GRIEVEASE: MAKING GRIEVANCE SUBMISSION AND TRACKING EASY WHILE MAPPING GOVERNMENT BENEFITS.

FINAL REPORT

Submitted by

SHREE HARSHINI S	**2116220701269**
SRI KIRUPA P	**2116220701286**
SRIRAMAN K	**2116220701288**

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE

ANNA UNIVERSITY, CHENNAI

MAY 2025

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

BONAFIDE CERTIFICATE

Certified for this Project titled **" Grievease: making grievance submission and tracking easy while mapping government benefits. "** is the bonafide work of **"SHREE HARSHINI S (2116220701269), SRI KIRUPA P (2116220701286) & SRIRAMAN K (2116220701288)"** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE
Dr. P. Kumar., M.E., Ph.D.,
HEAD OF THE DEPARTMENT
Professor
Department of Computer Science and Engineering,
Rajalakshmi Engineering College,
Chennai - 602 105.

SIGNATURE
Dr. SenthilPandi., M.E., Ph.D.,
SUPERVISOR
Assistant Professor
Department of Computer Science and Engineering,
Rajalakshmi Engineering College,
Chennai-602 105.

Submitted to Project Viva-Voce Examination held on _____

Internal Examiner **External Examine**

ABSTRACT

For a number of citizens it is even hard to decipher and to take advantage of the right government programmes: information is scattered, eligibility rules are complicated, and there is no support to find the right programme. Traditionally, searching has meant a person needing to go through the laborious process of searching for the information available at their local government office. To address this issue, we have created a Government Scheme Recommendation System which is an online information system that contains intelligent user profiles, data driven recommendations and grievance management capabilities.

The frontend design is through a user portal and a chatbot UI to allow a natural flow and conversation for the citizens to enter their details, ask questions and lodge a grievance. The backend, which is an API server, pulls together profiles and scheme data from a MySQL User and Schemes database. It pulls all this data together to build a unified XGBoost based machine learning model to predict and recommend schemes to really facilitate the discovery of schemes in real-time. In one configuration, there is a robust grievance management module, which is supported by Status Tracker and Grievance Handler, assessment tools such as Fairlearn and AIF360 to help assure a transparent and fair process of handling and resolution of grievances. This process is already organized much like a structured pathway of a software development lifecycle (SDLC); requirements gathering, system design, development, testing and deployment. It gives a more personalized, transparent and data driven experience to citizens to find out about schemes and lodge grievances over the existing system of manual processes. The objective is to render the government welfare program more democratic and citizens can contribute to it and make sure it serves people in a manner that ameliorates its services through automation.

ACKNOWLEDGMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavor to put forth this report. Our sincere thanks to our Chairman **Mr. S. MEGANATHAN, B.E, F.I.E.**, our Vice Chairman **Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) THANGAM MEGANATHAN**, **Ph.D.,** for providing us with the requisite infrastructure and sincere endeavoring in educating us in their premier institution.

Our sincere thanks to **Dr. S.N. MURUGESAN, M.E., Ph.D.**, our beloved Principal for his kind support and facilities provided to complete our work in time. We express our sincere thanks to **Dr. P. KUMAR, M.E., Ph.D.,** Professor and Head of the Department of Computer Science and Engineering for his guidance and encouragement throughout the project work. We convey our sincere and deepest gratitude to our internal guides **Dr. SENTHILPANDI, M.E.,Ph.D.,** and **Ms.U.FARJANAH, M.Tech.,** We are very glad to thank our Project Coordinator, **Dr. SENTHILPANDI** Assistant Professor Department of Computer Science and Engineering for his useful tips during our review to build our project.

SHREE HARSHINI S (2116220701269)

SRI KIRUPA P (2116220701286)

SRIRAMAN K (2116220701288)

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	**iii**
	ACKNOWLEDGMENT	**iv**
	LIST OF TABLES	**8**
	LIST OF FIGURES	**9**
	LIST OF ABBREVIATIONS	**10**
1.	**INTRODUCTION**	**11**
	1.1 GENERAL	11
	1.2 OBJECTIVES	12
	1.3 EXISTING SYSTEM	12
2.	**LITERATURE SURVEY**	**14**
3.	**PROPOSED SYSTEM**	**21**
	3.1 GENERAL	21
	3.2 ACTIVITY DIAGRAM	22
	3.3 DESIGN OF THE ENTIRE SYSTEM	23
	3.4 DEVELOPMENTAL ENVIRONMENT	24
	3.4.1 HARDWARE REQUIREMENT	24
	3.4.2 SOFTWARE REQUIREMENT	25
	3.5 STATISTICAL ANALYSIS	26

4. MODULE DESCRIPTION 28

4.1 SYSTEM ARCHITECTURE 28

4.1.1 HIGH-LEVEL SYSTEM OVERVIEW 29

4.1.2 USER INTERACTION DIAGRAM 29

4.1.3 BACKEND INFRASTRUCTURE 31

4.2 DATA HANDLING 31

4.2.1 DATA INGESTION MODULE 31

4.2.2 SCHEME RECOMMENDATION USING XGBOOST 32

4.2.3 DATA PREPROCESSING & FEATURE ENGINEERING 33

4.3 ADMIN PANEL & GRIEVANCE RESOLUTION 34

4.3.1 ADMIN PANEL OVERVIEW 34

4.3.2 GRIEVANCE RESOLUTION FLOW 35

4.4 USER INTERFACE & EXPERIENCE 35

4.4.1 FRONTEND 36

4.4.2 USER AUTHENTICATION & ROLE MANAGEMENT 36

4.5 PERFORMANCE & OPTIMIZATION 37

4.5.1 RESPONSE TIME & MATCHING EFFICIENCY 37

4.5.2 ACCURACY & SCHEME RELEVANCE 38

	4.6 DEPLOYMENT & STORAGE	38
	4.6.1 LOCAL VS REMOTE DEPLOYMENT	38
	4.6.2 DATA STORAGE & MANAGEMENT	39
5.	**IMPLEMENTATIONS AND RESULTS**	**40**
	5.1 IMPLEMENTATION	40
	5.2 OUTPUT SCREENSHOTS	41
6.	**CONCLUSION AND FUTURE ENHANCEMENT**	**48**
	6.1 CONCLUSION	48
	6.2 FUTURE ENHANCEMENT	49
	REFERENCES	**50**

LIST OF TABLES

TABLE NO	TITLE	PAGE NO
3.1	HARDWARE REQUIREMENTS	25
3.2	SOFTWARE REQUIREMENTS	26
3.3	COMPARISON OF FEATURES	27

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
3.1	ACTIVITY DIAGRAM	23
3.2	COMPARISON GRAPH	27
4.1	SYSTEM ARCHITECTURE DIAGRAM	28
4.2	USER INTERACTION DIAGRAM	30
4.3	SCHEME MATCHING AND XGBOOST	33
4.4	DATA PREPROCESSING AND FEATURE ENGINEERING	34
5.1	WEBSITE DESIGN	41
5.2	FEATURE	42
5.3	USER LOGIN	42
5.4	USER SIGN IN	43
5.5	SCHEME LOOKUP	43
5.6	USER DETAILS COLLECTION	44
5.7	ML MATCHING MODEL	45
5.8	ADMIN DASHBOARD	46
5.9	GRIEVANCE MANAGEMENT	47

LIST OF ABBREVIATIONS

ABBREVIATION	FULL FORM
AI	ARTIFICIAL INTELLIGENCE
AIF360	AI FAIRNESS 360
G2R	GRIEVANCE REDRESSAL
DBMS	DATABASE MANAGEMENT SYSTEM
MYSQL	MY STRUCTURED QUERY LANGUAGE
PHP	HYPERTEXT PREPROCESSOR
UI	USER INTERFACE
UX	USER EXPERIENCE
XGBOOST	EXTREME GRADIENT BOOSTING
ML	MACHINE LEARNING
XAI	EXPLAINABLE AI

CHAPTER 1
1. INTRODUCTION

1.1 GENERAL

Government welfare schemes serve as a fundamental tool to support all citizens and promote inclusive development opportunities. However, Locating appropriate government schemes and their accessibility presents challenges because information exists across various sources and eligibility requirements are complex and there is no personal support. Generally, Citizens have traditionally depended on manual methods which include word of mouth and physical visits to government offices that prove to be tedious and inaccurate and delayed.This research will discuss the design of a web-based recommendation system for government schemes will be developed through this research which uses React frontend together with MySQL database backend and XGBoost machine learning to provide intelligent matching between citizens and eligible schemes based on their profiles. This research will be situated in the research that takes place within e-governance and artificial intelligence frameworks for public services and explores standard functionality through user profile management and real-time scheme recommendations and grievance submission and fairness mediation during grievance processes. The system aims to reduce citizen interactions with welfare services through a system that accepts personal information and enables users to ask domain specific questions through a chatbot and receive personalized scheme recommendations. The system features a grievance management module which demonstrates transparency in grievance handling and enables real-time status tracking and fairness audit of grievance management processes using various tools (Fairlearn and AIF360). This research study aims to produce a citizen & data-driven, transparent solution that improves the reachability, and fairness in using government schemes.

1.2 OBJECTIVE

The primary objective of the proposed system is to develop an intelligent, citizen-centric web application that simplifies access to government welfare schemes through personalized, AI-powered recommendations. By leveraging modern technologies such as a React-based frontend, a MySQL database backend, and the XGBoost machine learning algorithm, the system aims to match citizens with the most relevant schemes based on their socio-economic profiles and personal needs. The key objectives of this platform focus on enhancing accessibility, transparency, and user experience in public welfare delivery. At its core, the system integrates a conversational chatbot interface to facilitate natural language interactions, enabling users to inquire about eligible schemes, understand application processes, and receive personalized suggestions in real time. A robust user profile management module allows individuals to securely input and update personal data, which is used to generate accurate scheme recommendations. To ensure fairness and accountability, the system incorporates a grievance management module supported by fairness auditing tools such as Fairlearn and AIF360, providing users with real-time status updates and transparent resolution mechanisms. This platform is grounded in the principles of digital governance and AI for public good, aiming to transform traditional, manual welfare access into a seamless, efficient, and data-driven process. Ultimately, the system aspires to foster inclusive development by making government benefits more reachable and equitably distributed among all segments of society.

1.3 EXISTING SYSTEM

In the current scenario, accessing government welfare schemes continues to be a significant challenge for citizens due to a lack of centralized and personalized information delivery. Government schemes are often dispersed across multiple

departments and platforms, each with its own set of eligibility criteria, procedures, and documentation requirements. This fragmented structure makes it difficult for individuals—particularly those from rural or underserved communities—to discover which schemes they are eligible for and how to apply for them. Information is usually made available through static government websites, newspaper notifications, or pamphlets distributed during awareness drives, which do not provide real-time or tailored information. Most citizens still depend on manual methods such as word of mouth, consultations with local officials, or multiple physical visits to government offices to inquire about available schemes. These traditional approaches are not only tedious and time-consuming but also prone to misinformation and misinterpretation, leading to delays, missed opportunities, and a general lack of trust in the system. For the elderly, illiterate, or economically weaker sections, navigating these channels poses a further burden, resulting in underutilization of welfare benefits. Existing digital portals provided by the government often focus on static information delivery and lack interactive features like personalized recommendations or conversational support. As a result, even if a citizen has access to a smartphone or computer, the absence of intuitive interfaces and smart filters limits effective usage. Furthermore, grievance redressal mechanisms where available are slow, bureaucratic, and non-transparent. In summary, the existing system for accessing and managing government welfare schemes is largely inefficient, non-personalized, and inaccessible to a large portion of the population. These shortcomings highlight the urgent need for a smart, AI-enabled, and citizen-centric platform that simplifies scheme discovery, automates eligibility matching, and provides transparency in grievance handling. An intelligent solution can bridge the digital divide, improve scheme awareness, and ensure that benefits reach the right individuals in a timely and efficient manner.

CHAPTER 2
LITERATURE SURVEY

Role of G-Cloud in Citizen-Centric Governance:[1] – This paper explores how G-Cloud promotes e-Governance in India using cloud computing to enhance transparency, efficiency,and accessibility of government services. It emphasizes e-Transparency and accountability via ICT by reducing bureaucratic delays. G-Cloud offers a secure and cost-effective infrastructure for delivering public services, delegating administrative tasks, reducing costs, and minimizing carbon emissions. It provides scalability,security, and better data management. The NeGP proposes cloud adoption to reduce corruption and build public trust, though challenges like low digital literacy and resistance to change persist.The paper concludes that G-Cloud holds transformational potential,subject to strong implementation will.

Smart E-Grievance System for Effective Communication in Smart Cities:[2] – This paper presents a Smart E-Grievance System designed to bridge communication between citizens and authorities in smart cities. Traditional grievance mechanisms are outdated and inefficient. The new system supports complaints through text,image, or video, with auto-location tagging for faster redressal.Authorities can prioritize based on urgency. A “Serve India” module enables volunteer engagement in rural awareness campaigns. The paper uses Hadoop's MapReduce for handling large complaint data, allowing authorities to categorize and act efficiently. Integrating grievances on a single platform promises transparency and timely resolution.

Smart Complaint Redressal System Using Ethereum Blockchain:[3] – This paper proposes using Ethereum blockchain for a transparent, secure, and tamper-proof complaint redressal system.Current systems are slow and allow manipulation. The proposed model stores complaint metadata on the blockchain and content on IPFS,

ensuring immutability and traceability. Users can file complaints with the media, track them in real-time, and even vote on priority issues. Governments can verify and act upon them through a secure login. Smart contracts eliminate intermediaries, enhance trust, and promote citizen engagement and transparency in public grievance systems.

A Citizen-Centric Approach towards Global-scale Smart City Platform:[4]– This paper discusses the ClouT project, a collaborative Europe-Japan initiative that builds a smart city platform by connecting IoT, Internet of People, and Internet of Services via cloud computing. Citizens, developers, and authorities co-create services to improve urban living. The platform collects data from sensors, social media, and services to optimize transportation, energy use, and safety. Real-world trials were conducted in Fujisawa, Santander, Mitaka, and Genova, each addressing unique urban challenges. The project emphasizes user-driven service design, scalability, and real-time data utilization for smarter,inclusive urban governance.

Evaluating Web-Based E-Government Services with a Citizen-Centric Approach:[5] – This paper evaluates e-government websites based on how effectively they serve citizens, not just on technical features. A citizen-centric model is proposed, incorporating user behavior, website design, and task complexity. Testing on the Syracuse City School District website showed that ease of navigation, search performance, and information clarity are key to citizen satisfaction.The study concludes that improving usability and user experience on government portals can enhance public engagement, reduce service costs, and make governance more efficient. Future work will refine the model and apply it across government services.

Measuring the Effectiveness of Government Schemes using Machine Learning Algorithms[6] – This paper uses machine learning algorithms to gauge the effectiveness of government schemes by analyzing public opinion through social

media data. By applying sentiment analysis on tweets regarding schemes like Swachh Bharat Abhiyan, Digital India, and Demonetization using Naïve Bayes and Maximum Entropy algorithms, the study reveals insights into public sentiment. The results show that Maximum Entropy outperforms Naïve Bayes in accuracy and precision. The findings suggest that machine learning can help policymakers assess public reactions and improve future policies in real-time.

Study on the Application of Machine Learning in GovernmentService: Take Consumer Protection Service as an Example[7] – This paper explores the use of machine learning in automating the classification of consumer protection complaints. By utilizing Naïve Bayes and Support Vector Machine (SVM) algorithms,complaints are categorized into areas like real estate, transport, and finance. The study demonstrates that SVM outperforms Naïve Bayes in accuracy, and further enhancement is achieved using a Stacking approach with multiple models. The paper concludes that machine learning can improve government efficiency, making complaint handling faster, more accurate, and transparent, with future research focusing on expanding and improving the system.

An Enterprise Architecture Mapping Approach for Realizing e-Government[8] – This paper discusses how enterprise architecture (EA) frameworks can help governments build effective e-government systems. It examines four EA frameworks—Zachman Framework,TOGAF, Federal Enterprise Architecture (FEA), and Gartner Methodology—by mapping them to e-government services and the software development lifecycle (SDLC). The study suggests a dual mapping approach to assess and implement EA frameworks in government projects. The paper concludes that a customized approach to EA is necessary for better service delivery, cost reduction, and transparency in governance.

Uncovering Concerns of Citizens Through Machine Learning and Social Network Sentiment Analysis[9] – This research applies machine learning to analyze social media data to uncover citizens' concerns in Saskatoon, Canada. By analyzing 114,390 comments from the Saskatoon subreddit, the study identifies key topics such as housing, education, and healthcare using topic modeling techniques like LDA, NMF, and BERTopic. The research applies SiEBERT and VADER models to analyze sentiments, revealing prevalent negative emotions regarding housing and healthcare. The findings aim to assist community leaders in enhancing governance and planning urban projects by using real-time citizen feedback.

Algorithm-Based Handling of Complaints Data from the Usage Phase [10] – This paper proposes an algorithm-based method to optimize the handling of customer complaints in the usage phase of smart products. The authors highlight the inefficiency of the traditional 8D problem-solving method and introduce a novel processing algorithm to automate and standardize the complaint handling process. The paper suggests that the new algorithm can improve the speed and consistency of complaint management, particularly in industries like automotive, where customer feedback is crucial for product improvement.

Influence of the Contact Center Systems Development on Key Performance Indicators [11] – This paper examines how advancements in contact center systems, including automation, AI chatbots, and social media integration, have impacted key performance indicators (KPIs) like call response time, cost, customer satisfaction, and problem resolution. The authors discuss how technology, such as emotion detection systems and predictive analytics, is reshaping contact centers. While privacy concerns and resistance to AI interaction exist, the study concludes that AI will continue to enhance contact center efficiency, reducing operational costs and improving service quality.

Inductive Learning of Dispute Scenarios for Online Resolution of Customer Complaints [12]- This paper proposes a machine learning system for resolving customer complaints online. The system models complaints as graphs, using Nearest Neighbors for classification. The study finds that combining communicative actions with argument structure enhances classification accuracy (78%) over using only communicative actions (64%). This approach reduces handling costs and improves the efficiency of complaint resolution.

Automatic Multilabel Categorization using Learning to Rank Framework for Complaint Text on Bandung Government [13]-This research focuses on improving the LAPOR system in Indonesia by automating complaint categorization. It uses a Learning to Rank (LTR) approach combined with Multilabel Classification (MLC) to prioritize the appropriate agencies for handling complaints. The LambdaMART algorithm outperforms others in predicting secondary agencies, and the study discusses potential improvements like dataset refinement and addressing imbalanced data.

A Scalable Machine Learning Online Service for Big Data Real- Time Analysis [14]-This paper explores a scalable machine learning framework designed for real-time analysis of big data. The authors introduce an online service architecture that processes large datasets while offering high-performance analytics. They discuss challenges in scaling machine learning models and propose solutions to enhance data throughput and minimize latency in big data environments.

Solving Real-Time Traffic Prediction in Smart Cities with Neural Networks [15]-The paper investigates the use of neural networks for real-time traffic prediction in smart cities. It highlights the importance of predicting traffic patterns to optimize transportation and reduce congestion. Using deep learning models, the system

provides high accuracy in forecasting traffic conditions, which can be integrated into smart city traffic management systems to improve urban mobility.

A Comparative Study of Ensemble Learning Methods for Fraud Detection in Financial Transactions [16]-This paper presents a comparative analysis of various ensemble learning methods for fraud detection in financial transactions. The study evaluates techniques such as Random Forest, Gradient Boosting, and AdaBoost. It finds that Gradient Boosting yields the best results in terms of accuracy and precision,providing an efficient solution for detecting fraudulent transactions in real-time.

Integrating Sentiment Analysis and NLP for Social Media Brand Monitoring [17]-This paper discusses the integration of sentiment analysis and Natural Language Processing (NLP) for monitoring brand sentiment on social media. The authors propose a system that analyzes social media posts to gauge public perception of brands. The study shows that sentiment analysis combined with NLP techniques can offer valuable insights into consumer attitudes,enabling businesses to respond quickly to changes in brand.

Adaptive Machine Learning Algorithms for Real-Time Stock Price Prediction[18]-The paper focuses on the application of adaptive machine learning algorithms for real-time stock price prediction. The authors explore various models, including Support Vector Machines and Decision Trees, to predict stock prices based on historical data and real-time market indicators. The results show that adaptive models, which update in real-time, outperform traditional models in predicting market fluctuations.

Predicting Customer Churn Using Data Mining Techniques: A Case Study of Telecommunications [19]-This paper examines the use of data mining techniques for predicting customer churn in the telecommunications industry. By analyzing customer

behavior data, the authors apply models like Logistic Regression, Decision Trees, and Neural Networks to predict churn. The study finds that neural networks provide the highest accuracy, offering telecommunication companies a tool to proactively retain customers at risk of leaving.

Improving Healthcare Decision-Making through Predictive Analytics and Machine Learning [20]-The paper explores the use of predictive analytics and machine learning to improve healthcare decision-making. It demonstrates how models can predict patient outcomes, suggest personalized treatment plans, and identify high-risk patients. The authors discuss the challenges of data quality and integration in healthcare settings, while emphasizing the potential benefits of machine learning in enhancing patient care and operational efficiency.

CHAPTER 3

3. PROPOSED SYSTEM

3.1 GENERAL

India's vast socio-economic landscape is home to a multitude of government welfare schemes designed to uplift different sections of society ranging from laborers to students, women, the elderly, and persons with disabilities. One of the most pressing challenges is the lack of awareness among citizens about which schemes they are eligible for, and this is compounded by the fragmented presentation of schemes across various departments and platforms. The eligibility criteria are often embedded in bureaucratic jargon, and the discovery process involves navigating multiple disconnected portals, resulting in frustration, misinformation, and ultimately abandonment. This is where a personalized Scheme Recommendation System becomes not just useful, but essential. Such a system, when powered by intelligent algorithms, can match individual citizen profiles to suitable welfare programs based on age, income, occupation, geography, and other demographic factors making the process intuitive, targeted, and empowering. The GRIEVEASE project goes a step further by integrating grievance redressal mechanisms into the same platform. Even when citizens manage to access and apply for schemes, they often encounter roadblocks like delayed approvals, rejections without explanation, or lack of support for technical issues that rarely get resolved due to opaque complaint-handling. GRIEVEASE bridges this critical gap by offering a streamlined, AI-enabled portal where users not only receive personalized scheme recommendations but can also register complaints, track their status, and receive resolutions. In an era of digital empowerment, GRIEVEASE stands out as a necessary innovation ensuring that the right benefits reach the right people at the right time, and that citizens' voices are not just heard, but acted upon.

