**Intelligent weather monitoring system powered by IOT**

Mrs.Shuchita Mudgil Shri.Rohan Rajoriya

Lecturer IT Sr.Lecturer IT

Kalaniketan Polytechnic College Jabalpur Kalaniketan Polytechnic College Jabalpur

**Abstract :-**

The system suggested in this document is a cutting-edge way to keep track of local meteorological conditions and make the data accessible from anywhere in the world. The Internet of Things (IoT), a cutting-edge and effective method of integrating the entire world of objects into a single network, is the technology that underpins it. Here, you may find items like sensors, electronic gadgets, and electronic devices for autos. The system uses sensors to monitor and regulate environmental factors including temperature and relative humidity. It then sends the information to a website where it is recorded as graphical statistics. Anywhere in the globe can access the data updated by the deployed system over the Internet.

Keywords- Internet of Things (IoT); Arduino Cloud, ESP8266

**INTRODUCTION**

The weather monitoring system is a practical implementation of the Internet of Things (IoT) and cloud technologies. It involves collecting and recording various weather parameters and using them for alerts, notifications, setting devices, and long-term analysis. Devices are used to collect, organize and display information, and the Internet of Things is expected to change the world by monitoring and controlling environmental phenomena.

The system consists of components such as the ESP8266 wifi module board, microcontroller board, DHT11 sensor, WIFI module, and web page to monitor weather conditions from any remote location.

**Litreature Survey :**

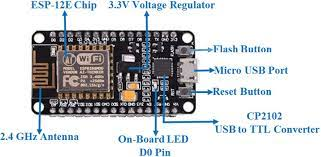
IOT has grown to be a popular subject of interest for academic institutions, major tech firms, and undoubtedly users or customers as well. Many IOT-based concepts, such as smart wearable technology, smart homes, and smart cities, have attracted a lot of attention. Nearly all Internet of Things applications use transducers and sensors attached to microcontrollers with wireless or wired data transmission to a remote cloud service or local data storage that transforms the raw data into significant information that can be used in a variety of other applications. As we worked on this project, we came across some examples of successful smart application development employing inexpensive Arduino or Raspberry Pi devices.These boards were used to create the majority of applications, including smart city and other automation projects. "Places can be equipped with sensors and monitor environmental conditions, cyclists or athletes can find the most "healthy" trips, and the city can respond by adjusting traffic orby planting more trees in some areas," according to [1]. All citizens will have access to the data, encouraging the development of applications that provide inhabitants with real-time information. Therefore, it is safe to conclude that this weather monitoring system will also be useful in some smart city initiatives. The authors of [2][3] used a single sensor, a composite DHT11 sensor, to measure both humidity and temperature.In the past, persons who worked from home or were occupied with household duties were unaware of the environmental conditions outside their place of residence or employment. People are unaware of the outside temperature, its level of humidity, whether it is raining or not, or whether it is quite low, high, or normal. In [4], a particle photon, an IoT board that is compatible with Arduino, was used to construct a weather monitoring system. It will also send us wishes in the morning, evening, and at night because it is equipped with a light sensor.The fact that "by deploying sensor devices in the environment, we can bring the environment into real life" was highlighted by the authors in [5] is fantastic.

**Design and Implementation:**

Using the MQTT protocol, the nodemcu board, which serves as a client in this system, publishes the sensor data into the topic Arduino Cloud/DEVICE ID/state in the Arduino Cloud is a Message Broker. The published data will be instantly stored in the Arduino cloud and made accessible in the cloud platform's visualisation tool. The client must subscribe to the commands started using cloud workflow on the subject "Arduino Cloud /DEVICE ID/commands" in order to receive the stored data from the cloud MQTT broker.

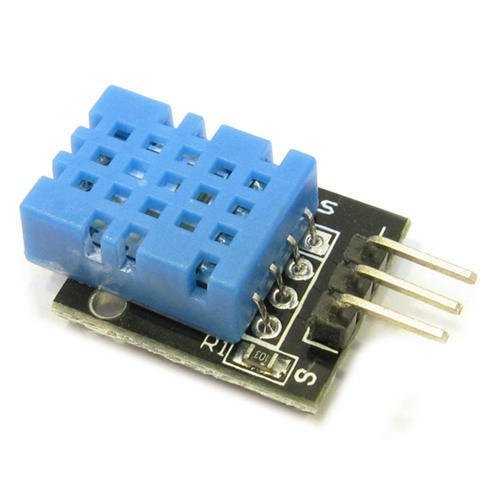
1. **NodeMCU**

The Arduino Uno WiFi module that we utilised is built within the Arduino Uno that we used.The board's ATmega328P processor and ESP8266 WiFi module combine with a TCP/IP protocol stack.The microcontroller must utilise a few AT commands in order to connect to the ESP8266 WiFi module and begin communicating.



1. **DHT-11 Sensor :**

DHT-11 is a digital temperature and humidity sensor with analogue input pins.



