

Second Year B. Tech (EL & CE)

Semester: IV

Subject: Basic IoT Laboratory

Name:

Class:

Roll No:

Batch:

Experiment No: 05

Name of the Experiment: Exploring cloud infrastructure for connecting IoT devices and sending and visualizing sensor data to open source cloud via Arduino IDE.

Performed on:

Submitted on:

Marks	Teacher's Signature with date

Aim: Exploring cloud infrastructure for connecting IoT devices and sending and visualizing sensor data to open source cloud via Arduino IDE.

Prerequisite: Basics of NodeMCU Model, Cloud concepts.

Objective:

1. Understand DHT11/DHT22 sensor
2. Sensor interfacing with NodeMCU (DHT11/DHT22)
3. Display Temperature/Humidity values on serial Monitor
4. Connect to WiFi, display WiFi IP and Display Temperature/Humidity values on serial Monitor
5. Understand Thingier.io cloud platform
6. Display Temperature/Humidity values on cloud platform

Components and equipment required:

NodeMCU, DHT11/22 sensor, Breadboard, Connecting Wires etc.

Theory:

Temperature sensor: It is a device, a thermocouple or RTD, that provides temperature measurement through an electrical signal.

Thermocouple: It is made from two dissimilar metals that generate electrical voltage in direct proportion to changes in temperature. The wires are joined together to form measuring junction and reference junction.

RTD: Resistor temperature detection is variable resistor that will change its electrical resistance in direct proportion to changes in temperature in precise, repeatable & linear manner.

The DHT22 sensor is used to measure the temperature and humidity. It is also known as AM2302. This sensor is cheap and also has better accuracy.

Specifications of DHT22

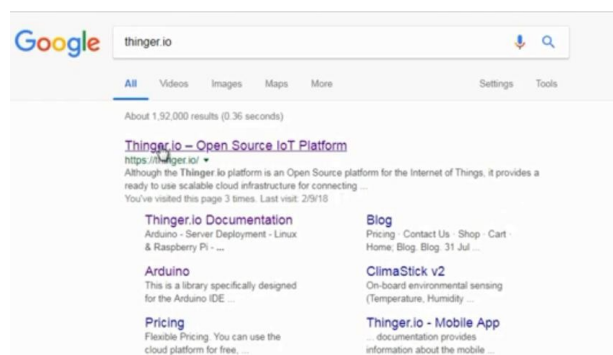
The specifications of the temperature and humidity sensor DHT22 are as follows:

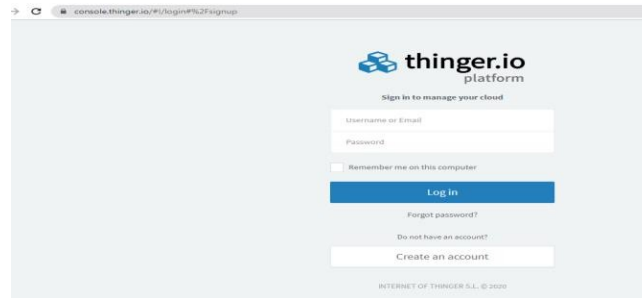
- Temperature range is from -40 to 125 degree Centigrade with accuracy of $\pm 0.5^{\circ}\text{C}$.
- Humidity range is from 0 to 100% with accuracy of $\pm 2-5\%$.
- Sampling rate is 0.5 Hz.
- Operating Voltage is 3-5V.
- Maximum Current while measuring is 2.5mA.

Setting up the Thingier.io Account:

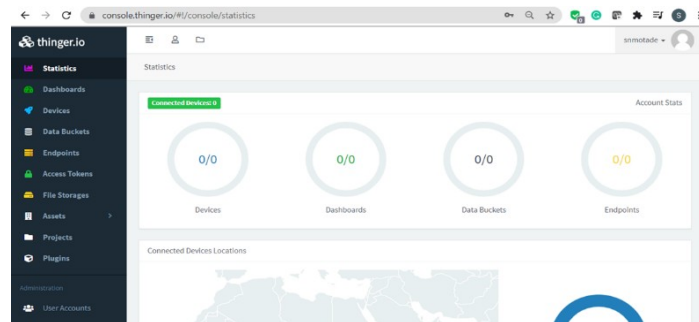
Thingier.io is a cloud IoT Platform that provides every needed tool to prototype, scale and manage connected products in a very simple way.

