**Assignment 4**

|  |  |
| --- | --- |
| Team ID | PNT2022TMID15971 |
| Project Name | Hazardous Area Monitoring for Industrial Plant powered by IoT |
| Team Members | Deva Dharshini S, Amancharla Vishnupriya, Dandolu Sujitha, V R Aruna |

#include <WiFi.h>

#include <PubSubClient.h>

#include <ArduinoJson.h>

WiFiClient wifiClient;

#define ORG "mxyrim"

#define DEVICE\_TYPE "NodeMCU"

#define DEVICE\_ID "12345"

#define TOKEN "12345678"

#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=5;

const int echopin=18;

String command;

String data="";

String lat="13.356563";

String lon="80.141428";

String name="point1";

String icon="fa-fire";

long duration;

int dist;

void setup()

{

**Serial**.begin(115200);

  pinMode(trigpin, OUTPUT);

  pinMode(echopin, 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(1000);

    }

    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(trigpin,LOW);

  digitalWrite(trigpin,HIGH);

  delayMicroseconds(10);

  digitalWrite(trigpin,LOW);

  duration=pulseIn(echopin,HIGH);

  dist=duration\*speed/2;

  dist=dist/4;

  dist=100-dist;

  if(dist>80){

    lat="13.356563";

    lon="80.141428";

  }else{

    lat="0.000000";

    lon="0.000000";

  }

  DynamicJsonDocument doc(1024);

  String payload;

  doc["Name"]=name;

  doc["Latitude"]=lat;

  doc["Longitude"]=lon;

  doc["Icon"]=icon;

  doc["GasPercent"]=dist;

  serializeJson(doc, payload);

  delay(3000);

**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");

  }

}



