

# SENDING DATA FROM RASPBERRY-PI TO IBM WATSON

<b>Team ID</b>	PNT2022TMID36136
<b>Project Name</b>	GAS LEAKAGE MONITORING AND ALERTING SYSTEM FOR INDUSTRIES

## **Team members:**

NITHISH KUMAR V R  
SANDEEP P B  
BEGAN BABU P  
KISHORE B L

## **AIM:**

To send sensor data (or any dummy data) from Raspberry –Pi to IBM Watson .In our case it is DHT sensors Data.

## **REQUIREMENTS:**

### **HARDWARE:**

- RASPBERRY-PI (3B)(WITH ETHERNET CABLE OR WIFI CONNECTED)
- USB MOUSE
- USB KEYBOARD
- VGA TO HDMI CABLE
- A MONITOR
- RASPBERRY’S POWER SUPPLY
- DHT-11 Sensor
- Connecting Wires

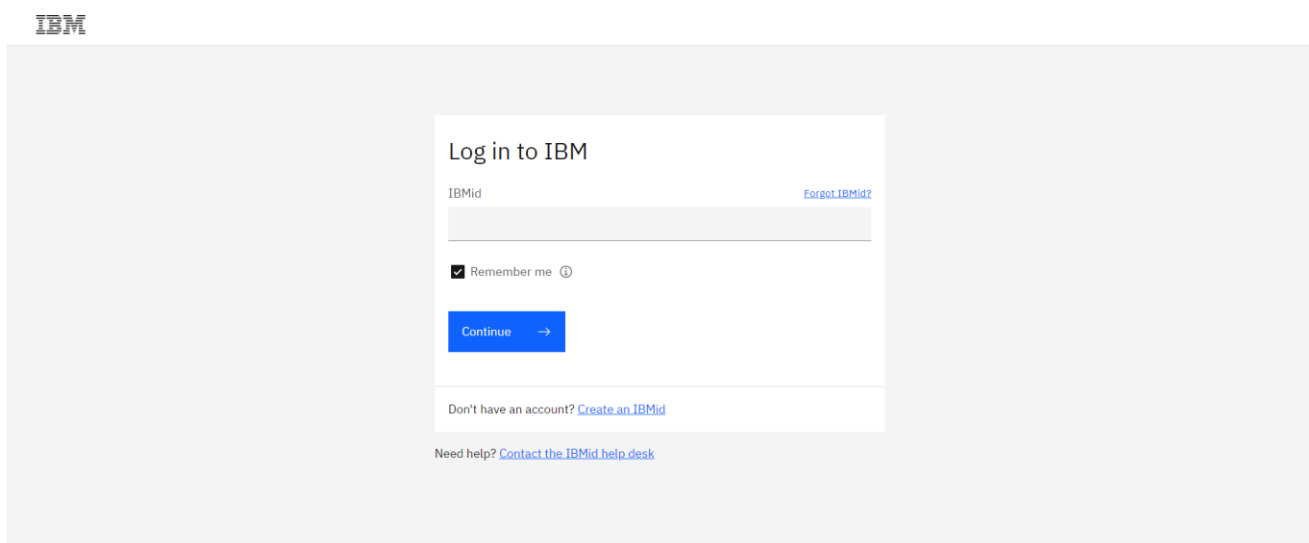
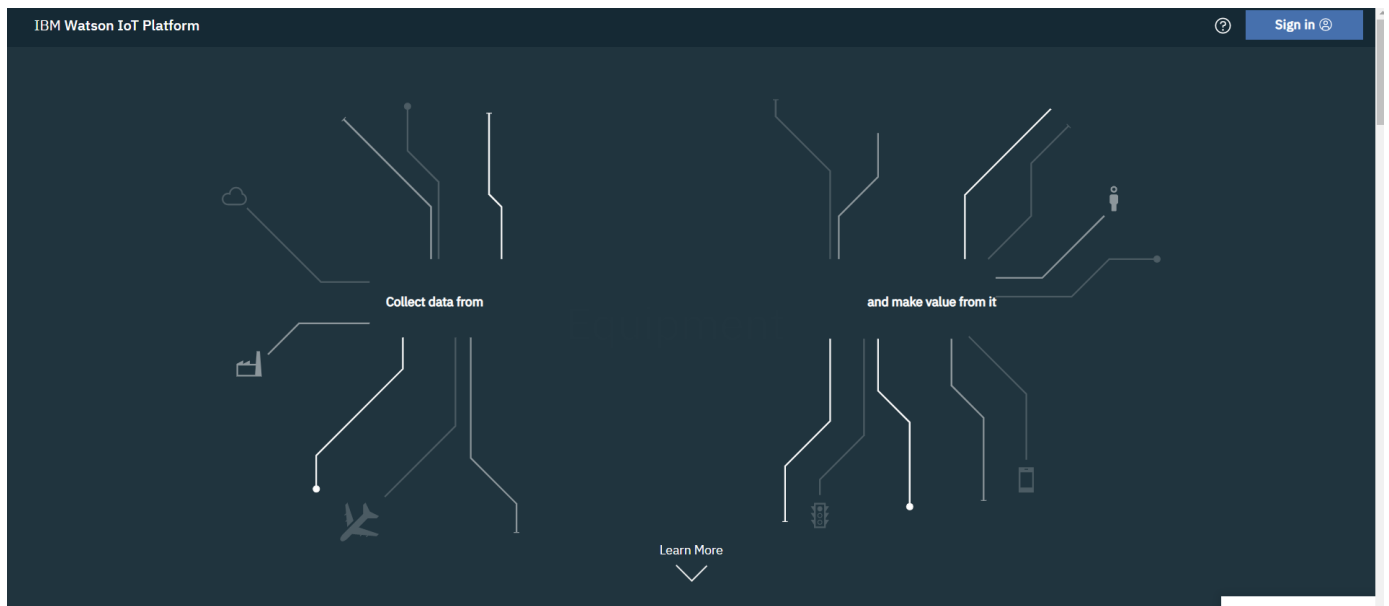
### **SOFTWARE:**

