# SMART WASTE MANAGEMENT SYSTEM FOR METROPOLITAN CITIES

**DOMAIN – INTERNET OF THINGS (IoT)**

# PROJECT REPORT

|  |  |  |
| --- | --- | --- |
| SUBMITTED BY  BARKAVI.B | - 963519106014 | TEAM ID:PNT2022TMID52303  MENTOR : Mrs. A P EVANJALIN |
| RESHMA.M | - 963519106027 | SPOC: Mr. MICHAEL FRANKLIN N |
| JEMISHA.F | - 963519106021 |  |
| SIVA JOTHI.S  AGNES.S  ABISHA .M | - 963519106031  -963519106004  -963519106002    - |  |

In the partial fulfillment of the requirements for the award of a degree of

BACHELOR OF ENGINEERING in COMPUTER SCIENCE AND ENGINEERING

STELLA MARY’S COLLEGE OF ENGINEERING ARUTHENGANVILAI

2022 – 2023

CONTENT

1. **INTRODUCTION**
   1. Project Overview
   2. Purpose

#### LITERATURE SURVEY

* 1. Existing problem
  2. References
  3. Problem Statement Definition

#### IDEATION & PROPOSED SOLUTION

* 1. Empathy Map Canvas
  2. Ideation & Brainstorming
  3. Proposed Solution
  4. Problem Solution fit

#### REQUIREMENT ANALYSIS

* 1. Functional requirement
  2. Non-Functional requirements

#### PROJECT DESIGN

* 1. Data Flow Diagrams
  2. Solution & Technical Architecture
  3. User Stories

#### PROJECT PLANNING & SCHEDULING

* 1. Sprint Planning & Estimation
  2. Sprint Delivery Schedule
  3. Reports from JIRA

#### CODING & SOLUTIONING (Explain the features added in the project along with code)

* 1. Feature 1
  2. Feature 2
  3. Database Schema (if Applicable)

#### TESTING

* 1. Test Cases
  2. User Acceptance Testing

#### RESULTS

* 1. Performance Metrics

#### ADVANTAGES & DISADVANTAGES

1. **CONCLUSION**

#### FUTURE SCOPE

1. **APPENDIX**

Source Code

GitHub & Project Demo Link

1. **INTRODUCTION**

Internet of Things is nothing but the applications performing with the help of internet access. IoT Communication over the internet has grown from user - user interaction to device – device interactions these days. The IoT concepts were proposed years back but still it’s in the initial stage of commercial deployment. Home automation industry and transportation industries are seeing rapid growth with IoT. The basic project idea is to design a smart waste detection system

which would automatically notify the officials about the current status of various garbage bins in the city, would have real time monitoring capabilities, which would be remotely controlled usingIoT techniques. This paper introduces you to the use of IoT on one such area, that is, Garbage Detection in smart ways using IoT and see how this can also be a major part of developing a city into a smart city.

## Project Overview

A big challenge in the urban cities is that of waste management as there is a rapid growth in the rate of urbanization and thus there is a need of sustainable urban development plans. As the concept of smart cities is very much trending these days and the smart cities cannot be complete without smart waste management system. There needs to be system that gives prior information of the filling of the bin that alerts the municipality so that they can clean the bin on time and safeguard the environment. To avoid all such situations we intend to propose a solution for this problem "Smart Garbage Bin", which will alarm and inform the authorized person when the garbage bin is about to fill. Then message will be send to the authorized person to collect the garbage from the particular area. The authorized person will send the message from his web application to the garbage collectors by sending a SMS. This system maintain a dry waste and a wet waste separately. This will help to reduce the overflow of the garbage bin and thus keeping the environment clean.

## Purpose

This project helps the citizens to make their surroundings and environment clean, pollution free and lead a healthy life throughout. It avoids the possibility garbage overflow, unhygienic environment, air-borne and water- borne disease, etc...

## LITERATURE SURVEY

* 1. **EXISTING PROBLEM**

In the existing system garbage is collected by the corporation weekly once or twice. Sometimes the garbage stinks and overflows from the bin and spread over the roads and pollutes the environment. This also produces a heavy air pollution and routes to various air-borne diseases Many a times the street dogs and other animals eat these waste and scatter these waste around the surroundings which creates the spread of various diseases and situation of unclean environment.

**Disadvantages of existing system**:

* + - Time consuming and less effective.
    - Overflow of waste from the bin.
    - Unhygienic Environment and look of the city.
    - Stinky smell and unpleasant situations.

**PROPOSED SYSTEM** :

In this proposed system there will be no issues repeated that of previous system. In this system the bin is designed in such a way that when the waste level reaches the threshold limit it

automatically closes the bin and intimates the alert to the admin. The bins are provided with low cost embedded device which helps in tracking the level of the garbage bins and a unique ID will be provided for every dustbin in the city .These details can be accessed by the concern authorities from their place with the help of internet and an immediate action can be made to clean the bin. The admin can monitor the level of the bin and can trace the location where it exists.

**Advantage**s :

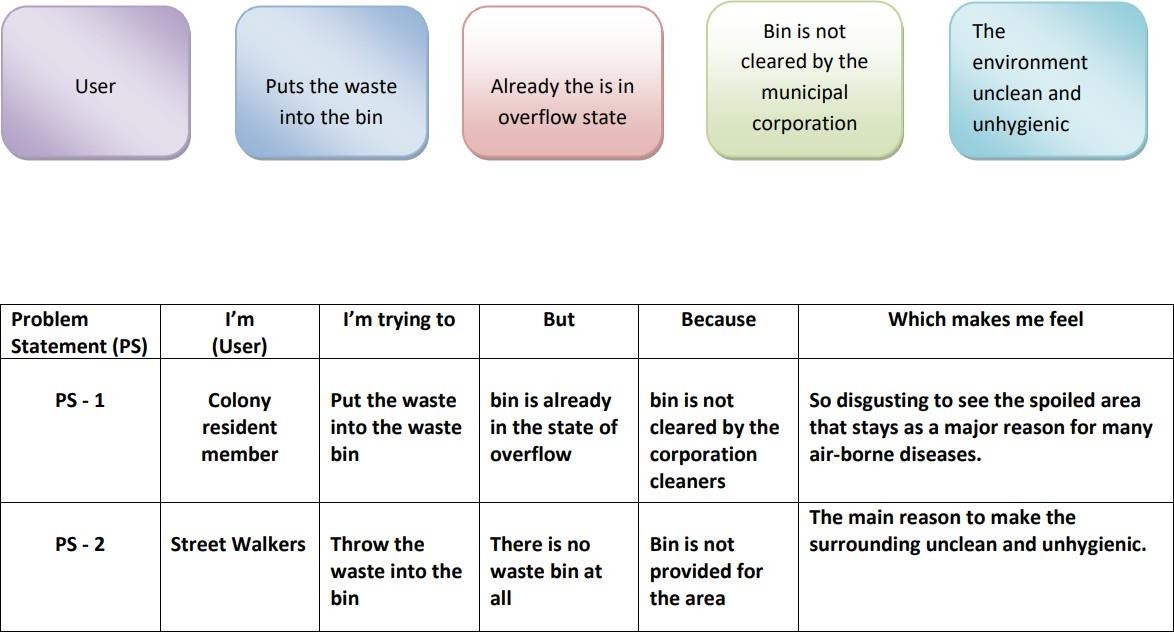
* + - Real time information on the fill level of the dustbin.
    - Deployment of dustbin based on the actual needs.
    - Cost Reduction and resource optimization.
    - Improves Environment quality .

## REFERENCES:

* + 1. Ikuo Ihara; Nagaoka University of Technology; Ultrasonic Sensing: Fundamentals and Its Applications to Non-destructive Evaluation.
    2. Arduino, “Available at [http://www.arduino.cc,](http://www.arduino.cc/)” 2010.
    3. M. Batty, “Smart Cities, Big Data,” Environment and Planning B: Planning and Design 2012, vol. 39, pp. 191– 93.
    4. Xu Li, Student Member, IEEE, Performance Evaluation of Vehicle-Based Mobile Sensor Networks for Traffic Monitoring.
    5. Yusuf Abdullahi Badamasi, The Working Principle Of An Arduino, Electronics, Computer and Computation (ICECCO), 2014 11th International Conference on 29 Sept.-1 Oct. 2014.

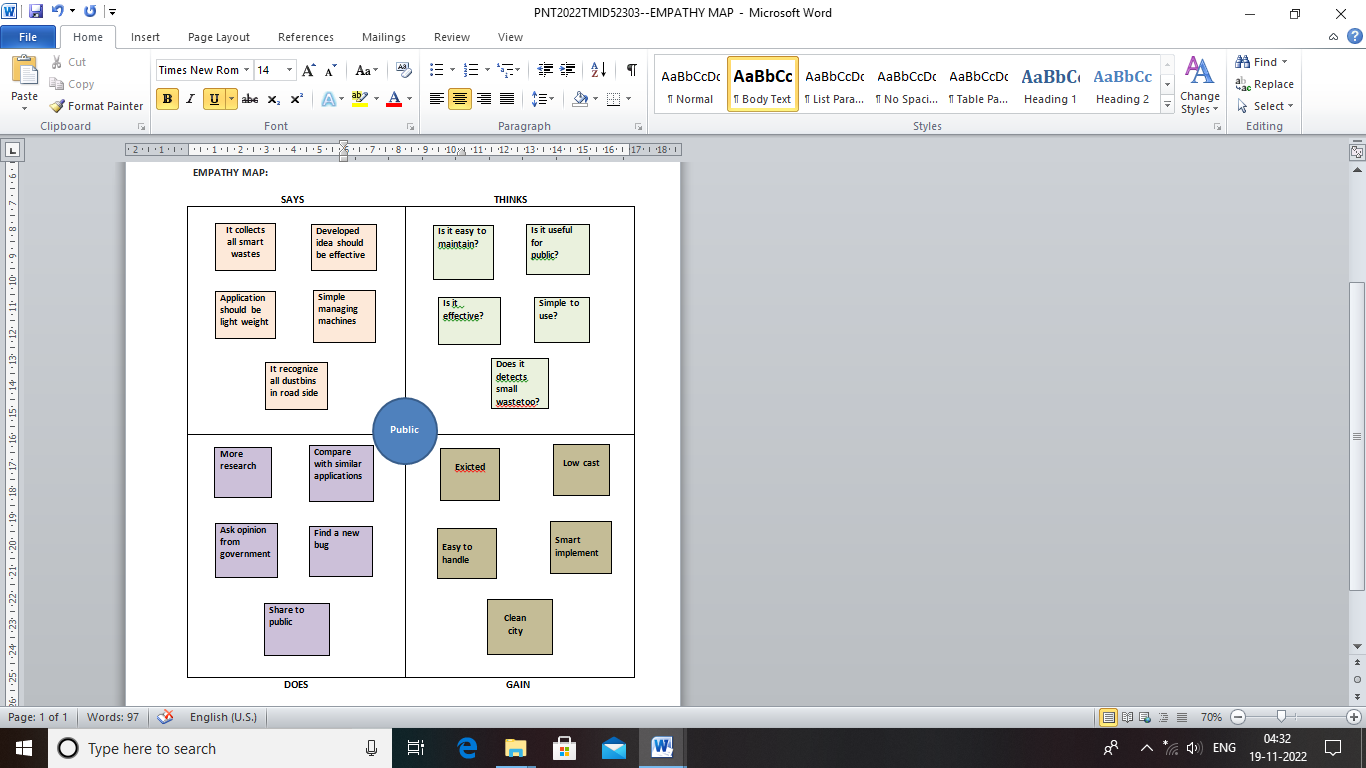
## Problem Statement

The waste management system provided earlier are not very reliable, efficient, cost effective and does not have any advanced processing features like automatic close of bin and alert intimations system .The following is a well articulated problem statement that allows you to find the ideal solution for the challenges faced.

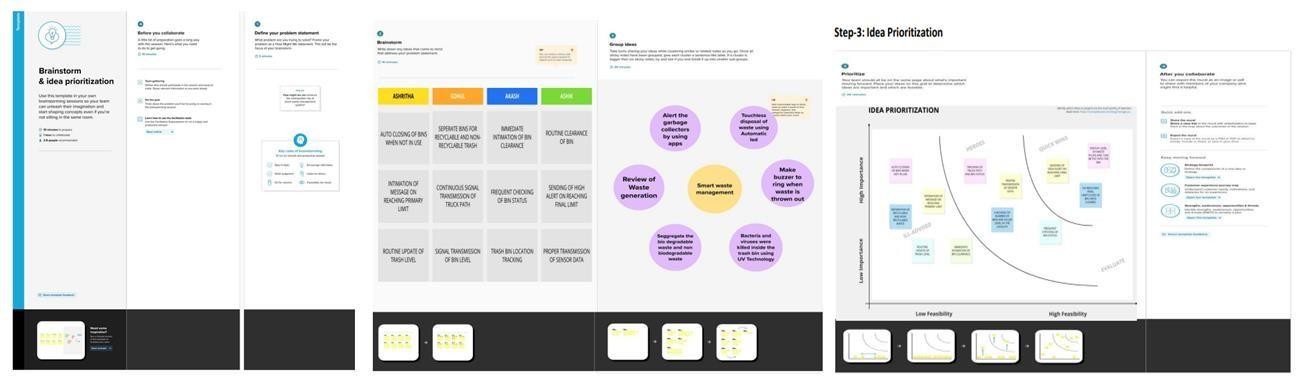


## IDEATION & PROPOSED SOLUTION

* 1. **EMPATHY MAP CANVAS**



## Ideation and Brainstorming



* 1. PROPOSED SOLUTION

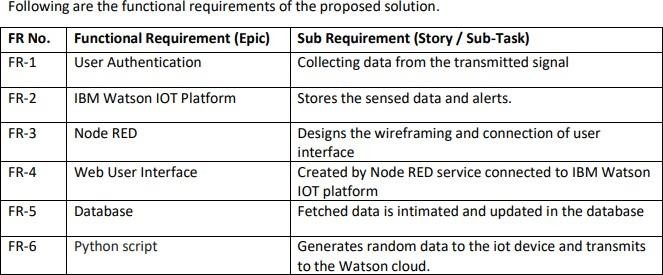
|  |  |  |
| --- | --- | --- |
| **S.No.** | **Parameter** | **Description** |
| 1. | Problem Statement (Problem to be solved) | * Rubbish and waste can cause air and water pollution. * Rotting garbage is also known to produce harmful gases mix with the air and cause breathing problem in people. * Due to improper waste disposal,we may face several problems like unpleasant odour and health problems. |
| 2. | Idea / Solution description | The key research objectives are as follows:   * The proposed system would be able to automate the solid waste monitoring process and management of the overall collection process using IOT (Internet of Things). * The Proposed system consists of main subsystems namely Smart Trash System(STS) and Smart Monitoring and Controlling Hut(SMCH). * To solve this problem of waste management for disposal for disposal using a smart refuse bin built with technologies. * Garbage truck weighing Mechanisms. |
| 3. | Novelty / Uniqueness | * Identify potential waste streams. * Create a waste management-focused community outreach plane |
| 4. | Social Impact / Customer Satisfaction | * Neighbourhood of landfills to communities, breeding of pests and loss in property values. * The IOT solution uses the data and selects optimum routes for waste collection trucks. |
| 5. | Business Model (Revenue Model) | Waste Management organises its operations into two reportable business segments:   * Solid Waste, comprising the Company’s waste collection, transfer, recycling and resource recovery, and disposal services, which are operated and managed locally by the Company’s various subsidiaries, which focus on distinct geographic areas. * Corporate and Other, comprising the Company’s other activities, including its development and operation of landfill gas-to- energy facilities in the INDIA, and its recycling brokerage services, as well as various corporate functions. |
| 6. | Scalability of the Solution | * Installing more bins for collecting recyclables like paper, glass, plastic. * Recycling not only save energy but also prevent the material from going to landfills & incineration and provides raw materials for new products. |

