Bachelor of Science with Honours in Computing Science/ Bachelor of Science with Honours in Cyber Security

COURSEWORK No. 2

(BIN IT)

Module Name : **Internet of Things**

Module Code : **PSB602IT**

Credits : **20 CR**

Due Date : **04 September 2026**

SUBMITTED BY

NAME : **DON LOO BOON KIAT**

CU Index No. : **504EHLOC**

Word Count : **2609**

Contents

1. Introduction 3
1.1 Project Brief 3
2. Background Study / Literature Review 3
2.1 Background 3
2.2 Case Studies 3
2.3 Key Problem 4
2.4 Proposed Solution and Rationale 4
2.5 Review of IoT Components and Enabling Technologies 5
3 Prototype Design & Architecture 5
3.1 Prototype Architecture 5
3.2 Prototype Implementation 6
3.2.1 Bin Simulator (bin_simulator.py) 7
3.2.2 Route Optimisation Logic (route_optimizer.py) 8
3.2.3 Live Dashboard 8
3.3 Prototype Evaluation (evaluation_baseline.py) 8
3.4 Security, Privacy and Trust Evaluation 9
4. Conclusion 10
5. List of References 10
6. Appendix 12

1. Introduction

1.1 Project Brief

In this report, the design and testing of the prototype application of the Internet of Things technology for the waste management industry is considered as part of efforts to solve the problem in connection with the UN Sustainable Development Goal #13 Climate Action. The digital transformation of industries using such technologies as cloud computing, automation, and the Internet of Things is becoming increasingly inevitable, since modern society demands efficiency and sustainability from companies and governments. The waste management is a sphere where IoT has already proved successful. At the same time, inefficiencies caused by this technology are still rather common, the garbage trucks operating on a fixed schedule drive to garbage containers that are far from being full, consuming unnecessary fuel and producing CO2 for no reason, while some other containers overfill during this time.

The report begins with a literature review. During the literature review, cases studies that describe IoT applications in the waste management industry were examined. Then, the problem and its description based on the results of the literature review were given, the solution to the problem was suggested, and different hardware, connectivity, and cloud platform options were reviewed. In this report the concept, architecture, and design of the prototype application are also considered, as well as the evaluation of the application's efficiency based on the real data obtained during the implementation of the solution.

2. Background Study / Literature Review

2.1 Background

The chosen area of study for the project is the management of municipal solid waste, specifically the collection of litter from public litter bins. Five real-life case studies have been analysed to determine the problems that this industry and society face today.

2.2 Case Studies

A trial of fill level sensors was conducted on the Derby City Council (UK) 200 litre bins which used to be collected based on a fixed paper schedule independent of the fill levels. This trial allowed for optimal routes to be developed and reduced number of bin collection by 53%, thus reducing overflow complaints and carbon footprint of the city council (REEN, 2023 & Simion, 2023).

Enevo (Finland), in the European Union Horizon 2020 SmartWASTE project, utilised wireless ultrasonic sensors to develop dynamic and demand-based daily collection plans rather than static routing. Results of the project reported collection cost savings of 20 – 50% in addition to emissions, road wear and noise reduction (European Commission, CORDIS). The inefficiencies due to the static routing are identified in the project and trucks are referred

to as being driven “blindly” between bins which are half-empty or overflowing (European Commission, CORDIS).

Sensoneo (Slovakia) conducted a pilot of an end-to-end smart waste system in Bratislava and smart route planning in Prague based on fill level and contamination sensors using radar technology. The report states that despite the narrow streets of Prague, optimised routing contributed to the 60% decrease in CO2 emissions from waste collection (European Commission, CORDIS).

The Edinburgh City Council (United Kingdom) is changing 3,000 communal glass recycling bins from fixed schedules to dynamic sensor-driven collection where bins are serviced only when nearly full of the hope that the collected data will help to identify bins with chronically low use which can be removed (BBC News).