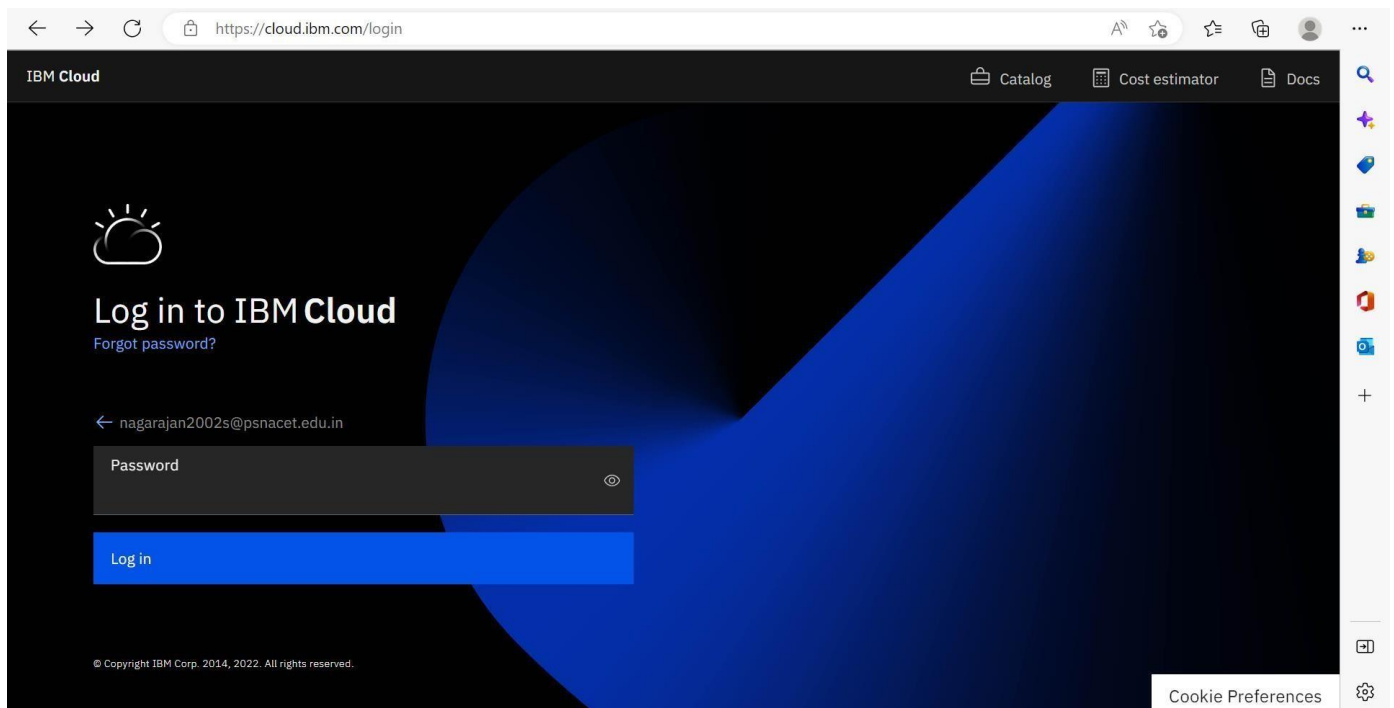
- IBM BLUEMIX ACCOUNT

## STEPS TO BE FOLLOWED

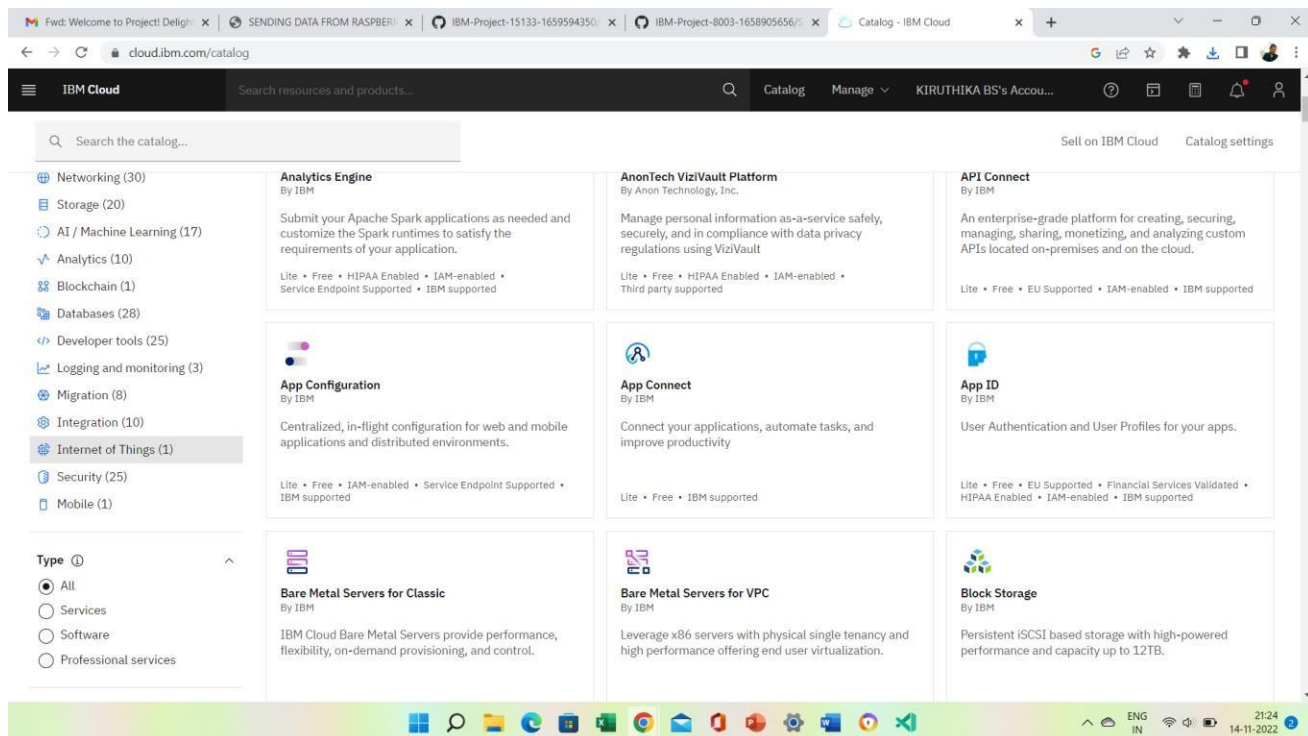
### Step-1: Create a device in IBM Watson:

- Firstly, login into your IBM-Bluemix account with your e-mail ID and Password.





➤ Click on catalog on your dashboard screen, then under platform go IoT.



➤ Check all details and click on create.

The screenshot shows the IBM Cloud catalog page for the Internet of Things Platform. The page is titled "Internet of Things Platform" and includes a description: "This service is the hub of all things IBM IoT, it is where you can set up and manage your connected devices so that your apps can access their live and historical data." The "Create" tab is selected, and the "Select a location" dropdown is set to "Frankfurt (eu-de)". The "Select a pricing plan" section shows a table with two plans: "Lite" and "Free". The "Lite" plan is selected, and its details are shown in a modal. The modal includes a summary of the plan's features and pricing, and a "Create" button. The "Free" plan is also listed with a "Add to estimate" button.

Plan	Features	Pricing
Lite	Includes up to 500 registered devices, and a maximum of 200 MB of each data metric Maximum of 500 registered devices Maximum of 500 application bindings Maximum of 200 MB of each of data exchanged, data analyzed and edge data analyzed	Free
Free		

➤ click on Launch

The screenshot shows the IBM Cloud service details page for the Internet of Things Platform. The page is titled "Internet of Things Platform-tk" and includes a "Launch" button. The "Launch" button is highlighted, and the "Docs" button is also visible. The page includes a section titled "Let's get started with IBM Watson IoT Platform" and a "Ready for the next level?" section. The "Ready for the next level?" section includes three options: "Lite", "Non-Production", and "Production". The "Lite" option is selected, and its details are shown in a modal. The modal includes a summary of the plan's features and pricing, and a "Launch" button.

Plan	Features	Pricing
Lite	The Lite service plan provides a lightweight development environment to get you started with the connectivity capabilities of Watson IoT Platform.	Free
Non-Production	The Non-Production service plan is a full-featured, fully-integrated offering that enables you to explore Watson IoT Platform to see how the service can fit into your IoT environment.	Starts at \$500 per month
Production	The Production service is a fully managed SaaS offering that enables you to manage and analyze enterprise IoT data.	Includes IBM Service & Support

- Dashboard of IBM Watson IoT platform,
- Click on Add device

IBM Watson IoT Platform

Browse Action Device Types Interfaces

Add Device

## Browse Devices

All Devices Diagnose

This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

Search by Device ID

Device Simulator

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
01234	Disconnected	KIRUTHIKA	Device	Nov 14, 2022 9:33 PM	
KIRUTHIKA_1	Disconnected	KIRUTHIKA	Device	Nov 14, 2022 10:07 PM	

Items per page 50 | 1-2 of 2 items

1 of 1 page

2 Simulations running

- After click on Add device this page will open

IBM Watson IoT Platform

Browse Action Device Types Interfaces

## Add Device

Identity Device Information Security Summary

Select a device type for the device that you are adding and give the device a unique ID.

Device Type

Device ID

Cancel Next

## Browse Devices

All Devices Diagnose

5 Simulations running

➤ Go to device type and fill the details.

IBM Watson IoT Platform

Browse Action **Device Types** Interfaces

### Add Type

Identity Device Information

Device types group devices that have similar characteristics, such as model number, firmware version, or location. Give the device type a unique name and a description that identifies characteristics that are shared by devices of this type.

Type ☐ Device Or ☐ Gateway

Name

Description

Cancel Next

5 Simulations running

➤ Click on Finish

IBM Watson IoT Platform

Browse Action **Device Types** Interfaces

### Add Type

Identity Device Information

These attributes will be used as a template for new devices that are assigned this device type

