

SENDING DATA FROM RASPBERRY-PI TO IBM WATSON

Date	3 NOVEMBER 2022
Team ID	PNT2022TMID05277
Project Name	GAS LEAKAGE MONITORING AND ALERTING SYSTEM FOR INDUSTRIES

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

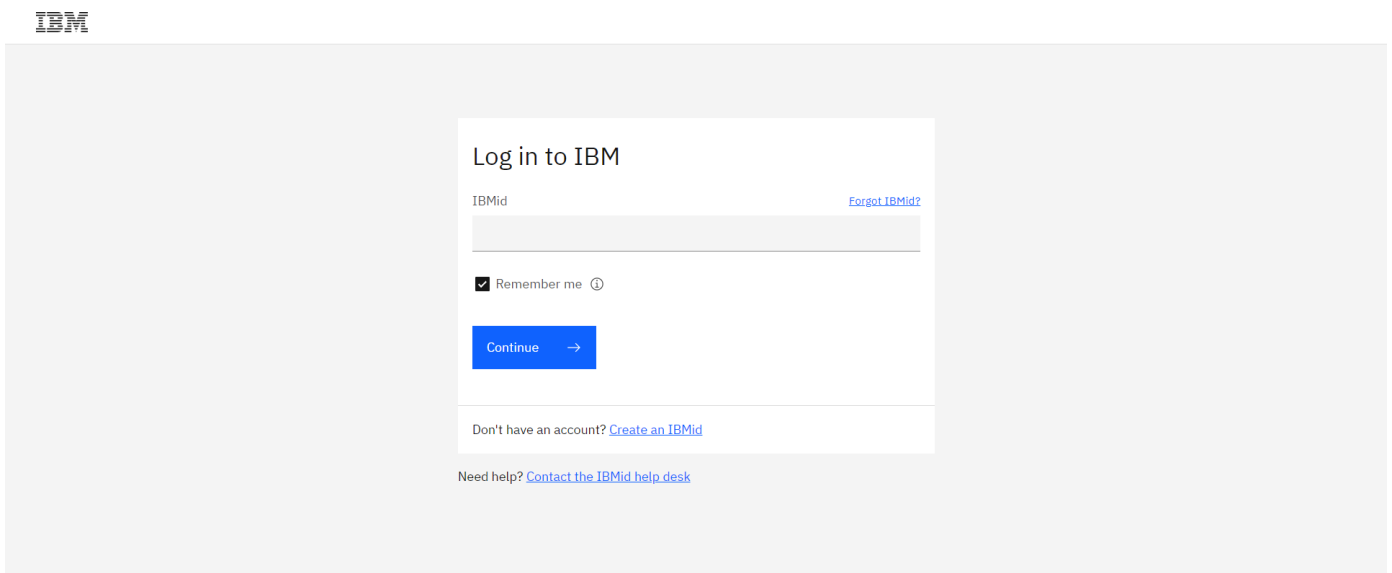
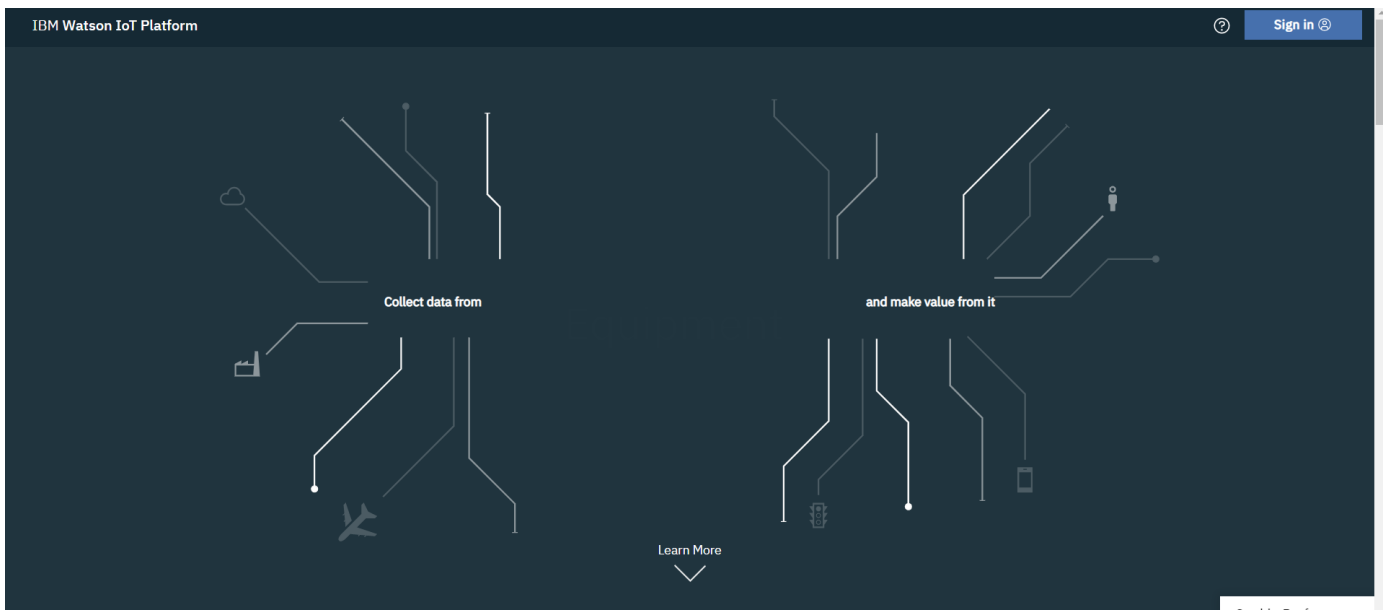
SOFTWARE:

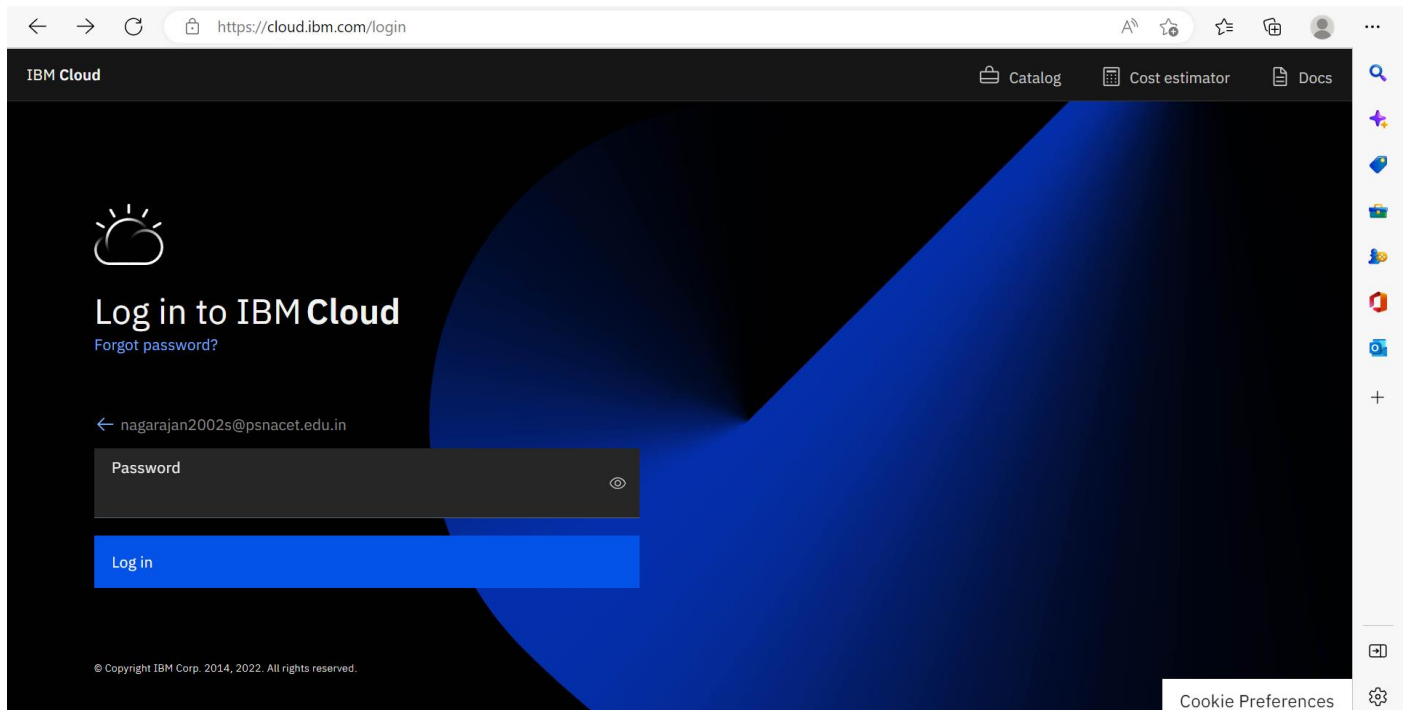
- IBM BLUEMIX ACCOUNT

STEPS TO BE FOLLOWED

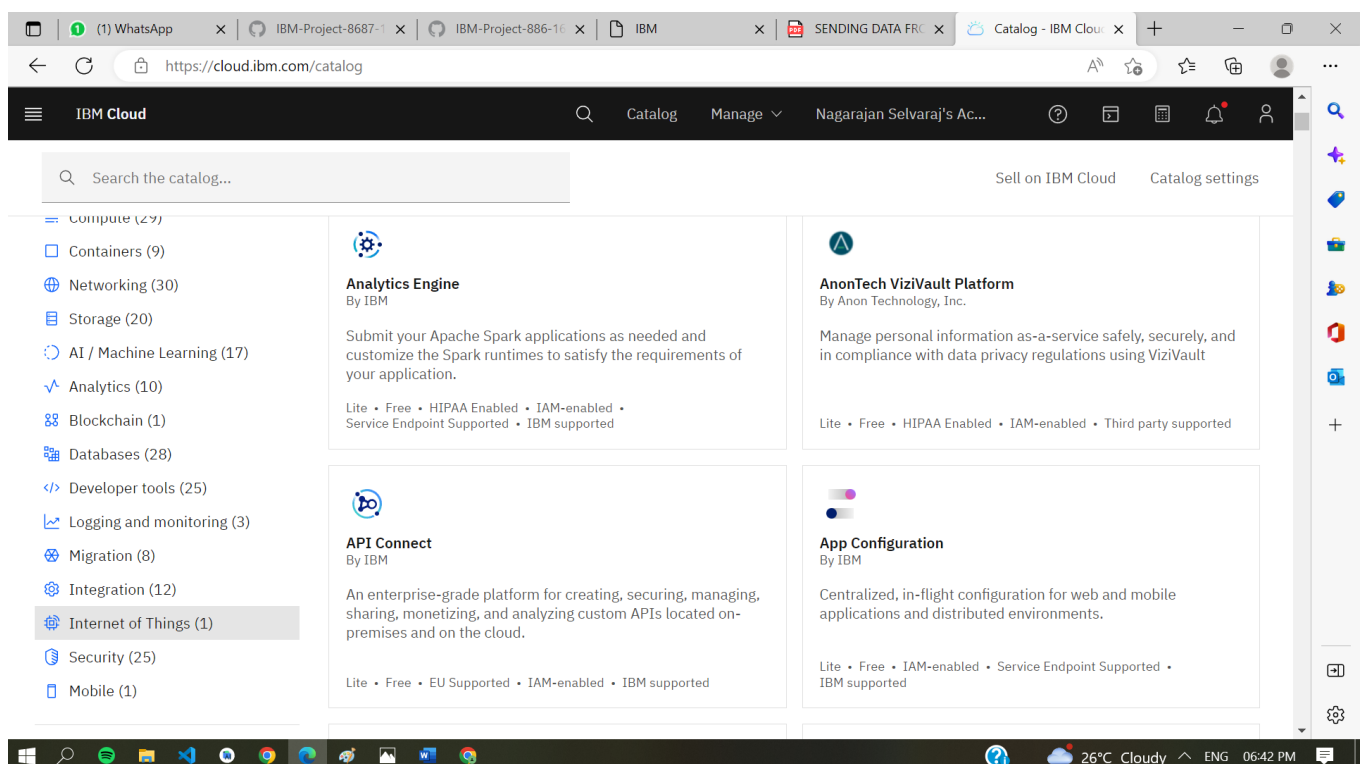
Step-1: Create a device in IBM Watson:

- 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' service. The page is divided into a left sidebar, a main content area, and a right summary panel.

Left Sidebar:

- Type: Service
- Provider: IBM
- Last updated: 08/15/2022
- Category: Internet of Things
- Compliance: IAM-enabled
- Location: Frankfurt

Main Content Area:

- Create** (selected) | About
- Select a location: Frankfurt (eu-de)
- Select a pricing plan: Displayed prices do not include tax. Monthly prices shown are for country or location: [United States](#)

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	Free

Right Summary Panel:

- Summary**
- Internet of Things Platform** **Free**
- Location: Frankfurt
- Plan: Lite
- Service name: Internet of Things Platform-0g
- Resource group: Default
- ☒ I have read and agree to the following license agreements: [Terms](#)
- Create** (button)
- Add to estimate (button)

- click on Launch

The screenshot shows the IBM Cloud Service Details page for the 'Internet of Things Platform-0g' service. The page is divided into a left sidebar, a main content area, and a right summary panel.

Left Sidebar:

- Manage** (selected)
- Plan
- Connections

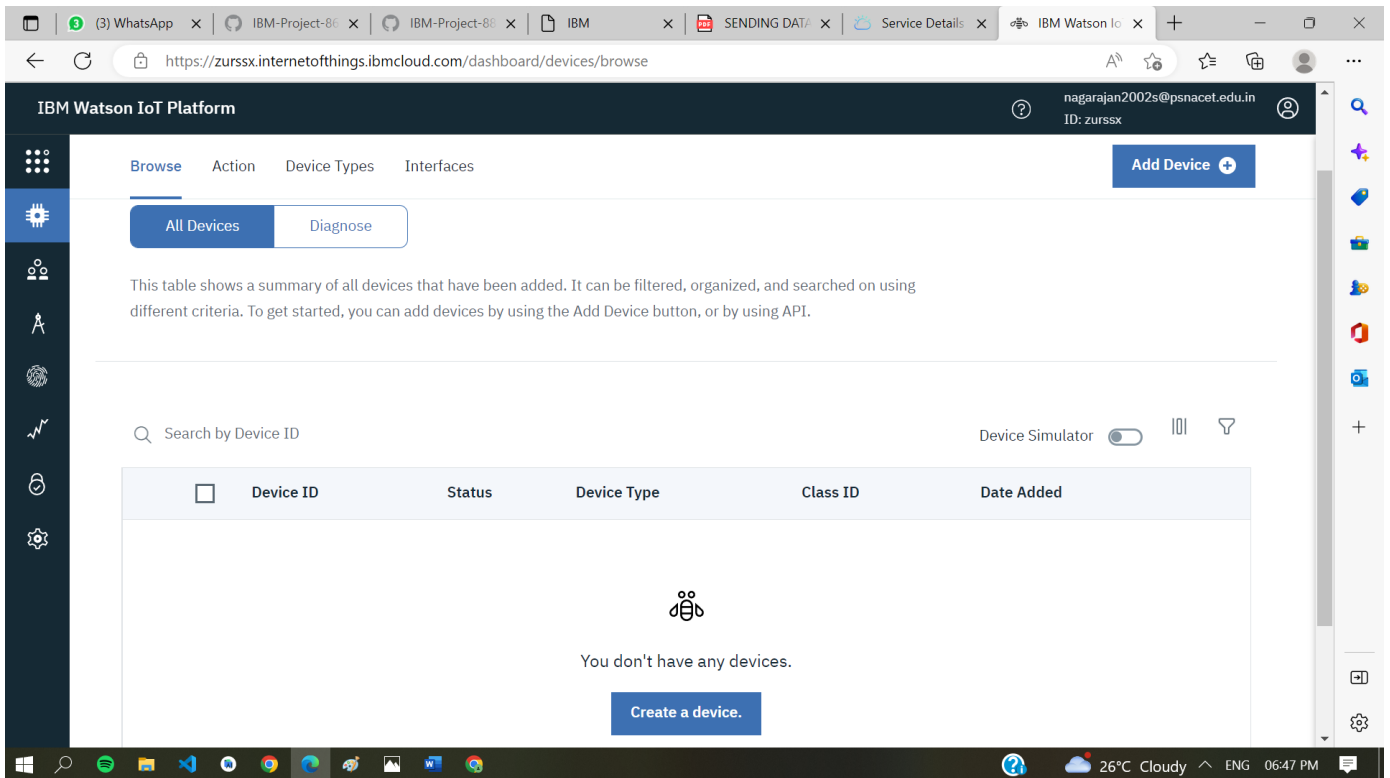
Main Content Area:

- Internet of Things Platform-0g** Active [Add tags](#)
- [Details](#) [Actions...](#)
- Let's get started with IBM Watson IoT Platform**
- Securely connect, control, and manage devices. Quickly build IoT applications that analyze data from the physical world.
- Launch** (button) **Docs** (button)
- Ready for the next level?**
- IBM Watson IoT Platform Journey**
- ☒ **Lite**
- ☐ **Non-Production**

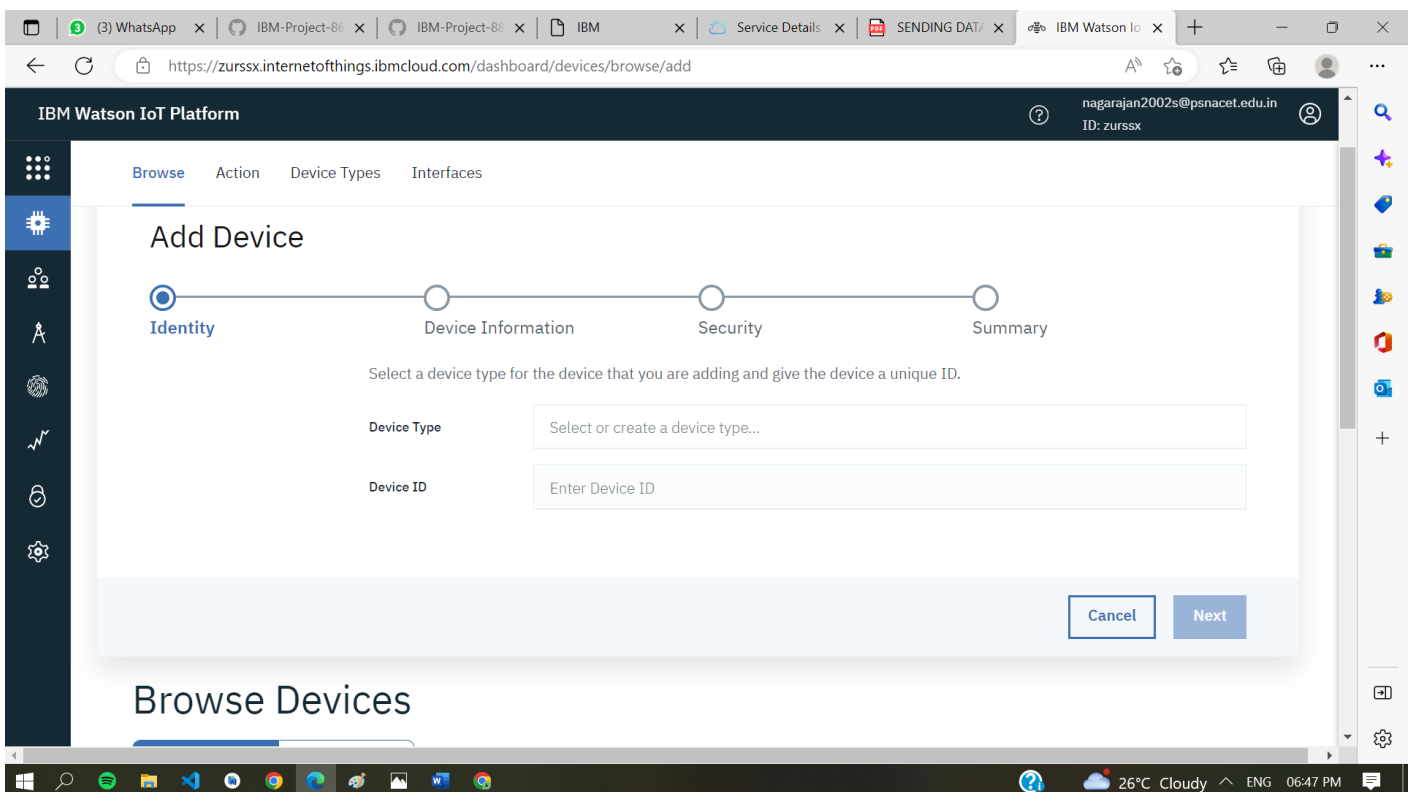
Right Summary Panel:

- Summary**
- Internet of Things Platform** **Free**
- Location: Frankfurt
- Plan: Lite
- Service name: Internet of Things Platform-0g
- Resource group: Default
- ☒ I have read and agree to the following license agreements: [Terms](#)
- Create** (button)
- Add to estimate (button)

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



- After click on Add device this page will open



- Go to device type and fill the details.

IBM Watson IoT Platform

Browser 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 Or

Name

The device type name is used to identify the device type uniquely and uses a restricted set of characters to make it suitable for API use.

Description

- Click on Finish

IBM Watson IoT Platform

Browser Action **Device Types** Interfaces

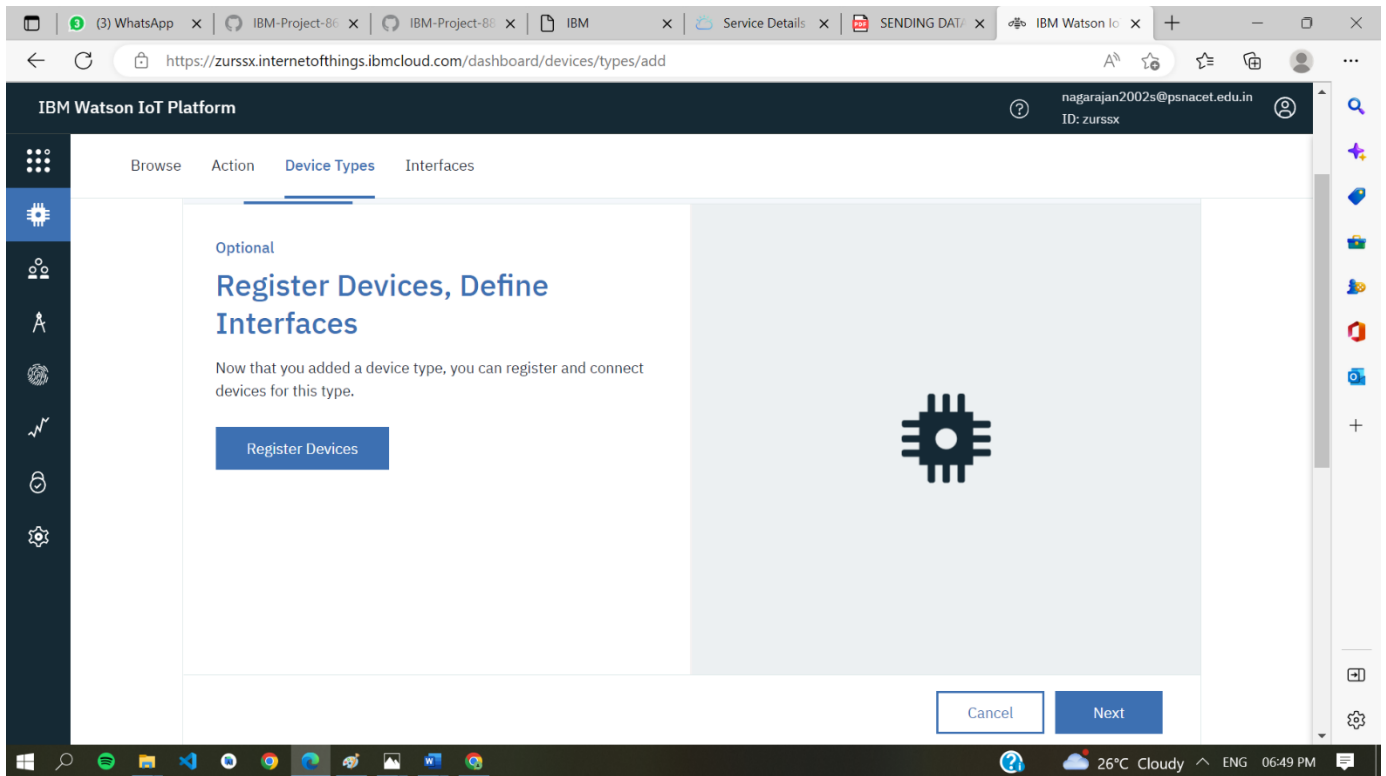
Add type

Identity **Device Information**

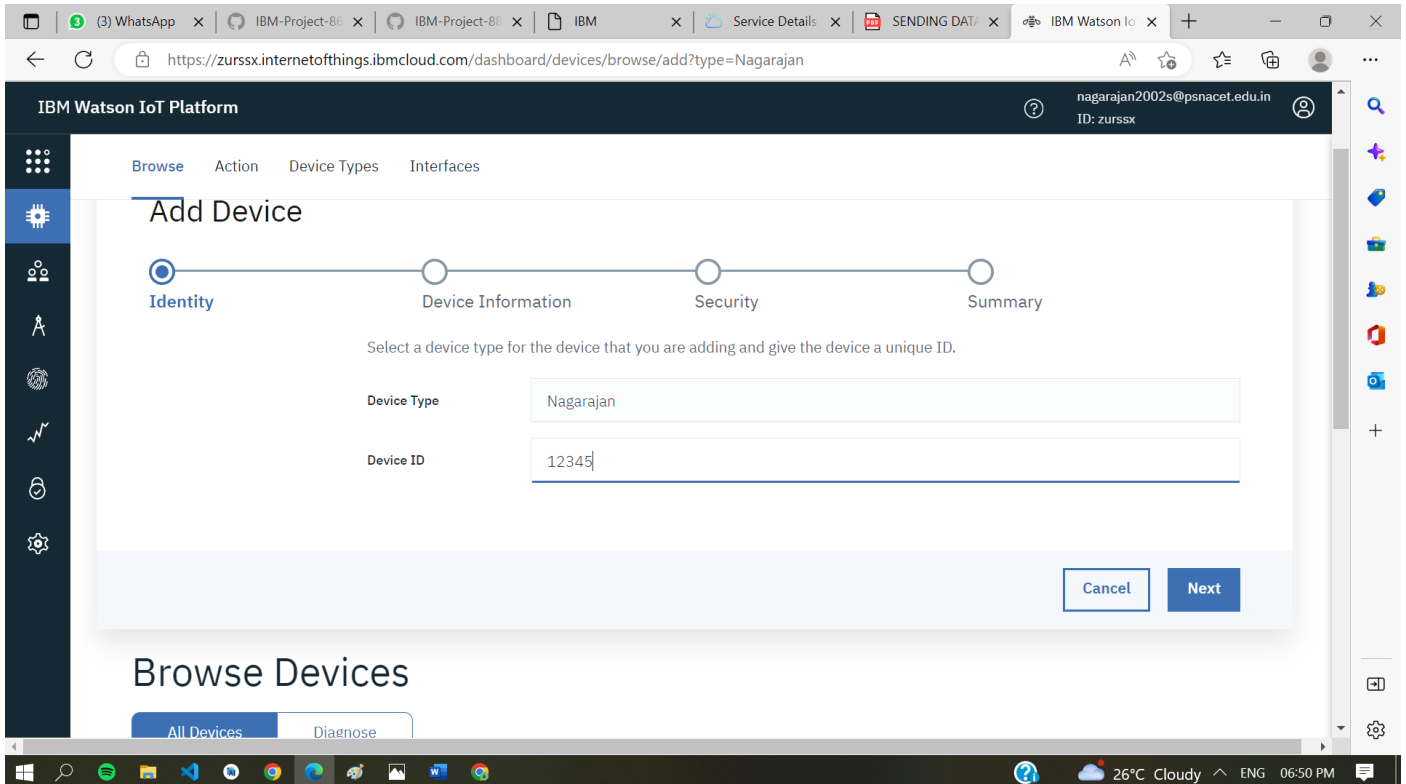
These attributes will be used as a template for new devices that are assigned this device type

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"/>

➤ 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 two columns of input fields for device metadata.

Field	Value
Serial Number	Enter Serial Number
Manufacturer	Enter Manufacturer
Model	Enter Model
Device Class	Enter Device Class
Description	Enter Description
Firmware Version	Enter Firmware Version
Hardware Version	Enter Hardware Version
Descriptive Location	Enter Descriptive Location

At the bottom left, there is a button labeled 'Add Metadata +'. The top navigation bar shows the user is logged in as 'nagarajan2002s@psnacet.edu.in' with ID 'zurssx'.

➤ Click on Next

The screenshot shows the 'Security' step of the 'Add Device' process. The progress bar shows 'Identity' and 'Device Information' as completed steps, with 'Security' being the current step. The page explains two options for selecting a device authentication token.

Auto-generated authentication token (default)

There are two options for selecting a device authentication token.

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 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.

Authentication Token

Enter an optional token

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.

➤ Click on Finish

IBM Watson IoT Platform

https://zurssx.internetofthings.ibmcloud.com/dashboard/devices/browse/add?type=Nagarajan

nagarajan2002s@psnacet.edu.in
ID: zurssx

Browse Action Device Types Interfaces

Add Device

Identity Device Information Security Summary

Verify that the following information is correct then select Finish

Device Type
Nagarajan

Device ID
12345

View Metadata

Security Token
To be generated

Back Finish

➤ Device is created

IBM Watson IoT Platform

https://y0ddy.internetofthings.ibmcloud.com/dashboard/devices/browse

nagarajan2002s@psnacet.edu.in
ID: zurssx

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
12345	Disconnected	Nagarajan	Device	Oct 31, 2022 11:38 AM	

Items per page 50 | 1-1 of 1 item

1 of 1 page

1 Simulation running

Activate Windows
Go to Settings to activate Windows.

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/libssl1.0.0_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 ... 115686 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
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[2562]: 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
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 1.6MB/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 (122kB)
    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 1.6MB/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/45/f9/45/bbc33ad957e82f7b71ba80e31d6d5a83d9d735a6d12e0c9418
Running setup.py bdist_wheel for dicttoxml ... done
Stored in directory: /home/pi/.cache/pip/wheels/20/d8/0d/acdc8f289011b7be7de71deebe642f683be0313dfff0493
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:

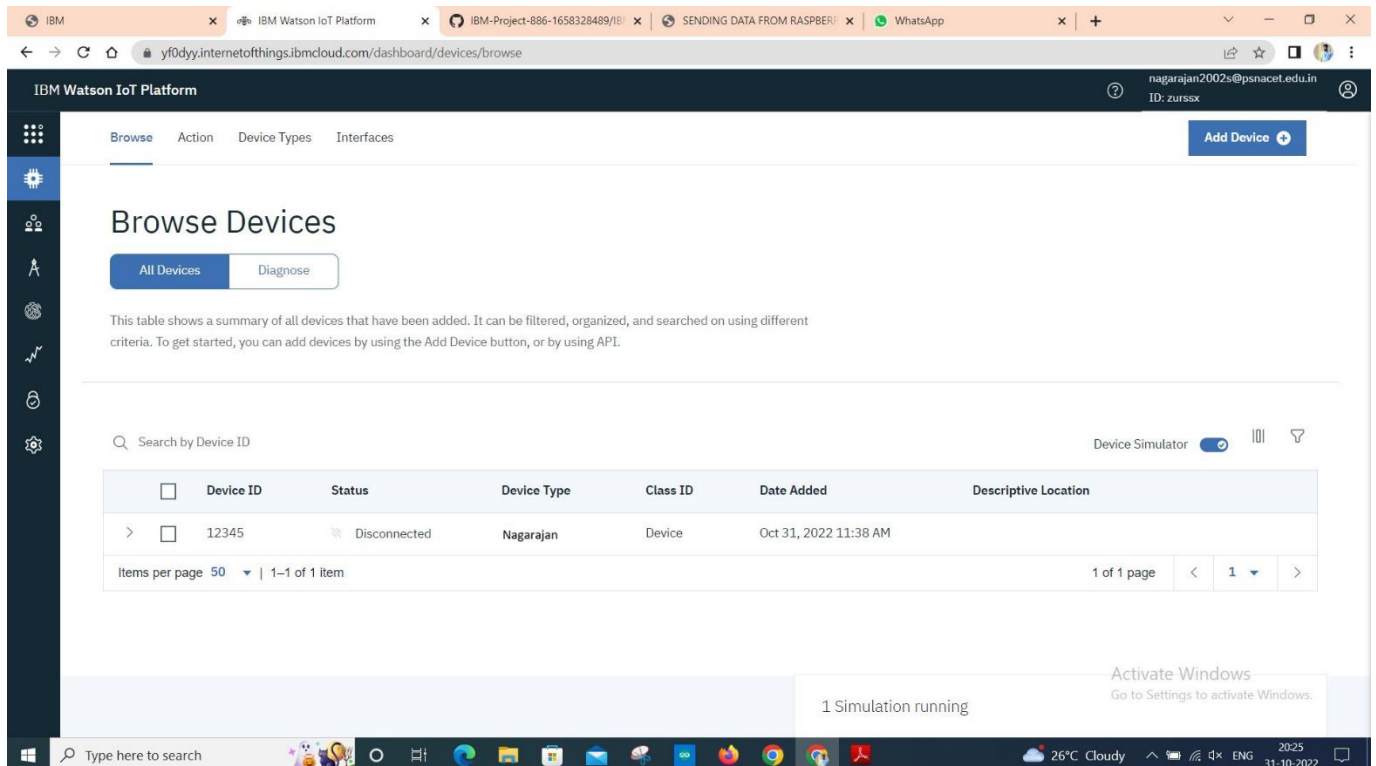
```

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 07:10:37,768 ibmiotf.device.Client INFO Connected successfully: d:gegtl4:mydevice:mydevice
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 = 28 C Humidity = 50 % to IBM Watson
Published Temperature = 28 C Humidity = 50 % to IBM Watson
Published Temperature = 28 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'. A sidebar on the left contains various icons. The main content area shows a table with columns: Device ID, Status, Device Type, Class ID, Date Added, and Descriptive Location. The first row is selected, showing device ID 12345, status 'Disconnected', device type 'Nagarajan', class ID 'Device', and date added 'Oct 31, 2022 11:38 AM'. Below the table, there are tabs for 'Identity', 'Device Information', 'Recent Events', 'State', and 'Logs'. The 'Recent Events' tab is active, showing a list of events with columns: Event, Value, Format, and Last Received. The events are listed as 'event_1' with values like '{"Hazardous Gas":61,"Temperature":88,"Humidit...}' and format 'json', all received 'a few seconds ago'. A status bar at the bottom indicates '1 Simulation running'.

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
12345	Disconnected	Nagarajan	Device	Oct 31, 2022 11:38 AM	

Event	Value	Format	Last Received
event_1	{"Hazardous Gas":61,"Temperature":88,"Humidit...	json	a few seconds ago
event_1	{"Hazardous Gas":20,"Temperature":36,"Humidit...	json	a few seconds ago
event_1	{"Hazardous Gas":79,"Temperature":56,"Humidit...	json	a few seconds ago
event_1	{"Hazardous Gas":52,"Temperature":82,"Humidit...	json	a few seconds ago
event_1	{"Hazardous Gas":26,"Temperature":33,"Humidit...	json	a few seconds ago

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 shows the 'Your boards' section of the IBM Watson IoT Platform. The top navigation bar includes 'Your boards' and 'Public boards'. A sidebar on the left contains various icons. The main content area displays two boards: 'USAGE OVERVIEW' with 3 cards and 'RISK AND SECURITY OVERVIEW' with 4 cards. Both boards are owned by the user. A large dashed box with a plus sign is visible, indicating a placeholder for a new board. Below the boards, there is a section titled 'Boards shared with you'. A status bar at the bottom indicates '1 Simulation 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.

IBM Watson IoT Platform

Usage Overview

3 Cards

Owned by you

Boards shared with you

Information

Members

Create a new board

Provide a name and description for your new board.

Board name

Description

☒ Make this board my landing page.

☐ Favorite (this also adds this board to your navbar)

Next

Create New Board

Activate Windows

Go to Settings to activate Windows.

