SPRINT-4

PROJECT	INDUSTRY-SPECIFIC INTELLIGENT FIRE MANAGEMENT SYSTEM
TEAM ID	PNT2022TMID49436

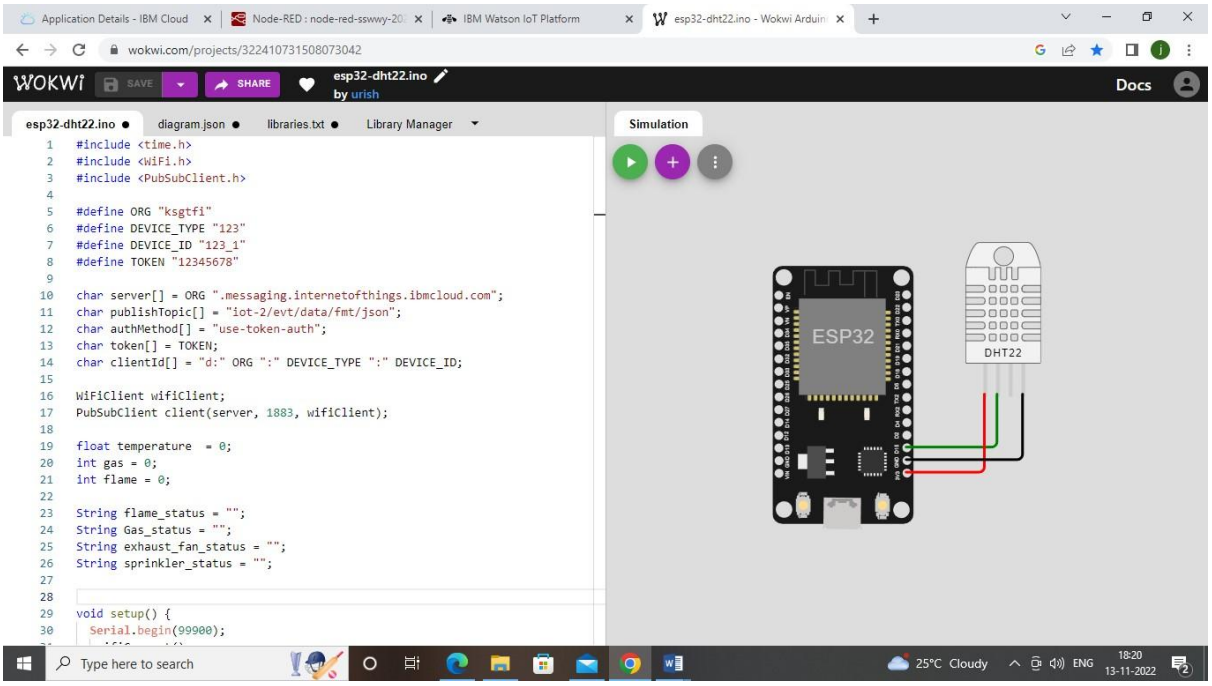
PYTHON CODE:

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

#define ORG "ksgtfi"
#define DEVICE_TYPE "123"
#define DEVICE_ID "123_1"
#define TOKEN "12345678"
 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-
2/evt/data/fmt/json";
char authMethod[] = "use-token-auth";
char token[] = TOKEN; char clientId[] = "d:" ORG ":"
DEVICE_TYPE ":" DEVICE_ID;

WiFiClient wifiClient;
PubSubClient client(server, 1883, wifiClient);
 float temperature  = 0;
int gas = 0; int flame
= 0;

String flame_status = "";
String Gas_status = "";
String exhaust_fan_status = "";
String sprinkler_status = "";


void setup()
{   Serial.begin(99900);
wifiConnect();    mqttConnect();
}
void loop() {
   srand(time(0));
//initial
variables and
random generated
data

