PROJECT DEVELOPMENT PHASE

Date	17 November 2022
Team ID	PNT2022TMID42644
Project Title	INDUSTRY – SPECIFIC INTELLIGENT FIRE MANAGEMENT SYSTEM

SPRINT 1 : (Simulation Creaction using Wokwi and Connecting IBM Cloud using Python Code) :

Simulation Creaction using Wokwi :

Code :

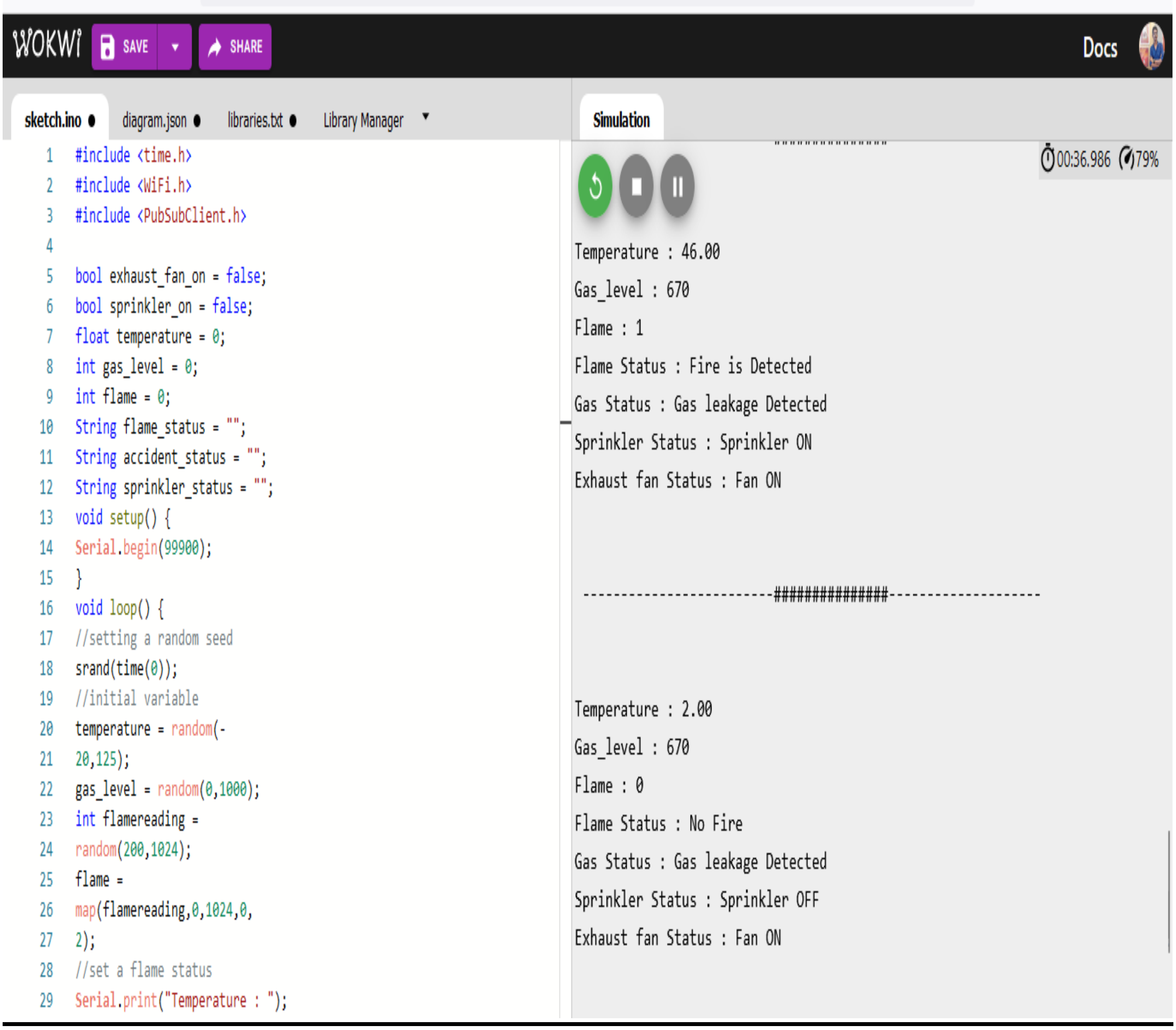
```
#include <time.h>
#include <WiFi.h>
#include <PubSubClient.h>

bool exhaust_fan_on = false;
bool sprinkler_on = false;
float temperature = 0;
int gas_level = 0;
int flame = 0;
String flame_status = "";
String accident_status = "";
String sprinkler_status = "";
void setup() {
```

```
Serial.begin(99900);
}
void loop() {
//setting a random seed
srand(time(0));
//initial variable
temperature = random(-
20,125);
gas_level = random(0,1000);
int flamereading =
random(200,1024);
flame =
map(flamereading,0,1024,0,
2);
//set a flame status
Serial.print("Temperature : ");
Serial.println(temperature);

Serial.print("Gas_level : ");
Serial.println(gas_level);

Serial.print("Flame : ");
Serial.println(flame);

switch (flame) {
```

```
case 0:
flame_status = "No Fire";
Serial.println("Flame Status : "+flame_status);
break;
case 1:
flame_status = "Fire is Detected";
Serial.println("Flame Status : "+flame_status);
break;
}
//Gas Detection
if(gas_level > 100){
Serial.println("Gas Status : Gas leakage Detected");
}
else{
exhaust_fan_on = false;
Serial.println("Gas Status : No Gas leakage Detected");
}
//send the sprinkler status
if(flame){
sprinkler_status =
"Sprinkler ON";
Serial.println("Sprinkler Status : "+sprinkler_status);
}
else{
sprinkler_status = "Sprinkler OFF";
```

