A Minor Project Report

on

COMPLAIN TRACK: PRIORITY RESPONSE SYSTEM FOR TRAIN ISSUES

Submitted in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

Under the guidance of

MRS.A.ANANYA M.E

Assistant Professor

Department of Artificial Intelligence and Machine Learning

Submitted by

DEEPANA D	(927622BAD006)
GAYATHRI V	(927622BAD014)
KIRUTHIKA R	(927622BAD030)

DEPARTMENT OF ARTIFICIAL INTELLIGENCE

M. KUMARASAMY COLLEGE OF ENGINEERING,

(Autonomous)

KARUR – 639 113

May 2025

M. KUMARASAMY COLLEGE OF ENGINEERING

(Autonomous Institution affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report, **"COMPLAIN TRACK: PRIORITY RESPONSE SYSTEM FOR TRAIN ISSUES"** is the bonafide work of **DEEPANA D (927622BAD006), GAYATHRI V (927622BAD014), KIRUTHIKA R (927622BAD030)**, who carried out the minor project work during the academic year **2024-2025** under my supervision. Certified further that, to the best of my knowledge, the work reported herein does not form part of any other minor project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Signature

Mrs.A.ANANYA M.E

SUPERVISOR

Department of Artificial Intelligence and Machine Learning,

M.Kumarasamy College of Engineering,

Thalavapalayam,

Karur-639113.

Signature

Dr. A. SELVI M. E., Ph.D.,

Head of the Department,

Department of Artificial Intelligence,

M.Kumarasamy College of Engineering,

Thalavapalayam,

Karur-639113.

INSTITUTION VISION & MISSION

Vision

To emerge as a leader among the top institutions in the field of technical education.

Mission

- Produce smart technocrats with empirical knowledge who can surmount the global challenges.
- Create a diverse, fully-engaged, learner-centric campus environment to provide quality education to the students.
- Maintain mutually beneficial partnerships with our alumni, industry and professional associations.

DEPARTMENT VISION & MISSION

Vision

To excel in education, innovation, and research in Artificial Intelligence and Data Science to fulfil industrial demands and societal expectations.

Mission

M1: To educate future engineers with solid fundamentals, continually improving teaching methods using modern tools.

M2: To collaborate with industry and offer top-notch facilities in a conducive learning environment.

M3: To foster skilled engineers and ethical innovation in AI and Data Science for global recognition and impactful research.

M4: To tackle the societal challenge of producing capable professionals by instilling employability skills and human values.

Programme Educational Objectives (PEOs)

PEO 1: Compete on a global scale for a professional career in Artificial Intelligence and Data Science.

PEO 2: Provide industry-specific solutions for the society with effective communication and ethics.

PEO 3: Hone their professional skills through research and lifelong learning initiatives.

Programme Outcomes (POs):

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO 9: Individual and team work: Function effectively as an individual, and as a member or

leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Programme Specific Outcomes (PSOs):

PSO1: Capable of finding the important factors in large datasets, simplify the data, and improve predictive model accuracy.

PSO2: Capable of analyzing and providing a solution to a given real-world problem by designing an effective program.

ABSTRACT

Complain Track is an intelligent, web-based complaint management system developed to enhance the efficiency and responsiveness of addressing passenger grievances within the railway sector. By moving away from manual tracking, it addresses challenges like lack of transparency and delayed responses through a centralized digital platform. Passengers can securely log in to register complaints, providing essential details and optionally uploading images or documents. The system features an automated categorization and prioritization module, using machine learning algorithms to classify complaints by severity, urgency, and safety concerns, ensuring critical issues are prioritized. Administrators benefit from a dedicated dashboard for efficient complaint management, while real-time notifications and status updates keep passengers informed throughout the resolution process. Ultimately, Complain Track reduces manual workloads, fosters timely resolutions, and improves communication between passengers and staff, contributing to a more reliable, transparent, and responsive railway ecosystem. It supports modernization efforts in public service delivery and helps drive sustainable urban development through efficient transport systems.

KEYWORDS:

Complaint Management System, Railway Services, Web Application, Machine Learning, Automated Prioritization, Real-time Status Updates, Passenger Experience, Issue Resolution, Administrative Dashboard, Transparency.

ABSTRACT WITH POs and PSOs MAPPING

ABSTRACT	POs MAPPED	PSOs MAPPED
Complain Track is an intelligent, web-based complaint management system aimed at improving how passenger grievances are handled in the railway sector. It replaces manual processes with a centralized digital platform, enabling passengers to securely submit complaints and track their status. Using machine learning, the system automatically categorizes and prioritizes complaints based on urgency, severity, and safety. Administrators manage complaints through a dedicated dashboard, while real-time enhance communication and transparency. The system contributes to more efficient public service delivery and supports sustainable urban development through improved transport systems.	PO2(3) PO3(3) PO5(3) PO6(3) PO7(2) PO10(3) PO11(3)	PSO1(3), PSO2(3)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Acronym	Expansions
NLP	Natural Language Processing
SVM	Support Vector Machine
DPDP	Digital Personal Data Protection
OTP	One Time Password
UAT	User Acceptance Test

ACKNOWLEDGEMENT

We gratefully remember our beloved **Founder Chairman, (Late) Thiru. M. Kumarasamy**, whose vision and legacy laid the foundation for our education and inspired us to successfully complete this project.

We extend our sincere thanks to **Dr. K. Ramakrishnan, Chairman**, and **Mr. K. R. Charun Kumar, Joint Secretary**, for providing excellent infrastructure and continuous support throughout our academic journey.

We are privileged to extend our heartfelt thanks to our respected Principal, **Dr. B. S. Murugan, B.Tech., M.Tech., Ph.D.,** for providing us with a conducive environment and constant encouragement to pursue this project work.

