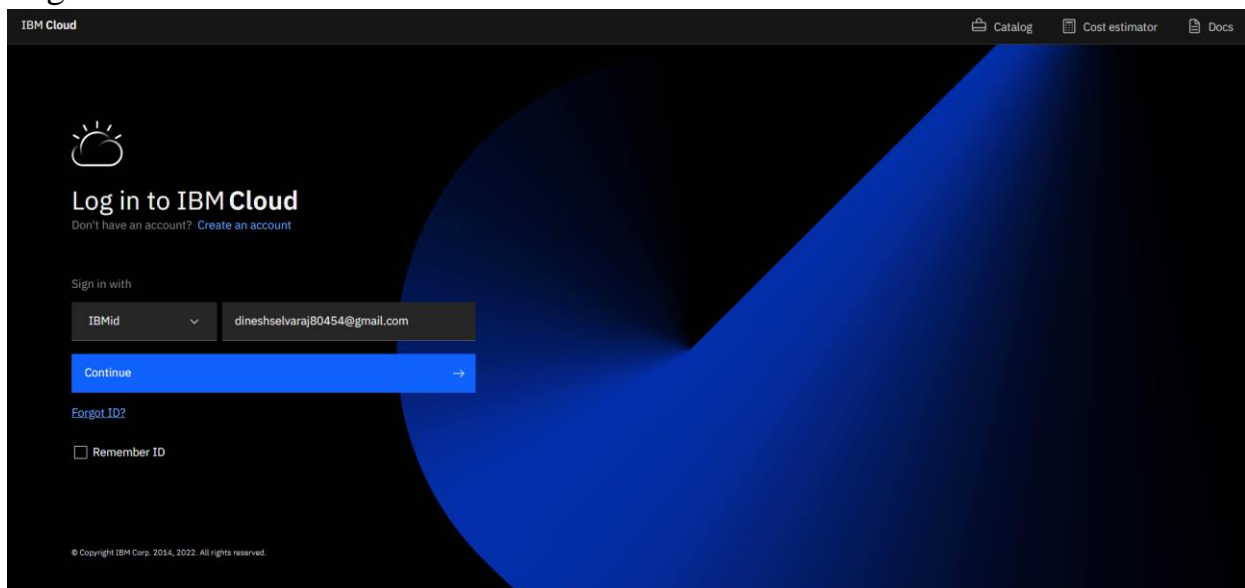


Project Development Phase Delivery of Sprint -2

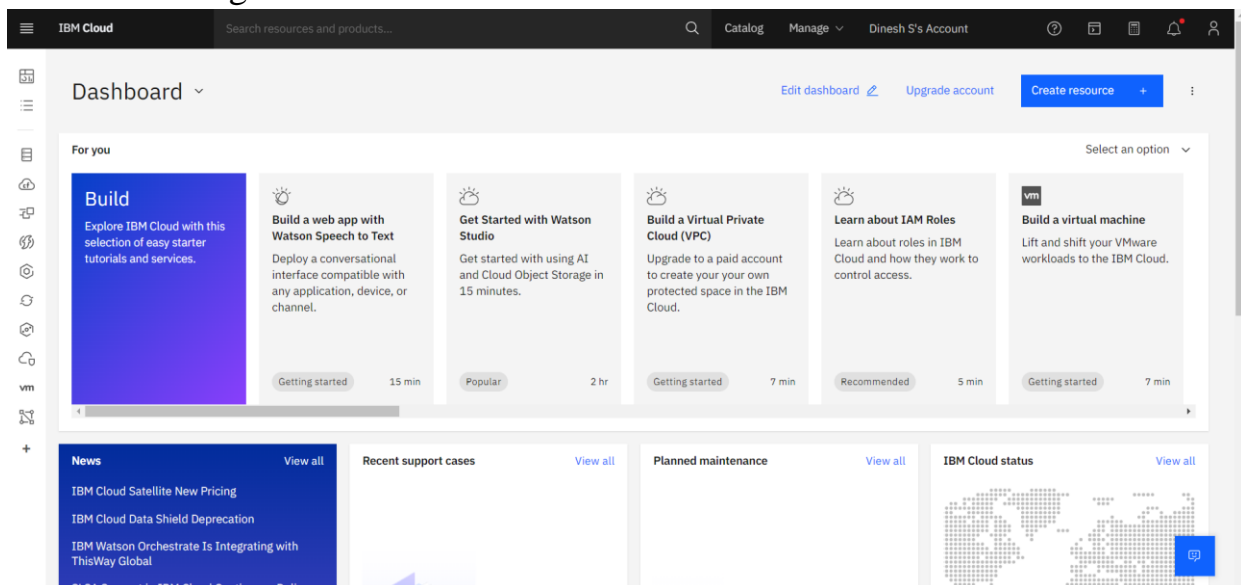
Team ID	PNT2022TMID31754
Project Name	Smart Farmer-IOT Enabled Smart FarmingApplication

In Sprint-2 we are going to develop the IBM Watson and making the connection to the node-red.