1. **Arduino IDE:**

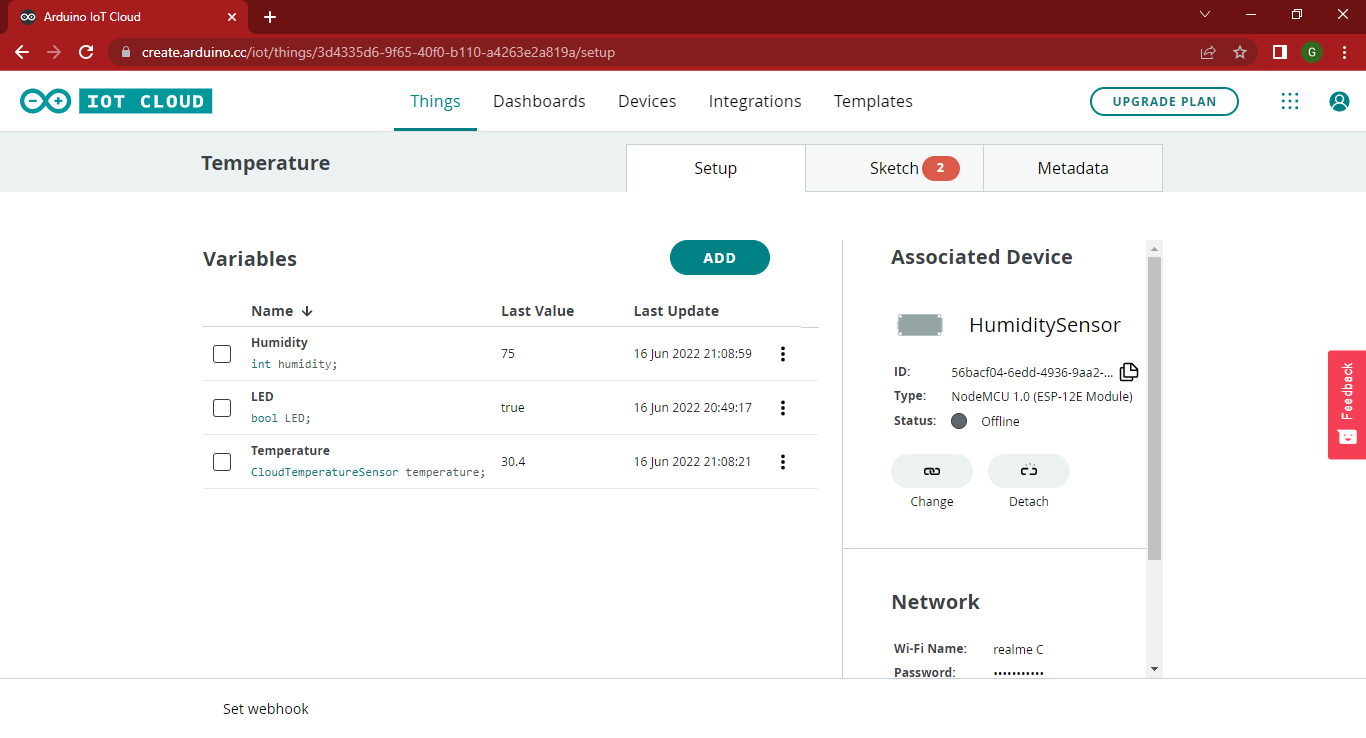
The Arduino IDE was the primary programme we used to programme our microcontroller board. It was created using C and C++.It offers a built-in library with several input and output options. Additionally, the programme is open source.The user code is flashed to Arduino code using the avrdude uploading tool by default.Sketches are the name for the programmes created in Arduino IDE. These are created in an editor and saved with the.ino file extension. It has a console where output, including error warnings and other data, is shown. This IDE supports the C and C++ programming languages by arranging the code according to some unique criteria.

**Steps and Procedure for performing the Project**

The "Arduino IOT Cloud platform" is the foundation of the entire project. Some of the main attributes of this IOT platform are listed below:

Things ,Dashboards, hardware, software, integrations, and templates

1. Launch the browser, type ARDUINO IOT CLOUD into the search bar, and navigate to [www.create.arduino.cc](http://www.create.arduino.cc)



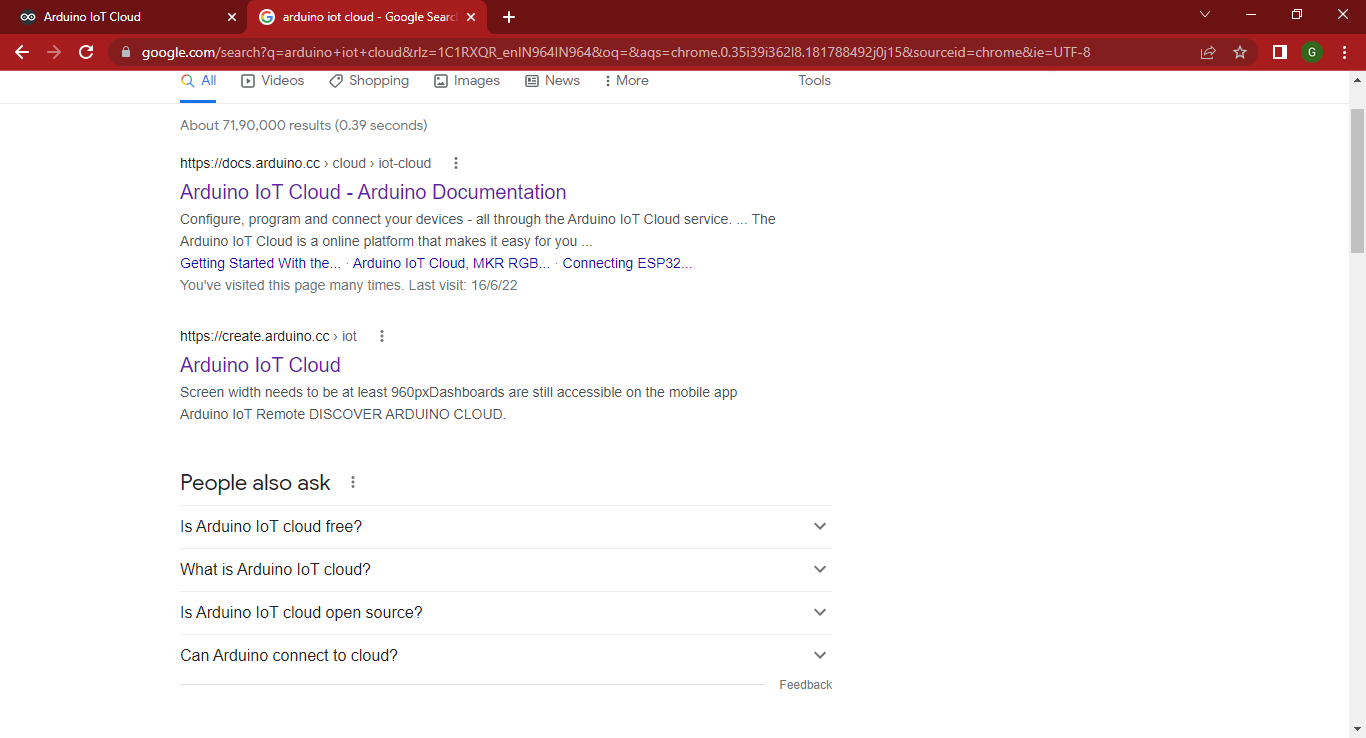


Fig : Snapshot Of Arduino Cloud

1. Each user can establish or sign up for their own account to utilise the capabilities of this IOT cloud platform, and the website holds all of their private information, including their Gmail address, password, and usernames. Users of "Create.arduino.cc" have a variety of alternatives for setting up an account. Option 1: Users can join up by selecting the CREATE ONE option, where they can enter all of their personal data, including Full Name, Date of Birth, Location, and so on. Option 2: Direct registration by choosing the Gmail option (no personal information is required).

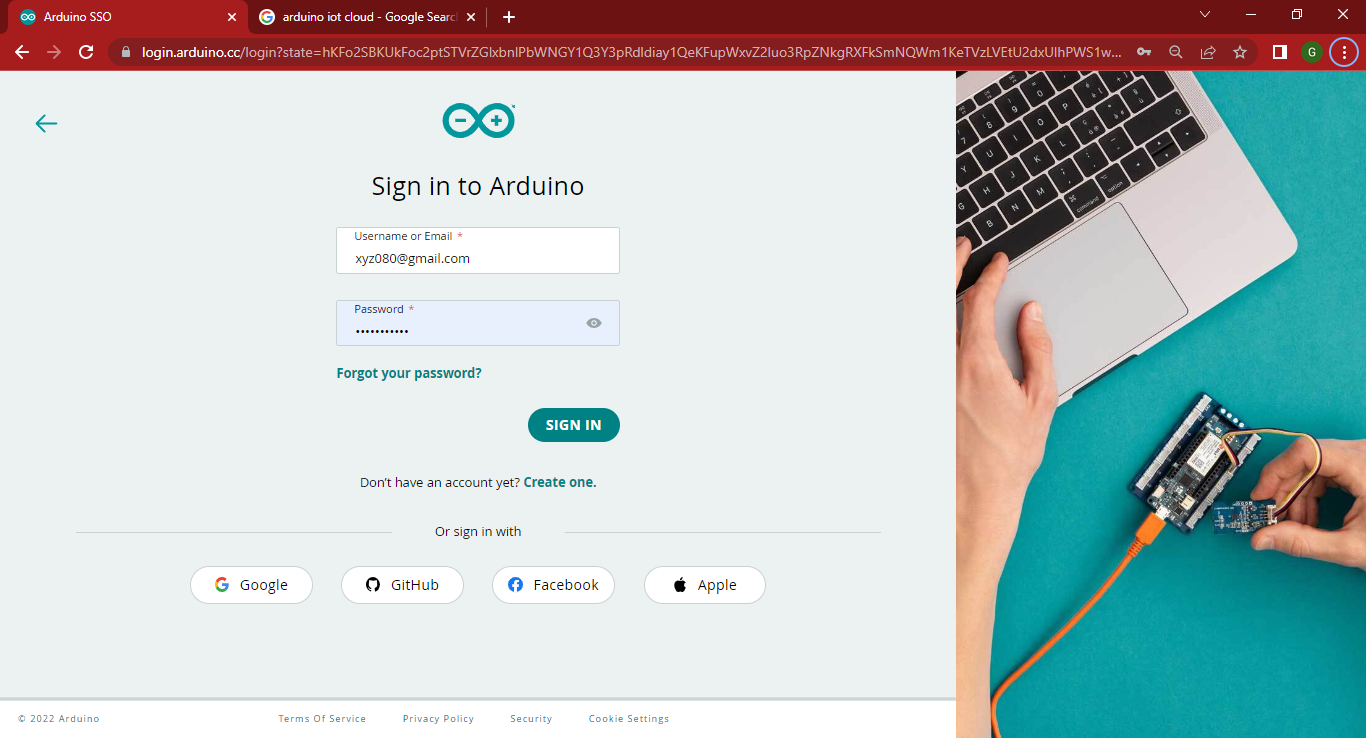
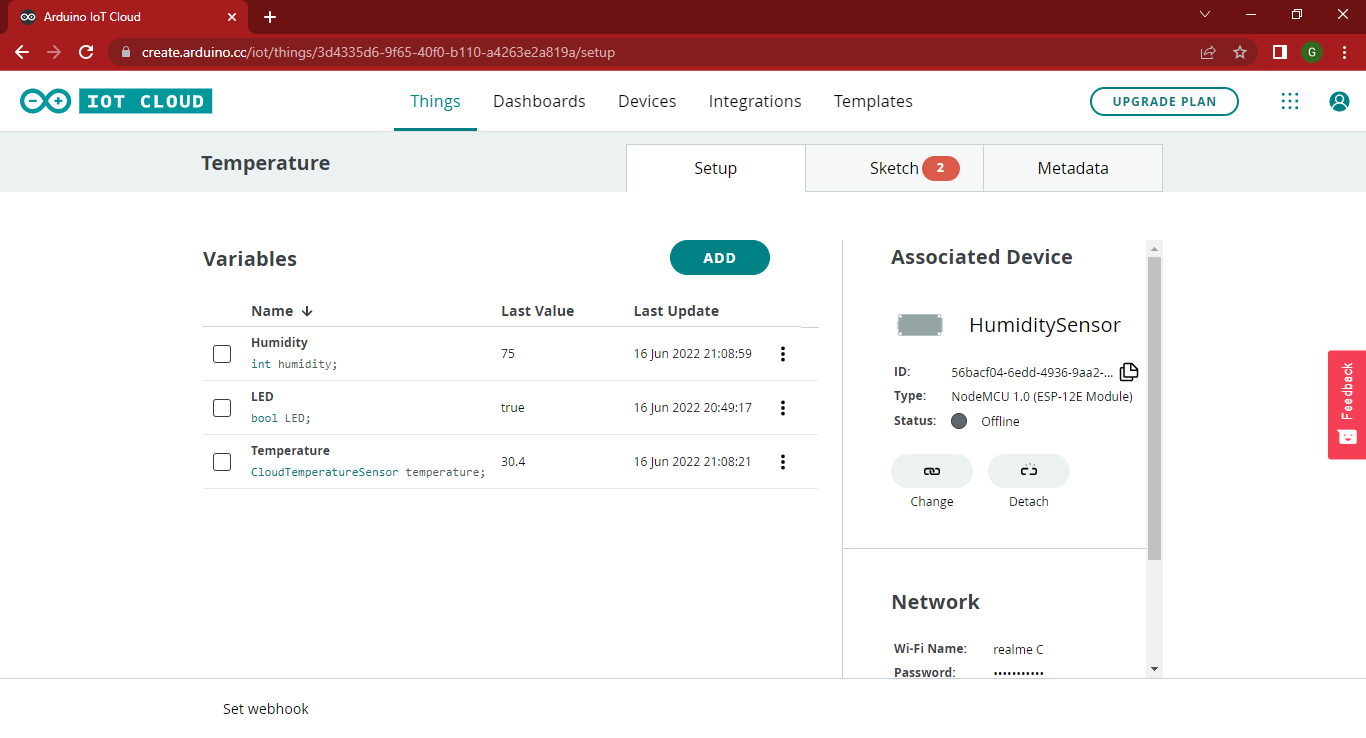


Fig : Signing in Arduino Cloud

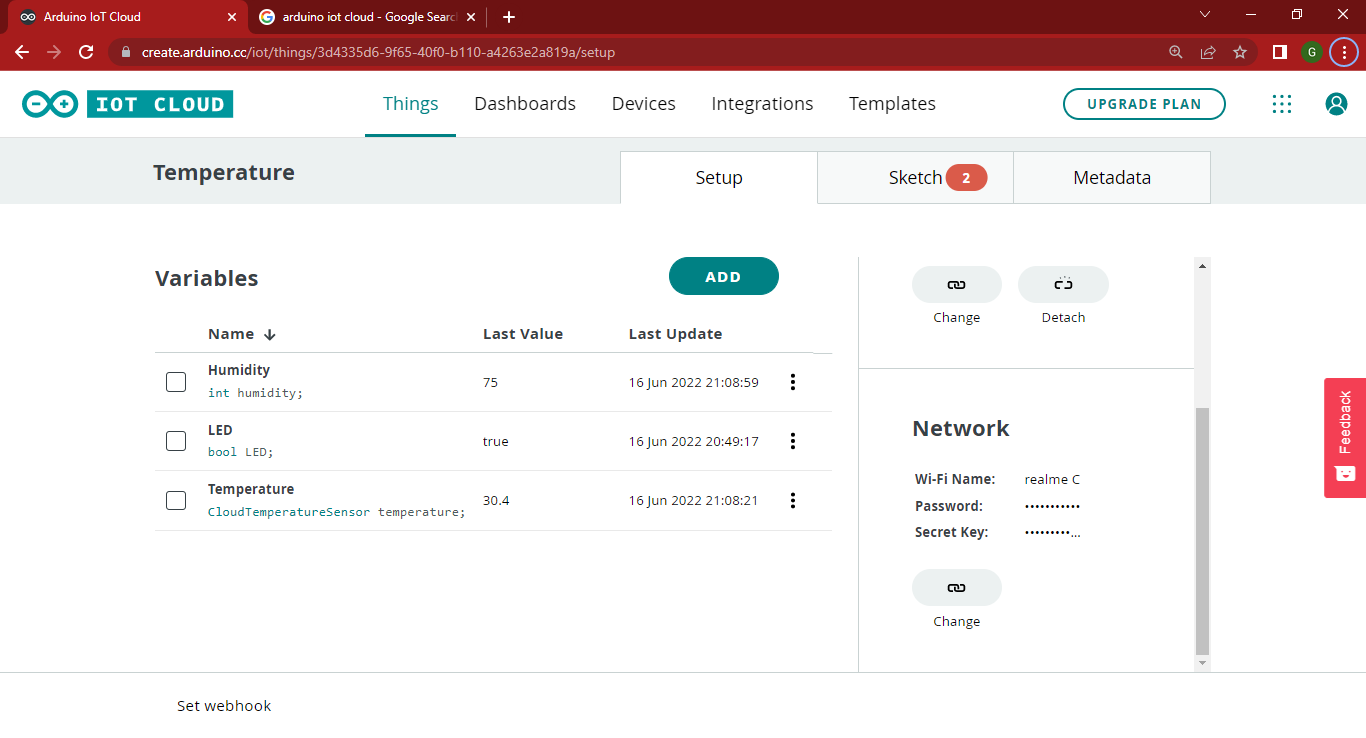
1. **Develop "THINGS"**

This option includes all the necessary data for running the project, such as humidity, LED, temperature, and network connectivity for pairing devices with one another to access the device through Bluetooth and WiFi.



*SELECT* -> Humidity – int humidity (integer type), LED – boolean type, and the Temperature – CloudTemperatureSensor.

1. By entering the correct SSID and your WiFi password, you can pair your device.



1. Develop "DASHBOARDS"

The Dashboard essentially provides a U.I (user interface) that allows users to connect with the IOT platform and is in charge of managing the overall Layout of the project. The following is a list of the widgets or parts that must be included in the dashboard for the device:

