IIS PROJECT -1
(B.TECH - CSE 2022-2023)

D Y PATIL
INTERNATIONAL
UNIVERSITY
AKURDI PUNE

Project Title:

Temperature Logger With Motion Sensing Using Node MCU

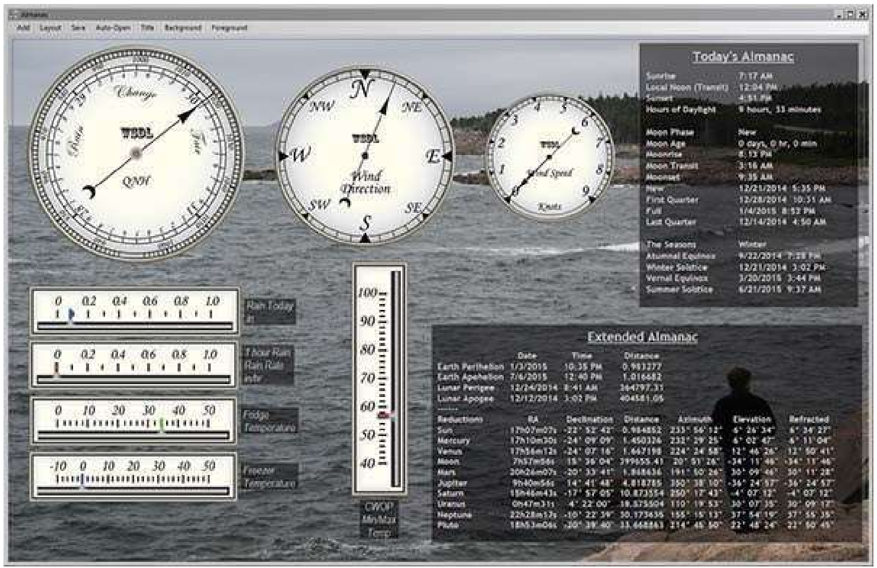
BY-

Nitin Jha- 20210802070

Rishab Debnath- 20210802094

Zaid Saudagar- 20210802127

Nikshay Kumar- 20210802137



Today's Almanac

Sunrise	7:17 AM
Local Noon (Transit)	12:04 PM
Sunset	4:51 PM
Hours of Daylight	9 hours, 33 minutes
Moon Phase	New
Moon Age	0 days, 0 hr, 0 min
Moonrise	8:13 PM
Moon Transit	3:16 AM
Moonset	9:35 AM
New	12/21/2014 5:35 PM
First Quarter	12/28/2014 10:31 AM
Full	1/4/2015 8:52 PM
Last Quarter	12/14/2014 4:50 AM
The Seasons	Winter
Autumnal Equinox	9/22/2014 7:29 PM
Winter Solstice	12/21/2014 3:02 PM
Vernal Equinox	3/20/2015 3:44 PM
Summer Solstice	6/21/2015 9:37 AM

Extended Almanac

	Date	Time	Distance
Earth Perihelion	1/3/2015	10:35 PM	0.983277
Earth Aphelion	7/6/2015	12:40 PM	1.016682
Lunar Perigee	12/24/2014	8:41 AM	364797.31
Lunar Apogee	12/12/2014	3:02 PM	404581.05

Reductions	RA	Declination	Distance	Azimuth	Elevation	Refracted
Sun	17h07m07s	-22° 52' 42"	0.984852	233° 56' 12"	6° 26' 34"	6° 34' 27"
Mercury	17h10m30s	-24° 09' 09"	1.450326	232° 29' 25"	6° 02' 47"	6° 11' 04"
Venus	17h56m12s	-24° 07' 16"	1.667198	224° 24' 58"	12° 46' 26"	12° 50' 41"
Moon	7h57m56s	15° 36' 04"	379655.41	20° 51' 26"	-34° 11' 46"	-34° 11' 46"
Mars	20h26m07s	-20° 33' 41"	1.868636	191° 50' 26"	30° 09' 46"	30° 11' 28"
Jupiter	9h40m56s	14° 41' 48"	4.818785	350° 38' 10"	-36° 24' 57"	-36° 24' 57"
Saturn	15h46m43s	-17° 57' 05"	10.873554	250° 17' 43"	-4° 07' 12"	-4° 07' 12"
Uranus	0h47m31s	4° 22' 00"	18.575504	110° 19' 53"	30° 07' 35"	30° 09' 17"
Neptune	22h28m53s	-10° 22' 39"	30.173636	155° 15' 13"	37° 54' 19"	37° 55' 35"
Pluto	18h53m06s	-20° 39' 40"	33.668863	214° 46' 50"	22° 48' 24"	22° 50' 45"

ABSTRACT

The weather logger with motion sensing is a device that can be used to monitor and record the temperature, humidity, and motion in an indoor or outdoor environment. The device consists of a temperature and humidity sensor, a motion sensor, a microcontroller, and a memory module. The temperature and humidity sensor can measure the temperature and humidity of the surrounding air, while the motion sensor can detect any movement in the environment.
The microcontroller is responsible for collecting data from the sensors and storing it in the memory module. It can also be programmed to send the data to a computer or a mobile device via Bluetooth or Wi-Fi. The device can be powered by a battery or a solar panel, making it suitable for long-term monitoring in remote locations.

The data collected by the weather logger with motion sensing can be used to analyze the environmental conditions in a particular location. For example, it can be used to monitor the temperature and humidity in a greenhouse or a data center, or to detect any unexpected motion in a secured area. The device can be customized to send alerts or notifications if certain conditions are met, such as if the temperature or humidity exceeds a certain threshold or if motion is detected during a specific time period.

Overall, the weather logger with motion sensing is a versatile device that can provide valuable insights into the environmental conditions of a particular location, making it a useful tool for a variety of applications.

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

- A weather logger is a device that is used to monitor and record environmental conditions such as temperature, humidity, air pressure, wind speed, and rainfall. It is a valuable tool for researchers, meteorologists, farmers, and anyone who needs to track weather patterns and make informed decisions based on the data collected.
- Weather loggers can range from simple devices that measure only one or two parameters to more advanced devices that can measure multiple parameters simultaneously. They can be installed indoors or outdoors, and can be powered by batteries, solar panels, or AC power.
- The data collected by weather loggers can be used for a variety of purposes, such as studying climate patterns, predicting weather changes, optimizing crop irrigation and fertilization, and monitoring indoor air quality. It can also be used to alert people to potential weather hazards, such as severe storms or heatwaves.
- Overall, weather loggers provide an efficient and cost-effective way to monitor and record environmental conditions over time, which can help inform decisions and improve outcomes in a variety of fields.

Temperature And Humidity

Temperature is a measure of the degree of hotness or coldness of an object or environment. It is an important environmental parameter that is measured and recorded by various devices, including thermometers and temperature sensors.Temperature is typically measured in degrees Celsius (°C) or degrees Fahrenheit (°F), and is influenced by various factors such as solar radiation, humidity, air pressure, and wind speed. Changes in temperature can have significant effects on the environment, including changes in weather patterns, plant growth, and animal behavior.Temperature is an important environmental parameter that plays a significant role in a wide range of applications, and accurate temperature measurement and monitoring is essential for understanding and managing environmental conditions.

Humidity is a measure of the amount of water vapor present in the air or in a gas. It is an important environmental parameter that is measured and recorded by various devices, including hygrometers and humidity sensors.Humidity can be measured in several ways, including relative humidity, absolute humidity, and specific humidity. Relative humidity is the most commonly used measure of humidity and represents the amount of water vapor present in the air relative to the maximum amount that could be present at a given temperature.

Motion sensing refers to the process of detecting movement or changes in an environment using sensors or cameras. This technology has become increasingly popular in recent years, as it has many applications in various fields, including security, gaming, and health monitoring.There are different types of motion sensors available, including passive infrared (PIR) sensors, ultrasonic sensors, microwave sensors, and image sensors. PIR sensors are commonly used in security systems, as they can detect the infrared radiation emitted by living beings. Ultrasonic sensors emit high-frequency sound waves and detect the echoes that bounce back, allowing them to measure distance and detect movement. Microwave sensors emit microwave radiation and detect the reflections, allowing them to detect motion through walls and other obstacles. Image sensors, such as those found in cameras, can detect motion by analyzing changes in the image.

CHAPTER 2

Literature survey

- A literacy survey in temperature logger is a survey that aims to assess the level of understanding and proficiency of individuals in using and interpreting temperature loggers. The survey can be conducted in various settings, such as in schools, workplaces, and research institutions.

- The survey can include questions that test the knowledge and skills of individuals in using temperature loggers, including the types of temperature loggers available, the principles of temperature measurement, the factors that can affect temperature readings, and the procedures for calibrating and maintaining temperature loggers.

- The survey can also assess the ability of individuals to interpret temperature data, including the ability to identify patterns and trends in temperature data, to recognize anomalies or outliers in the data, and to make informed decisions based on the data.

- The results of a literacy survey in temperature logger can provide valuable insights into the level of understanding and proficiency of individuals in using and interpreting temperature loggers. This information can be used to identify areas where additional training or education may be needed, to improve the accuracy and reliability of temperature data, and to optimize the use of temperature loggers in various applications.

The literacy survey on weather logger system would likely cover thefollowing areas:

1. Understanding of weather variables: The survey would assess respondents' knowledge of temperature, humidity, pressure, and other weather variables, including their units of measurement and how they are affected by different environmental factors.

2. Understanding of sensors: The survey would assess respondents' knowledge of different types of sensors used in weather logging systems, including their principles of operation and advantages and disadvantages.

3. Understanding of data logging and analysis: The survey would assess respondents' knowledge of how weather data is logged and stored, as well as their ability to analyze and interpret the data to draw meaningful conclusions.

4. Understanding of programming and electronics: The survey would assess respondents' knowledge of programming and electronics, including their ability to program microcontrollers and use tools such as breadboards, sensors, and data loggers.

5. Practical experience: The survey would also ask about respondents' practical experience with weather logging systems, including any projects they have completed or challenges they have faced.

Overall, the literacy survey on weather logging systems would aim to assess respondents' knowledge and skills related to weather monitoring and data analysis, as well as their practical experience in working with weather logging systems.

A motion sensing system is a device or technology that detects movement or changes in motion within its surrounding environment. Motion sensing systems are used in a wide range of applications, including security systems, gaming, and navigation.

There are several different types of motion sensing systems, including:

1. Passive Infrared (PIR) Sensors: PIR sensors detect changes in infrared radiation emitted by warm objects within their field of view. They are commonly used in security systems to detect the presence of people or animals.

2. Ultrasonic Sensors: Ultrasonic sensors emit high-frequency sound waves and detect changes in the echo reflected back from nearby objects. They are commonly used in parking sensors and robotics applications.

3. Microwave Sensors: Microwave sensors emit low-power microwaves and detect changes in the reflection of those waves caused by nearby objects. They are commonly used in motion detectors and traffic monitoring systems.

4. Image Sensors: Image sensors use cameras or other imaging technologies to detect changes in the visual field. They are commonly used in gaming and virtual reality systems.

Motion sensing systems can be used to automate tasks, enhance security, and improve safety in a wide range of environments. They can also be integrated with other technologies, such as artificial intelligence and machine learning algorithms, to improve their accuracy and performance.

Overall, motion sensing systems are a useful and versatile technology for detecting and tracking movement in a variety of applications. Their ability to detect motion and changes in motion

There has been significant previous work in the development of temperature loggers and motion sensing systems.

In the field of temperature loggers, many commercially available devices exist that can record temperature data over time. These devices can range from simple standalone data loggers to more complex systems that are integrated with wireless communication networks for remote data access and monitoring. Some examples of commercially available temperature loggers include products from companies such as Onset, Fluke, and Omega Engineering.

In addition to commercial devices, there have been several research studies focused on developing temperature loggers for specific applications. For example, a recent study published in the Journal of Agricultural and Food Chemistry developed a wireless temperature and humidity monitoring system for use in food storage facilities.

Similarly, motion sensing systems have been widely studied and developed for various applications. These systems range from simple motion detectors to more complex systems that utilize advanced sensing technologies such as computer vision and machine learning algorithms.

Some notable examples of motion sensing systems include the Kinect sensor developed by Microsoft for use in gaming and entertainment, and the LIDAR sensor used in autonomous vehicles for obstacle detection and navigation.

In terms of research studies, there has been significant work done in developing motion sensing systems for healthcare applications, such as fall detection and activity monitoring for elderly individuals.

Overall, there has been significant previous work in the development of temperature loggers and motion sensing systems, and these technologies continue to be actively researched and developed for a wide range of applications.

CHAPTER 3

HARDWARE REQUIREMENTS

1. Node MCU(ESP8266)

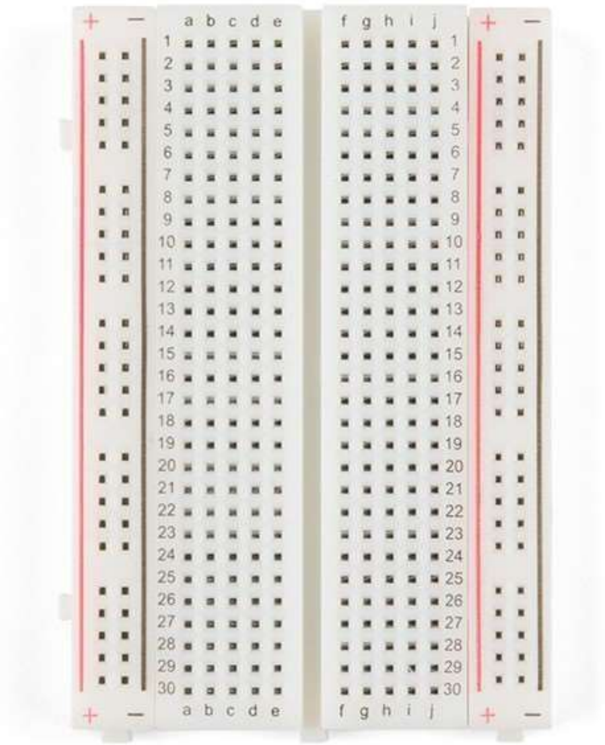
- ESP8266 is a low-cost, Wi-Fi-enabled microcontroller designed for the Internet of Things (IoT) applications. It was developed by Espressif Systems and was released in 2014. The ESP8266 has a built-in Wi-Fi module, which makes it easy to connect to wireless networks and communicate with other devices over the internet.The ESP8266 is based on the Tensilica L106 32-bit microcontroller unit (MCU) with a clock speed of 80 MHz. It also has 64KB of instruction RAM, 96KB of data RAM, and 4MB of flash memory for program storage. The microcontroller supports several communication protocols, including SPI, I2C, UART, and PWM.

-

2. BreadBoard

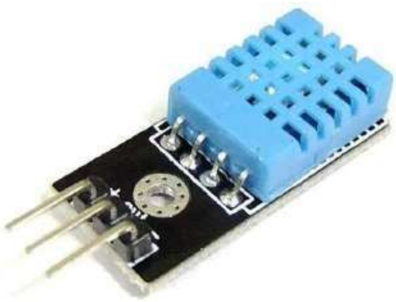
- A breadboard is a prototyping tool used in electronics to construct and test circuits without the need for soldering. It consists of a plastic board with a grid of small holes arranged in a regular pattern. The holes are interconnected by metal strips or wires inside the board, which allows components to be easily plugged in and connected.The layout of a breadboard is designed to mimic the layout of a typical electronic circuit, with rows and columns of interconnected holes. The board is typically divided into twosections, with the top section containing the power supply and the bottom section containing the ground connection.Components such as resistors, capacitors, and integrated circuits can be easily inserted into the holes on the breadboard and connected using jumper wires. The breadboard allows for quick and easy changes to the circuit design, since components can be easily moved or replaced.

-



3. Temperature Sensor

A temperature sensor is an electronic device that measures the temperature of its surrounding environment and converts the temperature into an electrical signal that can be read and interpreted by other devices. Temperature sensors are used in a wide range of applications, including environmental monitoring, industrial processes, and consumer electronics.



4. Ultrasonic Sensor

An ultrasonic sensor is a device that uses sound waves to detect the distance to, or presence of, an object. It emits high-frequency sound waves (ultrasound) and then listens for the echo reflected back from nearby objects. The time it takes for the echo to return is used to determine the distance to the object.

4. ***PIR Sensor***

A PIR (Passive Infrared) sensor is an electronic device that detects the infrared energy radiated by humans andanimals in their immediate vicinity. The sensor detects the movement of heat sources, which can include people, animals, and even cars, and then triggers an alarm or activates a device, such as turning on a light, when motion is detected.

6. Jumper Wires

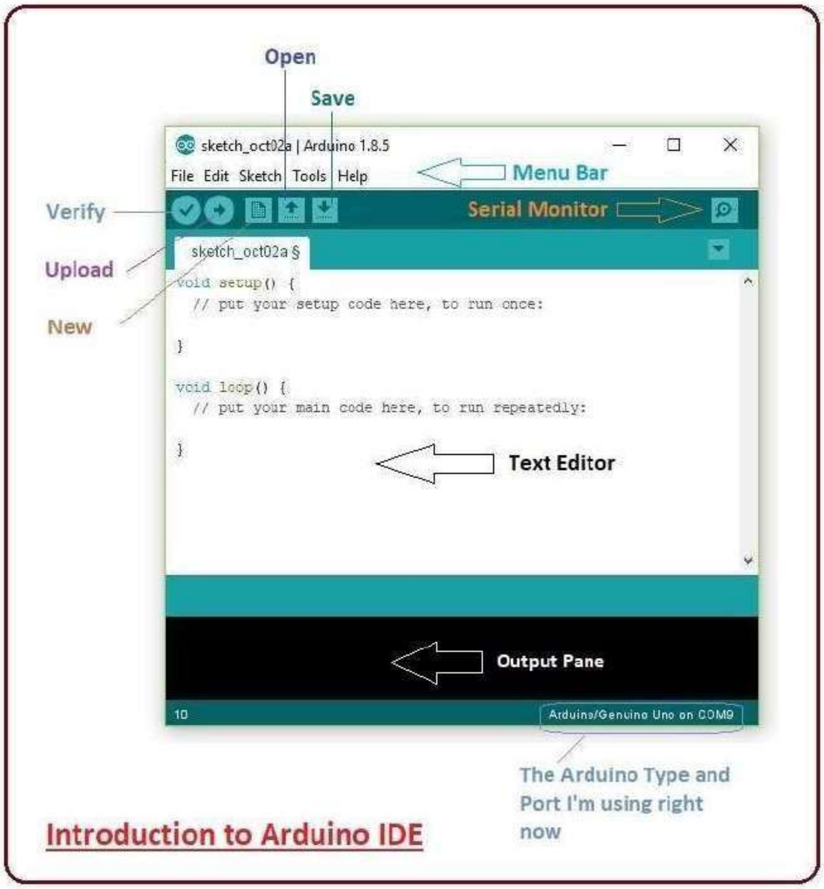
Jumper wires are a type of wire used to create connections between electronic \ components on a breadboard, printed circuit board (PCB), or other electronic circuits. They are typically made of insulated copper wire and come in different lengths, colors, and gauges.

CHAPTER 4

SOFTWARE REQUIREMENTS

1. Arduino IDE

Arduino IDE (Integrated Development Environment) is a software platform used to program and develop applications for Arduino boards. Arduino is an open-source electronics platform based on easy-to-use hardware and software.The Arduino IDE provides a simple, easy-to-use interface for writing, compiling, and uploading code to an Arduino board. It includes a code editor, a compiler, and a serial monitor for debugging and testing the code.The IDE supports C and C++ programming languages and provides a set of libraries that simplify the programming of complex functionalities. The libraries contain pre-written code for common tasks, such as controlling LEDs, reading sensors, and communicating with other devices. This allows developers to focus on creating the logic for their applications rather than dealing with low-level hardware details.

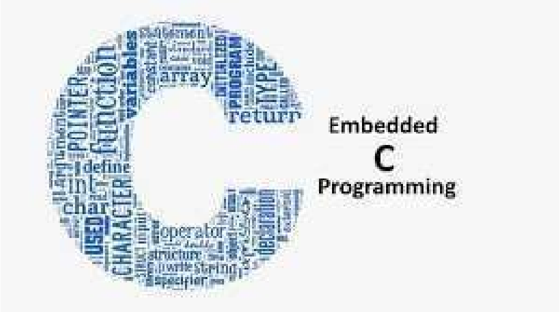

Open
Save
sketch_oct02a | Arduino 1.8.5
File Edit Sketch Tools Help
Menu Bar
Verify
Serial Monitor
Upload
sketch_oct02a §
New
Text Editor
Output Pane
10
Arduino/Genuino Uno on COM9
The Arduino Type and Port I'm using right now

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Introduction to Arduino IDE

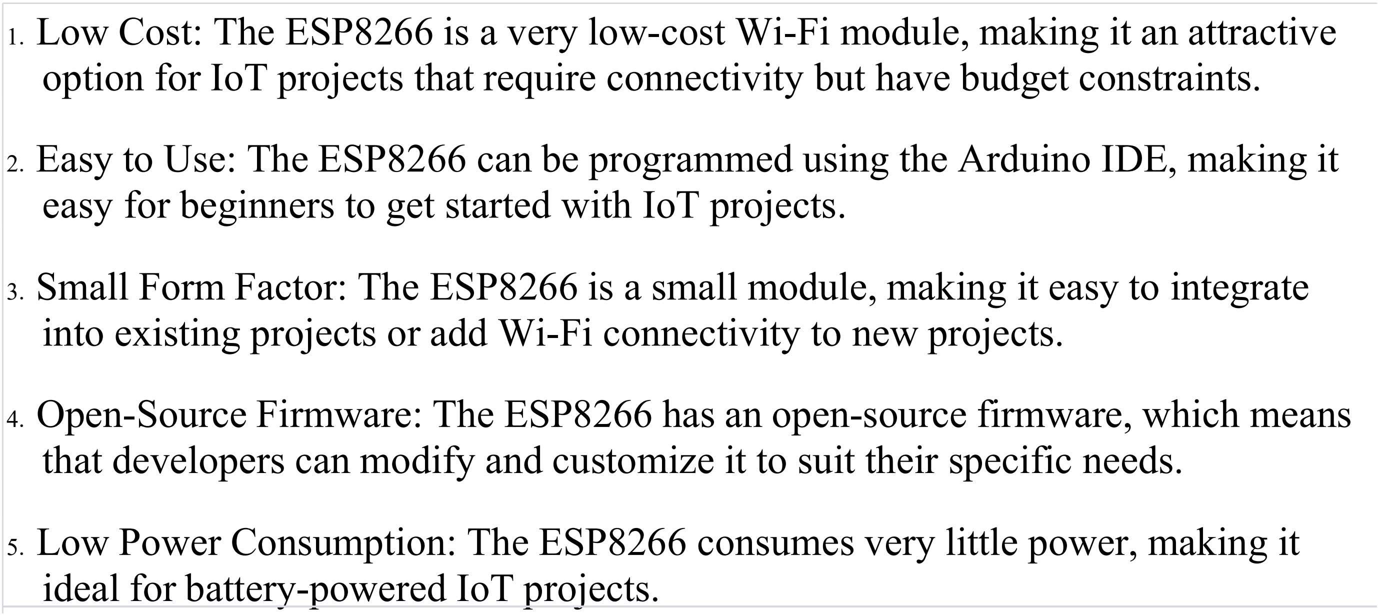
2. Embedded C

Embedded C is a programming language specifically designed for programming embedded systems, which are small computers integrated into a larger system or device to control its functionality. These systems are typically used in industries such as automotive, medical, aerospace, and consumer electronics.Embedded C is a subset of the C programming language, with added features specific to embedded systems. It includes built-in functions and macros for working with hardware, such as input/output (I/O) ports, timers, interrupts, and memory-mapped registers.



3. ESP 8266 wifi module

The ESP8266 is a low-cost Wi-Fi module that has gained popularity in recent years due to its ease of use and versatility. It is commonly used in Internet of Things (IoT) projects, allowing devices to connect to the internet and communicate with other devices over Wi-Fi.
Some of the key features of the ESP8266 Wi-Fi module include:



1. Low Cost: The ESP8266 is a very low-cost Wi-Fi module, making it an attractive option for IoT projects that require connectivity but have budget constraints.
2. Easy to Use: The ESP8266 can be programmed using the Arduino IDE, making it easy for beginners to get started with IoT projects.
3. Small Form Factor: The ESP8266 is a small module, making it easy to integrate into existing projects or add Wi-Fi connectivity to new projects.
4. Open-Source Firmware: The ESP8266 has an open-source firmware, which means that developers can modify and customize it to suit their specific needs.
5. Low Power Consumption: The ESP8266 consumes very little power, making it ideal for battery-powered IoT projects.

CHAPTER 5

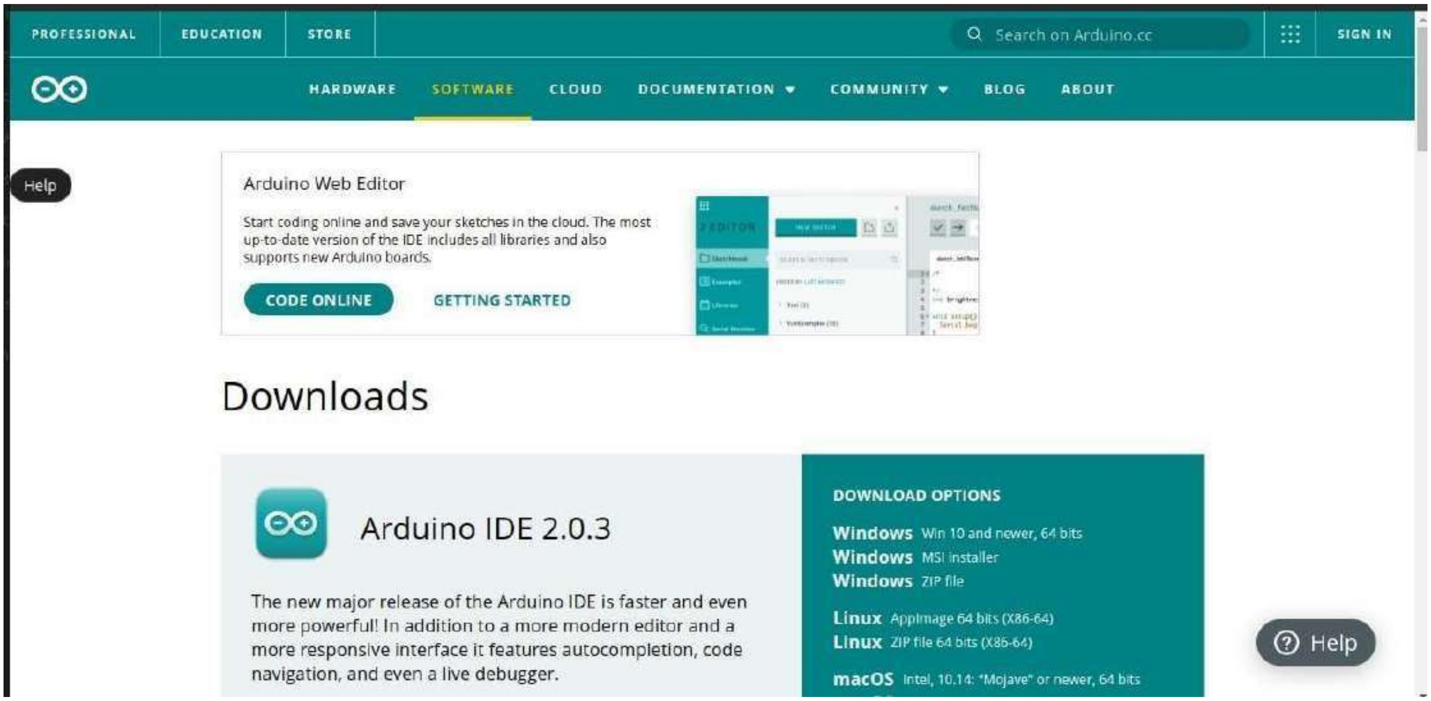
METHODOLOGY

STEP 1:- INSTALL ARDUINO IDE

- INSTALL ARDUINO IDE:-

Go to your search engine and enter this link:- https://www.arduino.cc/en/software this, link will redirect you to this page where you can install Arduino IDE compatable to your system.

-



Arduino Web Editor

Start coding online and save your sketches in the cloud. The most up-to-date version of the IDE includes all libraries and also supports new Arduino boards.

Downloads

Arduino IDE 2.0.3

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

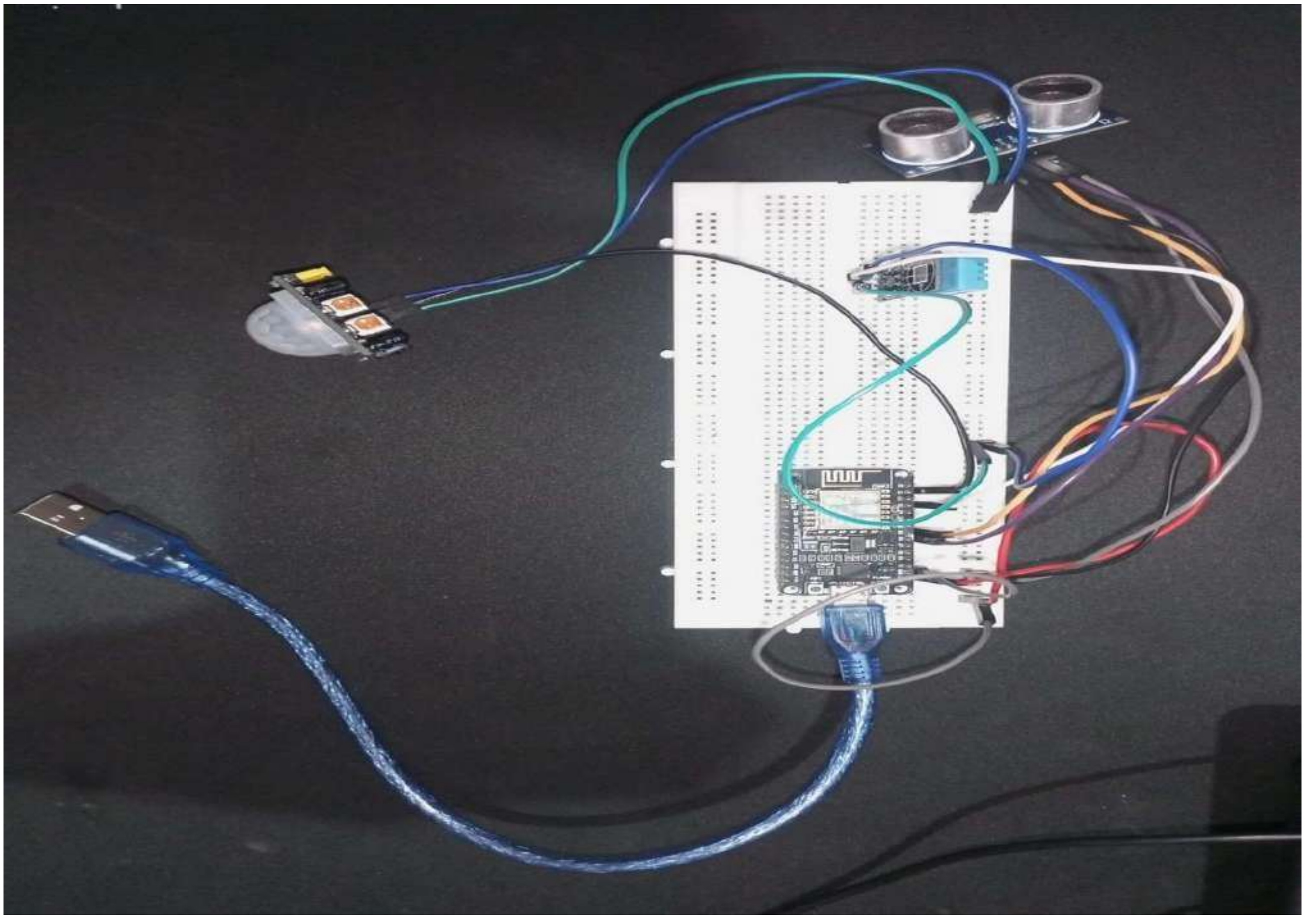
DOWNLOAD OPTIONS

Windows Win 10 and newer, 64 bits
Windows MSI installer
Windows ZIP file

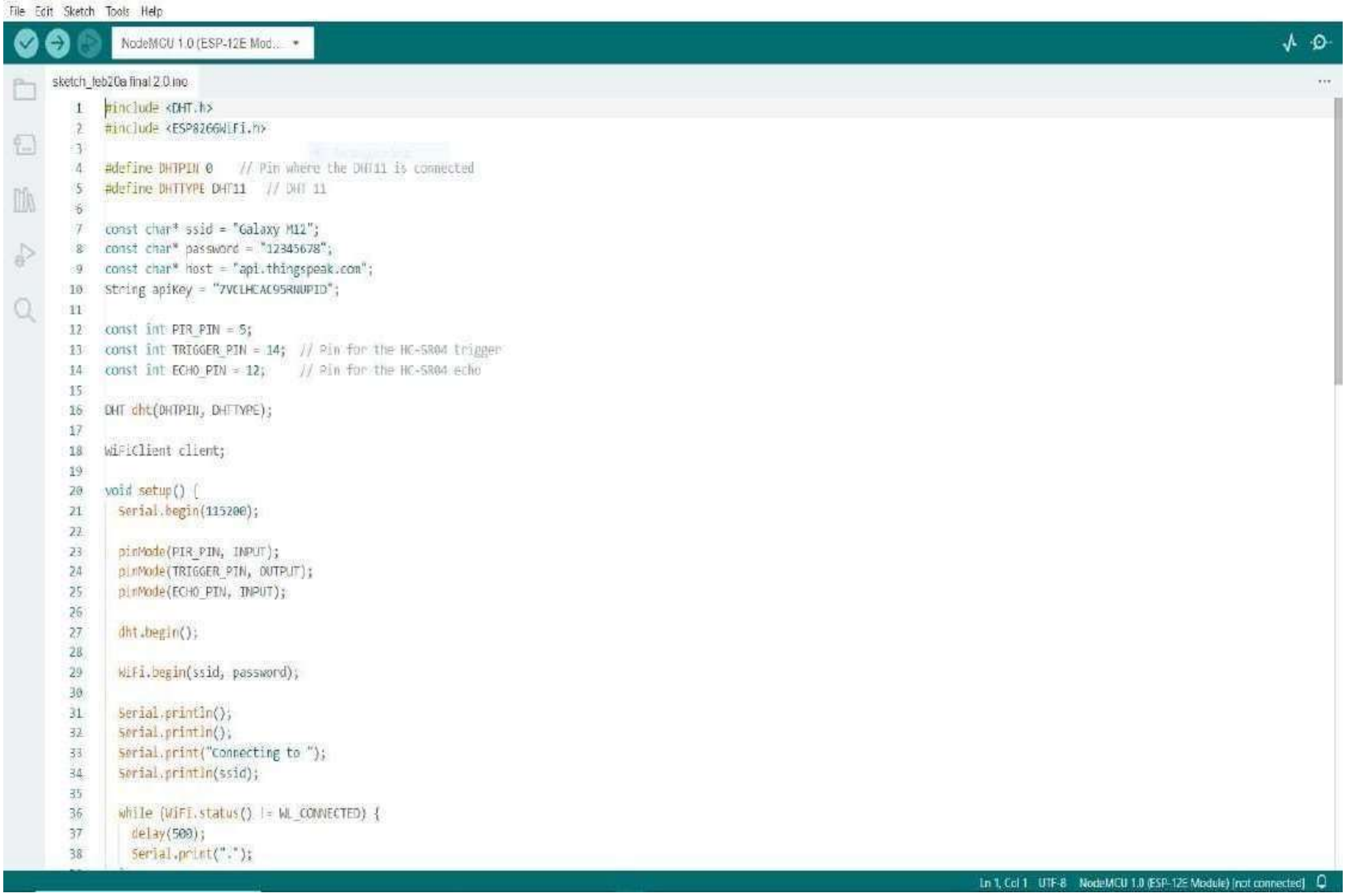
Linux AppImage 64 bits (X86-64)
Linux ZIP file 64 bits (X86-64)

macOS Intel, 10.14: "Mojave" or newer, 64 bits

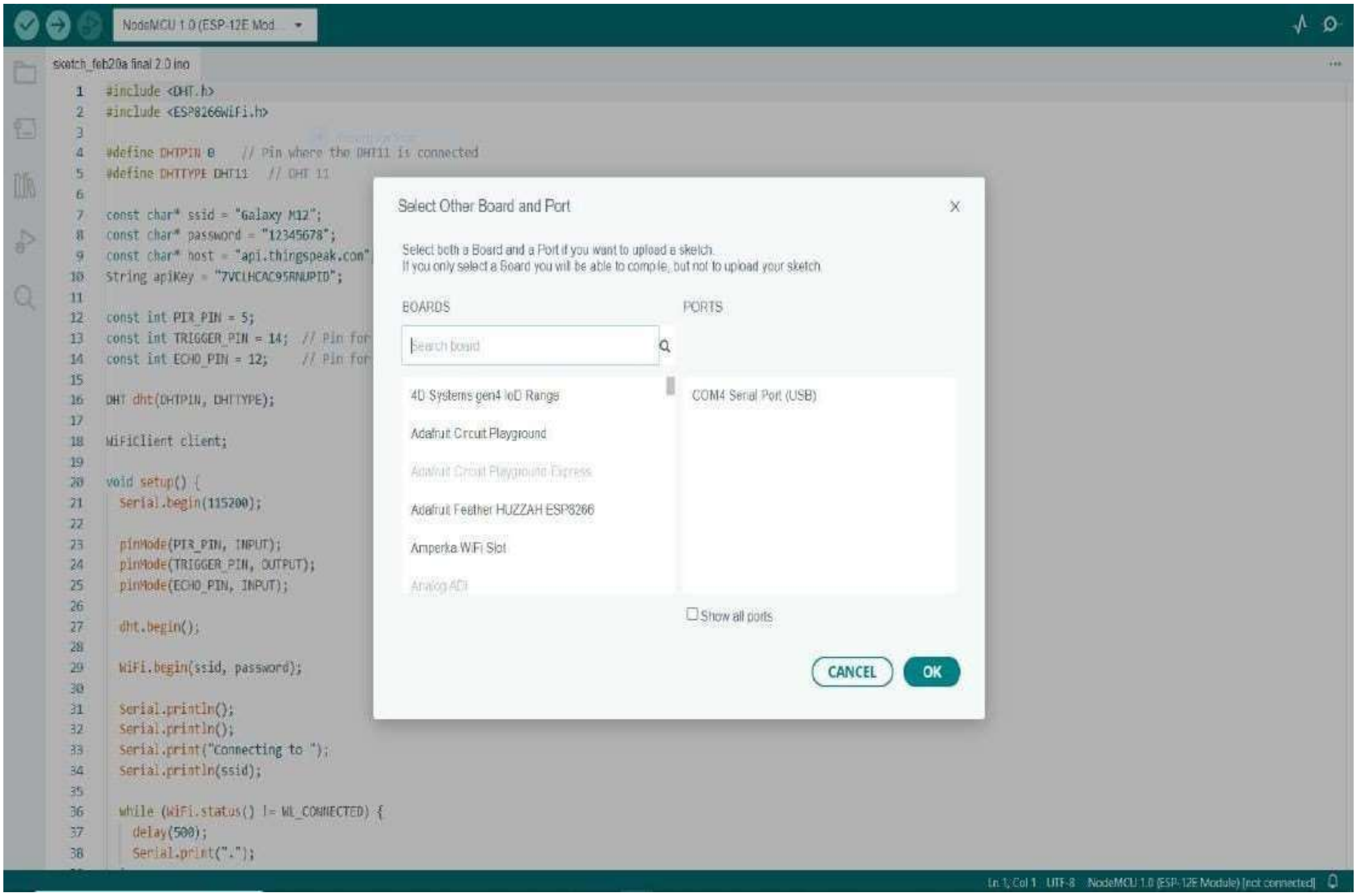
STEP 2:- Circuit Implementation:-



Step 3: Connecting the the hardware to the port


File Edit Sketch Tools Help
NodeMCU 1.0 (ESP-12E Mod...
sketch_feb20a final 2.0.ino
Ln 1, Col 1 UTF-8 NodeMCU 1.0 (ESP-12E Module) [not connected]

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

#define DHTPIN 0    // Pin where the DHT11 is connected
#define DHTTYPE DHT11   // DHT 11

const char* ssid = "Galaxy M12";
const char* password = "12345678";
const char* host = "api.thingspeak.com";
String apiKey = "7VCLHCAC95RNUPID";

const int PIR_PIN = 5;
const int TRIGGER_PIN = 14;  // Pin for the HC-SR04 trigger
const int ECHO_PIN = 12;     // Pin for the HC-SR04 echo

DHT dht(DHTPIN, DHTTYPE);

WiFiClient client;

void setup() {
  Serial.begin(115200);

  pinMode(PIR_PIN, INPUT);
  pinMode(TRIGGER_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);

  dht.begin();

  WiFi.begin(ssid, password);

  Serial.println();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
```



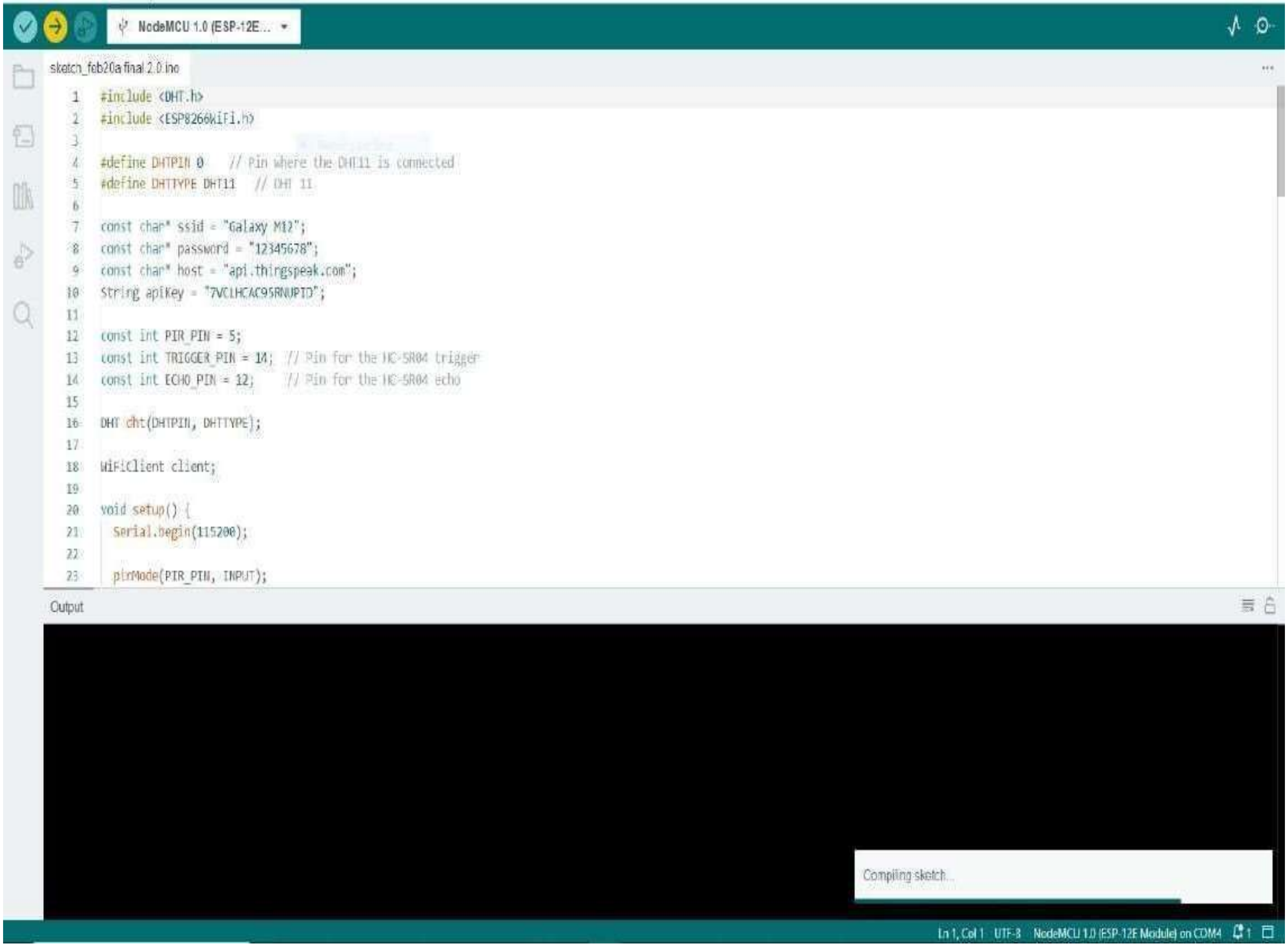
```
#include <DHT.h>
#include <ESP8266WiFi.h>

#define DHTPIN 0    // Pin where the DHT11 is connected
#define DHTTYPE DHT11   // DHT 11

const char* ssid = "Galaxy M12";
const char* password = "12345678";
const char* host = "api.thingspeak.com"
String apiKey = "7VCLHCAC95RNUPID";

const int PIR_PIN = 5;
const int TRIGGER_PIN = 14;  // Pin for
const int ECHO_PIN = 12;     // Pin for

DHT dht(DHTPIN, DHTTYPE);

WiFiClient client;

void setup() {
  Serial.begin(115200);

  pinMode(PIR_PIN, INPUT);
  pinMode(TRIGGER_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);

  dht.begin();

  WiFi.begin(ssid, password);