## PROBLEM SOLUTION FIT

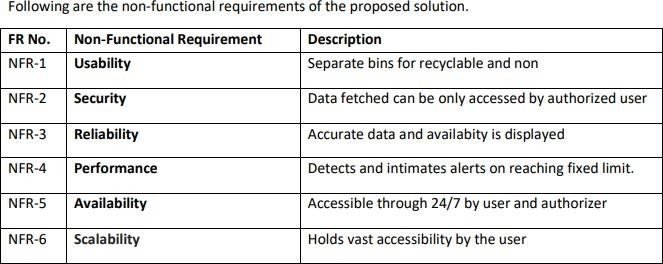
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **1.CUSTOMER SEGMENT(S)**  cs    Government and corporates managing the public | **6.CUSTOMER CONSTRAINTS**  CL  ✓ Indicating the waste level  ✓ Alerting through buzzer system  ✓ Low power requirement  ✓ User portable | **5. AVAILABLE SOLUTIONS**  AS   * **Recycling** - it has economic and environmental advantages. * **Incineration** - This disposal process can be a source of air pollution. * **Landfill** - significant cause of health and environmental problem Example: gas from these landfills is often incredibly dangerous * **Biological Reprocessing**- the ends of the stock is natural gas, which is used to produce heat an electricity. |  |
|  | **2. JOBS-TO-BE-DONE / PROBLEMS**  **PR**  ➢ Managing the wastes in metropolitan cities.  ➢ Providing a smart solution in the form of smart bin.  ➢ Reducing the pollution caused by the trashes.  ➢ Making the public more awared | **9. PROBLEM ROOT/ CAUSE**  RC  1. Lack of Public Awareness  2. Refusal to Learn About Compliance  3. Insufficient Investment in Waste Management  4. Lack of Proper Machinery  NEEDS:  Saving money protect the environment creating jobs builds resilience reduce emission and promote community | **7. BEHAVIOUR**  BE  ▪ Proper installation of bins at regular interval.  ▪ Providing enough awareness to people.  ▪ Correct disposal of trashes in the bin.  ▪ Standard discharging of wastes once the bin is filled  ▪ Keeping for reloading of waste. |  |
|  | **3. TRIGGERS**  TR  The amount of waste generated by an incident affects decisions regarding how to manage the waste, including the storage, treatment and disposal of the waste.  Available capacity is further limited if facilities still accept waste from daily activities during the incident response or choose not to accept incident-  generated waste at all. | **10. YOUR SOLUTION**  **SL**  To implement a smart bin built on a microcontroller based platform Arduino Uno board which is interfaced with GSM modem and Ultrasonic sensor which can gives the status of the waste present in the dustbin to the municipal authority. | **8. CHANNELS of BEHAVIOUR**  **CH**  ONLINE:  Information about the level of trashes filled is indicated and the data is transferred to the control room for each bin including its specifications(GSM module).  OFFLINE:  Placement of bins in the main hubs of the cities, taking necessary action of discharging wastes by the municipals. |  |
| **4.EMOTIONS: BEFORE / AFTER**  EM  BEFORE:  1. Improper management of wastes  2. Less control of public in waste disposal  AFTER:  1. Efficient way of trash monitoring  2. Improvement in cleanliness and public hygiene  3. Quick action of emptying  and refilling of bin |