We sincerely thank **Dr. A. Selvi, M.E., Ph.D.,** Professor and **Head, Department of Artificial Intelligence**, for her continuous support, valuable guidance, and motivation throughout the course of this project.

Our special thanks and deep sense of appreciation go to our **Project Supervisor, Mrs.A.Ananya,M.E** Assistant Professor, **Department of Artificial Intelligence,** for her exceptional guidance, continuous supervision, constructive suggestions, and unwavering support, all of which have been instrumental in the successful execution of this project.

We would also like to acknowledge **Mrs. A.Ananya., M.E., Assistant Professor, our Class Advisor,** and **Mrs. B. Sribharathi.,M.Sc., M. Phil.,** the **Project Coordinator**, for their constant encouragement and coordination that contributed to the smooth progress and completion of our project work.

We gratefully thank all the **faculty members of the Department of Artificial Intelligence** for their timely assistance, valuable insights, and constant support during various phases of the project.

Finally, we extend our profound gratitude to our **parents and friends** for their encouragement, moral support, and motivation, without which the successful completion of this project would not have been possible.

CHAPTER-1

INTRODUCTION

1.1. INTRODUCTION

In today's fast-paced world, the need for an efficient and responsive railway complaint management system has become increasingly important. The existing manual or semi-automated systems are often insufficient in addressing the concerns of passengers promptly and effectively. Delays in issue resolution, lack of transparency, and limited real-time communication between passengers and railway authorities result in frustration and a lack of trust in the system. Critical issues, especially those involving safety or medical emergencies, are not always prioritized appropriately due to the absence of intelligent mechanisms that can assess the severity of complaints. This project proposes the development of an intelligent and automated complaint management system tailored for the railway sector. The system aims to streamline the complaint registration and resolution process by introducing automated prioritization based on complaint severity, real-time status tracking, and enhanced communication channels between passengers and railway staff. By leveraging modern technologies, the proposed solution will significantly improve passenger satisfaction, accountability, and the overall efficiency of railway operations. Unlike traditional systems that rely heavily on manual intervention, the proposed system will utilize advanced algorithms to categorize and prioritize complaints. For instance, complaints involving health emergencies or security threats will be automatically flagged for immediate attention, ensuring faster response times. The integration of real-time notifications and updates will allow passengers to stay informed about the progress of their complaint, which enhances transparency and trust in the system. Moreover, the system will serve as a centralized platform where all complaints are logged, monitored, and analyzed. This will enable railway authorities to identify recurring issues, evaluate staff responsiveness, and implement long-term improvements. With a focus on both passenger convenience and operational efficiency, the intelligent complaint management system has the potential to transform the way railways handle grievances, ultimately leading to a safer and more reliable transportation experience.

1.2 PROBLEM STATEMENT

In the current railway system, passenger complaints are often managed manually, leading to delayed responses, lack of transparency, and ineffective issue resolution. Passengers frequently face difficulties in tracking the status of their complaints, and critical issues such as safety hazards or medical emergencies may not be prioritized appropriately. The limited communication between railway staff and passengers results in dissatisfaction and inefficiencies in service delivery. There is a need for an intelligent, automated system that can streamline the complaint management process, ensure timely responses based on severity, and provide real-time updates to enhance passenger trust and operational accountability.

1.3 OBJECTIVE

The objective of the Complain Track system is to develop an intelligent, real-time complaint management platform tailored for the railway sector, with the goal of efficiently capturing, categorizing, and prioritizing passenger grievances encountered during train journeys. By leveraging machine learning algorithms, the system can automatically assess the severity and urgency of each complaint ensuring that high-priority issues such as safety hazards, medical emergencies, and technical malfunctions are escalated immediately to the relevant authorities for prompt action. This not only enhances operational responsiveness but also significantly improves the passenger experience by fostering a sense of safety and trust. Furthermore, Complain Track aims to reduce manual processing by automating the entire complaint workflow from registration to resolution thereby minimizing human error and administrative delays.The platform also emphasizes transparency and accountability by providing passengers with real-time updates and notifications regarding the status of their complaints. Overall, the system is designed to modernize traditional complaint handling processes, support efficient decision-making, and contribute to the development of a reliable, technology-driven public transport service.

CHAPTER-2

LITERATURE REVIEW

CHAPTER 2

2.1 LITERATURE SURVEY

TABLE 2.1

FINDINGS ON AI MODELS IN RAILWAY COMPLAINT MANAGEMENT SYSTEMS

S.NO.	Paper Name	Author(s)	Algorithm Used
1.	AI-Based Complaint Management System for Indian Railways	Gupta, R., & Sharma, A.	NLP, Support Vector Machine (SVM)
2.	Real-Time Passenger Feedback System Using Sentiment Analysis	Iyer, N., & Thomas, P.	Sentiment Analysis, Naive Bayes
3.	Intelligent Grievance Prioritization in Public Transport Systems	Verma, K., & Rao, D.	Decision Trees, Rule-Based Prioritization
4.	IoT and AI for Emergency Response in Smart Rail Networks	Khan, A., & Mehta, S.	IoT Integration, Recurrent Neural Networks (RNN)
5.	Automated Helpdesk System Using Machine Learning for Transport Services	Banerjee, S., & Nair, V.	Random Forest, Clustering for Issue Categorization

The reviewed literature indicates a growing trend towards automation in managing public transport complaints. Gupta and Sharma demonstrated how NLP and SVM can be applied to automatically classify railway complaints into predefined categories.Verma and Rao introduced a priority-based model that utilizes decision trees for fast resolution of critical issues. Khan and Mehta's work highlighted the benefits of combining IoT with AI for managing emergencies in real-time through predictive alerts. Banerjee and Nair's system used clustering and random forest algorithms to optimize ticket classification and response.

TABLE 2.2

TECHNOLOGIES USED IN INTELLIGENT COMPLAINT MANAGEMENT SYSTEMS

S.NO.	Paper Name	Author(s)	Algorithm Used
1.	AI-Powered Real-Time Issue Tracking in Urban Rail Systems	Malhotra, S., & Dixit, K.	Python, Flask, REST APIs, MongoDB
2.	Smart Railway Complaint and Query Resolution System	Prasad, H., & Kulkarni, M.	Firebase, Android Studio, Google Dialogflow
3.	Voice-Enabled Grievance Redressal in Smart Transit	Rao, S., & Iyer, R.	Speech Recognition, NLP, Twilio API
4.	Cloud-Based Public Complaint Monitoring System	Joshi, M., & Shetty, A.	AWS (Lambda, S3), Node.js, DynamoDB
5.	AI Chatbot for Railway Complaint Automation	Singh, V., & Chaudhary, A.	Rasa NLP Framework, TensorFlow, SQLite

The reviewed literature reveals a growing adoption of advanced technological stacks for building scalable, intelligent complaint management systems in the transport sector. Malhotra and Dixit utilized Python, Flask, and MongoDB to develop a RESTful microservice architecture for real- time issue tracking in urban rail networks. Prasad and Kulkarni created a mobile solution using Firebase, Android Studio, and Dialog flow, allowing passengers to file complaints using voice or text with real-time sync capabilities. Rao and Iyer focused on speech recognition and Twilio API integration to implement voice-enabled grievance reporting, making complaint submission more accessible for differently-abled passengers. Joshi and Shetty developed a cloud-native solution on AWS, using serverless functions, cloud storage, and NoSQL databases to ensure scalability and low-latency responses. Singh and Chaudhary built an AI chatbot using the Rasa framework and TensorFlow, capable of understanding and responding to user queries using natural language and logging complaints in a local database.

TABLE 2.3

SURVEY FOR SOLUTIONS IN INTELLIGENT COMPLAINT MANAGEMENT

S.NO.	Paper Name	Author(s)	Algorithm Used
1.	Intelligent Complaint Prioritization Model for Railway Networks	Desai, N., & Mishra, L.	Priority assignment based on severity scores using AI ranking
2.	A Unified Dashboard for Smart Transport Complaint Analytics	Kumar, R., & Singh, P.	Centralized web dashboard with real-time updates and status logs
3.	Emergency Detection System in Railway Services Using Machine Intelligence	Batra, A., & Joshi, T.	AI-based emergency classifier using temporal and location data
4.	Automated Passenger Grievance Redressal Using NLP and Knowledge Graphs	Sharma, S., & Dubey, V.	NLP-based complaint classification with knowledge graph integration
5.	A Multi-Tiered Alert System for Railway Complaint Escalation	Meena, K., & Rao, V.	Escalation mechanism using rule-based filters and ML escalation bot

The reviewed studies offer solution-driven frameworks to enhance grievance handling in railway systems. Desai and Mishra proposed an AI-based priority model, assigning severity scores to complaints based on content and keywords, enabling faster responses to high-risk issues. Kumar and Singh introduced a centralized dashboard that visualizes complaint metrics, issue trends, and resolution times, promoting accountability among staff. Batra and Joshi developed a machine intelligence system capable of detecting emergencies using time and location patterns, which is critical for safety-focused transport systems. Sharma and Dubey used natural language processing combined with knowledge graphs to automatically understand the complaint context and map it to actionable resolutions.

CHAPTER - 3

FESABILITY STUDY

3. Feasibility Study

The feasibility study evaluates whether the Complain Track **s**ystem can be practically and successfully implemented. It considers five essential aspects: technical, operational feasibility.

3.1 Technical Feasibility

The Complain Track system is technically feasible due to its foundation on modern, stable, and scalable web technologies. It is a web-based application that integrates a machine learning model to classify and prioritize passenger complaints. The frontend uses React for building responsive user interfaces, Vite for fast development and optimized builds, and Tailwind CSS for modern, responsive design. The backend is developed with Node.js and Express.js**,** which provide a fast, non-blocking environment suitable for high-performance web applications. The system processes complaints in real time and assigns priority levels based on severity. Firebase handles data storage and real-time updates, ensuring that users and administrators are immediately informed of complaint status. Nodemailer is used for secure email-based OTP authentication, while role-based access control ensures data protection and controlled access. The system requires only a device with internet access and no special hardware, making it highly accessible, cost-effective, and easy to deploy.

3.2 Operational Feasibility

Operationally, Complain Track is designed for ease of use by both passengers and railway staff. Its intuitive interface allows users to quickly register, log in, and submit complaints without needing technical knowledge. Machine learning automates the classification and prioritization of issues, reducing the burden on support staff and enabling faster response to critical problems like safety hazards or medical emergencies. Real-time updates and role-based dashboards help administrators manage complaints efficiently. The system enhances transparency, accountability, and user satisfaction, which are essential for public-facing services like railways. Since it integrates smoothly with existing workflows and communication processes, the system is fully operable within the current railway infrastructure.

CHAPTER-4

PROJECT METHODOLOGY

4.1 Methodology

The Complain Track system will focusing on iterative development and continuous feedback. The project will be divided into sprints, each lasting two weeks, with each sprint delivering specific features such as complaint submission for complaint prioritization, and real-time tracking. The development will proceed in several phases: initially requirement analysis will be conducted to gather the needs of stakeholders, followed by system design where wireframes and mockups of the user interface will be created. Frontend development will utilize React**,** Vite, and Tailwind CSS to create a fast and responsive user interface. The backend will be built using Node.js and Express.js to handle API requests, real-time updates, and data processing. Machine learning will be integrated to automatically prioritize complaints based on their urgency, ensuring high-priority issues are addressed first. Testing will be continuous throughout development, including User Acceptance Testing (UAT) at the end of each sprint. Once the system is ready, it will be deployed using Firebase Hosting**,** and Firebase's real-time database will manage complaint tracking.

