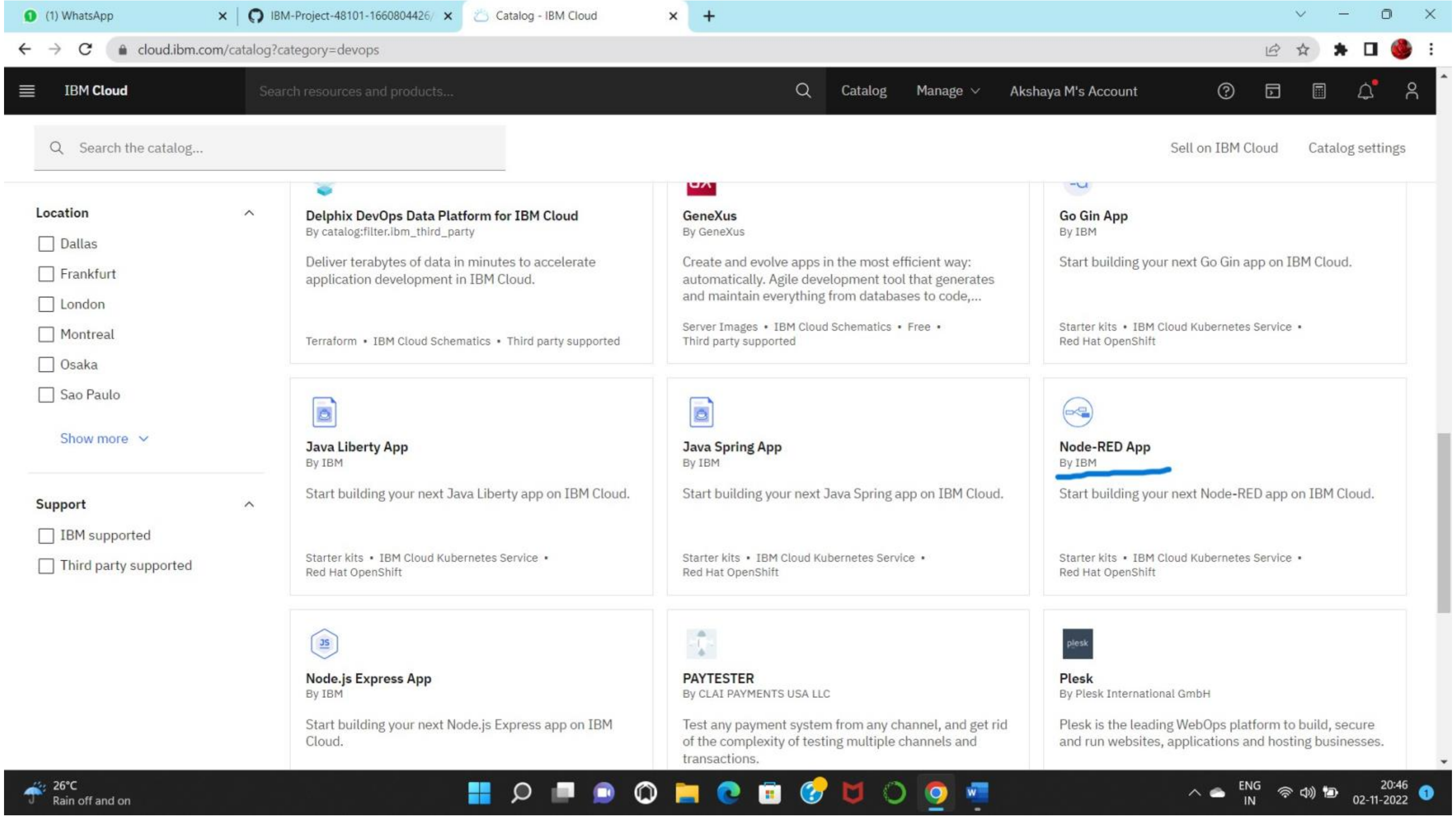
Create Node Red service

Team ID	PNT2022TMID49399
Project Name	Smart waste management system for metropolitan cities

Step 1: Login into IBM CLOUD account

Step2: In catalog, search for node red application



(1) WhatsApp | IBM-Project-48101-1660804426/ | Catalog - IBM Cloud

cloud.ibm.com/catalog?category=devops

IBM Cloud | Search resources and products... | Catalog | Manage | Akshaya M's Account

Search the catalog... | Sell on IBM Cloud | Catalog settings

Location

- □ Dallas
- □ Frankfurt
- □ London
- □ Montreal
- □ Osaka
- □ Sao Paulo

Show more

Support

- □ IBM supported
- □ Third party supported

Delphix DevOps Data Platform for IBM Cloud
By catalog:filter.ibm_third_party
Deliver terabytes of data in minutes to accelerate application development in IBM Cloud.
Terraform • IBM Cloud Schematics • Third party supported

GeneXus
By GeneXus
Create and evolve apps in the most efficient way: automatically. Agile development tool that generates and maintain everything from databases to code,...
Server Images • IBM Cloud Schematics • Free • Third party supported

Go Gin App
By IBM
Start building your next Go Gin app on IBM Cloud.
Starter kits • IBM Cloud Kubernetes Service • Red Hat OpenShift

Java Liberty App
By IBM
Start building your next Java Liberty app on IBM Cloud.
Starter kits • IBM Cloud Kubernetes Service • Red Hat OpenShift

Java Spring App
By IBM
Start building your next Java Spring app on IBM Cloud.
Starter kits • IBM Cloud Kubernetes Service • Red Hat OpenShift

Node-RED App
By IBM
Start building your next Node-RED app on IBM Cloud.
Starter kits • IBM Cloud Kubernetes Service • Red Hat OpenShift

Node.js Express App
By IBM
Start building your next Node.js Express app on IBM Cloud.

