12. FUTURE SCOPE
13. APPENDIX

Source Code

GitHub & Project Demo Link

1. INTRODUCTION

1.1 PROJECT OVERVIEW

Waste bins are part of our lives for decades and mostly its conditions are overflowing due to improper waste dumping, collection and management, which leads in foul smell and unhygienic condition, thus inherently results in environment pollution. Therefore, in this paper, design of a Waste Bin with real time monitoring is presented and a smart waste management system is proposed using the recent technical advancements of automation and Internet of Things (IoT). Th capacitance sensor in the bin continuously monitors the level of the bin in real time and communicates to the central cloud where the bins are connected. Ultrasonic sensor is used to open and close the lid of the bin whenever the persons are nearby the bin. Such smart bins are connected to the cloud, where the bin status are communicated, recorded and monitored by the local bodies through and android app or a centralized server. Thus the designed smart bin and proposed waste management system have better level of smartness compared to existing ones in metropolitan cities in a centralized manner.

1.2 PURPOSE

A waste management system is the strategy an organization uses to dispose, reduce, reuse, and prevent waste. Possible waste disposal methods are recycling, composting, incineration, landfills, bioremediation,waste to energy, and waste minimization.

2. LITERATURE SURVEY

2.1 EXISTING PROBLEMS

1. The process is not always cost-effective
2. The resultant product has a short life
3. The sites are often dangerous
4. The practices are not done uniformly
5. Waste management can cause more problems
6. Garbage segregation is very difficult

2.2 REFERENCES

[1] Prakash , Prabu," IoT Based Waste Management for Smart City", published in IJRCCE Volume 4 , Issue 2 , February 2016.

[2] Tarandeep Singh , Rita Mahajan , Deepak Bagai, "Smart Waste Management using Wireless Sensor Network", in IJRCCE Volume 4 ,Issue 6 , June 2016.

[3] S.S.Navghane, M.S.Killedar, Dr.V.M.Rohokale, " IoT Based Smart Garbage and Waste Collection Bin", IJARECE) Volume 5, Issue 5, May 2016.

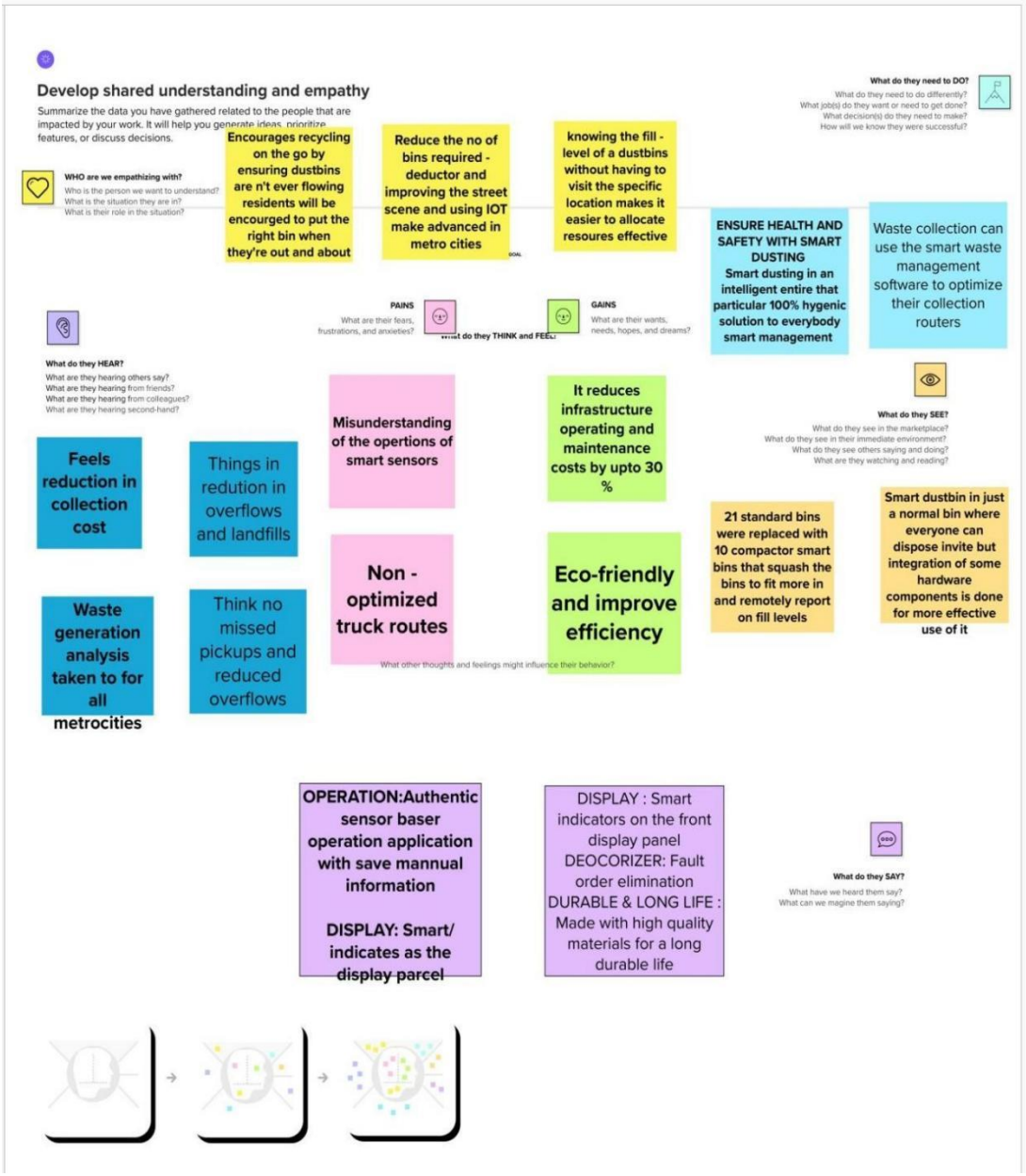
[4] Alexey Medvedev , Pert Fedchenkov,, AArkady Zaslavsky," Waste Management as an IoT Enabled Service in Smart Cities", Springer 2012.

2.3 PROBLEM STATEMENT DEFINITION

In today's world there is no proper management and control system for proper garbage collection. Humans have a tendency to avoid their duty. People in the societies use to throw garbage in filled garbage containers and garbage authorities also do not collect the garbage timely. Hence it leads to various types of pollution and many serious health issues.

3.IDEATIO AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



3.2 PROPOSED SOLUTION

In the case of the proposed solid waste management system, the bins are connect-ed to the internet to relay real-time information of the status of the bin. The rapid growth in population in recent years has led to more waste disposals, necessitating the need for a proper waste management system to avoid unhygienic living conditions. Implementation of the system translates to the bin being interfaced with microcontrol-ler-based system with ultrasonic sensors and a Wi-Fi module. The data which would be sent from the bins would be received, analysed and processed in the ThingSpeak cloud that displays the level of the garbage in the bin on a graph in its web page.

3.3 PROBLEM SOLUTION FIT

Define CS,Fit into CC,Explore AS,Differentiate.

1. CUSTOMER SEGMENTS(S):

- Waste holders, such as private individuals, properly owners or companiesare our customers.

5. AVAILABLE SOLUTIONS(AS):

- Shop eco-friendly with reusable bags.
- Join buy-and-sell groups.
- Digital trash bins are alternative to dustbins, because digital bins candetect the trash levels and send notification to the customers.

6. CUSTOMER:

- As it is technology based it requires internet access to work properly.
- Customers may use solar energy instead of electrical powers.
- Customers need to buy some internet of things(IOT) devices to access.

Focus on J&P,Tap into BE,Understand RC.

2. JOBS-TO-BE-DONE/PROBLEMS(J&P):

- Separate your waste.
- Create your composite waste.
- Growing pressure in outdated waste infrastructure, with declining level of capital investments and maintenance.

7. BEHAVIOUR(BE):

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

9.PROBLEM ROOT CAUSE(RC):

- Lack of industry experts.
- Emission of green house gasses.
- Poor recycling quality due to lack of education.

