PROJECT REPORT

1. INTRODUCTION

1.1 PROJECT OVERVIEW

The Internet and its applications have become an integral part of today's human lifestyle. It has become an essential tool in every aspect Due to the tremendous demand and necessity, researchers went beyond connecting just computers into the web. These researches led to the birth of an Internet of Things (IoT).Things (Physical Devices) that are connected to the Internet and sometimes these devices can be controlled from the internet is commonly called Internet of Things. Nowadays, there are a number of techniques which are purposefully used and are being built up for well management of garbage or solid waste. Sensors and IOT module i.e. Wi-Fi are the latest trends and are one of the best combinations to be used in the project. Hence a combination of both of these technologies is used in the project. Here we are using raspberry pi. A threshold value is set in the IOT. In these we use ultrasonic sensors .When that value is met then it will be sent to the officials through a module about the overload and also to clear the garbage as soon as possible. The same thing is displayed on the LCD, which is connected to the output port of the controller. IOT through data available on web portal about all area dustbin

1.2 PURPOSE

Using technology and innovation to optimize current systems will enable cities to become smarter, more efficient and save resources. Due to the growing population, the amount of waste being produced is vast and rapidly increasing. The management of this waste is therefore a significant area for much-needed improvement.

Currently, waste collection systems are in most cases outdated and result in pick-ups that are unnecessary or on the contrary – long-overdue. Unnecessary pickups result in 70% higher annual collection cost. When routes are planned inefficiently, congestion is created and more fuel is required to complete the collection. Overall, this contributes to a 50% higher carbon footprint.

With the use of IoT solutions for waste management, these issues can be solved by creating a more efficient pathway for garbage trucks. IoT sensor technology can be used to indicate when the emptying is actually needed. This customized and dynamic system for waste management can allow businesses, organizations, and citizens to all benefit.

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM

Proper management of food waste, a major component of municipal solid waste (MSW), is needed, especially in developing Asian countries where most MSW is disposed of in landfill sites without any pre treatment. Source separation can contribute to solving problems derived from the disposal of food waste. An organic waste source separation andcollection programme has been operated in model areas in Hanoi, Vietnam, since 2007. This study proposed three key parameters (participation rate, proper separation rate and proper discharge rate) for behaviour related to source separation of household organic waste, and monitored the progress of the programme based on the physical composition of household waste sampled from 558 households in model programme areas of Hanoi. The results showed that 13.8% of 558 households separated organic waste, and 33.0% discharged mixed (unseparated) waste improperly. About 41.5% (by weight) of the waste collected as organic waste was contaminated by inorganic waste, and one-third of the waste disposed of as organic waste by separators was inorganic waste. We proposed six hypothetical future household behaviour scenarios to help local officials identify a final or midterm goalfor the programme. We also suggested that the city government take further actions to increase the number of people participating in separating organic waste, improve the accuracy of separation and prevent non-separators from discharging mixed waste improperly.

2.2 REFRENCES

- https://www.mural.co/templates/empathy-map-canvas
- https://app.mural.co/t/veboosita1204/m/veboosita1204/1663922358124/34ba1c24af97 ed 215427f9e2458e1a0d606b1746?sender=ud2f59c937d58a4b323be0236
- J. A. Nathanson, “Solid-waste management | Britannica.com.”
- L. A. Manaf, M. A. A. Samah, and N. I. M. Zukki, “Municipal solid waste management in Malaysia: Practices and challenges,” Waste Manag., vol. 29, no. 11, pp. 2902–2906, Nov. 2009.
- S. Sharmin and S. T. Al-Amin, “A Cloud-based Dynamic Waste Management System for Smart Cities,” in Proceedings of the 7th Annual Symposium on Computing for Development - ACM DEV’16, 2016, pp. 1–4.
- B. R. Balakrishnan Ramesh Babu, A. K. Anand Kuber Parande, and C.A. Chiya Ahmed Basha, “Electrical and electronic waste: a global environmental problem,” Waste Manag. Res., vol. 25, no. 4, pp. 307–318, Aug. 2007.
- K. Kawai and L. T. M. Huong, “Key parameters for behaviour related to source separation of household organic waste: A case study in Hanoi, Vietnam,” Waste Manag. Res., vol. 35, no. 3, pp. 246–252, Mar. 2017

2.3 PROBLEM STATEMENT DEFINITION

In our current scenario, we see many garbage bins placed around us in cities which are overflowing due to increase in garbage day by day. This situation creates unhygienic condition for people and cause diseases so to avoid this situation we have planned to design “Smart Waste Management System for Metropolitan Cities” using IoT. Here we use multiple bins which is located throughout the cities were a device is develop such a way that it helps to track the levels of bins and unique id is provided for all the bins which helps to identify whether the bins are filled or not. Once the bin is filled it is detected and emptied.

3. IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviors and attitudes.

It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges.

Smart Waste Management System for Metropolitan Cities

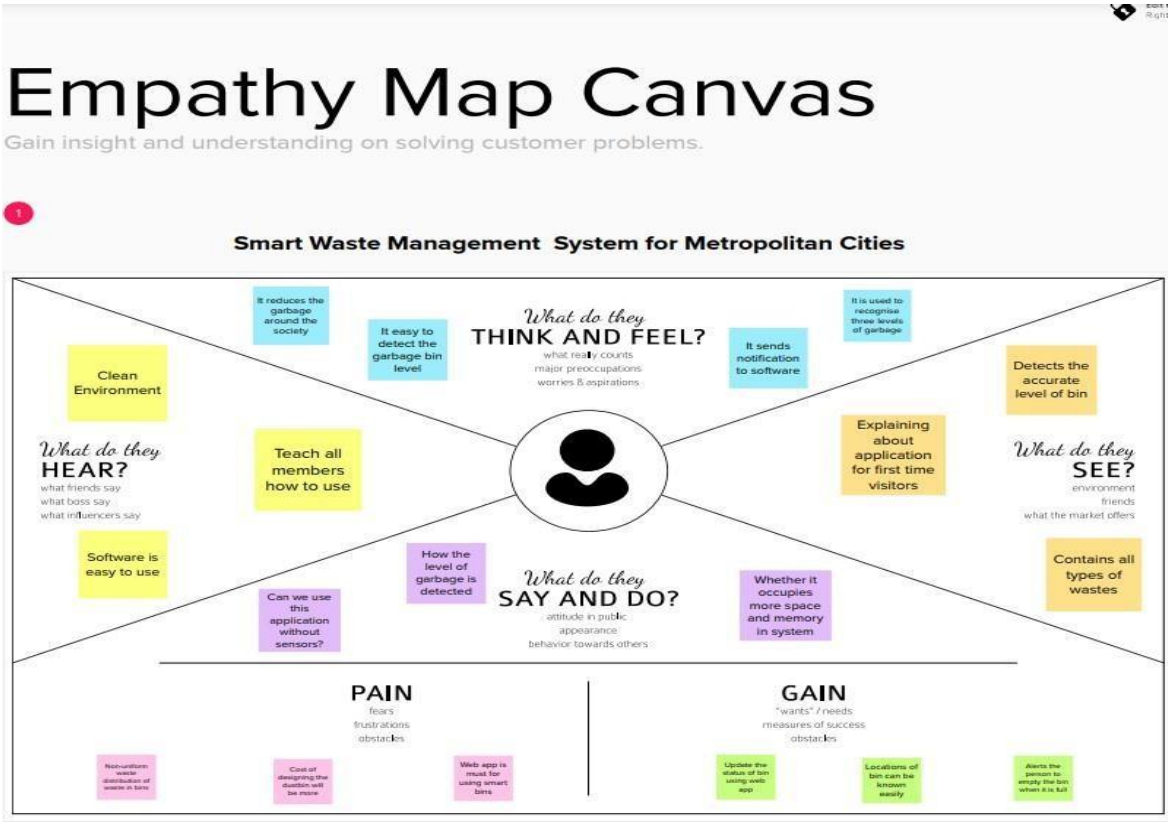

Empathy Map Canvas
Gain insight and understanding on solving customer problems.
1
Smart Waste Management System for Metropolitan Cities
What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
It reduces the garbage around the society
It easy to detect the garbage bin level
It sends notification to software
It is used to recognise three levels of garbage
What do they HEAR?
what friends say
what boss say
what influencers say
Clean Environment
Teach all members how to use
Software is easy to use
What do they SEE?
environment
friends
what the market offers
Detects the accurate level of bin
Explaining about application for first time visitors
Contains all types of wastes
What do they SAY AND DO?
attitude in public
appearance
behavior towards others
How the level of garbage is detected
Can we use this application without sensors?
Whether it occupies more space and memory in system
PAIN
fears
frustrations
obstacles
Non-uniform waste distribution of waste in bins
Cost of designing the dustbin will be more
Web app is must for using smart bins
GAIN
"wants" / needs
measures of success
obstacles
Update the status of bin using web app
Locations of bin can be known easily
Alerts the person to empty the bin when it is full

Figure3.1

Reference:

https://app.mural.co/t/veboosita1204/m/veboosita1204/1663922358124/34ba1c24af97ed 215427f9e2458e1a0d606b1746?sender=ud2f59c937d58a4b323be0236

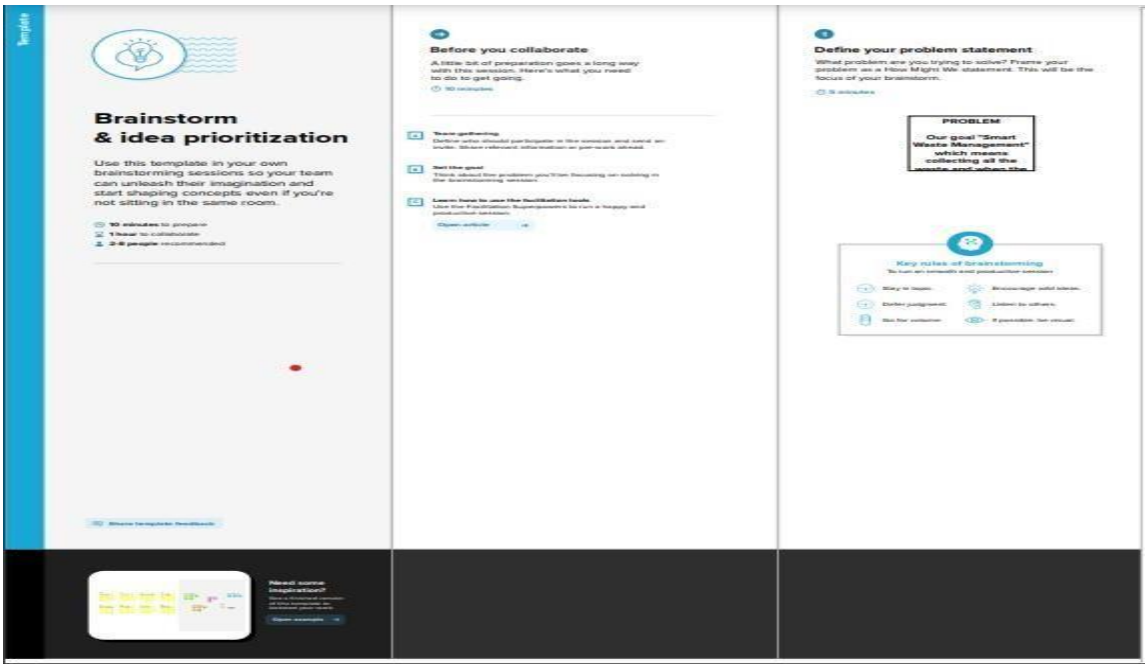
3.2 IDEATION & BRAINSTORMING

Brainstorm & Idea Prioritization Template:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions. Use this template in your own brainstorming sessions so your team can unleash theirimagination and start shaping concepts even if you're not sitting in the same room.

Reference: https://www.mural.co/templates/empathy-map-canvas

Step-1: Team Gathering, Collaboration and Select the Problem Statement



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
2-8 people recommended

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 gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

Set the goal
Think about the problem you'll be focusing on solving in the brainstorming session.

Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.

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

PROBLEM

Our goal "Smart Waste Management" which means collecting all the waste and when the

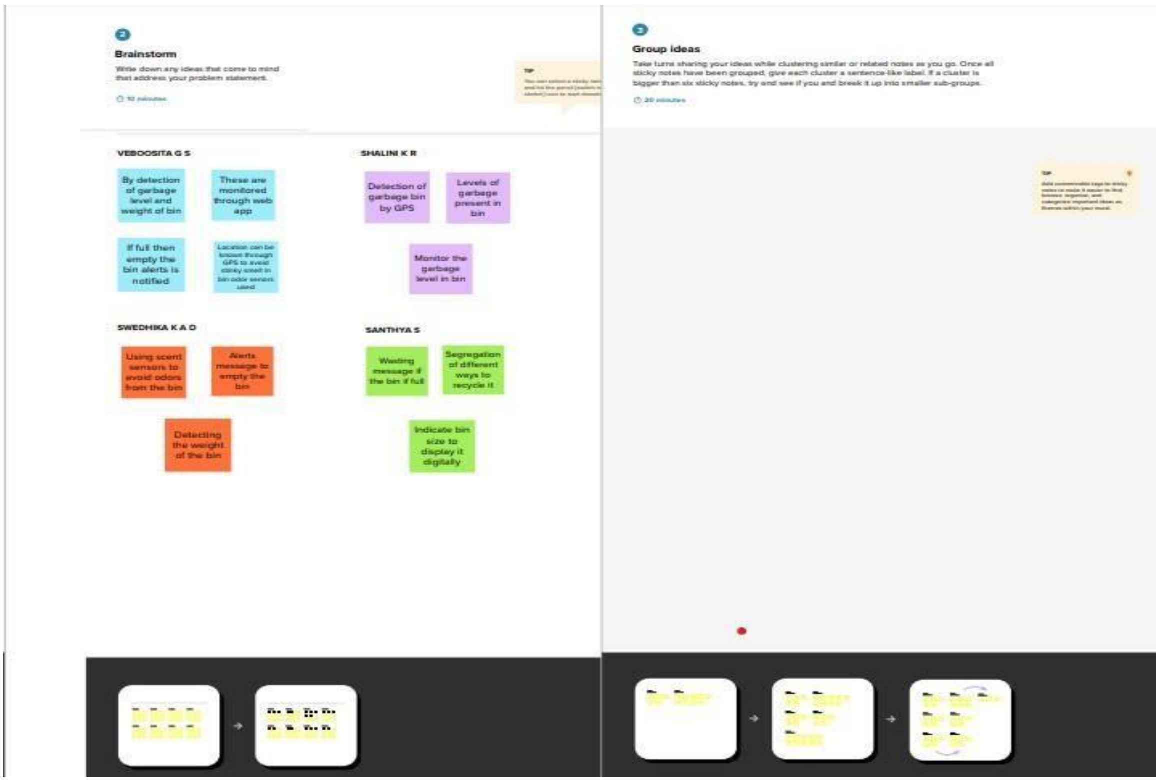
Key rules of brainstorming
To run an smooth and productive session

Stay in topic.
Defer judgment.
Go for volume.
Encourage wild ideas.
Listen to others.
If possible, be visual.

Need some inspiration?

Figure 3.1

Step-2: Brainstorm, Idea Listing and Grouping



2

Brainstorm

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

10 minutes

VEBOOSITA G S

By detection of garbage level and weight of bin

These are monitored through web app

If full then empty the bin alerts is notified

Location can be known through GPS to avoid stinky smell in bin odor sensors used

SHALINI K R

Detection of garbage bin by GPS

Levels of garbage present in bin

Monitor the garbage level in bin

SWEDHIKA K A D

Using scent sensors to avoid odors from the bin

Alerts message to empty the bin

Detecting the weight of the bin

SANTHYA S

Wasting message if the bin if full

Segregation of different ways to recycle it

Indicate bin size to display it digitally

3

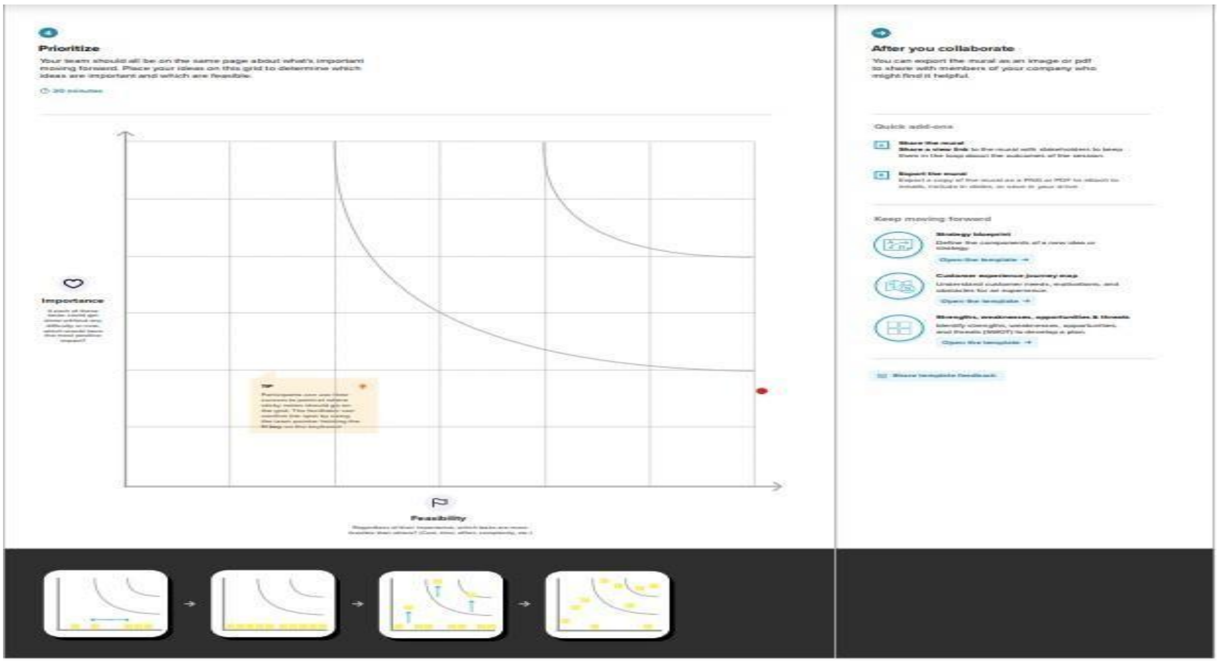
Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

Figure3.2

Step-3: Idea Prioritization



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.

20 minutes

Importance

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

After you collaborate

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

Quick add-ons

Share the mural

Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

Export the mural

Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

Keep moving forward

Strategy blueprint

Define the components of a new idea or strategy.

Open the template →

Customer experience journey map

Understand customer needs, motivations, and obstacles for an experience.

Open the template →

Strengths, weaknesses, opportunities & threats

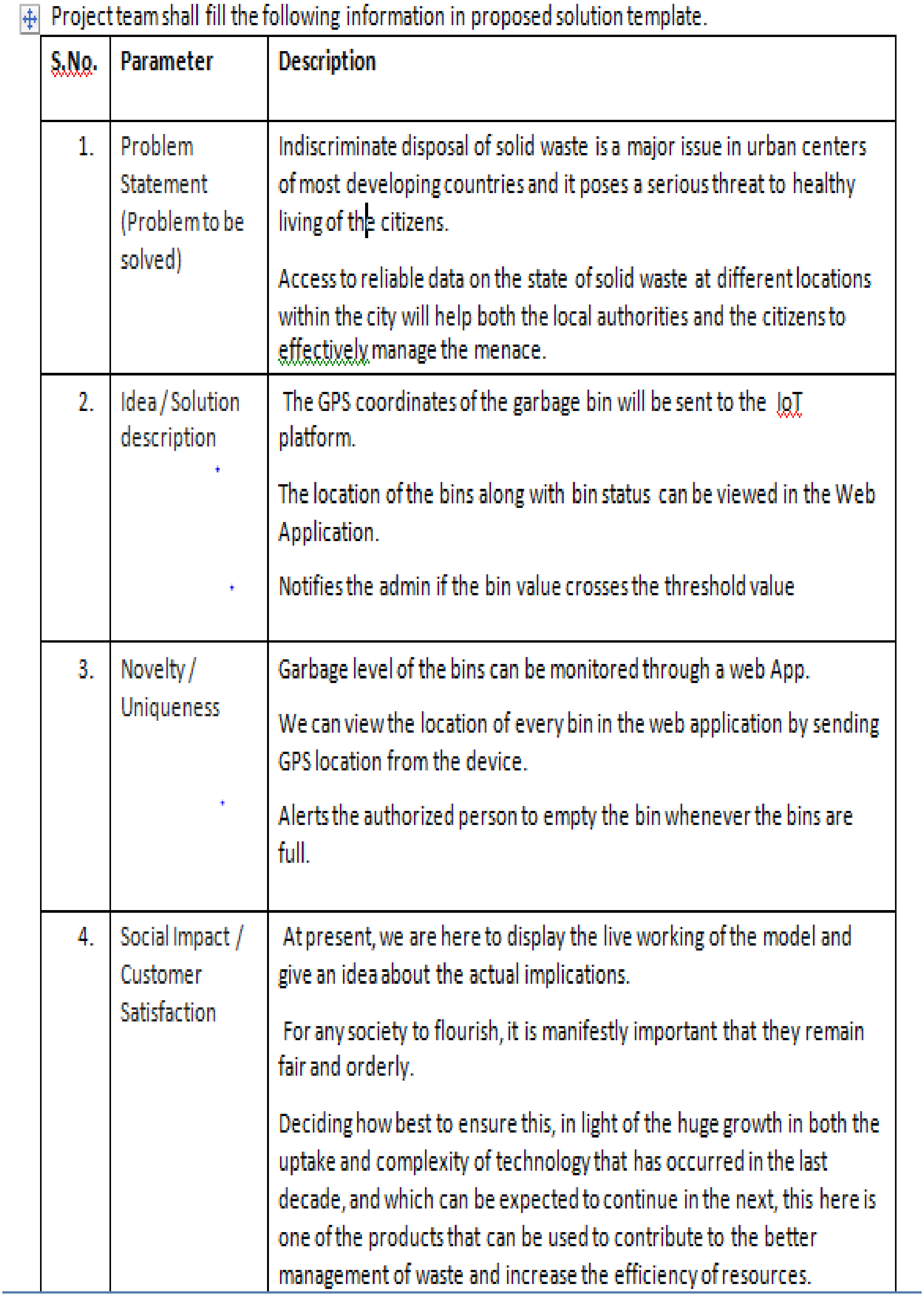
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.

Open the template →

Share template feedback

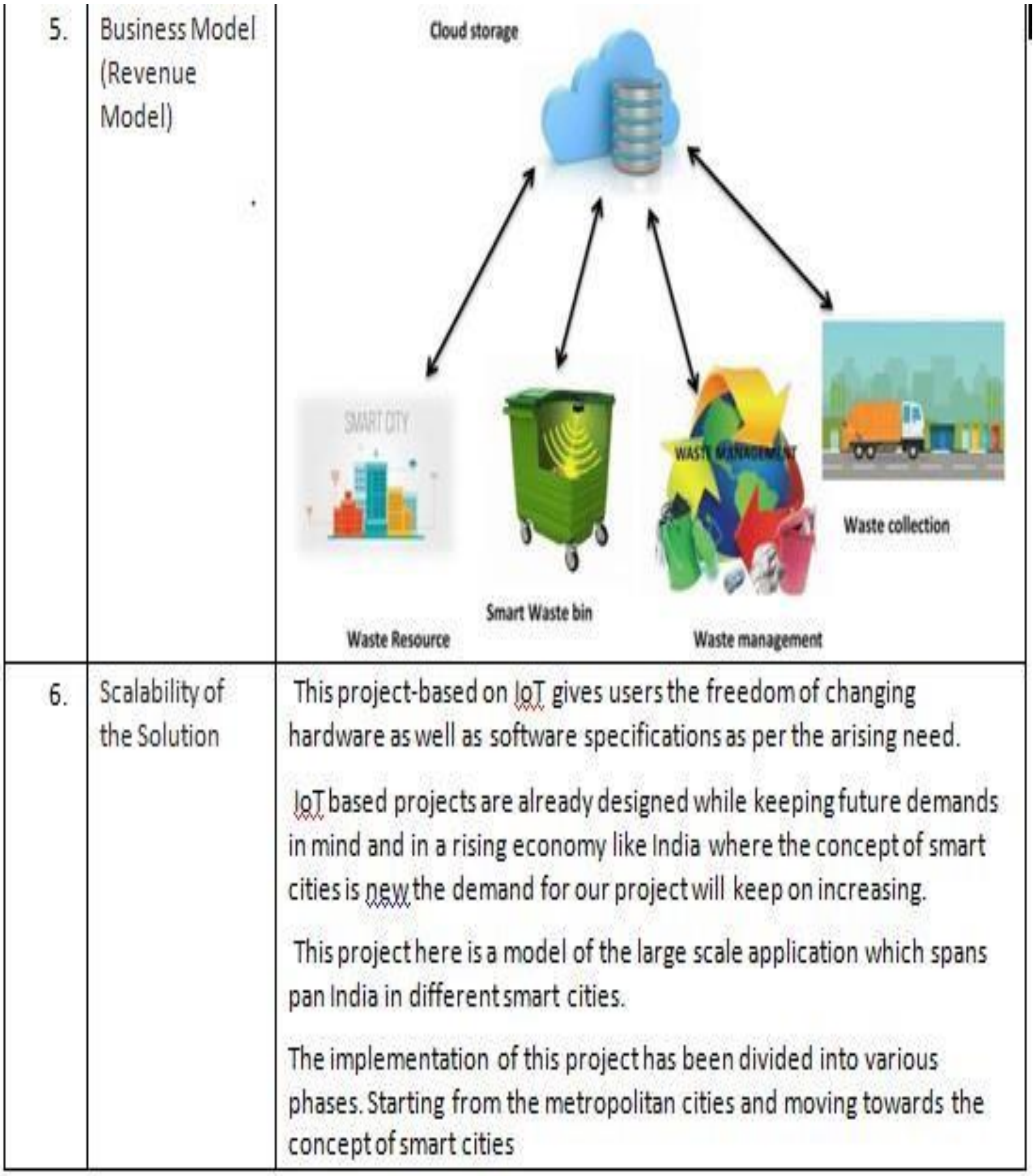
Figure3.3

3.2 PROPOSED SOLUTION



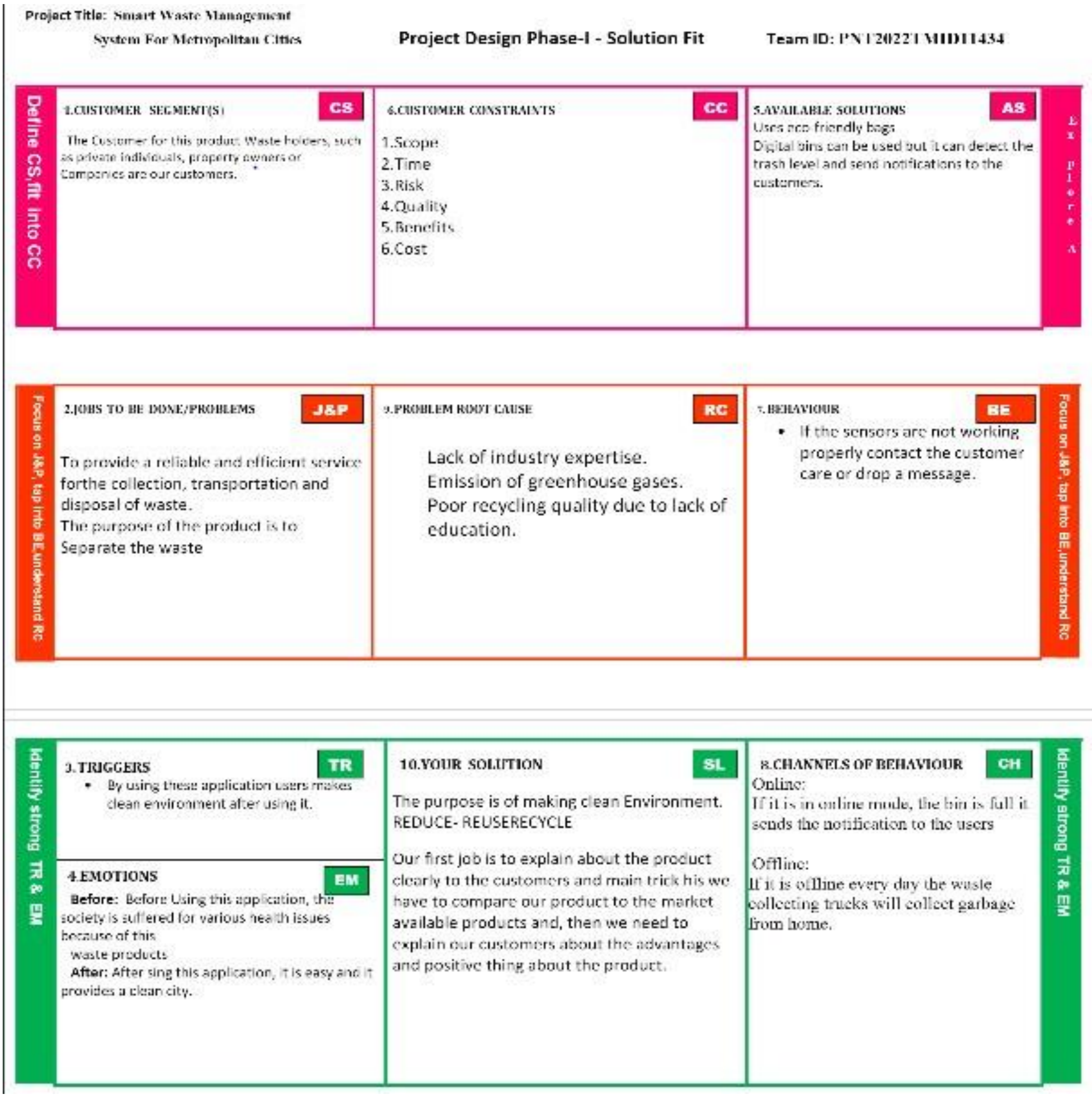
Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Indiscriminate disposal of solid waste is a major issue in urban centers of most developing countries and it poses a serious threat to healthy living of the citizens. Access to reliable data on the state of solid waste at different locations within the city will help both the local authorities and the citizens to effectively manage the menace.
2.	Idea / Solution description	The GPS coordinates of the garbage bin will be sent to the IoT platform. The location of the bins along with bin status can be viewed in the Web Application. Notifies the admin if the bin value crosses the threshold value
3.	Novelty / Uniqueness	Garbage level of the bins can be monitored through a web App. We can view the location of every bin in the web application by sending GPS location from the device. Alerts the authorized person to empty the bin whenever the bins are full.
4.	Social Impact / Customer Satisfaction	At present, we are here to display the live working of the model and give an idea about the actual implications. For any society to flourish, it is manifestly important that they remain fair and orderly. Deciding how best to ensure this, in light of the huge growth in both the uptake and complexity of technology that has occurred in the last decade, and which can be expected to continue in the next, this here is one of the products that can be used to contribute to the better management of waste and increase the efficiency of resources.



5.	Business Model (Revenue Model)	
6.	Scalability of the Solution	This project-based on IoT gives users the freedom of changing hardware as well as software specifications as per the arising need. IoT based projects are already designed while keeping future demands in mind and in a rising economy like India where the concept of smart cities is new the demand for our project will keep on increasing. This project here is a model of the large scale application which spans pan India in different smart cities. The implementation of this project has been divided into various phases. Starting from the metropolitan cities and moving towards the concept of smart cities

3.3 PROBLEM SOLUTION FIT



Project Title: Smart Waste Management System For Metropolitan Cities

Project Design Phase-I - Solution Fit

Team ID: PNT2022TMID11434

Define CS, fit into CC

1.CUSTOMER SEGMENT(S) CS

The Customer for this product Waste holders, such as private individuals, property owners or Companies are our customers.

6.CUSTOMER CONSTRAINTS CC

1.Scope
2.Time
3.Risk
4.Quality
5.Benefits
6.Cost

5.AVAILABLE SOLUTIONS AS

Uses eco-friendly bags
Digital bins can be used but it can detect the trash level and send notifications to the customers.

Explore A

Focus on J&P, tap into BE, understand RC

2.JOBS TO BE DONE/PROBLEMS J&P

To provide a reliable and efficient service forthe collection, transportation and disposal of waste.
The purpose of the product is to Separate the waste

9.PROBLEM ROOT CAUSE RC

Lack of industry expertise.
Emission of greenhouse gases.
Poor recycling quality due to lack of education.

7.BEHAVIOUR BE

- If the sensors are not working properly contact the customer care or drop a message.

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3.TRIGGERS TR

- By using these application users makes clean environment after using it.

4.EMOTIONS EM

Before: Before Using this application, the society is suffered for various health issues because of this waste products
After: After sing this application, it is easy and it provides a clean city.

10.YOUR SOLUTION SL

The purpose is of making clean Environment.
REDUCE- REUSERECYCLE

Our first job is to explain about the product clearly to the customers and main trick his we have to compare our product to the market available products and, then we need to explain our customers about the advantages and positive thing about the product.

8.CHANNELS OF BEHAVIOUR CH

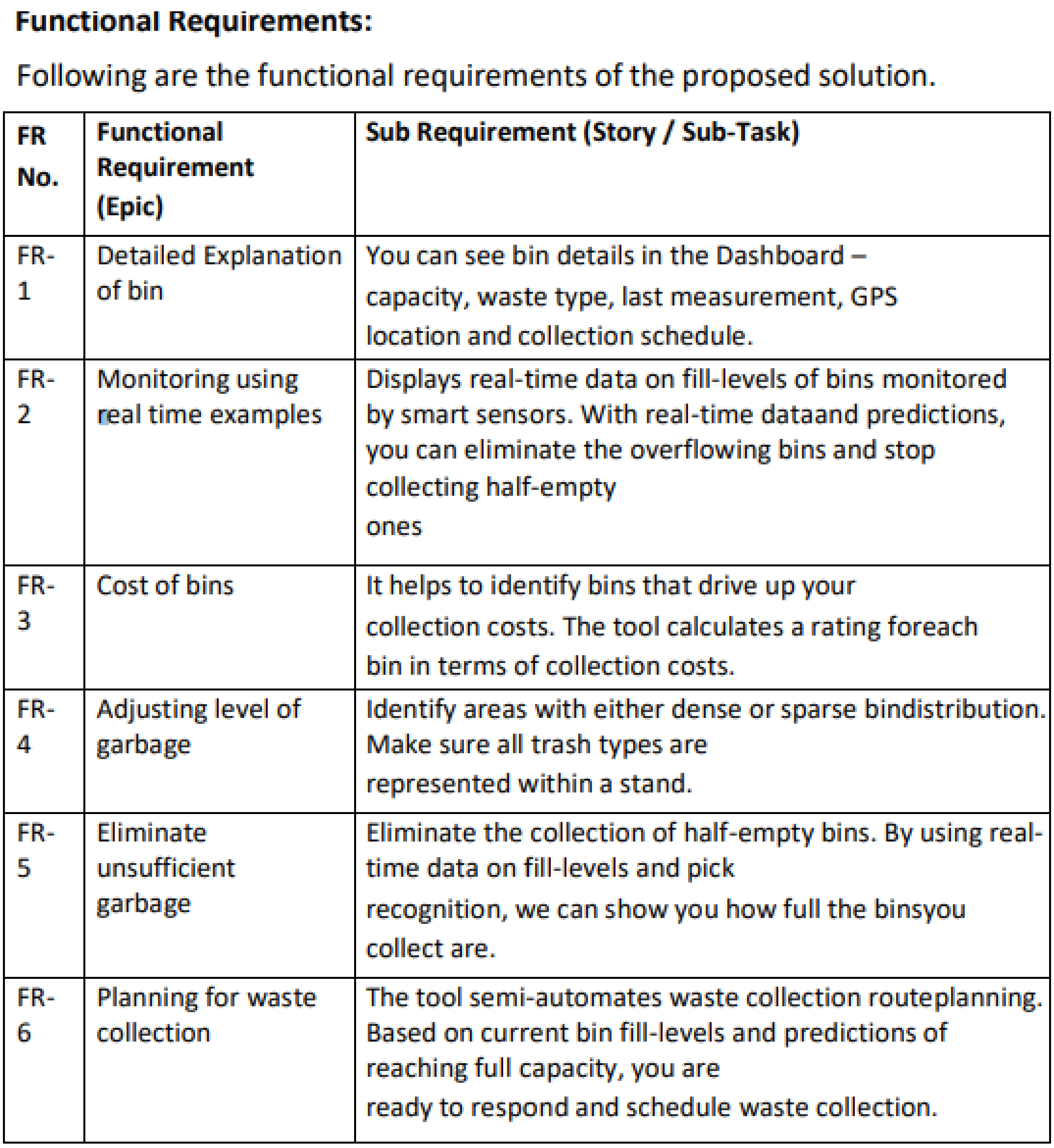
Online:
If it is in online mode, the bin is full it sends the notification to the users

Offline:
If it is offline every day the waste collecting trucks will collect garbage from home.

Identify strong TR & EM

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

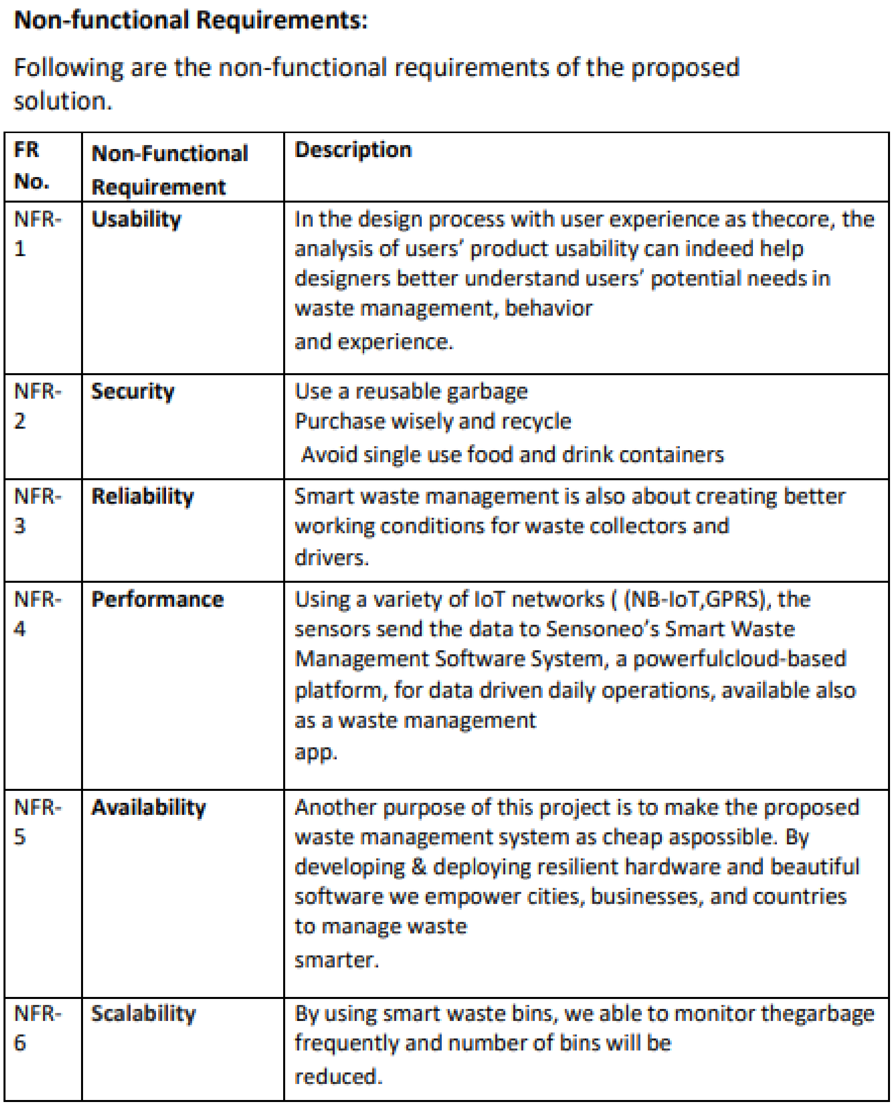


Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Detailed Explanation of bin	You can see bin details in the Dashboard – capacity, waste type, last measurement, GPS location and collection schedule.
FR-2	Monitoring using real time examples	Displays real-time data on fill-levels of bins monitored by smart sensors. With real-time dataand predictions, you can eliminate the overflowing bins and stop collecting half-empty ones
FR-3	Cost of bins	It helps to identify bins that drive up your collection costs. The tool calculates a rating foreach bin in terms of collection costs.
FR-4	Adjusting level of garbage	Identify areas with either dense or sparse bindistribution. Make sure all trash types are represented within a stand.
FR-5	Eliminate unsufficient garbage	Eliminate the collection of half-empty bins. By using real-time data on fill-levels and pick recognition, we can show you how full the binsyou collect are.
FR-6	Planning for waste collection	The tool semi-automates waste collection routeplanning. Based on current bin fill-levels and predictions of reaching full capacity, you are ready to respond and schedule waste collection.

4.2 NON-FUNCTIONAL REQUIREMENT



Non-functional Requirements:

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	In the design process with user experience as thecore, the analysis of users' product usability can indeed help designers better understand users' potential needs in waste management, behavior and experience.
NFR-2	**Security**	Use a reusable garbage Purchase wisely and recycle Avoid single use food and drink containers
NFR-3	**Reliability**	Smart waste management is also about creating better working conditions for waste collectors and drivers.
NFR-4	**Performance**	Using a variety of IoT networks ((NB-IoT,GPRS), the sensors send the data to Sensoneo's Smart Waste Management Software System, a powerfulcloud-based platform, for data driven daily operations, available also as a waste management app.
NFR-5	**Availability**	Another purpose of this project is to make the proposed waste management system as cheap aspossible. By developing & deploying resilient hardware and beautiful software we empower cities, businesses, and countries to manage waste smarter.
NFR-6	**Scalability**	By using smart waste bins, we able to monitor thegarbage frequently and number of bins will be reduced.

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 rightamount of the system requirement graphically. It shows how data enter andleaves the system, what changes the information, and where data is stored.

A smart waste management uses analytics to translate the data.

- The test conducted is the situation where the garbage bin is empty or its garbage level is very low.
- Then, the bin is filled.
- The notification is sent by the system.
- The garbage needs to be collected immediately.
- Location prone to overflow.
- The number of bins needed to avoid overflowing waste.
- The number of collection services that could be saved.
- The amount of fuel that could be saved.
- The driving distance that could be saved.
- No missed pickups of trashcans.
- New smart dustbins can be install by just connecting the IoT device to the cloud.

Example:

Data Flow Diagram

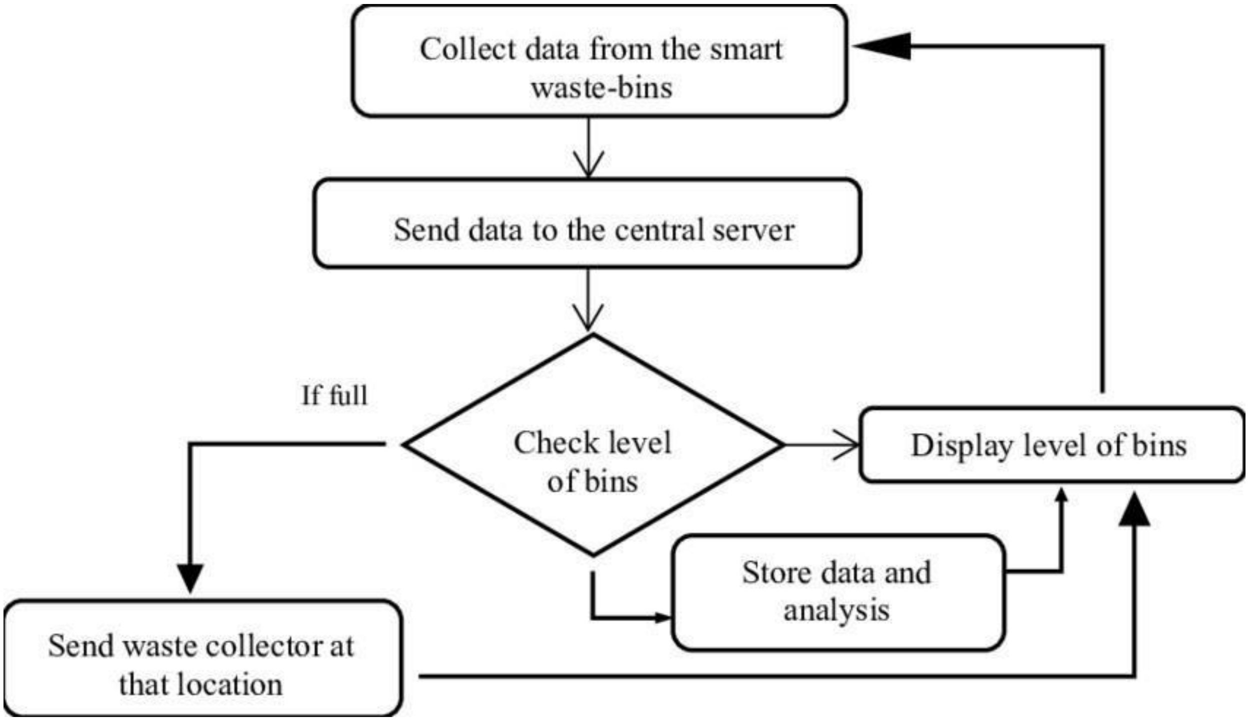

Collect data from the smart waste-bins
Send data to the central server
Check level of bins
If full
Display level of bins
Store data and analysis
Send waste collector at that location

Figure 5.1

5.2 SOLUTION & TECHNICAL ARCHITECTURE

The Deliverable shall include the architectural diagram as below and the information as per thetable1 & table 2

Component	Description	Technology
User Interface	IBM Watson IOT cloud platform	MQTT Protocol

Application Logic-1	The waste data arecollected using Sensors	Python
Application Logic-2	The collected data are monitored using IOT application	IBM Watson STT service
Application Logic-3	Based on data’s the alerting message will send to the workers for disposing the wastes.	IBM Watso n Assist ant
Database	MySQL is a relationaldatabase that is based on a tabular design. NoSQL is non-relational and has a document-based design	MySQL, NoSQL
Cloud Database	In this module will receive real time status updates from all the bins and continuously display it on web application and also push the notifications on client sides.	IBM DB2, IBM Cloud

File Storage	Data storage makes it easy to back up files for safekeeping and quick recovery in the event of an unexpected computing crash orcyberattack.	IBM Block Storage or Other Storage Service
External API-1	External APIs expose a project's internal resources to outside users or applications	IBM Weather API,etc.
External API-2	External API allow you to access third party resources that are available through RESTful webservices	Aadhar API, etc.
Machine Learning Model	The proper algorithm makes planning good. It will guide the goodness character and which path should be taken and which garbage bin should be collected First	Python IDLE or Anaconda navigatoror Jupitar
Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Cloud Server Configuration: Cloud deployment is the process of deploying an application through one or more hosting models— software as a service (SaaS), platform as a service (PaaS) and or infrastructure as a service (IaaS) that leverage the cloud Local Server Configuration: A local server gives you exclusive access to data and objects in a set of Windows	Cloud server-MySQL Local server-HTTP

Table-2: Application Characteristics:

S. No	Characteristics	Description	Technology
12.	Open-Source Frameworks	Transport, treatment, and disposal of waste together with monitoring and regulation.	Technology of Opensource framework is python.
13.	Security Implementations	Fundamental component of data security that dictates who's allowed to access and use company information and resources. Firewalls use a rule- based access control model with rules expressed in an access control list	Firewall
14.	Scalable Architecture	Using smart waste bins, reduce the number of bins inside town and cities because that we can able to monitor the garbage 24/7. It will be more cost efficient	IoT

15.	Availability	By developing & deploying resilient hardware and beautiful software we empower cities, businesses, and countries to manage waste smarter	IOT, RFID
16.	Performance	The Smart Sensors use ultrasound technology to measure the fill levels (along with other data) in bins several times a day. Using a variety of IoT networks ((NB-IoT, GPRS), the sensors send the data to Sensor's Smart Waste Management Software System, a powerful cloud-based platform, for data- driven daily operations, available also as a waste management app	IOT, GPRS

TECHNOLOGY ARCHITECTURE:

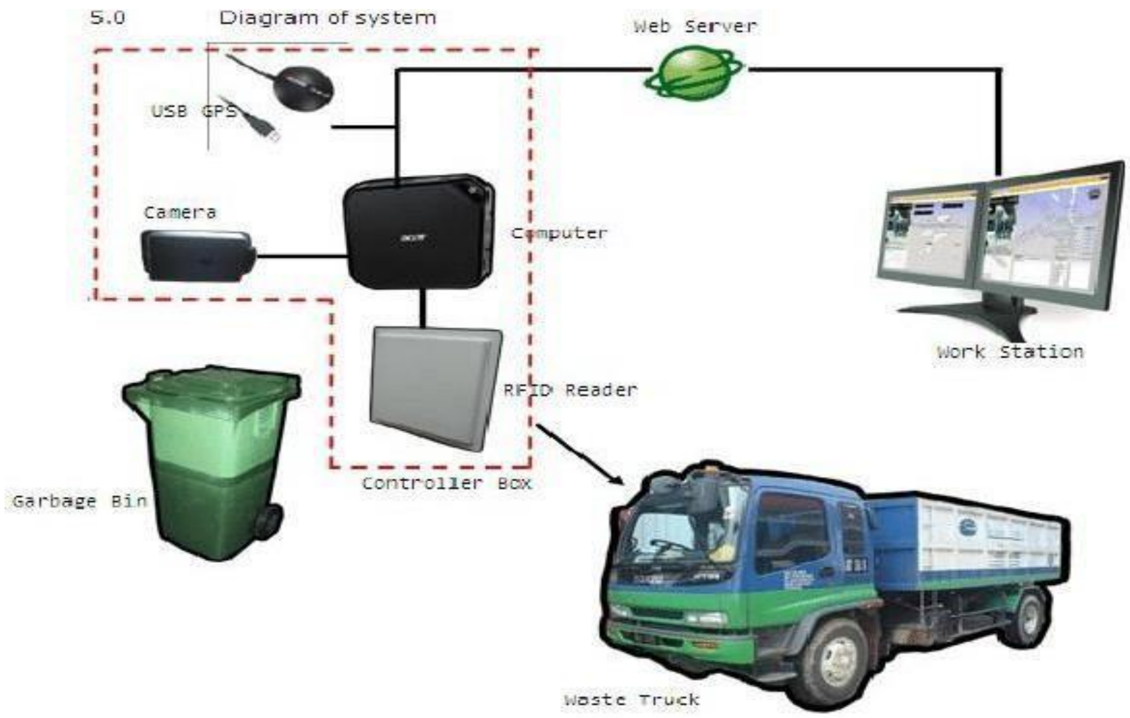


Figure 5.2

5.3. USER STORIES

Use the below template to list all the user stories for the product

Use the below template to list all the user stories for the product

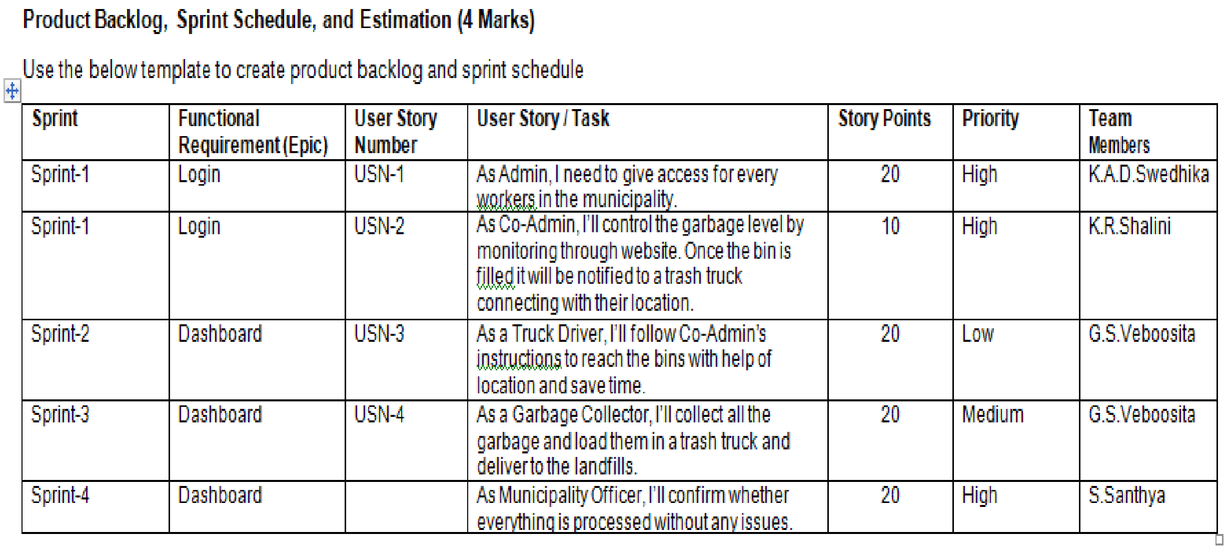
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Admin	Login	USN-1	Admin gives a user id and password for each and every workers and helps to manage	I can access my account / dashboard	Medium	Sprint-2
Assistant Admin	Login	USN-2	They help us to monitor the garbage level once it is filled alert message will be thrown with location	I can manage and monitor the garbage level	High	Sprint-1
Driver	Login	USN-3	They will follow the location	I can drive to reach the	Medium	Sprint-2

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
			where the garbage is filled and collect them in the truck	garbage where it is filled using location and collect them		
Garbage Collector	Login	USN-4	It will collect the trash and load it into the garbage truck and send to landfill	I can collect the trash and load them in truck	Medium	Sprint-2
Government Municipality	Login	USN-5	It will check the process without involving any issues	I can manage the process smoothly	High	Sprint-1

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Product Backlog, Sprint Schedule, and Estimation



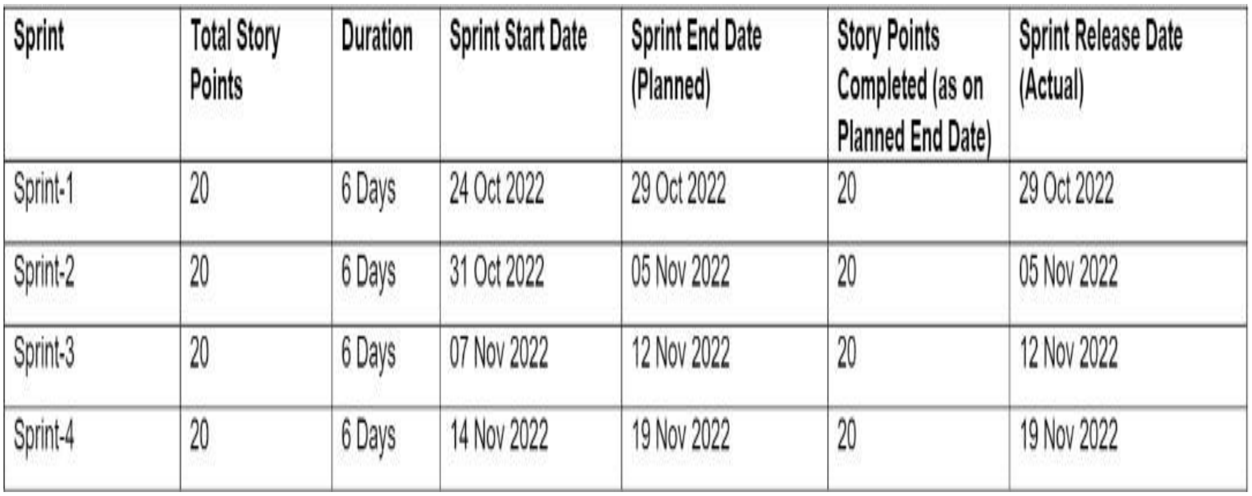
Product Backlog, Sprint Schedule, and Estimation (4 Marks)

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 Admin, I need to give access for every workers in the municipality.	20	High	K.A.D.Swedhika
Sprint-1	Login	USN-2	As Co-Admin, I'll control the garbage level by monitoring through website. Once the bin is filled it will be notified to a trash truck connecting with their location.	10	High	K.R.Shalini
Sprint-2	Dashboard	USN-3	As a Truck Driver, I'll follow Co-Admin's instructions to reach the bins with help of location and save time.	20	Low	G.S.Veboosita
Sprint-3	Dashboard	USN-4	As a Garbage Collector, I'll collect all the garbage and load them in a trash truck and deliver to the landfills.	20	Medium	G.S.Veboosita
Sprint-4	Dashboard		As Municipality Officer, I'll confirm whether everything is processed without any issues.	20	High	S.Santhya

6.2 SPRINT DELIVERY SCHEDULE

Project Tracker, Velocity & Burndown Chart:



Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
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:

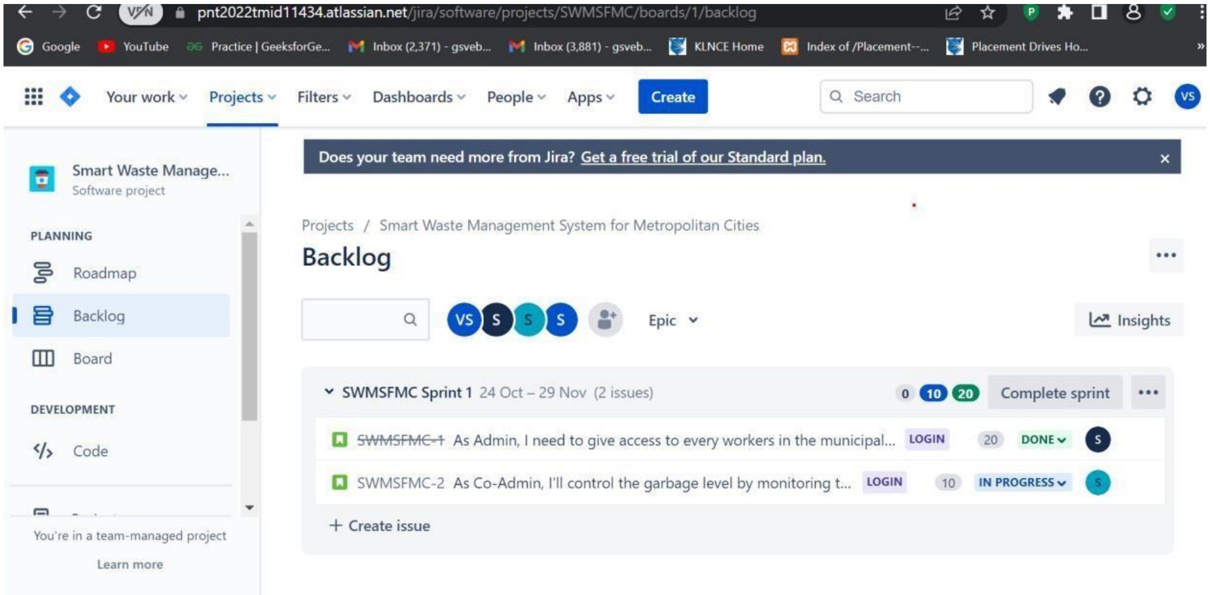
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let’s calculate the team’s average velocity (AV) per iteration unit (story points per day)

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

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

6.3 REPORTS FROM JIRA



Does your team need more from Jira? Get a free trial of our Standard plan.

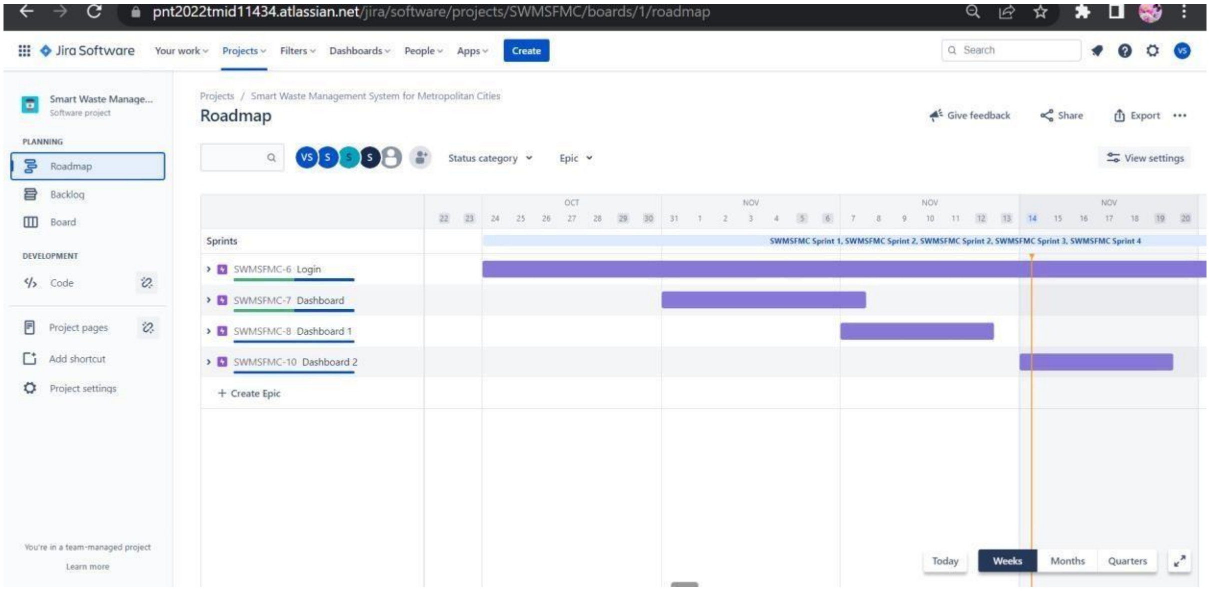
Projects / Smart Waste Management System for Metropolitan Cities

Backlog

SWMSFMC Sprint 1 24 Oct – 29 Nov (2 issues) 0 10 20

~~SWMSFMC-1~~ As Admin, I need to give access to every workers in the municipal...	LOGIN	20	DONE	S
SWMSFMC-2 As Co-Admin, I'll control the garbage level by monitoring t...	LOGIN	10	IN PROGRESS	S

Figure 6.1



Projects / Smart Waste Management System for Metropolitan Cities

Roadmap

Sprints: SWMSFMC Sprint 1, SWMSFMC Sprint 2, SWMSFMC Sprint 2, SWMSFMC Sprint 3, SWMSFMC Sprint 4

- SWMSFMC-6 Login
- SWMSFMC-7 Dashboard
- SWMSFMC-8 Dashboard 1
- SWMSFMC-10 Dashboard 2

Figure 6.2

7.CODING & SOLUTIONING

(EXPLAIN THE FEATURES ADDED IN THE PROJECT WITH CODE)

7.1 FEATURE 1

```
import wiotp.sdk.device
import time
import random
myConfig = {
"identity": {
"orgId": "zal46w",
"typeId": "Dumpster",
"deviceId":"12345"
},
"auth": {
"token": "12345678"
}
}
def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s"
     %cmd.data['command'])
    m=cmd.data['command']
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()
while True:
    temp=random.randint(-20,125)
    hum=random.randint(0,100)
    myData={'temperature':temp, 'humidity':hum}
    client.publishEvent(eventId="status", msgFormat="json", data=myData,
    qos=0,onPublish=None)
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
    time.sleep(2)
     client.disconnect()
```

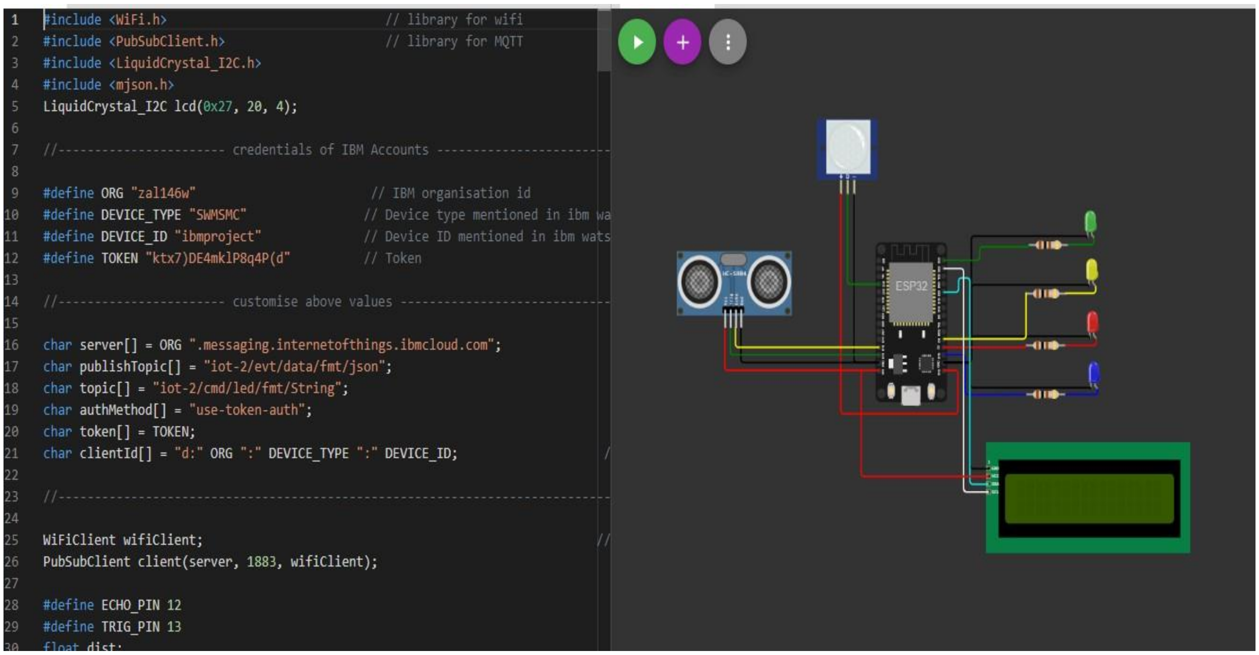
7.2 FEATURE 2

```
   import wiotp.sdk.device
import time
from geopy.geocoders import Nominatim
import random

myConfig = {
 "identity": {
 "orgId": "zal46w",
 "typeId": "Dumpster",
 "deviceId":"12345"},
 "auth": {
 "token": "12345678"
 }
}
id= [0]
geoloc=Nominatim(user_agent="geoapiExercises")

def init():
 lat, long = "9.914470", "78.143418"
 lat1, long1 = "9.9933491", "78.127579"
 lat2, long2 = "9.917916", "78.123496"
 location = geoloc.reverse(lat + "," + long)
 addr = location.raw['address']
 suburb = addr.get('suburb', ")
 city = addr.get('city', ")
 mydata = {'p': {'suburb1': suburb+", "+city, 'suburb2': "Tepakulam, "+city,
'suburb3': "KK Nagar,
"+city,'g_lat1':lat,'g_long1':long,'g_lat2':lat1,'g_long2':long1,'g_lat3':lat2,'g_lo
ng3':long2}}
 client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)


def dumpster_1():
 lat, long = "9.914470", "78.143418"
 location = geoloc.reverse(lat + "," + long)
 addr = location.raw['address']
 suburb = addr.get('suburb', ")
 city = addr.get('city', ")
 level = random.randint(1,100)
 weight = random.randint(1,1000)
 mydata = {'d': {'Level1': level, 'Weight1': weight, 'Lat1': lat, 'Long1':
long,'d_dump1':4}}
 if (level > 50 and weight > 500):
  mydata = {
   'd': {'dump1': dumpid, 'Level1': level, 'Weight1': weight, 'Lat1': lat, 'Long1':
long, 'd_dump1':1,'Suburb1': suburb, 'City1': city}}
client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
  print("pick")
  time.sleep(2)
 client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
 print("dump ", dumpid)
 print("Published data Successfully: %s", mydata)
def dumpster_2():
 lat, long = "9.9933491", "78.127579"
 location = geoloc.reverse(lat + "," + long)
 addr = location.raw['address']
 suburb = "Tepakulam"
 city = addr.get('city', ")
 level = random.randint(1,100)
 weight = random.randint(1,1000)
 mydata = {'d': {'Level2': level, 'Weight2': weight, 'Lat2': lat, 'Long2':
long,'d_dump2':4}}
 if (level > 50 and weight > 500):
  mydata = {
   'd': {'dump2': dumpid, 'Level2': level, 'Weight2': weight, 'Lat2': lat, 'Long2':
long,'d_dump2':2,'Suburb2': suburb, 'City2': city}}
client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
  print("pick")
  time.sleep(2)
 client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
 print("dump ", dumpid)
 print("Published data Successfully: %s", mydata)
```

```
def dumpster_3():
 lat, long = "9.917916", "78.123496"
 location = geoloc.reverse(lat + "," + long)
 addr = location.raw['address']
 suburb = "KK Nagar"
 city = addr.get('city', ")
 level = random.randint(1,100)
 weight = random.randint(1,1000)
 mydata = {'d': {'Level3': level, 'Weight3': weight, 'Lat3': lat, 'Long3':
long,'d_dump3':4}}

 if (level > 50 and weight > 500):
  mydata = {
   'd': {'dump3': dumpid, 'Level3': level, 'Weight3': weight, 'Lat3': lat, 'Long3':
long,'d_dump3':3,'Suburb3': suburb, 'City3': city}}
client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
  print("pick")
  time.sleep(2)
 client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
 print("dump ", dumpid)
 print("Published data Successfully: %s", mydata)


def myCommandCallback(cmd):
 print("Message received from IBM IoT Platform: %s" %
cmd.data['command'])
 m=cmd.data['command']
client = wiotp.sdk.device.DeviceClient(config=myConfig,
logHandlers=None)
client.connect()

 init()
 if dumpid == 1:
  dumpster_1()
  time.sleep(1)
 elif dumpid == 2:
  dumpster_2()
  time.sleep(1)
 elif dumpid==3:
  dumpster_3()
  time.sleep(1)
 mydata = {'d': {'d_dump1': 4}}
 client.publishEvent(eventId="status", msgFormat="json", data=mydata, qos=0, onPublish=None)
 client.commandCallback = myCommandCallback
time.sleep(2)
client.disconnect()
```

8. TESTING

8.1 TEST CASES



```
#include <WiFi.h>                                   // library for wifi
#include <PubSubClient.h>                           // library for MQTT
#include <LiquidCrystal_I2C.h>
#include <mjson.h>
LiquidCrystal_I2C lcd(0x27, 20, 4);

//---------------------- credentials of IBM Accounts ------------------------

#define ORG "zal146w"                             // IBM organisation id
#define DEVICE_TYPE "SWMSMC"                     // Device type mentioned in ibm wa
#define DEVICE_ID "ibmproject"                   // Device ID mentioned in ibm wats
#define TOKEN "ktx7)DE4mklP8q4P(d"               // Token

//---------------------- customise above values ----------------------------

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/data/fmt/json";
char topic[] = "iot-2/cmd/led/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;                         /

//-------------------------------------------------------------------------

WiFiClient wifiClient;                                                           //
PubSubClient client(server, 1883, wifiClient);

#define ECHO_PIN 12
#define TRIG_PIN 13
float dist;
```

REFRENCE:
https://wokwi.com/projects/348774440130052692

8.2 USER ACCEPTANCE TESTING

Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the Smart Waste Management System project at the time of the release to User Acceptance Testing (UAT).

Defect Analysis

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Sub total
By Design	10	4	3	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	78

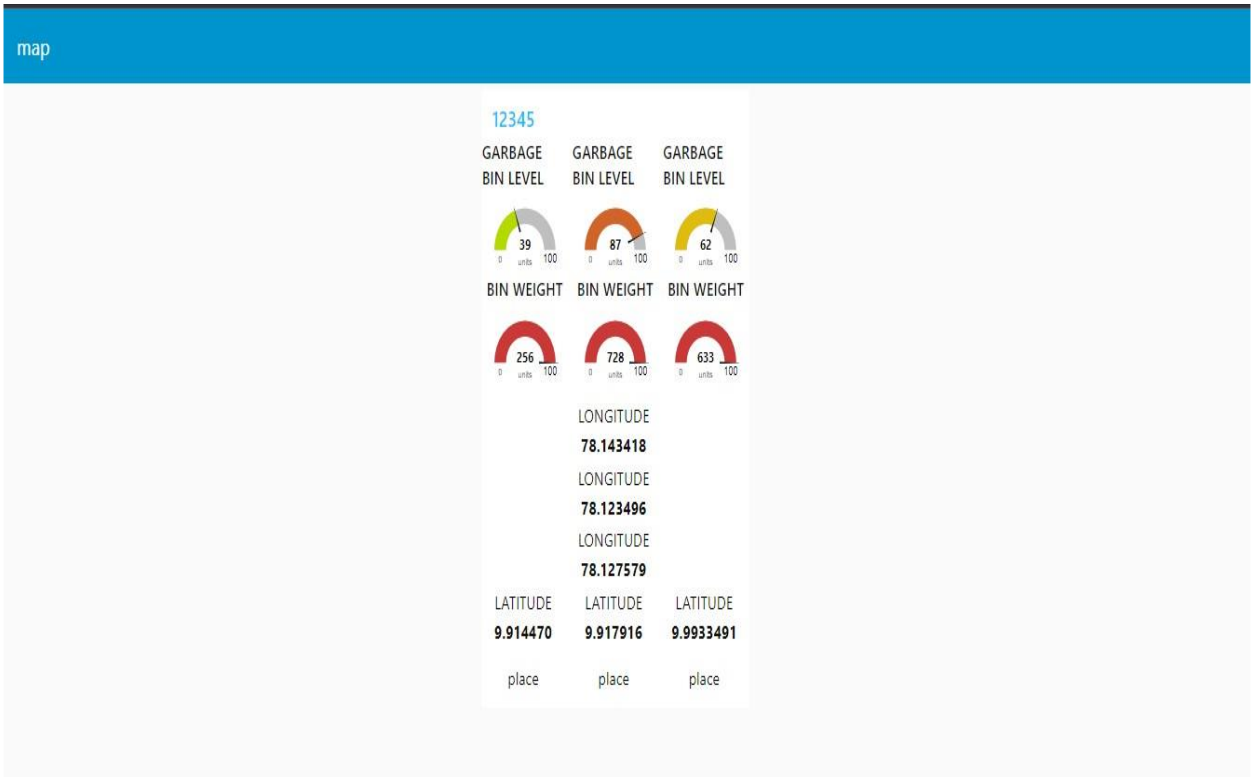
Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS

PERFORMANCE METRICS



map

12345

GARBAGE BIN LEVEL 39

GARBAGE BIN LEVEL 87

GARBAGE BIN LEVEL 62

BIN WEIGHT 256

BIN WEIGHT 728

BIN WEIGHT 633

LONGITUDE 78.143418

LONGITUDE 78.123496

LONGITUDE 78.127579

LATITUDE 9.914470

LATITUDE 9.917916

LATITUDE 9.9933491

place

place

place

Figure9.1

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- It saves time and money by using smart waste collection bins and systems equipped with fill level sensors. As smart transport vehicles go only to the filled containers or bins. It reduces infrastructure, operating and maintenance costs by upto 30%.
- It decreases traffic flow and consecutively noise due to less air pollution as result of less waste collection vehicles on the roads. This has become possible due to two way communication between smart dustbins and service operators.
- It keeps our surroundings clean and green and free from bad odour of wastes, emphasizes on a healthy environment and keep cities more beautiful.
- It further reduces manpower requirements to handle the garbage collection process.
- Applying a smart waste management process to the city optimizes management, resources and costs which makes it a “smart city”.
- It helps administration to generate extra revenue by advertisements on smart devices.

DISADVANTAGES

- System requires more waste bins for separate waste collection as per population in the city. This results in high initial cost due to expensive smart dustbins compared 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.
- It reduces manpower requirements which results in an increase in unemployment for unskilled people.
- The training has to be provided to the people involved in the smart waste management system.

11. CONCLUSION

The behavior 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 practice Recycle, Reuse and Reduce instead of producing waste.Waste disposal should be a priority for municipalities and governments.

12. FUTURE SCOPE

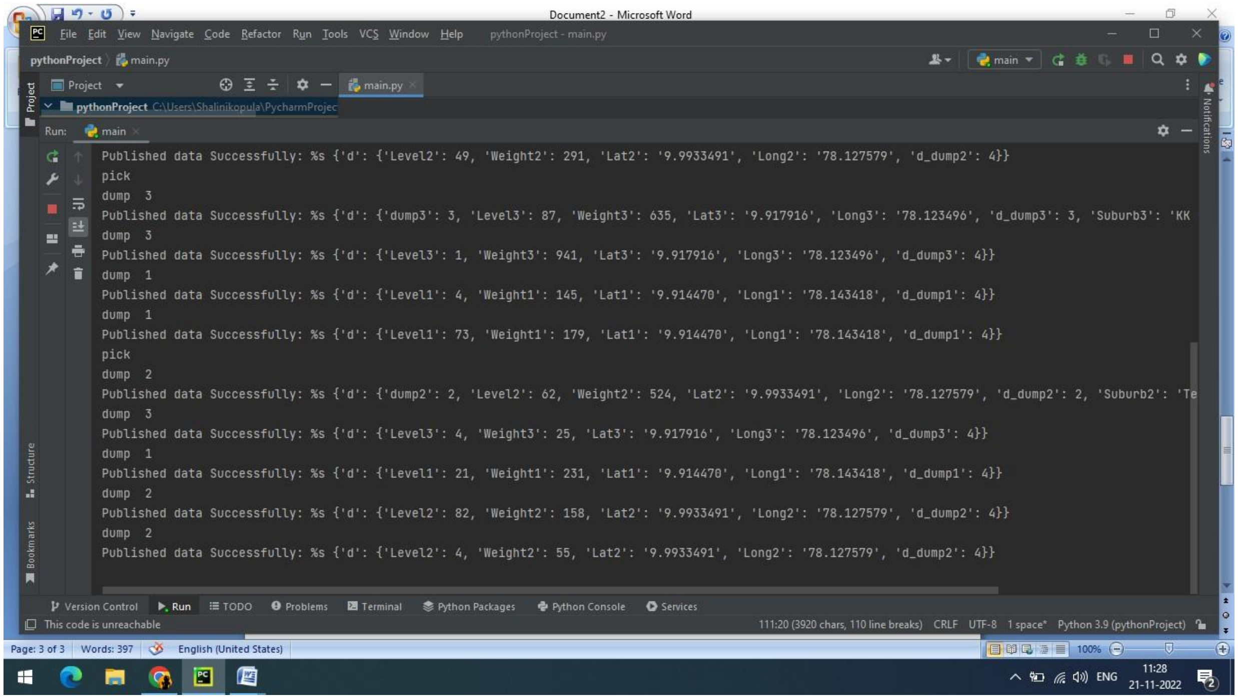
"Global Smart Waste Management System Market Size, Status and Forecast 2025" report provides the newest industry data and industry future trends, allowing you to identify the products and end users driving Revenue growth and profitability. The industry report lists the leading competitors and provides the insights strategic industry Analysis of the key factors influencing the market The Global Smart Waste Management System Market is projected to grow at a healthy growth rate from 2018 to 2025 according to new research. The study focuses on market trends, leading players, supply chain trends, technological innovations, key developments, and future strategies.

13. APPENDIX

13.1. SOURCE CODE

```
import wiotp.sdk.device
import time
from geopy.geocoders import Nominatim
import random

myConfig = {
 "identity": {
 "orgId": "zal46w",
 "typeId": "Dumpster",
 "deviceId":"12345"},
 "auth": {
 "token": "12345678"
 }
}
id= [0]
geoloc=Nominatim(user_agent="geoapiExercises")

def init():
 lat, long = "9.914470", "78.143418"
 lat1, long1 = "9.9933491", "78.127579"
 lat2, long2 = "9.917916", "78.123496"
 location = geoloc.reverse(lat + "," + long)
 addr = location.raw['address']
 suburb = addr.get('suburb', '')
 city = addr.get('city', '')
 mydata = {'p': {'suburb1': suburb+", "+city, 'suburb2': "Tepakulam, "+city,
'suburb3': "KK Nagar,
"+city,'g_lat1':lat,'g_long1':long,'g_lat2':lat1,'g_long2':long1,'g_lat3':lat2,'g_lo
ng3':long2}}
 client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)


def dumpster_1():
 lat, long = "9.914470", "78.143418"
 location = geoloc.reverse(lat + "," + long)
 addr = location.raw['address']
 suburb = addr.get('suburb', ")
 city = addr.get('city', ")
 level = random.randint(1,100)
 weight = random.randint(1,1000)
 mydata = {'d': {'Level1': level, 'Weight1': weight, 'Lat1': lat, 'Long1':
long,'d_dump1':4}}
 if (level > 50 and weight > 500):
  mydata = {
   'd': {'dump1': dumpid, 'Level1': level, 'Weight1': weight, 'Lat1': lat, 'Long1':
long, 'd_dump1':1,'Suburb1': suburb, 'City1': city}}
client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
  print("pick")
  time.sleep(2)
 client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
 print("dump ", dumpid)
 print("Published data Successfully: %s", mydata)
def dumpster_2():
 lat, long = "9.9933491", "78.127579"
 location = geoloc.reverse(lat + "," + long)
 addr = location.raw['address']
 suburb = "Tepakulam"
 city = addr.get('city', ")
 level = random.randint(1,100)
 weight = random.randint(1,1000)
 mydata = {'d': {'Level2': level, 'Weight2': weight, 'Lat2': lat, 'Long2':
long,'d_dump2':4}}
 if (level > 50 and weight > 500):
  mydata = {
   'd': {'dump2': dumpid, 'Level2': level, 'Weight2': weight, 'Lat2': lat, 'Long2':
long,'d_dump2':2,'Suburb2': suburb, 'City2': city}}
client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
  print("pick")
  time.sleep(2)
 client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
 print("dump ", dumpid)
 print("Published data Successfully: %s", mydata)

def dumpster_3():
 lat, long = "9.917916", "78.123496"
 location = geoloc.reverse(lat + "," + long)
 addr = location.raw['address']
 suburb = "KK Nagar"
 city = addr.get('city', '')
 level = random.randint(1,100)
 weight = random.randint(1,1000)
 mydata = {'d': {'Level3': level, 'Weight3': weight, 'Lat3': lat, 'Long3':
long,'d_dump3':4}}

 if (level > 50 and weight > 500):
  mydata = {
   'd': {'dump3': dumpid, 'Level3': level, 'Weight3': weight, 'Lat3': lat, 'Long3':
long,'d_dump3':3,'Suburb3': suburb, 'City3': city}}
client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
  print("pick")
  time.sleep(2)
 client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)
 print("dump ", dumpid)
 print("Published data Successfully: %s", mydata)


def myCommandCallback(cmd):
 print("Message received from IBM IoT Platform: %s" %
cmd.data['command'])
 m=cmd.data['command']
client = wiotp.sdk.device.DeviceClient(config=myConfig,
logHandlers=None)
client.connect()
while True:
 dumpid = random.randint(1,3)
 init()
 if dumpid == 1:
  dumpster_1()
  time.sleep(1)
 elif dumpid == 2:
  dumpster_2()
 time.sleep(1)
 elif dumpid==3:
  dumpster_3()
  time.sleep(1)

 mydata = {'d': {'d_dump1': 4}}
 client.publishEvent(eventId="status", msgFormat="json", data=mydata,
qos=0, onPublish=None)

 client.commandCallback = myCommandCallback
 time.sleep(2)
client.disconnect()
```

OUTPUT



```
Published data Successfully: %s {'d': {'Level2': 49, 'Weight2': 291, 'Lat2': '9.9933491', 'Long2': '78.127579', 'd_dump2': 4}}
pick
dump  3
Published data Successfully: %s {'d': {'dump3': 3, 'Level3': 87, 'Weight3': 635, 'Lat3': '9.917916', 'Long3': '78.123496', 'd_dump3': 3, 'Suburb3': 'KK
dump  3
Published data Successfully: %s {'d': {'Level3': 1, 'Weight3': 941, 'Lat3': '9.917916', 'Long3': '78.123496', 'd_dump3': 4}}
dump  1
Published data Successfully: %s {'d': {'Level1': 4, 'Weight1': 145, 'Lat1': '9.914470', 'Long1': '78.143418', 'd_dump1': 4}}
dump  1
Published data Successfully: %s {'d': {'Level1': 73, 'Weight1': 179, 'Lat1': '9.914470', 'Long1': '78.143418', 'd_dump1': 4}}
pick
dump  2
Published data Successfully: %s {'d': {'dump2': 2, 'Level2': 62, 'Weight2': 524, 'Lat2': '9.9933491', 'Long2': '78.127579', 'd_dump2': 2, 'Suburb2': 'Te
dump  3
Published data Successfully: %s {'d': {'Level3': 4, 'Weight3': 25, 'Lat3': '9.917916', 'Long3': '78.123496', 'd_dump3': 4}}
dump  1
Published data Successfully: %s {'d': {'Level1': 21, 'Weight1': 231, 'Lat1': '9.914470', 'Long1': '78.143418', 'd_dump1': 4}}
dump  2
Published data Successfully: %s {'d': {'Level2': 82, 'Weight2': 158, 'Lat2': '9.9933491', 'Long2': '78.127579', 'd_dump2': 4}}
dump  2
Published data Successfully: %s {'d': {'Level2': 4, 'Weight2': 55, 'Lat2': '9.9933491', 'Long2': '78.127579', 'd_dump2': 4}}
```

Document2 - Microsoft Word
File Edit View Navigate Code Refactor Run Tools VCS Window Help
pythonProject - main.py
pythonProject 〉 main.py
main
Project
main.py
pythonProject C:\Users\Shalinikopula\PycharmProjec
Run: main
Version Control
Run
TODO
Problems
Terminal
Python Packages
Python Console
Services
This code is unreachable
111:20 (3920 chars, 110 line breaks) CRLF UTF-8 1 space* Python 3.9 (pythonProject)
Page: 3 of 3 Words: 397 English (United States)
100%
ENG
11:28
21-11-2022

Figure13.1

13.2 GitHub & Project Demo Link

https://drive.google.com/file/d/1cR1yPIFdvH2livqtwtxSg8sB8vSbBA0W/view?usp=drivesdk