

SPRINT – 4

DATE	19 NOVEMBER 2022
TEAM ID	PNT2022TMID15274
PROJECT NAME	SMART WASTE MANAGEMENT FOR METROPOLITAN CITIES-IOT

WOWKI OUTPUT :

The screenshot shows the IBM Watson IoT Platform dashboard. A modal window is open for configuring a device type named 'lavi123'. The 'Events' section is active, showing a single event type named 'Sensor'. The 'Schedule' is set to '20' and 'Every Minute'. The 'Payload' is a JSON object with three fields: 'randomNumber', 'level', and 'weight', each with a random value between 0 and 100. The background shows a table of recent events for the device '12345'.

Event	Value	Format	Last Received
Sensor	{"randomNumber":34,"level":80,"weight":55}	json	a few seconds ago
Sensor	{"randomNumber":77,"level":62,"weight":12}	json	a few seconds ago
Sensor	{"randomNumber":32,"level":47,"weight":39}	json	a few seconds ago
Sensor	{"randomNumber":37,"level":22,"weight":74}	json	a few seconds ago
Sensor	{"randomNumber":71,"level":21,"weight":42}	json	a few seconds ago

The screenshot shows the Wokwi IDE interface. On the left, the Arduino code for 'esp32-blink.ino' is displayed. The code includes libraries for WiFi, PubSubClient, LiquidCrystal_I2C, and cJSON. It defines IBM credentials and MQTT topics. The main loop sends a JSON payload to an MQTT broker. On the right, a simulation of the ESP32 circuit is shown, including a breadboard, an LCD display, and various components. The LCD display shows 'Inches 66.9 cm 170.0'. The console output shows the MQTT client reconnecting and sending payloads.

```

1 #include <WiFi.h> // library for wifi
2 #include <PubSubClient.h> // library for MQTT
3 #include <LiquidCrystal_I2C.h>
4 #include <cjson.h>
5 LiquidCrystal_I2C lcd(0x27, 20, 4);
6
7 //----- credentials of IBM Accounts -----
8
9 #define ORG "s1e201" // IBM organisation id
10 #define DEVICE_TYPE "lavi123" // Device type mentioned in ibm watson iot platform
11 #define DEVICE_ID "12345" // Device ID mentioned in ibm watson iot platform
12 #define TOKEN "23456789" // Token
13
14 //----- customise above values -----
15
16 char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; // server name
17 char publishTopic[] = "iot-2/evt/data/fmt/json"; // topic name
18 char topic[] = "iot-2/cmd/led/fmt/String"; // cmd Representation
19 char authMethod[] = "use-token-auth"; // authentication method
20 char token[] = TOKEN;
21 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //Client id
22
23 //-----
24
25 WiFiClient wificlient; // creating a WiFi client
26 PubSubClient client(server, 1883, wificlient);
27
28 #define ECHO_PIN 12
29 #define TRIG_PIN 13
30 float dist;
31 String data3;
32 void setup()
33 {
34   Serial.begin(115200);
35   pinMode(LED_BUILTIN, OUTPUT);
36   pinMode(TRIG_PIN, OUTPUT);
  
```

Simulation console output:

```

Reconnecting MQTT client to s1e201.messaging.internetofthings.ibmcloud.com
IBM subscribe to cmd OK

Sending payload: {"Safe":169.97 }
Publish OK

Sending payload: {"Safe":169.97 }
Publish OK
  
```

