

# SENDING DATA FROM RASPBERRY-PI TO IBM WATSON

<b>Date</b>	10 NOVEMBER 2022
<b>Team ID</b>	PNT2022TMID28330
<b>Project Name</b>	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

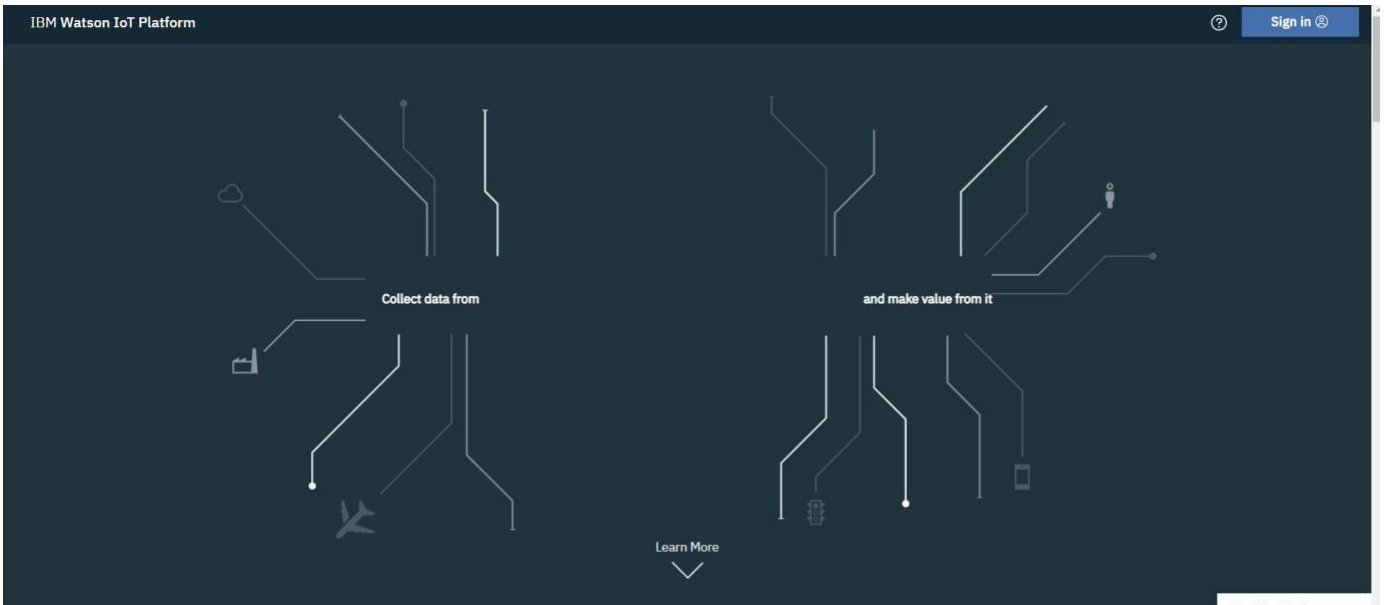
### SOFTWARE:

- 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.



IBM

## Log in to IBM

IBMid

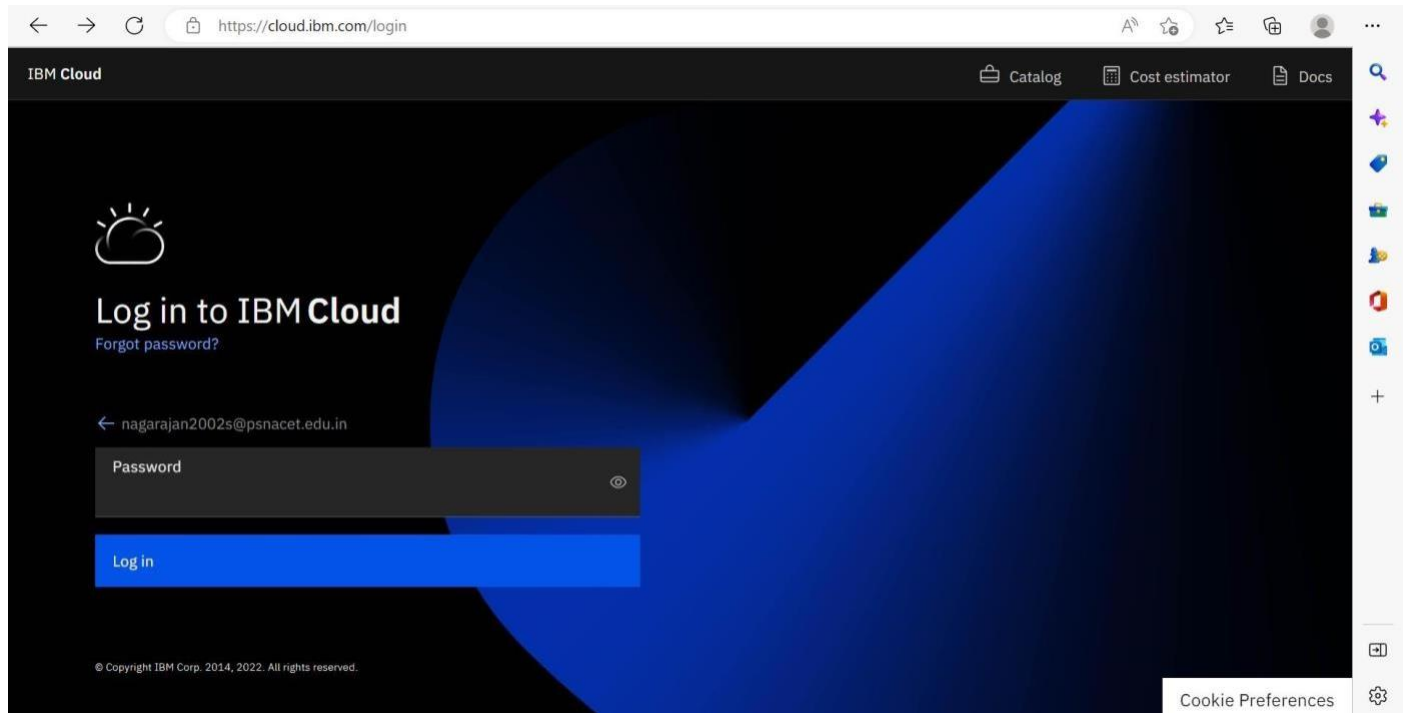
[Forgot IBMid?](#)

☒ Remember me ⓘ

Continue →

Don't have an account? [Create an IBMid](#)

Need help? [Contact the IBMid help desk](#)



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

IBM Cloud

Search the catalog...

Sell on IBM CloudCatalog settings

Computer (29)

Containers (9)

Networking (30)

Storage (20)

AI / Machine Learning (17)

Analytics (10)

Blockchain (1)

Databases (28)

Developer tools (25)

Logging and monitoring (3)

Migration (8)

Integration (12)

Internet of Things (1)

Security (25)

Mobile (1)

Analytics Engine

By IBM

Submit your Apache Spark applications as needed and customize the Spark runtimes to satisfy the requirements of your application.

Lite • Free • HIPAA Enabled • IAM-enabled • Service Endpoint Supported • IBM supported

API Connect

By IBM

An enterprise-grade platform for creating, securing, managing, sharing, monetizing, and analyzing custom APIs located on-premises and on the cloud.

Lite • Free • EU Supported • IAM-enabled • IBM supported

AnonTech ViziVault Platform

By Anon Technology, Inc.

