

# Project Development

## Phase

### Sprint - I

Date	19 NOVEMBER 2022
Team ID	PNT2022TMID18446
Project Name	Industry-Specific Intelligent Fire Management System

## OUTPUT:

WOKWI

SAVE

SHARE

esp32-dht22.ino by urish

Docs

esp32-dht22.ino

diagram.json

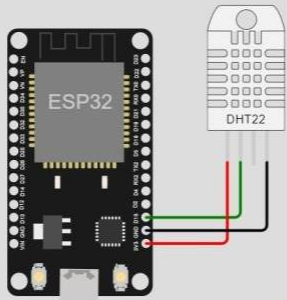
libraries.txt

Library Manager

```
1  /**
2   * ESP32 + DHT22 Example for Wokwi
3   *
4   * https://wokwi.com/arduino/projects/322410731508073042
5   */
6
7  #include "DHTesp.h"
8
9  const int DHT_PIN = 15;
10
11  DHTesp dhtSensor;
12
13  void setup() {
14    Serial.begin(115200);
15    dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
16  }
17
18  void loop() {
19    TempAndHumidity data = dhtSensor.getTempAndHumidity();
20    Serial.println("Temp: " + String(data.temperature, 2) + "°C");
21    Serial.println("Humidity: " + String(data.humidity, 1) + "%");
22    Serial.println("---");
23    delay(1000);
24  }
```

Simulation

+



## **CODE:**

```
#include <time.h>

bool exhaust_fan_on = false;
bool sprinkler_on = false;

float temperature = 0;
int gas = 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    =
  random(0,1000);  int
  flamereading =
random(200,1024);
  flame =
map(flamereading,0,1024,0,
2);

  //set a flame status

  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 > 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 =
"working";
        Serial.println("Sprinkler
Status : "+sprinkler_status);
    }
    else{ sprinkler_status =
"not
working";
        Serial.println("Sprinkler
Status : "+sprinkler_status);
    }
    //toggle the fan according
to gas
```

```

    if(gas > 100){ exhaust_fan_on
      = true;
      Serial.println("Exhaust
fan Status : Working");
    }
    else{ exhaust_fan_on =
      false;
      Serial.println("Exhaust
fan Status : Not Working");
    }

    Serial.println("");
    Serial.println("");
    Serial.println(" -----
/*****/-----
_____");
    Serial.println("");
    Serial.println("");

    delay(2000);

}

```

### **TEST CASES:**

S.NO	INPUT	OUTPUT	RESULT
1	Gas:62 Temperature:45.30 Flame:366	Exhaust Fan: Not Working Sprinkler: Not Working Status Logged: Done	PASSED
2	Gas:598 Temperature:51.40 Flame:412	Exhaust Fan: Working Sprinkler: Not Working Status Logged: Done	PASSED

--

	3	Gas:334 Temperature:49.30 Flame:912	Exhaust Fan: Working Sprinkler: Working Status Logged: Done	PASSED
	4	Gas:18 Temperature:67.90 Flame:745	Exhaust Fan: Not Working Sprinkler: Working Status Logged: Done	PASSED
	5	Gas: 354 Temperature:69.30 Flame:446	Exhaust Fan: Working Sprinkler: Not Working Status Logged: Done	PASSED