance the user interface with additional styling, animations, and interactive components to improve usability and visual appeal.
- Integrate third-party authentication providers (e.g., Google, Facebook, GitHub) to offer users alternative sign-in methods and streamline the registration/login process.
- Implement real-time notifications to alert user about new likes, comments, or updates to their posts, enhancing user engagement and interaction within the platform.

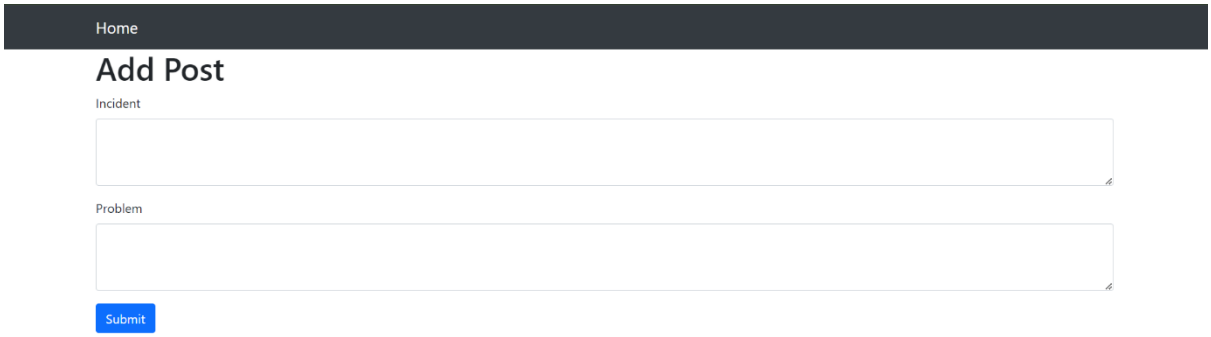
- Add functionality for user to report inappropriate content or users, and implement moderation features to manage reported content effectively.

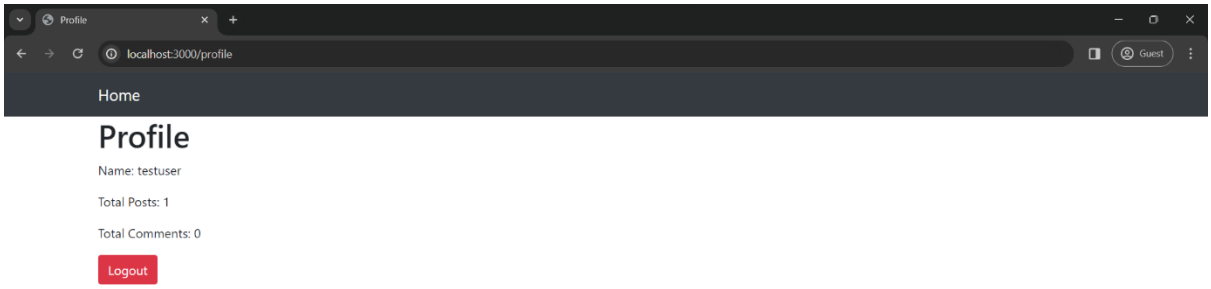
Final Output:






Home
Add Post
Profile
Logout
Light
Dark
Welcome test!
Posted by: test
Incident:
car accident
Problem:
brake down
1 Likes | 4 Comments
Delete
Like
Add a comment...
Comment
Posted by: vicky
Incident:


Home
Add Post
Incident
Problem
Submit


Profile
localhost:3000/profile
Guest
Home
Profile
Name: testuser
Total Posts: 1
Total Comments: 0
Logout