Sprint Delivery – 4

Date	17 November 2022
Team ID	PNT2022TMID28572
Project Name	Hazardous Area Monitoring for Industrial Power plant powered by IoT
Maximum Marks	4 Marks

WOKWI CODE:

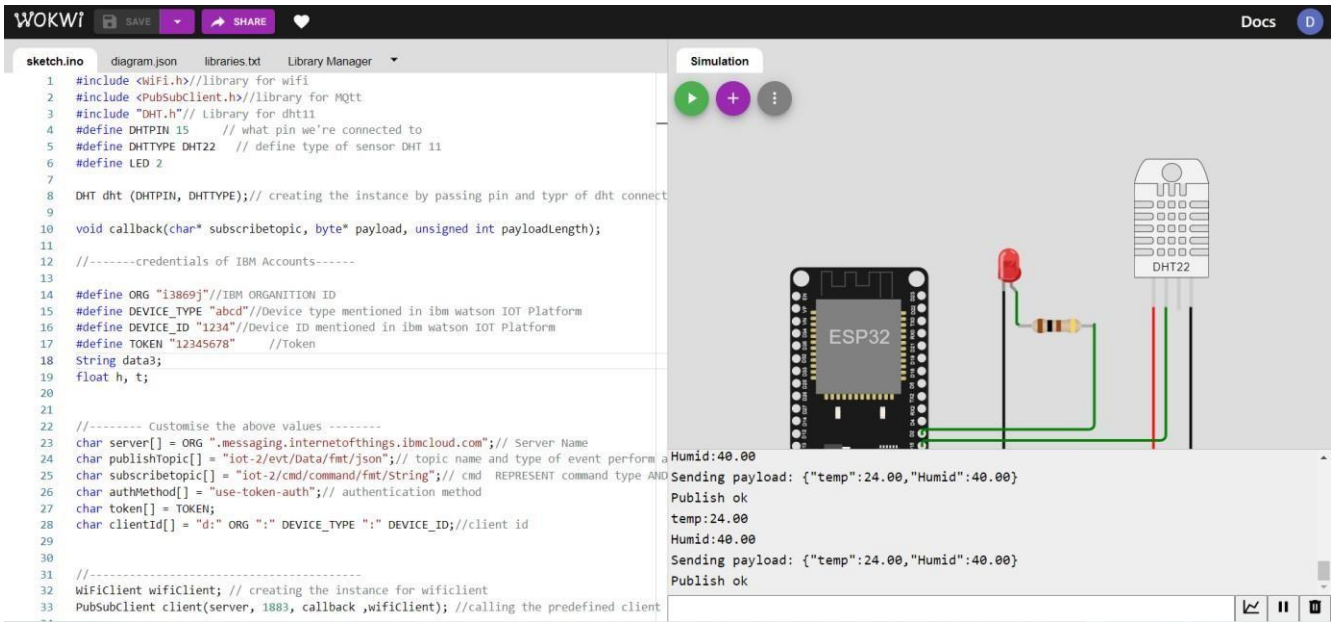
```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for
MQtt#include "DHT.h"// Library for
dht11
#define DHTPIN 15         // what pin we're connected
to #define DHTTYPE DHT22                  // define
type of sensor DHT 11#define LED 2

DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of
dhtconnected void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength);
//-------credentials of IBM Accounts------

#define ORG "iagqzu"//IBM ORGANITION ID
#define DEVICE_TYPE "Deepak"//Device type mentioned in ibm watson IOT
Platform#define DEVICE_ID "123"//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 formatin
which data to be send
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd REPRESENT command
typeAND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
chartoken[] = 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 predefinedclient
id by passing parameter like server id,portand wificredential
```

```
void setup()// configureing the ESP32
{
   Serial.begin(
   115200);
   dht.begin();
 pinMode(LED,OUTP
UT); delay(10);
                  Serial.p
rintln();


wifi
conn
ect()
;
mqtt
conn
ect()
;
} void loop()//
RecursiveFunction
{       h =
dht.readHumidity();t
=
dht.readTemperature
();
   Serial.print("tem
   p:");
   Serial.println(t);
   Serial.print("Hu
   mid:");
   Serial.println(h);
   PublishData(t, h);

delay(10
00);                  if (!client.loop()) {mqttconnect();
   }
}



/*.....retrieving to
Cloud. ..... */
 void PublishData(float temp, float humid)
{
```

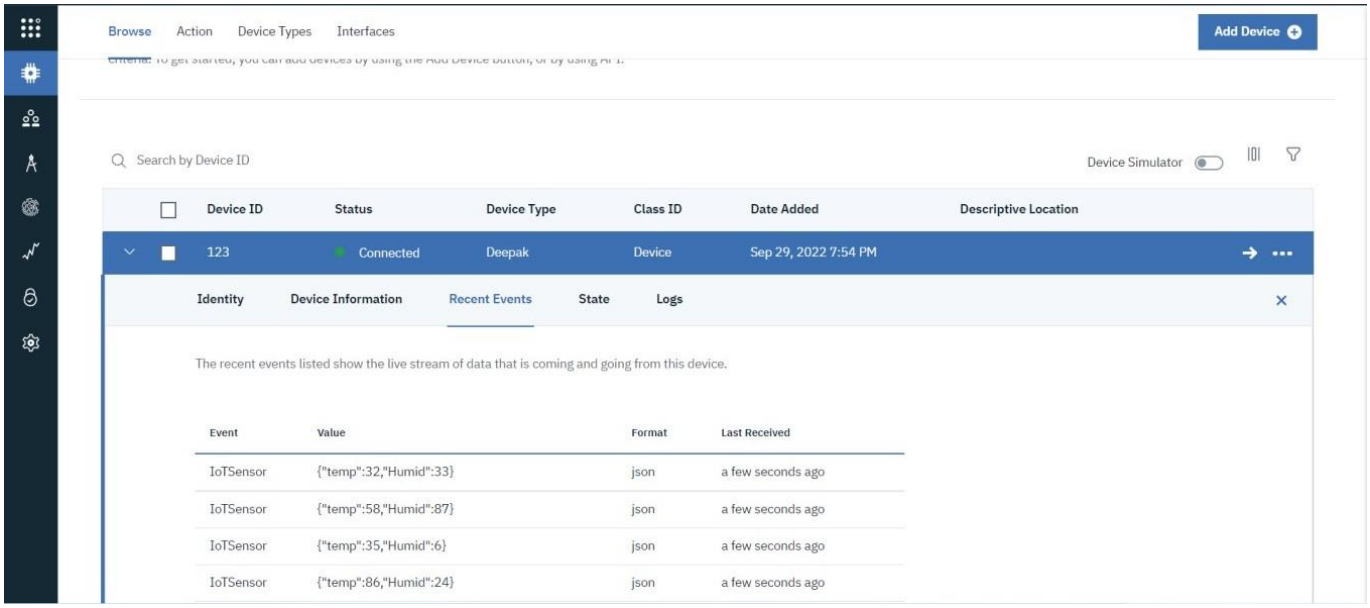
```
   mqttconnect();//function call for connecting to ibm
   /*            creating the String in in form JSon to update the data to ibm
cloud      */
   String payload =
"{\"temp\":";payload +=
temp;       payload += ","
"\"Humid\":";
               payload +=
humid;      payload +=
"}";


   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 itwill
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
forwificonnect
{
   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);
digitalWrite(LED,HIGH);   }    else
    {
Serial.println(
data3);
digitalWrite(L
ED,LOW);
} data3="";
}
```

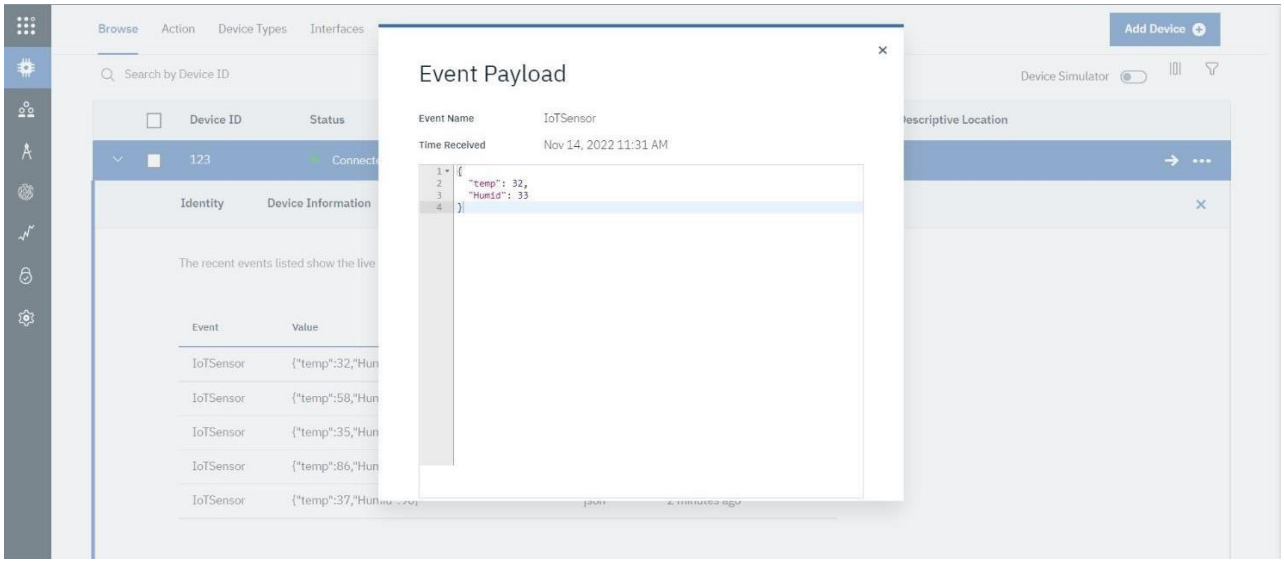
WOKWI OUTPUT:



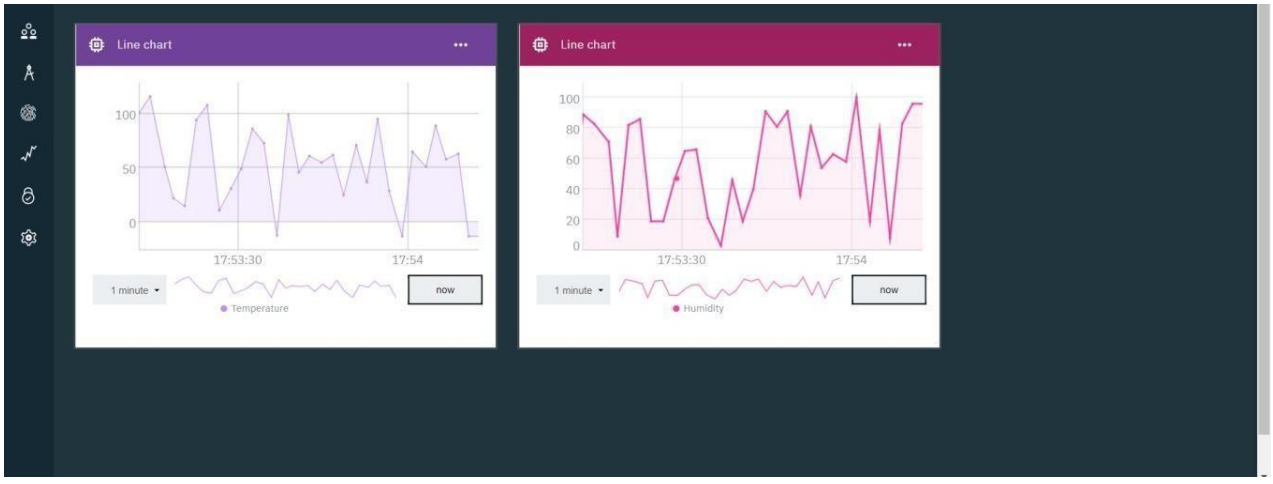
IBM WATSON PLATFORM DEVICE EVENT LOG:



DEVICE EVENT PAYLOAD:



DEVICE- BOARD:



IBM CLOUDANT DB LOG: