Project Report

# ResolveNow: Your Platform for Online Complaints

Category: Full Stack Development

## Skills Required:

HTML, CSS, JavaScript, Bootstrap, React.js, Node.js, MongoDB

## Project Description:

An online complaint registration and management system is a software application or platform that allows individuals or organizations to submit and track complaints or issues they have encountered. It can help optimize the complaint handling process and empower organizations to develop a safety management system to efficiently resolve customer complaints, while staying in line with industry guidelines and regulatory compliance obligations. It provides a centralized platform for managing complaints, streamlining the complaint resolution process, and improving customer satisfaction.

## Key Features:

* - User registration: Users can create accounts to submit complaints and track their progress.
* - Complaint submission: Users can enter details of their complaints, including relevant information such as name, description of the issue, address etc.
* - Tracking and notifications: Users can track the progress of their complaints, view updates, and receive notifications via email or SMS.
* - Interaction with agent: Users can communicate with the agent assigned to the complaint.
* - Assigning and routing complaints: The system assigns complaints to the appropriate department or personnel using intelligent routing algorithms.
* - Security and confidentiality: Ensures user data protection through authentication, encryption, access control, and regulation compliance.

## Description:

The Online Complaint Registration and Management System is a user-friendly software solution designed to streamline the process of submitting, tracking, and resolving complaints or issues encountered by individuals or organizations. It provides a centralized platform for efficient complaint management, allowing users to securely register complaints, track their progress in real-time, and interact with assigned agents for issue resolution. With features such as automatic notifications, intelligent complaint routing, and robust security measures, this system ensures timely and effective handling of complaints while prioritizing user details.

## Scenario:

Scenario: John, a customer, recently encountered a problem with a product he purchased online. He notices a defect in the item and decides to file a complaint using the Online Complaint Registration and Management System.

### User Registration and Login:

John visits the complaint management system's website and clicks on the "Sign Up" button to create a new account. He fills out the registration form, provides his full name, email address, and a secure password. After submitting the form, John receives a verification email and confirms his account. He then logs into the system using his email and password.

### Complaint Submission:

Upon logging in, John is redirected to the dashboard and clicks on the "Submit Complaint" button. He describes the issue in detail, attaches documents or images, and provides contact information. After reviewing, he submits the complaint.

### Tracking and Notifications:

John receives a confirmation of submission. He navigates to "My Complaints" and tracks the status in real-time. Email notifications inform him of updates like assignments or resolutions.

### Interaction with Agent:

Sarah, a customer service agent, is assigned the complaint. She reviews the complaint and contacts John through the system's chat feature. John responds and they discuss the issue further.

### Resolution and Feedback:

The company identifies the defect and offers a refund or replacement. John receives notification and provides positive feedback after resolution.

### Admin Management:

Admins monitor and assign complaints, ensuring smooth operation and policy compliance across the platform.

## Technical Architecture:

The technical architecture of our online complaint registration and management app follows a client-server model. The frontend, built with React.js, Bootstrap, and Material UI, connects to the backend using Axios and RESTful APIs.

The backend is developed with Express.js to handle server-side logic and communication. MongoDB is used for efficient and scalable data storage and retrieval, storing user profiles and complaints securely.

The complete system is integrated using technologies such as Socket.IO and WebRTC API to enable real-time messaging. This architecture supports efficient data exchange and ensures a seamless user experience for all roles: users, agents, and administrators.