**WEATHER STATION**

**Description**: Weather monitoring using Temperature Sensor and Smoke Sensor.

**Components Required:**

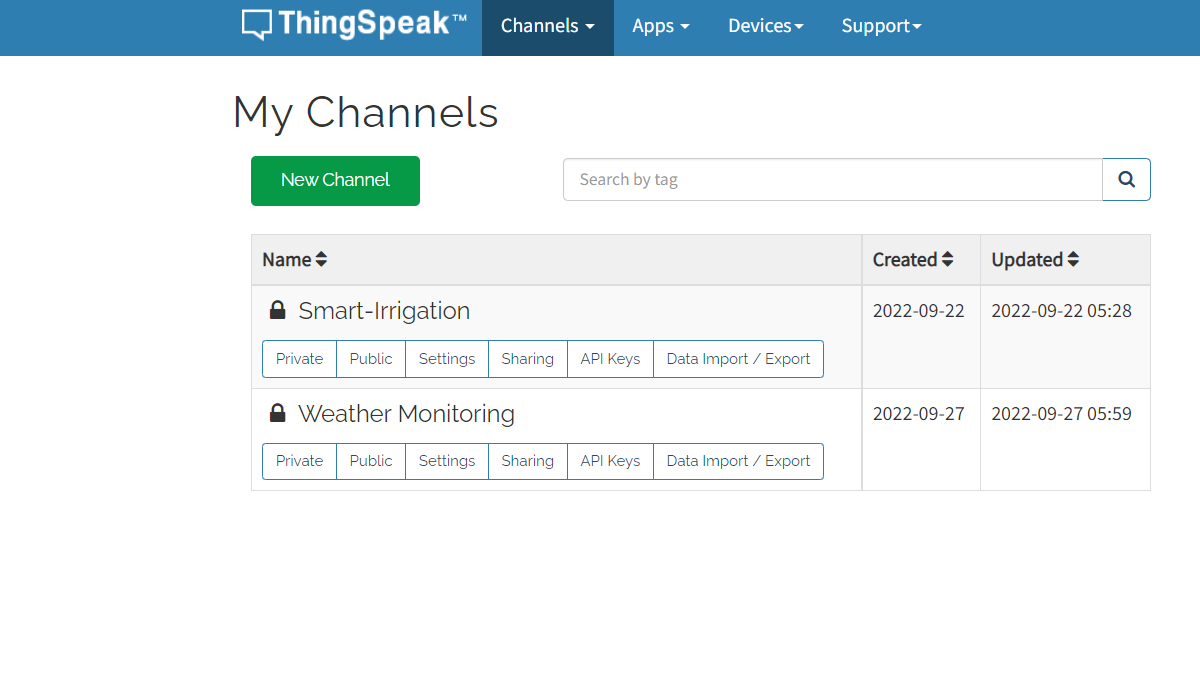
1. Arduino UNO
2. WIFI module
3. Smoke Sensor
4. Temperature Sensor (LM35)
5. Power Supply
6. Connectors
7. ThingSpeak (IOT analytic platform service)

**Creating Channel in ThingSpeak**

**Step 01:** Create ThingSpeak Account. The URL is given below.

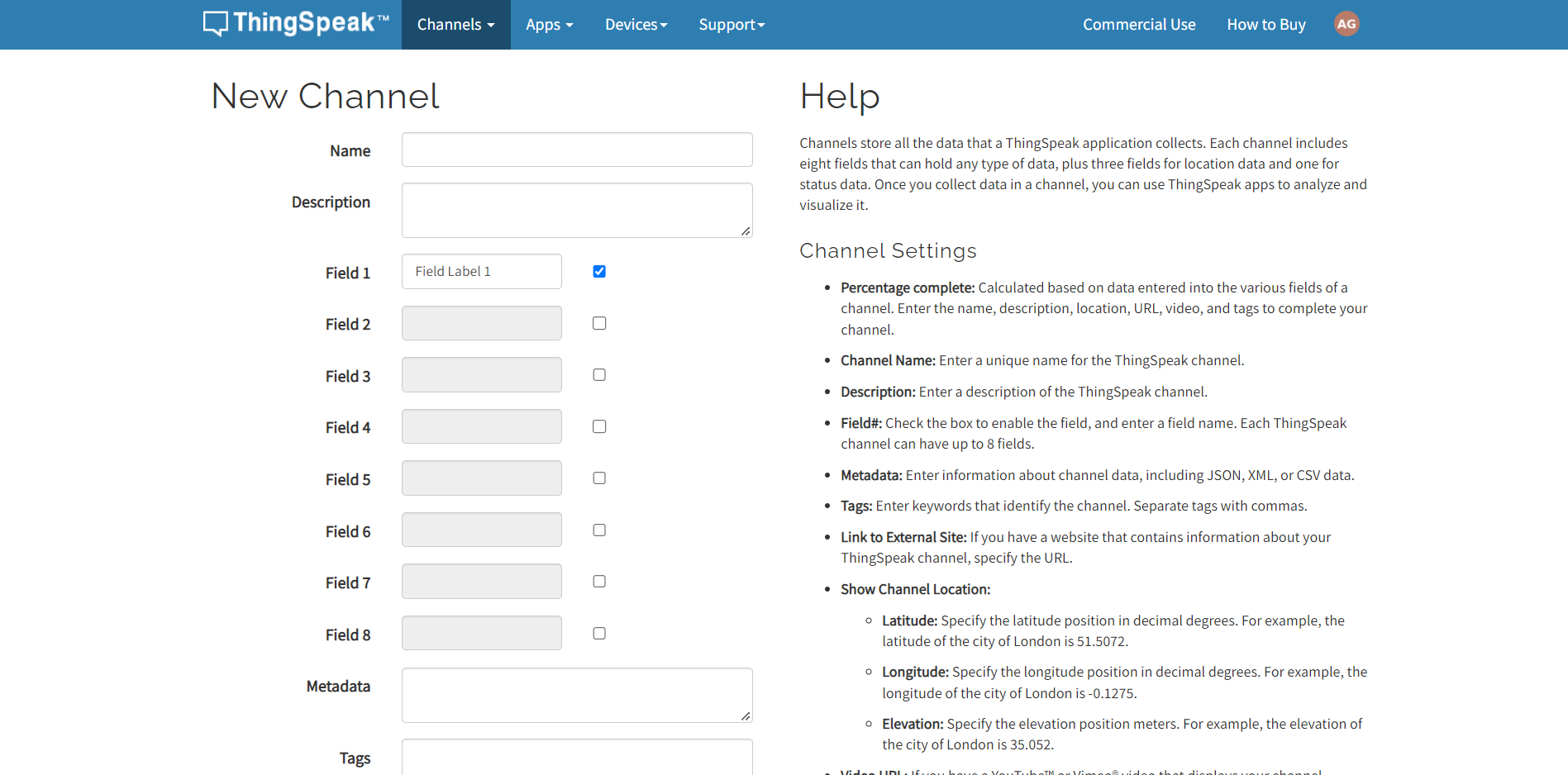
URL: <https://thingspeak.com/>

**Step 02:**

****

Click On New Channel.

**Step 03:** Providing Details



Enter the Name, and Description.

Choose the number of fields required.

Step 04: Save changes

**CODE:**

//WEATHER MONITORING PROJECT.01

#include <SoftwareSerial.h>

#define RX 8 // it works as a UART

#define TX 9 // so Rx is connected to Pin no 9 in Arduino Uno, Tx is connected to pin no 8 in UNO

SoftwareSerial esp8266(RX,TX);

String inputString = "";

boolean RX\_ST\_Flag = false;

boolean stringComplete = false;

String AP = "Nokia"; // CHANGE ME HOTSPOT NAME

String PASS = "12345678"; // CHANGE ME HOTSPOT NAME password

String API = "5AYF6UISFWRT03F9"; // CHANGE ME

String HOST = "api.thingspeak.com";

//String HOST = "184.106.153.149";

String PORT = "80";

String field1 = "field1";

String field2 = "field2";

String field3 = "field3"; //Extra Field is taken inCase if we use another sensor and forget to mention, we can use this

int countTrueCommand;

int countTimeCommand;

boolean found = false;

void setup()

{

delay(500);

Serial.begin(9600);

delay(500);

Serial.println("WEATHER MONITORING");

delay(500);

esp8266.begin(9600);

sendCommand("AT",5,"OK");

sendCommand("AT+CWMODE=1",5,"OK");

sendCommand("AT+CWJAP=\""+ AP +"\",\""+ PASS +"\"",20,"OK");

delay(500);

}

void loop()

{

int Temperature = analogRead(A0);

int Temperature\_Temp = (( Temperature/1024.0 )\*5000 ) / 10;

Serial.print( "Temperature = " );

Serial.println( Temperature\_Temp );

delay(2000);

int Smoke\_Sensor = analogRead(A1);

int Smoke\_Sensor\_Temp = ( ( Smoke\_Sensor/1024.0 )\*5000 ) / 10;

Serial.print( "Smoke\_Sensor = " );

Serial.println( Smoke\_Sensor\_Temp );

delay(2000);

// String getData = "GET /update?api\_key="+ API +"&"+ field1 +"="+String(Solar\_Volt\_Tx);+"&"+ field2 +"="+String(Wind\_Volt\_Tx);

String getData = "GET /update?api\_key="+ API +"&"+ field1 +"="+String(Temperature\_Temp)+"&"+ field2 +"="+String(Smoke\_Sensor\_Temp);

sendCommand("AT+CIPMUX=1",5,"OK");

sendCommand("AT+CIPSTART=0,\"TCP\",\""+ HOST +"\","+ PORT,15,"OK");

sendCommand("AT+CIPSEND=0," +String(getData.length()+4),4,">");

esp8266.println(getData);

delay(1500);

countTrueCommand++;

sendCommand("AT+CIPCLOSE=0",5,"OK");

delay(5000);

}

void sendCommand( String command, int maxTime, char readReplay[] )

{

Serial.print(countTrueCommand);

Serial.print(". at command => ");

Serial.print(command);

Serial.print(" ");

while(countTimeCommand < (maxTime\*1))

{

esp8266.println(command);//at+cipsend

if(esp8266.find(readReplay))//ok

{

found = true;

break;

}

countTimeCommand++;

}

if(found == true)

{

Serial.println("OYI");

countTrueCommand++;

countTimeCommand = 0;

}

if(found == false)

{

Serial.println("Fail");

countTrueCommand = 0;

countTimeCommand = 0;

}

found = false;

}

void serialEvent()

{

while (Serial.available())

{

char inChar = (char)Serial.read(); // get the new byte:

inputString += inChar; // add it to the inputString:

if(inChar == '#')

{

RX\_ST\_Flag = true;