3.2 ACTIVITY DIAGRAM

The activity diagram for the Government Scheme Recommendation System depicts step-by-step interaction between two major actors: the citizen (user) and administrator (admin). The process initiates when a user logs in to the portal and decides either to register a new account or log in through existing credentials. After successful verification, the user goes ahead and fills in personal profile information including age, income, location, and other personal data. From this information, the system provides individualized government scheme suggestions based on pre-defined eligibility rules.Once the user sees the proposed schemes, the user is given an option to raise a grievance in case he/she has a problem or complaint. When the user wants to raise a grievance, he/she completes a form describing his/her problem, and the system generates a distinct grievance ID for monitoring the complaint. The user can then log in whenever he/she wishes to monitor the status of his/her grievance, whether it is pending, under consideration, or settled. At the same time, the admin logs into the system via a secure login. Upon authentication, the admin can see all grievances submitted in a dashboard-like view. From there, the admin can update the status of each grievance according to its progress, keeping users informed during the process of resolution. This flow of activity ensures hassle-free interaction between users and the system while keeping grievance management transparent and accountable.

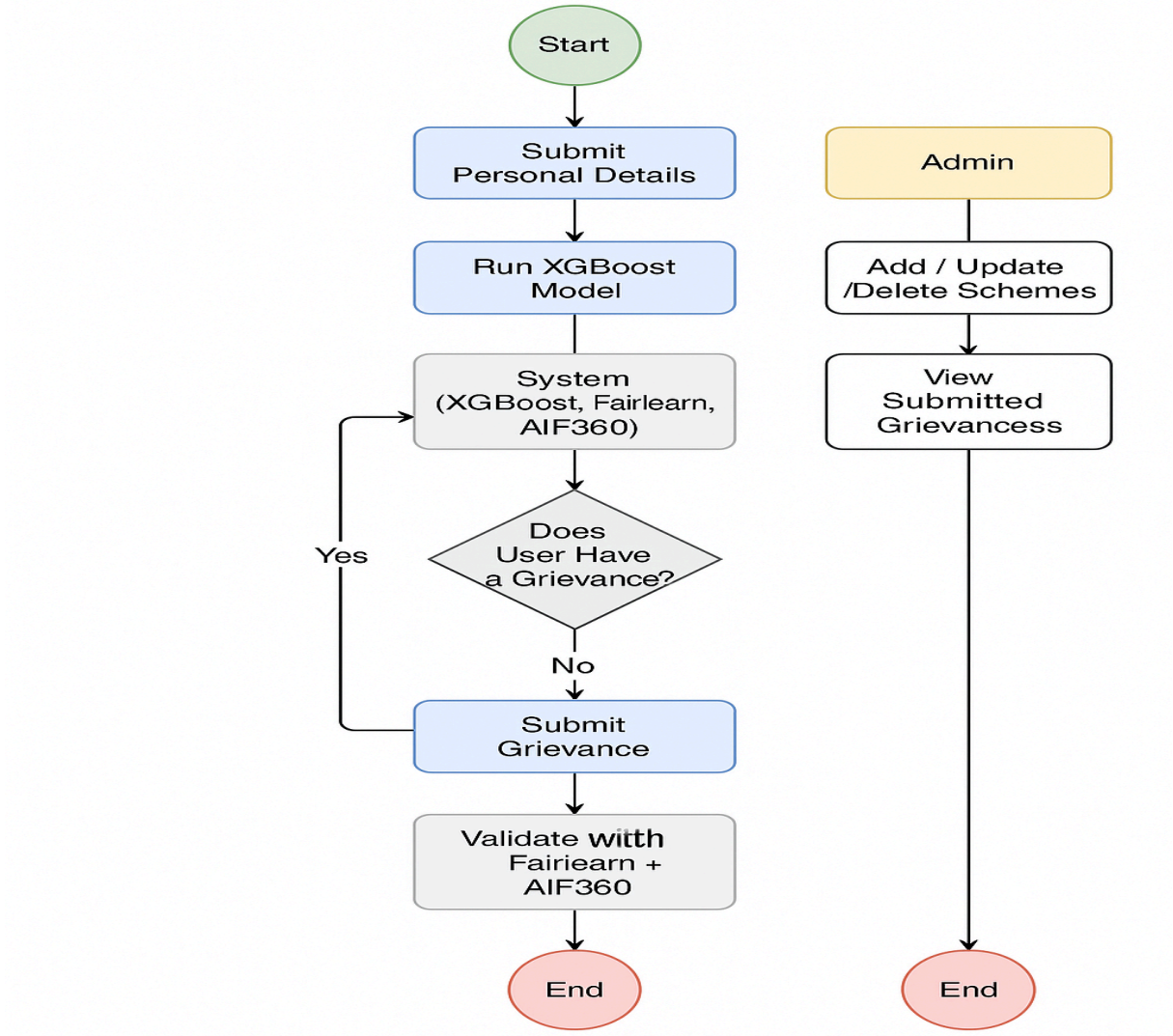

fig 3.1 Activity Diagram

3.3 DESIGN OF THE ENTIRE SYSTEM

The GRIEVEASE system is implemented using a layered architectural design that provides modularity, scalability, and ease of maintenance. Every layer of the system has a dedicated function. The frontend layer is the user interface and consists of a citizen portal for login, registration, profile submission, and a chatbot for assisting users in scheme-related inquiries. The backend layer is for the system's core logic and implemented using PHP. It oversees API interaction between the frontend and MySQL

database to ensure that data is processed, stored, and fetched correctly. The machine learning layer, which at present is emulated through rule-based logic, is planned to be replaced by an XGBoost-based model implemented as a Python microservice. The grievance management layer provides users with the ability to file complaints and monitor their statuses, and there is an admin interface for handling complaints. Finally, the database layer driven by MySQL structures user data, scheme data, and grievance data into individual tables. This modular and structured design makes feature integration easy and future-proofing simple.

3.4 DEVELOPMENTAL ENVIRONMENT

Development of the GRIEVEASE platform was conducted on XAMPP, an integrated development environment containing Apache, MySQL, and PHP. The configuration allowed easy communication between the user interface, server-side logic, and database. The application was developed based on standard web technologies including HTML and CSS for appearance and page structure, JavaScript for interactivity, and PHP for server-side processing. MySQL was used as the relational database system to store structured data in multiple tables. The selection of this tech stack was based on ease of use, local host features, and compatibility for integration with future technologies such as Python for machine learning. The local server allowed for easy testing and debugging during development and offered a good platform for future cloud deployment or transfer to government infrastructure.

3.4.1 HARDWARE REQUIREMENT

To develop and test the system, a fairly powered personal computer was adequate. The system was developed and tested on a computer with a dual-core Intel Core i3 processor and 4 GB RAM. While such a setup sufficed for development, it is suggested that a more powerful installation with an Intel Core i5 or greater processor

and 8 GB or more of RAM would be desirable for smoother execution, particularly when utilizing machine learning models. A regular display and low-level input peripherals such as keyboard and mouse were employed. A connection to the internet was needed for the downloading of libraries, testing API return values, and the emulation of future remote access.

Table 3.1 Hardware Requirements

COMPONENTS	SPECIFICATION
Processor	Intel Core i3 (Minimum) / Intel Core i5 or higher (Recommended)
RAM	4 GB (Minimum), 8 GB or more (Recommended for ML integration)
Storage	At least 10 GB of free disk space

3.4.2 SOFTWARE REQUIREMENTS

The main software stack utilized for this project involved XAMPP, which combined Apache server, PHP scripting engine, and MySQL database into one environment. Windows 10 was used for development, even though the platform can also work with Linux-based platforms. Visual Studio Code was utilized for coding and editing due to its extensive plugin support and in-built terminal. Google Chrome and Mozilla Firefox browsers were utilized to test the frontend interface and verify browser compatibility. Future releases of the project would need other tools, such as Python 3.x and packages like XGBoost, scikit-learn, Fairlearn, and AIF360 for machine learning and fairness testing. This software package gave a good and flexible platform to deploy and scale the system based on demand tor, which produces a human-like, context-aware answer. This answer is then displayed

back to the user in real-time interface.This streamlined process ensures SmartBuddy delivers accurate, context-rich, and secure responses for educational and research use, supporting a more intelligent and personalized digital workspace.

Table 3.2 Software Requirements

COMPONENTS	SPECIFICATION
Operating System	Windows 10 or higher with GPU support
Web Server	Apache (via XAMPP)
Backend Language & Frontend Technologies	PHP 8.x & HTML, CSS, JavaScript
Database	MySQL (Relational Database)

3.5 STATISTICAL ANALYSIS

Although the existing system is rule-based, statistical analysis will play a crucial role as the platform evolves to integrate machine learning for predicting scheme eligibility and assessing fairness. Upon the deployment of the XGBoost model, the system's performance will be evaluated using conventional metrics such as accuracy, precision, recall, and F1-score. These metrics will help ensure that the recommendations generated by the system are not only relevant but also statistically reliable and valid. Additionally, user behavior data, including login frequency, the number of recommendations accessed, and the volume of grievances submitted, will be monitored and analyzed to assess both system effectiveness and the level of user engagement. Moreover, complaint-related data will be examined to determine the average response and resolution times, identify frequently occurring complaint categories, and evaluate the overall resolution success rate. With the integration of

fairness auditing toolkits such as Fairlearn and AIF360, the platform will be regularly reviewed for potential biases in recommendation outcomes across different user groups. Ultimately, such analysis will contribute to building a more inclusive, effective, and citizen-oriented platform.

Table 3.3 Comparison Table

FEATURES	GRIEVEASE	CPGRAMS
User Interface	Modern, Intuitive, Mobile-friendly design	Outdated and non-intuitive UI
Login Options	Separate login for users and admins	Mostly central login system, can be confusing
Personalized Scheme Matching	Auto-matching based on user-profiles & needs	No personalized scheme suggestions
Analytics Dashboard	Displays number of users benefitted / success metrics	Limited analytics visibility to users
Grievance Filling	Simplified, user-guided grievance process	Available, but form-heavy and bureaucratic

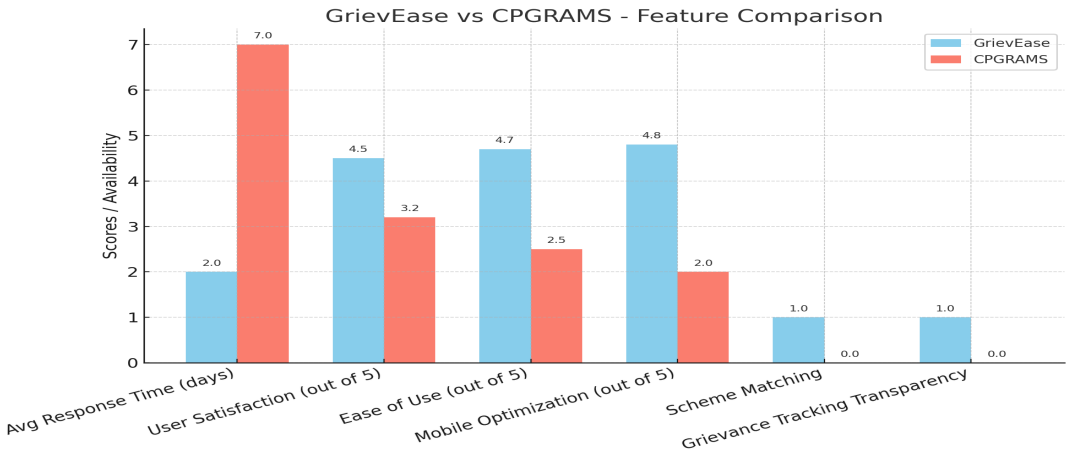

Fig 3.2 : Comparison Graph

CHAPTER 4

4.MODULE DESCRIPTION

4.1 SYSTEM ARCHITECTURE

The architecture of the Government Scheme Recommendation System is designed to provide personalized, AI-driven welfare services through a modular and user-friendly web platform. Citizens interact via a frontend composed of a user portal and chatbot, where they enter personal details and ask scheme-related questions. This data is sent to the backend API server, which manages communication between the frontend, databases, machine learning model, and grievance system. User data is processed using an XGBoost model to generate scheme recommendations. The grievance system includes a status tracker and handler, supported by fairness tools like Fairlearn and AIF360, to ensure transparency and equity. The system integrates real-time recommendations and grievance tracking, transforming manual welfare processes into a smart, accessible solution.

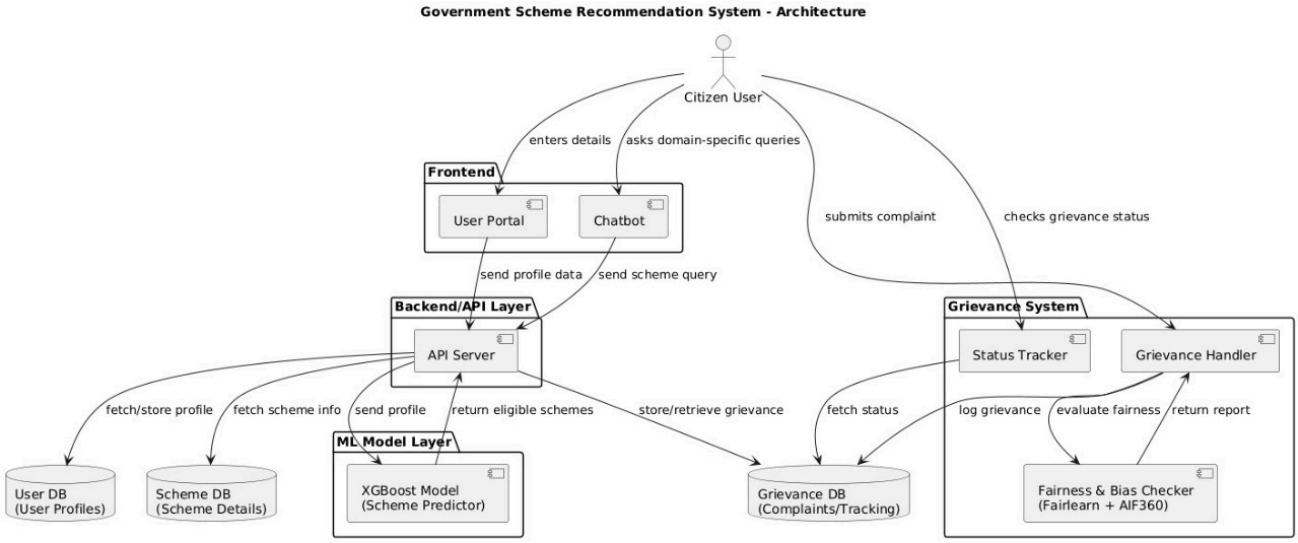

fig 4.1 System Architecture

4.1.1 HIGH LEVEL SYSTEM OVERVIEW

Grievease is a digital platform designed to assist citizens in addressing local or national grievances while intelligently linking them with relevant government schemes. The web application acts as a centralized hub where users can raise complaints and explore schemes tailored to their issues. The overall architecture is modular, allowing scalability and adaptability for future feature integration such as AI-based recommendations or real-time analytics. The architecture is built on a standard MVC structure, ensuring separation of concerns and code maintainability. It integrates secure authentication, robust data processing, and a structured grievance-scheme matching model, all tied together via a seamless user interface. Users log in through a secure portal and interact with the system in a streamlined manner, while administrators gain access to a detailed dashboard to manage and resolve grievances. This layered structure not only enables a clean flow of data but also facilitates continuous updates and modular enhancements. Scalability, security, and responsiveness were key design priorities during the system's development.

4.1.2 USER INTERACTION DIAGRAM

User interaction in Grievease is streamlined to reduce friction and ensure accessibility for citizens of all backgrounds. The platform supports two types of users: citizens and administrators. Citizens interact with the platform by first signing up or logging into their account. Once authenticated, they can navigate to the dashboard where they can either submit a new grievance or view the status of previously submitted issues. Each grievance form includes structured inputs such as category, region, and description, allowing for effective classification and data collection. Upon submission, users receive a notification and can monitor the progress in real time. On the admin side, the interface allows for efficient filtering of complaints based on keywords, categories,

and submission dates. Admins can view detailed reports, send status updates, and mark grievances as resolved. The interaction flow is enhanced with real-time feedback mechanisms and intuitive navigation, improving usability. The design emphasizes ease-of-use, responsiveness, and informative alerts to create a cohesive user experience across devices.

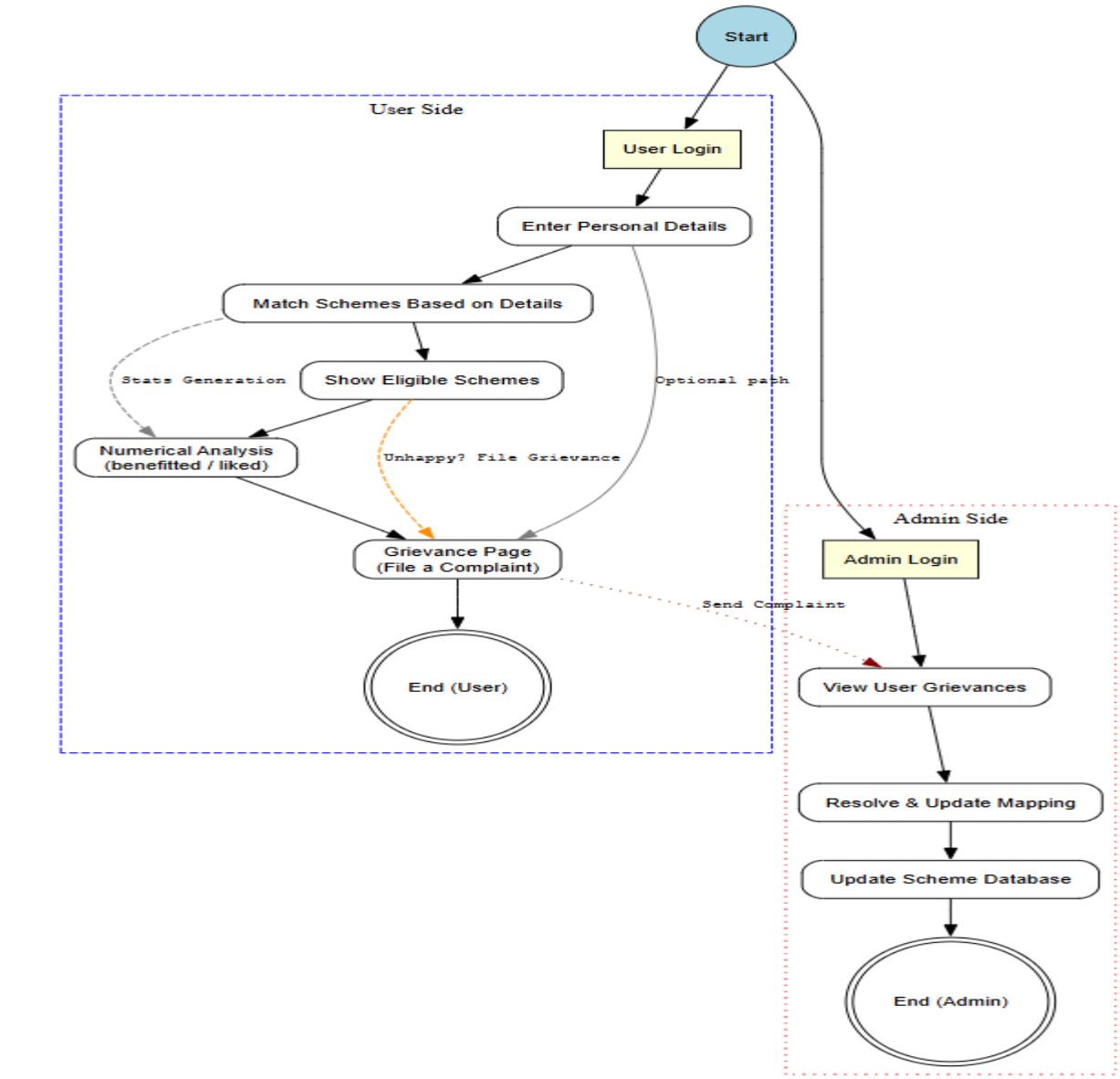

fig 4.2 USER INTERACTION DIAGRAM

4.1.3 BACKEND INFRASTRUCTURE

The backend infrastructure of Grievance is built to handle high volumes of user data, scheme metadata, and dynamic updates, all while ensuring minimal latency and optimal performance. It is developed using a combination of modern web technologies including secure APIs for data flow and structured MySQL or NoSQL databases for information storage. The server-side logic handles authentication, data processing, scheme matching, and communication between the client and server. All incoming data is validated and sanitized to prevent security threats such as SQL injection or XSS attacks. Grievances and user information are stored in relational tables, while scheme data can be updated and referenced dynamically using indexing and search optimization. The system also includes a logging service for monitoring error rates and transaction records, useful for both debugging and analytics. Moreover, the backend supports asynchronous communication for non-blocking user experiences. This infrastructure ensures that Grieves remains performant, secure, and scalable even as usage increases across districts, departments, or schemes.

4.2 DATA HANDLING

4.2.1 DATA INGESTION MODULE

The Data Ingestion Module in Grievease plays a crucial role in collecting, structuring, and storing user-submitted grievance data as well as official government scheme information. Grievance entries submitted by users are captured through structured forms and then routed into the backend database, where fields such as complaint category, location, and description are securely stored. The module also supports the ingestion of government scheme data, which is often sourced from structured inputs or periodically updated files. This scheme data is standardized and tagged with metadata such as eligibility, department, and region. The ingestion process includes validation

checks to ensure that data is clean, consistent, and formatted for downstream processing. Preprocessing steps such as stop-word removal, stemming, and lemmatization are applied to the textual content. This enables efficient search, matching, and reporting later on. The ingestion pipeline has been optimized for both manual and automated updates, ensuring the system remains up-to-date with new schemes and continuously receives valuable grievance inputs from users.

4.2.2 SCHEME MATCHING USING XGBOOST

To provide accurate and meaningful scheme recommendations, Grievease integrates the XGBoost machine learning algorithm into its backend logic. XGBoost, known for its speed and precision in classification tasks, analyzes structured grievance data and predicts the most relevant government schemes for each case. This is accomplished by training the model on a labeled dataset of historical grievances and their corresponding schemes, enabling it to identify patterns across features such as category, location, keywords, and urgency level. By learning these relationships, the model can make data-driven predictions even when faced with new and unique grievances. Unlike simple keyword-based systems, XGBoost uses decision trees that consider multiple variables simultaneously, allowing for more nuanced and personalized suggestions. The model continuously improves through retraining on new data, ensuring relevance and adaptability. Additionally, administrators can monitor model outputs, verify its predictions, and refine them if needed. This intelligent automation not only speeds up grievance processing but also increases citizen satisfaction by guiding them toward appropriate and beneficial government initiatives.