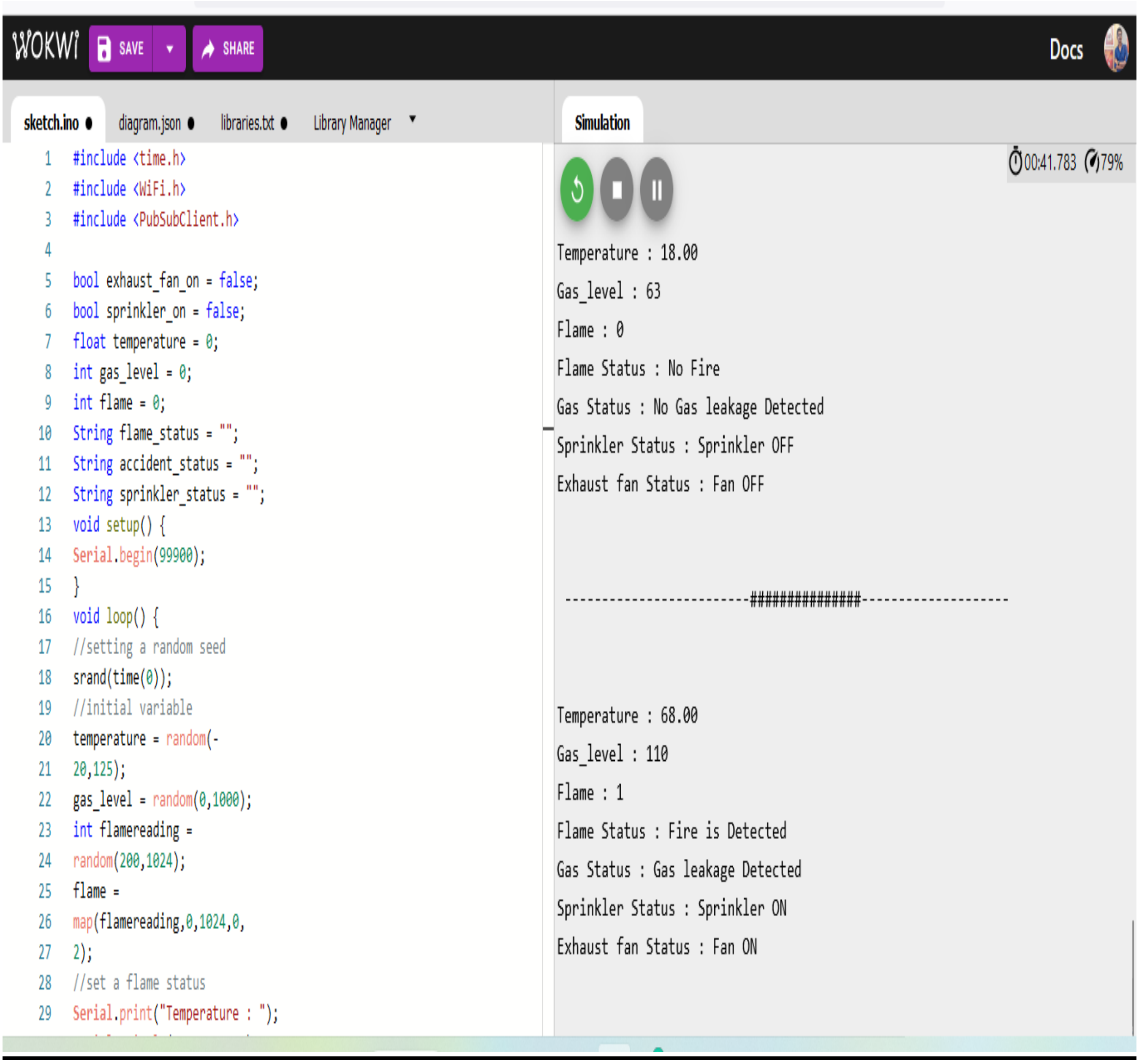
```
Serial.println("Sprinkler Status : "+sprinkler_status);
}
//toggle the fan according to gas
if(gas_level > 100){
exhaust_fan_on = true;
Serial.println("Exhaust fan Status : Fan ON");
}
else{
exhaust_fan_on = false;
Serial.println("Exhaust fan Status : Fan OFF");
}
Serial.println("");
Serial.println("");
Serial.println(" -------------------------###############--------------------");
Serial.println("");
Serial.println("");
delay(1000);
}
```