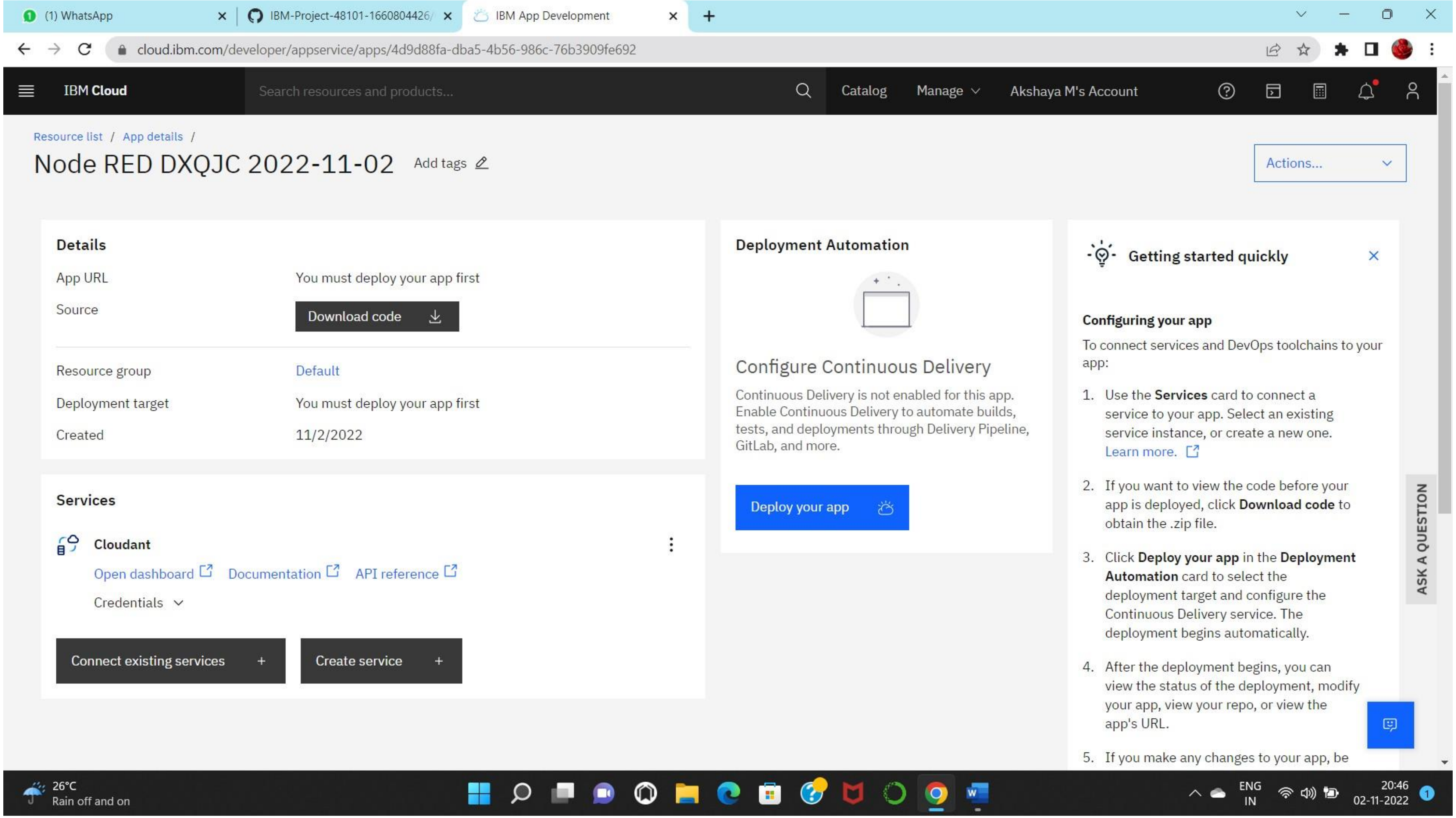
PAYTESTER
By CLAI PAYMENTS USA LLC
Test any payment system from any channel, and get rid of the complexity of testing multiple channels and transactions.

Plesk
By Plesk International GmbH
Plesk is the leading WebOps platform to build, secure and run websites, applications and hosting businesses.

26°C Rain off and on | ENG IN | 20:46 02-11-2022

Step 3: Enter the project details and click on create

Step 4: click on deploy option and deploy


(1) WhatsApp
IBM-Project-48101-1660804426/
IBM App Development
cloud.ibm.com/developer/appservice/apps/4d9d88fa-dba5-4b56-986c-76b3909fe692
IBM Cloud
Search resources and products...
Catalog
Manage
Akshaya M's Account
Actions...
ASK A QUESTION
26°C
Rain off and on
ENG
IN
20:46
02-11-2022

Resource list / App details /

Node RED DXQJC 2022-11-02 Add tags

Details

App URL	You must deploy your app first
Source	Download code
Resource group	Default
Deployment target	You must deploy your app first
Created	11/2/2022

Services

Cloudant

Open dashboard Documentation API reference

Credentials

Connect existing services + Create service +

Deployment Automation

Configure Continuous Delivery

Continuous Delivery is not enabled for this app. Enable Continuous Delivery to automate builds, tests, and deployments through Delivery Pipeline, GitLab, and more.

Deploy your app

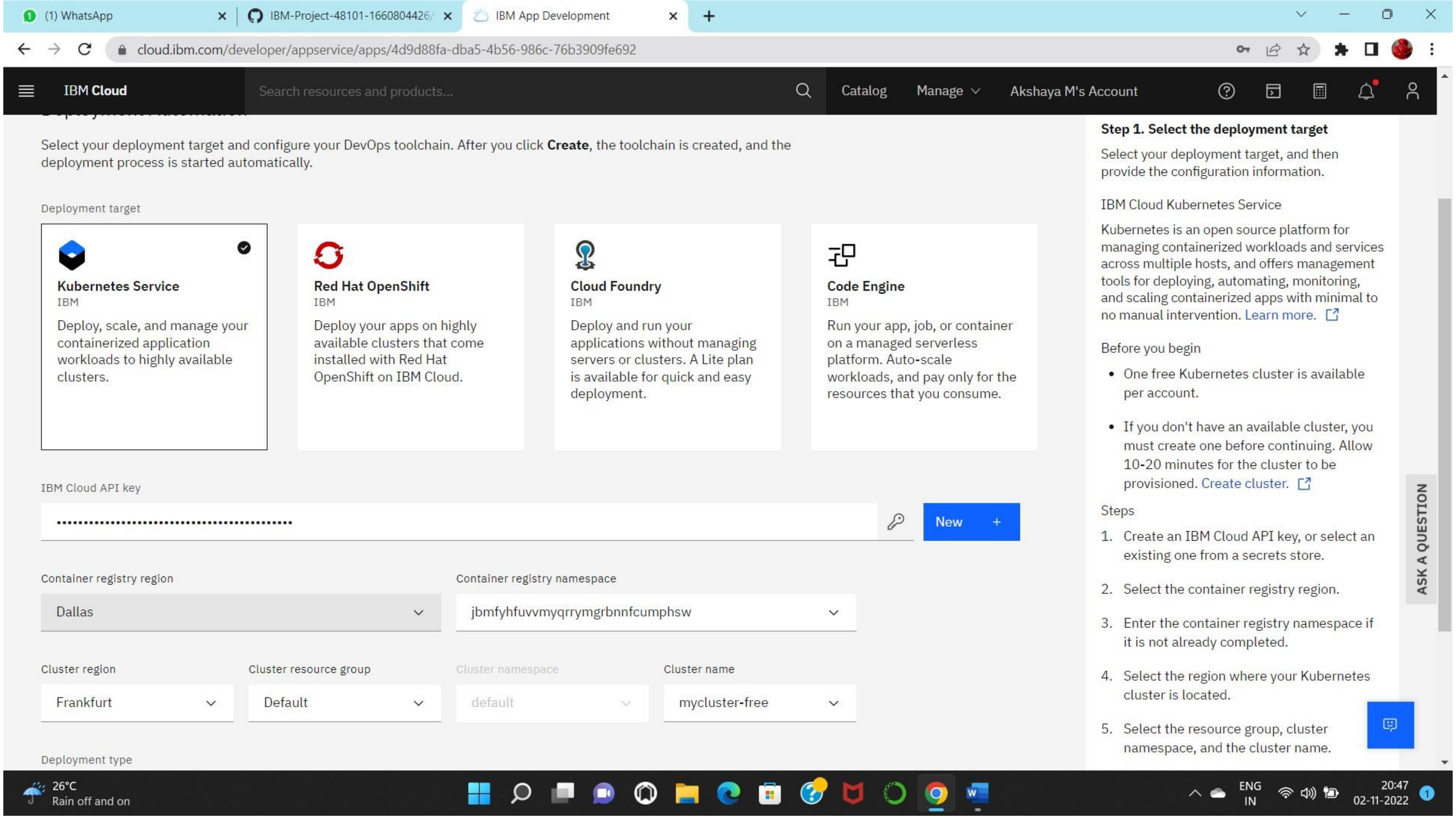
Getting started quickly

Configuring your app

To connect services and DevOps toolchains to your app:

1. Use the **Services** card to connect a service to your app. Select an existing service instance, or create a new one. Learn more.
2. If you want to view the code before your app is deployed, click **Download code** to obtain the .zip file.
3. Click **Deploy your app** in the **Deployment Automation** card to select the deployment target and configure the Continuous Delivery service. The deployment begins automatically.
4. After the deployment begins, you can view the status of the deployment, modify your app, view your repo, or view the app's URL.
5. If you make any changes to your app, be

Step 5: Set up the environment for deploying and click on create



Select your deployment target and configure your DevOps toolchain. After you click **Create**, the toolchain is created, and the deployment process is started automatically.

Deployment target

- **Kubernetes Service** IBM — Deploy, scale, and manage your containerized application workloads to highly available clusters.
- **Red Hat OpenShift** IBM — Deploy your apps on highly available clusters that come installed with Red Hat OpenShift on IBM Cloud.
- **Cloud Foundry** IBM — Deploy and run your applications without managing servers or clusters. A Lite plan is available for quick and easy deployment.
- **Code Engine** IBM — Run your app, job, or container on a managed serverless platform. Auto-scale workloads, and pay only for the resources that you consume.

IBM Cloud API key: ••

Container registry region: Dallas

Container registry namespace: jbmfyhfuvvmyqrrymgrbnnfcumphsw

Cluster region: Frankfurt

Cluster resource group: Default

Cluster namespace: default

Cluster name: mycluster-free

Deployment type

Step 1. Select the deployment target

Select your deployment target, and then provide the configuration information.

IBM Cloud Kubernetes Service

Kubernetes is an open source platform for managing containerized workloads and services across multiple hosts, and offers management tools for deploying, automating, monitoring, and scaling containerized apps with minimal to no manual intervention. Learn more.

Before you begin

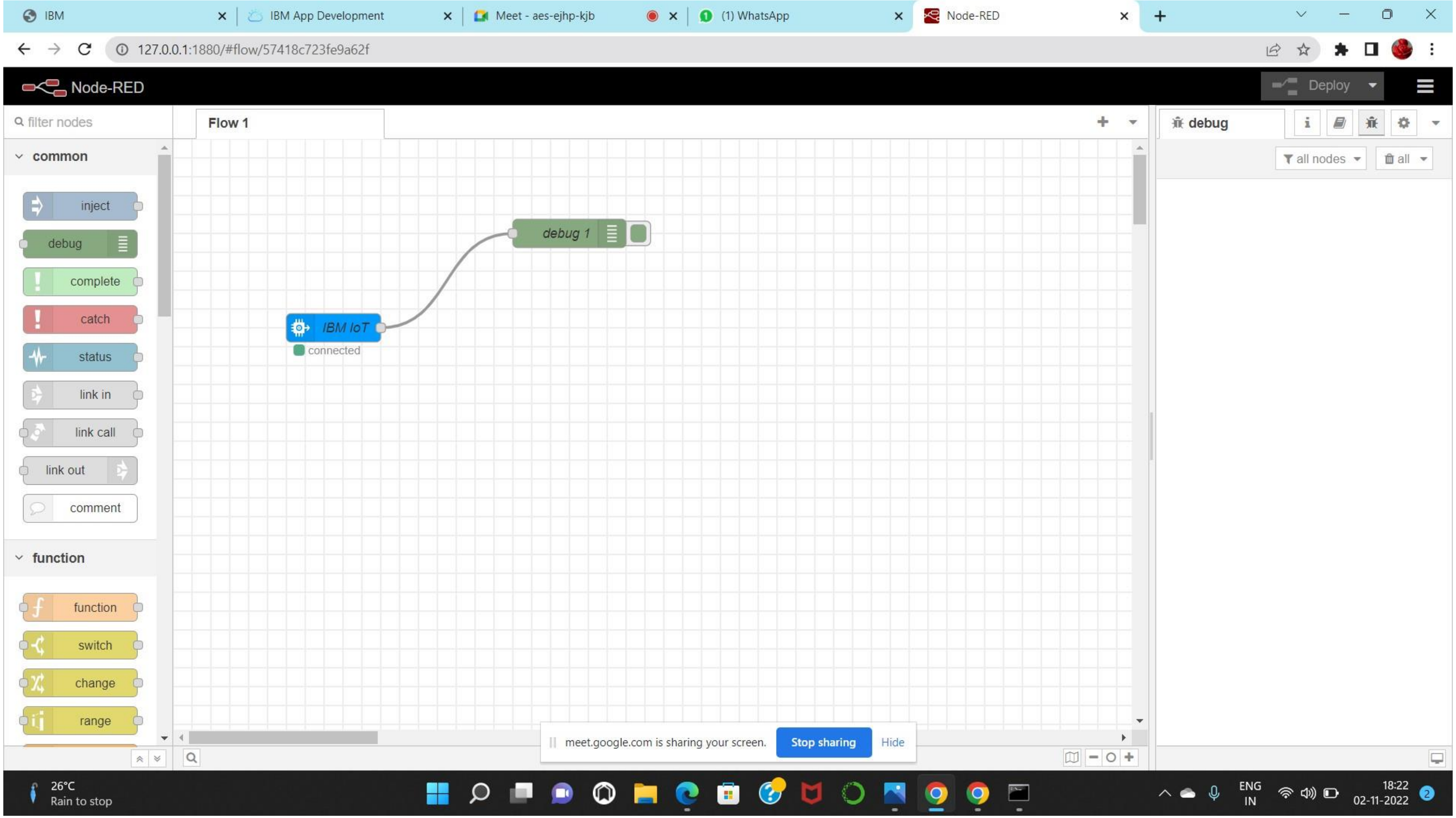
- One free Kubernetes cluster is available per account.
- If you don't have an available cluster, you must create one before continuing. Allow 10-20 minutes for the cluster to be provisioned. Create cluster.

Steps

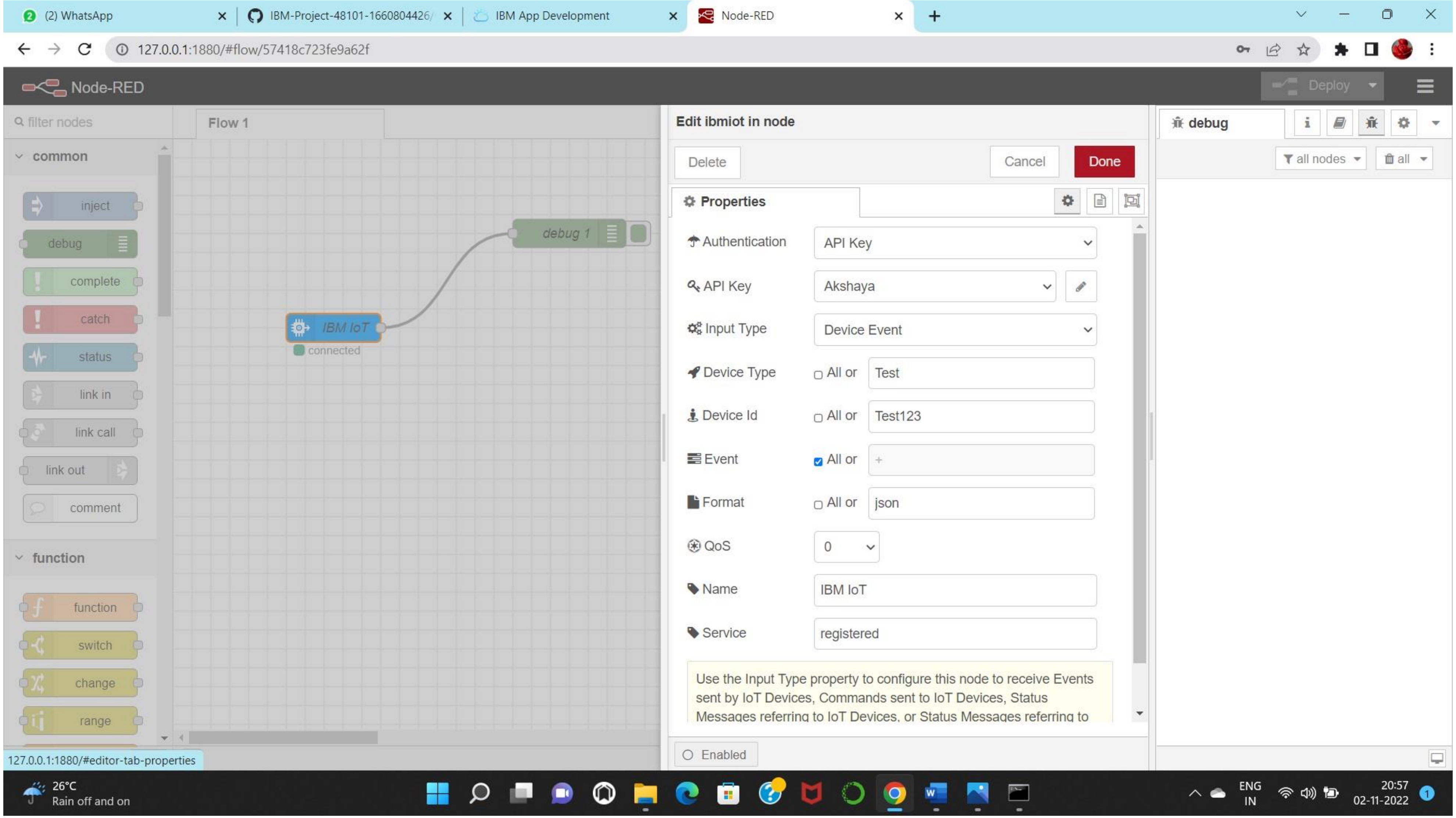
1. Create an IBM Cloud API key, or select an existing one from a secrets store.
2. Select the container registry region.
3. Enter the container registry namespace if it is not already completed.
4. Select the region where your Kubernetes cluster is located.
5. Select the resource group, cluster namespace, and the cluster name.

(1) WhatsApp
IBM-Project-48101-1660804426/
IBM App Development
cloud.ibm.com/developer/appservice/apps/4d9d88fa-dba5-4b56-986c-76b3909fe692
IBM Cloud
Search resources and products...
Catalog
Manage
Akshaya M's Account
New +
ASK A QUESTION
26°C
Rain off and on
ENG
IN
20:47
02-11-2022

Step 6: Now drag and drop the nodes and connect nodes with IOT Watson platform



Step 7: setup the settings that connects node red service with Watson IOT



Edit ibmiot in node

Authentication: API Key

API Key: Akshaya

Input Type: Device Event

Device Type: □ All or Test

Device Id: □ All or Test123

Event: ☑ All or +

Format: □ All or json

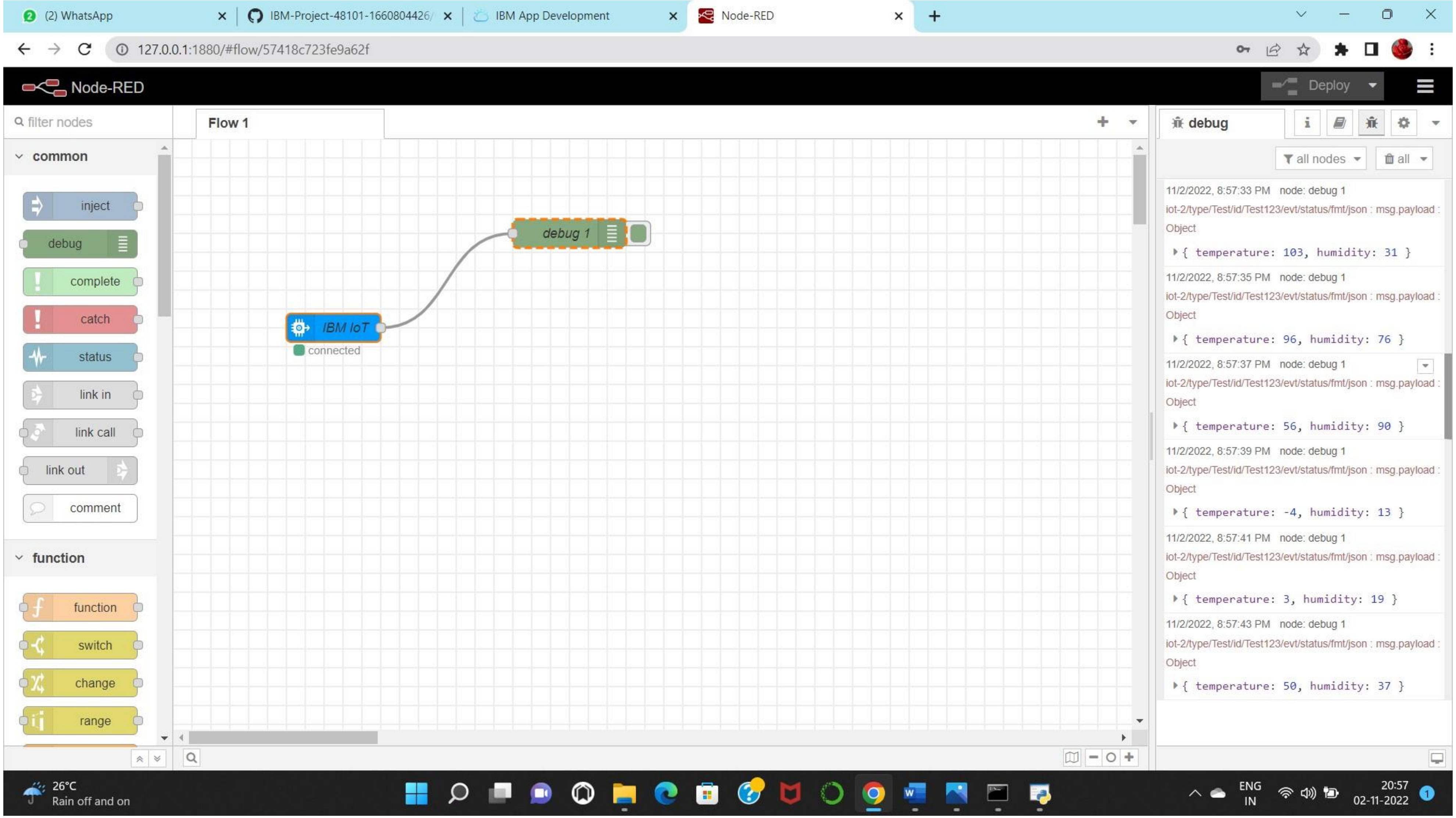
QoS: 0

Name: IBM IoT

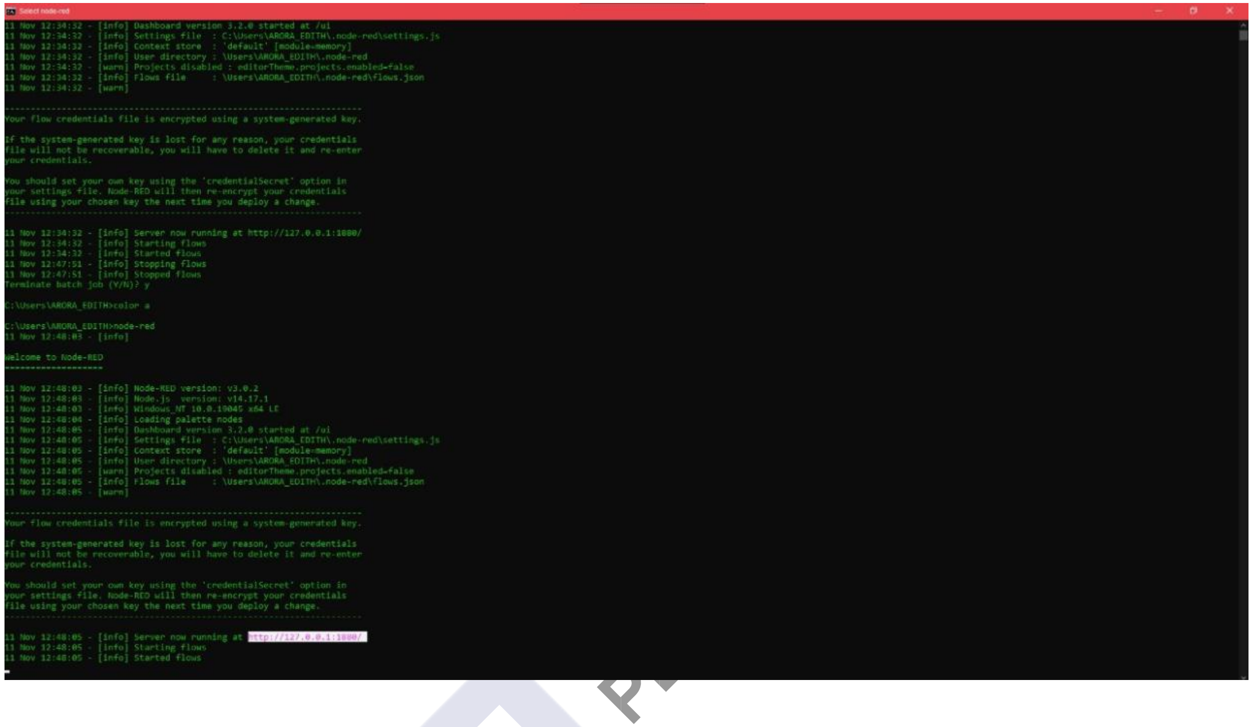
Service: registered

Use the Input Type property to configure this node to receive Events sent by IoT Devices, Commands sent to IoT Devices, Status Messages referring to IoT Devices, or Status Messages referring to

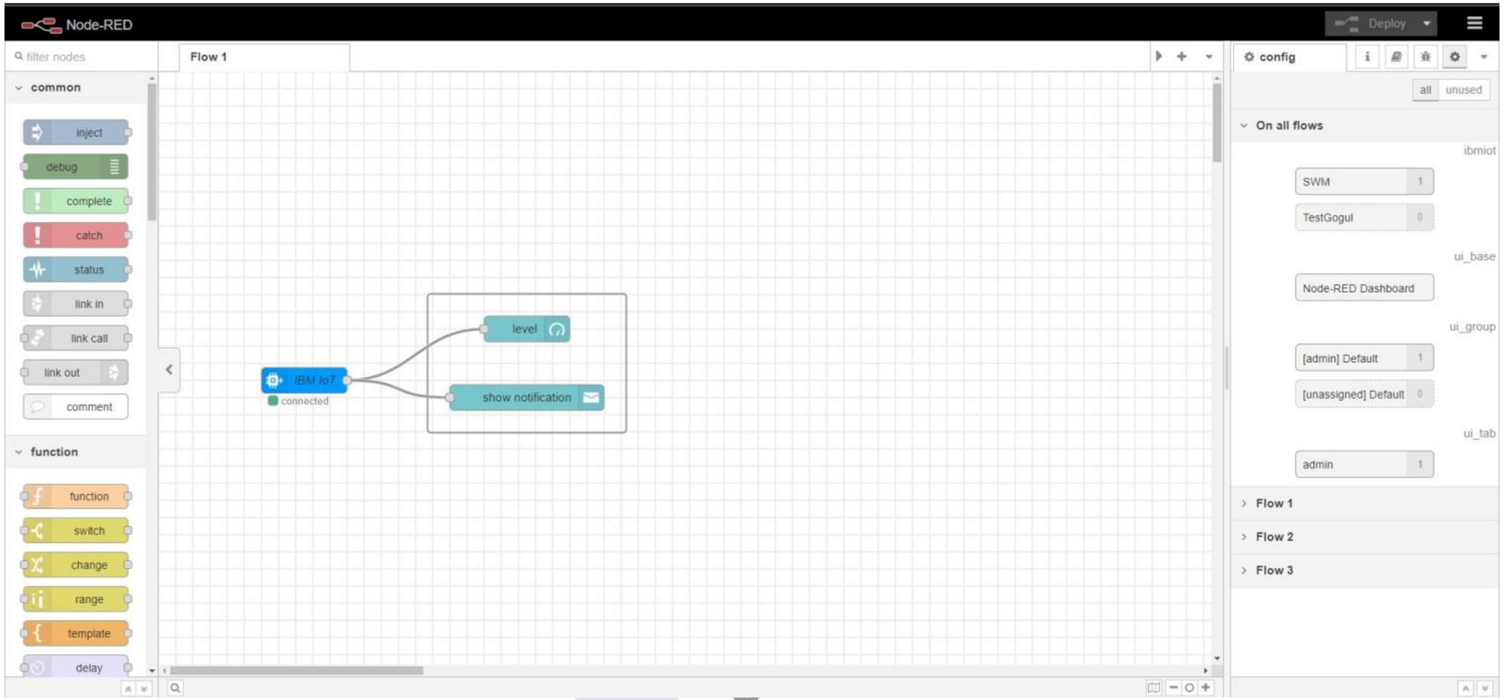
Step 8: Finally, output can be seen in node red service



```
11/2/2022, 8:57:33 PM   node: debug 1
iot-2/type/Test/id/Test123/evt/status/fmt/json : msg.payload :
Object
 ▸{ temperature: 103, humidity: 31 }
11/2/2022, 8:57:35 PM   node: debug 1
iot-2/type/Test/id/Test123/evt/status/fmt/json : msg.payload :
Object
 ▸{ temperature: 96, humidity: 76 }
11/2/2022, 8:57:37 PM   node: debug 1
iot-2/type/Test/id/Test123/evt/status/fmt/json : msg.payload :
Object
 ▸{ temperature: 56, humidity: 90 }
11/2/2022, 8:57:39 PM   node: debug 1
iot-2/type/Test/id/Test123/evt/status/fmt/json : msg.payload :
Object
 ▸{ temperature: -4, humidity: 13 }
11/2/2022, 8:57:41 PM   node: debug 1
iot-2/type/Test/id/Test123/evt/status/fmt/json : msg.payload :
Object
 ▸{ temperature: 3, humidity: 19 }
11/2/2022, 8:57:43 PM   node: debug 1
iot-2/type/Test/id/Test123/evt/status/fmt/json : msg.payload :
Object
 ▸{ temperature: 50, humidity: 37 }
```



```
11 Nov 12:34:32 - [info] Dashboard version 3.2.0 started at /ui
11 Nov 12:34:32 - [info] Settings file  : C:\Users\ARORA_EDITH\.node-red\settings.js
11 Nov 12:34:32 - [info] Context store  : 'default' [module=memory]
11 Nov 12:34:32 - [info] User directory : \Users\ARORA_EDITH\.node-red
11 Nov 12:34:32 - [warn] Projects disabled : editorTheme.projects.enabled=false
11 Nov 12:34:32 - [info] Flows file     : \Users\ARORA_EDITH\.node-red\flows.json
11 Nov 12:34:32 - [warn]

---------------------------------------------------------------------
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
---------------------------------------------------------------------

11 Nov 12:34:32 - [info] Server now running at http://127.0.0.1:1880/
11 Nov 12:34:32 - [info] Starting flows
11 Nov 12:34:32 - [info] Started flows
11 Nov 12:47:51 - [info] Stopping flows
11 Nov 12:47:51 - [info] Stopped flows
Terminate batch job (Y/N)? y

C:\Users\ARORA_EDITH>color a

C:\Users\ARORA_EDITH>node-red
11 Nov 12:48:03 - [info]

Welcome to Node-RED
===================

11 Nov 12:48:03 - [info] Node-RED version: v3.0.2
11 Nov 12:48:03 - [info] Node.js  version: v14.17.1
11 Nov 12:48:03 - [info] Windows_NT 10.0.19045 x64 LE
11 Nov 12:48:04 - [info] Loading palette nodes
11 Nov 12:48:05 - [info] Dashboard version 3.2.0 started at /ui
11 Nov 12:48:05 - [info] Settings file  : C:\Users\ARORA_EDITH\.node-red\settings.js
11 Nov 12:48:05 - [info] Context store  : 'default' [module=memory]
11 Nov 12:48:05 - [info] User directory : \Users\ARORA_EDITH\.node-red
11 Nov 12:48:05 - [warn] Projects disabled : editorTheme.projects.enabled=false
11 Nov 12:48:05 - [info] Flows file     : \Users\ARORA_EDITH\.node-red\flows.json
11 Nov 12:48:05 - [warn]

---------------------------------------------------------------------
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
---------------------------------------------------------------------

11 Nov 12:48:05 - [info] Server now running at http://127.0.0.1:1880/
11 Nov 12:48:05 - [info] Starting flows
11 Nov 12:48:05 - [info] Started flows
```


Node-RED
Deploy
filter nodes
common
inject
debug
complete
catch
status
link in
link call
link out
comment
function
function
switch
change
range
template
delay
Flow 1
IBM IoT
connected
level
show notification
config
all
unused
On all flows
ibmiot
SWM 1
TestGogul 0
ui_base
Node-RED Dashboard
ui_group
[admin] Default 1
[unassigned] Default 0
ui_tab
admin 1
Flow 1
Flow 2
Flow 3