SMART WASTE MANAGEMENT SYSTEM FOR METROPOLITAN CITIES

TEAM ID: PNT2022TMID12045
DOMAIN-INTERNET OF THINGS

PROJECT REPORT

Submitted by

VISHNU.K - 1913105

SWETHA.G.A - 1913097

RAAGHAVI.V - 1913071

THARAGA PRASATH - 1913098

In partial fulfillment of the award of the degree

of

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE AND ENGINEERING

K.S.R. COLLEGE OF ENGINEERING

TIRUCHENGODE – 673 215

NOVEMBER 2022

SMART WASTE MANAGEMENT SYSTEM FOR METROPOLITAN CITIES

CONTENTS

ABSTRACT

The main purpose of this project is to design and implement the smart waste management system.In the present day scenario, many times we see that the Garbage bins or Dust bins are placed at public places in the cities are overflowing due to increase in the waste every day.So we are planning to design Garbage Monitoring System using IoT.In this system, the garbage level detection is done and alerts the authorized person to empty the bin whenever the bins are full.The data are transferred to the cloud and from the cloud sends the data to the web application.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 node red

Our proposed model provide real time monitoring to the garbage bins placed in various locations.The garbage bins are build with a sensor module(Ultrasonic sensor) which continuously monitors the garbage bin.Any moment the garbage level passes over the critical level (i.e 80%),the system generates a

notification to the monitoring panel (admin panel /garbage cleaning team) and so cleaning team collects the garbage from the identified garbage bin.

Keywords: IOT,Ultrasonic sensor,cloud,GPS

CHAPTER - 1

INTRODUCTION

1.1 PROJECT OVERVIEW

We frequently observe overloaded trash cans or dustbins in our city, which makes the area unattractive and unsanitary for people to be in while also emitting a foul odour. The authorities are intending to execute a project named IoT Based Smart Garbage and Waste Collection Bins to prevent all such problems. They are turning more and more to smart solutions as a way to deal with budget cuts and aggressive environmental goals as IoT technology and new innovations become commercially available. Cities are experiencing environmental, social, and infrastructure issues as their populations grow, necessitating a new strategy for delivering public services, including garbage management. In order to make our cities livable and future-proof, smart waste management is a critical component of smart cities. Smart Bin sensors, Smart Waste Management Platform, Intelligent Routing, AI Recycling Robots and Container Tracking are efficient solutions to deal with waste fills across cities. Smart waste management offers a wide range of advantages, including increased operational effectiveness, cheaper operating costs, cleaner communities, and smaller carbon footprints along with increased recycling rates. This paper involves reasoning on why smart waste management is so important since the traditional waste management systems are ill-equipped to handle the additional trash produced by expanding populations, and the trash problem across the world isn't going away any time soon. Communities must embrace smart waste management technology that boosts productivity, cut collection costs, and divert more garbage from landfills in order to close the gap.

1.2 PURPOSE

The major purpose of this proposed system is to stop the dustbin overflowing around the road side and localities as smart bins are used in real time. This can create the surrounding area a

neat and clean while eliminating the overflow of the wastage outside the dustbin due to regular cleaning of the dustbin. Our purpose for this project are as follows:

Sensor based smart waste dustbin will judge the level of waste in it and send the message directly to the municipal corporation.

According to the filled level of the smart waste dustbin, the vehicles from the municipal corporation will choose the shortest path which will save their time.

The system is simple. If there is any problem with any equipment in the future, that part is easily replaceable with new one without any difficulty and delay.

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING PROBLEM

Manual collection of garbage is done former days which makes difficult to the municipal and garbage collector. Even if robot is used in the process of cleaning , it is not implemented in most places because the process is costlier.

2.2 REFRENCE

1] T. Anagnostopoulos, A. Zaslavsky, “Robust Waste Collection exploiting Cost Efficiency of IoT potentiality in Smart Cities”, IEEE 1st International Conference on Recent Advances in Internet of Things (RIoT), 2015, pp. 1-6.

[2] T. Anagnostopoulos, A. Zaslavsky, “Effective Waste Collection with Shortest Path Semi-Static and Dynamic Routing”, IEEE 14th International Conference on Next Generation Wired/Wireless Advanced Networks and Systems (NEW2AN) and 7th Conference on ruSMART, 2014, pp. 95-105.

[3] P. Muthukumaran, and S. B. Sarkar, “Solid waste disposal and water distribution system using the mobile adhoc network”, IEEE International Conference on Emerging Trends

inCommunication, Control, Signal Processing & Computing Applications (C2SPCA), 2013

[4] T. Gomes, N. Brito, J. Mendes, J. Cabral, and A. Tavares, “WECO: A wireless platform for monitoring recycling point spots”, IEEE 16th Mediterranean Electro technical Conference (MELECON), 2012, pp. 468-472.

[5] H. Lingling, L. Haifeng, X. Xu, and L. Jian, “An Intelligent Vehicle Monitoring System Based on Internet of Things”, IEEE 7th International Conference on Computational Intelligence and Security (CIS), 2011, pp. 231-233.

[6] P. P. Pereira, J. Eliasson, R. Kyusakov, J. Delsing, A. Raayatinezhad, and M. Johansson, “Enabling Cloud Connectivity for Mobile Internet of Things Applications”, In the Proceedings of the IEEE 7th International Symposium on Service-Oriented System Engineering (SOSE), 2013, pp. 515-526.

[7] J. Li,Y. Zhang, Y. F. Chen, K. Nagaraja, S. Li, and D. Raychaudhuri, “A Mobile Phone Based WSN Infrastructure for IoT over Future Internet Architecture” ,IEEE International Conference onInternet of Things and Cyber, Physical and Social Computing (iThings/CPSCom), 2013, pp. 426-433.

[8] O. Zhou, and X. Xiaopeng, “Research on In-vehicle Bus Network Based on Internet of Things”, IEEE 4th International Conference on Computational and Information Sciences (ICCIS), 2012, pp. 981-984.

[9] F. Reverter, M. Gasulla, and R. Pallas-Areny“Capacitive level sensing for solid-waste collection”, In the Proceedings of IEEE Conference on Sensors, vol. 1, 2003, pp. 7-11.

[10] A. Runka, B. Ombuki-Berman, and M. Ventresca, “Asearch space analysis for the waste collection vehicle routing problem with time windows”, In the Proceedings of the 11th Annual ACM Conference on Genetic and Evolutionary

2.3 PROBLEM STATEMENT DEFINITION

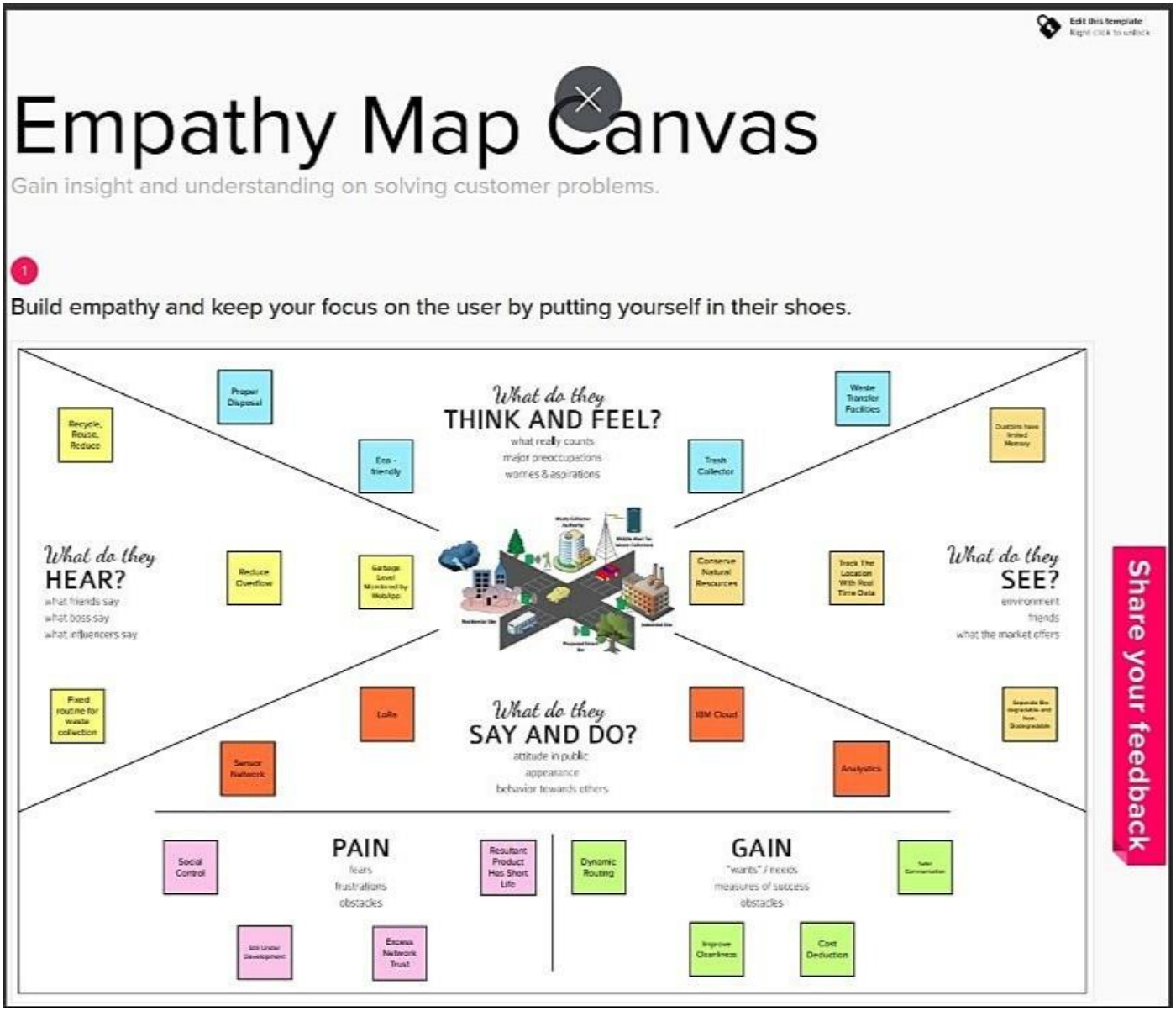
Design a smart waste collection system that allows citizens to segregate the various types of

solid waste they want to dispose and the municipal authorities to efficiently collect the same. The system should be mobile app (Android) based.

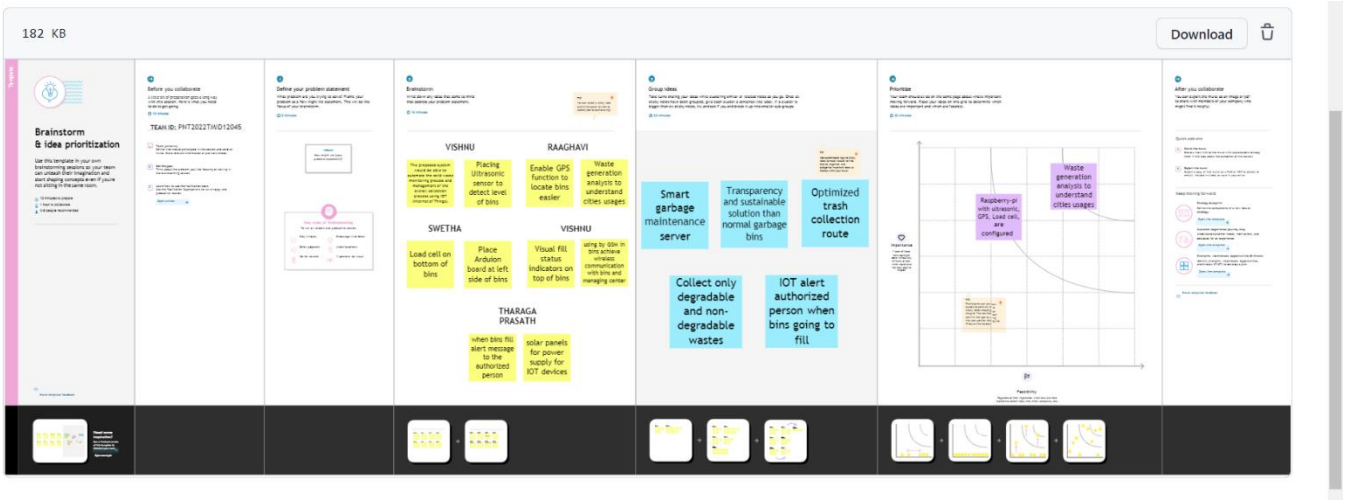
CHAPTER 3

IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



3.2 IDEATION & BRAINSTORMING



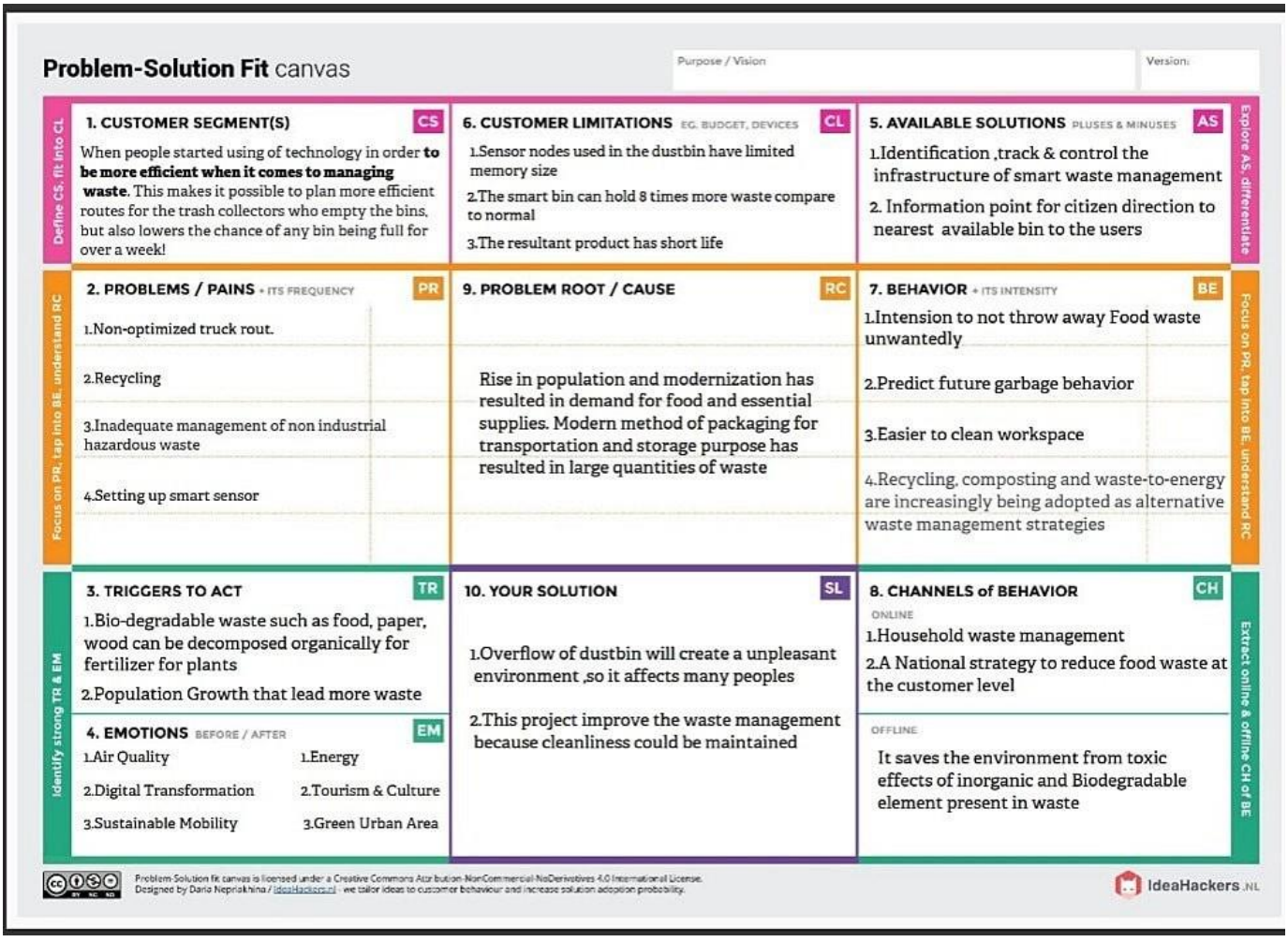
3.3 PROPOSED SOLUTION

S. No	Parameter	Description
1	Problem statement (problem to be solved)	Collection of garbage management in cities, towns and villages is a major concern and emerging problem in smart city paradigm, also lack of proper resource distribution in the process of garbage collection is great risk to sanitation, cleanliness and health.
2	Idea / Solution description	Collecting data from smart bins and alerting to the wastage collector. Data analysis for a smarter collection process. The data gathered in this process is analyzed and it is useful insights empowering the users in their decision making process.
3	Novelty / Uniqueness	Enhances scalability with computing power and storage available on demand, with no management burden.
4	Social Impact / Customer Satisfaction	It will help us to clean the cities and gives us healthy environment. When waste is disposed or recycled in a safe, ethical, and responsible manner, it helps reduce the negative impacts of the environment.

5	Business Model (financial Benefit)	Offering software as a service model to the government. One can earn good revenue from waste management business besides creating positive impacts in the society.
6	Scalability of Solution	In this model costumer gets benefits on using smart bins by providing prediction on day to day analysis in waste management system

This project presents the development of a smart garbage monitoring system in order to measure waste level in the garbage bin in real-time and to alert the authority, in particular cases.

3.4 PROPOSED SOLUTION FIT



CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Bin invention	1. Our proposed model provide real time monitoring to the garbage bins placed in various locations. 2. Overflow of dustbins will be notified.
FR-2	Real time monitoring	1. The garbage bins are monitored by smart sensors. 2. In addition to the percentage of fill-level, based on the historical data, the sensor predicts when the bin will become full, one of the functionalities that are not included even in the best waste management software. 3.With real-time data and predictions, you can eliminate the overflowing bins and stop collecting halfempty ones.
FR-3	Processing	1. Through sensor, the percentage of garbage levels will be detected. 2. When the garbage level moves to critical(i.e.80%), it gives alert notification to the security system. 3. After receiving the notification, the curbside collector collects the garbage.

