**IoT BASED SAFETY GADGET FOR CHILD SAFETY MONITORING AND NOTIFICATION**

ASSIGNMENT-4

Submitted by

P. Subitha

**Recent events. Write code and connection in wokwi for ultrasonic sensor when ever distance is less than 100 cms send alert to ibm cloud and display in device**

#include <WiFi.h>

#include <PubSubClient.h>

#define TRIGGER 2

#define ECHO 15

#define sound\_speed 0.034

int distance;

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

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

#define ORG "wp72r7"

#define DEVICE\_TYPE "iot-device-1"

#define DEVICE\_ID "123456789"

#define TOKEN "987654321"

String data3;

//-------- Customise the above values --------

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";

char publishTopic[] = "iot-2/evt/Data/fmt/json";

char subscribetopic[] = "iot-2/cmd/test/fmt/String";

char authMethod[] = "use-token-auth";

char token[] = TOKEN;

char clientId[] = "d:" ORG ":" DEVICE\_TYPE ":" DEVICE\_ID;

//-----------------------------------------

WiFiClient wifiClient;

PubSubClient client(server, 1883, callback ,wifiClient);

void setup()

{

Serial.begin(115200);

pinMode(TRIGGER, OUTPUT);

pinMode(ECHO, INPUT);

delay(10);

Serial.println();

wificonnect();

mqttconnect();

}

void loop()

{

digitalWrite(TRIGGER, HIGH);

delayMicroseconds(10);

digitalWrite(TRIGGER, LOW);

int duration=pulseIn(ECHO,HIGH);

distance=(duration\*sound\_speed)/2;

Serial.print("Distance:");

Serial.print(distance);

Serial.println("cms");

if(distance<100){

PublishData(distance);

}

delay(1000);

if (!client.loop()) {

mqttconnect();

}

}

/\*.....................................retrieving to Cloud...............................\*/

void PublishData(int d) {

mqttconnect();

String payload = "{\"message\":alert}";

Serial.print("Sending payload: ");

Serial.println(payload);

if (client.publish(publishTopic, (char\*) payload.c\_str())) {

Serial.println("Publish ok");

} else {

Serial.println("Publish failed");

}

}

void mqttconnect() {

if (!client.connected()) {

Serial.print("Reconnecting client to ");

Serial.println(server);

while (!!!client.connect(clientId, authMethod, token)) {

Serial.print(".");

delay(500);

}

initManagedDevice();

Serial.println();

}

}

void wificonnect()

{

Serial.println();

Serial.print("Connecting to ");

WiFi.begin("Wokwi-GUEST", "", 6);

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.println("WiFi connected");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

}

void initManagedDevice() {

if (client.subscribe(subscribetopic)) {

Serial.println((subscribetopic));

Serial.println("subscribe to cmd OK");

} else {

Serial.println("subscribe to cmd FAILED");

}

}

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++) {

data3 += (char)payload[i];

}

Serial.println("data: "+ data3);

data3="";

}