WOKWI SIMULATION

Team ID	PNT2022TMID02664
Project Name	Smart Waste Management System for Metropolitan Cities

Created an IOT device to sense the level of bins and do code for the device using the Wokwi platform.

CODE:

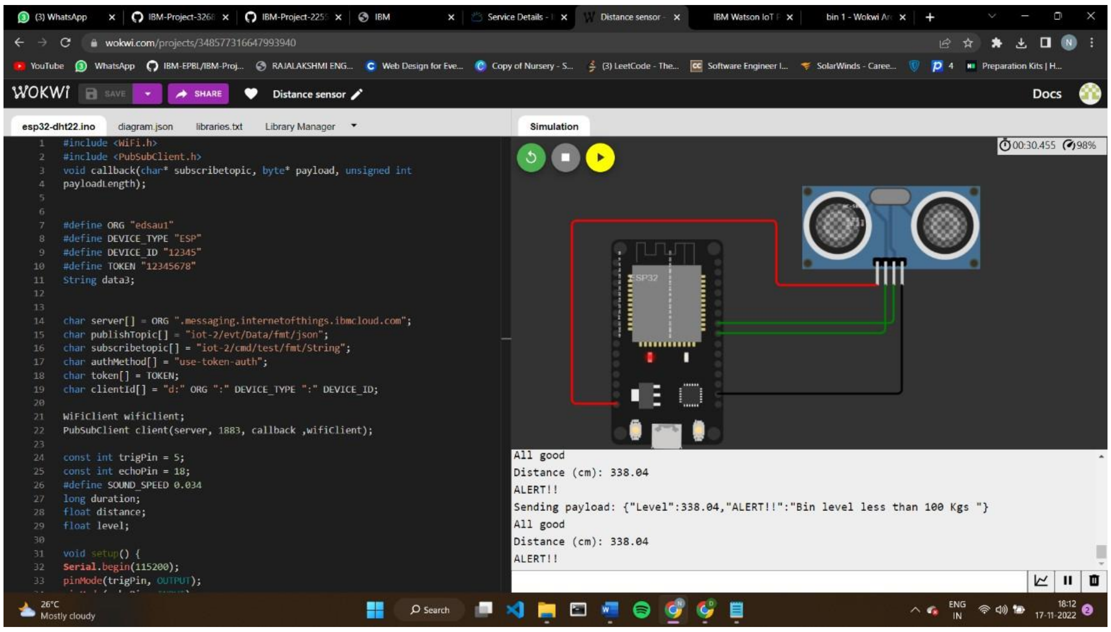
Run the code here: https://wokwi.com/projects/348577316647993940

```
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength);


#define ORG "edsau1"
#define DEVICE_TYPE "ESP"
#define DEVICE_ID "12345"
#define TOKEN "12345678"
String data3;


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);

const int trigPin = 5;
const int echoPin = 18;
#define SOUND_SPEED 0.034
long duration;
float distance;
float level;

void setup() {
Serial.begin(115200);
```

```
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
wificonnect();
mqttconnect();
}


void loop()
{
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = duration * SOUND_SPEED/2;
level = 400 - distance;
Serial.print("Distance (cm): ");
Serial.println(level);
if(level>300)
{
Serial.println("ALERT!!");
delay(1000);
PublishData(level);
delay(1000);
if (!client.loop()) {
mqttconnect();
}
}
else
{
Publishdata2(level);
delay(1000);
if (!client.loop()) {
mqttconnect();
}
}
delay(1000);
}

void PublishData(float dist) {
mqttconnect();
String payload = "{\"Level\":";
payload += dist;
payload += ",\"ALERT!!\":""\"Bin level less than 100 Kgs \"";
payload += "}";
Serial.print("Sending payload: ");
Serial.println(payload);
```

```
if (client.publish(publishTopic, (char*) payload.c_str())) {
Serial.println("All good");
} else {
Serial.println("Failed");
}
}


void Publishdata2(float dist) {
mqttconnect();
String payload = "{\"Level\":";
payload += dist;
payload += "}";
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++) {
//Serial.print((char)payload[i]);
data3 += (char)payload[i];
}
Serial.println("data: "+ data3);
data3="";
}
```

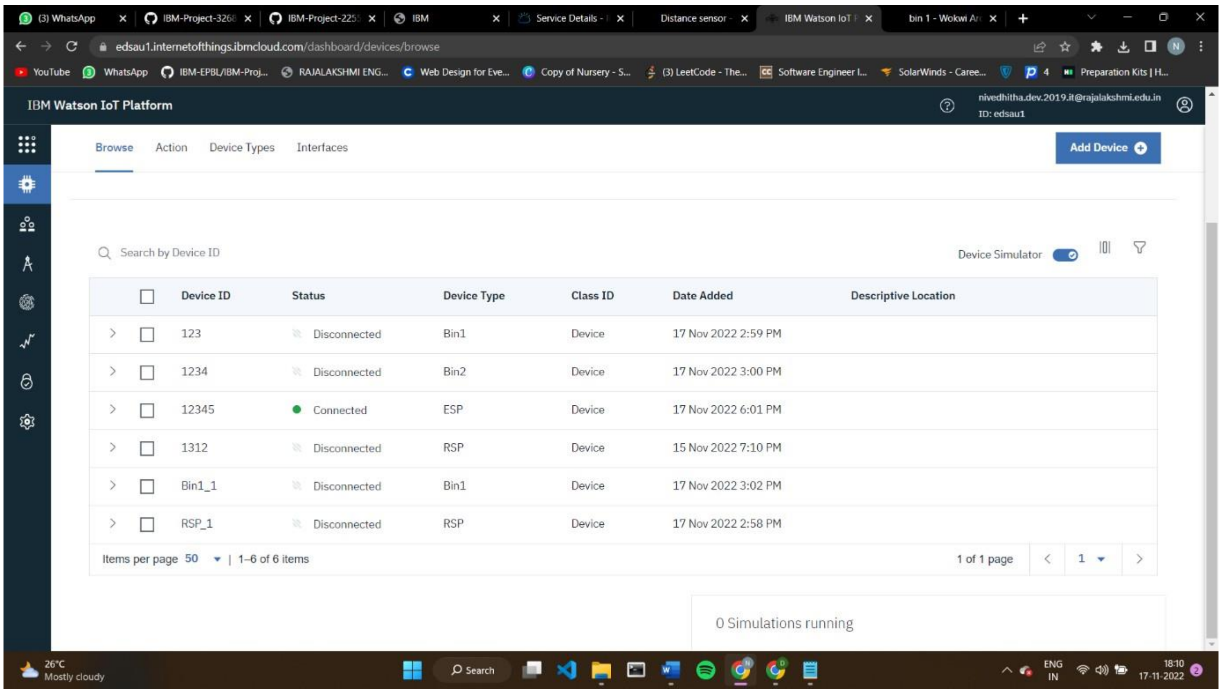
Sensor circuit:



```
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength);


#define ORG "edsau1"
#define DEVICE_TYPE "ESP"
#define DEVICE_ID "12345"
#define TOKEN "12345678"
String data3;


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);

const int trigPin = 5;
const int echoPin = 18;
#define SOUND_SPEED 0.034
long duration;
float distance;
float level;

void setup() {
Serial.begin(115200);
pinMode(trigPin, OUTPUT);
```

```
All good
Distance (cm): 338.04
ALERT!!
Sending payload: {"Level":338.04,"ALERT!!":"Bin level less than 100 Kgs "}
All good
Distance (cm): 338.04
ALERT!!
```

Watson IOT Platform:



IBM Watson IoT Platform

nivedhitha.dev.2019.it@rajalakshmi.edu.in
ID: edsau1

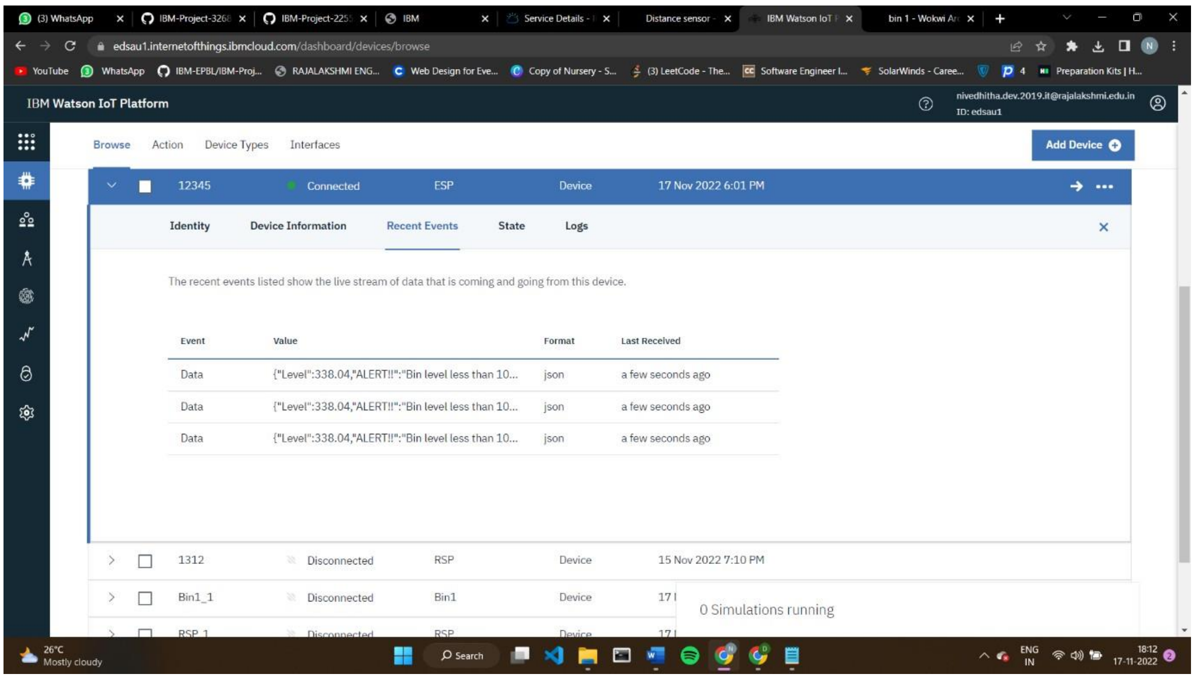
Browse | Action | Device Types | Interfaces | Add Device

Search by Device ID | Device Simulator

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
123	Disconnected	Bin1	Device	17 Nov 2022 2:59 PM	
1234	Disconnected	Bin2	Device	17 Nov 2022 3:00 PM	
12345	● Connected	ESP	Device	17 Nov 2022 6:01 PM	
1312	Disconnected	RSP	Device	15 Nov 2022 7:10 PM	
Bin1_1	Disconnected	Bin1	Device	17 Nov 2022 3:02 PM	
RSP_1	Disconnected	RSP	Device	17 Nov 2022 2:58 PM	

Items per page 50 | 1–6 of 6 items | 1 of 1 page

0 Simulations running



IBM Watson IoT Platform

nivedhitha.dev.2019.it@rajalakshmi.edu.in
ID: edsau1

Browse | Action | Device Types | Interfaces | Add Device

12345 | Connected | ESP | Device | 17 Nov 2022 6:01 PM

Identity | Device Information | Recent Events | State | Logs

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
Data	{"Level":338.04,"ALERT!!":"Bin level less than 10...	json	a few seconds ago
Data	{"Level":338.04,"ALERT!!":"Bin level less than 10...	json	a few seconds ago
Data	{"Level":338.04,"ALERT!!":"Bin level less than 10...	json	a few seconds ago

1312	Disconnected	RSP	Device	15 Nov 2022 7:10 PM
Bin1_1	Disconnected	Bin1	Device	17
RSP_1	Disconnected	RSP	Device	17

0 Simulations running