American Journal of
AI Digital Transformation and Regenerative Pharmacist
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0418 www.zesterapublications.com Original Research Paper

Urban Waste Tracker

A Smart IoT & AI-Based Municipal Solid Waste Management System

Ms. Puja Rani Das
Student, Dept. of CSE
GIFT Autonomous, Bhubaneswar

Mr. Bikash Ranjan Ghadai
Student, Dept. of CSE IoT
GIFT Autonomous, Bhubaneswar

Dr. Pratyush Ranjan Mohapatra
Associate Dean Academics, Dept. of CSE
GIFT Autonomous, Bhubaneswar

ABSTRACT

This paper presents the design and development of an Urban Waste Tracker — an intelligent, IoT-integrated and AI-powered system for real-time monitoring, collection routing, and citizen reporting of municipal solid waste (MSW). Rapid urbanisation has created enormous pressure on waste management infrastructure in Indian cities, resulting in overflowing bins, delayed collection cycles, and environmental degradation. The proposed system addresses these challenges by combining Internet-of-Things (IoT) sensors embedded in public dustbins with a cloud-connected mobile and web application that enables citizens to report waste accumulation, track collection vehicle routes, and view live bin-fill status. An Artificial Intelligence engine processes historical fill-rate data and predicts optimal collection schedules, thereby reducing fuel consumption and operational cost. The back-end is built on Node.js with Express, the mobile front-end on React Native, and the data layer on Firebase Realtime Database. Pilot testing across a simulated urban ward demonstrated a 35% reduction in missed-collection events and a 28% improvement in route efficiency. The results confirm that the platform can serve as a scalable, cost-effective backbone for smart-city waste governance.

Keywords — Urban Waste Management, IoT, Smart Dustbin, AI Route Optimisation, React Native, Firebase, Municipal Solid Waste, Smart City.

I. INTRODUCTION

Urbanisation in India is accelerating at an unprecedented pace. According to the Ministry of Housing and Urban Affairs, Indian cities generated approximately 62 million tonnes of municipal solid waste in 2023, with collection efficiency hovering below 70% in tier-2 and tier-3 cities. Overflowing bins, irregular collection, and a lack of real-time data are the principal causes of this inefficiency. Traditional waste management relies on fixed schedules that ignore the dynamic fill patterns of different bin locations, leading to either premature or delayed collection.

The Internet of Things (IoT) offers a transformative solution by enabling each dustbin to report its fill level, temperature, and odour index to a central cloud platform in real time. When this sensor data is combined with AI-based predictive analytics, waste collection can shift from a calendar-driven activity to a demand-driven one — collecting only when needed and routing vehicles along the most efficient paths.

This project, Urban Waste Tracker, delivers precisely that capability. Citizens can use a mobile application to report overflow, scan a QR code on any registered bin to check its status, and receive gamified incentives for responsible waste segregation. Municipal authorities access a web dashboard that visualises bin occupancy across the city, triggers automated dispatch notifications, and generates weekly analytics reports. The goal is to close the information gap between waste generators, the public infrastructure, and the collection workforce.

II. LITERATURE REVIEW

Research in smart waste management has grown rapidly since the mid-2010s. Early IoT-based systems such as the SmartBin project (Catania et al., 2013) demonstrated that ultrasonic sensors could measure bin fill levels with acceptable accuracy. Subsequent work by Anagnostopoulos et al. (2015) showed that transmitting this data over GSM networks to a cloud platform could enable responsive collection scheduling, reducing overflow incidents by up to 40%.

Parallel advances in machine learning opened new avenues for waste-type classification. Convolutional Neural Network (CNN) based approaches (Aral et al., 2018) achieved greater than 90% accuracy in classifying dry, wet, and recyclable waste from image data, enabling automated segregation at the source. These models, however, demand significant computational resources, creating a barrier to deployment on low-cost embedded hardware.

Received: 20-04-2026 | Accepted: 28-05-2026 | Published: 05-06-2026

American Journal of
AI Digital Transformation and Regenerative Pharmacist
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0418 www.zesterapublications.com Original Research Paper

Closer to the Indian context, Yadav and Bhatt (2020) proposed an RFID-integrated waste truck monitoring system for Ahmedabad municipality. While effective for tracking vehicle movement, the system lacked citizen participation features and did not incorporate predictive route planning. The Urban Waste Tracker addresses both limitations by adding a citizen-facing mobile interface and an AI scheduling module trained on historical sensor data.

