Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

Second Year B. Tech (EL & CE)

Semester: IV **Subject:** Basic IoT Laboratory

Name: Shreerang Mhatre **Class: SY**

Roll No: 29 **Batch: A2**

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: 02/03/2023

Submitted on: 04/03/2023

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 Thinger.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

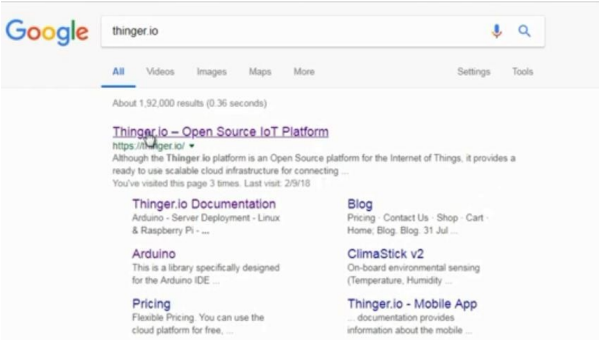
The specifications of the temperature and humidity sensor DHT22 are as follows:

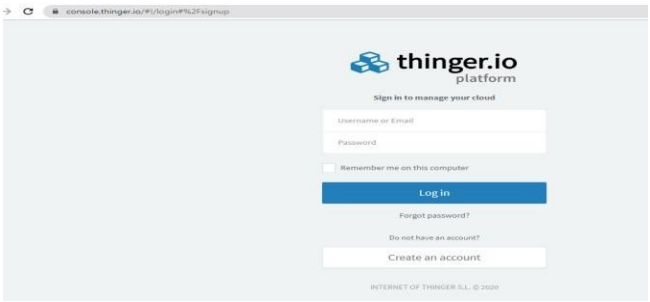
- Temperature range is from -40 to 125 degree Centigrade with accuracy of ±0.5 °C.
- Humidity range is from 0 to 100% with accuracy of ± 2-5%.
- Sampling rate is 0.5 Hz.
- Operating Voltage is 3-5V.
- Maximum Current while measuring is 2.5mA.

Setting up the Thinger.io Account:

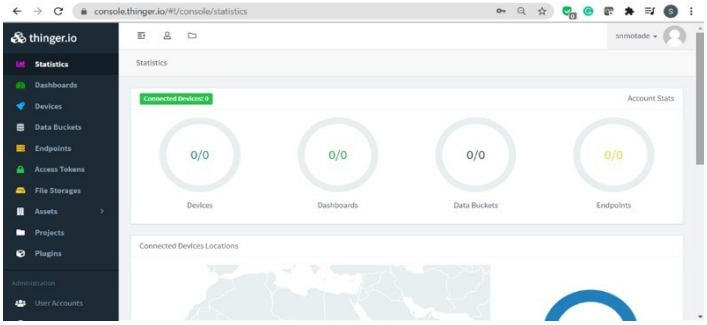
Thinger.io is a cloud IoT Platform that provides every needed tool to prototype, scale and manage connected products in a very simple way.

- **Free IoT platform:** Thinger.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 Thinger.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 Thinger.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.



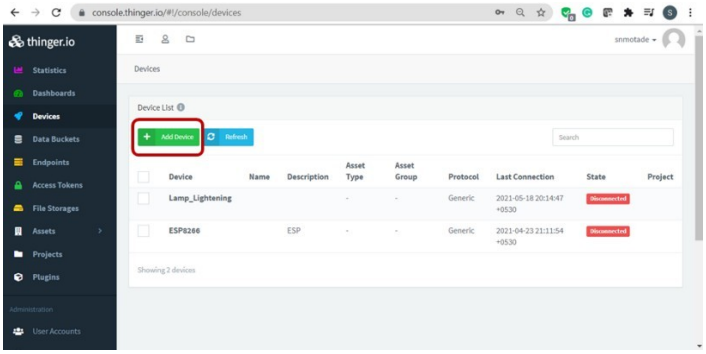


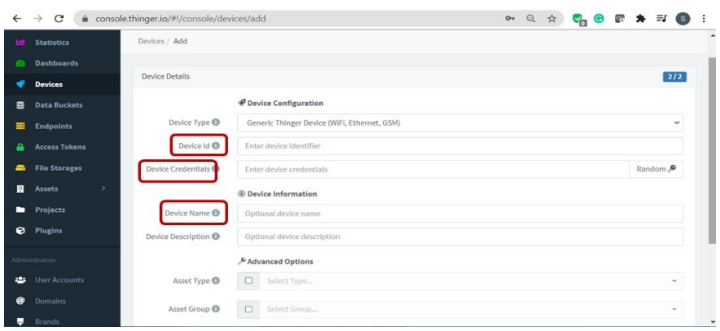
Thinger.io Cloud Platform:



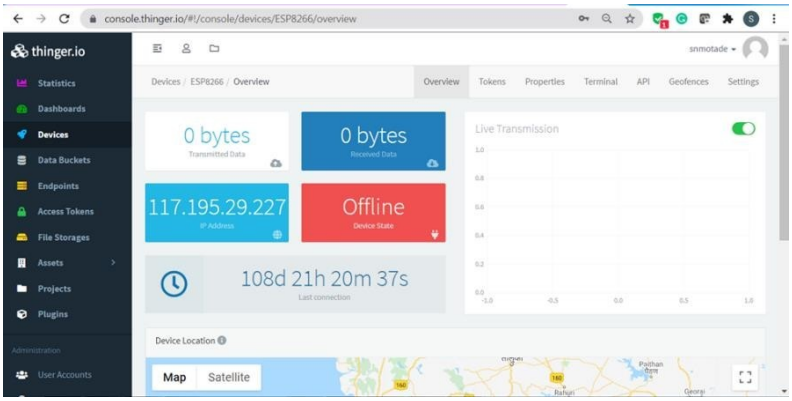
Create the device:

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.

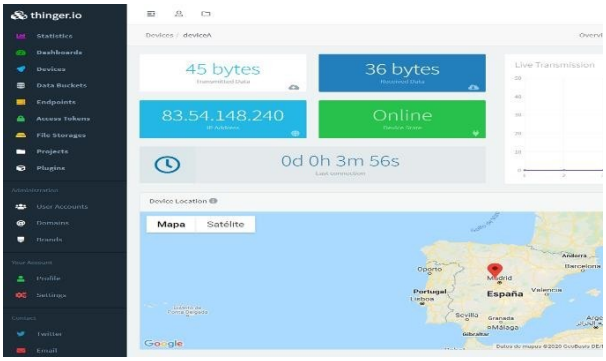




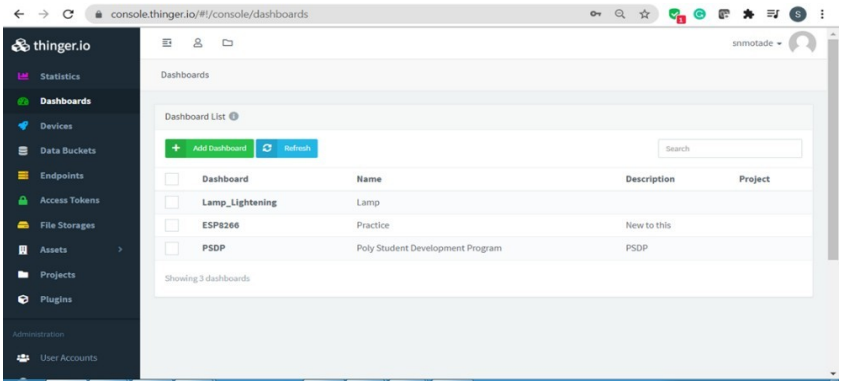
Once in the code our account identifier, device identifier, and device credentials have established, we can compile and flash the program. Meanwhile, we can open our device in the cloud console, just by clicking its identifier in the devices list. In the device screen, you will be able to see some information 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.

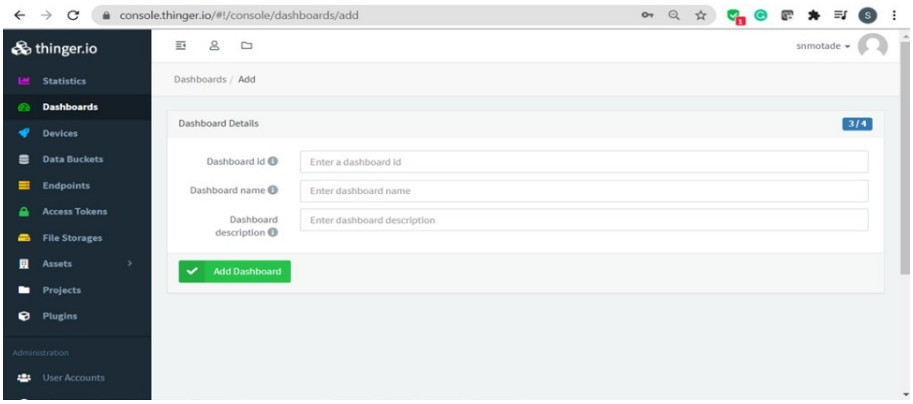


Once the device gets connected to the account, the interface will change its status, showing "Online" status, and some connection data like the IP address or the upload/download data amount:



Add Dashboard:





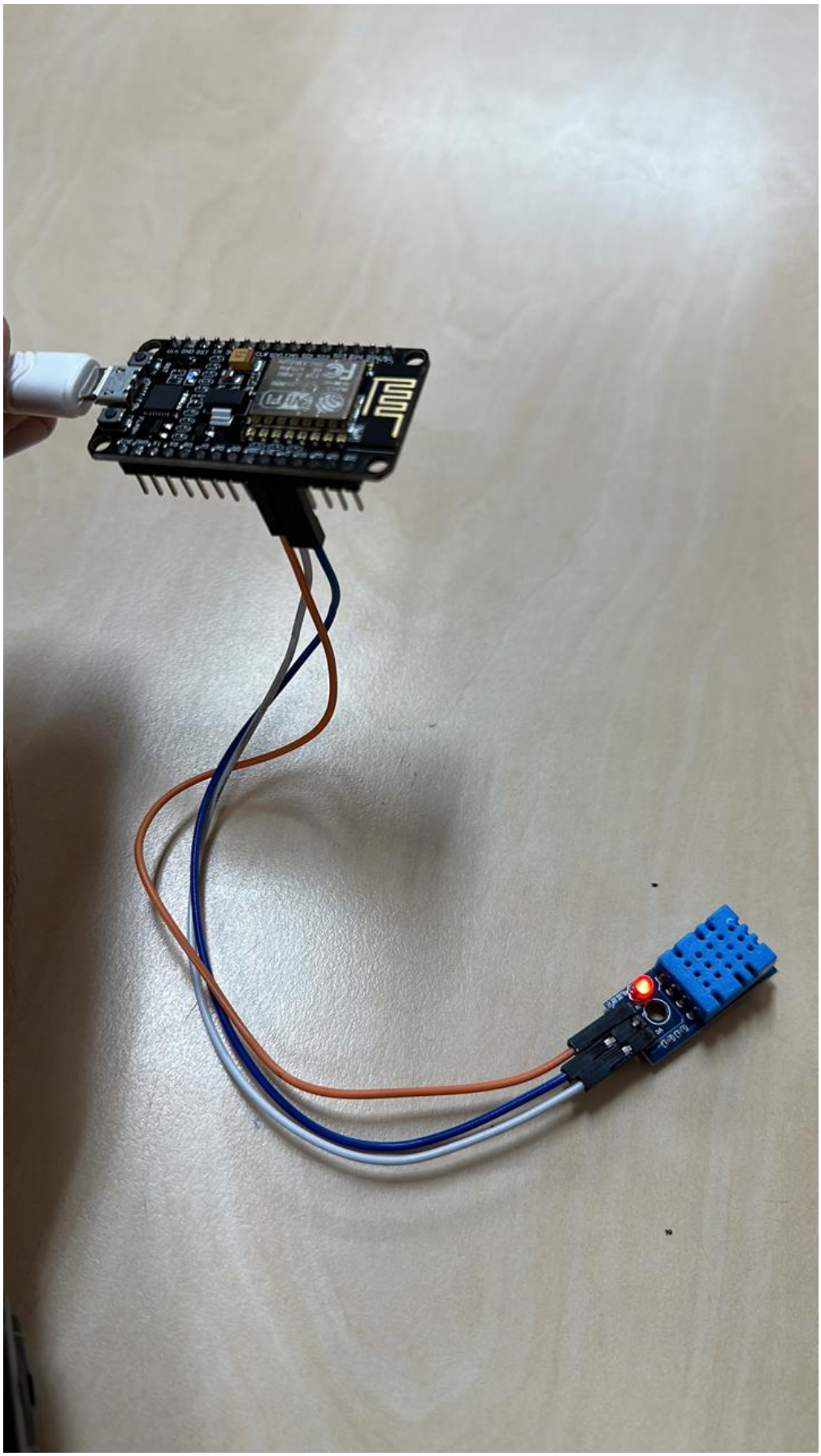
Conclusion:

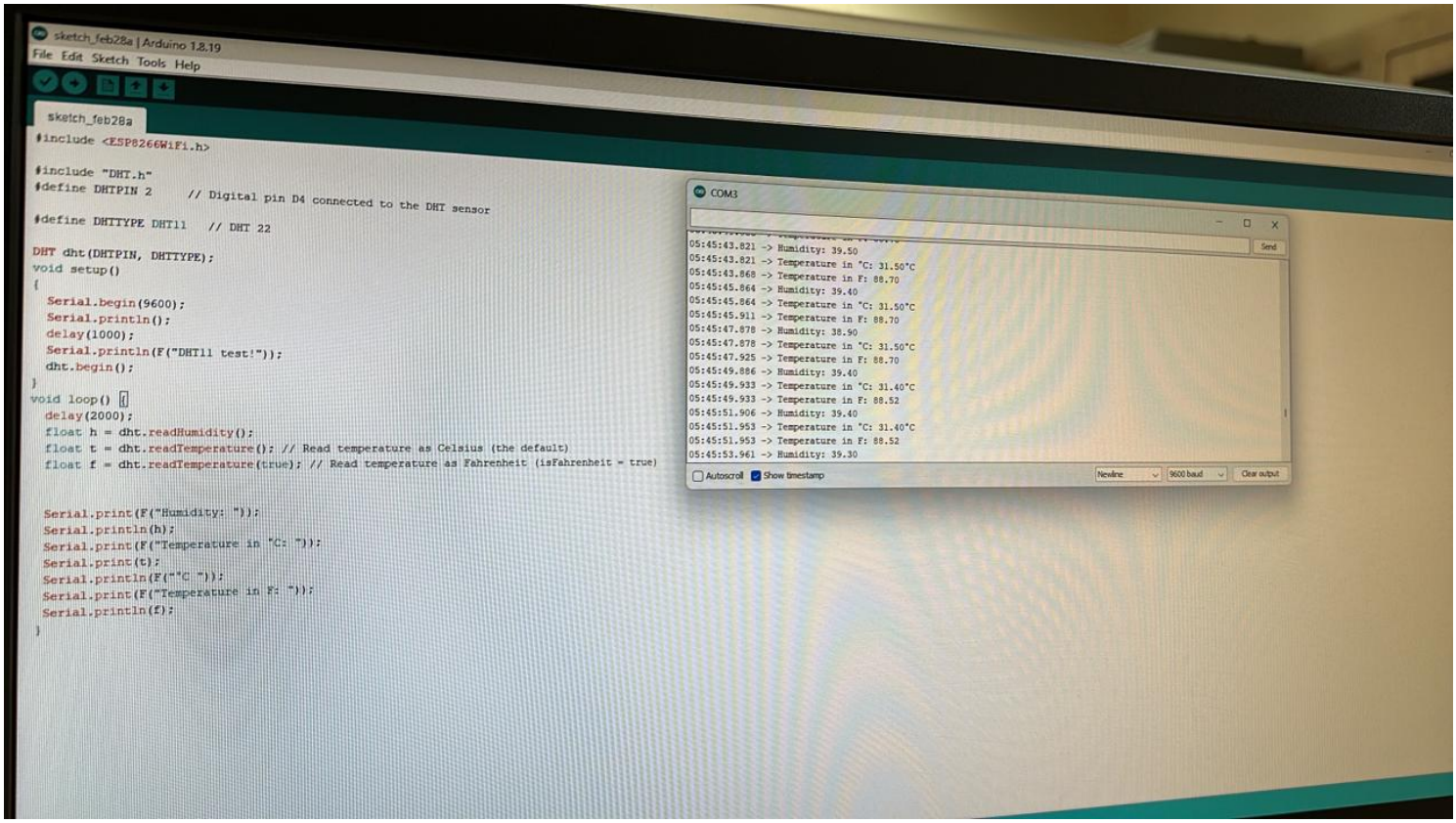
Post Lab Questions:

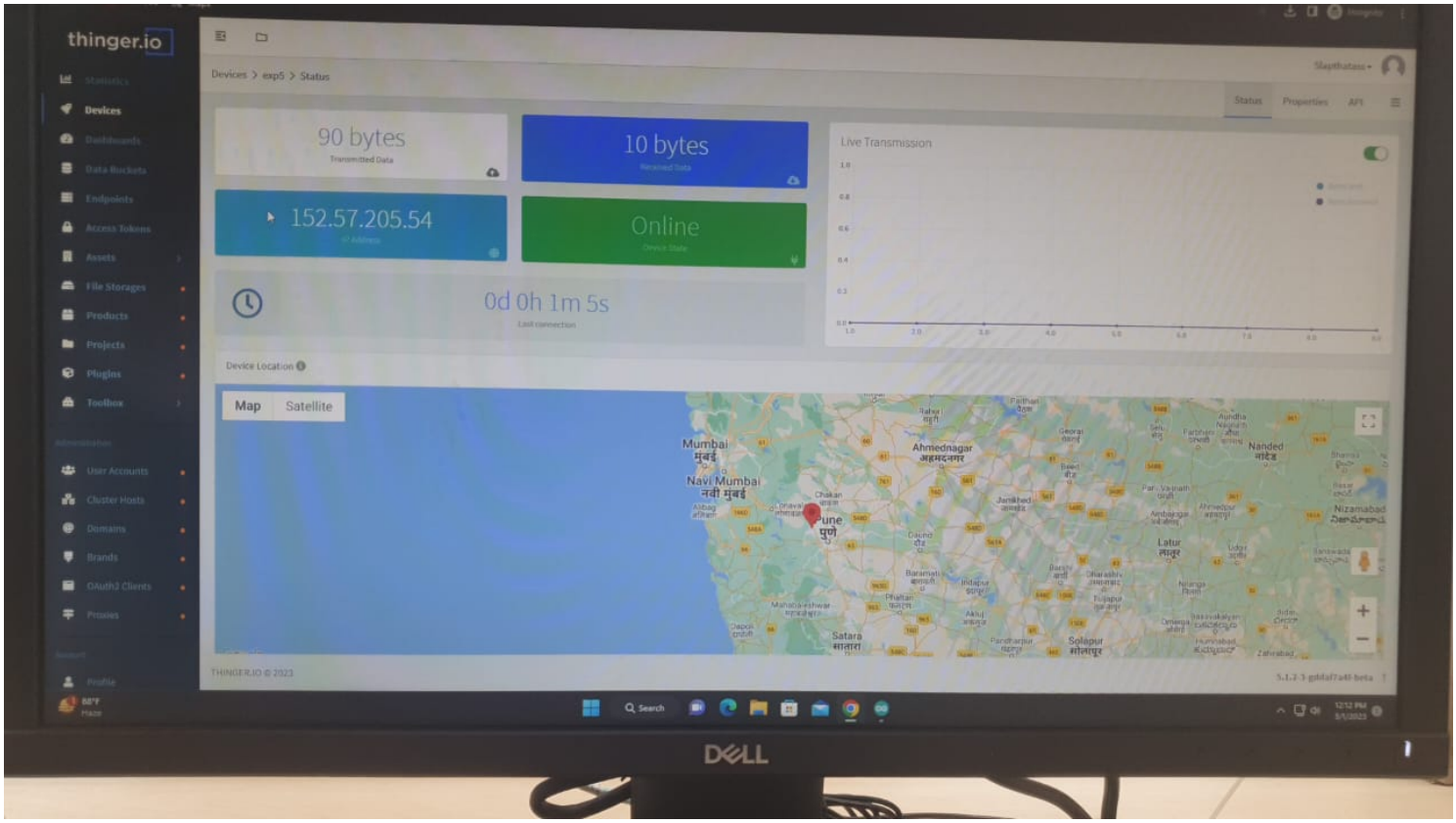
1. List and write features of any one IoT Cloud platforms.
2. Compare between DHT11 and DHT 22
3. What is the role of cloud in IoT?
4. What are the features of Thinger.io

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




sketch_feb28a | Arduino 1.8.19
File Edit Sketch Tools Help
sketch_feb28a
#include <ESP8266WiFi.h>
#include "DHT.h"
#define DHTPIN 2 // Digital pin D4 connected to the DHT sensor
#define DHTTYPE DHT11 // DHT 22
DHT dht(DHTPIN, DHTTYPE);
void setup()
{
Serial.begin(9600);
Serial.println();
delay(1000);
Serial.println(F("DHT11 test!"));
dht.begin();
}
void loop() {
delay(2000);
float h = dht.readHumidity();
float t = dht.readTemperature(); // Read temperature as Celsius (the default)
float f = dht.readTemperature(true); // Read temperature as Fahrenheit (isFahrenheit = true)
Serial.print(F("Humidity: "));
Serial.println(h);
Serial.print(F("Temperature in °C: "));
Serial.print(t);
Serial.println(F("°C "));
Serial.print(F("Temperature in F: "));
Serial.println(f);
}
COM3
Send
05:45:43.821 -> Humidity: 39.50
05:45:43.821 -> Temperature in °C: 31.50°C
05:45:43.868 -> Temperature in F: 88.70
05:45:45.864 -> Humidity: 39.40
05:45:45.864 -> Temperature in °C: 31.50°C
05:45:45.911 -> Temperature in F: 88.70
05:45:47.878 -> Humidity: 38.90
05:45:47.878 -> Temperature in °C: 31.50°C
05:45:47.925 -> Temperature in F: 88.70
05:45:49.886 -> Humidity: 39.40
05:45:49.933 -> Temperature in °C: 31.40°C
05:45:49.933 -> Temperature in F: 88.52
05:45:51.906 -> Humidity: 39.40
05:45:51.953 -> Temperature in °C: 31.40°C
05:45:51.953 -> Temperature in F: 88.52
05:45:53.961 -> Humidity: 39.30
Autoscroll
Show timestamp
Newline
9600 baud
Clear output


thinger.io
Devices > exp5 > Status
Status
Properties
API
90 bytes
Transmitted Data
10 bytes
Received Data
152.57.205.54
IP Address
Online
Device State
0d 0h 1m 5s
Last connection
Live Transmission
Device Location
Map
Satellite
Mumbai
मुंबई
Navi Mumbai
नवी मुंबई
Ahmednagar
अहमदनगर
Pune
पुणे
Satara
सातारा
Solapur
सोलापूर
Latur
लातूर
Nanded
नांदेड
Nizamabad
Devices
Dashboards
Data Buckets
Endpoints
Access Tokens
Assets
File Storages
Products
Projects
Plugins
Toolbox
User Accounts
Cluster Hosts
Domains
Brands
OAuth2 Clients
Proxies
Profile
THINGER.IO © 2023
DELL

Name: Shreerang Mhatre
Roll no: 29
Batch: A2

01/03/2023

MIT-WPU
MIT WORLD PEACE UNIVERSITY | PUNE

Exp no 5

* Post Lab Questions -

Q1) List & write features of any one IoT cloud platforms -

→ Device Management -

AWS IoT provides device management features that enable users to securely register, organize, & manage their IoT devices at scale. It also provides features such as over-the-air updates, remote device monitoring, and management of device configurations & policies. AWS IoT is a comprehensive IoT cloud platform that provides a range of features and capabilities for developing and deploying IoT solutions. Its device management, data management security & compliance & edge computing capabilities make it a powerful & flexible platform for building scalable & secure IoT solutions.

Q2) Compare between DHT11 & DHT22

	Parameter	DHT11	DHT22
①	Temp range	0° to 50°C	−40° to 80°C
②	Temp accuracy	±1-2°C	±0.5°C
③	Relative humidity range	30% to 90%	0% to 100%
④	Humidity accuracy	±4 to 5%	±2 to 5%
⑤	Resolution	8bits	16bits
⑥	Sampling period	≥1s	≥2s
⑦	Price / cost	Low cost	Higher cost

Q3) What is the role of cloud in IoT?

→ The Internet of things generates a huge amount of data per second. cloud computing helps in storing & analyzing this data to optimise IoT infrastructure. It also helps in the modernisation of operations by connecting legacy & smart devices, and machines to the internet & reducing the barriers between IT & OT teams with a unified view of systems. & data. It also reduces the extensive deployment costs for hardware or configure & manage networks & infrastructure relating to IoT.

Q4) What are the features of Thinger.io?

-> Features of thinger.io are -

1. Private cloud: It deploys in a flexible way and adapts to each project needs
2. Projects managements: All assets on thinger.io can be organised in terms of projects.
3. Allows many IoT device connection: It accepts almost any device in the market
4. store, view and analyse'.
One can see real time and historic data on its dashboard which can be further customised according to user requirement for analysis.