

AI-Powered Urban Community Service Hub: A Smart Complaint And Society Management System

Sidlambe Vaishanvi¹, Bhalerao Anushka², Jadhav Anjali³, Prof. Swati Dhadake⁴

^{1, 2, 3, 4}Dept of Computer Engineering

^{1, 2, 3, 4} SITS's Smt. Kashibai Navale College of Engineering

Abstract- The swift urban growth of cities has led to a significant need for smart digital solutions that can effectively oversee housing society activities. The AI-Driven Urban Community Service Centre is a multipurpose, mobile app that unites owners, tenants, residents, administrators, and service providers into a unified ecosystem. The solution integrates artificial intelligence to improve decision-making support while streamlining reservation administration, payment processing, and complaint classification. Whereas Natural Language Processing (NLP) analyzes and categorizes complaint messages, computer vision algorithms identify irregularities in provided images. Sentiment analysis facilitates the tracking of service quality comments, while optional voice-to-text conversion guarantees accessibility for all age groups. The backend system makes use of Flask microservices, Node.js, and a scalable MongoDB database. In urban areas, the proposed paradigm improves responsiveness, increases transparency, and decreases manual involvement. Future developments will make it possible to govern communities completely automatically using chatbots, blockchain verification, and predictive analytics. This approach shows how artificial intelligence (AI) has the potential to make city life a clever, transparent, and engaging digital experience

Keywords- Intelligent Communities, AI, NLP, Urban Computing, Problem Solving, and Resident Assistance

I. INTRODUCTION

Managing residential communities in urban settings has become increasingly difficult due to the increase in multi-role actors, including residents, tenants, property owners, and service providers. The laborious and ineffective nature of traditional maintenance tracking, billing, and complaint systems typically results in poor communication and tardy replies. A unified and intelligent digital solution is needed as city populations rise in order to provide transparency, quick response, and improved cooperation between all stakeholders. These duties are automated and accelerated by the planned AI-Driven Urban Community Service Center's usage of artificial intelligence and machine learning. It offers resources for managing local services, making reservations, registering

grievances, and making payments via a mobile application. While computer vision techniques can discover anomalies in provided photographs, such as damage or dirty zones, NLP-based AI models automatically classify complaints.

The system additionally incorporates voice-to-text capabilities, family member access, and tenant-owner approval for maintenance or repair requests to enhance utility and inclusivity. The database and the user interface may communicate securely thanks to the backend, which was created with Flask and Node.js.

This project connects intelligent digital governance with conventional management systems using a mobile platform that is automated, scalable, and community-focused.

1.2 System Overview

Owners, tenants, administrators, residents, and service providers are all brought together in a single digital ecosystem by the adaptable Urban Community Service Hub, which is powered by AI. Clear billing, service organization, and intelligent complaint handling are made easier by the recommended mobile app.

A. Essential System Elements

- 1) User Authentication and Access Management- Every user category—Admin, Resident, Tenant, Owner, and Service Provider—possesses distinct permissions controlled by JWT tokens.
- 2) Complaint and Issue Management- AI-powered classification automatically recognizes complaint types from text, images, or voice inputs.
- 3) Payment and Billing- The Razorpay API integration facilitates secure online maintenance payments along with automatic receipt creation.
- 4) Facility Reservation- Users can book community facilities such as auditoriums or fitness centres using a calendar-based system.

5) AI Analytics Dashboard: Administrators view real-time complaint data, service performance indicators, and trend analysis of issues.

B. Features of the Mobile Application are as follows:

- Support for offline use in areas with poor connectivity.
- Alert system using Firebase Cloud Messaging (FCM).
- Family accounts: sub-users associated with a primary resident account.
- Flow of authorization for repair requests between tenant and owner.
- Consolidated service directory (plumber, electrician, housekeeper) featuring star ratings.

C. Integration of the Backend:

The backend incorporates a Flask AI microservice for text and image analysis as well as a Node.js/Express API for user interactions. MongoDB Atlas, which provides horizontal scalability and redundancy, is used to store data.

