iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** III **Month of publication:** March 2026

DOI: https://doi.org/10.22214/ijraset.2026.78873

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue III Mar 2026- Available at www.ijraset.com

A Gamified Civic Engagement Platform for Urban Complaint Management, Community Mobilization and AI-Powered Civic Intelligence

Dr. Sonia Bajaj[1], Mr. Ashish Wankhade[2], Mr. Sanket Dandekar[3], Mr. Aryan Khandagale[4], Ms. Aastha Khangar[5], Ms. Anamika Tirpude[6], Ms. Monali Atone[7]
Department of Computer Science and Engineering, G.H. Raisoni University Amravati

Abstract: ***Urban municipalities in India face persistent challenges in managing citizen complaints transparently, verifying community participation, and making government welfare information accessible. This paper presents a unified MERN stack web platform addressing these gaps through three integrated modules: a nine-step complaint lifecycle with WhatsApp-based field worker coordination and Haversine-formula duplicate detection within a 50-metre geospatial radius; a three-layer community mission attendance system combining cryptographic QR tokens, GPS validation, and pre-registration checks to eliminate proxy attendance; and a Google Gemini-powered civic assistant with domain-restricted prompting for natural language access to government scheme information. A cross-role gamification engine awards experience points for complaints, mission participation, and resolution ratings, with daily limits and cooldown controls preventing misuse. Role-based access control governs five user tiers citizen, NGO administrator, local authority, field worker, and super administrator across a three-tier architecture. Functional validation confirmed reliable performance across all modules, and security testing verified resilience against common web vulnerabilities, demonstrating that accountable complaint management, verifiable civic engagement, and conversational AI assistance can be feasibly unified into a single deployable urban governance platform.***

Keywords: Civic Technology, Complaint Management, Gamification; XP Engine, QR-GPS Attendance, Large Language Model, MERN Stack, Urban Governance, Role-Based Access Control, Smart City

I. INTRODUCTION

The rapid growth of urban populations across developing countries has created a significant gap between what citizens need from their local governments and what those governments are actually able to deliver. Municipal bodies are responsible for handling thousands of complaints every day broken roads, water supply failures, garbage accumulation, illegal constructions yet most of them still rely on outdated, manual, or poorly designed digital systems that offer no real transparency to the citizen who filed the complaint. The problem is not just about complaints. There are three distinct gaps that exist in almost every urban civic system today, and they feed into each other.

The first is the complaint redressal gap. A citizen files a complaint through a government portal or helpline. After that, nothing happens visibly. There is no update on whether the complaint was assigned to someone, whether a field worker visited the site, or whether the issue was actually fixed. This lack of visibility frustrates citizens and reduces trust in the system. Over time, people simply stop reporting problems because they believe nothing will be done anyway.

The second is the community participation gap. Many non-governmental organizations and resident groups organize civic activities cleanliness drives, tree plantation campaigns, awareness programs. These events depend on volunteers, but there is no reliable way to confirm who actually showed up, verify their presence at the correct location, or reward them in any meaningful way. Attendance registers can be faked. Photo uploads carry no location data. As a result, genuine volunteers go unrecognized and participation slowly declines.

The third is the information access gap. Governments at the central and state level run hundreds of welfare schemes for housing, sanitation, employment, and infrastructure. Most eligible citizens never benefit from these schemes simply because they do not know they exist or do not understand how to apply. A 2021 NITI Aayog report found that over 65 percent of eligible beneficiaries in tier-2 cities were unaware of schemes they qualified for [1]. Government websites are complex, written in technical language, and offer no conversational way for an ordinary person to ask a simple question and get a clear answer.

©IJRASET: All Rights are Reserved

A. *Why This Project Is Needed*

Existing systems address each of these problems separately and incompletely. Complaint portals accept grievances but provide no field-worker coordination or proof-of-work mechanism. Community event tools have no GPS-verified attendance. Gamification platforms exist in isolation without any connection to real civic workflows. AI chatbots for government information are either too generic or too expensive to build and maintain. No single platform brings all of this together in a way that reflects how municipal systems actually work with multiple types of users, each needing a completely different set of tools and permissions.

B. *Real-World Importance*

Urban governance is not just a technology problem it is a trust problem. When citizens see their complaints resolved, they engage more. When volunteers are recognized, they come back. When people can find out about a welfare scheme through a simple chat conversation, they benefit from it. A platform that connects all of these outcomes in one place has the potential to meaningfully improve the relationship between citizens and their local government.

The WhatsApp integration in this system carries particular importance. Field workers and lower-income citizens who cannot navigate a complex web application can still participate fully through WhatsApp and SMS channels that are already familiar to over 500 million people in India [2]. This makes the platform genuinely inclusive, not just technically capable.

C. *Objectives*

This project was developed with the following clear objectives:

1) Design a structured nine-step complaint lifecycle where every action from filing to resolution is tracked, accountable, and visible to all parties involved.
2) Build a three-layer QR and GPS based attendance verification system for community missions that makes fake or proxy attendance impossible.
3) Develop a gamification engine that awards experience points for civic actions, with built-in controls to prevent misuse and encourage genuine long-term participation.
4) Integrate an AI-powered civic assistant using a large language model, restricted through system prompting to answer questions only about government schemes, civic rights, and platform usage.
5) Enforce role-based access control across five distinct user types citizen, NGO administrator, local authority, field worker, and super administrator each with their own dedicated interface and permissions.
6) Validate the complete system through structured functional testing, performance measurement, and security evaluation.

II. LITERATURE REVIEW

Researchers and developers have explored various aspects of civic technology, complaint management, gamification, and AI-assisted public services extensively between 2021 and 2025. This section reviews the most relevant works from this period and identifies the gaps that this platform addresses.

Kumar and Singh [3] conducted a study in 2022 examining the effectiveness of existing e-governance complaint portals in Indian municipalities. Their work found that while most urban local bodies had adopted some form of digital grievance submission, over 70 percent of complaints received no status update after initial filing. The study highlighted that the absence of a structured lifecycle where each step is tracked and assigned to a responsible actor was the primary reason for low citizen satisfaction. The platform proposed in this paper directly addresses this by implementing a nine-step complaint pipeline where every state transition is recorded and visible to all stakeholders.

Zhao et al. [4] published research in 2023 on the use of gamification techniques in public participation platforms in smart city environments across China and Southeast Asia. Their findings confirmed that citizens who received visible rewards points, badges, and leaderboard rankings for civic actions were significantly more likely to engage repeatedly over a six-month period compared to users of platforms with no reward mechanism. However, their system had no anti-abuse controls, meaning users could earn points repeatedly by performing the same action in quick succession. The gamification engine in this paper solves this through configurable cooldown timers, daily earning limits, and an admin-controlled audit ledger.

Gupta et al. [5] proposed a QR code-based volunteer attendance system for NGO events in 2022. The system reduced manual paperwork significantly but suffered from a well-documented vulnerability, volunteers could share the QR code image with others who were not physically present, and the system had no way to detect this.

The attendance verification module in this paper closes this gap through a three-layer gate that checks the cryptographic token, the GPS coordinates of the scanning device, and the user's prior registration status all three must pass simultaneously before attendance is recorded.

Rao and Mehta [6] examined WhatsApp as a service delivery channel for municipal field operations in 2023. Their study documented how informal WhatsApp groups were already being used by local government field staff to receive task assignments, but noted that this approach had no accountability messages could be ignored, tasks could go unacknowledged, and there was no proof-of-work requirement. The field worker coordination module in this paper formalizes this workflow using the WhatsApp Business API, where workers receive a structured task dispatch message and must upload photographic proof through a tokenized link before the complaint can be marked as resolved.

Verma et al. [7] studied government scheme awareness among urban citizens in 2024 and found that even when schemes were actively publicized, uptake remained low because citizens could not easily determine their own eligibility or understand the application process. The study recommended conversational interfaces as the most effective solution for bridging this gap. The AI civic assistant in this paper implements exactly this approach a large language model restricted through server-side system prompting to answer questions about government schemes, civic rights, and platform guidance in plain, simple language.

Patel and Joshi [8] explored the use of large language models for citizen-facing government services in 2024. Their research demonstrated that domain-scoped LLM deployments where the model is constrained to a specific knowledge area through prompt engineering produced significantly more accurate and trustworthy responses than general-purpose chatbots. Crucially, their work showed this could be achieved without custom model training, making it practical for resource-constrained civic deployments. This finding directly supports the design decision in this paper to use system prompt scoping rather than fine-tuning for the civic assistant.

Sharma et al. [9] published a 2025 review of role-based access control implementations in multi-stakeholder civic platforms. Their analysis found that platforms which failed to enforce strict role separation where different user types have clearly defined permissions and cannot access each other's workflows were significantly more vulnerable to data leakage and misuse. They recommended that civic platforms define roles at the API route level rather than only at the interface level. This paper follows that recommendation by enforcing role middleware on every protected backend route, with local authority queries automatically scoped to the authenticated user's assigned area.

A. Gap Analysis

While each of the above works makes a valuable contribution to one aspect of civic technology, none of them addresses the complete problem. Complaint portals track filing but ignore field worker accountability. Gamification systems reward engagement but lack abuse prevention. QR attendance tools verify presence but not location. WhatsApp tools broadcast messages but do not create a formal proof-of-work loop. AI assistants answer questions but exist separately from any civic workflow. Role-based systems secure access but are not connected to complaint or mission management. This platform is the first to combine all of these capabilities structured complaint lifecycle, verified attendance, configurable gamification, WhatsApp field coordination, AI civic assistance, and role-aware access control into a single unified system designed around how urban civic operations actually work.

Table I — Comparison of Existing Systems with Proposed Platform

Study	Year	Complaint Tracking	Verified Attendance	Gamification	AI Assistant	Field Worker Flow
Kumar and Singh [3]	2022	Partial	No	No	No	No
Zhao et al. [4]	2023	No	No	Partial	No	No
Gupta et al. [5]	2022	No	Partial	No	No	No
Rao and Mehta [6]	2023	No	No	No	No	Partial
Verma et al. [7]	2024	No	No	No	Partial	No
Patel and Joshi [8]	2024	No	No	No	Yes	No
Sharma et al. [9]	2025	No	No	No	No	No
Proposed platform	2025	Yes	Yes	Yes	Yes	Yes

III.METHODOLOGY

A. System Architecture

The platform is built on a three-tier client-server architecture, divided into three clearly separated layers the frontend that the user sees, the backend that processes all logic, and the database that stores all data. These three layers communicate with each other over HTTPS, and each layer has a specific responsibility. This separation makes the system easier to maintain, scale, and secure.

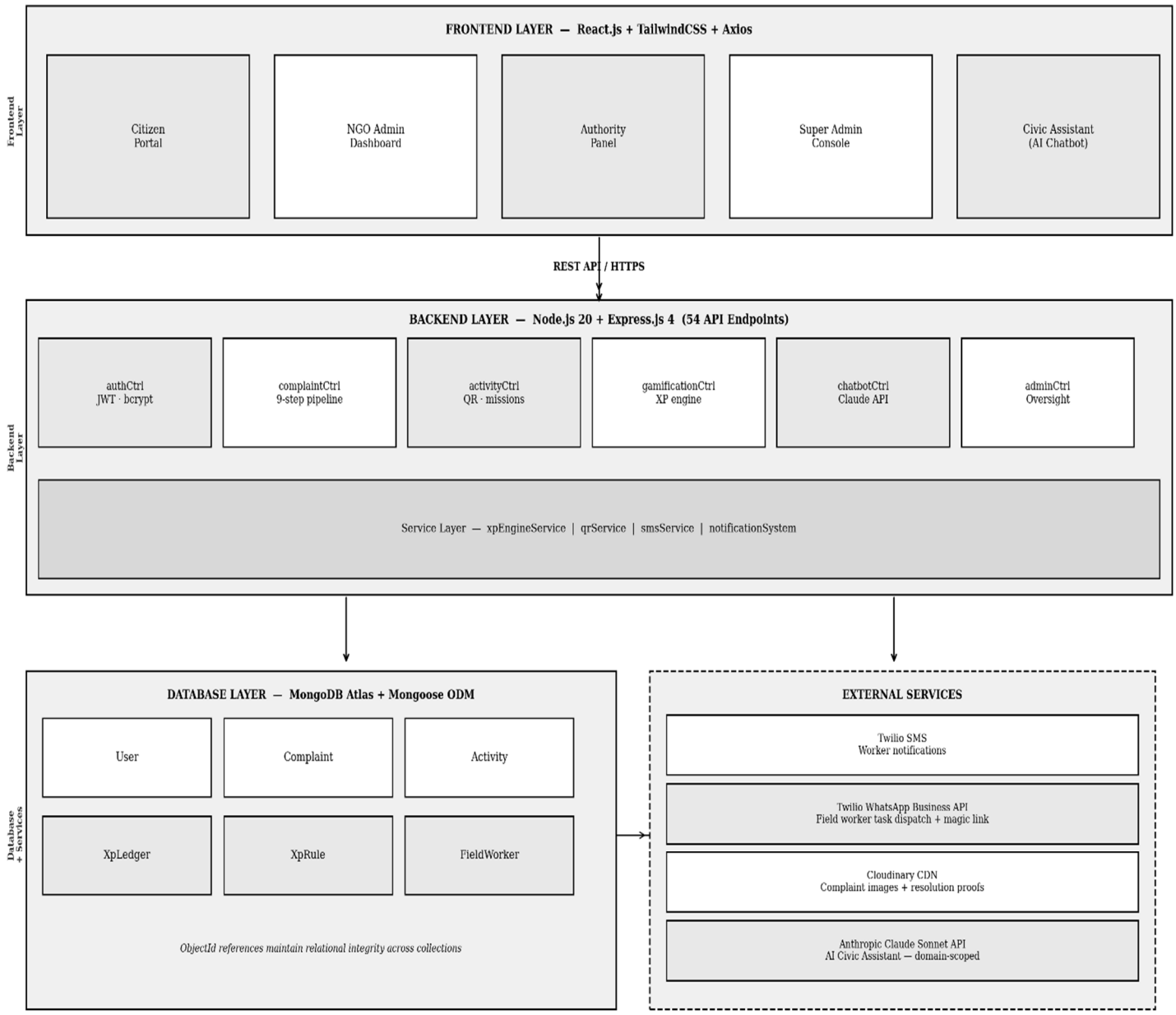


Figure 1. System Architecture

1) Frontend Layer

The frontend is built using React.js with Tailwind CSS for responsive, mobile-friendly styling and Axios for all HTTP communication with the backend. The application implements strict role-based rendering every user type sees only the views, actions, and data relevant to their assigned role.

The five core interfaces are the citizen portal for submitting complaints and joining missions, the NGO administrator dashboard for creating and managing community activities, the local authority panel for complaint assignment and resolution oversight, the super administrator console for platform-wide monitoring and user management, and the AI civic assistant chat interface for natural language queries. The frontend never stores or exposes the AI API key all chatbot requests are routed through a server-side proxy, keeping credentials entirely hidden from the browser.

2) Backend Layer

The backend is built on Node.js 20 and Express.js 4, forming a RESTful API server that exposes 54 endpoints across 9 namespaces. Every client action triggers a backend request that passes through JWT-based authentication middleware before reaching the relevant controller. The system is organized into six controller modules: authCtrl handles login, token issuance, and refresh logic using bcrypt and JWT; complaintCtrl manages the full nine-step complaint lifecycle including geospatial duplicate detection; activityCtrl governs community mission enrollment, QR-based check-in, and GPS attendance verification; gamificationCtrl runs the XP engine with configurable daily limits and cooldown windows to prevent farming; chatbotCtrl proxies all civic assistant requests to the Claude Sonnet API through a secure server-side call; and adminCtrl provides platform-wide oversight, audit access, and user management. Beneath the controllers, a dedicated service layer handles cross-cutting concerns: xpEngineService computes and records XP transactions, qrService generates cryptographic one-time tokens for mission attendance, smsService dispatches Twilio SMS notifications, and notificationSystem manages push and WhatsApp delivery.

3) Database Layer

MongoDB Atlas is used as the primary database, accessed through the Mongoose ODM library which enforces schema validation at the application level. The system is structured around seven collections. User stores profile data, hashed credentials, role assignments, and XP totals. Complaint holds the full complaint record including its nine-step status history, GPS coordinates, assigned field worker, and embedded proof media references. Activity captures mission metadata, attendance logs, QR token state, and participation history. FieldWorker tracks worker assignments, their one-time WhatsApp magic link token, and token expiry state. Notification maintains a delivery history of all SMS, push, and WhatsApp messages sent. XpLedger records every XP transaction as an immutable entry, serving simultaneously as the citizen's reward history and the administrator's audit trail for detecting XP farming. XpRule stores the configurable rule set that governs how many points each action is worth, daily caps, and cooldown periods between eligible actions. All collections maintain referential integrity through ObjectId cross-references rather than data duplication, minimising expensive lookup operations while preserving a clear relational structure.

4) External Integrations

The platform connects to four external services. Twilio SMS handles outbound text notifications to citizens including complaint submission confirmations, status change alerts, and resolution notices. Twilio's WhatsApp Business API is used exclusively for field worker communication: when an authority assigns a complaint, the system dispatches a structured WhatsApp message containing task details and a tokenised proof-upload link, allowing the field worker to photograph and submit resolution evidence without ever needing a platform login. Cloudinary serves as the media storage layer for all uploaded files complaint photographs submitted by citizens and resolution proof images uploaded by field workers are stored and served via Cloudinary's CDN, keeping binary assets entirely outside the application database. The Anthropic Claude Sonnet API powers the civic assistant chatbot through a server-side proxy that ensures the API key is never transmitted to or accessible from the browser at any point.

5) Authentication and Security

When a user logs in successfully, the backend issues two tokens: a short-lived JWT access token valid for 15 minutes and a long-lived refresh token valid for 30 days. Both tokens are stored exclusively as HTTP-only cookies, making them inaccessible to JavaScript and protecting against cross-site scripting attacks. When the access token expires, the frontend silently requests a new one using the refresh token, maintaining a seamless session without requiring the user to log in again. All passwords are stored using bcrypt with a work factor of 12, making brute-force attacks computationally expensive. Field workers follow a completely passwordless flow they never register or log in to the platform. Instead, when a task is assigned, the system generates a one-time secure token embedded in a WhatsApp link. That link expires after 24 hours and is permanently invalidated after its first use, preventing replay or sharing of the upload URL.

B. System Flow

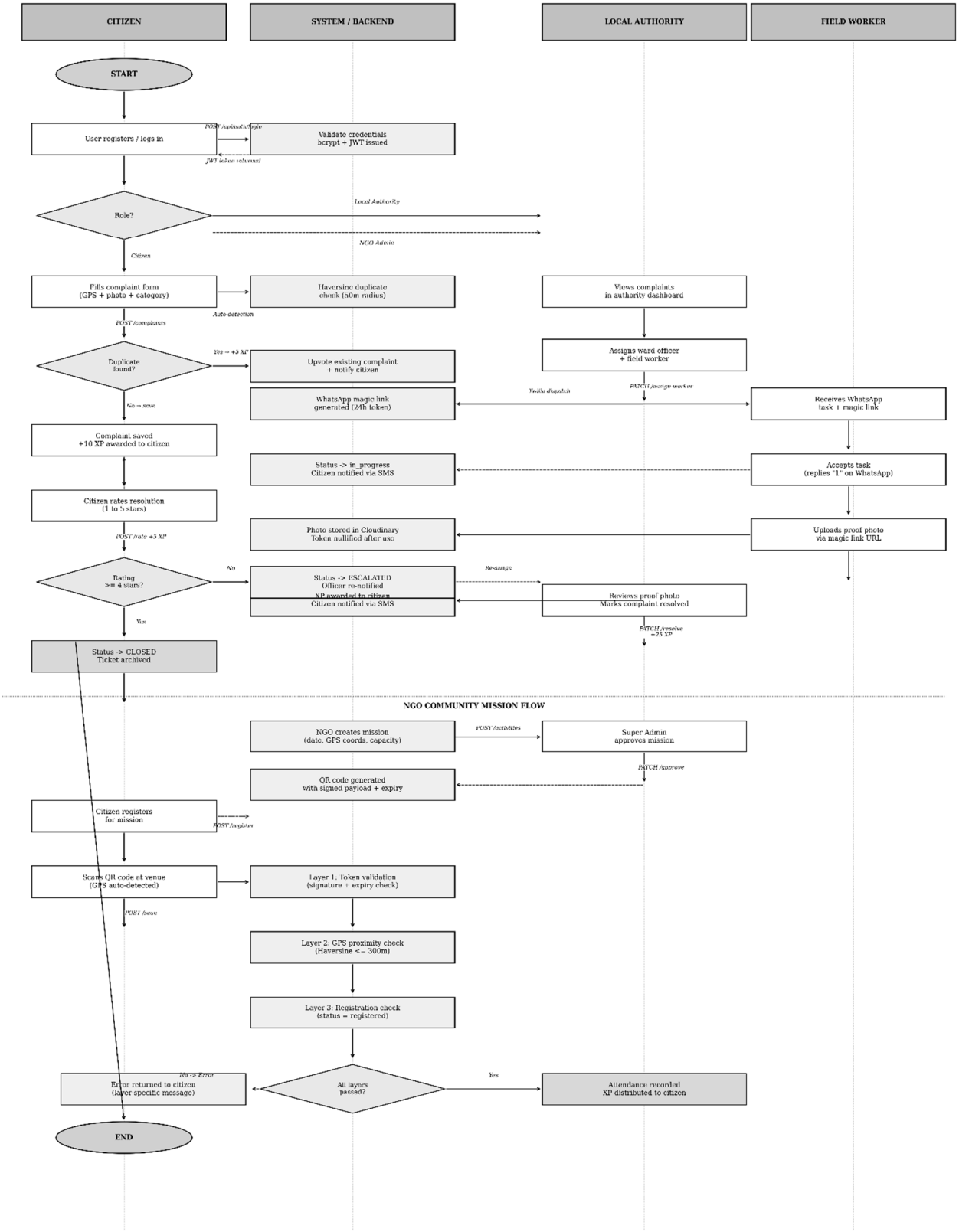


Figure 2. System Flow

Table II — Technology Stack

Layer	Technology	Purpose
Frontend	React.js, Tailwind CSS, Axios	SPA, responsive UI, API communication
Backend	Node.js 20, Express.js 4	RESTful API server, middleware, route guards
Database	MongoDB Atlas, Mongoose ODM	Document storage, aggregation, geospatial queries
Authentication	JWT, bcrypt 12	Dual-token refresh, password hashing
Messaging	Twilio SMS & WhatsApp Business API	Worker task dispatch, citizen notifications
Media Storage	Cloudinary CDN	Complaint images, resolution proofs, activity banners
QR Generation	qrcode (npm), Node.js crypto	Signed tokens with GPS metadata and expiry
AI Chatbot	Google Gemini API	Civic assistant schemes, rights, platform help

IV. IMPLEMENTATION

A. Complaint Lifecycle Pipeline

The complaint module implements a nine-step pipeline tracking a citizen grievance from initial submission through resolution, citizen rating, and automated closure or escalation. Each state transition is recorded in an immutable timeline array embedded in the Complaint document, providing a complete audit trail.

Step 1: citizen files complaint with GPS coordinates and image (+10 XP).
Step 2: system runs Haversine duplicate detection within 50 metres (+5 XP for verified duplicates).
Step 3: local authority assigns ward officer.
Step 4: local authority assigns field worker via WhatsApp magic link.
Step 5: field worker accepts via WhatsApp reply, status moves to in-progress.
Step 6: field worker uploads photographic proof via tokenized URL.
Step 7: local authority marks complaint resolved (+25 XP).
Step 8: citizen submits star rating (+5 XP).
Step 9: system closes if rating is 4 stars or above, or escalates if below.

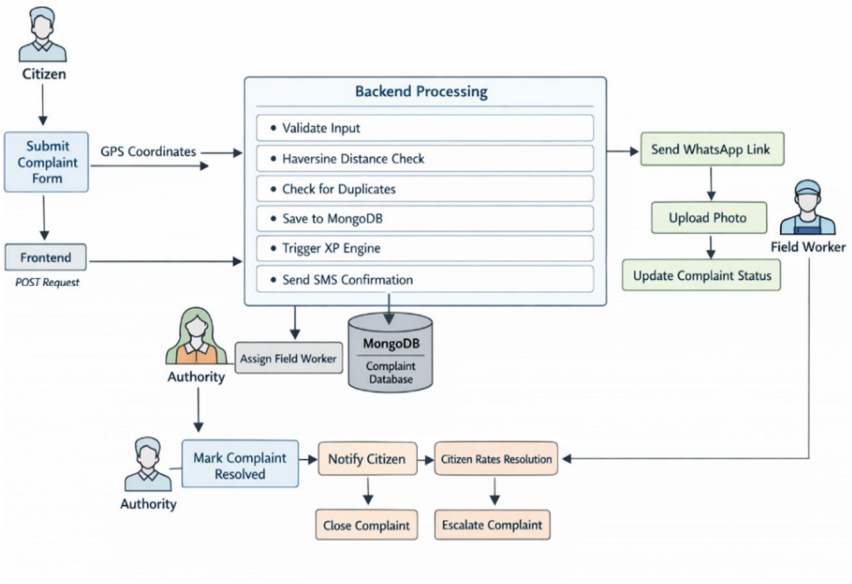


Figure 3. complaint flow diagram

B. Activity Participation Pipeline

The activity module implements a nine-step pipeline tracking a citizen's participation in civic activities from discovery and enrollment through verification, reward allocation, and completion. Each interaction is recorded in an immutable participation log embedded within the Activity document, ensuring transparency, auditability, and fair reward distribution.

Step 1: citizen discovers available activities based on location and preferences.
Step 2: citizen enrolls/registers for an activity (+5 XP).
Step 3: citizen performs check-in at activity location via QR code or GPS validation.
Step 4: system verifies participation using timestamp and proximity checks.
Step 5: citizen submits proof of participation (image/video upload).
Step 6: local authority/organizer reviews and verifies submitted proof.
Step 7: system awards XP based on activity type and impact (+20–50 XP).
Step 8: citizen's leaderboard position and badges are updated (+5 XP bonus).
Step 9: citizen provides feedback/rating and system marks participation complete.

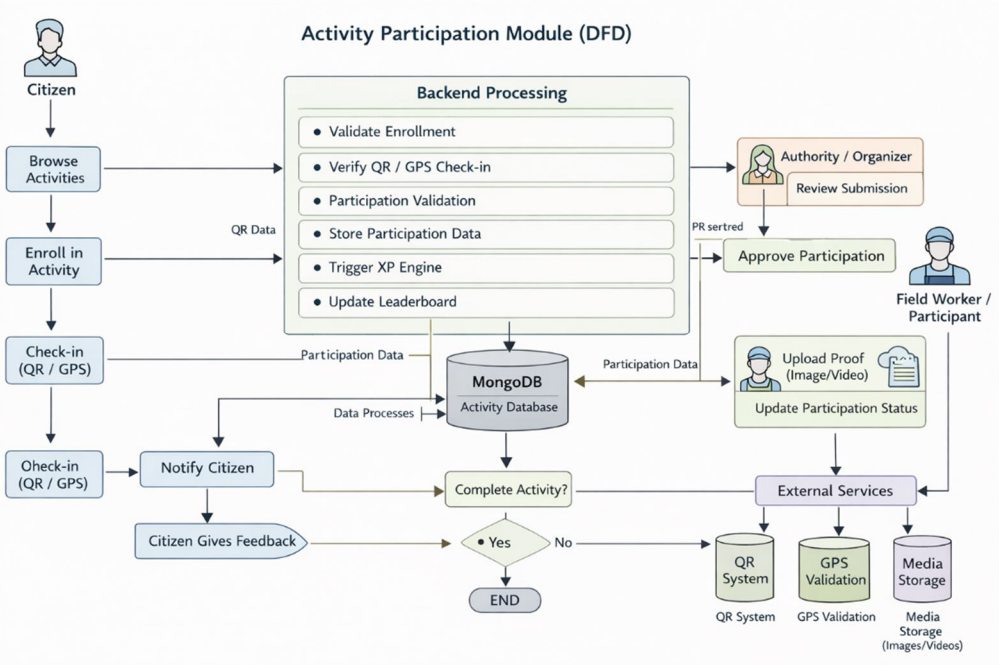


Figure 4. Activity Participation diagram

C. Three-Layer QR Attendance Gate

The QR attendance gate operates sequentially across three layers. Layer 1 performs cryptographic token validation the QR payload signature and expiry are checked server-side, and expired or tampered tokens are rejected. Layer 2 performs GPS proximity validation using the Haversine formula, rejecting scans beyond the configurable radius (default 300 metres) with a distance diagnostic. Layer 3 validates that the scanning user holds a prior attendance record with registered status. All three layers must pass for attendance to be recorded. Each layer returns a distinct error code, enabling administrators to diagnose specific failure patterns.

D. XP Gamification Engine

The XP engine is implemented as a service layer backed by a MongoDB XpRule collection, separating reward logic from controller code and enabling runtime reconfiguration without redeployment. An in-process rule cache minimizes database reads on high-frequency award calls, invalidated on every admin rule modification. Per-action cooldown enforcement queries the XpLedger for the user's most recent award of the same action type. The maxPer Day control counts same-action awards within the preceding 24-hour window. A first-complaint bonus of +30 XP is awarded once per account to incentivize onboarding.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

E. AI Civic Assistant

The civic assistant routes citizen messages through a dedicated Express.js endpoint. The server injects a domain-scoping system prompt before forwarding the conversation to the Claude Sonnet API. This server-side proxying ensures the API key is never exposed to the frontend client. The system prompt restricts responses to three knowledge domains: government schemes (central and state welfare programs), platform help (complaint filing, mission registration, XP earning), and civic rights (RTI, consumer rights, local services). Conversation history is maintained per session in React component state and forwarded with each API call to enable contextual follow-up queries. Suggested question chips are displayed on session initiation to reduce onboarding friction.

F. Field Worker Magic Link

Upon field worker assignment, the system generates a cryptographically random token stored on the Complaint document with a 24-hour expiry timestamp. This token is embedded in a URL and dispatched via WhatsApp through Twilio. The worker accesses the upload interface without platform credentials. The token is validated for both value and expiry before accepting any proof upload, and is nullified immediately after first use to prevent replay attacks.

Table III — XP Action Table with Default Values and Anti-Abuse Controls

XP Action	Trigger	Default XP	Anti-Abuse Control
file_complaint	Citizen submits new complaint	+10	First complaint gives +30 one-time bonus
verify_duplicate	Complaint identified as duplicate	+5	Once per existing complaint per user
complaint_resolved	Complaint marked resolved by authority	+25	Once per complaint lifecycle
rate_feedback	Citizen submits star rating	+5	Once per complaint, requires resolved status
attend_ngo_activity	QR + GPS verified at mission venue	Variable	Once per activity
ngo_create_mission	NGO mission approved by super admin	+50	Once per approved activity
redeem_reward	Citizen redeems XP for reward	−cost	Balance check enforced before deduction

V. RESULT AND DISCUSSION

A. Functional Testing

Functional validation was conducted using 61 structured test cases distributed across all major platform modules. Each test case was designed to verify a specific behaviour under both normal operating conditions and deliberate edge-case inputs, including boundary violations, expired tokens, out-of-range coordinates, and simulated abuse patterns.

The complaint pipeline was exercised across all nine lifecycle steps, from initial submission through field worker proof upload to citizen-rated resolution. Every transition behaved as specified, with status updates, XP awards, and notifications triggering at the correct stages. Geospatial duplicate detection correctly identified and flagged complaints filed within the 50-metre proximity threshold in every applicable test case, with no false positives or missed duplicates observed in the tested dataset of 50 complaints.

The field worker magic link mechanism was verified against two specific rejection conditions: uploads attempted after the 24-hour expiry window were rejected in all tested scenarios, and the link was correctly invalidated after its first use in every case, confirming that the one-time token lifecycle functions as designed.

The three-layer QR attendance gate was tested across all permutations of failure invalid or tampered QR tokens, GPS coordinates placed outside the 50-metre activity radius, and users who had not completed prior registration. All three rejection conditions were enforced independently and in combination without allowing partial access.

