**Project Design Phase**

**Solution Architecture**

|  |  |
| --- | --- |
| Date | 26 -052025 |
| Team ID | LTVIP2025TMID46583 |
| Project Name | Reslove Now |
| Maximum Marks | 4 Marks |

**Solution Architecture – Resolve Now**

The solution architecture of Resolve Now is designed to deliver a secure, responsive, and scalable online complaint registration and management system that effectively addresses the needs of users, agents, and administrators. It follows a modular client-server architecture built using the MERN stack (MongoDB, Express.js, React.js, Node.js) and is capable of future enhancements such as real-time communication and AI-based routing.

**Key Architectural Components:**

***Frontend (Client Interface – React.js):***

* Built with React.js and styled using Bootstrap + Material UI
* Fully responsive UI supporting mobile, tablet, and desktop views
* Role-based components (User, Agent, Admin) rendered based on login session
* Axios used for asynchronous communication with backend APIs
* Animations via AOS (Animate On Scroll) for smoother user experience

***Backend (Server Logic – Node.js + Express):***

* RESTful APIs developed using Express.js
* Handles authentication, session management, complaint routing, chat history, and admin logic
* Implements business logic for:
  + Complaint registration
  + Complaint status updates
  + Agent assignment
  + Escalation handling
* Middleware ensures secure routes and error handling

***Database (MongoDB + Mongoose):***

* NoSQL database to store complaints, users, agents, messages, etc.
* Mongoose schemas for:
  + Users (name, phone, role)
  + Complaints (category, description, status, timestamps)
  + Assigned complaints (agent mapping)
  + Chat/Feedback
* Indexed for fast querying and performance

***Authentication & Authorization:***

* JWT-based authentication ensures secure access
* Role-based authorization:
  + Users can submit/view complaints
  + Agents can view assigned tickets and respond
  + Admins have full control over data and routing
* Login sessions stored securely

***Real-Time & Notification Layer (Planned):***

* Integration with Socket.io for real-time updates & chat
* WebRTC-based audio/video call option for complaint escalation (future enhancement)
* Notifications via email/SMS on complaint status changes

***Admin Dashboard and Analytics:***

* View all complaints with filters (pending/resolved)
* Assign agents based on workload
* View platform metrics: resolution time, complaint counts, agent performance
* Export data for internal use/reporting

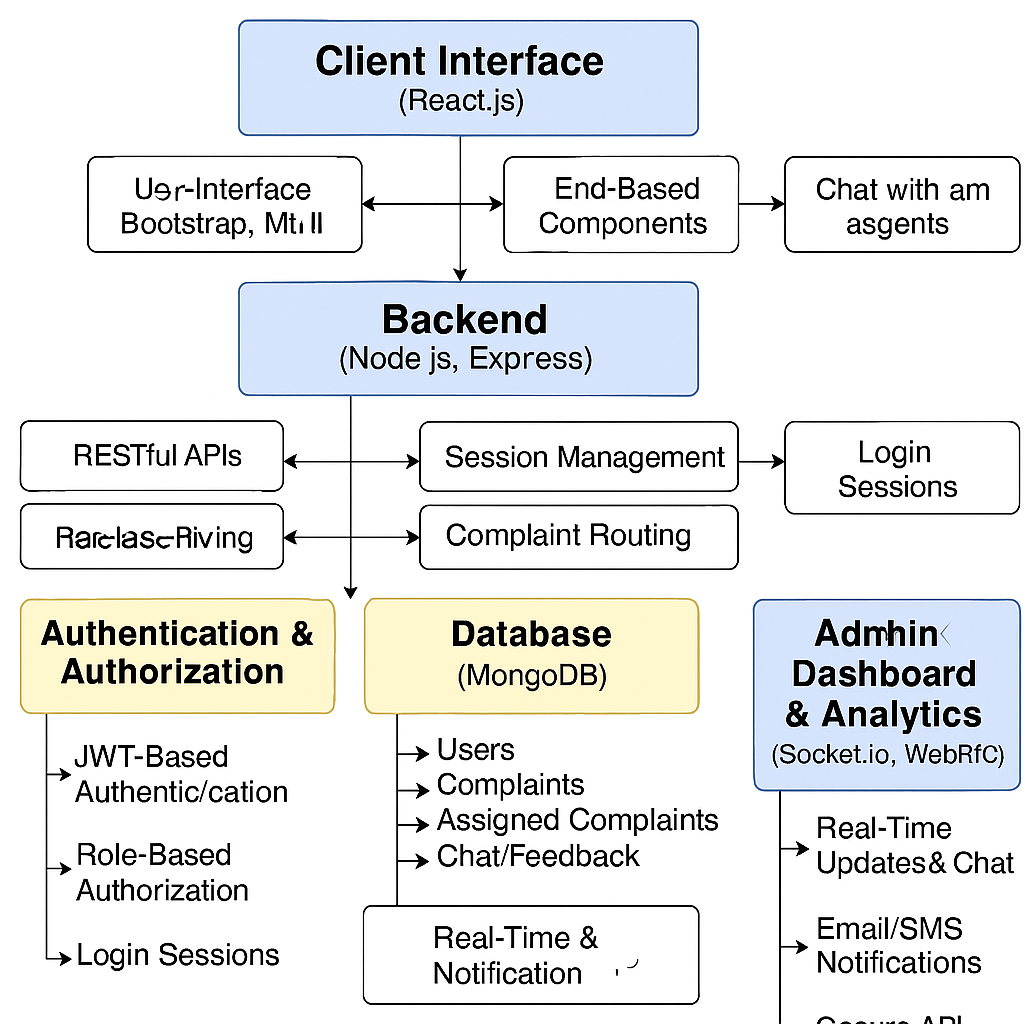
***Deployment & Scalability:***

* Hosted on Render or Railway for both frontend and backend
* .env used for environment config and API security
* Easily scalable architecture supports multiple departments or city-wide expansion
* API-driven design supports integration with external government systems or legacy tool

***Security Considerations:***

* Secure routing with encrypted tokens
* Input validation and rate limiting to prevent spam
* Role-based route protection

**Example - Solution Architecture Diagram:**



The solution architecture of *Resolve Now* is built using a modular, scalable, and role-based client-server model. It integrates a React.js frontend for a responsive and interactive user experience, and a Node.js + Express.js backend to manage core logic, authentication, and complaint processing. MongoDB serves as the database to store user, complaint, and resolution data in a flexible and efficient structure.

Authentication is handled through secure JWT tokens, enabling role-based access for users, agents, and administrators. The architecture also includes dashboards tailored to each role, allowing seamless complaint submission, status tracking, agent assignment, and resolution monitoring. Additionally, the system is designed to support real-time communication using technologies like Socket.io and WebRTC, enabling future features like live chat and video support. With this setup, Resolve Now delivers a secure, responsive, and future-ready platform for streamlined complaint management across public or private organization