www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 12, December 2025, pp : 2111-2114

e-ISSN : 2583-1062
Impact Factor : 7.001

GARBAGE MONITORING SYSTEM

Namrata Brahma Prakash[1], Krithika M[2], Keerthi G R[3], Varun R[4], Naveen M[5]

[1]Assistant professor, Computer Science and Engineering, T. john Institute of Technology, Bengaluru, Karnataka, India.

[2,3,4,5]Student, Computer science and Engineering, T. john Institute of Technology, Bengaluru, Karnataka, India.

DOI: https://www.doi.org/10.58257/IJPREMS37904

ABSTRACT

This study investigates a garbage monitoring system designed to address waste management challenges efficiently. The system employs smart sensors to monitor garbage levels in bins and transmit data to a central server via IoT technology. The design utilizes microcontrollers and wireless communication modules to provide real-time updates on bin status. The analysis highlights the effectiveness of the system in optimizing collection schedules, reducing operational costs, and minimizing environmental hazards. Results indicate a significant improvement in waste management efficiency, with a potential reduction in overflow incidents. The study concludes that the proposed system enhances urban cleanliness and contributes to sustainable waste management practices.

Keywords: Garbage monitoring, IoT, waste management, smart sensors, real-time updates, environmental sustainability

1. INTRODUCTION

Effective solid waste management is a critical component of urban infrastructure, essential for mitigating environmental degradation and ensuring public health. Traditional waste collection methods often suffer from inefficiencies, including untimely collection and resource wastage, leading to issues such as overflowing bins and increased operational costs. The integration of Internet of Things (IoT) technologies presents a transformative approach, enabling real-time monitoring and dynamic optimization of waste collection systems. Current research emphasizes the deployment of smart sensors, wireless communication modules, and cloud-based analytics to provide continuous data on waste levels, facilitating predictive maintenance and route optimization. This study introduces a garbage monitoring system that leverages IoT-enabled smart sensors for real-time bin status tracking, data transmission, and centralized decision-making. By enhancing operational efficiency and reducing environmental impacts, the proposed system represents a significant advancement in sustainable waste management technologies.

2. METHODOLOGY

The garbage monitoring system utilizes IoT-enabled smart sensors to track garbage levels in real-time, transmitting data via wireless communication to a centralized server. Performance analysis involved testing sensor accuracy, data transmission reliability, and optimization of waste collection schedules, demonstrating the system's efficiency in reducing overflow incidents and operational costs.

Requirement Gathering: All the stakeholders, such as the QA managers and the operational staff, were engaged in order to come up with the gaps that exist in the current manual garbage collection systems. These discussions revealed issues such as inefficient collection schedules, overflow incidents, and lack of real-time monitoring, which led to the need for an automated garbage monitoring system.

Workshops System and Design Interviews: The design phase for the garbage monitoring system focused on identifying functional and non-functional requirements, followed by the development of detailed architectural diagrams and prototypes. The system was designed to send alert notifications to authorities when garbage bins reach full capacity or experience overflow. For the user interface, tools such as Figma were used to design a clear and intuitive notification dashboard. The architecture was structured to ensure smooth integration of the frontend, backend, and database, enabling real-time data collection from sensors, processing, and timely alert generation for authorities.

Development: The system was developed using Agile methodology, with a focus on iterative development and regular feedback. Key technologies included:

Frontend Development: React Native and Flutter for cross-platform compatibility, providing authorities with a user-friendly interface for receiving alert notifications.

Backend Development: Java for handling business logic and managing database interactions, ensuring timely alert generation and seamless communication between sensors and authorities regarding garbage bin status.

Database: MySQL for structured data storage.

Testing: In this paper, the approach used to design the garbage monitoring system included implementing a comprehensive testing strategy. Unit Testing was performed to validate the reliability and effectiveness of individual components, such as sensors and alert notification triggers. Response Testing was carried out to ensure the timely and accurate delivery of alert notifications to authorities when garbage bins reach full capacity or experience overflow. Additional comprehensive testing types, including performance and stress testing, were conducted to assess the system's reliability and performance under varying conditions.

Unit Testing: To validate individual components.

Integration Testing: To ensure smooth communication between system components, integration testing was performed, focusing on data flow from sensors to the backend and notification delivery to authorities.

System Testing: The system's overall performance was evaluated under real-world conditions to ensure reliable and timely alert notifications for authorities when garbage bins reached full capacity or experienced overflow.

Deployment: The application was deployed gradually in stages, beginning with a controlled 'pilot' test in select areas. Feedback from this phase was used to refine the system and prepare for full-scale implementation.

Maintenance and Updates: A maintenance strategy was developed to address bug fixing, implement future updates, and introduce new features based on user feedback, ensuring continuous improvement in the alert notification system and overall performance.

3. PROJECT OVERVIEW

The system digitizes the waste management process by providing real-time tracking of garbage levels and sending automated alert notifications to authorities when bins reach full capacity or experience overflow. It enhances operational efficiency with features like real-time data monitoring, automated scheduling, and seamless communication between sensors and authorities. The system also supports scalability, ensuring that it can be expanded to cover more areas while maintaining reliable alert delivery and improving waste collection efficiency.

4. SYSTEM OVERVIEW

The Garbage Monitoring System is a mobile application designed to enhance waste management by providing real-time tracking of garbage levels and sending automated alert notifications to authorities when bins reach full capacity or experience overflow. Its intuitive interface allows both technically proficient and non-proficient users to easily monitor garbage status, receive notifications, and take timely actions. The system's automatic features for monitoring bin levels, generating alerts, and optimizing waste collection schedules streamline the entire waste management process, ensuring a cleaner, more efficient urban environment.

5. KEY FEATURES

- Real-time monitoring of garbage bin levels with automated alert notifications to authorities when bins are full or overflowing.
- Role-based access and permissions for authorities and waste management personnel.
- Notification alerts accessible on mobile and web platforms for timely communication.
- Data tracking and reporting for efficient waste collection scheduling and optimization.

6. LITERATURE SURVEY

Modern waste management systems increasingly utilize IoT and smart sensors for real-time monitoring and data-driven decision-making. Recent studies have explored the integration of IoT-based systems for detecting garbage bin levels and notifying relevant authorities to optimize waste collection schedules (Singh et al., 2023). Additionally, predictive analytics has been applied to forecast waste accumulation trends, enabling municipalities to allocate resources more efficiently (Sharma et al., 2022). The use of robust offline functionality ensures that alert systems remain operational even in areas with intermittent network connectivity, a critical feature for rural deployments (Kumar and Patel, 2023). By incorporating these technological advancements, the Garbage Monitoring System addresses the limitations of traditional waste management practices, enhancing responsiveness and operational efficiency while contributing to the development of smart urban infrastructure.

7. ARCHITECTURE

Architectural Overview: The system's architecture seamlessly integrates multiple components to function as a unified whole: a responsive mobile front end for user interactions, a powerful backend for real-time processing, and a reliable database for secure data storage.

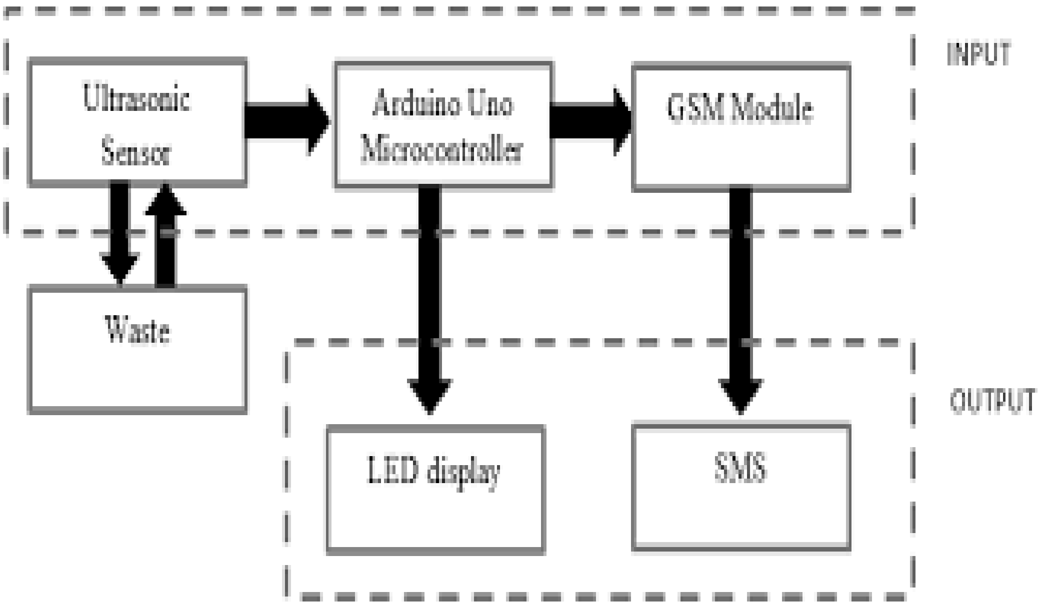
Frontend: A user-friendly mobile interface, developed using React Native or Flutter, allows authorities to receive and manage alert notifications intuitively and efficiently.

Backend: At the core of the system, a robust Java-based backend handles the processing of sensor data, generating and dispatching timely alert notifications to authorities with precision.

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 04, Issue 12, December 2025, pp : 2111-2114
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

Database: A MySQL database anchors the architecture by securely storing garbage bin status, user information, and notification history, ensuring data integrity and accessibility.

Backend Services: Backend services act as the system's neural network, coordinating real-time monitoring of garbage bin levels, processing sensor data, and triggering alert notifications to authorities with seamless synchronization.



Context analysis diagram

DESIGN

Supervisor Design:

The system is designed to facilitate seamless interactions between authorities and backend services. Users can comfortably navigate the interface with an intuitive layout and responsive design, ensuring timely access to alert notifications and relevant data in a user-friendly environment.

User Interface Design:

Leveraging Figma wireframes, the interface provides an aesthetically pleasing and highly functional user experience. Authorities can easily monitor garbage bin levels and respond to notifications via the intuitive design.

INTEGRATION

Backend Syncing:

Garbage bin data is synced in real time, ensuring consistency across all platforms and accurate alert notifications.

Query Parameters and Input Data:

The system efficiently processes input data from IoT sensors, ensuring precise and secure delivery of alerts to authorities while minimizing risks of data exposure.

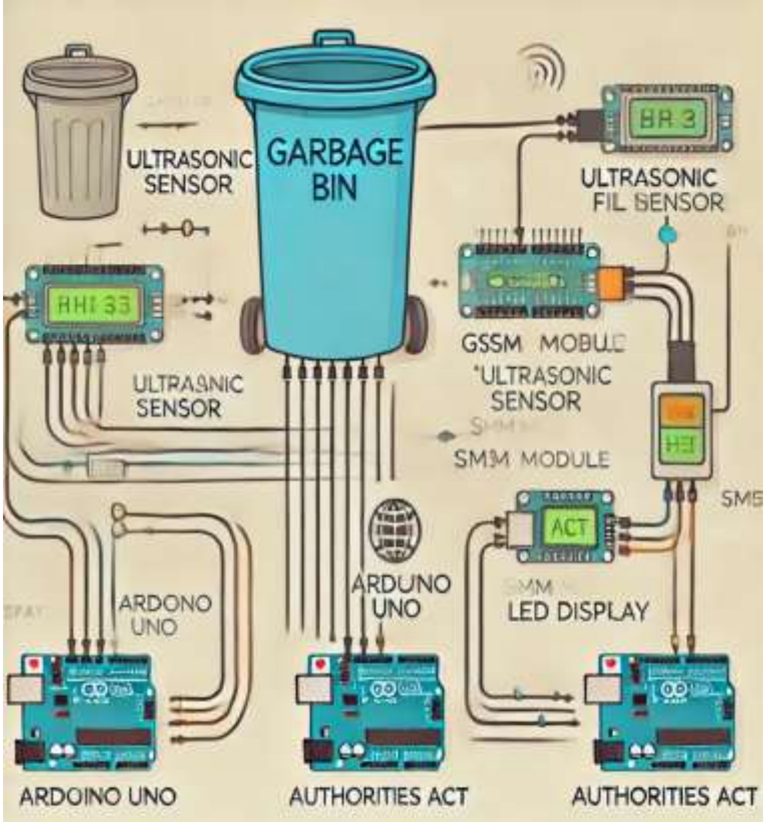
8. TESTING AND VALIDATION

Testing Plan:

The system is evaluated using test cases for functionality, performance, and user experience.

Test Case ID	Description	Expected Result	Status
TC-001	Monitor garbage bin levels	Accurate status updates for bins	Pass/Fail
TC-005	Trigger alert notifications	Authorities receive timely alerts	Pass/Fail
TC-006	Sync data across platforms	Consistent and updated bin data on all platforms	Pass/Fail
TC-009	Ensure data integrity	Data remains consistent after operations	Pass/Fail
TC-010	Validate alert delivery	Correct authority details displayed and notified	Pass/Fail

IMPLEMENTATION AND DESIGN



TECHNOLOGY STACK

Frontend: React Native, Flutter

Backend: Java

Database: MySQL

Wireframes: Figma

IDEs: Visual Studio Code, Android Studio

9. CONCLUSION

The Garbage Monitoring System uses ultrasonic sensors (for garbage level detection), Arduino Uno (for data processing), GSM module 800C (for alerts), and LED displays (for real-time feedback). The system is supported by a Java backend, MySQL for data storage, and a mobile app built with React Native or Flutter, ensuring efficient monitoring and alerting.

10. REFERENCES

[1] Aalok V. Agarwala, Paul G. Firth, Meredith A. Albrecht, Lisa Warren, and Guido Musch. 2015. An electronic checklist improves transfer and retention of critical information at intraoperative handoff of care. *Anesth. Analg.* 120, 1 (January 2015), 96–104.

[2] Jochen Bergs, Johan Hellings, Irina Cleemput, Ö Zurel, Vera De Troyer, Monique Van Hiel, Jean-Luc Demeere, Donald Claeys, and Dominique Vandijck. 2014. Systematic review and meta-analysis of the effect of the World Health Organization surgical safety checklist on postoperative complications. Br J Surg 101, 3 (February 2014), 150–158.

[3] Aalok V. Agarwala, Paul G. Firth, Meredith A. Albrecht, Lisa Warren, and Guido Musch. 2015. An electronic checklist improves transfer and retention of critical information at intraoperative handoff of care. Anesth. Analg. 120, 1 (January 2015), 96–104.

[4] Aleksandra Sarcevic, Brett Rosen, Leah Kulp, Ivan Marsic, and Randall Burd. 2016. Design challenges in converting a paper checklist to digital format for dynamic medical settings. In Proceedings of the 10th EAI International Conference on Pervasive Computing Technologies for Healthcare (Pervasive Health 2016*)*

[5] Heidi S. Kramer and Frank A. Drews. Checking the lists: A systematic review of electronic checklist use in health care. Journal of Biomedical Informatics