The investigation carried out by Brouwer et al. (2023) is a seven-month academic comparison of the results achieved using ultrasonic sensors monitoring compared to truck drivers’ visual estimations for one Portuguese waste management company that serves 36 cities. Using the predictive model based on Gaussian process, the researchers prove that both methods can significantly decrease the volume of collection and the number of overflows in comparison with the scheduled model. Yet, the research proves that the total implementation of the sensors involves real costs and precision limitations.

2.3 Key Problem

For all five cities, the same inefficiency keeps coming back regardless of country or bin, collection takes place based on fixed schedules rather than actual filling status of each bin. The result is over-servicing for some bins, unnecessary trips, fuel usage, and emissions from bins that fill faster than the schedule. The situation is discussed by Enevo relative to a worldwide baseline of over 400 million containers being collected every day (European Commission, CORDIS), while the effect of solving the problem is stated to be considerable, 20 to 50% cost savings (Enevo), 60% CO2 reduction (Sensoneo), and 53% less servicing trips (Derby). Hence, it becomes clear that this issue is an unnecessary emission factor in urban areas associated with SDG 13. Target 13.2 (climate considerations in planning) and Target 13.3 (institutional capacity building for climate change mitigation) for municipal governments. However, as Brouwer et al. (2023) state, this problem is not fully resolved as accuracy and cost of sensors is a genuine concern.

2.4 Proposed Solution and Rationale

The suggested solution is IoT-based and revolves around the bin fill-level monitoring and optimisation of routes for collecting waste. The system is based on the methodology used by Enevo / Sensoneo / Derby, the simulated ultrasonic sensors send data about fill-level, identification number, and location to the cloud-based back-end, the latter stores this data as live data, a routine for route optimisation detects the bins that exceed certain threshold of their fill and creates a route for collecting waste from those bins only, the results are

displayed in a real-time dashboard. This architectural solution directly reflects the sensing-to-routing approach responsible for the efficiency gains mentioned in the above examples.

2.5 Review of IoT Components and Enabling Technologies

Component	Options considered	Decision for prototype	Rationale
Sensing Hardware	HC-SR04	HC-SR04	Cheapest, sufficient accuracy for bin-scale fill estimation
Connectivity Protocol	HTTPS REST, MQTT, CoAP	HTTPS REST (Firebase Admin SDK)	Simplest, native Firebase compatibility
Long-range connectivity	WiFi, NB-IoT	WiFi (Simulated)	Adequate for a single collection zone
Cloud platform	Firebase Firestore, AWS IoT Core, Azure IoT Hub, ThingsBoard	Firebase Firestore	Free tier, simple Admin SDK ingestion, real-time listeners suit a live dashboard, fastest path to a working prototype

Table 1: Comparison of IoT hardware, connectivity, and cloud platform options

3 Prototype Design & Architecture

3.1 Prototype Architecture

The conceptual architecture is structured as a five-stage pipeline as illustrated in Figure 1 below:

I. The bin simulator implemented in Python simulates the ultrasonic sensors and sends data over the HTTPS
II. Firebase Firestore keeps track of live fill level and location information for each bin
III. The route optimisation module requests for information on bins that have exceeded the threshold from Firebase Firestore and creates an optimal route based on a nearest neighbour approach
IV. The live dashboard displays live data on the bin information and the route information using Firebase Firestore listeners
V. The bin collection team implements the optimal route.

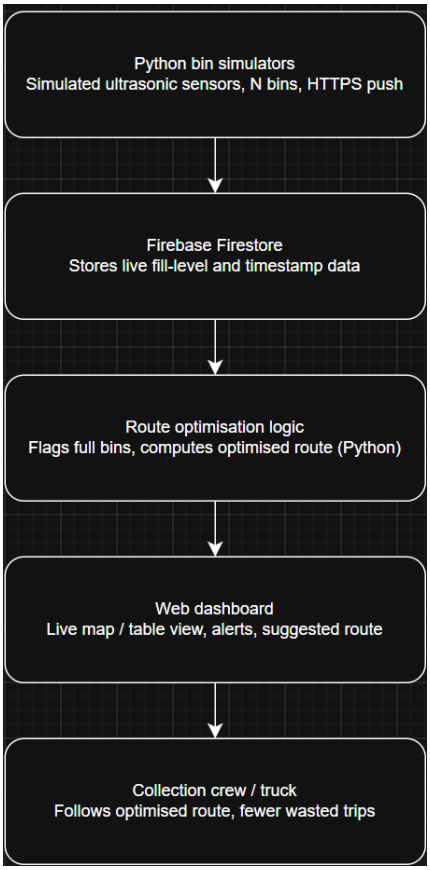


Figure 1: Layered architecture of the smart bin monitoring prototype

Python is used in the virtual sensing layer as well as the route optimisation logic implemented on the server. Firebase Firestore is chosen as a cloud storage, as opposed to AWS IoT Core and Azure IoT Hub, since Firestore supports real-time listeners, while the former two have higher implementation complexity, the user interface is built on top of HTML and JavaScript, and uses Firebase Web SDK, specifically on Snapshot listeners, which update the browser view immediately without any manual actions from the users.

3.2 Prototype Implementation

The Firebase backend was configured first, a Firestore database was created and a service account key generated to authenticate the Python simulator as an administrative client (Figure 2).

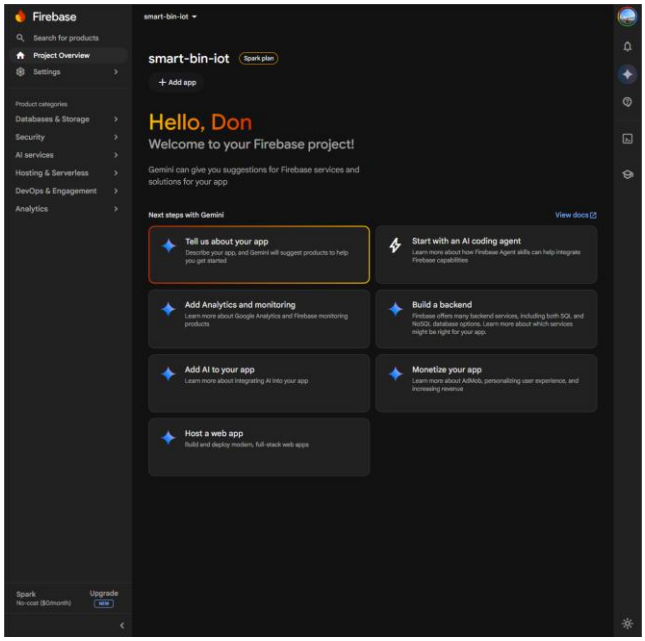


Figure 2 - Firebase project initialisation in the Google Cloud console

3.2.1 Bin Simulator (bin_simulator.py)

The most important novelty introduced in the simulator is bin_simulator.py, which models sensors in a more realistic manner without just producing random numbers. The fill level increases at a random rate per tick, with a small probability of occurrence of a large deposit of waste, which mimics to random garbage disposal, and the fill level goes back to close to zero after collection. At each tick, a packet (bin ID, fill level, coordinates, timestamp) is sent using the Firebase Admin SDK to Firestore, as both a live document and a historical one, which would be done by ESP32 devices via HTTPs.

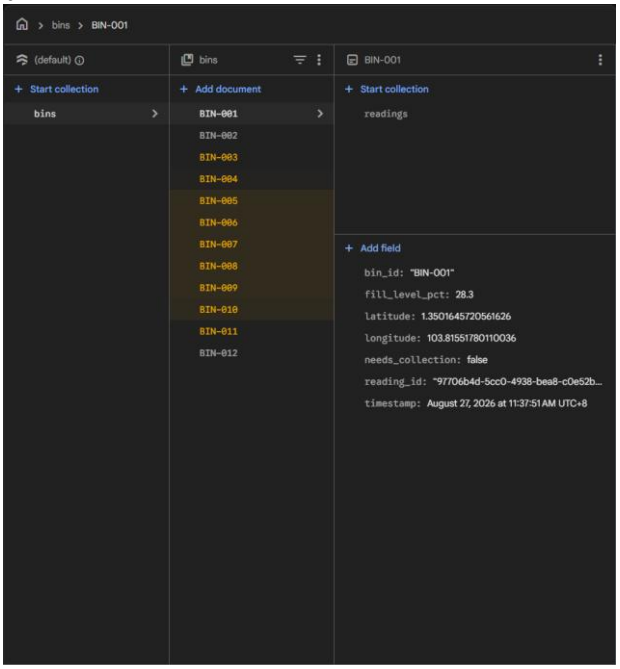


Figure 3 - Live Firestore document (BIN-001) written by the Python simulator

3.2.2 Route Optimisation Logic (route_optimizer.py)

In this module, Firestore is queried in order to discover all the bins that have been tagged as needing_collection (bins at fill level above 80%) and then a collection route is created by a greedy nearest neighbour algorithms. This procedure starts from a particular location and calculates distance between two points using haversine distance function. The reason why we chose nearest neighbour algorithm was because it is an established and justifiable algorithmic choice as opposed to solving the Travelling Salesman Problem which would be unnecessary and harder to justify in a viva situation.

3.2.3 Live Dashboard

The dashboard of the web application (Figure 4) gives a real-time operational overview based on the Firestore on Snapshot event listeners that will dynamically change whenever new data and routes are added. The individual bins are represented through a radial fill-level gauge which changes from teal to amber as it reaches the collection point limit along with the optimised route details.



Figure 4 - Live dashboard showing bin fill levels and the optimised collection route

3.3 Prototype Evaluation (evaluation_baseline.py)

In the routing evaluation scenario, the performance of the IoT-based optimisation was tested against the static schedule which reflects the conventional method outlined in all five case studies in the Background Study, visiting each and every bin regardless of its fill level. Two routing schemes, therefore, were computed on the same real-life Firestore data based on the same nearest-neighbour heuristic, making it possible to single out fill-level filtering as the only variable. The main performance criteria used in the evaluation were as follows, number

of stops made, total distance of the route (in km), and percentage reductions in both stops and distance.

Given a fleet of 12 bins partially filled (three of 12 bins being at the collection threshold of 80% fill level) the visit-all route took 12 stops in 7.269km distance while the optimised route took 3 stops in 3.197km distance (see Figure 5). It means that there were 9 stops less (i.e., a 75% decrease in collection stops) and the route distance was 4.072km shorter (or a 56% reduction in distance).

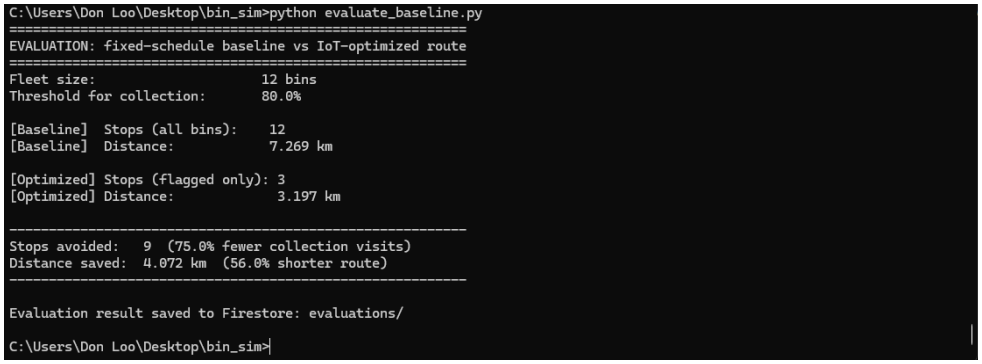


Figure 5 - Terminal output comparing the fixed-schedule baseline against the IoT-optimised route

The numbers are of the same magnitude as in the real-life situation (Derby 53%, Sensoneo 60%), implying that the prototype can tackle the problem at hand in an equally effective manner as those commercially implemented solutions can.

However, there are several limitations of this particular analysis which should be noted instead of being disregarded, similar to the limitations of the major commercial systems outlined by Brouwer et al (2023). First of all, the determination of route distance using haversine equation will inevitably result in an underestimation of the distance as in an urban environment the vehicle will take the shortest path along the street network rather than go in a straight line between the two points. Additionally, the results are derived from only one run of the software at a particular point in time and based on simulated rather than real hardware sensor data.

3.4 Security, Privacy and Trust Evaluation

Bin IT prototype being an IoT solution with sensors which are connected to the internet and a cloud database as well as a publicly exposed dashboard inherits security, privacy, and trust issues that characterize all such systems and are mapped below to OWASP (2018) IoT Top 10. A weak point of the system is the fact that the Firestore ingestion endpoint accepts any write request coming from the Admin SDK service-account key without any authentication per device, which falls under OWASP's "weak, guessable or hardcoded passwords" category (OWASP, 2018). Thus a leaking key or a spoofing sensor can send fake fill-level data and redirect trucks away from full bins. Another weakness in this regard is the fact that the dashboard reads the Firestore without access control based on the roles, making the location of the bins and their route known to any user with the project URL, which raises privacy issues because the pattern of recurring trucks give away the location of the bins. A lack of the rate limiting on the ingestion endpoint also makes it susceptible to a denial-of-service attack, in which the cloud cost is increased due to flooding the Firestore with false readings and resulting in unnecessary trips.

Some measures that may be applied are, securing the Firestore Security Rules by only allowing write requests from the authenticated service account linked to particular bin ID and limiting read actions to authenticated dashboard users, instead of using an open access strategy, using short-lived tokens for each sensor instead of a single static key, which means that one leaked credential will not allow for spoofing the entire fleet of sensors, keeping the currently used secure HTTPs transport layer, and implementing anomaly detection to spot unlikely changes between readings indicating a sensor fault rather than injection.

4. Conclusion

This report has provided IoT-system for intelligent monitoring of waste-bin fill level, which is aimed at solving the fixed schedule collection problem in five different cases found in the Background Study. Using simulation data, the prototype demonstrated a 75% reduction in stops and 56% reduction in route length compared to a fixed schedule baseline, consistent with the order of magnitude found in the commercial and municipal systems described above.

The limitation of this prototype are the usage of simulated sensors instead of real devices, straight line distance calculation instead of road network-based distance calculation, and a one-time rather than longitudinal experiment. Further work is needed to evaluate the performance under different fill-state situations and real hardware devices, as well as adding the road network routing to the prototype and implementing the above security countermeasures. Specifically, it is important to implement the improved Firestore security rule set before putting this prototype into practice.

5. List of References

BBC News Bin sensors could bring end to regular collection. Available at: web.archive.org/web/*/https://feeds.bbci.co.uk/news/articles/czr3vy2zjngo

Brouwer, Y., Barbosa-Povoa, A.P., Antunes, A.P. and Ramos, T.R.P. (2023) ‘Comparison of different waste bin monitoring approaches: An exploratory study’, Waste Management & Research.

REEN. (2023, July 11). Derby City Council to roll out waste management technology after successful trial. https://reen.com/derby-city-council-to-roll-out-waste-management-technology-after-successful-trial/

Simion, C. (2023, October 3). Council scoops award for ‘rubbish’ innovation. Marketing Derby. https://www.marketingderby.co.uk/news/council-scoops-award-for-rubbish-innovation/

European Commission, CORDIS Analytics technology for your waste containers. Available at: https://cordis.europa.eu/article/id/125211-analytics-technology-for-your-waste-containers

European Commission, CORDIS Smarter waste collection and increased sustainability. Available at: https://cordis.europa.eu/article/id/411422-smarter-waste-collection-and-increased-sustainability

European Commission, CORDIS SmartCircle project results summary. Available at: https://cordis.europa.eu/project/id/101010676/results/es

6. Appendix

Live Dashboard URL - https://smart-bin-iot-29dcf.web.app

Firebase / Firestore console link - https://console.firebase.google.com/project/smart-bin-iot-29dcf/overview

Source code file listing – bin_simulator.py, route_optimizer.py, evaluate_baseline.py, index.html