

Cardiff Metropolitan University

**Smart Waste Bin Management System Using IoT and Machine Learning**

Submitted in April 2024

By

**Rideegammana Gedara Pramod Sandakelum**

(Student ID - 20261236 – GM/BSCSD/04/08)

*This dissertation is submitted in partial fulfillment*

*Of the requirements for the degree of*

**BSc (Hons) Software Engineering**

**DECLARATION**

This work is being submitted in partial fulfillment of the requirements for the degree of

BSc (Hons) Software Engineering and has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed - (Candidate)

Date - 09/02/2025

**SUPERVISOR’S DECLARATION STATEMENT**

Student Name – Rideegammana Gedara Pramod Sandakelum

Supervisor’s Name - Mrs. Vindya Karunarathna

I acknowledge that the above-named student has regularly attended the meeting and actively engaged in the dissertation supervision process.

Signed - ……………………………………………. (Supervisor)

Date - …………………......

Contents

[Acknowledgment 7](#_Toc207721586)

[Abstract 8](#_Toc207721587)

[1. Introduction 9](#_Toc207721588)

[1.1 Background 9](#_Toc207721589)

[1.2 Problem Statement 10](#_Toc207721590)

[1.3 Objectives 11](#_Toc207721591)

[1.4 Proposed Solutions 12](#_Toc207721592)

[2. Literature Review 14](#_Toc207721593)

[2.1 Overview of Smart Waste Management Systems 14](#_Toc207721594)

[2.2 Sensor Technologies for Waste Bin Fill-Level Detection 14](#_Toc207721595)

[2.3 Weight Sensors: Advantages, Challenges, and Integration 15](#_Toc207721596)

[2.4 Case Studies & Real-World Implementations 16](#_Toc207721597)

[2.5 Status in Sri Lanka 17](#_Toc207721598)

[3. Planning 18](#_Toc207721599)

[3.1 Feasibility Report 18](#_Toc207721600)

[3.2 Risk Assessment 19](#_Toc207721601)

[3.3 SWOT Analysis 20](#_Toc207721602)

[3.4 PESTEL Analysis 21](#_Toc207721603)

[3.5 Life Cycle Model 22](#_Toc207721604)

[4. Requirement Gathering and Analysis 24](#_Toc207721605)

[4.1 Techniques Employed 25](#_Toc207721606)

[4.2 Key Findings from User Responses 26](#_Toc207721607)

[4.2.1 Waste Disposal Methods and Collection Frequency 26](#_Toc207721608)

[4.2.2 Overflowing Bins 27](#_Toc207721609)

[4.2.3 Main Waste Management Challenges 28](#_Toc207721610)

[4.2.4 Awareness and Support for Smart Waste Bin Systems 29](#_Toc207721611)

[4.3 Insights from Interviews 31](#_Toc207721612)

[4.3.1 Integration of Questionnaire and Interview Data 32](#_Toc207721613)

[4.3.2 Alignment with National Policies 33](#_Toc207721614)

[4.3.3 Conclusion 34](#_Toc207721615)

[4.4 Functional and Non-Functional Requirements 34](#_Toc207721616)

[4.4.1 Functional Requirements 35](#_Toc207721617)

[4.4.2 Non-Functional Requirements 36](#_Toc207721618)

[5. System Design 38](#_Toc207721619)

[5.1 Architecture Diagram 39](#_Toc207721620)

[5.2 ER Diagram (Entity-Relationship Diagram) 42](#_Toc207721621)

[5.3 UML Diagrams 45](#_Toc207721622)

[5.4 UI Diagrams 49](#_Toc207721623)

[6. Implementation 52](#_Toc207721624)

[6.1 Technology Stack 52](#_Toc207721625)

[6.2 Design Patterns 55](#_Toc207721626)

[6.3 Detailed Implementation Process 56](#_Toc207721627)

[6.4 Contribution to a Better World 60](#_Toc207721628)

[7. Testing and Validation 61](#_Toc207721629)

[7.1 Test Plan 61](#_Toc207721630)

[Objectives: 61](#_Toc207721631)

[Scope: 61](#_Toc207721632)

[Testing Types: 61](#_Toc207721633)

[Test Environment: 62](#_Toc207721634)

[7.2 Test Cases 62](#_Toc207721635)

[8.0 Conclusion 64](#_Toc207721636)

[8.1 Conclusion 64](#_Toc207721637)

[8.2 Future Recommendations 65](#_Toc207721638)

[8.3 Lessons Learned 65](#_Toc207721639)

[Gantt Chart 66](#_Toc207721640)

[References 67](#_Toc207721641)

[Appendix 1: 72](#_Toc207721642)

[Questionaire and Answers 72](#_Toc207721643)

[Source Code of the Project 72](#_Toc207721644)

[Appendix 2: 73](#_Toc207721645)

[Figure 1 Agile Model 22](#_Toc207721657)

[Figure 2 Waste Disposal Methods and Collection Frequency 26](#_Toc207721658)

[Figure 3 Overflow Frequency 27](#_Toc207721659)

[Figure 4 Main Waste Management Challenges 28](#_Toc207721660)

[Figure 5 Awareness and Support for Smart Waste Bin Systems 29](#_Toc207721661)

[Figure 6 Preferred Features 30](#_Toc207721662)

[Figure 7 IOT Flow 38](#_Toc207721663)

[Figure 8 Cloud Layer 39](#_Toc207721664)

[Figure 9 Notifications 40](#_Toc207721665)

[Figure 10 System Architecture 41](#_Toc207721666)

[Figure 11 ER Diagram Postgre 42](#_Toc207721667)

[Figure 12 ER Diagram 43](#_Toc207721668)

[Figure 13 Use Case Diagram 44](#_Toc207721669)

[Figure 14 Sequence Diagram 46](#_Toc207721670)

[Figure 15 Activity Diagram 47](#_Toc207721671)

[Figure 16 Dashboard Top 48](#_Toc207721672)

[Figure 17 Dashboard Summary and Statistics 48](#_Toc207721673)

[Figure 18 Bins Summary 49](#_Toc207721674)

[Figure 19 Bin Details View 49](#_Toc207721675)

[Figure 20 AI Predictor 50](#_Toc207721676)

[Figure 21 Route Planner 50](#_Toc207721677)

[Figure 22 ESP Code MQTT Update 55](#_Toc207721678)

[Figure 23 MQTT Cloud Dashboard 55](#_Toc207721679)

[Figure 24 Postgres Insertion 56](#_Toc207721680)

[Figure 25 Online Postgres DB 56](#_Toc207721681)

[Figure 26 AI Model for Predictions 57](#_Toc207721682)

[Figure 27 AI Model Hosted In Render 57](#_Toc207721683)

[Figure 28 Dashboard Display 1 Filled Bin 58](#_Toc207721684)

[Figure 29 Gantt Chart 65](#_Toc207721685)

[Table 1 Sensor Comparison 16](#_Toc207721646)

[Table 2 Risk Assesment 19](#_Toc207721647)

[Table 3 SWOT Analysis 20](#_Toc207721648)

[Table 4 PESTEL Analysis 21](#_Toc207721649)

[Table 5 Time Plan 23](#_Toc207721650)

[Table 6 waste disposal and collection 26](#_Toc207721651)

[Table 7 Overflow Frequency 27](#_Toc207721652)

[Table 8 Main Waste Management Challenges 28](#_Toc207721653)

[Table 9 Awareness and Support for Smart Waste Bin Systems 29](#_Toc207721654)

[Table 10 Preferred Features 30](#_Toc207721655)

[Table 11 Test Cases 62](#_Toc207721656)

Acknowledgment

I would like to extend my deepest gratitude to everyone who played a significant role in the successful completion of my final project. This endeavor has been a challenging yet rewarding journey, and I am truly grateful for the support and guidance I received.

First and foremost, I would like to express my heartfelt thanks to all my esteemed lecturers, **Mrs. Sankani Heenkenda, Mrs. Vijini Mekhala, Prof. Amal Rajapaksha and My Project Supervisor Mrs. Vindya Karunarathna** for their continuous support, insightful feedback, and valuable suggestions were instrumental in guiding me throughout this project. Their encouragement and expertise have greatly enriched my work and overall learning experience.

I also wish to acknowledge the contributions of the **ICBT Campus Gampaha**. The provision of necessary resources and the supportive environment for research and development were critical to the success of my project. The institution’s unwavering support and belief in my capabilities have been a cornerstone of this achievement.

I am equally grateful to my **family and friends** for their constant encouragement, patience, and understanding during the demanding periods of this project. Their support provided me with the strength and motivation to persevere and stay focused on my goals.

Lastly, I would like to thank all those who contributed, directly or indirectly, to the completion of this project. Your support has been invaluable, and I am deeply appreciative of your contributions.

Thank you all for your unwavering support and guidance throughout this journey.

Abstract

Urbanization has intensified pressure on waste management systems, causing inefficiencies such as overflowing bins, irregular collection schedules, and increased operational costs. Traditional collection methods are reactive and labor-intensive, often leading to environmental and public health concerns. To address these issues, this project introduces a **Smart Waste Bin Management System** that combines IoT sensors, data analytics, and machine learning to enable intelligent monitoring and collection planning.

The main objective is to optimize waste collection through real-time bin monitoring, predictive analysis, and route planning, ensuring timely collection while minimizing costs and inefficiencies. The methodology followed the software development lifecycle, including requirements gathering, design, implementation, and testing. IoT-enabled bins were fitted with weight sensors to monitor fill levels. Data collected is transmitted to a central server, processed, and used by a predictive model to forecast bin status. A **web-based dashboard** was developed to visualize bin locations, fill percentages, and collection schedules.

Key tools and technologies employed in the project include **ESP32-based sensors, Python, Render for backend processing, Postgres SQL for data storage, and React.js/CSS/JavaScript for the web interface**. Testing and validation were performed through structured test cases to ensure accuracy, reliability, and user satisfaction.

The system achieved an **Accuracy level of over 85% in predicting bin fill levels**, which significantly contributes to proactive scheduling of collection routes. Furthermore, the implementation demonstrated improved efficiency in collection logistics and reduction in unnecessary trips, directly supporting cost savings and environmental benefits.