Identify strong TR &EM.

3. TRIGGERS(TR):

- Seeing how neighbors are having a clean environment after using itpeople will get admire by seeing others.

4.EMOTIONS BEFORE/AFTER(EM):

- Before using this technology, society is suffered by serious health issues.
- After using this technology, they feel good as it provides clean society

8. CHANNELS OF BEHAVIOUR(CH):

ONLINE:

- If it is in online mode,after the bin is getting filled it sends notification message to the authorized persons.

OFFLINE:

- If it is in offline mode,everyday waste collecting trucks will collect the garbage waste from the home.

10. YOUR SOLUTION(SL):

- Our solution is to manage the garbage level and sending indication message to the authorized persons.
- The view of making clean environment. REDUCE-REUSE-RECYCLE[R3]

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

Functional Requirement defines a function of a software system and how the system must behave when presented with specific inputs or conditions. All the conditions may occur in system and these may include calculations, data manipulation and processing and other specific functionality. Some conditions are needed to specify the logical input otherwise the system will not function as per its implementation. In this system following are the functional requirements for the admin and the user.

- The Admin has to login by using valid user name and password.
- After login successful he can do some operations such as add contents, view all contents, list all searching history, list ranking of images, list of all personalized search, attacker details, recover contents, list of all user and logout.
- The admin can add n-number of contents. If the admin want to add a new content, then admin will enter a URL, domain, title, description, uses, related images of the particular content, then submit and that data will stored in data base.
- The Admin can view list of all users. Here button, it will display all personalized search details.
- The time delay Generation chart results. This chart shows the time delay by using greedy DP and time delay using greedy IP.
- The user can attack contents, and then user should enter name to attack, and click on attack button.
- The Attributes are Privacy protection, personalized web search, utility, risk, profile, profile based personalization, Admin, users.
- All register users are stored with the details such as user ID, user name, E mail ID, mobile no, Location, date of birth, address, pin code, general key and personalized key.
- The admin can view the attacker details. If admin clicks on attacker details button, the admin will get attacker information.

- There are n numbers of users are present. User should register before doing some operations. After registration successful he has to login by using authorized user name and password.
- Login successful user will do some operations such as query search, personalized search, attack content details, view comparison graph and logout.
- The user can search query. Before searching any query, the user should request generalization, then proxy server will provide a generalization of profile.
- The user can view the comparison between greedy DP & greedy IL.
- After personalized searching, the greedy IL will be generated. If the user clicks on personalized search.

4.2 NON-FUNCTIONAL REQUIREMENTS

Non – Functional requirements, as the name suggests, are those requirements that are not directly concerned with the specific functions delivered by the system. They may relate to emergent system properties such as reliability response time and store occupancy. Alternatively, they may define constraints on the system such as the capability of the Input Output devices and the data representations used in system interfaces. Many non- functional requirements relate to the system as whole rather than to individual system features. This means they are often critical than the individual functional requirements. The following non-functional requirements are worthyof attention. The key non- functional requirements are

- Security:

The system should allow a secured communication between server, Admin and users.

- Energy Efficiency:

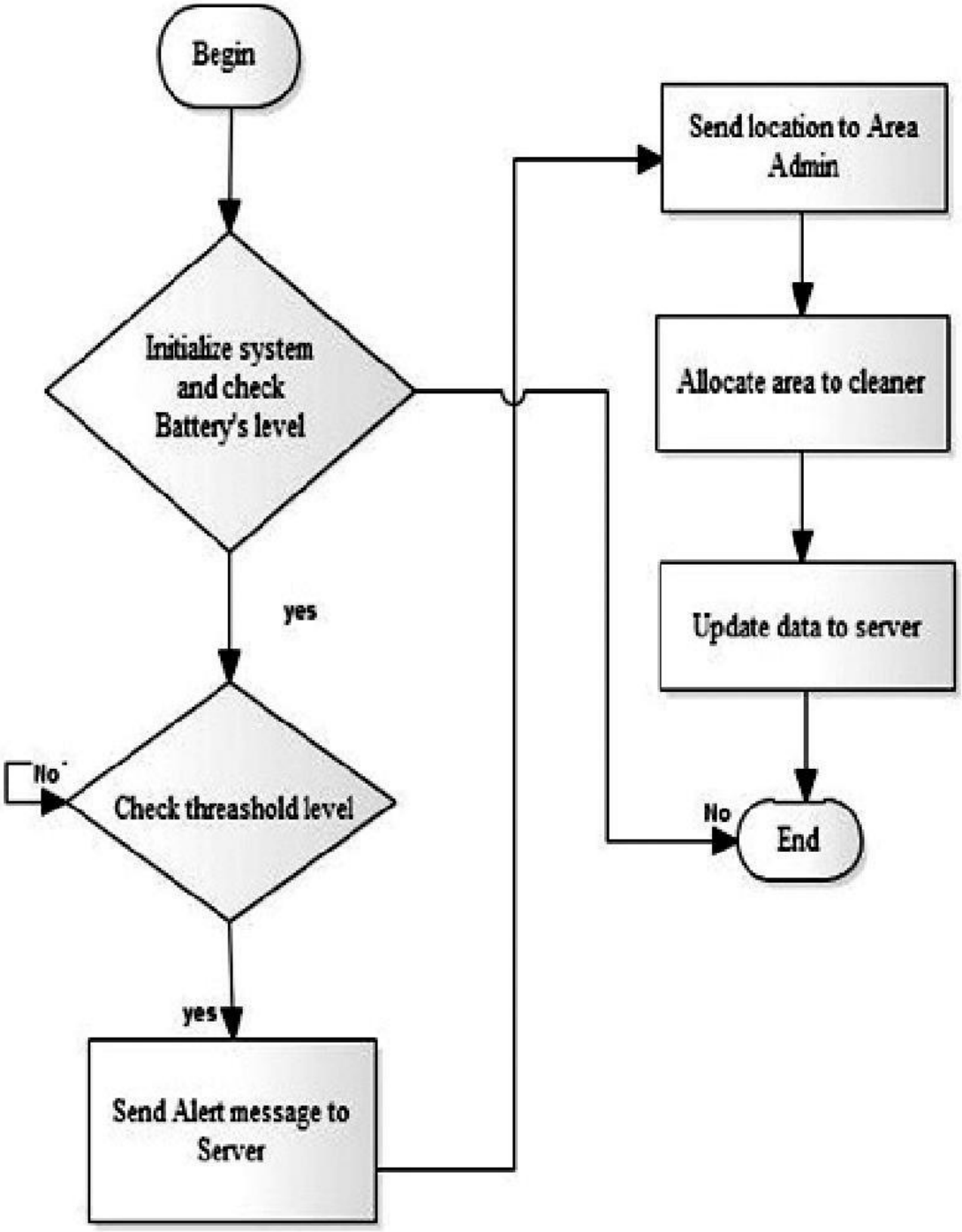
The Energy consumed by the Users to receive the File information from the server and admin

- Reliability:

The system should be reliable and must not degrade the performance ofthe existing system and should not lead to the hanging of the system.

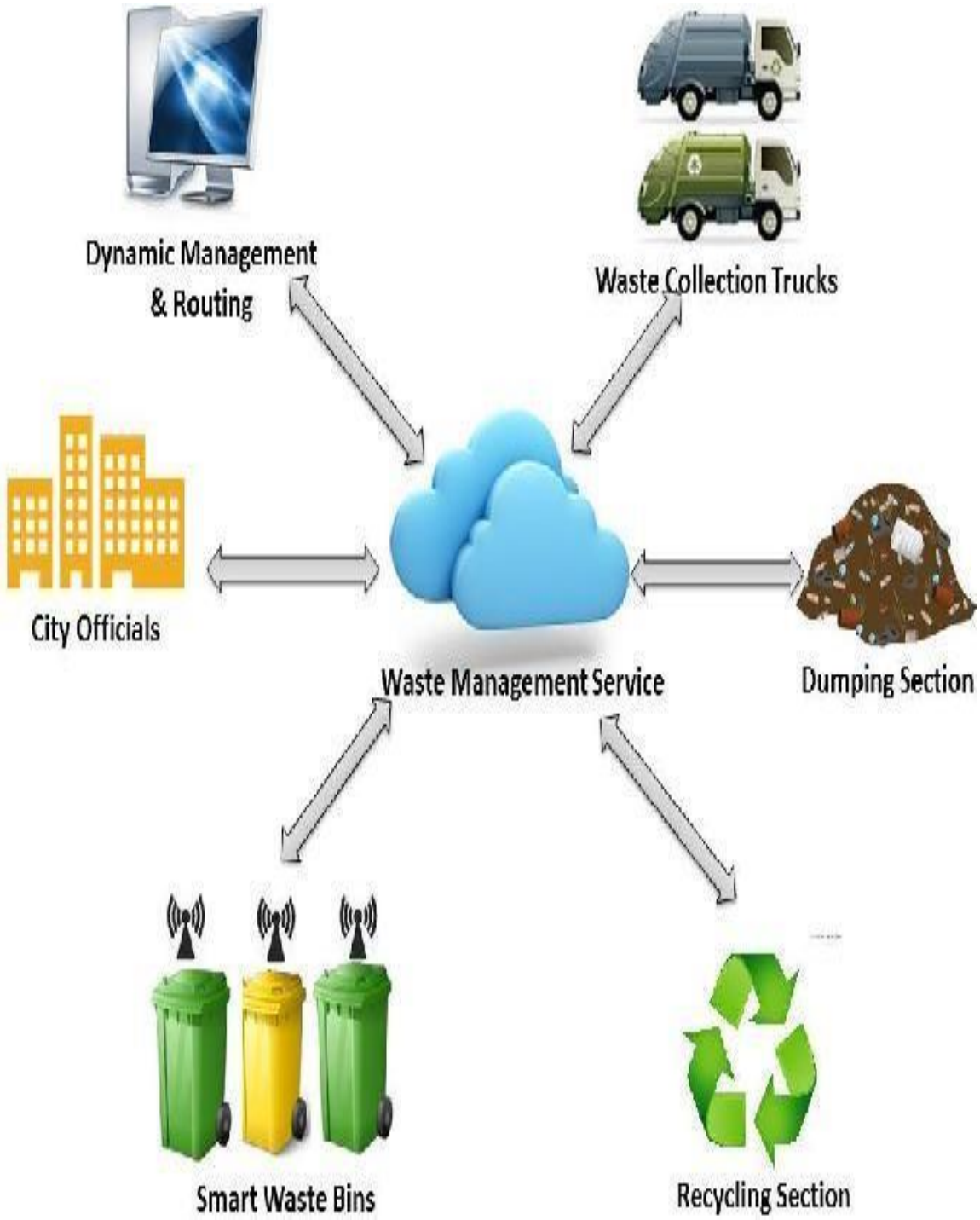
5. PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS



5.2 SOUTION & TECHNICAL ARCHITECTURE

Solution architecture is a complex process – with many sub-processes –that bridges the gap between business problems and technology solutions



5.3 USER STORIES

UserType	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Admin(who manages server)	Webserver login	USN-1	As a admin, I can able to track the truck driver name, id, contact number, location, and also the location of the dustbin.	I can Manage and direct workers through web server	High	Sprint-1
Co-Admin	Login	USN-2	As a co-admin I'll monitor the workers, whether the work has been done properly, checking the availability of workers and also monitor the waste collected by the truck driver within the time	I can monitor the garbage bin activity	High	Sprint-1

Customer (Web user)	User	USN-3	As a user , I can able to raise queries to higher authorities about the maintenance and disposal of waste	I can raise queries	Medium	Sprint-2

6.PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

PHASE	TITLE	DESCRIPTION
Ideation Phase	Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers, research publications etc.
	Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements
	Ideation	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.

Phase-1	Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.
	Problem Solution Fit	Prepare problem - solution fit document.
	Solution Architecture	Prepare solution architecture document.
Phase-2	Customer Journey	Prepare the customer journey maps to understand the user interactions & experiences with the application (entry to exit)
	FunctionalRequirement	Prepare the functional and Nonfunctional requirement document.
	Data Flow Diagrams	Draw the data flow diagrams and submit for review.
	Technology Architecture	Prepare the technology architecture diagram.
Project planning phase	Prepare Milestone & Activity List	Prepare the milestones & activity list of the project.

Project development phase	Project Development - Delivery of Sprint-1, 2, 3 & 4	Develop & submit the developed code by testing it.

6.2 SPRINT DELIVERY SCHEDULE

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**	**Team Members**
Sprint-1	Login	USN-1	Administrator, I need to give user id and passcode for ever workers over there in municipality	10	High	Eswar
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	10	High	Eswari

			location of the bin.			
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	Jevitha
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	Jevitha
Sprint-3	Dashboard	USN-4	As a Local Garbage Collector, I'll gather all the waste from the garbage, load it onto a garbage	20	Medium	Ashley Ebetha

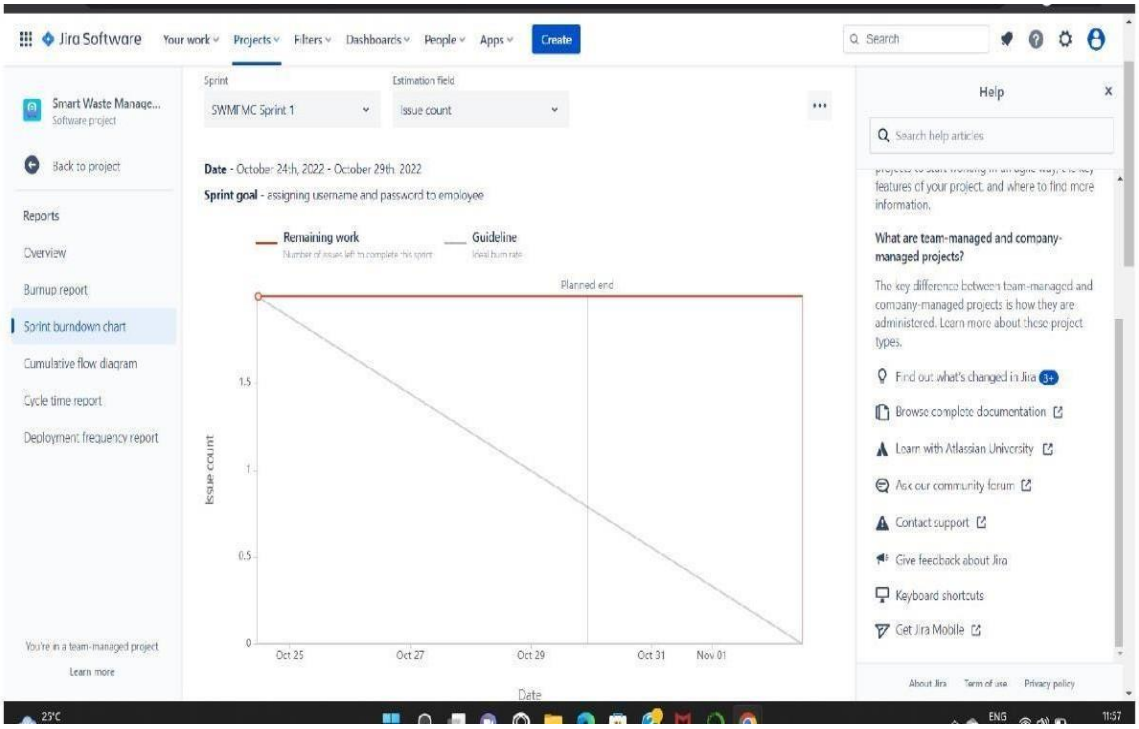
			truck, and deliver it to Landfills			
Sprint-4	Dashboard	USN-5	As a Municipality officer, I'll make sure everything is proceeding as planned and without any problem	20	High	Ashika

PROJECT TRACKER, VELOCITY & BURNDOWN CHART

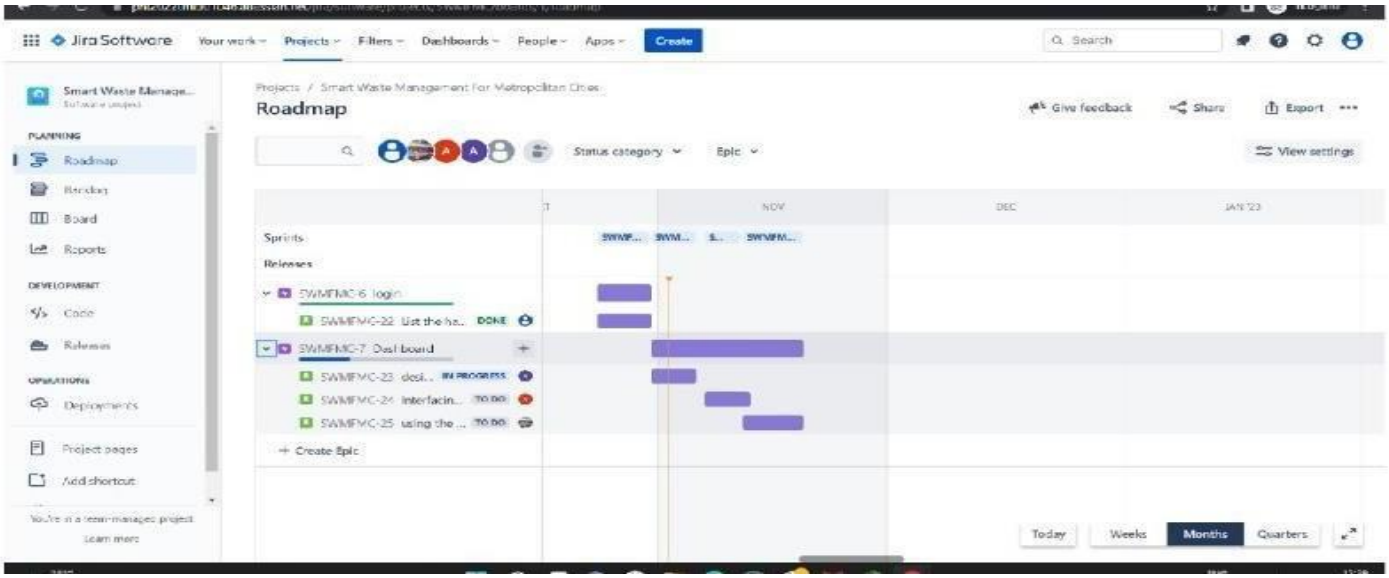
Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date (Actual)**
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days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

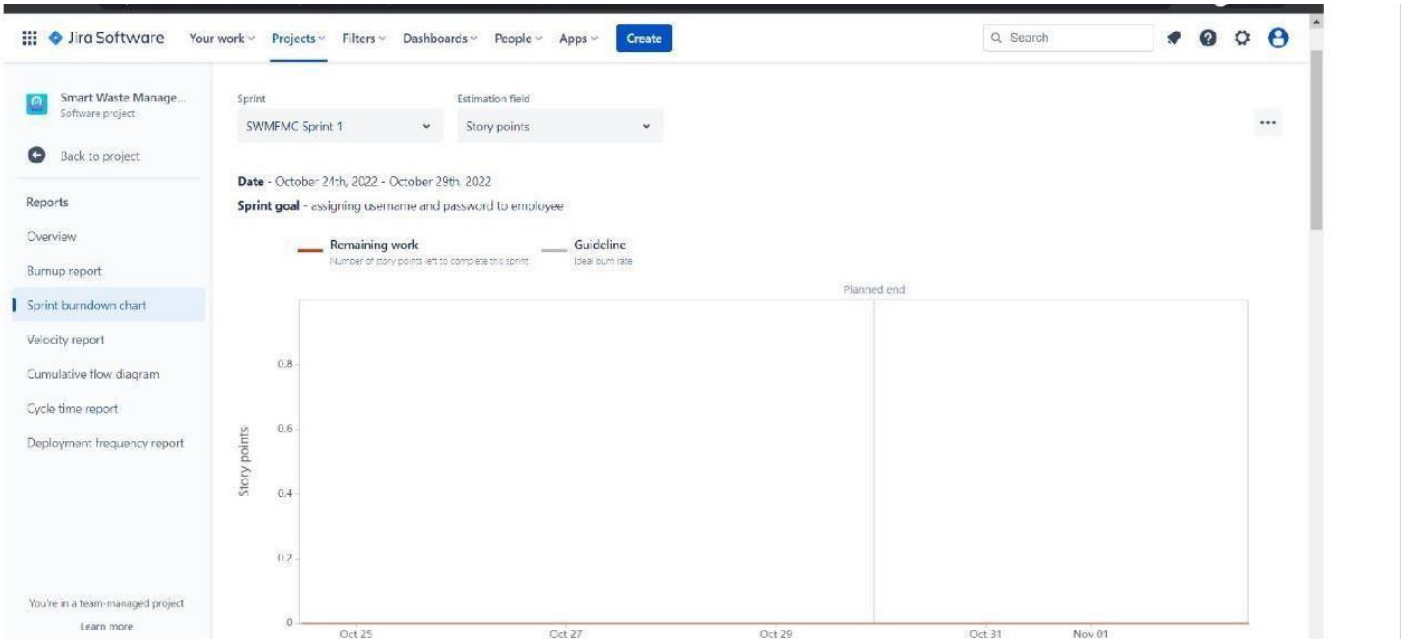
6.3 Reports from JIRA:

Burnout Chart:

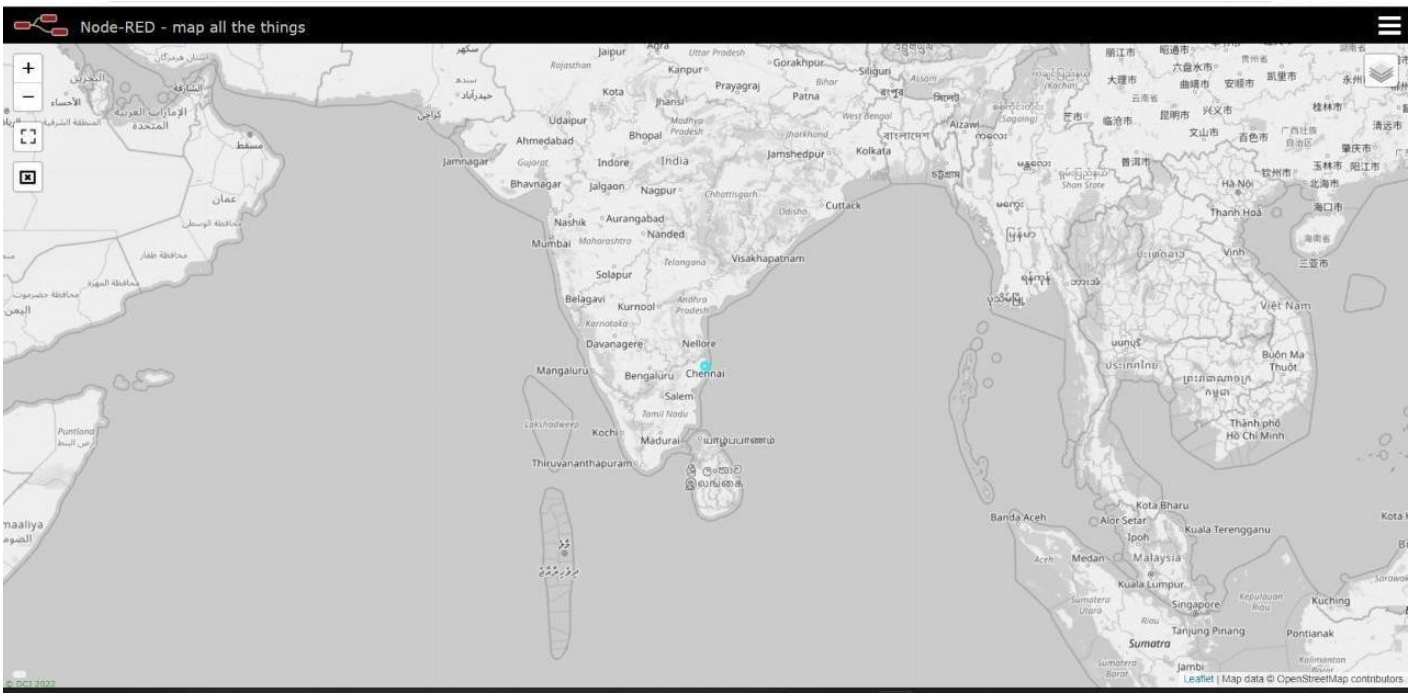


Road Map:

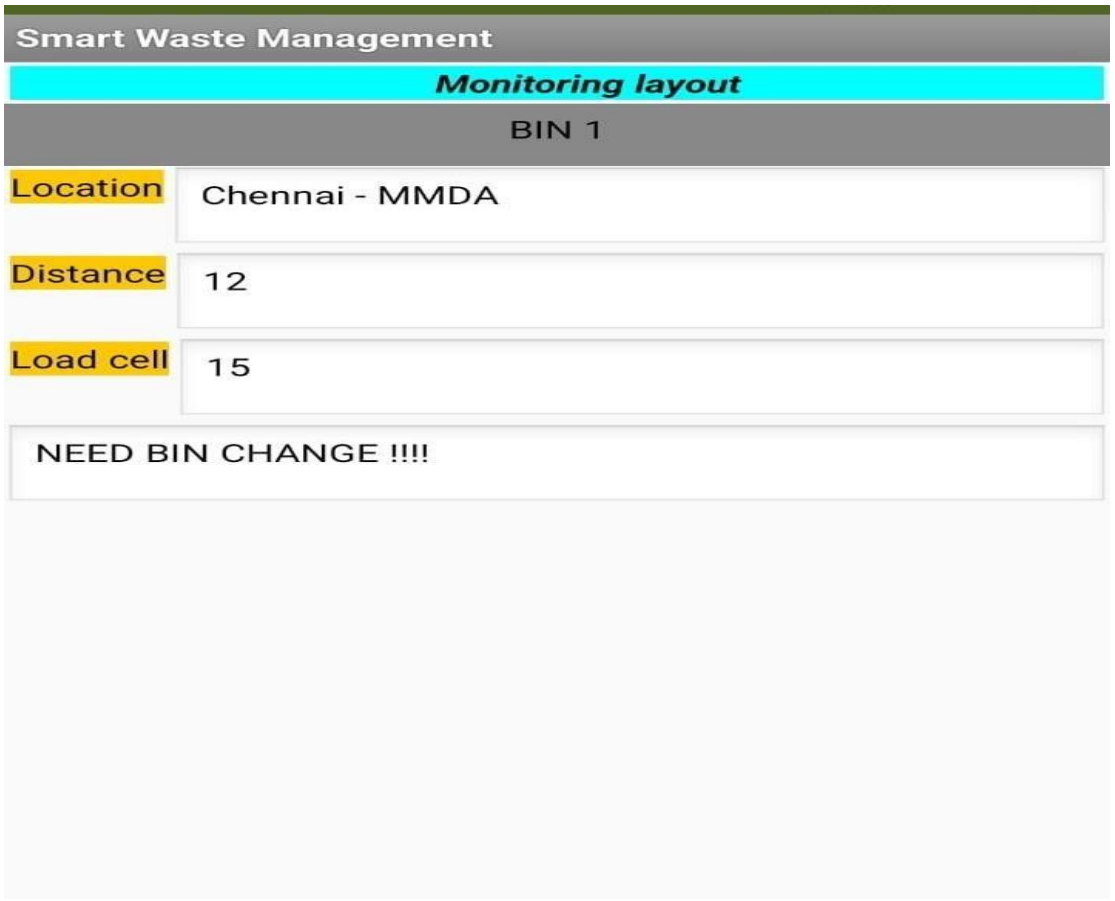




7.1 Feature 1- LOCATION TRACKER:



4.2 Feature 2- LIVE UPDATE ON COLLECTED DATA:



8.TESTING

8.1 TESTCASES:

TESTCASEID	FEATURETYPE	COMPONENT	TESTSCENARIO	PREREQUISTE	STEPS TO EXECUTE	TESTDATA	EXECUTEDRESULT	ACTUALRESULT	STATUS	COMMENTS	TCFORAUTOMATION(Y/N)	BUGID	EXECUTEDBY
LOGINPAGE_TC_001	FUNCTONA L	HOME PAGE	VERIFY THE USER IS ABLE TO SEE THE LOGIN/SIG N UP WEN USER CLICK ON MY ACCOUNT BUTTON		1.ENTER URL AND CLICK GO 2.VERIFY LOGIN/SIGN UP	https://169.51.204.219.30106	L0gin page is visible	Working as expected	PASS	Successful			
LOGINPAGE_TC_002	UI	LOGIN PAGE_TC_002 UI HOME PAGE	VERIFY THE USER IS ABLE TO SEE THE LOGIN/SIG N UP WEN USER CLICK ON MY ACCOUNT BUTTON		1.ENTER URL AND CLICK GO 2.VERIFY LOGIN/SIGN UP Elements a.ID text b o x B.	https://169.51.204.219.30106	Application should show below UI element	Working as expected	PASS	Successfull			

					password text box c..login button D.new user E.already have an account								
LOGIN PAGE_TC_003	FUNCTIONAL	LOGIN PAGE	VERIFY THE USER IS ABLE TO SEE THE LOGIN/SIG N UP WEN USER CLICK ON MY ACCOUNT BUTTON		1.enter url and click go 2.click on my account 3.Enter valid ID 4.Enter valid password 5.click on login	Id:1111 password:5678	User should navigate your home page.	Working as expected	PASS	Successful			
LOGIN PAGE_TC_	FUNCTIONAL	LOGIN	VERIFY THE USER IS ABLE TO SEE THE LOGIN/SIG N UP WEN USER		1.enter url and click go 2.click on my account	Id:1111 password:5678	Confirmation message sent	Working as expected	PASS	Successful			

004			CLICK ON MY ACCOUNT BUTTON		3.Enter valid ID 4.Enter valid password 5.click on login butvton								
LOGINPAGE_TC_005	UI	LOGIN PAGE	VERIFY THE USER IS ABLE TO SEE THE LOGIN/SIG N UP WEN USER CLICK ON MY ACCOUNT BUTTON		1.enter url and click go 2.click on my account 3.Enter valid ID 4.Enter valid password 5.click on login button	Id:1111 password:5678	Confirmation message e sent	Working as expected	PASS	Successful			
LOGINPAGE_TC_00	FUNCTIONAL	LOGIN PAGE FOR ADMIN	VERIFY THE USER IS ABLE TO SEE THE LOGIN/SIG N UP WEN USER CLICK ON		1.enter url and click go 2.click on my account	Id:1111 password:5678	Customer database is visible	Working as expected	PASS	Successful			

6			MY ACCOUNT BUTTON		3.Enter valid ID 4.Enter valid password 5.click on login button								

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 [Product Name] project 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	10	4	2	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	7

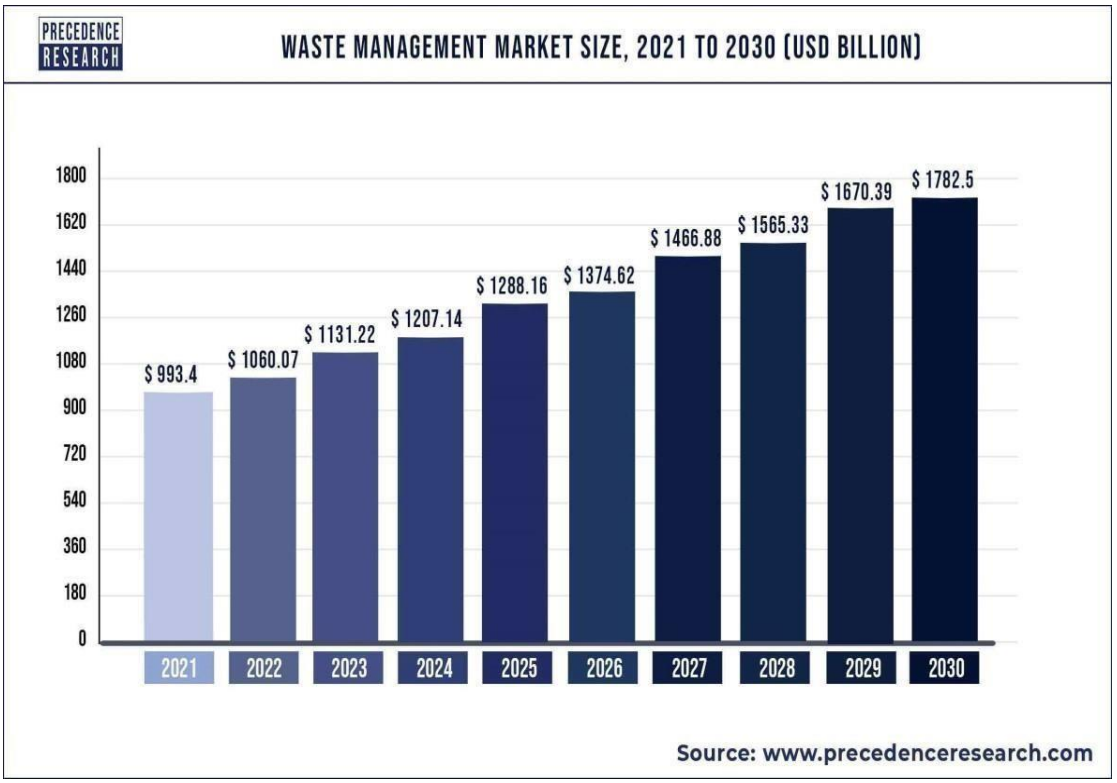
1. TEST CASE ANALYSIS:

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

9.1 PERFORMANCE METRICEES:



10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- Reduction in Collection Cost
- No Missed Pickups
- Reduced Overflows
- Waste Generation Analysis
- CO2 Emission Reduction

DISADVANTAGES

- System requires a greater 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.

11. CONCLUSION

A Smart Waste Management system that is more effective than the one in use now is achievable by using sensors to monitor the filling of bins. Our conception of a "smart waste management system" focuses on monitoring waste management, offering intelligent technology for waste systems, eliminating human intervention, minimizing human time and effort, and producing a healthy and trash-free environment. The suggested approach can be implemented in smart cities where residents have busy schedules that provide little time for garbage management. If desired, the bins might be put into place in a metropolis where a sizable container would be able to hold enough solid trash for a single unit. The price might be high.

12. FUTURE SCOPE

There are several future works and improvements for the proposed system, including the following:

1. Change the system of user authentication and atomic lock of bins, which would aid in protecting the bin from damage or theft.
2. The concept of green points would encourage the involvement of residents or end users, making the idea successful and aiding in the achievement of collaborative waste management efforts, thus fulfilling the idea of Swachh Bharath.
3. Having case study or data analytics on the type and times waste is collected on different days or seasons, making bin filling predictable and removing the reliance on electronic components, and fixing the coordinates.
4. Improving the Server's and Android's graphical interfaces

