

# Sprint-2

**Team ID :** PNT2022TMID31718

**Project Title :** Industry-specific intelligent fire management system

## CODE:

```
int LED_PIN = 2; // the current reading from the input pin and
int Motor_PIN= 12; // Pin for ventilation fan const int mq2 = 4;
int value = 0; // Flame
int flame_sensor_pin = 10 ; // initializing pin 10 as the sensor digital
output pin
int flame_pin = HIGH ; // current state of sensor
#define PIN_LM35 39
#define ADC_VREF_mV 3300.0 #define
ADC_RESOLUTION 4096.0
void setup()
{
  Serial.begin(115200); pinMode(LED_PIN,
  OUTPUT); pinMode(mq2, INPUT);
  pinMode ( flame_sensor_pin , INPUT ); // declaring sensor pin as input
  pin for Arduino
  pinMode(BUZZER_PIN, OUTPUT);
}
void temperature()
{
  int adcVal = analogRead(PIN_LM35);
  float milliVolt = adcVal * (ADC_VREF_mV / ADC_RESOLUTION); float
  tempC = milliVolt / 10; Serial.print("Temperature: ");
  Serial.print(tempC);
  Serial.print("'C"); if(tempC
  > 60)
  {
    Serial.println("Alert");
    digitalWrite(Motor_PIN, HIGH); // turn on
  }
  else
  {
    digitalWrite(Motor_PIN, LOW); // turn off
  }
}
void GasSensors()
{int gassensorAnalogmq2 = analogRead(mq2);
  Serial.print("mq2 Gas Sensor: ");
  Serial.print(gassensorAnalogmq2);
```

```

Serial.print("\t");
Serial.print("\t");
Serial.print("\t");
if (gassensorAnalogmq2 > 1500)
{
  Serial.println("mq2Gas");//message to user
  Serial.println("Alert");
}
else
{
  Serial.println("No mq2Gas");//message to user
}
}
void flamesensor()
{
  flame_pin = digitalRead ( flame_sensor_pin ) ; // reading from the
  sensor
  if (flame_pin == LOW ) // applying condition
  {
    Serial.println ( " ALERT: FLAME DETECTED" ) ;
    digitalWrite ( Motor_PIN , HIGH ) ;// if state is high, then turn high
    the BUZZER
  }
  else
  {
    Serial.println ( " NO FLAME DETECTED " ) ;
    digitalWrite ( Motor_PIN , LOW ) ; // otherwise turn it low
  }
}
void loop()
{
  temperature(
);
  GasSensors();
  flame

```