Simulation Output :



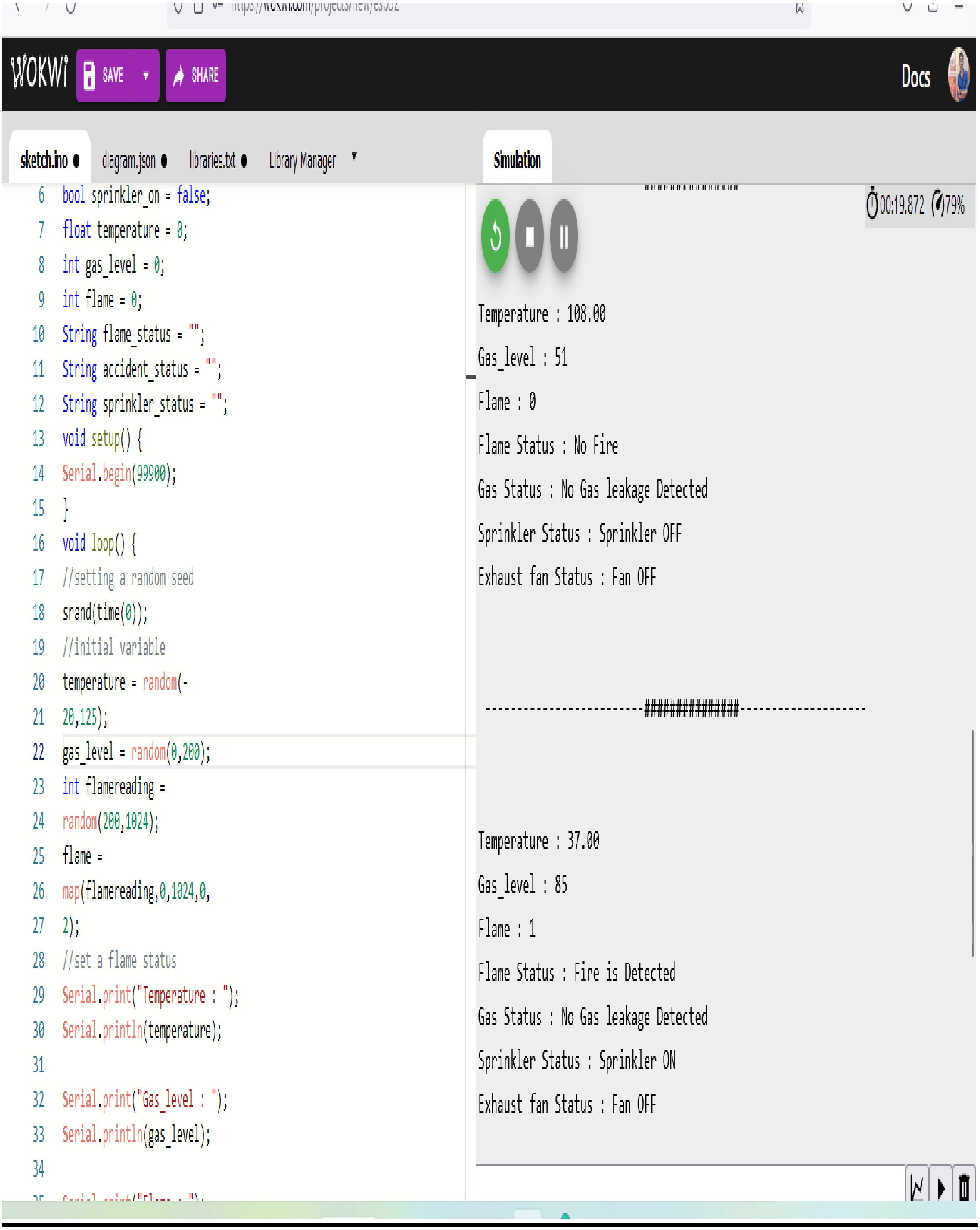
```
#include <time.h>
#include <WiFi.h>
#include <PubSubClient.h>

bool exhaust_fan_on = false;
bool sprinkler_on = false;
float temperature = 0;
int gas_level = 0;
int flame = 0;
String flame_status = "";
String accident_status = "";
String sprinkler_status = "";
void setup() {
Serial.begin(99900);
}
void loop() {
//setting a random seed
srand(time(0));
//initial variable
temperature = random(-
20,125);
gas_level = random(0,1000);
int flamereading =
random(200,1024);
flame =
map(flamereading,0,1024,0,
2);
//set a flame status
Serial.print("Temperature : ");
```

```
Temperature : 46.00
Gas_level : 670
Flame : 1
Flame Status : Fire is Detected
Gas Status : Gas leakage Detected
Sprinkler Status : Sprinkler ON
Exhaust fan Status : Fan ON


 -----------------------##############-------------------


Temperature : 2.00
Gas_level : 670
Flame : 0
Flame Status : No Fire
Gas Status : Gas leakage Detected
Sprinkler Status : Sprinkler OFF
Exhaust fan Status : Fan ON
```



```
#include <time.h>
#include <WiFi.h>
#include <PubSubClient.h>

bool exhaust_fan_on = false;
bool sprinkler_on = false;
float temperature = 0;
int gas_level = 0;
int flame = 0;
String flame_status = "";
String accident_status = "";
String sprinkler_status = "";
