

Full Stack Development with MERN

Project Documentation format

1. Introduction

- **Project Title:** ResolveNow – Online Complaint Registration & Management System
- **Team Members:**
 1. Dwarampudi Dinesh Eswar Reddy- Backend Developer
 2. Ragipindi Sailaja Reddy- Frontend Developer
 3. Naga Prathimasri Kanuri- Database & Integration Developer
 4. Endala Harshavardhan-Testing & Deployment Engineer

2. Project Overview

- **Purpose:** ResolveNow is a web-based complaint management system developed using the MERN stack. The purpose of the project is to provide a centralized digital platform where users can submit complaints, track their status, communicate with assigned agents, and receive notifications. It helps organizations manage complaints efficiently, improve transparency, and increase customer satisfaction.

- **Features:**

- User Registration & Login
- Email Verification
- Complaint Submission with Attachments
- Complaint Tracking Dashboard
- Admin Panel for Assignment
- Real-time Chat (Socket.io)
- Email Notifications
- Feedback System
- Role-Based Access (User, Agent, Admin)

3. Architecture

- **Frontend:** The frontend is built using React.js with Bootstrap for responsive UI design. Axios is used to communicate with backend REST APIs. The frontend consists of components such as Login, Registration, Dashboard, Complaint Form, Chat Window, and Admin Panel.
- **Backend:** The backend is developed using Node.js and Express.js. It handles authentication, complaint management logic, status updates,

assignment processing, and chat functionality. JWT (JSON Web Token) is used for secure authentication and authorization.

Database: MongoDB is used as the primary database to store:

- User details
- Complaint details
- Chat messages
- Feedback records

MongoDB Atlas is used for cloud deployment.

4. Setup Instructions

- **Prerequisites:**

- Node.js (v16 or above)
- MongoDB (Local or MongoDB Atlas)
- Git
- VS Code

Installation:

1. Clone repository

```
git clone <repository-link>
```

2. Install frontend dependencies

```
cd client
```

```
npm install
```

3. Install backend dependencies

```
cd server
```

```
npm install
```

4. Create .env file in server folder

```
MONGO_URI=your_mongodb_connection
```

```
JWT_SECRET=your_secret_key
```

```
EMAIL_USER=your_email
```

```
EMAIL_PASS=your_password
```

5. Folder Structure

• **Client:**

```
client/
  └── src/
    ├── components/
    ├── pages/
    ├── services/
    ├── App.js
    └── index.js
  └── public/
    └── package.json
```

• **Server:**

```
server/
  ├── controllers/
  ├── models/
  ├── routes/
  ├── middleware/
  ├── config/
  ├── server.js
  └── package.json
```

6. Running the Application

• **Frontend:**

```
cd client
```

```
npm start
```

• **Backend:**

```
cd server
```

```
npm start
```

7. API Documentation

Method	Endpoint	Description
POST	/api/auth/register	Register new user
POST	/api/auth/login	Login user
GET	/api/auth/profile	Get user profile

8. Authentication

Authentication is implemented using **JWT (JSON Web Tokens)**.

- User logs in
- Server generates JWT token
- Token stored in local storage
- Protected routes verified using middleware
- Passwords encrypted using bcrypt

Role-based access control is implemented for:

- User
- Agent
- Admin

9. User Interface

• The UI includes:

- Registration Page
- Login Page
- User Dashboard
- Complaint Submission Form
- Chat Interface
- Admin Monitoring Panel

Screenshots should include:

- Login Page
- Dashboard
- Complaint Form
- Admin Panel

10. Testing

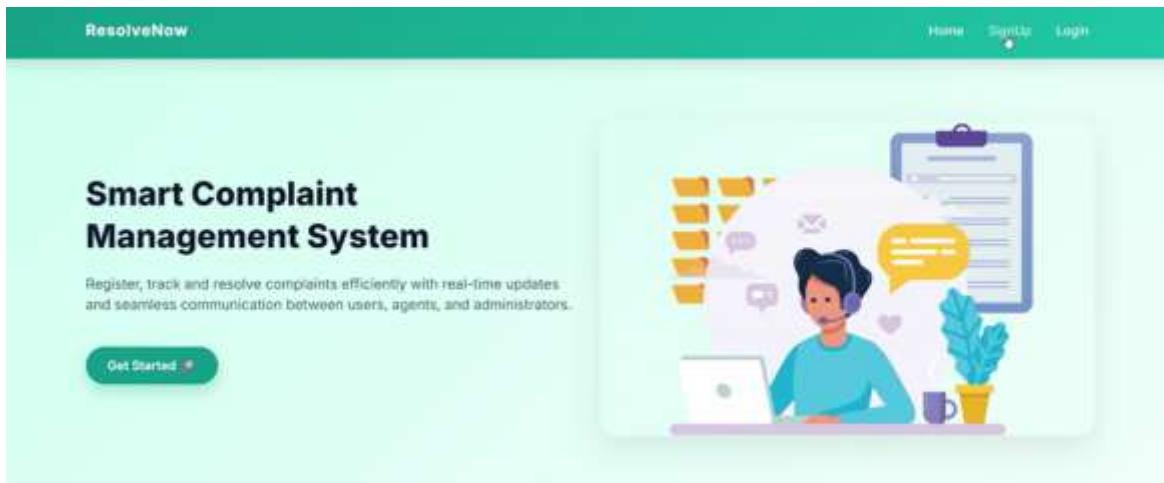
• Testing strategy includes:

- Manual functional testing
- API testing using Postman
- Authentication testing
- Integration testing
- Error handling validation

11. Screenshots or Demo

GITHUB LINK:

<https://github.com/sailaja-37/ResolveNoww.git>

This screenshot shows the 'Register a Complaint' form. The top navigation bar includes 'Hi, Dinesh', 'Complaint Register', 'Status', and 'Log Out'. The main form has fields for 'Name' (Dinesh), 'Address' (Enter your address), 'City' (City), 'State' (State), 'Pincode' (Pincode), and 'Postal Code' (Postal Code). There is also a 'Description' field with a placeholder 'Describe your complaint in detail' and a large text area below it. A 'Submit Complaint' button is at the bottom right of the form.

This screenshot shows the user dashboard for 'chinni'. The top navigation bar includes 'Hi chinni', 'Dashboard', 'Users / Agents', and 'Log Out'. The main content area is divided into two sections: 'Users Complaints' and 'Agents'. The 'Users Complaints' section displays a card for 'Dinesh' with details: Address: 1-146, Basam Street, Almora, Ponnanchira Mandal, W.D district, Andhra Pradesh, 534298; City: Almora - west godavari; State: Andhra Pradesh; Pincode: 534298; Comment: Certified Theft; Status: completed. The 'Agents' section shows a card for 'raghu' with the email 'Email: raghu@gmail.com'.

12. Known Issues

- Real-time chat may experience delay under heavy load
- Email notifications depend on SMTP configuration
- No AI-based complaint categorization implemented yet

13. Future Enhancements

- AI-based automatic complaint categorization
- Mobile application version
- SMS notification integration
- Analytics dashboard with graphical reports
- Multi-organization support
- Cloud auto-scaling implementation