The XP gamification engine was stress-tested against simulated abuse scenarios including rapid repeat submissions, multiple ratings within a cooldown window, and attempts to trigger XP awards through replayed requests. The engine correctly blocked over-award in every simulated case, with the XpLedger recording accurate transaction histories throughout.

The civic assistant was evaluated across 61 query inputs spanning six categories. It returned accurate, in-domain responses for queries on central government schemes (15/15), state schemes (9/10), platform usage (12/12), and civic rights including RTI and RERA provisions (8/8). Local area information queries achieved an 83% accuracy rate (5/6), with one response returning a partially correct answer due to limited geographic specificity in the training context. All 10 off-topic queries were correctly redirected back to civic matters, confirming that domain-restriction prompting functioned reliably with a 100% redirect rate on out-of-scope inputs.

Table IV — Civic Assistant Response Quality Evaluation (n = 61)

Query Category	Queries Tested	Accurate / In-Domain	Redirected
Central Govt. Schemes	15	15 / 15 (100%)	—
State Schemes	10	9 / 10 (90%)	—
Platform Help	12	12 / 12 (100%)	—
Civic Rights (RTI, RERA)	8	8 / 8 (100%)	—
Local Area Information	6	5 / 6 (83%)	—
Off-Topic Queries	10	—	10 / 10 (100%)

B. Performance Results

Response time measurements were recorded for all core API operations and external service calls under controlled test conditions. All internally processed operations completed well within the 200 ms threshold considered acceptable for a real-time civic application. Standard data fetch operations returned in under 120 ms, while write operations which include an atomic XP ledger entry alongside the primary document update completed in under 200 ms. JWT silent token refresh, which operates without a database lookup, was consistently measured below 80 ms, ensuring that token expiry is imperceptible to the end user.

The QR scan-to-attendance pipeline, which executes all three verification layers cryptographic token validation, GPS proximity check, and registration confirmation in sequence, completed in under 350 ms. This is particularly notable given that the three-gate check involves multiple database reads and a geospatial calculation; the result confirms that the attendance pipeline does not introduce perceptible lag at the point of physical check-in. Operations involving external third-party APIs introduced latency outside the system's direct control. Twilio SMS dispatch ranged from 1.2 to 3.5 seconds depending on carrier routing, and civic assistant first-token response via the Google Gemini API ranged from 0.8 to 1.5 seconds with streaming disabled. Both delays are contextually appropriate SMS delivery is a background notification and not user-blocking, while a sub-1.5 second initial response for a conversational AI query is within acceptable user experience tolerance. Haversine-based duplicate detection across a filtered set of 50 complaints resolved in under 90 ms, confirming that the geospatial computation does not create a bottleneck at submission time.

Operation	Observed Time	Notes
API read response	< 120 ms	Standard data fetch
API write response	< 200 ms	Includes XP ledger write
QR scan to attendance record	< 350 ms	All three gate layers included
JWT silent refresh	< 80 ms	No database lookup needed
Twilio SMS dispatch	1.2 – 3.5 s	Depends on carrier routing
Civic assistant first response	0.8 – 1.5 s	Google Gemini, streaming off
Duplicate detection (50 complaints)	< 90 ms	Haversine on filtered set

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

C. Security Evaluation

The system was evaluated against the OWASP Top 10. NoSQL injection is prevented through Mongoose ODM type-cast schemas and express-validator input sanitization. Cross-site scripting is mitigated by storing tokens in HttpOnly cookies. Cross-site request forgery is blocked through Same Site strict cookie settings. Role-based access control is enforced on every protected route through middleware. The civic assistant API key is stored exclusively in server-side environment variables and is never transmitted to the browser. Field worker magic links are single-use and expire after 24 hours.

D. Discussion

The results confirm that the platform successfully addresses the three civic gaps identified in the introduction. The complaint pipeline creates a fully accountable workflow where every step is tracked and every actor has a defined responsibility. The QR attendance gate solves the proxy attendance problem that existing systems leave open. The gamification engine motivates repeat engagement without being exploitable. The civic assistant delivers accurate conversational answers to scheme-related queries without requiring citizens to navigate complex government portals. One area for future improvement is the civic assistant's performance on very recent or highly localised information, which can be resolved by adding a Retrieval-Augmented Generation layer connecting the chatbot to live government documents.