void setup() {
Serial.begin(99900);
}
void loop() {
//setting a random seed
srand(time(0));
//initial variable
temperature = random(-
20,125);
gas_level = random(0,1000);
int flamereading =
random(200,1024);
flame =
map(flamereading,0,1024,0,
2);
//set a flame status
Serial.print("Temperature : ");
```

```
Temperature : 18.00
Gas_level : 63
Flame : 0
Flame Status : No Fire
Gas Status : No Gas leakage Detected
Sprinkler Status : Sprinkler OFF
Exhaust fan Status : Fan OFF



 -----------------------##############-------------------



Temperature : 68.00
Gas_level : 110
Flame : 1
Flame Status : Fire is Detected
Gas Status : Gas leakage Detected
Sprinkler Status : Sprinkler ON
Exhaust fan Status : Fan ON
```



```
bool sprinkler_on = false;
float temperature = 0;
int gas_level = 0;
int flame = 0;
String flame_status = "";
String accident_status = "";
String sprinkler_status = "";
void setup() {
Serial.begin(99900);
}
void loop() {
//setting a random seed
srand(time(0));
//initial variable
temperature = random(-
20,125);
gas_level = random(0,200);
int flamereading =
random(200,1024);
flame =
map(flamereading,0,1024,0,
2);
//set a flame status
Serial.print("Temperature : ");
Serial.println(temperature);

Serial.print("Gas_level : ");
Serial.println(gas_level);

Serial.print("Flame : ");
```

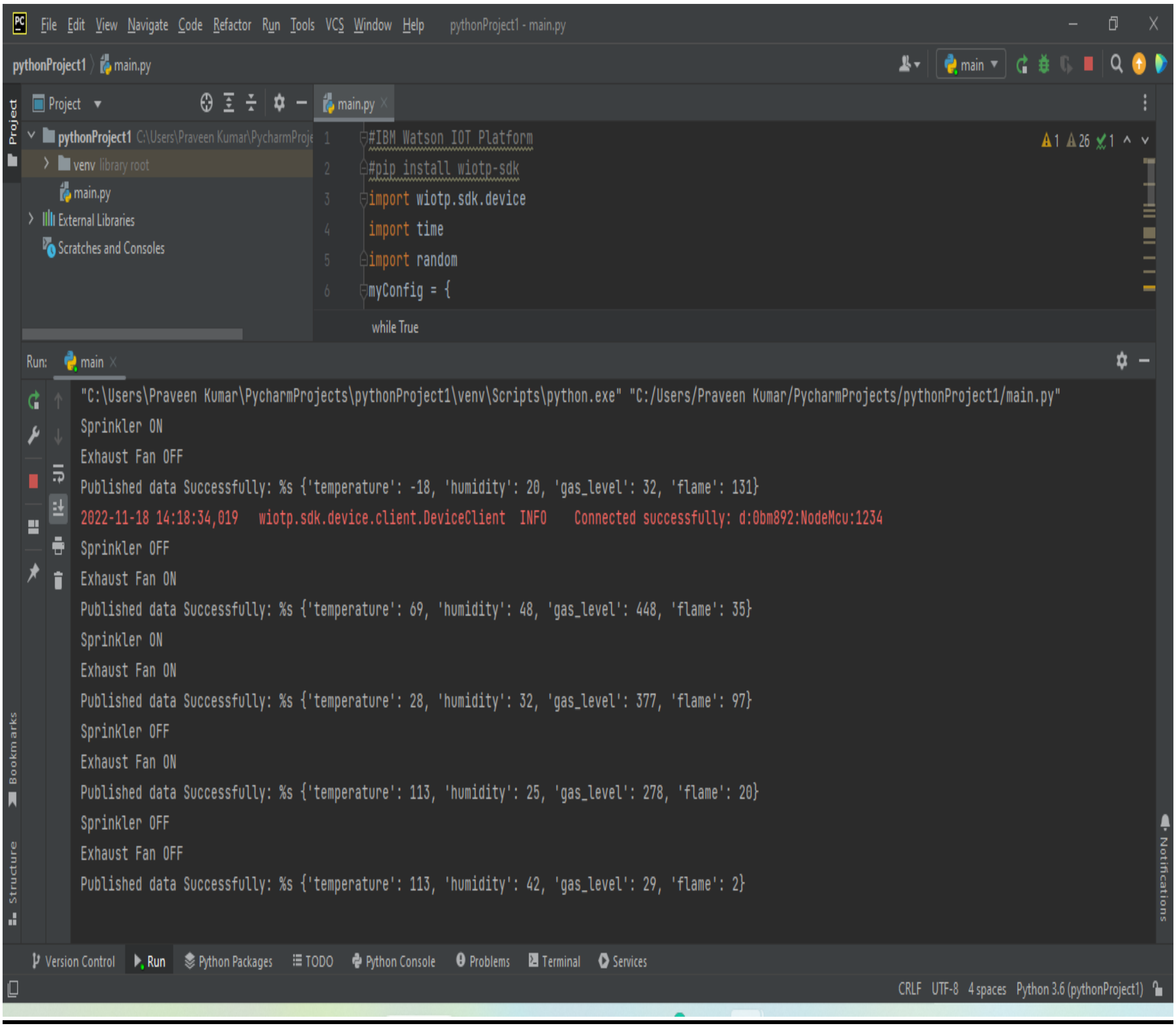
```
                      ##############
Temperature : 108.00
Gas_level : 51
Flame : 0
Flame Status : No Fire
Gas Status : No Gas leakage Detected
Sprinkler Status : Sprinkler OFF
Exhaust fan Status : Fan OFF


 ----------------------##############------------------


Temperature : 37.00
Gas_level : 85
Flame : 1
Flame Status : Fire is Detected
Gas Status : No Gas leakage Detected
Sprinkler Status : Sprinkler ON
Exhaust fan Status : Fan OFF
```

Connecting IBM Cloud using Python Code :

Code:

```
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "0bm892",
        "typeId": "NodeMcu",
        "deviceId":"1234"
    },
    "auth": {
        "token": "12345678"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" %
cmd.data['command'])
    m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig,
logHandlers=None)
client.connect()

while True:
    temperature=random.randint(-20,125)
    humidity=random.randint(0,100)
    gas_level =random.randint(0,500)
    flame=random.randint(0,200)

    if (flame>80):
        print("Sprinkler ON")
    else:
        print("Sprinkler OFF")

    if (gas_level>100):
        print("Exhaust Fan ON")
    else:
        print("Exhaust Fan OFF")

    myData={'temperature':temperature, 'humidity':humidity,
'gas_level': gas_level,'flame':flame}
    client.publishEvent(eventId="status", msgFormat="json",
data=myData, qos=0, onPublish=None)
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
    time.sleep(2)
client.disconnect()
```

OUTPUT :

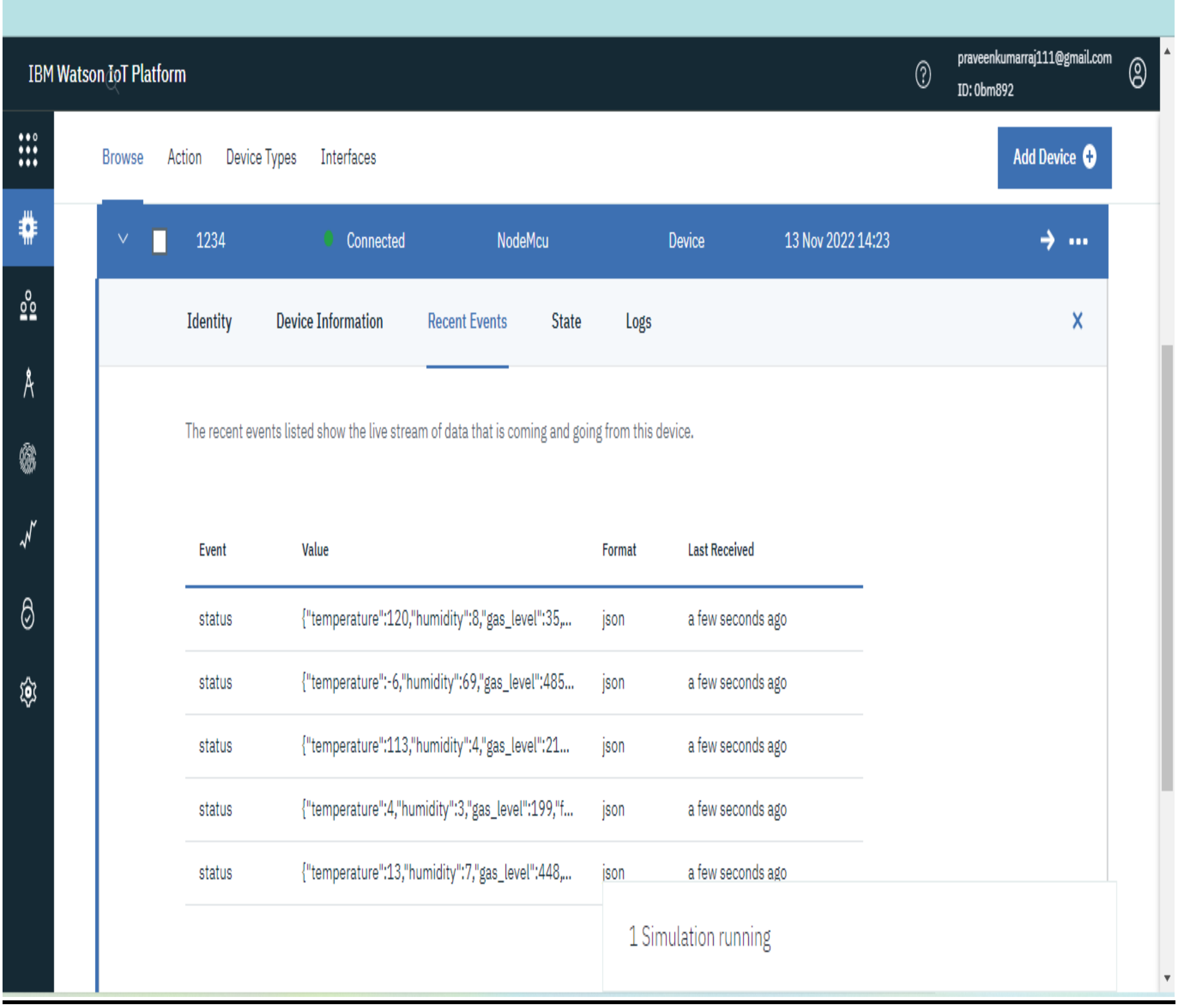


```python
#IBM Watson IOT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig = {
```

while True

```
"C:\Users\Praveen Kumar\PycharmProjects\pythonProject1\venv\Scripts\python.exe" "C:/Users/Praveen Kumar/PycharmProjects/pythonProject1/main.py"
Sprinkler ON
Exhaust Fan OFF
Published data Successfully: %s {'temperature': -18, 'humidity': 20, 'gas_level': 32, 'flame': 131}
2022-11-18 14:18:34,019   wiotp.sdk.device.client.DeviceClient  INFO    Connected successfully: d:0bm892:NodeMcu:1234
Sprinkler OFF
Exhaust Fan ON
Published data Successfully: %s {'temperature': 69, 'humidity': 48, 'gas_level': 448, 'flame': 35}
Sprinkler ON
Exhaust Fan ON
Published data Successfully: %s {'temperature': 28, 'humidity': 32, 'gas_level': 377, 'flame': 97}
Sprinkler OFF
Exhaust Fan ON
Published data Successfully: %s {'temperature': 113, 'humidity': 25, 'gas_level': 278, 'flame': 20}
Sprinkler OFF
Exhaust Fan OFF
Published data Successfully: %s {'temperature': 113, 'humidity': 42, 'gas_level': 29, 'flame': 2}
```

OUTPUT IN IBM CLOUD :



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

Event	Value	Format	Last Received
status	{"temperature":120,"humidity":8,"gas_level":35,...	json	a few seconds ago
status	{"temperature":-6,"humidity":69,"gas_level":485...	json	a few seconds ago
status	{"temperature":113,"humidity":4,"gas_level":21...	json	a few seconds ago
status	{"temperature":4,"humidity":3,"gas_level":199,"f...	json	a few seconds ago
status	{"temperature":13,"humidity":7,"gas_level":448,...	json	a few seconds ago

1 Simulation running