- **Free IoT platform:** Thingier.io provides a lifetime freemium account with only few limitations to start learning and prototyping when your product becomes ready to scale, you can deploy a Premium Server with full capacities within minutes.
- **Simple but Powerful:** Just a couple code lines to connect a device and start retrieving data or controlling its functionalities with our web-based Console, able to connect and manage thousands of devices in a simple way.
- **Hardware agnostic:** Any device from any manufacturer can be easily integrated with Thingier.io's infrastructure.
- **Extremely scalable & efficient infrastructure:** thanks to our unique communication paradigm, in which the IoT server subscribes device resources to retrieve data only when it is necessary, a single Thingier.io instance is able to manage thousands of IoT devices with low computational load, bandwidth and latencies.
- **Open-Source:** most of the platform modules, libraries and APP source code are available in our Github repository to be downloaded and modified with MIT license.



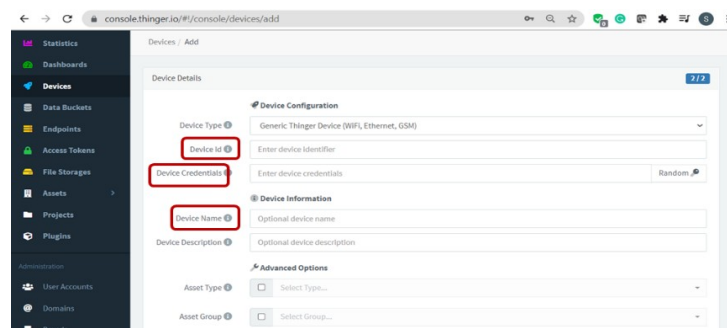
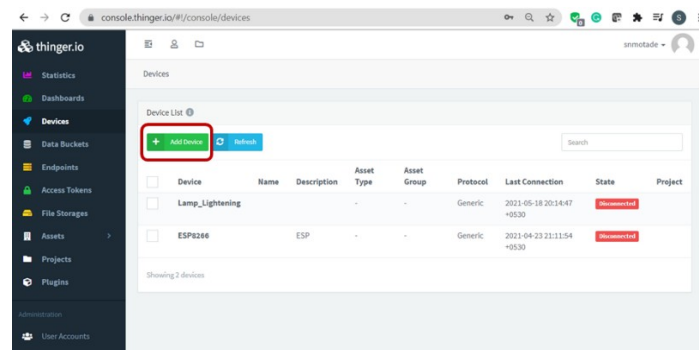


Thinger.io Cloud Platform:

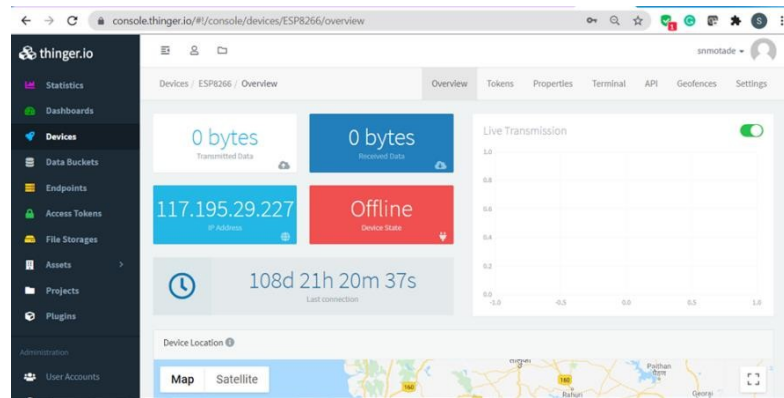


Create the device:

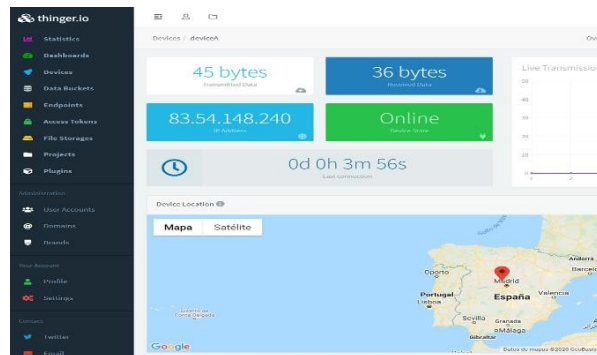
The first step with Thinger.io (except for not connected devices like Sigfox) is creating a device profile, which will relate the hardware device with the user account. Any device in Thinger.io must be registered to get access to the cloud. Each one has its own identifier and credentials and is related to the user account. All device creation and management processes are performed from the devices tab in the main menu.



