An Intelligent Urban Civic Complaint Management System with Automated Classification, Smart Worker Assignment, Escalation Mechanisms, and Resolution Tracking

Dr. Priya S, Arvind M, Mukhil Amsath P K, Nithivarsha T P, Ranjith Kumar A
Associate Professor, Department of Computer Science and Engineering
Student, Department of Computer Science and Engineering

Abstract—**Urban municipalities today are grappling with a persistent problem: grievance systems that simply do not keep up with the pace of city life. Delayed responses, poor coordination between departments, and a general lack of accountability have frustrated both citizens and administrators alike. This paper introduces a practical solution — an AI-powered Urban Civic Complaint Management System built around the convergence of mobile technology, cloud computing, and intelligent automation. Through a smartphone application, residents can report problems by attaching a photo, a brief description, and their GPS location. Engineers manage the entire complaint journey through role-specific dashboards, while an automated escalation engine ensures that unresolved issues are pushed up the chain of command based on predefined Service Level Agreement (SLA) timelines. The system uses DistilBERT, a compact transformer model, for zero-shot complaint classification, and filters out duplicate submissions by combining semantic similarity checks with geographic proximity calculations via the Haversine formula. Senior administrators get access to real-time dashboards that translate raw complaint data into meaningful operational insights. Testing confirms that this integrated approach genuinely improves response times, brings greater transparency to the resolution process, and raises citizen satisfaction levels in urban service management.**

Index Terms—Civic Complaint Management, DistilBERT, Escalation Management, Firebase, Flutter, Smart Governance

I. INTRODUCTION

Cities are growing faster than ever, and the pressure on public infrastructure has never been higher. Roads wear out, power lines fail, water pipes burst — and when they do, residents need a reliable way to report the problem and get it fixed. Traditional reporting mechanisms, whether through telephone helplines or paper-based complaint registers, have long struggled to keep pace with these demands. They introduce unnecessary delays, leave citizens with no visibility into what is happening with their complaint, and often result in issues simply falling through the cracks.

The answer to this challenge lies in the technology that most urban residents already carry in their pockets. Mobile applications and cloud platforms offer a fundamentally better way to handle civic complaints — faster reporting, real-time status tracking, and the kind of large-scale coordination that was simply impossible with manual processes. While the system described here is designed to work across multiple public service domains, its first deployment focuses on the Electricity Sector. This was a deliberate choice: electricity operations involve complex multi-tiered oversight structures, with Assistant Engineers (AE) and Executive Engineers (EE) sharing responsibility for different aspects of fault management. Getting this hierarchy right in a digital system is a meaningful engineering challenge, and solving it here lays the groundwork for broader rollouts.

II. RELATED WORK

Research into AI-assisted complaint and grievance management has been steadily building over the past few years. Sheikh et al. [1] developed an NLP-driven complaint registration platform capable of automatically sorting incoming submissions into predefined fault categories — an early demonstration that text classification could meaningfully reduce manual triage work. In parallel, Wagh et al. [2] took a data analytics approach, designing a grievance system that mines historical records to uncover recurring patterns and help administrators plan more effective responses.

Chatbot-based solutions have also gained traction. Jadhav et al. [3] proposed a system where machine learning classifiers work alongside conversational agents, reducing the need for direct human intervention in citizen-facing interactions. On the administrative side, Kavitha and Amutha [4] developed an AI-assisted platform that strengthens classification accuracy while actively supporting engineering decision-making. Sinha and Pal [5] took a different angle, applying sentiment analysis to complaint data so that urgent cases could be distinguished from routine ones and prioritised accordingly. Kalbande and Shinde [6] specifically focused on automated classification pipelines designed to speed up the processing of urban complaints reported by citizens.

Despite the progress these contributions represent, a common thread runs through most existing solutions: they manage individual aspects of the complaint process well but lack an integrated, end-to-end approach. Structured escalation policies and dynamic workload-aware worker allocation, in particular, remain underexplored. The system presented here closes that gap by bringing AI-driven classification, SLA-governed escalation, and intelligent task assignment together within a single, cohesive cloud platform.