  Serial.println();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
```

Select Other Board and Port

Select both a Board and a Port if you want to upload a sketch.
If you only select a Board you will be able to compile, but not to upload your sketch.

BOARDS
PORTS
Search board
4D Systems gen4 IoD Range
Adafruit Circuit Playground
Adafruit Circuit Playground Express
Adafruit Feather HUZZAH ESP8266
Amperka WiFi Slot
Analog ADI
COM4 Serial Port (USB)
□ Show all ports
CANCEL
OK
Ln 1, Col 1   UTF-8   NodeMCU 1.0 (ESP-12E Module) [not connected]

STEP 5: Including the required Libraries

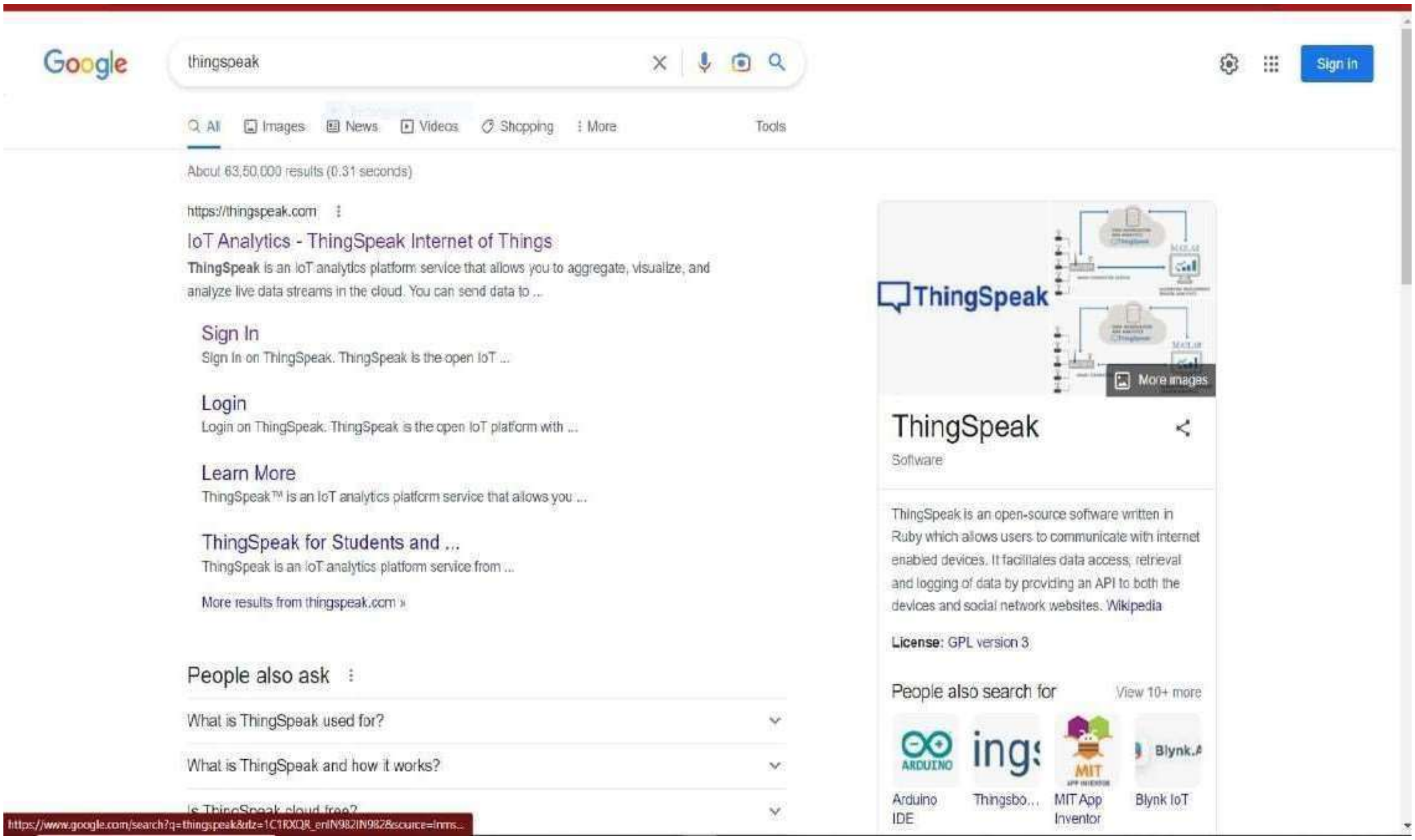

NodeMCU 1.0 (ESP-12E...
sketch_feb20a final 2.0.ino
Output
Ln 1, Col 1 UTF-8 NodeMCU 1.0 (ESP-12E Module) on COM4

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

#define DHTPIN 0    // Pin where the DHT11 is connected
#define DHTTYPE DHT11   // DHT 11

const char* ssid = "Galaxy M12";
const char* password = "12345678";
const char* host = "api.thingspeak.com";
String apiKey = "7VCLHCAC95RNUPTD";

const int PIR_PIN = 5;
const int TRIGGER_PIN = 14;  // Pin for the HC-SR04 trigger
const int ECHO_PIN = 12;     // Pin for the HC-SR04 echo

DHT dht(DHTPIN, DHTTYPE);

WiFiClient client;

void setup() {
  Serial.begin(115200);

  pinMode(PIR_PIN, INPUT);
```

Compiling sketch...

R

Step 6: Creating a API for Demonstration:



About 63,50,000 results (0.31 seconds)

https://thingspeak.com

IoT Analytics - ThingSpeak Internet of Things

ThingSpeak is an IoT analytics platform service that allows you to aggregate, visualize, and analyze live data streams in the cloud. You can send data to ...

Sign In

Sign In on ThingSpeak. ThingSpeak is the open IoT ...

Login

Login on ThingSpeak. ThingSpeak is the open IoT platform with ...

Learn More

ThingSpeak™ is an IoT analytics platform service that allows you ...

ThingSpeak for Students and ...

ThingSpeak is an IoT analytics platform service from ...

More results from thingspeak.com »

People also ask

What is ThingSpeak used for?

What is ThingSpeak and how it works?

Is ThingSpeak cloud free?

https://www.google.com/search?q=thingspeak&rlz=1C1RXQR_enIN982IN982&source=lnms...

ThingSpeak

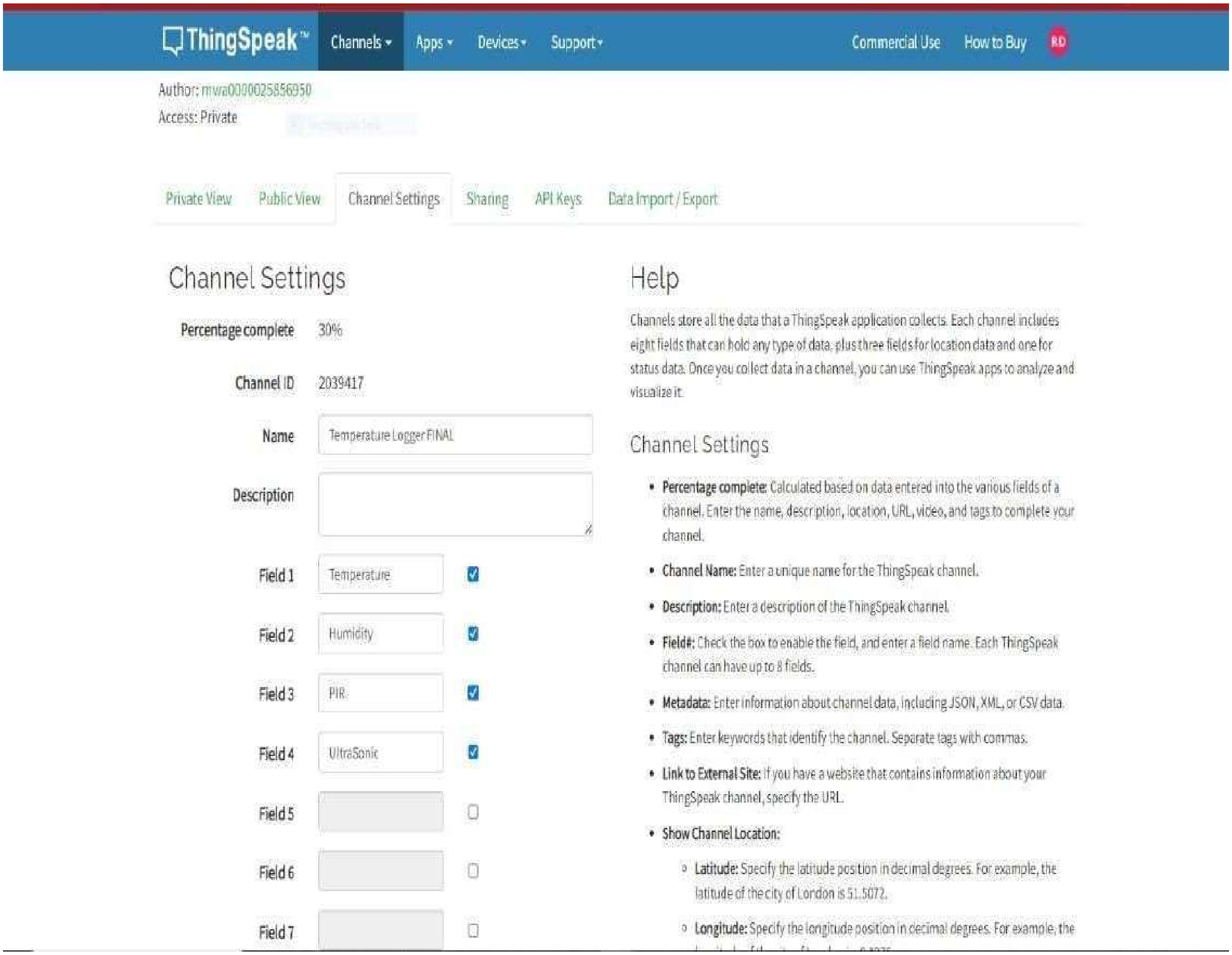
Software

ThingSpeak is an open-source software written in Ruby which allows users to communicate with internet enabled devices. It facilitates data access, retrieval and logging of data by providing an API to both the devices and social network websites. Wikipedia

License: GPL version 3

People also search for

View 10+ more



Author: mwa0000025856950

Access: Private

Channel Settings

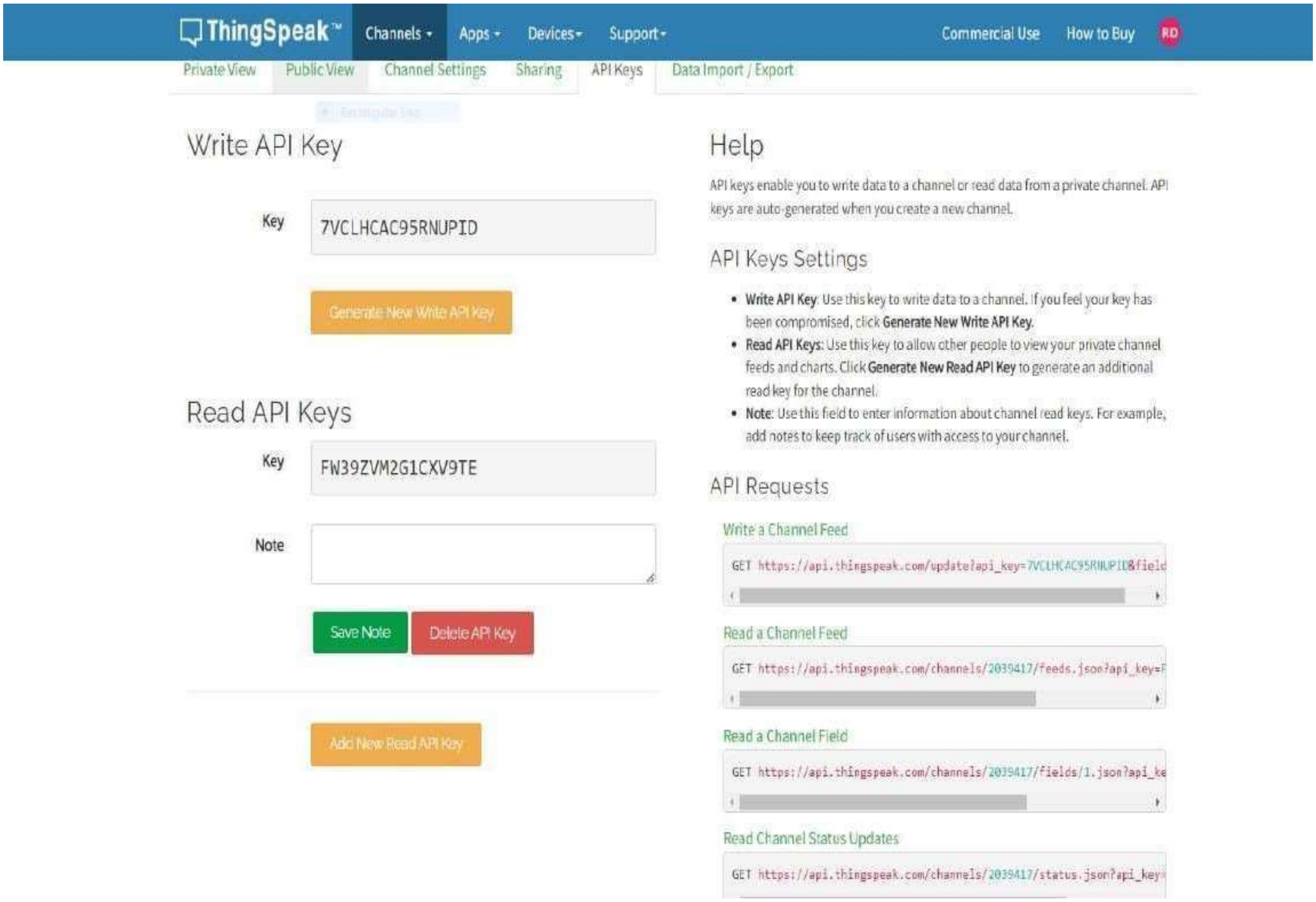
Percentage complete	30%	
Channel ID	2039417	
Name	Temperature Logger FINAL	
Description		
Field 1	Temperature	☑
Field 2	Humidity	☑
Field 3	PIR	☑
Field 4	UltraSonic	☑
Field 5		□
Field 6		□
Field 7		□

Help

Channels store all the data that a ThingSpeak application collects. Each channel includes eight fields that can hold any type of data, plus three fields for location data and one for status data. Once you collect data in a channel, you can use ThingSpeak apps to analyze and visualize it.

Channel Settings

- **Percentage complete:** Calculated based on data entered into the various fields of a channel. Enter the name, description, location, URL, video, and tags to complete your channel.
- **Channel Name:** Enter a unique name for the ThingSpeak channel.
- **Description:** Enter a description of the ThingSpeak channel.
- **Field#:** Check the box to enable the field, and enter a field name. Each ThingSpeak channel can have up to 8 fields.
- **Metadata:** Enter information about channel data, including JSON, XML, or CSV data.
- **Tags:** Enter keywords that identify the channel. Separate tags with commas.
- **Link to External Site:** If you have a website that contains information about your ThingSpeak channel, specify the URL.
- **Show Channel Location:**
  - **Latitude:** Specify the latitude position in decimal degrees. For example, the latitude of the city of London is 51.5072.
  - **Longitude:** Specify the longitude position in decimal degrees. For example, the



Write API Key

Key: 7VCLHCAC95RNUPID

Read API Keys

Key: FW39ZVM2G1CXV9TE

Note:

Help

API keys enable you to write data to a channel or read data from a private channel. API keys are auto-generated when you create a new channel.

API Keys Settings

- **Write API Key**: Use this key to write data to a channel. If you feel your key has been compromised, click **Generate New Write API Key**.
- **Read API Keys**: Use this key to allow other people to view your private channel feeds and charts. Click **Generate New Read API Key** to generate an additional read key for the channel.
- **Note**: Use this field to enter information about channel read keys. For example, add notes to keep track of users with access to your channel.

API Requests

Write a Channel Feed

```
GET https://api.thingspeak.com/update?api_key=7VCLHCAC95RNUPID&field
```

Read a Channel Feed

```
GET https://api.thingspeak.com/channels/2039417/feeds.json?api_key=F
```

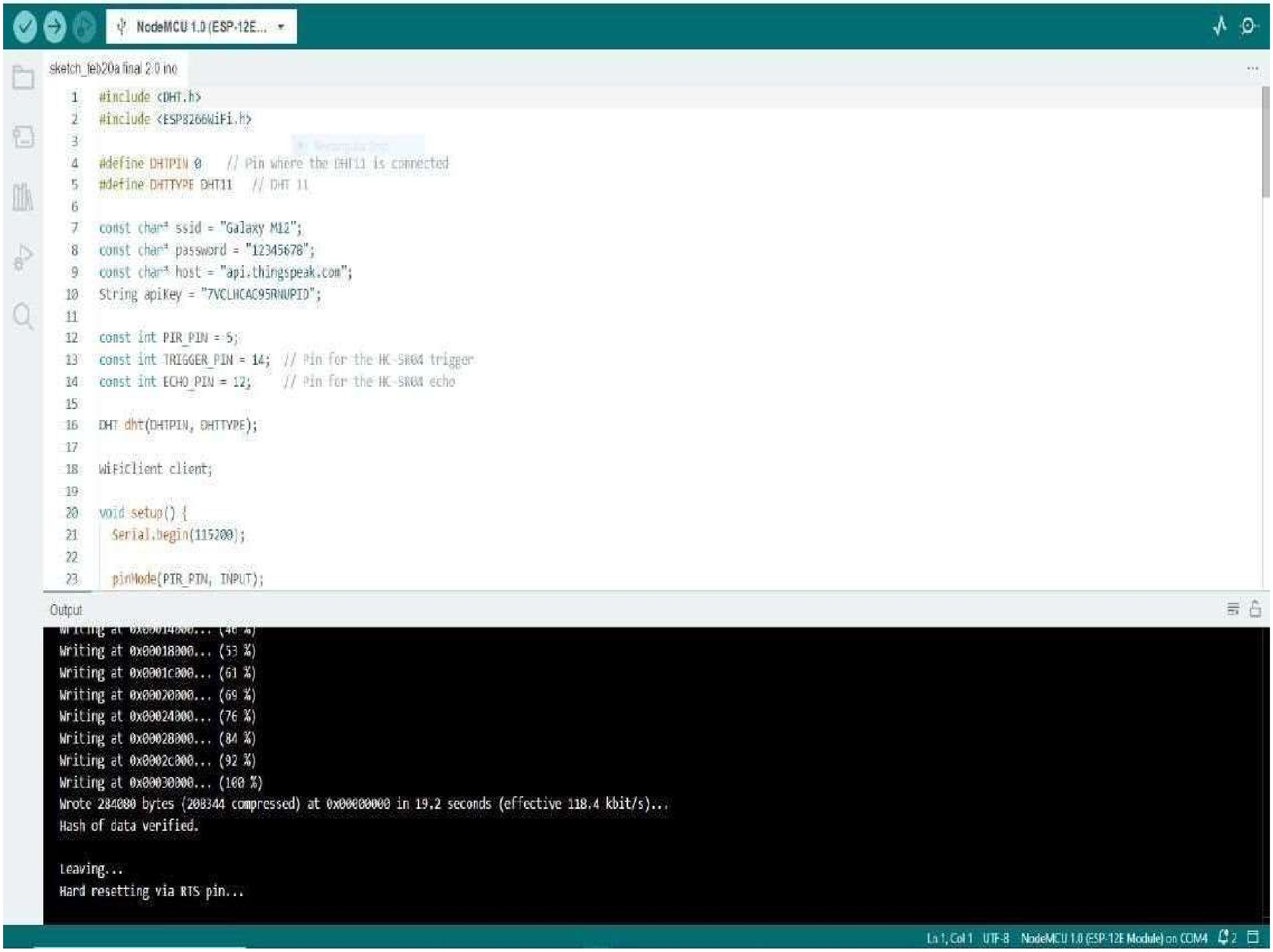
Read a Channel Field

```
GET https://api.thingspeak.com/channels/2039417/fields/1.json?api_ke
```