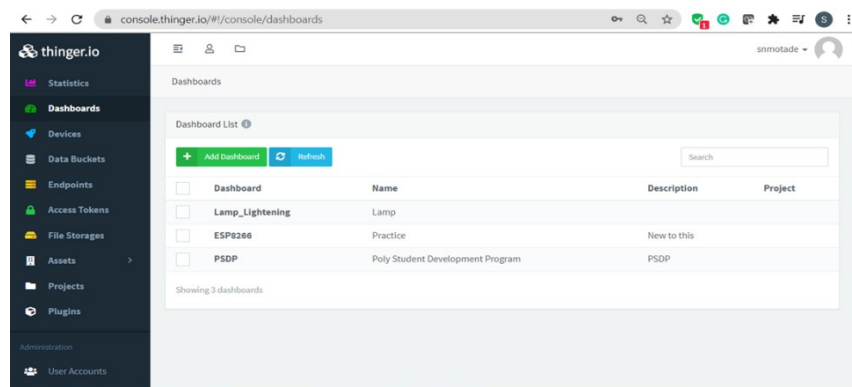
Once in the code our account identifier, device identifier, and device credentials have established, we can compile and flash the program. Meanwhile, we can open our device in the cloud console, just by clicking its identifier in the devices list. In the device screen, you will be able to see some information about your device, like its IP address, connection status, or sent/received information in real time. By default, our device will appear as disconnected, as shown in the picture below.

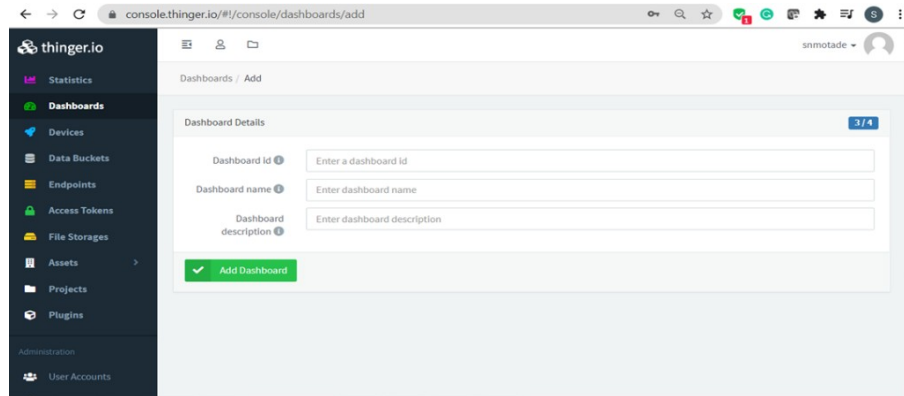


Once the device gets connected to the account, the interface will change its status, showing "Online" status, and some connection data like the IP address or the upload/download data amount:



Add Dashboard:





Conclusion:

Post Lab Questions:

1. List and write features of any one IoT Cloud platforms.
2. Compare between DHT11 and DHT 22
3. What is the role of cloud in IoT?
4. What are the features of Thinger.io

Additional links for more information:

1. Monitoring DHT11 Sensor Data with NodeMCU ESP8266 and Thinger IO
<https://www.youtube.com/watch?v=nWRtz7jRvVE>
2. Installing DHT11/DHT22 sensor libraries for Arduino/NodeMCU
<https://iot4beginners.com/installing-dht11-dht22-libraries-for-arduino-nodemcu/>
3. DHT11 Temperature & Humidity sensor on NodeMCU using Arduino IDE
<https://roboindia.com/tutorials/dht11-nodemcu-arduino/>
4. Thinger.io Documentation
<https://docs.thinger.io/features>



Dr. Vishwanath Karad

MIT WORLD PEACE
UNIVERSITY | PUNE

TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS