project development phase

sprint 2

Date	14 NOV 2022
Team ID	PNT2022TMID11461
Project Name	IOT Based Smart Crop Protection System For Agriculture

To detect a soil moisture

code:

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "DHT.h"// Library for dht11
#define DHTPIN 15     // what pin we're connected to
#define DHTTYPE DHT22   // define type of sensor DHT 11


DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht connected

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

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

#define ORG "ii5wx2"//IBM ORGANITION ID
#define DEVICE_TYPE "abcd"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "1234"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "12345678"    //Token
String data3;
float h, t;


//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event perform and
```

```
format in which data to be send
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command type AND COMMAND
IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id


//-------------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by
passing parameter like server id,portand wificredential


void setup()// configureing the ESP32
{
  Serial.begin(115200);
  dht.begin();

  Serial.println();
  wificonnect();
  mqttconnect();
}

void loop()// Recursive Function
{

  h = dht.readHumidity();
  t = dht.readTemperature();
  int s=random(100);
  Serial.print("temp:");
  Serial.println(t);
  Serial.print("Humid:");
  Serial.println(h);
  Serial.print("Moisture:");
  Serial.println(s);
```

```
  PublishData(t, h,s);
  delay(1000);
  if (!client.loop()) {
    mqttconnect();
  }
}



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

void PublishData(float temp, float humid,int Moisture) {
  mqttconnect();//function call for connecting to ibm
  /*
     creating the String in in form JSon to update the data to ibm cloud
  */
  String payload = "{\"temp\":";
  payload += temp;
  payload += "," "\"Humid\":";
  payload += humid;
   payload += "," "\"Moisture\":";
  payload += Moisture;
  payload += "}";


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


  if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish ok");// if it sucessfully upload data on the cloud then it will print
publish ok in Serial monitor or else it will print publish failed
  } 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() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Connecting to ");

  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the connection
  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);
 if(data3=="lighton")
  {
Serial.println(data3);

  }
  else
  {
Serial.println(data3);

  }
data3="";
}
```

WhatsApp | esp32-dht | Google Ma | IBM Cloud | Service De | IBM Watso | IBM | IBM-30185 | IBM-Proje

wokwi.com/projects/322410731508073042

PYTHON-SMTP | Akshay Saini - YouT... | Top 50 Array Codin... | Explore - LeetCode | YouTube | hackerrank | NextStep- Tata Con... | app imp | ictact | Software Engineeri...

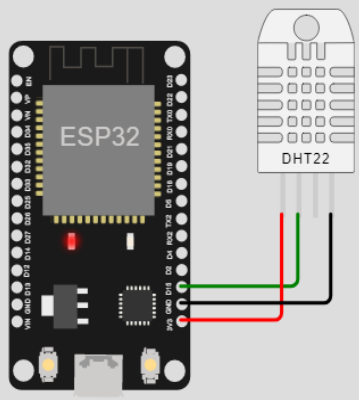
WOKWi SAVE SHARE esp32-dht22.ino by urish Docs

esp32-dht22.ino ● diagram.json libraries.txt ● Library Manager

```
68  /*.....retrieving to Cloud.....
69
70  void PublishData(float temp, float humid,int Moisture) {
71    mqttconnect();//function call for connecting to ibm
72    /*
73       creating the String in in form JSon to update the data to ibm cloud
74    */
75    String payload = "{\"temp\":";
76    payload += temp;
77    payload += "," "\"Humid\":";
78    payload += humid;
79     payload += "," "\"Moisture\":";
80    payload += Moisture;
81    payload += "}";
82
83
84    Serial.print("Sending payload: ");
85    Serial.println(payload);
86
87
88    if (client.publish(publishTopic, (char*) payload.c_str())) {
89      Serial.println("Publish ok");// if it sucessfully upload data on the clou
90    } else {
91      Serial.println("Publish failed");
92    }
93
94  }
95
96
```

Simulation

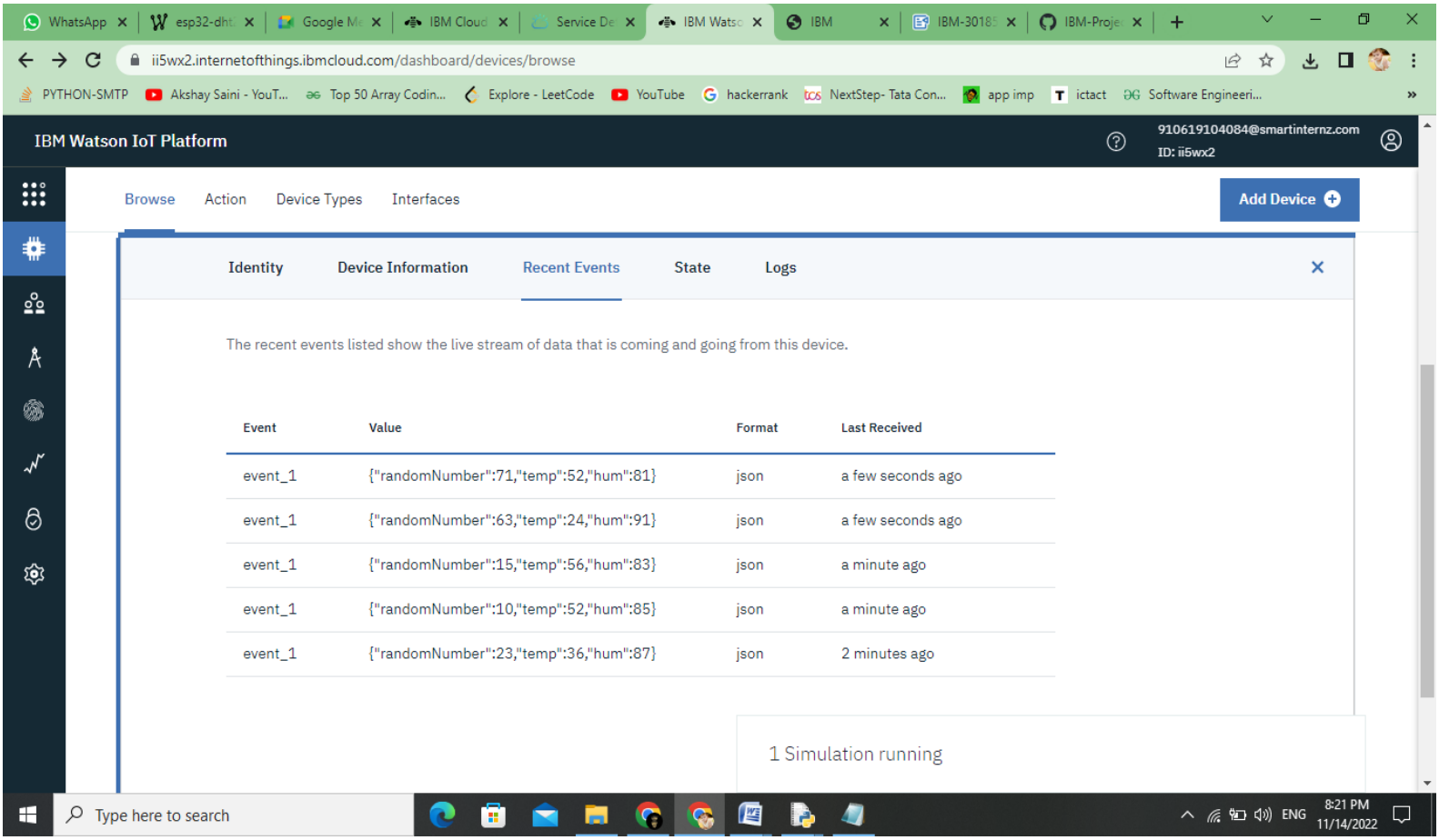
00:06.632 71%




```
Publish ok
temp:24.00
Humid:40.00
Moisture:8
Sending payload: {"temp":24.00,"Humid":40.00,"Moisture":8}
Publish ok
temp:24.00
```

Type here to search

ENG 8:20 PM 11/14/2022


IBM Watson IoT Platform
910619104084@smartinternz.com
ID: ii5wx2
Browse
Action
Device Types
Interfaces
Add Device
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
event_1
{"randomNumber":71,"temp":52,"hum":81}
json
a few seconds ago
event_1
{"randomNumber":63,"temp":24,"hum":91}
json
a few seconds ago
event_1
{"randomNumber":15,"temp":56,"hum":83}
json
a minute ago
event_1
{"randomNumber":10,"temp":52,"hum":85}
json
a minute ago
event_1
{"randomNumber":23,"temp":36,"hum":87}
json
2 minutes ago
1 Simulation running
Type here to search
ENG
8:21 PM
11/14/2022