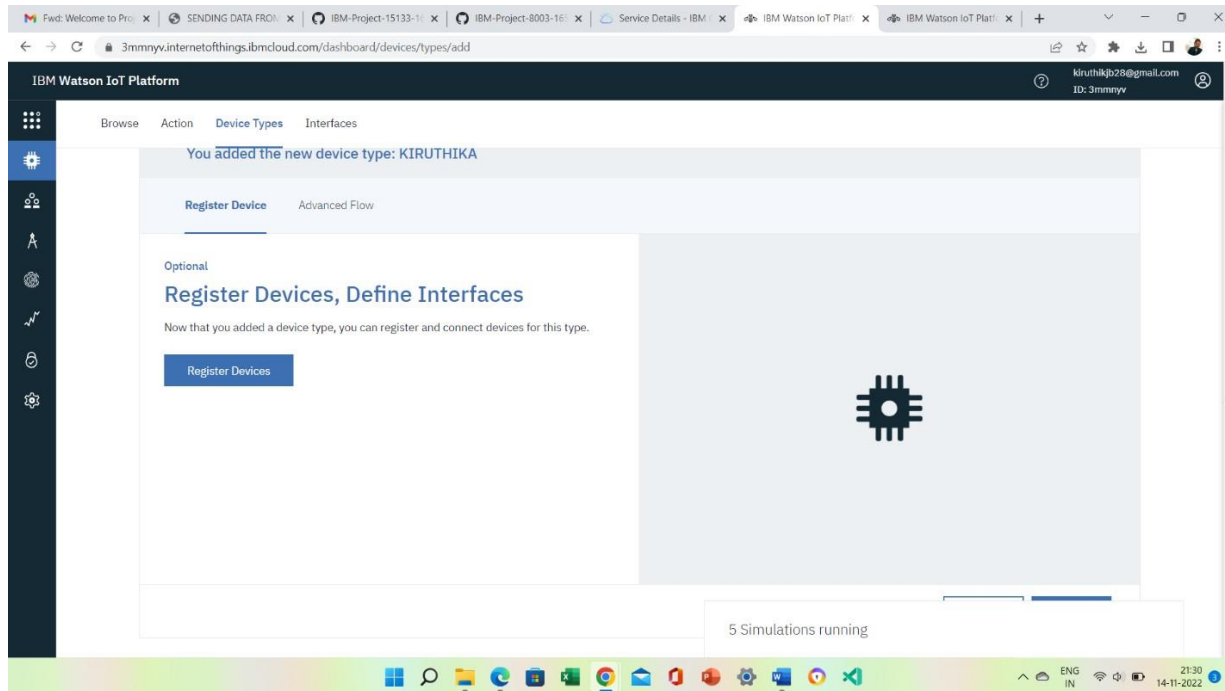
Edit Metadata

Serial Number	<input type="text" value="Enter Serial Number"/>	Manufacturer	<input type="text" value="Enter Manufacturer"/>
Model	<input type="text" value="Enter Model"/>	Device Class	<input type="text" value="Enter Device Class"/>
Description	<input type="text" value="Enter Description"/>	Firmware Version	<input type="text" value="Enter Firmware Version"/>
Hardware Version	<input type="text" value="Enter Hardware Version"/>	Descriptive Location	<input type="text" value="Enter Descriptive Location"/>

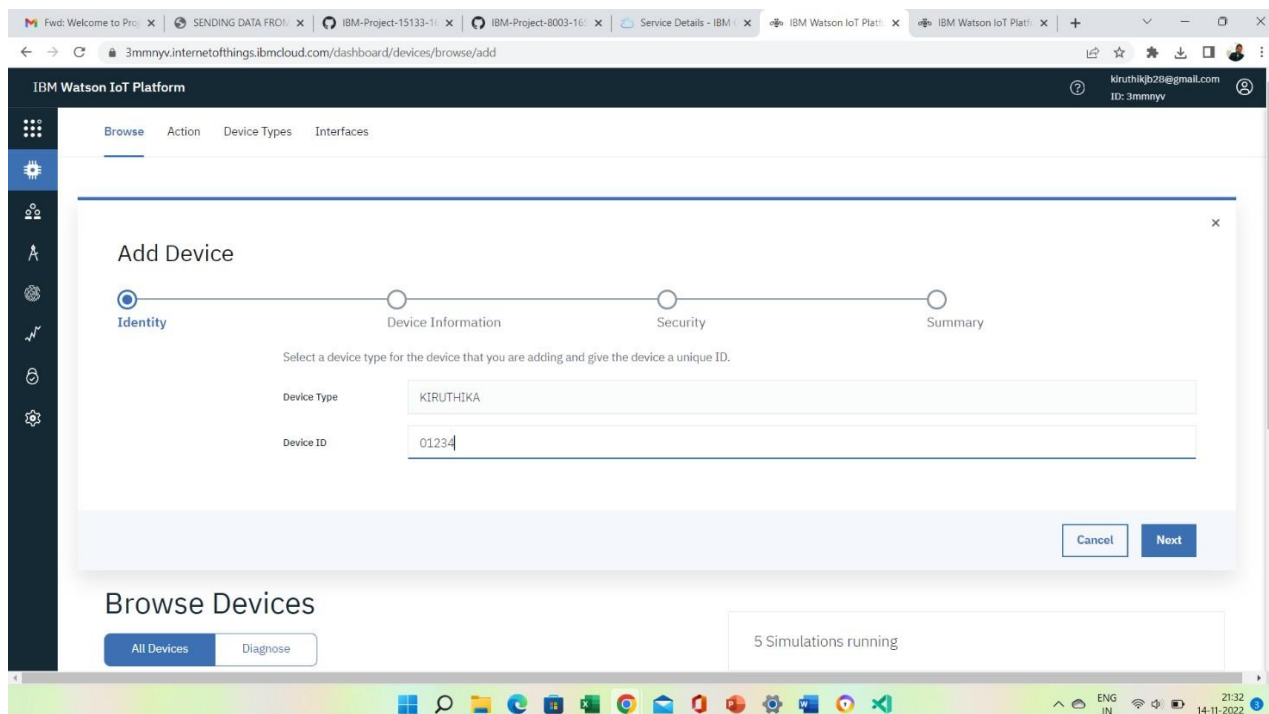
Back Finish

2 Simulations running

➤ Click on Register Device.



➤ Choose the device and give device ID and then click on next.



## ➤ Click on Next

The screenshot shows the 'Add Device' page in the IBM Watson IoT Platform. The progress bar indicates that the 'Identity' step is complete, and the 'Device Information' step is currently active. The page contains several input fields for device details:

- Serial Number:** Enter Serial Number
- Model:** Enter Model
- Description:** Enter Description
- Hardware Version:** Enter Hardware Version
- Manufacturer:** Enter Manufacturer
- Device Class:** Enter Device Class
- Firmware Version:** Enter Firmware Version
- Descriptive Location:** Enter Descriptive Location

At the bottom right, there are 'Back' and 'Next' buttons. A status bar at the bottom indicates '5 Simulations running'.

## ➤ Click on Next

The screenshot shows the 'Add Device' page in the IBM Watson IoT Platform, specifically the 'Security' step. The progress bar shows that 'Identity' and 'Device Information' are complete, and 'Security' is the current step. The page provides information about authentication tokens:

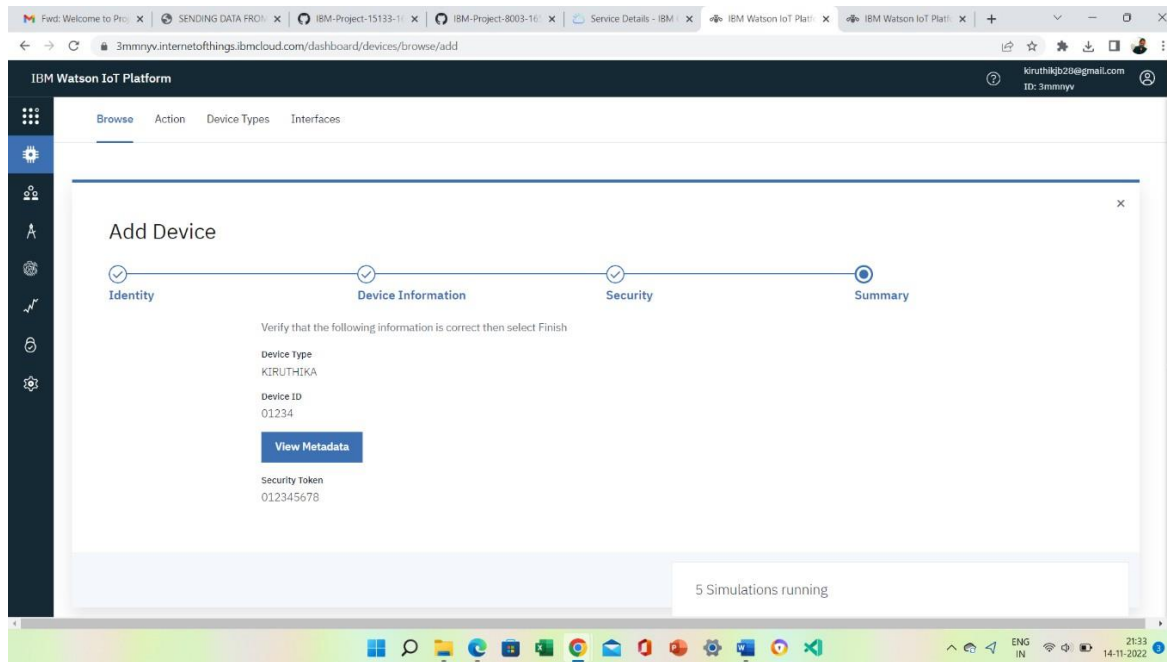
- Auto-generated authentication token (default):** Allow the service to generate an authentication token for you. Tokens are 18 characters and contain a mix of alphanumeric characters and symbols. The token is returned to you at the end of the device registration process.
- Self-provided authentication token:** Provide your own authentication token for this device. The token must be between 8 and 36 characters and contain a mix of lowercase and uppercase letters, numbers, and symbols, which can include hyphens, underscores, and periods. Do not use repeated characters, dictionary words, user names, or other predefined sequences.

Below the text, there is an input field labeled 'Authentication Token' with the placeholder text 'Enter an optional token'. A note below the field states: 'Make a note of the generated token. Lost authentication tokens cannot be recovered. Tokens are encrypted before being stored. Authentication token are encrypted before we store them.'

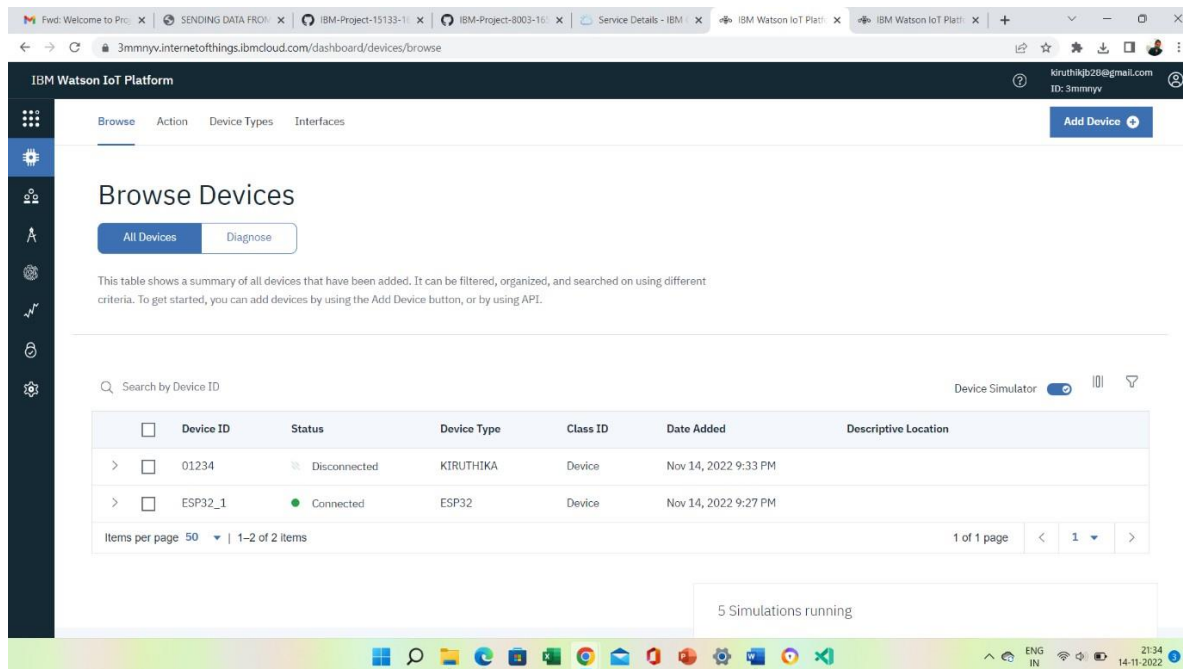
At the bottom right, there are 'Back' and 'Next' buttons. A status bar at the bottom indicates '5 Simulations running'.



## ➤ Click on Finish



## ➤ Device is created



## STEP-2: INSTALLING NECESSARY PACKAGES ON YOUR PI:

- Now we are going to install necessary packages on your pi.
- Open your terminal in your pi and type the following commands
- `curl -LO https://github.com/ibm-messaging/iot-raspberrypi/releases/download/1.0.2.1/iot_1.0-2_armhf.deb`
- `sudo dpkg -i iot_1.0-2_armhf.deb`
- `service iot status`

Following are the images as to what appears on your pi's terminal when u type these commands

```
File Edit Tabs Help
2017-10-23 06:55:22-- http://ftp.nl.debian.org/debian/pool/main/o/openssl/lib
ssl1.0.0_1.0.1t-1-deb8u6_armhf.deb
Resolving ftp.nl.debian.org (ftp.nl.debian.org)... 130.89.149.21, 2001:67c:2564:
a120::21
Connecting to ftp.nl.debian.org (ftp.nl.debian.org)[130.89.149.21]:80... connect
ed.
HTTP request sent, awaiting response... 200 OK
Length: 867950 (848K) [application/x-debian-package]
Saving to: 'libssl1.0.0_1.0.1t-1-deb8u6_armhf.deb'

libssl1.0.0_1.0.1t- 100%[=====] 847.61K  358KB/s   in 2.4s

2017-10-23 06:55:25 (358 KB/s) - 'libssl1.0.0_1.0.1t-1-deb8u6_armhf.deb' saved [
867950/867950]

pi@raspberrypi:~$ sudo dpkg -i libssl1.0.0_1.0.1t-1-deb8u6_armhf.deb
Selecting previously unselected package libssl1.0.0:armhf.
(Reading database ... 115606 files and directories currently installed.)
Preparing to unpack libssl1.0.0_1.0.1t-1-deb8u6_armhf.deb ...
Unpacking libssl1.0.0:armhf (1.0.1t-1-deb8u6) ...
Setting up libssl1.0.0:armhf (1.0.1t-1-deb8u6) ...
pi@raspberrypi:~$ curl -LO https://github.com/ibm-messaging/iot-raspberrypi/rel
eases/download/1.0.2.1/iot_1.0-2_armhf.deb
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left  Speed
100 164 0 164 0 0 157 0 --:--:-- 0:00:01 --:--:-- 157
100 609 0 609 0 0 457 0 --:--:-- 0:00:01 --:--:-- 457
100 110k 100 110k 0 0 20117 0 0:00:03 0:00:03 --:--:-- 48190
pi@raspberrypi:~$ sudo dpkg -i iot_1.0-2_armhf.deb
(Reading database ... 115626 files and directories currently installed.)
Preparing to unpack iot_1.0-2_armhf.deb ...
Unpacking iot (1.0-1) over (1.0-1) ...
Setting up iot (1.0-1) ...
Processing triggers for systemd (232-25-deb9u1) ...
pi@raspberrypi:~$ service iot status
* iot.service - LSB: iot service
   Loaded: loaded (/etc/init.d/iot; generated; vendor preset: enabled)
   Active: active (running) since Mon 2017-10-23 06:56:25 UTC; 17s ago
     Docs: man:systemd-sysv-generator(8)
    CGroup: /system.slice/iot.service
            └─2562 /opt/iot/iot /dev/null

Oct 23 06:56:24 raspberrypi systemd[1]: Starting LSB: iot service...
Oct 23 06:56:24 raspberrypi iot[2567]: Starting the iot program
Oct 23 06:56:25 raspberrypi iot[2562]: **** IoT Raspberry Pi Sample has started ****
Oct 23 06:56:25 raspberrypi iot[2562]: Config file not found. Going to Quickstart mode
Oct 23 06:56:25 raspberrypi iot[2562]: Running in Quickstart mode
Oct 23 06:56:25 raspberrypi systemd[1]: Started LSB: iot service.
```

- Then open your terminal and type `pip install ibmiotf`

```
File Edit Tabs Help
pi@raspberrypi:~$ pip install ibmiotf
Collecting ibmiotf
  Downloading ibmiotf-0.3.0.tar.gz (58kB)
    100% |#####| 61kB 510kB/s
Collecting dicttoxml==1.7.4 (from ibmiotf)
  Downloading dicttoxml-1.7.4.tar.gz
    100% |#####| 61kB 1.7MB/s
Collecting iso8601==0.1.10 (from ibmiotf)
  Downloading iso8601-0.1.12-py2.py3-none-any.whl
Collecting paho-mqtt==1.2 (from ibmiotf)
  Downloading paho-mqtt-1.3.1.tar.gz (80kB)
    100% |#####| 81kB 916kB/s
Collecting pytz==2014.7 (from ibmiotf)
  Using cached pytz-2017.2-py2.py3-none-any.whl
Collecting requests==2.5.0 (from ibmiotf)
  Downloading requests-2.18.4-py2.py3-none-any.whl (88kB)
    100% |#####| 92kB 1.0MB/s
Collecting requests-toolbelt==0.7.0 (from ibmiotf)
  Downloading requests-toolbelt-0.8.0-py2.py3-none-any.whl (54kB)
    100% |#####| 61kB 1.6MB/s
Collecting xmldict==0.10.2 (from ibmiotf)
  Downloading xmldict-0.11.0-py2.py3-none-any.whl
Collecting urllib3<1.23,>=1.21.1 (from requests==2.5.0->ibmiotf)
  Downloading urllib3-1.22-py2.py3-none-any.whl (132kB)
    100% |#####| 133kB 1.4MB/s
Collecting idna<2.7,>=2.5 (from requests==2.5.0->ibmiotf)
  Downloading idna-2.6-py2.py3-none-any.whl (56kB)
    100% |#####| 61kB 1.7MB/s
Collecting chardet<3.1.0,>=3.0.2 (from requests==2.5.0->ibmiotf)
  Downloading chardet-3.0.4-py2.py3-none-any.whl (133kB)
    100% |#####| 143kB 1.0MB/s
Collecting certifi==2017.4.17 (from requests==2.5.0->ibmiotf)
  Using cached certifi-2017.7.27.1-py2.py3-none-any.whl
Building wheels for collected packages: ibmiotf, dicttoxml, paho-mqtt
Running setup.py bdist_wheel for ibmiotf ... done
Stored in directory: /home/pi/.cache/pip/wheels/7e/f9/45/bbc33ad957e82f7b71ba0e316d65a83d9d735a0d12e0c0418
Running setup.py bdist_wheel for dicttoxml ... done
Stored in directory: /home/pi/.cache/pip/wheels/45/62/59/9691bb33ec6a7b2ae66a13765401b50def5468024078e12cce
Running setup.py bdist_wheel for paho-mqtt ... done
Stored in directory: /home/pi/.cache/pip/wheels/20/d8/0d/acdc8f2890111b7e7de71deebef0642fb3be0313dfff0493
Successfully built ibmiotf dicttoxml paho-mqtt
Installing collected packages: dicttoxml, iso8601, paho-mqtt, pytz, urllib3, idna, chardet, certifi, requests, requests-toolbelt, xmldict, ibmiotf
Successfully installed certifi-2017.7.27.1 chardet-3.0.4 dicttoxml-1.7.4 ibmiotf-0.3.0 idna-2.6 iso8601-0.1.12 paho-mqtt-1.3.1 pytz-2017.2 requests-2.18.4 requests-toolbelt-0.8.0 urllib3-1.22 xmldict-0.11.0
pi@raspberrypi:~$
```

- I have sent DHT-11 Sensors data to ibm bluemix .To get the code u need to login into IOT GYAN.
- Then I get the image as follows in my pi's shell:

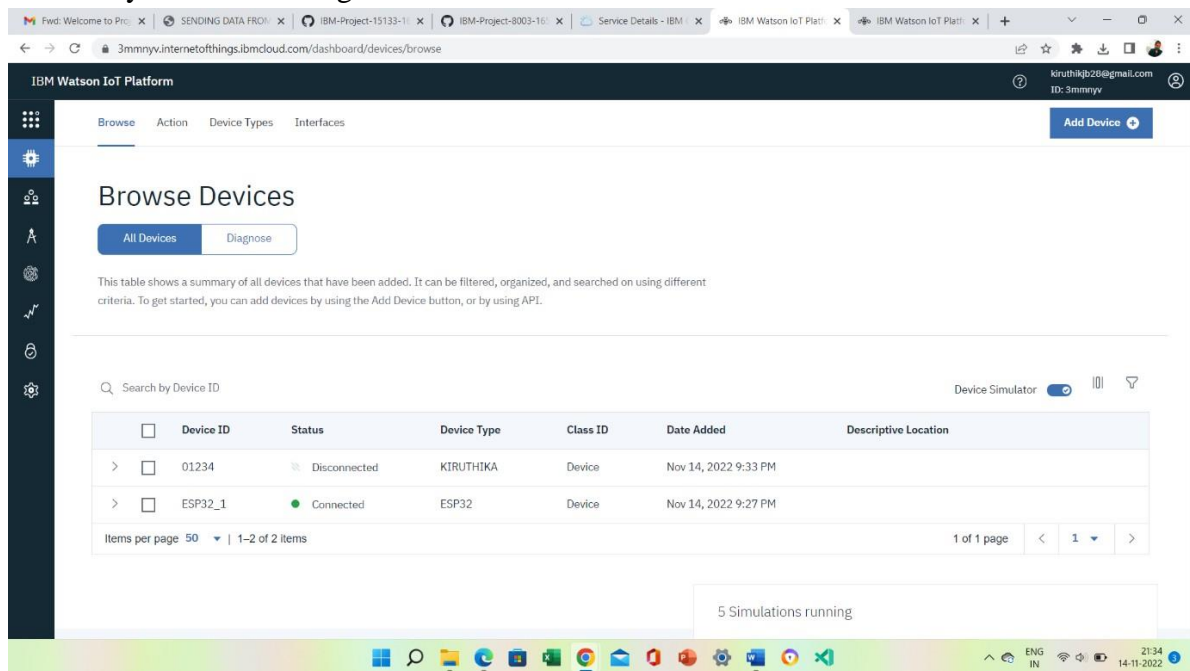
```

File Edit Shell Debug Options Window Help
Python 2.7.13 (default, Jan 19 2017, 14:48:08)
[GCC 6.3.0 20170124] on linux2
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /home/pi/Downloads/dht11toibmiot.py =====
2017-10-23 07:10:37,765 ibmiotf.device.Client INFO Connected successfully: d:gegtl4:mydevice:mydevice
SensorData Invalid
Published Temperature = 28 C Humidity = 50 % to IBM Watson
SensorData Invalid
Published Temperature = 28 C Humidity = 50 % to IBM Watson
SensorData Invalid
Published Temperature = 28 C Humidity = 50 % to IBM Watson
SensorData Invalid
Published Temperature = 28 C Humidity = 50 % to IBM Watson
Published Temperature = 29 C Humidity = 50 % to IBM Watson
Published Temperature = 29 C Humidity = 50 % to IBM Watson

```

### Step-3: checking your data sent on IBM Bluemix:

- After you have sent your sensors data you can check whether it is received at your iot platform Just look at the image below and if u see the same wifi kind of symbol on your created device then your data is being received.



- After double clicking on your created device you can see the received data as shown in image

The screenshot displays the IBM Watson IoT Platform interface. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. The main content area shows a list of devices. The device 'KIRUTHIKA\_1' is selected, and its details are shown in a modal window. The modal has tabs for 'Identity', 'Device Information', 'Recent Events', 'State', and 'Logs'. The 'Recent Events' tab is active, showing a table of recent events.

Event	Value	Format	Last Received
event_1	{"randomNumber":36}	json	a few seconds ago
event_1	{"randomNumber":96}	json	a few seconds ago
event_1	{"randomNumber":6}	json	a few seconds ago
event_1	{"randomNumber":55}	json	a few seconds ago
event_1	{"randomNumber":6}	json	a few seconds ago

2 Simulations running

#### Step-4: Creating boards and cards for visualization of data:

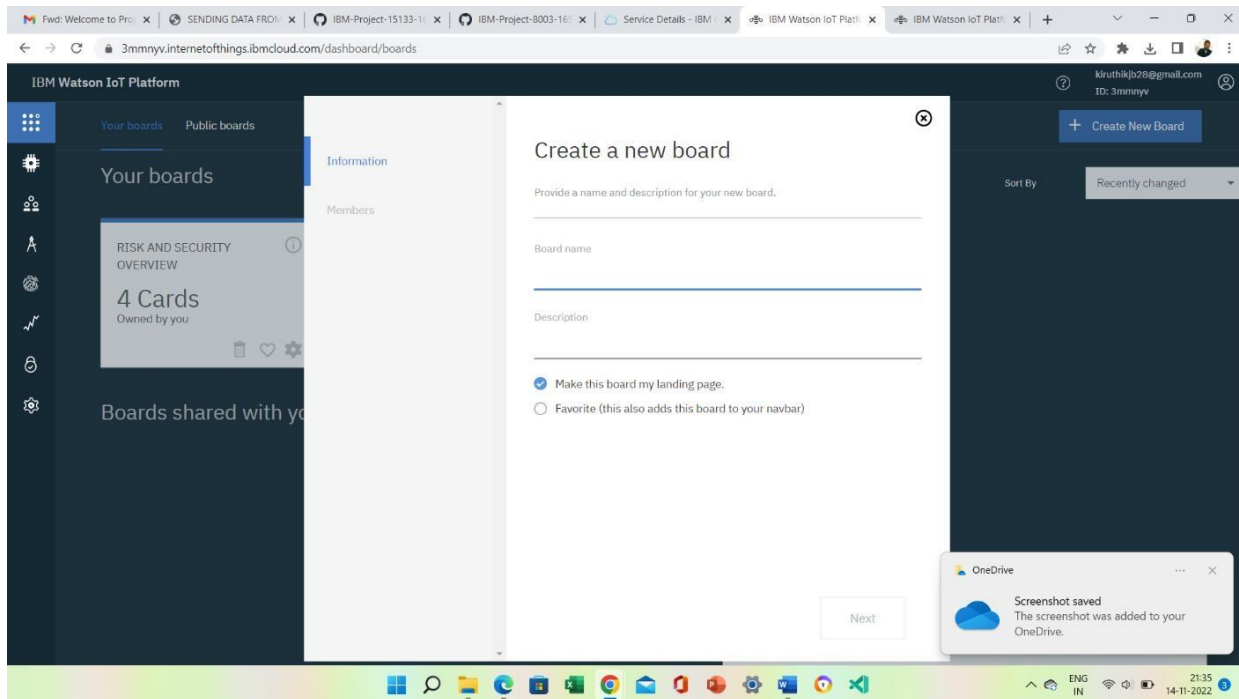
- In your Watson platform you have an option called board .Click on it and you get the following window on your screen

The screenshot displays the IBM Watson IoT Platform interface for creating boards. The top navigation bar includes 'Your boards' and 'Public boards'. The main content area shows a list of boards. Two boards are visible: 'RISK AND SECURITY OVERVIEW' with 4 cards and 'USAGE OVERVIEW' with 3 cards. A large dashed box with a plus sign indicates where to click to create a new board.

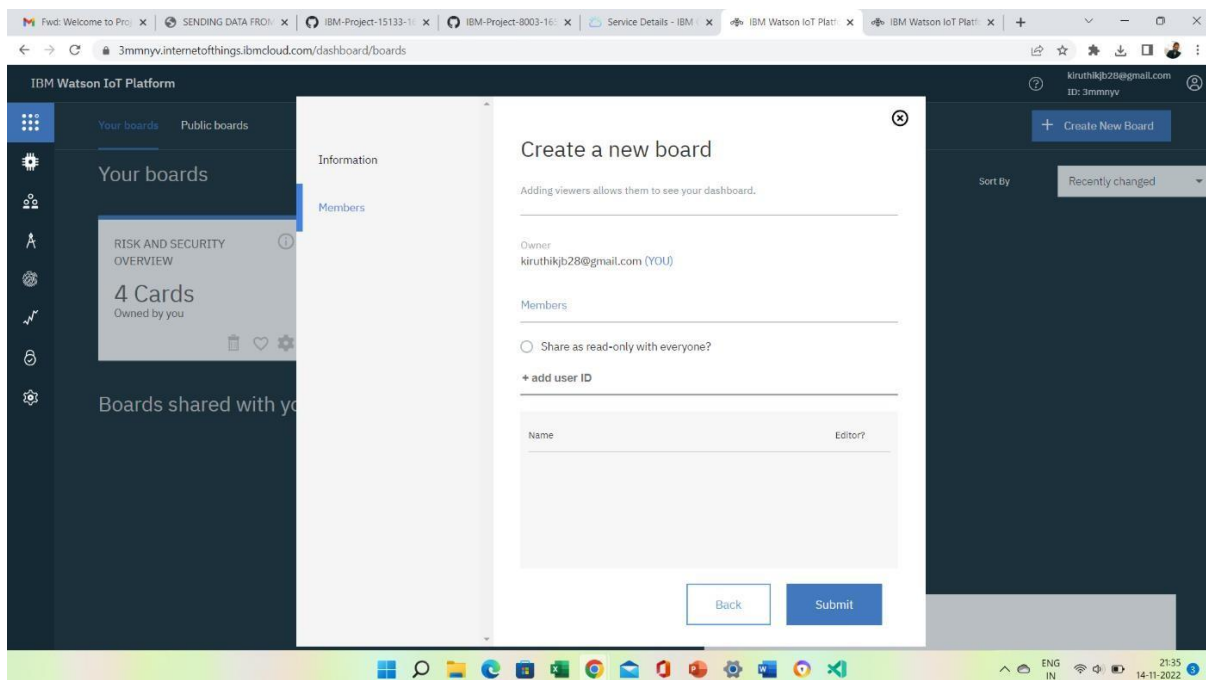
5 Simulations running

- Click on Create a new board to create a board .

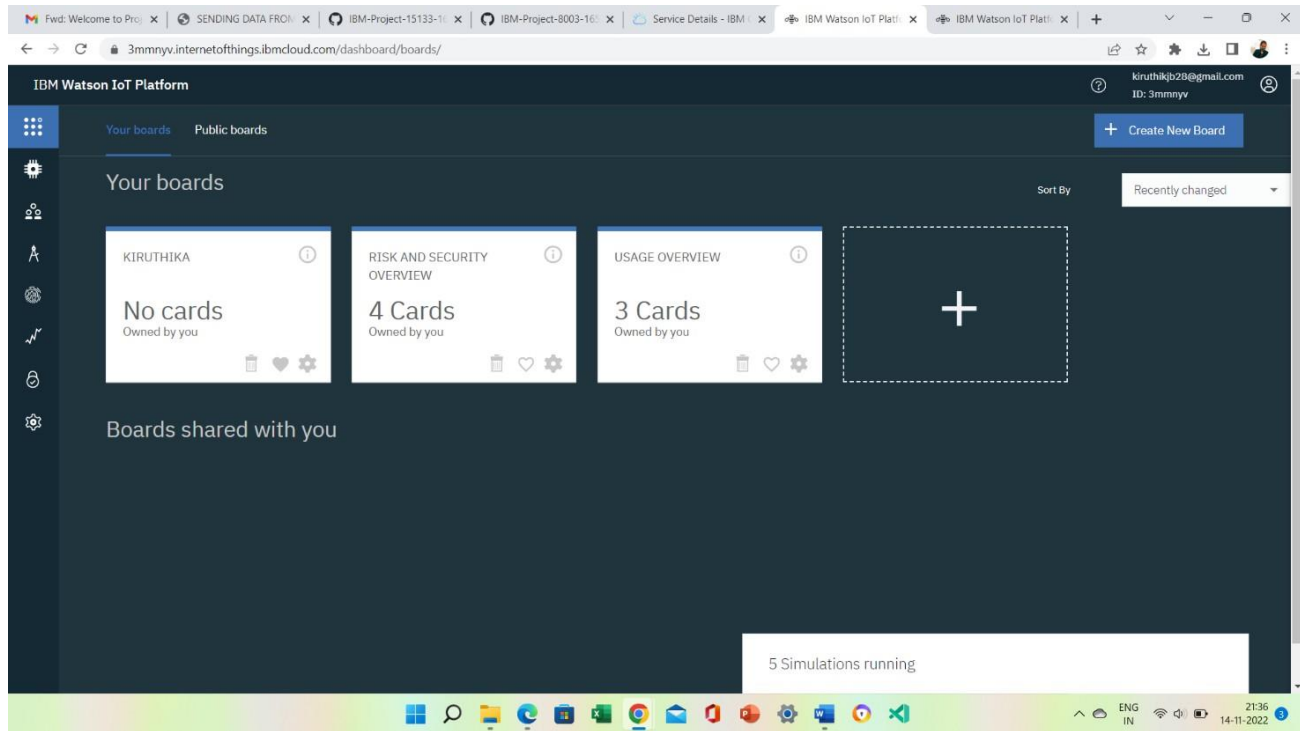
The given below window appears give a name and description to your board as shown in the window below.



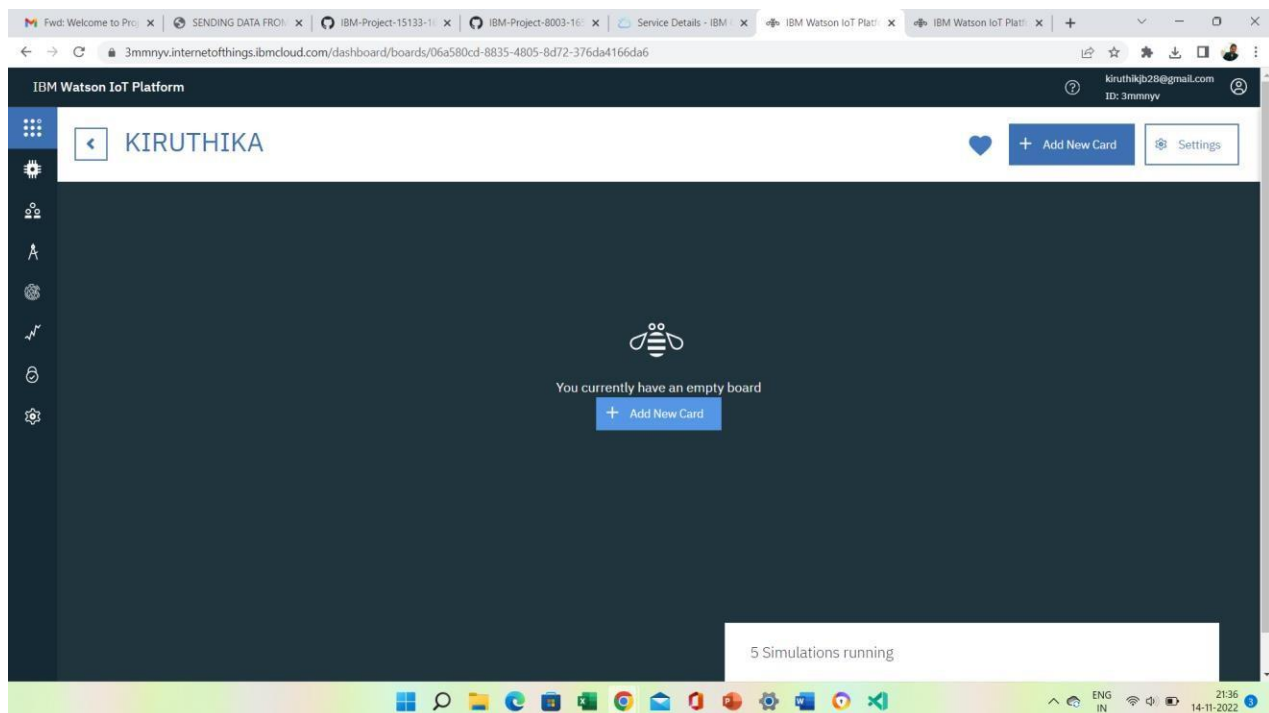
- Then click on Next you get the below window then again click on Submit



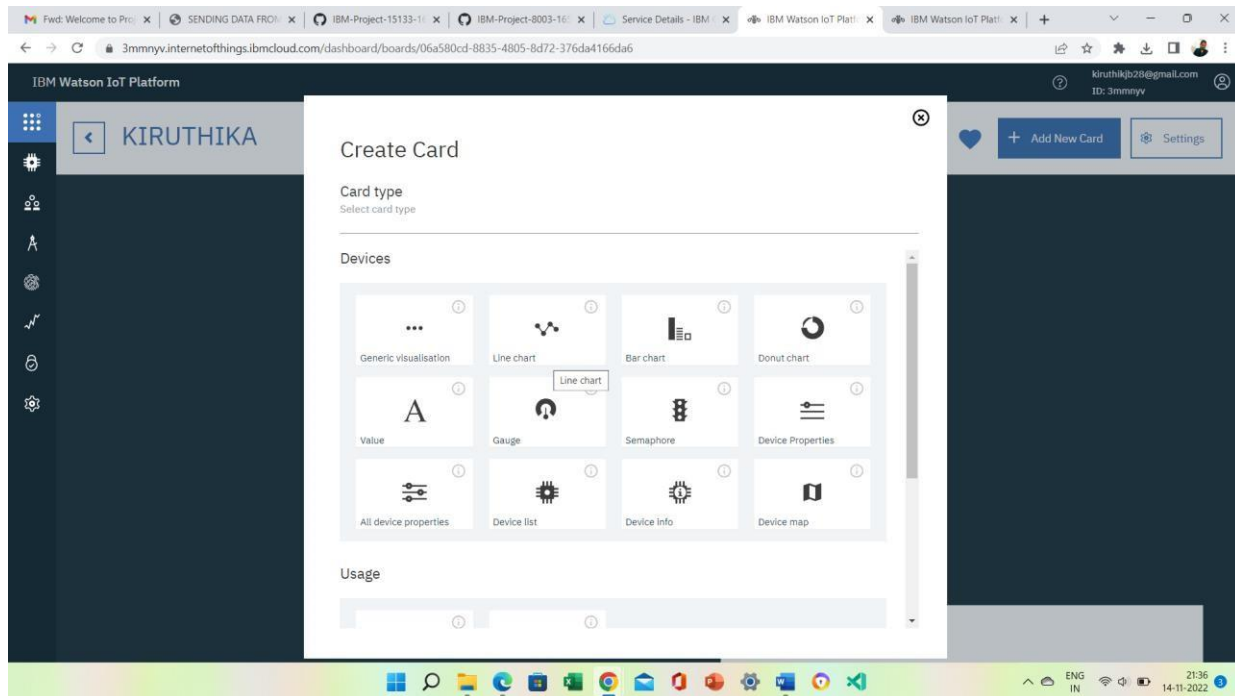
- Then double click on your boards name which you have created.



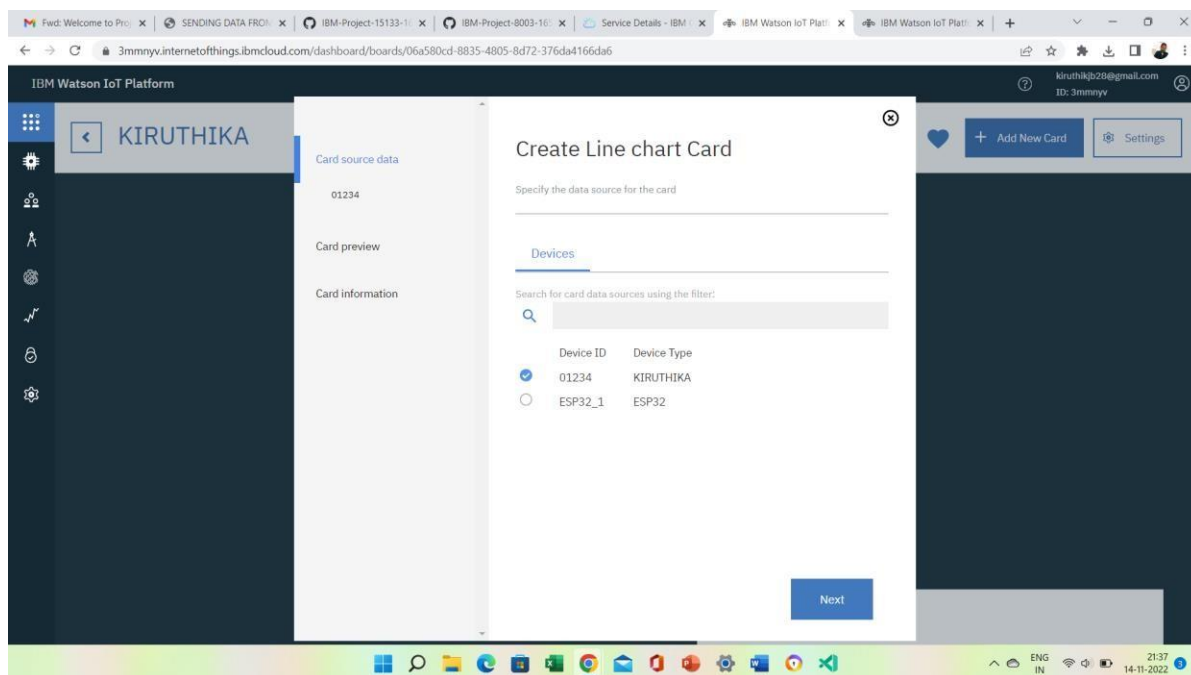
- Click on Add New Card



- Select the type of Graph u want accordingly and click next



- You get the below window, choose the Device and click on Next.





- Select the event, properly to be visualized on your graph and click next. In my case it is humidity

IBM Watson IoT Platform

KIRUTHIKA

Card source data

01234

Card preview

Card information

### Create Line chart Card

Connect data set

Temperature

Event

event 1

Property

Temperature

Name

Temperature

Type

Number

Unit

100

Min

0

Max

100

Back Next

- Then select the size of the graph and color of the graph board you want and click next

IBM Watson IoT Platform

KIRUTHIKA

Card source data

01234

Card preview

Card information

### Create Line chart Card

Enter title and description of the card

Title

Line chart

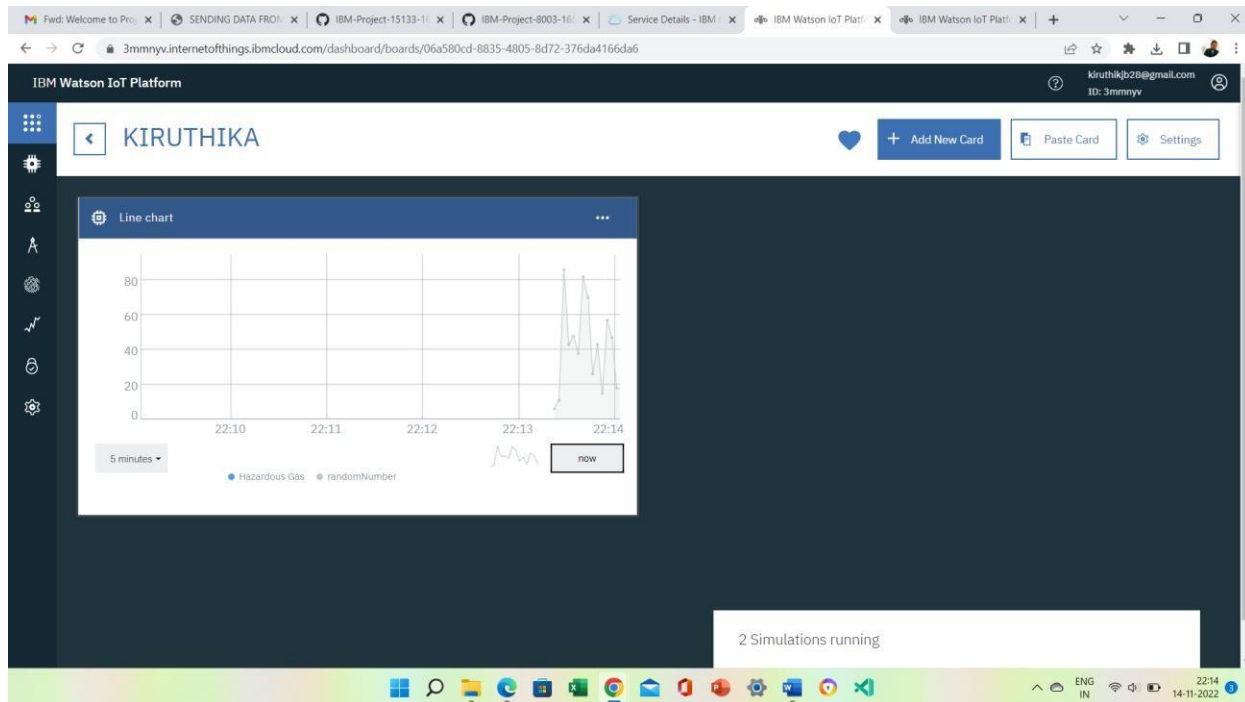
Color scheme

A line chart to display time series information with historic and live data

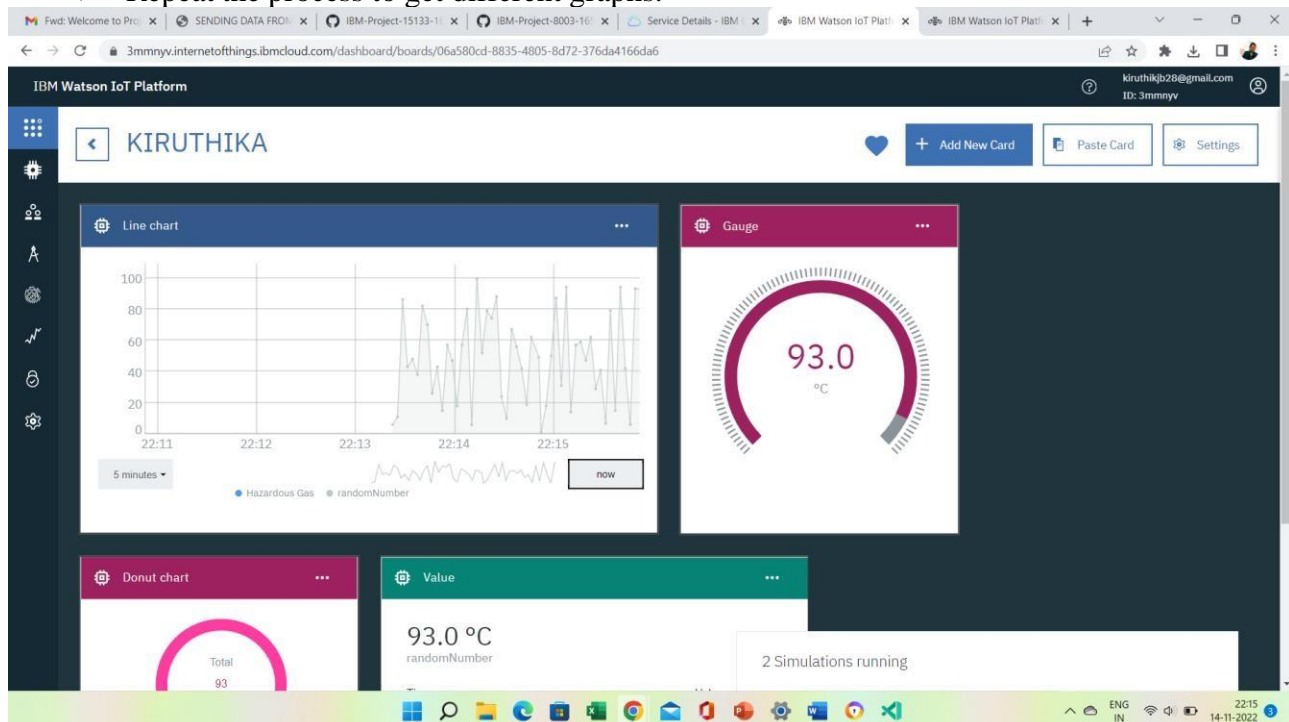
Back Submit

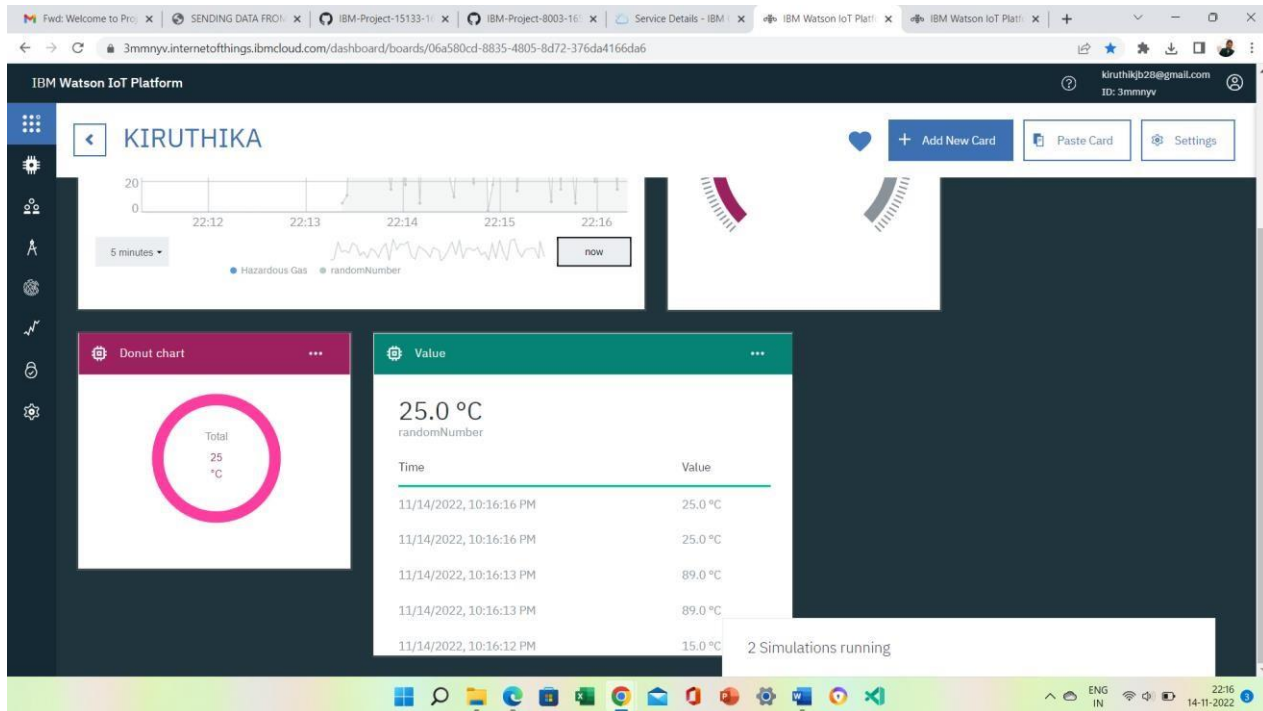


➤ Here is the graph



➤ Repeat the process to get different graphs.





## RESULT:

Hence, we were able to send data from our pi to IBM Watson and visualize it on a graph.