    temperature = random(-20,125);    gas =
 random(0,1000);    int flamereading =
```

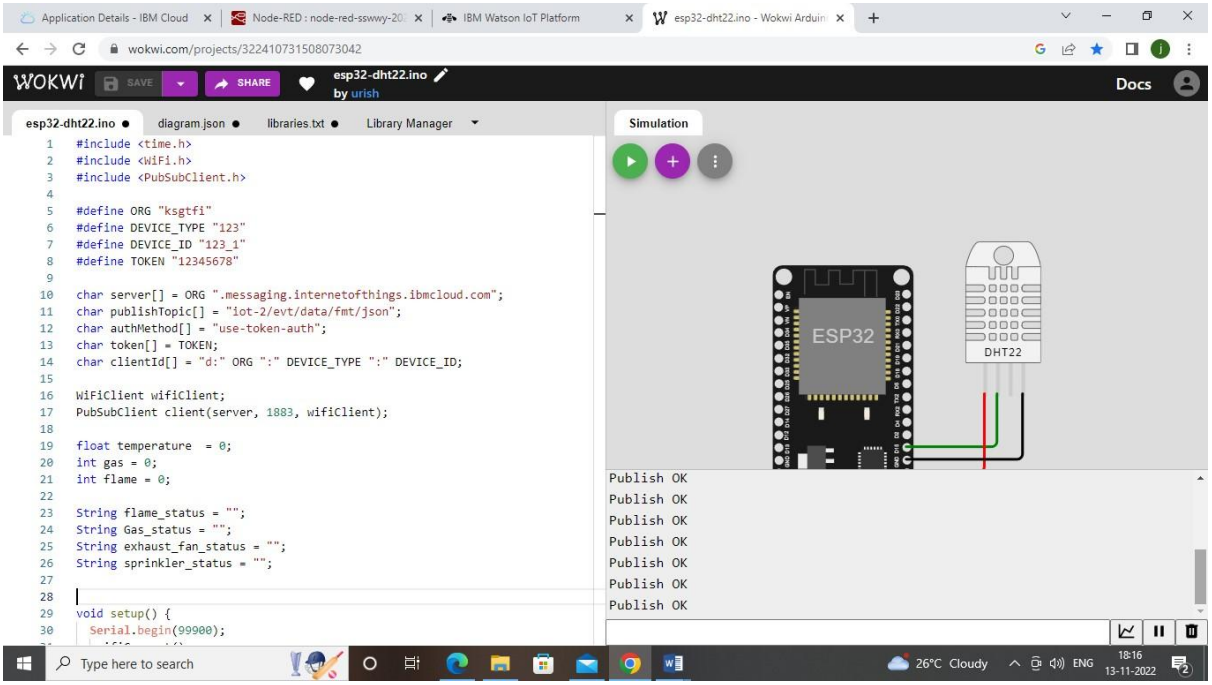
```
 random(200,1024);    flame =
 map(flamereading,200,1024,0,2);

    //set a flame status
     switch (flame) {     case 0:
flame_status = "No Fire";         break;
case 1:          flame_status = "Fire is
Detected";         break;
    }

    //send the sprinkler status

if(flame==1){
        sprinkler_status = "Working";
    }     else{         sprinkler_status
= "Not Working";

    }

    //toggle the fan according to gas reading

    if(gas > 100){
        Gas_status = "Gas Leakage is Detected";
exhaust_fan_status = "Working";
     }
else{
        Gas_status = "No Gas Leakage is Detected";
exhaust_fan_status = "Not Working";
    }

    //json format for IBM Watson

    String payload = "{";
payload+="\"gas\":";
payload+=gas;     payload+=",";
payload+="\"temperature\":";
payload+=(int)temperature;
payload+=",";
payload+="\"flame\":";
payload+=flamereading;
payload+=",";
payload+="\"fire_status\":\""+fl
ame_status+"\",";
payload+="\"sprinkler_status\":\
""+sprinkler_status+"\",";
payload+="\"Gas_status\":\""+Gas
_status+"\",";
    payload+="\"exhaust_fan_status\":\""+exhaust_fan_status+"\"}";
       if(client.publish(publishTopic, (char*)
payload.c_str()))
```

```
    {
        Serial.println("Publish OK");
    }
else{
        Serial.println("Publish failed");
    }
    delay(1000);
        if
(!client.loop())

{       mqttConnect()
;
    }
}

void wifiConnect()
{
  Serial.print("Connecting to ");
  Serial.print("Wifi");
  WiFi.begin("Wokwi-GUEST", "", 6);
while (WiFi.status() != WL_CONNECTED)
  {
    delay(500);
    Serial.print(".");
  }
  Serial.print("WiFi connected, IP address: ");
  Serial.println(WiFi.localIP());

}

void mqttConnect()
{   if
(!client.connected())
  {
    Serial.print("Reconnecting MQTT client to ");
Serial.println(server);     while
(!client.connect(clientId, authMethod, token)) {
      Serial.print(".");
 delay(500);
    }