4.2 FEATURES

1. Complaint Submission: Allows passengers to easily submit complaints through a user-friendly form.
2. Automated Categorization: Complaints are automatically categorized based on their content (e.g., safety, delays).
3. Real-Time Tracking: Passengers and admins can track the status of complaints in real time.
4. User Authentication: Secure login system with OTP verification via Nodemailer to ensure only verified users can access the system.
5. Admin Dashboard: Admins can view, manage, and resolve complaints, assign priorities, and track progress.
6. Role-Based Access: Different interfaces for passengers and admins, providing specific functionalities to each user type.
7. Real-Time Notifications: Automated email notifications for complaint status updates and important actions.
8. Responsive Design: Optimized for both desktop and mobile devices using React and Tailwind CSS.

4.3 ARCHITECTURE

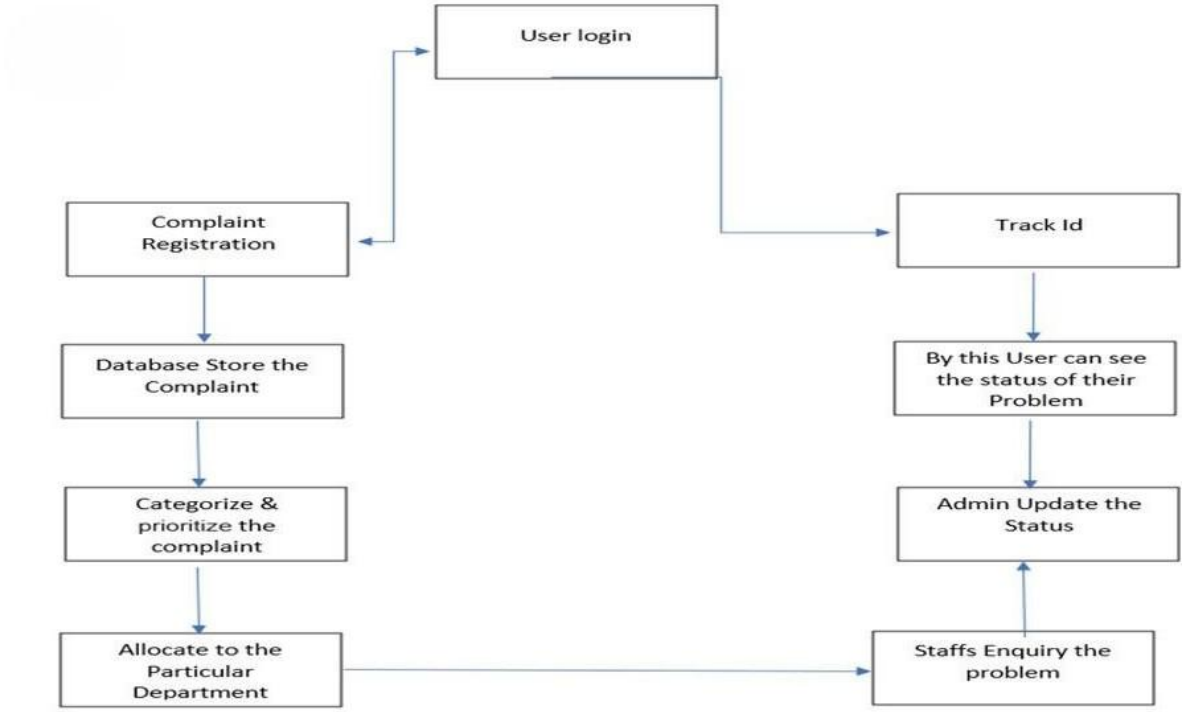


FIGURE 4.3 COMPLAIN TRACK ARCHITECTURE

CHAPTER-5

RESULT

SOURCE CODE:
Admin Dashboard

```
import React, { useState, useEffect } from 'react';
import { getFirestore, collection, getDocs } from 'firebase/firestore';
import { initializeApp } from 'firebase/app';
import '../assets/admin.css'; // Import your custom CSS for styling
import {
    AreaChart,
    Area,
    XAxis,
    YAxis,
    CartesianGrid,
    Tooltip,
    ResponsiveContainer,
  } from 'recharts';

const firebaseConfig = {
  apiKey: "AIzaSyAth9C58p4ues2bKlJSx4pZE47SaxQqauo",
  authDomain: "jpro-firebase.firebaseapp.com",
  projectId: "jpro-firebase",
  storageBucket: "jpro-firebase.firebasestorage.app",
  messagingSenderId: "747381925733",
  appId: "1:747381925733:web:372a903a009406635b638c",
  measurementId: "G-WHT8YLLE30",
};

const app = initializeApp(firebaseConfig);
const db = getFirestore(app);

const AdminDashboard = () => {
  const [activeTab, setActiveTab] = useState('Users');
  const [users, setUsers] = useState([]);
  const [enquiries, setEnquiries] = useState([]);
  const [majorEnquiries, setMajorEnquiries] = useState([]);
  const [analytics, setAnalytics] = useState({});
  const [analyticsData, setAnalyticsData] = useState([]);
  const [analyticsOverview, setAnalyticsOverview] = useState({});

  useEffect(() => {
    if (activeTab === 'Users') {
```

```
      fetchUsers();
    } else if (activeTab === 'Enquiries' || activeTab === 'Major Enquiries' || activeTab === 'Analytics') {
      fetchEnquiries();
    }
  }, [activeTab]);

  const fetchUsers = async () => {
    try {
      const response = await fetch('http://localhost:5000/api/getUsers');
      const data = await response.json();
      setUsers(data);
    } catch (error) {
      console.error('Error fetching users:', error);
    }
  };

  const fetchEnquiries = async () => {
    try {
      const querySnapshot = await getDocs(collection(db, 'userForms'));
      const enquiryData = querySnapshot.docs.map(doc => ({ id: doc.id, ...doc.data() }));

      const processedEnquiries = enquiryData.map(enquiry => {
        const priorityScore = calculatePriority(enquiry);
        return {
          ...enquiry,
          priorityScore,
          priorityClass:
            priorityScore > 75 ? 'High' : priorityScore > 50 ? 'Medium' : 'Low',
        };
      });

      setEnquiries(processedEnquiries);
      const highPriorityEnquiries = processedEnquiries.filter(e => e.priorityClass === 'High');
      setMajorEnquiries(highPriorityEnquiries);

      if (activeTab === 'Analytics') {
        generateAnalytics(processedEnquiries);
      }
    } catch (error) {
      console.error('Error fetching enquiries:', error);
    }
```

```
  };

  const calculatePriority = enquiry => {