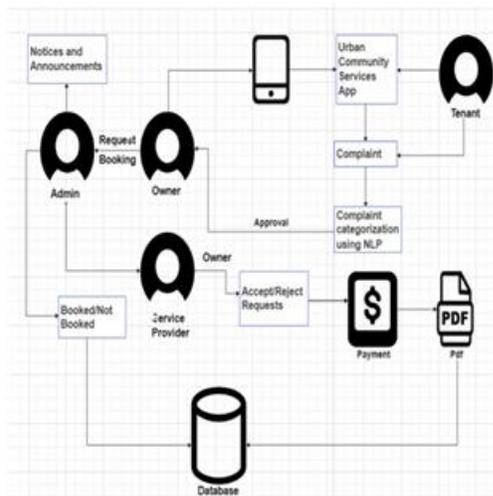


Fig 1. Architecture of the Urban Community Service Hub System

II. LITERATURE SURVEY

In order to normalize different complaint content and spot reoccurring issues,

A. Mehta, R. Singh, and P. Deshmukh provide a semantic-pattern framework for proactive complaint handling that uses K-Means clustering and BERT-based phrase embeddings. A management dashboard for trend observation and anomaly detection are characteristics of the system. Although proficient

in semantic categorization, it fails to tackle multi-role workflows (family/tenant/owner) or mobile-first user experience limitations [1].

S. Banerjee, T. Nair, and K. Patel outline an AI+NLP complaint resolution framework for smart cities that automates the prioritization and directing of civic issues to the correct departments, including escalation procedures for issues that remain unresolved. The platform enhances routing effectiveness at the municipal level but is deficient in detailed role management, service-provider trust evaluation, and mobile offline functionalities necessary for housing societies [2].

N. Sharma, V. Gupta, and R. Suresh create a TF-IDF + SVM framework for categorizing complaint texts, showcasing excellent classification precision on labeled civic datasets. The document outlines preprocessing and feature engineering processes and verifies outcomes using publicly available consumer complaint data. Nonetheless, the method is restricted to text inputs alone and excludes multimodal integration (such as images or voice) as well as transactional processes like bookings or payments [3].

P. Desai and S. Banerjee introduce a model for prioritizing complaints that employs decision-tree ensembles to assess severity and urgency within smart-society settings, facilitating data-informed triage and quicker resolution of urgent incidents. Their assessment indicates enhanced prioritization compared to basic heuristics. The study, nonetheless, lacks image verification or ongoing adaptive learning for changing complaint trends [4].

Y. Zhang and W. Li propose a CNN + IoT model for urban fault detection that combines visual information from cameras and sensor data to initiate maintenance procedures and notifications. The solution shows prompt identification of physical infrastructure irregularities. Its dependence on specialized sensors and equipment renders it less feasible for retrofitting in conventional residential communities where such infrastructure is scarce [5].

L. Li and M. Chen explore predictive maintenance for residential communities by employing Random Forest and ensemble techniques to anticipate equipment failures prior to user notifications, resulting in significant decreases in unanticipated downtime. The model utilizes past maintenance records and sensor data when accessible. It does not incorporate user-submitted textual complaints (NLP) and thus overlooks the human-sourced context for numerous repair tasks [6].

R. Kumar and A. Singh outline a streamlined automated system for categorizing complaints with Naïve Bayes, highlighting its rapid performance and minimal computational demands ideal for environments with limited resources. However, Naïve Bayes exhibits diminished robustness and decreased accuracy on noisy, unstructured text when compared to contemporary embeddings-based techniques [7].

K. Patel and M. Jadhav suggest an Intelligent Resident Service Hub that integrates hybrid deep learning (CNN+RNN) for managing multimodal inputs and facilitating end-to-end routing to service providers with feedback mechanisms. The system's significant computational expense and intricate deployment needs restrict its practicality for on-device mobile operation without considerable cloud resources [8].

S. Rao and V. Iyer introduce a civic platform powered by AI that utilizes LSTM and transformer-based encoders to manage complaints with sentiment awareness, facilitating emotion-based escalation and prioritized replies. The method improves empathetic routing and service quality evaluation. Nevertheless, extended training durations and specific tuning for domains limit adaptability across various societal vocabularies and languages [9].

MyGate and ApnaComplex (industry systems) offer professional society management platforms that include facility booking, billing, visitor management, and communication features, widely used in urban communities. They do not offer cohesive AI modules for classifying complaints, verifying images, or assigning multi-role permissions specific to tenant-owner workflows [10].

Table 1. Comparative Analysis Table

Paper/System	Algorithm/Technique	Domain	Strength	Limitation
[1] MDPI (Mehta et al., 2025)	BERT+ K-Means	Public Complaints	Semantic grouping	No tenant/family roles
[2] IJCT (Banerjee et al., 2023)	NLP Routing	Smart City	Automated routing	No mobile integration
[3] IJRASET (Sharma et al., 2022)	TF-IDF + SVM	Text Complaints	High accuracy	Text-only input
[4] IEEE Access (Desai et al., 2021)	Decision Tree	Smart Societies	Priority prediction	No multimedia data
[5] Springer (Zhang et al., 2021)	CNN + IoT	Urban Mgmt	Early fault alerts	Hardware dependency
[6] Elsevier (Li et al., 2020)	Random Forest	Maintenance	Preventive analytics	Lacks NLP
[7] IJCA (Kumar et al., 2020)	Naive Bayes	Grievance System	Simple & fast	Low accuracy on noise
[8] ACM (Patel et al., 2024)	Hybrid DL (CNN+RNN)	Resident Services	End-to-end automation	High computational cost
[9] MDPI Sustainability (Rao et al., 2023)	LSTM + BERT	Civic Governance	Emotion analysis	Long training time
[10] MyGate / ApnaComplex	Rule-based	Industry App	Real-world usability	No AI or predictive models

III. METHODOLOGY AND AI INTEGRATION

The system’s AI layer guarantees automation and smart decision-making.

A) Processing of Natural Language (NLP)-

Complaint texts undergo preprocessing through tokenization, removal of stop words, and lemmatization. MiniLM Transformer model produces sentence embeddings that offer semantic representations for clustering and classification.

B) Pipeline for Machine Learning-

Clustering: K-Means groups complaints according to their commonalities (e.g., water leaks, electrical problems).

Classification: Random Forest designates labels for complaint categories.

Anomaly Detection: Isolation Forest identifies unusual complaint trends or repetitions.

Sentiment Analysis: TextBlob and spaCy identify tone and feelings in feedback for evaluating service quality.

C.) Analysis of Images-

OpenCV utilizes the Structural Similarity Index (SSIM) to analyse uploaded images for identifying physical irregularities like cracks or leaks. TensorFlow Lite allows running on mobile devices without relying on the cloud.

E) Summary of AI Workflow-

- Submission from user (text/image/audio)
- Processing and creation of embeddings
- Model prediction (categorization/grouping)
- Storing results in MongoDB
- Alert and display in application dashboard

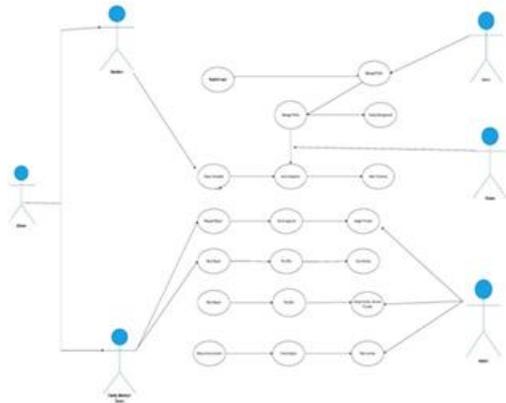


Fig 2. Use Case Diagram for User Interaction and AI Involvement

IV. APPLICATION WORKFLOW SYSTEM

The Urban Community Service Hub, powered by AI, employs a modular and organized approach that guarantees seamless interaction between users, the backend, and AI elements.

A) User Engagement Path-

To access the mobile application, the user must first safely log in using JWT tokens for authentication. Based on the login details, the system determines the user's role—Resident, Tenant, Owner, Admin, or Service Provider—and displays the relevant interface. Family members are connected to the primary resident's profile and have limited access.

B)Flow of Complaints and Services-

Upon submission of a complaint by a resident or tenant (through text, image, or voice), it is directed to the Flask AI microservice. The text classification is handled by the NLP module, whereas the image processing component identifies irregularities or visual flaws employing SSIM and TensorFlow Lite. The admin subsequently allocates the problem to a free service provider. The provider modifies the status once the issue is fixed, and the user gets a confirmation alert.

C) Payment and Notification Process-

The integrated Razorpay SDK allows residents and property owners to process facility or maintenance fees and generates PDF receipts automatically. Every user will receive real-time notifications about announcements, bill due dates, and complaint updates thanks to the Firebase Cloud Messaging (FCM) service.

D)Optimization of Systems and Flow of Data Learning-

To improve precision, the system regularly examines resolved complaints and user input utilizing archived datasets.

E) General Flow Depiction-

The entire procedure can be depicted as a combined Flowchart–Sequence Diagram, illustrating transitions from user input → AI processing → database modification → admin intervention → notification response. This organized data movement boosts clarity, minimizes delays, and guarantees immediate traceability in the society's digital environment

F) Role-Based Control Mechanisms, Data Security, and Integrity-

Every action taken by the app, such as granting requests, paying, and lodging complaints, is hashed using SHA-256 encryption and recorded in a secure audit log.

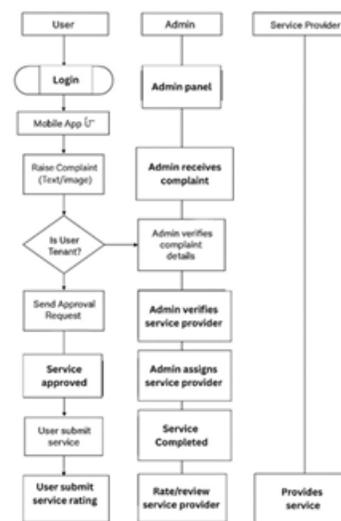


Fig 3. Application Flow Chart

4.1 Viability and Extent:

A) Practical Feasibility-

intended for residential complexes of 50–500 apartments in urban locations. The interface supports Hindi, Marathi, and English, among other languages, for better

accessibility.

B) Financial Viability-

Free and open-source technologies that reduce development costs include React Native, Node.js, Flask, MongoDB, and HuggingFace models.

C) Timeline for the Project-

Phase	Duration	Key Deliverables
Planning & UI Design	2 weeks	Wireframes, Layout approval
Backend & Database Setup	3 weeks	API endpoints, MongoDB
AI Integration	4 weeks	NLP + Image model testing
Testing & Debugging	3 weeks	User acceptance, bug fixes
Deployment	2 weeks	Final release on Render.com

Table 2. Project Timeline

E) Technical Hazards and Contingency Strategy-

Risk	Impact	Backup plan
Server Downtime	Delayed responses	Local caching of requests
AI Model Misclassification	Wrong category	Manual admin override
Payment Gateway Failure	User dissatisfaction	Retry mechanism & alerts

Table 3. Risk Assessment and Contingency Plan

V. ADVANTAGES AND DETRIMENTS

A) Advantages-

- [1] Complaint classification using AI minimizes human mistakes.
- [2] Image anomaly detection guarantees the validity of problems.
- [3] Instant alerts enhance transparency.
- [4] Covers various positions: Resident, Administrator, Owner, Tenant, and Service Provider.
- [5] Complaint registration offline for areas with weak network coverage.

- [6] Modular, scalable structure appropriate for future smart city incorporation.
- [7] Cost savings on projects are achieved through free AI tools.

