DELIVERY OF SPRINT – 3

Team ID	PNT2022TMID04344
Project Name	Project –Smart Waste management System
Date	11 November 2022
Marks	4 Marks

Transferring the data from the sensors to the IBM IoT Watson cloud

I. Ino code :

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

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

#define ORG "ede5q6"                    // IBM organisation id
#define DEVICE_TYPE "MyTesting"              // Device type mentioned in ibm watson iot
platform
#define DEVICE_ID "12345"              // Device ID mentioned in ibm watson iot platform
#define TOKEN "lrTLbr*22xoKqc)Wko"        // 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(4, 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;
  return 300;
}

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);

  }
  else
  {
    digitalWrite(15, LOW);
  }

 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 > 150 && cm < 250)
  {
    digitalWrite(4, HIGH);
    Serial.println("Warning!!,Trash is about to cross 50% of bin level");
    digitalWrite(2, LOW);
    digitalWrite(23, LOW);
  }
  else if(cm > 250 && cm <=400)
  {
    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");
 }




  if(cm <= 100)
{
digitalWrite(21,HIGH);
// String payload = "{\"HighAlert !Trash bin is about to be full\":\"";
// payload += cm;
// payload += "\" }";
String payload="{\"data\":";
payload+="\"HighAlert !Trash bin is about to be full\"";
payload+="," "\"gap\":";
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");
}
}
///////////////////////////

if(cm > 150 && cm < 250)
{
digitalWrite(22,HIGH);
String payload="{\"data\":";
payload+="\"warning! Trash is about to cross 50% of bin level\"";
payload+="," "\"gap\":";
payload+=cm;
payload+="}";
// String payload="";
// // String payload = "{\"warning! Trash is about to cross 50% of bin leve\":\"";
// payload += cm;
// payload += "\" }";
Serial.print("\n");
Serial.print("Sending distance: ");
Serial.println(cm);
if(client.publish(publishTopic, (char*) payload.c_str()))
{
Serial.println("Publish OK");
}
else
{
Serial.println("Publish FAILED");
}
}
///////////////////////////

if(cm > 250 && cm <=400)
{
digitalWrite(21,HIGH);

// String data="Bin is available";
// float gap=cm;
// String payload="{\Data"
// String data="{\"data\":\"Bin is available\" ,";
// String payload = "\"load\":\"";
String payload="{\"data\":";
payload+="\"Trash is in low level(Space available)\"";
payload+="," "\"gap\":";
payload+=cm;
payload+="}";
// payload += cm;
// payload += "\" }";
// data+=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();
}
```

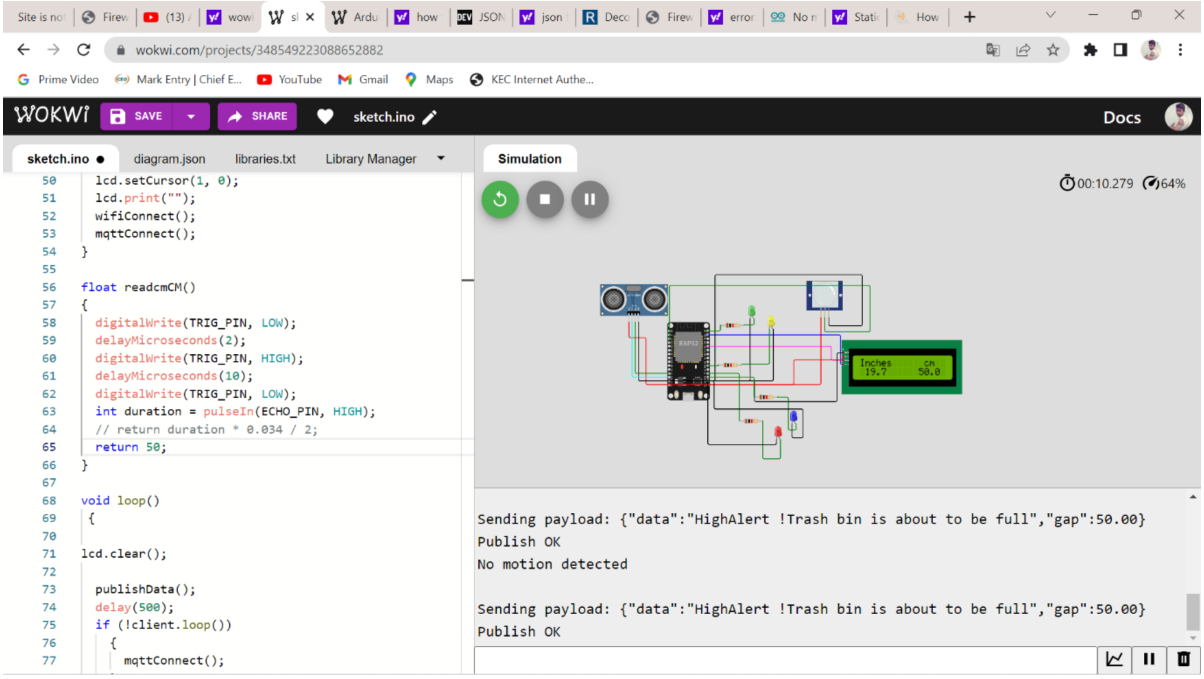
Diagram.json

```
{
  "version": 1,
  "author": "ganesh Kumar",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 76, "left": 19.31, "attrs": {} },
    {
      "type": "wokwi-hc-sr04",
      "id": "ultrasonic1",
      "top": -16.86806233723958,
      "left": -148.66666666666669,
      "attrs": {}
    },
    {
      "type": "wokwi-pir-motion-sensor",
      "id": "pir1",
      "top": -25.33,
      "left": 367.47,
      "attrs": {}
    },
    {
      "type": "wokwi-lcd1602",
      "id": "lcd1",
      "top": 122.59,
      "left": 454.48,
      "attrs": { "pins": "i2c" }
    },
    {
      "type": "wokwi-led",
      "id": "led1",
      "top": 28.02,
      "left": 211.04,
      "attrs": { "color": "limegreen" }
    },
    {
      "type": "wokwi-resistor",
      "id": "r1",
      "top": 76.14,
      "left": 152.31,
      "attrs": { "value": "1000" }
    },
    {
      "type": "wokwi-led",
      "id": "led2",
      "top": 53.99,
      "left": 259.59,
      "attrs": { "color": "yellow" }
    },
    {
      "type": "wokwi-resistor",
      "id": "r2",
      "top": 175.42,
      "left": 147.07,
      "attrs": { "value": "1000" }
    },
    {
      "type": "wokwi-led",
      "id": "led3",
      "top": 330.14,
      "left": 277.16,
      "attrs": { "color": "red" }
    },
    {
      "type": "wokwi-resistor",
      "id": "r3",
      "top": 315.38,
      "left": 199.1,
      "attrs": { "value": "1000" }
    },
    {
      "type": "wokwi-led",
      "id": "led4",
      "top": 291.12,
      "left": 316.19,
      "attrs": { "color": "blue" }
    },
    {
      "type": "wokwi-resistor",
      "id": "r4",
      "top": 255.49,
      "left": 236.26,
      "attrs": { "value": "1000" }
    }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "ultrasonic1:VCC", "esp:VIN", "red", [ "v39.24", "h42.88", "v6" ] ],
    [ "ultrasonic1:GND", "esp:GND.2", "black", [ "v0" ] ],
    [ "ultrasonic1:ECHO", "esp:D12", "green", [ "v0" ] ],
    [ "ultrasonic1:TRIG", "esp:D13", "cyan", [ "v0" ] ],
    [ "pir1:VCC", "esp:3V3", "red", [ "v0" ] ],
    [ "lcd1:SCL", "esp:D22", "blue", [ "h-5.67", "v-201.1" ] ],
    [ "pir1:GND", "esp:GND.2", "black", [ "v20.74", "h84.02", "v-128", "h-368.67",
"v151.33" ] ],
    [ "pir1:OUT", "esp:D34", "green", [ "v36.07", "h112.81", "v-116.67", "h-376.67" ] ],
    [ "lcd1:SDA", "esp:D21", "magenta", [ "h-28.61", "v-51.54" ] ],
    [ "esp:VIN", "lcd1:VCC", "red", [ "h311.3", "v-61.84", "h0.82" ] ],
    [ "lcd1:GND", "esp:GND.1", "black", [ "h-14.75", "v122.42", "h-206.35", "v-57.91" ] ],
    [ "led1:C", "esp:GND.1", "black", [ "v106.72", "h0.28", "v-4.08" ] ],
    [ "led1:A", "r1:2", "green", [ "v0" ] ],
    [ "r1:1", "esp:D23", "green", [ "v0" ] ],
    [ "esp:RX2", "r2:1", "green", [ "h0" ] ],
    [ "r2:2", "led2:C", "green", [ "v-1.37", "h56.81" ] ],
    [ "led2:A", "esp:GND.1", "black", [ "v0" ] ],
    [ "led3:A", "r3:2", "green", [ "v47.79", "h-31.84" ] ],
    [ "r3:1", "esp:D2", "green", [ "v0" ] ],
    [ "led3:C", "esp:GND.1", "black", [ "v11.93", "h-135.76" ] ],
    [ "led4:A", "r4:2", "green", [ "v15.09", "h-19.18", "v-82.28", "h-3.16" ] ],
    [ "r4:1", "esp:D15", "green", [ "v0" ] ],
    [ "led4:C", "esp:GND.1", "black", [ "v34.07", "h27.74", "v-71.73", "h-222.57", "v-67.51" ]
]
  ]
}
```

Libraries.txt

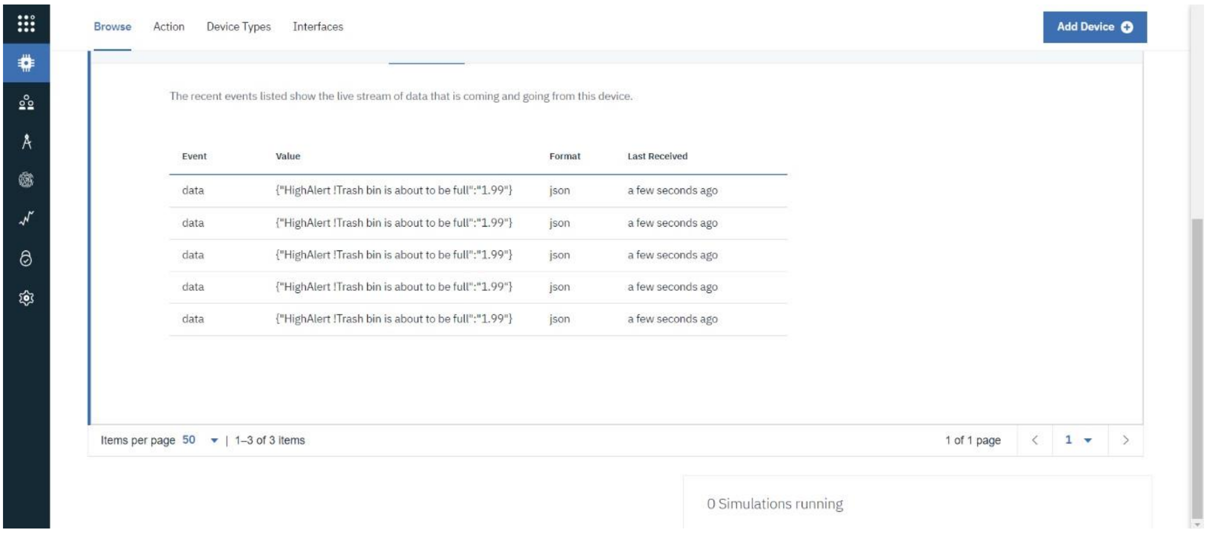
```
# Wokwi Library List
# See https://docs.wokwi.com/guides/libraries

# Automatically added based on includes:
LiquidCrystal I2C

PubSubClient
ArduinoJson
```

IV.Output on cloud, when Bin is fulled/ about to get filled



```
  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;
  return 50;
}

void loop()
 {

lcd.clear();

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

```
Sending payload: {"data":"HighAlert !Trash bin is about to be full","gap":50.00}
Publish OK
No motion detected

Sending payload: {"data":"HighAlert !Trash bin is about to be full","gap":50.00}
Publish OK
```



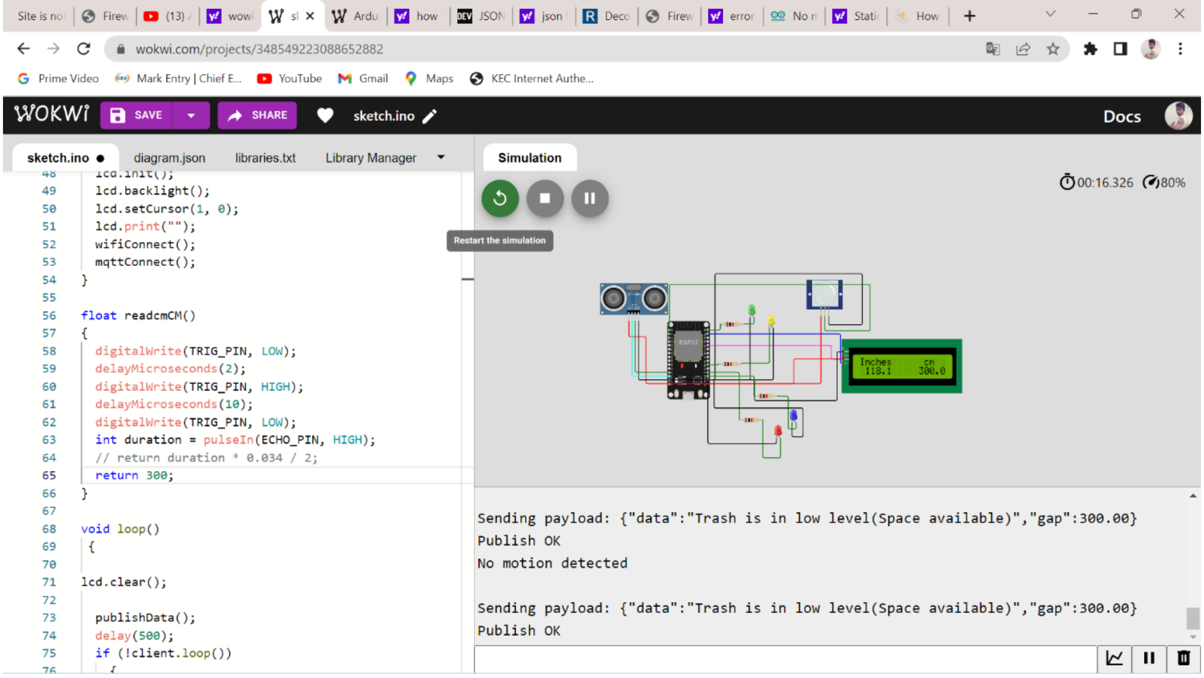
The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
data	{"HighAlert !Trash bin is about to be full":"1.99"}	json	a few seconds ago
data	{"HighAlert !Trash bin is about to be full":"1.99"}	json	a few seconds ago
data	{"HighAlert !Trash bin is about to be full":"1.99"}	json	a few seconds ago
data	{"HighAlert !Trash bin is about to be full":"1.99"}	json	a few seconds ago
data	{"HighAlert !Trash bin is about to be full":"1.99"}	json	a few seconds ago

Items per page 50 | 1–3 of 3 items   1 of 1 page   1

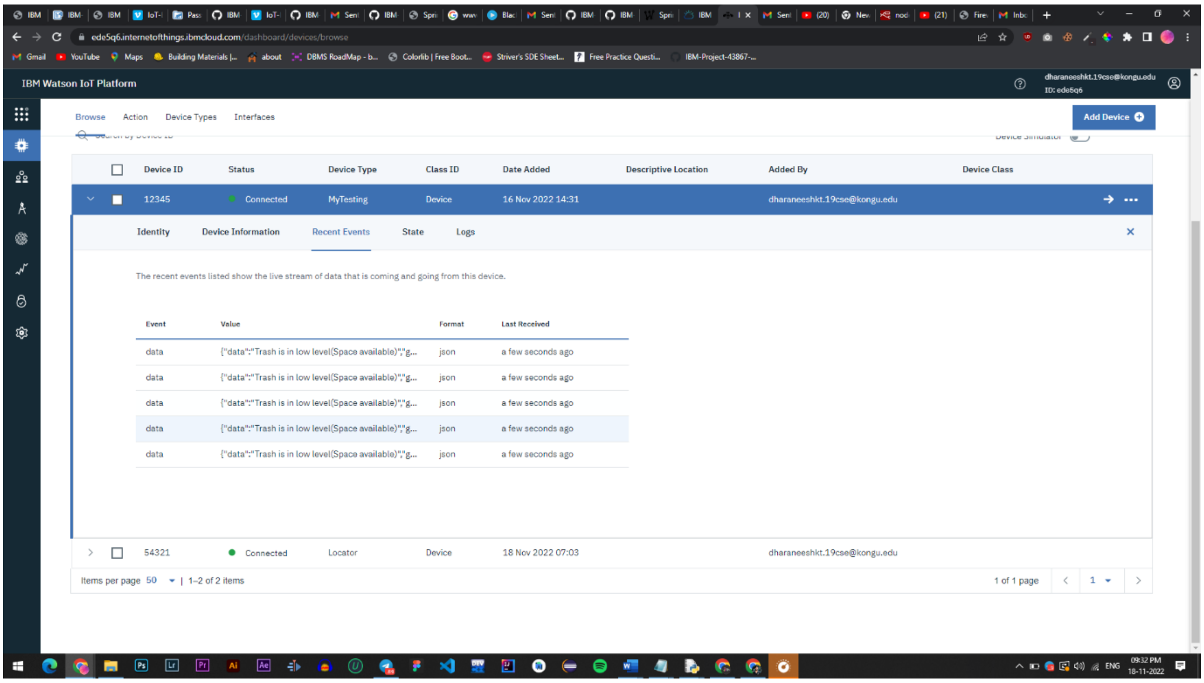
0 Simulations running

V.Output on cloud, when Bin is Empty



```
  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;
  return 300;
}

void loop()
 {

lcd.clear();

  publishData();
  delay(500);
  if (!client.loop())
    {
```

```
Sending payload: {"data":"Trash is in low level(Space available)","gap":300.00}
Publish OK
No motion detected

Sending payload: {"data":"Trash is in low level(Space available)","gap":300.00}
Publish OK
```



Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By	Device Class
12345	● Connected	MyTesting	Device	16 Nov 2022 14:31		dharaneeshkt.19cse@kongu.edu	
54321	● Connected	Locator	Device	18 Nov 2022 07:03		dharaneeshkt.19cse@kongu.edu	

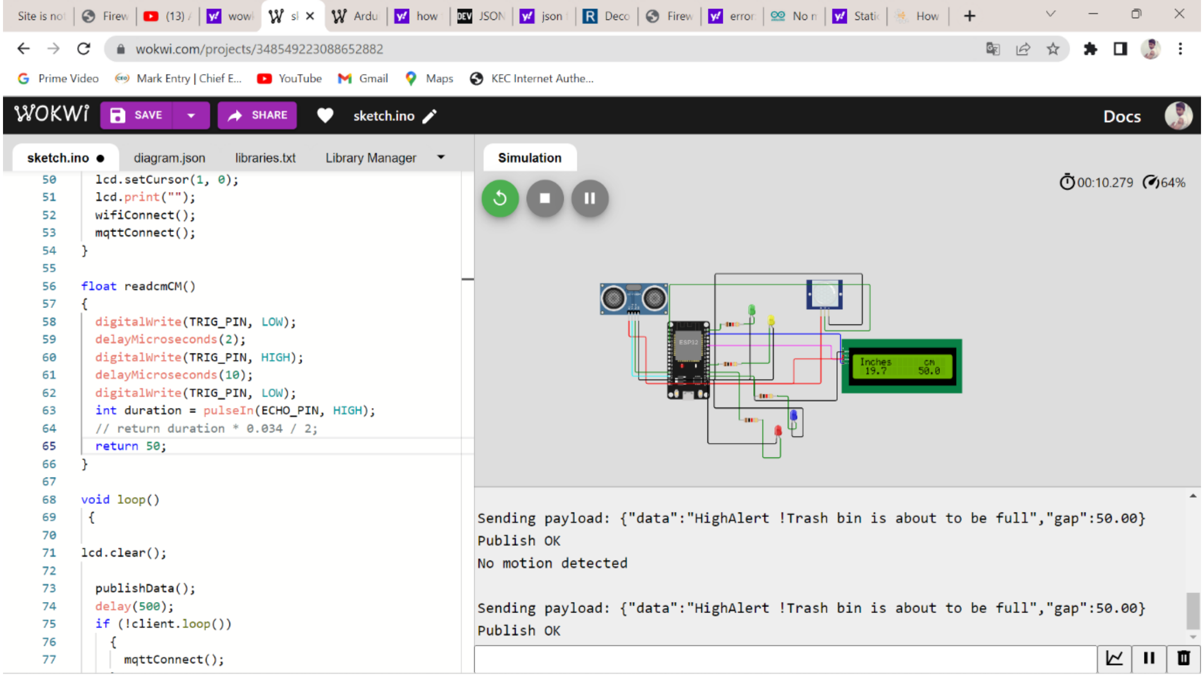
The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
data	{"data":"Trash is in low level(Space available)","g...	json	a few seconds ago
data	{"data":"Trash is in low level(Space available)","g...	json	a few seconds ago
data	{"data":"Trash is in low level(Space available)","g...	json	a few seconds ago
data	{"data":"Trash is in low level(Space available)","g...	json	a few seconds ago
data	{"data":"Trash is in low level(Space available)","g...	json	a few seconds ago

Items per page 50 | 1–2 of 2 items

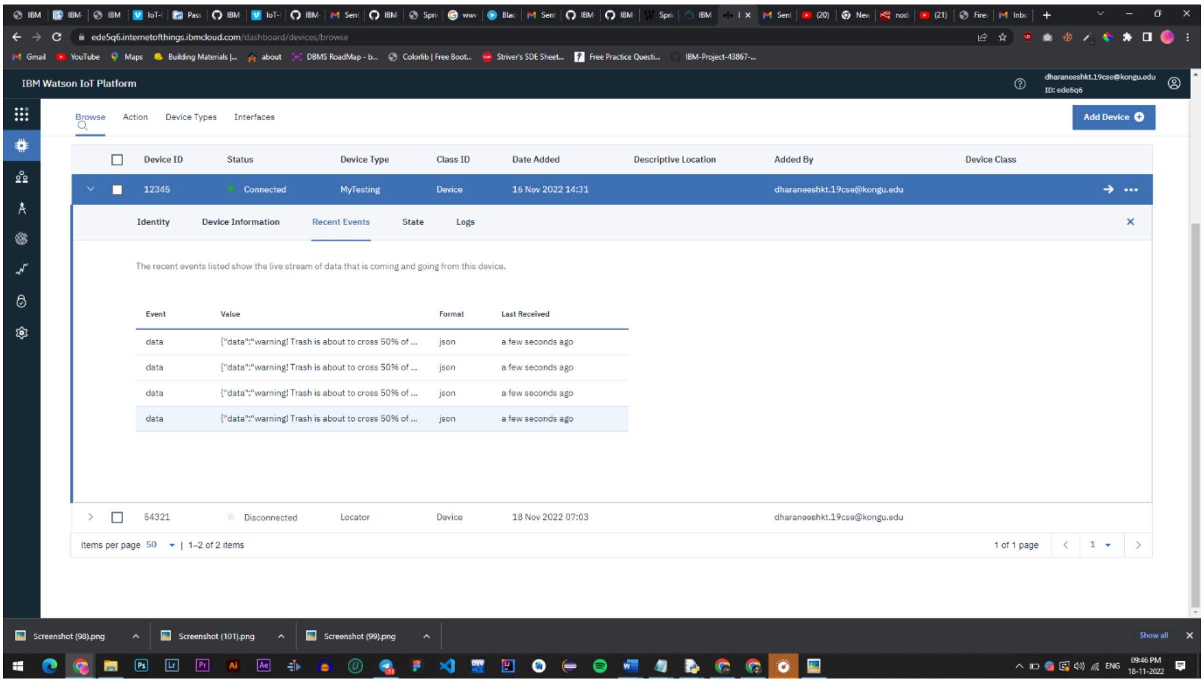
1 of 1 page

VI.Output on cloud, when Bin is about to cross 50% of storage



```
  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;
  return 50;
}

void loop()
 {

lcd.clear();

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

```
Sending payload: {"data":"HighAlert !Trash bin is about to be full","gap":50.00}
Publish OK
No motion detected

Sending payload: {"data":"HighAlert !Trash bin is about to be full","gap":50.00}
Publish OK
```



	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By	Device Class
□	12345	Connected	MyTesting	Device	16 Nov 2022 14:31		dharaneeshkt.19cse@kongu.edu	
□	54321	Disconnected	Locator	Device	18 Nov 2022 07:03		dharaneeshkt.19cse@kongu.edu	

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
data	{"data":"warning! Trash is about to cross 50% of ...	json	a few seconds ago
data	{"data":"warning! Trash is about to cross 50% of ...	json	a few seconds ago
data	{"data":"warning! Trash is about to cross 50% of ...	json	a few seconds ago
data	{"data":"warning! Trash is about to cross 50% of ...	json	a few seconds ago

Items per page 50 | 1–2 of 2 items   1 of 1 page