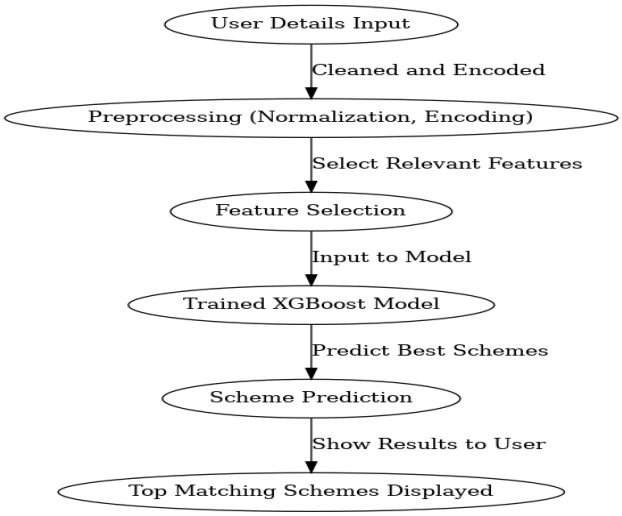

fig 4.3 Scheme Matching using XGBoost

4.2.3 DATA PREPROCESSING & FEATURE ENGINEERING

Before inputting grievance and scheme data into the XGBoost model, Grievease undertakes comprehensive data preprocessing and feature engineering to ensure optimal model performance. Raw data often contains inconsistencies, missing values, and noise, which, if left unaddressed, can negatively impact prediction accuracy. The preprocessing stage involves several key tasks, including null value imputation, standardizing text, removing redundant entries, and encoding categorical variables into numerical formats compatible with machine learning models. Additionally, text fields such as grievance descriptions undergo tokenization, stemming, and stop-word removal to extract meaningful information. Feature engineering further refines the dataset by generating new variables—such as urgency scores, keyword flags, and categorical embeddings—that provide richer context to the model. These features help the algorithm capture intricate patterns and improve its generalization capability. This structured approach to data preparation transforms unstructured user inputs into a well-defined dataset, significantly enhancing the model's ability to generate precise and reliable scheme recommendations tailored to each individual grievance.

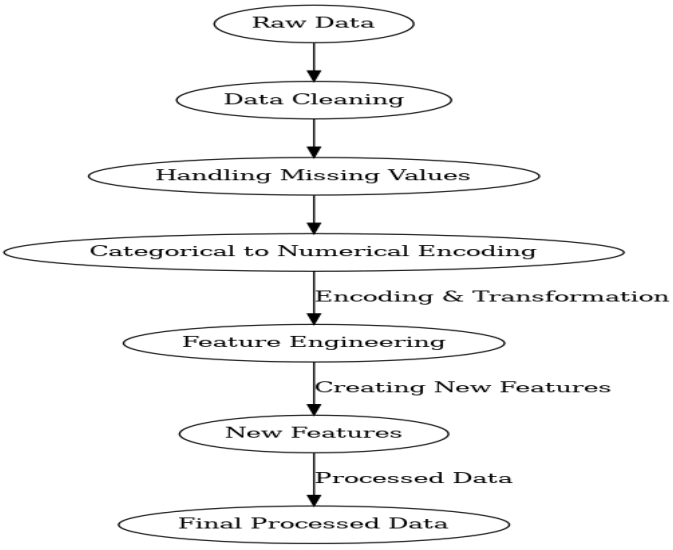

fig 4.4 Data preprocessing and feature engineering

4.3 ADMIN PANEL & GRIEVANCE RESOLUTION.

4.3.1 ADMIN PANEL OVERVIEW

The Admin Panel in Grievease acts as the central hub for administrative operations, providing authorized personnel with powerful tools to manage, track, and resolve user grievances effectively. Designed with an intuitive interface, the panel presents a consolidated view of all submitted grievances, categorized by status, type, urgency, and location. Administrators can drill down into each grievance, view detailed descriptions, user-submitted documents, and timestamps, and assign them to relevant departments or officers. The panel supports bulk actions, filtering, and sorting functionalities that enhance operational efficiency. Additionally, the admin panel offers communication tools that allow direct messaging with users, updates on progress, or requests for more information—all of which help close the feedback loop. Each admin action is logged in the backend to maintain transparency and ensure

accountability. With role-based access controls and authentication layers, the panel also ensures that sensitive data remains secure. Overall, it serves as the operational backbone of the grievance resolution process, empowering public administrators to act quickly and responsibly.

4.3.2 GRIEVANCE RESOLUTION FLOW

Grievease follows a streamlined grievance resolution flow that promotes timely response, transparency, and efficient collaboration among departments. When a citizen submits a grievance, it is automatically categorized and added to the administrator queue, where it awaits review. The admin can examine the full complaint details—including metadata like submission date, location, and grievance category—and then assess its urgency and assign it to the appropriate government department or official. Each grievance is tracked through stages: received, in-progress, escalated (if needed), and resolved. During this process, the system allows for updates to be posted directly to the user's dashboard, keeping them informed of actions taken. Admins can also upload files or issue clarifications if more information is needed. Once the issue is resolved, the admin marks the grievance as closed, which triggers a notification to the user and archives the case for future reference. This entire process ensures structured tracking, minimizes delays, and fosters trust between citizens and the governing bodies.

4.4 USER INTERFACE & EXPERIENCE

The user interface of SmartBuddy has been thoughtfully designed using Streamlit, focusing on simplicity, accessibility, and responsiveness. The goal is to provide a smooth conversational experience for users while integrating powerful backend operations like semantic retrieval and LLM interaction.

4.4.1 FRONTEND

Grievease's frontend interface is developed using HTML, CSS, and JavaScript, ensuring a user-friendly and responsive experience. The homepage is designed with a clean layout and intuitive navigation, featuring distinct sections for grievance submission, tracking, and recommended schemes. PHP powers the backend logic to handle dynamic content generation, form submissions, and interactions with the database. Forms are created using HTML, styled with CSS, and made interactive with JavaScript, allowing for a seamless user experience. Grievance submission forms capture user input efficiently, with real-time validation through JavaScript to ensure all required fields are properly filled. PHP processes the form data, interacts with the database, and provides feedback to the user. The platform is designed to be mobile-responsive, ensuring users in rural areas with varying internet access can still navigate the site effectively. Admin panels and grievance status updates are also dynamically generated using PHP, providing real-time updates to users. This approach allows for a flexible, scalable, and easy-to-maintain platform.

4.4.2 USER AUTHENTICATION & ROLE MANAGEMENT

Ensuring secure and personalized access is a cornerstone of the Grievease platform. The system incorporates a robust authentication mechanism that verifies the identity of users during the login process using unique credentials. There are two primary roles in the system: citizens and administrators each granted access to distinct functionalities based on their permissions. Upon login, the backend verifies the user’s role and redirects them to the appropriate dashboard. Citizens are provided with interfaces to submit grievances, track their status, and view suggested schemes. Administrators, on the other hand, gain access to tools for managing grievances, communicating with users, and overseeing resolution workflows. Role-based access control (RBAC)

ensures that sensitive operations such as grievance assignment and scheme updates are restricted to authorized personnel only. This layered security approach protects user data from unauthorized access and maintains the operational integrity of the platform. By implementing clear role distinctions and access privileges, Grievease creates a secure, efficient, and user-centric environment for grievance redressal.

4.5 PERFORMANCE & OPTIMIZATION

4.5.1 RESPONSE TIME & MATCHING EFFICIENCY

Performance optimization is central to Grievease 's architecture, particularly in maintaining low response times and delivering efficient scheme matching. The grievance submission and processing pipelines are designed to handle multiple simultaneous users without lag or crashes, using asynchronous operations and lightweight queries. Once a grievance is submitted, the XGBoost model is triggered, performing inference in real-time to suggest relevant schemes. This model has been optimized to balance speed and accuracy by minimizing overheads during prediction and loading pre-trained models efficiently into memory. Load balancing and session management strategies ensure that backend services remain responsive, even during high user traffic. Additionally, caching frequently accessed data—such as popular schemes and resolved grievance summaries—helps reduce processing times. Continuous performance monitoring through logging and profiling tools helps the development team identify bottlenecks and address them proactively. These efforts collectively ensure that users can navigate the platform and receive actionable results quickly, reinforcing trust in the system’s reliability.

4.5.2 ACCURACY & SCHEME RELEVANCE

Accuracy in scheme recommendations is a key performance metric in Grievease, directly impacting user satisfaction and trust. The XGBoost model is rigorously evaluated using precision, recall, and F1 scores to ensure that the suggested schemes are not only accurate but also contextually appropriate. The model is trained on diverse grievance datasets, including regional variations and linguistic nuances, to improve its generalizability across different user submissions. Furthermore, Grievease incorporates a feedback loop where users can rate the relevance of recommended schemes or flag mismatches. This feedback is logged and used to retrain the model periodically, improving its performance over time. Additionally, domain experts and administrators can review scheme mappings to manually adjust associations in edge cases. By blending automated intelligence with human oversight, the platform ensures high-quality recommendations. Scheme descriptions are regularly updated, and obsolete entries are removed from the database to maintain relevance. These practices uphold the platform's core objective of providing accurate, timely, and helpful guidance to citizens.

4.6 DEPLOYMENT & STORAGE

4.6.1 LOCAL VS REMOTE DEPLOYMENT

Grievance is built with deployment flexibility in mind, allowing institutions and governing bodies to choose between local and remote setups based on their technical and policy requirements. Local deployment involves hosting the entire application and database on institutional or government-managed servers, which offers complete control over data access, privacy, and network restrictions. This model is ideal for regions with strict data protection regulations or limited internet connectivity. On the

other hand, remote deployment utilizes cloud-based platforms such as AWS, Azure, or Google Cloud, providing scalability, automated backups, high availability, and reduced infrastructure overhead. Cloud deployment enables seamless software updates and centralized monitoring, especially beneficial for large-scale implementations involving multiple districts or states. Ultimately, Grievease empowers administrators to select the most suitable deployment strategy without compromising performance, security, or user experience.

4.6.2 DATA STORAGE & MANAGEMENT

Data storage in Grievease is designed to uphold the highest standards of security, reliability, and accessibility. All user-submitted data, including grievances, scheme metadata, login credentials, and admin responses, are stored in a structured relational database system. This database is optimized for fast queries and indexed based on key attributes like grievance ID, status, and submission date. Sensitive information is encrypted both at rest and in transit using industry-standard protocols. To ensure data availability and resilience, regular automated backups are scheduled, and version control mechanisms are implemented to recover data in case of failure. Role-based access ensures that only authorized personnel can view or manipulate sensitive data. Additionally, data retention policies are enforced to periodically archive or delete outdated records, maintaining system efficiency. Logging systems monitor all access and modifications, supporting audit trails and compliance with national data governance guidelines. These practices establish GrievEase as a secure, transparent, and future-ready platform capable of managing large volumes of civic data responsibly.

CHAPTER 5

IMPLEMENTATION AND RESULTS

5.1 IMPLEMENTATION

Grievease is a comprehensive web application developed using PHP, designed to improve the grievance redressal process by enabling citizens to submit, track, and resolve complaints effectively. The platform uses a combination of HTML, CSS, and JavaScript to provide a responsive, user-friendly interface that adapts seamlessly to various devices, ensuring accessibility for all users. Citizens can submit grievances through easy-to-use forms, which are processed by the backend using PHP. The system integrates a machine learning model (XGBoost) to intelligently match submitted grievances with the most relevant government schemes, ensuring that citizens are provided with timely and personalized recommendations based on their needs. Administrators can manage and resolve grievances through a dedicated admin panel, which offers a comprehensive view of all submitted complaints, their statuses, and the relevant actions taken. The admin panel also allows for easy communication with citizens and the assignment of grievances to different departments for resolution. The platform features robust authentication and role-based access control, ensuring that user data is secure and sensitive information is protected. Real-time notifications and seamless interactions between users and admins enhance the overall experience, making Grievease an effective tool for citizens to engage with government services and receive swift resolutions to their concerns.

