#include "MQ135.h"

#include <SoftwareSerial.h>

#define DEBUG true

SoftwareSerial esp8266(9,10); // This makes pin 9 of Arduino as RX pin and pin 10 of Arduino as the TX pin

const int sensorPin= 0;

int air\_quality;

#include <LiquidCrystal.h> 

LiquidCrystal lcd(12,11, 5, 4, 3, 2);

void setup() {

pinMode(8, OUTPUT);

lcd.begin(16,2);

lcd.setCursor (0,0);

lcd.print ("circuitdigest ");

lcd.setCursor (0,1);

lcd.print ("Sensor Warming ");

delay(1000);

Serial.begin(115200);

esp8266.begin(115200); // your esp's baud rate might be different

  sendData("AT+RST\r\n",2000,DEBUG); // reset module

  sendData("AT+CWMODE=2\r\n",1000,DEBUG); // configure as access point

  sendData("AT+CIFSR\r\n",1000,DEBUG); // get ip address

  sendData("AT+CIPMUair\_quality=1\r\n",1000,DEBUG); // configure for multiple connections

  sendData("AT+CIPSERVER=1,80\r\n",1000,DEBUG); // turn on server on port 80

pinMode(sensorPin, INPUT);        //Gas sensor will be an input to the arduino

lcd.clear();

}

void loop() {

MQ135 gasSensor = MQ135(A0);

float air\_quality = gasSensor.getPPM();

if(esp8266.available()) // check if the esp is sending a message 

  {

    if(esp8266.find("+IPD,"))

    {

     delay(1000);

     int connectionId = esp8266.read()-48; /\* We are subtracting 48 from the output because the read() function returns the ASCII decimal value and the first decimal number which is 0 starts at 48\*/ 

     String webpage = "<h1>IOT Air Pollution Monitoring System</h1>";

       webpage += "<p><h2>";   

       webpage+= " Air Quality is ";

       webpage+= air\_quality;

       webpage+=" PPM";

       webpage += "<p>";

     if (air\_quality<=1000)

{

  webpage+= "Fresh Air";

}

else if(air\_quality<=2000 && air\_quality>=1000)

{

  webpage+= "Poor Air";

}

else if (air\_quality>=2000 )

{

webpage+= "Danger! Move to Fresh Air";

}

webpage += "</h2></p></body>"; 

     String cipSend = "AT+CIPSEND=";

     cipSend += connectionId;

     cipSend += ",";

     cipSend +=webpage.length();

     cipSend +="\r\n";

     sendData(cipSend,1000,DEBUG);

     sendData(webpage,1000,DEBUG);

     cipSend = "AT+CIPSEND=";

     cipSend += connectionId;

     cipSend += ",";

     cipSend +=webpage.length();

     cipSend +="\r\n";

     String closeCommand = "AT+CIPCLOSE="; 

     closeCommand+=connectionId; // append connection id

     closeCommand+="\r\n";

     sendData(closeCommand,3000,DEBUG);

    }

  }

lcd.setCursor (0, 0);

lcd.print ("Air Quality is ");

lcd.print (air\_quality);

lcd.print (" PPM ");

lcd.setCursor (0,1);

if (air\_quality<=1000)

{

lcd.print("Fresh Air");

digitalWrite(8, LOW);

}

else if( air\_quality>=1000 && air\_quality<=2000 )

{

lcd.print("Poor Air, Open Windows");

digitalWrite(8, HIGH );

}

else if (air\_quality>=2000 )

{

lcd.print("Danger! Move to Fresh Air");

digitalWrite(8, HIGH);   // turn the LED on

}

lcd.scrollDisplayLeft();

delay(1000);

}

String sendData(String command, const int timeout, boolean debug)

{

    String response = ""; 

    esp8266.print(command); // send the read character to the esp8266

    long int time = millis();

    while( (time+timeout) > millis())

    {

      while(esp8266.available())

      {

        // The esp has data so display its output to the serial window 

        char c = esp8266.read(); // read the next character.

        response+=c;

      }  

    }

    if(debug)

    {

      Serial.print(response);

    }

    return response;

}