* LED Switch
* Temperature Gauge
* Humidity Value
* Temperature Chart

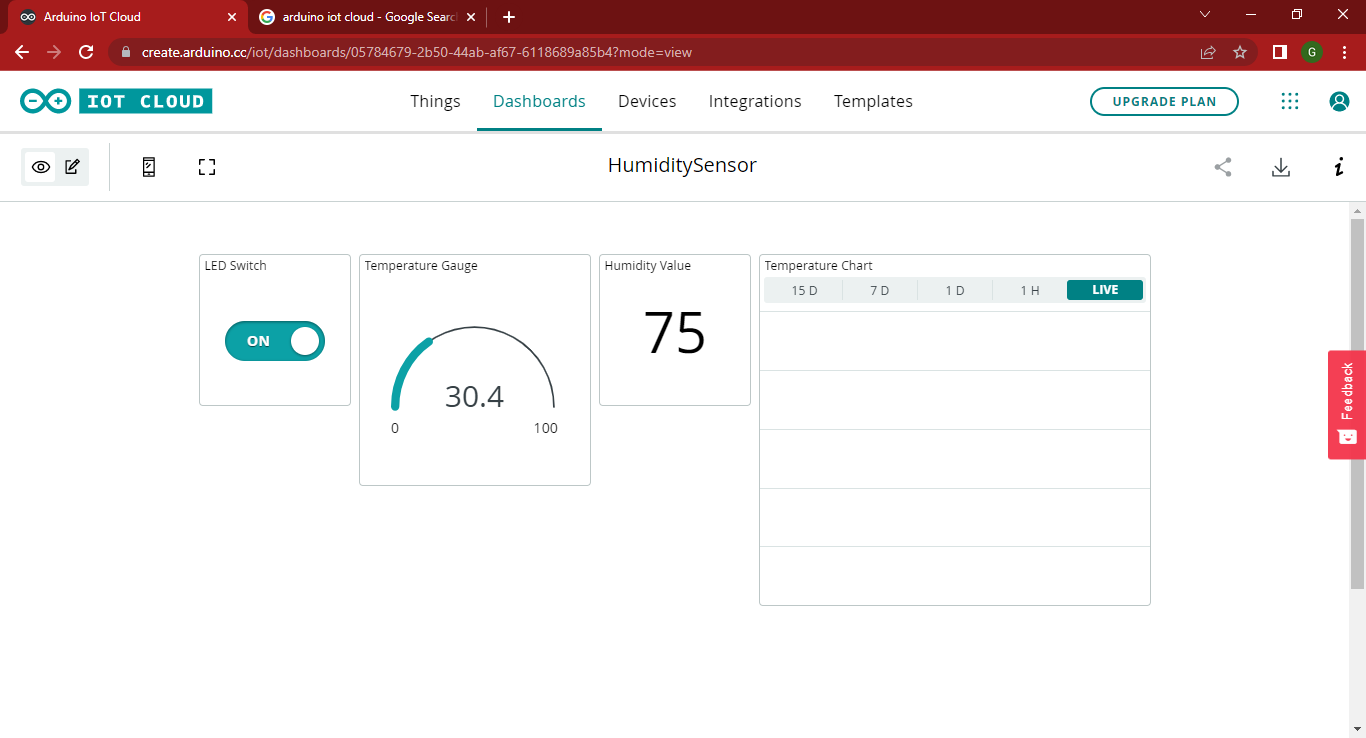
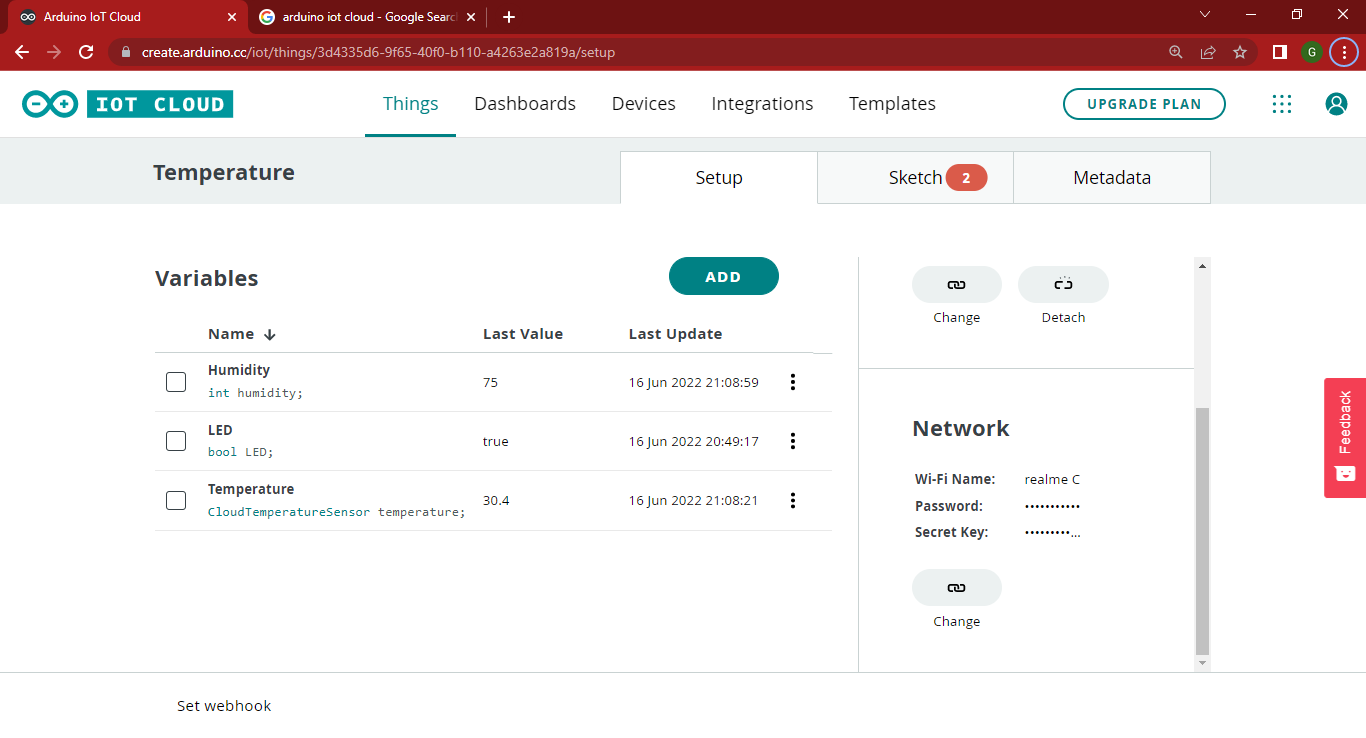
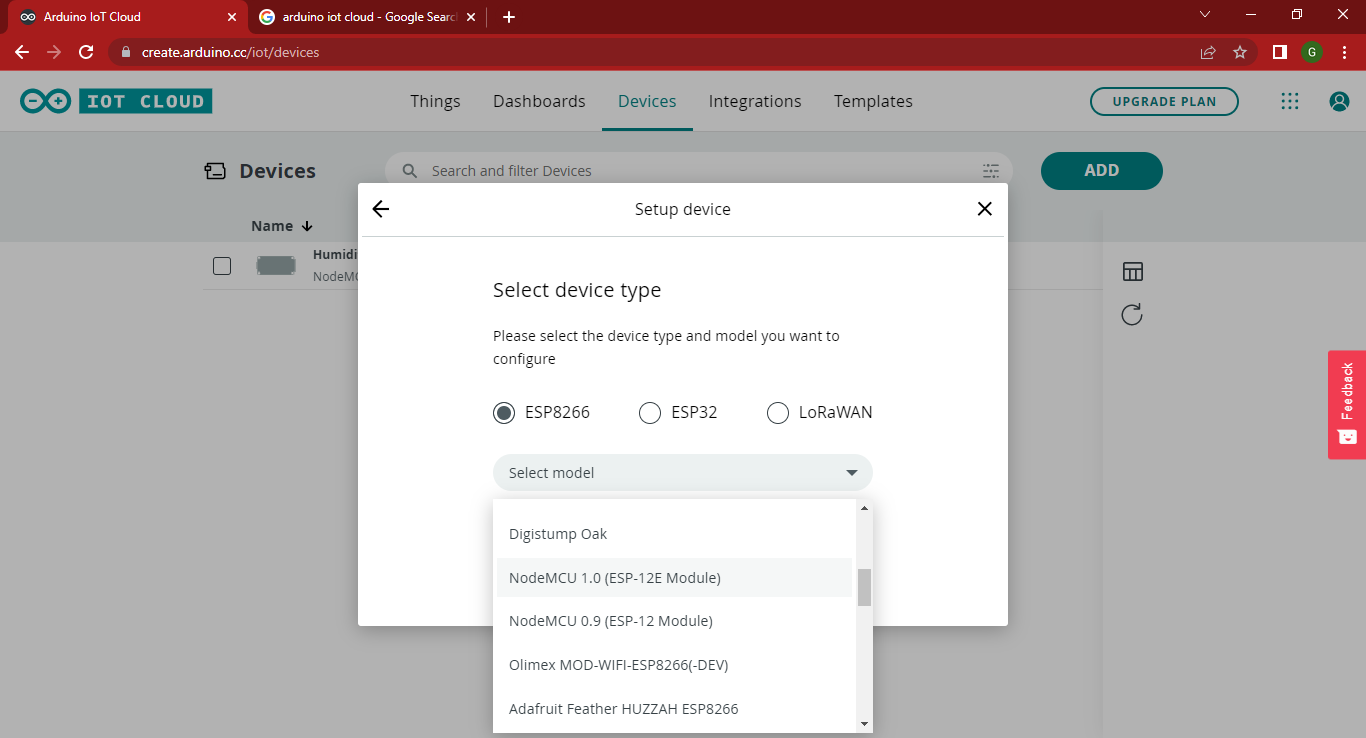