    const typeWeights = {
      'Medical Assistance': 100,
      Security: 80,
      'Divyangjan Facilities': 60,
      'Coach - Maintenance': 50,
      Miscellaneous: 30,
    };
    const typeWeight = typeWeights[enquiry.type] || 10;
    const urgencyScore = enquiry.date ? new Date(enquiry.date).getTime() : Date.now();

    return typeWeight + Math.max(0, 100 - (Date.now() - urgencyScore) / (1000 * 60 * 60 *
24));
  };

  const generateAnalytics = processedEnquiries => {
    const typeCounts = processedEnquiries.reduce((acc, enquiry) => {
      acc[enquiry.type] = (acc[enquiry.type] || 0) + 1;
      return acc;
    }, {});

    const highPriorityCount = processedEnquiries.filter(e => e.priorityClass === 'High').length;
    const mediumPriorityCount = processedEnquiries.filter(e => e.priorityClass ===
'Medium').length;
    const lowPriorityCount = processedEnquiries.filter(e => e.priorityClass === 'Low').length;

    setAnalyticsData(
      Object.keys(typeCounts).map(type => ({
      name: type,
        High: processedEnquiries.filter(e => e.type === type && e.priorityClass ===
'High').length,
        Medium: processedEnquiries.filter(e => e.type === type && e.priorityClass ===
'Medium').length,
        Low: processedEnquiries.filter(e => e.type === type && e.priorityClass ===
'Low').length,
      }))
    );

    setAnalyticsOverview({
      highPriorityCount,
```

```
    mediumPriorityCount,
    lowPriorityCount,
  });
};

const renderAnalytics = () => (
  <div>
    <h3>Analytics Overview</h3>
    <p><strong>Total High Priority Enquiries:</strong>
{analyticsOverview.highPriorityCount}</p>
    <p><strong>Total Medium Priority Enquiries:</strong>
{analyticsOverview.mediumPriorityCount}</p>
    <p><strong>Total Low Priority Enquiries:</strong>
{analyticsOverview.lowPriorityCount}</p>

    <h4>Enquiry Type and Priority Levels</h4>
    <ResponsiveContainer width="100%" height={400}>
      <AreaChart data={analyticsData}>
        <CartesianGrid strokeDasharray="3 3" />
        <XAxis dataKey="name" />
        <YAxis />
        <Tooltip />
        <Area type="monotone" dataKey="High" stackId="1" stroke="#ff4d4d" fill="#ffcccc"
/>
        <Area type="monotone" dataKey="Medium" stackId="1" stroke="#ffd633"
fill="#fff0b3" />
        <Area type="monotone" dataKey="Low" stackId="1" stroke="#4d94ff" fill="#b3d9ff"
/>
      </AreaChart>
    </ResponsiveContainer>
  </div>
);

const renderTable = () => {
  if (activeTab === 'Users') {
  return (
    <table className="data-table">
      <thead>
        <tr>
          <th>UID</th>
          <th>Email</th>
                : 'green',
```

```
          }}
        >
          <td>{enquiry.trackId}</td>
          <td>{enquiry.email}</td>
          <td>{enquiry.type}</td>
          <td>{enquiry.subType}</td>
          <td>{enquiry.date}</td>
          <td>{enquiry.priorityClass}</td>
          <td>{enquiry.status || 'In Progress'}</td>
          <td>{enquiry.description}</td>
        </tr>
      ))}
    </tbody>
  </table>
 );
} else if (activeTab === 'Major Enquiries') {
 return (
  <table className="data-table">
    <thead>
      <tr>
        <th>Track ID</th>
        <th>Email</th>
        <th>Type</th>
        <th>Priority</th>
        <th>Status</th>
        <th>Description</th>
      </tr>
    </thead>
    <tbody>
      {majorEnquiries.map(enquiry => (
        <tr key={enquiry.id}>
          <td>{enquiry.trackId}</td>
          <td>{enquiry.email}</td>
          <td>{enquiry.type}</td>
          <td>{enquiry.priorityClass}</td>
          <td>{enquiry.status || 'In Progress'}</td>
          <td>{enquiry.description}</td>
        </tr>
      ))}
    </tbody>
```

```
    </div>
  );
};

export default AdminDashboard;
```

