Project Development Phase
Sprint 3

Date	12 November 2022
Team ID	PNT2022TMID04334
Project Name	Gas leakage monitoring and alerting system for industries

Sprint 3:

The sensed values are published to IBM cloud continuously.

The values are sent through payload and are received in Json format.

Code:

```
#include <WiFi.h>
#include <PubSubClient.h>
#include "DHTesp.h"
#include<stdio.h>
#include <stdlib.h>
#define LED 2
const int DHT_PIN = 15;
DHTesp dhtSensor;
int gas;
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
#define ORG "oyi7sh"
#define DEVICE_TYPE "Gas_leakage"
#define DEVICE_ID "154555"
#define TOKEN "WoOgbWlz4q-F4KQKc-"
String data3;
```

```
IPAddress myDns(127, 0, 0, 53);
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);
  dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
  pinMode(LED, OUTPUT);
  delay(10);
  wificonnect();
  mqttconnect();
}
void loop()
{
    TempAndHumidity  data = dhtSensor.getTempAndHumidity();
    gas=random(10000);
    Serial.println("Temp: " + String(data.temperature, 2) + "°C");
    Serial.println("Humidity: " + String(data.humidity, 1) + "%");
    Serial.println("gas_val " + String(gas));
    PublishData(String(data.temperature,2),String(data.humidity,
1),String(gas),int(data.temperature),int(data.humidity),int(gas));
    delay(1000);
    if (!client.loop()) {
      mqttconnect();
    }
```

```
}
void PublishData(String temp,String hum,String gas1,int temp1,int hum1,int gas2)
{
  mqttconnect();
  if (gas2>2000)
  {
    digitalWrite(LED, HIGH);
    Serial.println("Fire alert");
  }
  else
  {
    digitalWrite(LED, LOW);
    Serial.println("Normal");
  }
  String payload = "{\"temperature\":";
  payload += temp;
  payload += "," "\"humidity\":\"";
  payload += hum;
  payload += "\"";
  payload += "," "\"gas_level\":\"";
  payload += gas1;
  payload += "\"}";

  Serial.print("Sending payload: ");
  // Serial.println(payload);
  if (client.publish(publishTopic, (char*) payload.c_str()))
  {
    Serial.println("Data sent successfully");
  }
  else
  {
    Serial.println("Data sent failure");
```

```
  }
  Serial.println("---");
}
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());
```

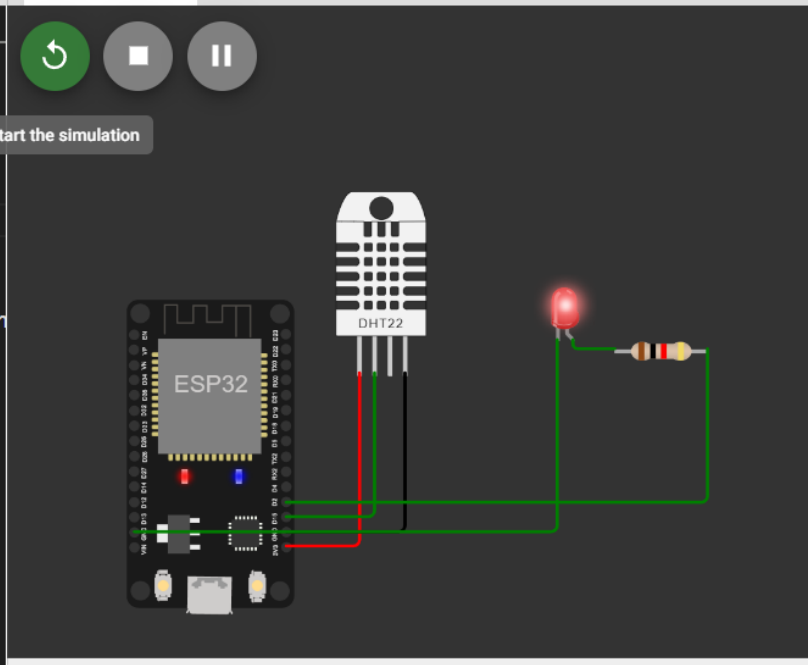
```
}
void initManagedDevice()
{
  if (client.subscribe(subscribetopic))
  {
    Serial.println((subscribetopic));
    Serial.println("subscribe to cmd OK");
  }
  else
  {
    Serial.println("subscribe to cmd FAILED");
  }
}
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{
  Serial.print("callback invoked for topic: ");
  Serial.println(subscribetopic);
  for (int i = 0; i < payloadLength; i++)
  {
    data3 += (char)payload[i];
  }
  data3="";
}
```

OUTPUT:

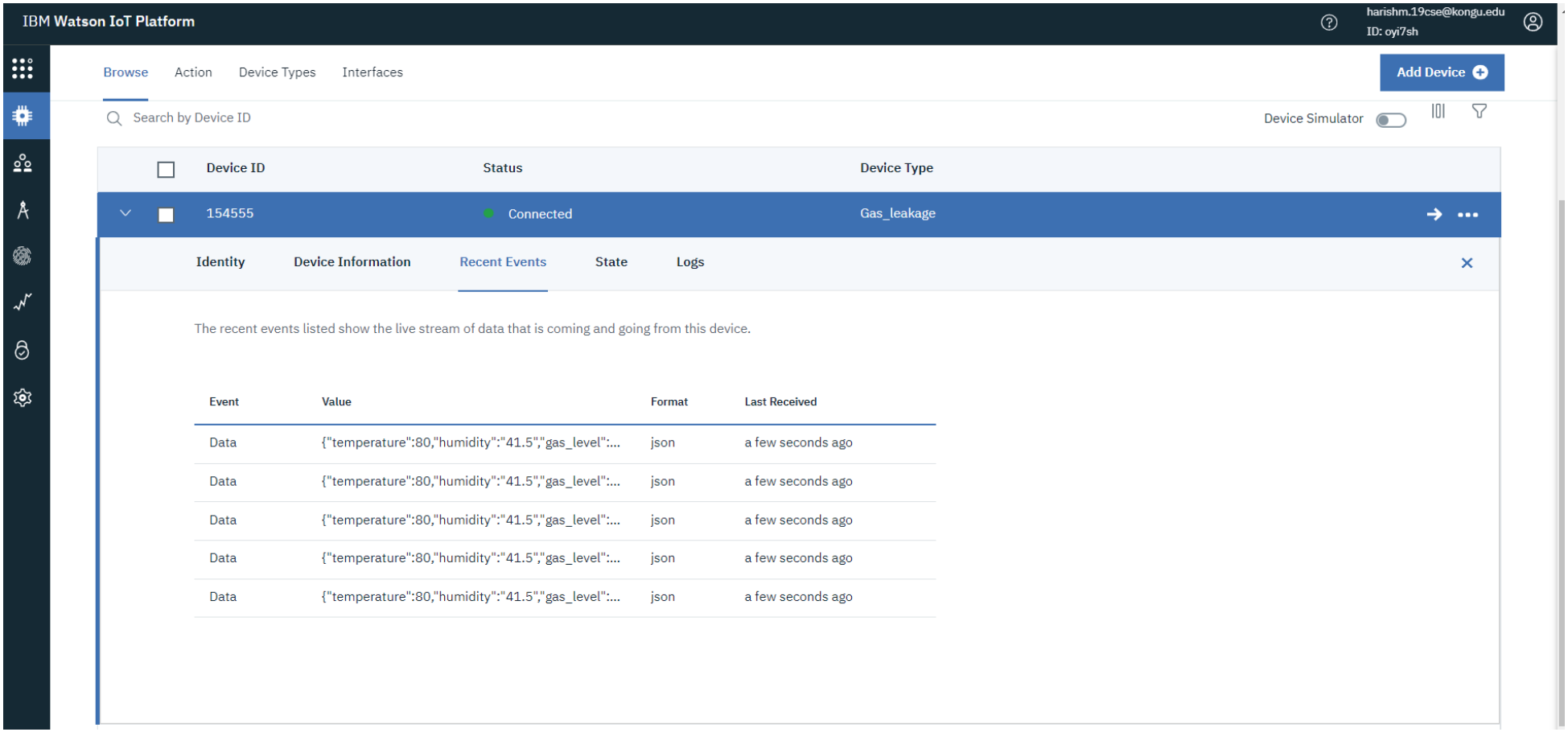
sketch.ino | diagram.json | libraries.txt | Library Manager

