

SPRINT 1

| | |
|--------------|--|
| Date | 05 November 2022 |
| Team ID | PNT2022TMID04749 |
| Project Name | Industry-Specific Intelligent Fire Management System |

Wokwi project link:

<https://wokwi.com/projects/348649268096008787>

Output:

The screenshot displays the Wokwi web IDE interface. On the left, the code for `esp32-dht22.ino` is shown, which includes libraries for WiFi, MQTT, and the DHT22 sensor. It defines a MQTT client and a DHT22 sensor, and sets up a loop to read temperature and humidity data. On the right, the simulation window shows the hardware connection between the ESP32 and the DHT22 sensor. Below the hardware diagram, the serial output of the program is displayed, showing the connection status and the sensor readings.

```
1 #include <WiFi.h>
2 #include <PubSubClient.h>
3 #include <time.h>
4 #include "DHTesp.h"
5 #define temp_pin 15
6 void callback(char* topic, byte* payload, unsigned int payloadLength);
7 #define ORG "bx33ga"
8 #define DEVICE_TYPE "Sound"
9 #define DEVICE_ID "2002"
10 #define TOKEN "12345678"
11 String data;
12 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
13 char publishTopic[] = "iot-2/evt/Data/fmt/json";
14 char subscribeTopic[] = "iot-2/cmd/test/fmt/String";
15 char authMethod[] = "use-token-auth";
16 char token[] = TOKEN;
17 char clientId[] = "d:" + ORG + ":" + DEVICE_TYPE + ":" + DEVICE_ID;
18 WiFiClient wifiClient;
19 PubSubClient client(server, 1883, callback, wifiClient);
20
21 const int DHT_PIN = 15;
22
23 DHTesp dhtSensor;
24
25
26 bool exhaust_fan_on = false;
27 bool sprinkler_on = false;
28
29 float temperature = 0;
30 int gas = 0;
31 int flame = 0;
32
33 String flame_status = "";
34
35
36
37 void setup() {
38   Serial.begin(999000);
39
40   wifiConnect();
41   mqttConnect();
42   dhtSensor.setup(DHT_PIN, DHTesp::DHT22);
43 }
44
45
46 void loop() {
47   client.loop();
48   if (client.publish(publishTopic, data)) {
49     Serial.println("Published: " + data);
50   }
51   if (client.subscribe(subscribeTopic)) {
52     Serial.println("Subscribed: " + subscribeTopic);
53   }
54   float temp = dhtSensor.getTemperature();
55   float hum = dhtSensor.getHumidity();
56   String tempStr = String(temp, 2);
57   String humStr = String(hum, 2);
58   data = tempStr + "," + humStr;
59   Serial.println("Temperature: " + tempStr + " Humidity: " + humStr);
60 }
```

Connecting to.....
WIFI CONNECTED
IP address:
10.10.0.2
Reconnecting to bx33ga.messaging.internetofthings.ibmcloud.com
iot-2/cmd/test/fmt/String
subscribe to cmd ok

Temperature: 24.00°C
Humidity: 40.0%

Sending payload: ("temp normal")
publish ok
Flame Status : No Fire
Gas Status : There is no sign of a gas leak
Sprinkler Status : not functioning
Exhaust fan Status : Not functioning

Temperature: 24.00°C
Humidity: 40.0%

Sending payload: ("temp normal")
publish ok
Flame Status : No Fire
Gas Status : There is no sign of a gas leak
Sprinkler Status : not functioning
Exhaust fan Status : Not functioning