Recent literature also highlights the importance of gamification in encouraging waste segregation behaviour (Escrig et al., 2021). Point-reward systems linked to correct bin usage have shown measurable improvement in segregation compliance in pilot programmes across Europe. Our system incorporates a similar incentive model adapted to the Indian urban context.

III. SYSTEM ANALYSIS

A. Existing System Analysis

Existing municipal waste management in most Indian cities is manual and schedule-driven. Collection vehicles follow fixed routes irrespective of whether bins are full or empty, wasting fuel and driver time. Citizen grievance systems, where they exist, are slow, paper-based, or phone-hotline dependent with no real-time acknowledgment. There is no centralised database of bin locations, capacities, or fill histories, making macro-level planning extremely difficult for municipal engineers.

B. Proposed System

The Urban Waste Tracker replaces static scheduling with a sensor-driven, AI-optimised, citizen-participatory ecosystem. Each registered dustbin is fitted with an ultrasonic fill sensor and a microcontroller (ESP32) that publishes readings to Firebase every 10 minutes via Wi-Fi or NB-IoT. A cloud-hosted AI module consumes this stream, predicts when each bin will reach 80% capacity, and auto-generates optimised collection routes for the ward's fleet. Citizens interact through a React Native application while administrators use a React.js web dashboard.

IV. SYSTEM REQUIREMENTS

A. Functional Requirements

The system shall:

- Register, authenticate, and profile citizen users and municipal officer accounts.
- Accept fill-level readings from IoT sensors and store them in the cloud database.
- Allow citizens to report overflowing or damaged bins with photo evidence via the mobile app.
- Display a live city-wide heat-map of bin fill levels on the administrator dashboard.
- Generate AI-optimised daily collection routes and push them to driver mobile devices.
- Issue push notifications to citizens when their reported complaint is resolved.
- Maintain an audit trail of all collection events and sensor readings for at least 12 months.

B. Non-Functional Requirements

The system shall:

- Respond to sensor data ingestion within 2 seconds under a load of 500 concurrent sensors.
- Serve the mobile application with a cold-start time of less than 3 seconds on a 4G connection.
- Store all data on encrypted cloud storage (AES-256) with role-based access control.
- Achieve a minimum uptime of 99.5% per month.
- Comply with the Indian IT Act 2000 and relevant municipal data governance guidelines.

C. Hardware and Software Requirements

Hardware: ESP32 microcontroller, HC-SR04 ultrasonic sensor, MQ-135 gas/odour sensor, SIM7600 NB-IoT module, 5V solar charging module, 3000 mAh LiPo battery pack, Android/iOS smartphone.

Software: Node.js 20 LTS, Express 4.x, React Native 0.73, React.js 18, Firebase Realtime Database and Cloud Functions, Python 3.11 (AI module), scikit-learn, Google Maps Platform API, Visual Studio Code, Android Studio.

V. SYSTEM DESIGN

A. System Architecture

The Urban Waste Tracker employs a four-tier architecture: (1) the Sensor Layer — embedded IoT hardware in each dustbin; (2) the Communication Layer — NB-IoT or Wi-Fi bridges to the cloud; (3) the Application Layer — a Node.js/Express REST API and Python AI micro-service hosted on Google Cloud Run; and (4) the Presentation Layer — React Native mobile apps

American Journal of
AI Digital Transformation and Regenerative Pharmacist
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0418 www.zesterapublications.com Original Research Paper

for citizens and drivers, and a React.js web dashboard for administrators. Firebase Realtime Database acts as the shared data backbone, enabling sub-second synchronisation across all connected clients.

B. ER Diagram

The data model comprises five entities: USER (user_id, name, email, role, ward_id, points), BIN (bin_id, location, capacity_litres, ward_id, sensor_status), SENSOR_READING (reading_id, bin_id, fill_percent, temperature, odour_index, timestamp), COMPLAINT (complaint_id, user_id, bin_id, photo_url, status, created_at, resolved_at), and COLLECTION_EVENT (event_id, vehicle_id, route_json, ward_id, scheduled_time, completion_time). Relationships: one User files many Complaints; one Bin has many Sensor Readings and many Complaints; one Collection Event covers multiple Bins.

C. AI Route Optimisation Module

The route optimisation module uses a Gradient Boosting Regressor trained on 90 days of historical fill-rate data per bin to predict time-to-full for each bin. Predicted fill times are fed into a Nearest Neighbour heuristic for the Capacitated Vehicle Routing Problem (CVRP), partitioning bins into clusters based on collection vehicle capacity and geographic proximity. The module is scheduled to regenerate routes every night at 11:00 PM and updates dynamically if a citizen report triggers a bin status change during the day.

D. Security Design

Firebase Authentication manages all user identity with JWT tokens valid for one hour. Role-based access control separates citizen, driver, and administrator capabilities at both the API and database rule levels. IoT devices authenticate via device-specific HMAC-SHA256 signed payloads to prevent spoofed sensor readings. All REST API endpoints are served over TLS 1.3, and personally identifiable information is encrypted at rest using Firebase's built-in AES-256 encryption.

VI. IMPLEMENTATION

A. Development Environment

The development stack uses JavaScript/TypeScript for the Node.js backend and React Native frontend, Python 3.11 for the AI micro-service, Arduino C++ for the ESP32 firmware, Visual Studio Code as the primary IDE, Android Studio for mobile emulation, and Postman for API testing. All services are containerised with Docker and deployed to Google Cloud Run for auto-scaling capability.

B. IoT Firmware

The ESP32 firmware, written in Arduino C++, initialises the HC-SR04 sensor and takes three distance measurements every 10 minutes, averaging them to derive a fill percentage. The MQ-135 provides a gas index on a 0–100 scale. Both values, along with device timestamp and battery voltage, are packed into a compact JSON payload and published to the Firebase Realtime Database over HTTPS. Deep-sleep mode between readings keeps average power consumption below 40 mW, enabling the solar-charged battery to sustain the device indefinitely in daylight conditions.

C. Backend Implementation

The Node.js/Express server exposes RESTful endpoints for bin registration, sensor data ingestion, complaint submission, route retrieval, and user management. An event listener on the Firebase database node triggers a Cloud Function whenever a bin's fill level crosses 80%, immediately notifying the ward officer and flagging the bin for priority collection. The Python AI service runs as a separate Cloud Run container, exposing a single /predict-routes endpoint that the main server calls nightly.

D. Frontend Implementation

The React Native mobile application offers three role-based views: a Citizen Home screen showing nearby bins, a complaint submission form with camera integration, and a personal points dashboard; a Driver Navigation screen that displays the day's optimised route on Google Maps with turn-by-turn guidance; and an Administrator Dashboard showing the ward heat-map, real-time bin statuses, collection logs, and analytics charts built with Victory Native.

VII. TESTING

A. Objectives of Testing

American Journal of
AI Digital Transformation and Regenerative Pharmacist
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0418 www.zesterapublications.com Original Research Paper

The objectives of testing are to verify end-to-end data flow from IoT sensor to dashboard, validate AI route prediction accuracy, stress-test the backend under concurrent sensor updates, confirm mobile UI correctness across Android and iOS devices, and ensure security controls prevent unauthorised data access.

B. Types of Testing

Unit Testing validated individual modules: sensor parsing functions, route-optimisation algorithms, and Firebase rule expressions. Integration Testing confirmed seamless interaction between the ESP32 firmware, Firebase, the Node.js API, and the React Native client. System Testing exercised the complete pipeline with 50 simulated bin sensors running over a 72-hour window. User Acceptance Testing (UAT) was conducted with a group of 20 participants — 15 citizens and 5 simulated municipal officers — who rated the app usability at 4.3 out of 5.

C. Testing Results

Sensor readings were ingested with a mean latency of 1.4 seconds. The AI module predicted bin fill times within a mean absolute error of 18 minutes against ground-truth overflow timestamps in the holdout test set. Route optimisation reduced total vehicle distance by 28% compared to fixed routes. All security tests, including JWT expiry enforcement and firewall rule validation, passed without exceptions.

VIII. RESULTS AND DISCUSSION

A. Performance Analysis

The Urban Waste Tracker was deployed in a simulated ward environment comprising 50 virtual dustbins and 3 collection vehicles over a two-week evaluation period. Key metrics are summarised below:

Metric	Result
Sensor-to-cloud latency (mean)	1.4 seconds
Bin fill prediction error (MAE)	18 minutes
Route distance reduction	28%
Missed-collection reduction	35%
Mobile app cold-start time	2.7 seconds
UAT usability score	4.3 / 5.0
System uptime (test period)	99.8%

B. Discussion

The results validate the core hypothesis that demand-driven, sensor-informed collection scheduling is substantially more efficient than fixed-calendar approaches. The 28% route distance reduction directly translates to fuel savings and lower carbon emissions — a significant outcome given the scale of urban fleets. The 35% reduction in missed-collection events was achieved primarily through the AI module's proactive flagging of bins approaching overflow rather than reactive citizen reporting alone. Limitations observed during testing include the dependency on reliable Wi-Fi or NB-IoT coverage in all bin locations, the AI model's reduced accuracy for bins in high-footfall areas with irregular fill patterns, and the battery drain caused by the MQ-135 sensor in ambient temperatures above 35°C — a concern relevant to Odisha summers. These issues will be addressed in the next development cycle.

IX. FUTURE ENHANCEMENTS

Several enhancements are planned to broaden the system's impact and technical depth:

- Computer Vision Segregation Aid: Integration of a lightweight CNN model on the mobile app to classify waste type (dry / wet / e-waste / hazardous) from a camera image before it is deposited, guiding citizens towards the correct bin.
- Blockchain Audit Trail: Immutable recording of collection events and citizen complaint resolutions on a permissioned blockchain to ensure transparency and accountability for municipal authorities.

American Journal of
AI Digital Transformation and Regenerative Pharmacist
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0418 www.zesterapublications.com Original Research Paper

- Multilingual Voice Interface: Odia, Hindi, and English voice commands within the mobile application to maximise accessibility for low-literacy users.
- Predictive Bin Maintenance: Anomaly detection on sensor streams to identify failing hardware before it causes data gaps.
- Integration with Smart City Command Centres: Publishing bin and route data as an open API for consumption by Smart City dashboards operating under the MoHUA Smart Cities Mission.
- Carbon Credit Tracking: Automated calculation and reporting of emission reductions achieved through optimised routing to facilitate participation in voluntary carbon credit markets.

X. CONCLUSION

The Urban Waste Tracker successfully integrates Internet of Things sensing, cloud computing, artificial intelligence, and mobile application development into a unified platform for smart municipal solid waste management. The system was designed, implemented, and tested against realistic operational scenarios, demonstrating significant improvements in collection efficiency, missed-collection rates, and user satisfaction.

The project illustrates how modern software and embedded technologies can directly address one of the most visible urban governance challenges in rapidly growing Indian cities. By giving voice to citizens, intelligence to infrastructure, and visibility to administrators, the Urban Waste Tracker lays a practical foundation for the next generation of smart-city environmental services. Future work will focus on scaling the system to a live municipal ward, refining the AI module with real-world training data, and exploring sustainability through solar-powered sensor nodes and community-owned infrastructure models.

REFERENCES

[1] Ministry of Housing and Urban Affairs, Government of India, "Annual Report on Municipal Solid Waste Management," New Delhi, 2023.

[2] V. Catania, D. Ventura, "An Approach for Monitoring and Smart Planning of Urban Solid Waste Management Using Smart-M3 Platform," Proceedings of the 15th Conference of FRUCT, 2013.

[3] T. Anagnostopoulos, A. Zaslavsky, A. Medvedev, S. Khoruzhnikov, "Robust Waste Collection Exploiting Cost Efficiency of IoT Potentials in Smart Cities," Proceedings of the 1st International Conference on Recent Advances in Information Technology, 2015.

[4] R. A. Aral, S. R. Keskin, M. Kaya, M. Hacıömeroğlu, "Classification of TrashNet Dataset Based on Deep Learning Models," IEEE International Conference on Big Data, 2018.

[5] A. Yadav, P. Bhatt, "IoT-Based Smart Waste Management System for Smart City," International Journal of Innovative Technology and Exploring Engineering, vol. 9, no. 4, 2020.

[6] R. Escrig, J. M. Fuertes-Pallás, M. A. Brun-Usan, "Gamification for Waste Segregation: A Systematic Review," Sustainability, vol. 13, no. 18, 2021.

[7] Google Firebase Documentation. [Online]. Available: https://firebase.google.com/docs

[8] Espressif ESP32 Technical Reference Manual. [Online]. Available: https://www.espressif.com/en/support/documents/technical-documents

[9] React Native Official Documentation. [Online]. Available: https://reactnative.dev/docs/getting-started

[10] Node.js Documentation, OpenJS Foundation. [Online]. Available: https://nodejs.org/en/docs

[11] Google Maps Platform — Directions API. [Online]. Available: https://developers.google.com/maps/documentation/directions

[12] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.