Final Code

Team ID	PNT2022TMID27888
Project Name	Smart Waste Management System for Metropolitan Cities

SOURCE CODE (For Smart Bin):

```
#include <ESP32Servo.h>

#include <LiquidCrystal_I2C.h>


#include <HX711.h>


#define DATA_PIN  12

#define CLOCK_PIN 14


#include <WiFi.h>

#include <PubSubClient.h>

WiFiClient wifiClient;

#define ORG "uuyxja"

#define DEVICE_TYPE "NodeMcu"

#define DEVICE_ID "12345"

#define TOKEN "23323850"

#define speed 0.034


char server[] = ORG ".messaging.internetofthings.ibmcloud.com";

char publishTopic[] = "iot-2/evt/Data/fmt/json";

char topic[] ="iot-2/cmd/home/fmt/String";

char authMethod[] = "use-token-auth";

char token[] = TOKEN;

char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
```

```
PubSubClient client(server, 1883, wifiClient);
void publishData();
const int trigpin=2;
const int echopin=15;
String command;
String data="";
long duration;
float dist;

LiquidCrystal_I2C LCD = LiquidCrystal_I2C(0x27, 16, 2);
Servo servo;
int trigPin1 = 2;
int echoPin1 = 15;
int trigPin2 = 18;
int echoPin2 = 5;
int duration1;
int distance1;
int duration2;
int distance2;

void setup()
{
Serial.begin(115200);
LCD.begin(16,2);
LCD.init();
LCD.backlight();
LCD.clear();

servo.attach(23);
Serial.begin(115200);
pinMode(trigPin1, OUTPUT);
```

```
pinMode(echoPin1, INPUT);
pinMode(trigPin2, OUTPUT);
pinMode(echoPin2, INPUT);
wifiConnect();
mqttConnect();
}

void loop() {
publishData();
delay(500);
if (!client.loop()) {
  mqttConnect();
}
}

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(client.subscribe(topic));
    Serial.println("subscribe to cmd OK");
}
else {
 Serial.println("subscribe to cmd FAILED"); }
 }
void publishData()
{
 digitalWrite(trigPin1, LOW);
 delayMicroseconds(2);
 digitalWrite(trigPin1, HIGH);
 delayMicroseconds(10);
 digitalWrite(trigPin1, LOW);
 duration1 = pulseIn(echoPin1, HIGH);
 distance1= duration1*0.034/2;
 //Serial.println(distance1);
 delay(100);

 digitalWrite(trigPin2, LOW);
 delayMicroseconds(2);
```

```
digitalWrite(trigPin2, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin2, LOW);
duration2 = pulseIn(echoPin2, HIGH);
distance2= duration2*0.034/2;
//Serial.println(distance2);
delay(100);

LCD.setCursor(0,1);
LCD.print("Fill Status ");



if(distance2>300 && distance2<=400){
  LCD.setCursor(12,1);
  LCD.print("25% ");
  String payload = "{\"Bin_Level\":";
  payload += "25";
  payload += ",\"Weight\":";
  payload += "12.5";
  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(distance2 > 200 && distance2<= 299){
  LCD.setCursor(12,1);
```

```
  LCD.print("50%");
  String payload = "{\"Bin_Level\":";
  payload += "50";
  payload += ",\"Weight\":";
  payload += "25";
  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(distance2 >50 && distance2<=199){
  LCD.setCursor(12,1);
  LCD.print("75%");
  String payload = "{\"Bin_Level\":";
  payload += "75";
  payload += ",\"Weight\":";
  payload += "37.5";
  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{
   LCD.setCursor(12,1);
   LCD.print("100%");
   String payload = "{\"Bin_Level\":";
   payload += "100";
   payload += ",\"Weight\":";
   payload += "50";
   payload += "}";
   Serial.print("Sending payload: ");
   Serial.println(payload);
   if (client.publish(publishTopic, (char*) payload.c_str())) {
     Serial.println("Publish OK");
     } else {
       Serial.println("Publish FAILED");
       }
  }
  if(distance1<=50){
   LCD.setCursor(0,0);
   LCD.print("Dustbin is open ");
   servo.write(90);
  }
  else{
   LCD.setCursor(0,0);
   LCD.print("Dustbin is close ");
   servo.write(0);
  }
}
```