

Project Development Phase
Sprint-3

Date	23 October 2022
Team ID	PNT2022TMID40415
Project Name	Industry Specific-Intelligent Fire Management System
Maximum Marks	2

```
#define BLYNK_TEMPLATE_ID "TMPL-uhc59_T"
#define BLYNK_DEVICE_NAME "Fire alert"
#define BLYNK_AUTH_TOKEN "jkfkhu5fzDC9_PBdtssloT9OmXq3THwb"
```

```
#define BLYNK_FIRMWARE_VERSION    "0.1.0"
```

```
#define BLYNK_PRINT Serial
```

```
//#define BLYNK_DEBUG
```

```
#define APP_DEBUG
```

```
#include <ESP8266WiFi.h>
```

```
#include <BlynkSimpleEsp8266.h>
```

```
#include "DHT.h"
```

```
#define DHTPIN 5
```

```
#define DHTTYPE DHT22
```

```
DHT dht(DHTPIN, DHTTYPE);
```

```
char auth[]=BLYNK_AUTH_TOKEN;
```

```
char ssid[]="OPPO A52";
```

```
char pass[]="6380604277";
```

```
int Gas=A0;
```

```
int Flame=4;
```

```
int buzz=2;

int redLight=3;

int greenLight=4;

float sensorvalue;

int flamevalue;

void setup() {

    pinMode(Gas, INPUT);

    pinMode(Flame, INPUT);

    pinMode(buzz,OUTPUT);

    pinMode(redLight,OUTPUT);

    pinMode(greenLight,OUTPUT);

    Serial.begin(115200);

    Blynk.begin(auth,ssid,pass);

    dht.begin();

}

void loop() {

    sensorvalue = analogRead(Gas);

    flamevalue= digitalRead(Flame);

    Blynk.run();

    Blynk.virtualWrite(V0,sensorvalue);

    Blynk.virtualWrite(V1,!flamevalue);

    Serial.print("Gas value:");

    Serial.println(sensorvalue);

    Serial.print("flame state:");

    Serial.println(!flamevalue);


    float h = dht.readHumidity();

    float t = dht.readTemperature();


    if (isnan(h) || isnan(t)) {
```

```
Serial.println("Failed to read from DHT sensor!");  
return;  
}
```

```
Serial.print("Humidity: ");  
Serial.print(h);  
Serial.print(" %\t");  
Serial.print("Temperature: ");  
Serial.print(t);
```

```
if(flamevalue==0){  
    tone(buzz,1000,200);  
    digitalWrite(redLight,HIGH);  
    digitalWrite(greenLight,LOW);  
}  
else{  
    noTone(buzz);  
    digitalWrite(redLight,LOW);  
    digitalWrite(greenLight,HIGH);  
}
```

```
if(sensorvalue>500){  
    tone(buzz,1000,200);  
    digitalWrite(redLight,HIGH);  
    digitalWrite(greenLight,LOW);  
}  
else{  
    noTone(buzz);  
    digitalWrite(redLight,LOW);  
    digitalWrite(greenLight,HIGH);  
}
```