III. PROBLEM STATEMENT

Today's civic complaint systems carry the weight of outdated processes. Fault reports are still being logged by hand and passed along through informal channels, producing the predictable results: poor traceability, uneven distribution of work among field staff, and resolution timelines that test the patience of citizens and erode their confidence in public institutions.

Perhaps the most critical gap is the absence of any formal escalation mechanism. When a complaint sits unresolved beyond what any reasonable person would consider an acceptable timeframe, there is no automatic process to bring it to the attention of a more senior official. Everything depends on someone remembering to follow up. This single shortcoming is enough to allow serious infrastructure faults to linger far longer than they should, and it undermines the perception of accountability that good governance requires.

IV. PROPOSED SYSTEM

The system introduced in this paper is a full-featured digital platform for managing the entire complaint lifecycle, initially deployed within Electricity Board (EB) operations. Rather than serving one group in isolation, it connects residents, field workers, and administrators at every level — Assistant Engineers (AE), Assistant Executive Engineers (AEE), and Executive Engineers (EE) — within a single unified environment that takes care of complaint intake, routing, and resolution with minimal manual effort.

A. Complaint Submission and Geospatial Mapping

When a resident spots an electrical fault — a supply outage, a damaged transformer, or a billing irregularity — the mobile application walks them through a straightforward reporting process. They provide a text description, attach a photograph, and the app automatically records their GPS coordinates. What sets this system apart is its Geospatial Section Mapping Service, which uses Point-in-Polygon spatial algorithms to match the complainant's +location against a GeoJSON-encoded map of Electricity Board jurisdictional boundaries. This means every complaint lands directly in the inbox of the right section office — no manual routing, no guesswork.

B. Role-Based Administration and Hierarchy

The platform mirrors the actual administrative hierarchy of the Electricity Board, giving each level of official exactly the view they need. Assistant Engineers receive immediate notifications for complaints in their territory and coordinate the deployment of Wiremen and Line Inspectors. Assistant Executive Engineers and Executive Engineers, meanwhile, have visibility across multiple sections — tracking aggregate performance, outstanding cases, and the overall operational health of the areas under their supervision.

C. AI-Driven Classification and Prioritisation

Complaints are classified automatically using DistilBERT, a lightweight transformer model that performs zero-shot inference on the submitted text. It assigns each complaint to one of three fault categories: Safety Hazards (such as exposed live wires), Infrastructure Failures (such as transformer breakdowns), or Service Requests (such as meter-related concerns). On top of this, each complaint receives a Criticality Score that captures its urgency — ensuring that safety-critical faults reach the top of the processing queue ahead of routine maintenance tasks.

D. Automated Escalation and Resolution Tracking

The platform's escalation engine is what gives it teeth. Governed by Service Level Agreements, it watches every complaint against its resolution deadline. If a high-priority complaint — say, a power outage — is neither resolved by the assigned lineman nor acknowledged by the responsible AE within the stipulated window, the system automatically elevates it to the AEE. If it remains unaddressed, it escalates again to the EE. Every step of this journey, from the initial dispatch of a field worker through to final power restoration confirmation, is recorded as an auditable trail.

V. SYSTEM MODULES

The platform is structured around five functional modules, each owning a distinct phase of the complaint management lifecycle.

1. Citizen Complaint Module

This is the front door of the system — the interface through which residents report faults. Using the mobile app, they submit a description, a photograph, and their location. The record is saved to the cloud database immediately, and citizens can check on their complaint's progress at any point without having to call anyone.

2. Complaint Management Module

This module gives engineers the tools they need to do their job well. They can inspect submitted complaints, verify the reported details, and track cases from the moment they arrive through to final resolution. A unified view of open, in-progress, and resolved complaints helps the engineering team stay coordinated and avoid duplication of effort.

3. Worker Assignment Module

Rather than leaving task allocation to chance or individual judgment, this module automates it. Field assignments are distributed based on current workload, so no single operative ends up swamped while others sit idle. The result is faster on-site attendance and a more equitable distribution of responsibility across the team.

4. Escalation Module

This module is the system's accountability backbone. It continuously monitors each complaint against its SLA threshold and, when a deadline is breached, automatically elevates the case through the management hierarchy — from AE to AEE to EE as required. No complaint can fall through the cracks indefinitely.

5. Analytics Module

The Analytics Module turns operational data into actionable intelligence. Dashboards display complaint volumes, average resolution times, and individual worker performance indicators. For administrators, this is the difference between reacting to problems and anticipating them.

VI. TECHNOLOGY STACK

- Frontend: Flutter Mobile Application
- Backend: Python Processing Services
- Database: Firebase Firestore
- Authentication: Firebase Authentication
- Cloud Infrastructure: Firebase Cloud Platform
- Mapping Services: GPS-based Location APIs

VII. SYSTEM ARCHITECTURE

The platform follows a three-tier architecture with a clearly defined presentation layer, application layer, and data layer. Each tier carries specific responsibilities and interacts with the others through well-defined interfaces.

Presentation Layer

Citizens interact with the system through a Flutter mobile application that handles fault submission and progress tracking. Engineers access their tools through web-based dashboards that display complaint queues, assignment controls, and status updates.

Application Layer

This is where the system's intelligence lives. The application layer handles fault classification through DistilBERT, manages worker assignments, enforces the SLA-based escalation rules, and coordinates the complete complaint tracking pipeline.

Data Layer

Firebase Firestore provides the persistence backbone, storing complaint records, user profiles, assignment histories, and resolution states. Changes propagate to all connected clients in real time, keeping every stakeholder working from the same picture.

Supporting services such as GPS geocoding and file storage are handled by external cloud providers. This keeps the core platform lean while still benefiting from specialist infrastructure where it matters.

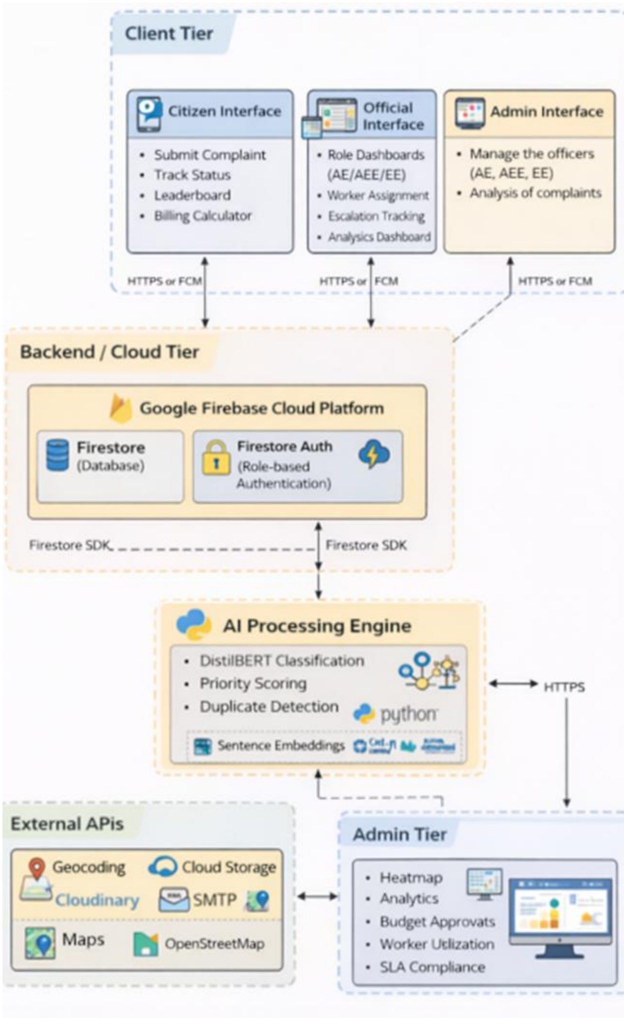

Fig. 1. System Architecture Diagram

VIII. IMPLEMENTATION

The citizen-facing interface was built with Flutter, which allows a single codebase to run on Android devices without compromise. Firebase Firestore was chosen as the cloud database for its low-latency, real-time synchronisation capabilities, keeping the mobile app and backend in constant alignment. User authentication — for both residents and engineering personnel — is handled by Firebase Authentication, which ensures that complaint records remain accessible only to the right people.

Backend services responsible for AI inference and business logic are written in Python, a natural choice given the maturity of its machine learning ecosystem. Bringing these pieces together — Flutter on the client side, Firebase for data management, and Python for intelligent processing — produces an architecture that scales well, remains maintainable over time, and can absorb the kind of complaint volumes a real city generates.

DistilBERT inference runs through the Hugging Face Transformers library, while sentence-level embeddings for semantic comparison are generated using the SentenceTransformers toolkit, with PyTorch as the underlying deep learning runtime. Python backend scripts listen continuously for new Firestore entries and, upon detecting one, invoke the relevant models and write the enriched results — category, priority level, duplicate flag, and confidence score — back to the database in near real time.

IX. AI-BASED COMPLAINT CLASSIFICATION

The intelligence layer of this platform is built around a multi-stage AI pipeline designed to process each incoming complaint with as little human involvement as possible. Zero-shot classification via DistilBERT allows the system to categorise free-text complaint descriptions without needing a labelled training dataset — a significant practical advantage in civic settings where clean, annotated data is rarely available. The supported categories cover electricity supply faults, road surface damage, sanitation deficiencies, and water distribution failures.

Before any text reaches the model, it goes through a normalisation step: everything is converted to lowercase, and URLs, punctuation, and other non-semantic tokens are stripped out. The cleaned text is then fed into DistilBERT, which returns a ranked list of candidate categories along with confidence probabilities.

Urgency scoring picks up where classification leaves off. A rule-based weighting scheme considers factors such as the presence of hazard-related keywords, the severity of the affected infrastructure, and how frequently similar complaints have been raised in the same area. The resulting score is bucketed into three priority tiers — High, Medium, and Low — which directly influences the order in which complaints appear in the engineer's queue.

Duplicate detection draws on the Sentence Transformer model (all-MiniLM-L6-v2) to encode complaint descriptions as dense semantic vectors. Cosine similarity scores between vectors identify submissions that are saying essentially the same thing. At the same time, geographic deduplication is achieved by computing physical distances between complaint coordinates using the Haversine formula. A submission is flagged as a probable duplicate when both its semantic similarity and its spatial proximity to an existing complaint exceed configurable thresholds.

Two additional filters round out the quality control layer. A spam detection routine catches bursts of identical complaints sent by a single user within a short time window. Conversely, a risk-zone detector aggregates genuinely distinct complaints from multiple users within a geographic cluster, surfacing potential infrastructure hotspots for proactive attention before they escalate into larger failures.

The entire pipeline is triggered by a persistent Firebase Firestore event listener attached to the complaints collection. The moment a new complaint document is detected, the listener hands the record to the processing pipeline, and the enriched output — fault category, priority level, duplicate flag, and model confidence — is written back to the record in near real time.

A. AI Processing Pipeline

As soon as a complaint is submitted, the data is committed to Firebase Firestore. A dedicated backend listener detects the insertion event and kicks off the processing sequence, which runs through the following stages in order:

- Text Preprocessing — Normalisation and sanitisation of the raw complaint text to remove noise and standardise formatting.
- Fault Classification — Zero-shot category inference via DistilBERT, assigning each complaint to the most appropriate fault domain.
- Urgency Scoring — Rule-based heuristic assignment of a High, Medium, or Low priority tier based on keywords and contextual signals.
- Duplicate Detection — Semantic vector comparison using SentenceTransformer embeddings, combined with Haversine-based spatial proximity checking.
- Spam Filtering — Rate-based suppression of repetitive submissions from individual users within short time windows.

Once all stages are complete, the complaint record is annotated with the inferred category, priority tier, and confidence score, preparing it for efficient triage by the assigned engineer.

X. COMPLAINT RESOLUTION WORKFLOW

Step 1 - Resident submits a fault report through the mobile application.

Step 2 - The AI pipeline classifies the complaint and assigns a priority tier.

Step 3 - The Assistant Engineer reviews the case and dispatches a field operative.

Step 4 - The field operative attends the site and carries out the remedial work.

Step 5 - The engineer inspects and validates the completed repair.

Step 6 - The resident confirms the resolution and submits an experience rating.

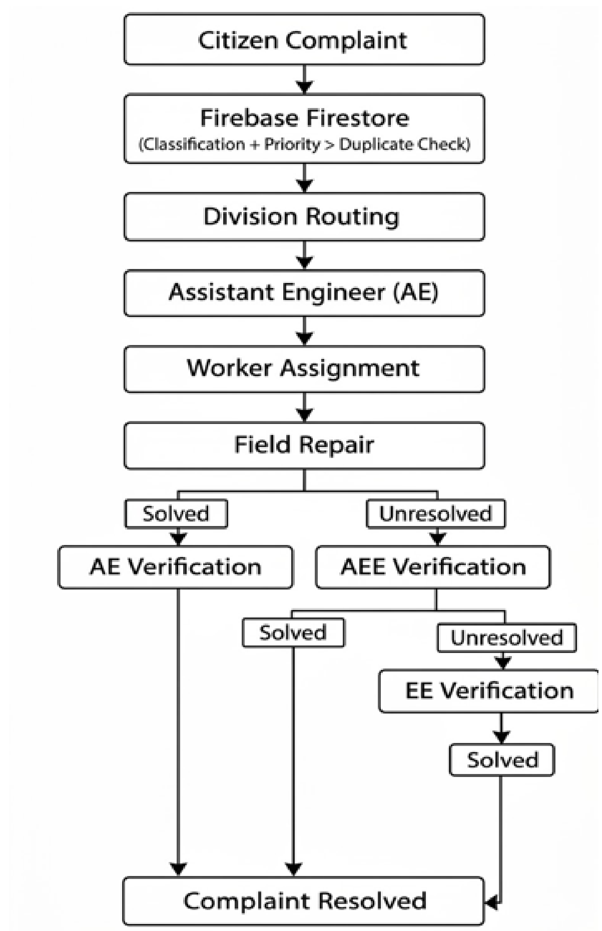

Fig. 2. Complaint Resolution Workflow

XI. ADVANTAGES OF PROPOSED SYSTEM

Accelerated Complaint Processing — Direct mobile submission removes the queuing delays and transcription errors that come with telephone or counter-based reporting. Citizens can log a fault in under a minute, and the complaint is in the system immediately.

Enhanced Accountability — Every complaint carries a unique identifier and is tracked through each stage of its lifecycle. Citizens have verifiable proof that their submission is being acted upon, which in itself is a meaningful improvement over the opacity of traditional systems.

Proactive Escalation — The SLA engine does not wait for someone to remember to follow up. When a complaint stalls, it is automatically escalated to the appropriate authority, which means critical infrastructure faults receive executive attention even when frontline staff are stretched.

Optimised Resource Utilisation — Workload-aware assignment logic distributes tasks more evenly across field workers, preventing the twin problems of overloading some operatives while others are underutilised. The outcome is more consistent response times across all sections.

Evidence-Based Administration — Real-time dashboards convert raw complaint data into performance metrics that administrators can actually use. Decisions about staffing, infrastructure investment, and operational priorities can be grounded in evidence rather than intuition.

XII. FUTURE WORK

Cross-Departmental Expansion

The modular design of this platform means it is ready to grow. Bringing in additional civic departments — Water Works, Public Works, Urban Sanitation — is a natural next step that would give citizens a single touchpoint for all their complaints, regardless of which public service is involved. The coordination benefits of this kind of integration across departments would be substantial.

Predictive Fault Detection

Looking further ahead, time-series analysis and machine learning models trained on historical complaint data could allow the system to spot recurring patterns and anticipate failures before they happen. A power outage that affects the same neighbourhood every monsoon season, for instance, should not require repeated complaints to trigger a response. Moving from reactive to proactive maintenance would reduce downtime and make better use of limited resources.

IoT and Smart Grid Integration

Sensors embedded in electrical infrastructure can monitor parameters like voltage, current, and equipment health in real time. When anomalies are detected, the system could automatically generate and log a complaint without any citizen intervention. This would dramatically reduce reporting delays and, in some cases, allow faults to be addressed before they affect service at all.

Regional Language Support

For the platform to genuinely serve all citizens, it needs to speak their language. Extending NLP capabilities to support Tamil and other regional languages would significantly broaden adoption among non-English-speaking populations and make digital grievance redressal a more equitable service.

AI-Optimised Field Routing

Incorporating real-time geolocation data of field workers into the assignment algorithm would allow the system to dispatch the nearest available operative rather than simply the least busy one. Factoring in urgency levels and real-world traffic conditions could further reduce response times and improve workforce utilisation across the board.

Smart City Federation

As urban environments become more instrumented, the complaint management platform could become one node within a broader municipal IoT network. Seamless data exchange between urban systems would support a more holistic view of city operations, enabling inter-departmental coordination and more resilient infrastructure planning.

XIII. RESULTS AND DISCUSSION

Functional testing of the system was carried out using a curated dataset of synthetic complaints designed to reflect realistic fault scenarios across power supply disruptions, road surface deterioration, and water delivery failures. The DistilBERT classification engine consistently assigned complaints to the correct fault categories, which validates the practical case for zero-shot inference in civic applications where properly labelled training data is hard to come by.

The deduplication subsystem reliably caught co-located, semantically similar submissions, preventing unnecessary duplicate work orders from being sent to field personnel. Worker assignment performed as expected, distributing tasks in a balanced way that kept individual workloads within reasonable bounds. The SLA-based escalation engine correctly triggered upward referrals in every test case where a resolution deadline was breached. Taken together, these results demonstrate that the proposed platform delivers genuine improvements in complaint throughput, process transparency, and cross-level administrative coordination.

XIV. CONCLUSION

The Intelligent Urban Civic Complaint Management System developed through this work offers a technically sound and practically deployable framework for bringing municipal grievance redressal into the digital age. By combining a Flutter-++based mobile interface with Firebase cloud services and a Python-driven AI engine, the platform automates the full complaint lifecycle — from structured intake and intelligent classification through to SLA-enforced escalation and resolution confirmation. The results show that this integrated approach meaningfully reduces administrative overhead, strengthens the channel of communication between citizens and authorities, and creates an accountable, auditable governance process that can be adapted to a wide range of urban service domains.

REFERENCES

[1] Sheikh, A., Iqbal, A., et al., “AI-Based Complaint Registration and Sorting System using NLP and Text Classification,” 2023.

[2] Wagh, S., et al., “AI Based Predictive Grievance Management System with Graphical Analysis,” 2023.

[3] Jadhav, V., et al., “Smart Complaint Management System using Machine Learning and Chatbot,” 2022.

[4] Kavitha, S., and Amutha, B., “Complaint Management System using Artificial Intelligence,” 2022.

[5] Sinha, A., and Pal, S., “An AI-Driven Approach to Grievance Redressal Systems with Sentiment-Based Prioritization,” 2021.

[6] Kalbande, D., and Shinde, A., “Smart Citizen Complaint System using AI Based Classification,” 2021.

[7] Sharma, K., et al., “Automated Complaint Tracking and Resolution using AI and Cloud Infrastructure,” 2021.

[8] Patil, N., and More, S., “Design of Grievance Redressal System using Predictive Analytics,” 2020.

[9] Sahu, S., et al., “A Novel Approach to Complaint Analysis using Text Mining and NLP,” 2020.

[10] Kulkarni, A., et al., “AI-Based Grievance Management System,” 2020.