Manage personal information as-a-service safely, securely, and in compliance with data privacy regulations using ViziVault

Lite • Free • HIPAA Enabled • IAM-enabled • Third party supported

App Configuration

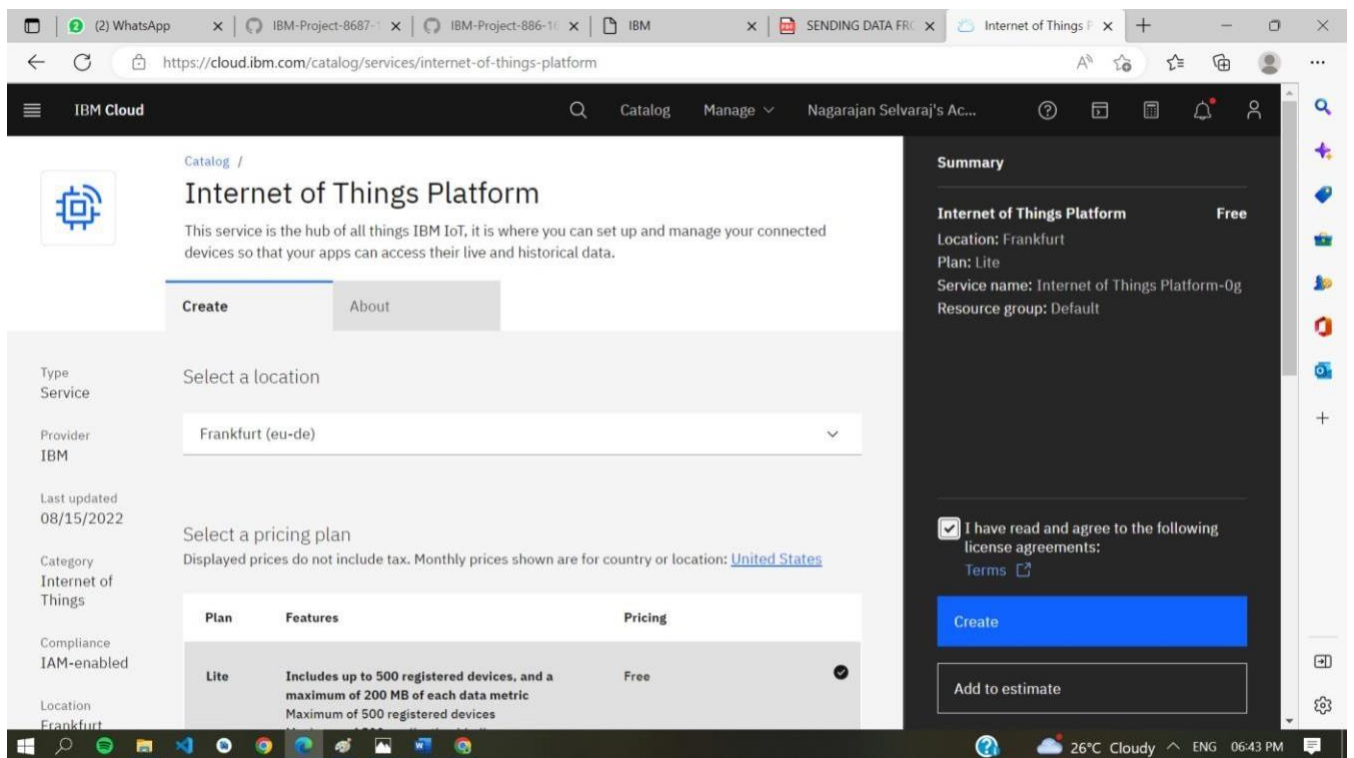
By IBM

Centralized, in-flight configuration for web and mobile applications and distributed environments.

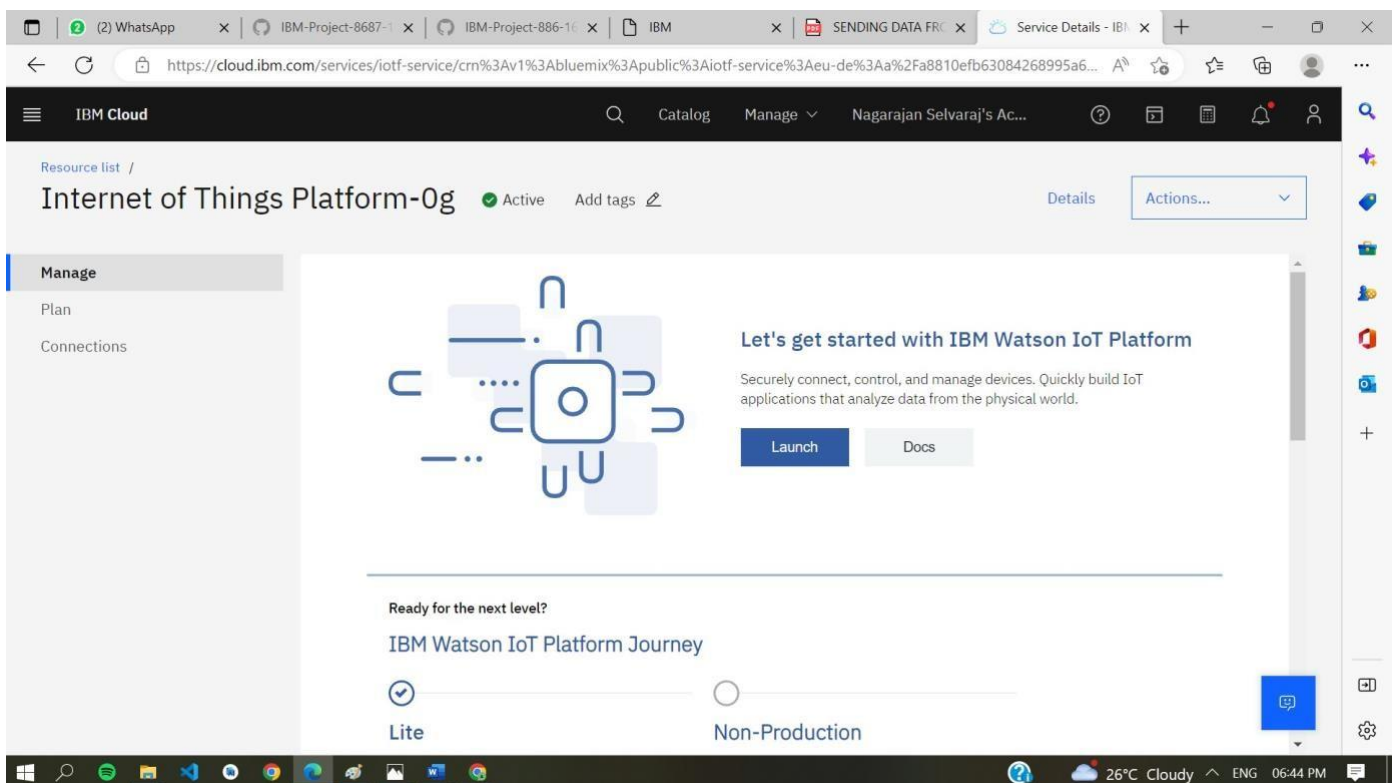
Lite • Free • IAM-enabled • Service Endpoint Supported • IBM supported

Windows taskbar with icons for WhatsApp, IBM projects, and system status (26°C Cloudy, 06:42 PM).

Check all details and click on create.