Fig : Dashboard Creation

1. Develop "DEVICES"

This option includes all of the device (Hardware) configurations and data for the Esp8266 - Node MCU in the project mentioned above.

* Select “Set up a third party device”
* Search for Node MCU (1.0) module
* After completing the above steps successfully a *Secret-Key* will be generated and providing the right key is mandatory for pairing the device to access the Wifi and Bluetooth connectivity.



**Note :** The secret Key must be accurate.

1. Produce a "SKETCH" and submit the CODES

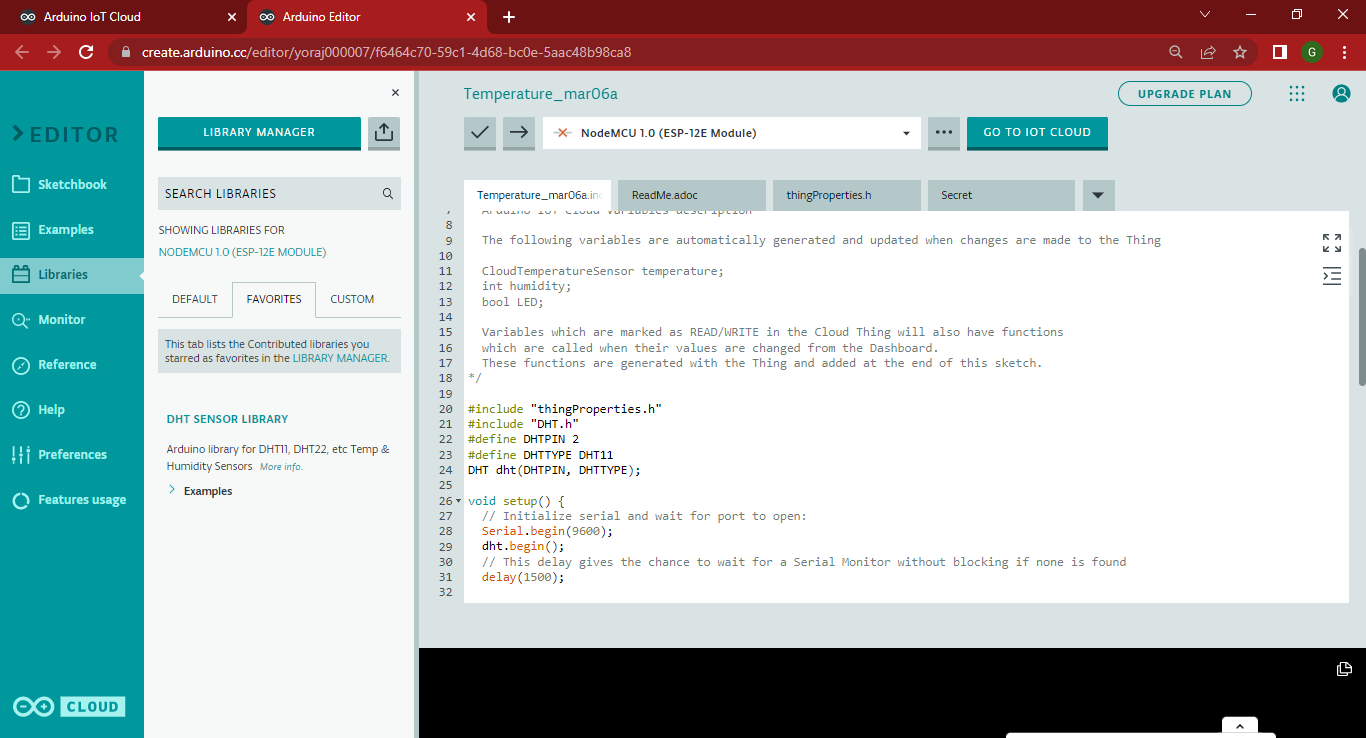
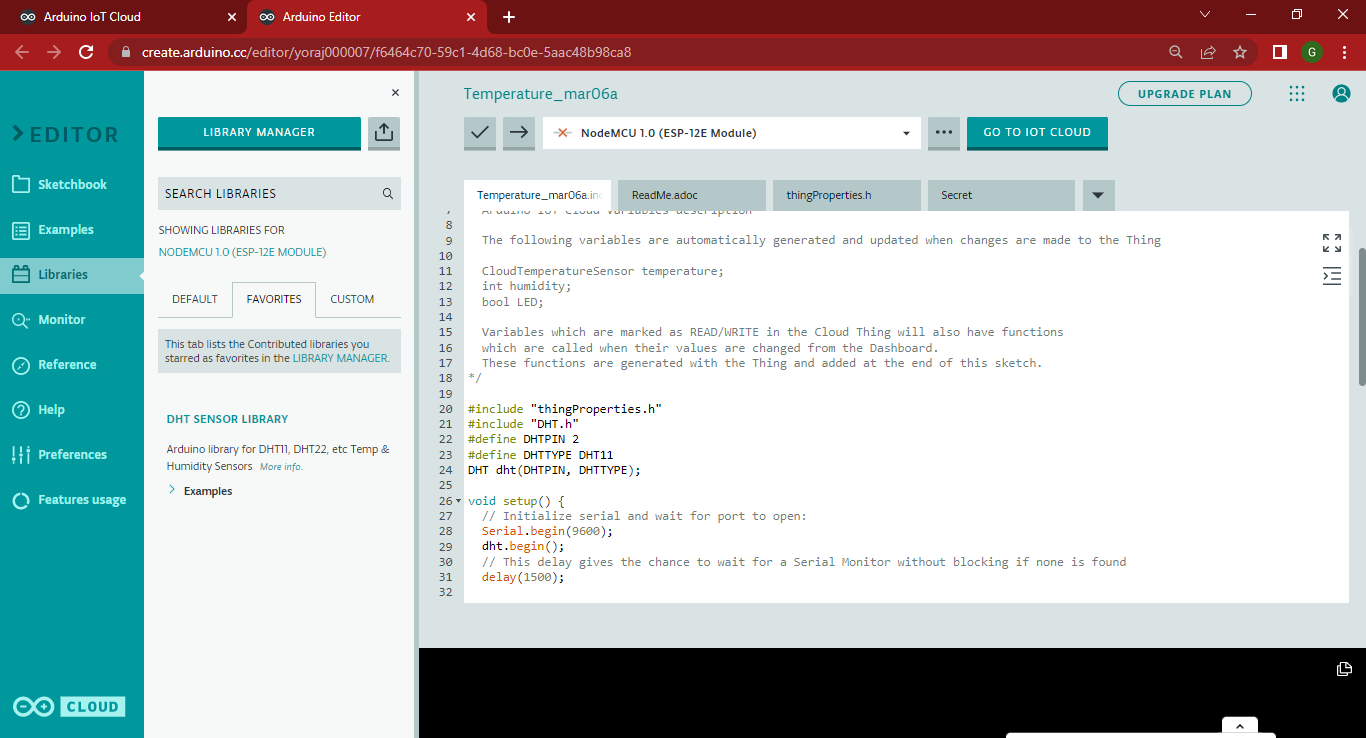
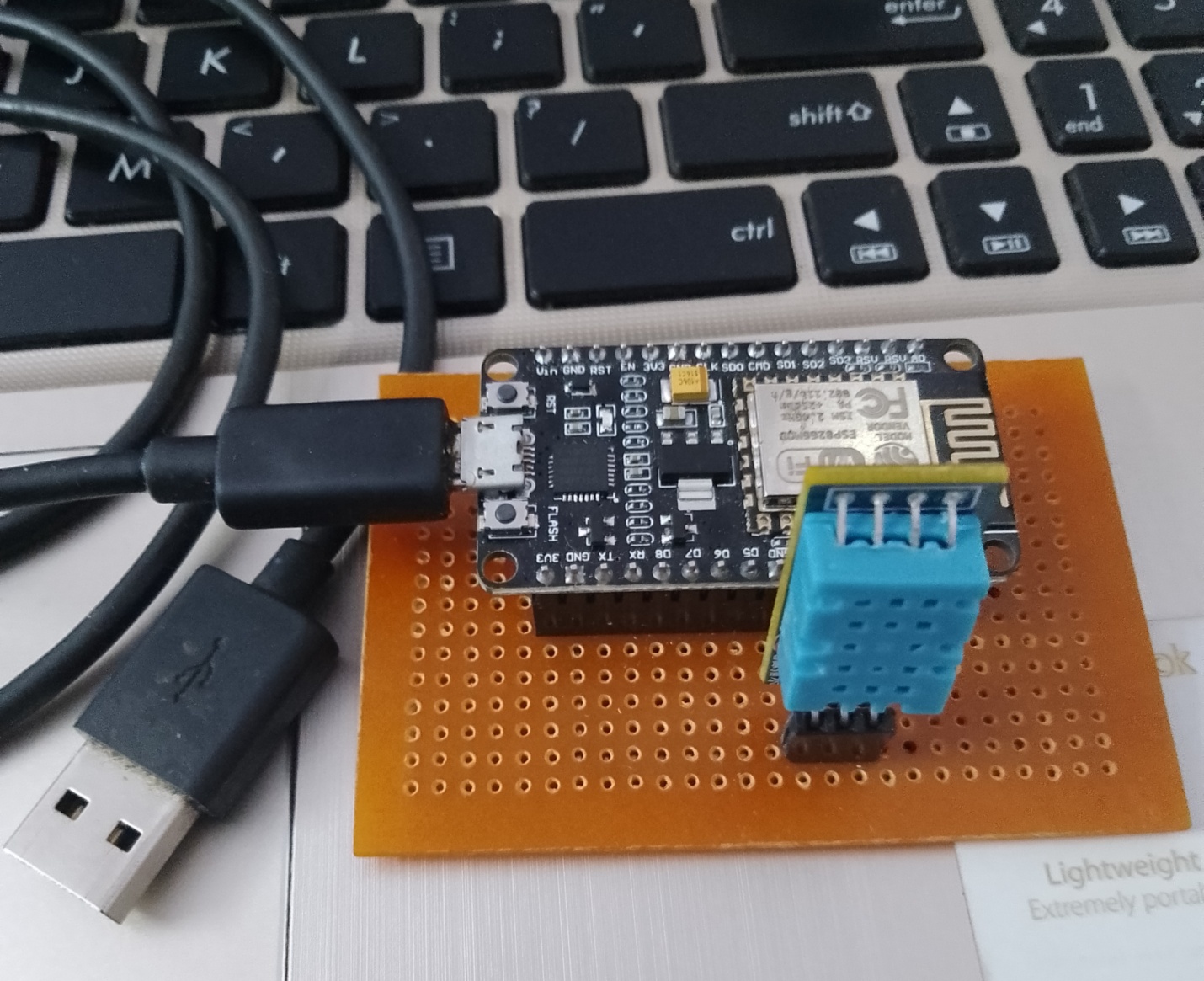
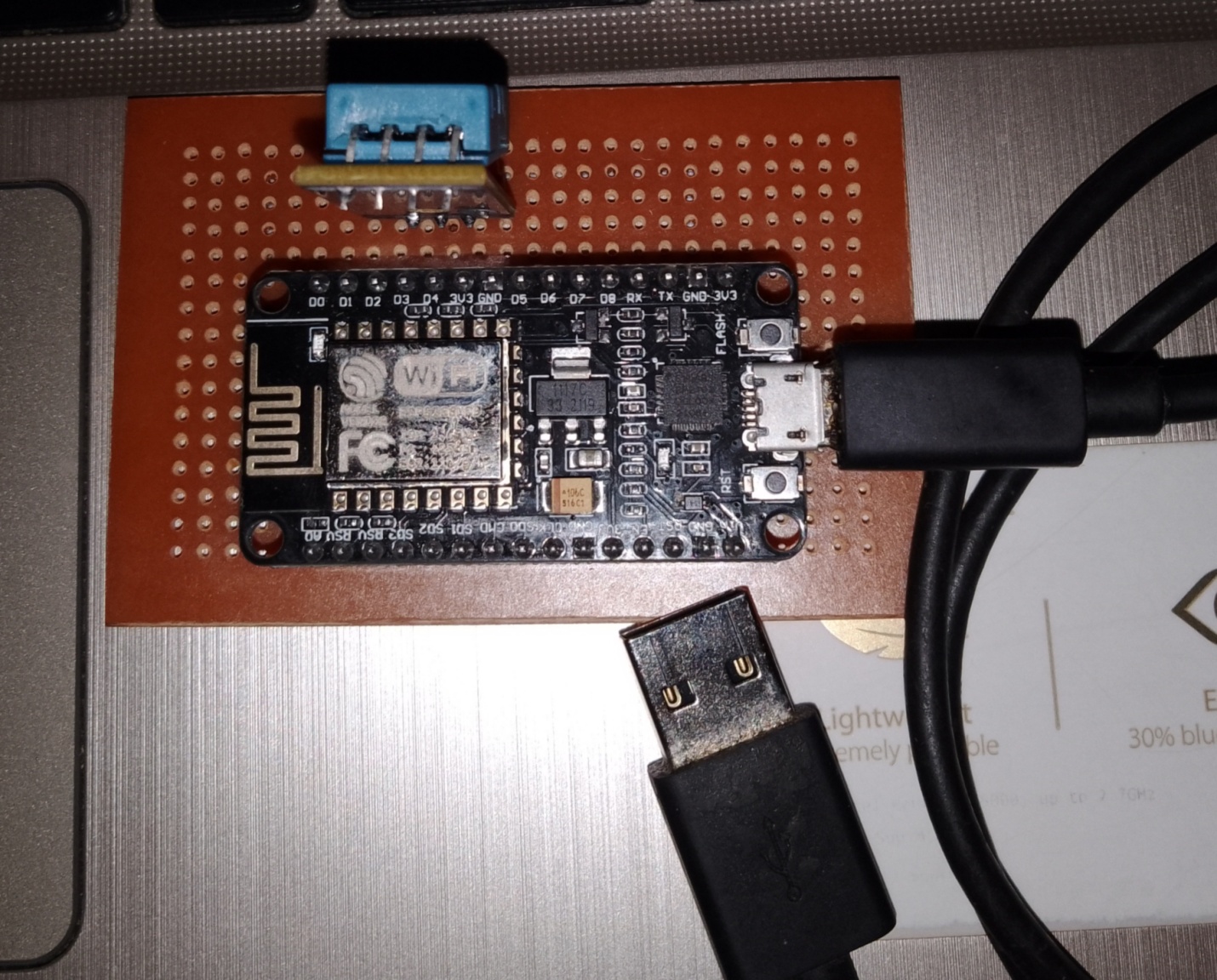
The programme "Sketch" controls all the logical and programming elements necessary to operate the device and get readings for the temperature and humidity of a certain area. It is the brain of the entire project. preserving Bluetooth and wireless connectivity for the gadget. The sketch serves as a vehicle for all the code and programming required to operate the device. 

