**PROJECT TITLE:RESOLVE NOW**

(PLATFORM FOR ONLINE COMPLAINTS)

1. INTRODUCTION

1.1 Project Overview

ComplainConnect is a digital complaint management platform designed to empower citizens, customers, and employees to lodge complaints seamlessly and have them resolved efficiently. Built using the robust MERN (MongoDB, Express.js, React.js, Node.js) stack, the application bridges the gap between people and the relevant authorities by offering an accessible, transparent, and user-friendly interface.

This platform allows users to register complaints, track status, upload supporting documents, and receive updates. It also equips administrators with powerful tools for managing, analyzing, and resolving complaints in a timely manner.

1.2 Purpose

The goal is to streamline the complaint lodging process, reduce bureaucratic delays, and ensure user satisfaction by creating a responsive and accountable system. The platform serves sectors like government services, customer support, educational institutions, or internal workplace grievance systems.

**2. IDEATION PHASE**

**2.1 Problem Statement**

In many institutions, there is no centralized, efficient, and transparent way for individuals to register and track complaints. Users are often unaware of where to report, leading to frustration and unresolved issues. The solution is a platform that ensures every complaint is heard, tracked, and resolved.

**2.2 Empathy Map Canvas**

* **Says:** “I never know who to contact for my issue.”
* **Thinks:** “Will my complaint even be looked at?”
* **Does:** Tries calling or emailing departments, often with no response.
* **Feels:** Frustrated, unheard, and discouraged.

**2.3 Brainstorming**

Key features envisioned:

* User registration and complaint submission
* Complaint tracking with status updates
* Admin dashboard for assignment and resolution
* Priority and category tagging
* Analytics for complaint trends
* Notifications via email/SMS

**3. REQUIREMENT ANALYSIS**

**3.1 Customer Journey Map**

1. **Register/Login**
2. **Submit Complaint**
3. **Upload Evidence**
4. **Receive Acknowledgement**
5. **Track Complaint Status**
6. **Receive Resolution or Feedback**
7. **Rate Resolution or Reopen Case**

**3.2 Solution Requirements**

**Functional Requirements:**

* User authentication and role-based access
* Complaint form with category, urgency, and attachment support
* Admin and resolver dashboards
* Status update system (e.g., Submitted → In Process → Resolved)
* Feedback/rating mechanism

**Non-Functional Requirements:**

* Mobile responsiveness
* Real-time updates and notifications
* Secure data storage and encryption
* Scalable architecture
* API integration (email, SMS, etc.)

**3.3 Data Flow Diagram**

* **UI Layer:** React.js interface
* **Web Server:** Node.js with Express
* **API Gateway:** Routes all frontend requests
* **Auth Service:** User login/registration with JWT
* **Complaint Service:** Handles creation, update, and retrieval
* **Database:** MongoDB for users, complaints, comments, etc.

**3.4 Technology Stack**

* **Frontend:** React.js
* **Backend:** Node.js + Express.js
* **Database:** MongoDB
* **Authentication:** JWT
* **Dev Tools:** GitHub, VS Code, Postman, npm
* **Deployment:** Render / Netlify / Heroku

**4. PROJECT DESIGN**

**4.1 Problem Solution Fit**

ComplainConnect enables fast, structured complaint registration and resolution tracking—minimizing delays and ensuring transparency.

**4.2 Proposed Solution**

* Complaint registration with detailed forms
* File attachments (e.g., images, PDFs)
* Role-based dashboards for users, staff, and admins
* Priority management for escalations
* Email/SMS status updates
* Analytics for authorities

**4.3 Architecture**

* **Frontend (React.js):** Form rendering, complaint view, filters
* **Backend (Express.js):** Routing, validation, security
* **MongoDB:** Complaint and user data
* **JWT Auth:** Secure login and session handling
* **API Layer:** Enables integration with external services

**5. PROJECT PLANNING & SCHEDULING**

**5.1 Planning Phases**

1. Requirements gathering and ideation
2. UI/UX wireframing
3. Database and backend design
4. Frontend development
5. API integration and testing
6. Deployment and maintenance

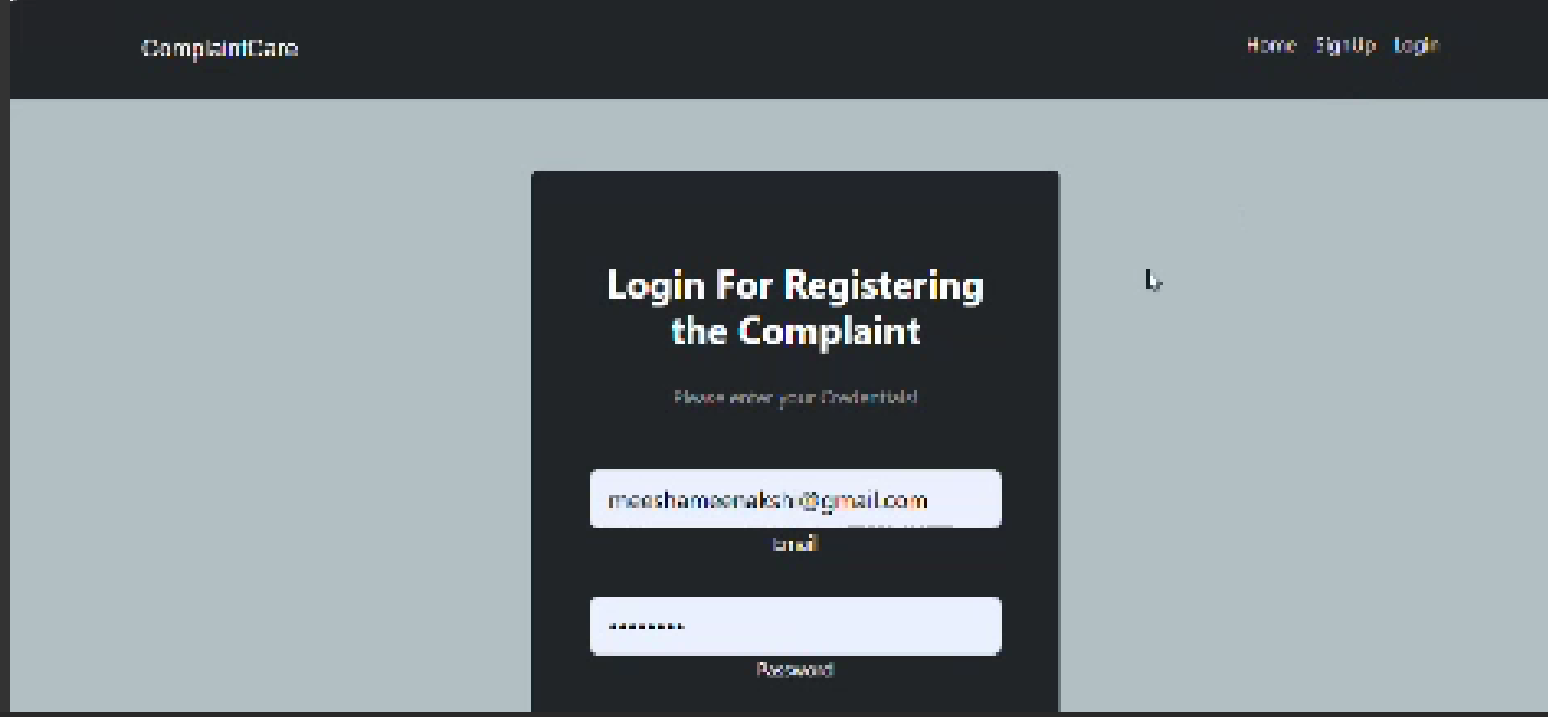
**6. FUNCTIONAL & PERFORMANCE TESTING**

**6.1 Testing Process**

* **Functional Testing:** Complaint submission, status update, notifications
* **Load Testing:** Multiple simultaneous complaint submissions
* **Security Testing:** Data protection, input validation
* **Cross-browser/device testing**

7. **RESULTS**

**7.1 Output Screenshots**



**8. ADVANTAGES & DISADVANTAGES**

**Advantages**

* Centralized complaint system
* Real-time updates
* Transparent tracking
* Secure and scalable
* Time-saving for users and admins

**Disadvantages**

* No offline complaint submission
* Depends on internet availability
* Admin overload without escalation flow

**9. CONCLUSION**

ComplainConnect offers a modern and efficient solution for handling complaints across organizations. It enhances user trust and accountability while improving operational transparency. Built with the MERN stack, it provides an extensible platform for future enhancements.

**10. FUTURE SCOPE**

* Mobile app version
* AI-based complaint categorization
* Chatbot for FAQ and support
* Integration with government/enterprise systems
* Multi-language support

**11. APPENDIX**

* **Source Code:** https://github.com/Vedasahithi07/smartbridge-ResloveNow-