IJARCCE

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154230

BUILDING MANAGEMENT SYSTEM

Vatsal Jain[1], Darshan Kandalgaonkar[2], Gautam Enjam[3], Deepesh Ahire[4], Rakesh Suryawanshi[5]

Dept of Computer Engineering, A.C. Patil College of Engineering, Navi Mumbai, India[1]
Dept of Computer Engineering, A.C. Patil College of Engineering, Navi Mumbai, India[2]
Dept of Computer Engineering, A.C. Patil College of Engineering, Navi Mumbai, India[3]
Dept of Computer Engineering, A.C. Patil College of Engineering, Navi Mumbai, India[4]
Dept of Computer Engineering, A.C. Patil College of Engineering, Navi Mumbai, India[5]

Abstract: Managing residential buildings usually relies on standard First-In-First-Out (FIFO) ticketing. This rigid approach often causes problems because minor tasks can easily delay urgent emergencies. This paper presents a Smart Building Management System that solves this scheduling flaw by combining a dynamic triage algorithm with a resident-focused micro-economy. Instead of treating all complaints equally, the system applies an exponential aging formula to tickets, strictly capped at a maximum score of 100. The queueing engine constantly adjusts these priorities by scanning text descriptions for danger keywords, tracking SLA deadlines, and boosting scores when multiple neighbors report the exact same problem. Concurrently, a point-based gamified economy runs in the background. Automated tasks track maintenance payments and adjust resident trust scores based on their financial reliability. Our testing shows that linking contextual math to a gamified ledger successfully creates a self-correcting environment. Emergencies receive immediate action, routine tasks avoid starvation, and residents are financially incentivized to maintain good standing.

Keywords: Algorithmic Starvation, Building Management Systems, Context-Aware Triage, Gamification, Micro-Economy, Queuing Theory

I. INTRODUCTION

Urban residential complexes have grown significantly in recent years, generating a substantial amount of interest among developers in modernizing facility management platforms. Most existing software solutions rely heavily on traditional First-In-First-Out (FIFO) queuing models for handling resident complaints and maintenance requests. This is primarily because such linear systems are relatively simple to implement and work reasonably well for small-scale operations. However, as buildings scale and the volume of requests increases, treating all incoming tasks equally creates a significant administrative bottleneck. Routine tasks (e.g., cosmetic repairs, minor plumbing leaks, or general inquiries) often flood the system, inadvertently pushing highly critical emergencies to the back of the queue. This phenomenon, known computationally as task starvation, makes static scheduling a potentially dangerous approach in high-density living environments.

Furthermore, there has traditionally been a disconnect between maintenance scheduling and the financial behavior of the residents who request these services. It should be noted, however, that while digital billing ledgers exist, administrators still frequently rely on manual cross-referencing to track timely maintenance payments and apply late penalties. As long as property managers must manually bridge the gap between service requests and financial compliance, the overall efficiency of the residential ecosystem remains severely limited. Beyond doubt, contemporary management architectures should equip administrators with tools that go beyond passive digital logbooks; they should actively facilitate self-regulating environments where residents are intrinsically incentivized to maintain good standing.

To address these ongoing challenges, this paper proposes a modernized Smart Building Management System architecture. We introduce a dynamic triage algorithm that abandons the rigid limitations of the static FIFO approach. Instead, it continuously recalculates the actual urgency of a task through a combination of lexical danger keyword detection, strict Service Level Agreement (SLA) deadlines, and crowd-sourced community feedback. Parallel to this scheduling logic, the architecture incorporates a decentralized, gamified micro-economy. By utilizing automated scheduled routines, the system seamlessly maps resident payment behaviors to a decaying trust score, dynamically enforcing financial compliance without the need for manual intervention.

II. LITERATURE SURVEY

Recently, researchers have focused heavily on how Internet of Things (IoT) tools can modernize everyday urban infrastructure. This is largely because smart building platforms now generate massive amounts of daily data—ranging from routine maintenance logs to urgent security alerts [1]. In early iterations of these systems, developers typically relied on greedy heuristics or simple First-In-First-Out (FIFO) models to handle incoming tasks. While these linear methods are easy to program and work fine when traffic is low, they struggle significantly during peak hours. In fact, studies show that when standard linear queues become overloaded, the system frequently misses deadlines for highly critical tasks, which can ultimately compromise overall building safety [2].

To get around these static limitations, a number of proposed solutions lean into dynamic priority scheduling. In these setups, a task's priority isn't permanently locked in when it is created; instead, it shifts based on changing factors like execution deadlines or sudden drops in energy [3]. For example, dynamic schedulers often use heap sorting to stretch the system's capacity so that high-value tasks are not dropped [2]. However, applying these models directly to human-centric property management reveals a notable gap. Current schedulers almost exclusively look at hard numerical metrics (e.g., battery levels or network latency). They rarely consider the actual textual context of what a resident is complaining about, nor do they factor in how many people in the community are simultaneously affected by the exact same issue.

Moving beyond just computational scheduling, getting humans to actually comply with building rules—especially regarding financial administration and paying maintenance bills on time—presents a completely different hurdle. To tackle this sort of behavioral compliance, many developers have turned to gamification, which essentially involves using game-design elements in non-game environments [4]. By weaving in mechanics like leaderboards, loyalty tiers, and automated point systems, researchers have shown that gamification can significantly boost user motivation in both educational settings and corporate offices [5].

It should be noted, however, that while dynamic scheduling and gamified economies both work well on their own, we rarely see them combined into a single, cohesive architecture for residential buildings. Most platforms currently on the market either focus strictly on task management (ignoring behavioral compliance) or they act as basic billing portals without any intelligent task sorting. Therefore, it seems clear that modern facilities need a hybrid approach. They require a system that not only triages tasks dynamically based on linguistic context and community feedback but also manages resident financial behavior through a decentralized, gamified ledger.

III. PROBLEM STATEMENT

Current complaint management systems exhibit the following limitations:

- Equal treatment of critical and non-critical complaints
- Starvation of low-priority tasks
- Lack of contextual understanding of complaint descriptions
- Absence of deadline-based escalation

IV. PROPOSED SYSTEM

To successfully bridge the gap between static queueing and behavioral compliance, the proposed architecture relies on two distinct but interconnected computational engines: a context-aware triage calculator and a decentralized penalty ledger. By processing these engines asynchronously, the system achieves a self-regulating ecosystem without overloading the primary server thread.

A. Context-Aware Triage Algorithm

Previous scheduling approaches in facility management often attempt to prioritize tasks by blindly multiplying weight factors together. While functional in theory, this linear approach frequently results in unbounded priority inflation, where older, minor complaints eventually mathematically outrank immediate life-safety emergencies. To solve this, the proposed algorithm calculates a modified base score using contextual multipliers and then adds an exponential aging curve. Most importantly, the final priority score is mathematically capped at a strict ceiling of 100. This ceiling guarantees that an extreme anomaly—such as a fire or critical electrical hazard—immediately halts the queue and cannot be mathematically surpassed by an aging cosmetic repair. The priority equation is defined as:

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154230

$$P_{final} = min\left((B_s \times M_{lexical} \times M_{crowd} \times M_{SLA}) + (\alpha \times T^{\beta}), 100\right)$$

In this formula, B_s represents the base severity assigned to the task category. $M_{\{lexical\}}, M_{\{crowd\}}$, and M_{SLA} act as dynamic multipliers that adjust the baseline depending on dangerous keyword detection, the volume of community impact, and standard deadline breaches. Finally, T represents the total hours elapsed, combined with α and β variables to plot a non-linear exponential aging curve that actively prevents task starvation.

B. Gamified Economy & Automated Ledgers

Parallel to the triage engine, the system continuously enforces financial and behavioral compliance through a background ledger. Instead of relying on manual administrative oversight, the architecture utilizes automated cron jobs. If a resident fails to pay their monthly maintenance bill, or if their system activity is flagged as malicious spam by an administrator, the backend mathematically deducts points from their loyalty balance. Conversely, on-time payments yield positive point generation, creating a financial micro-economy that intrinsically motivates residents to maintain a positive algorithmic trust score.

C. Integrated Execution Workflow

To ensure real-time responsiveness, the dual engines do not wait for human interaction. The following pseudocode outlines the automated 60-second background sweep that powers both the triage updates and the gamified ledger:
While the server remains active, the system performs a background sweep at regular intervals of 60 seconds to continuously monitor and update complaint priorities and user records. During each cycle, every complaint is evaluated by calculating the elapsed time since its submission. The system then adjusts its priority using multiple factors, including detection of critical or danger-related keywords and the number of residents affected by the issue.
An exponential aging mechanism is applied to ensure that older complaints gradually gain higher priority over time. If a complaint's final priority exceeds a predefined threshold, it is immediately flagged as critical, and an emergency alert is sent to the administrator through real-time communication channels.
Simultaneously, the system reviews all resident accounts to identify overdue payments or misuse such as spam complaints. Appropriate penalties are applied by deducting loyalty points from the respective users.
Finally, all complaints are reorganized in descending order of their computed priority, ensuring that the most urgent issues are addressed first. This continuous process maintains an adaptive and responsive complaint management system.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154230

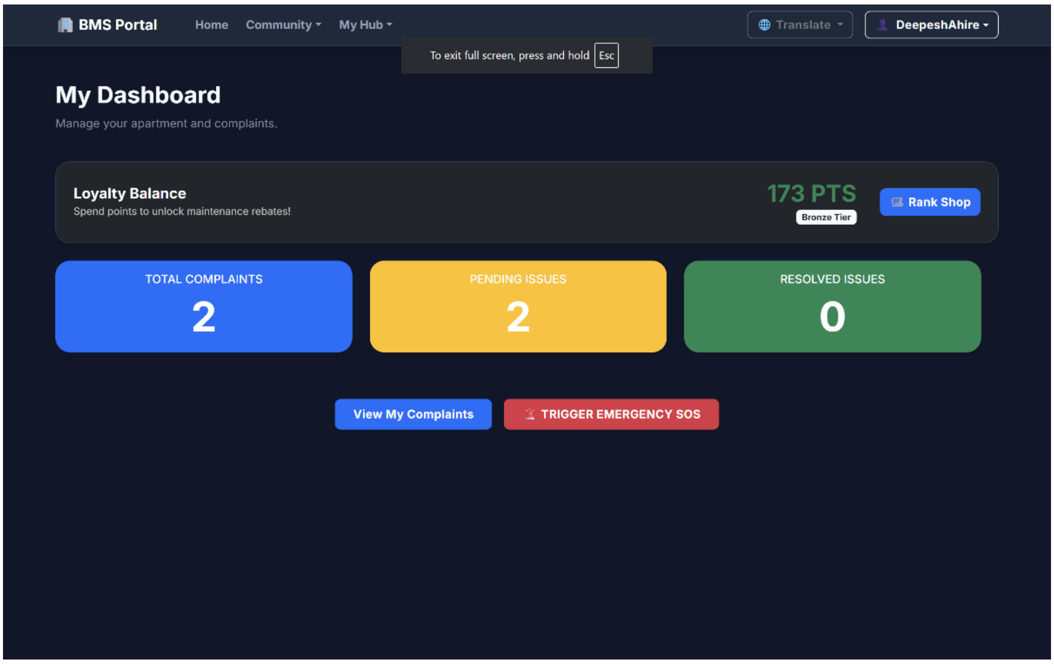

Fig. 1 Dashboard overview displaying complaints, pending issues, and loyalty points.

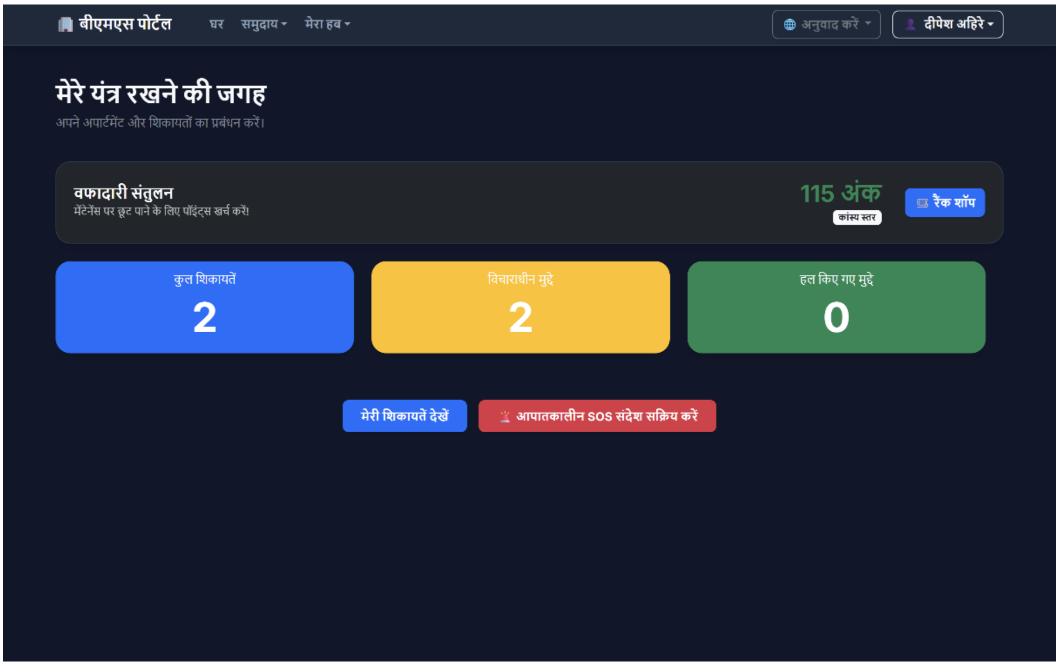

Fig. 2 Multi-language support interface for broader accessibility.

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154230

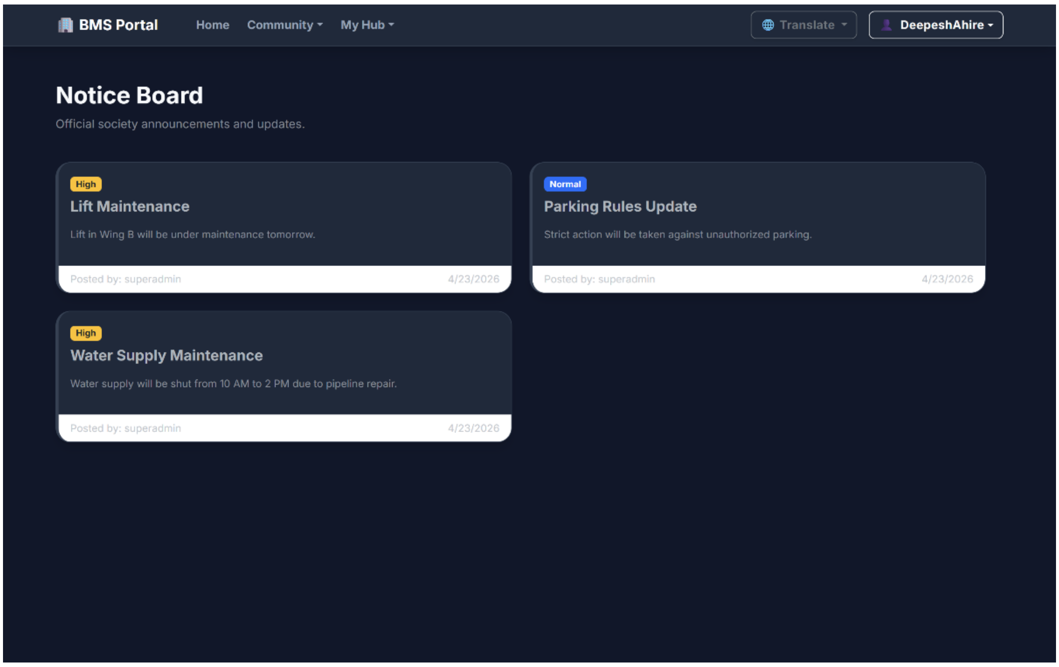

Fig. 3 Notice board showing society announcements and maintenance updates.

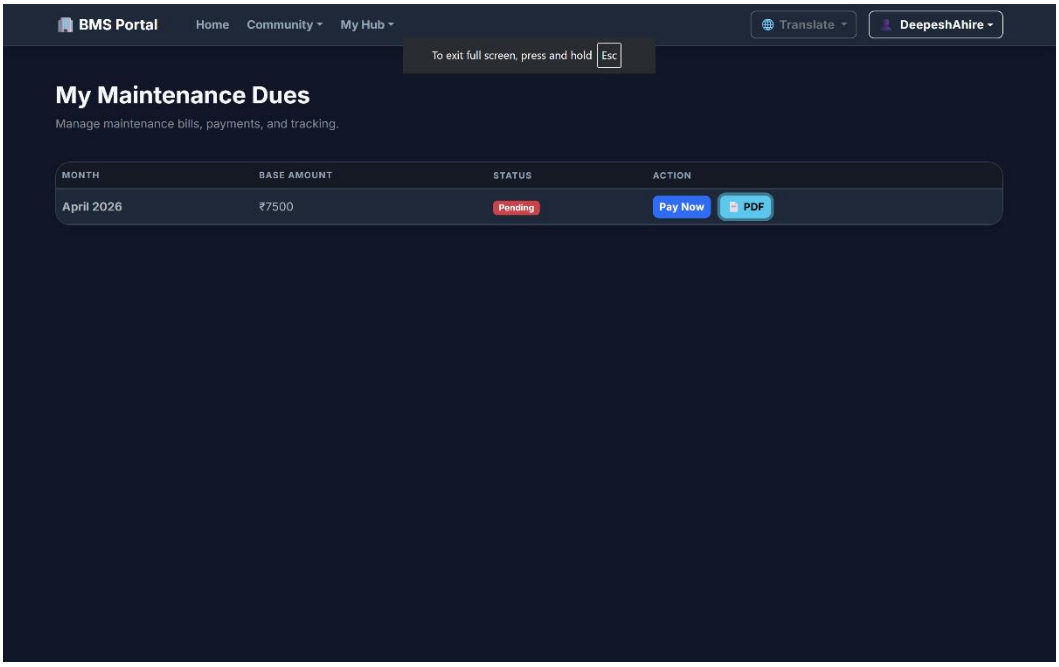

Fig. 4 Maintenance dues interface displaying monthly charges, payment status, and options for online payment and receipt download.

IJARCCE
ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026
DOI: 10.17148/IJARCCE.2026.154230

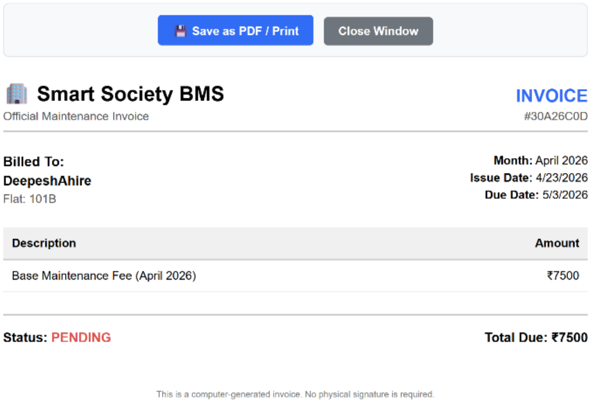

Fig. 5 System-generated maintenance invoice for resident billing.