Fig: Sketch Code

1. To record the humidity, import the "DHT SENSOR LIBRARY" library from the library manager.





**Result :**

When sensor measurements are transferred to the cloud for analysis, the data is examined, and the system is then informed if a parameter's specific value deviates from the intended range. A graph is also created to show the trends.

**Future Scope:**

The proposed cloud-based and IoT-based weather monitoring system can be expanded to include a GPS device, more sensors, and the ability to send environmental parameters as a message or notification. It can also be used for other automation projects.

**References :**

[1] Hammi, B., Khatoun, R., Zeadally, S., Fayad, A., & Khoukhi, L. (2018). IoT technologies for smart cities. IET Networks, 7(1), 1-13. doi: 10.1049/iet-net.2017.0163

[2] S. Zafar, G. Miraj, R. Baloch, D. Murtaza, and K. Arshad, “An IoT Based Real-Time Environmental Monitoring System Using Arduino and Cloud Service”, Eng. Technol. Appl. Sci. Res., vol. 8, no. 4, pp. 3238-3242, Aug. 2018.

[3] S. D. Shewale, S. N. Gaikwad, “An IoT based real-time weather monitoring system using Raspberry Pi”, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering”, Vol. 6, No. 6, pp. 4242-4249, 2017

[4] K, H. (2020). IOT Based Weather Monitoring System Using Particle Photon. Retrieved 10 April 2020, from <https://www.engineersgarage.com/contributions/iot-based-weather-monitoring-systemusing-particle-photon/>

[5] Babu, R.Suresh & Thillainathan, Palaniappan & Anushya, K & Kowsalya, M & Krishnadevi, M. (2018). IoT Based Weather Monitoring System. [6] A Greenhouse Monitoring and Crop Prediction System Implemented using Iot, Arduino Uno and Nodemcu. (2020). International Journal Of Recent Technology And Engineering, 8(4S5), 1-4. doi: 10.35940/ijrte.d1001.1284s519.

[6] F. Meneghello, M. Calore, D. Zucchetto, M. Polese and A. Zanella, "IoT: Internet of Threats? A Survey of Practical Security Vulnerabilities in Real IoT Devices," in IEEE Internet of Things Journal, vol. 6, no. 5, pp. 8182-8201, Oct. 2019. doi: 10.1109/JIOT.2019.2935189

[7] N. Y. Parotkin and V. V. Zolotarev, "Information Security of IoT Wireless Segment," 2018 Global Smart Industry Conference (GloSIC), Chelyabinsk, 2018, pp. 1-7. doi: 10.1109/GloSIC.2018.8570144

[8] S. Siboni et al., "Security Testbed for Internet-of-Things Devices," in IEEE Transactions on Reliability, vol. 68, no. 1, pp. 23-44, March 2019. doi: 10.1109/TR.2018.2864536

[9] Y. Liu, Y. Kuang, Y. Xiao and G. Xu, "SDN-Based Data Transfer Security for Internet of Things," in IEEE Internet of Things Journal, vol. 5, no. 1, pp. 257-268, Feb. 2018 doi:10.1109/JIOT.2017.2779180

[10] P. Fremantle and P. Scott, “A survey of secure middleware for the Internet of Things,” PeerJ Computer Science, vol. 3, p. e114, May 2017