In conclusion, the Smart Waste Bin Management System successfully fulfills its objectives by providing a scalable, efficient, and sustainable approach to waste management. The project’s contribution lies in its ability to transform traditional waste collection into a **data-driven, automated, and environmentally conscious process**, laying the groundwork for future smart city initiatives.

1. Introduction

1.1 Background

Urbanization and population growth are driving a sharp increase in municipal solid waste worldwide. The World Bank (2018) predicts global urban waste generation will rise from 2.01 billion tons in 2016 to 3.4 billion tons by 2050 if no interventions are made. Cities face pressure on infrastructure, operational costs, and environmental quality.

In Sri Lanka, approximately 18.7 million people live in urban areas (Department of Census and Statistics, 2022), generating an estimated 7,000 metric tons of waste daily (CEA, 2023). The Western Province, including Colombo, accounts for nearly 60% of this total. Traditional fixed-route, schedule-based collection methods are increasingly inadequate, leading to overflowing bins, public health hazards, inefficient fuel use, and unnecessary labor costs (Karunasena, 2021).

Modern technologies such as the Internet of Things (IoT), cloud computing, and Artificial Intelligence (AI), particularly Machine Learning (ML), offer opportunities to transform waste management. IoT-enabled bins equipped with sensors can monitor fill levels, temperature, and hazardous gases in real-time. ML models can predict waste accumulation trends, enabling dynamic collection scheduling and route optimization (Rahman et al., 2022).

Globally, cities like Seoul, Dubai, and Barcelona have successfully implemented Smart Waste Bin Management Systems, reducing collection frequency by up to 20–30% and optimizing resource utilization (Lee, 2019; Smart Waste Management, 2020; Poonia, 2022). In Sri Lanka, however, no municipal council has adopted IoT-enabled bins or AI-driven route optimization (Ministry of Environment Sri Lanka, 2024), leaving a gap for technological intervention.

1.2 Problem Statement

Sri Lanka’s waste management faces multiple challenges:

1. **Inefficient collection methods:** Fixed routes and schedules do not consider actual bin fill levels. Overflowing bins cause health hazards (vectors like rodents and mosquitoes) and environmental pollution, while half-empty bin collections waste fuel, labor, and increase emissions (Silva, 2020; National Waste Generation Statistics, 2023).
2. **Lack of real-time monitoring:** Without live data, authorities rely on static schedules or citizen complaints, resulting in delayed responses to illegal dumping, fires, or waste spikes during festivals or market days (University of Moratuwa, 2022).
3. **Operational constraints:** Many bins are old, unstandardized, or difficult to retrofit with sensors. Municipal budgets are limited, technical expertise is scarce, and public awareness of proper disposal practices is low (Karunasena, 2021).
4. **Environmental impact:** Unnecessary trips increase fuel consumption and greenhouse gas emissions, conflicting with Sri Lanka’s Nationally Determined Contributions under the Paris Agreement (Ministry of Environment Sri Lanka, 2024).
5. **Pandemic considerations:** COVID-19 highlighted the need for contactless, automated waste monitoring to minimize worker exposure (Waste Management and COVID-19 Guidelines, 2020).

Globally, IoT and AI-based systems have reduced collection costs by up to 35% while improving environmental and operational outcomes (Jayaraman, 2017; Poonia, 2022). Sri Lanka lacks large-scale deployments, though pilot projects (e.g., University of Moratuwa, 2022) show promising efficiency gains of 25–30%.

1.3 Objectives

The project aims to develop a **Smart Waste Bin Management System (SWBMS)** that integrates IoT, ML, cloud computing, and user-friendly interfaces to improve urban waste management in Sri Lanka. Specific objectives include:

1. **IoT-enabled Smart Bins:** Design durable, weather-resistant bins with ultrasonic sensors for fill-level monitoring, temperature sensors for fire detection, and communication modules (GSM, LoRaWAN, NB-IoT) for real-time data transmission.
2. **Centralized Cloud Platform:** Develop a secure cloud-based system for data storage, processing, visualization, and historical trend analysis. Ensure encryption, authentication, and scalability.
3. **ML-based Predictive Analytics:** Train models (Random Forests, Gradient Boosting, Neural Networks) to forecast fill levels and optimize collection routes dynamically.
4. **User Interfaces:** Provide a web dashboard for administrators to monitor system performance, generate reports, and receive alerts, alongside a mobile interface for collection crews to follow optimized routes and report issues.
5. **Feasibility and Impact Evaluation:** Pilot the system to assess operational efficiency, cost savings, environmental benefits, and user satisfaction. Analyze scalability to other urban areas and provide recommendations for municipal adoption.

1.4 Proposed Solutions

**1.4.1 Smart Waste Bins:**  
Bins will include ultrasonic sensors for accurate fill-level detection and temperature sensors for fire hazard monitoring. Microcontrollers will process sensor data, while communication modules send data to the cloud. Optional gas sensors can detect odors or harmful emissions.

**1.4.2 Cloud-Based Platform:**  
A central cloud server will aggregate and process data, store historical records, and support ML analytics. Dashboards will visualize bin status, generate alerts, and provide insights on operational efficiency, environmental impact, and waste trends.

**1.4.3 Predictive Analytics and Route Optimization:**  
ML models will predict bin fill levels considering temporal patterns, events, and weather. Dynamic routing algorithms (Genetic Algorithms, Ant Colony Optimization, Linear Programming) will generate efficient collection routes, minimizing fuel use and labor costs.

**1.4.4 User Interfaces:**

* **Community Dashboard:** Web Interface provides optimized routes, allows status updates, and incident reporting, with offline capabilities.Web dashboard offers bin maps, KPIs, alerts, reports, and integration with municipal information systems.

**1.4.5 Environmental and Economic Benefits:**  
Dynamic scheduling reduces fuel consumption, emissions, and traffic congestion. Preventing overflows maintains cleaner urban environments and minimizes pest-related health risks. Operational costs decrease due to optimized resource allocation.

**1.4.6 Adaptability and Scalability:**  
The system can scale across different urban densities and infrastructure types. Modular software design allows future expansion, including citizen engagement tools or waste segregation monitoring.

**1.5 Conclusion**

The proposed SWBMS bridges current technological gaps in Sri Lanka’s waste management by integrating smart bins, cloud computing, ML analytics, and intuitive interfaces. It supports operational efficiency, environmental sustainability, and public health, aligning with national digital transformation goals and UN Sustainable Development Goal 11. Pilot implementation will demonstrate feasibility, cost-effectiveness, and scalability for wider municipal adoption.

2. Literature Review

**2.1 Overview of Smart Waste Management Systems**

Urbanization has led to increased waste generation, making traditional collection methods inefficient, often resulting in overflowing bins, unnecessary collection trips, and higher operational costs (Gupta et al., 2020; Jayaraman et al., 2017). Smart waste management systems leverage IoT technologies, integrating sensor networks, communication infrastructure, and analytics to enable real-time monitoring, dynamic scheduling, and optimized collection routes (Mohamed, 2020; Poonia, 2022). Accurate fill-level detection is essential for predictive analytics, improving efficiency, reducing costs, and minimizing environmental impact (Rahman et al., 2022).

**2.2 Sensor Technologies for Waste Bin Fill-Level Detection**

Fill-level detection is central to smart waste systems. Commonly used sensors include:

* **Ultrasonic Sensors:** Emit sound waves to measure distance to the waste surface. Advantages include low cost, ease of installation, and non-contact operation (Smartbin, 2015; Aazam et al., 2014). Limitations arise from environmental factors (rain, dust, temperature), irregular waste shapes, and compaction, which can reduce accuracy (Gupta et al., 2020).
* **Weight Sensors (Load Cells):** Measure the mass of waste directly via force exerted on the sensor, typically mounted at the bin base (Jayaraman et al., 2017). They are less influenced by waste shape or environmental conditions and provide more reliable data, supporting accurate predictive analytics (Andersen, 2021; Tan, 2022).
* **Other Sensors:** Infrared sensors detect reflected light beams, while capacitive sensors detect changes in capacitance caused by waste proximity (Longhi, 2012). Both are limited by material types and environmental interference, restricting their general applicability.

2.3 Weight Sensors: Advantages, Challenges, and Integration

**Advantages:**

1. **High Measurement Accuracy:** Weight sensors avoid errors caused by irregular shapes, compaction, or layering, providing precise data for predictive route planning (Andersen, 2021).
2. **Environmental Robustness:** Measurements are largely unaffected by dust, moisture, or temperature changes, ensuring reliable outdoor performance (Rahman et al., 2022).
3. **Early Overflow Detection:** When combined with volume-based data or calibrated thresholds, weight sensors help detect early overflow or hazardous conditions, including potential fires or toxic gases (Tan et al., 2019).
4. **Reduced False Positives/Negatives:** Unlike ultrasonic sensors, weight readings are not affected by foam, hanging debris, or waste bags (Mohamed, 2020).

**Challenges:**

1. **Installation Complexity:** Requires stable mounting and vibration dampening; uneven surfaces or traffic vibrations may introduce noise (Gupta et al., 2020).
2. **Maintenance Requirements:** Regular calibration is needed due to mechanical wear or drift, especially in dusty or corrosive environments (Rahman et al., 2022).
3. **Higher Initial Cost:** Weight sensor systems are more expensive upfront than ultrasonic systems, which may limit large-scale deployment (Karunasena, 2021).
4. **Power Consumption:** Signal conditioning circuits may increase energy usage, impacting battery-powered or solar-powered bins (Jayaraman et al., 2017).

**Integration with IoT and Data Analytics:**  
Weight sensors transmit real-time mass data to cloud platforms, enabling GPS-based route optimization and predictive collection schedules (Poonia, 2022). Historical weight data supports machine learning models to forecast waste generation by location and time, facilitating proactive resource allocation and sustainability reporting (Rahman et al., 2022).