13. APPENDIX

13.1 source code:

```
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()
```

```
[
    {
        "id": "6ed1189c17ed0439",
        "type": "tab",
        "label": "Flow 1",
        "disabled": false,
        "info": "",
        "env": []
    },
    {
        "id": "ce1790a002f55f3a",
```

```
"type": "ibmiot in",
"z": "6ed1189c17ed0439",
"authentication": "apiKey",
"apiKey": "bf9996433728395e",
"inputType": "evt",
"logicalInterface": "",
"ruleId": "",
"deviceId": "BIN1ID",
"applicationId": "",
"deviceType": "BIN1",
"eventType": "+",
"commandType": "",
"format": "json",
"name": "IBM IoT",
"service": "registered",
"allDevices": "",
"allApplications": "",
"allDeviceTypes": false,
"allLogicalInterfaces": "",
"allEvents": true,
"allCommands": "",
"allFormats": "",
"qos": 0,
"x": 250,
"y": 180,
"wires": [
    [
        "b678812da97d9d1a",
        "f720c62cad238799",
```

```
            "35b263513ea4f373"
        ]
    ]
},
{
    "id": "b678812da97d9d1a",
    "type": "debug",
    "z": "6ed1189c17ed0439",
    "name": "msg.payload",
    "active": true,
    "tosidebar": true,
    "console": false,
    "tostatus": false,
    "complete": "payload",
    "targetType": "msg",
    "statusVal": "",
    "statusType": "auto",
    "x": 610,
    "y": 180,
    "wires": []
},
{
    "id": "f720c62cad238799",
    "type": "function",
    "z": "6ed1189c17ed0439",
    "name": "Distance 1",
    "func": "msg.payload = msg.payload.dist\nglobal.set('d',msg.payload)\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
```

```
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 430,
    "y": 220,
    "wires": [
        [
            "5dcbaf252dc78b06",
            "b678812da97d9d1a"
        ]
    ]
},
{
    "id": "35b263513ea4f373",
    "type": "function",
    "z": "6ed1189c17ed0439",
    "name": "LOAD cell 1",
    "func": "msg.payload =msg. payload.load\nglobal.set('l', msg.payload)\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 430,
    "y": 300,
    "wires": [
        [
            "b7ac8ba401c6cab8"
        ]
```

```
    ]
},
{
    "id": "5dcbaf252dc78b06",
    "type": "ui_gauge",
    "z": "6ed1189c17ed0439",
    "name": "",
    "group": "f3b64a4198b3c46c",
    "order": 1,
    "width": 4,
    "height": 4,
    "gtype": "gage",
    "title": "Distance 1",
    "label": "Cm",
    "format": "{{value}}",
    "min": 0,
    "max": "100",
    "colors": [
        "#00b500",
        "#e6e600",
        "#ca3838"
    ],
    "seg1": "",
    "seg2": "",
    "className": "",
    "x": 710,
    "y": 240,
    "wires": []
},
```

```
{
    "id": "b7ac8ba401c6cab8",
    "type": "ui_gauge",
    "z": "6ed1189c17ed0439",
    "name": "",
    "group": "f3b64a4198b3c46c",
    "order": 2,
    "width": 4,
    "height": 4,
    "gtype": "gage",
    "title": "LOAD CELL 1",
    "label": "KG",
    "format": "{{value}}",
    "min": 0,
    "max": "100",
    "colors": [
        "#00b500",
        "#e6e600",
        "#ca3838"
    ],
    "seg1": "",
    "seg2": "",
    "className": "",
    "x": 720,
    "y": 300,
    "wires": []
},
{
    "id": "5de18859cabb1a5d",
```

```
    "type": "http in",
    "z": "6ed1189c17ed0439",
    "name": "",
    "url": "/sensor",
    "method": "get",
    "upload": false,
    "swaggerDoc": "",
    "x": 210,
    "y": 420,
    "wires": [
        [
            "80650c336af78c61"
        ]
    ]
},
{
    "id": "5ab7d1be9c4e2831",
    "type": "http response",
    "z": "6ed1189c17ed0439",
    "name": "",
    "statusCode": "",
    "headers": {},
    "x": 710,
    "y": 400,
    "wires": []
},
{
    "id": "80650c336af78c61",
    "type": "function",
```

```
    "z": "6ed1189c17ed0439",
    "name": "function 1",
    "func": "msg.payload = { \"dist\": global.get('d'), \"load\": global.get('l')}\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 460,
    "y": 420,
    "wires": [
        [
            "5ab7d1be9c4e2831"
        ]
    ]
},
{
    "id": "e0022c1a3e189dea",
    "type": "ibmiot in",
    "z": "6ed1189c17ed0439",
    "authentication": "apiKey",
    "apiKey": "bf9996433728395e",
    "inputType": "evt",
    "logicalInterface": "",
    "ruleId": "",
    "deviceId": "BIN2ID",
    "applicationId": "",
    "deviceType": "BIN2",
    "eventType": "+",
```

```
    "commandType": "",
    "format": "json",
    "name": "IBM IoT",
    "service": "registered",
    "allDevices": "",
    "allApplications": "",
    "allDeviceTypes": false,
    "allLogicalInterfaces": "",
    "allEvents": true,
    "allCommands": "",
    "allFormats": "",
    "qos": 0,
    "x": 250,
    "y": 500,
    "wires": [
        [
            "2a22e946c6d5f734",
            "233a55d8b0e40a46",
            "a5ed197df7ced05a"
        ]
    ]
},
{
    "id": "233a55d8b0e40a46",
    "type": "function",
    "z": "6ed1189c17ed0439",
    "name": "Distance 2",
    "func": "msg.payload = msg.payload.dist\nglobal.set('d',msg.payload)\nreturn msg;",
    "outputs": 1,
```

```
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 450,
    "y": 540,
    "wires": [
        [
            "2a22e946c6d5f734",
            "9b44a1863803e38a"
        ]
    ]
},
{
    "id": "a5ed197df7ced05a",
    "type": "function",
    "z": "6ed1189c17ed0439",
    "name": "LOAD cell 2",
    "func": "msg.payload =msg. payload.load\nglobal.set('l', msg.payload)\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 450,
    "y": 600,
    "wires": [
        [
            "40ccb32035a0f55f"
```

```
        ]
    ]
},
{
    "id": "2a22e946c6d5f734",
    "type": "debug",
    "z": "6ed1189c17ed0439",
    "name": "msg.payload",
    "active": true,
    "tosidebar": true,
    "console": false,
    "tostatus": false,
    "complete": "payload",
    "targetType": "msg",
    "statusVal": "",
    "statusType": "auto",
    "x": 650,
    "y": 480,
    "wires": []
},
{
    "id": "9b44a1863803e38a",
    "type": "ui_gauge",
    "z": "6ed1189c17ed0439",
    "name": "",
    "group": "f3b64a4198b3c46c",
    "order": 4,
    "width": 4,
    "height": 4,
```

```
    "gtype": "gage",
    "title": "Distance 2",
    "label": "Cm",
    "format": "{{value}}",
    "min": 0,
    "max": "100",
    "colors": [
      "#00b500",
      "#e6e600",
      "#ca3838"
    ],
    "seg1": "",
    "seg2": "",
    "className": "",
    "x": 710,
    "y": 540,
    "wires": []
},
{
    "id": "40ccb32035a0f55f",
    "type": "ui_gauge",
    "z": "6ed1189c17ed0439",
    "name": "",
    "group": "f3b64a4198b3c46c",
    "order": 5,
    "width": 4,
    "height": 4,
    "gtype": "gage",
    "title": "LOAD CELL 2",
```

```
    "label": "KG",
    "format": "{{value}}",
    "min": 0,
    "max": "100",
    "colors": [
        "#00b500",
        "#e6e600",
        "#ca3838"
    ],
    "seg1": "",
    "seg2": "",
    "className": "",
    "x": 720,
    "y": 580,
    "wires": []
},
{
    "id": "60298a7291818343",
    "type": "http in",
    "z": "6ed1189c17ed0439",
    "name": "",
    "url": "/sensor",
    "method": "get",
    "upload": false,
    "swaggerDoc": "",
    "x": 190,
    "y": 660,
    "wires": [
        [
```

```
            "616151913ceb65e2"
        ]
    ]
},
{
    "id": "616151913ceb65e2",
    "type": "function",
    "z": "6ed1189c17ed0439",
    "name": "function 2",
    "func": "msg.payload = { \"dist\": global.get('d'), \"load\": global.get('l')}\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 420,
    "y": 660,
    "wires": [
        [
            "332391e22b2af8e8"
        ]
    ]
},
{
    "id": "332391e22b2af8e8",
    "type": "http response",
    "z": "6ed1189c17ed0439",
    "name": "",
    "statusCode": "",
```

```
    "headers": {},
    "x": 670,
    "y": 660,
    "wires": []
},
{
    "id": "4d33e05e616db2bb",
    "type": "ibmiot in",
    "z": "6ed1189c17ed0439",
    "authentication": "apiKey",
    "apiKey": "bf9996433728395e",
    "inputType": "evt",
    "logicalInterface": "",
    "ruleId": "",
    "deviceId": "BIN3ID",
    "applicationId": "",
    "deviceType": "BIN3",
    "eventType": "+",
    "commandType": "",
    "format": "json",
    "name": "IBM IoT",
    "service": "registered",
    "allDevices": "",
    "allApplications": "",
    "allDeviceTypes": false,
    "allLogicalInterfaces": "",
    "allEvents": true,
    "allCommands": "",
    "allFormats": "",
```

```
    "qos": 0,
    "x": 250,
    "y": 760,
    "wires": [
        [
            "c7ddb56ba52e82df",
            "fbf611802e58a9d1",
            "231892da1f5ab0fb"
        ]
    ]
},
{
    "id": "1c11c86fbb36f097",
    "type": "http in",
    "z": "6ed1189c17ed0439",
    "name": "",
    "url": "/sensor",
    "method": "get",
    "upload": false,
    "swaggerDoc": "",
    "x": 190,
    "y": 900,
    "wires": [
        [
            "cdea8fe7bd7a2f5e"
        ]
    ]
},
{
```

```
    "id": "c49cd92e337f886b",
    "type": "debug",
    "z": "6ed1189c17ed0439",
    "name": "msg.payload",
    "active": true,
    "tosidebar": true,
    "console": false,
    "tostatus": false,
    "complete": "payload",
    "targetType": "msg",
    "statusVal": "",
    "statusType": "auto",
    "x": 750,
    "y": 940,
    "wires": []
},
{
    "id": "c7ddb56ba52e82df",
    "type": "debug",
    "z": "6ed1189c17ed0439",
    "name": "msg.payload",
    "active": true,
    "tosidebar": true,
    "console": false,
    "tostatus": false,
    "complete": "payload",
    "targetType": "msg",
    "statusVal": "",
    "statusType": "auto",
```

```
    "x": 690,
    "y": 760,
    "wires": []
},
{
    "id": "71be31afc89560dd",
    "type": "function",
    "z": "6ed1189c17ed0439",
    "name": "Distance 4",
    "func": "msg.payload = msg.payload.dist\nglobal.set('d',msg.payload)\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 470,
    "y": 980,
    "wires": [
        [
            "c49cd92e337f886b",
            "b88ea394cc4571c3"
        ]
    ]
},
{
    "id": "fbf611802e58a9d1",
    "type": "function",
    "z": "6ed1189c17ed0439",
    "name": "Distance 3",
```

```
    "func": "msg.payload = msg.payload.dist\nglobal.set('d',msg.payload)\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 470,
    "y": 780,
    "wires": [
        [
            "c7ddb56ba52e82df",
            "240d2e6c8f487fd8"
        ]
    ]
},
{
    "id": "231892da1f5ab0fb",
    "type": "function",
    "z": "6ed1189c17ed0439",
    "name": "LOAD cell 3",
    "func": "msg.payload =msg. payload.load\nglobal.set('l', msg.payload)\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 470,
    "y": 820,
    "wires": [
```

```
        [
            "e18c17929284e061"
        ]
    ]
},
{
    "id": "a0cbff62cdd2e77c",
    "type": "function",
    "z": "6ed1189c17ed0439",
    "name": "LOAD cell 4",
    "func": "msg.payload =msg. payload.load\nglobal.set('l', msg.payload)\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 470,
    "y": 1020,
    "wires": [
        [
            "c7a15e2a5bf9c2da"
        ]
    ]
},
{
    "id": "cdea8fe7bd7a2f5e",
    "type": "function",
    "z": "6ed1189c17ed0439",
    "name": "function 3",
```

```
    "func": "msg.payload = { \"dist\": global.get('d'), \"load\": global.get('l')}\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 420,
    "y": 900,
    "wires": [
        [
            "9bfb685be1503933"
        ]
    ]
},
{
    "id": "9bfb685be1503933",
    "type": "http response",
    "z": "6ed1189c17ed0439",
    "name": "",
    "statusCode": "",
    "headers": {},
    "x": 690,
    "y": 900,
    "wires": []
},
{
    "id": "240d2e6c8f487fd8",
    "type": "ui_gauge",
    "z": "6ed1189c17ed0439",
```

```
    "name": "",
    "group": "f3b64a4198b3c46c",
    "order": 15,
    "width": 4,
    "height": 4,
    "gtype": "gage",
    "title": "Distance 3",
    "label": "Cm",
    "format": "{{value}}",
    "min": 0,
    "max": "100",
    "colors": [
        "#00b500",
        "#e6e600",
        "#ca3838"
    ],
    "seg1": "",
    "seg2": "",
    "className": "",
    "x": 830,
    "y": 800,
    "wires": []
},
{
    "id": "b88ea394cc4571c3",
    "type": "ui_gauge",
    "z": "6ed1189c17ed0439",
    "name": "",
    "group": "f3b64a4198b3c46c",
```

```
    "order": 18,
    "width": 4,
    "height": 4,
    "gtype": "gage",
    "title": "Distance 4",
    "label": "Cm",
    "format": "{{value}}",
    "min": 0,
    "max": "100",
    "colors": [
        "#00b500",
        "#e6e600",
        "#ca3838"
    ],
    "seg1": "",
    "seg2": "",
    "className": "",
    "x": 750,
    "y": 1000,
    "wires": []
},
{
    "id": "e18c17929284e061",
    "type": "ui_gauge",
    "z": "6ed1189c17ed0439",
    "name": "",
    "group": "f3b64a4198b3c46c",
    "order": 16,
    "width": 4,
```

```
    "height": 4,
    "gtype": "gage",
    "title": "LOAD CELL 3",
    "label": "KG",
    "format": "{{value}}",
    "min": 0,
    "max": "100",
    "colors": [
      "#00b500",
      "#e6e600",
      "#ca3838"
    ],
    "seg1": "",
    "seg2": "",
    "className": "",
    "x": 840,
    "y": 840,
    "wires": []
},
{
    "id": "147146a0342debce",
    "type": "ibmiot in",
    "z": "6ed1189c17ed0439",
    "authentication": "apiKey",
    "apiKey": "bf9996433728395e",
    "inputType": "evt",
    "logicalInterface": "",
    "ruleId": "",
    "deviceId": "BIN4ID",
```

```
    "applicationId": "",
    "deviceType": "BIN4",
    "eventType": "+",
    "commandType": "",
    "format": "json",
    "name": "IBM IoT",
    "service": "registered",
    "allDevices": "",
    "allApplications": "",
    "allDeviceTypes": false,
    "allLogicalInterfaces": "",
    "allEvents": true,
    "allCommands": "",
    "allFormats": "",
    "qos": 0,
    "x": 230,
    "y": 1000,
    "wires": [
        [
            "71be31afc89560dd",
            "a0cbff62cdd2e77c",
            "c49cd92e337f886b"
        ]
    ]
},
{
    "id": "c7a15e2a5bf9c2da",
    "type": "ui_gauge",
    "z": "6ed1189c17ed0439",
```

```
    "name": "",
    "group": "f3b64a4198b3c46c",
    "order": 19,
    "width": 4,
    "height": 4,
    "gtype": "gage",
    "title": "LOAD CELL 4",
    "label": "KG",
    "format": "{{value}}",
    "min": 0,
    "max": "100",
    "colors": [
        "#00b500",
        "#e6e600",
        "#ca3838"
    ],
    "seg1": "",
    "seg2": "",
    "className": "",
    "x": 760,
    "y": 1040,
    "wires": []
},
{
    "id": "3cec67f2e3359287",
    "type": "http in",
    "z": "6ed1189c17ed0439",
    "name": "",
    "url": "/sensor",
```

```
    "method": "get",
    "upload": false,
    "swaggerDoc": "",
    "x": 230,
    "y": 1080,
    "wires": [
        [
            "c08f7bb853942b70"
        ]
    ]
},
{
    "id": "c08f7bb853942b70",
    "type": "function",
    "z": "6ed1189c17ed0439",
    "name": "function 4",
    "func": "msg.payload = { \"dist\": global.get('d'), \"load\": global.get('l')}\nreturn msg;",
    "outputs": 1,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 460,
    "y": 1080,
    "wires": [
        [
            "07773f295c5a783c"
        ]
    ]
```

```
},
{
    "id": "07773f295c5a783c",
    "type": "http response",
    "z": "6ed1189c17ed0439",
    "name": "",
    "statusCode": "",
    "headers": {},
    "x": 670,
    "y": 1080,
    "wires": []
},
{
    "id": "517648d26c93720d",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 3,
    "width": 4,
    "height": 1
},
{
    "id": "e3086afde0717b7f",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 6,
```

```
    "width": 4,
    "height": 1
},
{
    "id": "329502bdd11bd52a",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 7,
    "width": 4,
    "height": 1
},
{
    "id": "7321a3216f223378",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 8,
    "width": 4,
    "height": 1
},
{
    "id": "f830dc54889698c4",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
```

```
    "order": 9,
    "width": 4,
    "height": 1
},
{
    "id": "3e4d0a1c6525c3dd",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 10,
    "width": 4,
    "height": 1
},
{
    "id": "f5fa3cc71404c6c2",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 11,
    "width": 4,
    "height": 1
},
{
    "id": "eacd5aaaa83c8e0a",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
```

```
    "group": "f3b64a4198b3c46c",
    "order": 12,
    "width": 4,
    "height": 1
},
{
    "id": "e9c145684f7c3c5b",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 13,
    "width": 24,
    "height": 1
},
{
    "id": "4bf3afc9ad5605e6",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 14,
    "width": 24,
    "height": 1
},
{
    "id": "84a1cdcb1acbd2ce",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
```

```
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 17,
    "width": 4,
    "height": 1
},
{
    "id": "8e3c9b05b4f4659f",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 20,
    "width": 4,
    "height": 1
},
{
    "id": "d3debe038a3f028a",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 21,
    "width": 4,
    "height": 1
},
{
    "id": "cdf1f0030c949ca8",
    "type": "ui_spacer",
```

```
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 22,
    "width": 4,
    "height": 1
},
{
    "id": "28a4dbc012c368f4",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 23,
    "width": 4,
    "height": 1
},
{
    "id": "bbd70391a5bc8257",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 24,
    "width": 4,
    "height": 1
},
{
    "id": "b94286c6f5fc0e17",
```

```
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 25,
    "width": 4,
    "height": 1
},
{
    "id": "4dcd3b1b88e0508e",
    "type": "ui_spacer",
    "z": "6ed1189c17ed0439",
    "name": "spacer",
    "group": "f3b64a4198b3c46c",
    "order": 26,
    "width": 4,
    "height": 1
},
{
    "id": "bf9996433728395e",
    "type": "ibmiot",
    "name": "Gogul",
    "keepalive": "60",
    "serverName": "",
    "cleansession": true,
    "appId": "",
    "shared": false
},
{
```

```
    "id": "f3b64a4198b3c46c",
    "type": "ui_group",
    "name": "control",
    "tab": "2b19469befff9adb",
    "order": 2,
    "disp": true,
    "width": "24",
    "collapse": false,
    "className": ""
  },
  {
    "id": "2b19469befff9adb",
    "type": "ui_tab",
    "name": "control",
    "icon": "control",
    "disabled": false,
    "hidden": false
  }
```


git hub link: **https://github.com/IBM-EPBL/IBM-Project-48008-1660803856**

Project demo link: **https://drive.google.com/file/d/19nj9-aljLTidUJJEjgdjfRewaTu_Pcvw/view?usp=drivesdk**