

ASSIGNMENT 4

Date	2 Nov 22
Name	Nirmal Kumar A
Team ID	PNT2022TMID38289
Project Name	SmartFarmer - IoT Enabled Smart Farming Application

QUESTION :

Write code and connection in wovki for ultrasonic sensor. Whenever distance is less than 100 cms send “alert” to IBM cloud and display in device recent events

CODE :

```
#include <WiFi.h> //library for wifi
#include <PubSubClient.h> //library for MQTT
WiFiClient wifiClient;
String data3;
#define ORG "iuyrs1"
#define DEVICE_TYPE "Nirmal"
#define DEVICE_ID "1908"
#define TOKEN "!F5Z0g2D*aR&(Hp?Kr"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/ Nirmal /fmt/json";
char topic[] = "iot-2/cmd/event_1/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient);

const int trigpin=5;
const int echopin=18;
```

```

String command;
String data="";

long duration;
float dist;

void setup()
{
  Serial.begin(115200);
  pinMode(led, OUTPUT);
  pinMode(trigpin, OUTPUT);
  pinMode(echopin, INPUT);
  wifiConnect();
  mqttConnect();
}

void loop(){
  bool isNearby = dist < 100;
  digitalWrite(led, isNearby);

  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("IBM 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;
    if(dist<100){
        String payload = "{\"Alert Distance\":";
        payload += dist;
        payload += "}";
        Serial.print("\n");
        Serial.print("Sending payload: ");
        Serial.println(payload);
        if (client.publish(publishTopic, (char*) payload.c_str())){
            Serial.println("Publish OK");
        }
    }
    if(dist>100){
        String payload = "{\"Distance\":";
        payload += dist;
        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");
    }
}
}

```


OUTPUT :

1) When Distance greater than 100 cm

The screenshot shows the Wokwi IoT simulator interface. On the left, the sketch code is displayed, which includes MQTT client libraries and defines variables for an IBM IoT device. The code publishes distance data to a specific topic. On the right, the simulation window shows the ESP32 hardware connected to a sensor module and a red LED. The console output indicates the device has connected to WiFi and successfully published a payload: {"Distance":399.98}.

```
1 #include <WiFi.h> //library for wifi
2 #include <PubSubClient.h> //library for MQTT
3 WiFiClient wificlient;
4 String data3;
5 #define ORG "iuyrs1"
6 #define DEVICE_TYPE "Nirmal"
7 #define DEVICE_ID "1908"
8 #define TOKEN "!F520g2D*aR&(Hp?Kp"
9 #define speed 0.034
10 #define led 14
11 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
12 char publishTopic[] = "iot-2/evt/Nirmal/fmt/json";
13 char topic[] = "iot-2/cmd/event_1/fmt/String";
14 char authMethod[] = "use-token-auth";
15 char token[] = TOKEN;
16 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
17 PubSubClient client(server, 1883, wificlient);
18
19
20
21 const int trigpin=5;
22 const int echopin=18;
23 String command;
24 String data="";
25
26 long duration;
27 float dist;
```

Simulation console output:

```
Connecting to Wifi...WiFi connected, IP address: 10.10.0.2
Reconnecting MQTT client to
iuyrs1.messaging.internetofthings.ibmcloud.com
IBM subscribe to cmd OK

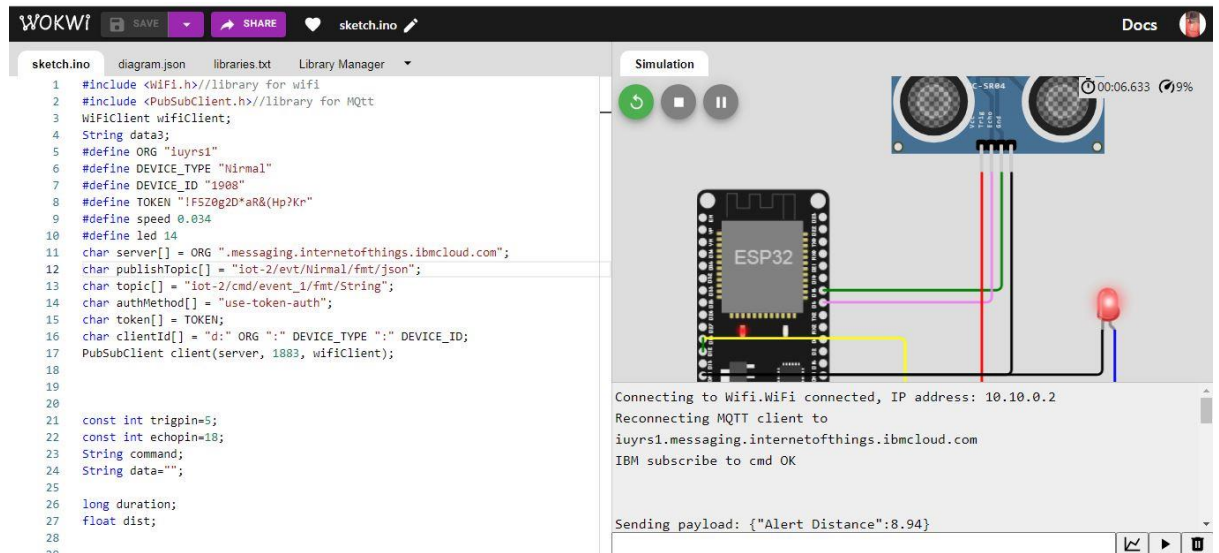
Sending payload: {"Distance":399.98}
Publish OK
```

IBM RECENT EVENTS

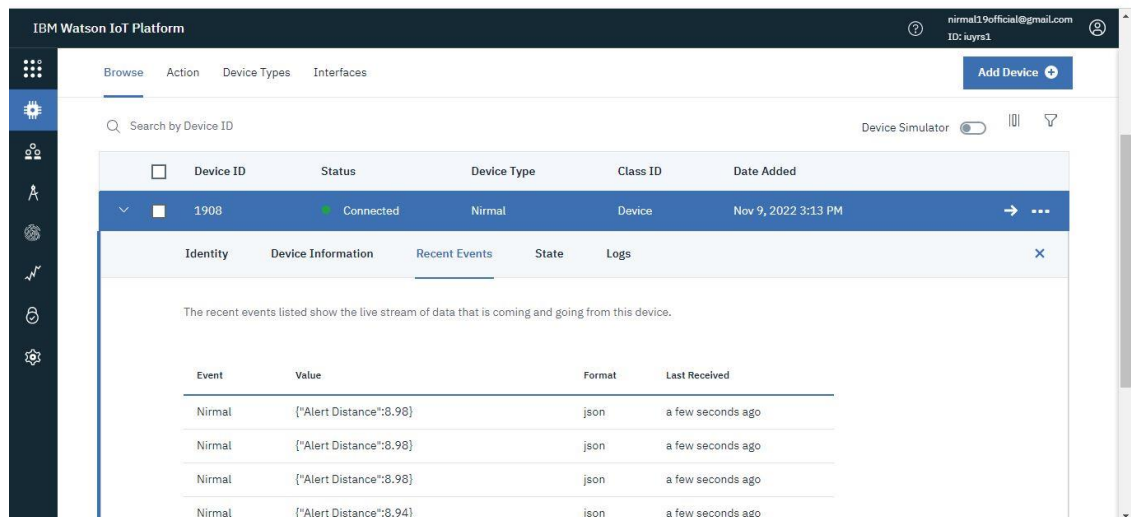
The screenshot shows the IBM IoT dashboard interface. The 'Recent Events' tab is selected for device 1908. The table below displays the live stream of data events.

Event	Value	Format	Last Received
Nirmal	{"Distance":399.96}	json	a few seconds ago
Nirmal	{"Distance":399.96}	json	a few seconds ago
Nirmal	{"Distance":399.94}	json	a few seconds ago
Nirmal	{"Distance":399.96}	json	a few seconds ago
Nirmal	{"Distance":400.23}	json	a few seconds ago

2) When distance less than 100



IBM RECENT EVENTS



WOKWI LINK-

<https://wokwi.com/projects/347781259790910034>