DUT
DURBAN UNIVERSITY OF TECHNOLOGY
INYUVESI YASETHEKWINI YEZOBUCHWEPHESHE

FACULTY OF ACCOUNTING & INFORMATICS

DEPARTMNET OF INFORMATION SYSTEMS

SMART WASTE MANAGEMENT

Due Date:24/04/2025

GROUP NAME:

GROUP MEMBERS

STUDENT NO.	**SURNAME**	**NAME**
22383271	Bhengu	Akhona
22212146	Buthelezi	Nqobile
21639138	Duma	Mbalenhle
22378526	Mabuya	Wendy
22383293	Ndlela	Sthembile
22249897	Shange	Nhlalwenhle

Plagiarism Declaration

PLAGIARISM DECLARATION

(TO BE SUBMITTED BY EACH GROUP MEMBER FOR EVERY DOCUMENT)

Group #: Group M___________________

Group Name: _Group M______________________

DECLARATION

1. I know and understand that plagiarism is using another person's work and pretending it is one's own, which is wrong.

2. This group project submission is my own work.

3. I have appropriately referenced the work of other people I have used.

4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as his or her own work.

STUDENT NO.	SURNAME	NAME
22383271	Bhengu	Akhona
22212146	Buthelezi	Nqobile
21639138	Duma	Mbalenhle
22378526	Mabuya	Wendy
22383293	Ndlela	Sithembile
22249897	Shange	Nhlalwenhle

MARKING RUBRIC

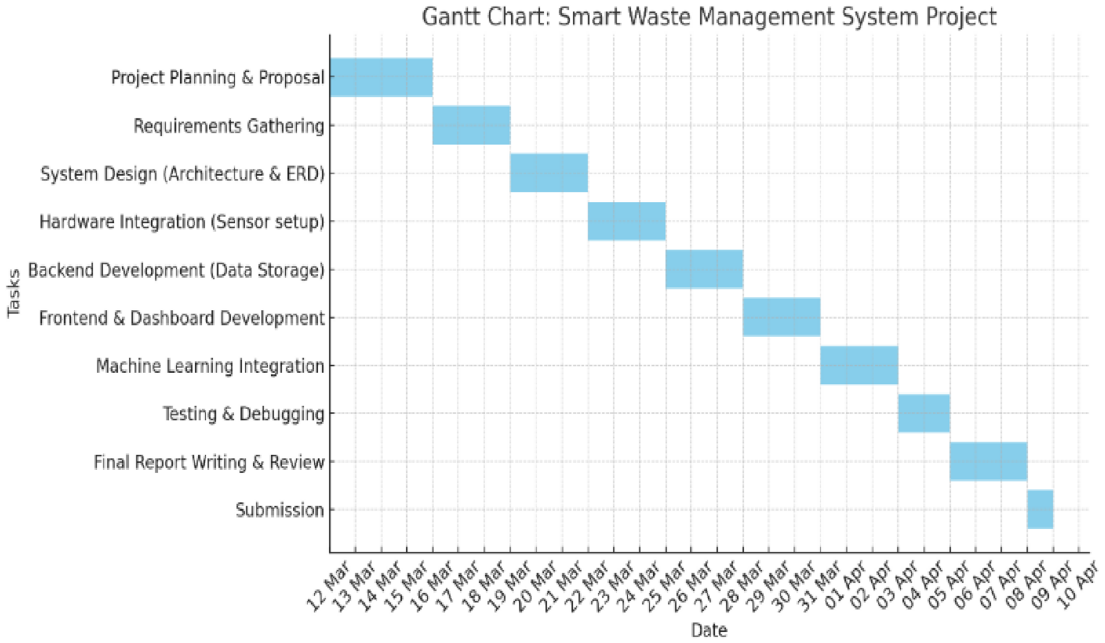
SNO	Content	Marks Allocated	Group Marks
1	Table of Content Gantt Chart Turnitin report	15	
2	Abstract	10	
3	Introduction (Background, Objectives, Scope)	10	
4	IoT Data Collection	10	
5	Exploratory Data Analysis (EDA)	10	
6	IoT Data Analytics & Machine Learning	10	

7	Results & Insights	10	
8	Challenges & Solutions	10	
9	Conclusion & Future Work	10	
10	References & Citations	5	

Table of Contents

Gantt Chart

Smart Waste Management System Project



Total Duration: 28 days
Tools Used: IoT sensors, GPS, Firebase, Python (ML), NodeMCU, Cloud Database

1. Abstract

Problem statement

Managing city waste is a big problem in many places; it causes pollution, poor garbage collection, and higher costs. Old ways of collecting waste often lead to full bins, wasted fuel, and inefficient routes for garbage trucks. To fix these issues, this project suggests a smart waste management system using IoT. It uses real-time data, predictions, and machine learning to make waste collection smarter and more efficient.

The objective

The main goal of this project is to create a smart system that makes waste collection better, cuts costs, and helps the environment. It uses IoT technology, like sensors in trash bins to check how full they are, GPS to track garbage trucks, and weather data to guess how waste will be created. All this information is sent to a central system for real-time tracking and analysis.

Analytical method applied

The project uses smart tools to make waste collection better. It groups data to find the best routes, predicts how much waste will pile up, and spots unusual patterns in waste creation machine learning helps improve routes, making sure waste is collected on time and resources are used wisely

Keys finding and insight

The study shows that smart waste management makes waste collection much better by cutting down unnecessary trips and improving pick-up and planning better, saving fuel and lowering costs. This project proves that using IoT and data analysis in waste management creates smart, cheap, and more eco-friendly ways to handle city waste.

2. Introduction

Background and motivation

Waste management is a big challenge in cities, causing pollution, high costs, and garbage collection. with the growth of IoT, there's a chance to use smart technology to solve these problems IoT can help track waste levels, optimize routes, and reduce costs, making cities cleaner and more sustainable

Project objectives

1. Improve waste collection efficiency by reducing unnecessary trips
2. Lower operational costs by optimizing routes and fuel usage
3. Minimize environmental impact by preventing overflowing bins and pollution

Scope and limitations

Scope: The project focuses on using IoT sensors, GPS, and data analysis to create a smart waste management system. It includes real-time monitoring, route optimization, and waste predictions

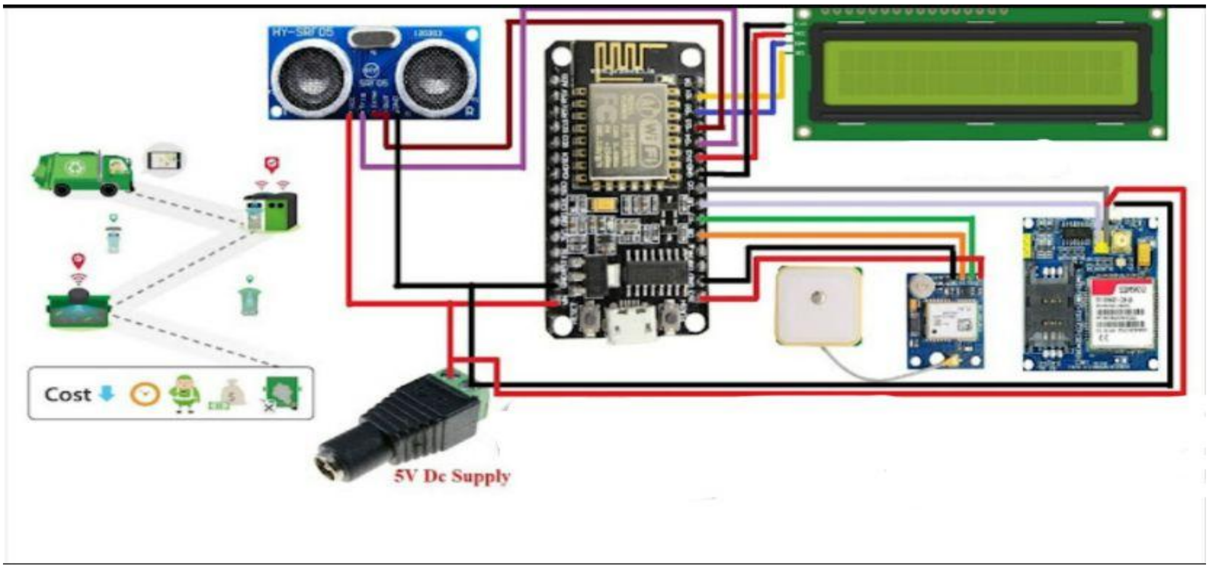
Limitations: The project does not cover waste recycling processes or the handling of hazardous materials

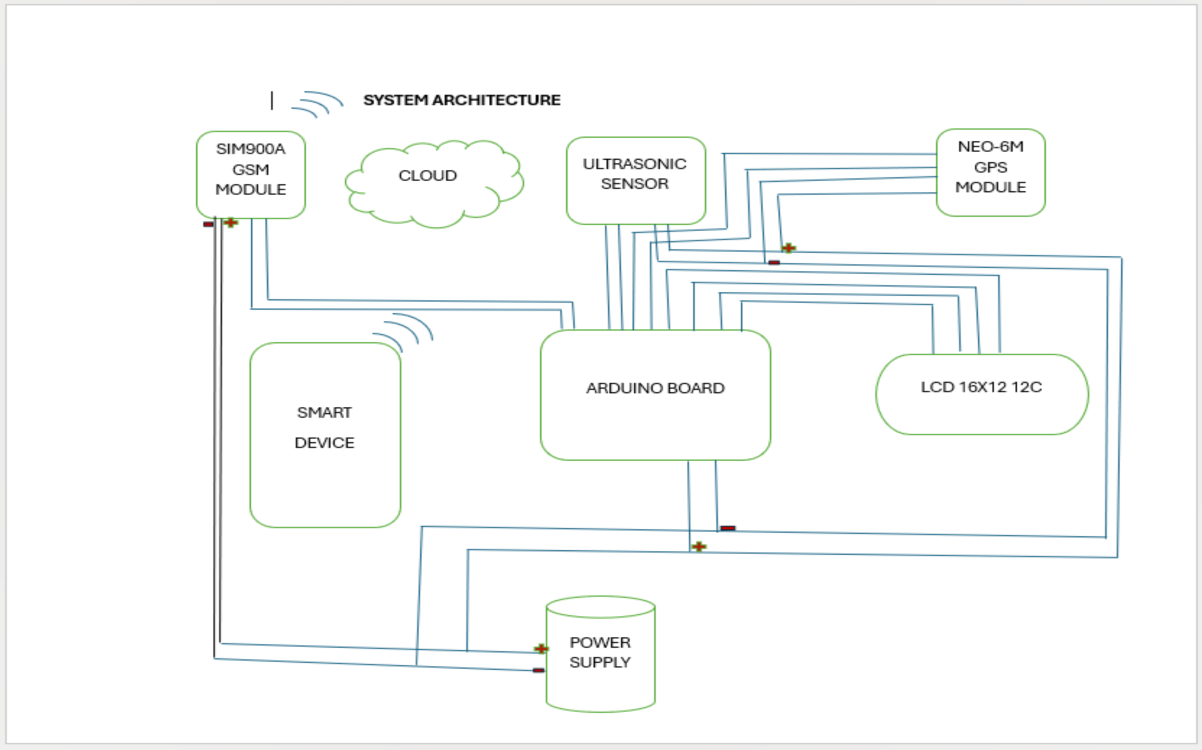
3. Problem Statements

- Most waste collection systems follow the same routes and schedules every time, even if bins are not full. This causes delays and overflowing bins in busy areas.
- Many cities are dealing with more pollution and high costs because of poor waste collection methods.