1. **. REQUIREMENT ANALYSIS**

## FUNCTIONAL REQUIREMENT

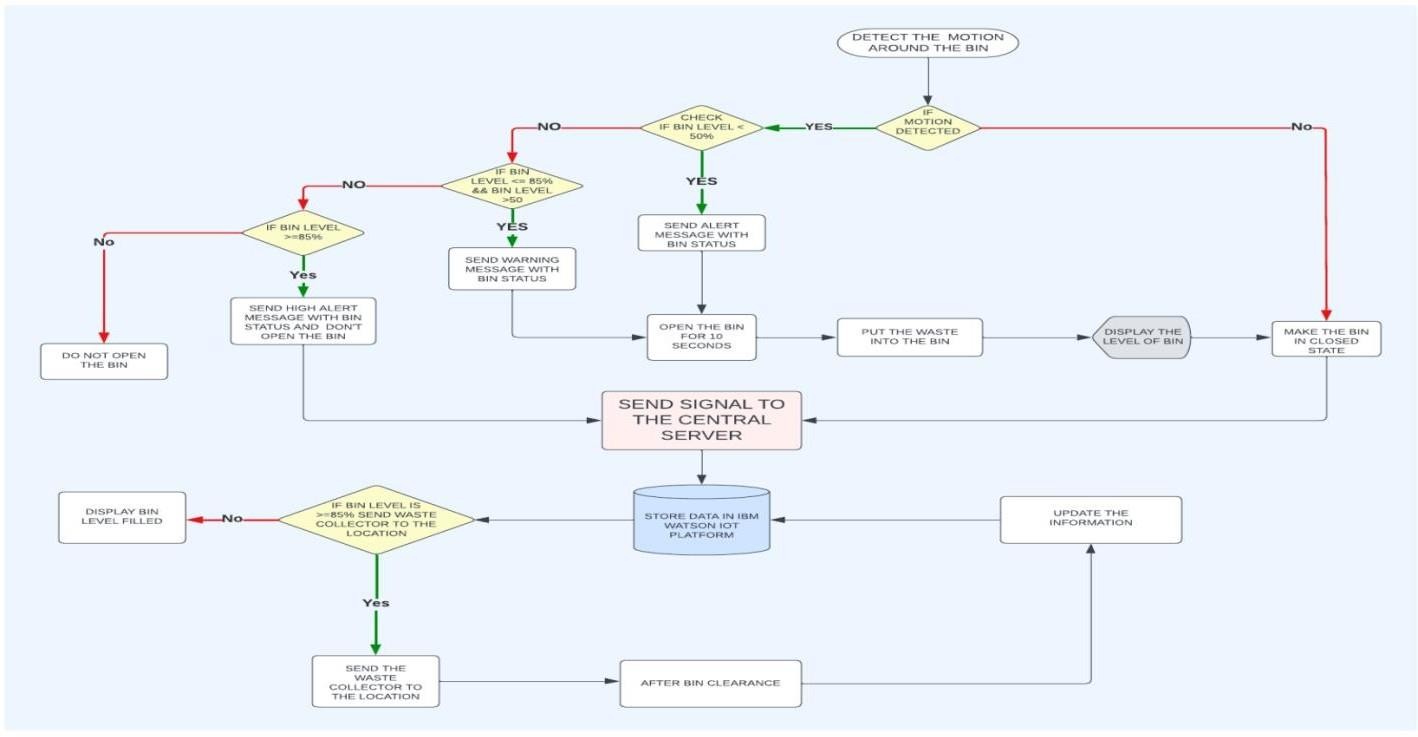


* 1. **NON-FUNCTIONAL REQUIREMENT**

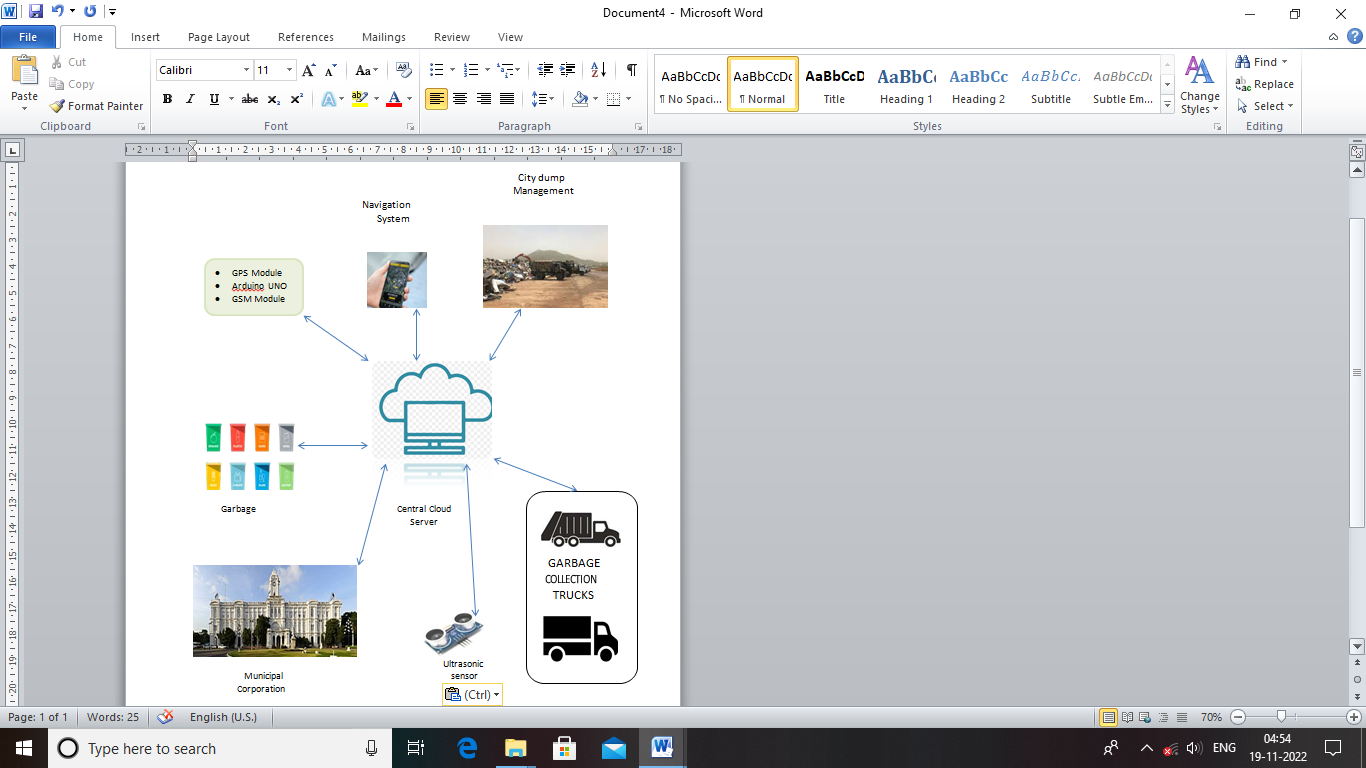


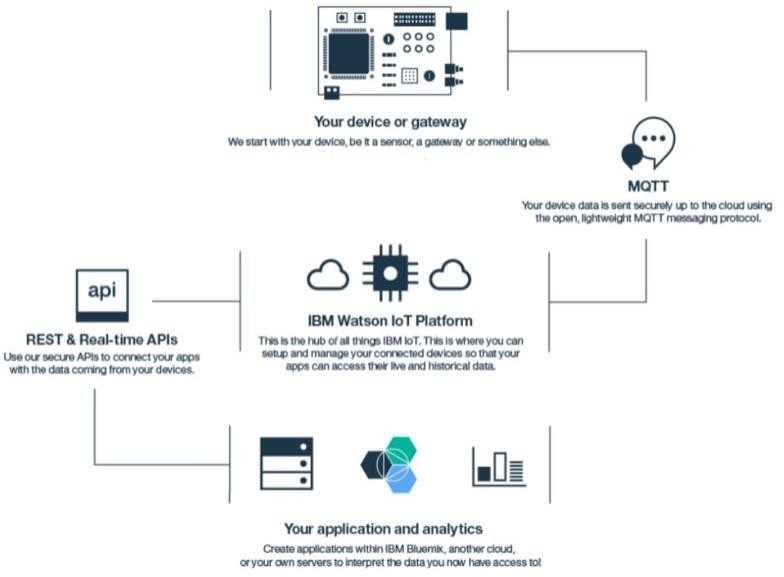
## PROJECT DESIGN

* 1. **. DATA FLOW DIAGRAM**

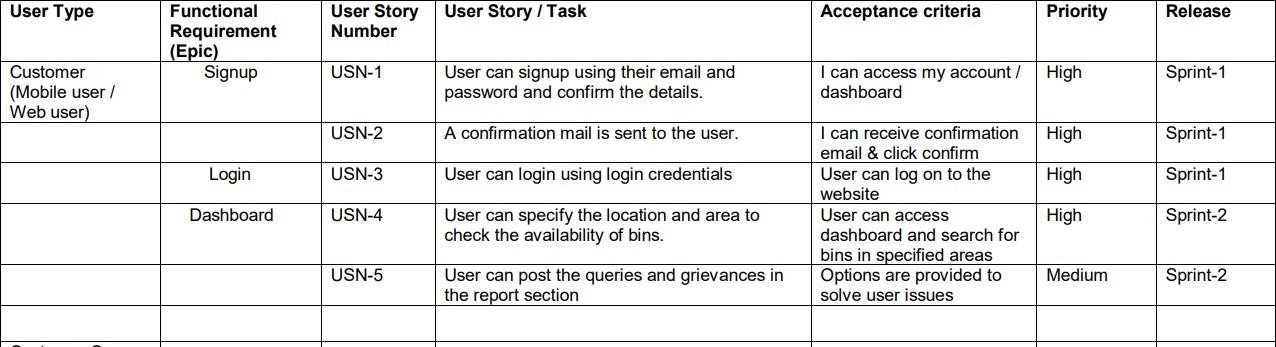


## 5.2. SOLUTION ARCHITECTURE AND TECHNICAL ARCHITECTURE



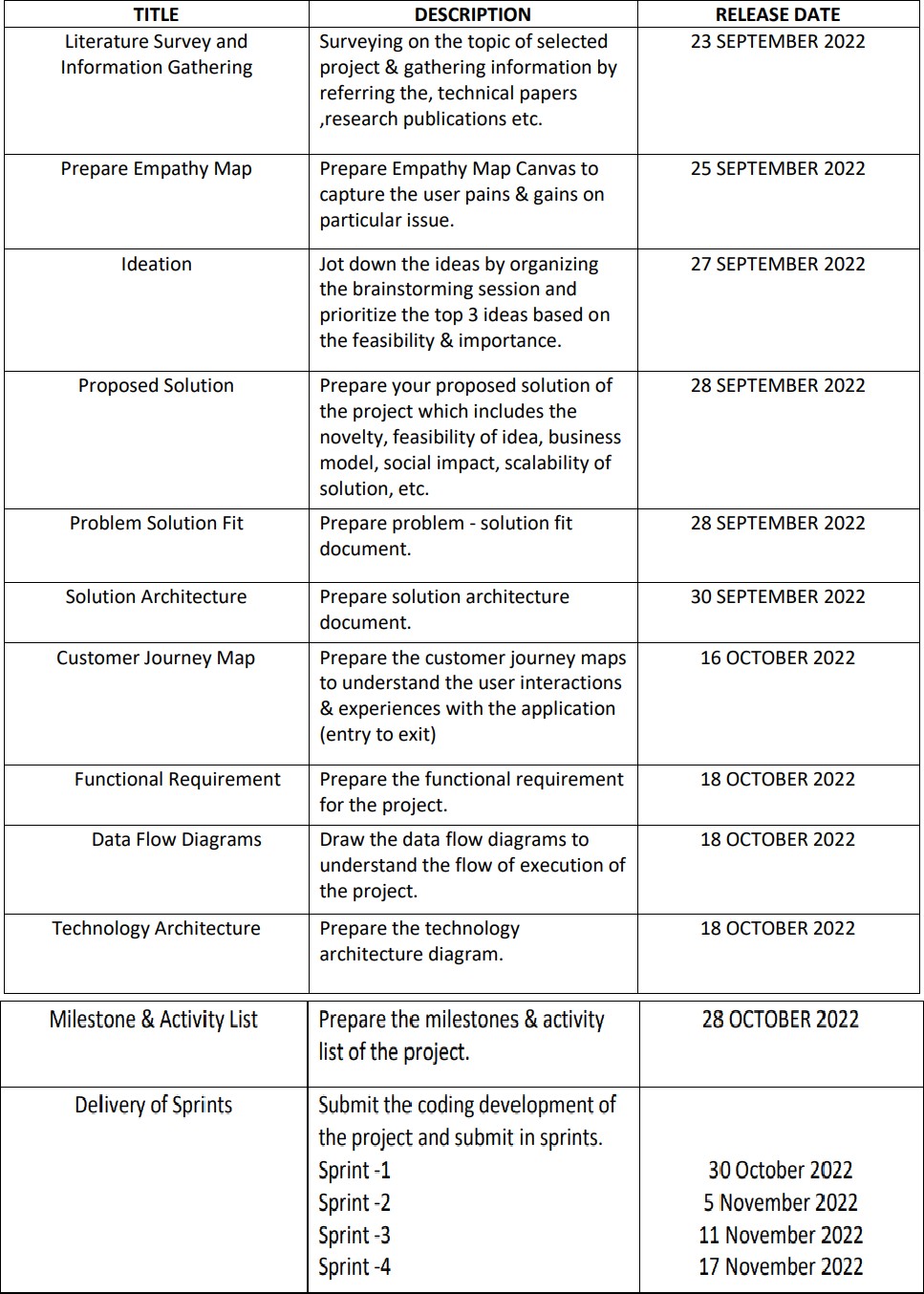


**5.3 USER STORIES**

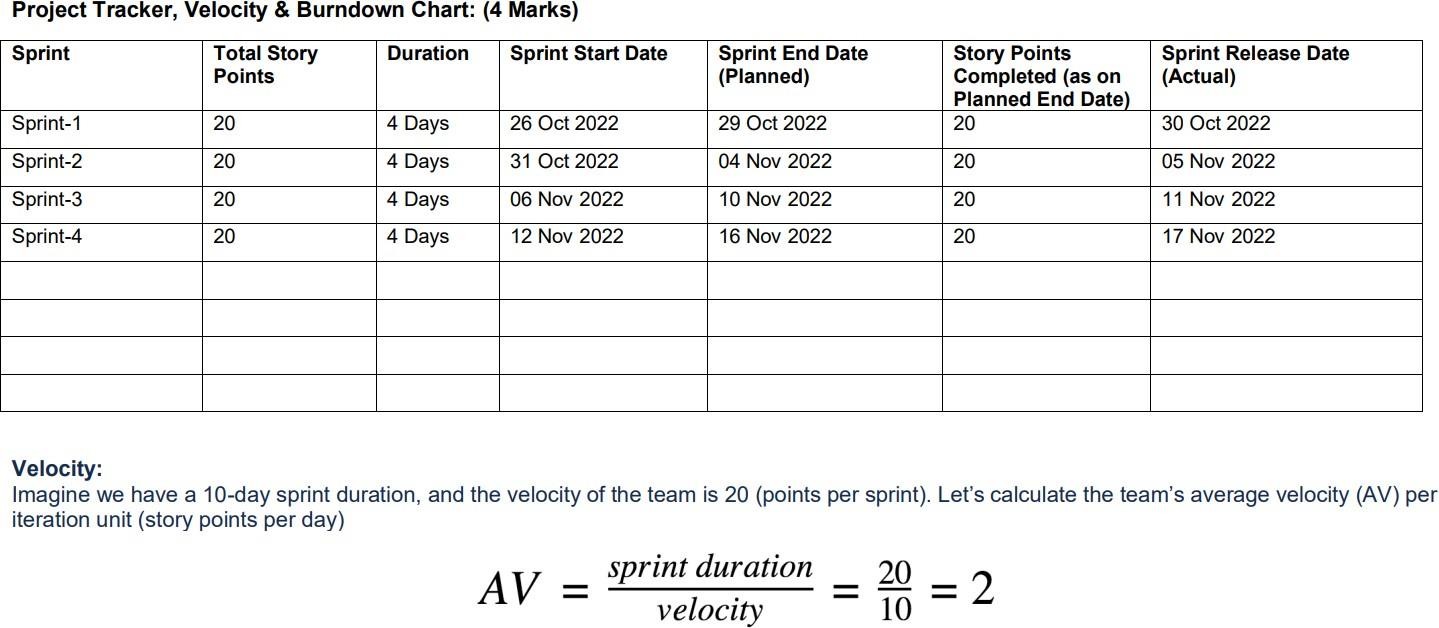


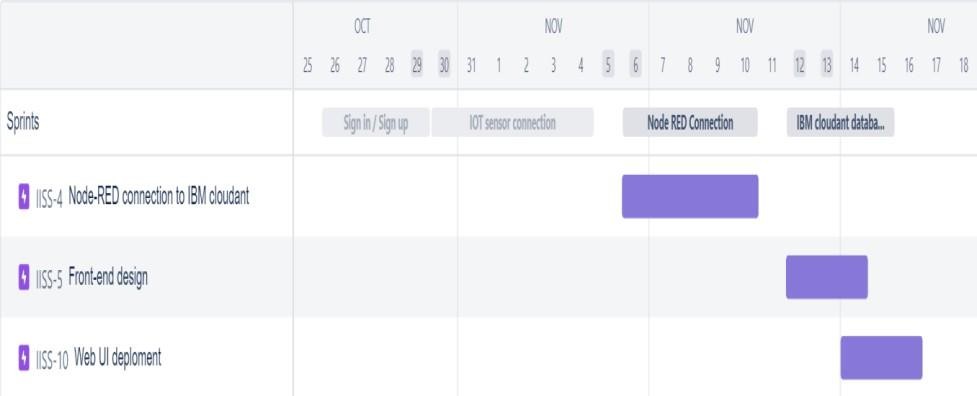
## PROJECT PLANNING AND SCHEDULING

* 1. **. SPRINT PLANNING AND ESTIMATION**

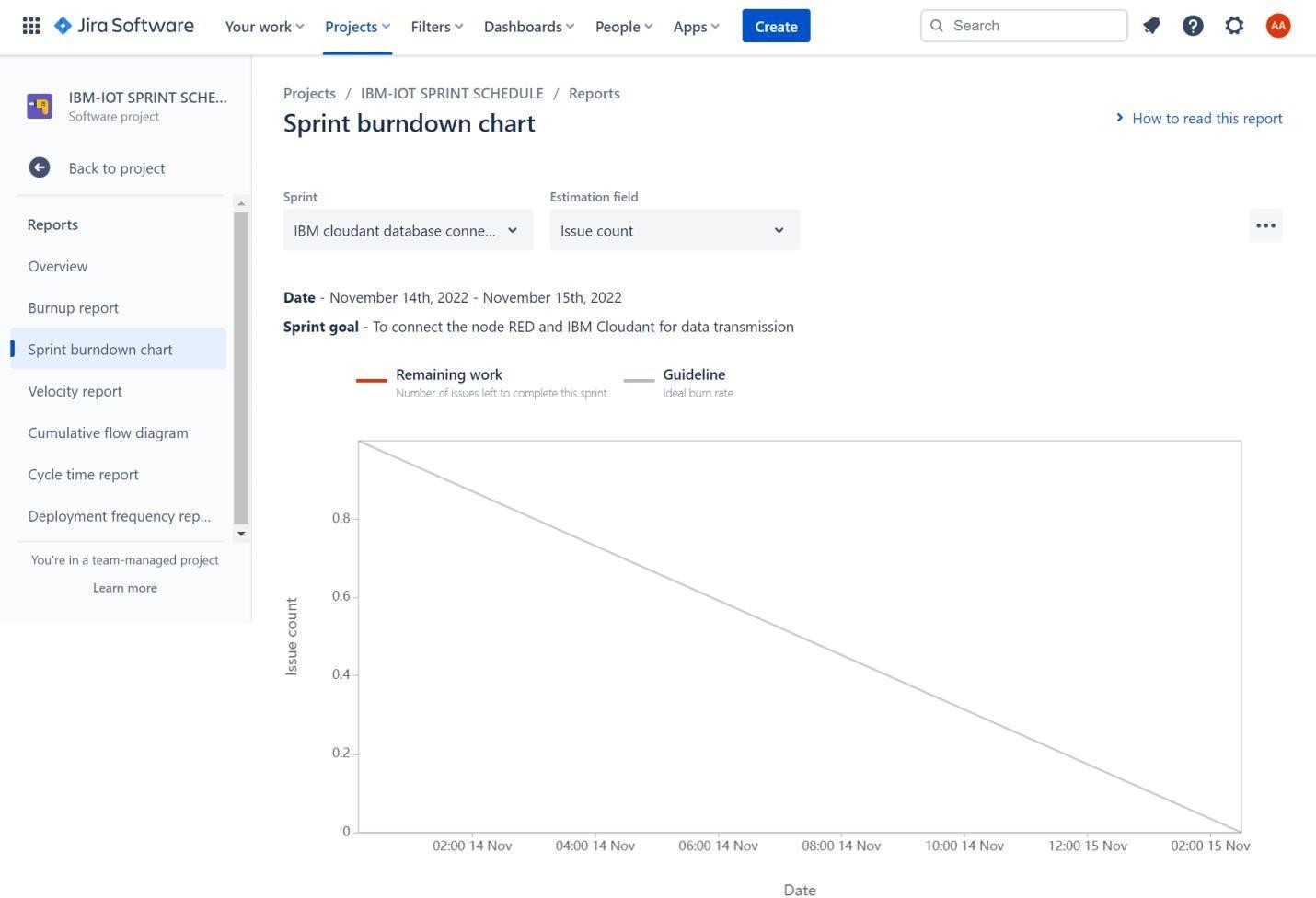


## SPRINT DELIVERY SCHEDULE



* 1. **JIRA REPORTS ROADMAP**

## SPRINT BURNDOWN CHART



1. **CODING AND SOLUTION**

## Wokwi code for Sensor transmission

#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 "9gbe4w" // IBM organisation id

#define DEVICE\_TYPE "SWMSMC" // Device type mentioned in ibm watson iot platform

#define DEVICE\_ID "ibmproject" // Device ID mentioned in ibm watson iot platform #define TOKEN "sUNA41tG6-Pq)0rk5X" // 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;

String data3;

bool SealBin = true; 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 <= 100) //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 > 100 && cm < 180)

{

digitalWrite(4, HIGH);

**Serial**.println("Warning!!,Trash is about to cross 50% of bin level"); digitalWrite(2, LOW);

digitalWrite(23, LOW);

}

else if(cm > 180)

{

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 <= 100)

{

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 <= 180)

{

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 if(cm > 180)

{

digitalWrite(23,HIGH); String payload = "{"; 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");

}

}

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();

}

//handles commands from user side

void callback(char\* subscribetopic, byte\* payload, unsigned int payloadLength)

{

**Serial**.print("callback invoked for topic: ");

**Serial**.println(subscribetopic);

for (int i = 0; i < payloadLength; i++) {

data3 += (char)payload[i];

}

**Serial**.println("data: "+ data3);

const char \*s =(char\*) data3.c\_str(); double pincode = 0;

const char \*buf; int len;

if (mjson\_find(s, strlen(s), "$.command", &buf, &len)) // And print it

{

String command(buf,len); if(command=="\"SealBin\"")

{

SealBin = true;

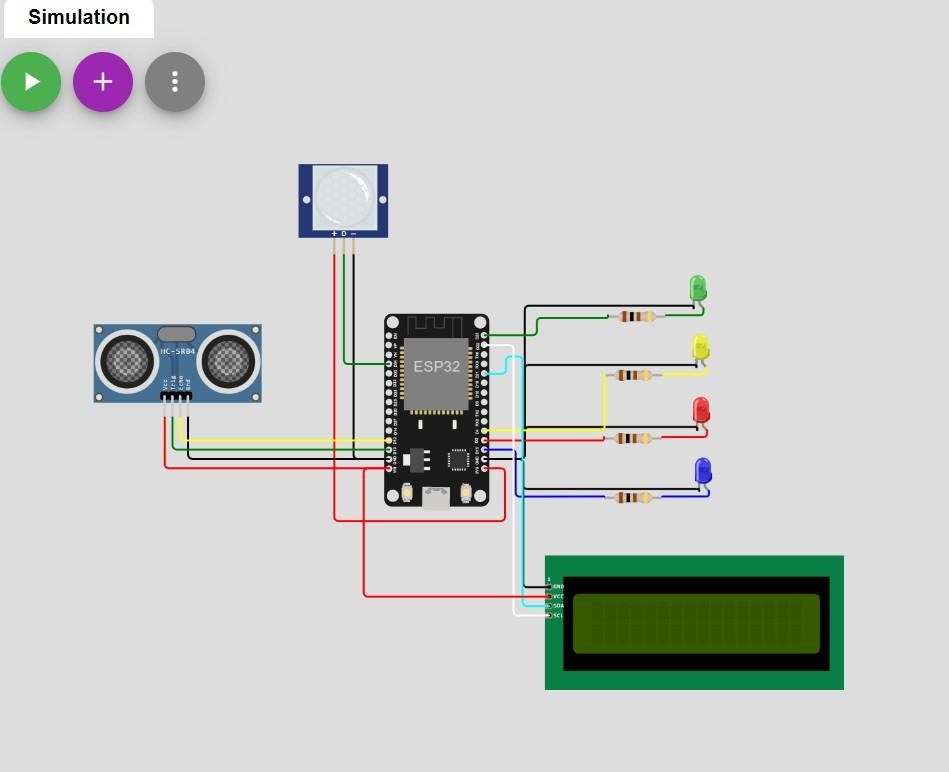
}

}

data3="";

}

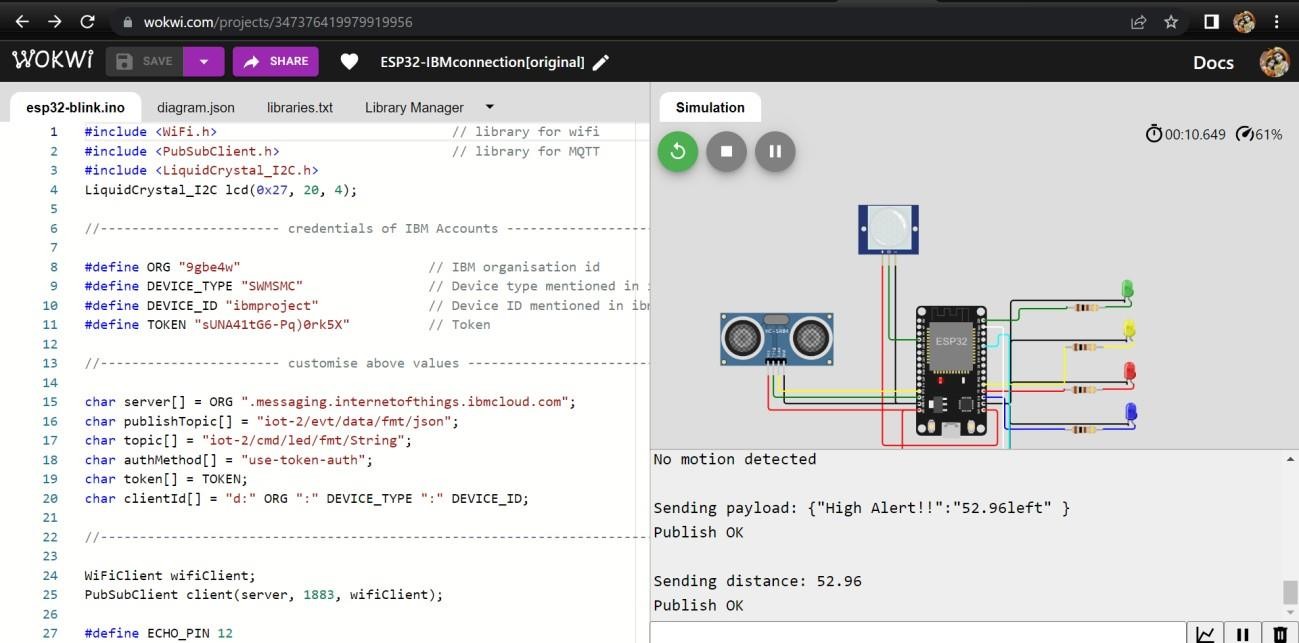
# . Sensor Connection Setup



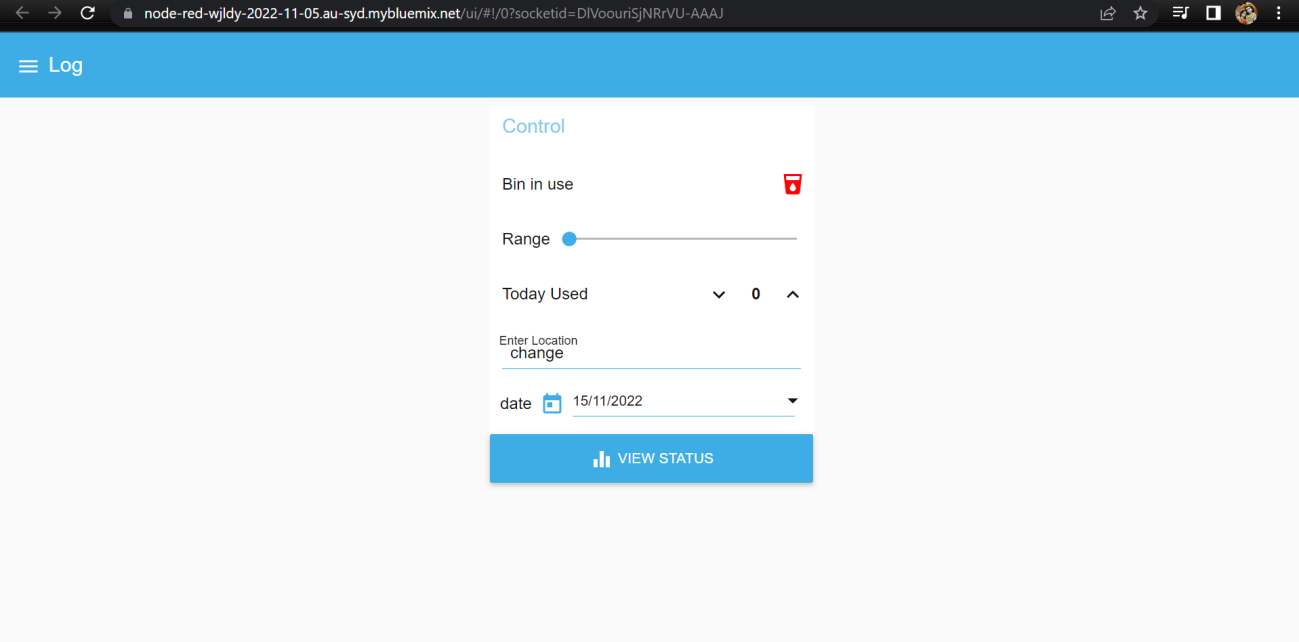
PHYSICAL COMPONENTS:

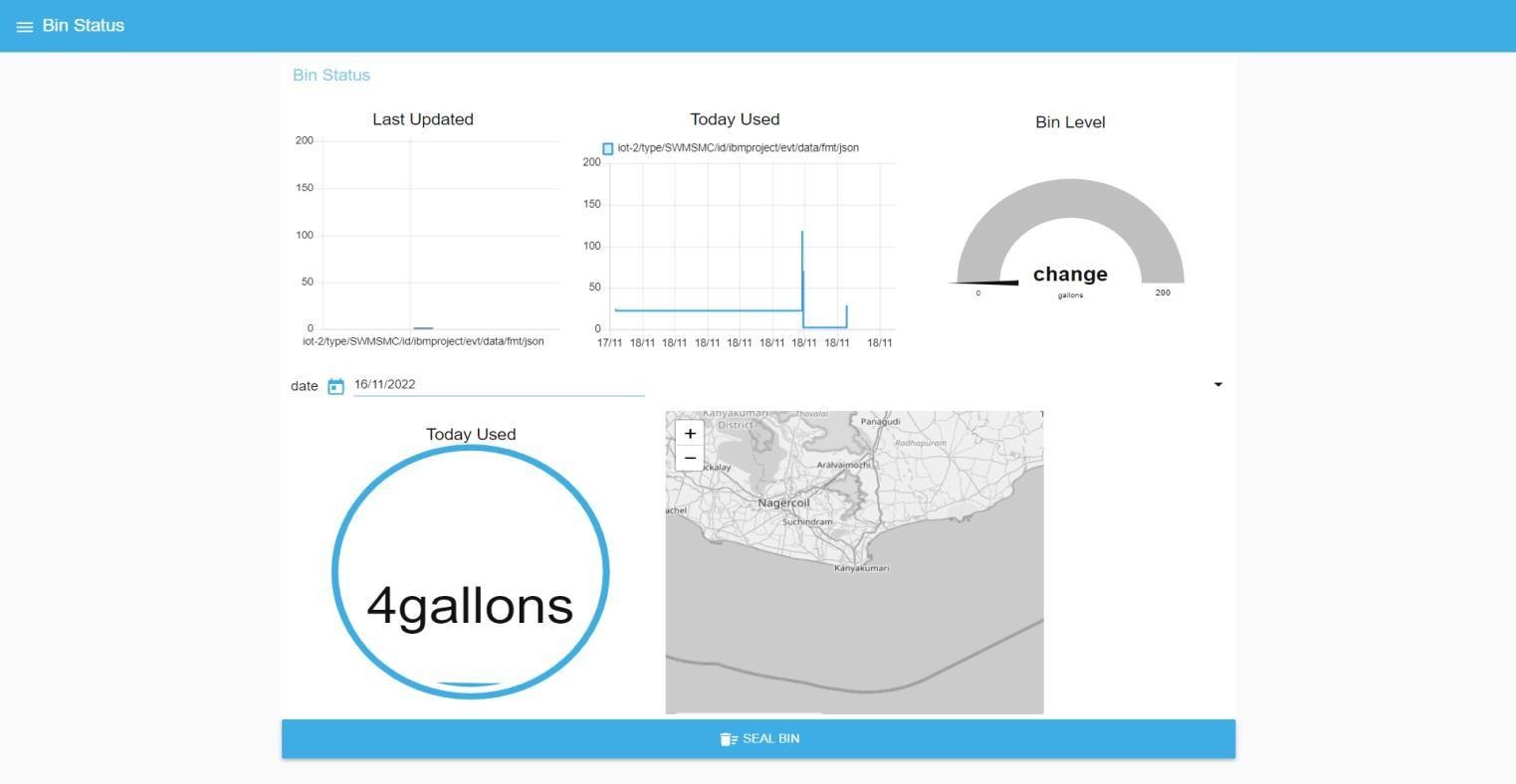
* + - PIR MOTION SENSOR
    - ULTRASONIC DISTANCE SENSOR
    - ESP32-ARDUINO MICROCONTROLLER

OUTPUT: WOKWI SETUP



WEB UI





*The admin gets notification when the bin detects motion and if the bin level crosses 50 percent it indicates warning and if it crosses 90 percent it gives a High alert and closes the bin. If the admin wants to seal the bin the admin can command seal bin until it is accessed for cleaning.*

# Test Case:

Maximum Size of Bin : 200 cm Safe limit: below 100 cm Minimum threshold limit of bin: 100 cm

Maximum threshold limit of bin: 180 cm

|  |  |  |  |
| --- | --- | --- | --- |
| **S.no** | **Bin Level**  **(cm filled)** | **Bin Status** | **Location** |
| *1* | *45* | *Safe* | *Kanyakumari* |
| *2* | *78* | *Safe* | *Coimbatore* |
| *3* | *112* | *Warning* | *Trichy* |
| *4* | *169* | *Warning* | *Chennai* |
| *5* | *186* | *Warning* | *Ooty* |
| *6* | *193* | *High\_Alert* | *Tirunelveli* |
| *8* | *0* | *Safe* | *Chengalpattu* |
| *9* | *35* | *Safe* | *Madurai* |
| *10* | *101* | *Warning* | *Salem* |
| *11* | *132* | *Warning* | *Thanjavore* |
| *12* | *158* | *Warning* | *Vellore* |
| *13* | *93* | *High\_Alert* | *Erode* |
| *14* | *93* | *High\_Alert* | *Karur* |
| *15* | *93* | *High\_Alert* | *Cuddalore* |
| *16* | *30* | *Safe* | *Kumbakonam* |
| *17* | *110* | *Warning* | *Ambur* |
| *18* | *180* | *Warning* | *Sivakasi* |
| *19* | *195* | *High\_Alert* | *Neyveli* |
| *20* | *80* | *Safe* | *Krishnagiri* |

**Note:** *The bin location provided above is default. When the user access the bin , the location and status of the bin displayed to the admin.*

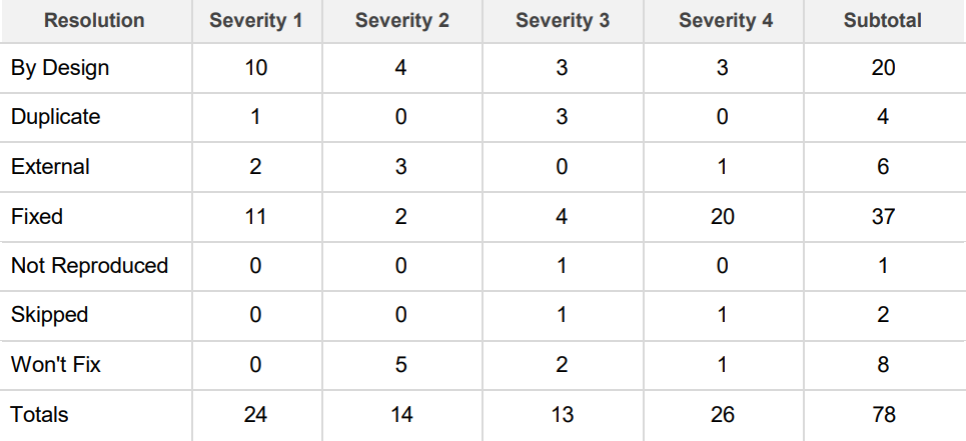
# USER ACCEPTANCE TESTING

#### 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).

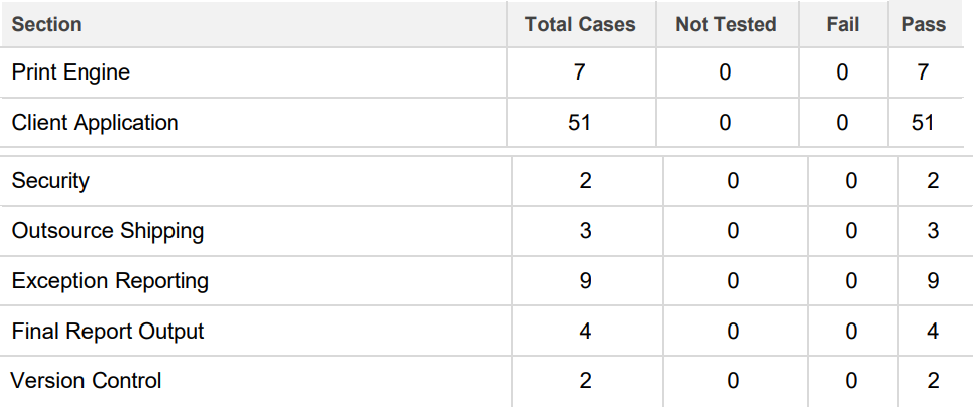
#### Defect Analysis

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



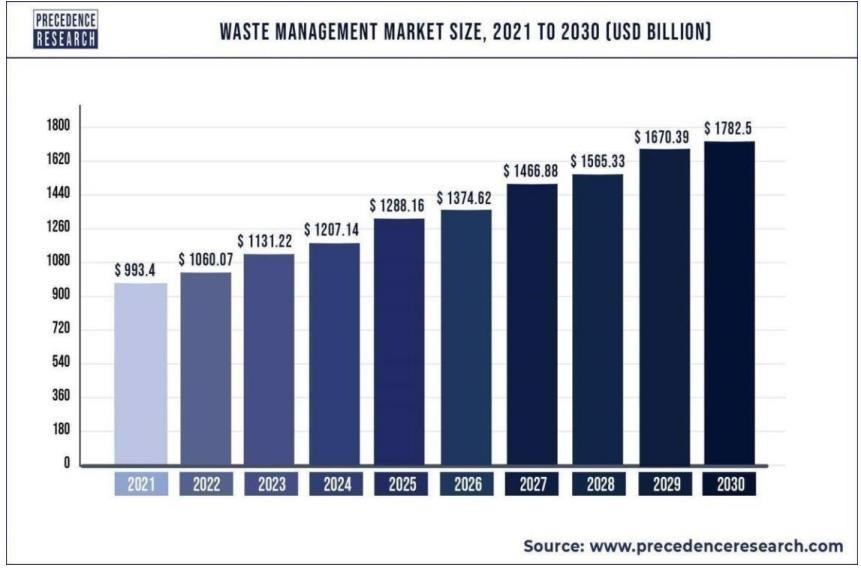
#### TEST CASE ANALYSIS

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

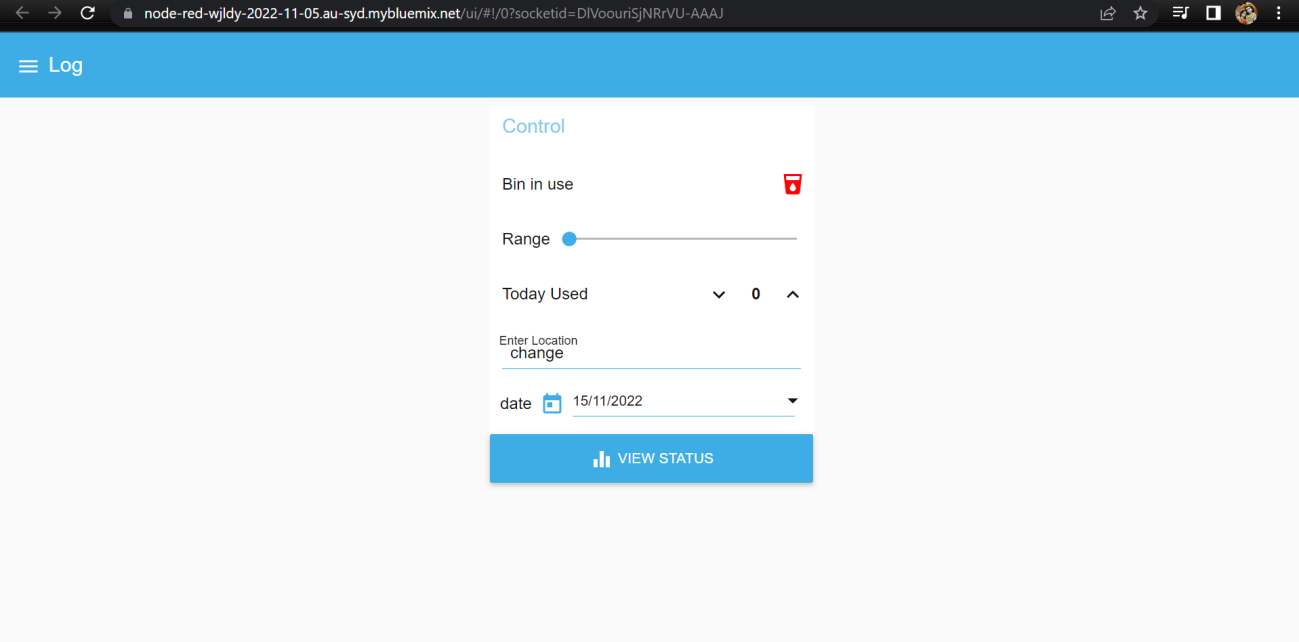


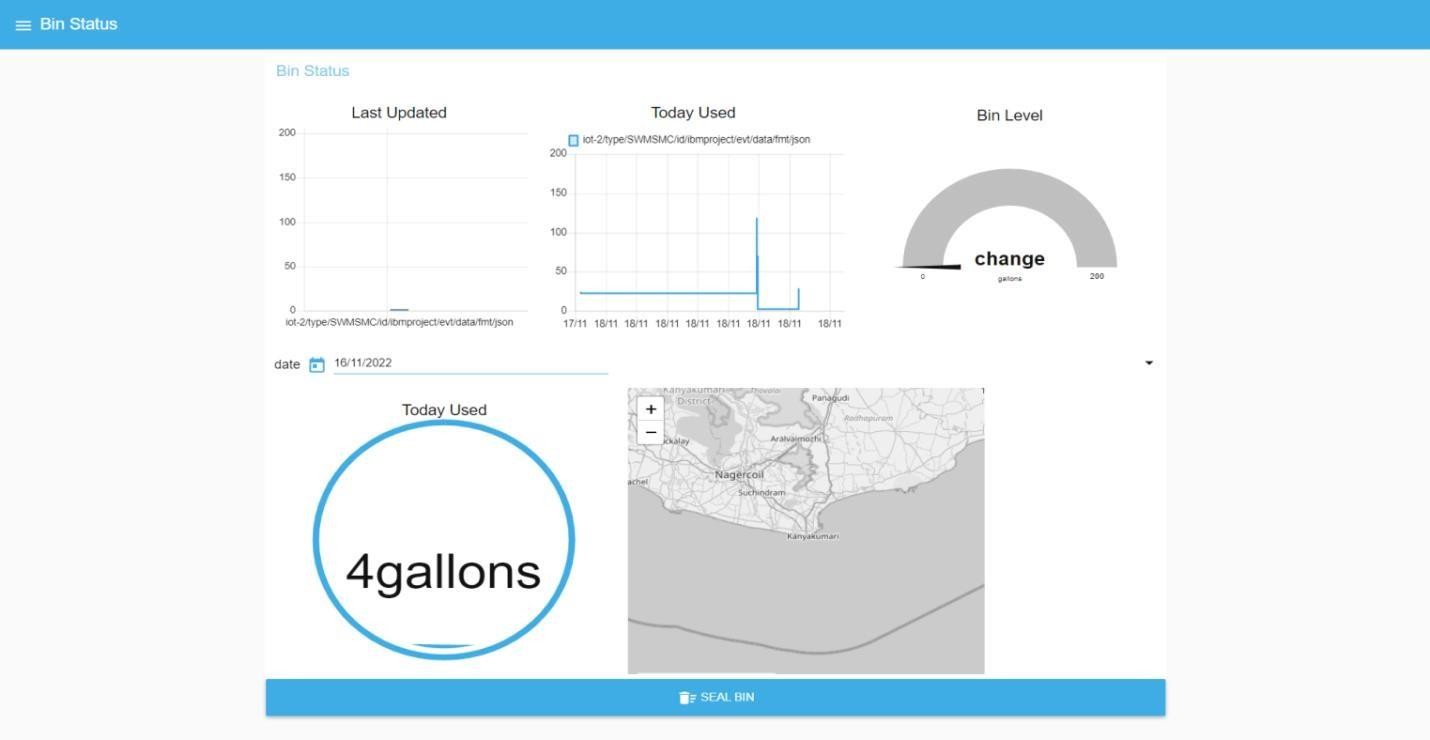
# RESULTS

## Performance Metrics



**9.2 . Admin Web UI**





## ADVANTAGES AND DISADVANTAGES

* 1. **ADVANTAGES**
* Reduction in Collection Cost
* No Missed Pickups
* Reduced Overflows
* Waste Generation Analysis
* CO2 Emission Reduction

## 10.2 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.

# 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. But these may price bit high.

# FUTURE SCOPE:

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

* Changes the system of user authentication and atomic lock of bins, which would aid in protecting the bin from damage or theft.
* 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’.
* Having case study or data analytics on the type and times waste is collected on different days or seasons, making the bin level predictable and remove the reliance on electronic components, and fixing the coordinates.
* Improving the Server's and Android's graphical interfaces

# 14. APPENDIX

### Esp32 - Microcontroller :

ESP32 is a series of low-cost, low-power system on a chip microcontrollers with integrated Wi-Fi and dual-mode Bluetooth.

* + Memory: 320 KiB
  + SRAM CPU: Tensilica Xtensa LX6 microprocessor @ 160 or 240 MHz
  + Power: 3.3 V DC
  + Manufacturer: Espressif Systems
  + Predecessor: ESP8266
* **Sensors :**
  + PIR motion sensor: PIR sensors allow you to sense motion, almost always used to detect whether a human has moved in or out of the sensors range.
  + Ultrasonic Distance Sensor : Ultrasonic Sensors measure the distance to the target by measuring the time between the emission and reception.

## 13.1. Source code

#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 "9gbe4w" // IBM organisation id

#define DEVICE\_TYPE "SWMSMC" // Device type mentioned in ibm watson iot platform

#define DEVICE\_ID "ibmproject" // Device ID mentioned in ibm watson iot platform #define TOKEN "sUNA41tG6-Pq)0rk5X" // 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;

String data3;

bool SealBin = true; 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 <= 100) //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 > 100 && cm < 180)

{

digitalWrite(4, HIGH);

**Serial**.println("Warning!!,Trash is about to cross 50% of bin level"); digitalWrite(2, LOW);

digitalWrite(23, LOW);

}

else if(cm > 180)

{

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 <= 100)

{

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 <= 180)

{

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 if(cm > 180)

{

digitalWrite(23,HIGH); String payload = "{"; 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");

}

}

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();

}

//handles commands from user side

void callback(char\* subscribetopic, byte\* payload, unsigned int payloadLength)

{

**Serial**.print("callback invoked for topic: ");

**Serial**.println(subscribetopic);

for (int i = 0; i < payloadLength; i++) {

data3 += (char)payload[i];

}

**Serial**.println("data: "+ data3);

const char \*s =(char\*) data3.c\_str(); double pincode = 0;

const char \*buf; int len;

if (mjson\_find(s, strlen(s), "$.command", &buf, &len)) // And print it

{

String command(buf,len); if(command=="\"SealBin\"")

{

SealBin = true;

}

}

data3="";

}

## 13.2 . GITHUB LINK

**LINK : https://github.com/IBM-EPBL/IBM-Project-51721-1660982220**