    Serial.println();
  }
}
```

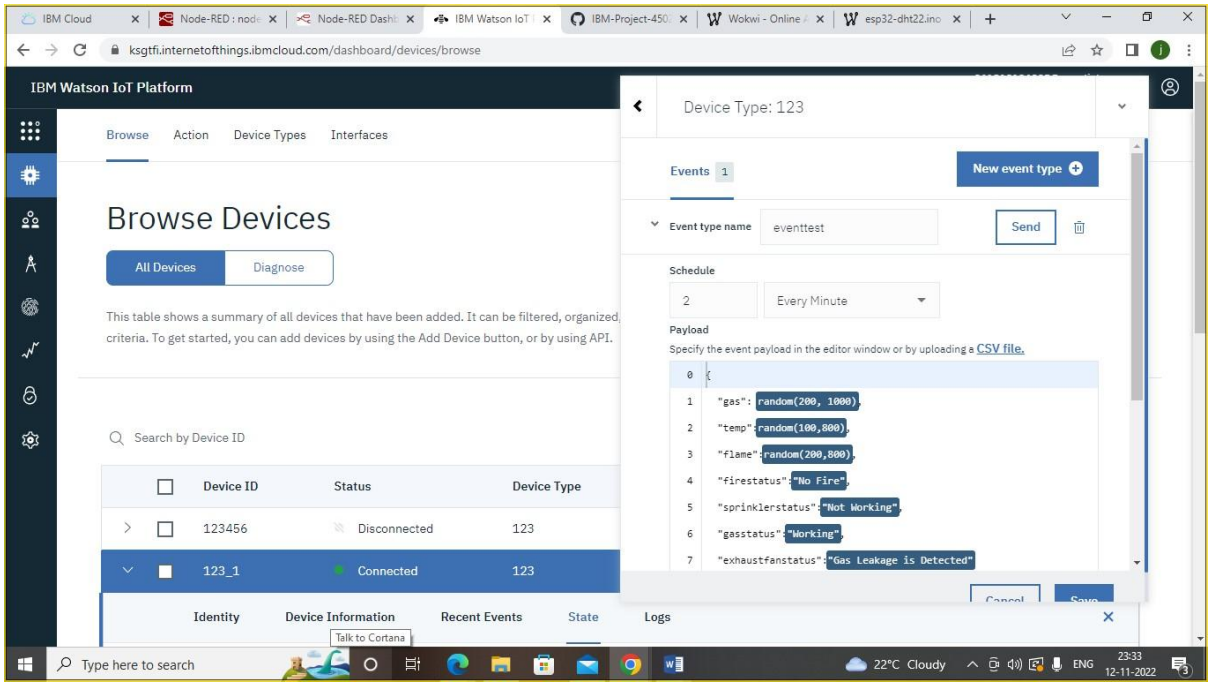
WOKWI CONNECTION:



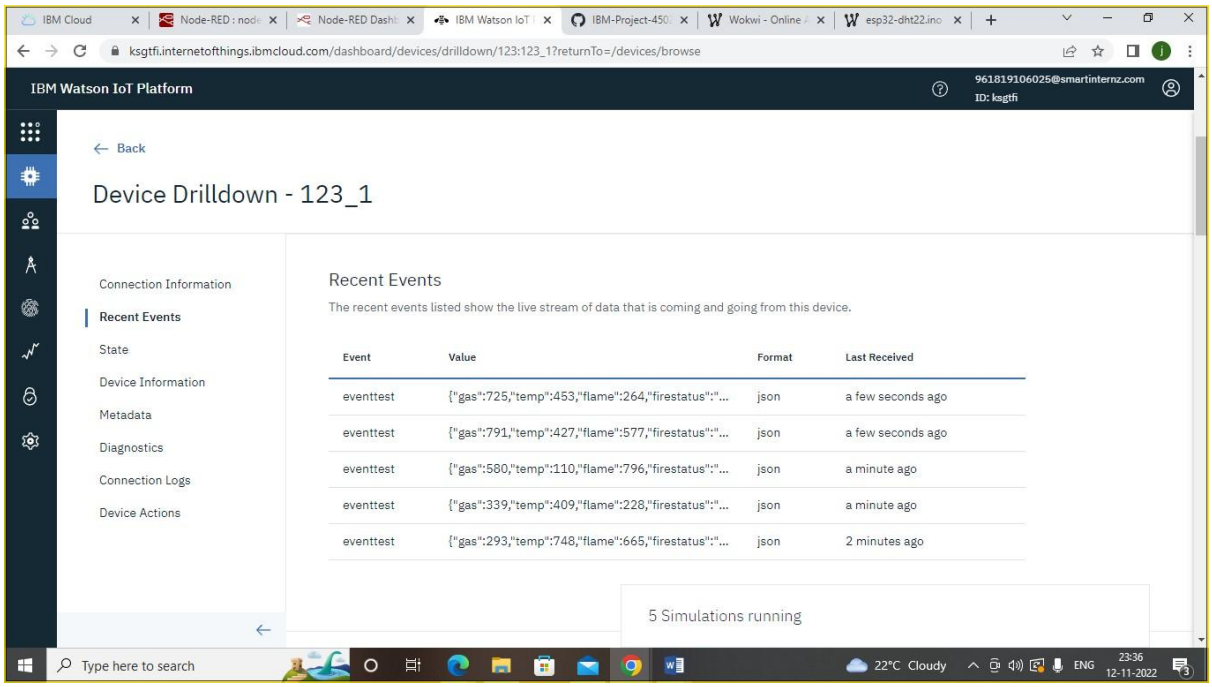
WOKWI OUTPUT:

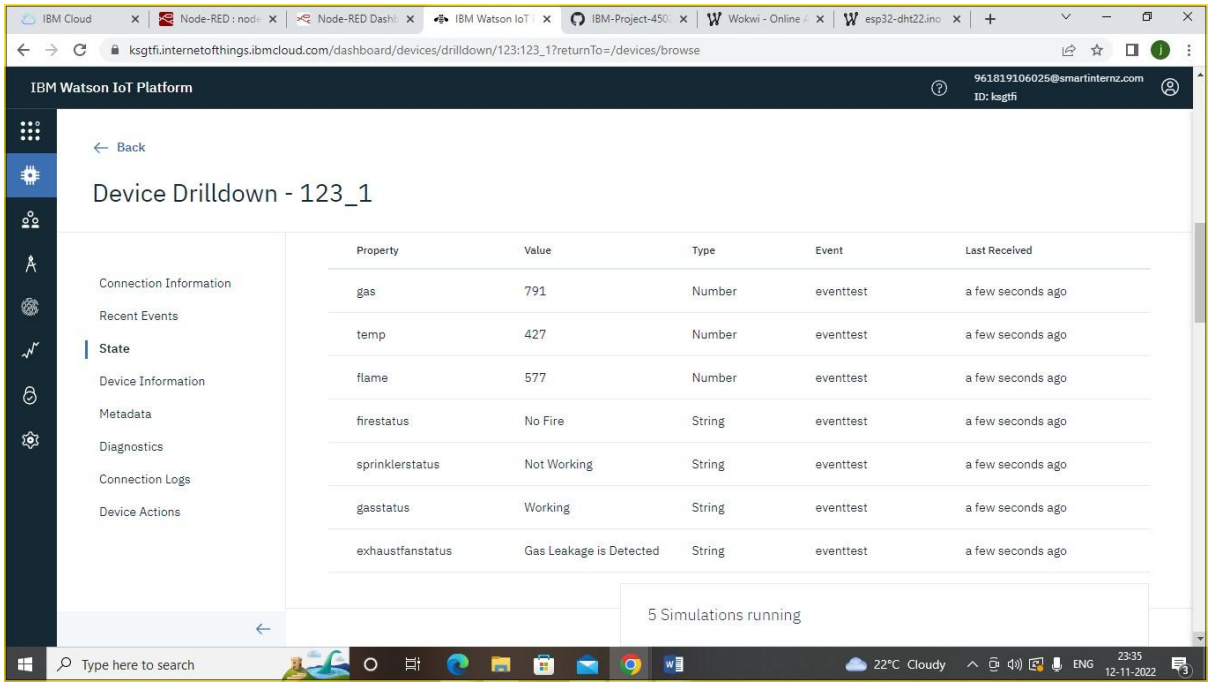


WATSON IOT PLATFORM:

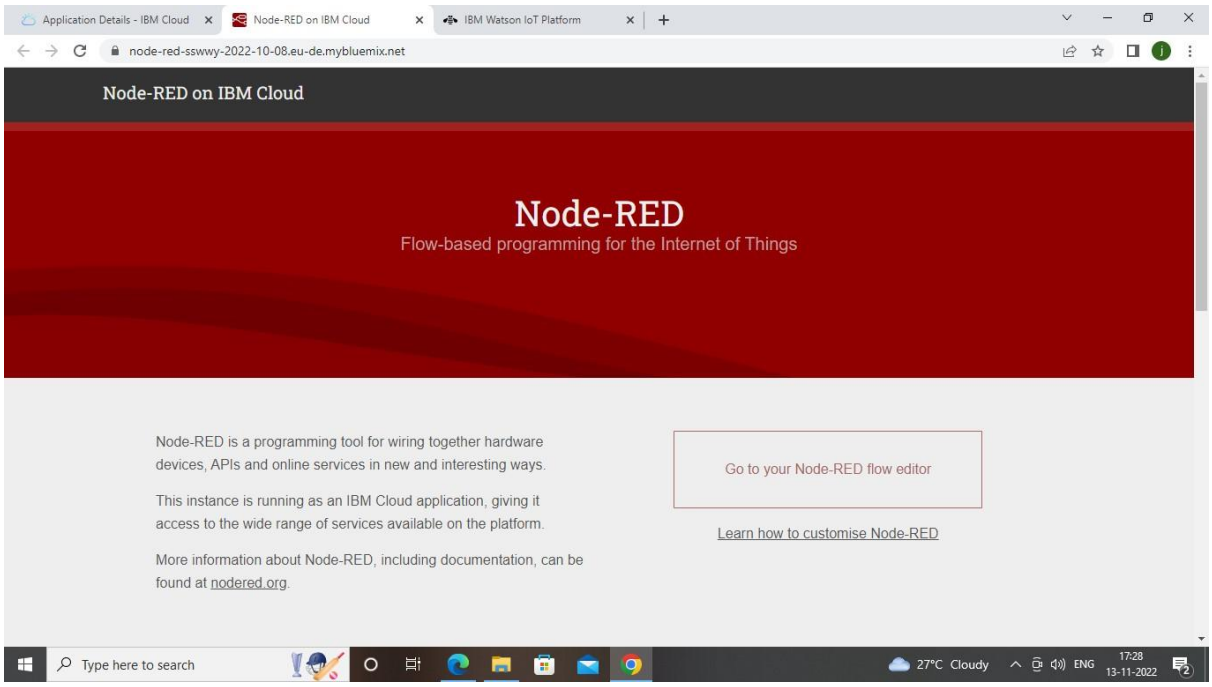

IBM Watson IoT Platform
Browse
Action
Device Types
Interfaces
Browse Devices
All Devices
Diagnose
Search by Device ID
Device ID
Status
Device Type
123456
Disconnected
123
123_1
Connected
123
Identity
Device Information
Recent Events
State
Logs
Device Type: 123
Events 1
New event type
Event type name
eventtest
Send
Schedule
2
Every Minute
Payload
Specify the event payload in the editor window or by uploading a CSV file.
{
"gas": random(200, 1000),
"temp":random(100,800),
"flame":random(200,800),
"firestatus":"No Fire",
"sprinklerstatus":"Not Working",
"gasstatus":"Working",
"exhaustfanstatus":"Gas Leakage is Detected"
Cancel
Save
Type here to search
22°C Cloudy
ENG
23:33
12-11-2022

OUTPUT:

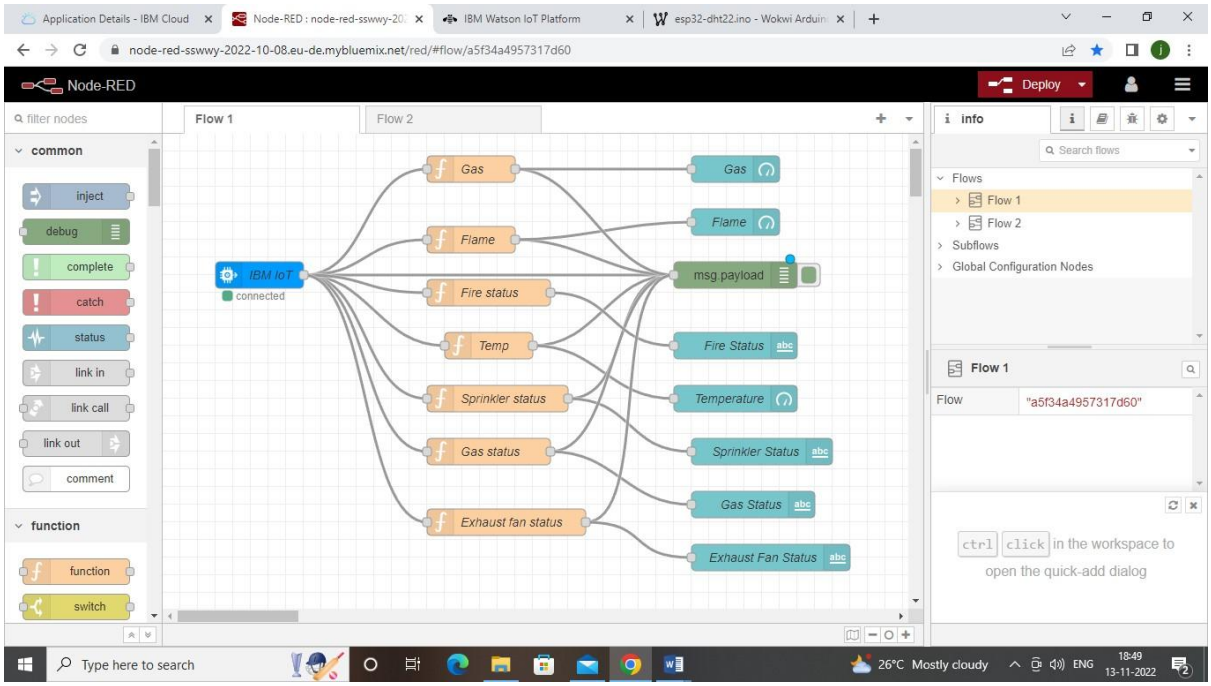

IBM Watson IoT Platform
961819106025@smartinternz.com
ID: ksgtfi
Back
Device Drilldown - 123_1
Connection Information
Recent Events
State
Device Information
Metadata
Diagnostics
Connection Logs
Device Actions
Recent Events
The recent events listed show the live stream of data that is coming and going from this device.
Event
Value
Format
Last Received
eventtest
{"gas":725,"temp":453,"flame":264,"firestatus":"...
json
a few seconds ago
eventtest
{"gas":791,"temp":427,"flame":577,"firestatus":"...
json
a few seconds ago
eventtest
{"gas":580,"temp":110,"flame":796,"firestatus":"...
json
a minute ago
eventtest
{"gas":339,"temp":409,"flame":228,"firestatus":"...
json
a minute ago
eventtest
{"gas":293,"temp":748,"flame":665,"firestatus":"...
json
2 minutes ago
5 Simulations running
Type here to search
22°C Cloudy
ENG
23:36
12-11-2022



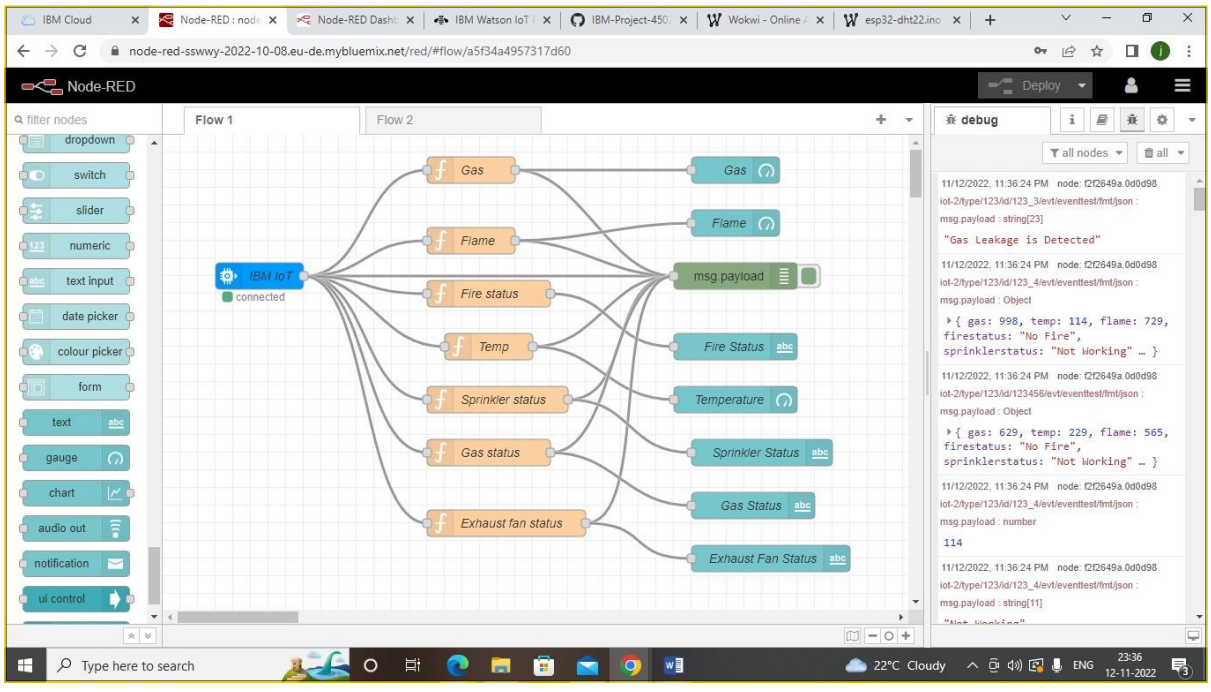
NODE-RED :

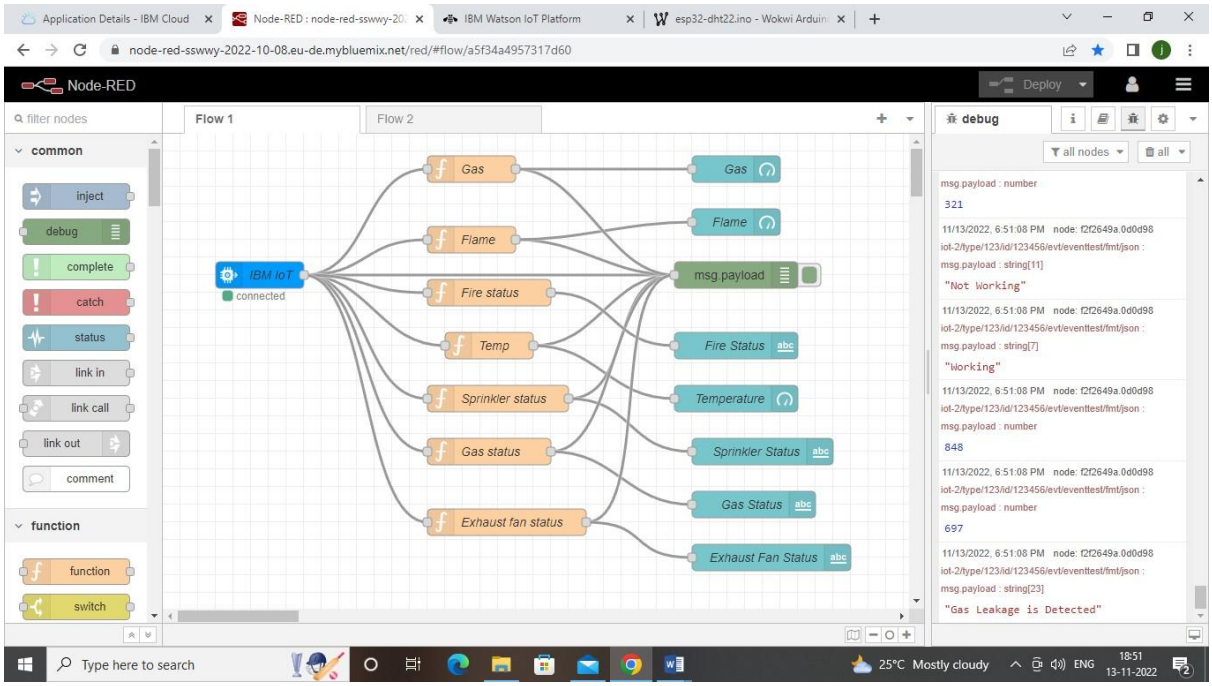


WEB APPLICATION USING NODE-RED :

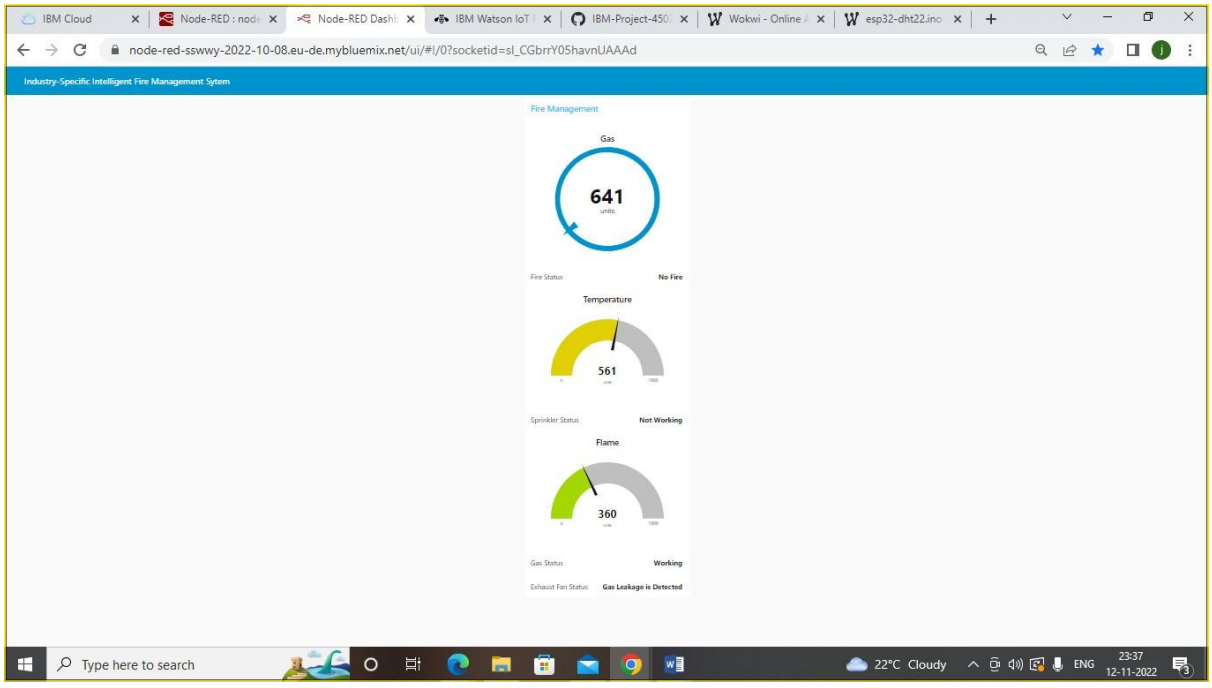

Application Details - IBM Cloud
Node-RED : node-red-sswwy-20
IBM Watson IoT Platform
esp32-dht22.ino - Wokwi Arduin
node-red-sswwy-2022-10-08.eu-de.mybluemix.net/red/#flow/a5f34a4957317d60
Node-RED
Deploy
filter nodes
Flow 1
Flow 2
common
inject
debug
complete
catch
status
link in
link call
link out
comment
function
function
switch
Gas
Flame
Fire status
Temp
Sprinkler status
Gas status
Exhaust fan status
IBM IoT
connected
Gas
Flame
msg payload
Fire Status
Temperature
Sprinkler Status
Gas Status
Exhaust Fan Status
info
Search flows
Flows
Flow 1
Flow 2
Subflows
Global Configuration Nodes
Flow 1
Flow
"a5f34a4957317d60"
ctrl
click
in the workspace to open the quick-add dialog
Type here to search
26°C Mostly cloudy
ENG
18:49
13-11-2022

OUTPUT:


IBM Cloud
Node-RED : node
Node-RED Dashb
IBM Watson IoT
IBM-Project-450
Wokwi - Online
esp32-dht22.ino
node-red-sswwy-2022-10-08.eu-de.mybluemix.net/red/#flow/a5f34a4957317d60
Node-RED
Deploy
filter nodes
Flow 1
Flow 2
dropdown
switch
slider
numeric
text input
date picker
colour picker
form
text
gauge
chart
audio out
notification
ui control
Gas
Flame
Fire status
Temp
Sprinkler status
Gas status
Exhaust fan status
IBM IoT
connected
Gas
Flame
msg payload
Fire Status
Temperature
Sprinkler Status
Gas Status
Exhaust Fan Status
debug
all nodes
all
11/12/2022, 11:36:24 PM node: f2f2649a.0d0d98
iot-2/type/123/id/123_3/evt/eventtest/fmt/json : msg.payload : string[23]
"Gas Leakage is Detected"
11/12/2022, 11:36:24 PM node: f2f2649a.0d0d98
iot-2/type/123/id/123_4/evt/eventtest/fmt/json : msg.payload : Object
{ gas: 998, temp: 114, flame: 729, firestatus: "No Fire", sprinklerstatus: "Not Working" … }
11/12/2022, 11:36:24 PM node: f2f2649a.0d0d98
iot-2/type/123/id/123456/evt/eventtest/fmt/json : msg.payload : Object
{ gas: 629, temp: 229, flame: 565, firestatus: "No Fire", sprinklerstatus: "Not Working" … }
11/12/2022, 11:36:24 PM node: f2f2649a.0d0d98
iot-2/type/123/id/123_4/evt/eventtest/fmt/json : msg.payload : number
114
11/12/2022, 11:36:24 PM node: f2f2649a.0d0d98
iot-2/type/123/id/123_4/evt/eventtest/fmt/json : msg.payload : string[11]
Type here to search
22°C Cloudy
ENG
23:36
12-11-2022



NODE-RED DASHBOARD STATUS:



Successfully get the status for Flame, Fire, Gas , Temperture in Node-red Dashboard Using Node-Red Platform.