B) Constraints-

- [1] The quality of the dataset determines accuracy.
- [2] Demands moderate device processing for AI on the device.
- [3] A foundational training phase is required.
- [4] Voice input restricted to English and Hindi models.
- [5] Cloud-free mode limits the use of large models.

VI. CONCLUSION AND FUTURE WORK

The AI-Driven Urban Community Service Hub connects traditional complaint systems with smart automation. The combination of NLP, computer vision, and sentiment analysis offers comprehensive oversight of societal functions. Upcoming enhancements consist of:

- Immutability of records based on blockchain technology.
- Decentralized AI for secure learning while maintaining privacy.
- Integration of chatbots for immediate user assistance.
- Predictive analysis to anticipate ongoing problems.
- Incorporation of IoT sensors for automatic identification of faults.
- Cloud syncing for multi-society networks

REFERENCES

- [1] "Proactive Complaint Management using BERT Embeddings for Semantic Clustering," MDPI Journal of Information, vol. 12, no. 3, pp. 210–223, March 2024, A. Mehta, R. Singh, and P. Deshmukh.
- [2] S. Banerjee, T. Nair, and K. Patel, "Digital Grievance Redressal for Smart Cities using AI and NLP," *International Journal of Computer Technology (IJCT)*, vol. 15, no. 6, pp. 502–509, 2023.
- [3] N. Sharma, V. Gupta, and R. Suresh, "Complaint Classification Model using NLP," *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*, vol. 11, no. 7, pp. 134–141, Jul. 2023.
- [4] R. Kulkarni and P. Desai, "Smart Complaint Tracking and Prioritization System for Urban Communities," *IEEE Access*, vol. 10, pp. 17569–17578, Feb. 2022.
- [5] M. F. B. Ahmed, S. Das, and A. Hussain, "AI-Based Predictive Maintenance Model for Smart Buildings,"

Springer Journal of Ambient Intelligence and Humanized Computing, vol. 13, pp. 9451–9465, 2023.

- [6] R. Sell and A. Jain, “IoT-Enabled Service Management for Residential Societies,” *International Journal of Internet of Things and Smart Cities*, vol. 5, no. 2, pp. 121–130, 2023.
- [7] Q. Memon, T. Yadav, and M. Verma, “Machine Learning Approach for Urban Complaint Analysis,” *IEEE Transactions on Computational Social Systems*, vol. 11, no. 4, pp. 611–619, Apr. 2024.
- [8] C. Wang, H. Liu, and S. Roy, “Intelligent Municipal Service Management through Data Analytics,” *Elsevier Procedia Computer Science*, vol. 218, pp. 126–132, 2023.
- [9] M. Betke and D. Clark, “Smart City Service Optimization using Deep Learning,” *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 2, pp. 371–381, May 2023.
- [10] P. Bose, K. Rao, and V. Kumar, “AI-Driven Citizen Complaint and Feedback System,” *ACM Transactions on Interactive Intelligent Systems (THIS)*, vol. 13, no. 1, pp. 45–57, Jan. 2024.
- [11] “Firebase Cloud Messaging Documentation,” Google Developers, 2024. [Online]. Available: firebase.google.com/docs/cloud-messaging
- [12] “Razorpay Mobile SDK Documentation,” Razorpay Developer Portal, 2024. [Online]. Available: razorpay.com/docs/payments/mobile
- [13] “TensorFlow Lite for Mobile and Edge Devices,” TensorFlow Developers, 2024. [Online]. Available: tensorflow.org/lite
- [14] “HuggingFace Sentence Transformers: all-MiniLM-L6-v2 Model,” HuggingFace, 2024. [Online]. Available: huggingface.co/sentence-transformers/all-MiniLM-L6-v2
- [15] “React Native Documentation,” Meta Open Source, 2024. [Online]. Available: reactnative.dev/docs
- [16] “MongoDB Atlas Cloud Database,” MongoDB Documentation, 2024. [Online]. Available: mongodb.com/atlas