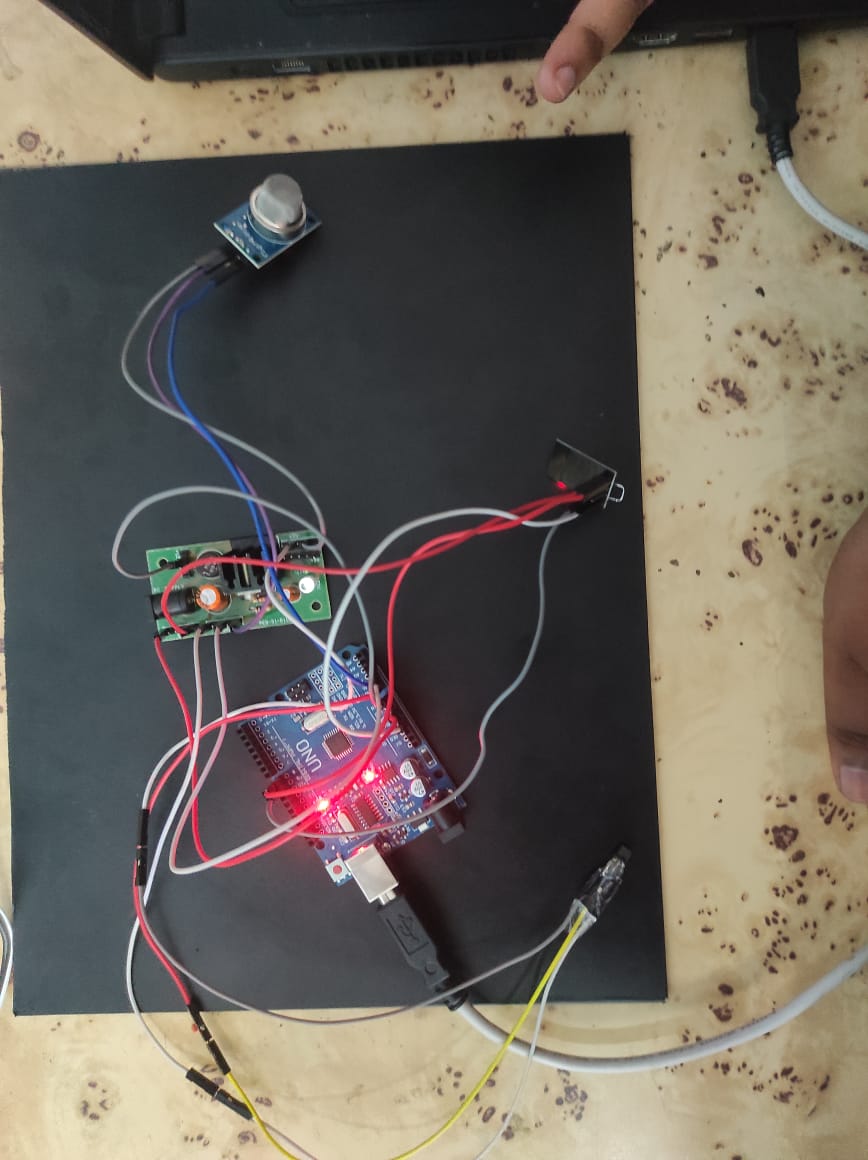
}

}

}

NOTE: Change the API key, String AP (Hotspot Name), String PASS (Hotspot password)

**CIRCUIT CONNECTION**



TEMPERATURE SENSOR

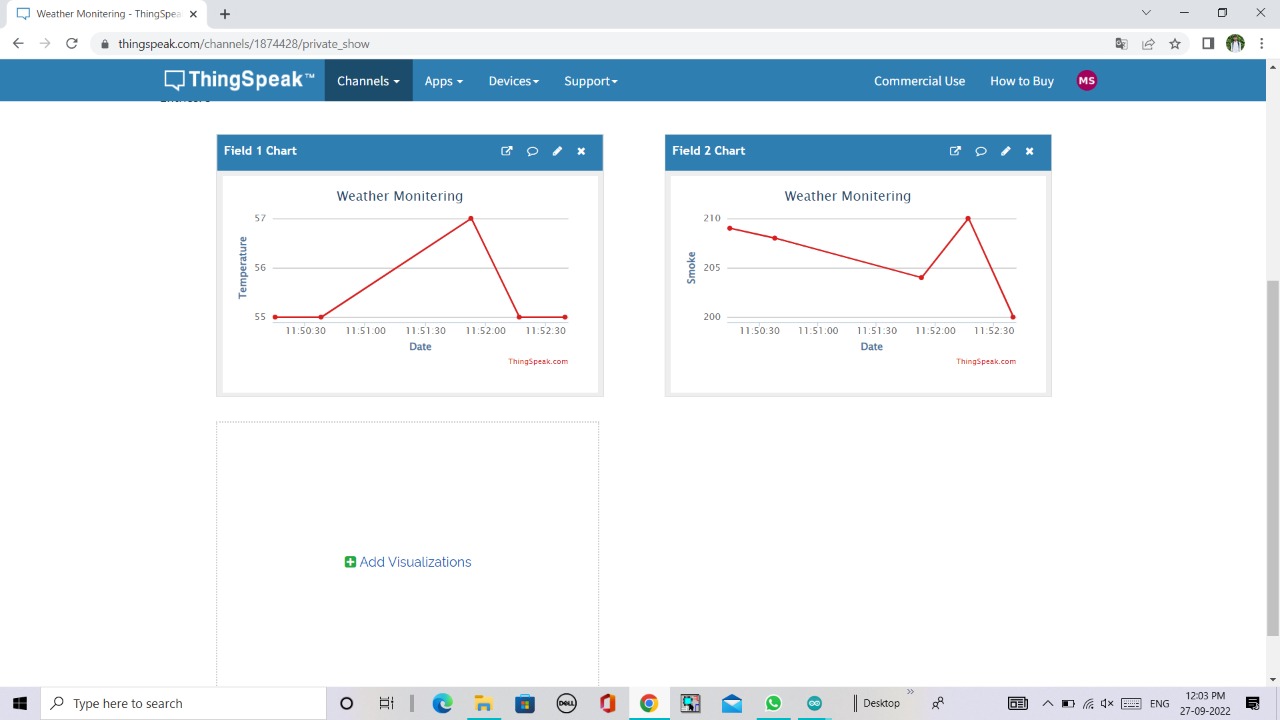
ARDUINO UNO

WIFI MODULE

SOMKE SENSOR

POWER SUPPLY

**RESULT SNAPSHOT:**



**Conclusion:** The above snapshots show the Real-Time Monitoring of Temperature and Smoke.