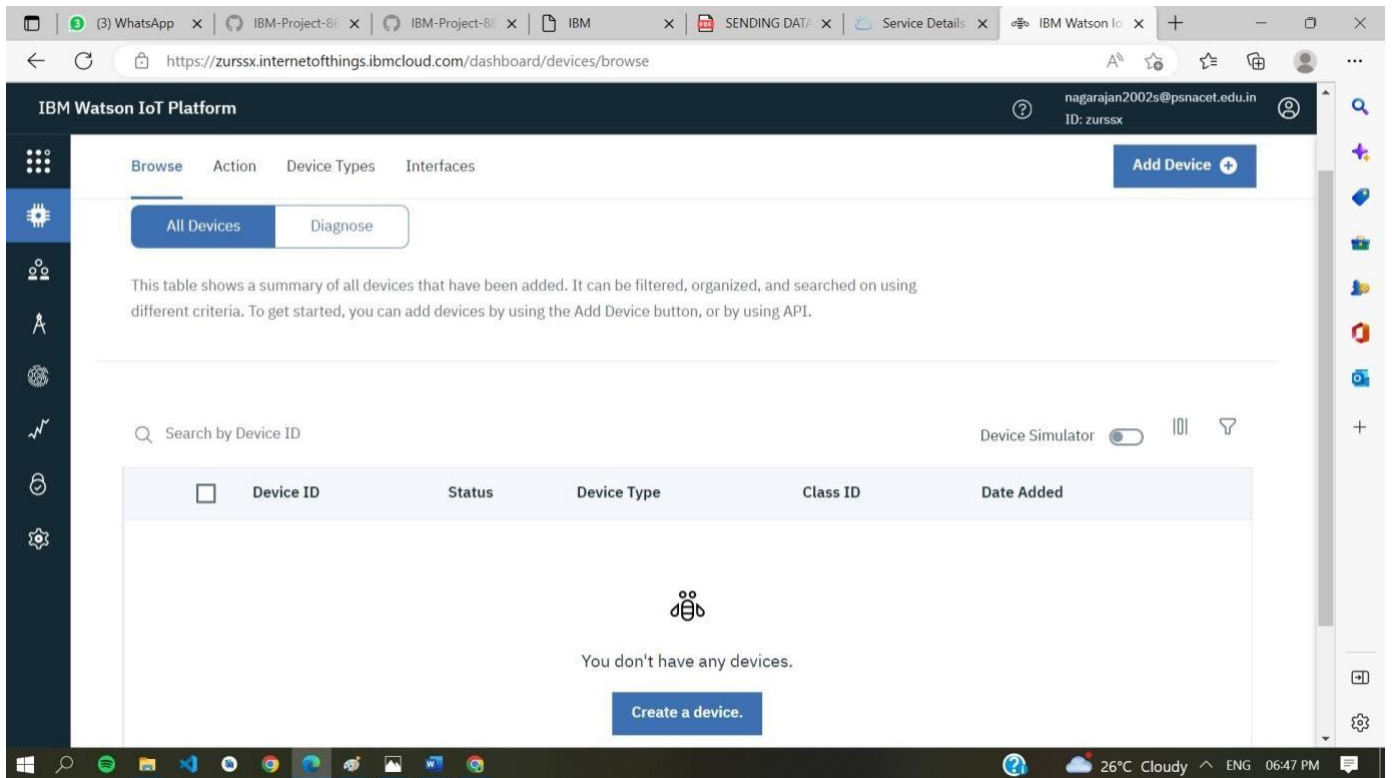


click on Launch

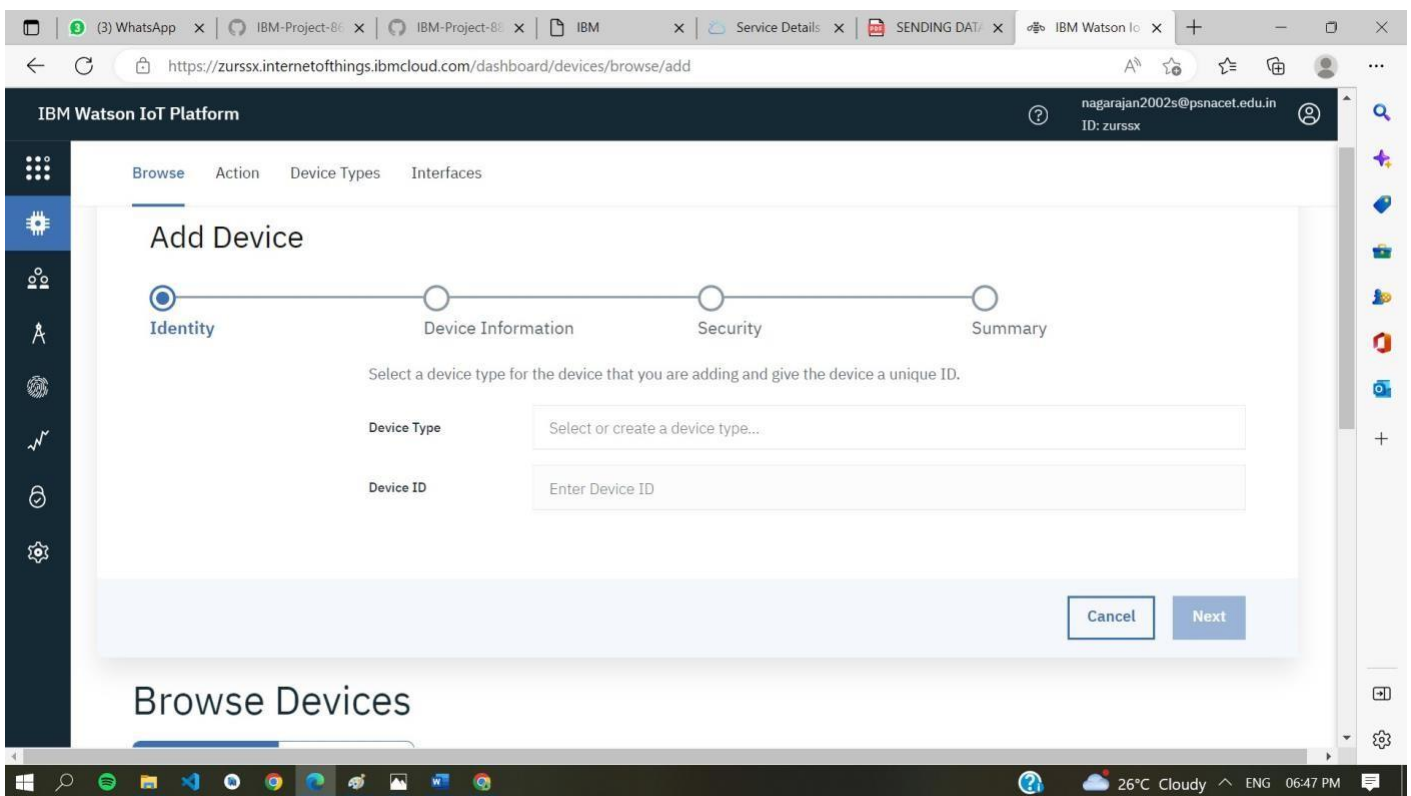


Dashboard of IBM Watson IoT platform,

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

https://zurssx.internetofthings.ibmcloud.com/dashboard/devices/types/add

nagarajan2002s@psnacet.edu.in  
ID: zurssx

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  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

https://zurssx.internetofthings.ibmcloud.com/dashboard/devices/types/add

nagarajan2002s@psnacet.edu.in  
ID: zurssx

Browse Action Device Types Interfaces

## Add type

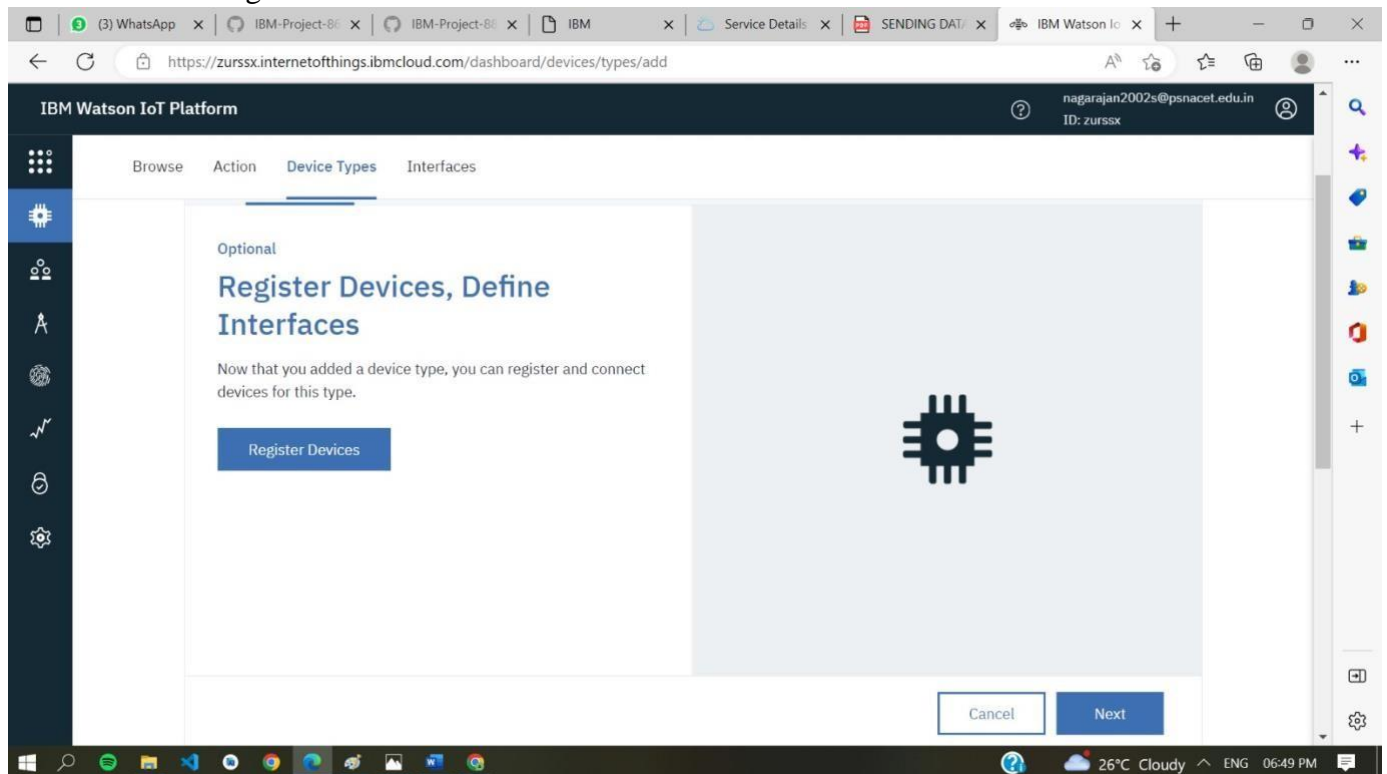
Identity Device Information

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

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

The screenshot shows the 'Add Device' page in the IBM Watson IoT Platform. The page has a dark blue header with the platform name and a user profile. A navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. The main content area is titled 'Add Device' and features a progress bar with four steps: 'Identity' (selected), 'Device Information', 'Security', and 'Summary'. Below the progress bar, a message states: 'Select a device type for the device that you are adding and give the device a unique ID.' There are two input fields: 'Device Type' with the value 'Nagarajan' and 'Device ID' with the value '12345'. At the bottom right, there are 'Cancel' and 'Next' buttons. The browser's address bar shows the URL: 'https://zurssx.internetofthings.ibmcloud.com/dashboard/devices/browse/add?type=Nagarajan'. The Windows taskbar at the bottom shows the time as 06:50 PM and the temperature as 26°C Cloudy.

Next

The screenshot shows the 'Add Device' page in the IBM Watson IoT Platform, now at the 'Device Information' step. The progress bar shows 'Identity' as a completed step and 'Device Information' as the current step. A message states: 'You can modify the default device information and enter more information about the device for identification purposes.' The form contains several input fields arranged in two columns. The left column includes 'Serial Number' (placeholder: 'Enter Serial Number'), 'Model' (placeholder: 'Enter Model'), 'Description' (placeholder: 'Enter Description'), and 'Hardware Version' (placeholder: 'Enter Hardware Version'). The right column includes 'Manufacturer' (placeholder: 'Enter Manufacturer'), 'Device Class' (placeholder: 'Enter Device Class'), 'Firmware Version' (placeholder: 'Enter Firmware Version'), and 'Descriptive Location' (placeholder: 'Enter Descriptive Location'). At the bottom left, there is a button labeled 'Add Metadata +'. The browser's address bar and the Windows taskbar are the same as in the previous screenshot.

○ Click on Next

Click on

The screenshot shows the 'Add Device' page in the IBM Watson IoT Platform, specifically the 'Security' tab. The page has a dark blue header with the IBM Watson IoT Platform logo and user information (nagarajan2002s@psnacet.edu.in, ID: zurssx). The left sidebar contains various icons for navigation. The main content area has a progress bar at the top with four steps: Identity, Device Information, Security (selected), and Summary. Below the progress bar, there are two options for selecting a device authentication token: 'Auto-generated authentication token (default)' and 'Self-provided authentication token'. The 'Auto-generated' option is selected. Below these options, there is a text input field labeled 'Authentication Token' with the placeholder text 'Enter an optional token'. Below the input field, there is a note: 'Make a note of the generated token. Lost authentication tokens cannot be recovered. Tokens are encrypted before being stored.' and a statement: 'Authentication token are encrypted before we store them.'

IBM Watson IoT Platform

Identity Device Information Security Summary

There are two options for selecting a device authentication token.

**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 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

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.

Finish

The screenshot shows the 'Add Device' page in the IBM Watson IoT Platform, specifically the 'Device Information' tab. The page has a dark blue header with the IBM Watson IoT Platform logo and user information (nagarajan2002s@psnacet.edu.in, ID: zurssx). The left sidebar contains various icons for navigation. The main content area has a progress bar at the top with four steps: Identity, Device Information (selected), Security, and Summary. Below the progress bar, there is a note: 'You can modify the default device information and enter more information about the device for identification purposes.' Below this note, there are several input fields for device information: 'Serial Number', 'Model', 'Description', 'Hardware Version', 'Manufacturer', 'Device Class', 'Firmware Version', and 'Descriptive Location'. Each input field has a placeholder text. At the bottom of the form, there is a button labeled 'Add Metadata +'. The bottom of the screen shows a Windows taskbar with various application icons and system information (26°C Cloudy, ENG, 06:50 PM).

IBM Watson IoT Platform

Identity Device Information Security Summary

You can modify the default device information and enter more information about the device for identification purposes.

Serial Number

Model

Description

Hardware Version

Manufacturer

Device Class

Firmware Version

Descriptive Location

Add Metadata +

○ Click on

○ Device is created

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
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/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 ... 115696 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 29117    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
Collecting iso8601>=0.1.10 (from ibmiotf)
  Downloading iso8601-0.1.12-py2.py3-none-any.whl (124kB)
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 1.0MB/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% |#####| 81kB 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/f9/45/bbc33ad957e82f7b71ba80e316d65a83d9d735ad12e0c0418
Running setup.py bdist_wheel for dicttoxml ... done
Stored in directory: /home/pi/.cache/pip/wheels/45/62/59/96910b33ec6a7b2ae66a13765401b50def5468024078e12cce
Running setup.py bdist_wheel for paho-mqtt ... done
Stored in directory: /home/pi/.cache/pip/wheels/28/08/6d/acdc8f2890111b7de71deebef6642f83be0313dfff0493
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:

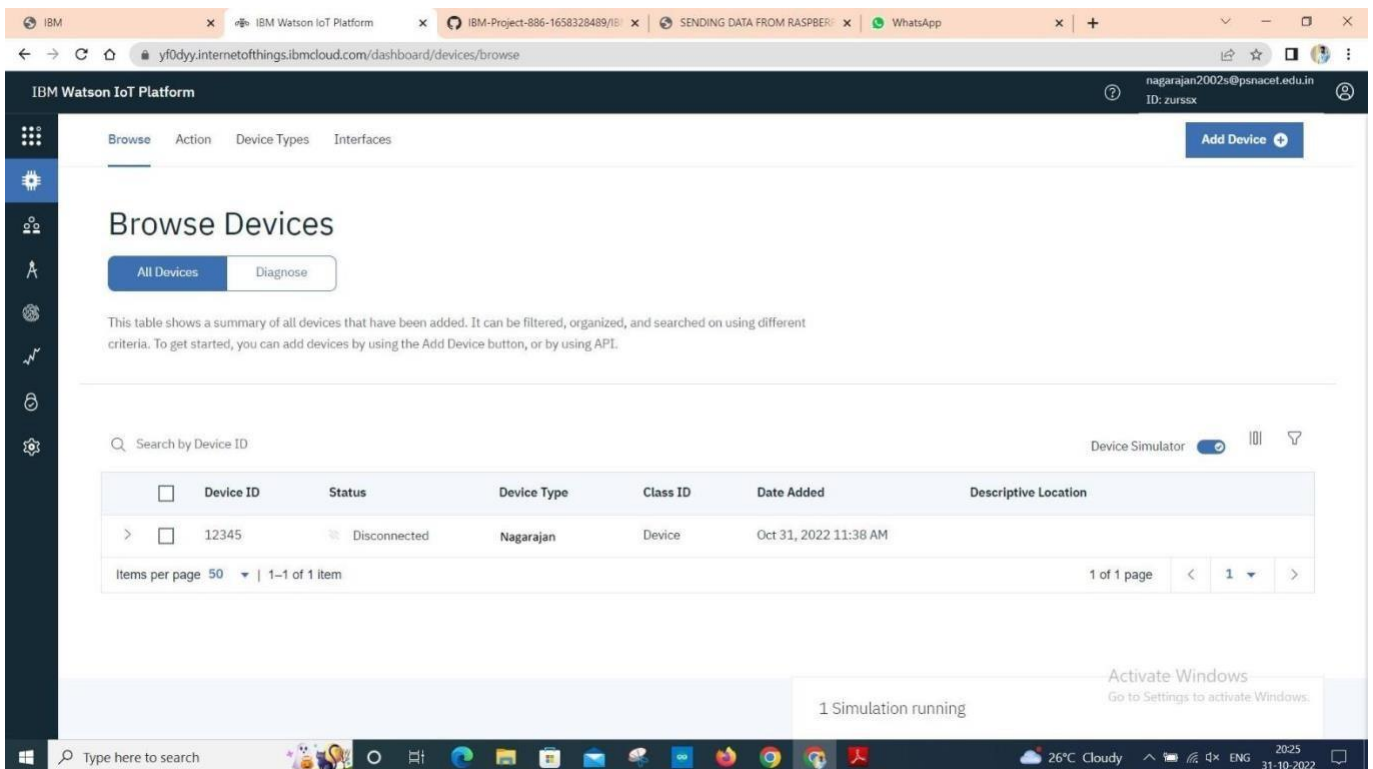
```

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:geg14: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 = 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'. A sidebar on the left contains various icons. The main content area shows a table of devices with columns: Device ID, Status, Device Type, Class ID, Date Added, and Descriptive Location. A device with ID 12345 is selected, showing its status as 'Disconnected' and type as 'Nagarajan'. Below this, a 'Recent Events' tab is active, displaying a table of events. The events table has columns: Event, Value, Format, and Last Received. The events are listed as 'event\_1' with values like '{\"Hazardous Gas\":61,\"Temperature\":88,\"Humidit...}' in 'json' format, received 'a few seconds ago'. A '1 Simulation running' notification is visible at the bottom right.

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:

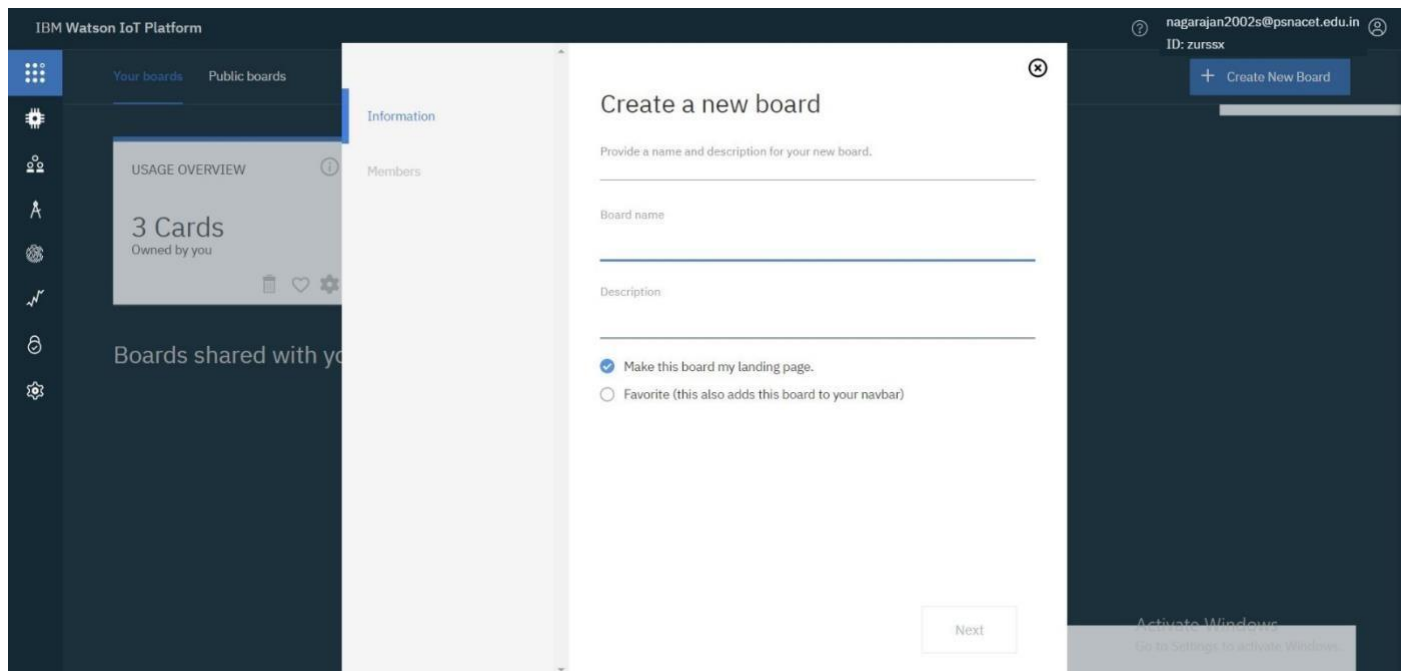
- In your Watson platform you have an option called board .Click on it and you get the following window on your screen

The screenshot shows the 'Your boards' section of the IBM Watson IoT Platform. It features two existing boards: 'USAGE OVERVIEW' with 3 cards and 'RISK AND SECURITY OVERVIEW' with 4 cards. A large dashed box with a plus sign indicates where to click to create a new board. A 'Create New Board' button is visible in the top right. Below the boards, there is a section for 'Boards shared with you'. A '1 Simulation running' notification is visible at the bottom right.

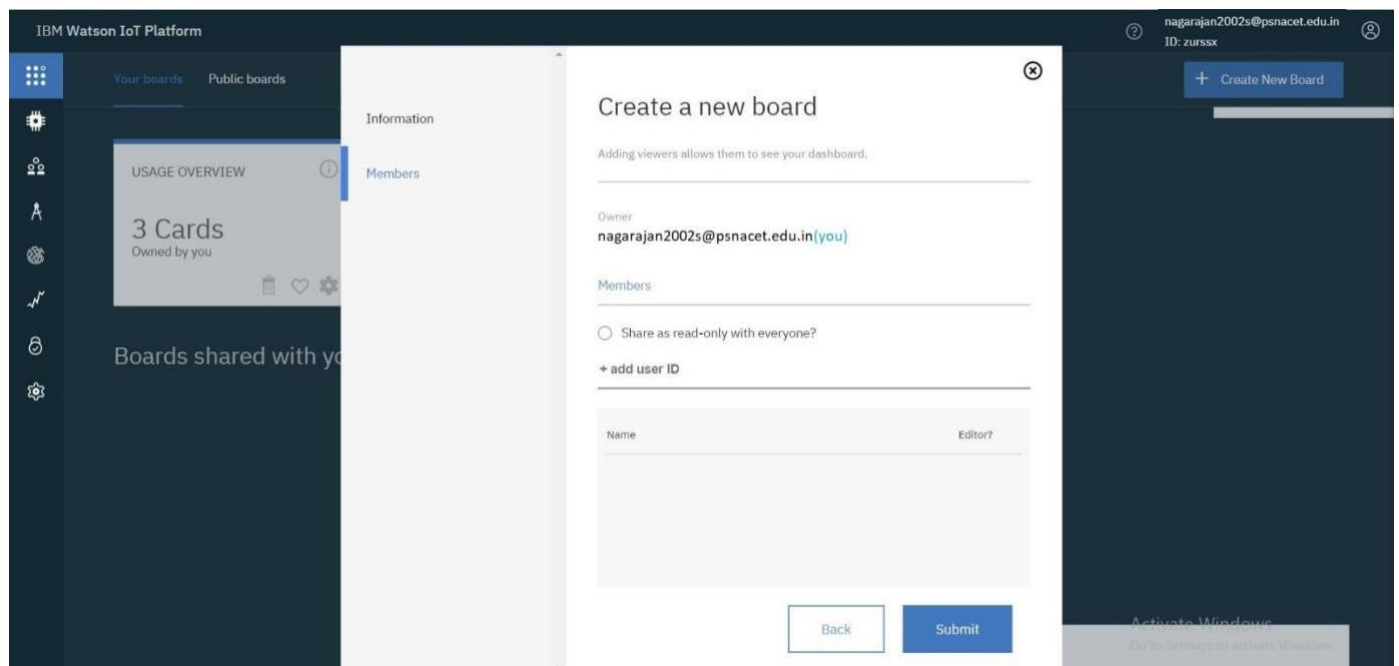
- Click on Create a new board to create a board .

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

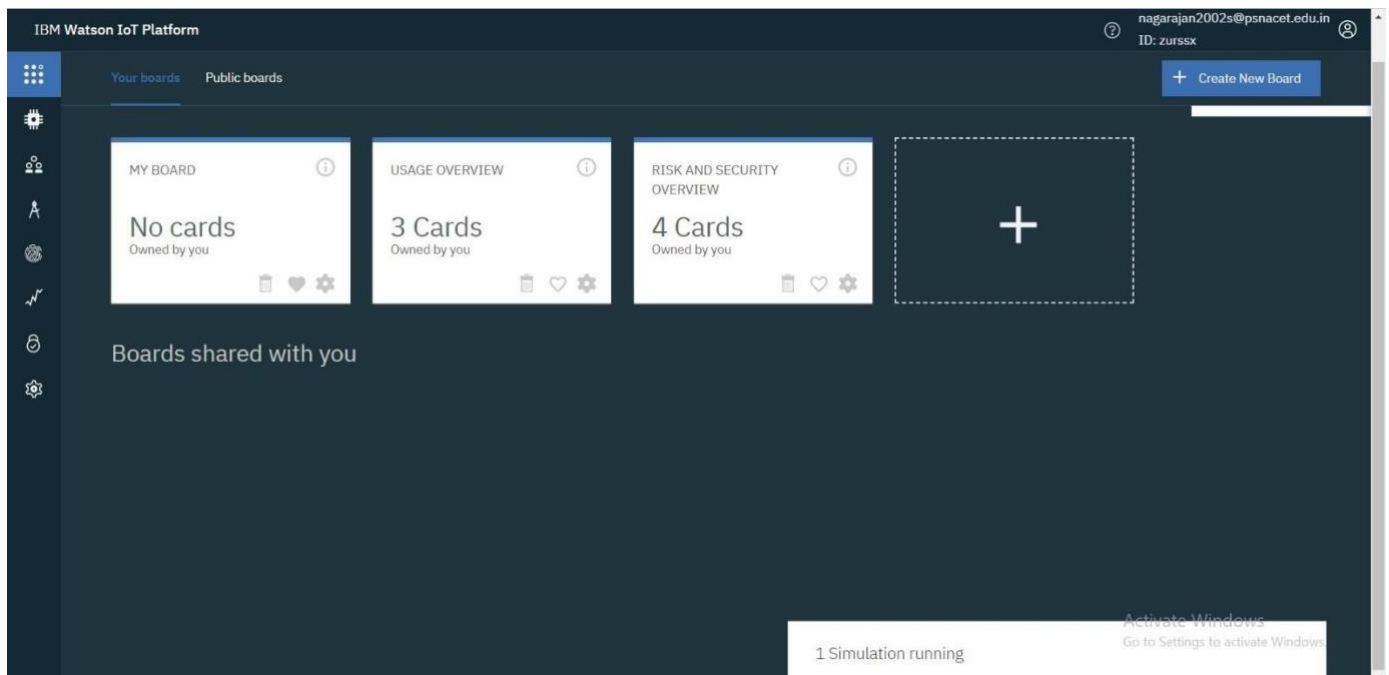




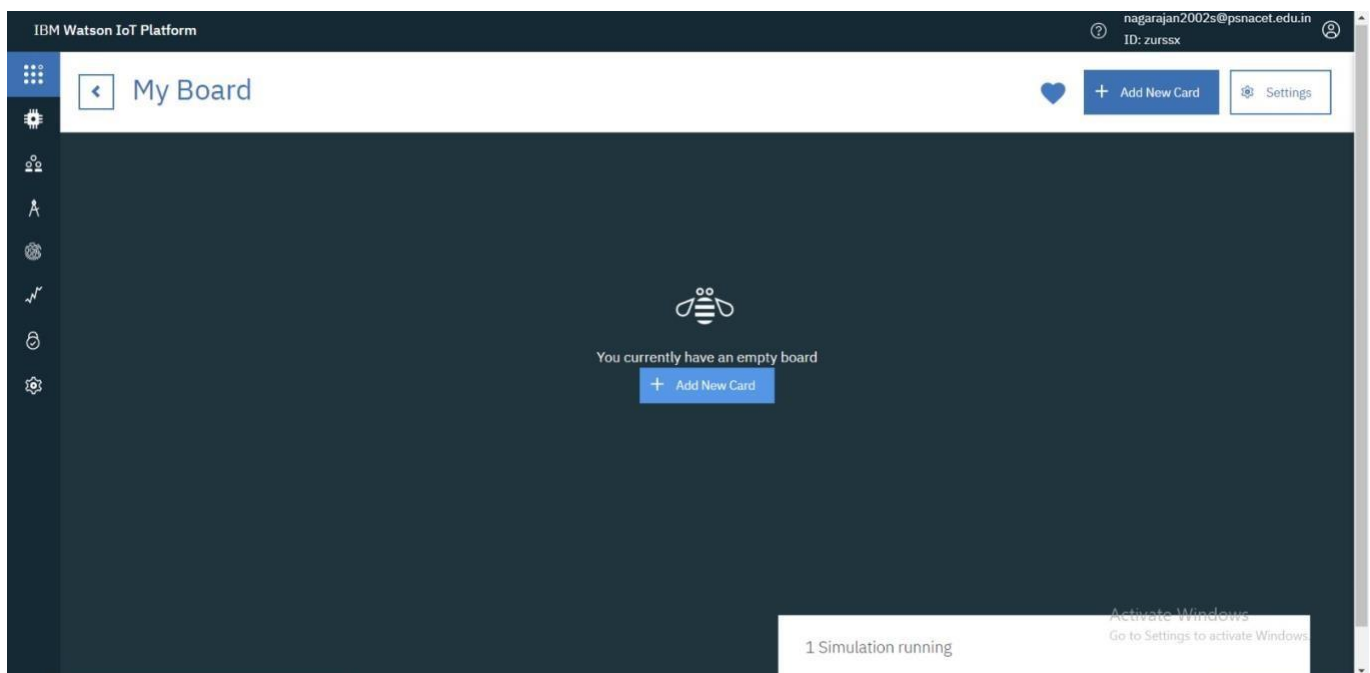
- Then click on Next you get the below window then again click on Submit



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

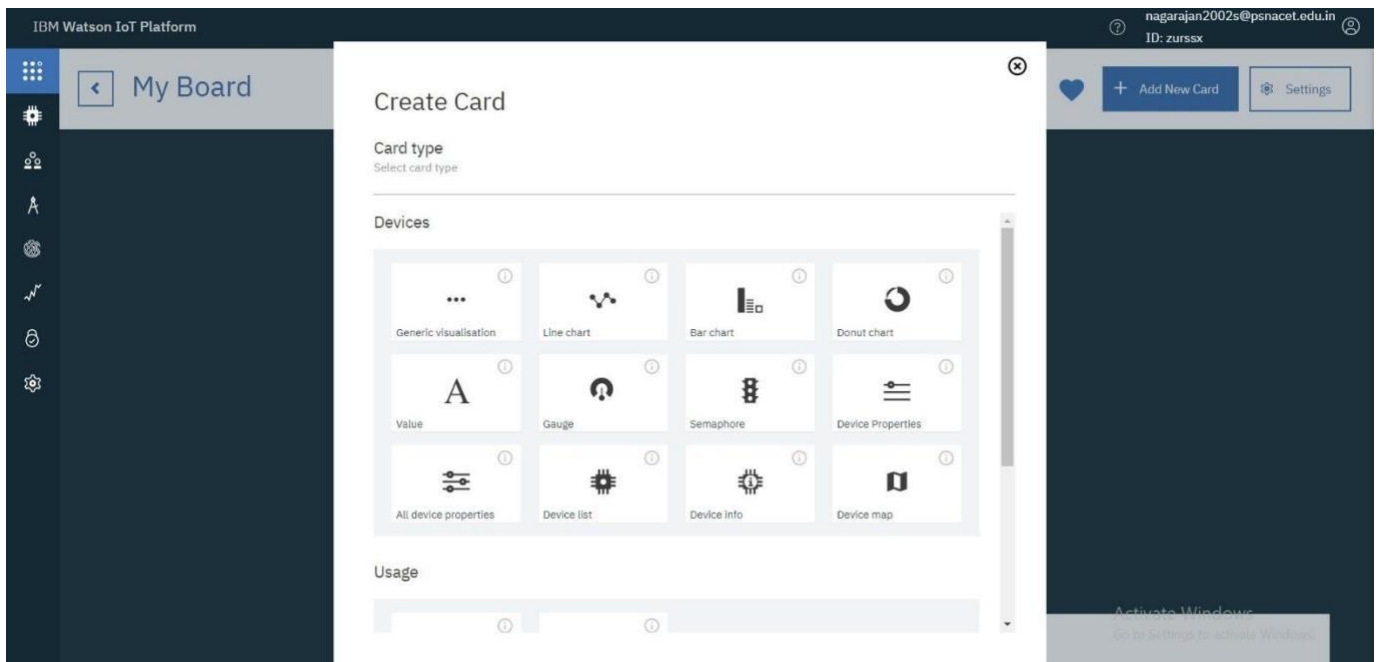


- Click on Add New Card

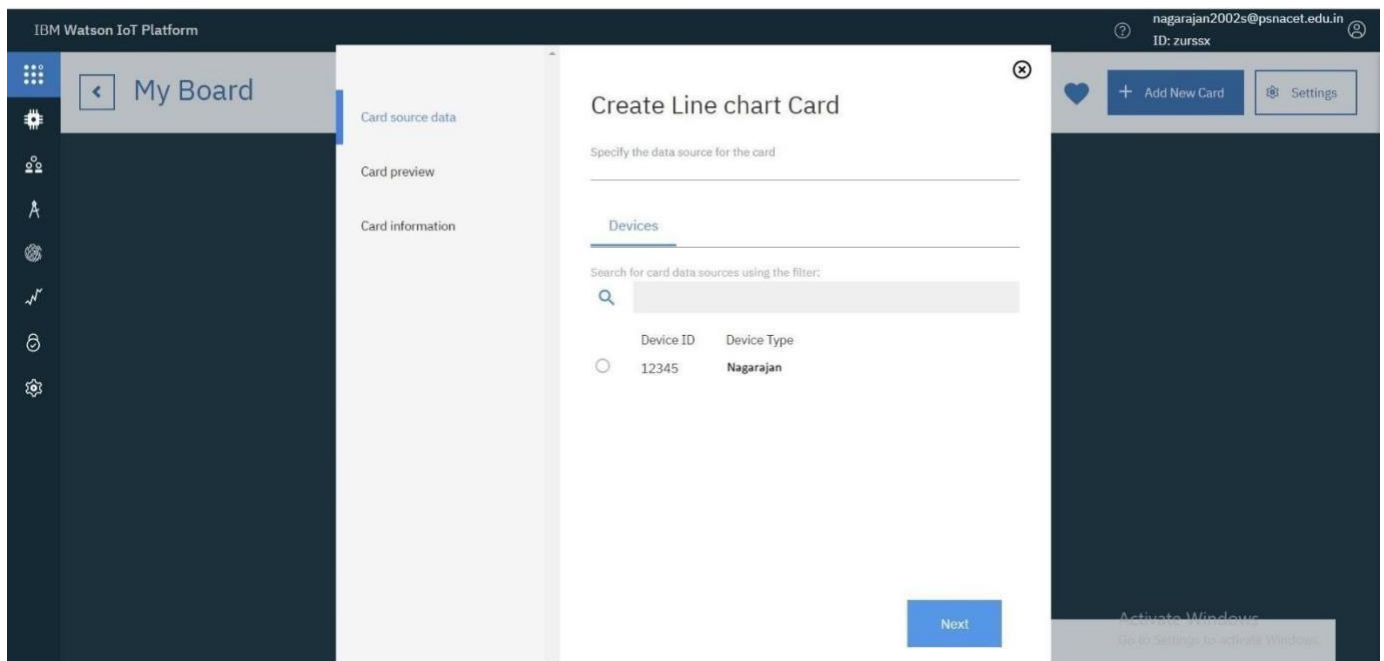


- Select the type of Graph u want accordingly and click next

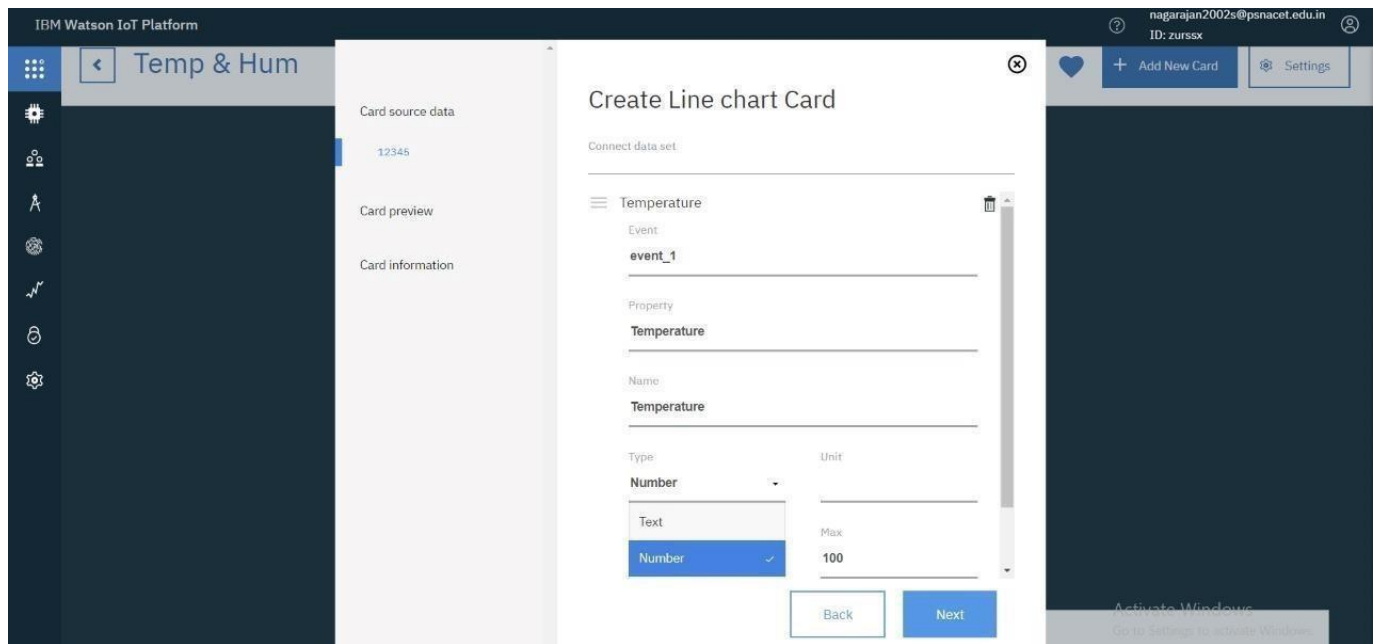




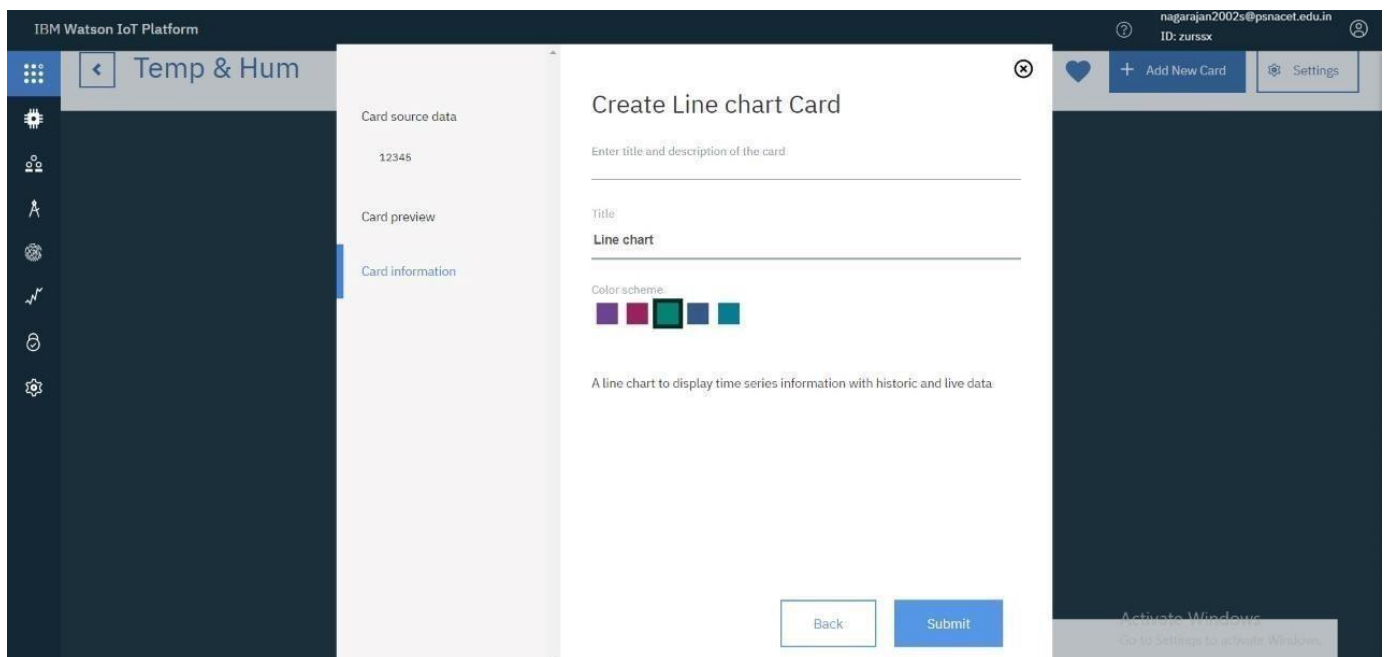
- You get the below window, choose the Device and click on Next.



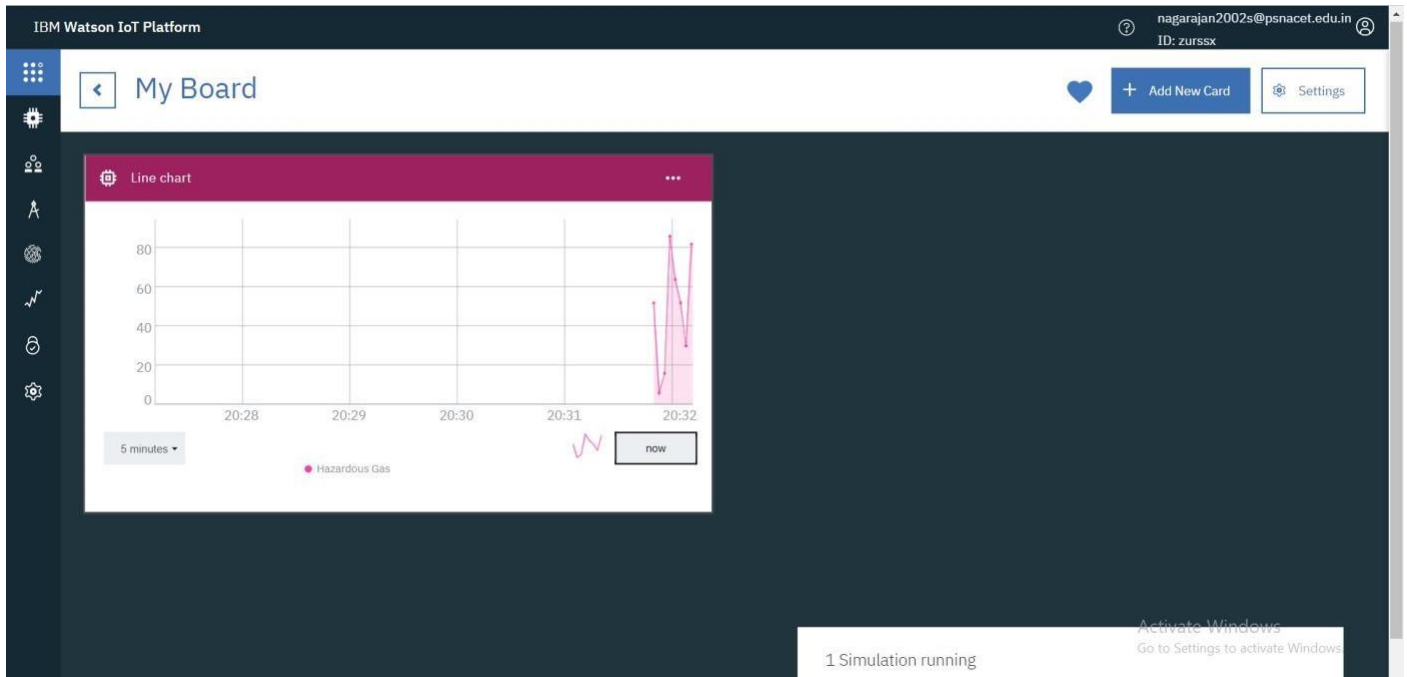
- Select the event, properly to be visualized on your graph and click next. In my case it is humidity



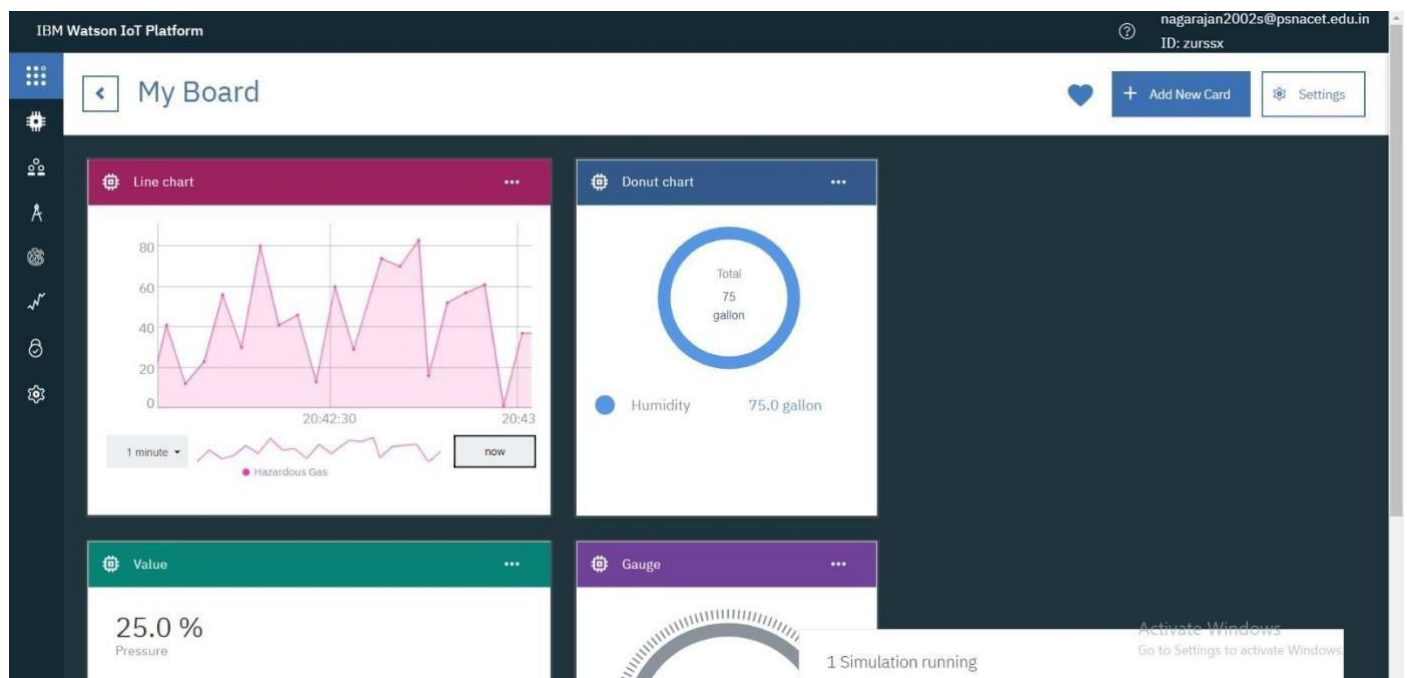
- Then select the size of the graph and color of the graph board you want and click next

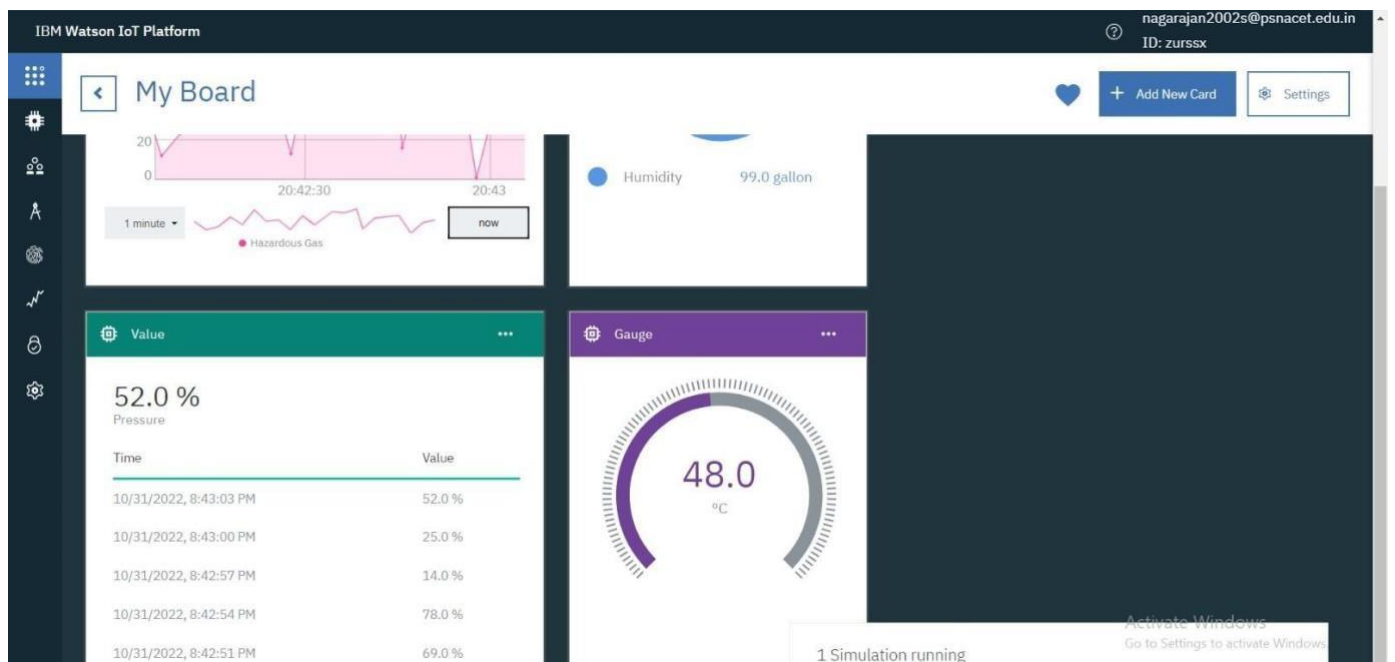


- Here is the graph



- 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.