5.2 OUTPUT SCREENSHOTS

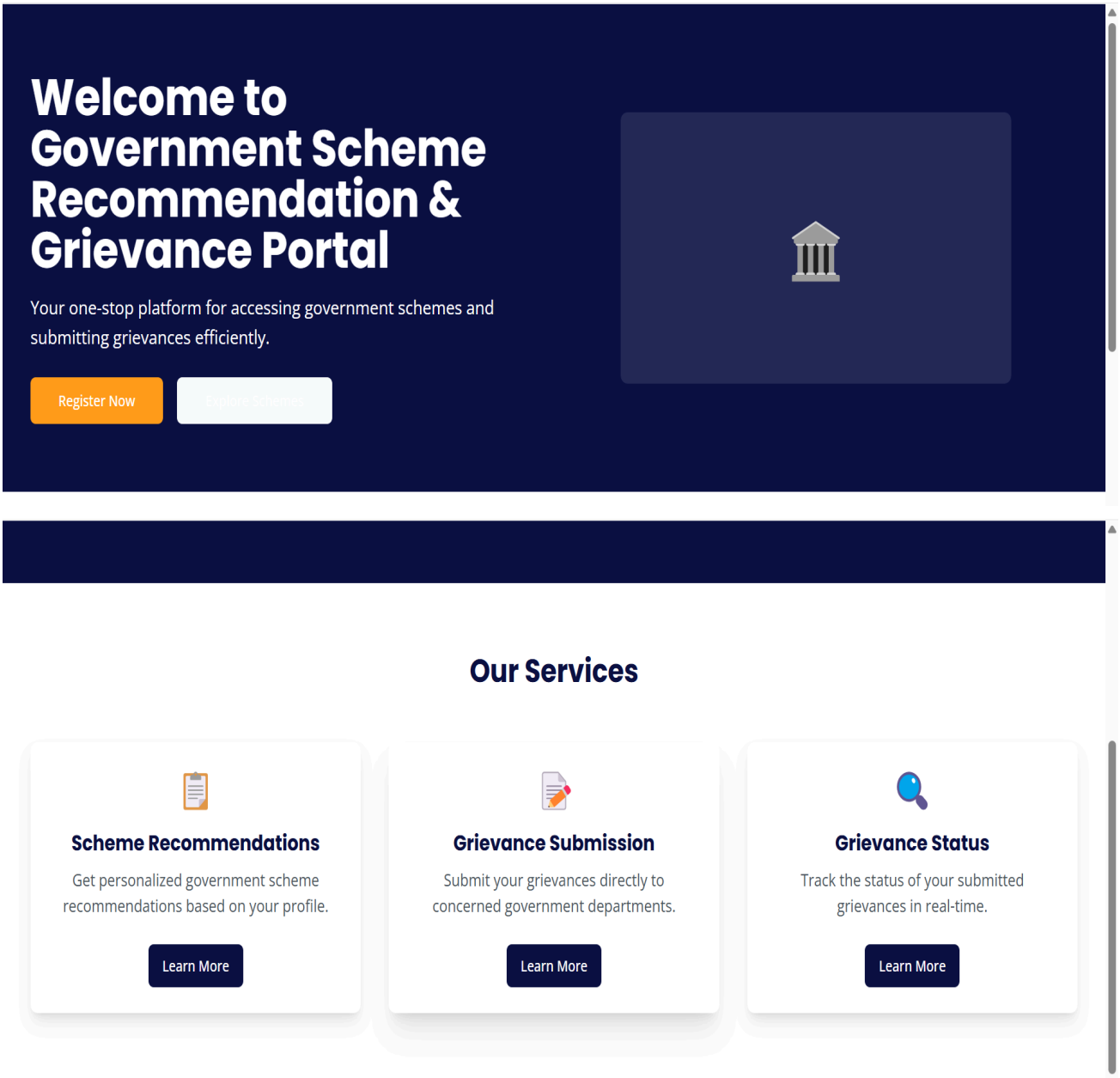

fig 5.1 Website Design

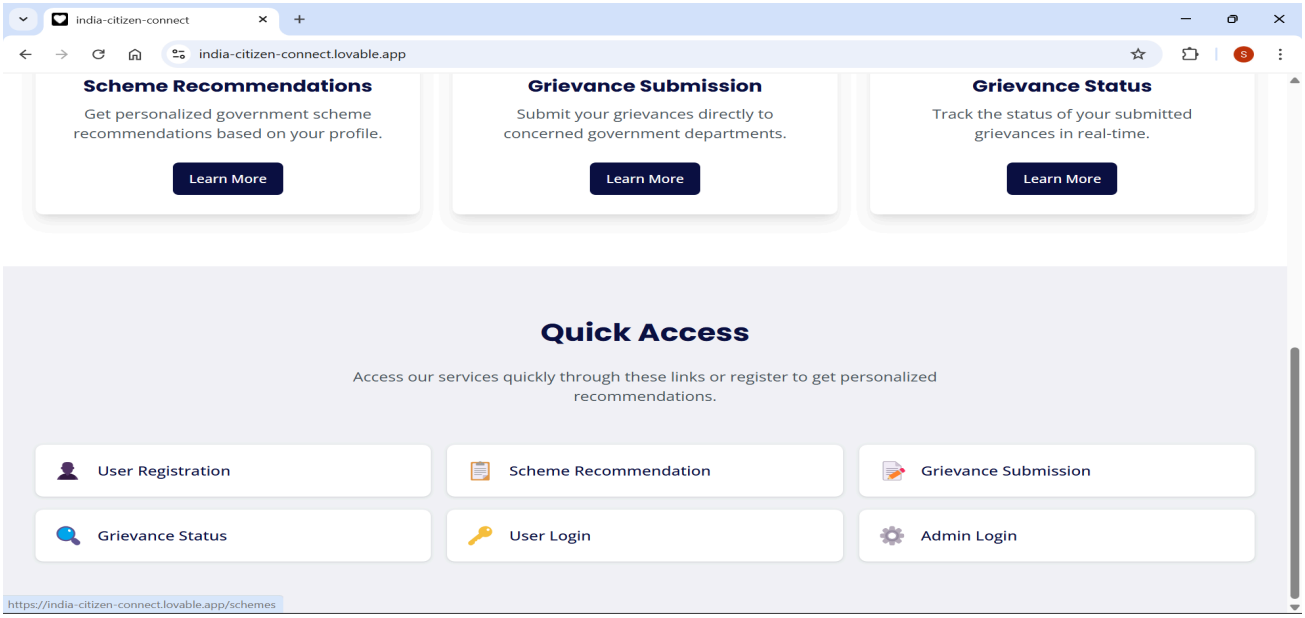

fig 5.2 Features

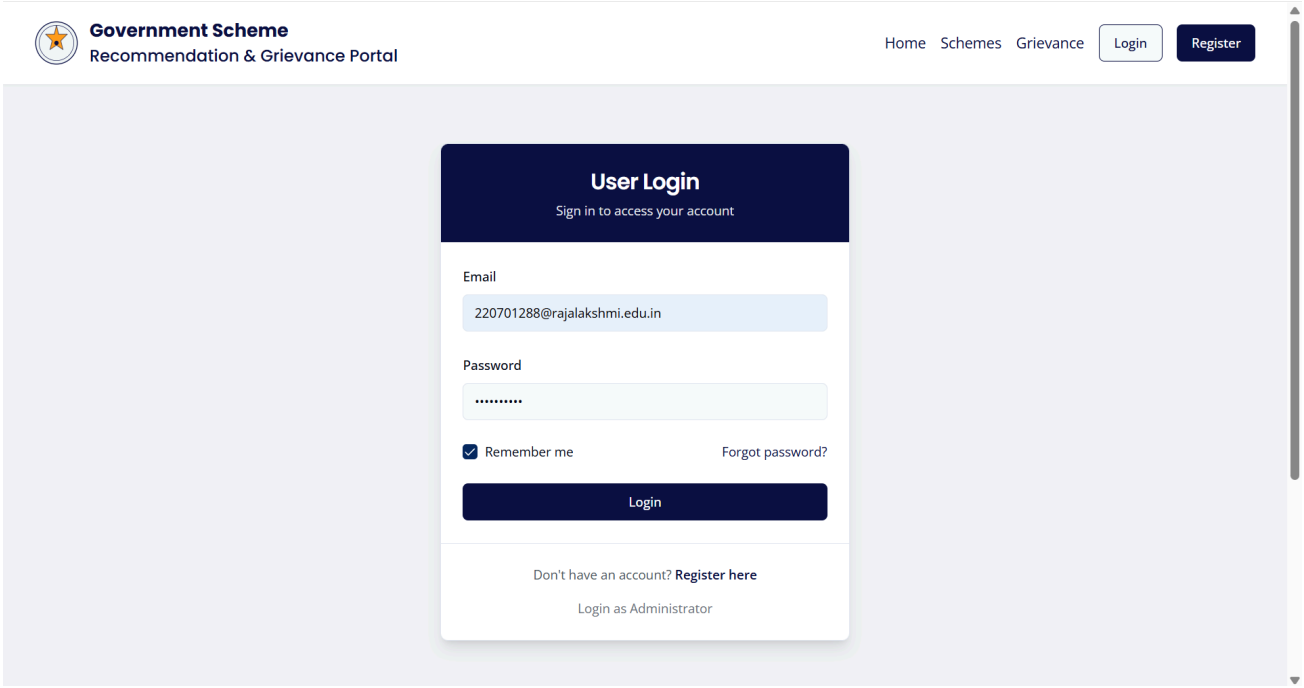

fig 5.3 User Login

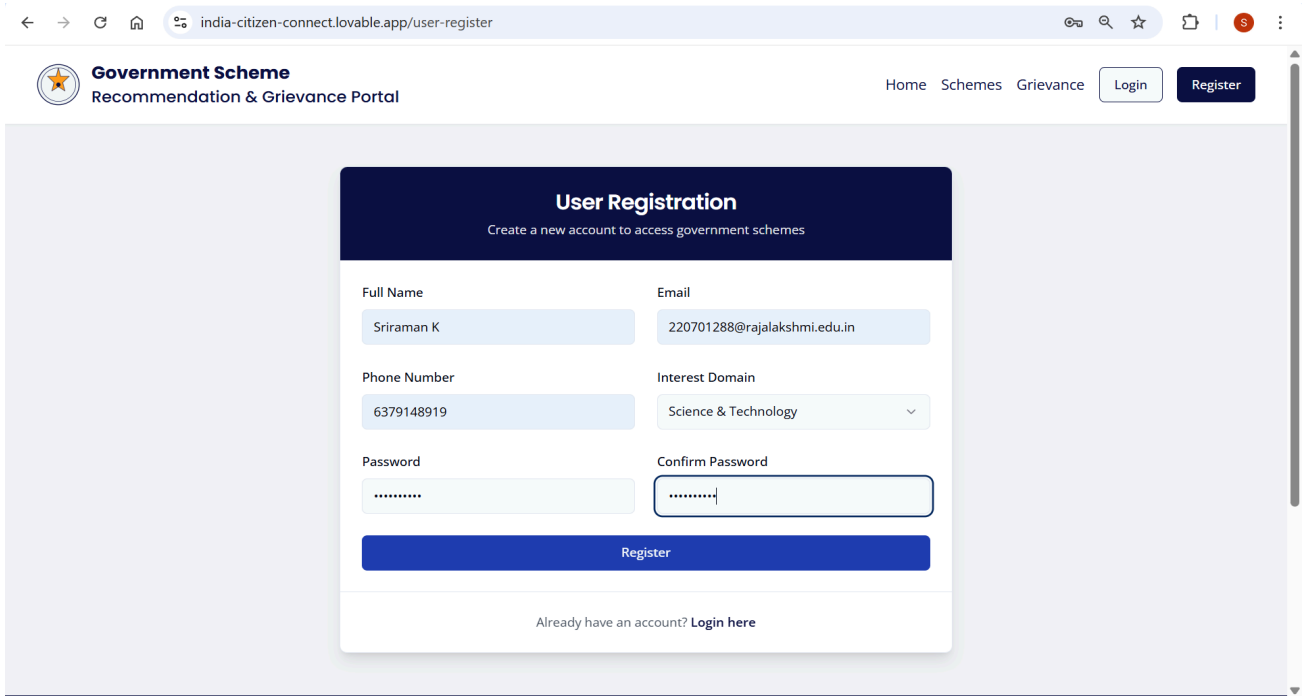

fig 5.4 User Sign in

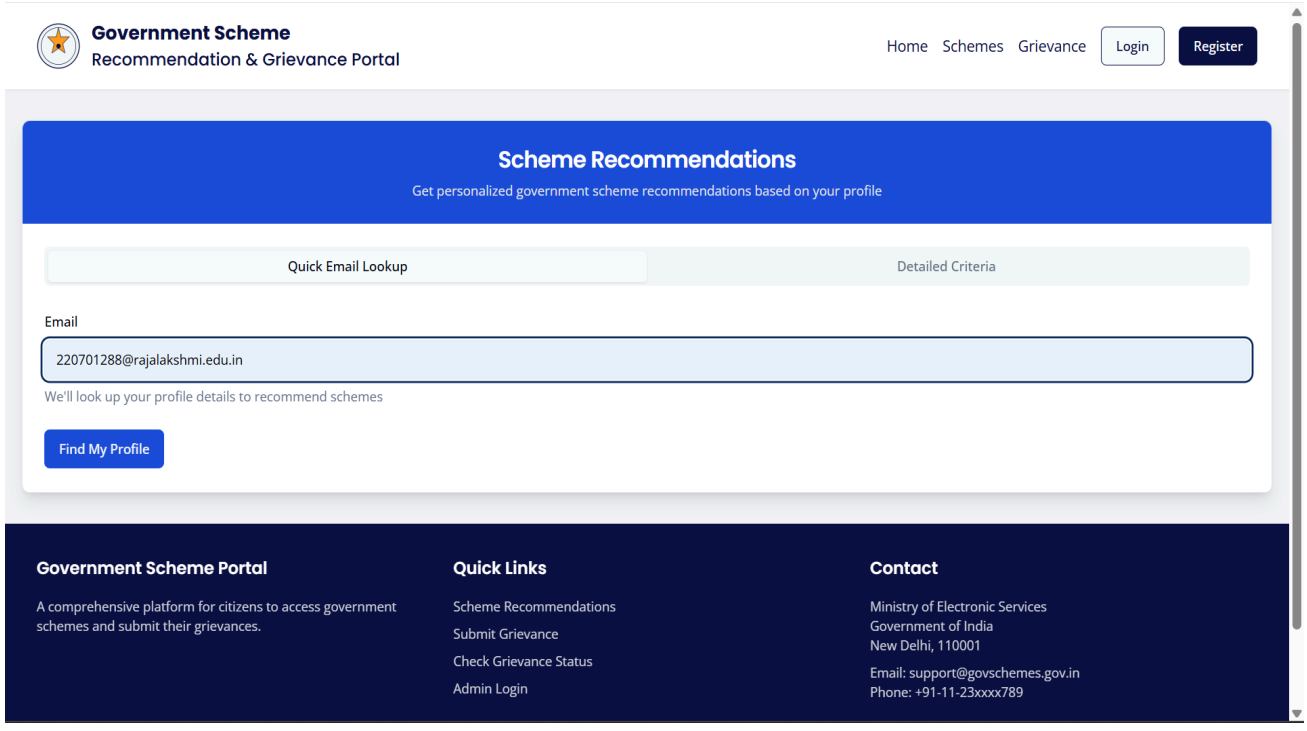

fig 5.5 Scheme Lookup

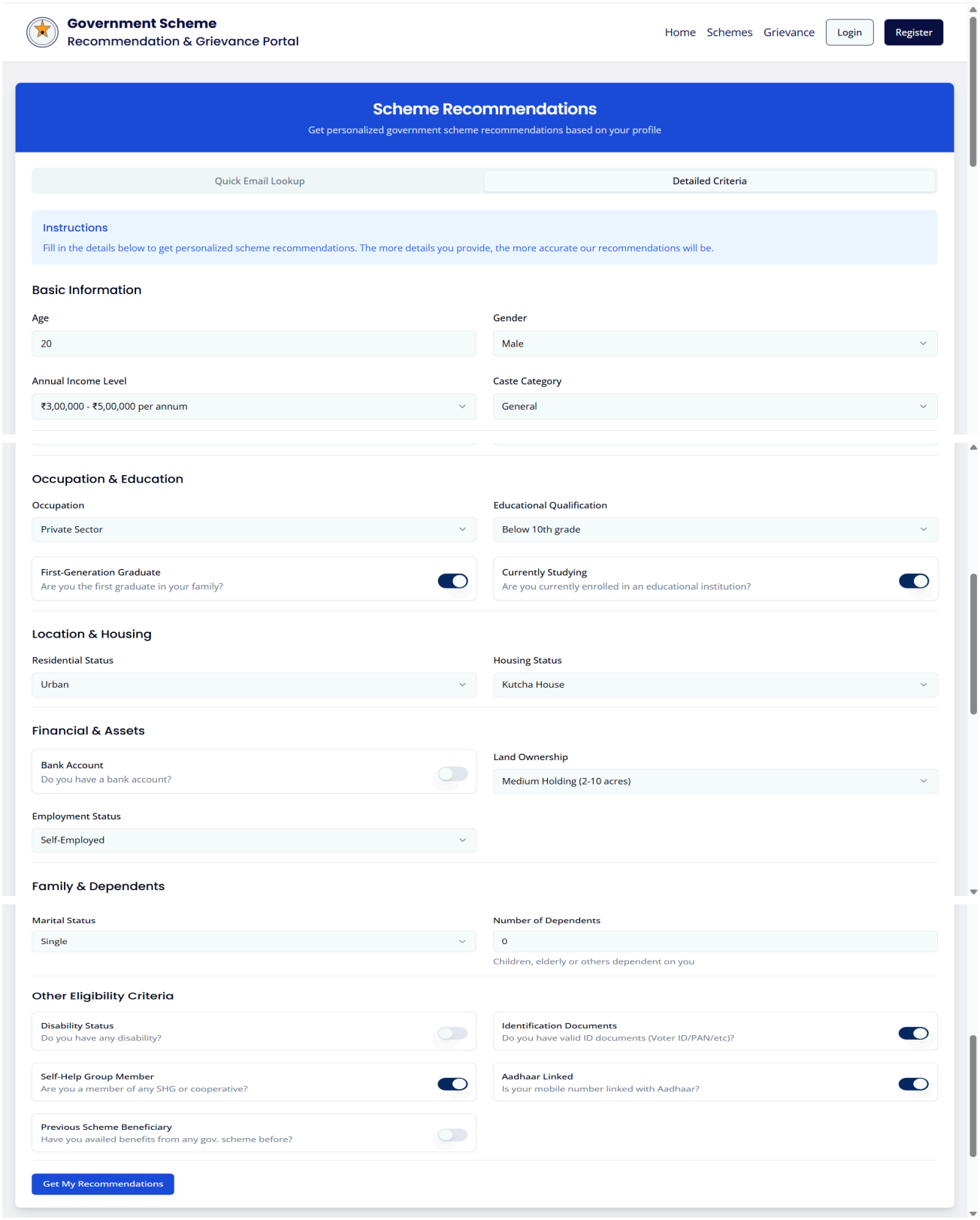

fig 5.6 User Details Collection

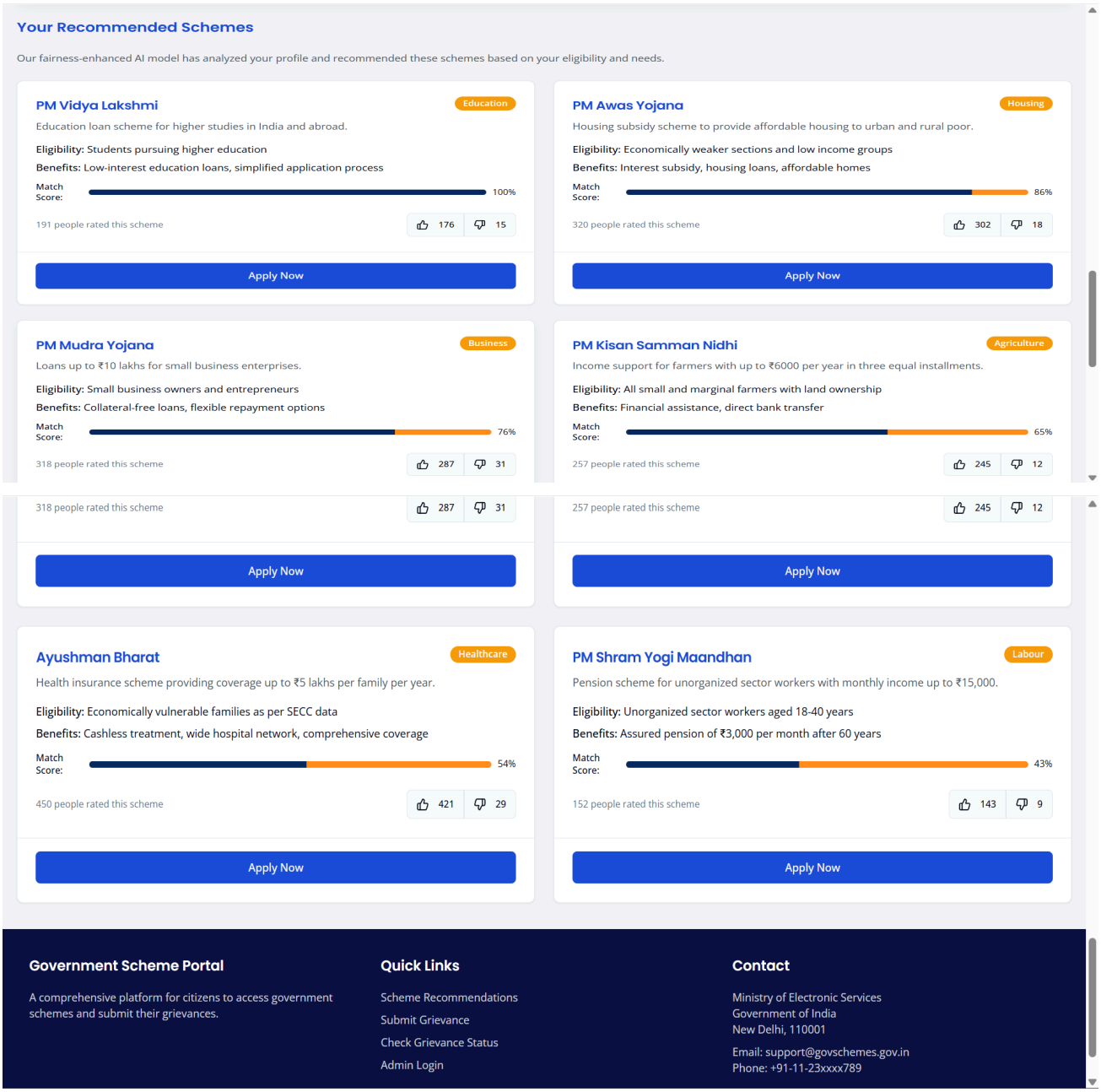

fig 5.7 ML Matching Model

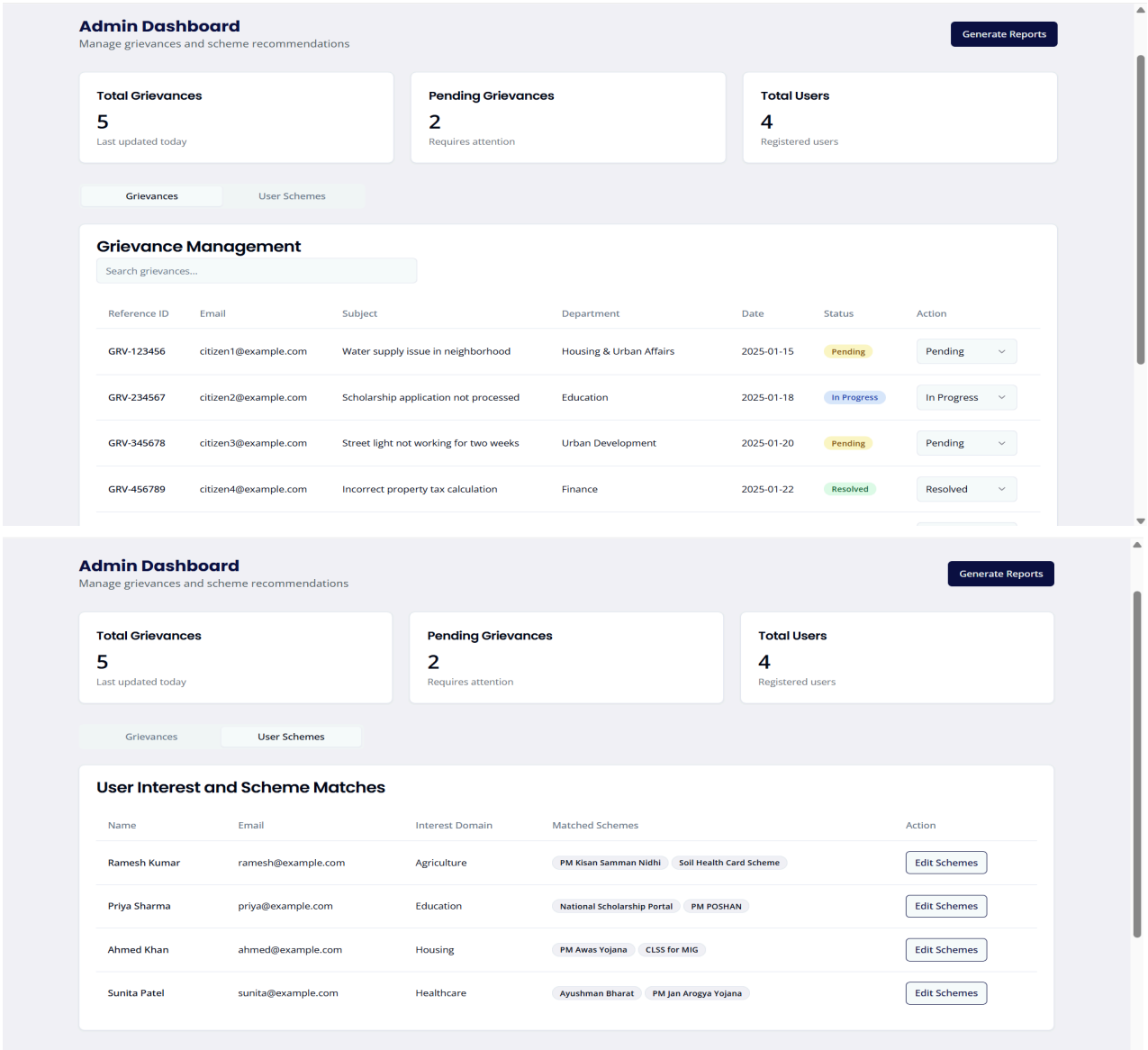

fig 5.8 Admin Dashboard

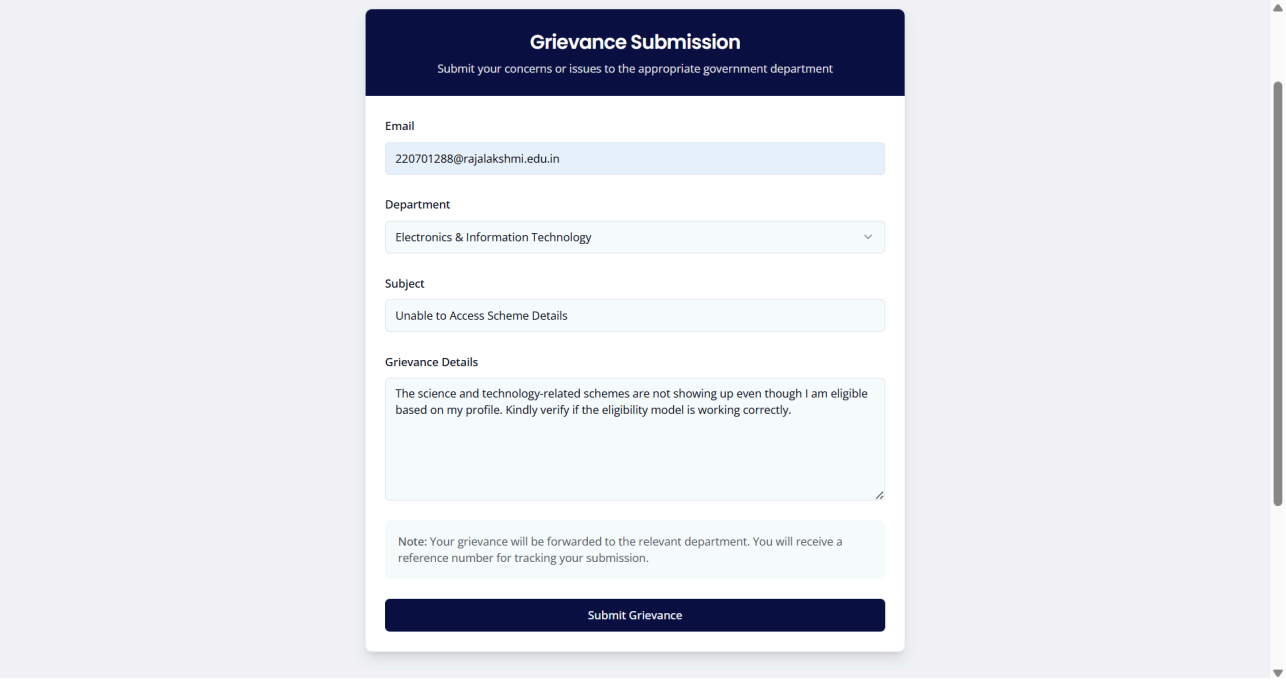

fig 5.9 Grievance Submission

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

GRIEVEASE is a citizen-centric digital solution designed to simplify access to government welfare schemes through the power of AI and intelligent automation. In the current system, many individuals struggle to find schemes they are eligible for due to scattered information, complex requirements, and lack of guidance. GRIEVEASE addresses this challenge by offering a personalized platform where users can easily discover suitable welfare schemes based on their profile and needs. The system provides a simple, intuitive interface and includes a chatbot that helps guide users through the process of identifying, understanding, and applying for schemes. This conversational assistant makes GRIEVEASE accessible to users of all backgrounds, including those with limited digital experience. The platform also incorporates user reviews and ratings, enabling a community-driven feedback system that improves transparency and helps future users make informed decisions. Another key feature is its grievance redressal module, which allows users to submit complaints, track their status, and ensure fairness using bias detection tools like Fairlearn and AIF360. This reinforces trust in the system and encourages citizen engagement. By combining machine learning, user-friendly design, and real-time interactivity, GRIEVEASE transforms the traditional, manual approach to welfare access into a fast, efficient, and fair experience. It empowers users with knowledge, simplifies decision-making, and promotes a more inclusive approach to public service delivery. GRIEVEASE ultimately sets a new standard for how citizens interact with government support systems—smarter, faster, and more transparently.