Track Enquiry

```
import React, { useState } from 'react';
import { getFirestore, collection, query, where, getDocs } from 'firebase/firestore';
import { initializeApp } from 'firebase/app';

const firebaseConfig = {
  apiKey: "AIzaSyAth9C58p4ues2bKlJSx4pZE47SaxQqauo",
  authDomain: "jpro-firebase.firebaseapp.com",
  projectId: "jpro-firebase",
  storageBucket: "jpro-firebase.firebasestorage.app",
  messagingSenderId: "747381925733",
  appId: "1:747381925733:web:372a903a009406635b638c",
  measurementId: "G-WHT8YLLE30",
};

const app = initializeApp(firebaseConfig);
const db = getFirestore(app);

const TrackEnquiry = () => {
  const [trackId, setTrackId] = useState('');
  const [enquiry, setEnquiry] = useState(null);
  const [error, setError] = useState('');

  const handleTrackEnquiry = async () => {
    if (!trackId.startsWith('TRACKID')) {
    setError('Invalid TRACKID format.');
    setEnquiry(null);
      return;
    }

    try {
      const q = query(collection(db, 'userForms'), where('trackId', '==', trackId));
      const querySnapshot = await getDocs(q);
```

```
      if (!querySnapshot.empty) {
        const result = querySnapshot.docs[0].data();
        setEnquiry(result);
        setError('');
      } else {
        setError('No enquiry found for this TRACKID.');
        setEnquiry(null);
      }
    } catch (err) {
      console.error('Error fetching enquiry:', err);
      setError('An error occurred while fetching the enquiry.');
      setEnquiry(null);
    }
  };

  return (
    <div className="flex flex-col items-center min-h-screen bg-gradient-to-r from-gray-200
to-gray-400 text-gray-800 p-6">
      <header className="text-center mt-10">
        <h1 className="text-3xl font-bold mb-6">Track Your Enquiry</h1>
      </header>

      <div className="w-full max-w-md bg-white shadow-lg p-6 rounded-lg">
        <input
          type="text"
          value={trackId}
          onChange={(e) => setTrackId(e.target.value)}
          placeholder="Enter TRACKID"
          className="w-full border border-gray-300 rounded-md p-3 mb-4 focus:ring-2
focus:ring-blue-500"
        />
        <button
          onClick={handleTrackEnquiry}
          className="w-full bg-blue-500 text-white py-3 rounded-md hover:bg-blue-600
transition"
        >
          Track
        </button>
        {error && <p className="text-red-500 mt-4">{error}</p>}
      </div>

      {enquiry && (
```

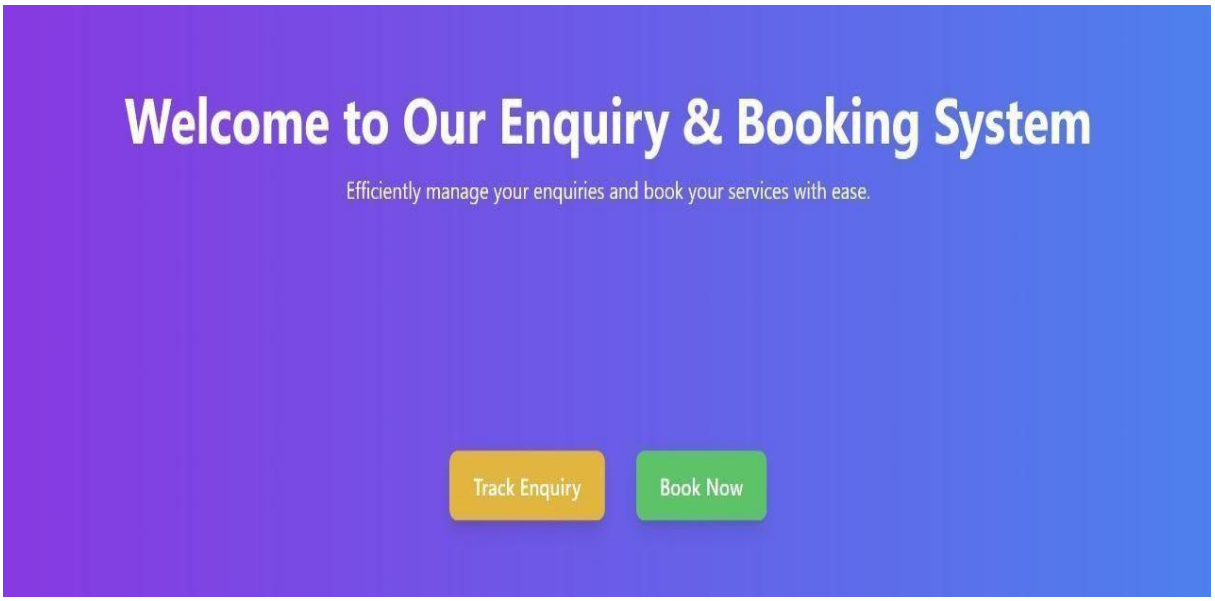
```
        <div className="mt-8 bg-white shadow-lg p-6 rounded-lg">
          <h2 className="text-xl font-bold mb-4">Enquiry Details</h2>
          <p><strong>Type:</strong> {enquiry.type}</p>
          <p>
            <strong>Status:</strong>{' '}
            {enquiry.status ? enquiry.status : <span className="text-yellow-500">In
Progress</span>}
          </p>
          <p><strong>Description:</strong> {enquiry.description}</p>
          {enquiry.status === 'Fixed' && (
            <p className="text-green-500 font-bold mt-4">This enquiry is resolved.</p>
          )}
        </div>
      )}
    </div>
  );
};

export default TrackEnquiry;
```