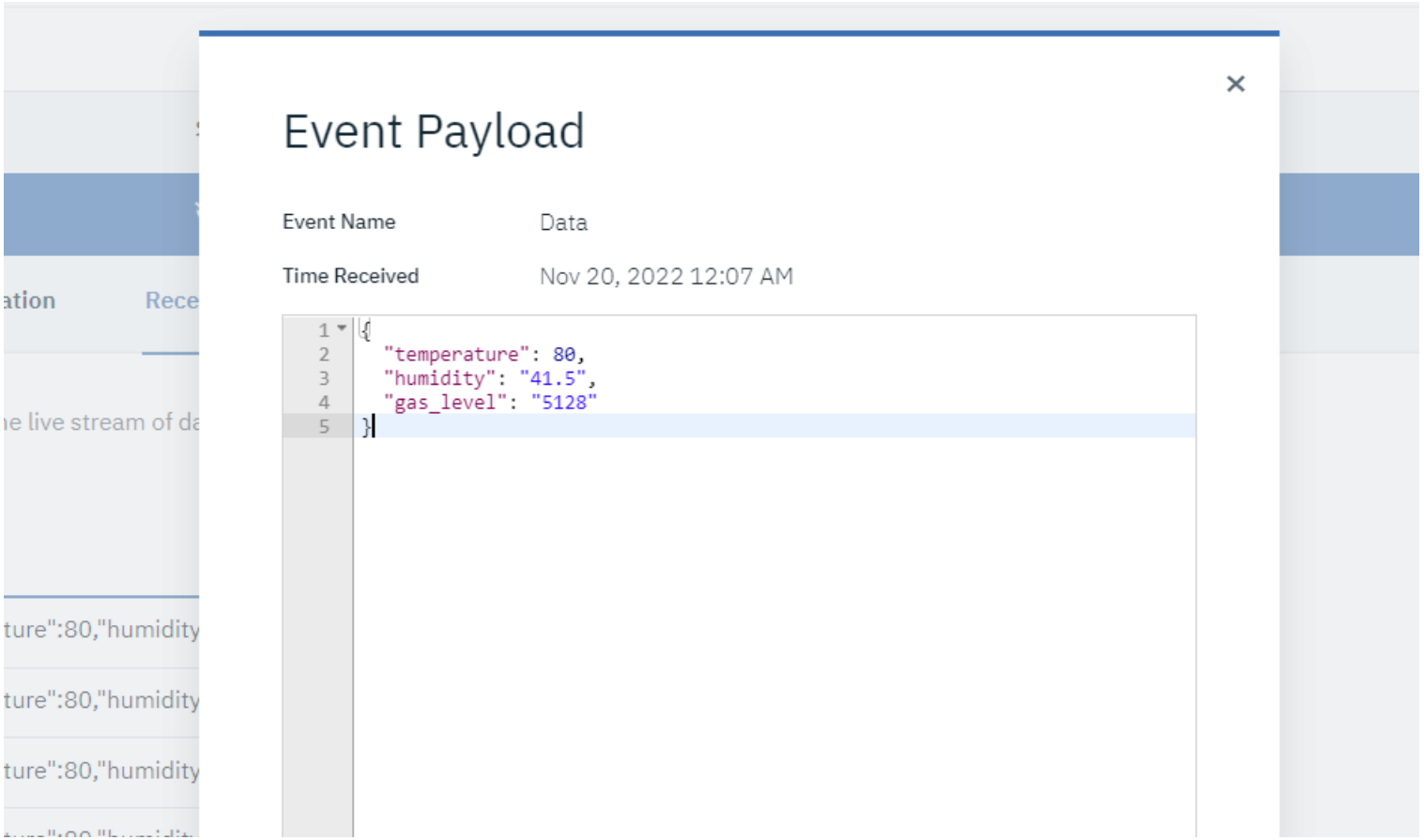
```
#include <WiFi.h>
#include <PubSubClient.h>
#include "DHTesp.h"
#include<stdio.h>
#include <stdlib.h>
#define LED 2
const int DHT_PIN = 15;
DHTesp dhtSensor;
int gas;
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength
#define ORG "oyi7sh"
#define DEVICE_TYPE "Gas_leakage"
#define DEVICE_ID "154555"
#define TOKEN "WoOgbWlz4q-F4KQKc-"
String data3;
IPAddress myDns(127, 0, 0, 53);
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);
  dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
```

Simulation




```
---
Temp: 80.00°C
Humidity: 41.5%
gas_val 3487
Fire alert
Sending payload: Data sent successfully
---
```


IBM Watson IoT Platform
harishm.19cse@kongu.edu
ID: oyi7sh
Browse
Action
Device Types
Interfaces
Add Device
Search by Device ID
Device Simulator
Device ID
Status
Device Type
154555
Connected
Gas_leakage
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
{"temperature":80,"humidity":"41.5","gas_level":...
json
a few seconds ago
Data
{"temperature":80,"humidity":"41.5","gas_level":...
json
a few seconds ago
Data
{"temperature":80,"humidity":"41.5","gas_level":...
json
a few seconds ago
Data
{"temperature":80,"humidity":"41.5","gas_level":...
json
a few seconds ago
Data
{"temperature":80,"humidity":"41.5","gas_level":...
json
a few seconds ago



Thus sprint 3 has been completed successfully