4.2 NON FUNCTIONAL REQUIREMENT

Following are the non-functional requirements of the proposed solution

.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	A Smart Smart city waste management technology allows crews to empty bins before they become overflowing with trash or recycling, and before infestation becomes an issue.
NFR-2	**Security**	Innovations in waste reduction technologies allow us to better monitor, prevent, and manage our waste. This includes appliances that deal with waste sustainably, smartphone apps to track waste and help us develop eco-friendly habits, and sensors to accurately measure what we have and what we're tossing.
NFR-3	**Reliability**	Smart Bins help to create a cleaner, safer, more hygienic environment and enhanced operational efficiency while reducing management costs, resources, and road-side emissions.
NFR-4	**Performance**	Smart waste management is also about creating better working conditions for waste collectors and drivers. Instead of driving the same collection routes and servicing empty bins, waste collectors will spend their time more efficiently, taking care of bins that need servicing.
NFR-5	**Availability**	The system should be available all the time when required.
NFR-6	**Scalability**	Using smart bin reduces the number of bins inside cities because we able to monitor the garbage 24/7 more efficient and scalability when we move smarter.

CHAPTER 5

PROJECT DESIGN

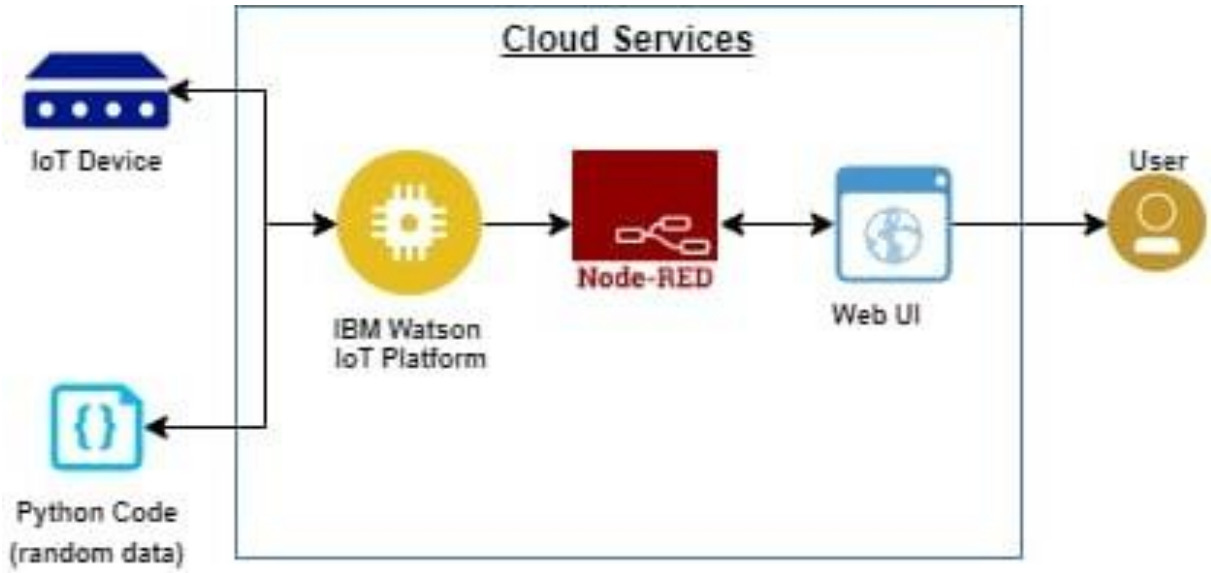
5.1 DATA FLOW DIAGRAM

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

Garbage level detection in bins.Getting the weight of the garbage in the bin. Alerts the authorized person to empty the bin whenever the bins are full.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.



5.3 USER STORIES

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, and password and confirming my password.	3	High	vishnu
Sprint-1		USN-2	As a user, I will receive a confirmation email once I have registered for the application.	2	High	swetha
Sprint-2		USN-3	As a user, I can register for the application through Facebook.	2	Low	raaghavi
Sprint-1		USN-4	As a user, I can register for the application through Gmail.	5	Medium	tharaga Prasath
Sprint 4		USN-5	As a user, I can register for the application through Twitter.	2	Low	vishnu
Sprint-1	Login	USN-6	As a user, I can log into the application by entering my email & password	3	High	swetha
Sprint-2	Dashboard	USN-7	As a user, I need to give permission to access My Contacts, Location, and Storage.	5	High	raaghavi
Sprint-2		USN-8	As a user, I get access to the dashboard which shows a map with marked zones	5	High	tharaga Prasath
Sprint-1	Registration	USN-9	As a management, I need to register my hospitals on the site.	2	high	vishnu

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-3	Login	USN-10	As a management, I need to login into my dashboard with my given hospital id and password.	5	medium	vishnu
Sprint-2	Dashboard	USN-11	As a management, I need to enter the case information of the patient that visits our hospital.	5	high	raaghavi
Sprint-3		USN-12	As a management, I need to store all the patient information on the cloud	5	high	Tharaga Prasath
Sprint-2	Services	USN-13	As an admin, I need to provide valid information about the pandemic out there.	5	high	swetha
Sprint-3		USN-14	As an admin, I need to provide medical advice through a chatbot.	5	medium	vishnu
Sprint-3		USN-15	As an admin, I need to provide medical recommendations by collaborating with top hospitals.	5	low	tharaga Prasath
Sprint-4		USN-16	As an admin, I need to alert the user when they enter pandemic zones.	3	Medium	raaghavi
Sprint-3		USN-17	As an admin, I need to provide preventive measures when they travel through it.	5	high	vishnu
Sprint 4		USN-18	As an admin, I need to provide special services for premium users by giving services like monitoring health by their smart bands.	3	low	swetha

Sprint-4	Data collections	USN-19	As an admin, I need to store all the user information on the cloud	5	Medium	tharaga Prasath
Sprint-4		USN-20	As an admin, I need to collect the list of viruses & bacteria present in this world.	5	High	vishnu

CHAPTER 6

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

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	29 Oct 2022	swetha
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	05 Nov 2022	vishnu
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	12 Nov 2022	tharaga Prasath
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	19 Nov 2022	raaghavi

6.2 Sprint Delivery Schedule:

Project Tracker, Velocity:

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	30	30 Oct 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	49	07 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	50	14 Nov 2022

CHAPTER 7

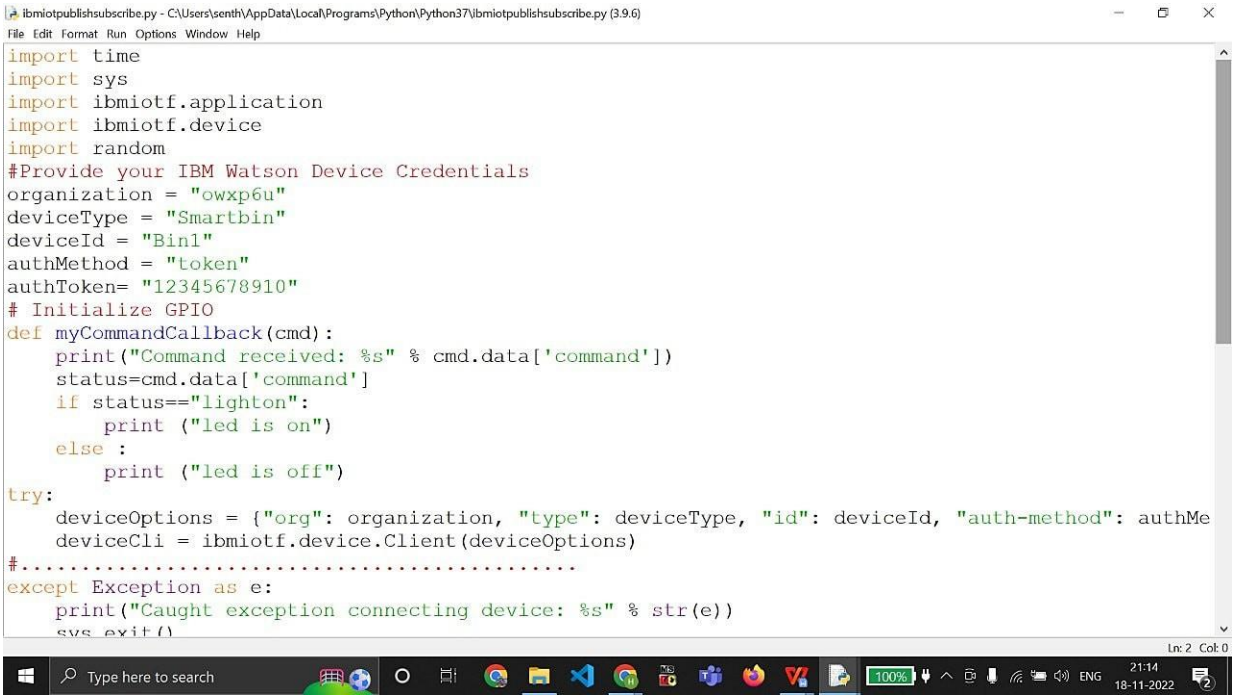
CODING & SOLUTIONING

7.1 FEATURE 1

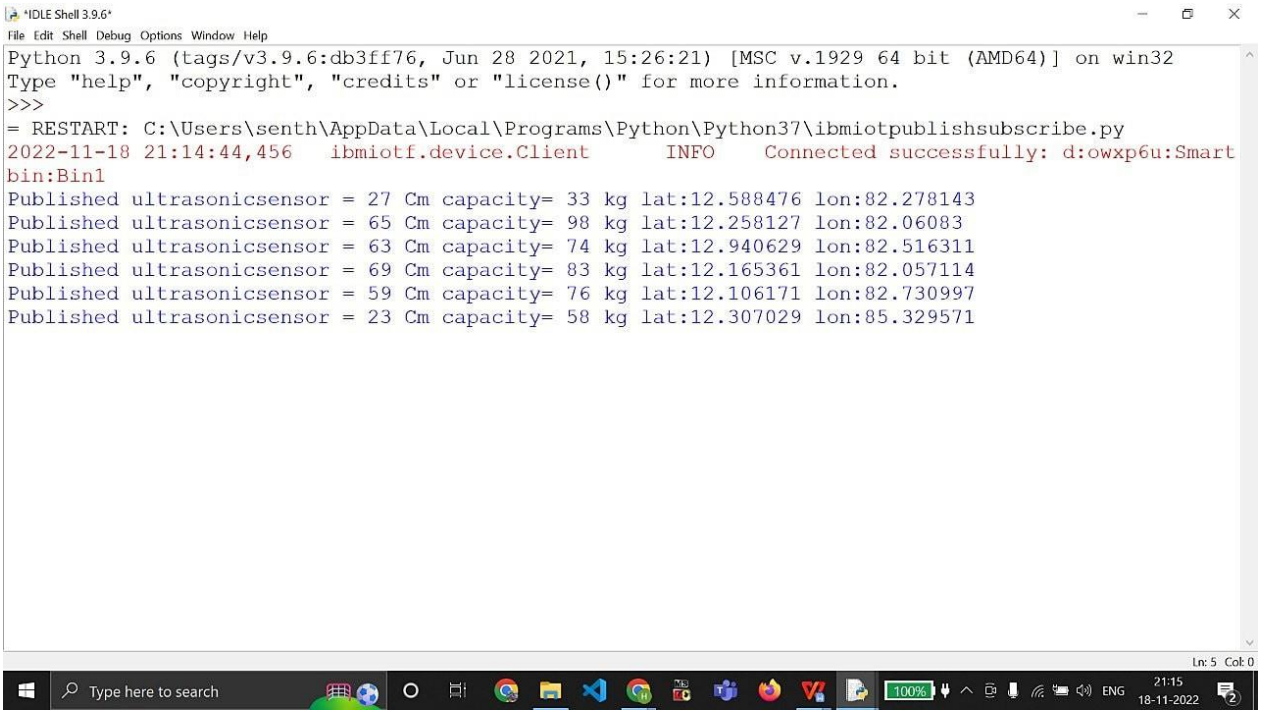
Python code:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "owxp6u"
deviceType = "Smartbin"
deviceId = "Bin1"
authMethod = "token"
authToken= "12345678910"
# Initialize GPIO
def myCommandCallback(cmd):
   print("Command received: %s" % cmd.data['command'])
   status=cmd.data['command']
   if status=="lighton":
      print ("led is on")
   else :
      print ("led is off")
try:
   deviceOptions = {"org": organization, "type": deviceType, "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 "hello" with value "world" into the cloud as an event of type "greeting"
10 times
deviceCli.connect()

while True:
      #Get Sensor Data from DHT11
      time.sleep(5)
      ultrasensor=random.randint(0,80)
      capacity=random.randint(0,100)
      lat=round(random.uniform(12.03,13.05),6)
      lon=round(random.uniform(80.80,85.90),6)
      data = { 'ultrasonicsensor' : ultrasensor, 'capacity': capacity,'lat':lat,'lom':lon}
```

```
    #print data
    def myOnPublishCallback():
        print ("Published ultrasonicsensor = %s Cm" % ultrasensor, "capacity= %s kg" %
capacity,"lat:%s"%lat,"lon:%s"%lon)

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoTF")
    time.sleep(1)

    deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()
```



Python code output:



7.2 FEATURE 2

Wokwi code:

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

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

#define ORG "owxp6u"                              // IBM organisation id
#define DEVICE_TYPE "ESP-32"                      // Device type mentioned in ibm watson iot platform
#define DEVICE_ID "DHT-22"                      // Device ID mentioned in ibm watson  iot  platform
#define TOKEN "harini@2001"                 // Token

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

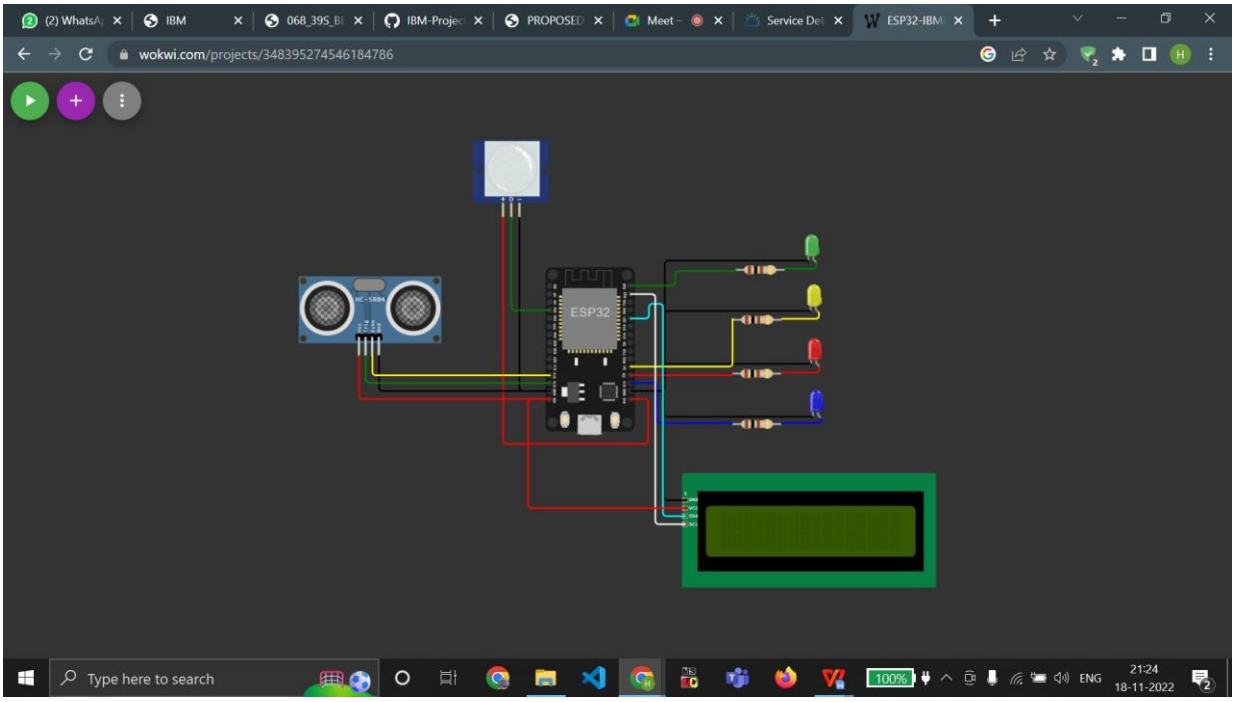
```
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
// server name
char publishTopic[] = "iot-2/evt/data/fmt/json";
// topic name and type of event perform and format in which data to be
send
char topic[] = "iot-2/cmd/led/fmt/String";
// cmd Represent type and command is test format of strings
char authMethod[] = "use-token-auth";
// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
//Client id

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


WiFiClient wifiClient;
// creating instance for wificlient
PubSubClient client(server, 1883, wifiClient);

#define ECHO_PIN 12
#define TRIG_PIN 13
float dist;

void setup()
{
  Serial.begin(115200);
  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
  //pir pin
  pinMode(34, INPUT);

  //ledpins
  pinMode(23, OUTPUT);
  pinMode(2, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(15, OUTPUT);


  lcd.init();
  lcd.backlight();
  lcd.setCursor(1, 0);
  lcd.print("");
  wifiConnect();
  mqttConnect();
}

float readcmCM()
```

```
{
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);
  int duration = pulseIn(ECHO_PIN, HIGH);
  return duration * 0.034 / 2;
}

void loop()
 {

  lcd.clear();

  publishData();
  delay(500);
  if (!client.loop())
    {
      mqttConnect();                                    // function call
to connect to IBM
    }
}


/*.....retrieving to cloud.....
.....*/

void wifiConnect()
{
  Serial.print("Connecting to ");
  Serial.print("Wifi");
  WiFi.begin("Wokwi-GUEST", "", 6);
  while (WiFi.status() != WL_CONNECTED)
    {
      delay(500);
      Serial.print(".");
    }
  Serial.print("WiFi connected, IP address: ");
  Serial.println(WiFi.localIP());
}
void mqttConnect()
  {
    if (!client.connected())
      {
        Serial.print("Reconnecting MQTT client to ");
        Serial.println(server);
        while (!client.connect(clientId, authMethod, token))
          {
```

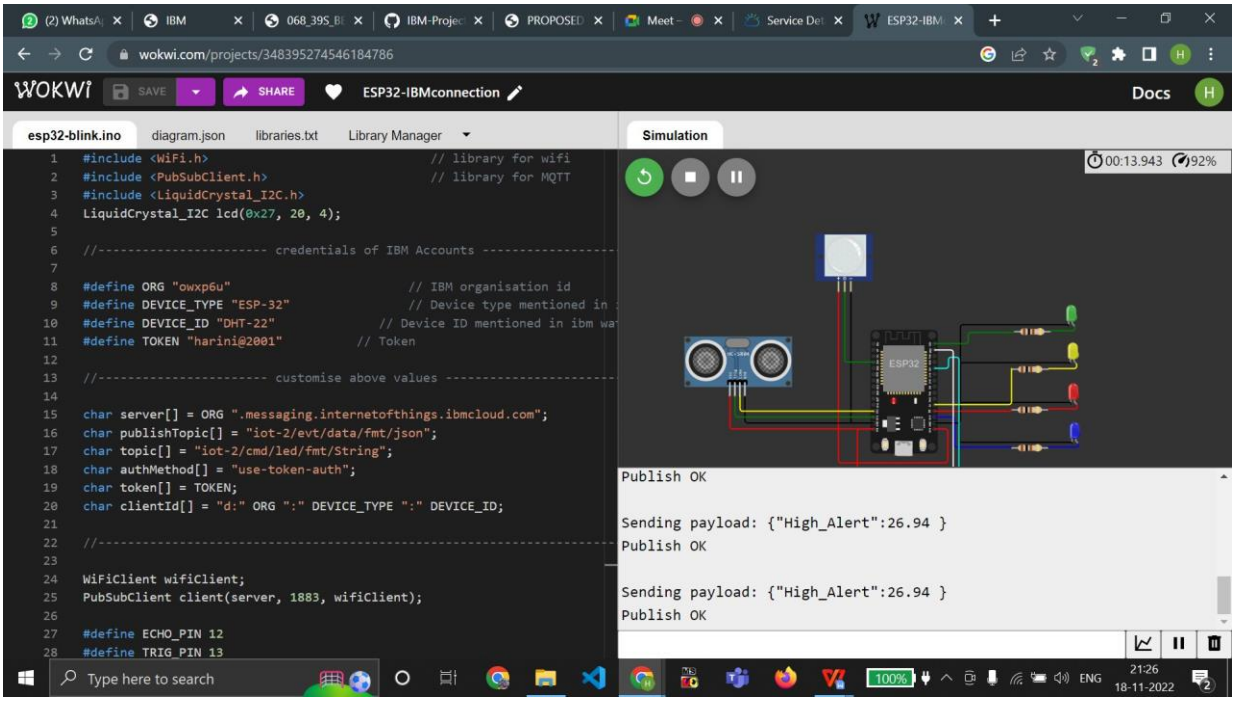
```
            Serial.print(".");
            delay(500);
          }
        initManagedDevice();
        Serial.println();
      }
  }
void initManagedDevice()
  {
    if (client.subscribe(topic))
      {
        Serial.println("IBM subscribe to cmd OK");
      }
    else
      {
        Serial.println("subscribe to cmd FAILED");
      }
  }
void publishData()
{
  float cm = readcmCM();

  if(digitalRead(34))                                    //pir motion
detection
  {
    Serial.println("Motion Detected");
    Serial.println("Lid Opened");
    digitalWrite(15, HIGH);


 if(digitalRead(34)== true)
 {
  if(cm <= 60)                                                    //Bin level
detection
  {
    digitalWrite(2, HIGH);
    Serial.println("High Alert!!!,Trash bin is about to be full");
    Serial.println("Lid Closed");
    lcd.print("Full! Don't use");
    delay(2000);
    lcd.clear();
    digitalWrite(4, LOW);
    digitalWrite(23, LOW);
  }
  else if(cm > 60 && cm < 120)
  {
    digitalWrite(4, HIGH);
    Serial.println("Warning!!,Trash is about to cross 50% of bin level");
```

```
    digitalWrite(2, LOW);
    digitalWrite(23, LOW);

  }
  else if(cm > 120)
  {
    digitalWrite(23, HIGH);
    Serial.println("Bin is available");
    digitalWrite(2,LOW);
    digitalWrite(4, LOW);

  }
    delay(10000);
    Serial.println("Lid Closed");
 }
 else
 {
   Serial.println("No motion detected");
    digitalWrite(2, LOW);
    digitalWrite(15, LOW);
    digitalWrite(4, LOW);
    digitalWrite(23, LOW);
 }


}
  else
  {
    digitalWrite(15, LOW);

  }


  if(cm <= 60)
{
digitalWrite(21,HIGH);
String payload = "{\"High_Alert\":";
payload += cm;
payload += " }";
Serial.print("\n");
Serial.print("Sending payload: ");
Serial.println(payload);

if (client.publish(publishTopic, (char*) payload.c_str()))          // if
data is uploaded to cloud successfully,prints publish ok else prints
publish failed
{
Serial.println("Publish OK");
}
```

```
}
else if(cm <= 120)
{
digitalWrite(22,HIGH);
String payload = "{\"Warning\":";
payload += cm ;
payload += " }";
Serial.print("\n");
Serial.print("Sending payload: ");
Serial.println(payload);
if(client.publish(publishTopic, (char*) payload.c_str()))
{
Serial.println("Publish OK");
}
else
{
Serial.println("Publish FAILED");
}
}
else
{
Serial.println();
}

  float inches = (cm / 2.54);                                    //print
on lcd
  lcd.setCursor(0,0);
  lcd.print("Inches");
  lcd.setCursor(4,0);
  lcd.setCursor(12,0);
  lcd.print("cm");
  lcd.setCursor(1,1);
  lcd.print(inches, 1);
  lcd.setCursor(11,1);
  lcd.print(cm, 1);
  lcd.setCursor(14,1);
  delay(1000);
  lcd.clear();
}
```

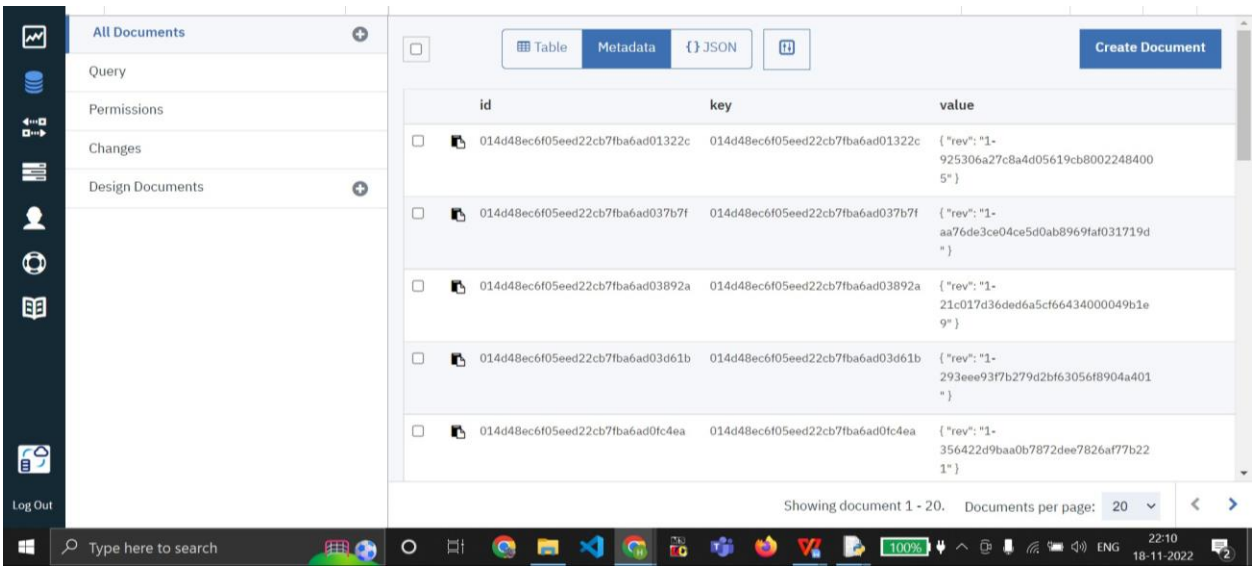

wokwi.com/projects/348395274546184786
ESP32
Type here to search
ENG
21:24
18-11-2022

Wokwi code:

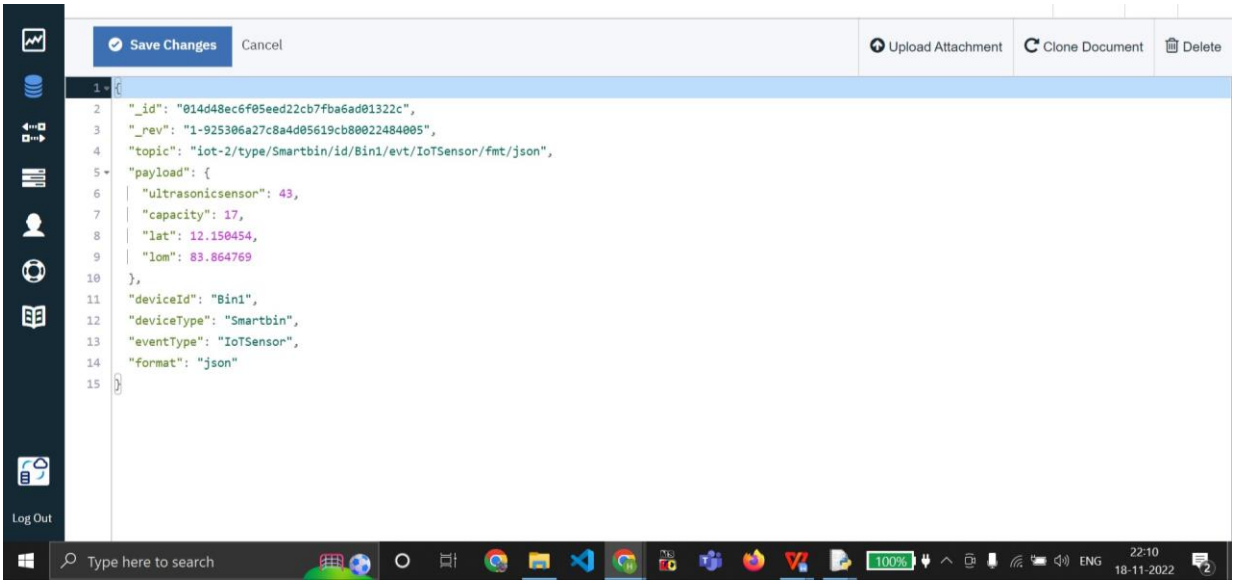

wokwi.com/projects/348395274546184786
WOKWI
SAVE
SHARE
ESP32-IBMconnection
Docs
esp32-blink.ino
diagram.json
libraries.txt
Library Manager
Simulation
00:13.943
92%
#include <WiFi.h> // library for wifi
#include <PubSubClient.h> // library for MQTT
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 20, 4);
//---------- credentials of IBM Accounts ----------
#define ORG "owxp6u" // IBM organisation id
#define DEVICE_TYPE "ESP-32" // Device type mentioned in
#define DEVICE_ID "DHT-22" // Device ID mentioned in ibm wa
#define TOKEN "harini@2001" // 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
ESP32
Publish OK
Sending payload: {"High_Alert":26.94 }
Publish OK
Sending payload: {"High_Alert":26.94 }
Publish OK
Type here to search
ENG
21:26
18-11-2022

7.3 DATABASE SCHEMA

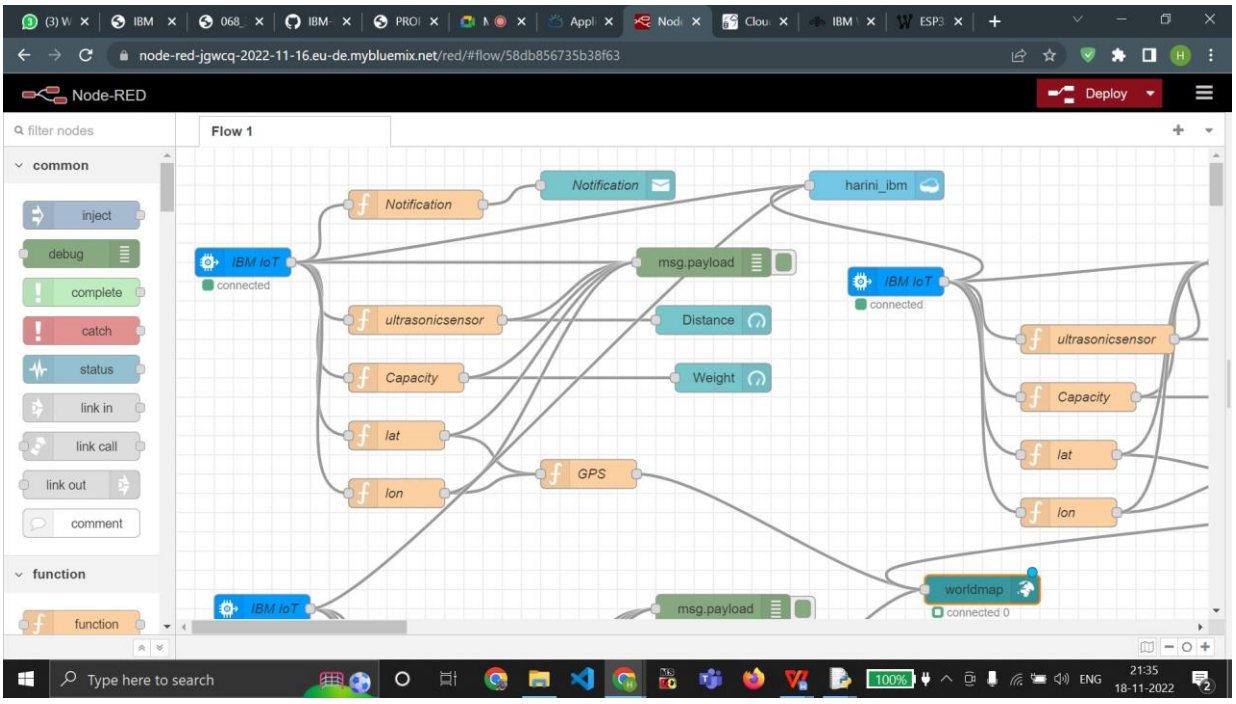
METADATA:



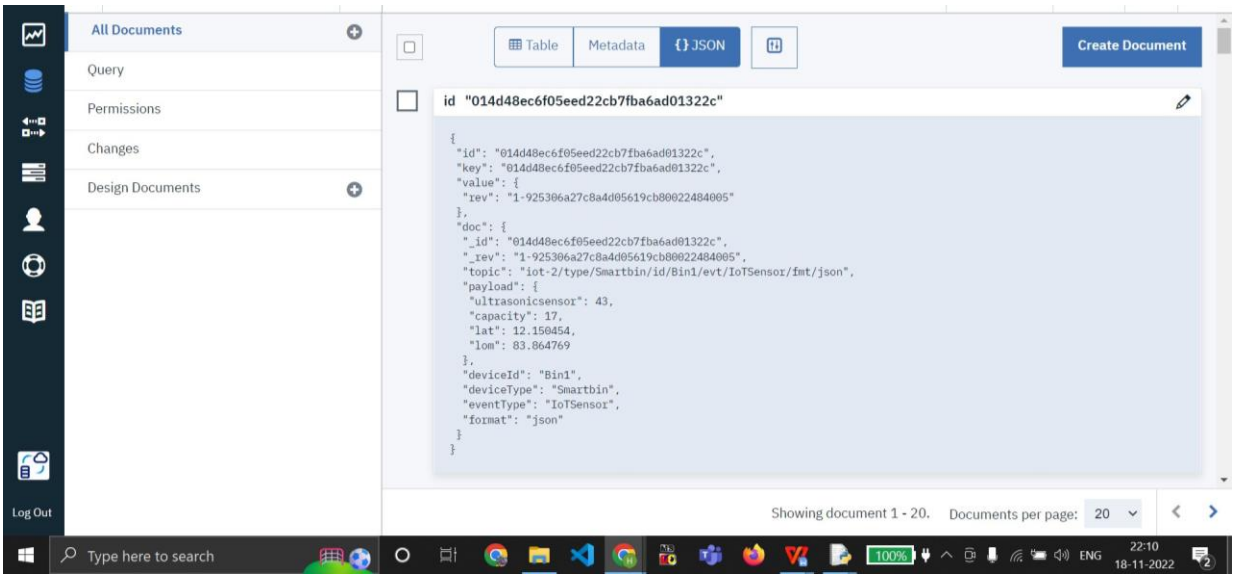
METADATA INFO:



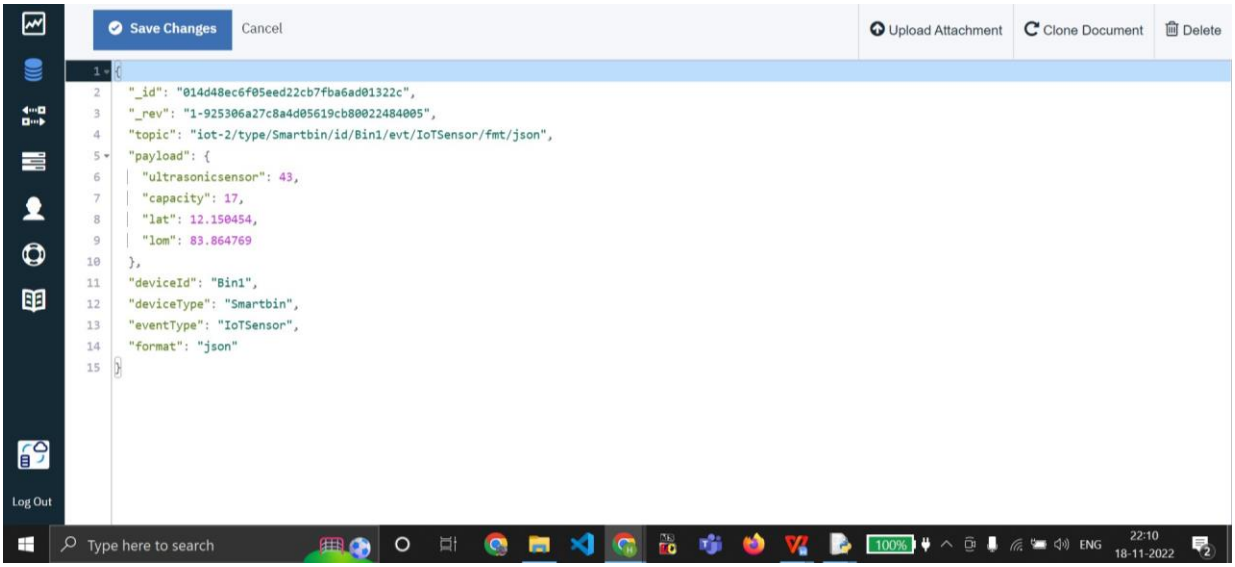
CLOUDANT DIAGRAM:



JSON CODE:



CLOUDANT DOCUMENT:

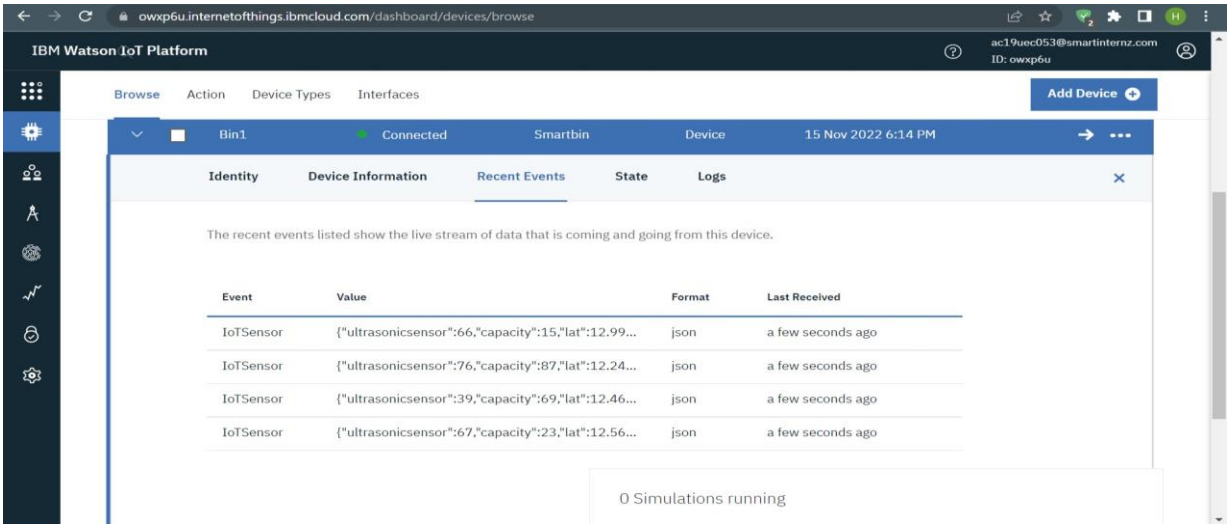


CHAPTER 8

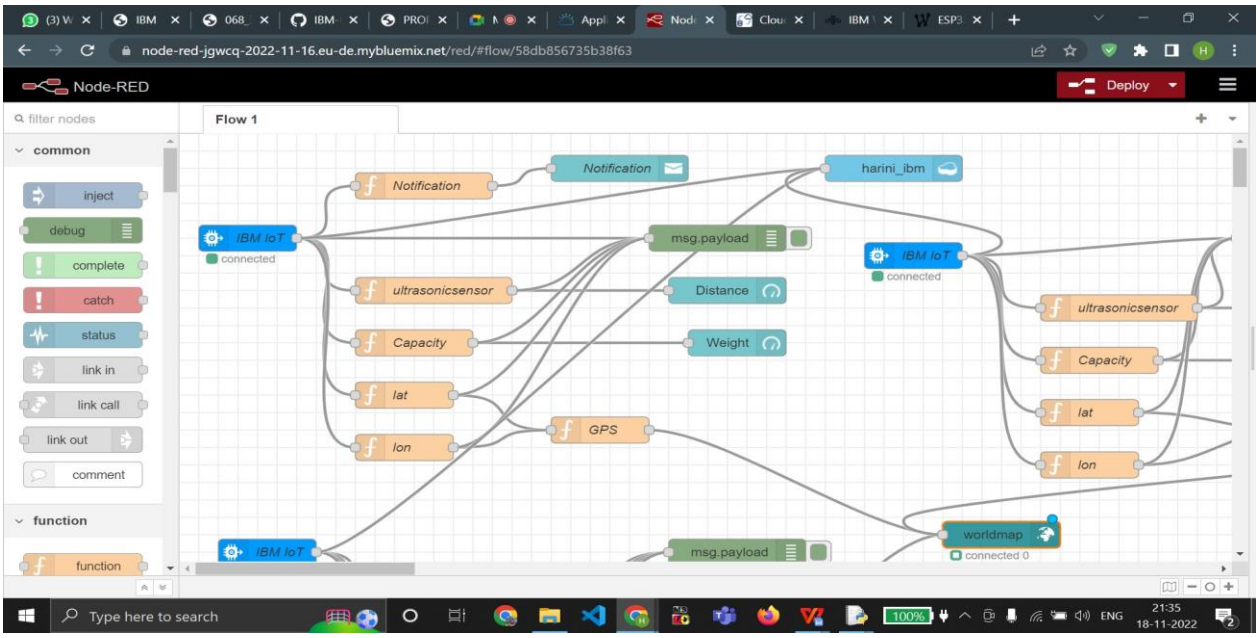
TESTING

8.1 TEST CASES

IBM WATSON output



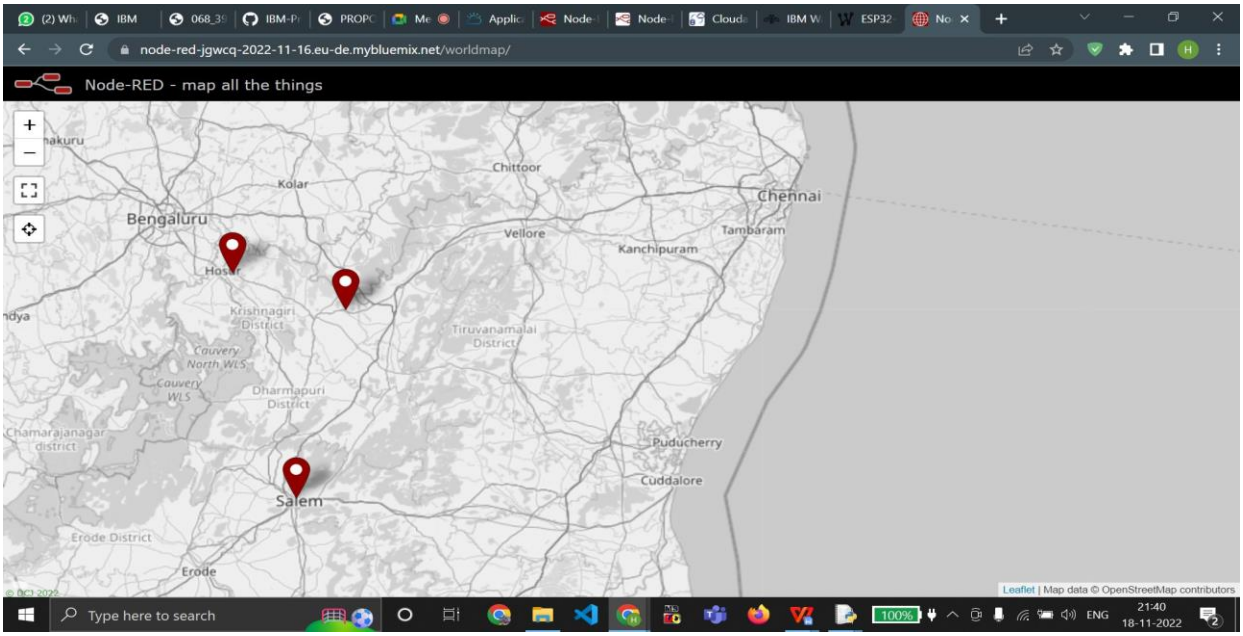
8.2 NODE RED:



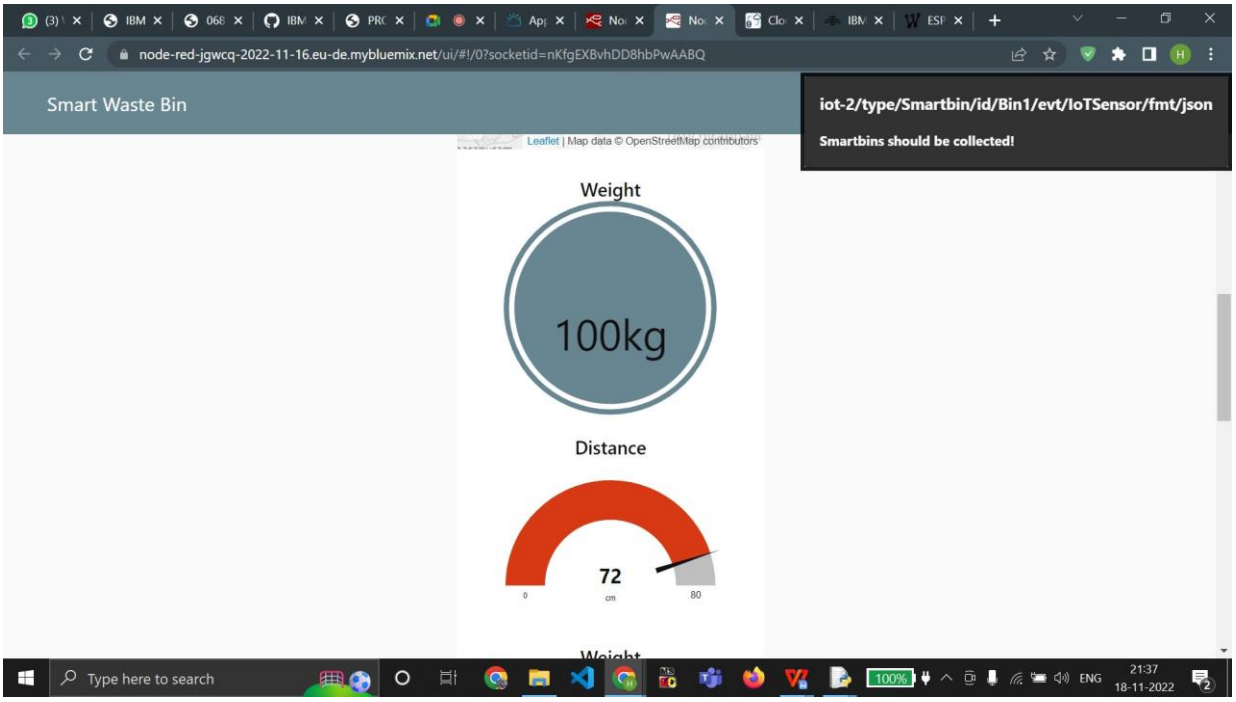
CHAPTER 9

RESULTS

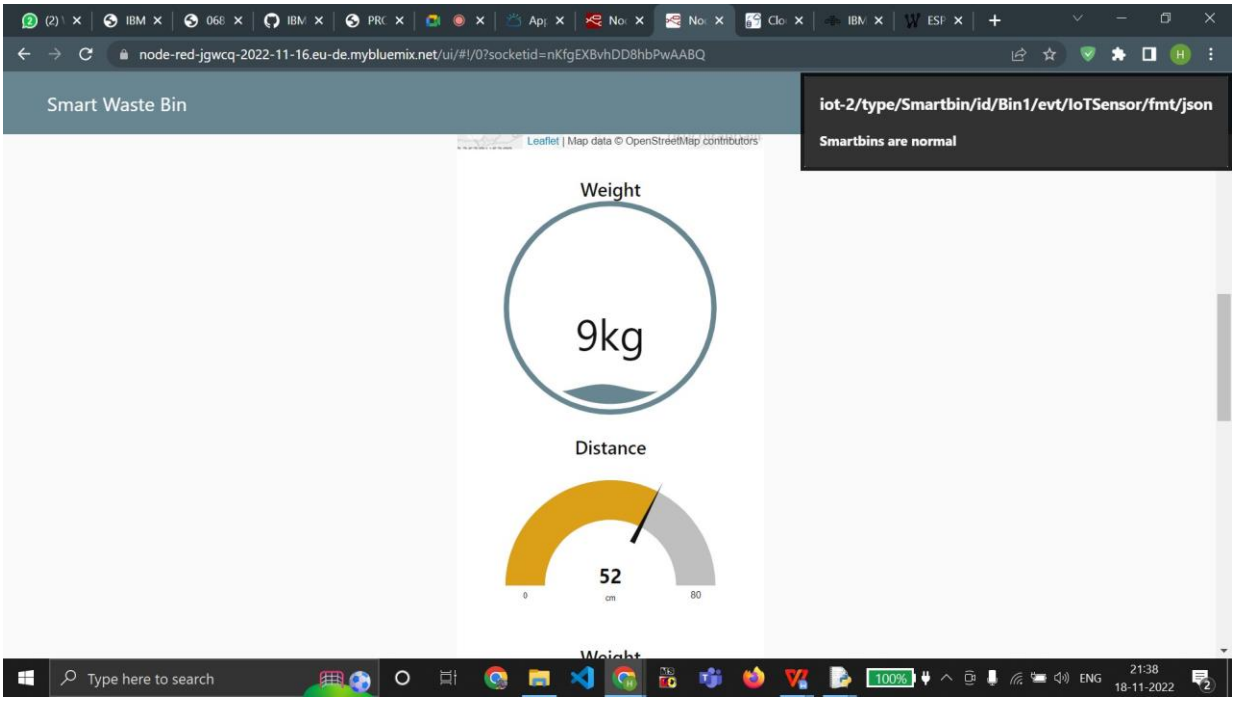
9.1WEB APP MAP



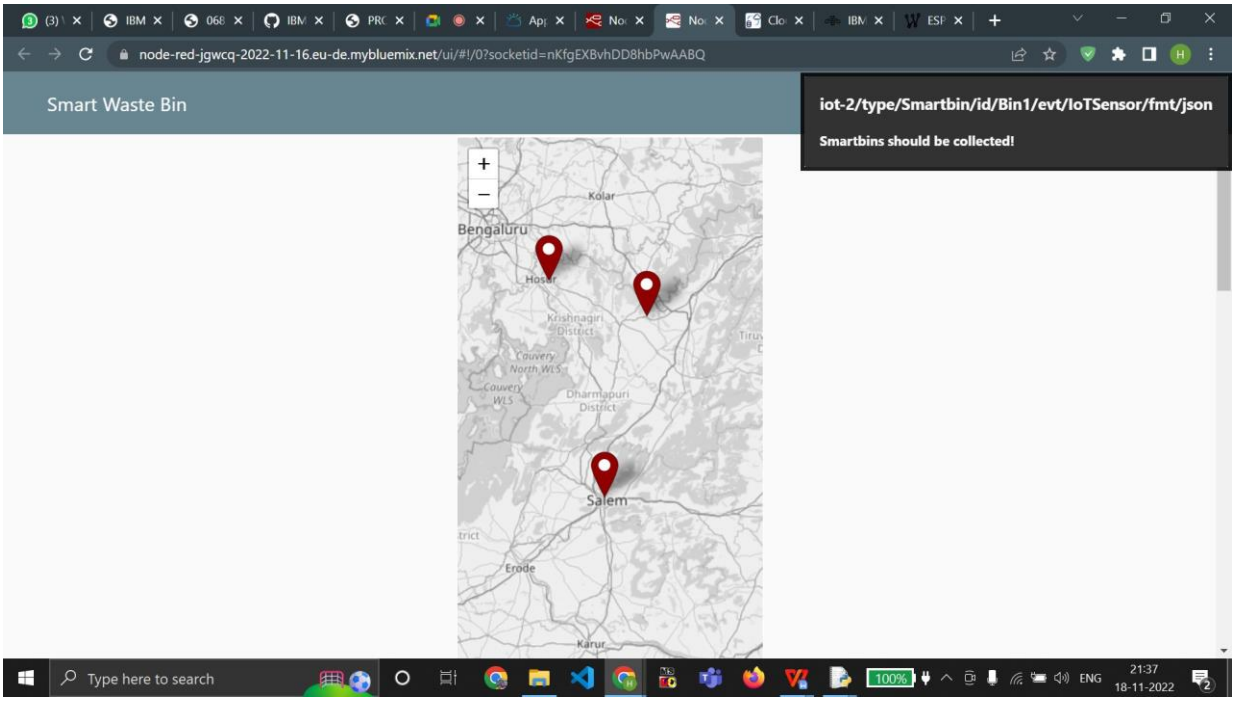
SMART BIN WHEN IT SHOULD BE COLLECTED:



SMART BIN WHEN IT IS NORMAL:



SMART BIN LOCATION:



10. ADVANTAGES AND DISADVANTAGES

Advantages of Smart Waste Management

Following are the benefits or advantages of Smart Waste Management:

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 healthy environment and keep cities more beautiful.

It further reduces manpower requirements to handle the garbage collection process.

Applying 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 of Smart Waste Management

Following are the drawbacks or disadvantages of Smart Waste Management:

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 trainining has to be provided to the people involved in the smart waste management system

11. CONCLUSION

The cloud based system for waste collection in smart cities.

The development system provides improved database for garbage collection time and quantity of garbage collection at each location.

By implementing this project, we will avoid overflowing of garbage from the container in residental area which is previously either loaded manually or with the help of loaders in traditional trucks.

Development of application for city administration municipality staff.

12. FUTURE SCOPE

Future of Solid Waste Disposal & Management in India

With increasing population, urbanization and expanding economic activities, Solid waste disposal and management is a challenge in India. Here are some statistics which shows the solid waste management issues in India:-

According to the Central Pollution Control Board of India, the per capita generation of waste has increased from 0.26 kg/day to 0.85 kg/day. Close to 90% of waste is disposed of without proper treatment causing environmental pollution.

Total of approximately 143,449 MT of municipal waste is generated daily. However, only 35,062 tons of waste is treated.

13. APPENDIX

SOURCE CODE:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "owxp6u"
deviceType = "Smartbin"
deviceId = "Bin1"
authMethod = "token"
authToken= "12345678910"
# Initialize GPIO
def myCommandCallback(cmd):
   print("Command received: %s" % cmd.data['command'])
   status=cmd.data['command']
   if status=="lighton":
      print ("led is on")
   else :
      print ("led is off")
try:
   deviceOptions = {"org": organization, "type": deviceType, "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 "hello" with value "world" into the cloud as an event of type "greeting"
10 times
deviceCli.connect()

while True:
      #Get Sensor Data from DHT11
      time.sleep(5)
      ultrasensor=random.randint(0,80)
```

```
        capacity=random.randint(0,100)
        lat=round(random.uniform(12.03,13.05),6)
        lon=round(random.uniform(80.80,85.90),6)
        data = { 'ultrasonicsensor' : ultrasensor, 'capacity': capacity,'lat':lat,'lom':lon}
        #print data
        def myOnPublishCallback():
            print ("Published ultrasonicsensor = %s Cm" % ultrasensor, "capacity= %s kg" %
capacity,"lat:%s"%lat,"lon:%s"%lon)

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(1)
        deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-54993-1663328851