MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, Accredited by NAAC, NBA & Affiliated to Anna University)

DESIGNING YOUR FUTURE
Estd. 2000

Department of Electronics And Communication Engineering

SMARTFARMER - IOT ENABLED SMART FARMING APPLICATION

TEAM ID: PNT2022TMID18947

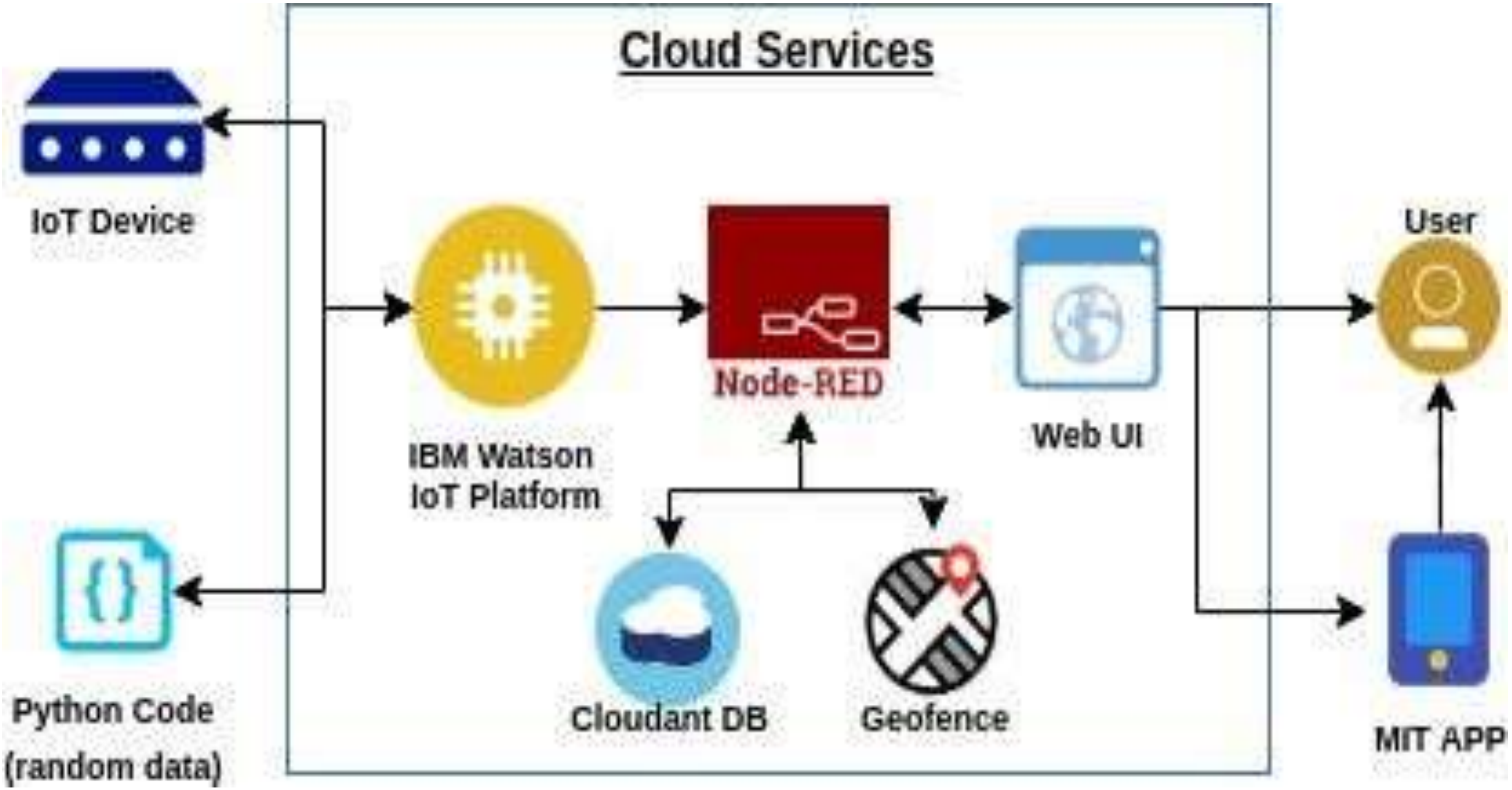
TEAM LEAD:

KOKILA P - MECR19EC044

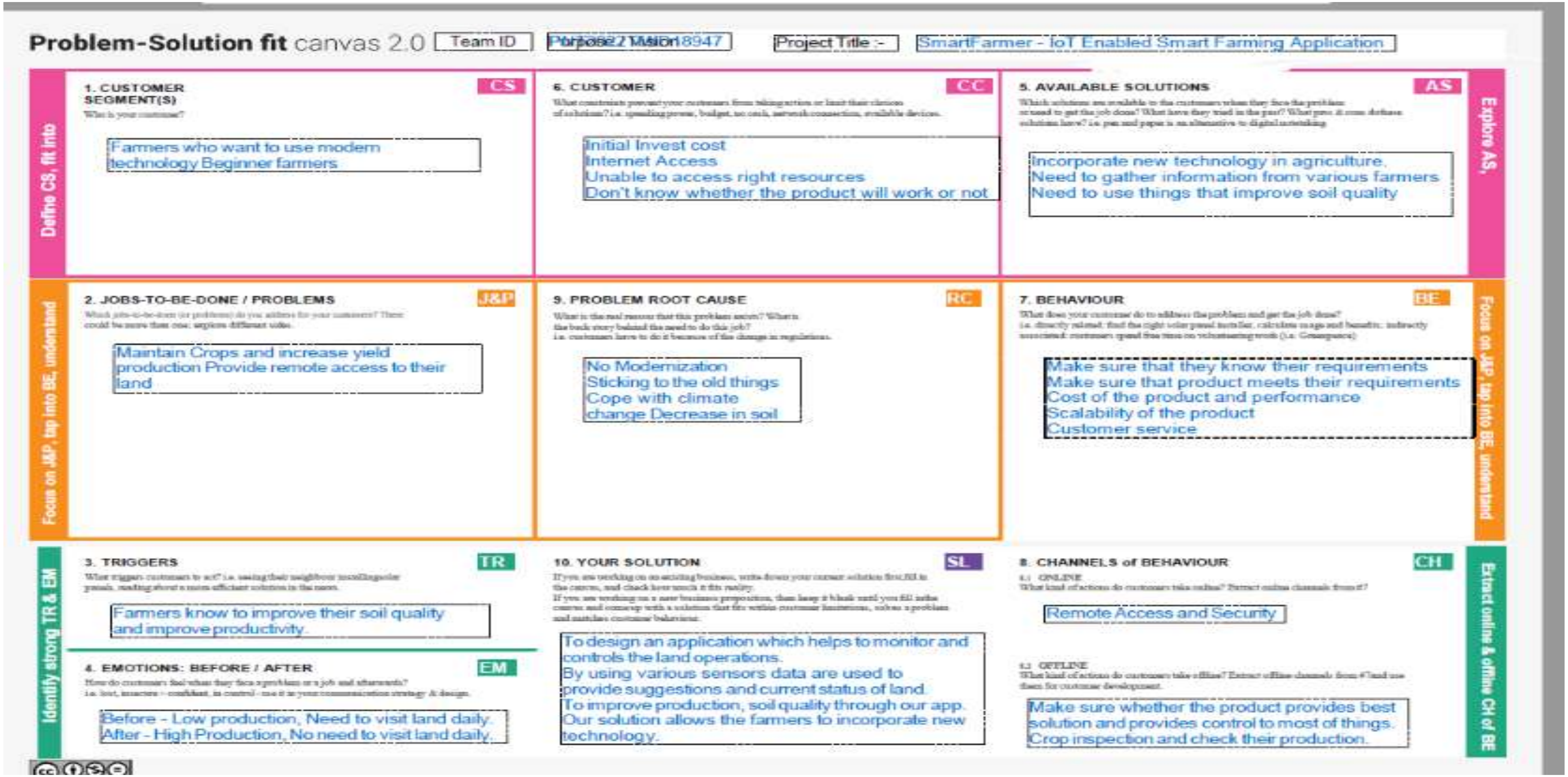
TEAM MEMBERS:

1.KRISHNAKUMAR S -MECR19EC045
2.MADHAVAN M -MECR19EC046
3.MADHAVARAJAN R -MECR19EC047

Solution and Technical Architecture



Problem Solution Fit



Problem-Solution fit canvas 2.0 | Team ID | PNT2022TMID18947 | Project Title :- | SmartFarmer - IoT Enabled Smart Farming Application

Define CS, fit into
Explore AS,
Focus on J&P, tap into BE, understand
Identify strong TR & EM
Extract online & offline CH of BE

1. CUSTOMER SEGMENT(S) CS

Who is your customer?

Farmers who want to use modern technology Beginner farmers

6. CUSTOMER CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

Initial Invest cost
Internet Access
Unable to access right resources
Don't know whether the product will work or not

5. AVAILABLE SOLUTIONS AS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

Incorporate new technology in agriculture.
Need to gather information from various farmers
Need to use things that improve soil quality

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

Maintain Crops and increase yield production Provide remote access to their land

9. PROBLEM ROOT CAUSE RC

What is the real reason that this problem exists? What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

No Modernization
Sticking to the old things
Cope with climate change Decrease in soil

7. BEHAVIOUR BE

What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Make sure that they know their requirements
Make sure that product meets their requirements
Cost of the product and performance
Scalability of the product
Customer service

3. TRIGGERS TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

Farmers know to improve their soil quality and improve productivity.

4. EMOTIONS: BEFORE / AFTER EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

Before - Low production, Need to visit land daily.
After - High Production, No need to visit land daily.

10. YOUR SOLUTION SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

To design an application which helps to monitor and controls the land operations.
By using various sensors data are used to provide suggestions and current status of land.
To improve production, soil quality through our app.
Our solution allows the farmers to incorporate new technology.

8. CHANNELS of BEHAVIOUR CH

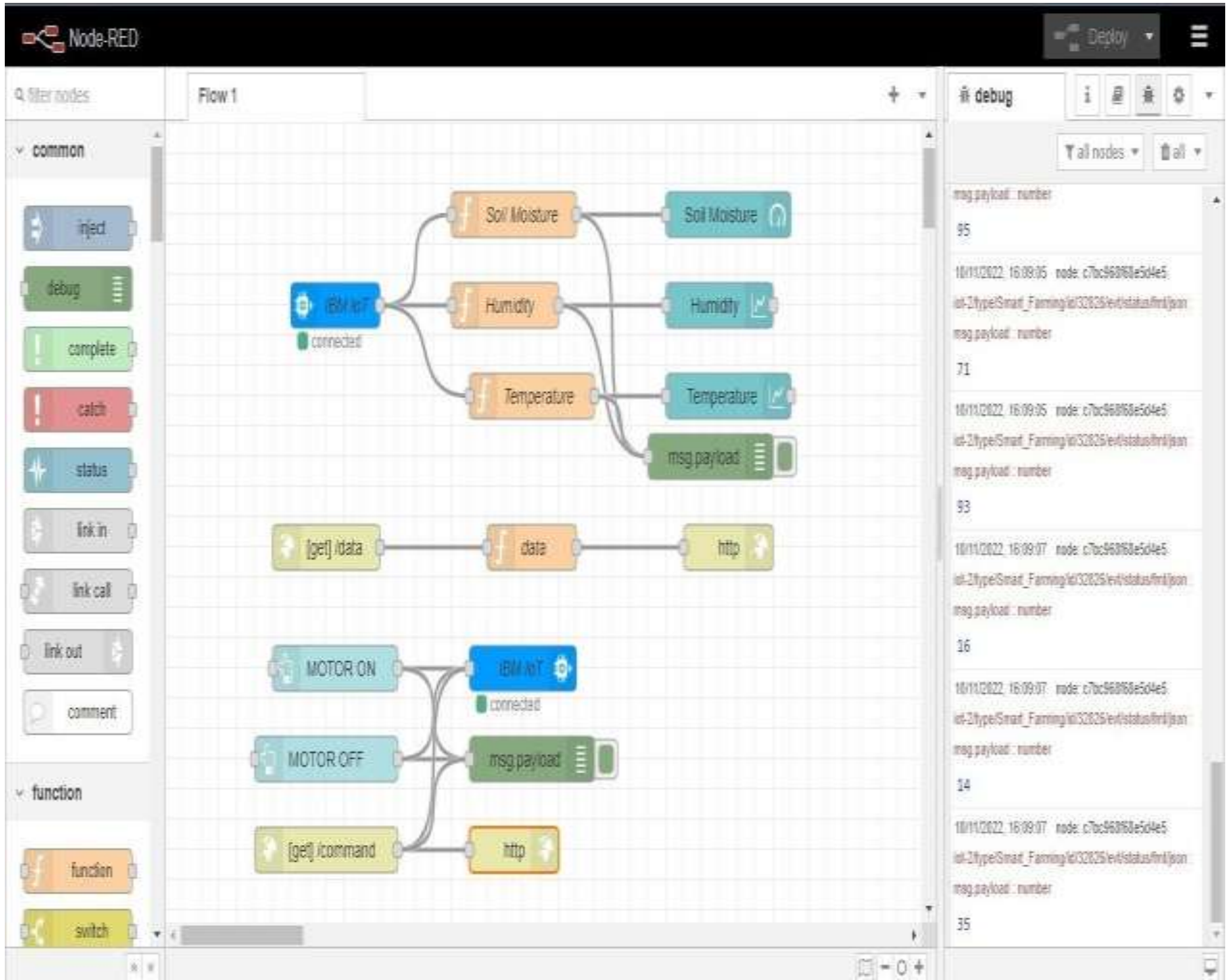
8.1 ONLINE
What kind of actions do customers take online? Extract online channels from #7

Remote Access and Security

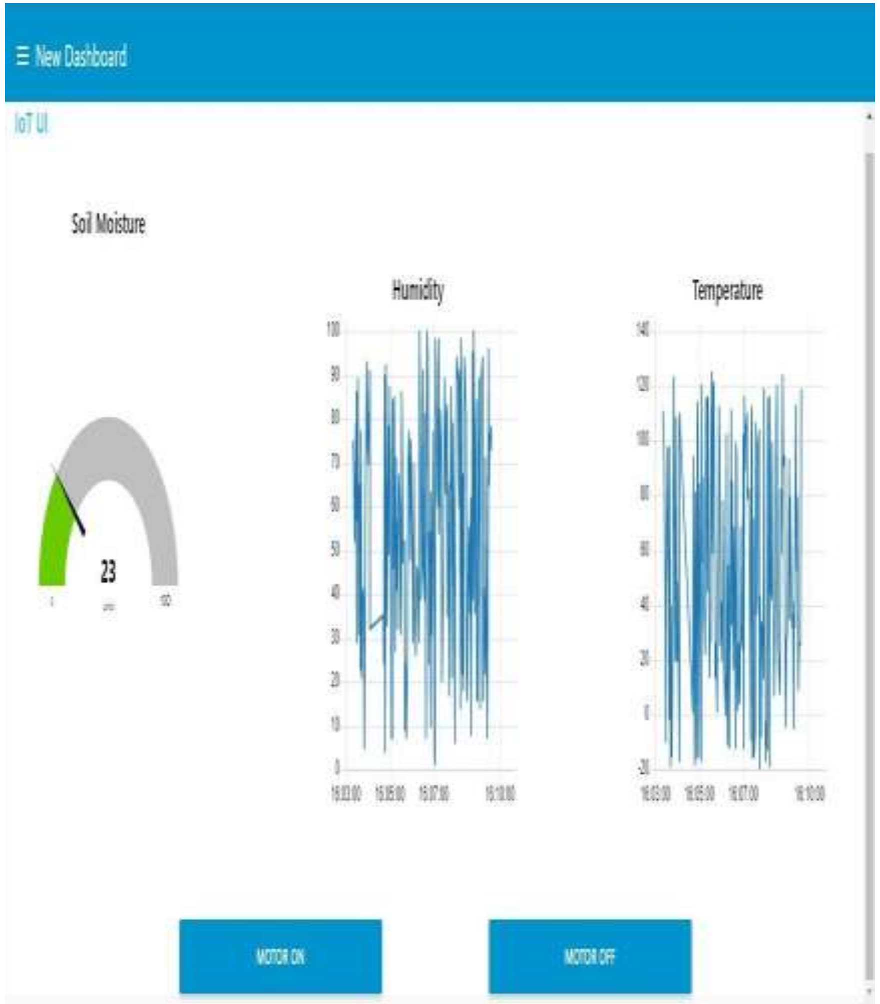
8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

Make sure whether the product provides best solution and provides control to most of things.
Crop inspection and check their production.

Node-RED &Soil Moisture Status



```
msg.payload : number
95
18/11/2022, 16:09:05   node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json
msg.payload : number
71
18/11/2022, 16:09:05   node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json
msg.payload : number
93
18/11/2022, 16:09:07   node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json
msg.payload : number
16
18/11/2022, 16:09:07   node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json
msg.payload : number
14
18/11/2022, 16:09:07   node: c7bc968f68e5d4e5
iot-2/type/Smart_Farming/id/32826/evt/status/fmt/json
msg.payload : number
35
```



Steps:

1. Create a Node-RED starter application

2. Create an Internet of Things Platform service

3. Enable the Continuous Delivery feature

4. Open the Node-RED visual programming editor

5. Configure your Node-RED application

6. Add extra nodes to your Node-RED palette

7. Launch the Watson IoT Platform

Step 1. Create a Node-RED starter application

Follow these steps to create a Node-RED starter application in the IBM Cloud.

1. Log in to IBM Cloud.

2. Open the catalog and search for **node-red**.

3. Click on the **Node-RED App** tile.

   This will show you an overview of the Starter Kit and what it provides.

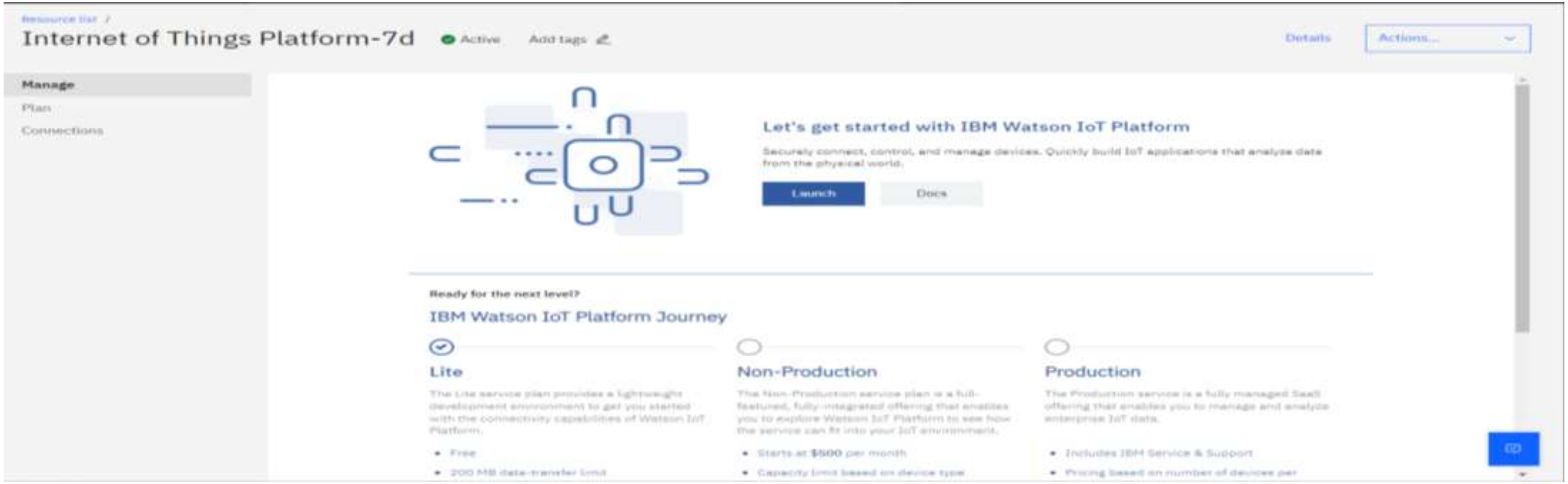
4. On the *Create* tab, a randomly generated **App name** will be suggested. Either accept that default name or provide a unique name for your application. This will become part of the application URL.

   Note: If the name is not unique, you will see an error message and you must enter a different name before you can continue.

5. The Node-RED starter application requires an instance of the **Cloudant database service** with IBM Cloud IAM and Cloudant credentials to store your application flow configuration. Select the region the service should be created in and what pricing plan it should use.

Step 2: Create an Internet of Things Platform service

The next step is to create an Internet of Things Platform service instance.



Internet of Things Platform-7d

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.

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
- 200 MB data-transfer limit

Non-Production

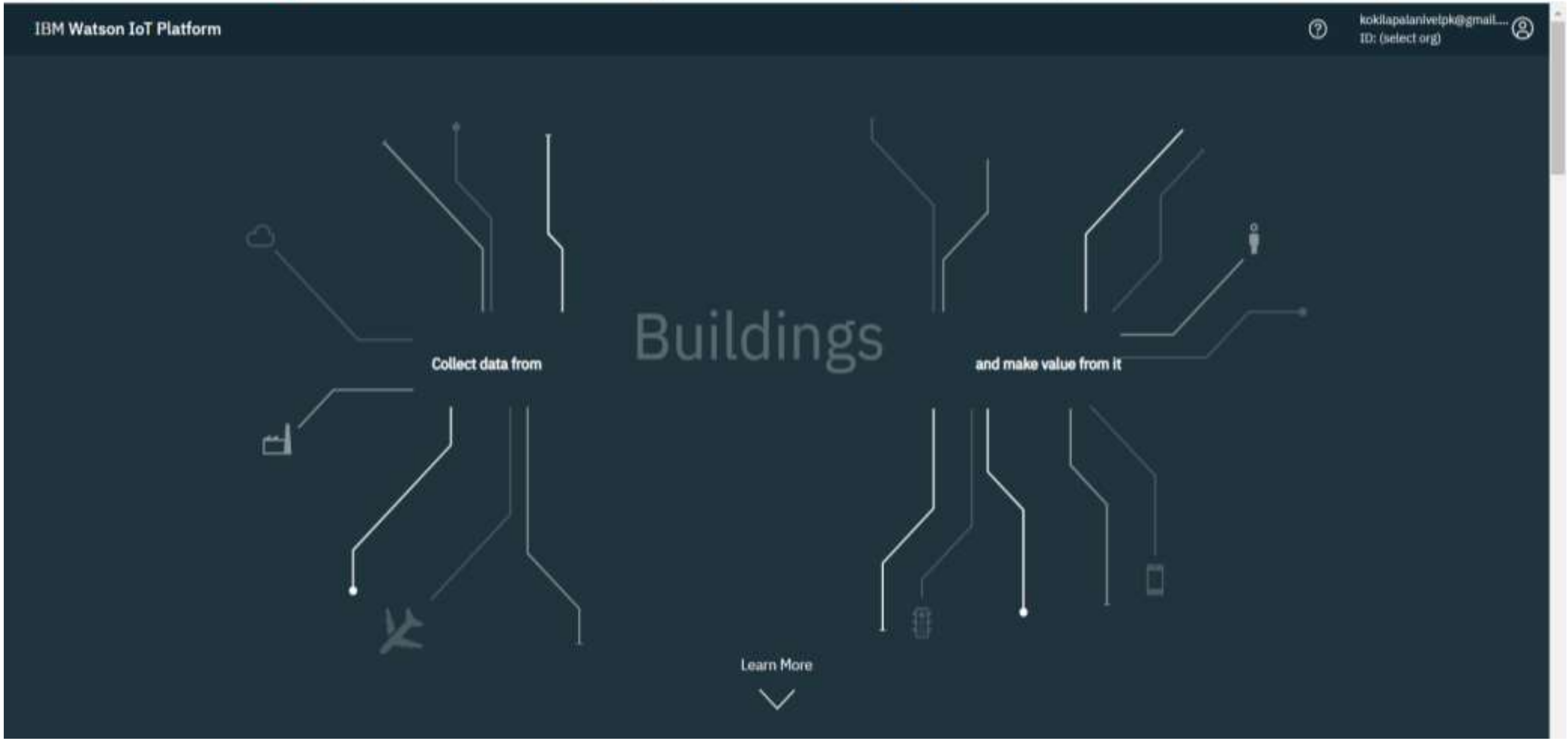
The Non-Production service plan is a full-featured, fully integrated offering that enables you to explore Watson IoT Platform to see how the service can fit into your IoT environment.

- Starts at **$500** per month
- Capacity limit based on device type

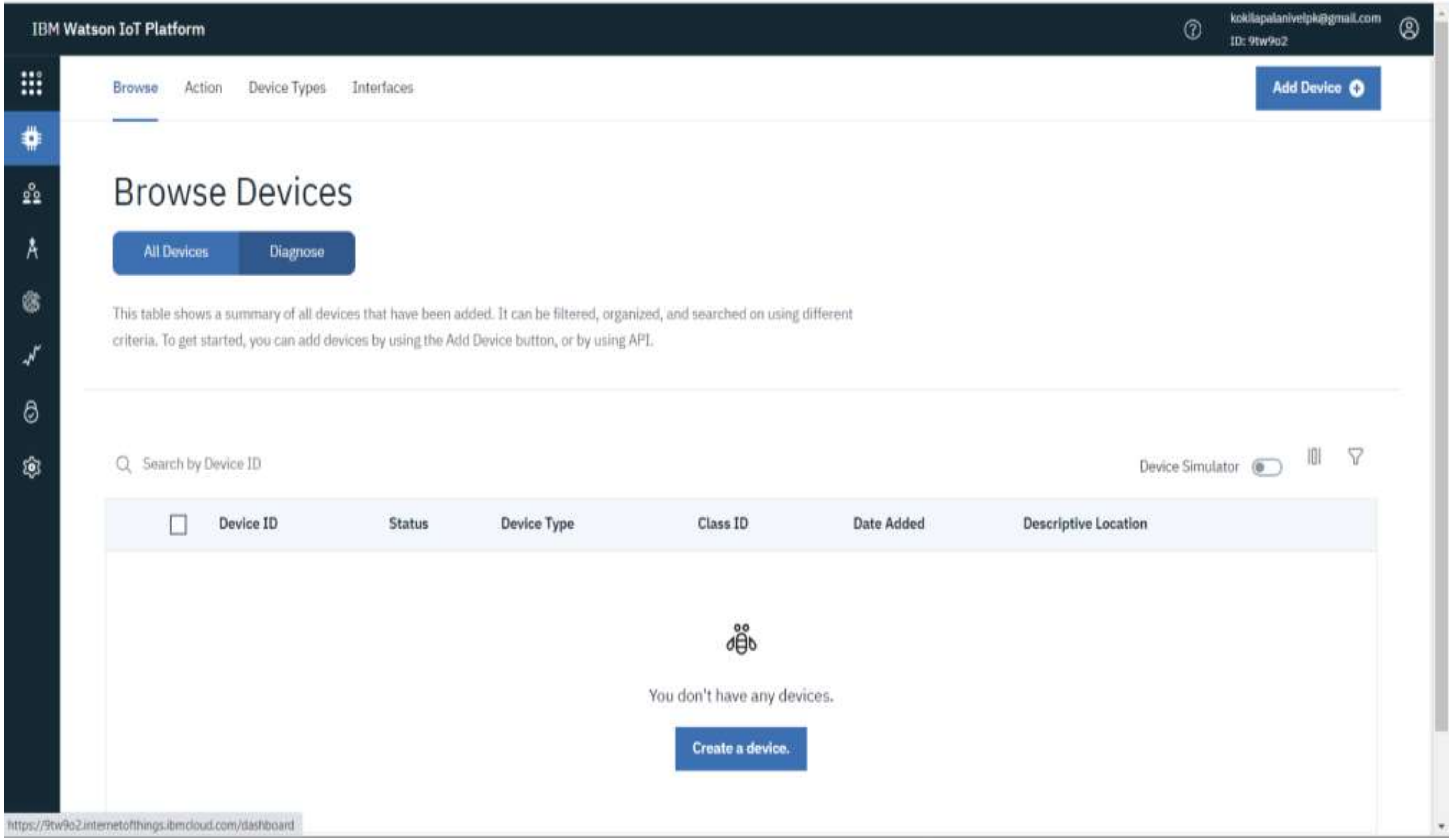
Production

The Production service is a fully managed SaaS offering that enables you to manage and analyze enterprise IoT data.

- Includes IBM Service & Support
- Pricing based on number of devices per


IBM Watson IoT Platform
kokilapalanivelpk@gmail....
ID: (select org)
Collect data from
Buildings
and make value from it
Learn More

1. Click on the **Create service** button.



Browse Devices

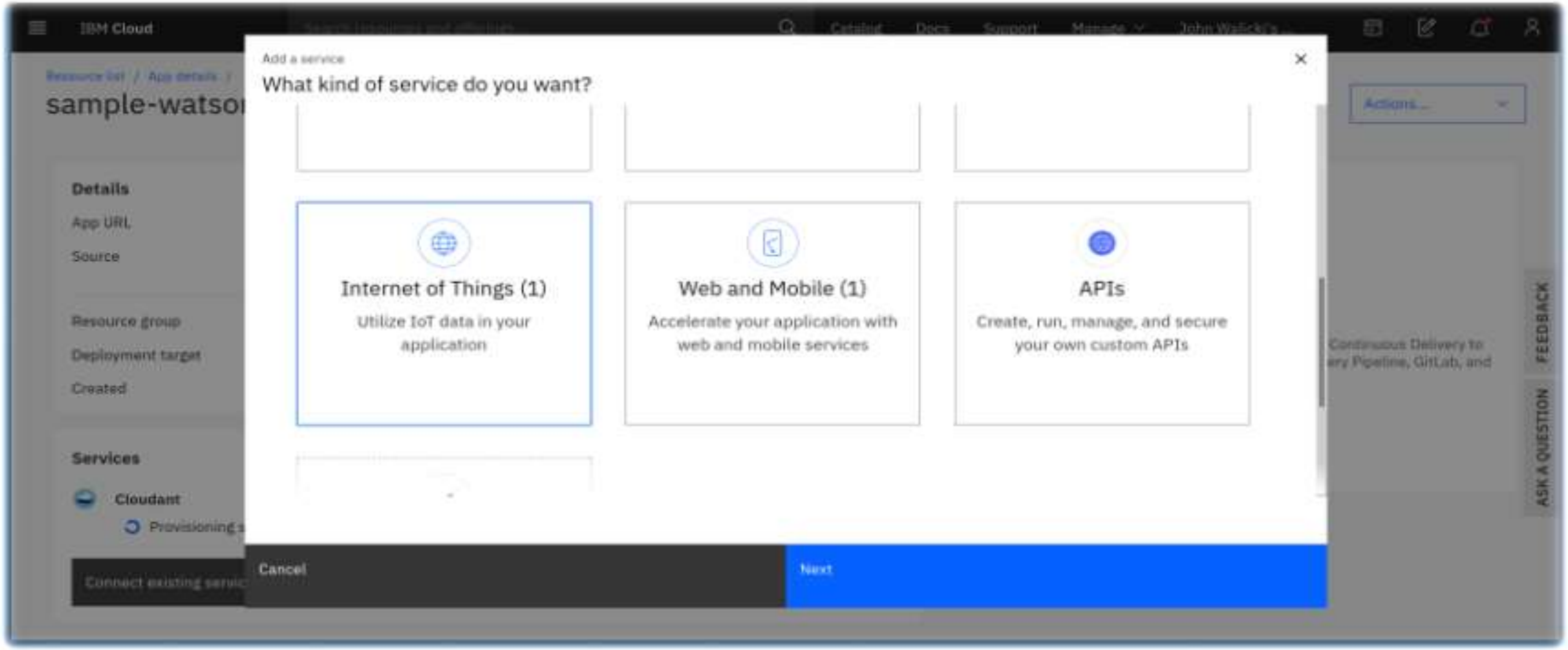
This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

□	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location

You don't have any devices.

2. Scroll down in the list of services to the **Internet of Things** card.

3. Select the **Internet of Things** card and click on the **Next** button.



Add a service

What kind of service do you want?

Internet of Things (1)

Utilize IoT data in your application

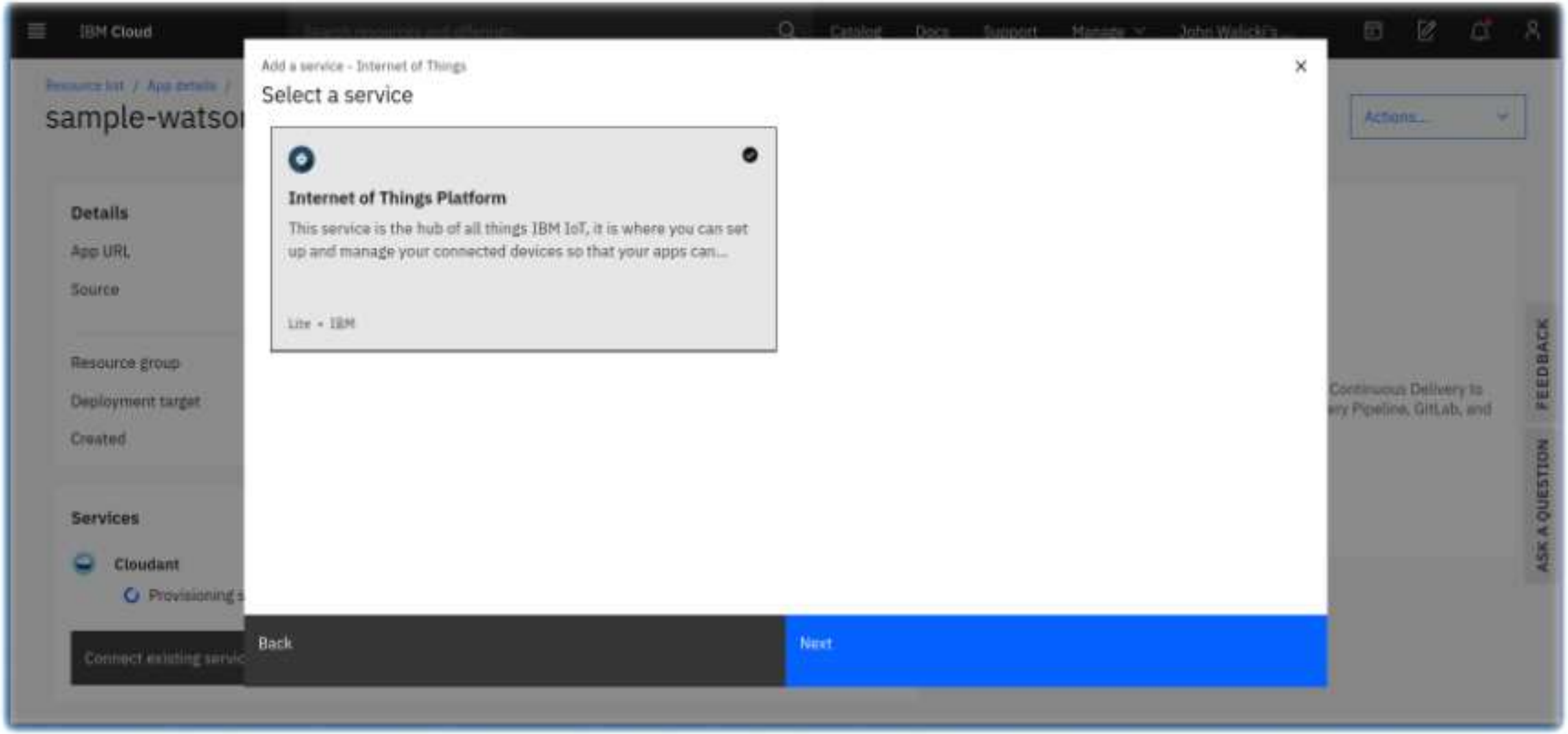
Web and Mobile (1)

Accelerate your application with web and mobile services

APIs

Create, run, manage, and secure your own custom APIs

4. Select the **Internet of Things Platform** card and click on the **Next** button.



Add a service - Internet of Things

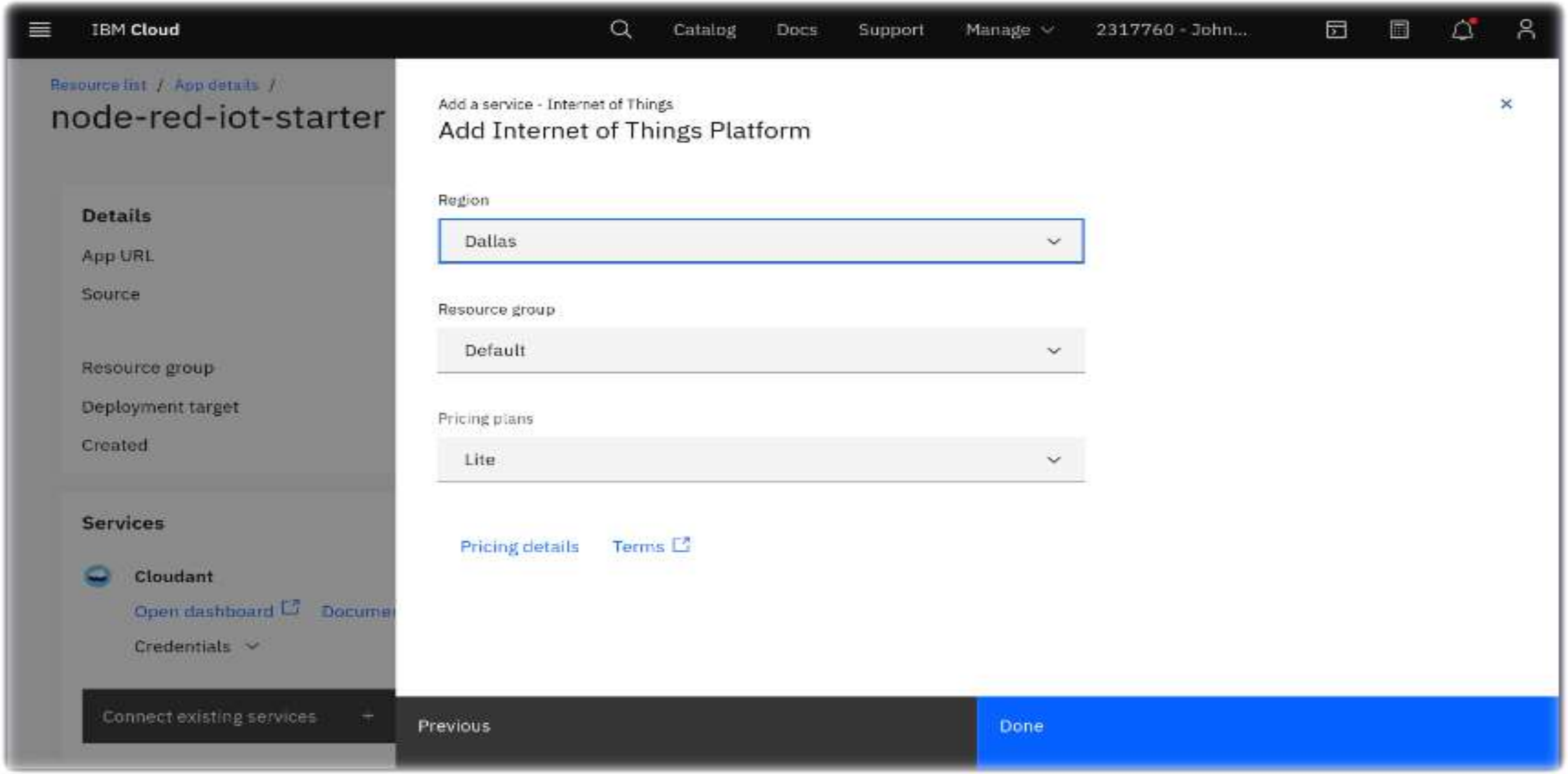
Select a service

Internet of Things Platform

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...

Lite • IBM

5. Select the region (try to match the region you selected previously), and then click **Done**.



Add a service - Internet of Things

Add Internet of Things Platform

Region: Dallas

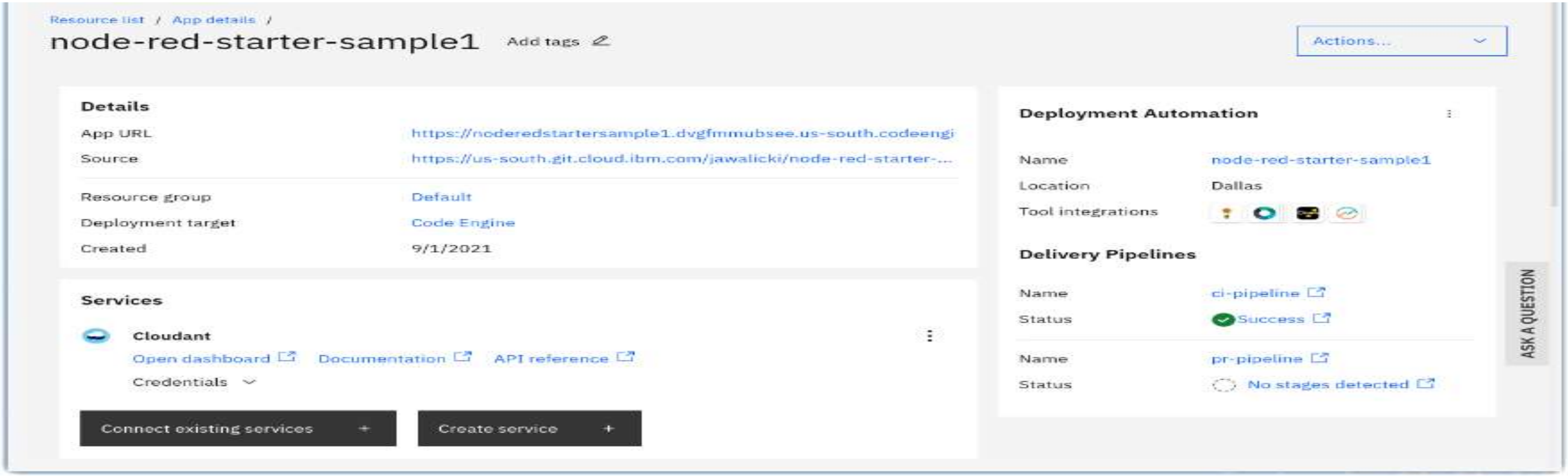
Resource group: Default

Pricing plans: Lite

Step 4. Open the Node-RED application

Now that you've deployed your Node-RED application, let's open it up! You may have to refresh your page.

On the application details page, you should now see the **App URL**, **Source** and **Deployment target** fields filled in.



Resource list / App details /

node-red-starter-sample1

Details

App URL https://noderedstartersample1.dvgfmmubsee.us-south.codeengi

Source https://us-south.git.cloud.ibm.com/jawalicki/node-red-starter-...

Resource group Default

Deployment target Code Engine

Created 9/1/2021

Services

Cloudant

Credentials

Deployment Automation

Name node-red-starter-sample1

Location Dallas

Tool integrations

Delivery Pipelines

Name ci-pipeline

Status Success

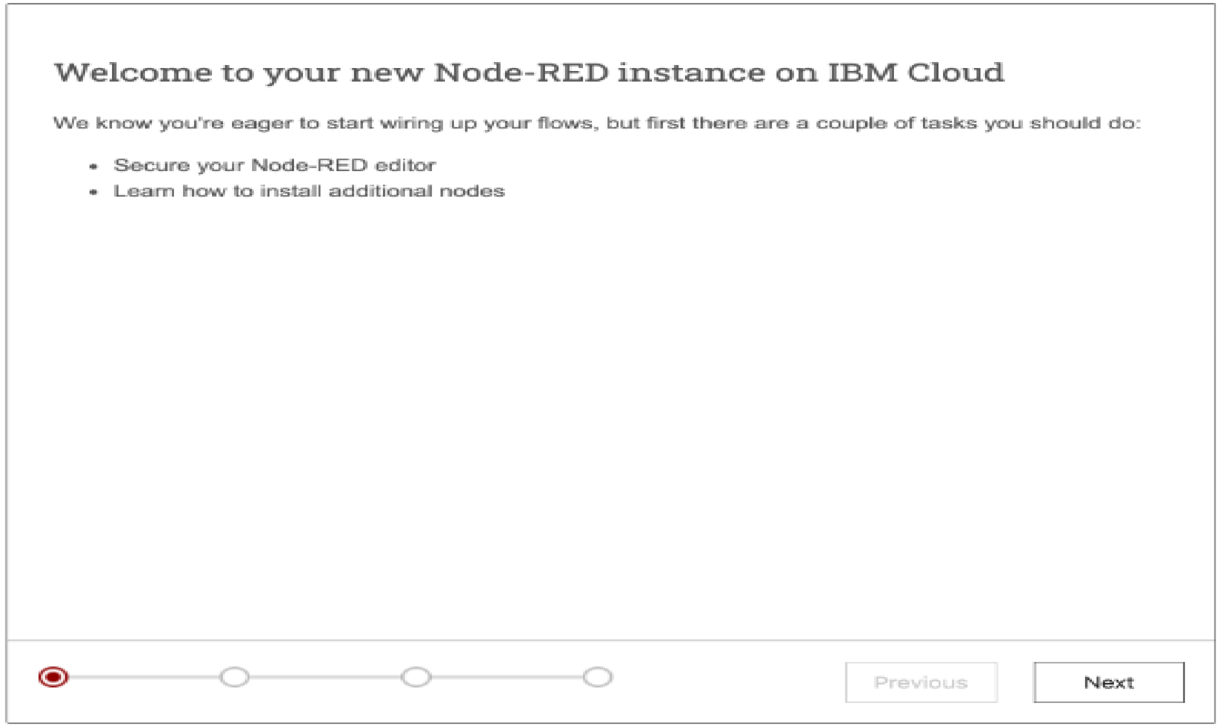
Name pr-pipeline

Status No stages detected

Step 5. Configure your Node-RED application

The first time you open your Node-RED app, you'll need to configure it and set up security.

1. A new browser tab will open with the Node-RED start page.



Welcome to your new Node-RED instance on IBM Cloud

We know you're eager to start wiring up your flows, but first there are a couple of tasks you should do:

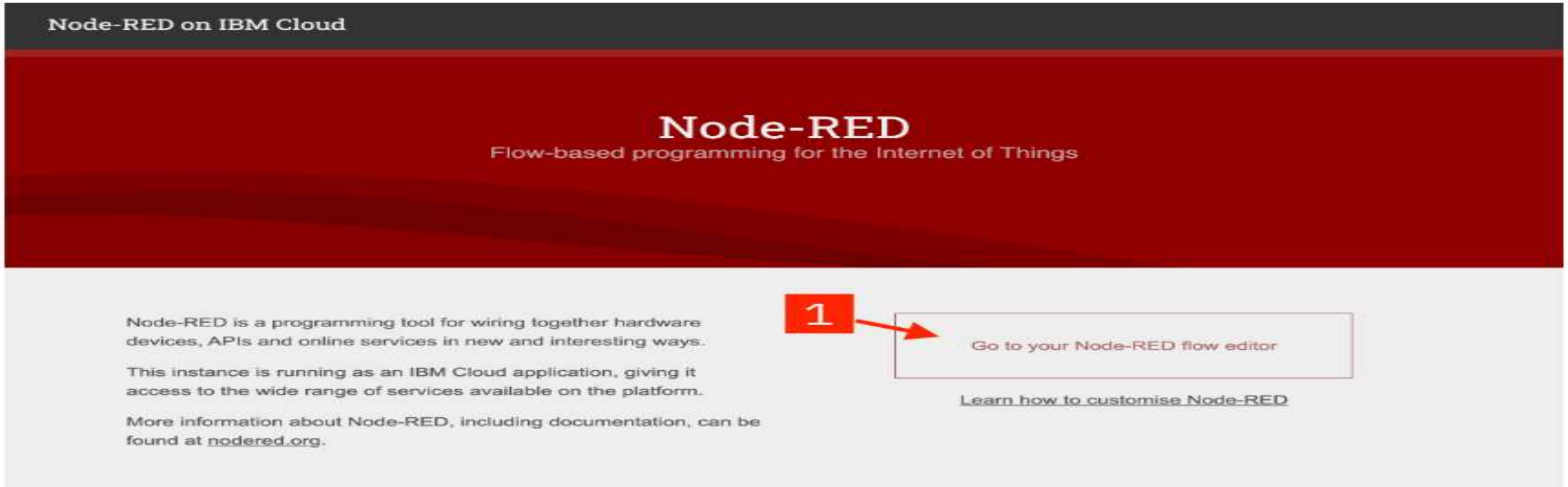
- Secure your Node-RED editor
- Learn how to install additional nodes

2. On the initial screen, click **Next** to continue.

3. Secure your Node-RED editor by providing a **username** and **password.** If you need to change these at any point, you can either edit the values in the Cloudant database, or override them using *environment variables*. The documentation on nodered.org ↗ describes how to do this. Click **Next** to continue.

4. The final screen summarizes the options you've made and highlights the environment variables you can use to change the options in the future. Click **Finish** to proceed.

5. Node-RED will save your changes and then load the main application. From here you can click the **Go to your Node-RED flow editor** button to open the editor.

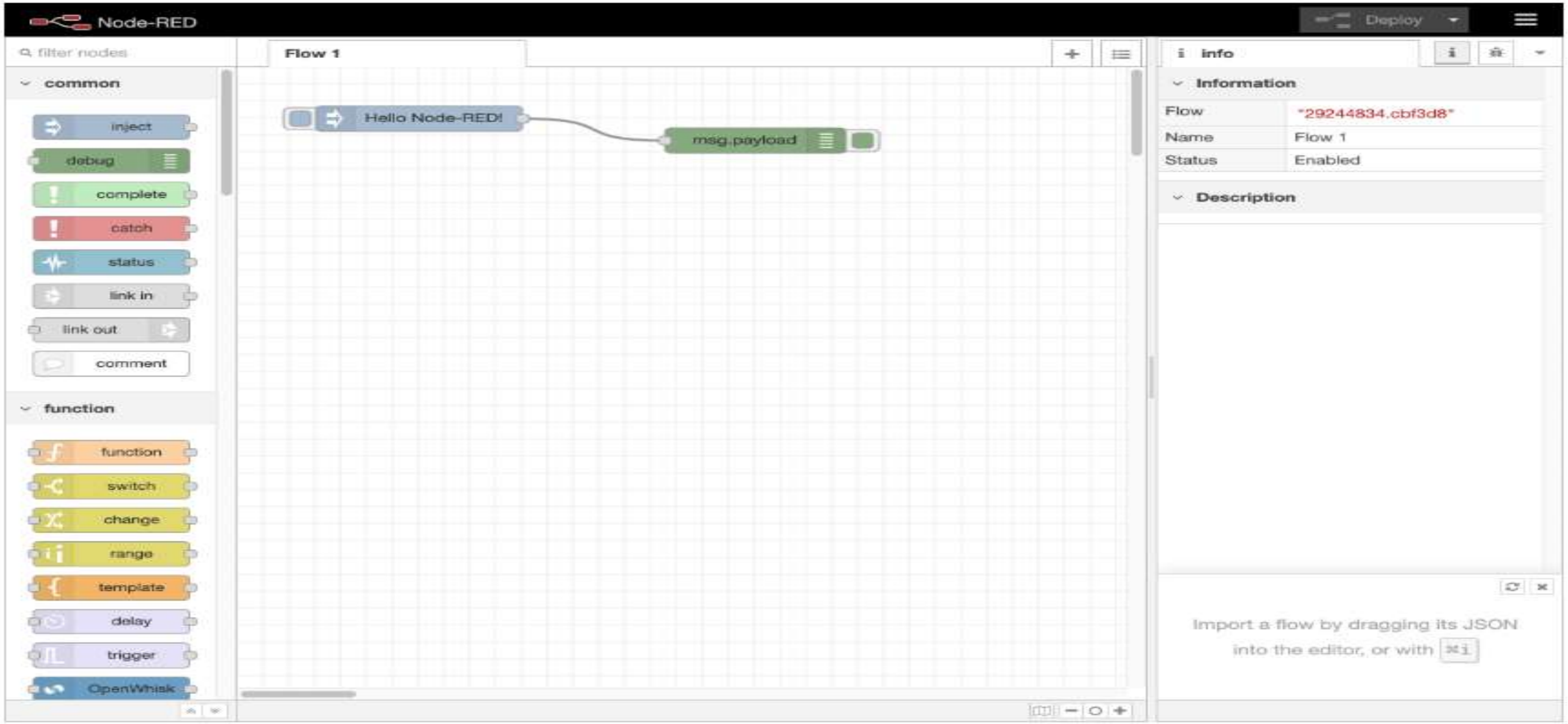

Node-RED on IBM Cloud
Node-RED
Flow-based programming for the Internet of Things
1
Go to your Node-RED flow editor
Learn how to customise Node-RED

Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.

This instance is running as an IBM Cloud application, giving it access to the wide range of services available on the platform.

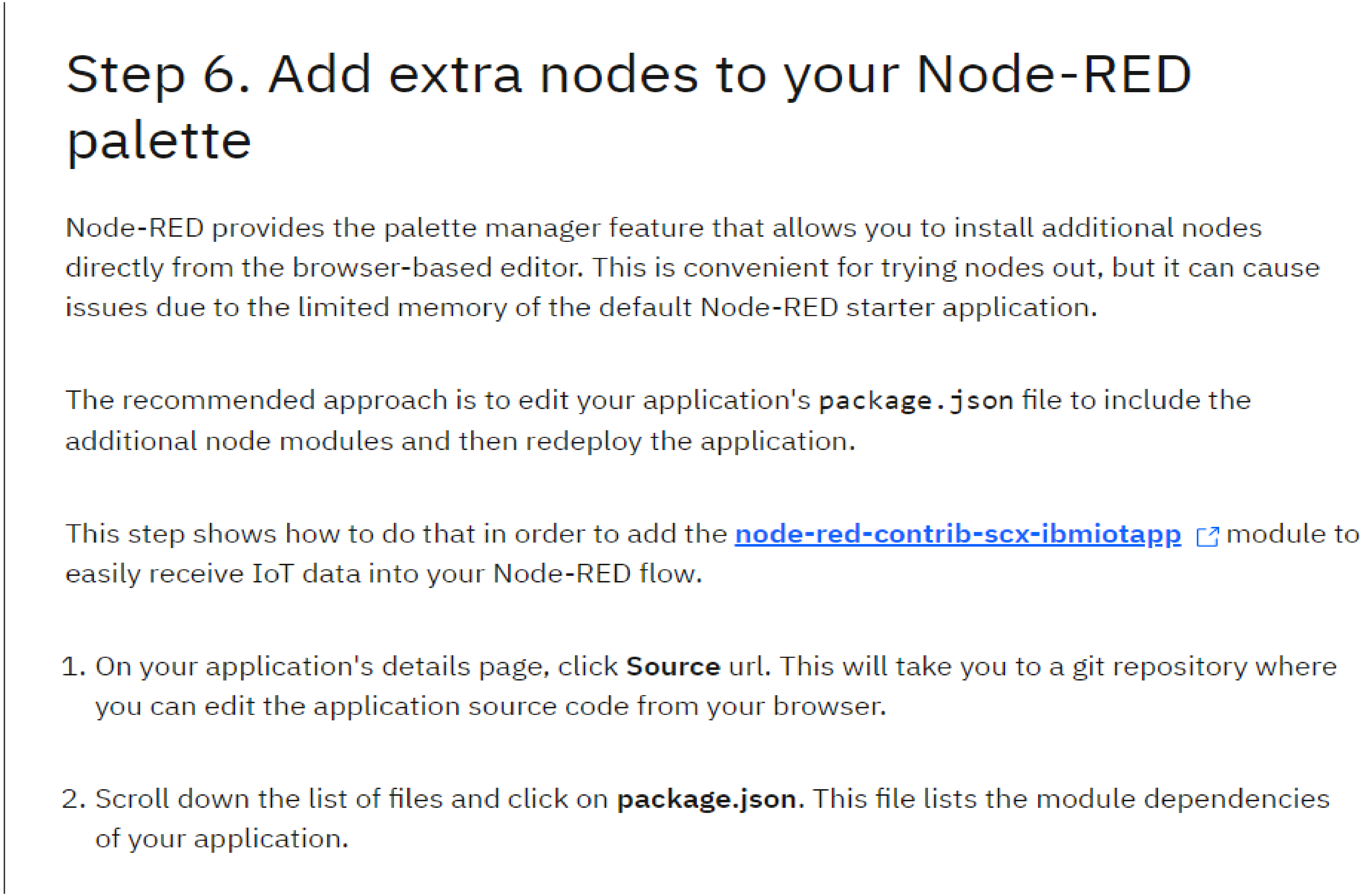
More information about Node-RED, including documentation, can be found at nodered.org.

The Node-RED editor opens showing the default flow.



Flow	"29244834.cbf3d8"
Name	Flow 1
Status	Enabled

Import a flow by dragging its JSON into the editor, or with ⌘i



Step 6. Add extra nodes to your Node-RED palette

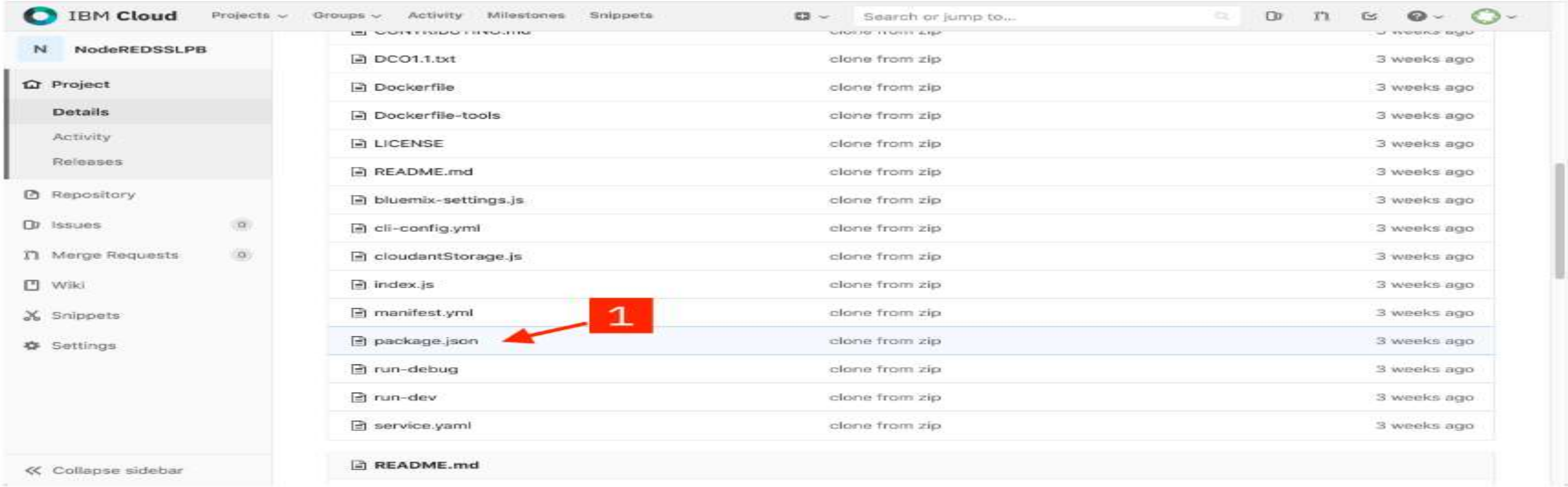
Node-RED provides the palette manager feature that allows you to install additional nodes directly from the browser-based editor. This is convenient for trying nodes out, but it can cause issues due to the limited memory of the default Node-RED starter application.

The recommended approach is to edit your application's `package.json` file to include the additional node modules and then redeploy the application.

This step shows how to do that in order to add the node-red-contrib-scx-ibmiotapp module to easily receive IoT data into your Node-RED flow.

1. On your application's details page, click **Source** url. This will take you to a git repository where you can edit the application source code from your browser.

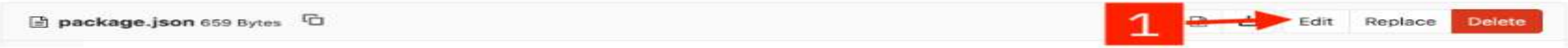
2. Scroll down the list of files and click on **package.json**. This file lists the module dependencies of your application.



DCO1.1.txt	clone from zip	3 weeks ago
Dockerfile	clone from zip	3 weeks ago
Dockerfile-tools	clone from zip	3 weeks ago
LICENSE	clone from zip	3 weeks ago
README.md	clone from zip	3 weeks ago
bluemix-settings.js	clone from zip	3 weeks ago
cli-config.yml	clone from zip	3 weeks ago
cloudantStorage.js	clone from zip	3 weeks ago
index.js	clone from zip	3 weeks ago
manifest.yml	clone from zip	3 weeks ago
package.json	clone from zip	3 weeks ago
run-debug	clone from zip	3 weeks ago
run-dev	clone from zip	3 weeks ago
service.yaml	clone from zip	3 weeks ago

README.md

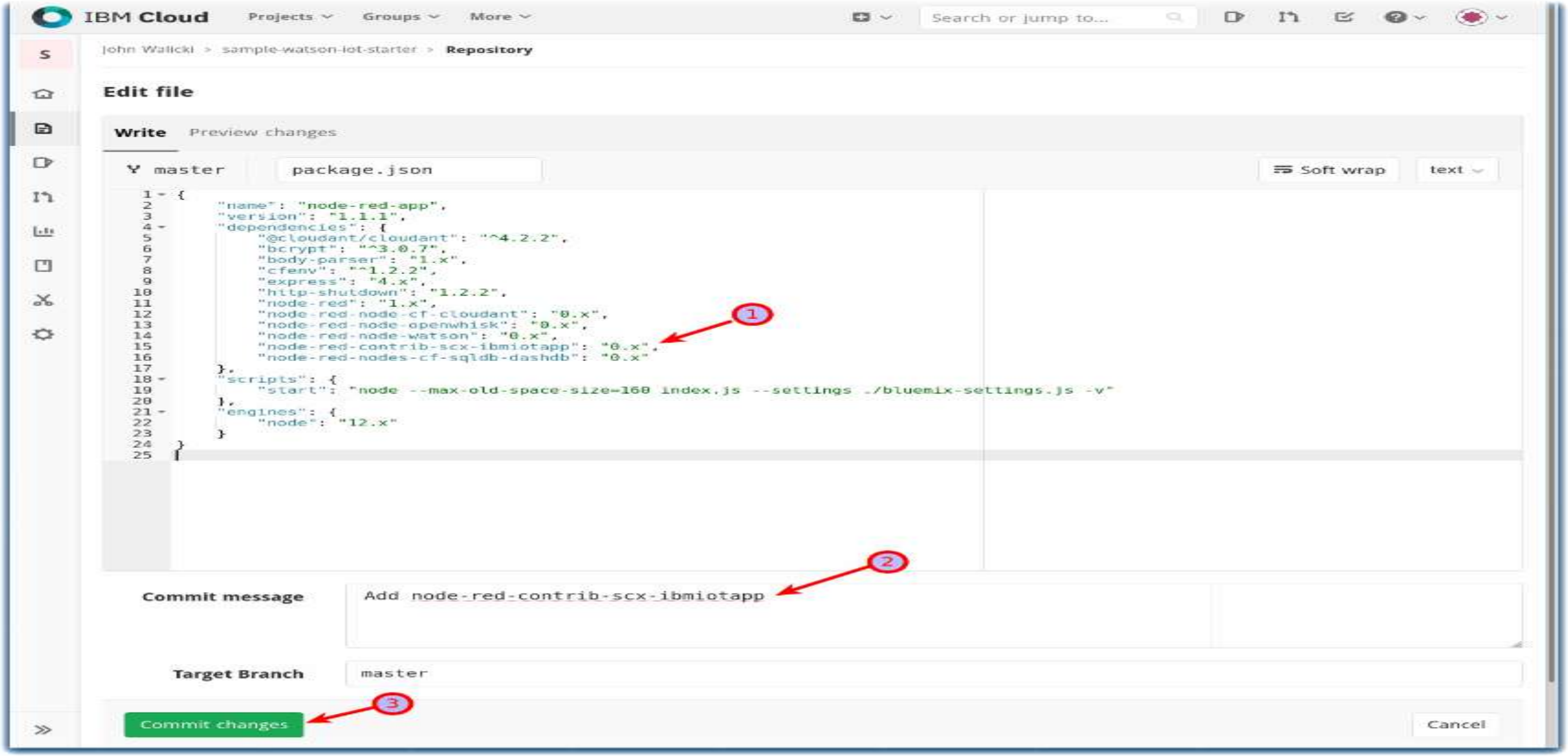
3. Click the **Edit** button



4. Add the following entry to the top of the **dependencies** section (1):

```
"node-red-contrib-scx-ibmiotapp": "0.x",
```

Add a **Commit message** (2) and click **Commit changes** (3)

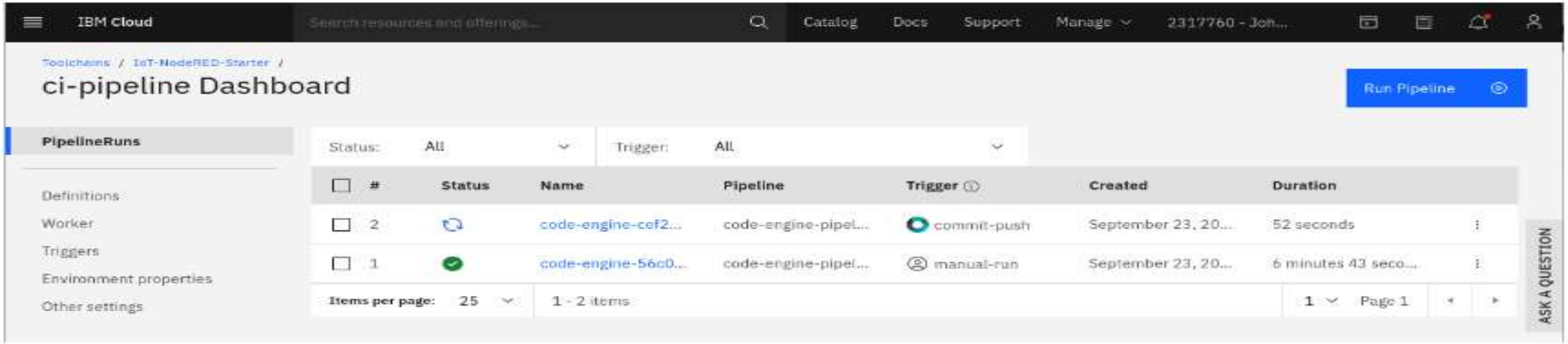


```
{
    "name": "node-red-app",
    "version": "1.1.1",
    "dependencies": {
        "@cloudant/cloudant": "^4.2.2",
        "bcrypt": "^3.0.7",
        "body-parser": "1.x",
        "cfenv": "~1.2.2",
        "express": "4.x",
        "http-shutdown": "1.2.2",
        "node-red": "1.x",
        "node-red-node-cf-cloudant": "0.x",
        "node-red-node-openwhisk": "0.x",
        "node-red-node-watson": "0.x",
        "node-red-contrib-scx-ibmiotapp": "0.x",
        "node-red-nodes-cf-sqldb-dashdb": "0.x"
    },
    "scripts": {
        "start": "node --max-old-space-size=160 index.js --settings ./bluemix-settings.js -v"
    },
    "engines": {
        "node": "12.x"
    }
}
```

Commit message Add node-red-contrib-scx-ibmiotapp

Target Branch master

5. At this point, the Continuous Delivery pipeline will automatically run to build and deploy that change into your application. If you view the Delivery Pipeline you can watch its progress. The Build section shows you the last commit made and the Deploy section shows the progress of redeploying the application.



ci-pipeline Dashboard

Status: All

Trigger: All

□	#	Status	Name	Pipeline	Trigger	Created	Duration
□	2	↻	code-engine-cef2...	code-engine-pipel...	commit-push	September 23, 20...	52 seconds
□	1	✓	code-engine-56c0...	code-engine-pipel...	manual-run	September 23, 20...	6 minutes 43 seco...

Items per page: 25 · 1 - 2 items · 1 · Page 1

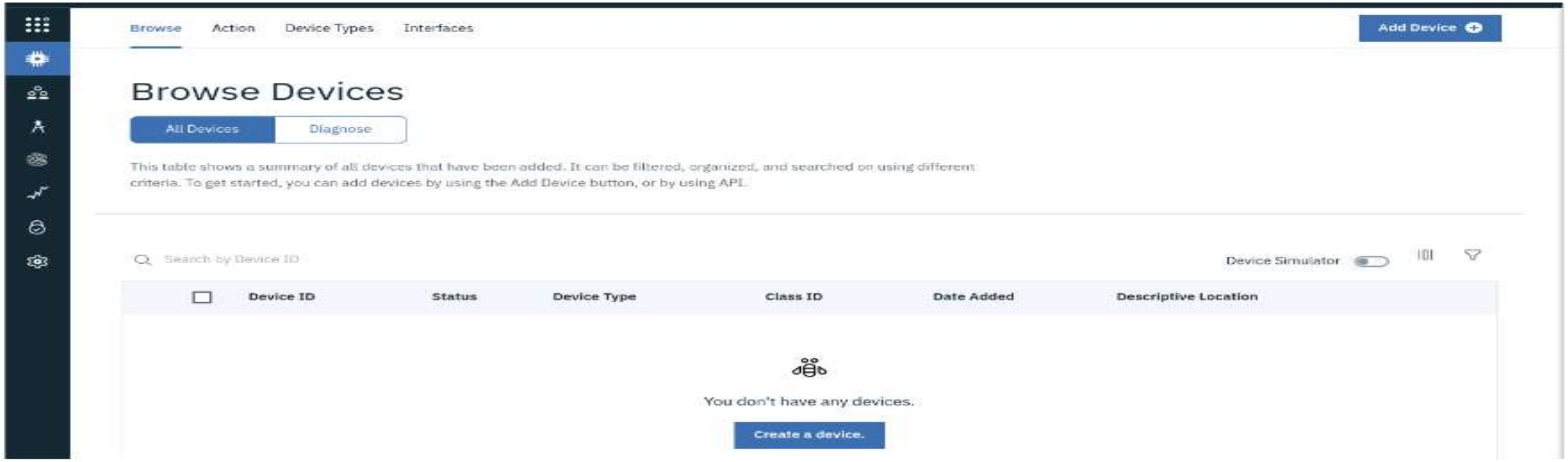
6. Once the Deploy stage completes, your application will have restarted and now have the `node-red-contrib-scx-ibmiotapp` nodes preinstalled.

Step 7. Launch the Watson IoT Platform

Now that you've configured the services and setup the Node-RED Visual Programming Editor, you can launch the Watson IoT Platform.

1. Return to the [IBM Cloud Application Details](#) page.

2. In the **Services** section, for the Internet of Things Platform service, click the **Open Dashboard** link.

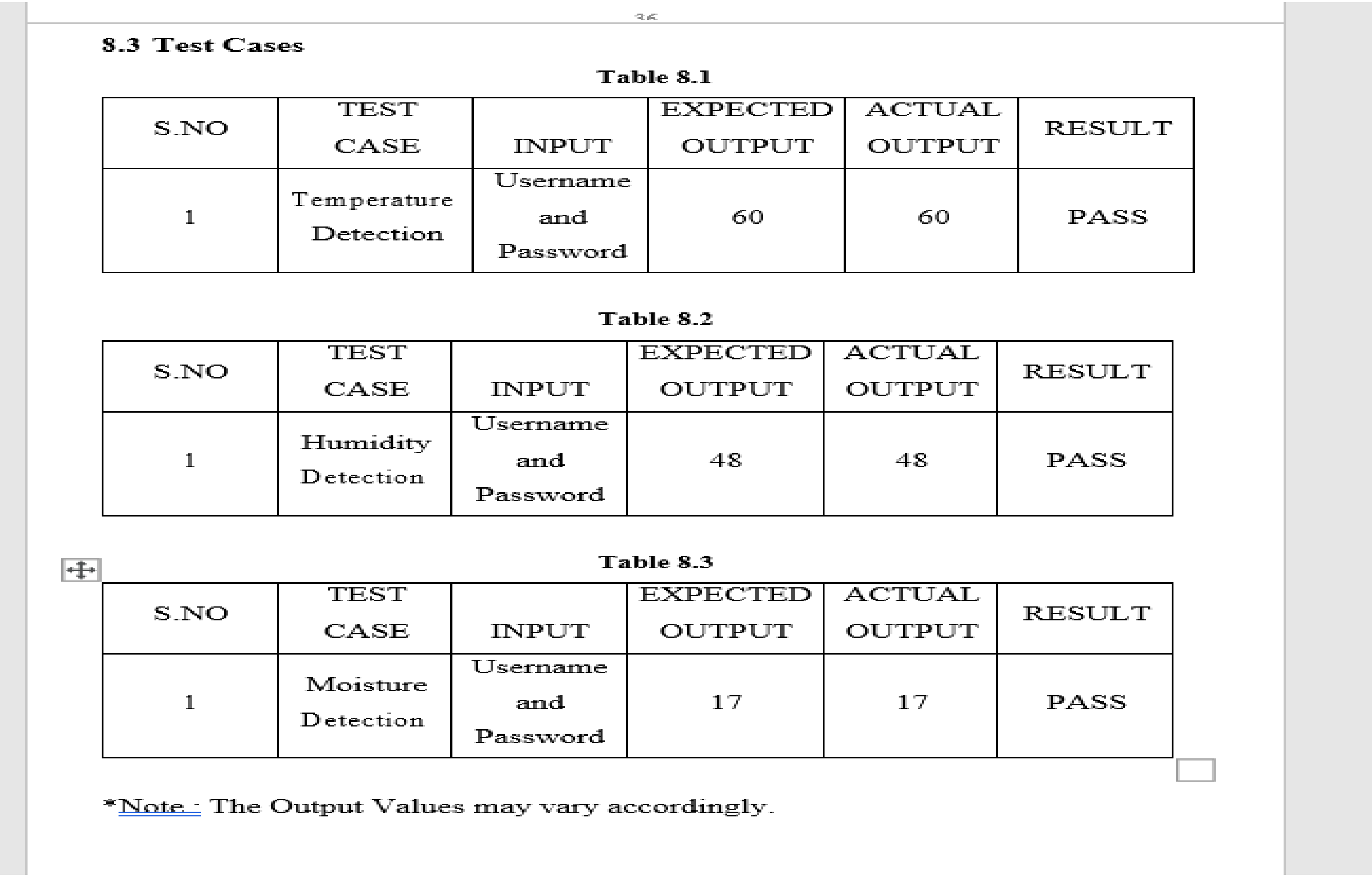


Browse Devices

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.

□	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location

You don't have any devices.



8.3 Test Cases

Table 8.1

S.NO	TEST CASE	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	RESULT
1	Temperature Detection	Username and Password	60	60	PASS

Table 8.2

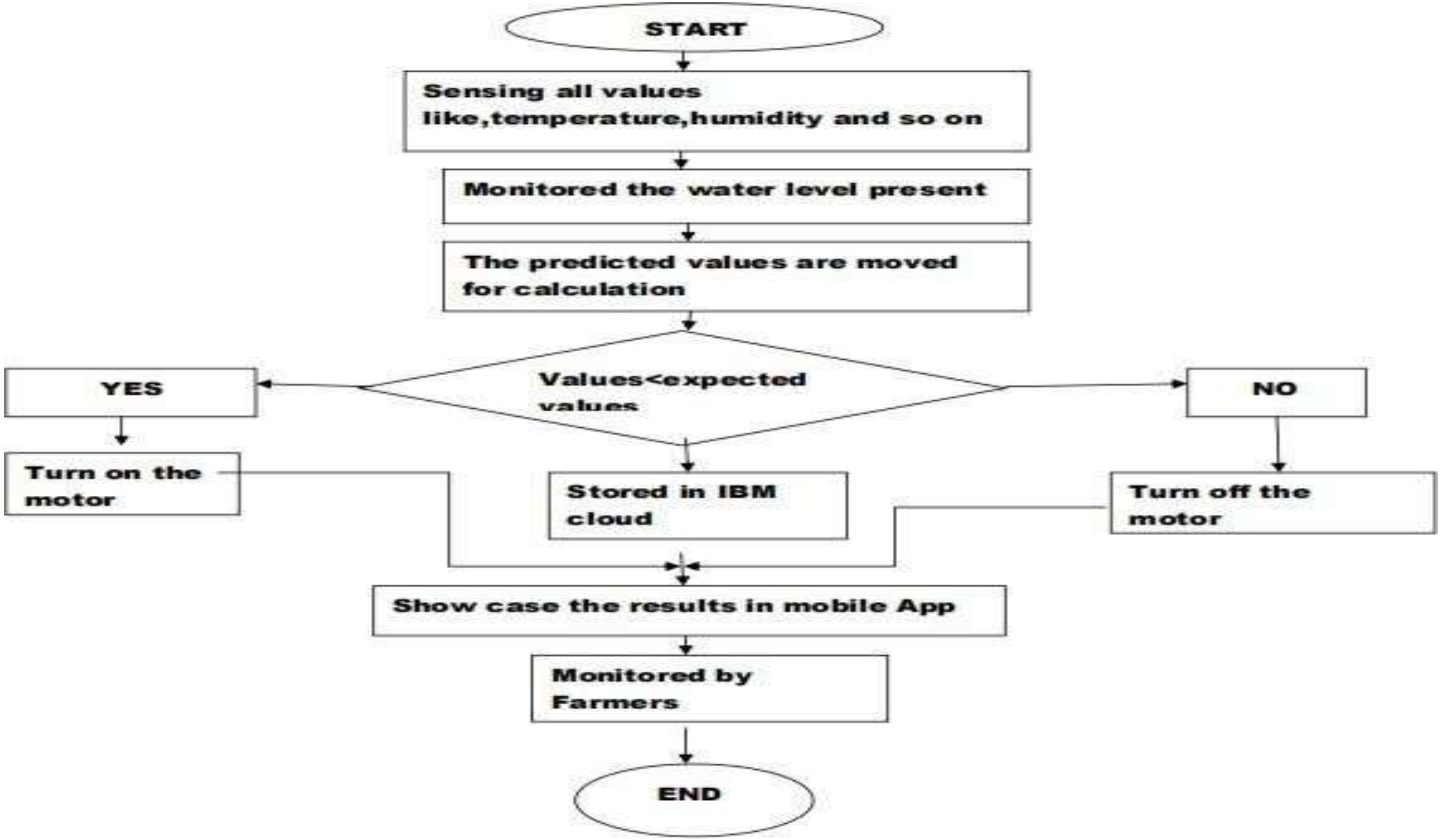
S.NO	TEST CASE	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	RESULT
1	Humidity Detection	Username and Password	48	48	PASS

Table 8.3

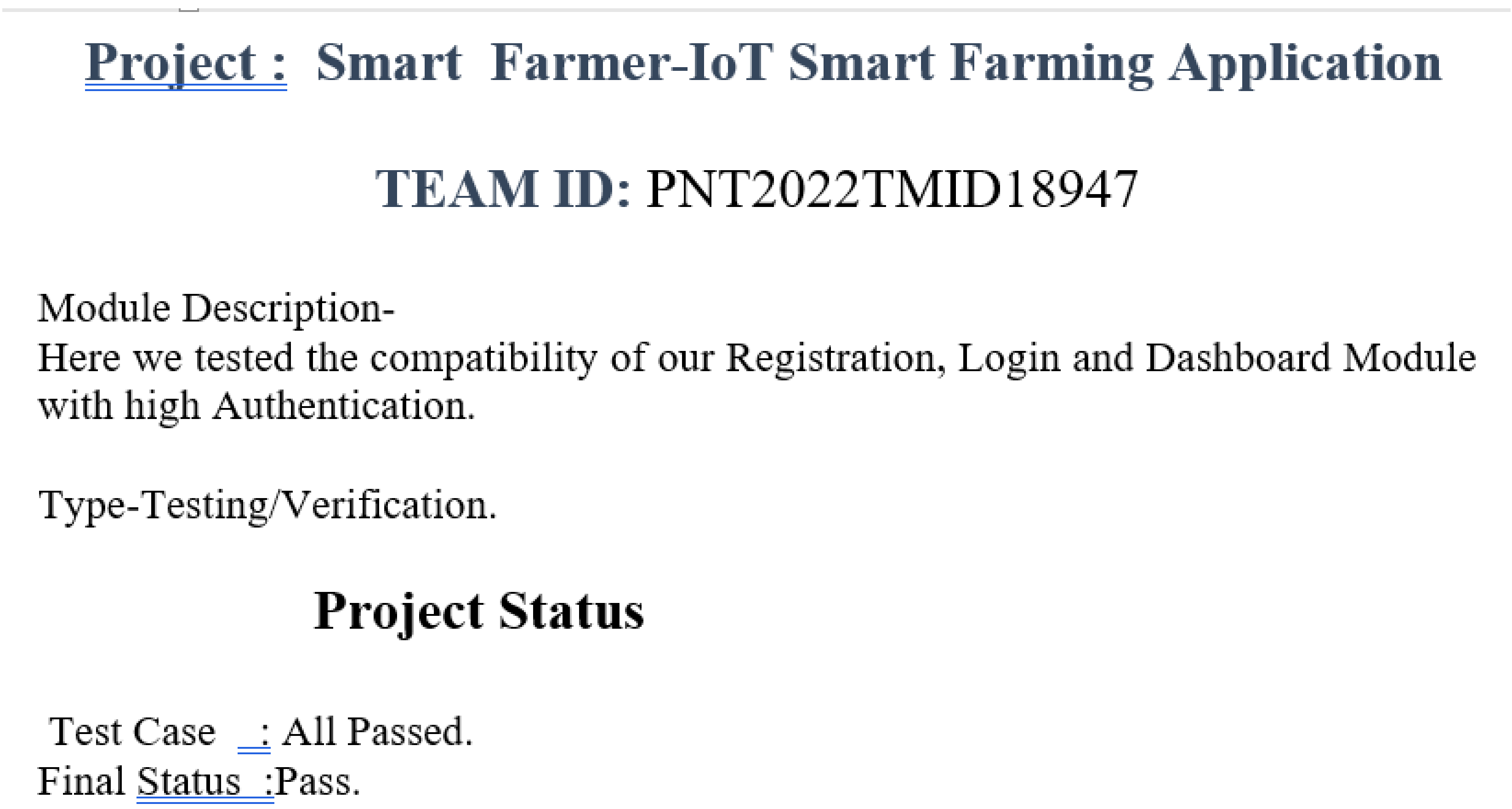
S.NO	TEST CASE	INPUT	EXPECTED OUTPUT	ACTUAL OUTPUT	RESULT
1	Moisture Detection	Username and Password	17	17	PASS

*Note : The Output Values may vary accordingly.

Data Flow Diagram:


START
Sensing all values like,temperature,humidity and so on
Monitored the water level present
The predicted values are moved for calculation
Values<expected values
YES
NO
Turn on the motor
Stored in IBM cloud
Turn off the motor
Show case the results in mobile App
Monitored by Farmers
END

Testing Results of Sprint 1:



Project : Smart Farmer-IoT Smart Farming Application

TEAM ID: PNT2022TMID18947

Module Description-
Here we tested the compatibility of our Registration, Login and Dashboard Module with high Authentication.

Type-Testing/Verification.

Project Status

Test Case : All Passed.
Final Status :Pass.

ADVANTAGES AND DISADVANTAGES

Advantages:

- As it is a mobile friendly application one can access all the metrics in one touch.
- It has clean User interface so that user have smooth control over the application.
- The consumption of electric power is less as compared to other application.
- The moisture level and the temperature levels are monitored at regular intervals.
- It can run on all android versions.
- The application requires less memory and storage space.

Disadvantages:

- When the network connectivity is poor the performance of the application will beaffected
- As it is platform dependent it cannot run on all devices.
- The application will produce inaccurate values when there is a fault or any change inAPI.
- The user should be more aware on the results produced.

CONCLUSION

- In this work, we successfully develop a system that can help in an automatedirrigation system by analyzing the moisture level of the ground.
- The smart irrigation system proves to be a useful system as it automates and regulates the watering without any manual intervention. The primary applications for this project are for farmers and gardeners who do not have enough time to watercrops/plants.
- The farmers are facing major problems in watering their agriculture fields. So that the Farmers can Watering their plant Smart.

FUTURE SCOPE

- It helps in automatic irrigation for crops and also helps to maintain the water levelin field.
- The system will notify on the critical conditions. As this is an automated device it can works even the absence

❖ References

Mobile Integrated Smart Irrigation Management and Monitoring System Using IOTDate of Conference: 06-08
April 2017Publisher: IEEE Date Added to IEEE Xplore: 08February 2018DOI:
10.1109/ICCSP.2017.8286792
IoT Based Smart Irrigation Monitoring And Controlling SystemDate Added to IEEE Xplore: 15 January 2018 ISBN Information:
Electronic ISBN: 978-1-5090...Dateof Conference: 19-20 May 2017
INSPEC Accession Number: 17504411
Smart Waste Collection Monitoring and Alert System via IoTDate Added to IEEE Xplore:
24 June 2019DOI:
10.1109/ISCAIE.2019.874376
Print on Demand (PoD) ISBN: 978-1-5386-854