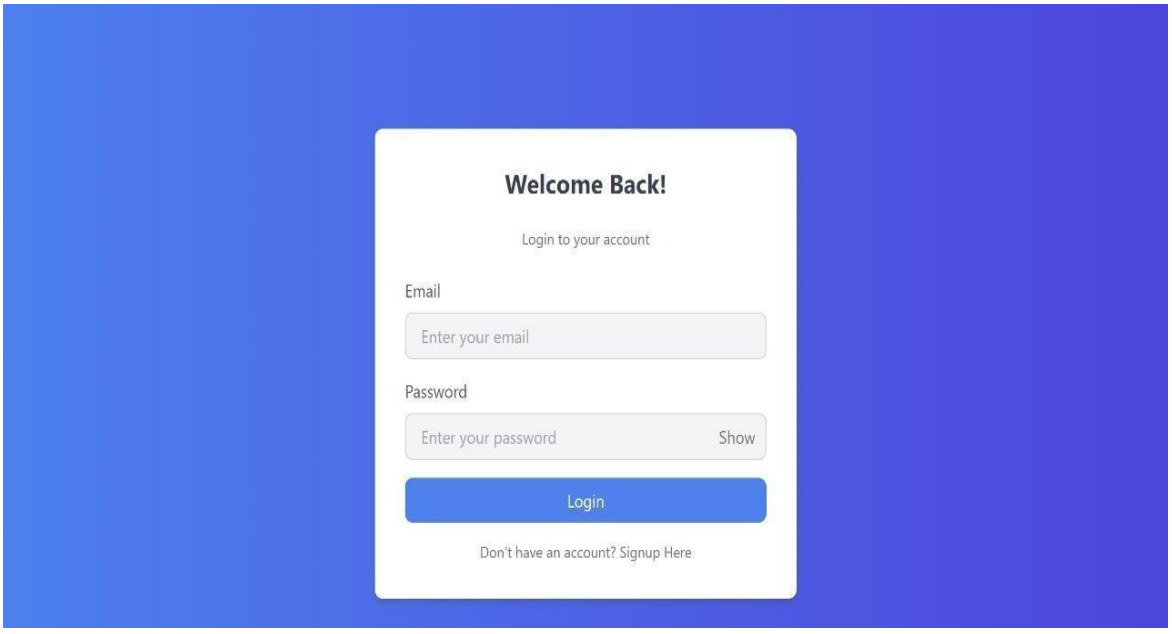
Firebase

```
{
  "firestore": {
    "rules": "firestore.rules",
    "indexes": "firestore.indexes.json"
  }
}
{
  "indexes": [],
  "fieldOverrides": []
}
{
  "projects": {
    "default": "jpro-firebase"
  }
}
```

