

## PROJECT DEVELOPMENT PHASE

### SPRINT-4

Date	17 NOVEMBER 2022
Team ID	PNT2022TMID17430
Project Name	Project - Hazardous Area Monitoring for Industrial Plant powered by IoT
Maximum Marks	4 Marks

#### CODE :

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <DHT.h>

WiFiClient wifiClient;
String data3;

#define DHTTYPE DHT11
#define DHTPIN 4
#define MQTPIN 34
DHT dht(DHTPIN, DHTTYPE);

#define ORG "v6wg8x"
#define DEVICE_TYPE "projectFinal"
#define DEVICE_ID "FinalDeliverable"
#define TOKEN "A1ymH))p*JB&iMWNpY"
#define speed 0.034

void callback(char* topic, byte* payload, unsigned int payloadLength);

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char topic[] = "iot-2/cmd/test/fmt/String";
char authMethod[] = "use-token-auth";
```

```
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, callback , wifiClient);
void publishData();
```

```
String command;
String data = "";
```

```
long duration;
float dist;
```

```
void setup()
{
  Serial.begin(115200);
  dht.begin();
  wifiConnect();
  mqttConnect();
}
```

```
void loop() {
  publishData();
  delay(500);

  if (!client.loop()) {
    mqttConnect();
  }
}
```

```
void wifiConnect() {
  Serial.print("Connecting to "); Serial.print("Wifi");
  WiFi.begin("JerroldWi-Fi","75779901");
  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()
{
    int sensorValue = analogRead(MQTPIN); //MQT 135 connected to GPIO 34 (Analog ADC1_CH6)
    Serial.print("AirQua=");
    Serial.print(sensorValue, DEC);
    Serial.println(" PPM");
    float humid = dht.readHumidity();
    float temp = dht.readTemperature(true);

```

```

String payload = "{\"Humidity\":\"";
payload += humid;
payload += "\"}";
if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish OK");
}
payload = "{\"Temperature\":\"";
payload += temp;
payload += "\"}";
if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish OK");
}
payload = "{\"Air Quality\":\"";
payload += String(sensorValue);
payload += "\"}";
if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish OK");
}
}

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++) {
        dist += (char)payload[i];
    }
    Serial.println("data:" + data3);
    if (data3 == "lighton") {
        Serial.println(data3);
    }
    data3 = "";
}

```