

## **IBM – NALAIYA THIRAN PROJECT**

### **SMART WASTE MANAGEMENT SYSTEM FOR METROPOLITAN CITIES**

**INDUSTRY MENTOR : DINESH**

**FACULTY MENTOR : RAMESH V**

**TEAM ID : PNT2022TMID44517**

**TEAM LEAD : GOBIKA V**

**TEAM MEMBER : GEETHANJALI S**

**TEAM MEMBER : GOPIKA K**

**TEAM MEMBER : KOWSALYA S**

## **ABSTRACT**

The paper is based on the concept of Automation used in waste management system under the domain of Climate Change and Cleanliness and Hygiene. Dumping garbage onto the streets and in public areas is a common synopsis found in all developing countries and this mainly end up affecting the environment and creating several unhygienic conditions. In order to deal with these problems Smart waste bin is an ideology put forward which is a combination of hardware and software technologies i.e. connecting Wi-Fi system to the normal dustbin in order to provide free internet facilities to the user for a particular period of time. The technology awards the user for keeping the surrounding clean and thus work hand in hand for the proper waste management in a locality. Smart waste bin uses multiple technologies firstly the technology for measuring the amount of trash dumped secondly the movement of the waste and lastly sending necessary signals and connecting the user to the Wi-Fi system. The proposed system will function on client server model, a cause that will assure clean environment, good health, and pollution free society.

## TABLE OF CONTENT

CHAPTER	CONTENTS	PAGE NO
1	<b>INTRODUCTION</b> 1.1 PROJECT OVERVIEW 1.2 PURPOSE	05
2	<b>LITERATURE SURVEY</b> 2.1 EXISTING PROBLEM 2.2 REFERENCES 2.3 PROBLEM STATEMENT DEFINITION	06
3	<b>IDEATION &amp; PROPOSED SOLUTION</b> 3.1 EMPATHY MAP CANVAS 3.2 IDEATION & BRAINSTROMING 3.3 PROPOSED SOLUTION 3.4 PROBLEM SOLUTION FIT	09
4	<b>REQUIREMENT ANALYSIS</b> 4.1 FUNCTIONAL REQUIREMENT 4.2 NON-FUNCTIONAL REQUIREMENTS	14
5	<b>PROJECT DESIGN</b> 5.1 DATA FLOW DIAGRAMS 5.2 SOLUTION & TECHNICAL ARCHITECTURE 5.3 USER STORIES	17

<b>6</b>	<b>PROJECT PLANNING &amp; SCHEDULING</b> 6.1 SPRINT PLANNING & ESTIMATION 6.2 SPRINT DELIVERY SCHEDULE 6.3 REPORTS FROM JIRA	<b>20</b>
<b>7</b>	<b>CODING &amp; SOLUTIONING</b> 7.1 FEATURE 1 7.2 FEATURE 2	<b>26</b>
<b>8</b>	<b>TESTING</b> 8.1 TEST CASES 8.2 USER ACCEPTANCE TESTING	<b>40</b>
<b>9</b>	<b>RESULTS</b> 9.1 PERFORMANCE METRICS	<b>44</b>
<b>10</b>	<b>ADVANTAGES &amp; DISADVANTAGES</b>	<b>47</b>
<b>11</b>	<b>CONCLUSION</b>	<b>48</b>
<b>12</b>	<b>FUTURE SCOPE</b>	<b>49</b>
<b>13</b>	<b>APPENDIX</b> SOURCE CODE GITHUB & PROJECT DEMO LINK	<b>50</b>

# **1. INTRODUCTION**

## **1. PROJECT OVERVIEW**

Today big cities around the world are facing a common problem, managing the city waste effectively without making city unclean. Today's waste management systems involve a large number of employees being appointed to attend a certain number of dumpsters this is done every day periodically. This leads to a very inefficient and unclean system in which some dumpsters will be overflowing some dumpsters might not be even half full. This is caused by variation in population density in the city or some other random factor this makes it impossible to determine which part needs immediate attention. Here a waste management system is introduced in which each dumpster is embedded in a monitoring system that will notify the corresponding personal if the dumpster is full. In this system, it is also possible to separate wet and dry waste into two separate containers. This system provides an effective solution to the waste management problem

## **2. PURPOSE**

Today big cities around the world are facing a common problem, managing the city waste effectively without making city unclean. Today's waste management systems involve a large number of employees being appointed to attend a certain number of dumpsters this is done every day periodically. This leads to a very inefficient and unclean system in which some dumpsters will be overflowing some dumpsters might not be even half full. This is caused by variation in population density in the city or some other random factor this makes it impossible to determine which part needs immediate attention. Here a waste management system is introduced in which each dumpster is embedded in a monitoring system that will notify the corresponding personal if the dumpster is full. In this system, it is also possible to separate wet and dry waste into two separate containers. This system provides an effective solution to the waste management problem

## 2. LITERATURE SURVEY

### 2.1 EXISTING PROBLEM:

This is not an original idea, IOT based dustbin was implemented and effectuated much before. Some authors presented systems where the sensors in the bin checked if the bin are filled up to the brim or not. If it was filled an automated message was sent to the server end of the system, through the Arduino SIM module, which used the application of the Arduino board. Once the server received the message it forwarded the message to the worker in charge, if the worker was available, he would notify his/her presence by accepting the work and would reach the required destination.

If the worker was not available, the work would be transferred to another worker. Some authors also implemented real time waste management system by using smart dustbins to check the filled level of dustbins whether they were filled. By implementing this proposed system, the cost reduction, resource optimization, effective usage of smart dustbins was carried out. This system indirectly reduced traffic in the city.

The System informed the status of each and every dust bin in real time so that the concerned authority can send the garbage collection vehicle only when the dustbin is full. This system assured the cleaning of dustbins soon when the garbage level reached its maximum. If the dustbin was not cleaned in specific time, then the records were sent to the higher authority who took appropriate action against the concerned contractor. Progressively the Dustbin with Wi-Fi Router attached in it was also introduced.

The Dustbin had a Passive Infrared Sensor. The Wi-Fi router was programmed to display the temporary connecting code. When the user threw trash in the dustbin, the PIR sensor detected the trash and sent signals to the microcontroller. The microcontroller detected the signals and forwarded it to the router device. The router verified the signals and generated random codes and then forwarded it again to the microcontroller. The microcontroller scanned the signals and forwarded it to the LCD Display. The LCD Display displayed it. The user entered the random code generated by the router on the PHP interface which was hosted on the server. The user then used the Master Wi-Fi password to connect to the internet. The user got the internet access for 10 minutes and automatically got disconnected.

- More complications in the processing.
- many controlling units linked with each other

- higher implementation cost

## 2.2 REFERENCES

1. P. Suresh, Vijay. Daniel, R.H. Aswathy, Dr. V. Parthasarathy, "A State-of-the-Art review on Internet of Things" International Conference on Science Engineering and Management Research (ICSEMR), IEEE, DOI: 10.1109/ICSEMR.2014.7043637 19 February 2015.
2. Parkash, Prabu V "IoT Based Waste Management for Smart City" International Journal of Innovative Research in Computer and Communication Engineering, Vol. 4, Issue 2, DOI:10.15680/IJIRCCE.2016. 0402029, February 2016.
3. Evaluation on the Performance of Urban Domestic Sewage Treatment Plants in China - 2011 Dongmei Han; Guojun Song
4. Teemu Nuortioa, Jari Kyto"jokib, Harri Niskaa, Olli Bra"ysyb "Improved route planning and scheduling of waste collection and transport", Expert Systems with Applications 30 (2006) 223– 232, Elsevier
5. M. Arebey, M. Hannan, H. Basri, and H. Abdullah, "Solid waste monitoring and management using RFID, GIS and GSM", The IEEE Student Conference on Research and Development (SCOREd), 16-18 November 2009, UPM Serdang, Malaysia, 2009
6. M. Hannan, M. Arebey, R. A. Begum, and H. Basri, "Radio Frequency Identification (RFID) and communication technologies for solid waste bin and truck monitoring system", Waste Management, Vol. 31, pp. 2406-2413, 2011.
7. S. Longhi, D. Marzioni, E. Alidori, G. Di Buo, M. Prist, M.Grisostomi, et al., "Solid Waste Management Architecture Using Wireless Sensor Network Technology", The 5th International Conference on New Technologies, Mobility and Security (NTMS), 7-10 May 2012, Istanbul, pp. 1-5, 2012. 147

- 8.** Waikhom Reshmi, RamKumar Sundaram, M. Rajeev Kumar, “Sensor Unit for Waste Management: A Better Method,”International conference on Science, Engineering and Management Research, ©2014 IEEE



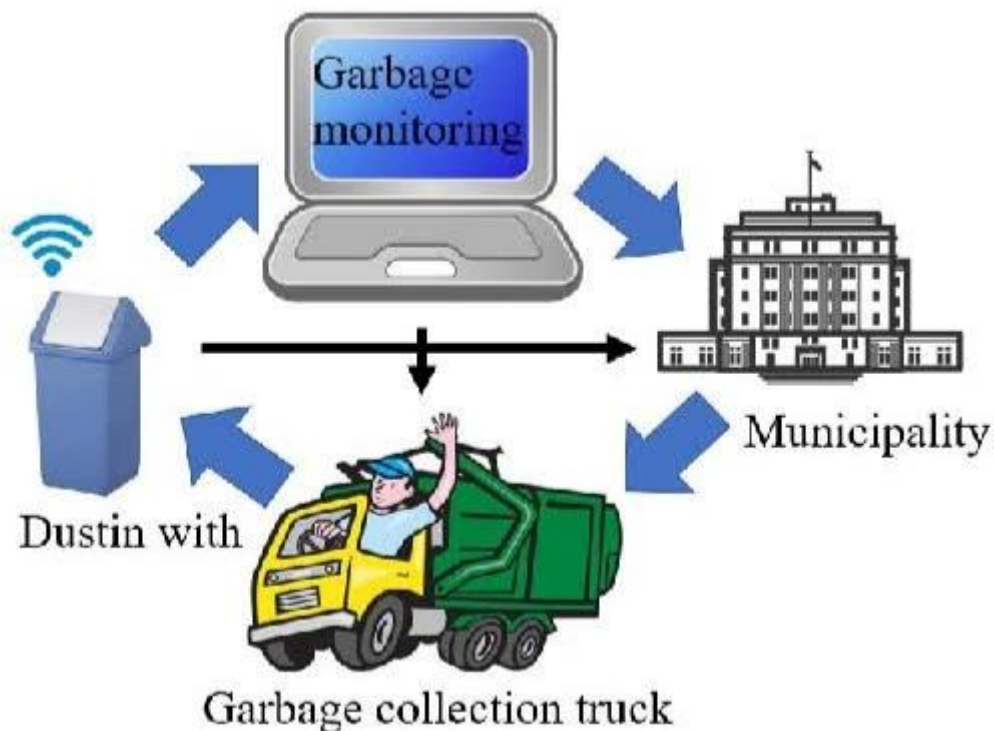
### 2.3 PROBLEM STATEMENT DEFINITION

The current process of waste management starts with the waste being created by people in the cities and disposed in trash bins near its creation point. The disposed trash is collected by municipality or private company trucks at the predefined times and transferred to temporary collection centers. The trash at the collection centers is then sent for recycling.

This process in current city setting solves the waste problem partially while it creates other problems such as;

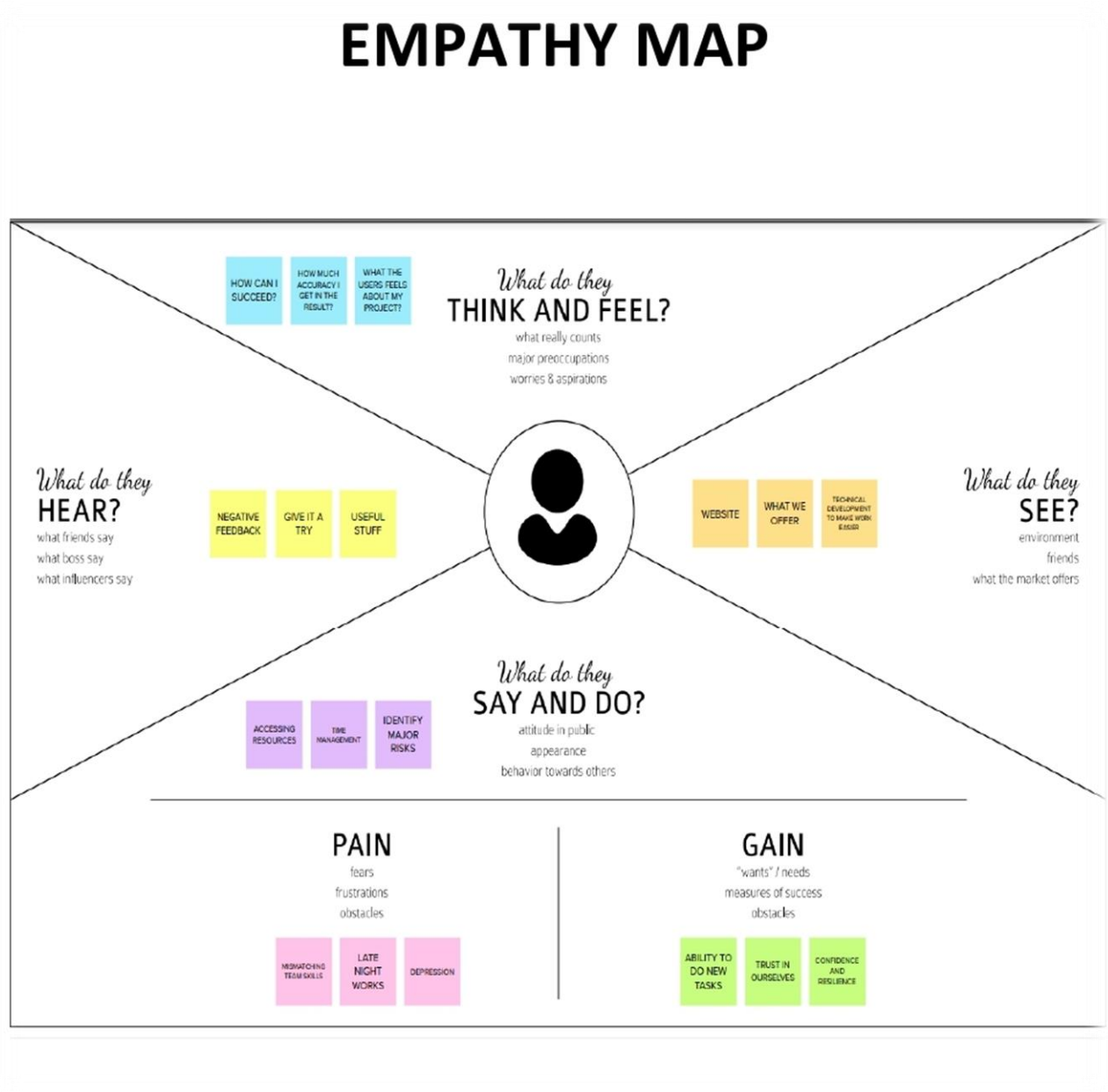
- Some trash bins are overfilled while others are underfilled by the trash collection time,
- overfilled trash bins create unhygienic conditions,
- unoptimized truck routes result in excessive fuel usage and environmental pollution and
- all collected trash is combined which complicates sorting at the recycling facility.

Some of these problems can be mitigated by implementing smart waste management systems.



### 3. IDEATION & PROPOSED SOLUTION

#### 3.1 EMPATHY MAP CANVAS:



## 3.2 IDEATION & BRAINSTORMING

### Brainstorm & idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

- 10 minutes to prepare
- 1 hour to collaborate
- 2-4 people recommended

Share template feedback

#### 1 Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

- Team getting: Define who should participate in the session and send an invite. Give relevant information as pre-work ahead.
- Get things right: Think about the problem you're focusing on solving in the brainstorming session.
- Learn how to use the facilitation tools: Use the facilitation support to see a happy and productive session.

Open guide

#### 2 Define your problem statement

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

**How Might We**

With the existing methods of collecting and disposal it is near impossible to manage such amount of waste in the future as around 30% of waste end up on the roads and public places due to ineffective disposing and collecting methods.

**Key rules of brainstorming**

To set an smooth and productive session

- Stay on topic
- Encourage wild ideas
- Defer judgement
- Let ideas flow
- Go for volume
- Quantity over quality

#### 3 Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

**Team Lead**

- Separation at source in all metropolitan and local municipalities
- Preventing ecological degradation
- Reduces energy and cost
- Creating a cleaner environment
- Share the exact location of the Bin
- Helps Council to collect Garbage before Overflowing
- Increase thermal treatment
- Protect the health and well-being of people
- Sort of recyclable materials & increasing efficiency
- Making a safer working environment for employers
- Sends SMS for municipality to empty the Bin

**M1**

**M2**

**M3**

- Reducing the need for human workers
- People have been enjoying a cleaner
- A reduction in the number of waste bins need
- Responsibility Towards environment
- Reduce time and effort in efficient manner
- Updates that either bin is full or empty
- Helpful to develop Green and Smart cities
- Reduces the requirement of man power
- Detects the current status of garbage bin

#### 4 Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

10 minutes

**Importance**

1. Rank or Rate each idea on an importance scale (1-5) or a similar scale that represents importance.

**Feasibility**

1. Rank or Rate each idea on a feasibility scale (1-5) or a similar scale that represents feasibility.

**Updates that either bin is full or empty**

**Sort of recyclable materials & increasing efficiency**

**Helpful to develop Green and Smart cities**

**A reduction in the number of waste bins need**

**Helps Council to collect Garbage before Overflowing**

**Helpful to develop Green and Smart cities**

#### 5 After you collaborate

You can export the results as an image or pdf to share with members of your company who might find it helpful.

Quick actions

- Share the result: Share a view that the result with stakeholders to keep them in the loop about the outcomes of the session.
- Export the result: Export a copy of the result as a PNG or PDF to share with stakeholders to keep them in the loop about the outcomes of the session.

Keep moving forward

- Strategy to improve: Define the core aspects of a new idea or strategy.
- Open for feedback
- Customize experience: See how you can help. Understand customer needs, behaviors, and attitudes to focus on improvement.
- Open for feedback
- Strengthen, weaknesses, opportunities, threats: Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
- Open for feedback

Share template feedback

Need some inspiration?

Check out the ideas from other users who have used this template.

Get inspired

Need some inspiration?

Check out the ideas from other users who have used this template.

Get inspired

Need some inspiration?

Check out the ideas from other users who have used this template.

Get inspired

Need some inspiration?

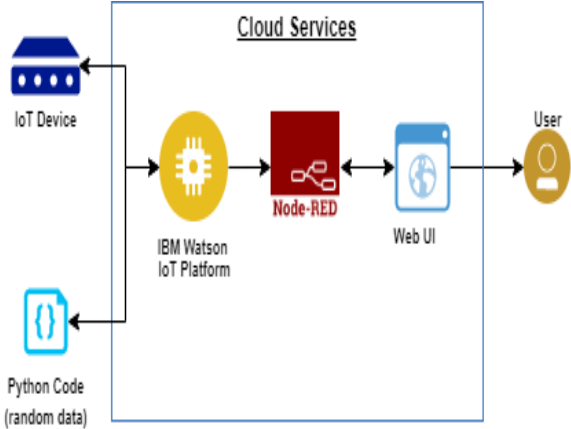
Check out the ideas from other users who have used this template.

Get inspired

### 3.3 PROPOSED SOLUTION

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement( Problem to be solved)	As the population is growing, the garbage is also increasing. This huge unmanaged accumulation of garbage is polluting the environment, spoiling the beauty of the area and also leading to the health hazard to overcome these, smart waste management for metropolitan cities are introduced.
2.	Idea/Solution description	<ul style="list-style-type: none"> <li>➤ Garbage level detection in bins.</li> <li>➤ Getting the weight of the garbage in the bin.</li> <li>➤ Alerts the authorized person to empty the bin whenever the bins are full.</li> <li>➤ Garbage level of the bins can be monitored through a web App.</li> <li>➤ We can view the location of bins in the web application by sending GPS location from the device.</li> </ul>
3.	Novelty/ Uniqueness	A system that detects the level of garbage in the dustbins with the help of sensor systems and send this information to the authorized control room. Weight sensor determines the weight of the garbage in the dustbin and Infrared (IR) sensor is used to detect the waste level in the dustbins.

4.	Social Impact/ Customer Satisfaction	Municipality authority is the customer whose work will get easy and they get satisfied if they know to use this system the garbage collector can complete their work within the time without leaving any place. The workload for the garbage collector and authority may get reduced 75%
5.	Business Model (Revenue Model)	<p>The cost for the truck fuel get reduced, man power reduced so, wages for them reduced but the process is cost efficient for implementation and everyone should get training for better performance.</p>  <pre> graph LR     IoT[IoT Device] --&gt; Watson[IBM Watson IoT Platform]     Watson --&gt; NodeRED[Node-RED]     NodeRED --&gt; WebUI[Web UI]     WebUI --&gt; User((User))     Python[Python Code (random data)] --&gt; Watson     subgraph Cloud_Services [Cloud Services]         Watson         NodeRED         WebUI     end </pre>
6.	Scalability of the Solution	The process can also denote the climate change in the area, what is the average time the bin takes to fill, denoting the weight of liquid and solid separately, percentage of biodegradable and non-biodegradable waste.

# Consumer Trends Canvas

1

**Share your feedback**



## 4. REQUIREMENT ANALYSIS

### 4.1 FUNCTIONAL REQUIREMENT

Following are the functional requirements of the proposed solution.

<b>FR No.</b>	<b>Functional Requirement (Epic)</b>	<b>Sub Requirement (Story / Sub-Task)</b>
FR-1	<b>User Registration</b>	Registration through Gmail
FR-2	<b>User Confirmation</b>	Confirmation via Email Confirmation via OTP
FR-3	<b>GPS Access</b>	GPS access to know the location
FR-4	<b>Bin level Analysing</b>	Acquire the levels of Waste bins in a regular interval of time.
FR-5	<b>Transport Router</b>	To make a efficient route for the collection of garbages around a area.

## 4.2 NON-FUNCTIONAL REQUIREMENTS

Following are the non-functional requirements of the proposed solution.

<b>NFR No.</b>	<b>Non-Functional Requirement</b>	<b>Description</b>
NFR-1	<b>Usability</b>	<ul style="list-style-type: none"> <li>• A smart solution has been proposed to make the waste sorting more simple and accurate , and improve the user experience, usability, and satisfaction.</li> <li>• It aims to optimize ease of use while offering maximum functionality.</li> </ul>
NFR-2	<b>Security</b>	<ul style="list-style-type: none"> <li>• The information of the users will be highly secured,the accounts are verified with Gmail.</li> <li>• If the products are misplaced then the GPS driven sensor gives an alert.</li> </ul>
NFR-3	<b>Reliability</b>	<ul style="list-style-type: none"> <li>• Operates in a defined environment without failure resulting in less manpower, emissions, fuel use and traffic congestion.</li> </ul>

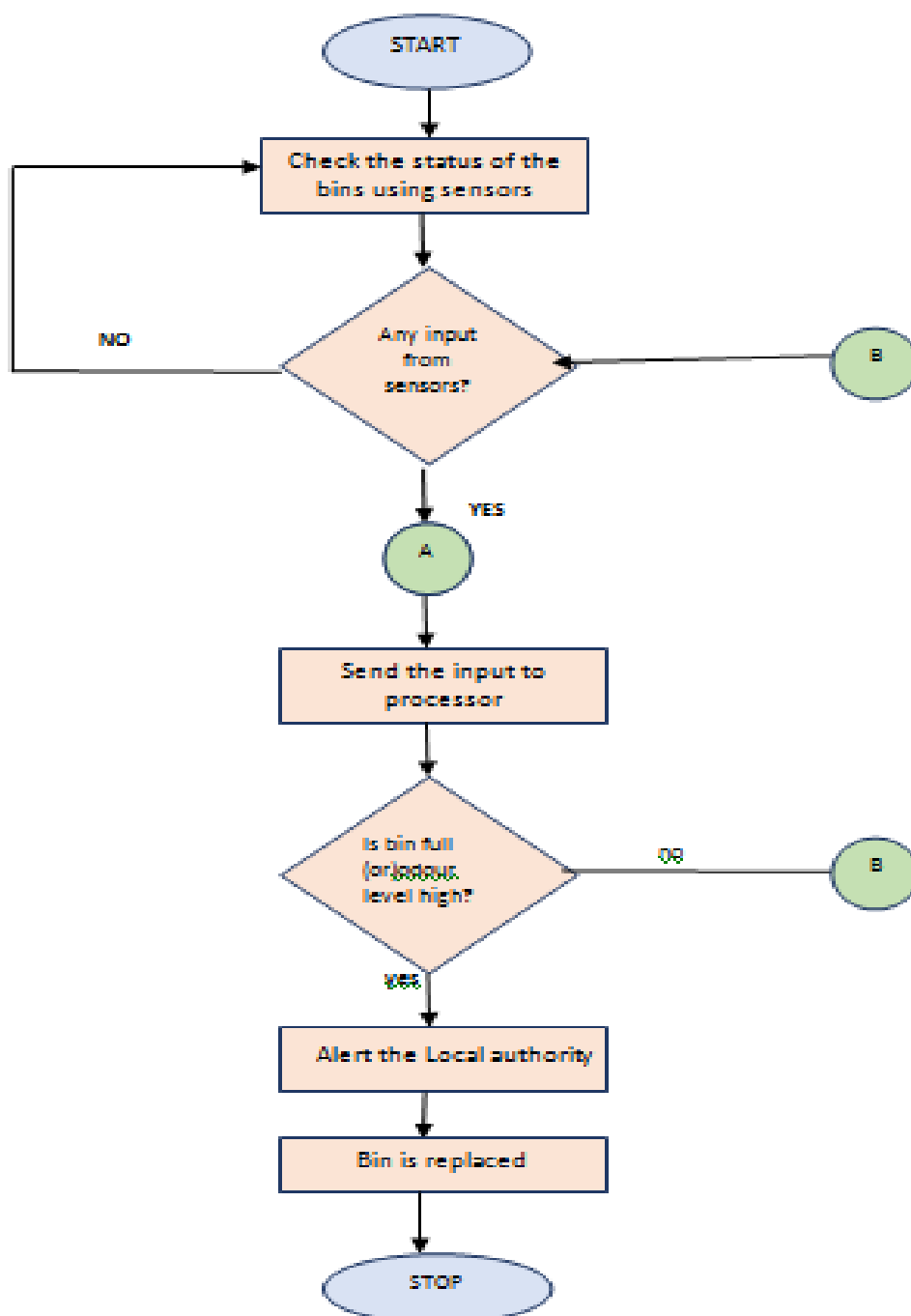


NFR-4	<b>Performance</b>	<ul style="list-style-type: none"> <li>• The system will provide accurate reports, thus increasing the efficiency of the system.</li> <li>• The real-time monitoring of the garbage level with the help of sensors and wireless communication will reduce the total number of trips required of Garbage collecting truck.</li> <li>• This will reduce the total expenditure associated with the garbage collection.</li> </ul>
NFR-5	<b>Availability</b>	<ul style="list-style-type: none"> <li>• The smart waste bins are available in Convention centers, buildings, stadiums, and transportation facilities and captures high-quality waste data and informs staff when it gets full.</li> </ul>
NFR-6	<b>Scalability</b>	<ul style="list-style-type: none"> <li>• A versatile scalable smart waste-bin system based on limited waste management could potentially lead to great improvements.</li> <li>• Once these smart bins are implemented on a large scale by replacing the traditional bins, the waste can be quickly managed to its efficient level as it avoids unnecessary lumping of wastes on roadside.</li> </ul>

## 5.PROJECT DESIGN

### 5.1 DATA FLOW DIAGRAMS

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



## 5.2 SOLUTION & TECHNICAL ARCHITECTURE:

- The general architecture which uses the cluster of smart waste bins connected through IoT in the out door environment. It uses a GPS to communicate the status of the smart waste bins enabling the effective waste management system.
- The proposed system could be considered as a robotic smart waste bin where the bins could mobilize, localize its location and communicate its status to the cloud.
- The smartness is achieved by having ultrasonic sensors, Node MCU, Capacitive sensors, servomotors, microcontroller integrated to form an autonomous system.

### Solution Architecture Diagram:



### 5.3 USER STORIES

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Login	USN-1	As a Administrator, I need to give user id and passcode for ever workers over there in municipality	10	High	Gobika
Sprint-1	Login	USN-2	As a Co-Admin, I'll control the waste level by monitoring them vai real time web portal. Once the filling happens, I'll notify trash truck with location of bin with bin ID	10	High	Gobika
Sprint-2	Dashboard	USN-3	As a Truck Driver, I'll follow Co-Admin's Instruction to reach the filling bin in short roots and save time	20	Low	Geethanjali
Sprint-3	Dashboard	USN-4	As a Local Garbage Collector, I'll gather all the waste from the garbage, load it onto a garbage truck, and deliver it to Landfills	20	Medium	Gopika
Sprint-4	Dashboard	USN-5	As a Municipality officer, I'll make sure everything is proceeding as planned and without any problems	20	High	Kowsalya

## 6. PROJECT PLANNING & SCHEDULE

### 6.1 SPRINT PLANNING & ESTIMATION:

TITLE	DESCRIPTION	DATE
<b>Literature Survey &amp; Information Gathering</b>	Literature survey on the selected project & gathering information by referring the, technical papers, research publications etc.	07 October 2022
<b>Prepare Empathy Map</b>	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	07 October 2022
<b>Ideation</b>	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	07 October 2022
<b>Proposed Solution</b>	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	11 October 2022
<b>Problem Solution Fit</b>	Prepare problem - solution fit document.	11 October 2022
<b>Solution Architecture</b>	Prepare solution architecture document.	11 October 2022

<b>Customer Journey</b>	Prepare the customer journey maps to understand the user interactions & experiences with the application (entry to exit).	11 October 2022
<b>Functional Requirement</b>	Prepare the functional requirement document.	11 October 2022
<b>Data Flow Diagrams</b>	Draw the data flow diagrams and submit for review.	08 NOVEMBER 2022
<b>Technology Architecture</b>	Prepare the technology architecture diagram.	08 NOVEMBER 2022
<b>Prepare Milestone &amp; ActivityList</b>	Prepare the milestones & activity list of the project.	08 NOVEMBER 2022
<b>Project Development - Delivery of Sprint-1, 2, 3 &amp; 4</b>	Develop & submit the developed code by testing it.	08 NOVEMBER 2022

## 6.2 SPRINT DELIVERY SCHEDULE:

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Login	USN-1	As a Administrator, I need to give user id and passcode for ever workers over there in municipality	10	High	Gobika Geethanjali Gopika Kowsalya
Sprint-1	Login	USN-2	As a Co-Admin, I'll control the waste level by monitoring them vai real time web portal. Once the filling happens, I'll notify trash truck with location of bin with bin ID	10	High	Gobika Geethanjali Gopika Kowsalya
Sprint-2	Dashboard	USN-3	As a Truck Driver, I'll follow Co-Admin's Instruction to reach the filling bin in short roots and save time	20	Low	Gobika Geethanjali Gopika Kowsalya
Sprint-3	Dashboard	USN-4	As a Local Garbage Collector, I'll gather all the waste from the garbage, load it onto a garbage truck, and deliver it to Landfills	20	Medium	Gobika Geethanjali Gopika Kowsalya
Sprint-4	Dashboard	USN-5	As a Municipality officer, I'll make sure everything is proceeding as planned and without any problems	20	High	Gobika Geethanjali Gopika Kowsalya

<b>Sprint</b>	<b>Total Story Points</b>	<b>Duration</b>	<b>Sprint Start Date</b>	<b>Sprint End Date (Planned)</b>	<b>Story Points Completed (as on Planned End Date)</b>	<b>Sprint Release Date (Actual)</b>
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

**Velocity:**

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day).

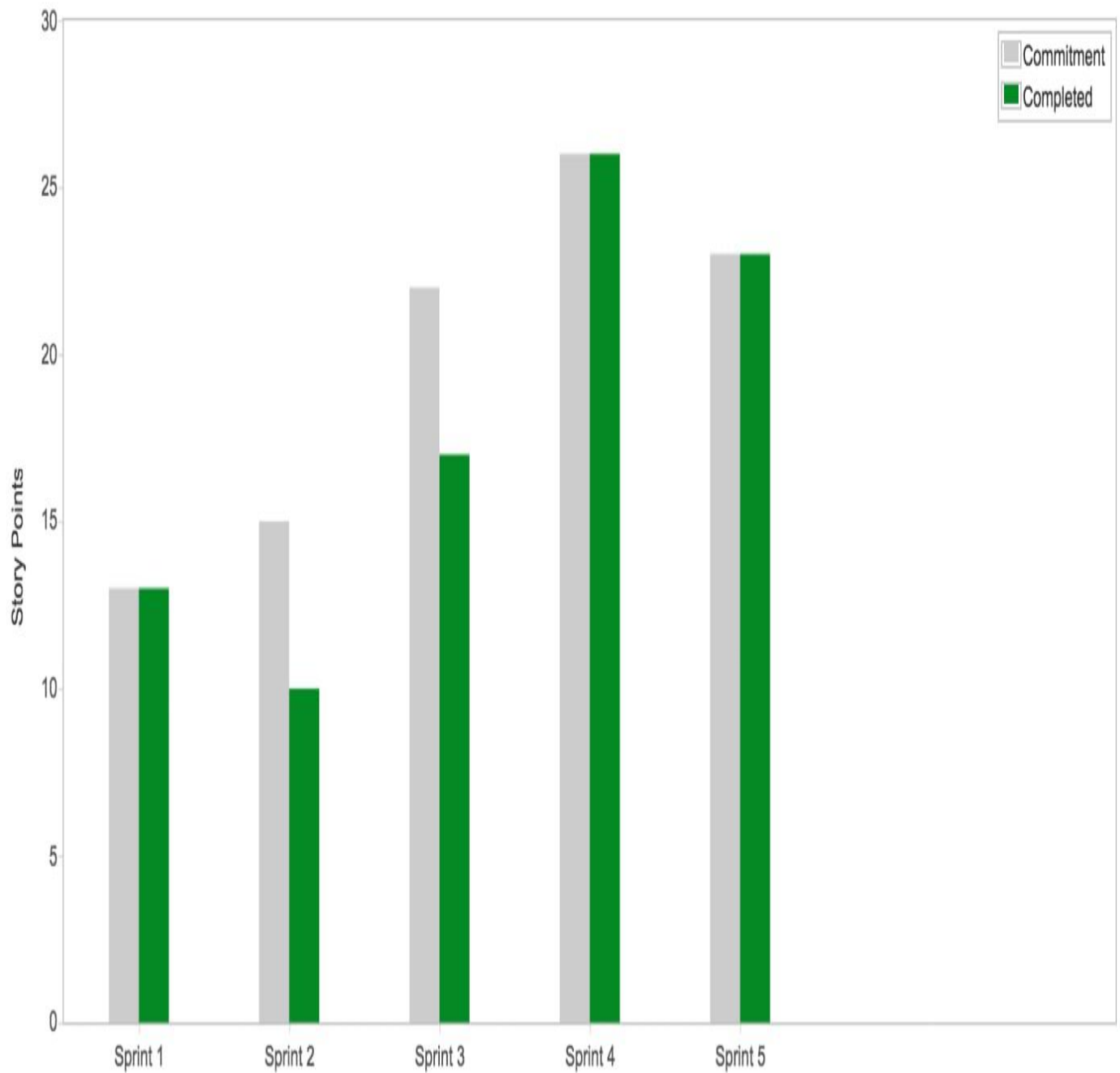
$$AV = \frac{\text{sprint duration}}{\text{velocity}} = \frac{20}{10} = 2$$



## 6.3 REPORTS FROM JIRA:

### VELOCITY CHART

Velocity Chart



JIRA DASH BOARD



Marketing Assignments

Project or Saved Filter:

Marketing

Search

Project or saved filter to use as the basis for the graph.

Advanced Search

Statistic Type:

Assignee

Select which type of statistic to display for this filter.

Auto refresh

☐ Update every 15 minutes

Save

Cancel

2 minutes ago

Add a Gadget

Search gadgets

All 52

Jira 50

Charts 12

Wallboard 5

Jira Service Management 1

### Pie Chart

By Atlassian

Add

Displays the matching issues for a project or filter as a pie chart.

Jira Charts

### Recently Created Chart

By Atlassian

Add

Displays recently created issues for a specified project as a bar chart

Jira Charts

### Resolution Time

By Atlassian

Add

Displays a bar graph of elapsed time to resolve issues for a project or filter.

Jira Charts

## 7.CODING & SOLUTIONING

### 7.1 FEATURE-1:

```
import requests
import json
import ibmiotf.application
import ibmiotf.device
import time
import random
import sys

# watson device details

organization = "4yi0vc"
devicType = "BIN1"
deviceId = "BIN1ID"
authMethod= "token"
authToken= "123456789"

#generate random values for randomo variables (temperature&humidity)

def myCommandCallback(cmd):
    global a
    print("command recieved:%s" %cmd.data['command'])
    control=cmd.data['command']
    print(control)

try:
    deviceOptions={"org":organization,"type":devicType,"id":deviceId,"auth-
method":authMethod,"auth-token":authToken}
    deviceCli = ibmiotf.device.Client(deviceOptions)
except Exception as e:
    print("caught exception connecting device %s" %str(e))
    sys.exit()
```

#connect and send a datapoint "temp" with value integer value into the cloud as a type of event for every 10 seconds

```
deviceCli.connect()
```

```
while True:
```

```
    distance= random.randint(10,70)
```

```
    loadcell= random.randint(5,15)
```

```
    data= {'dist':distance,'load':loadcell}
```

```
    if loadcell < 13 and loadcell > 15:
```

```
        load = "90 %"
```

```
    elif loadcell < 8 and loadcell > 12:
```

```
        load = "60 %"
```

```
    elif loadcell < 4 and loadcell > 7:
```

```
        load = "40 %"
```

```
    else:
```

```
        load = "0 %"
```

```
    if distance < 15:
```

```
        dist = 'Risk warning:' 'Dumpster poundage getting high, Time to collect :) 90 %'
```

```
    elif distance < 40 and distance >16:
```

```
        dist = 'Risk warning:' 'dumpster is above 60%'
```

```
    elif distance < 60 and distance > 41:
```

```
        dist = 'Risk warning:' '40 %'
```

```
    else:
```

```
        dist = 'Risk warning:' '17 %'
```

```
    if load == "90 %" or distance == "90 %":
```

```
        warn = 'alert : ' ' Dumpster poundage getting high, Time to collect :)'
```

```

elif load == "60 %" or distance == "60 %":

    warn = 'alert :' 'dumpster is above 60%'
else :
    warn = 'alert :' 'No need to collect right now '
def myOnPublishCallback(lat=10.678991,long=78.177731):
    print("Gandigramam, Karur")
    print("published distance = %s " %distance,"loadcell:%s " %loadcell,"lon = %s " %long,"lat =
%s" %lat)
    print(load)
    print(dist)
    print(warn)

time.sleep(10)

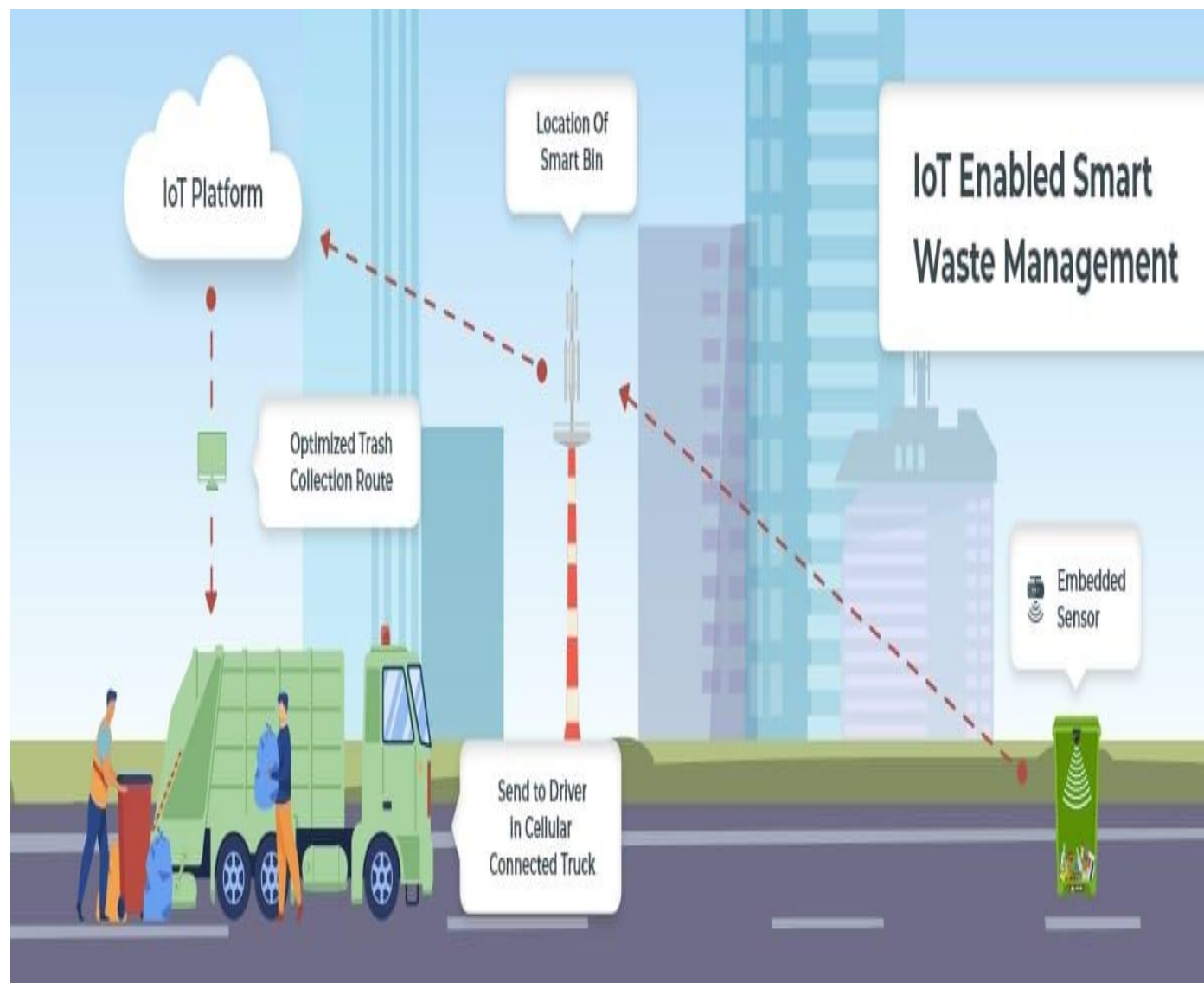
success=deviceCli.publishEvent("IoTSensor","json",warn,qos=0,on_publish=
myOnPublishCallback)

success=deviceCli.publishEvent("IoTSensor","json",data,qos=0,on_publish=
myOnPublishCallback)

if not success:
    print("not connected to ibmiot")
    time.sleep(30)

deviceCli.commandCallback=myCommandCallback
#disconnect the device
deviceCli.disconnect()

```



## 7.2 FEATURE-2:

### HTML CODE :

```
<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Smart Waste Management System</title>

<!-- Bootstrap 4 CSS CDN -->

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-
bootstrap/4.5.2/css/bootstrap.min.css" />

<!-- Fontawesome CSS CDN -->

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.14.0/css/all.min.css"
/>
<link rel="stylesheet" href="css/style.css" />

</head>

<body class="bg-info">
<div class="container">

<!-- Login Form Start -->

<div class="row justify-content-center wrapper" id="login-box">

<div class="col-lg-10 my-auto myShadow">

<div class="row">

<div class="col-lg-7 bg-white p-4">

<h1 class="text-center font-weight-bold text-primary">Sign in</h1>

<hr class="my-3" />

<form action="#" method="post" class="px-3" id="login-form">
```

```
<div class="input-group input-group-lg form-group">

<div class="input-group-prepend">
<span class="input-group-text rounded-0"><i class="far fa-envelope fa-lg fa-fw"></i></span>
</div>

<input type="email" id="email" name="email" class="form-control rounded-0" placeholder="E-
Mail" required />
</div>

<div class="input-group input-group-lg form-group">

<div class="input-group-prepend">
<span class="input-group-text rounded-0"><i class="fas fa-key fa-lg fa-fw"></i></spa

</div>

  <input type="password" id="password" name="password" class="form-control rounded-0"
minlength="5"placeholder="Password" required autocomplete="off" />
</div>

<div class="form-group clearfix">

<div class="custom-control custom-checkbox float-left">
<input type="checkbox" class="custom-control-input" id="customCheck" name="rem" />

<label class="custom-control-label" for="customCheck">Remember me</label>
</div>

<div class="forgot float-right">

<a href="#" id="forgot-link">Forgot Password?</a>
</div>

</div>

<div class="form-group">

<input type="submit" id="login-btn" value="Sign In" class="btn btn-primary btn-lg btn-block
myBtn" />

</div>

</form>
```



```
</div>
```

```
<div class="col-lg-5 d-flex flex-column justify-content-center myColor p-4">
```

```
<h1 class="text-center font-weight-bold text-white">Welcome Friend!</h1>
```

```
<hr class="my-3 bg-light myHr" />
```

```
<p class="text-center font-weight-bolder text-light lead">Start your initiative to make your  
environmentclean</p>
```

```
<button class="btn btn-outline-light btn-lg align-self-center font-weight-bolder mt-4 myLinkBtn"  
id="register-link">Sign Up</button>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<!-- Login Form End -->
```

```
<!-- Registration Form Start -->
```

```
<div class="row justify-content-center wrapper" id="register-box" style="display: none;">
```

```
<div class="col-lg-10 my-auto myShadow">
```

```
<div class="row">
```

```
<div class="col-lg-5 d-flex flex-column justify-content-center myColor p-4">
```

```
<h1 class="text-center font-weight-bold text-white">Welcome Back!</h1>
```

```
<hr class="my-4 bg-light myHr" />
```

```
<p class="text-center font-weight-bolder text-light lead">To stay connected Please login with  
your personalinfo.</p>
```

```
<button class="btn btn-outline-light btn-lg font-weight-bolder mt-4 align-self-center myLinkBtn"  
id="login-link">Sign In</button>
```

```
</div>
```

```
<div class="col-lg-7 bg-white p-4">
```

```
<h1 class="text-center font-weight-bold text-primary">Create Account</h1>
```

```
<hr class="my-3" />
```

```
<form action="#" method="post" class="px-3" id="register-form">
```

```
<div class="input-group input-group-lg form-group">
```

```
<div class="input-group-prepend">
```

```
<span class="input-group-text rounded-0"><i class="far fa-user fa-lg fa-fw"></i></span>
```

```
</div>
```

```
<input type="text" id="name" name="name" class="form-control rounded-0" placeholder="Full  
Name" required />
```

```
</div>
```

```
<div class="input-group input-group-lg form-group">
```

```
<div class="input-group-prepend">
```

```
<span class="input-group-text rounded-0"><i class="far fa-envelope fa-lg fa-fw"></i></span>
```

```
</div>
```

```
<input type="email" id="remail" name="email" class="form-control rounded-0" placeholder="E-  
Mail" required />
```

```
</div>
```

```
<div class="input-group input-group-lg form-group">
```

```
<div class="input-group-prepend">
```

```
<span class="input-group-text rounded-0"><i class="fas fa-key fa-lg fa-fw"></i></span>
```

```
</div>
```

```
<input type="password" id="rpassword" name="password" class="form-control rounded-0"  
minlength="5" placeholder="Password" required />
```

```
</div>
```

```
<div class="input-group input-group-lg form-group">
```

```
<div class="input-group-prepend">
```

```
<span class="input-group-text rounded-0"><i class="fas fa-key fa-lg fa-fw"></i></span>
```

```
</div>
```

```
<input type="password" id="cpassword" name="cpassword" class="form-control rounded-0" minlength="5" placeholder="Confirm Password" required />
```

```
</div>
```

```
<div class="form-group">
```

```
<div id="passError" class="text-danger font-weight-bolder"></div>
```

```
</div>
```

```
<div class="form-group">
```

```
<input type="submit" id="register-btn" value="Sign Up" class="btn btn-primary btn-lg btn-block myBtn" />
```

```
</div>
```

```
</form>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<!-- Registration Form End -->
```

```
<!-- Forgot Password Form Start -->
```

```
<div class="row justify-content-center wrapper" id="forgot-box" style="display: none;">
```

```
<div class="col-lg-10 my-auto myShadow">
```

```
<div class="row">
```

```
<div class="col-lg-7 bg-white p-4">
```

```
<h1 class="text-center font-weight-bold text-primary">Forgot Your Password?</h1>
```

```
<hr class="my-3" />
```

```
<p class="lead text-center text-secondary">To reset your password, enter the registered e-
```

mail address and we will send you password reset instructions on your e-mail!</p>

```
<form action="#" method="post" class="px-3" id="forgot-form">
```

```
<div id="forgotAlert"></div>
```

```
<div class="input-group input-group-lg form-group">
```

```
<div class="input-group-prepend">
```

```
<span class="input-group-text rounded-0"><i class="far fa-envelope fa-lg"></i></span>
```

```
</div>
```

```
<input type="email" id="femail" name="email" class="form-control rounded-0"
```

```
placeholder="E-Mail" required />
```

```
</div>
```

```
<div class="form-group">
```

```
<input type="submit" id="forgot-btn" value="Reset Password" class="btn btn-primary btn-lg  
btn-block myBtn" />
```

```
</div>
```

```
</form>
```

```
</div>
```

```
<div class="col-lg-5 d-flex flex-column justify-content-center myColor p-4">
```

```
<h1 class="text-center font-weight-bold text-white">Reset Password!</h1>
```

```
<hr class="my-4 bg-light myHr" />
```

```
<button class="btn btn-outline-light btn-lg font-weight-bolder myLinkBtn align-self-center" id="back  
link">Back</button>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<!-- Forgot Password Form End -->
```

</div>

<!-- jQuery CDN -->

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

<script src="js/script.js"></script>

</body>

</html>

### **CSS CODE :**

@import

url("https://fonts.googleapis.com/css?family=Maven+Pro:400,500,600,700,800,900&display=swap");

\* {

margin:

0;

padding: 0;

box-sizing: border-box;

font-family: "Maven Pro", sans-serif;

}

.wrapper

{ height: 100vh;}

.myColor

{

background-image: linear-gradient(to right, #324bf3 50%, #f9d423 150%);

}

.myShadow {

box-shadow: 0 10px 10px rgba(0, 0, 0, 0.5);

```
}  
  
.myBtn {  
  border-  
  radius:  
  50px; font-  
  weight:  
  bold;font-  
  size:  
  20px;  
  
  background-image: linear-gradient(to right, #0acffe  
  0%,#495aff 100%);border: none;  
}  
  
.myBtn:hover {  
  background-image: linear-gradient(to right, #495aff 0%, #0acffe 100%);  
}  
  
.myHr {  
  height:  
  2px;  
  border-radius: 100px;  
}  
  
.myLinkBtn  
{ border-  
  radius:  
  100px;width:  
  50%;  
  border: 2px solid #fff;  
}  
  
@media (max-width: 720px) {  
  .wrapper {  
    margin: 2px;  
  }  
}
```

**JS CODE:**

```
$(function () {  
  
    $("#register-link").click(function () {  
  
        $("#login-box").hide();  
  
        $("#register-box").show();  
  
    });  
  
    $("#login-link").click(function () {  
  
        $("#login-box").show();  
  
        $("#register-box").hide();  
  
    });  
  
    $("#forgot-link").click(function () {  
        $("#login-box").hide();  
        $("#forgot-box").show();  
  
    });  
  
    $("#back-link").click(function () {  
  
        $("#login-box").show();  
  
        $("#forgot-box").hide();  
  
    });  
});
```

The screenshot shows a web browser window with the address bar displaying 'codepad.co/playground'. The page features a 'Sign in' form with the following elements:

- Sign in** (h2)
- ☐ E-Mail
- Password
- ☐ Remember me
- [Forgot Password?](#)
- 

Below the form, there is a 'Welcome Friend!' section with the text 'Start your initiative to make your environment clean' and a  button.

The left sidebar shows the code editor with HTML, CSS, and JavaScript code. The HTML code includes a 'Forgot Password' form and a 'Sign in' button. The CSS code defines the layout and styling of the form. The JavaScript code handles the form submission and shows/hides the form.

The bottom status bar shows the temperature as 25°C Cloudy, the time as 07:40 PM, and the date as 30-10-2022.

The screenshot shows a web browser window with the address bar displaying 'codepad.co/playground'. The page features a 'Welcome Back!' form with the following elements:

- Welcome Back!** (h2)
- To stay connected Please login with your personal info.
- 

Below the form, there is a 'Create Account' section with the following elements:

- 
- 
- 
- 
- 

The left sidebar shows the code editor with HTML, CSS, and JavaScript code. The HTML code includes a 'Welcome Back!' form and a 'Create Account' form. The CSS code defines the layout and styling of the form. The JavaScript code handles the form submission and shows/hides the form.

The bottom status bar shows the temperature as 25°C Cloudy, the time as 07:41 PM, and the date as 30-10-2022.



## 8. TESTING

### 8.1 TEST CASES

Table 1 Testcase scenarios of Arduino for developing Smart Garbage Bin

Test case description	Testcase notation	Input	Requirements	Testcase status
Sends an alert message and displays on the web browser monitoring page as garbage bin found to be 'EMPTY.'	T <sub>1</sub>	Null	Garbage bin should not have waste in it	Pass
Sends an alert message and displays on the web browser monitoring page as garbage bin found to be 'MEDIUM.'	T <sub>2</sub>	Garbage filling	Garbage bin should be filled to its intermediate level	Pass
Sends an alert message and displays on the web browser monitoring page as garbage bin found to be 'NEARLY FULL.'	T <sub>3</sub>	Garbage filling	Garbage bin should be filled to an above intermediate level	Pass
Sends an alert message and displays on the web browser monitoring page as garbage bin found to be 'FULL.'	T <sub>4</sub>	Filled	Garbage bin should be filled to its maximum level	Pass
Sends an alert message and displays on the web browser monitoring page as garbage bin found to be 'THRESHOLD CROSSED'	T <sub>5</sub>	Spillover	Garbage bin should be filled to a level that crosses the threshold limit	Pass

Activate Windows  
Go to PC settings to activate Windows

Table 2 Testcase scenarios of Smart Garbage Bin web-based application.

Test case description	Required input	Information and related requirements	Test case status indicating pass or fail
The user or concerned service provider should register with the required details	User input details for registration	User Name, Email ID, Phone Number, and Security Password	Pass
The user or concerned service provider tried to log in to the monitoring portal with registered details	User login details	User Name, Security Password	Pass or Fail
Monitoring website portal indicating home, user, SGB status	User monitoring home screen should be display	The developed prototype for Smart Garbage Bin must be kept 'ON.'	Pass <small>Activate Windows Go to PC settings to activate</small>

Table 3 Experimental validation of Smart Garbage Bin test cases



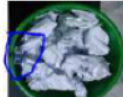

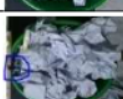
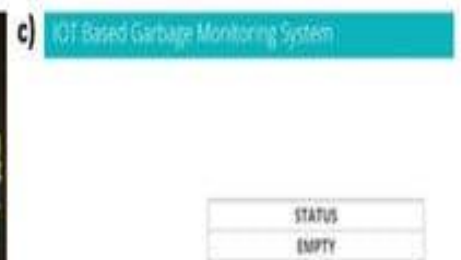
Testcase notation	Input	Input experimental visuals	Remarks	Testcase validation
T <sub>1</sub>	Null		Garbage bin does not have waste in it	Pass
T <sub>2</sub>	Garbage filling		The garbage bin is filled to its intermediate level	Pass
T <sub>3</sub>	Garbage filling		The garbage bin is filled to an above intermediate level	Pass
T <sub>4</sub>	Filled		The garbage bin is filled to its maximum level	Pass
T <sub>5</sub>	Spillover		The garbage bin is filled to a level that crosses the threshold limit	Pass

Table 4 Smart Garbage Bin status identification and experimental results

Test cases	Bin condition	LCD	Buzzer	Web portal status	Mobile SMS sent and received
Empty	Filling	Sending SMS... EMPTY	No	EMPTY	Yes
Medium	Filling	Sending SMS... MEDIUM	No	MEDIUM	Yes
Nearly full	Filling	Sending SMS... NEARLY FULL	Yes	NEARLY FULL	Yes
Full	Filled	Sending SMS... FULL	Yes	FULL	Yes
Threshold Crossed	Spil over	Sending SMS... THRESHOLD CROSSED	Yes	THRESHOLD CROSSED	Yes



## 8.2 USER ACCEPTANCE TESTING

### 1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the web UI which provides “Clean and green environment“ at the time of the release to User Acceptance Testing (UAT).

### 2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	8	3	2	1	14
Duplicate	2	0	3	0	5
External	2	1	0	1	4
Fixed	9	2	4	8	23
Not Reproduced	0	0	0	0	0
Skipped	0	0	1	1	2
Won't Fix	0	2	1	1	4
Totals	21	8	11	12	52

### 3. Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	12	0	0	12
Client Application	30	0	0	30
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	7	0	0	7
Final Report Output	4	0	0	4
Version Control	3	0	0	3

## 9.RESULTS

### 9.1 PERFORMANCE METRICS

The result of the constructed detection system and simulations of the entire system are detailed with the aid of diagrams as found in figure 7 – 12. A table indicating the tests run on both existing and proposed systems is also presented under this section.

#### A. Smart Waste Bin

The final system implementation on the smart bin can be seen in figure 7a and b. The detection system is mounted on the lid of the bin with the ultrasonic sensor located beneath the bin lid to have a line of sight detection of the garbage level.

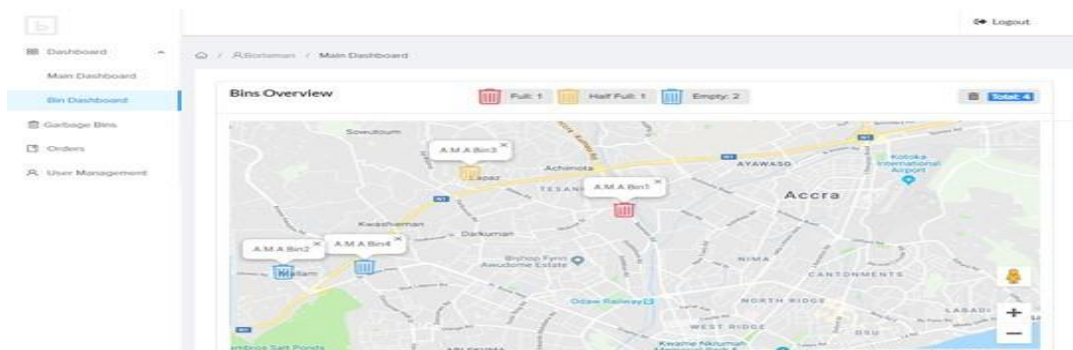


Smart Waste Bin

Garbage Level  
Detection System

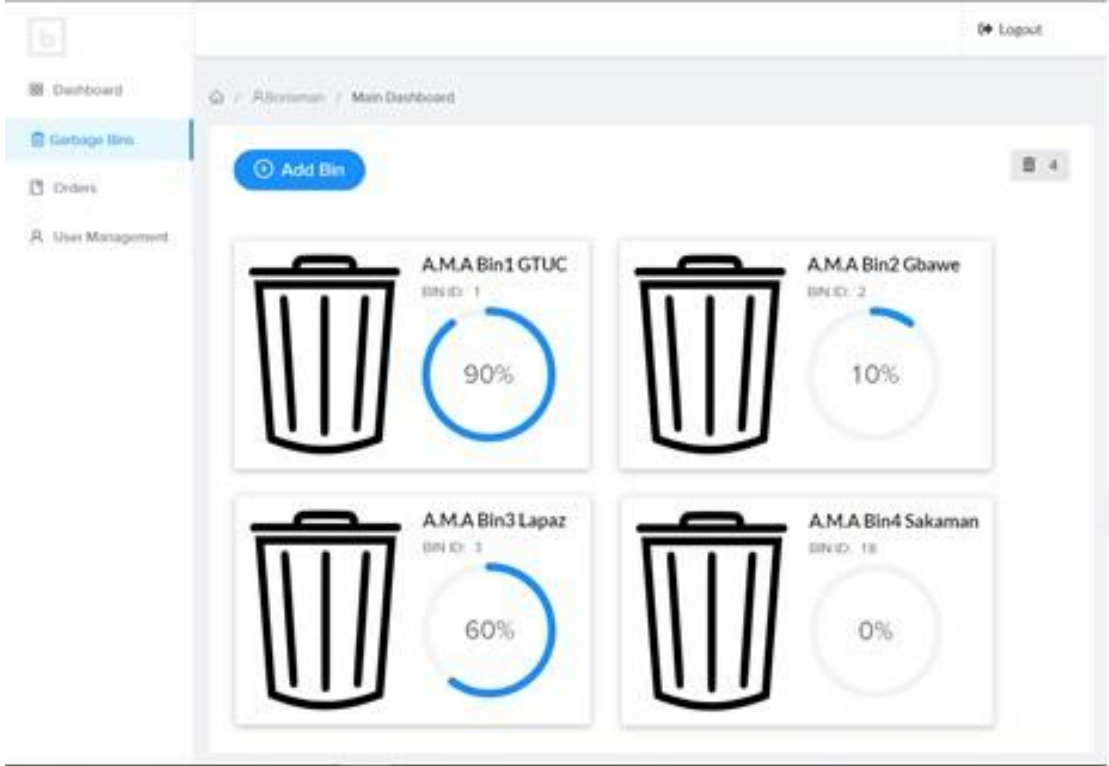
#### B. Desktop Application System

The system worked well meeting all the requirements set earlier and most especially in terms of response to queries. The constructed bin and three extra simulated bins were in selected areas in Accra, the capital city of Ghana. present the geolocations, bin status information and work order information respectively.



Route Information for Bins on Desktop Application





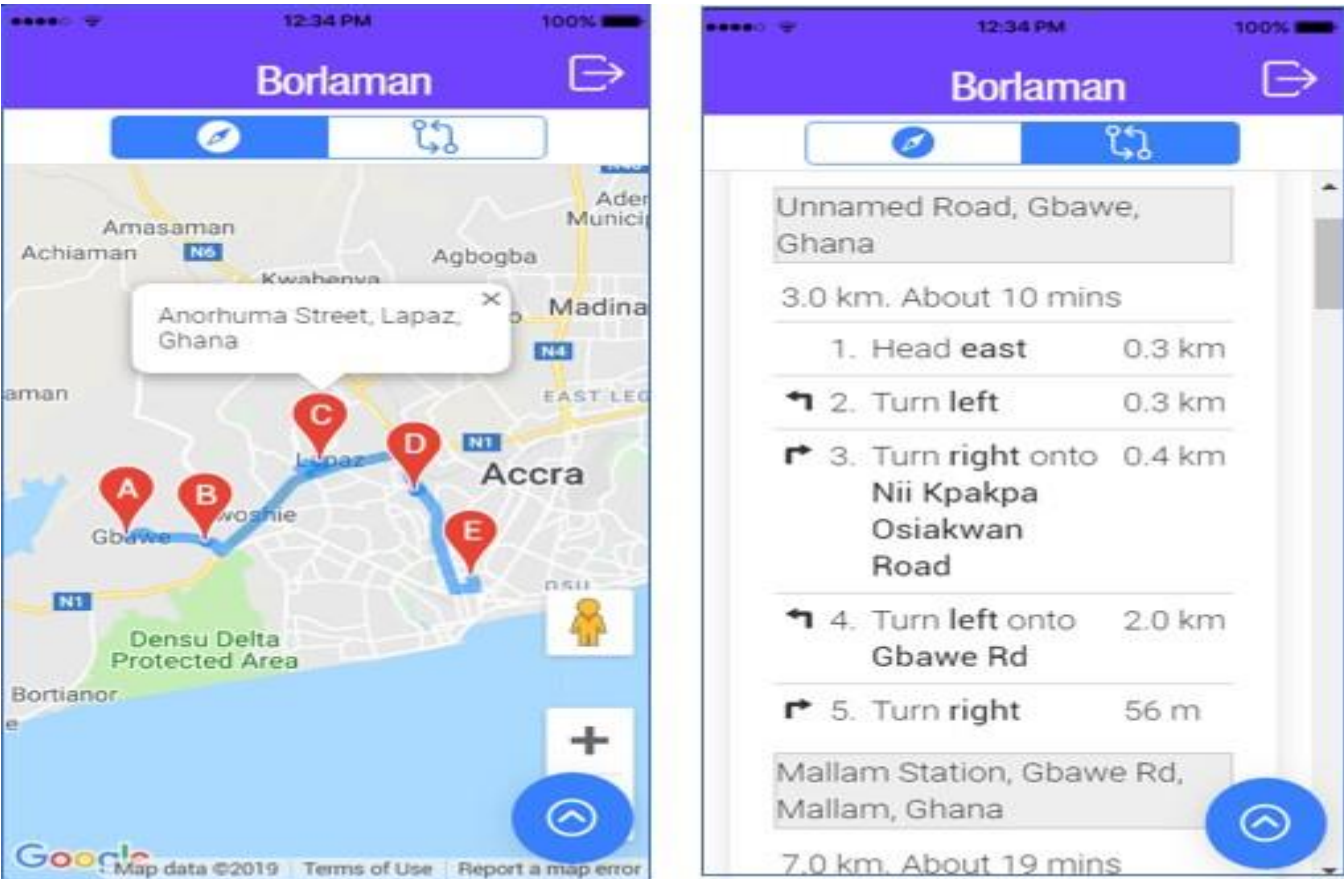
Display of Bin Status on Desktop Application

#	Name	Username	PhoneNumber	Email	Role	Status	Toggle Status
1	Ekow Baah	ekow	0501099579	nyanta@borlakor.ci	Borlaman	On Duty	Absent
2	Administrator Borla	admin	0205845289	admin@borlakor.cc	Admin	Absent	On Duty
3	Robert Adjel-Laryea	bob	0557144981	radjellaryea@gmail	Borlaman	On Duty	Absent
4	Christopher Micheal	xenya	0543894558	xenya@gmail.com	Admin	Absent	On Duty

Work Order on Desktop Application

C. The Mobile Application

The android application was named Borlaman, which is a coined term from Twi (a language in Ghana) implying a garbage collector. Figure 11 provides the map and route information to the next waste bin for collection which can be accessed on the garbage collector’s phone



Route Information for Bins on Mobile

## 10. ADVANTAGES & DISADVANTAGES

### ADVANTAGES:

The proposed plan has many advantages, it is also cogent enough to be implemented in every street of a developing nation. the advantages lie in its easy and valuable functioning. This will not only improve the streets we live in, but also provide a pavement for better working system.

- Efficient and effective Functioning.
- Cleaner Environs
- Better health issues.
- Pollution free and stinking free environs
- Smart cities
- Technology development
- Tourist attraction.

Once implemented, this method would be easy to work on. The garbage will be dumped into the bins thereby reducing the health-threats imposed by the trash present all-around.

### DISADVANTAGES:

- System requires more number of waste bins for separate waste collection as per population in the city. This results into high initial cost due to expensive smart dustbins compare to other methods.
- Sensor nodes used in the dustbins have limited memory size.
- Wireless technologies used in the system such as zigbee and wifi have shorter range and lower data speed. In RFID based systems, RFID tags are affected by surrounding metal objects (if any).
- It reduces man power requirements which results into increase in unemployments for unskilled people.
- The training has to be provided to the people involved in the smart waste management system.



## 11. CONCLUSION

This project is very effective in managing waste in any big city. Rather than using conventional periodic collection methods here priority system is used so the city is clean all the time without any overflowing dumpsters. It has been tested and verified properly to make sure all the different parts work together for a smooth function of the whole system.

This work has proposed a framework for waste management with an optimized routing and alternative SMS alert to garbage collectors. In this work, contribution to the body of knowledge relates to the methodology outlined for solving a waste management problem which is receiving a lot of attention globally. The power source to the hardware system is a 9V alkaline battery which could be a challenge when not replaced early after it has drained down. The system can further be miniaturized in further works plus the provision of solar energy.

Improper disposal and improper maintenance of domestic waste create issues in public health and environment pollution thus this paper attempts to provide practical solution towards managing the waste collaborating it with the use of IOT i.e. providing free internet facilities for a specific time once the trash is dumped into the bin. the proposed system will definitely help to overcome all the serious issues related to waste and keep the environment clean.

The behaviour of generating garbage is too dangerous not only for today's generation, but also for future generations. It is critical to educate people and encourage them to practise Recycle, Reuse, and Reduce instead of producing waste. Waste disposal should be a priority for municipalities and governments

## 12. FUTURE SCOPE

Every project is always has scope for improvement, perhaps the most pressing issue of separation of waste is when their dispose simultaneously. The waste segregator can be improvised to include the separation of paper and plastic, safe segregation of biomedical waste generated at home, compact and aesthetic Mechanical design.

The moisture sensor can be implemented hand in hand with the other sensors and the compartments for segregating the dry and wet waste can be created which will solve the issues related to waste segregation.

The future of Waste management starts and proceeds with technological adjustments. Like every other industry, to proceed, the waste management industry needs to become digitized and data-driven to advance its work field. The future is smart and competitive! Especially for businesses, they are required to be one step ahead of their competitors. When smart waste management solutions are applied over time, the data is collected. These data in hand sensors can be used to identify fill patterns, optimize driver routes and schedules, and reduce operational costs. These sensors' cost is steadily decreasing, making smart bins more feasible to implement and more attractive to companies or city leaders.

When we say that the future is smart, it also means that it is practical. The selection of the containers minimizes the need for trash collection crews. The amount of labor and time spent on collection processes is minimized, and ultimately it's profitable. In addition to hardware, the time spent is reduced for management and reduced by using easy to use compact and comprehensive platforms and mobile apps for both ends of the waste management Processes

## 12. APPENDIX

### SOURCE CODE:

# Waste management system

Backend## Getting Started

### Installing

Dependencies

####python3.7

we are using python as a backend language

Follow instructions to install the latest version of python for your platform in the [python docs](https://docs.python.org/3/using/unix.html#getting-and-installing-the-latest-version-of-python)

##### Windows

[python3.7](https://www.python.org/downloads/windows/)

#### virtual environment

We recommend working within a virtual environment whenever using Python for projects. This keeps your dependencies for each project separate and organized. Instructions for setting up a virtual environment for your platform can be found in the python docs

#### PIP Dependencies

Once you have your virtual environment setup and running, install dependencies by navigating to the `/backend` directory and running:

```
```shell script
```

```
pip install -r requirements.txt
```

```
```
```

This will install all of the required packages we selected within the `requirements.txt` file.#### key dependencies

- [Flask](https://flask.palletsprojects.com/en/1.1.x/)

- [SQLALCHEMY](https://www.sqlalchemy.org/)

- [Flask-CORS](https://flask-cors.readthedocs.io/)

## Setup Database

## Running the Server ## API References ### Getting Started ### Error Handling ###  
Endpoints

#### GET /areas

- ##### General

\* Return a list of areas objects and number of total areas

- ##### Sample

\* Request

shell script

curl https://wastes-management.herokuapp.com/api/areas

\* Response

json

```
{
  "areas": [
    {
      "area_code": 22, "area_name": "Nattaraja Nagar", "area_size": 100.0,
      "city": "Komarapalaiyam", "latitude": "342342",
      "longitude": "42342"
    },
    {
      "area_code": 33, "area_name": "Nattaraja Nagar ", "area_size": 100.0,
      "city": "Komarapalaiyam",
      "latitude": "45345",
      "longitude": "423"
    },
    {
      "area_code": 44,
      "area_name": "Vasanthanagar",
      "area_size": 120.0,
      "city": "Pallipalaiyam"
    }
  ]
}
```

```

        "latitude": "45344635",
        "longitude": "423423"
    }
],
    "total_areas": 3
}
```

```

#### #### GET /areas/{area\_code}

##### - ##### General

\* Return a specific area object by area code

##### - ##### Sample

\* Request

shell script

```
curl https://wastes-management.herokuapp.com/api/areas/22
```

\* Response

```
json
{
```

```

    "area": { "area_code":
        22,
        "area_name": "Puthur"
        "area_size": 100.0,
        "city": "Kolathur",
        "latitude": "342342",
        "longitude": "42342"
    }

```

```
}
```

##### - ##### General

\* Return a list of baskets objects based on a specific area,

\* The object that return include list of baskets, total number of basket in this area

##### - ##### Sample

\* Request

shell script

```
curl https://wastes-management.herokuapp.com/api/areas/22/baskets
```

\* Response

json

```
{
  "baskets": [
    {
      "basket_height": 90,
      "basket_length": 40,
      "basket_width": 40,
      "id": 6,
      "latitude": "534535534",
      "level": "0% ",
      "longitude": "435345",
      "software_version":
        "v2.0"
    },

```

##### GET /areas/{area\_code}/baskets

```
{
  "basket_height": 90,
  "basket_length": 40,
  "basket_width": 40,
  "id": 7,
  "latitude": "5345345",
  "level": "0% ",
  "longitude": "5345",
  "software_version":
    "v2.0"
},
{
  "basket_height": 90,
  "basket_length": 40,
  "basket_width": 40,
  "id": 5,
  "latitude": "534534",
  "level": "83% ",
  "longitude":

```

```

        "534534534",
        "software_version":
        "v2.0"
    }
],
    "total_baskets": 3
}
```

```

#### #### GET /areas/{area\_code}/users

##### - ##### General

- \* Return a list of users objects based on a specific area,
- \* The object that return include list of users, total number of users in this area

##### - ##### Sample

##### \* Request

```shell script

```
curl https://wastes-management.herokuapp.com/api/areas/22/users
```

##### \* Response

```json

```

{
    "total_users": 0,
    "users": []
}

```

```

#### #### POST /areas

##### - #####General

- \* Insert new area in the system using the submitted longitude, latitude and area code
- \* Return success message and area object if created successfully

##### - #####Sample

##### \* Request

```shell script

```
curl -X POST https://wastes-management.herokuapp.com/api/areas -H "Content-Type:application/json" -d '{ "area_code": 33, "longitude": 4234432, "latitude": 324242 }'
```

```
---
```

```
* Response
```

```
```json
```

```
{
  "area": {
    "area_code":
      55,
    "area_name": "Kolathukadu",
    "area_size": 100.0,
    "city": "Pallipalaiyam Agarakaram",
    "latitude": "43424",
    "longitude": "423434"
  },
  "success": true
}
```

```
---
```

```
##### Get /baskets
```

```
- ##### General
```

```
* Return a list of baskets objects and number of total basket
```

```
- ##### Sample
```

```
* Request
```

```
```shell script
```

```
curl https://wastes-management.herokuapp.com/api/baskets
```

```
```
```

```
* Response
```

```
```json
```

```
{
  "baskets": [
    {
      "basket_height": 90,
      "basket_length": 40,
      "basket_width": 40,
      "id": 1,
      "latitude": "42342423",

```



```

        "level": "33% ",
        "longitude":
            "534534534",
        "software_version":
            "v1.0"
    },
    {
        "basket_height": 90,
        "basket_length": 40,
        "basket_width": 40,
        "id": 2,
        "latitude": "345353535",
        "level": "44% ",
        "longitude":
            "53453453",
        "software_version":
            "v1.0"
    },
    {
        "basket_height": 90,
        "basket_length": 40,
        "basket_width": 40,
        "id": 3,
        "latitude": "34535345",
        "level": "88% ",
        "longitude": "545353",
        "software_version":
            "v1.0"
    }
],
    "total_baskets": 9
}

```

#### Get /baskets/{basket\_id}

- ##### General

\* return a specific basket by id

- ##### Sample

\* Request

```shell script

curl https://wastes-management.herokuapp.com/api/baskets/1

```

\* Response

```json

{

"basket": {

"basket\_height":

90,

"basket\_length": 40,

"basket\_width": 40,

"id": 1,

"latitude": "42342423",

"level": "33%"

"longitude": "534534534",

"software\_version": "v1.0"

}

}

```

##### GET /baskets/{basket\_id}/wastes

- #####General

\* Return a list of wastes object based on a specific basket,

\* the object that return include basket id, wastes, total size of wastes that generated by this basket

- #####Sample

\* Request

```shell script

curl https://wastes-management.herokuapp.com/api/baskets/1/wastes

```

\* Response

```json

{

```

    "basket_id": 1,
    "total_size":
0.048, "wastes": [
    {
        "basket_id": 1,
        "date_of_creation": "Mon, 25 Jan 2021 18:42:35 GMT",
        "size": 0.016,
        "type": "bio"
    },
    {
        "basket_id": 1,
        "date_of_creation": "Mon, 25 Jan 2021 18:42:46 GMT",

```

\* Response

```

```json
{
    "size": 0.016,
    "type": "bio"
},
{
    "basket_id": 1,
    "date_of_creation": "Mon, 25 Jan 2021 18:42:50 GMT",
    "size": 0.016,
    "type": "bio"
}
]
}
```

```

#### #### POST /baskets

##### - #####General

- \* Create new basket using the submitted longitude, latitude and area code
- \* you can set basket height, width, length, version manually,
- \* Return success message and basket object if created successfully

##### - #####Sample

- \* Request

```

```shell script

```

```
curl -X POST https://wastes-management.herokuapp.com/api/baskets -H
"Content-Type:application/json" -d '{ "area_code": 33, "longitude": 4234432,
"latitude": 324242 }'
```

shell script

```
curl -X POST https://wastes-management.herokuapp.com/api/baskets -H "Content-
Type: application/json" -d '{ "area_code": 33, "longitude": 4234432, "latitude": 324242,
"basket_height:120, "basket_width": 50, "basket_length": 50, "basket_version": "v4.0" }'
```

```
"basket": {
  "basket_height":
    90,
  "basket_length": 40,
  "basket_width": 40,
  "id": 10,
  "latitude": "324242",
  "level": "0%",
  "longitude": "4234432",
  "software_version":
    "v1.0"
},
"success": true
}
```
```

#### PATCH /baskets

- #####General

- \* Update the basket software version
- \* Return the number of updated baskets

- #####Sample

- \* Request

```shell script

```
curl -X PATCH https://wastes-management.herokuapp.com/api/baskets -H
"Content-Type:application/json" -d '{ "software_version": "V2.0" }'
```

```

    * Response
    json
    {
        "baskets_update": 10
    }

    ##### PATCH /baskets/{basket_id}

    #####General

    * Update basket level by submitted basket level

    * Return success message

    - #####Sample

    * Request
    ```shell script

    curl -X PATCH https://wastes-management.herokuapp.com/api/baskets/1 -H
    "Content-Type:application/json" -d '{ "level": 0}'

    ```

    * Response
    ```json
    {
        "success": true
    }
    ```

    ##### DELETE /baskets/{basket_id}

    - #####General

    * Update the basket software version

    * Return the number of updated baskets

    - #####Sample

    * Request
    ```shell script
    curl -X DELETE https://wastes-management.herokuapp.com/api/baskets/1

    ```

    * Response
    ```json
    {
        "success": true
    }

```

#### GET /users

- ##### General

\* Return a list of user object

- ##### Sample

\* Request

```shell script

curl https://wastes-management.herokuapp.com/api/users

```

\* Response

```json

```
{
  "user": [
    {
      "Date_of_birth": null,
      "email": "gobika.v@jkkn.ac.in",
      "first_name": "ahemd",
      "gender": "male",
      "last_name":
      "hosam",
      "user_name":
      "ahmed"
    },
    {
      "Date_of_birth": null,
      "email":
      "mahmoudamr@gamil.com",
      "first_name": "mahmoud",
      "gender": "male",
      "last_name": "amr",
      "user_name":
      "mahmoud2"
    },
    {
      "Date_of_birth": null,
      "email":
```

```

        "ahemd.esmail@gamil.com",
        "first_name": "ahmed",

        "gender": "male",
        "last_name": "esmail",
        "user_name":
        "ahmed2"
    }
]
}
```

```

```
##### GET /users/{user_name}
```

```
- ##### General
```

```
* Return specific user object based on user_name
```

```
- ##### Sample
```

```
* Request
```

```
```shell script
```

```
curl https://wastes-management.herokuapp.com/api/users/meladsamuel
```

```
* Response
```

```
##### GET /users
```

```
- ##### General
```

```
* Create new user by submitted user name, first name, last name, email, password, gender
```

```
* Return success message and user object
```

```
- ##### Sample
```

```
* Request
```

```
```shell script
```

```
curl -X POST https://wastes-management.herokuapp.com/api/users -H "Content-
Type: application/json" -d '{ "user_name": "ali", "first_name": "ali", "last_name":
"emad", "email": "ali.emad@gamil.com", "password": "123", "gender": "male",
"area_code": 22 }'`
```

```
* Response
```

```
```json
{
```

```

    "success":true,

    "user": { "Date_of_birth":null

    ,

    "email":"ali.emad@gamil.com",

    "first_name":"ali",

    "gender":"male",

    "last_name":"emad

    ", "user_name":"ali"

    }

}
```

```

#### #### GET /vehicles

##### - ##### General

\* Return a list of vehicles objects

##### - ##### Sample

\* Request

```shell script

curl https://wastes-management.herokuapp.com/api/vehicles

```

\* Response

```json

```

#### #### GET /vehicles/{plate\_number}

##### - ##### General

\* Return a list of vehicles objects

##### - ##### Sample

\* Request

```shell script

curl https://wastes-management.herokuapp.com/api/vehicles

\* Response

`json

#### #### POST /vehicles

##### - ##### General



\* Insert new vehicles in the system by submitted plate number, container size, tank size, employee ssn

\* Return success message and list of vehicles object

- ##### Sample

\* Request

```shell script

```
curl -X POST https://wastes-management.herokuapp.com/api/vehicles -H
"Content-Type:application/json" -d '{"plate_number": 543, "container_size": 6.0,
"tank_size": 100.0,
"employee_ssn": 29854364445354}'
```

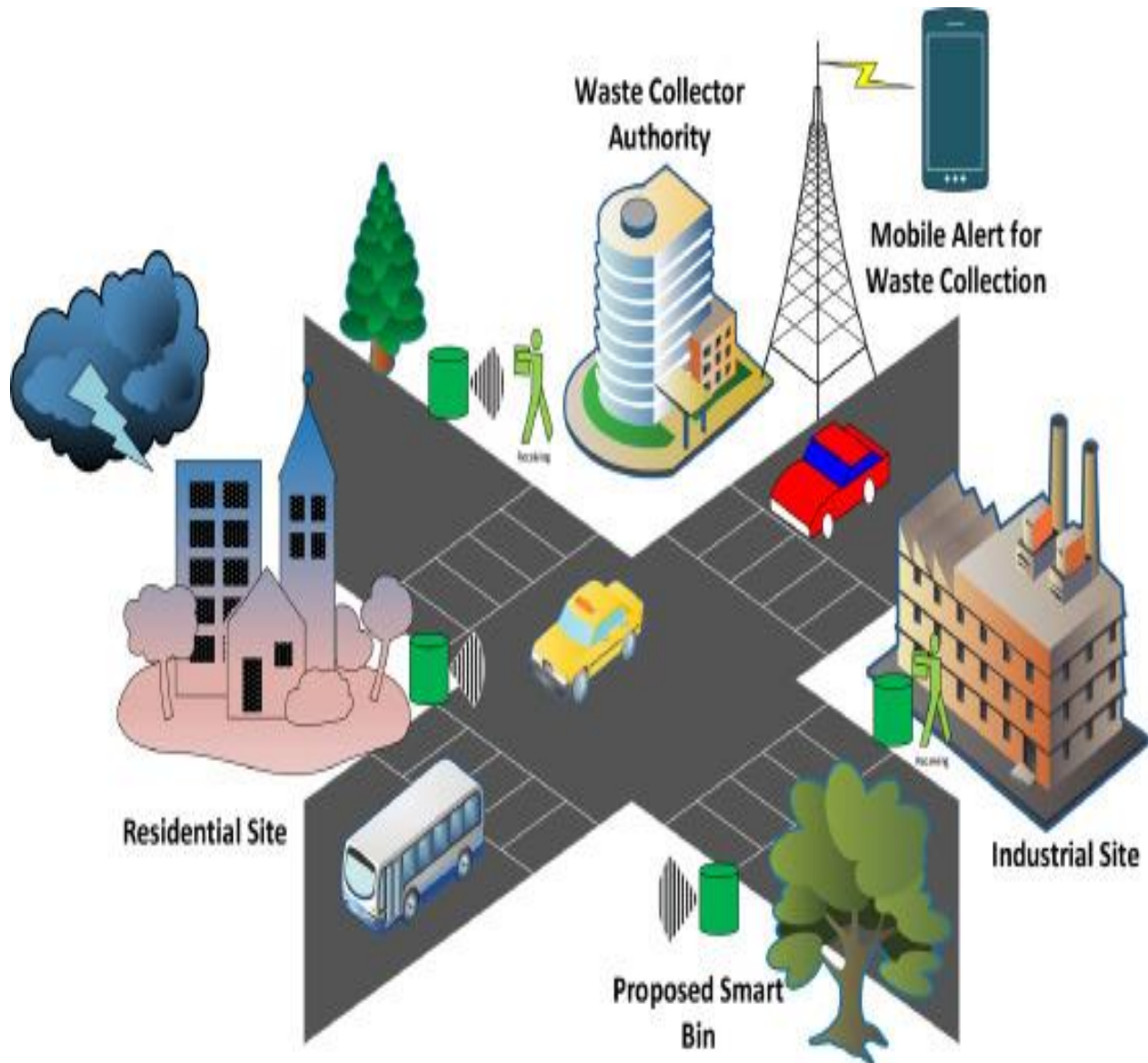
```

\* Response

```json

```
{
  "success":
  true, "vehicle":
  [
    {
      "container_size":
      6.0, "driver": {
        "SSN": 29854364445354,
        "date_of_birth": "Sun, 11 Oct 1998 00:00:00
        GMT", "full_name": "مصطفى صابر محمد",
        "phone": "011432523482",
        "user_name": "mostafa"
      },
      "plate_number":
      543, "tank_level":
      null, "tank_size":
      100.0
    }
  ]
}
```

```



## **GITHUB & PROJECT DEMO LINK**

### **GITHUB LINK:**

**<https://github.com/IBM-EPBL/IBM-Project-47507-1660799988>**