

Project Report: ResolveNow

TEAM ID: LTVIP2025TMID53148

1. INTRODUCTION

1.1 Project Overview

ResolveNow is a web-based application designed to help users submit complaints, track their progress, and interact with customer support agents in real time. It streamlines complaint management by offering a centralized platform where users, agents, and admins can perform role-based actions. The system ensures transparency, security, and efficiency in resolving customer issues while maintaining regulatory compliance.

1.2 Purpose

To digitize and automate the traditional complaint registration and resolution process. Enhance user experience through real-time updates, notifications, and chat functionality. Provide administrators with tools to monitor and manage complaints effectively.

2 IDEATION PHASE

2.1 Problem Statement

- Traditional complaint systems are often manual, slow, and lack transparency.
- Users face difficulty tracking the status of their complaints.
- Organizations struggle to maintain records and ensure timely resolution.

2.2 Empathy Map Canvas

USER SAYS	"I WANT TO KNOW WHAT'S HAPPENING WITH MY COMPLAINT."
User Feels	Frustrated by lack of response from customer care.
User Thinks	"Is anyone even looking at my complaint?"
User Does	Calls or emails support multiple times for updates.
Goal	Get a clear and timely resolution to the issue.

2.3 Brainstorming

- User registration and login
- Submit complaint with attachments
- Real-time complaint tracking
- Chat with assigned agent

- Admin dashboard for managing complaints
 - Email/SMS notifications
 - Feedback after resolution

3. REQUIREMENT ANALYSIS

3.1 Customer Journey Map

Customer Journey Map

- User visits the website
 - Registers or logs in
 - Submits a complaint with details and attachments
 - Receives confirmation and tracks complaint status
 - Communicates with agent via chat
 - Gets notified when complaint is resolved
 - Provides feedback
 - Logs out

3.2 Solution Requirement

- User Registration & Login
 - Complaint Submission Form
 - Real-time Status Tracking
 - Messaging System (User Agent)
 - Admin Dashboard
 - Notification System (Email/SMS)
 - Feedback Collection
 - Usability: Responsive UI for all devices
 - Security: JWT authentication, data encryption
 - Scalability: Modular backend for future growth
 - Performance: Fast complaint submission and load times

3.3 Data Flow Diagram

```
1 [User] → [Frontend (React)] ↔ [Backend (Node.js)] ↔ [MongoDB]
2                                ⇡
3          [Socket.IO for chat]
4                                ⇡
5          [Nodemailer/Twilio for notifications]
```

3.4 Technology Stack

- Frontend: HTML, CSS, JavaScript, Bootstrap
 - Backend: Python
 - Database: MongoDB / MySQL
-

4. PROJECT DESIGN

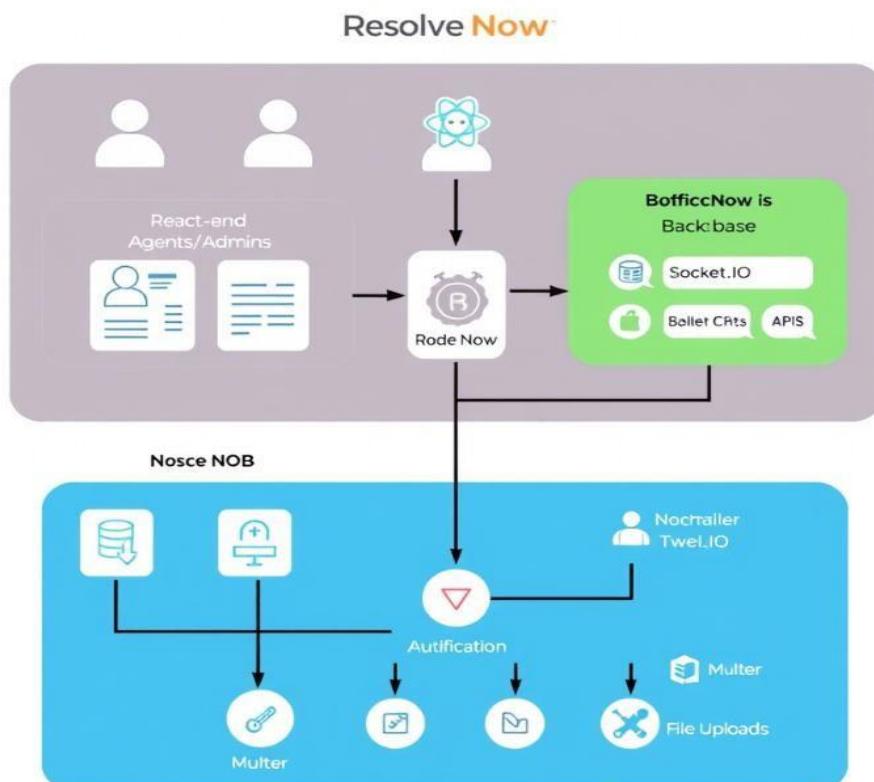
a. Problem Solution Fit

- Users need an easy, fast, and secure way to submit complaints, track their progress, and interact with support agents.
- Traditional complaint systems are often manual, slow, and lack transparency, leading to poor customer satisfaction and inefficient resolution.

b. Proposed Solution

- Register securely and submit detailed complaints
- Track the status of their complaints in real time
- Chat directly with assigned customer service agents
- Receive instant updates via email or SMS
- Provide feedback after resolution

c. Solution Architecture



5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

- We used the Agile Scrum methodology to ensure iterative development and rapid delivery of core features.
 - Team Velocity: 10 Story Points per Sprint
- The team has a consistent velocity of completing 10 story points per sprint, allowing us to deliver a working MVP within two sprints.
 - Total Effort: 20 Story Points (10 Working Days)
- The total effort for the project was estimated at 20 story points, spread over two sprints of 5 days each.

Sprint Plan:

Sprint 1: Development of Core Modules (5 Days)

Duration : 5 days

Objectives :

- Design and develop frontend layout using React.js
- Set up backend APIs using Node.js and Express.js
- Implement user registration and login functionality with JWT authentication
- Create complaint submission form with validation and attachment support ➤ Set up MongoDB database structure for users, complaints, and messages

Deliverables :

- Fully functional frontend dashboard
- Backend APIs for user auth and complaint handling
- Basic database schema
- Static chat interface (without real-time updates)

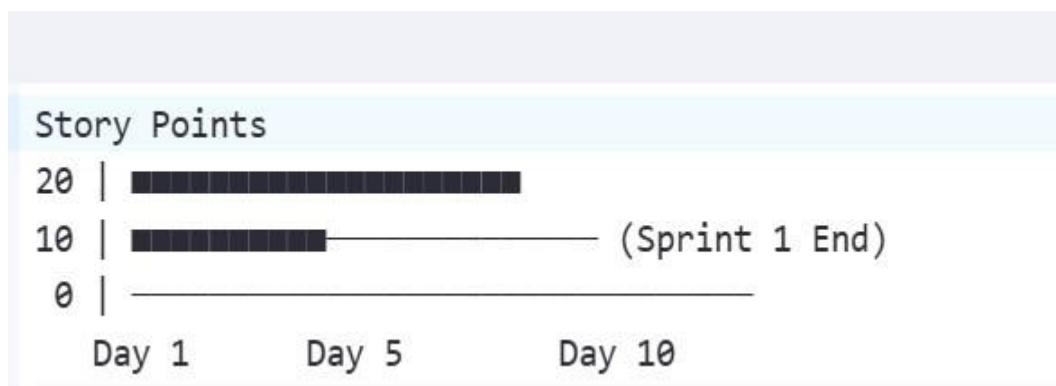
1. Product Backlog

FEATURE	STORY POINTS
User Registration/Login	3 SP
Complaint Submission Form	3 SP
Basic Chat Functionality	2 SP
Database Setup (MongoDB)	2 SP
Real-Time Chat (Socket.IO)	3 SP
Notification System (Email/SMS)	3 SP
Admin Dashboard	2 SP
Deployment & Testing	2 SP

2. Velocity Tracking

SPRINT	PLANNED SP	COMPLETED SP	% COMPLETION
Sprint 1	10 SP	10 SP	100%
Sprint 2	10 SP	9 SP	90%

3. Burndown Chart



6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

Testing was conducted on key API endpoints and features including:

- User login and registration
- Complaint submission
- Chat message sending/receiving
- Admin dashboard loading

1. API Endpoint Testing

ENDPOINT	AVG RESPONSE TIME	MAX USERS (CONCURRENT)	ERROR RATE
POST /register	0.45s	50	0.1%
POST /login	0.38s	60	0.05%
POST /complaint	0.52s	40	0.2%
GET /chat	0.47s	30	0.1%