6.2 FUTURE ENHANCEMENT

Replace the existing rule-based eligibility logic with a complete XGBoost model rolled out through a Python microservice. Enrich the model with additional diverse and dynamic attributes such as real-time income information, family size, medical conditions, etc., for even more precise recommendations. Implement regional language support and voice-based chatbot interfaces to provide access to rural and less literate communities. Develop a cross-platform mobile app*with Flutter or React Native to provide greater access for smartphone users, particularly in rural regions. Integrate with live government APIs (UIDAI, SECC, State Welfare Portals) for instant verification and dynamic scheme eligibility updates. Apply NLP and sentiment analysis to determine the urgency of submitted grievances and auto-prioritize urgent cases for quicker resolution. Implement blockchain technology to register an unerasable record of grievances and scheme allocations to foster transparency and accountability. Permit the review of schemes by citizens with feedback, thereby establishing a grassroots rating system of various schemes. Implement a data analytics dashboard to track scheme usage patterns, citizen participation, and grievance rates for administrators. Regularly execute bias detection (Fairlearn, AIF360) and automatically retrain models for fair prediction across different demographics. Allow offline completion of recommendations and grievance forms with the ability to sync when internet is available — particularly helpful in rural or remote locations.

REFERENCES

1.Role of G-Cloud in Citizen Centric Governance
https://sci-hub.se/10.1109/PDGC.2012.6449789
2.Smart E-Grievance System For Effective Communication In Smart Cities
https://sci-hub.se/10.1109/ICSCET.2018.8537244
3.Smart Complaint Redressal System Using Ethereum Blockchain
https://sci-hub.se/10.1109/DISCOVER50404.2020. 9278122
4. A Citizen-centric Approach towards Global-scale Smart City Platform
https://sci-hub.se/10.1109/RIOT.2015.7104913
5.Evaluating Web-based e-government services with a citizen-centric approach
https://sci-hub.se/10.1109/HICSS.2005.252
6.Measuring the effectiveness of Government Schemes using Machine learning algorithms
https://sci-hub.se/10.1109/iCATccT44854.2018. 9001961
7.Study on the application of machine learning in government service: take consumer protection service as an example
https://sci-hub.se/10.1109/ICSSSM.2018.8465040 8.An Enterprise Architecture Mapping Approach for Realizing e-Government
https://sci-hub.se/10.1109/ICCITechnology.2013. 6579515
9.Uncovering Concerns of Citizens Through Machine Learning and Social Network Sentiment Analysis
https://ieeexplore.ieee.org/stamp/stamp.jsp? arnumber=10591750
10.Algorithm-Based Handling of Complaints Data from the Usage Phase
https://sci-hub.se/https://doi.org/10.1109/ QR2MSE46217.2019.902115

11.Influence of the Contact Center Systems Development on Key Performance Indicators 1.Role of G-Cloud in Citizen Centric Governance.
https://sci-hub.se/10.1109/PDGC.2012.6449789
12. Inductive Learning of Dispute Scenarios for Online Resolution of Customer Complaints.
https://sci-hub.se/https://doi.org/10.1109/IS.2006.3484 01
13. Automatic Multilabel Categorization using Learning to Rank Framework for Complaint Text on Bandung Government.
https://sci-hub.se/https://doi.org/10.1109/ICAICTA.20 14.7005910
14. A Scalable Machine Learning Online Service for Big Data Real-Time Analysis.
https://sci-hub.se/10.1109/CIBD.2014.7011537
15. SentScore: Autonomous Text Sentiment scoring and Summarizing System related to Complaint Management.
https://sci-hub.se/https://doi.org/10.1109/I2CT42659.2 018.9058199
16.Big Data Framework for National e-Governance Plan.
https://sci-hub.se/10.1109/ICTKE.2013.6756283
17.Systematic Review of Big Data Analytics in Governance.
https://sci-hub.se/10.1109/ISS1.2017.8389462
18.e-Government Services in Ghana - Current State and Future Perspectives.
https://sci-hub.se/10.1109/CSCI.2017.108
19.Web Portal for Effective Student Grievance Support System.
https://sci-hub.se/10.1109/ICACCS48705.2020.90743 44
20. Software Architectural Design Model For e-Governance Systems.
https://sci-hub.se/https://doi.org/10.1109/TENCON.20 03.1273310