**Project Report**: **ResolveNow**

TEAM ID: LTVIP2025TMID55000

# 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 rolebased actions. The system ensures transparency, security, and efficiency in resolving customer issues while maintaining regulatory compliance.

## 1.2 Purpose

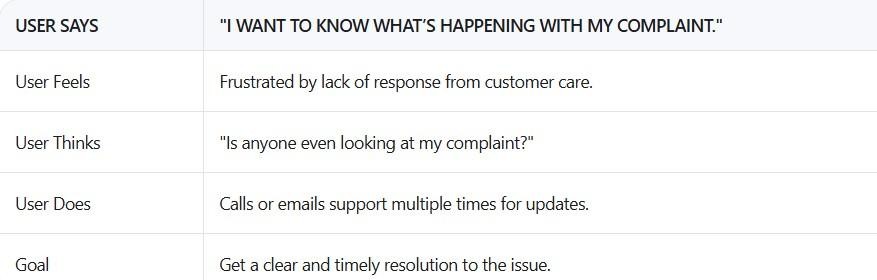
To digitize and automate the traditional complaint registration and resolution process.Enhance user experience through real-time updates, notifications, and chat functionality.Provide administrators with tools to monitor and manage complaints effectively.

# 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



## 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



## 3.4 Technology Stack

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

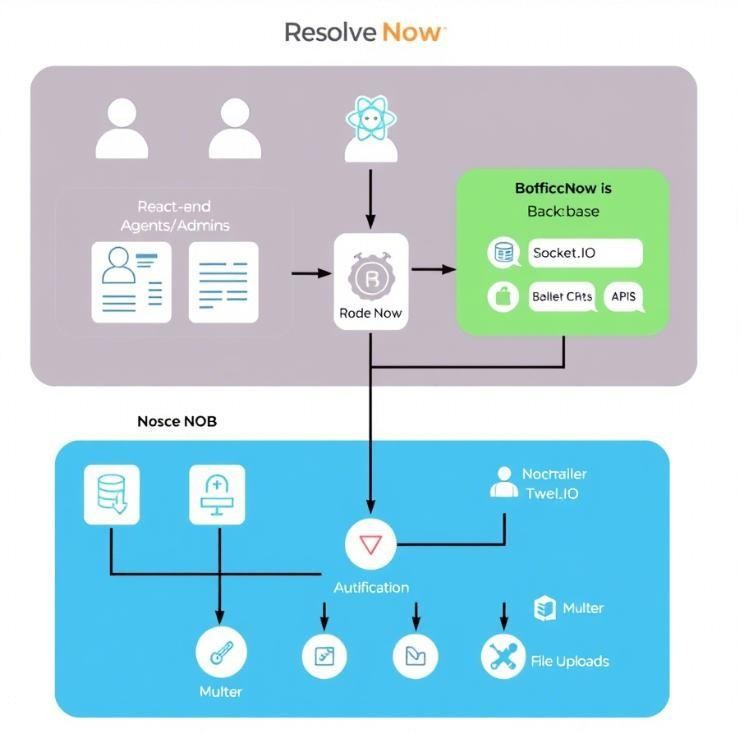
# 4. PROJECT DESIGN

## a. Problem Solution Fit

o Users need an easy, fast, and secure way to submit complaints, track their progress, and interact with support agents. o 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 s SCHEDULING

## 5.1Project 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

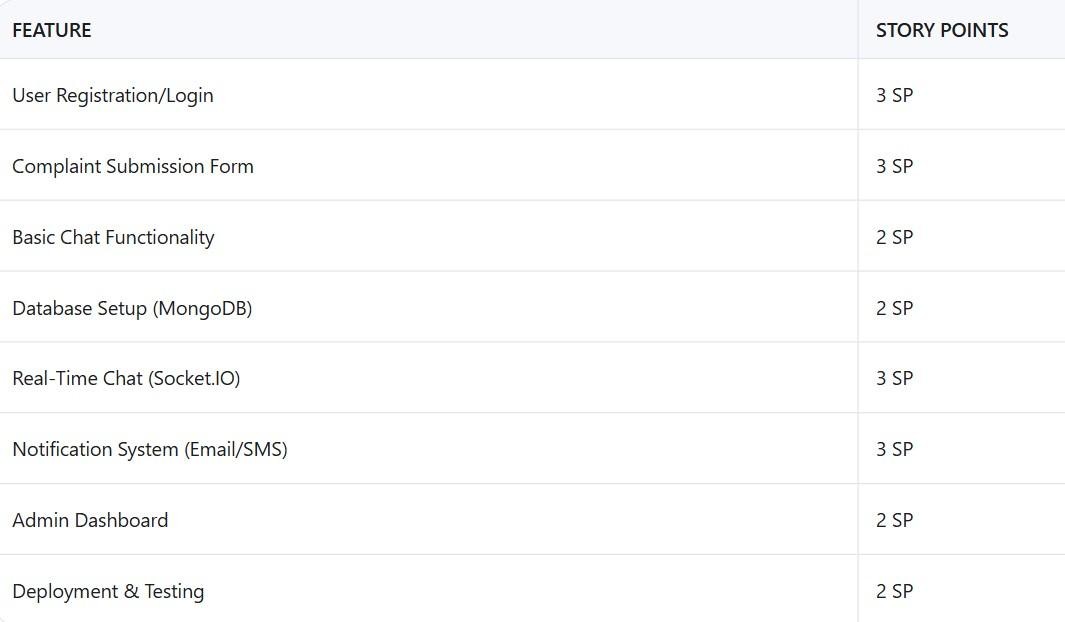
Objectives:

* 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)

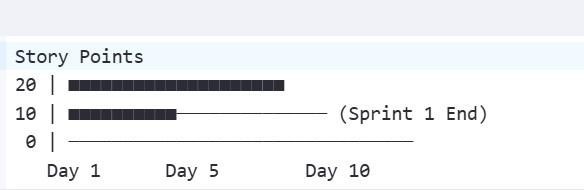
### 1. Product Backlog



**2.Velocity Tracking**



#### 3. Burndown Chart



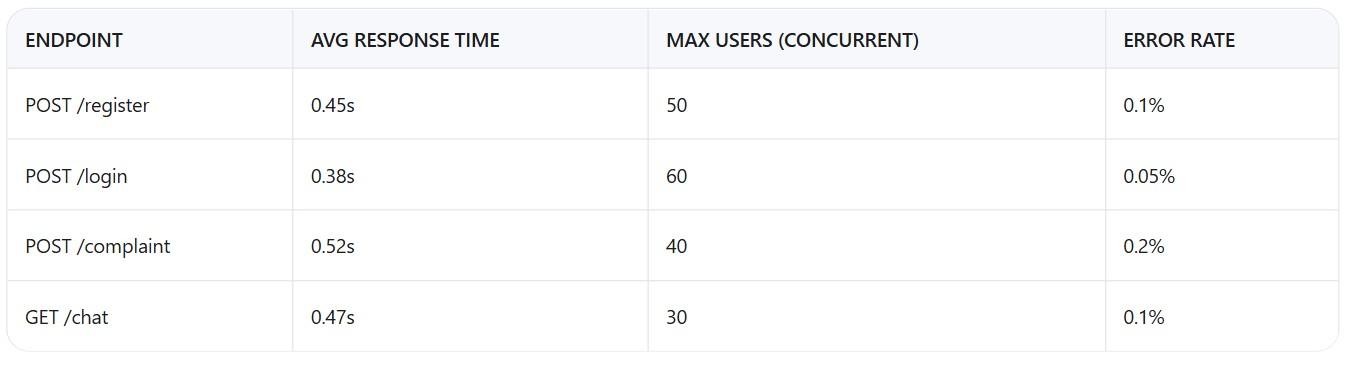
# 6. FUNCTIONAL AND PERFORMANCE TESTING

## 6.1Performance 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