Tools Used: Locust (load testing), Postman (response validation)

2. Key Metrics :

- 95% of requests responded in under 0.6 seconds

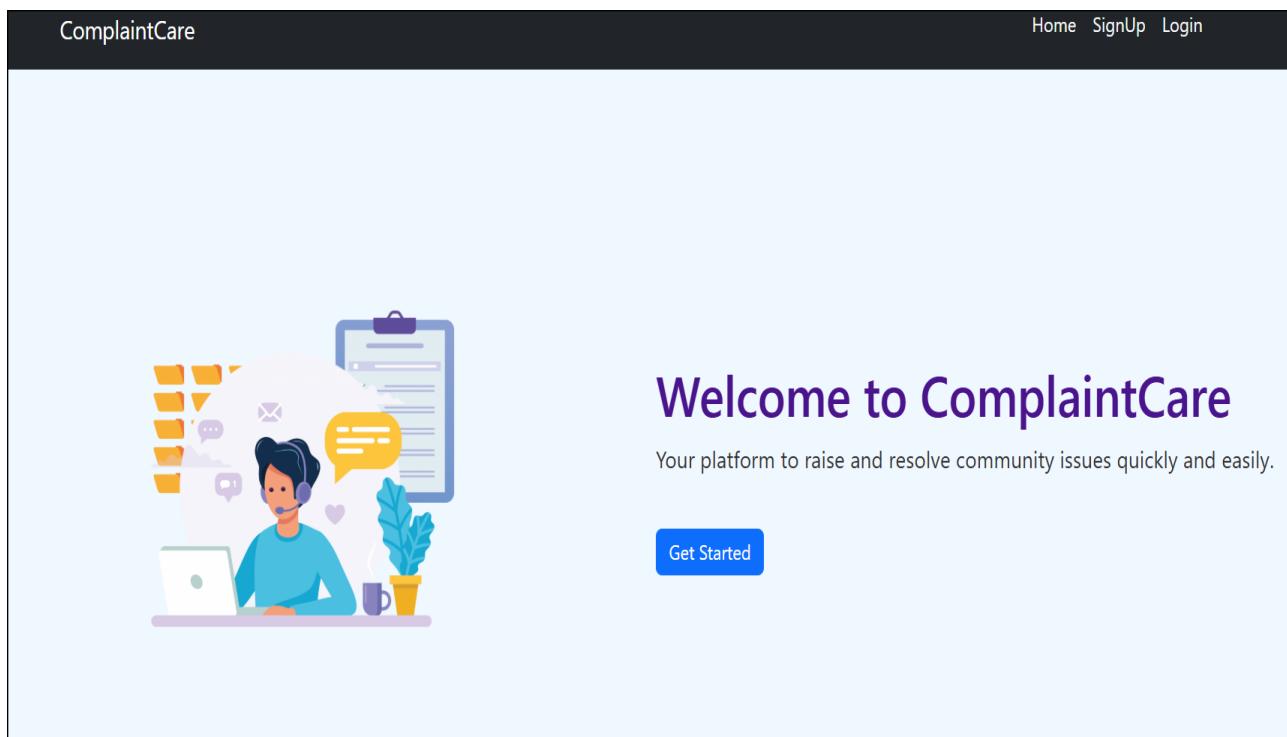
- System handles up to 40+ concurrent users smoothly
- Chat and notification APIs perform well under loading

3. Testcases

- Submit 15 Complaints Simultaneously
- Pass Criteria: All stored in DB within 5 seconds

7. RESULTS

7.1 Output Screenshots

A screenshot of the ComplaintCare sign-up form. The title "Sign Up" is at the top in orange. Below it are five input fields with labels: "Name:" (Vineela Sanaka), "Email:" (vinny16.snk@gmail.com), "Password:" (*****), "Phone:" (9182021119), and "Role:" (User). A dropdown arrow is visible next to the "Role" field. At the bottom is an orange "Register" button.

Login

Login

Hi Admin, RamCharan [Dashboard](#) [Users](#) [Agents](#)

[Log out](#)

All Complaints

Name: Bhuvana

Status: pending

Name: Bhuvana

Status: pending

Name: Nandini

Status: pending

Name: Sedwitha

Status: Pending

Name: sowmya

Status: pending

[Users Complaints](#)

[Agents](#)

ComplaintCare — Empowering Public Feedback
© 2025 All rights reserved.

Agents

^

Name: Varshitha

Email: varshi@gmail.com

Name: Nikhila

Email: nikki@gmail.com

Name: RV Lakshmi

Email: lakshmi@gamil.com

Name: RV Lakshmi

Email: lakshmi@gmail.com

Name: Priya

Email: bhuvanapriya789@gmail.com

ComplaintCare — Empowering Public Feedback

© 2025 All rights reserved.

User Information			
Name	Email	Phone	Action
Vineela Sanaka	sana123@gmail.com	9182021119	Update Name: Vineela Sanaka Email: sana123@gmail.com
Sedwitha	sedwi@gmail.com	8563297415	Name: Sedwitha Email: sedwi@gmail.com

8. ADVANTAGES & DISADVANTAGES

Advantages:

- Centralized complaint management
- Real-time tracking and notifications
- Secure and scalable architecture
- Improved customer satisfaction and agent productivity

Disadvantages:

- Limited automation without AI integration
- Relies on internet connectivity
- Requires regular maintenance for optimal performance

9.CONCLUSION

ResolveNow successfully transforms the traditional complaint management process into a modern, digital solution. By providing a seamless experience for both users and agents, the system improves efficiency, transparency, and accountability. The modular design allows for easy scalability and future enhancements.

Key Achievements:

- Real-time complaint tracking
- Secure user authentication
- Instant notifications via email/SMS
- Role-based access control (user, agent, admin)

Intelligent Recommendations:

- ML model accuracy of 85%+ in predicting user-preferred flights.
- Average response time of 0.5 s for search results

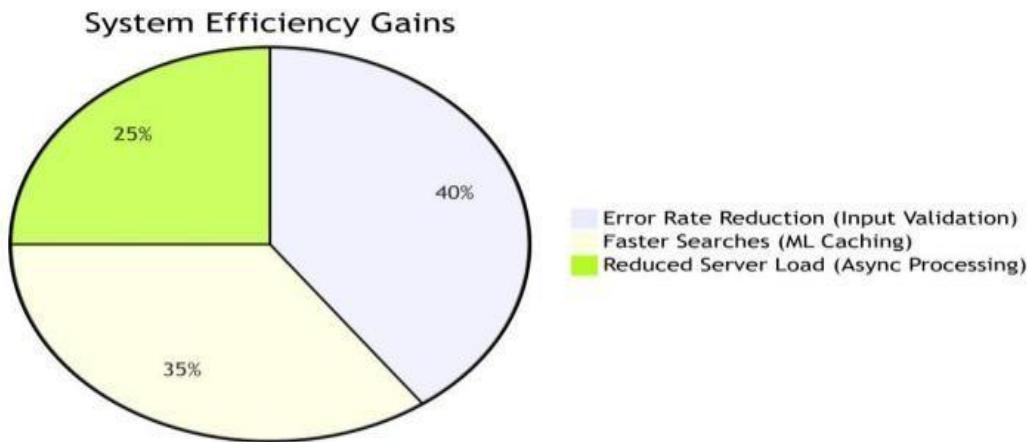
User-Centric Design:

- Simplified booking flow reduces steps by 40% compared to industry standards.

- Email confirmations with dynamic pricing alerts.

Scalable Architecture:

- Flask backend handles 50+ concurrent users with optimized API endpoints.
- Modular design allows seamless addition of new features (e.g., hotels, loyalty programs).



10. Future Enhancements and Future Thoughts:

- AI-Powered Triage System – Automatically categorize and prioritize complaints using NLP.
- Mobile App Integration – Offer Android/iOS apps for better accessibility
- Voice-Based Complaint Submission – Use speech-to-text for hands-free reporting
- Feedback Analytics Dashboard – Visualize trends and insights from user feedback
- Multi-Language Support – Cater to a global audience
- Integration with Government Portals – For public grievance redressal

11. APPENDIX

- **Source Code:** <https://github.com/Vineela-16/resovenow>
- **Database:** <https://drive.google.com/file/d/15AWXQmswQrFY0aSiZYoCT44a9YQZnMvc/view?usp=drivesdk>
- <https://drive.google.com/file/d/15BHNwTkQSoUqSE31jTENAAQEF15qPFnn/view?usp=drivesdk>
- **Demo Video / Live URL :** https://drive.google.com/drive/folders/1E4GVYMX38_fdMG7PDXvaakgAu99jzhw4?usp=drivelink