Read Channel Status Updates

```
GET https://api.thingspeak.com/channels/2039417/status.json?api_key=
```

Step 7 :- Running the Uploading the code in the hardware Device

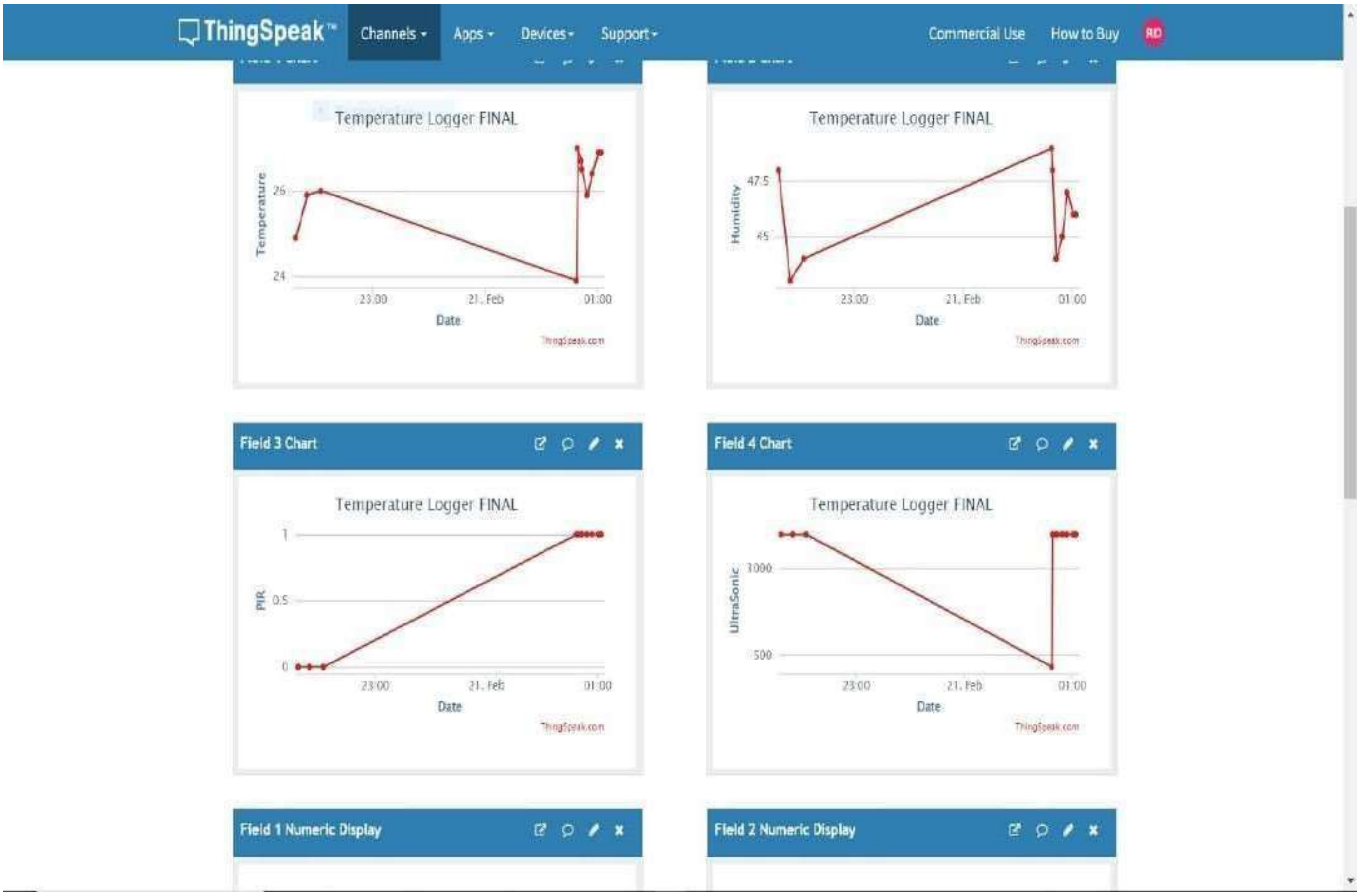


NodeMCU 1.0 (ESP-12E...
sketch_feb20a final 2.0.ino
Output
Ln 1, Col 1   UTF-8   NodeMCU 1.0 (ESP-12E Module) on COM4   2

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

#define DHTPIN 0    // Pin where the DHT11 is connected
#define DHTTYPE DHT11   // DHT 11

const char* ssid = "Galaxy M12";
const char* password = "12345678";
const char* host = "api.thingspeak.com";
String apiKey = "7VCLHCAC95RNUPID";

const int PIR_PIN = 5;
const int TRIGGER_PIN = 14;  // Pin for the HC-SR04 trigger
const int ECHO_PIN = 12;     // Pin for the HC-SR04 echo

DHT dht(DHTPIN, DHTTYPE);

WiFiClient client;

void setup() {
  Serial.begin(115200);

  pinMode(PIR_PIN, INPUT);
```

```
Writing at 0x00014000... (46 %)
Writing at 0x00018000... (53 %)
Writing at 0x0001c000... (61 %)
Writing at 0x00020000... (69 %)
Writing at 0x00024000... (76 %)
Writing at 0x00028000... (84 %)
Writing at 0x0002c000... (92 %)
Writing at 0x00030000... (100 %)
Wrote 284080 bytes (208344 compressed) at 0x00000000 in 19.2 seconds (effective 118.4 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...
```

STEP 8 :- Demonstration



ALGORITHMIC TREE FOR THE CODE

```
setup()
├─ Serial.begin(115200)
├─ pinMode(PIR_PIN, INPUT)
├─ pinMode(TRIGGER_PIN, OUTPUT)
├─ pinMode(ECHO_PIN, INPUT)
├─ dht.begin()
├─ WiFi.begin(ssid, password)
├─ while (WiFi.status() != WL_CONNECTED)
│   └─ delay(500)
└─ client.connect(host, 80)
   └─ post data to ThingSpeak

loop()
├─ float h = dht.readHumidity()
├─ float t = dht.readTemperature()
├─ float duration, distance
├─ digitalWrite(TRIGGER_PIN, LOW)
├─ delayMicroseconds(2)
├─ digitalWrite(TRIGGER_PIN, HIGH)
├─ delayMicroseconds(10)
├─ digitalWrite(TRIGGER_PIN, LOW)
├─ duration = pulseIn(ECHO_PIN, HIGH)
├─ distance = duration * 0.034 / 2
├─ if (isnan(h) || isnan(t))
│   └─ print error message
├─ else
│   ├─ construct HTTP POST request with data
│   └─ client.connect(host, 80)
│       └─ post data to ThingSpeak
└─ delay(15000)
```

FUTURE SCOPE OF FACE RECOGNITION

The future applications and uses of temperature loggers and motion sensing systems are vast, as these technologies continue to be improved and integrated with other emerging technologies. Here are some potential future applications and uses:

- Smart Homes: Temperature loggers and motion sensing systems can be used in smart home systems to monitor and control heating and cooling, as well as detect and respond to motion.

- Agriculture: Temperature loggers can be used to monitor and optimize crop growth and storage conditions, while motion sensing systems can be used for animal monitoring and crop surveillance.

- Healthcare: Temperature loggers and motion sensing systems can be used to monitor patients in real-time, detecting changes in body temperature and movement to detect potential health issues.

- Automotive Industry: Temperature loggers can be used to monitor the performance of car engines, while motion sensing systems can be used for collision avoidance and navigation in self-driving cars.

- Security: Motion sensing systems can be used for security and surveillance purposes in various settings, such as airports, public transportation systems, and buildings.

- Robotics: Temperature loggers and motion sensing systems can be used in robotics applications to optimize performance and monitor robot behavior.

- Industrial Applications: Temperature loggers and motion sensing systems can be used in industrial settings to monitor and optimize manufacturing processes, detect faults in machinery, and improve worker safety.

Overall, temperature loggers and motion sensing systems have numerous potential future applications and uses, and as these technologies continue to advance and become more integrated with other emerging technologies, their potential applications will likely expand even further.

CONCLUSION

- Temperature loggers and motion sensing systems are important technologies that have a wide range of applications in various industries. Temperature loggers are used to record temperature data over time, and can provide valuable insights into the performance of systems or equipment.

- Motion sensing systems can detect movement or changes in motion, and can be used for security, navigation, and automation purposes.There has been significant previous work in the development of temperature loggers and motion sensing systems, both commercially and through research studies. Commercially available temperature loggers range from simple data loggers to more complex systems with wireless communication capabilities. Motion sensing systems range from simple motion detectors to more advanced systems that use computer vision and machine learning algorithms.

- Researchers have also developed temperature loggers and motion sensing systems for specific applications, such as wireless temperature and humidity monitoring for food storage facilities, and healthcare applications such as fall detection and activity monitoring for elderly individuals.

- Overall, temperature loggers and motion sensing systems are important technologies that continue to be actively researched and developed for a wide range of applications. These technologies have the potential to improve efficiency, safety, and security in various industries, and can provide valuable insights into systems and processes.