SPRINT-1

Team ID	PNT2022TMID41191
Project Name	Industry-specific intelligent fire management system

CONFIGURING ESP32 USING WOKWI PROJECTS:

Program:-

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

```
#include "DHTesp.h"
#include <cstdlib>
#include <time.h>

const int DHT_PIN = 15;

bool is_exhaust_fan_on = false;
bool is_sprinkler_on = false;

float temperature  = 0;

int gas_ppm = 0;
int flame = 0;
int flow = 0;

String flame_status = "";
String accident_status = "";
String sprinkler_status = "";

DHTesp dhtSensor;


void setup() {
  Serial.begin(99900);

  /**** sensor pin setups ****/
```

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

```
  /**** sensor pin setups ****/
  dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
  //if real gas sensor is used make sure the senor is heated up for acurate readings
  /*
    - Here random values for readings and stdout were used to show the
      working  of the devices as physical or simulated devices are not
      available.
  */
}

void loop() {

  TempAndHumidity  data = dhtSensor.getTempAndHumidity();

  //setting a random seed
  srand(time(0));

  //initial variable activities like declaring , assigning
  temperature  = data.temperature;
  gas_ppm = rand()%1000;
  int flamereading = rand()%1024;
  flame = map(flamereading,0,1024,0,1024);
  int flamerange = map(flamereading,0,1024,0,3);
  int flow = ((rand()%100)>50?1:0);

```

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

```
51  //set a flame status based on how close it is.....
52  switch (flamerange) {
53  case 2:    // A fire closer than 1.5 feet away.
54    flame_status = "Close Fire";
55    break;
56  case 1:    // A fire between 1-3 feet away.
57    flame_status = "Distant Fire";
58    break;
59  case 0:    // No fire detected.
60    flame_status = "No Fire";
61    break;
62  }
63
64  //toggle the fan according to gas in ppm in the room
65  if(gas_ppm > 100){
66    is_exhaust_fan_on = true;
67  }
68  else{
69    is_exhaust_fan_on = false;
70  }
71
72  //find the accident status 'cause fake alert may be caused by some mischief activities
73  if(temperature < 40 && flamerange ==2){
74    accident_status = "need auditing";
75    is_sprinkler_on = false;
```

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

```
75    is_sprinkler_on = false;
76  }
77  else if(temperature < 40 && flamerange ==0){
78    accident_status = "not found";
79    is_sprinkler_on = false;
80  }
81  else if(temperature > 50 && flamerange == 1){
82    is_sprinkler_on = true;
83    accident_status = "moderate";
84  }
85  else if(temperature > 55 && flamerange == 2){
86    is_sprinkler_on = true;
87    accident_status = "severe";
88  }else{
89    is_sprinkler_on = false;
90    accident_status = "none";
91  }
92
93
94  //send the sprinkler status
95  if(is_sprinkler_on){
96    if(flow){
97      sprinkler_status = "working";
98    }
99    else{
100     sprinkler status = "not working";
```

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

```
100     sprinkler_status = "not working";
101   }
102 }
103 else if(is_sprinkler_on == false){
104   sprinkler_status = "it should not!";
105 }
106 else{
107   sprinkler_status = "Error!!";
108 }
109
110 //Obivously the output.It is like json format 'cause it will help us for future sprints
111 String out = "{\n\t\"senor_values\":{";
112 out+="\n\t\t\"gas_ppm\":"+String(gas_ppm)+",";
113 out+="\n\t\t\"temperature\":"+String(temperature,2)+",";
114 out+="\n\t\t\"flame\":"+String(flame)+",";
115 out+="\n\t\t\"flow\":"+String(flow)+",\n\t}";
116 out+="\n\t\"output\":{";
117 out+="\n\t\t\"is_exhaust_fan_on\":"+String((is_exhaust_fan_on)?"true":"false")+",";
118 out+="\n\t\t\"is_sprinkler_on\":"+String((is_sprinkler_on)?"true":"false")+",";
119 out+="\n\t}";
120 out+="\n\t\"messages\":{";
121 out+="\n\t\t\"fire_status\":"+flame_status+",";
122 out+="\n\t\t\"flow_status\":"+sprinkler_status+",";
123 out+="\n\t\t\"accident_status\":"+accident_status+",";
124 out+="\n\t}";
125 out+="\n}";
```

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

```
  out+="\n\t\t\"gas_ppm\":"+String(gas_ppm)+",";
  out+="\n\t\t\"temperature\":"+String(temperature,2)+",";
  out+="\n\t\t\"flame\":"+String(flame)+",";
  out+="\n\t\t\"flow\":"+String(flow)+",\n\t}";
  out+="\n\t\"output\":{";
  out+="\n\t\t\"is_exhaust_fan_on\":"+String((is_exhaust_fan_on)?"true":"false")+",";
  out+="\n\t\t\"is_sprinkler_on\":"+String((is_sprinkler_on)?"true":"false")+",";
  out+="\n\t}";
  out+="\n\t\"messages\":{";
  out+="\n\t\t\"fire_status\":"+flame_status+",";
  out+="\n\t\t\"flow_status\":"+sprinkler_status+",";
  out+="\n\t\t\"accident_status\":"+accident_status+",";
  out+="\n\t}";
  out+="\n}";
  Serial.println(out);

  delay(2000);
}

```

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

```
{
  "version": 1,
  "author": "Uri Shaked",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": 0, "left": 0, "attrs": {} },
    { "type": "wokwi-dht22", "id": "dht1", "top": -15.53, "left": 145.5, "attrs": {} }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [] ],
    [ "dht1:SDA", "esp:D15", "green", [ "v0" ] ],
    [ "dht1:VCC", "esp:3V3", "red", [ "v0" ] ],
    [ "dht1:GND", "esp:GND.1", "black", [ "v0" ] ]
  ]
}
```

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

```
# Wokwi Library List
# See https://docs.wokwi.com/guides/libraries

DHT sensor library for ESPx
```

OUTPUT:

```
{
        "senor_values":{
                "gas_ppm":0,
                "temperature":24.00,
                "flame":45,
                "flow":0,
        }
        "output":{
                "is_exhaust_fan_on":false,
                "is_sprinkler_on":false,
        }
        "messages":{
                "fire_status":No Fire,
                "flow_status":it should not!,
                "accident_status":not found,
        }
}
```

SIMULATION: