**Online Complaint Registration and Management System**

**🧭 ResolveNow – Guided Project Walkthrough**

**🎯 Objective**

**Build a full-stack complaint management web application where users can:**

* **Register/Login securely**
* **File complaints**
* **Track complaint progress**
* **Chat with assigned agents**

**Admins can:**

* **Manage users/agents**
* **Assign & resolve complaints**
* **Send updates and notifications**

**🔧 Step 1: Clone the Repository**

**git clone https://github.com/Salman-shaik116/Resolve\_online\_complaints.git**

**cd Resolve\_online\_complaints**

**🧪 Step 2: Install Project Dependencies**

**🖥️ Backend Setup**

**cd backend**

**npm install**

**Create a .env file:**

**MONGO\_URI=your\_mongodb\_connection\_string**

**JWT\_SECRET=your\_jwt\_secret**

**EMAIL\_USER=your\_email@example.com**

**EMAIL\_PASS=your\_email\_password**

**Start the backend server:**

**npm start**

**The backend runs on http://localhost:5000.**

**🌐 Frontend Setup**

**Open a new terminal tab:**

**cd frontend**

**npm install**

**npm start**

**This launches React frontend on http://localhost:3000.**

**👥 Step 3: User Roles and Actions**

**👤 Normal User**

* **Register/Login**
* **File complaint**
* **View complaint history and status**
* **Chat with agent**

**🛠️ Agent**

* **Login**
* **See assigned complaints**
* **Respond to user via chat**
* **Mark complaints as resolved**

**👨‍💼 Admin**

* **Login (admin credentials in DB or seed data)**
* **View all complaints**
* **Assign complaints to agents**
* **Manage users/agents**
* **Change complaint statuses**

**🔐 Step 4: Authentication Flow**

* **JWT tokens are issued on login**
* **Stored in localStorage**
* **Protected routes check for valid token using middleware**
* **bcryptjs handles password hashing**

**✉️ Step 5: Email Notification Setup**

**Uses nodemailer to send updates:**

* **Admin assigns/resolves complaint → user gets email**
* **Set up Gmail or SMTP credentials in .env**

**🧱 Folder Overview**

**Resolve\_online\_complaints/**

**│**

**├── backend/**

**│ ├── models/ # User, Complaint, Message schemas**

**│ ├── controllers/ # Business logic**

**│ ├── routes/ # API endpoints**

**│ └── server.js # Express entry point**

**│**

**├── frontend/**

**│ ├── src/**

**│ │ ├── components/ # Navbar, ComplaintForm, ChatWindow, etc.**

**│ │ ├── pages/ # Login, Register, Dashboard, AdminPanel**

**│ │ └── App.js**

**📲 API Quick Preview**

| **Endpoint** | **Method** | **Description** |
| --- | --- | --- |
| **/auth/register** | **POST** | **Register new user** |
| **/auth/login** | **POST** | **Login and get JWT** |
| **/complaints** | **GET/POST** | **Get/create user complaint** |
| **/admin/complaints** | **GET** | **Admin fetches all complaints** |
| **/admin/complaints/:id** | **PUT** | **Admin updates status** |

**🎯 What You Can Build On Top**

**Here are some mini-projects to extend it:**

**1. Complaint Status Filtering**

**Add filters by date/category/status in the user and admin dashboard.**

**2. Real-Time Chat (Socket.io)**

**Replace chat polling with real-time agent-user messaging.**

**3. PDF Complaint Report**

**Allow users to export their complaint details as a PDF report.**

**4. Notification Bell**

**Implement a dashboard notification center using a custom component.**

**5. Analytics for Admin**

**Add a page showing:**

* **Total complaints this month**
* **Resolved vs pending**
* **Agent-wise resolution rate**

**📹 Demo & Resources**

* **Demo Video: in GitHub**
* **Docs PDF: also, in GitHub**
* **DB Schema Info: backend/models/**

**🤝 Contribution Guide**

**Want to improve this project?**

**# Create your own branch**

**git checkout -b feature-name**

**# Make your changes and commit**

**git commit -m "Added new feature"**

**# Push and create a pull request**

**git push origin feature-name**