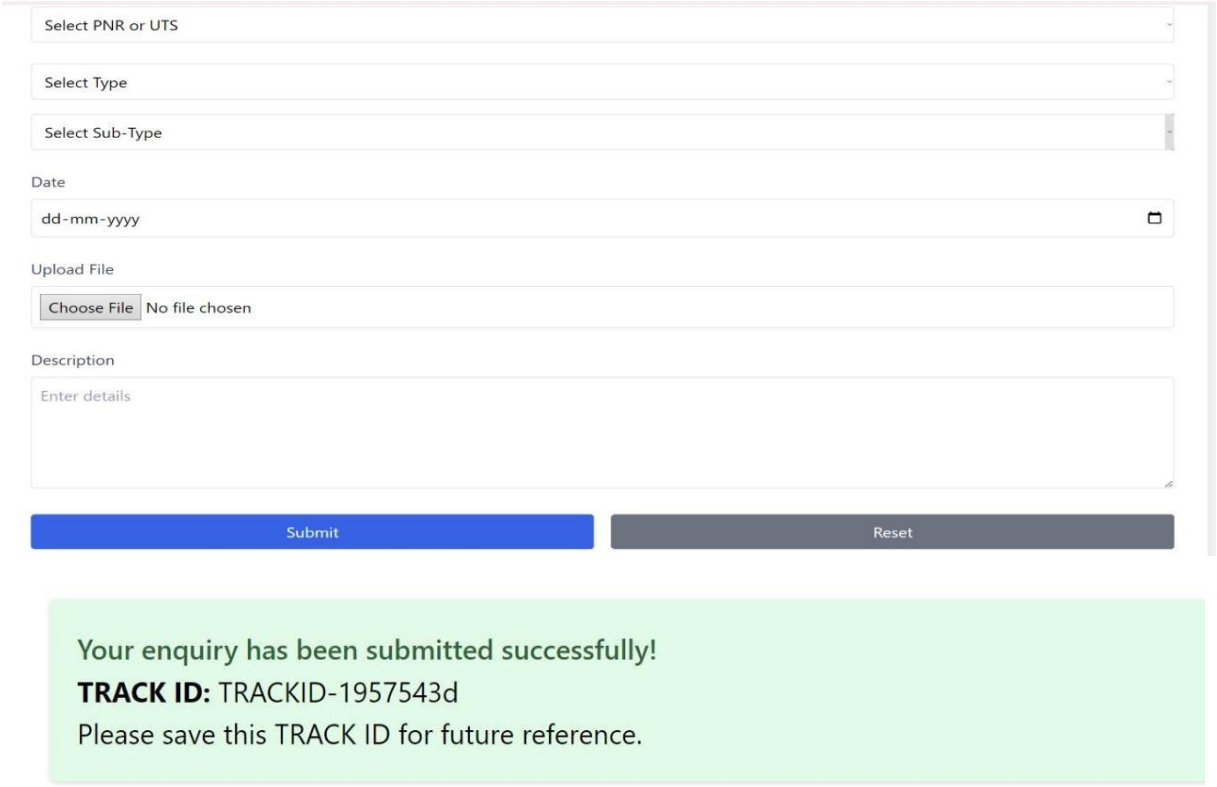
5.1 LANDING PAGE



5.2 LOGIN PAGE



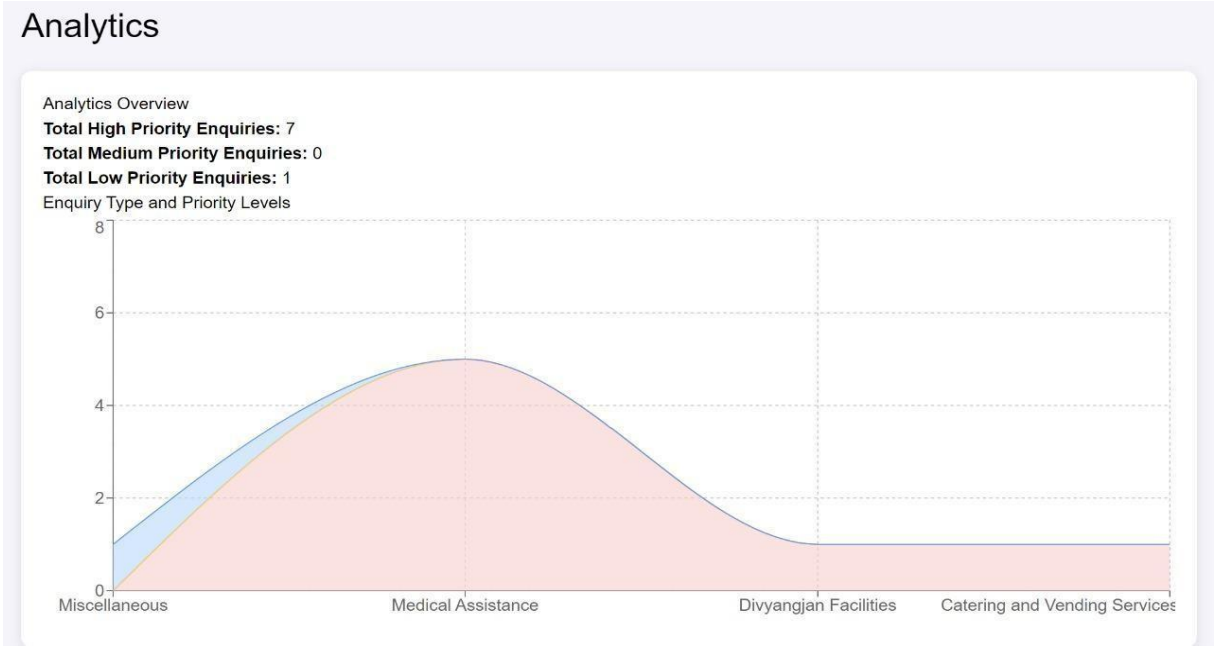
5.3 COMPLAIN REGISTRATION PAGE



5.4 STAFF DASHBOARD

5.5 ADMIN DASHBOARD

5.6 ANALYSIS



CHAPTER-6

CONCLUSION

CONCLUSION

In conclusion, the Complain Track system offers a comprehensive, intelligent solution to revolutionize the way railway passenger complaints are managed. By leveraging modern technologies such as machine learning for complaint categorization and prioritization, the system ensures that critical issues are promptly addressed based on their severity and urgency. The real-time notifications and status updates provide passengers with full transparency, improving communication and ensuring that they remain informed throughout the complaint resolution process. This not only enhances the user experience but also fosters trust and satisfaction among passengers, contributing to a more reliable and responsive railway service. Moreover, the system's powerful analytics and reporting tools enable railway authorities to track complaint trends, monitor response times, and assess the overall effectiveness of complaint handling. This data-driven approach empowers administrators to make informed decisions and implement continuous improvements to service delivery. Overall, Complain Track represents a significant step toward modernizing railway operations, increasing operational efficiency, and ensuring a higher standard of service for passengers. By enhancing complaint management, it aligns with the goals of improving public transport systems, ensuring passenger safety, and promoting sustainable and accountable governance in the railway sector.

CHAPTER-7

REFERENCES

REFERENCE

1. Roy, T. S.; Namratha, N.; Naga Malleswari, T. Y. J. (2023). Voice E-Mail Synced with Gmail for Visually Impaired. Third International Conference on Artificial Intelligence and Smart Energy (ICAIS), Coimbatore, India, pp. 802–807. DOI: 10.1109/ICAIS56108.2023.10073879

2. Samant, Rutuja; Yadav, Gaurang; Poojary, Diksha; Tandlekar, Guruprasad; Ahirrao, Manisha. (2022). Indian Railways Tweets Classification System using Naive Bayes Classifier. IJIRT, Volume 8, Issue 11, ISSN: 2349-6002.

3. Jamwal, G.; Vasukidevi, T. N.; Malleswari, T.; Vijayakumar, L. C. S.; Reddy, A. S. A. L. G.; Gupta. (2022). Real-Time Conversion of American Sign Language to Text with Emotion using Machine Learning. I-SMAC 2022, Nepal. DOI: 10.1109/I-SMAC55078.2022.9987362

4. Mithra, S.; Nagamalleswari, T. (2022). An Analysis of Deep Learning Models for Dry Land Farming Applications. Applied Geomatics. DOI: 10.1007/s12518-022-00425-3

5. Almiya, Kirandeep. (2022). Automatic Ticket Classification Using NLP. GitHub. Retrieved from: https://github.com/kirandalmiya/AutomaticTicketClassification_NLP

6. Singh, Anubhav; Gupta, Meenu; Saxena, Anmol; Ahmed, Shakeel. (2021). Multi-class Railway Complaints Categorization Using Neural Networks: RailNeural. Journal of Rail Transport Planning and Management, Vol. 20, Article 100265. DOI: 10.1016/j.jrtpm.2021.100265

7. Drishti IAS. (2021). AI Powered Grievance Management Application. Drishti IAS Daily News Analysis. Retrieved from: https://www.drishtiias.com/daily-news-analysis/ai-powered-grievance-management-application

8. Kumar, Sachin & Nezhurina, Marina. (2020). Sentiment Analysis on Tweets for Trains Using Machine Learning. International Journal of Innovative Research in Science, Engineering and Technology, Vol. 9, Issue 5.

9. Shrivastva, Chitresh. (2017). Indian Railway and Social Media – The Way Forward to Improving Business to Consumer Interaction. International Journal of Engineering Development and Research, Vol. 5, Issue 4.

10. Lisanti, Giuseppe; Karaman, Svebor; Pezzatini, Daniele; Del Bimbo, Alberto. (2015). A Multi-Camera Image Processing and Visualization System for Train Safety Assessment. arXiv:1507.07815. Retrieved from: https://arxiv.org/abs/1507.07815