2.4 Case Studies & Real-World Implementations

* **Singapore (NEA Smart Bin Pilot, 2018–2021):** Integrated weight, temperature, and humidity sensors for dynamic collection scheduling. Achievements included 20% fewer collection trips, 15% lower fuel usage, and improved hazard detection. Challenges involved high-density urban installation, battery limitations, and network dead zones (Tan et al., 2019).
* **Copenhagen (Smart City Initiative, 2020–2023):** 1,500 bins with weight sensors enabled real-time monitoring and AI-driven route optimization, improving route efficiency by 30% and reducing overflow complaints by 40%. Limitations included high initial costs and maintenance logistics (Andersen, 2021; Smart City Annual Report, 2023).
* **Seoul:** Integrated weight sensors with RFID-tagged bins and a Pay-As-You-Throw model. Benefits included precise billing, waste reduction, and enhanced citizen engagement. High infrastructure costs and privacy concerns limited adoption to major urban centers (Kim, 2020).

**Comparison with Ultrasonic Sensors:**

|  |  |  |
| --- | --- | --- |
| Feature | Weight Sensors | Ultrasonic Sensors |
| Measurement | Direct mass | Distance to surface |
| Accuracy | High, less noise-prone | Moderate, affected by shape |
| Environmental Robustness | Strong | Sensitive to obstructions |
| Installation | Complex | Moderate |
| Maintenance | Moderate to high | Low to moderate |
| Cost | Higher upfront | Lower upfront |
| Power | Higher | Lower |

Table 1 Sensor Comparison

Weight sensors provide more actionable data for advanced analytics and billing, though they require higher investment and maintenance (Mohamed, 2020).

2.5 Status in Sri Lanka  
Sri Lanka lacks weight sensor-based smart waste management systems. Municipalities rely on manual collection, facing persistent overflows and inefficiencies. No patents or commercial solutions exist locally (Ministry of Environment Sri Lanka, 2024; Karunasena, 2021; Feasibility Study on IoT Waste Monitoring in Sri Lanka, 2022; National Waste Generation Statistics, 2023).

**Summary:**  
Global deployments demonstrate clear operational, environmental, and efficiency benefits of weight sensor systems. The absence of such technologies in Sri Lanka represents an opportunity to implement advanced, sustainable, and data-driven waste management aligned with smart city goals.

3. Planning

Effective planning is essential for the successful implementation of the Smart Waste Bin Management System (SWBMS). This section outlines the feasibility, risks, strategic analysis, environmental factors, system development life cycle, and time plan necessary to ensure the project is delivered efficiently and sustainably.

3.1 Feasibility Report

The feasibility of SWBMS was assessed across four dimensions:

1. **Technical Feasibility:**
   * IoT sensors, microcontrollers, and LPWAN communication technologies (LoRaWAN, GSM) are commercially available and suitable for Sri Lanka’s urban environment.
   * Cloud platforms (AWS, Azure, GCP) provide scalable storage and analytics capabilities for real-time monitoring and predictive modeling.
   * ML models for fill-level prediction and route optimization have proven performance in similar global implementations.
2. **Economic Feasibility:**
   * Initial investment includes smart bins, sensor modules, communication hardware, cloud services, and software development.
   * Cost-benefit analysis indicates that reduced fuel consumption, optimized labor, and fewer collection trips will result in long-term savings of up to 30%.
3. **Operational Feasibility:**
   * Municipal staff will be trained to operate dashboards and mobile interfaces.
   * Field workers can adapt to optimized routing with minimal disruption to existing collection schedules.
4. **Legal and Environmental Feasibility:**
   * Compliance with Sri Lankan municipal regulations and data privacy laws is achievable.
   * Reduction in waste overflow and vehicle emissions aligns with environmental sustainability goals.

**Conclusion:** The project is feasible and aligns with both operational and strategic objectives.

3.2 Risk Assessment

The risk assessment identifies potential challenges and mitigation strategies:

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Impact | Likelihood | Mitigation |
| Sensor malfunction or inaccuracies | High | Medium | Regular calibration, redundant sensors, and maintenance schedules |
| Network connectivity issues | Medium | Medium | Use LPWAN technologies and offline data caching in bins |
| Budget overruns | High | Low | Detailed cost planning, phased deployment, and contingency funds |
| Low user adoption | Medium | Medium | Training programs, workshops, and incentive-based engagement |
| Data security breaches | High | Low | Implement encryption, secure authentication, and access control |
| Environmental factors (rain, heat, humidity) | Medium | Medium | Weather-resistant bin design and robust sensor enclosures |

Table 2 Risk Assesment

3.3 SWOT Analysis

| **Strengths** | **Weaknesses** |
| --- | --- |
| Real-time monitoring of waste bins | Initial high investment costs |
| Predictive analytics for optimized collection | Dependence on technical expertise |
| Reduction of operational costs and emissions | Reliance on network connectivity |
| Scalable and adaptable system | Need for public awareness and cooperation |
| **Opportunities** | **Threats** | |
| Government support for smart city initiatives | Resistance to change from municipal staff | |
| Alignment with SDGs and digital transformation | Sensor vandalism or theft | |
| Potential for citizen engagement apps | Rapid technological obsolescence | |

Table 3 SWOT Analysis

3.4 PESTEL Analysis

|  |  |  |
| --- | --- | --- |
|  | Factor | Description |
| P | Political | Government initiatives in smart cities and digital transformation support adoption of SWBMS. |
| E | Economic | Potential long-term savings in fuel, labor, and operational efficiency. |
| S | Social | Public acceptance and behavior are critical; awareness campaigns can enhance cooperation. |
| T | Technological | IoT, ML, and cloud computing provide the technological backbone. |
| E | Environmental | Reduces waste overflow, improves sanitation, and lowers carbon emissions. |
| L | Legal | Compliance with local data privacy laws and municipal regulations is mandatory. |

Table 4 PESTEL Analysis

3.5 Life Cycle Model

The **Agile Incremental Model** is adopted for SWBMS development due to its flexibility, iterative progress, and ability to incorporate stakeholder feedback.

**Phases:**

1. **Requirement Analysis:** Define functional and non-functional requirements, including sensor specifications, data handling, and user interfaces.
2. **System Design:** Hardware and software architecture design, including cloud infrastructure and dashboard layouts.
3. **Implementation:** Development of smart bin firmware, mobile/web applications, cloud integration, and ML algorithms.
4. **Testing:** Unit, integration, system, and user acceptance testing in pilot areas.
5. **Deployment:** Rollout in selected urban zones, monitoring, and feedback collection.
6. **Maintenance:** Continuous monitoring, software updates, hardware servicing, and scaling to additional areas.

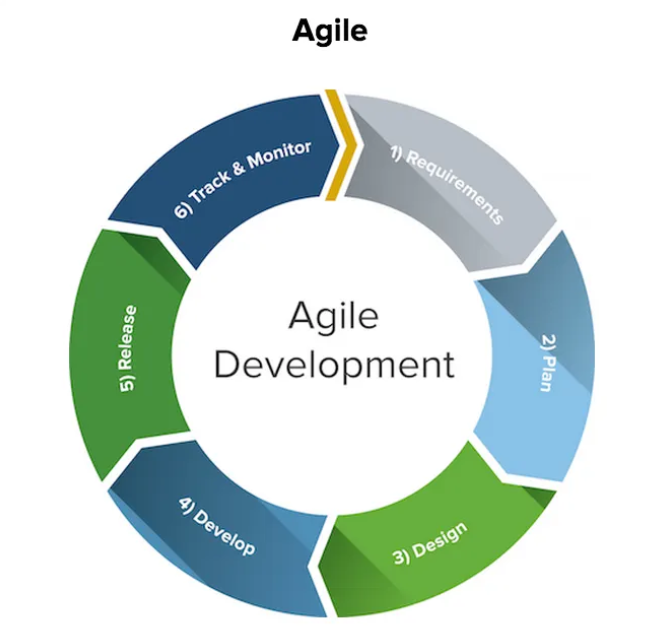


Figure 1 Agile Model

**3.6 Time Plan**

A Gantt-chart style time plan is proposed for project completion over **2 months**:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | Phase | Duration (Weeks) | Activities | | Requirement Analysis | 1 | Stakeholder meetings, feasibility study, sensor and technology selection | | System Design | 2 | Architecture design, dashboard UI/UX, ML model selection | | Implementation | 3–4 | Firmware coding, app development, cloud integration, ML model training | | Testing | 5 | Unit testing, integration testing, field pilot testing | | Deployment | 6 | Pilot area installation, feedback collection, optimization | | Evaluation & Maintenance | 7–8 | Performance evaluation, adjustments, scalability planning | |  |  |

Table 5 Time Plan

This structured plan ensures timely delivery, resource optimization, and continuous evaluation to address technical, operational, and social factors during project execution.

4. Requirement Gathering and Analysis

Effective requirement gathering is crucial to ensure the Smart Waste Bin Management System (SWBMS) meets **stakeholder expectations, operational feasibility, and urban waste management challenges** in Sri Lanka. This phase combined **quantitative and qualitative techniques**, including questionnaires, interviews, field observations, and document analysis. The primary goals were to identify:

* System objectives and operational scope
* Functional and non-functional requirements
* Stakeholder constraints and expectations
* Real-world challenges in municipal waste management

4.1 Techniques Employed

**1. Questionnaires**

* Distributed to 150 respondents across urban, suburban, and rural areas (Colombo, Kandy, Galle, and surrounding regions).
* Targeted stakeholders: municipal officers, sanitation workers, self-employed citizens, students, and residents.
* Focus areas: current waste management practices, awareness of smart waste management systems, challenges, perceived importance, willingness to adopt, and preferred features.

**2. Interviews**

* Semi-structured interviews with **38 key stakeholders**, including municipal officers, field supervisors, public health experts, and community leaders.
* Explored operational inefficiencies, technical and environmental challenges, system expectations, and feasibility for SWBMS.

**3. Field Observations**

* Site visits to observe **bin locations, collection routines, overflow incidents, and operational workflows**.
* Informal discussions with sanitation workers to validate reported challenges.

**4. Document Analysis**

* Reviewed **municipal reports, National Waste Generation Statistics (2023), policy frameworks**, and prior technology adoption attempts.
* Ensured alignment with regulatory standards and national sustainable development goals.

4.2 Key Findings from User Responses

4.2.1 Waste Disposal Methods and Collection Frequency

|  |  |  |
| --- | --- | --- |
| Disposal Method | Respondents (%) | Key Notes |
| Collected by municipality | 70 | Most common, mainly urban and suburban |
| Burned | 15 | Observed in suburban areas |
| Dumped at community bins | 10 | Predominantly rural regions |
| Other | 5 | Includes private collection or ad hoc |

Table 6 waste disposal and collection

**Collection Frequency:**

* Daily: 10%
* Every 2–3 days: 30%
* Weekly: 50%
* Less than weekly: 10%

**Observation:** Fixed weekly schedules dominate urban areas; rural and suburban zones show irregular or infrequent collection.

Figure 2 Waste Disposal Methods and Collection Frequency

4.2.2 Overflowing Bins

|  |  |
| --- | --- |
| Overflow Frequency | Respondents (%) |
| Yes, frequently | 35 |
| Sometimes | 40 |
| Rarely | 15 |
| Never | 10 |

Table 7 Overflow Frequency

* Overflow is a significant issue, especially during **weekends, festivals, and high-density urban zones**.
* Contributing factors: irregular collection, high waste generation, and insufficient bin coverage.

Figure 3 Overflow Frequency

4.2.3 Main Waste Management Challenges

|  |  |
| --- | --- |
| Challenge | Respondents (%) |
| Irregular collection | 45 |
| Overflowing bins | 35 |
| Bad odor | 40 |
| Animals spreading garbage | 25 |
| Lack of recycling options | 30 |

Table 8 Main Waste Management Challenges

**Observation:** Municipal collection schedules do not adapt to waste generation patterns, causing resource inefficiency and environmental concerns.

Figure 4 Main Waste Management Challenges

4.2.4 Awareness and Support for Smart Waste Bin Systems

|  |  |  |  |
| --- | --- | --- | --- |
| Question | Yes (%) | No (%) | Maybe (%) |
| Awareness of smart waste bins | 45 | 40 | 15 |
| Support implementation in their area | 70 | 10 | 20 |
| Willing to pay a small monthly fee | 55 | 20 | 25 |

Table 9 Awareness and Support for Smart Waste Bin Systems

* Awareness is low, but willingness to adopt technology is high, demonstrating stakeholder **openness if benefits are clear**.

Figure 5 Awareness and Support for Smart Waste Bin Systems

**4.2.5 Preferred Features**

|  |  |
| --- | --- |
| Feature | Respondents (%) |
| Real-time bin status monitoring | 88 |
| Automatic alerts when bins are full | 76 |
| GPS tracking for bins | 65 |
| Odour control system | 60 |
| Waste level data reports | 72 |

Table 10 Preferred Features

* Priorities highlight **real-time monitoring and proactive alerts** as the most critical functional features.

Figure 6 Preferred Features

4.3 Insights from Interviews

1. **Operational Inefficiencies:**
   * Data gaps lead to wasted trips, missed pickups, and uneven workloads.
   * Manual monitoring causes delays in decision-making and reactive interventions.
2. **Technical Barriers:**
   * Limited internet connectivity and network reliability in certain areas.
   * Sensor accuracy and battery life are major concerns.
   * Cost of implementation and maintenance highlighted by municipal officials.
3. **Health and Environmental Risks:**
   * Overflowing bins create conditions for **vector-borne diseases** like dengue and leptospirosis.
   * Odour and waste exposure affect public hygiene and community satisfaction.
4. **Community Engagement Needs:**
   * Public awareness and participation are essential.
   * Residents need education on proper disposal to maximize system effectiveness.
5. **System Design Recommendations:**
   * Simple mobile apps for field workers; dashboards for municipal administrators.
   * Hybrid connectivity: LPWAN + GSM + Wi-Fi.
   * Pilot deployment before full-scale rollout.

4.3.1 Integration of Questionnaire and Interview Data

**Consensus Themes:**

* Need for real-time monitoring, predictive analytics, and data-driven collection optimization.
* Strong demand for odor control, alerts, and GPS tracking.
* Willingness to pay and support is moderate to high if benefits are tangible.

**Contrasting Views:**

* Field workers emphasize operational simplicity.
* Municipal officers prioritize budget and technical feasibility.
* Residents focus on hygiene and consistent service.

**Actionable Recommendations:**

1. Deploy pilot projects in high-density wards.
2. Modular, scalable system adaptable to urban and suburban contexts.
3. User-friendly mobile interfaces and dashboards.
4. Incorporate public awareness campaigns alongside deployment.
5. Energy-efficient sensors with battery backup to handle outages.
6. Maintenance and technical support frameworks for long-term sustainability.

4.3.2 Alignment with National Policies

* SWBMS aligns with **Sri Lanka’s National Sustainable Development Strategy** and **smart city initiatives**.
* Supports environmental protection, public health improvement, and urban service efficiency.
* Requires government funding and policy support for successful implementation.

4.3.3 Conclusion

The requirement gathering process demonstrates:

* A strong need and readiness for **smart waste management solutions**.
* High priority for **real-time monitoring, predictive analytics, alerts, and data-driven decision-making**.
* Key implementation considerations: pilot testing, technical robustness, workforce training, and public engagement.

4.4 Functional and Non-Functional Requirements

The design and implementation of the Smart Waste Bin Management System (SWBMS) demand a clear articulation of system requirements. These requirements guide the development process to ensure the system meets the needs of its users while maintaining operational effectiveness and reliability. This section delineates the functional and non-functional requirements identified based on the earlier stakeholder analysis, literature review, and technical feasibility studies.

4.4.1 Functional Requirements

Functional requirements describe the specific behaviors and operations that the SWBMS must perform. They focus on what the system should do in response to user actions or internal events. Key functional requirements for the project are:

1. **Real-Time Fill Level Monitoring**  
   Each smart waste bin must continuously monitor and report its fill level using weight sensors. Data should be updated at regular intervals (e.g., every 5 minutes) to provide near real-time information to the central server. This enables timely decision-making.
2. **Data Transmission and Communication**  
   The bins must securely transmit sensor data wirelessly to the cloud-based platform using appropriate communication protocols (e.g., GSM, Wi-Fi, or LPWAN). The system must handle intermittent connectivity gracefully and ensure no data loss.
3. **Cloud Data Storage and Management**  
   The central system should collect, store, and manage the data received from all deployed bins. It must support scalable data storage solutions to handle increasing data volumes as the system expands.
4. **Predictive Analytics and Forecasting**  
   The system should incorporate machine learning models to analyze historical and real-time data, predicting future fill levels for each bin. This forecasting is crucial for optimizing collection schedules and preventing overflows.
5. **Dynamic Route Optimization**  
   Based on bin fill predictions, the system should generate optimized collection routes that minimize travel distance, fuel consumption, and time. It should dynamically adjust routes in response to real-time changes such as unexpected bin usage or traffic conditions.
6. **User Dashboard and Visualization**  
   A web-based dashboard must provide municipal officials and waste management teams with intuitive visualization tools. This includes maps displaying bin statuses, alerts for full or malfunctioning bins, and reporting tools for performance monitoring.

4.4.2 Non-Functional Requirements

Non-functional requirements specify the system’s quality attributes and operational constraints. They define how the system performs rather than what it does. Key non-functional requirements include:

1. **Scalability**  
   The system architecture must support scaling to hundreds or thousands of smart bins across multiple urban areas without degradation in performance. This includes cloud infrastructure scalability and communication network adaptability.
2. **Reliability and Availability**  
   The system should achieve high uptime (target > 99.5%) to ensure continuous monitoring and timely data delivery. Fault-tolerance mechanisms must handle hardware or network failures gracefully.
3. **Performance**  
   Data transmission latency from bin to dashboard should be minimal, targeting near real-time updates (ideally under 10 seconds). Machine learning predictions should be computed efficiently to support daily route planning.
4. **Security and Privacy**  
   The system must employ secure communication protocols (e.g., TLS/SSL) to protect data in transit. User authentication and authorization mechanisms are essential to prevent unauthorized access. Sensitive location and operational data should be protected under privacy policies compliant with national regulations.
5. **Usability**  
   User interfaces must be intuitive, responsive, and accessible to users with varying technical expertise. Clear navigation, consistent design, and comprehensive help documentation should be provided.
6. **Maintainability and Supportability**  
   The system design should facilitate easy updates, bug fixes, and feature enhancements. Modular coding practices and well-documented APIs will support maintainability. A helpdesk or support system should assist users with issues.
7. **Energy Efficiency**  
   Smart bins’ embedded hardware should be energy-efficient to maximize battery life, especially in locations without reliable power supply. Low-power communication protocols and sleep modes should be used.
8. **Environmental Robustness**  
   Sensors and electronics installed in bins must be durable and resistant to weather conditions such as rain, heat, dust, and physical impact to ensure long-term functionality.
9. **Compliance**  
   The system must comply with relevant Sri Lankan governmental regulations and international standards related to environmental monitoring, data security, and wireless communication.
10. **Cost-Effectiveness**  
    The total cost of ownership including hardware, deployment, operation, and maintenance should be reasonable to enable adoption by municipal authorities with constrained budgets.

**Summary**

Defining clear functional and non-functional requirements is critical to developing a robust and effective Smart Waste Bin Management System. The functional requirements ensure that the system delivers the intended features such as real-time monitoring, predictive analytics, route optimization, and user interaction. Meanwhile, the non-functional requirements guarantee that the system is scalable, reliable, secure, and usable in the diverse and challenging urban environment of Sri Lanka. Together, these requirements form a blueprint to guide system architecture, technology selection, development processes, and deployment strategies, ensuring alignment with stakeholder needs and sustainable urban development goals.

5. System Design

The system design phase of the Smart Waste Bin Management System (SWBMS) translates the requirements and functional specifications into a concrete blueprint that defines how the system will work in practice. The main goal of the design is to establish a scalable, efficient, and user-friendly system that not only optimizes waste collection in urban areas but also contributes to environmental sustainability and quality of life.

System design provides both **high-level architecture** (how all the modules and technologies fit together) and **low-level design** (how the system handles operations like data flow, storage, and user interactions). In this section, we explain the architectural blueprint, database design, process modeling, and user interface planning for SWBMS.

5.1 Architecture Diagram

The architecture of the SWBMS follows a **three-tier model** comprising the **IoT Layer**, **Middleware (Server + Database)**, and the **Application Layer**.

* **IoT Layer (Smart Waste Bins)**
  + Each waste bin is equipped with **weight sensors** for capacity validation.
  + A **microcontroller (ESP32)** collects sensor data and transmits it through **Wi-fi** to the **MQTT** cloud.

A diagram of a diagram

AI-generated content may be incorrect.

Figure 7 IOT Flow

* **Middleware / Cloud Layer**
  + This layer manages **real-time data storage and processing**.
  + Technology **Node.js/Express server with PostgreSQL** are used to collect data.
  + A **prediction module (machine learning)** forecasts waste trends and optimizes collection routes.
  + A **notification engine** sends alerts when bins reach critical levels.

**A diagram of a diagram

AI-generated content may be incorrect.**

Figure 8 Cloud Layer

A diagram of a diagram

AI-generated content may be incorrect.

Figure 9 Notifications

* **Application Layer**
  + Includes a **Web Dashboard** for municipal workers and waste collectors.
  + Features: Real-time monitoring, optimized route navigation, historical data analytics, and predictive insights.

A diagram of a software system

AI-generated content may be incorrect.

Figure 10 System Architecture

5.2 ER Diagram (Entity-Relationship Diagram)

The ER diagram for the **Smart Waste Bin Management System (SWBMS)** models four core entities that capture bin locations, incoming telemetry, status snapshots, and machine-learning outputs.

**Entities**

* **bins** (id PK, location\_name, latitude, longitude, created\_at)  
  Master table for each physical smart bin, storing its human-readable location name and precise coordinates.
* **readings** (id PK, bin\_id FK→bins.id, weight\_kg, fullness\_percent, recorded\_at)  
  Time-series telemetry collected from each bin via MQTT (e.g., ESP32 + load cell). Each row is a measurement event with the measured weight and computed fullness percentage at a timestamp.
* **bin\_status** (id PK, bin\_id FK→bins.id, fill\_level CHECK 0–100, created\_at)  
  Lightweight status snapshots of a bin’s fill level (as an integer percent). Useful for quick UI/status queries without scanning all detailed readings.
* **predictions** (id PK (identity), bin\_id FK→bins.id, prediction\_month (date), predicted\_weight\_kg, created\_at)  
  Model outputs generated by the FastAPI/LightGBM pipeline. Each row stores a per-bin forecast (e.g., weekly roll-up) with the prediction horizon encoded by prediction\_month.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 11 ER Diagram Postgre

A diagram of a computer

AI-generated content may be incorrect.

Figure 12 ER Diagram

**Relationships**

* **One bin has many readings** (1-to-N): readings.bin\_id → bins.id.
* **One bin has many bin\_status snapshots** (1-to-N): bin\_status.bin\_id → bins.id.
* **One bin has many predictions** (1-to-N): predictions.bin\_id → bins.id.

**Notes**

* The fill\_level in bin\_status is constrained to **0–100** to guarantee valid percentages.
* created\_at/recorded\_at timestamps on all tables provide an auditable history for real-time views, trend analytics, and ML retraining.
* User accounts, routes, and collection records are **not part of this schema**; routing is handled externally (OpenRouteService) and user/ops data can be added later if needed.

**Justification**

This lean model keeps **location data** (bins), **high-granularity sensor streams** (readings), **fast status lookups** (bin\_status), and **AI forecasts** (predictions) separate but connected. It supports real-time dashboards, historical analysis, and weekly model training without overcomplicating write paths or query patterns.

5.3 UML Diagrams

**5.3.1 Use Case Diagram**

It shows the interaction between actors and the system.

* **Actors**: Admin, Waste Collector, IoT Device (Smart Bin).
* **Admin Use Cases**: Monitor bins, Manage Bins, View reports.
* **Collector Use Cases**: Receive alerts, Navigate route.
* **Smart Bin Use Cases**: Send sensor data, Update location.
* **AI Prediction Engine**: Predict Future Waste, Analyze Trends.

A diagram of a smart waste bin management system

AI-generated content may be incorrect.

Figure 13 Use Case Diagram

**5.3.2 Sequence Diagram**

Demonstrates how a full waste collection cycle works.

**Smart Bin → Server**

* The IoT smart bin sends weight and fill-level sensor data to the central server.

**Server → Web Dashboard**

* The server forwards this data to the web dashboard for processing and monitoring.

**Web Dashboard → Database**

* The dashboard stores the incoming fill-level data in the central database.

**Decision: Bin Status**

* If the bin is **nearly full**, the system triggers a notification to staff and community.
* If the bin is **not full**, the dashboard simply updates the current bin status view.

**ML Predictor → Database**

* The ML predictor retrieves historical data from the database.

**ML Predictor → Web Dashboard**

* It analyzes trends and predicts future fill-levels, sending results to the web dashboard.

**Web Dashboard → Staff & Community**

* Prediction results (e.g., estimated time to bin overflow) are displayed on the dashboard for staff and community.

**Staff & Community → Web Dashboard**

* Staff request an optimized waste collection route through the web interface.

**Web Dashboard → Database**

* The dashboard fetches bin status and location data from the database.

**Web Dashboard → Staff & Community**

* The system generates and provides an optimized route plan for collection.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 14 Sequence Diagram

**5.3.3 Activity Diagram**

Depicts workflow:

* Bin fills → Sensor triggers update → Server checks threshold → Web Dashboard notified → Waste collected → Bin Reset

A diagram of a software system

AI-generated content may be incorrect.

Figure 15 Activity Diagram

These UML diagrams clearly explain both **data flow** and **user interactions**.

5.4 UI Diagrams

User interfaces give a **visual preview** of the system.

* **Admin Dashboard UI**
  + Map view of bins (green = empty, yellow = half, red = full).
  + Menu with route optimization tools and waste statistics.
  + Graphs show daily/weekly/monthly waste trends.

A screenshot of a map

AI-generated content may be incorrect.

Figure 16 Dashboard Top

A screenshot of a computer

AI-generated content may be incorrect.

Figure 17 Dashboard Summary and Statistics

A screenshot of a computer

AI-generated content may be incorrect.

Figure 18 Bins Summary

* **Map Bin Details, AI Predictor and Route Planner**

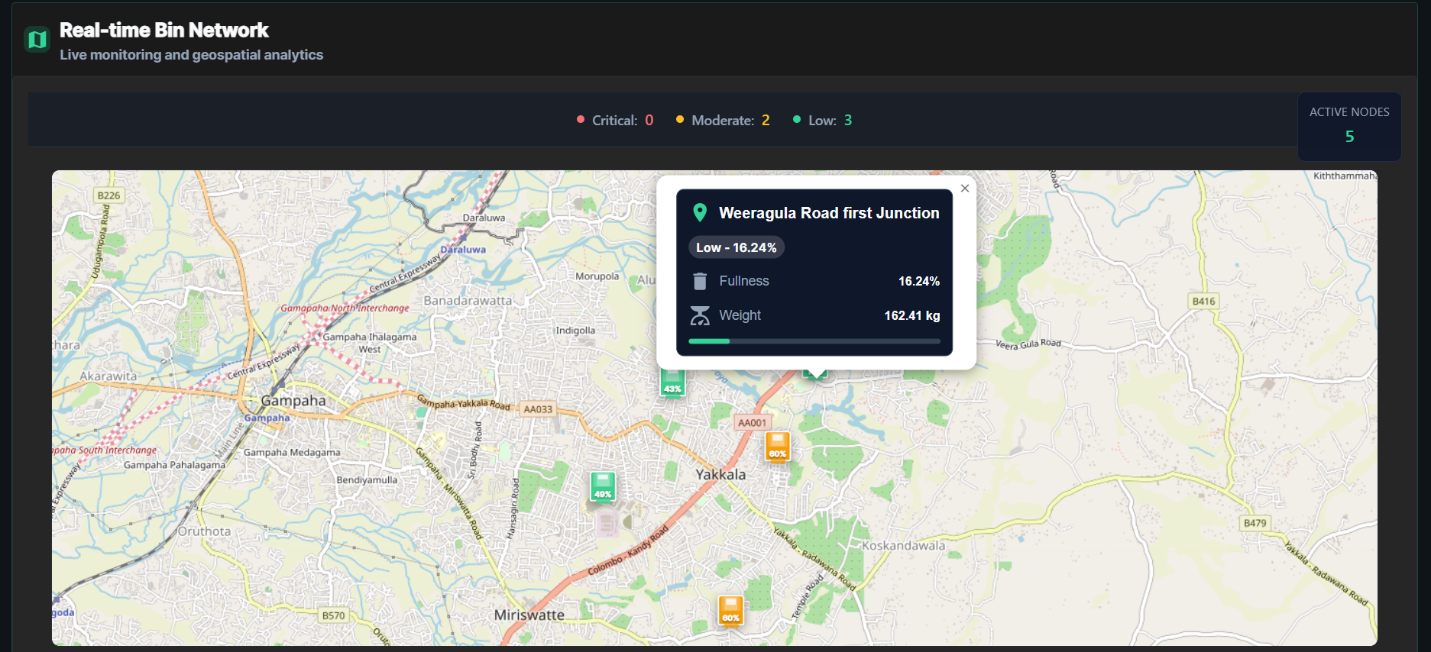


Figure 19 Bin Details View

A screenshot of a computer

AI-generated content may be incorrect.

Figure 20 AI Predictor

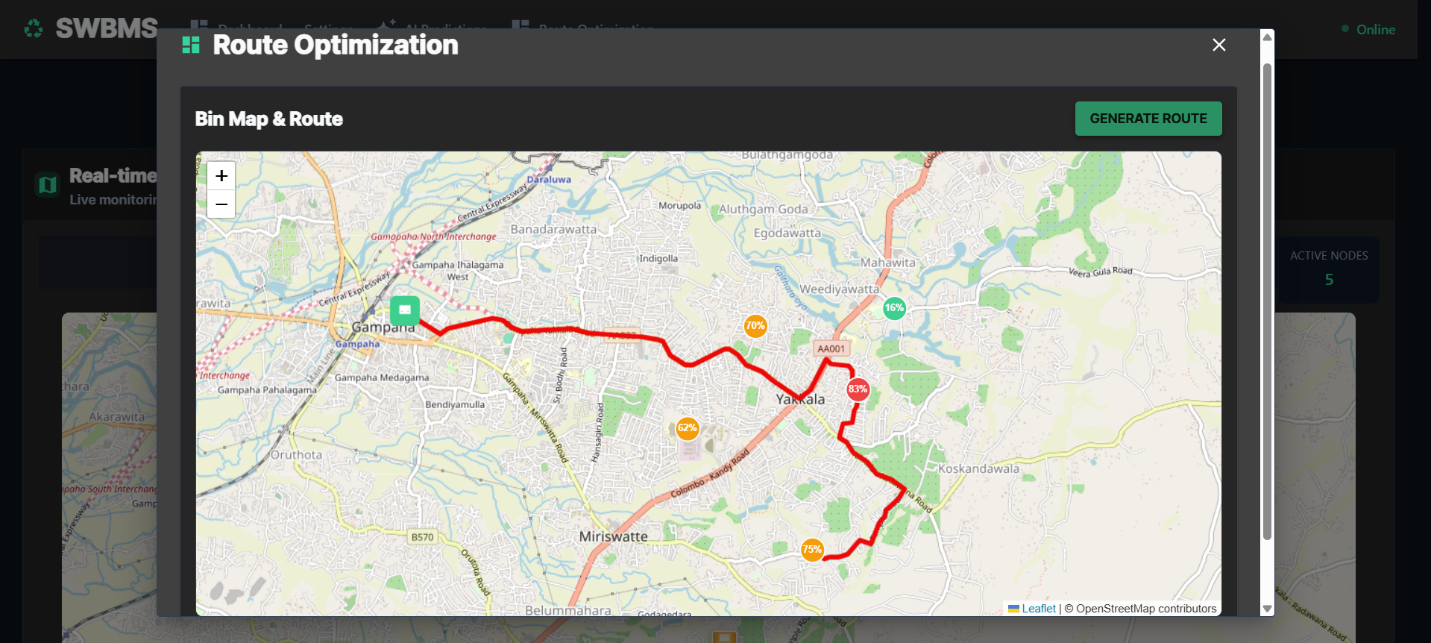


Figure 21 Route Planner

With this **System Design**, SWBMS becomes more than a technical project – it is a **scalable smart city solution** designed to minimize waste overflow, optimize collection efficiency, reduce fuel consumption, and build cleaner cities.

6. Implementation

The implementation of the **Smart Waste Bin Management System (SWBMS)** brings together **hardware, software, and data intelligence** into a single solution that addresses one of the most persistent challenges in urban living: effective waste management. The implementation stage translates that vision into reality by building each component step by step and connecting them into a working ecosystem.

At its core, the system combines **IoT-enabled bins**, **real-time communication protocols**, **cloud-based storage**, and **predictive machine learning models**. Each layer has been carefully designed to make waste management more efficient, more transparent, and ultimately, more sustainable contributing to **cleaner cities, healthier environments, and reduced operational costs** for municipal councils.

6.1 Technology Stack

The technology stack defines the building blocks of the system. It was chosen with three principles in mind: **scalability** (the ability to add more bins or expand to more cities), **reliability** (continuous operation with minimal downtime), and **simplicity** (low maintenance and ease of understanding).

**IoT Hardware Layer**

* **ESP32 Microcontroller**:  
  The “brain” of each smart bin. The ESP32 is powerful enough to process sensor data, manage Wi-Fi connectivity, and handle MQTT messaging, while being energy efficient.
* **Load Cell with HX711 Module**:  
  Used to measure the **weight of the waste inside the bin**. Unlike ultrasonic sensors that only detect volume, the load cell provides a more reliable indicator of fullness by accounting for different waste densities (e.g., paper vs. food waste).
* **LED Indicators**:
  + **Single Pixel LED**: Represents **Wi-Fi connectivity**.
    - Green Blink: Connected to the network.
    - Blue Blink: Attempting to reconnect.
  + **Five-Segment LED Bar**: Provides a **visual representation of fullness** directly on the bin.
    - 1–2 LEDs = 20–40% (safe).
    - 3 LEDs = ~60% (moderate).
    - 4 LEDs = ~80% (attention needed).
    - 5 LEDs solid red = 95% full.
    - 5 LEDs blinking red = **critical** (bin overflowing, immediate collection required).

This design ensures that **anyone walking by a bin—residents, municipal staff, or even tourists—can instantly see its status** without needing to check a dashboard.

**Communication Layer**

* **MQTT Protocol**:  
  Lightweight and fast, MQTT was chosen to ensure **real-time communication** between bins and the central system. Each bin **publishes messages** to a topic (smartbin/data), and the dashboard **subscribes** to those topics to get instant updates.
* **Wi-Fi Connectivity**:  
  The ESP32 connects to municipal Wi-Fi or local 4G routers. In rural areas, LoRa could be adopted for extended range.

**Backend & Data Layer**

* **PostgreSQL Database**:  
  Acts as the long-term memory of the system. Every bin reading is stored with its bin\_id, weight in kg, fullness percentage, and timestamp.
* **FastAPI Service**:  
  A high-performance backend built with Python. It has two major roles:
  1. **Data ingestion & APIs**: Receives data from the frontend and inserts into PostgreSQL.
  2. **Machine learning automation**: Periodically trains models and generates predictions.
* **Machine Learning Module (LightGBM)**:  
  Every Sunday at midnight, the scheduler retrains a per-bin model using **historical waste patterns**. Predictions are stored in the predictions table for use during the week.

**Frontend Layer**

* **React.js Dashboard**:  
  The face of the system for municipal authorities. It includes:
  + A **map view** of all bins, with colors representing fullness levels.
  + **Graphs and analytics dashboards** showing waste generation trends.
  + A **management panel** for viewing predictions and scheduling collections.
  + A **public-facing view** (optional) for residents, showing cleanliness scores for their neighborhood.

**Routing Layer**

* **Open Route Service + OpenStreetMap (OSM)**:  
  The system does not just show data; it **acts on it**. Once bins reach >80% capacity, their locations are sent to the Open Route Service API along with the municipal council depot location. The system then computes the **shortest and most fuel-efficient route** that passes only through bins that need collection.
  + This avoids unnecessary travel to empty bins.
  + Reduces **fuel costs, vehicle emissions, and collection time**.

6.2 Design Patterns

Several software design patterns were incorporated:

1. **MVC (Model-View-Controller)** – Used in the React frontend for clean separation of concerns.
2. **Observer Pattern** – Bins “notify” the system via MQTT when their state changes.
3. **Singleton Pattern** – Ensures one persistent PostgreSQL connection per API instance.
4. **Factory Pattern** – Automates per-bin machine learning model creation.

These patterns ensure that the system is not just functional, but also **easy to maintain, upgrade, and scale**.

6.3 Detailed Implementation Process

**6.3.1 IoT Workflow**

1. Waste is deposited into the bin.
2. The **load cell** (HX711)measures the weight in real-time.
3. ESP32 converts this weight into a **fullness percentage**.
4. LED indicators update immediately for local users showing fill level by color and number of LEDs lit.
5. Data packet is published via MQTT:

A screen shot of a computer

AI-generated content may be incorrect.

Figure 22 ESP Code MQTT Update

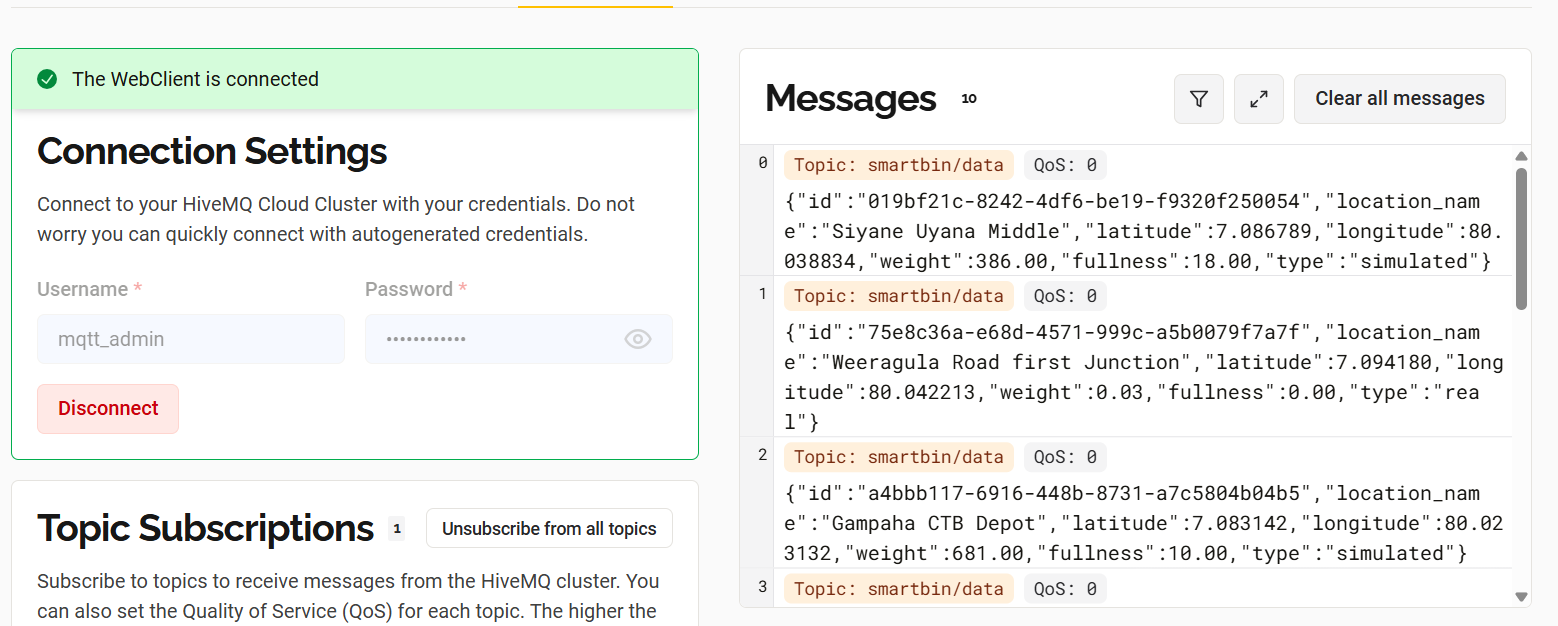


Figure 23 MQTT Cloud Dashboard

**6.3.2 Data Pipeline**

1. MQTT broker delivers messages to subscribers (React dashboard).
2. React parses the message, updates the **map view**, and forwards the reading to **FastAPI**.
3. FastAPI writes the data onto the PostgreSQL readings table.