**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

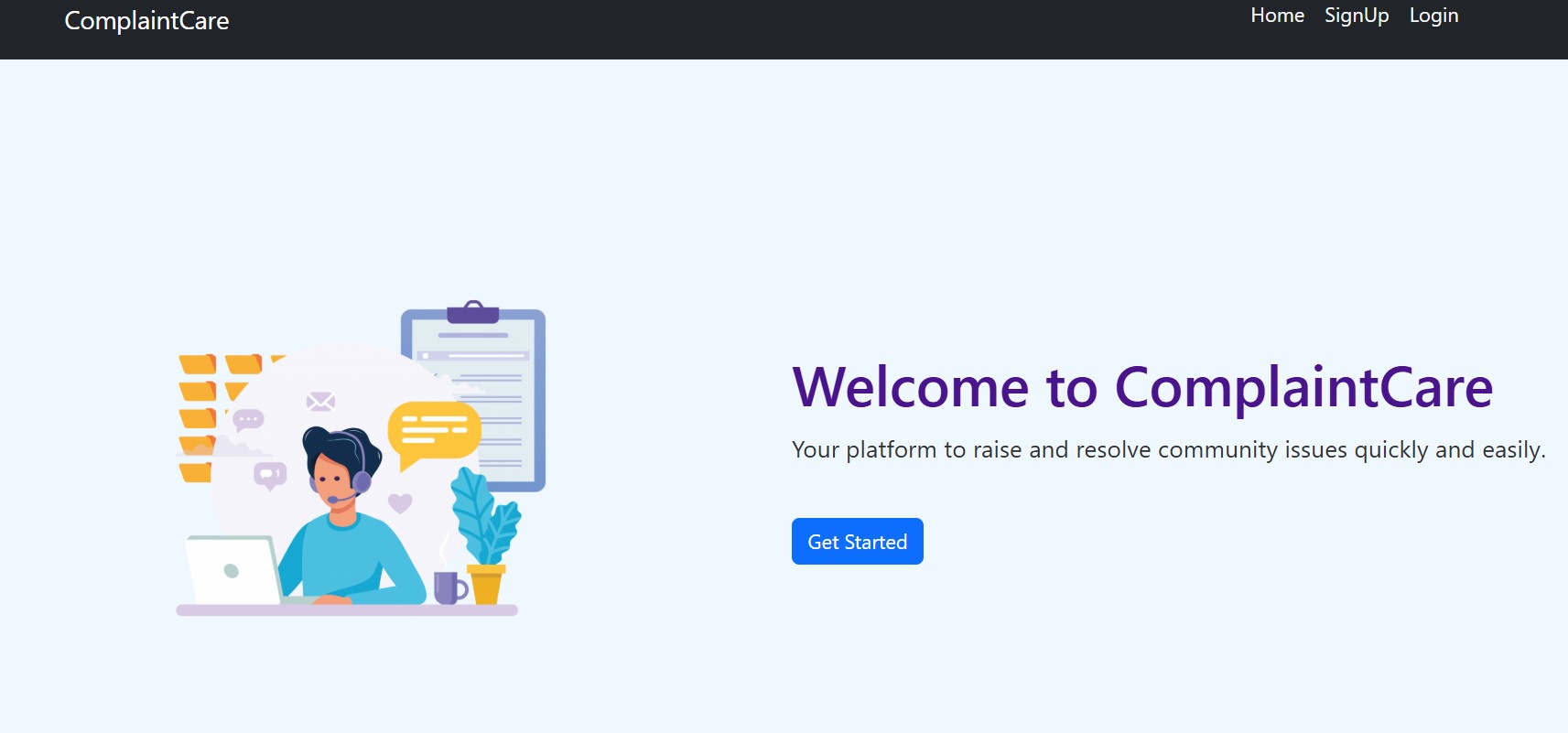
### 3.Testcases

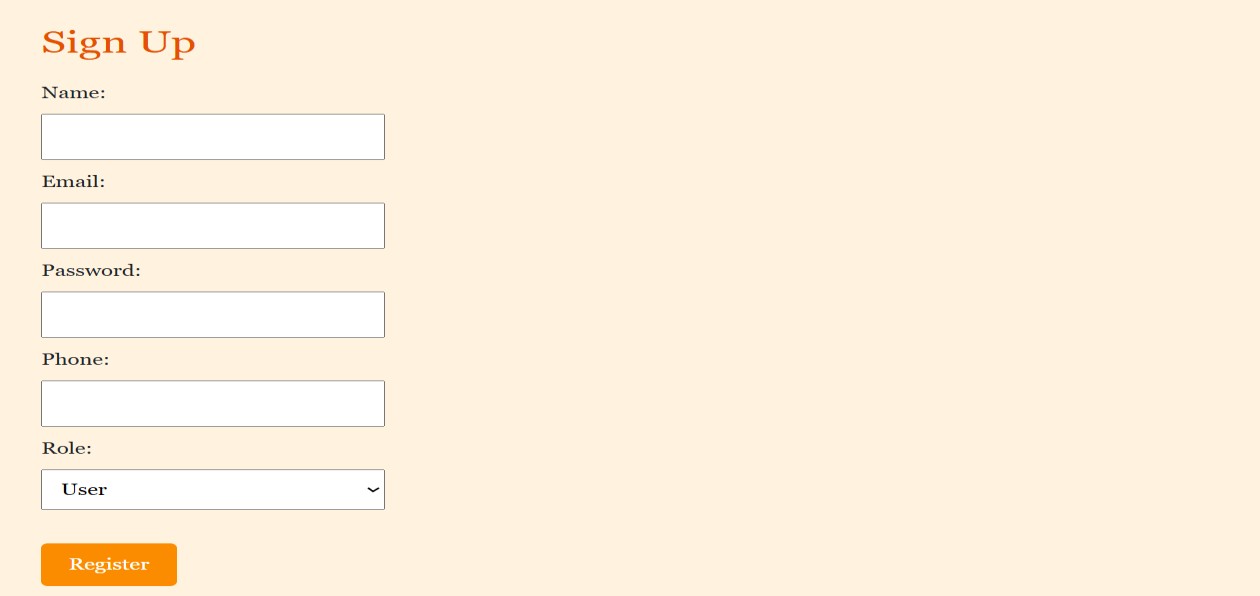
* Submit 15 Complaints Simultaneously
* Pass Criteria: All stored in DB within 5 seconds

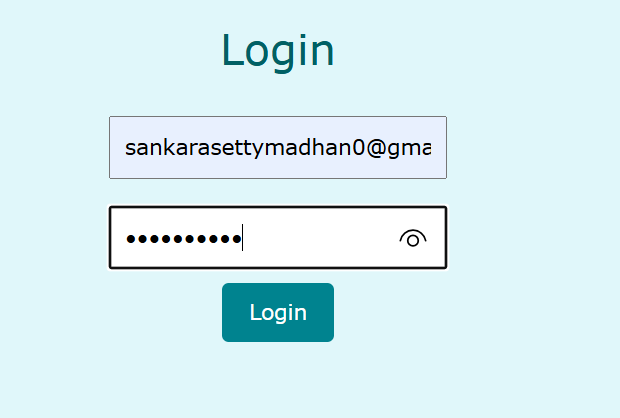
# 7. RESULTS

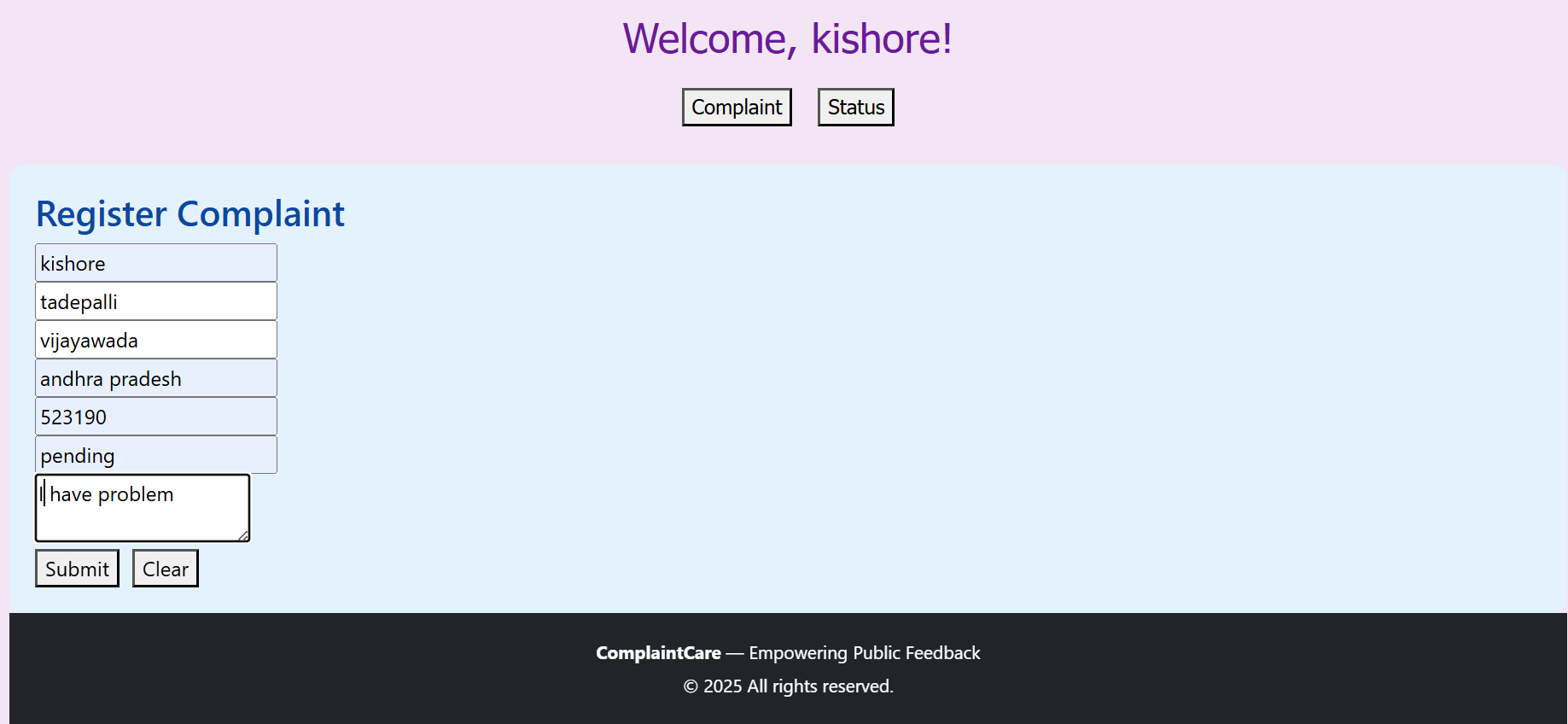
7.1

Output Screenshots





****



## 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

Intelligent Recommendations:

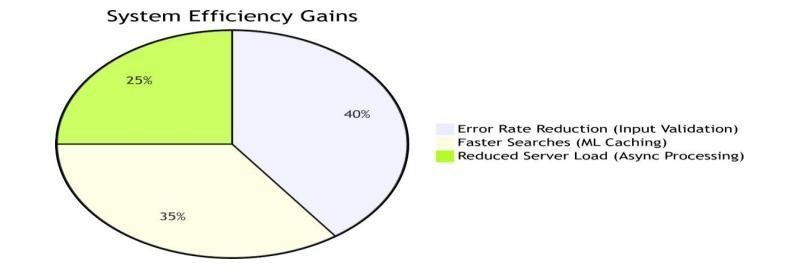
* ML model accuracy of 85%+ in predicting user-preferred flights.
* Average response time of 0.5 s for search results

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).



1. Future Enhancements and Future Thouthts:
   * + 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
2. APPENDIX

**Source Code:** <https://github.com/kishoresaila/online-complaint-project.git>

* + Demo Video / Live URL : <https://drive.google.com/file/d/1RdIMoxP4pVkCWXIi7wW3KAts-1mwQPrE/view?usp=sharing>