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

IBM Watson IoT Platform

Usage Overview

3 Cards

Owned by you

Boards shared with you

Information

Members

Create a new board

Adding viewers allows them to see your dashboard.

Owner

nagarajan2002s@psnacet.edu.in

Members

☐ Share as read-only with everyone?

+ add user ID

Name

Editor?

Back

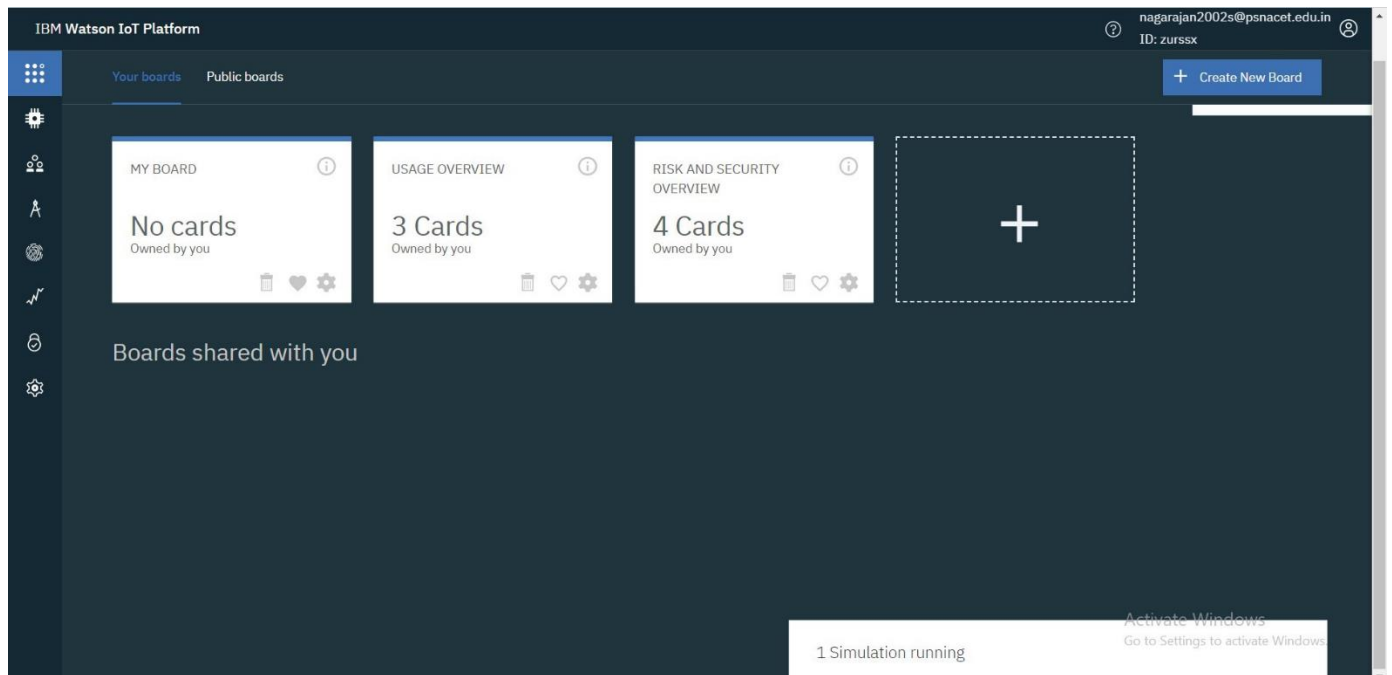
Submit

Create New Board

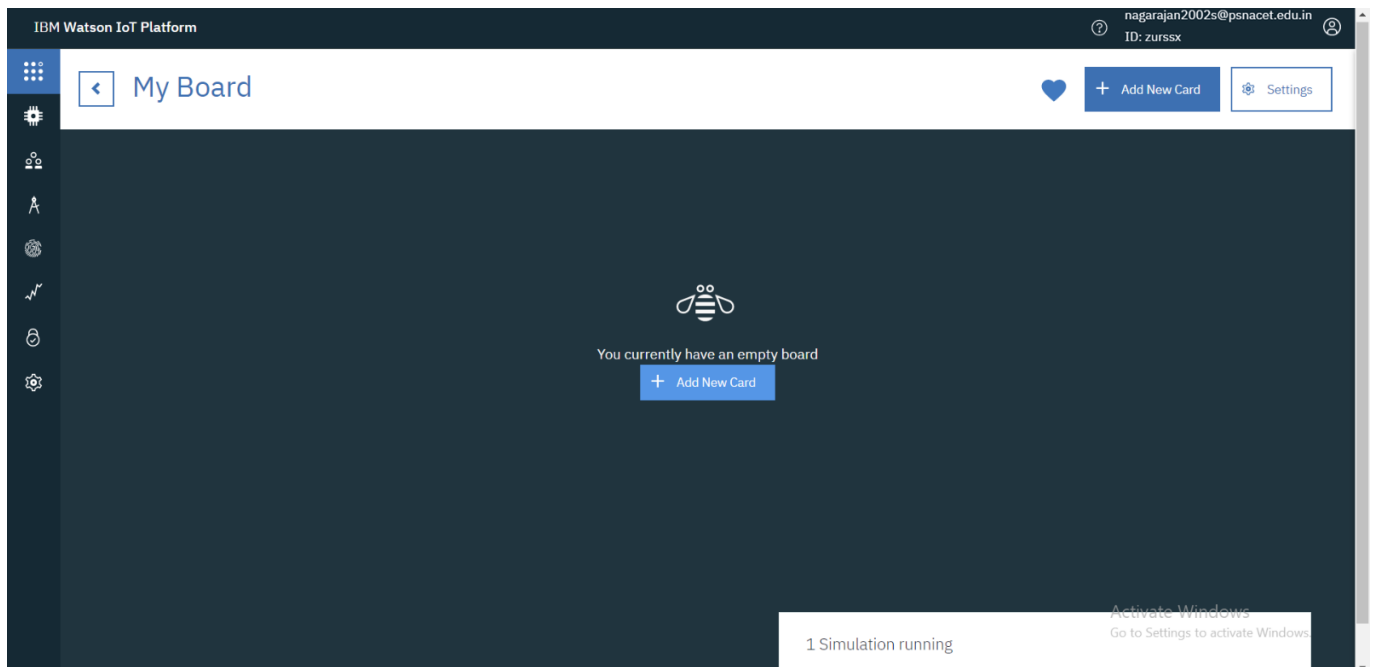
Activate Windows

Go to Settings to activate Windows.

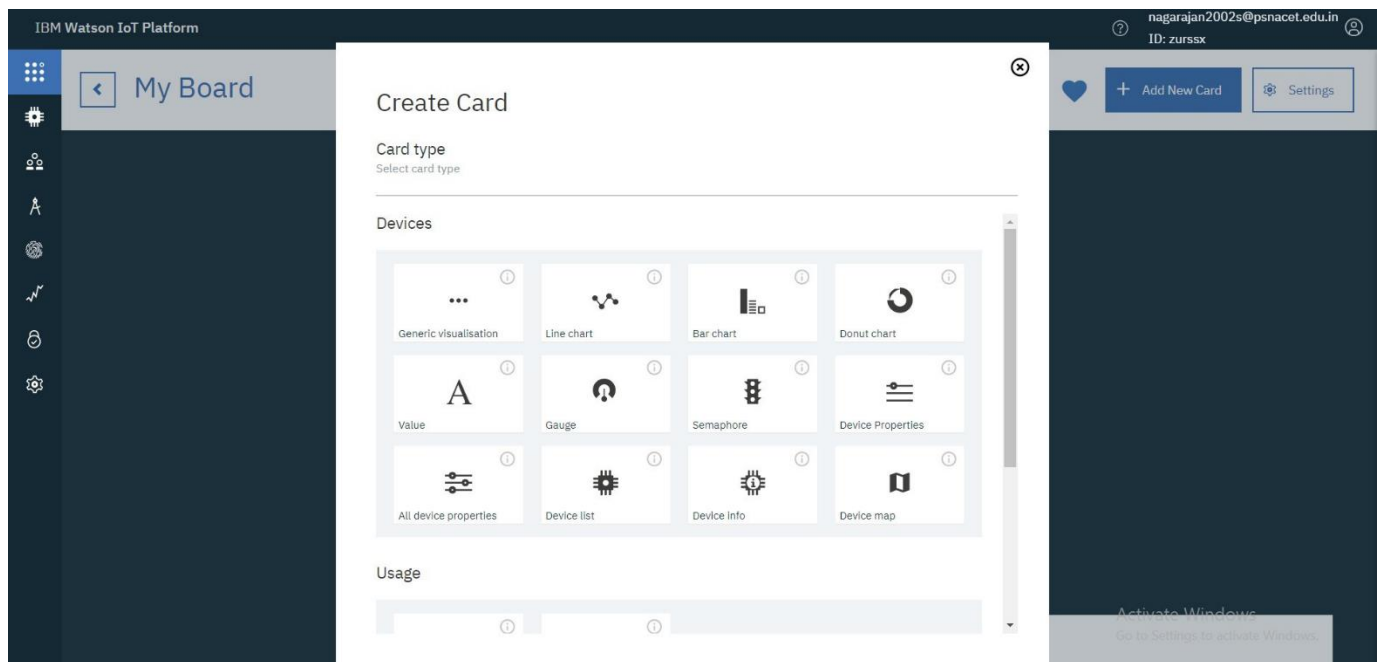
- Then double click on your boards name which you have created.



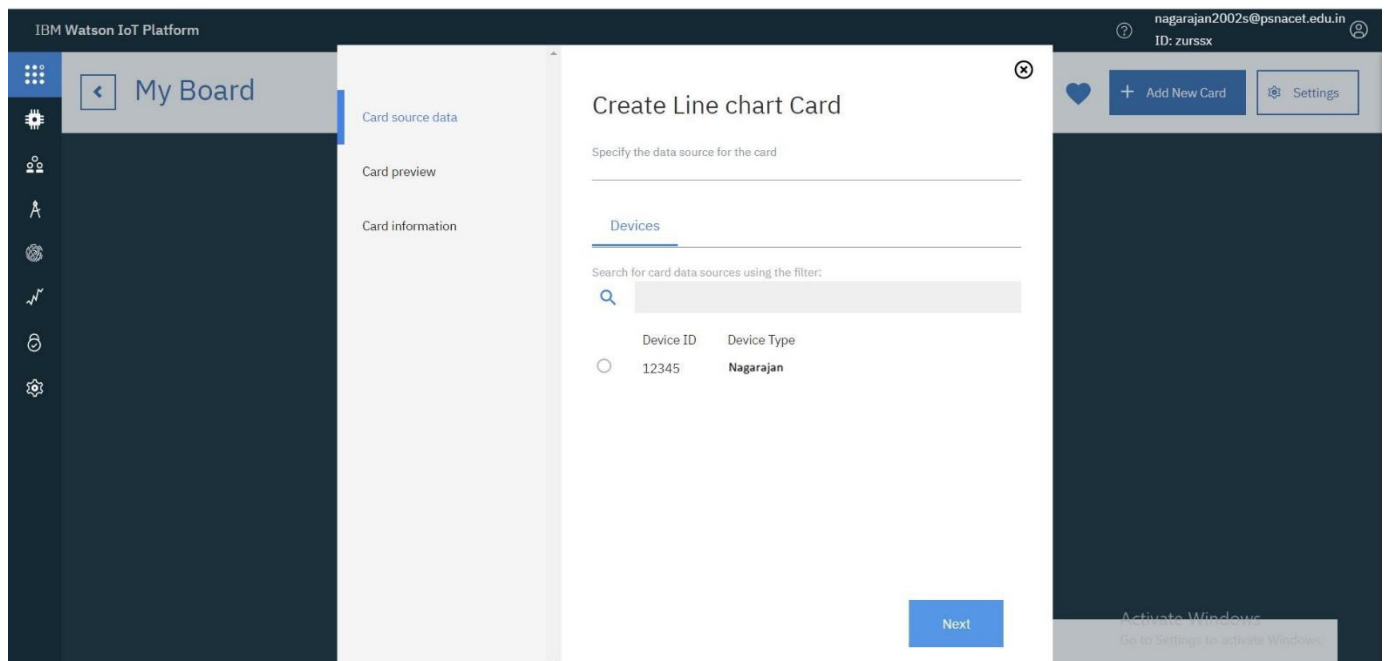
- Click on Add New Card



- 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

Temp & Hum

Card source data

12345

Card preview

Card information

Create Line chart Card

Connect data set

Temperature

Event

event_1

Property

Temperature

Name

Temperature

Type

Number

Unit

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

Temp & Hum

Card source data

12345

Card preview

Card information

Create Line chart Card

Enter title and description of the card

Title

Line chart

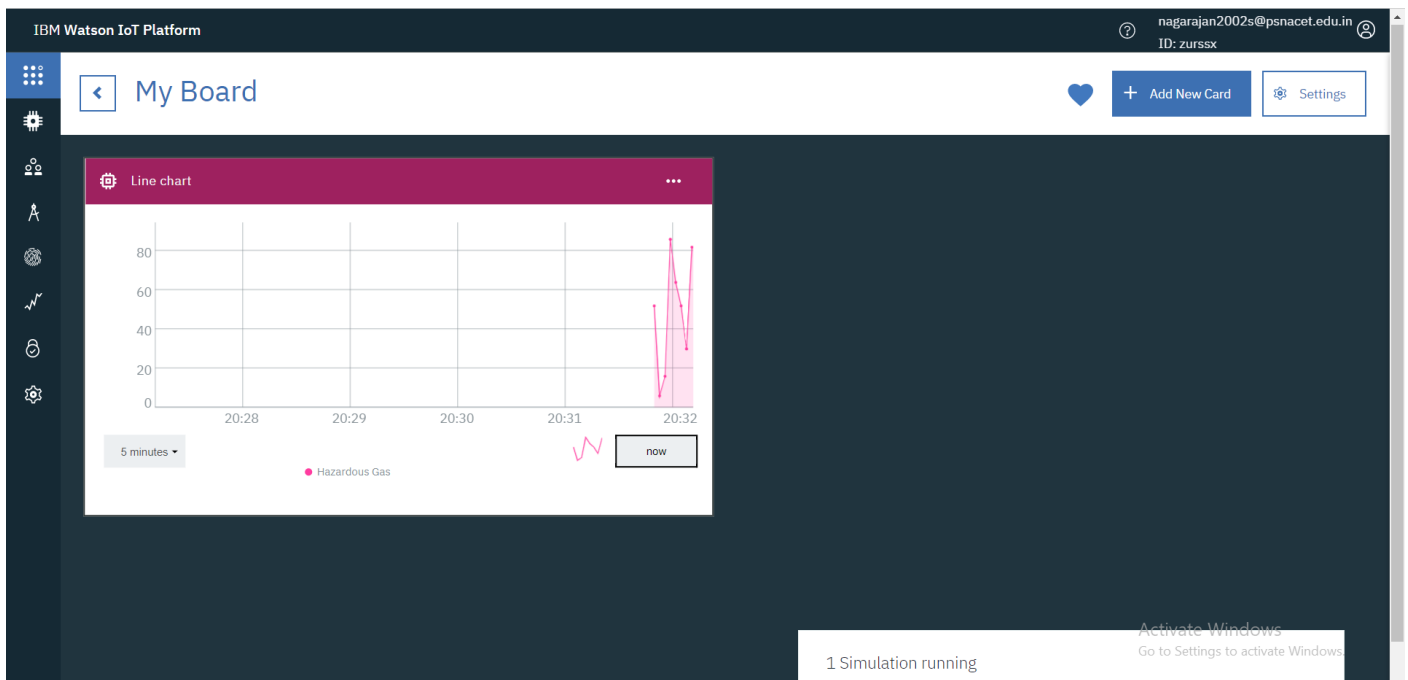
Color scheme

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

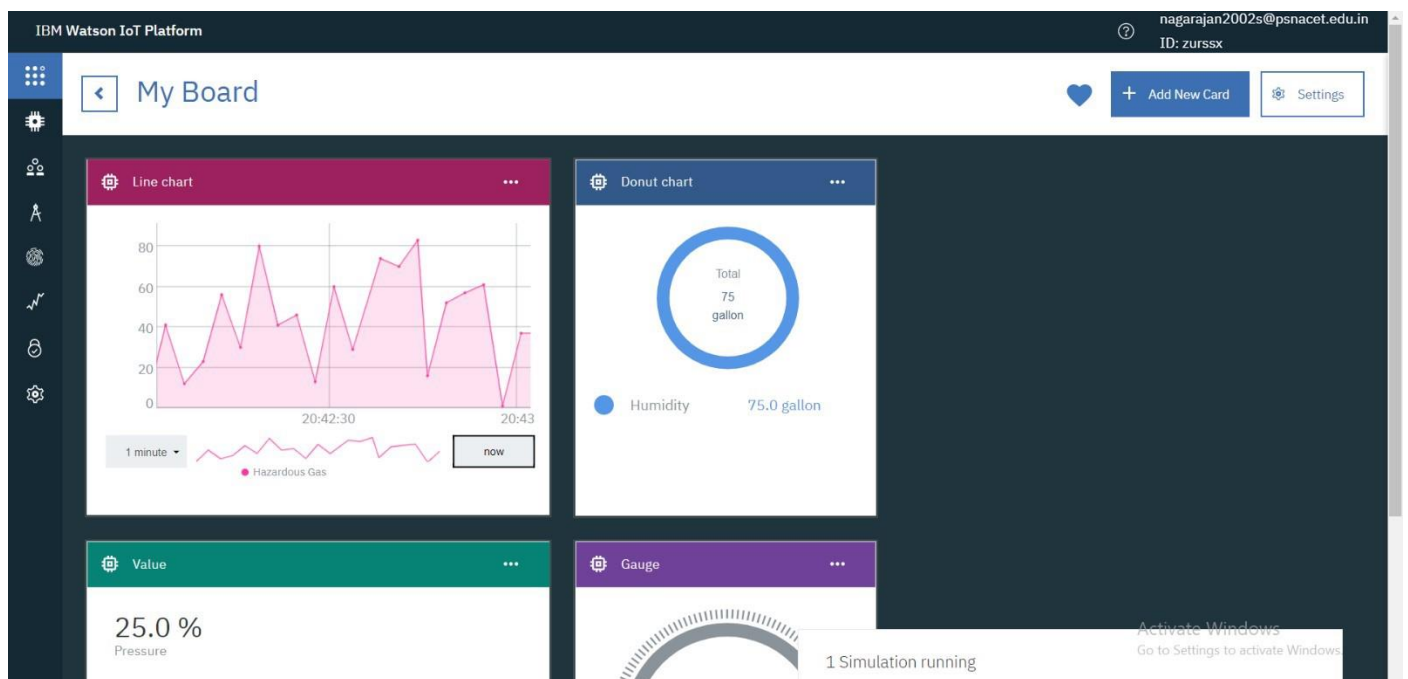
Back

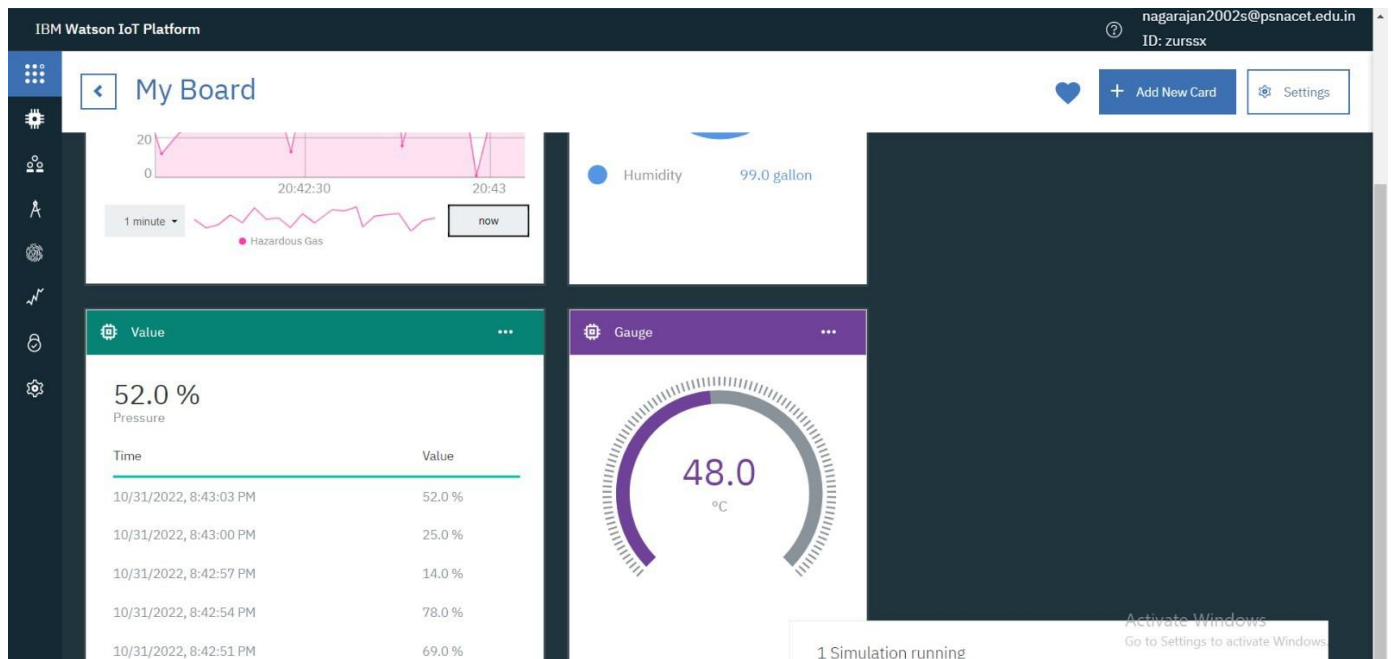
Submit

➤ 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.