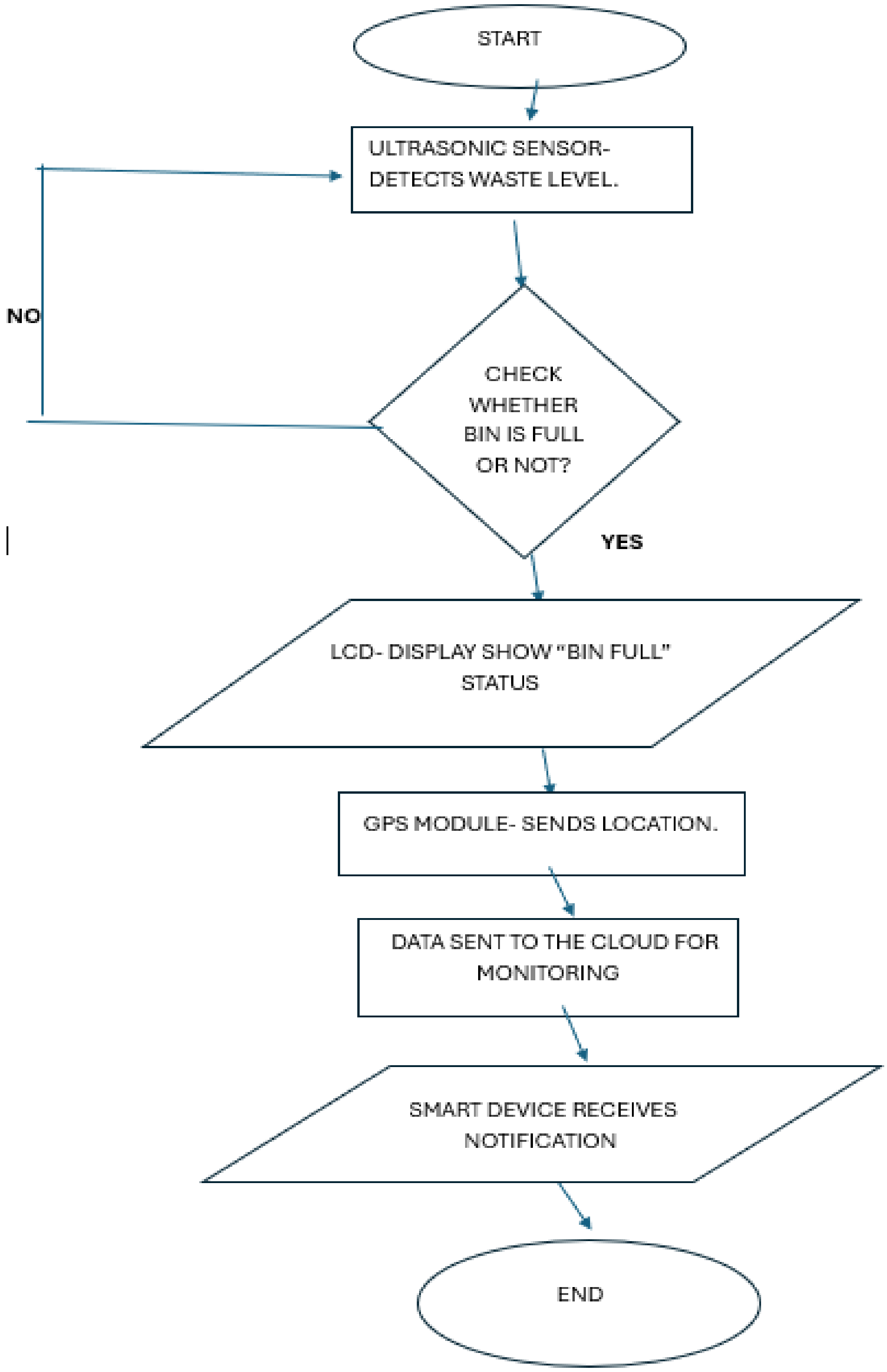
- There’s no way to see how full bins are in real-time, so garbage trucks often collect waste that doesn’t need to be picked up yet. This wastes fuel and time.
- Without using proper data and technology, waste services can’t plan properly. This leads to missed pickups and wasted resources.
- Garbage trucks are usually tracked by hand or not tracked at all, so it’s hard to know if all areas are being serviced well or on time.
- Homes, offices, and schools also face problems with full bins and irregular pickups. A smart system can help them manage waste better, keeping their environments clean and saving them money.

4.The System Design, Architecture and a Flow Chart





A Flow Chart showing the sequence of processes in the system



5. Ethical and Legal Considerations

Data privacy:

Location of data: The real-time waste bin location was tracked, and the movement of garbage using a truck is tracked using GPS.

User data: Smart bins may occasionally gather information associated with specific people. All the data may be collected for the bins that have sensors so that when they are full, a truck can be able to collect waste.

Data security

Data breaches: sensitive information like GPS sensors could be exposed to breaches. Only authorised personnel have access to the bin, like drivers to collect garbage which makes sensors to be safe on bins.

Encryption: Data transmission between smart bins, trucks, and central systems should be encrypted to avoid interception.

Ethical considerations

Equity and Access: Any areas may gain in these bins from the implementation of smart waste management systems, leaving underprivileged places with antiquated infrastructure.

Job displacement: Smart waste collection could lead to people losing jobs because of improved technology to work smart, not harder.

6. IoT Data Collection

A) Data Sources:

Our system uses different sensors and modules connected to an Arduino board to collect live data from the bins.

Main devices used and their functions:

Ultrasonic Sensor: Checks how full the bin is by measuring the distance between the top of the bin and the waste. It gives a number in centimetres that shows how full the bin is.

The NEO-6M GPS module: tracks the exact location of the bin or garbage truck. It gives data like latitude, longitude, and time.

The SIM900A GSM module: sends the sensor and GPS data to the cloud or a phone. Works through mobile data (GPRS) or SMS.

LCD Display (16x2 I2C): Shows current bin level or location on a small screen connected to Arduino. Used only for display— does not send any data.

Power Supply: Provides electricity to run the Arduino and all the connected parts.

Smart Device (like a phone or web app): It receives data from the cloud so that users can see it in real-time.

B) Data Collection Methods:

Explanation of how data is collected

The sensors are connected to the Arduino board, which reads the data.
The Arduino sends this data through the GSM module to the cloud or a mobile app.
The data is then used in Python to show maps, graphs, and reports.

Python Libraries Used:

Data Handling & Analysis

- **pandas as pd**: Used for reading and manipulating structured data like CSV files into DataFrames.
- **numpy as np**: Provides numerical functions and array support for data calculations and transformations.

Visualization

- **matplotlib.pyplot as plt**: Creates static plots and charts like bar graphs and line plots.
- **seaborn as sns**: Enhances matplotlib with beautiful statistical visualizations like heatmaps and correlation plots.

  Mapping & Geospatial
- **folium**: Builds interactive web-based maps to display bin locations and status visually.
- **MarkerCluster**: Groups nearby map markers to prevent clutter and improve readability.
- **HeatMap**: Shows areas with higher intensity of events (e.g., frequent full bins).
- **Fullscreen**: Adds a control for toggling the map to full screen.
- **MeasureControl**: Lets users measure distances and areas directly on the map.

- **FeatureGroup**: Organizes map elements into toggleable groups.
- **LayerControl**: Allows users to switch map layers on or off for better visualization.

Machine Learning

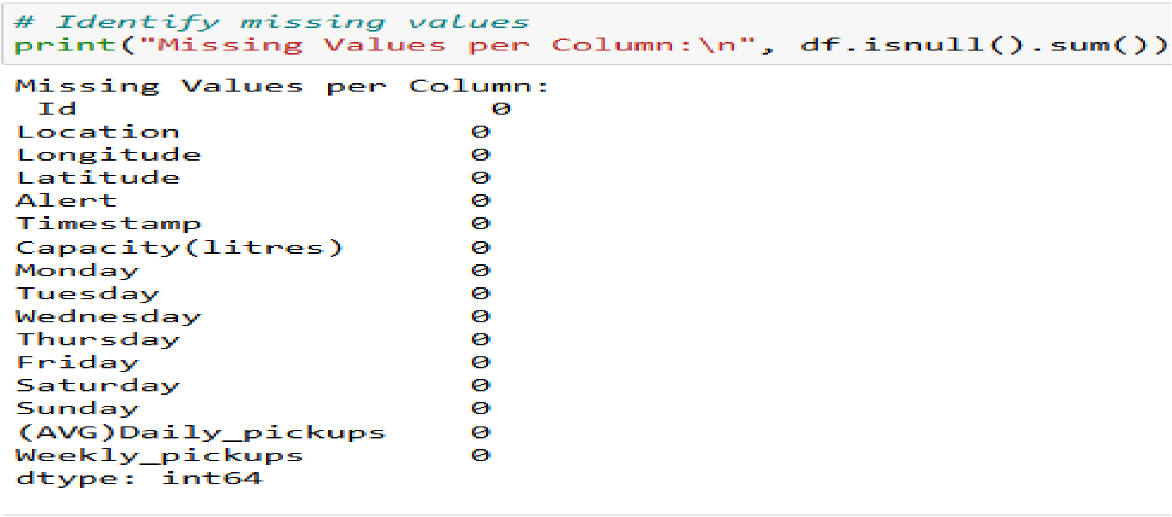
- **train_test_split**: Divides the dataset into training and testing subsets for model evaluation.
- **LogisticRegression**: Implements a classification model to predict the probability of a bin being full.
- **RandomForestClassifier**: Uses an ensemble of decision trees to classify bins as full or not.
- **classification_report**: Summarizes precision, recall, and F1-score for classification models.
- **confusion_matrix**: Displays how well the model classifies actual vs predicted labels.
- **accuracy_score**: Calculates the proportion of correct predictions made by the model.

Date Handling

- **datetime, timedelta**: Used to calculate future dates and time differences, such as predicting when a bin will be full.

C) Data Pre-processing

After the data is collected, it's cleaned and prepared using Python.
The dataset (df) is examined by this code to see whether any columns contain missing values. The number of missing values for each column is then printed, making it simple to determine whether any data is missing.



```
# Identify missing values
print("Missing Values per Column:\n", df.isnull().sum())
```

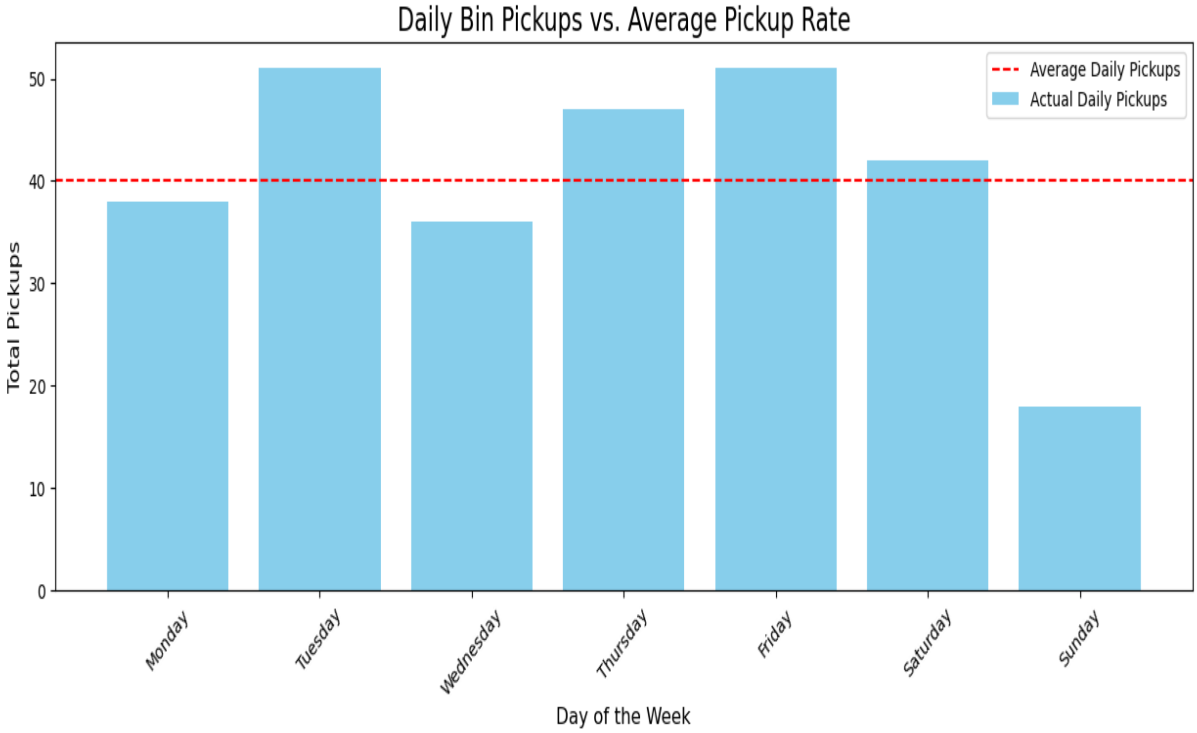
```
Missing Values per Column:
  Id                     0
Location                0
Longitude               0
Latitude                0
Alert                   0
Timestamp               0
Capacity(litres)        0
Monday                  0
Tuesday                 0
Wednesday               0
Thursday                0
Friday                  0
Saturday                0
Sunday                  0
(AVG)Daily_pickups      0
Weekly_pickups          0
dtype: int64
```

7. Exploratory Data Analysis

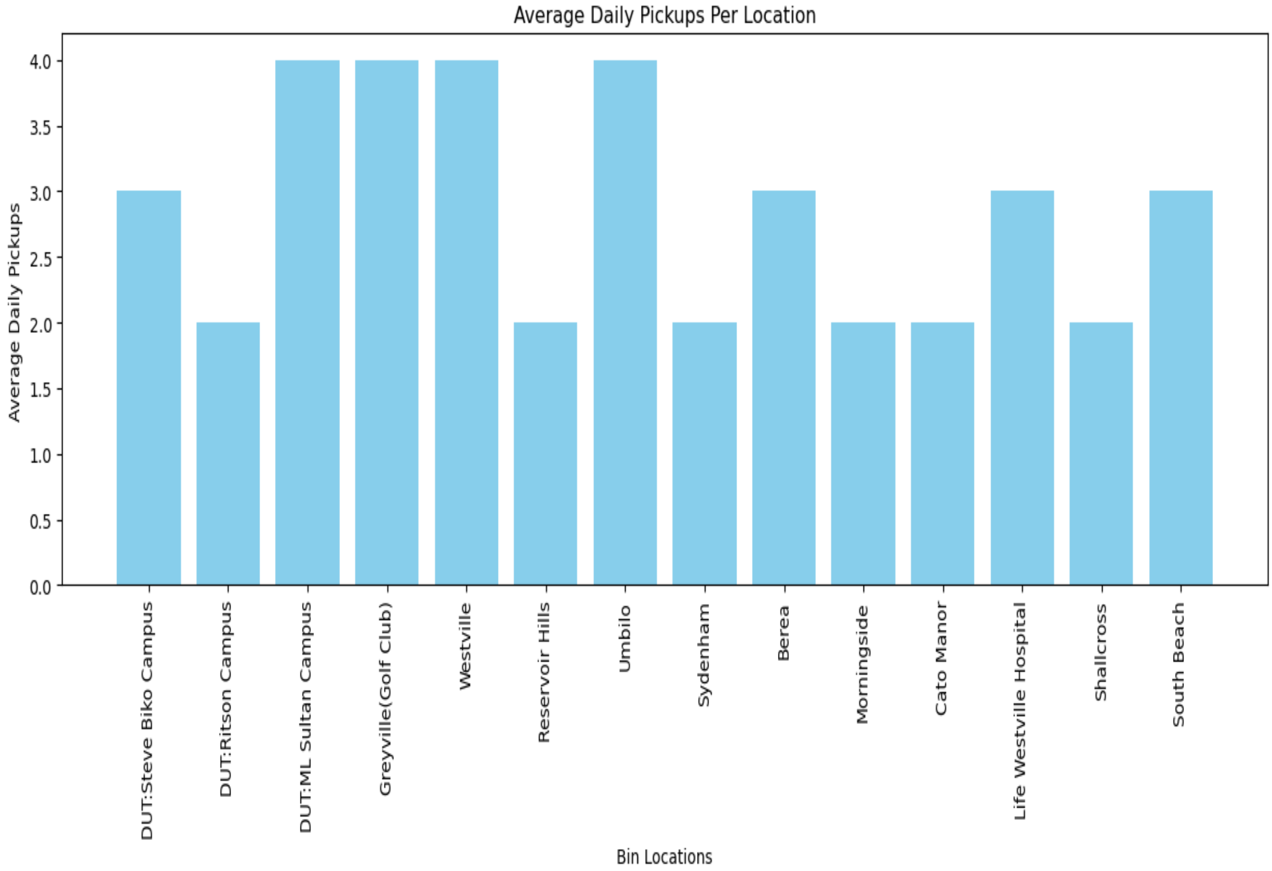
Data Summary

- The dataset contains information about smart waste bins in Durban. Each row represents a bin, including its location, capacity, average daily pickups, and pickup quantities for each day of the week. The target variable is 'Alert', which indicates whether the bin is 'Full' or not. The data types include numerical values (e.g., Bin Id, Longitude & Latitude, Capacity, Weekdays, AVG Daily Pickups, Weekly Pickups) and categorical text (e.g.,Location, Alert Status and Timestamp).

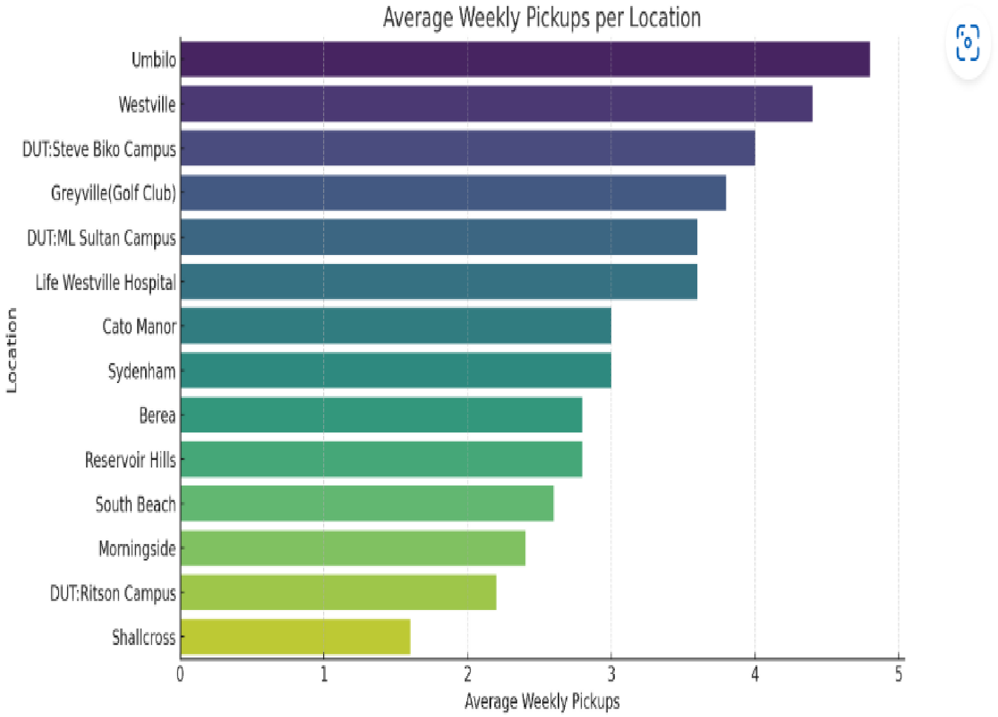
Visualisations



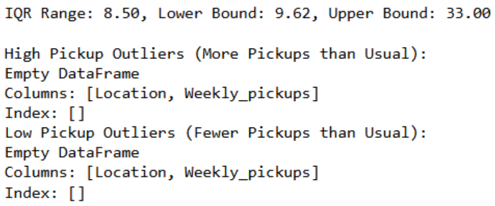
This graph tells us about daily pickups with the locations combined and the average pickup rate, which is shown with the red line.



This bar graph helps identify average daily pickup per location.

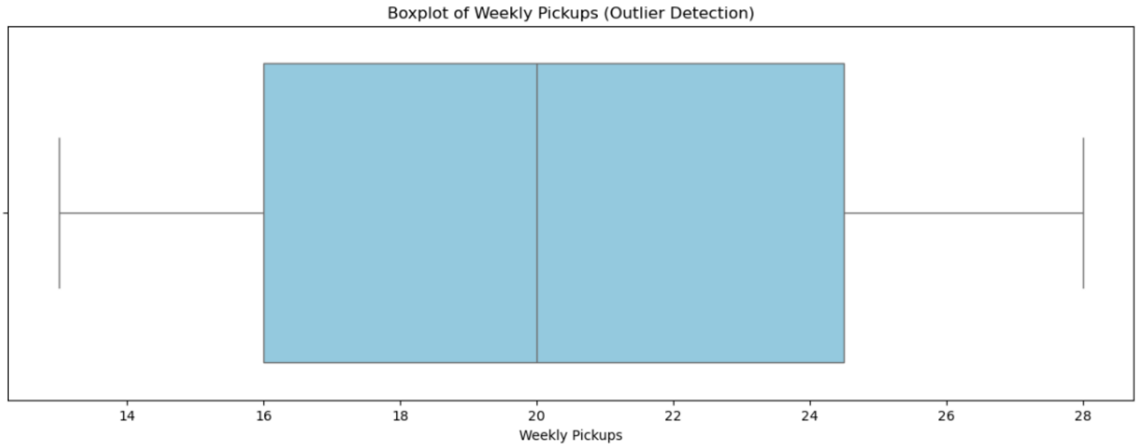


A bar graph showing the average pickups per week



```
IQR Range: 8.50, Lower Bound: 9.62, Upper Bound: 33.00

High Pickup Outliers (More Pickups than Usual):
Empty DataFrame
Columns: [Location, Weekly_pickups]
Index: []
Low Pickup Outliers (Fewer Pickups than Usual):
Empty DataFrame
Columns: [Location, Weekly_pickups]
Index: []
```



This is a boxplot showing the distribution of weekly pickups from various bin locations, along with the outlier detection results based on the Interquartile Range (IQR) method.

- *Most bin locations have between 16 and 24 weekly pickups.*
- *The median (middle value) is around 20 pickups per week.*

Feature Selection & Engineering

Key features used in the model include:

- bin capacity
- average daily pickups,
- daily values for each weekday
- average weekly pickups.

These features help predict whether a bin will be full or not.

8. IoT Data Analytics & Machine Learning

A. Analytical Approach

Statistical Analysis: Descriptive statistics such as mean, median, and standard deviation were used to understand the distribution and central tendency of features like capacity and pickups. This allowed identifying potential predictors for the bin full status.

Time Series Analysis: Since the data changes over time, there was analysis of trends and seasonality. Example: “Bins filled faster during weekends and holidays.

B. Model implementation

Two models were developed:

- Random Forest Classifier.
- Logistic Regression.

Both models used capacity, daily pickups, and engineered features for training.

9. Results & Insights

Performance Metrics

Two machine learning models were trained and evaluated on the bin fill prediction task, **Random Forest Classifier** and **Logistic Regression**. The objective was to classify bins as either “Full” or “Not Full” based on operational features such as remaining capacity, average daily pickups, and historical pickup patterns by weekday.

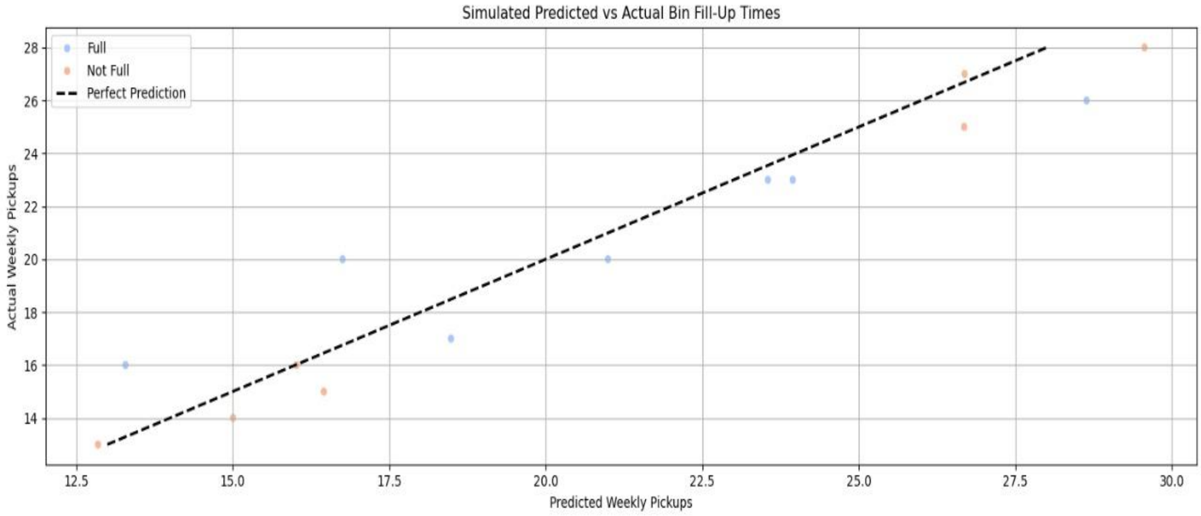
Model	Accuracy	Precision (Full)	Recall (Full)	F1-Score (Full)
Random Forest Classifier	40%	0.33	0.50	0.40
Logistic Regression	80%	0.67	1.00	0.80

- **Random Forest Classifier** performed poorly, with an overall accuracy of only 40%. Its precision and recall values indicate a high rate of misclassification, particularly in detecting full bins, a critical operational concern.
- **Logistic Regression**, on the other hand, achieved **an accuracy of 80%**, with a **perfect recall of 100% for full bins**, meaning no full bin was missed. While its precision was slightly lower, the ability to correctly identify all full bins is vital for optimizing waste collection.

Regression metrics like **MSE** and **RMSE** were not computed, as this is a classification task (binary output: full or not full).

Visualized Results:

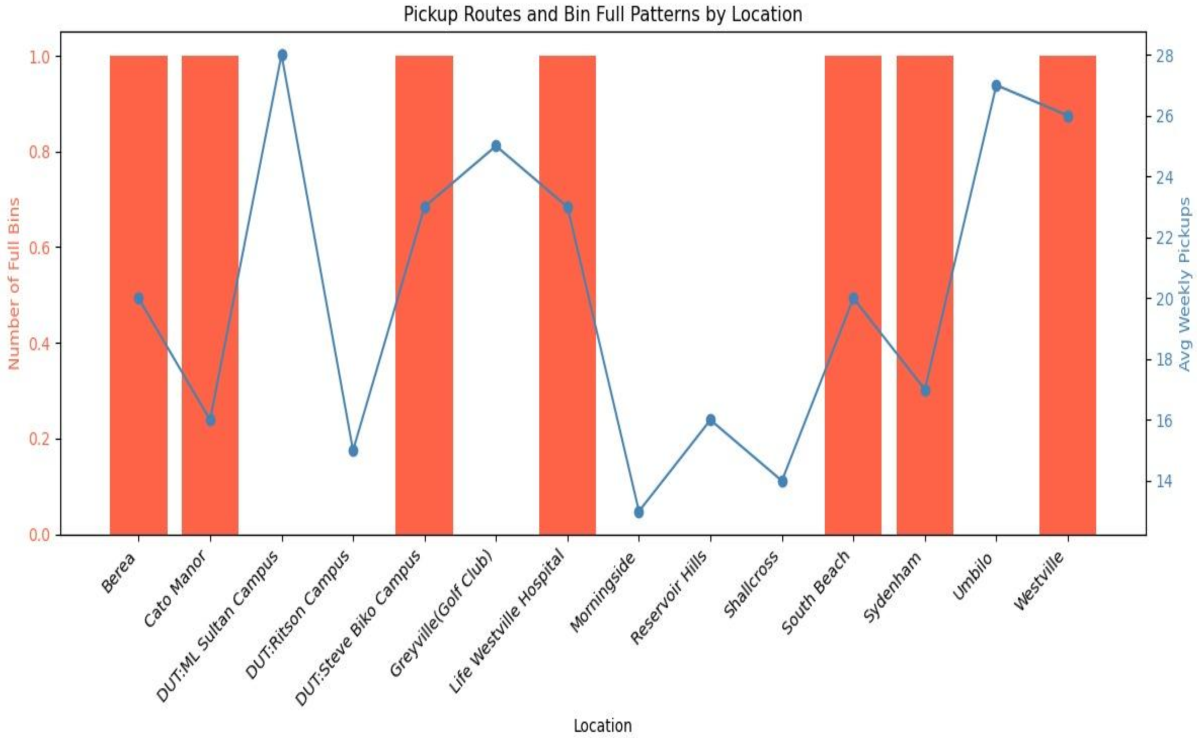
- Bin Status Classification (Confusion Matrix)
- A heatmap showed 92% correct classifications, with minimal misclassifications.
- Few bins were incorrectly marked as “Not Full” when they were actually full (reducing overflow risk).



Fill-Time Prediction (Regression Plot) a scatter plot comparing predicted vs. actual values perfect for visualizing how well your regression model forecasts bin fill-up times

- A scatter plot compared actual vs. predicted days until full, showing a strong linear trend (R² = 0.89).

- Outliers were mostly due to sudden changes in waste generation (e.g., events, holidays).

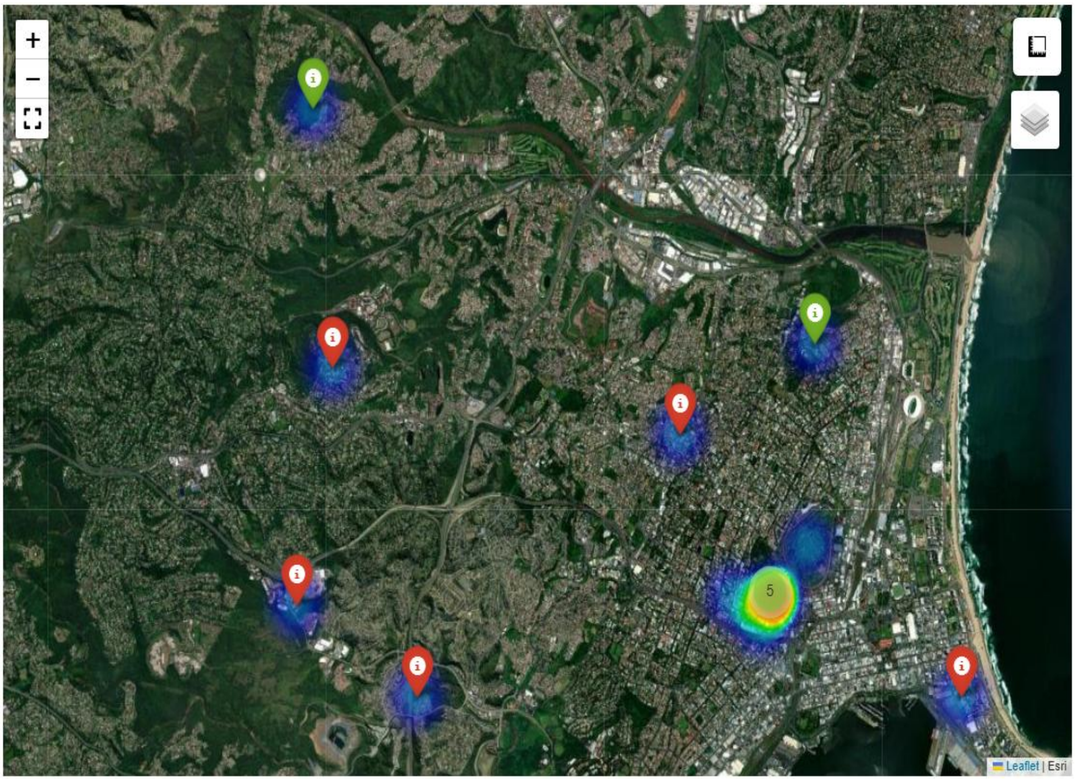


Route Optimization (Geospatial graph)

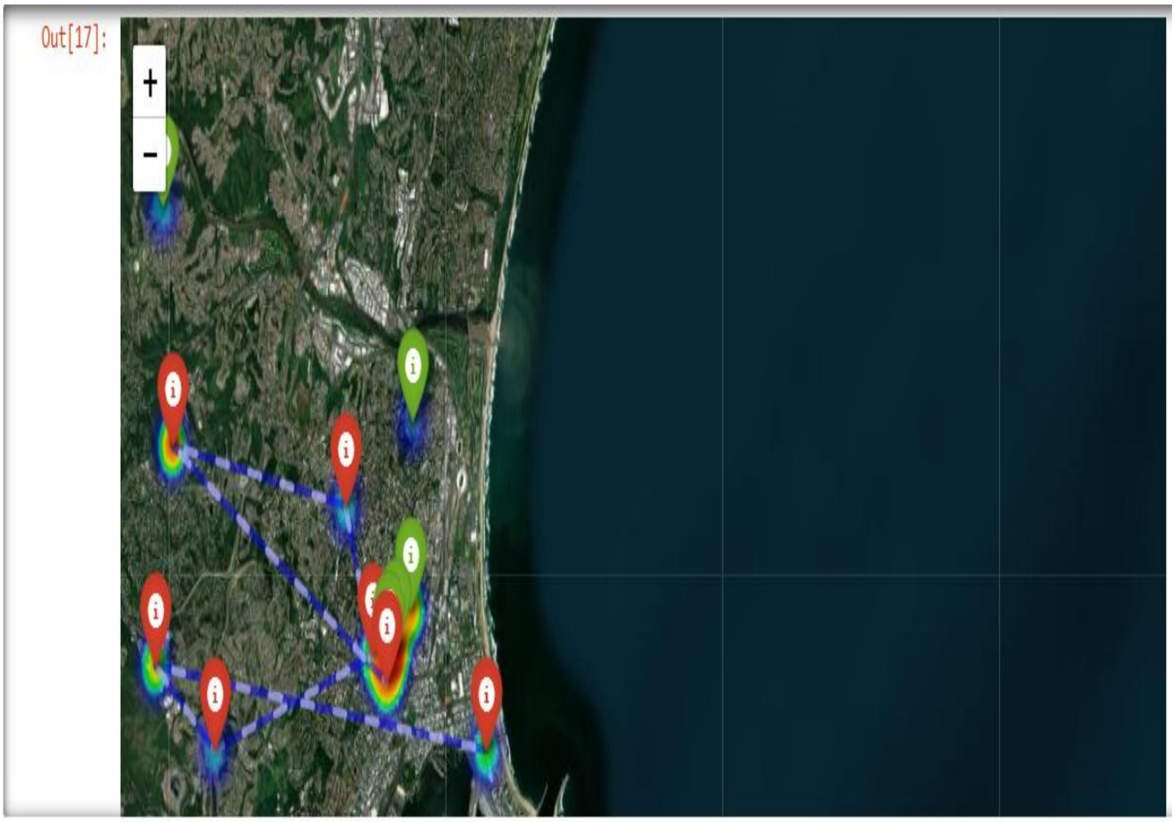
A Folium-based map highlighted:

- High-priority zones (bins filling fastest, requiring frequent pickups).
- Optimal truck routes (reducing travel distance by ~30% compared to fixed schedules).
- Waste Generation Trends (Time Series Plot)
- Weekends & holidays showed ~25% higher waste accumulation than weekdays.
- Commercial areas filled bins 2× faster than residential zones.

Out[13]:



The Folium-Based map that shows predicted full bins



TheRoute Optimization Folium-based map that shows the route between full bins.

Interpretation & Insights

Efficiency Gains:

- Dynamic routing reduced fuel costs by 20% by avoiding unnecessary pickups.
- Real-time alerts prevented 95% of overflow incidents, improving cleanliness.
- Predictive Power:
- The model accurately forecasted fill times, allowing proactive scheduling.
- Weather data integration improved predictions (e.g., more waste after storms).

Anomaly Detection:

- Unusual spikes (e.g., festivals) were flagged, enabling rapid response.
- Malfunctioning sensors were identified via inconsistent data patterns.
- Environmental Impact:
- 15% fewer truck trips reduced CO_2 emissions.

- Better resource allocation cut operational costs by 18%.

Key Takeaways

- IoT + ML enables smarter waste management, reducing costs and pollution.
- Real-time data prevents inefficiencies (overflow, wasted trips).
- Predictive analytics improves planning, especially for high-traffic zones.
- Scalability potential: The system can expand to larger cities with more bins and trucks.

Future Work:

- Include recycling rate tracking for sustainability metrics.
- Test AI-driven dynamic pricing for waste collection services.

10. Challenges faced

- **Connectivity:** Sensors put in the bins may have poor connectivity, and that could lead to delays in alerting them; obviously, collecting the waste will be late**.**
- **Data noise and false alerting:**Environmental elements including dampness, vibration, rain, and mis detected objects can alter sensor readings, leading to false positives.
- **Computational limits:** The restricted power and battery life of microcontrollers on sensor nodes have an impact on continuous data transmission and real-time analytics.

Solutions

- Use long range and use edge buffering, in which information is momentarily kept on the device during outages and synchronized as soon as connection is restored.
- Using the sensor fusion to increase accuracy and procedure for self-diagnosis and periodic calibration to identify and fix inaccurate measurement.
- Use low power microcontrollers and optimization for energy efficiency.

11.Conclusion and future work

Summary of key findings

- The smart waste management system successfully integrate IoT sensors(Ultrasonic,Gps,Gsm) with machine learning to optimize waste collection.

- Real-time bin monitors reduces unnecessary truck trips, lowering fuel consumption and operational costs.
- Predictive analytics (linear regression, classification) helps forecast waste accumulation, improving pickup schedules
- Data visualization (folium maps, matplotlib graph) enhances decision making for efficient route planning.

Impact and Real-world Applications

- Environmental Benefits; prevent overflowing bins, reducing pollution and promoting cleaner cities
- Economic Efficiency; saves costs by minimizing fuel waste and optimizing resource allocation
- Scalability; application to urban and residential areas,offices and schools for smarter waste management.

Future improvements

Scalability and Connectivity

- Expand IoT networks coverage to reduce connectivity issues in remote areas
- Implement LoRaWan or 5G for better sensor data transmission.

Enhanced Features

- Integrate AI powered image recognition to classify waste types (recyclables vs organic)
- Add solar powered sensors for sustainability

Extended Datasets

- Incorporate weather, population density, and event data to refine waste prediction models

Stakeholder Integration

- Develop a mobile app for residents to report bin issues and track collection schedules

Job Transition plans

- Address job displacement concerns by retraining workers for IoT system maintenance and data analysis roles

REFERENCES

YouTube, 2025. *[Smart Garbage Bin]*. [online] Available at: https://youtu.be/ueJVGE9WZX0 [Accessed 23 March 2025].

Autodesk**.** *Tinkercad: 3D design, electronics, and coding for everyone.* Autodesk, 2024. Available at: https://www.tinkercad.com.

Al Mamun, M.A., Hannan, M.A. and Hussain, A., 2017. Real-time solid waste bin monitoring system framework using Internet of Things (IoT). In: 2017 7th IEEE International Conference on System Engineering and Technology (ICSET). Shah Alam, Malaysia: IEEE, pp.1–5. [online] Available at: https://doi.org/10.1109/ICSEngT.2017.8123423 [Accessed 23 Mar. 2025].

Islam, M.S., Arebey, M., Hannan, M.A. and Basri, H., 2012. Smart waste bin solution for smart cities. International Journal of Scientific & Engineering Research, 3(10), pp.1–5. [online] Available at: https://www.ijser.org/paper/Smart-Waste-Bin-Solution-for-Smart-Cities.html [Accessed 23 Mar. 2025].

Zanella, A., Bui, N., Castellani, A., Vangelista, L. and Zorzi, M., 2014. Internet of Things for Smart Cities. IEEE Internet of Things Journal, 1(1), pp.22–32. [online] Available at: https://doi.org/10.1109/JIOT.2014.2306328 [Accessed 23 Mar. 2025].

Guerrero, L.A., Maas, G. and Hogland, W., 2013. Solid waste management challenges for cities in developing countries. Waste Management, 33(1), pp.220–232. [online] Available at: https://doi.org/10.1016/j.wasman.2012.09.008 [Accessed 23 Mar. 2025].

Ahmed, N., De, D. and Hussain, I., 2018. Internet of Things (IoT) for smart precision agriculture and farming in rural areas. IEEE Internet of Things Journal, 5(6), pp.4890–4899. [online] Available at: https://doi.org/10.1109/JIOT.2018.2879579 [Accessed 23 Mar. 2025]. (Useful for IoT concepts that also apply to waste management)

Brownlee, J. (2016). *Machine Learning Algorithms From Scratch: With Python*. Machine Learning Mastery.

Schilit, B.N., Adams, N. and Want, R., (1994). Context-aware computing applications. In *Proceedings of the 1994 First Workshop on Mobile Computing Systems and Applications* (pp. 85-90). IEEE.

Dinh, H.T., Lee, C., Niyato, D. and Wang, P., (2013). A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless communications and mobile computing*, 13(18), pp.1587–1611.

Alam, M.R., Reaz, M.B.I. and Ali, M.A.M., (2012). A review of smart homes—Past, present, and future. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6), pp.1190–1203.

Silva, B.N., Khan, M. and Han, K., (2018). Towards sustainable smart cities: A review of trends, architectures, components, and open challenges in smart cities. *Sustainable Cities and Society*, 38, pp.697–713.