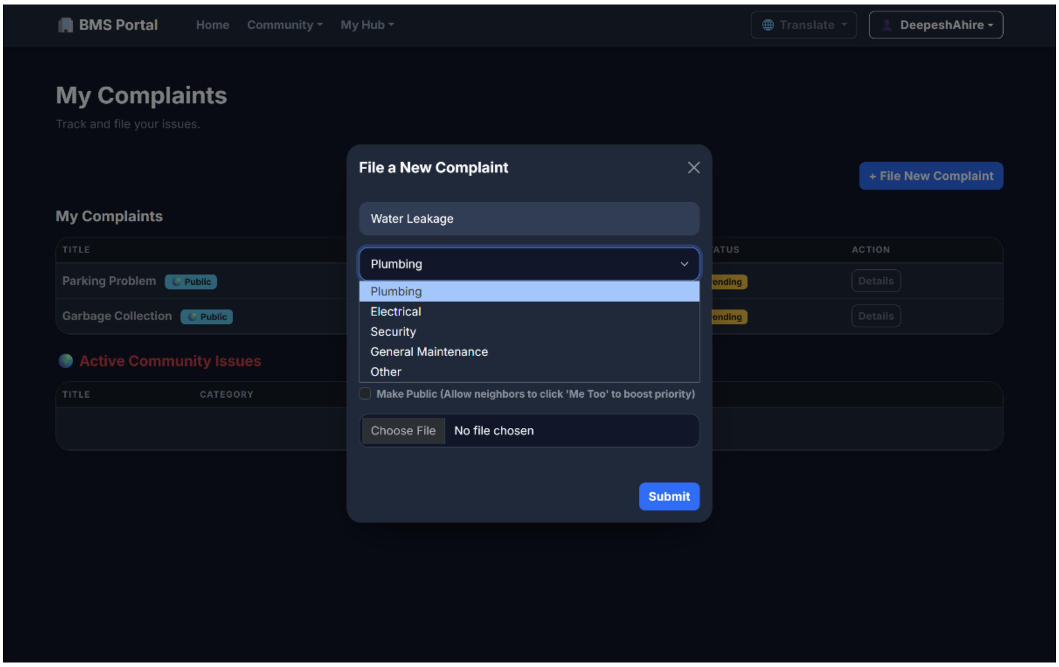

Fig. 6 Form for submitting a new complaint with category and description.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026
DOI: 10.17148/IJARCCE.2026.154230

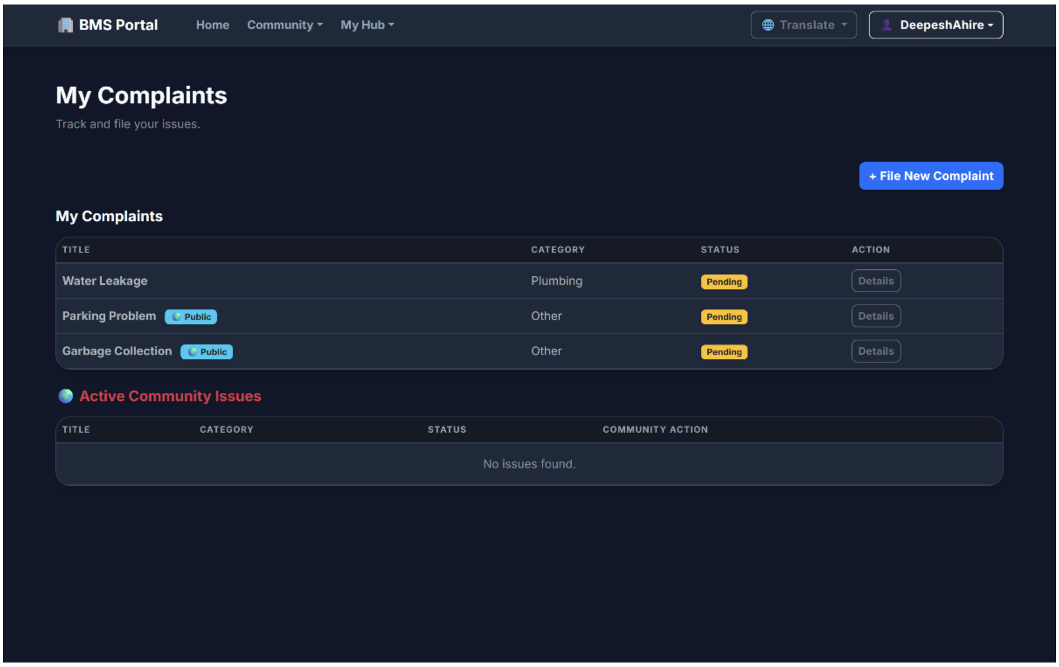

Fig. 7 Detailed complaint view with status tracking and updates.

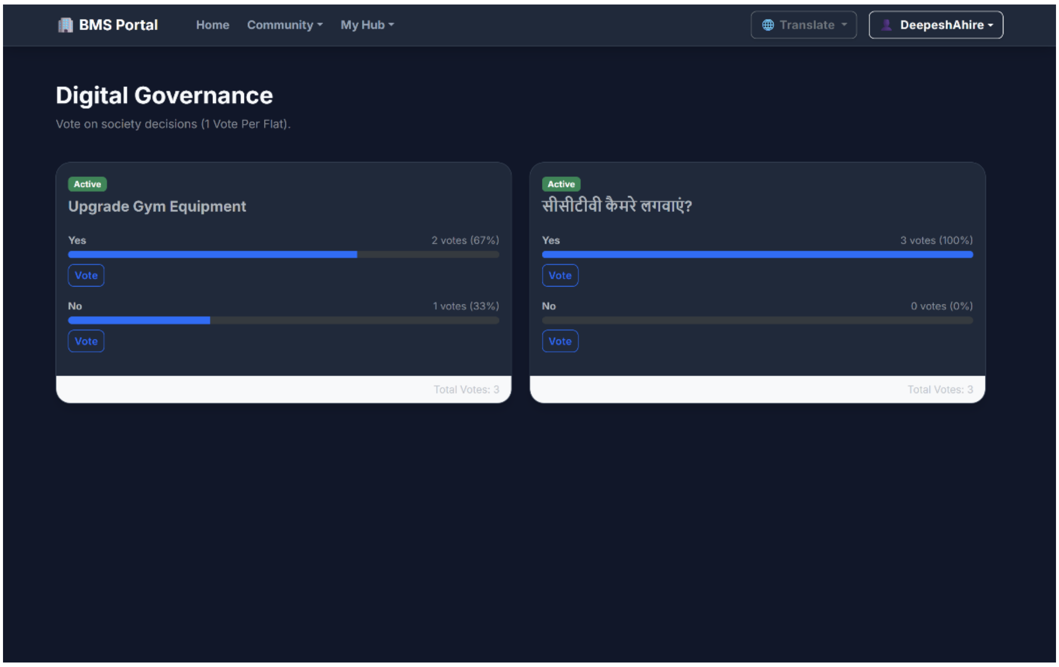

Fig. 8 Digital governance module for voting on community decisions.

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁂ Peer-reviewed & Refereed journal ⁂ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154230

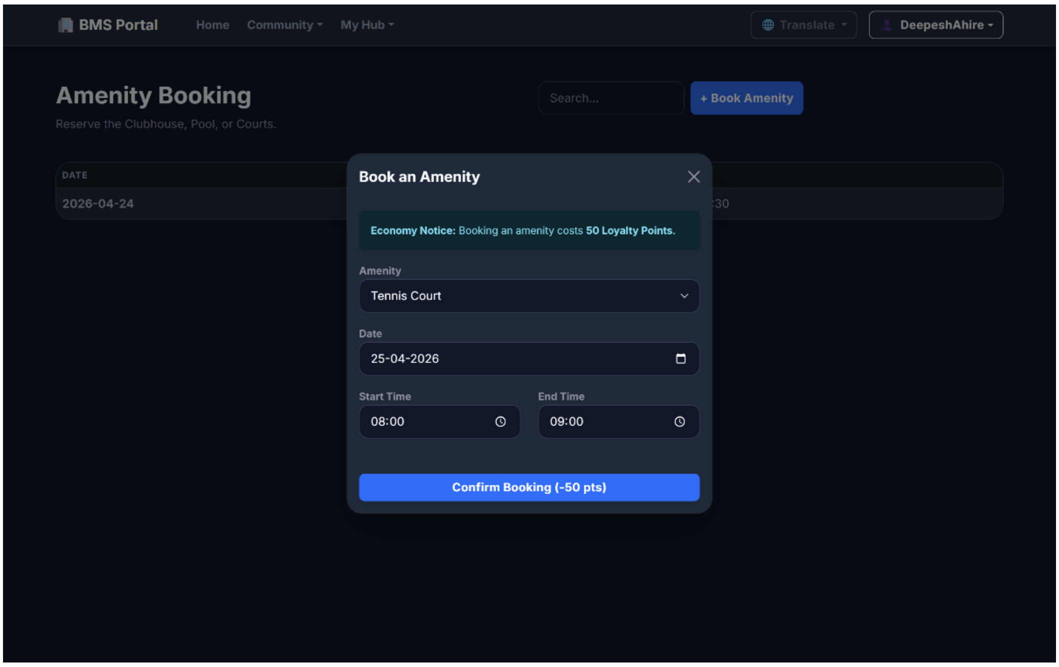

Fig. 9 Amenity booking interface for reserving facilities with date and time selection.

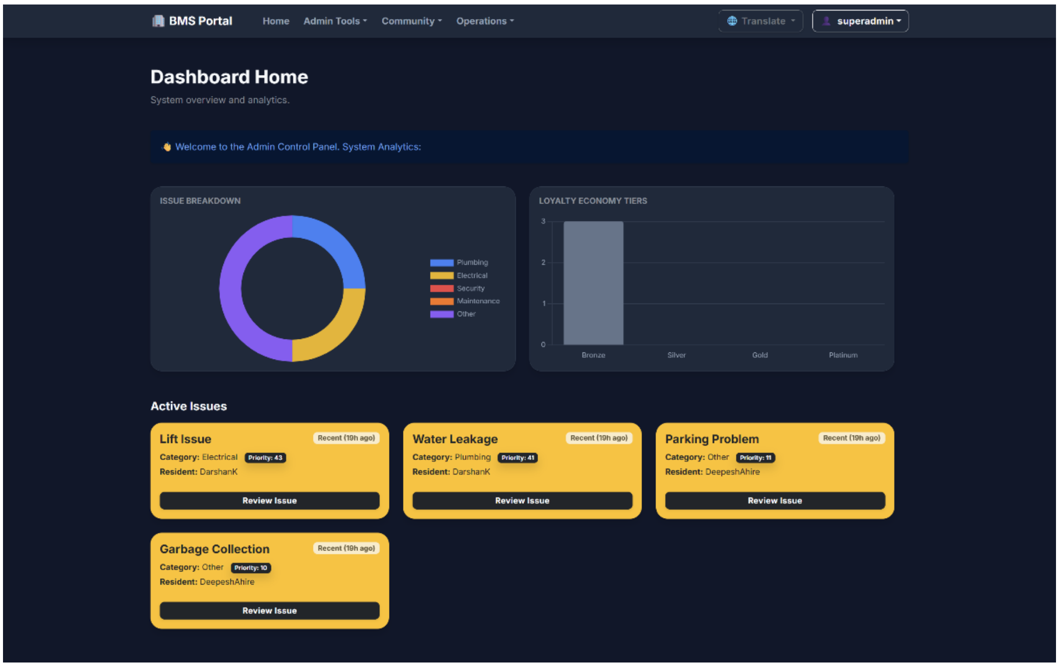

Fig. 10 Admin dashboard displaying system analytics, issue distribution, loyalty tiers, and prioritized active complaints for efficient management.

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154230

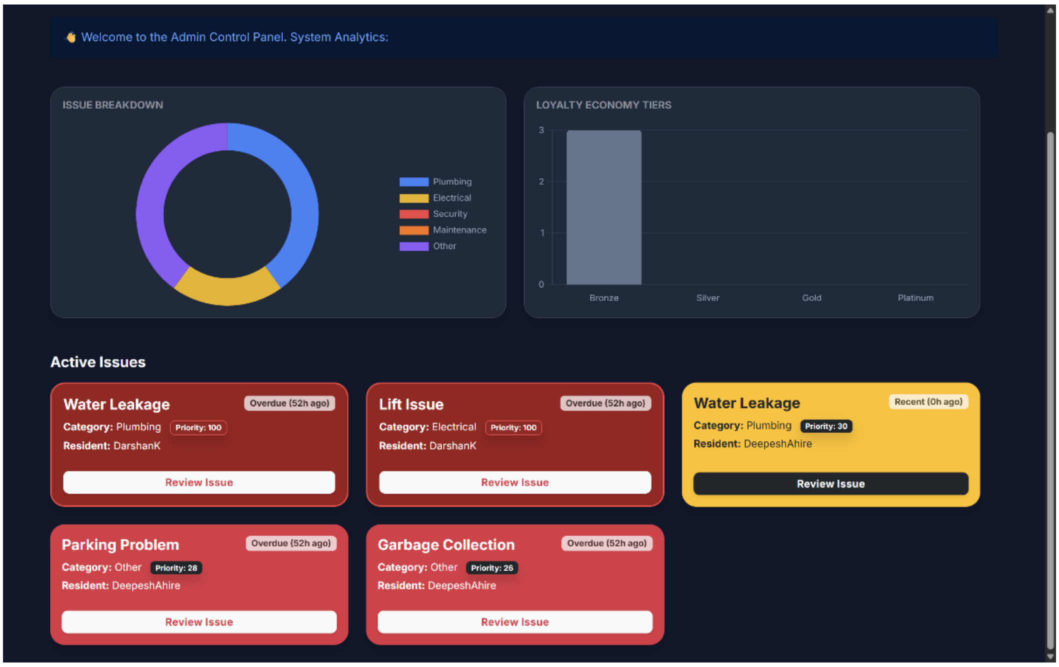

Fig. 11 Admin dashboard highlighting overdue and recent complaints with priority levels.

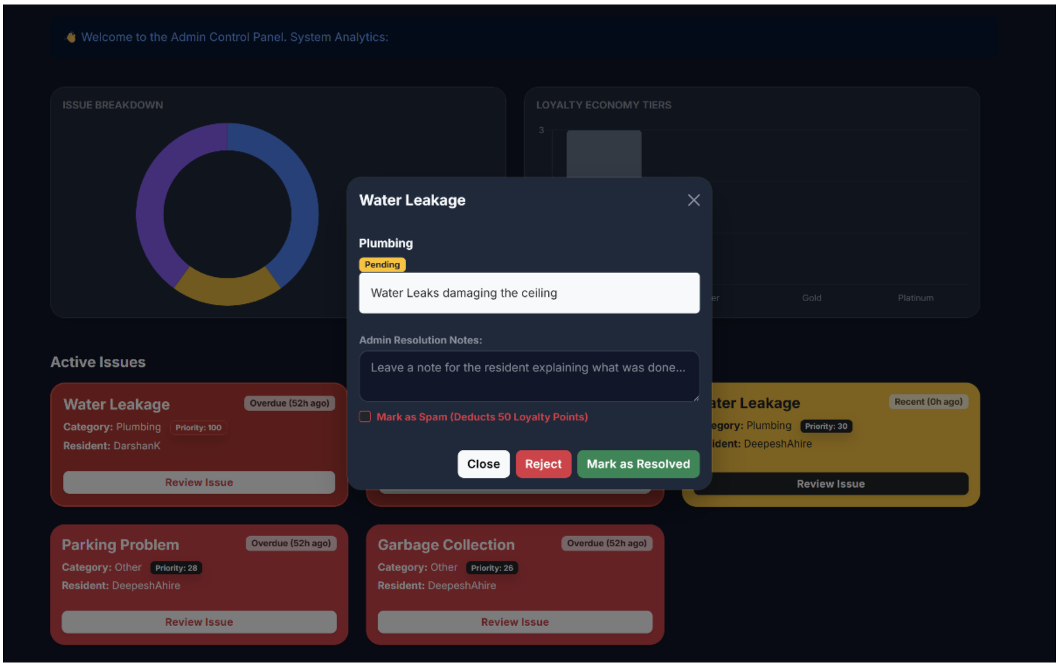

Fig. 12 Admin complaint handling interface with resolution controls.

ISSN (O) 2278-1021, ISSN (P) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁙ Peer-reviewed & Refereed journal ⁙ Vol. 15, Issue 4, April 2026
DOI: 10.17148/IJARCCE.2026.154230

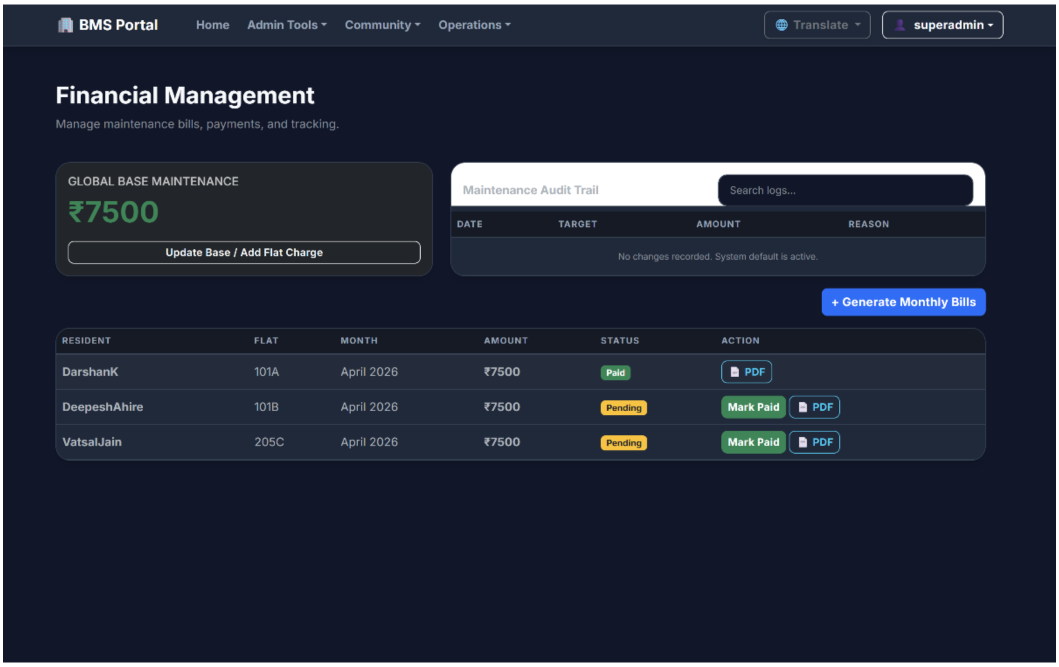

Fig. 13 Administrative dashboard for society financial management and tracking.

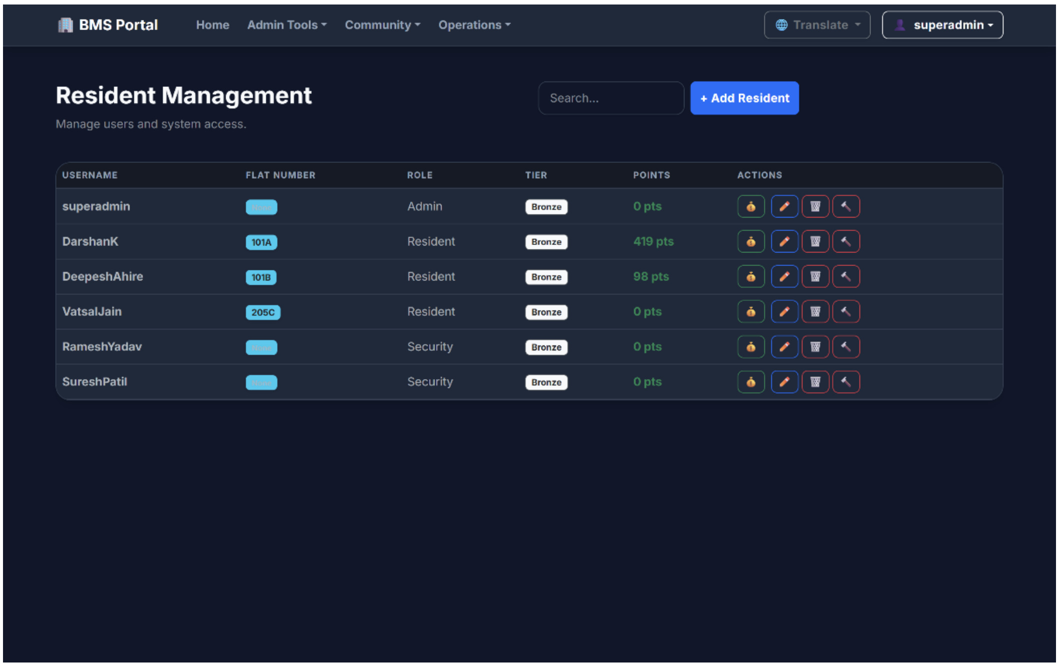

Fig. 14 Admin panel overview for managing users and system data.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁒ Peer-reviewed & Refereed journal ⁒ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154230