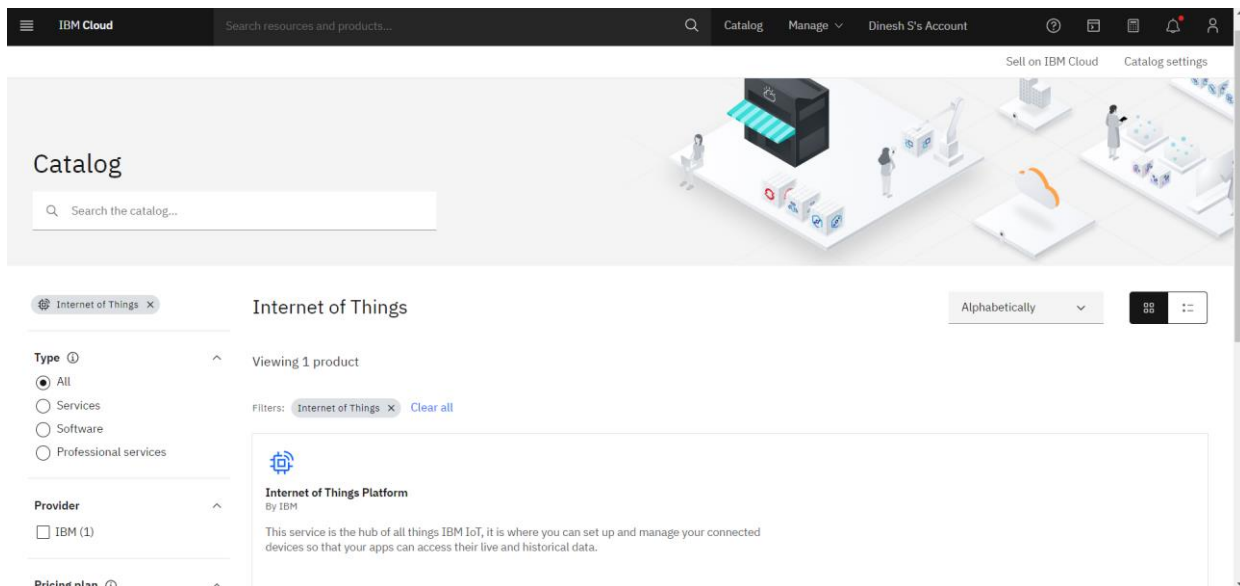
Login into IBM cloud:



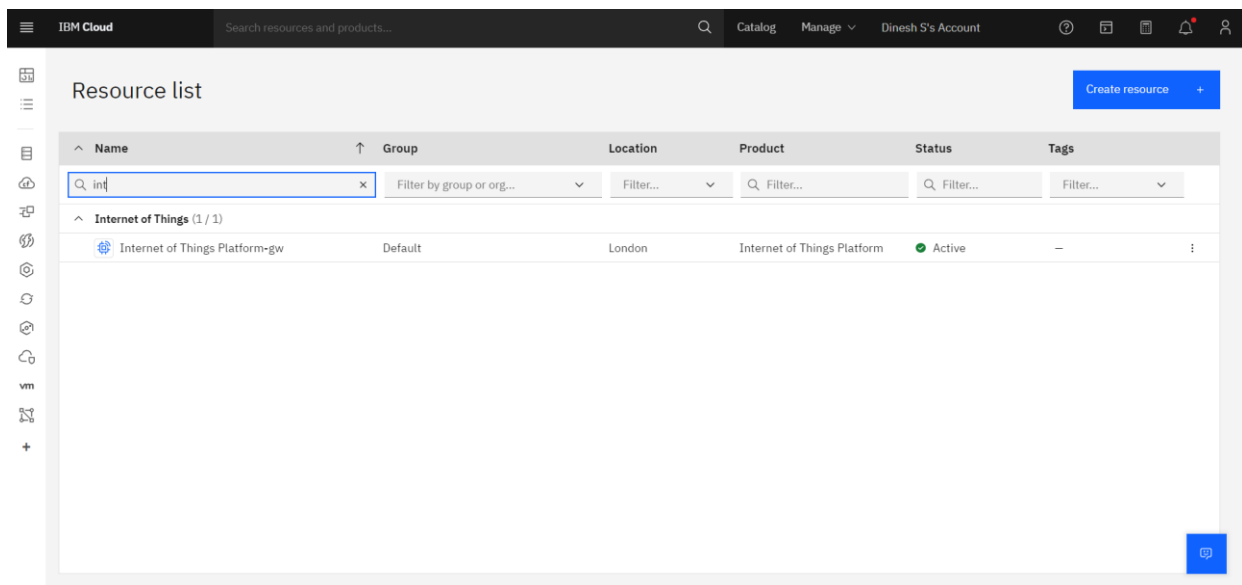
Click the Catlag button



After clicking the catalog .Select the Internet of Things and then click.



If you have already existing plan we can continue or we have to create new one.



Next window will be appear after clicking the exiting plan and click the launch button.

Now we have to Register Device in the Watson platform. click register device button.

IBM Cloud

Search resources and products...

Catalog

Manage

Dinesh S's Account

Resource list /

Internet of Things Platform-gw

Active

Add tags


Details

Actions...

Manage

Plan

Connections



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

Docs

Ready for the next level?

IBM Watson IoT Platform Journey

✓

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

IBM Watson IoT Platform

?

dineshseivraj00454@gmail.com

ID: 94g1g1

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.

Q Search by Device ID

Device Simulator

III

▼

<input type="checkbox"/>	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
>	<input type="checkbox"/> 1234567	Disconnected	Arduino	Device	16 Nov 2022 10:11	
>	<input type="checkbox"/> 637929	Disconnected	ESP32_Controller	Device	16 Nov 2022 12:00	

Items per page 50 | 1-2 of 2 items

1 of 1 page

<

1

>

Type the device type and device id and then click the next button.

IBM Watson IoT Platform

1234567 Disconnected Arduino Device

Identity Device Information Recent Events State Logs

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
event_1	{"Temperature":94,"pressure":62}	json	a few seconds ago
event_1	{"Temperature":80,"pressure":75}	json	a few seconds ago
event_1	{"Temperature":68,"pressure":19}	json	a few seconds ago
event_1	{"Temperature":74,"pressure":20}	json	a few seconds ago
event_1	{"Temperature":69,"pressure":13}	json	a few seconds ago

Simulations

1/50 Simulations Running

+ New Simulation

Device Type: Arduino

1 Event

1234567

1 x Create Simulated Device Use Registered Device

20 events sent 760 bytes sent

637929 Disconnected ESP32_Controller Device 16 Nov 2022 12:00

Now we have to give one authentication token that token is more than 8 characters and below 36 characters.

IBM Watson IoT Platform

1234567 Disconnected Arduino Device 16 Nov 2022 10:11

Identity Device Information Recent Events State Logs

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
event_1	{"Temperature":35,"pressure":17}	json	a few seconds ago
event_1	{"Temperature":21,"pressure":86}	json	a few seconds ago
event_1	{"Temperature":45,"pressure":4}	json	a few seconds ago
event_1	{"Temperature":26,"pressure":49}	json	a few seconds ago
event_1	{"Temperature":41,"pressure":41}	json	a few seconds ago

Simulations

1 Simulation running

637929 Disconnected ESP32_Controller Device

Then click the finish button.

My Device Credentials:

Organization ID : 9lg1g1

Device Type : Arduino

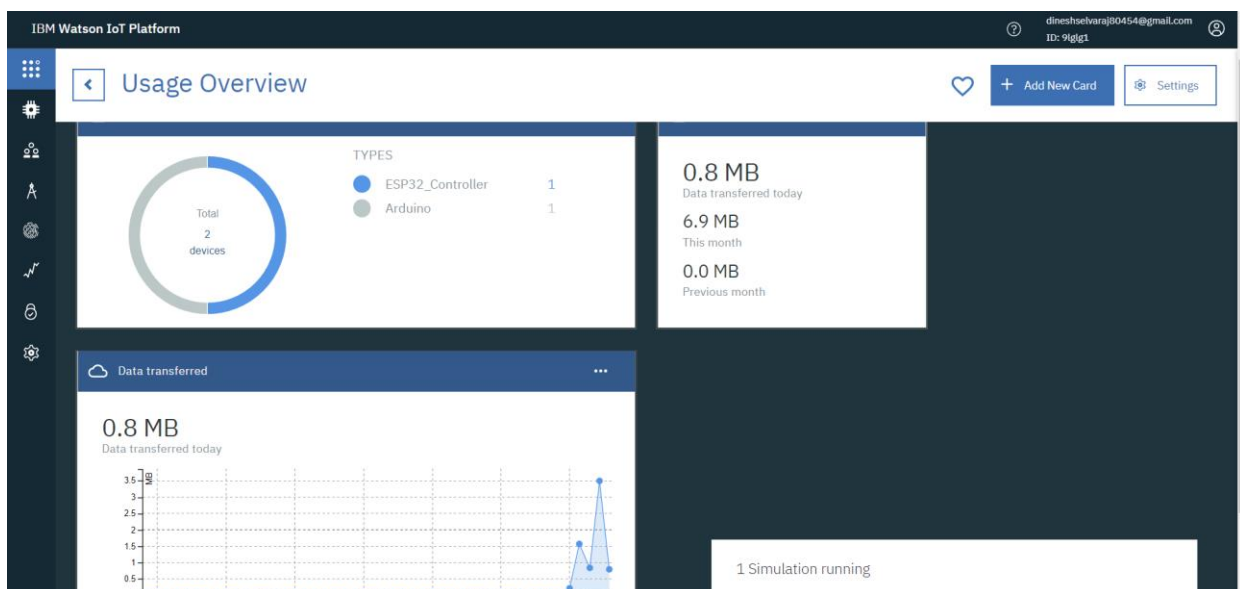
Device ID : 1234567

Authentication Method : use-token-auth

Authentication Token : 123456789

Now we have to add boards. Because we can data as graph model.

After adding boards we can run simulation and see the simulation as shown below. You will receive the simulator data in cloud .



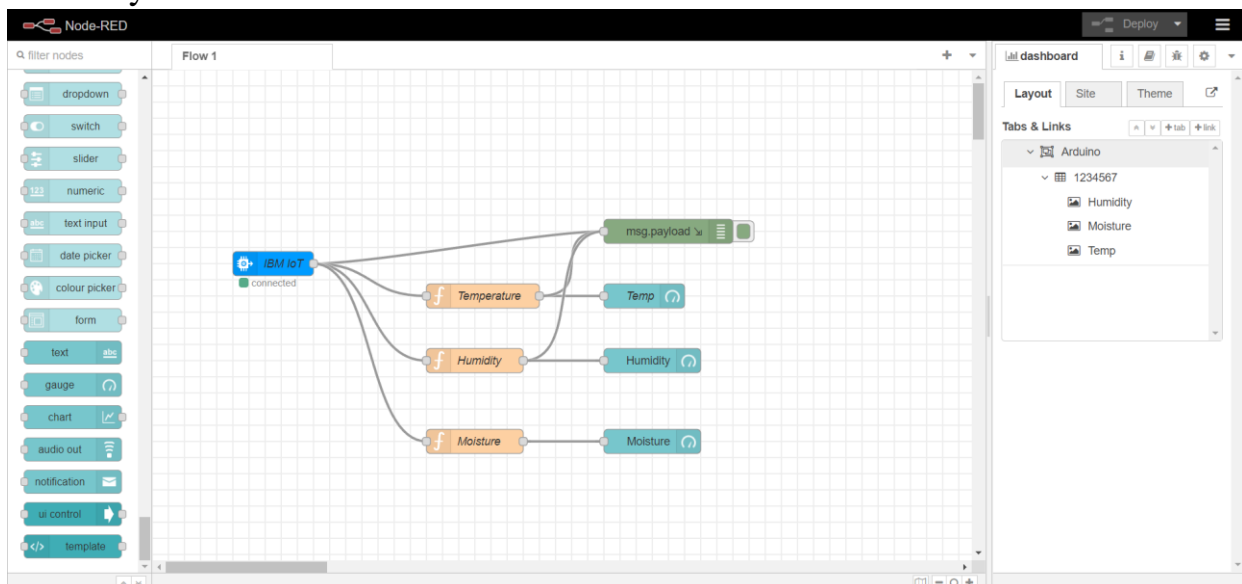
Now getting the random temperature and humidity values in the IBM Watson.

You can see the received data in Recent Events under your device. So finally we can generate temperature and humidity values as like real sensors.

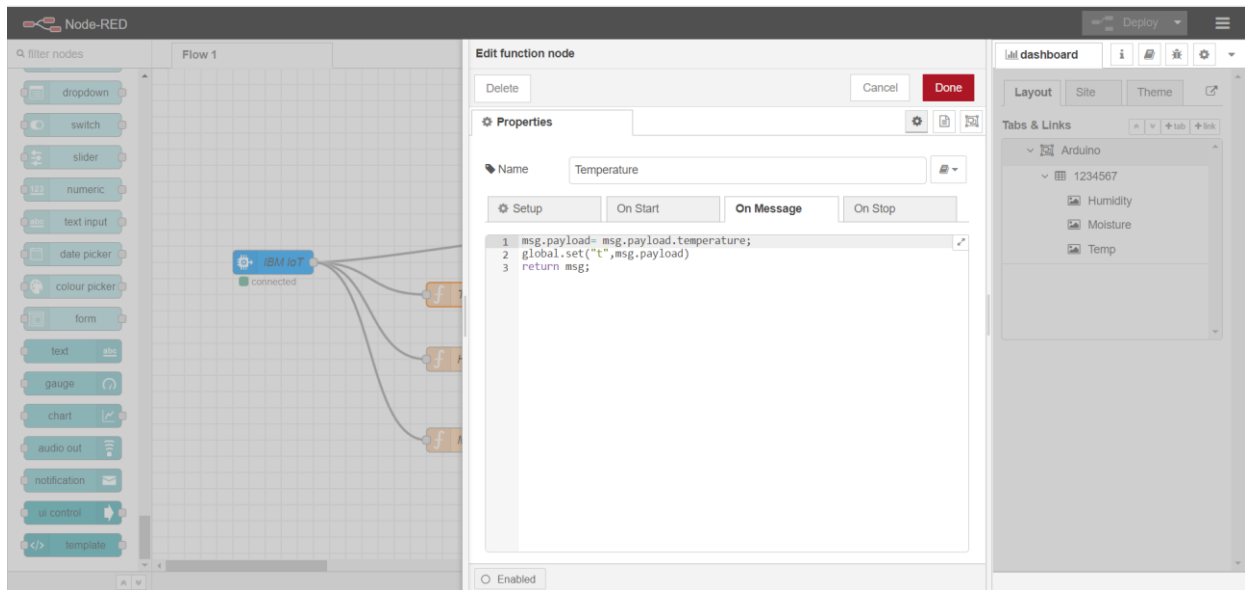


Now Configuration the Node-Red with IBM Watson Platform to collect the IBM cloud data:

The IBM Watson is added to Node-Red workflow. Then the appropriate device credentials obtained earlier are entered into the node to connect and fetch device telemetry to Node-Red.



Once it is connected to the Node-Red it receives the data from the Watson. Displaying the data using debug node in the left side of the workspace. And also see the results in the debug node



Connect function node and write the Java script code to get each reading separately.

Function node is rename as the temperature and humidity. And write json code on message.

Json code for Temperature:

```
msg.payload= msg.payload.temperature
global.set('t',msg.payload) return msg.
```

Json code for Humidity:

```
msg.payload= msg.payload.humidity
global.set('h',msg.payload) return msg.
```

Finally connect the Gauge nodes from node-red to see the data in the node-red dashboard UI:

Now we can see the output in the node-red dashboard.

Node-RED interface showing a flow named "Flow 1" and the configuration for the "ibmiot out" node.

Flow 1: The flow starts with an "IBM IoT" node (connected). It branches into three function nodes: "Temperature", "Humidity", and "Moisture". These three function nodes then connect to a "motor on" node, which in turn connects to a "motor off" node.

Edit ibmiot out node:

- Authentication: API Key
- API Key: 6ad679d0f449a3fc
- Output Type: Device Event
- Device Type: Arduino
- Device Id: 1234567
- Event Type: event_1
- Format: json
- Data: data
- QoS: 0
- Name: IBM IoT
- Service: registered

config: The configuration panel shows the node is enabled and lists the flow "Flow 1".

Home dashboard interface showing sensor data and motor control buttons.

ibm:

- humidity:** A gauge showing a value of 48 units (range 0 to 100).
- moisture:** A gauge showing a value of 48 units (range 4 to 100).
- temperature:** A gauge showing a value of 100 units (range 0 to 100).

motor:

- MOTOR OFF button
- MOTOR ON button