FINAL CODE

TEAM ID :PNT2022TMID52298

```
#include<LiquidCrystal.h>
#IBM Watson IOT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "gv6nq",
        "typeId": "GasLeakageAndMonitoring",
        "deviceId":"12345"
    },
    "auth": {
        "token": "use-token-auth"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']
```

```
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()



LiquidCrystal lcd(6, 7, 8, 9, 10, 11);
float gasPin = A0;
float gasLevel;
int ledPin = 2;
int buttonPin = 3;
int buzzPin = 4;
int buttonState;
int fan = 5;

void setup(){
  pinMode(ledPin, OUTPUT);
  pinMode(buttonPin, INPUT);
  pinMode(gasPin,INPUT);
  pinMode(fan,OUTPUT);
  Serial.begin(9600);
  lcd.begin(16, 2);
  lcd.setCursor(0,0);
  lcd.print(" Welcome");
  lcd.setCursor(0,2);
  lcd.print("PNT2022TMID51246");
  delay(500);
  lcd.clear();
}

void loop(){
  // Read the value from gas sensor and button
  gasLevel = analogRead(gasPin);
  buttonState = digitalRead(buttonPin);

  // call the function for gas detection and button work
  gasDetected(gasLevel);
  buzzer(gasLevel);
  exhaustFanOn(buttonState);
}

// Gas Leakage Detection & Automatic Alarm and Fan ON
void gasDetected(float gasLevel){
  if(gasLevel >= 200){
   digitalWrite(buzzPin,HIGH);
    digitalWrite(ledPin,HIGH);
    digitalWrite(fan,HIGH);
    lcd.setCursor(0,0);
   lcd.print("GAS:");
    lcd.print(gasLevel);
   lcd.setCursor(0,2);
   lcd.print("FAN ON");
   delay(1000);
   lcd.clear();
  }else{
   digitalWrite(ledPin,LOW);
    digitalWrite(buzzPin,LOW);
    digitalWrite(fan,LOW);
    lcd.setCursor(0,0);
   lcd.print("GAS:");
    lcd.print(gasLevel);
   lcd.setCursor(0,2);
   lcd.print("FAN OFF");
   delay(100);
   lcd.clear();
  }
}
//BUZZER
void buzzer(float gasLevel){
if(gasLevel>=200)
  {
  for(int i=0; i<=30; i=i+10)
  {
  tone(4,i);
  delay(300);
  noTone(4);
  delay(4300);
  }
  }
}

// Manually Exhaust FAN ON
void exhaustFanOn(int buttonState){
  if(buttonState == HIGH){
    digitalWrite(fan,HIGH);
    lcd.setCursor(0,0);
   lcd.print("Button State:");
    lcd.print(buttonState);
    lcd.setCursor(0,2);
    lcd.print("FAN ON");
    delay(10000);
    lcd.clear();
  }



}
```