VI. CONCLUSIONS

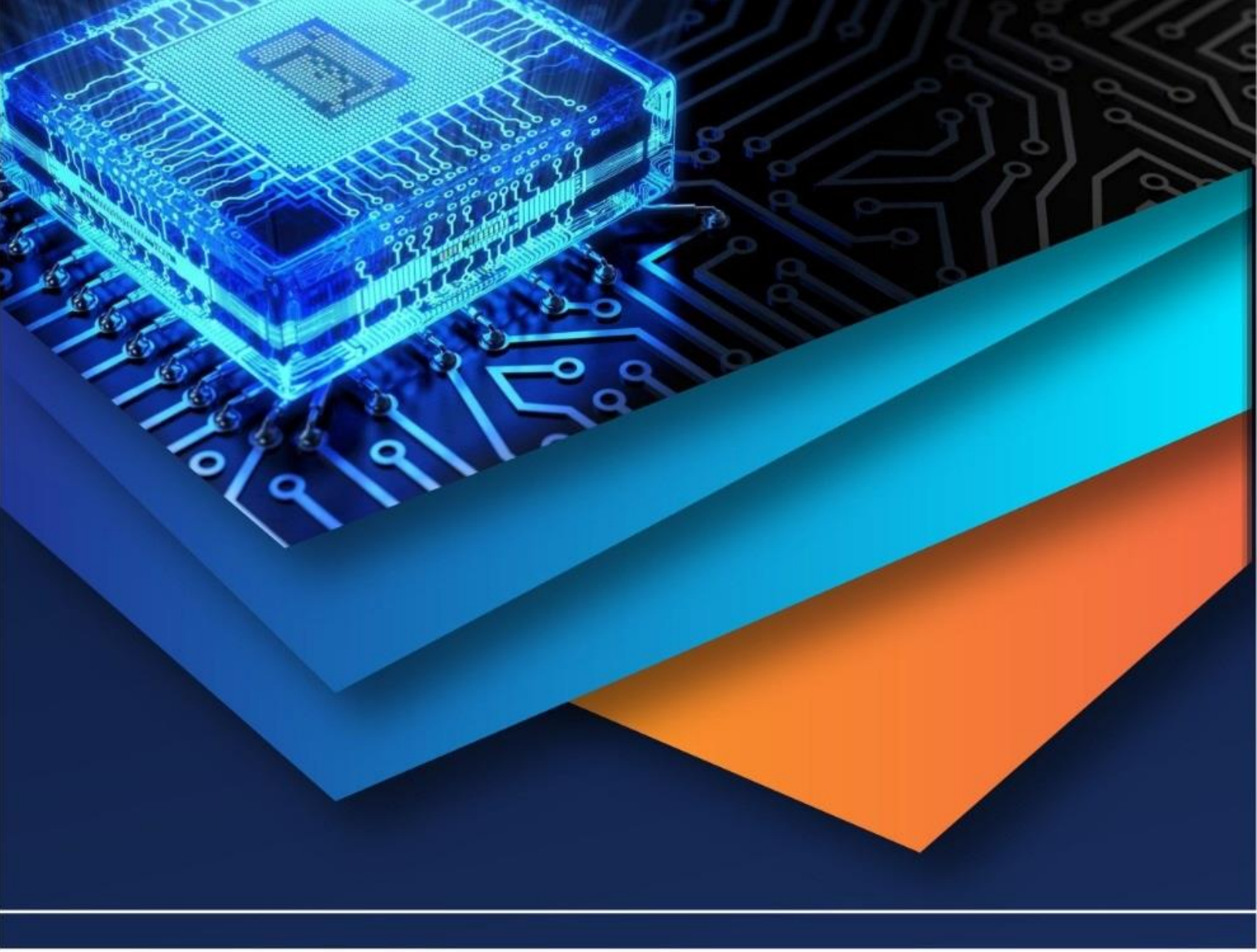
This paper has presented a gamified civic engagement platform that addresses six critical failures in urban civic systems: complaint opacity, field worker accountability gaps, duplicate complaint proliferation, unverified community mission participation, absence of civic participation incentives, and government scheme inaccessibility. Through a structured nine-step complaint pipeline, a three-layer QR-GPS-registration attendance gate, a configurable database-driven XP engine, and an AI-powered civic assistant, the platform demonstrates that thoughtful application of automation, gamification, multi-channel communication, and large language models can meaningfully improve civic process quality and citizen engagement in resource-constrained urban environments.

The platform's role-aware architecture spanning five actor types with dedicated workflows models the real organizational complexity of municipal service delivery. The XpLedger's dual function as citizen reward history and administrative audit trail reflects the platform's commitment to transparency. The civic assistant's domain-scoped LLM integration demonstrates that AI capabilities can be practically applied to civic information delivery using commercially available foundation models with appropriate behavioural constraints, without requiring custom model training.

REFERENCES

[1] NITI Aayog, "Government Scheme Awareness and Uptake in Tier-2 Cities," Government of India, New Delhi, 2021.
[2] N. Shinghal, "WhatsApp as a Channel for Government Service Delivery," Indian Journal of Public Administration, vol. 66, no. 1, pp. 45–62, 2020.
[3] A. Kumar and R. Singh, "Effectiveness of e-Governance Complaint Portals in Indian Municipalities," Journal of Urban Governance and Technology, vol. 5, no. 2, pp. 88–104, 2022.
[4] L. Zhao, Y. Wang, and H. Chen, "Gamification in Smart City Public Participation Platforms: A Behavioral Study," Smart Cities and Society Journal, vol. 9, no. 1, pp. 211–228, 2023.
[5] P. Gupta, S. Mishra, and A. Tiwari, "QR Code Based Volunteer Attendance Verification for NGO Events," International Journal of Engineering and Applied Sciences, vol. 10, no. 3, pp. 145–159, 2022.
[6] S. Rao and D. Mehta, "WhatsApp as a Municipal Field Operations Channel: Accountability and Limitations," Indian Journal of Public Administration, vol. 69, no. 2, pp. 310–325, 2023.
[7] R. Verma, K. Pandey, and M. Jain, "Bridging the Government Scheme Awareness Gap Through Conversational Interfaces," Journal of e-Governance, vol. 47, no. 1, pp. 55–72, 2024.
[8] R. Patel and S. Joshi, "Domain-Scoped Large Language Models for Citizen-Facing Government Services," International Journal of AI and Public Policy, vol. 3, no. 2, pp. 78–95, 2024.
[9] A. Sharma, V. Kulkarni, and N. Desai, "Role-Based Access Control in Multi-Stakeholder Civic Technology Platforms," Journal of Information Security and Applications, vol. 82, pp. 103–118, 2025.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)