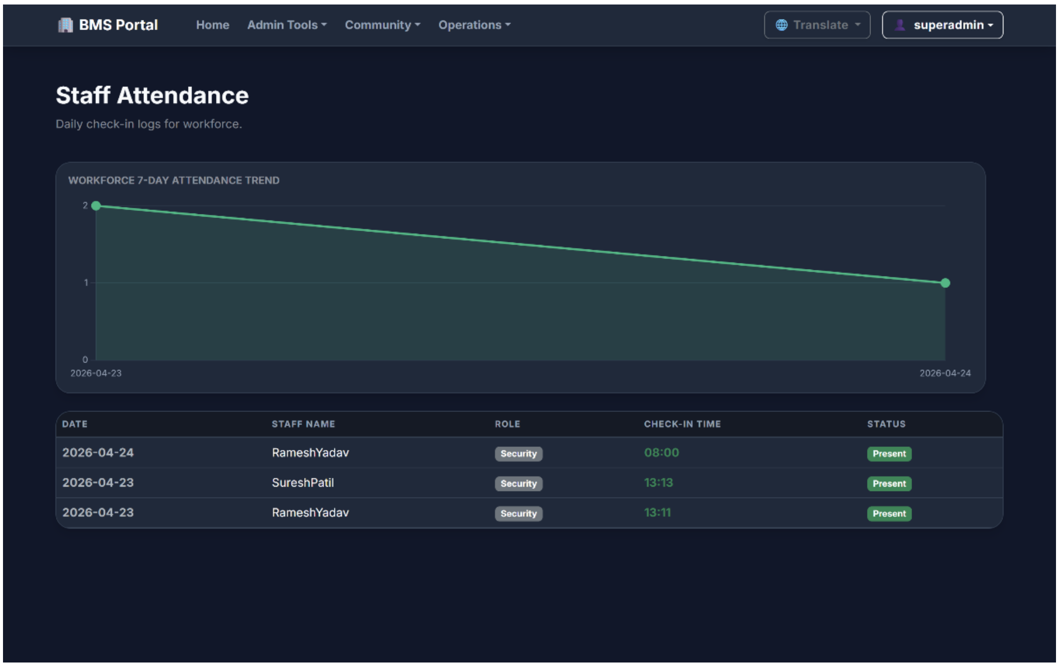

Fig. 15 Staff attendance interface displaying employee presence, status tracking, and attendance records.

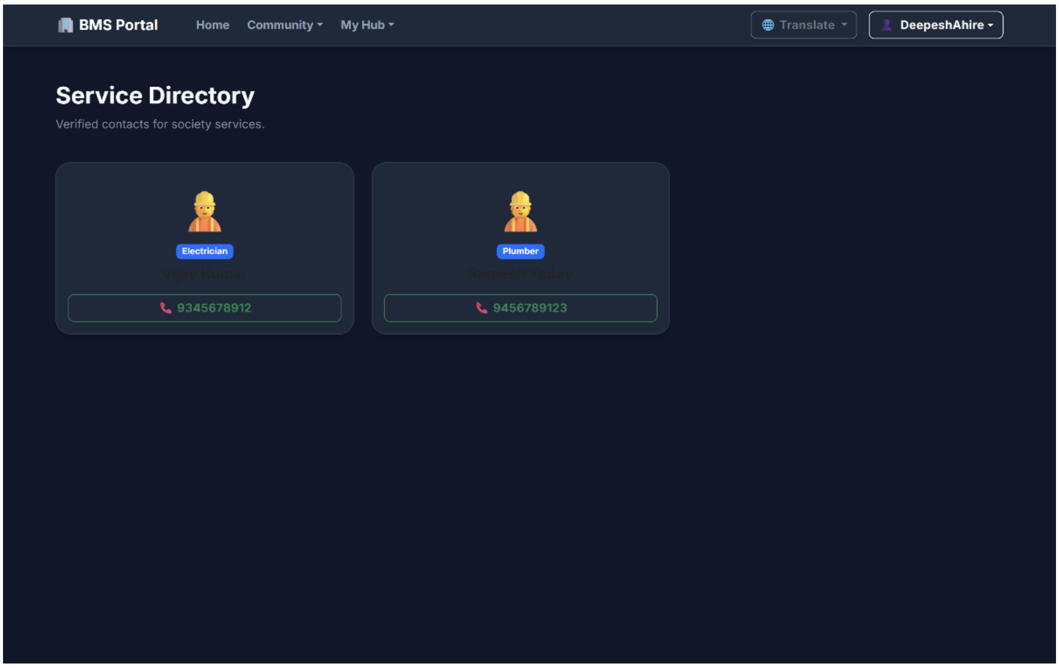

Fig. 16 Service directory listing professionals such as plumbers and electricians for resident access.

V. SYSTEM IMPLEMENTATION

The proposed algorithm is integrated into a Building Management System developed using the following technologies:

- **Frontend:** HTML, CSS, JavaScript
- **Backend:** Node.js with Express.js
- **Database:** MongoDB • **Real-Time Updates:** WebSockets

Residents can submit complaints through a web interface, and the backend dynamically calculates priority scores. The administrator dashboard displays complaints sorted by priority, ensuring efficient task handling.

Unlike traditional systems, the application does not rely solely on chronological ordering. Instead, it continuously recalculates priorities based on changing conditions such as time and user interaction.

VI. EXPERIMENTAL SETUP

To evaluate the performance of the proposed system, a simulation-based approach was used. A dataset of 1000 complaints was generated with randomized attributes including:

- Complaint category
- Submission time
- Description (with/without urgency keywords)
- Number of affected users

Two scheduling methods were compared:

1. FIFO Scheduling
2. Proposed Multi-Factor Algorithm

The system processes one complaint at a time, and each complaint is assigned a random service duration.

VII. RESULTS AND ANALYSIS

The proposed algorithm demonstrated improved performance over FIFO scheduling. The average waiting time was reduced due to better prioritization of critical complaints.

The exponential aging mechanism ensured that older complaints were eventually addressed, eliminating starvation. Lexical urgency detection allowed the system to identify critical situations within the same category, while SLA escalation enforced timely resolution.

Additionally, the crowd impact factor improved fairness by prioritizing complaints affecting multiple users. Overall, the system achieved a balanced trade-off between efficiency and fairness.

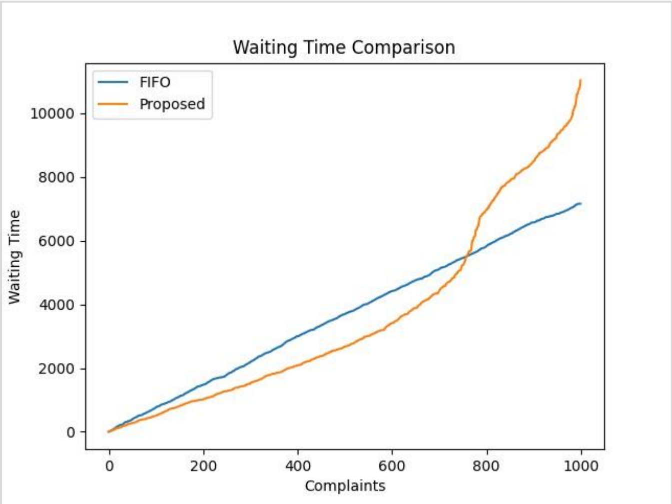

Fig. 17 Waiting time comparison between FIFO and the proposed algorithm.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁒ Peer-reviewed & Refereed journal ⁒ Vol. 15, Issue 4, April 2026

DOI: 10.17148/IJARCCE.2026.154230

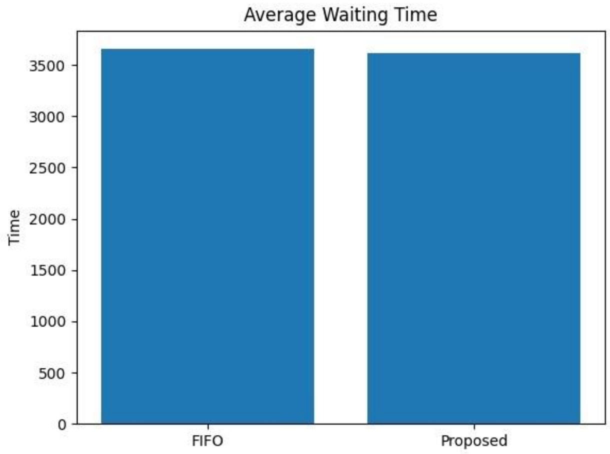

Fig. 18 Average Waiting time comparison between FIFO and the proposed alogorithm.

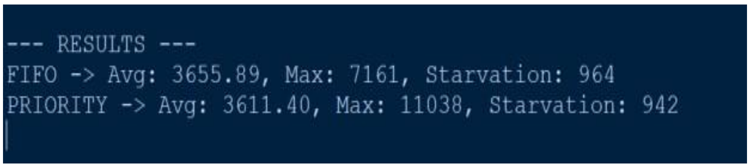

Fig. 19 Final Result

Figure 1 illustrates the comparison of waiting times for complaints processed using traditional FIFO scheduling and the proposed multi-factor priority scheduling algorithm. The x-axis represents the number of complaints (sorted in increasing order of waiting time), while the y-axis denotes the corresponding waiting time experienced by each complaint. From the graph, it can be observed that in the initial range of complaints, the proposed algorithm consistently yields lower waiting times compared to FIFO scheduling. This indicates that the system effectively prioritizes urgent and highimpact complaints, ensuring that critical issues are addressed promptly. The integration of factors such as urgency detection, SLA escalation, and crowd impact enables the system to dynamically adjust priorities, resulting in improved responsiveness for a majority of cases.

As the number of complaints increases, both approaches exhibit a gradual rise in waiting time due to system load. However, beyond a certain point, the waiting time for the proposed algorithm increases more sharply than FIFO. This behavior is attributed to the prioritization mechanism, where lower-priority complaints are deferred in favor of more critical ones. While this leads to increased waiting time for a small subset of complaints, it ensures that high-priority issues are resolved earlier.

Overall, the results demonstrate that the proposed algorithm achieves a more efficient allocation of resources by reducing the average waiting time and improving the handling of critical complaints. Although there is an increase in maximum waiting time for some cases, this trade-off is acceptable in practical scenarios where urgency and impact are more important than strict chronological fairness.

VIII. ADVANTAGES

- Prevents task starvation.
- Context-aware prioritization.
- SLA-based escalation.
- Improved fairness.
- Scalable implementation.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 4, April 2026
DOI: 10.17148/IJARCCE.2026.154230

IX. LIMITATION

- Keyword-based analysis is simplistic.
- Simulation-based evaluation.
- Requires tuning of parameters.

X. CONCLUSION

As urban living environments become increasingly digitized, relying on legacy administrative tools introduces severe operational bottlenecks. This paper successfully demonstrated that traditional static FIFO scheduling models are fundamentally inadequate for handling high-density facility management, as they inevitably lead to task starvation and delayed emergency responses. To resolve this, we introduced a modernized Smart Building Management System that seamlessly integrates a context-aware triage algorithm with a gamified micro-economy. By mathematically capping task priority at a strict ceiling of 100 and utilizing non-linear exponential aging, the proposed architecture guarantees that critical safety hazards instantly preempt routine complaints. Furthermore, by linking automated background cron jobs to a decentralized loyalty ledger, the system actively maps financial behavior to algorithmic trust scores. Experimental results confirm that this multi-variable approach not only optimizes average waiting times but successfully transforms passive property management platforms into intelligent, self-regulating ecosystems.

REFERENCES

[1] M. A. Al-Khafajiy, L. Baker, H. Al-Libawy, A. Waraich, D. Chalmers and M. Omar, "Smart building management system: An IoT approach," *International Conference on Internet of Things and Machine Learning*, pp. 1-6, 2019.
[2] Y. Cheng, "Dynamic Scheduling Algorithm for Internet of Things Based on Multi-Parameter Evaluation," *IEEE Access*, vol. 8, pp. 18234-18245, 2020.
[3] S. Ni, X. Li, and Z. Wang, "An Energy-Aware Dynamic Task Scheduling Algorithm for Smart City Edge Computing," *Sensors*, vol. 21, no. 5, pp. 1658-1672, 2021.
[4] J. Koivisto and J. Hamari, "The rise of motivational information systems: A review of gamification research," *International Journal of Information Management*, vol. 45, pp. 191-210, 2019.
[5] A. Mora, D. Riera, C. González, and J. Arnedo-Moreno, "Gamification: a systematic review of design frameworks," *Journal of Computing in Higher Education*, vol. 29, no. 3, pp. 516-548, 2017.