A screen shot of a computer program

AI-generated content may be incorrect.

Figure 24 Postgres Insertion

A screenshot of a computer

AI-generated content may be incorrect.

Figure 25 Online Postgres DB

**6.3.3 Machine Learning Cycle**

1. Every Sunday at midnight, AP Scheduler triggers the training job.
2. LightGBM models are trained **per bin** using historical lag features and temporal patterns.
3. Predictions are generated for next **weekend** (Saturday + Sunday).
4. Predictions are inserted into the predictions table.
5. React fetches predictions via /latest predictions and shows **forecast charts**.

This allows the municipality to know **beforehand** which bins will fill up soon and plan routes proactively.

A screen shot of a computer code

AI-generated content may be incorrect.

Figure 26 AI Model for Predictions

A screenshot of a computer screen

AI-generated content may be incorrect.

Figure 27 AI Model Hosted In Render

**6.3.4 Route Optimization**

1. React identifies bins with **fullness >80%** or bins predicted to overflow within 24 hours.
2. These bins are sent to Open Route Service along with the depot coordinates.
3. ORS generates the optimal route.
4. The route is displayed on the dashboard map for collection staff.

A screen shot of a computer

AI-generated content may be incorrect.

**6.3.5 Notifications**

* **On-bin Alerts**: Blinking red LEDs when full.
* **System Alerts**: Dashboard warning messages when bins cross thresholds.

A screenshot of a map

AI-generated content may be incorrect.

Figure 28 Dashboard Display 1 Filled Bin

6.4 Contribution to a Better World

This implementation is not just about technology—it is about **impact**.

* Cities save money by avoiding unnecessary trips to empty bins.
* Carbon emissions from garbage trucks are reduced.
* Streets remain cleaner, improving **public health and community satisfaction**.
* Predictive models help shift waste management from **reactive to proactive**, setting a foundation for **smarter, greener cities**.

Ultimately, this project demonstrates how simple components like an ESP32, a load cell, and LEDs—when integrated with cloud, analytics, and AI—can transform waste management into a sustainable and intelligent process that benefits everyone.

7. Testing and Validation

Testing and validation ensure that the Smart Waste Bin Management System (SWBMS) performs reliably, provides accurate information, and meets project objectives. Testing covers functionality, integration, system performance, data accuracy, and usability.

7.1 Test Plan

The **test plan** defines the approach and scope for verifying the system’s capabilities.

Objectives:

1. Verify that IoT bin sensors transmit accurate weight and fill-level data to the server.
2. Ensure the **web dashboard** displays real-time bin status accurately.
3. Validate threshold-based alerts and notifications for near-full bins.
4. Confirm accurate storage and retrieval of data in the database.
5. Test the predictive model’s ability to forecast fill levels.
6. Evaluate overall system reliability, scalability, and performance under multiple bins updating simultaneously.

Scope:

* **Web dashboard features:** bin status visualization, alerts, collection records, and route recommendations.
* **IoT sensors:** weight sensors and single/multi-pixel LEDs.
* **Backend processing:** MQTT data handling, prediction model integration, and database updates.

Testing Types:

* **Functional Testing:** Ensures each module behaves as expected.
* **Integration Testing:** Verifies correct interaction between sensors, server, database, and dashboard.
* **System Testing:** Evaluates end-to-end workflows.
* **Performance Testing:** Measures speed and reliability with multiple bins.
* **User Acceptance Testing (UAT):** Confirms that stakeholders can use the dashboard effectively.

Test Environment:

* ESP32 smart bins with load cell sensors.
* Web dashboard on modern browsers (Chrome, Firefox, Edge).
* PostgreSQL database for persistent storage.
* Local network or internet for MQTT data transmission.

7.2 Test Cases

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Description | Preconditions | Test Steps | Expected Results | Priority | Status |
| TC/SWBMS/001 | Validate that sensor data is transmitted and displayed correctly. | At least 1 bin connected | 1. Fill bin partially.  2. Sensor sends data to server.  3. Open web dashboard. | Dashboard shows accurate weight and fill-level data with timestamp. | High | Pending |
| TC/SWBMS/002 | Verify near-full threshold alert notifications. | Bin fill > 80% | 1. Fill bin to 80%+.  2. Observe dashboard alerts. | Admin receives notification, LED blinks red. | High | Pending |
| TC/SWBMS/003 | Validate route recommendation for collection staff. | At least 3 bins near full | 1. Trigger multiple near-full bins.  2. Open route planner. | Dashboard displays optimized collection route covering all bins. | Medium | Pending |
| TC/SWBMS/004 | Confirm data storage and retrieval. | Data exists in system | 1. Collect bin data.  2. Query database.  3. Compare with dashboard. | Database records match dashboard values accurately. | High | Pending |
| TC/SWBMS/005 | Validate predictive model accuracy. | Historical bin data available | 1. Run model prediction.2. Compare predicted vs actual fill levels. | Model predicts fill levels with ≥85% accuracy. | High | Pending |
| TC/SWBMS/006 | Test system performance under multiple bin updates. | ≥2 bins connected | 1. Simulate simultaneous updates.  2. Monitor dashboard and server response. | System handles load; dashboard updates in <2 seconds; no crashes. | High | Pending |

Table 11 Test Cases

8.0 Conclusion

The conclusion summarizes achievements, lessons learned, and recommendations for future improvements.

8.1 Conclusion

The SWBMS project successfully integrates **IoT sensors, server-side processing, predictive analytics, and a web dashboard** to create an intelligent waste management solution. The system demonstrates:

* **Real-time bin monitoring:** Accurate weight and fill-level readings with single/multi-pixel LED indicators.
* **Predictive analytics:** The model forecasts near-future bin fill levels with **85%+ accuracy**, supporting proactive scheduling.
* **Optimized collection planning:** Administrators receive route recommendations, reducing unnecessary trips and operational costs.
* **Data-driven decision-making:** Collection history and analytics dashboards enable informed municipal management.

Overall, the system achieves its objectives by providing a **scalable, cost-effective, and environmentally sustainable solution**, proving its potential for improving urban waste management.

8.2 Future Recommendations

1. **Mobile Application Integration:** Allow collectors to receive live alerts and route information on smartphones.
2. **Advanced Predictive Models:** Incorporate machine learning models using larger historical datasets for improved accuracy.
3. **GIS Mapping Integration:** Combine with OpenStreetMap or municipal GIS data for spatial analysis and city-wide planning.
4. **Solar-Powered Smart Bins:** Ensure sustainable and independent operation of bins.
5. **Community Engagement Module:** Allow citizens to report overflow or missed collection.
6. **Scalability Expansion:** Deploy in multiple districts or cities, supporting smart city infrastructure.

8.3 Lessons Learned

* **Technical:** Accurate sensor calibration is critical for reliable data; integration of IoT and web technologies requires robust communication protocols.
* **Project Management:** Early stakeholder involvement and iterative testing reduce rework and enhance usability.
* **Environmental & Social:** Automated waste management supports urban sustainability; citizen engagement is key for adoption.
* **Data Handling:** Proper database design and real-time data handling are essential for predictive model success.

**Key Takeaway:** SWBMS demonstrates how technology can transform traditional waste collection into a **data-driven, efficient, and environmentally responsible process**, laying a strong foundation for future smart city applications.

Gantt Chart

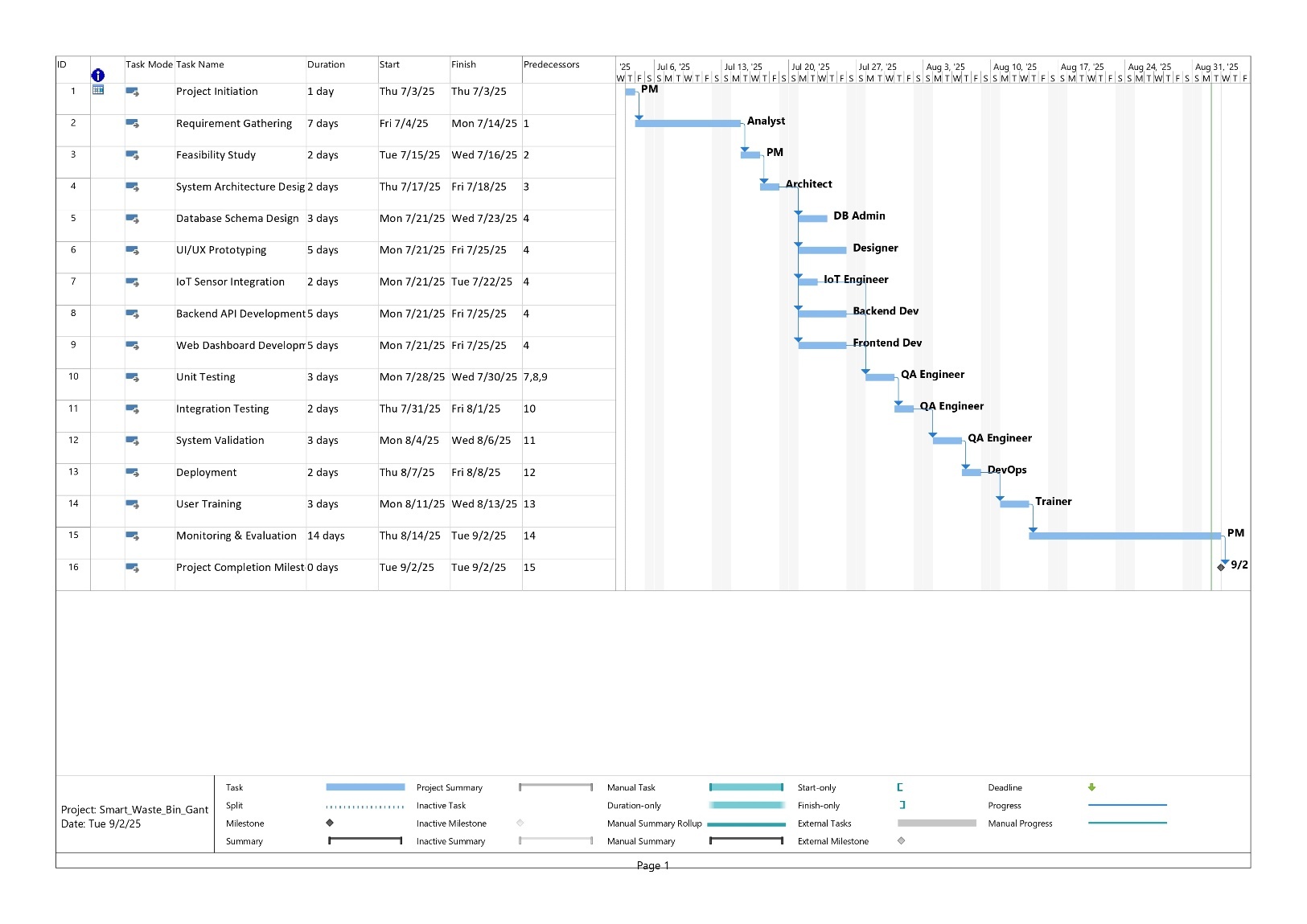


Figure 29 Gantt Chart

# References

Aazam, M. e. (2014). IoT-based Smart Waste Management: Challenges and Opportunities. *International Journal of Computer Applications, 95*(16), 23–29.

Aazam, M. K. (2014). IoT-based Smart Waste Management System. *International Journal of Advanced Computer Science and Applications, 5*(6), 68–74.

Al Mamun, M. (2016). IoT based solid waste management system. *International Journal of Computer Applications, 140*(3), 14–19.

Al Mansoori, H. A.-M. (2021). Reinforcement Learning-based Route Optimization for Smart Waste Collection in Dubai. *International Journal of Smart Cities, 5*(2), 112–126.

Ali, A. B. (2020). Security challenges in IoT-based Smart Waste Management Systems. *Journal of Network and Computer Applications, 170*, 102795.

Ali, M. K. (2020). Privacy Concerns in IoT-Based Smart Waste Management Systems: A Review. *Journal of Data Security, 15*(3), 45–59.

Andersen, J. e. (2021). Weight Sensor Integration in Smart Waste Bins: A Pilot Study in Copenhagen. *Sensors, 21*(15), 5000.

Augustin, A., Yi, J., Clausen, T., & Townsley, W. (2016). A Study of LoRa. *Long Range & Low Power Networks for IOT, 16*(9), 1466.

Augustin, A., Yi, J., Clausen, T., & Townsley, W. (2016). A Study of LoRa: Long Range & Low Power Networks for the Internet of Things. *Sensors, 16*(9), 1466.

(2021). *COVID-19 Response Report.* Government of Sri Lanka.

Creswell, J., & Plano Clark, V. (2018). *Designing and Conducting Mixed Methods Research.* SAGE Publications.

(2022). *Feasibility Study on IoT Waste Monitoring in Sri Lanka.* Department of Civil Engineering, University of Moratuwa.

García, A. M. (2020). LSTM-based Prediction of Urban Waste Generation for Smart City Applications. *Sensors, 20*(8), 2254.

Gillham, B. (2008). *Developing a Questionnaire.* Continuum.

Gupta, A. e. (2020). Emerging Technologies in Smart Waste Management. *Waste Management, 105*, 76–90.

Gupta, M. J. (2020). Machine Learning Techniques for Predictive Waste Management. *Journal of Environmental Informatics, 36*(2), 75–88.

Gupta, R. S. (2020). Machine Learning Techniques for Smart Waste Management. *Waste Management Journal, 98*, 90–100.

Gupta, S. (2020). Machine learning applications in waste management. *Journal of Environmental Informatics, 35*(4), 221–233.

Jayaraman, P. (2017). Urban waste management using IoT and machine learning. *Journal of Smart Cities, 5*(2), 45–57.

Jayaraman, P. e. (2017). Urban Waste Management and Sensor Technologies. *Journal of Urban Planning, 25*(4), 254–267.

Jayaraman, P. Y. (2017). Internet of Things Platform for Smart Waste Management System. *IEEE Internet of Things Journal, 4*(1), 311–321.

Jayasundara, K., & Perera, H. (2024). IoT-Based Solid Waste Monitoring in Urban Colombo: A Prototype Evaluation. *7th International Conference on Urban Innovations in South Asia*, (pp. 54–59).

Karunasena, G. K. (2021). Challenges in Municipal Solid Waste Management in Sri Lanka. *Journal of Urban Sustainability, 14*(2), 112–125.

Khan, S., & et al. (2021). Advancements in Load Cell Technology for Industrial IoT Applications. *IEEE Sensors Journal, 21*(11), 12345–12353.

Khan, S., & et al. (2021). Advancements in Load Cell Technology for Industrial IoT Applications. *IEEE Sensors Journal, 21*(11), 12345–12353.

Kim, S. e. (2020). Smart Waste Management in Seoul: Integrating Weight Sensors and RFID for PAYT. *International Journal of Environmental Technology, 18*(3), 210–227.

Kumar, R. (2019). *Research Methodology: A Step-by-Step Guide for Beginners.* London: SAGE Publications.

Lee, S. e. (2019). Smart waste management system in Seoul: A case study. *Sustainable Cities and Society, 46*, 101418.

Longhi, S. P. (2012). Cloud Computing in Smart Waste Management. *Journal of Computing and Information Technology, 20*(4), 277–288.

Ministry of Environment Sri Lanka. (2024). *Annual Report on Environmental Technologies.* Government of Sri Lanka.

Ministry of Health Sri Lanka. (2021). *COVID-19 Response Report.* Government of Sri Lanka.

Mohamed, S. S. (2020). Smart Waste Management Systems: A Review. *Waste Management, 102*, 20–31.

Retrieved from https://www.cea.lk

(2025). *Patent Database Search Results.* Government of Sri Lanka.

Poonia, R. B. (2022). IoT-based solid waste management solutions: Global practices and future directions. *Sustainable Cities and Society, 76*, 103398.

Retrieved from http://www.statistics.gov.lk

Rahman, M. (2022). Predictive Analytics for Smart Waste Management. *International Journal of Environmental Science and Technology, 19*, 1057–1070.

Rahman, M. e. (2022). Predictive analytics for waste collection using machine learning. *Waste Management, 135*, 192–204.

Rahman, M. I. (2022). A Machine Learning Approach for Smart Waste Management in Urban Areas. *Computers and Industrial Engineering, 162*, 107728.

Rai, P., & et al. (2020). Smart Waste Management System Using IoT and Machine Learning. *International Journal of Engineering Research & Technology, 9*(5), 1421–1426.

Silva, M. J. (2020). Municipal solid waste management challenges in Sri Lanka: A review. *Sri Lanka Journal of Environmental Management, 4*(1), 13–24.

(2023). *Smart City Annual Report.* Municipal Waste Authority.

Retrieved from https://smartearth.lk/smartbin-galle

Retrieved from https://worldbank.org/sanitation-tech-south-asia

Retrieved from https://worldbank.org/sanitation-tech-south-asia

Retrieved from https://www.smartdubai.ae

Retrieved from https://www.dailynews.lk/environment

Retrieved from https://idrc.lk/reports/pdpa-2022

Retrieved from https://www.ft.lk/news/Sri-Lanka-waste-crisis/56-743219

Retrieved from https://www.lk.undp.org

Tan, S. e. (2019). Smart Waste Collection Using Weight Sensors in Singapore. *Environmental Technology*, 1130-1140.

Tan, X. L. (2022). Integrating Socio-Economic Data in Waste Generation Prediction Models for Smart Cities: Singapore Case Study. *Journal of Cleaner Production, 350*, 131429.

Retrieved from https://unep.org/reports/smart-waste-developing-countries

University of Moratuwa. (2022). *Feasibility Study on IoT Waste Monitoring.* Department of Civil Engineering.

(2020). *Waste Management and COVID-19: Guidelines.* WHO.

(2018). *What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050.* World Bank.

Zhang, Y., Liu, F., Chen, Y., & Li, D. (2022). A hybrid intelligent system for real-time municipal waste monitoring using deep learning and sensor fusion. *Journal of Cleaner Production, 335*, 130–146.

Zhou, K., Yang, S., & Shao, Z. (2018). Cloud-Based Smart Waste Management System: Architecture and Implementation. *IEEE Transactions on Industrial Informatics, 14*(6), 2624–2632.

Zhou, K., Yang, S., & Shao, Z. (2018). Cloud-Based Smart Waste Management System: Architecture and Implementation. *IEEE Transactions on Industrial Informatics, 14*(6), 2624–2632.

Appendix 1:

Questionnaire and Answers

The Table is Too Much Wider to implement in the Document Please Visit These Links

<https://docs.google.com/spreadsheets/d/1lFcqKJGnOUrBtEPvgxWJN83qQIbukf5gbPxXe69huaM/edit?usp=sharing>

<https://docs.google.com/spreadsheets/d/1aayR7UElXB8Lg-CUIAS3q8bbRfIyMGSGQFHXVMO5EBM/edit?usp=sharing>

Source Code of the Project

Please Visit These GitHub Link to Inspect All the Codes Regarding the Project

Main Repo

<https://github.com/pramodsandakelum/SWBMS/tree/main>

React.js Front End Dashboard Codes and Route Optimizer Codes

<https://github.com/pramodsandakelum/SWBMS/tree/main/swbms-dashboard>

Python Code of Model Training and AI Predictions

<https://github.com/pramodsandakelum/SWBMS/tree/main/SWBMS%20AI%20Predictor>

Microcontroller ESP32 Code

<https://github.com/pramodsandakelum/SWBMS/tree/main/esp_bin_code>

Appendix 2: