

```

#include <WiFi.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>

WiFiClient wifiClient;

#define ORG "oa3490"
#define DEVICE_TYPE "TestDeviceType"
#define DEVICE_ID "12345"
#define TOKEN "-A)0raS44f)fdjYBVS"
#define speed 0.034

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/abcd_1/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="14.167589";
String lon="80.248510";
String name="point2";
String icon="";

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;
    if(dist<100){
        dist=100- dist; icon="fa- trash";
    }
}

```

```

}else{ dist=0;
    icon="fa-trash- o";
}

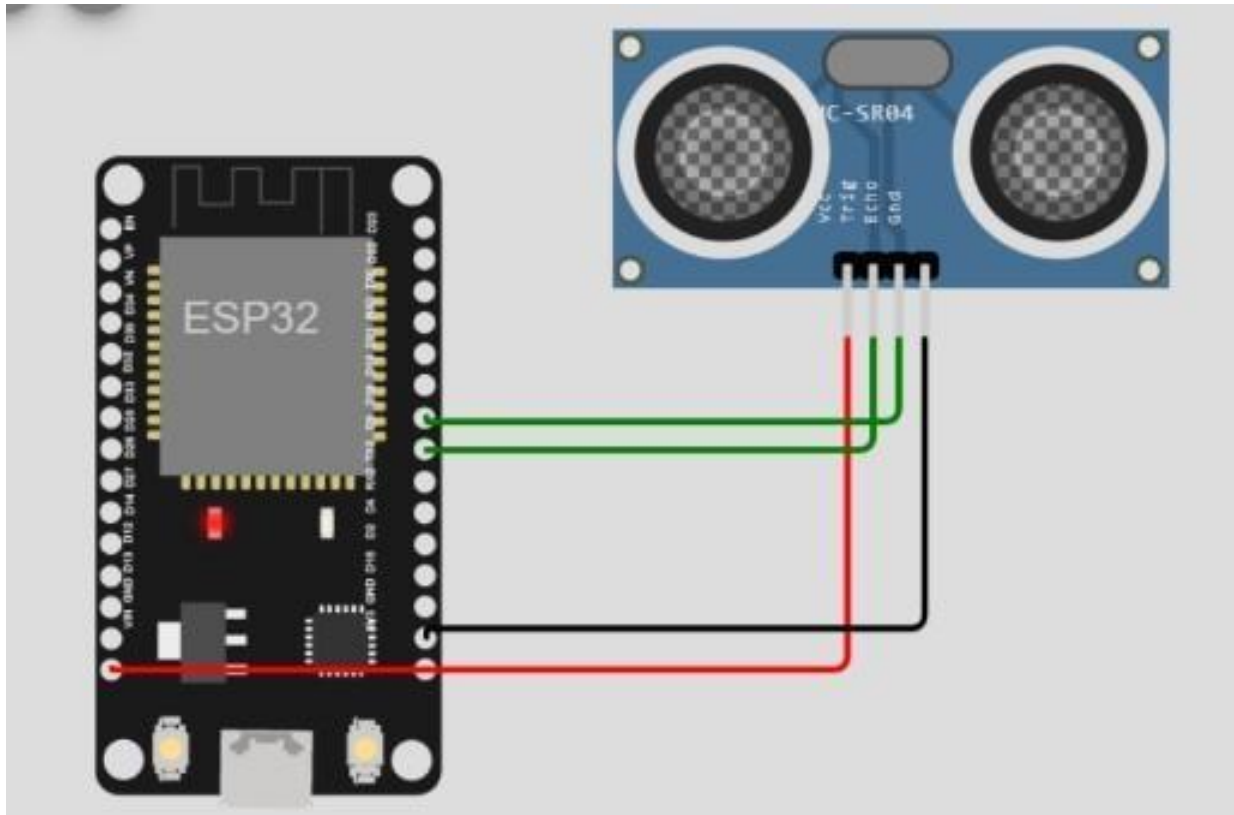
DynamicJsonDocument doc(1024) ;
String payload; doc["Name"]=
name; doc["Latitude"]= lat;
doc["Longitude"]= lon;
doc["Icon"]= icon;
doc["FillPercent"]= dist;
serializeJson(doc, payload);
delay(3000) ; Serial.print("\n")
;

Serial.print("Sending payload: ") ; Serial.println( payload);

if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish OK") ;
} else {
    Serial.println("Publish FAILED") ;
}
}

```

## Connections:



**Output:**

**WOKWI** SAVE SHARE ♥

sketch.ino diagram.json libraries.txt Library Manager

```

1 #include <wifi.h>
2 #include <PubSubClient.h>
3 #include <ArduinoJson.h>
4
5 WiFiClient wificlient;
6
7 #define ORG "0a3490"
8 #define DEVICE_TYPE "TestDevicetype"
9 #define DEVICE_ID "12345"
10 #define TOKEN "-A)0raS44f)fdjyBVS"
11 #define speed 0.034
12
13 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
14 char publishTopic[] = "iot-2/evt/abcd_1/fmt/json";
15 char topic[] = "iot-2/cmd/home/fmt/String";
16 char authMethod[] = "use-token-auth";
17 char token[] = TOKEN;
18 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
19 PubSubClient client(server, 1883, wificlient);
20 void publishData();
21
22 const int trigpin=5;
23 const int echopin=18;
24 String command;
25 String data="";
26 String lat="14.167589";
27 String lon="80.248510";
28 String name="point2";
29 String icon="";
30
31 long duration;
32 int dist;
33
34 void setup()
35 {

```

**Simulation** Docs SIGN IN

00:36.677 10

Editing Ultrasonic Distance Sensor  
Distance: 94cm

trash,"FillPercent":6}  
Publish OK

Sending payload:  
{ "Name": "point2", "Latitude": "14.167589", "Longitude": "80.248510", "Icon": "fa-trash", "FillPercent": 6 }  
Publish OK

**Output :( IBM Cloud)**

Microsoft Edge DevTools interface showing the 'Add Device' page. The page displays a table of device information for a device named '12345'.

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
12345	Disconnected	TestDeviceType	Device	Oct 25, 2022 12:17 PM	

Below the table, there are tabs for Identity, Device Information, Recent Events, State, and Logs. The 'Recent Events' tab is selected, showing a list of events with columns for Event, Value, Format, and Last Received.

Event	Value	Format	Last Received
event_1	{"Alert Distance":8}	json	a few seconds ago
event_1	{"Alert Distance":81}	json	a few seconds ago
event_1	{"Alert Distance":56}	json	a few seconds ago
event_1	{"Alert Distance":98}	json	a few seconds ago
event_1	{"Alert Distance":72}	json	a few seconds ago

The status of the device is 'Disconnected'. The page also shows a '1 Simulation running' notification at the bottom right.