A Project Report
on
IoT based SMART AGRICULTURE SYSTEM

Table of Contents

ABSTRACT

The Internet of Things (IoT) technology has brought revolution to every field of common man's life by making everything smart and intelligent. IoT refers to a network of things which make a self configuring network. The development of Intelligent Smart Farming IoT based devices is day by day turning the face of agriculture production by not only enhancing it but also making it cost-effective and reducing wastage. The aim / objective of this report is to propose IoT based Smart Agriculture System to assist farmers in getting Live Data (Temperature, Humidity and Soil Moisture) for efficient environment monitoring which will enable them to increase their overall yield and quality of products. The Farmer gets the real time weather forecast data by using external platforms like Open Weather API. Based on all the parameters he can water his crop by controlling the motors using the mobile application. Even if the farmer is not present near his crop he can water his crop by controlling the motors using the mobile application from anywhere.

1. INTRODUCTION

1.1 Overview

The objective of this report is to propose the IoT based on the Smart Agriculture System which will enable the farmers to have live data of soil moisture,temperature,humidity values and can remotely control the motor on and off to water the crop.

1.2 Purpose

The purpose of this project is to yield the crop properly by getting the live data of soil moisture, temperature, humidity. The farmer can see the live values and according to that, He can sow the seeds and water his land, moreover it saves a substantial amount of time. The farmer can activate his motor which is in the farm land with just one click.

2. LITERATURE SURVEY

2.1 Existing problem

Farming is known as the backbone of our country. Due to high pollution the temperatures are not recording with stable values, they are varying day by day. So, due to this sudden changes in the atmosphere the farmer is not able to estimate the temperature, humidity also soil moisture, hence the crop is not yielding properly and farmers are going into debts. To water the land the farmer must walk miles on his bare foot or peddle his cycle for a long distance, it is consuming huge amount of time and it is laborious work to do everyday.

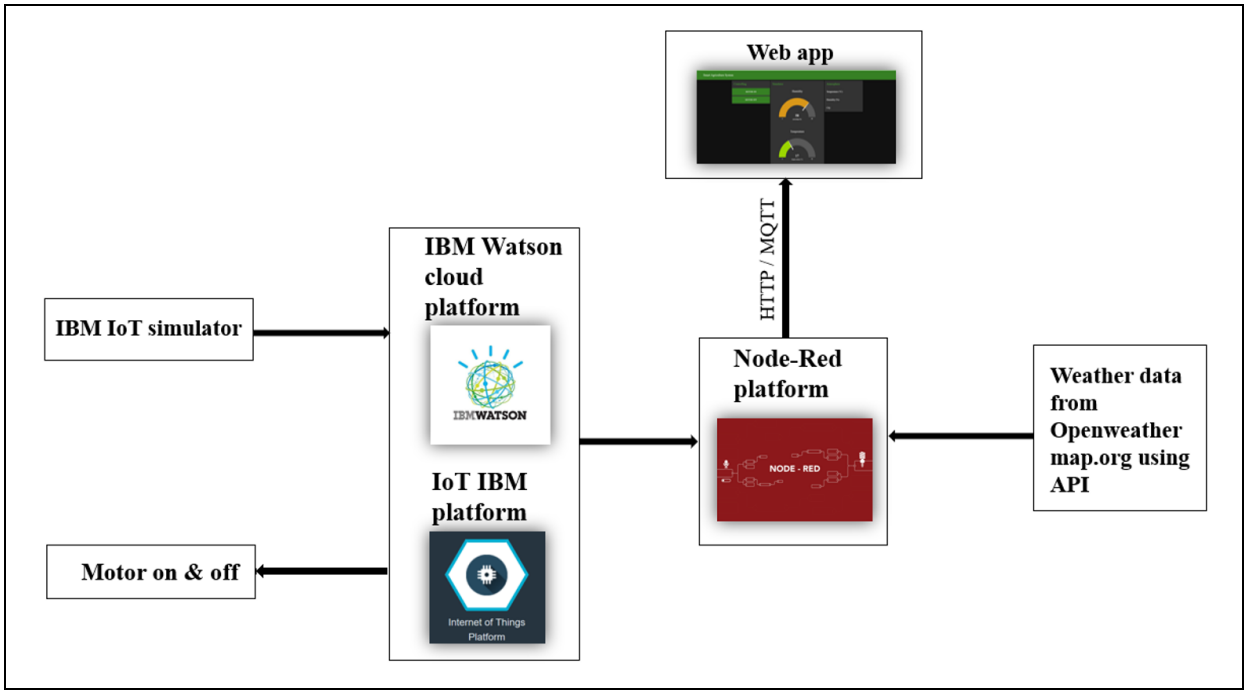
2.2 Proposed Solution

The proposed solution for this problem is the Smart Agriculture System based on IoT. It can monitor soil moisture and climatic conditions to grow and yield a good crop. The farmer can also get the real time weather forecasting data by using external platforms like Open Weather API. Farmer is provided a mobile app using which he can monitor the temperature,humidity and soil moisture parameters along with weather forecasting

details. Based on all the parameters he can water his crop by controlling the motors using the mobile application.Even if the farmer is not present near his crop he can water his crop by controlling the motors using the mobile application from anywhere. Here we are using the Online IoT simulator for getting the Temperature and Humidity values.

3. THEORITICAL ANALYSIS

3.1 Block diagram



3.2 Software Designing

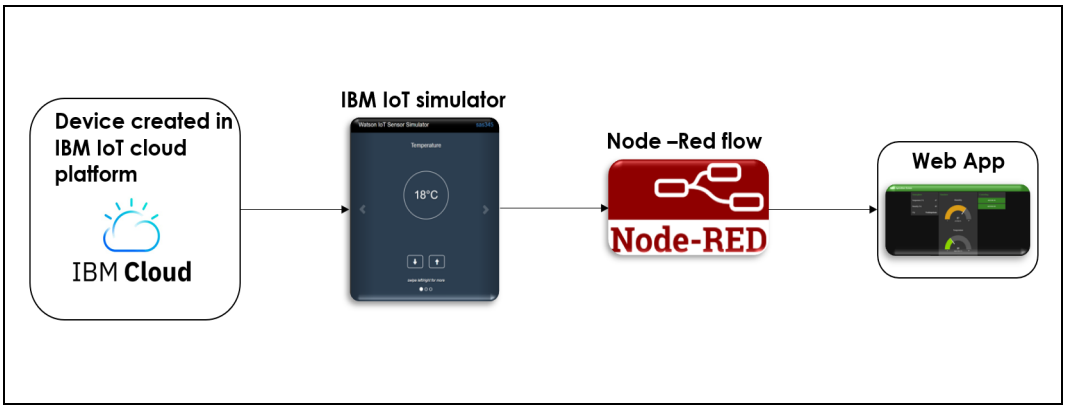
- Flow of data from IBM IoT simulator to Web app
- Flow of data from Openweathermap.org to Web app

3.2.1 Flow of data from IBM IoT simulator to Web app

The flow of data from IBM IoT simulator to web app is

- Device is created in the IBM IoT platform.
- After creating we will get the device credentials, save them in notepad for further purpose.
- Open IBM IoT simulator and give the device credentials.

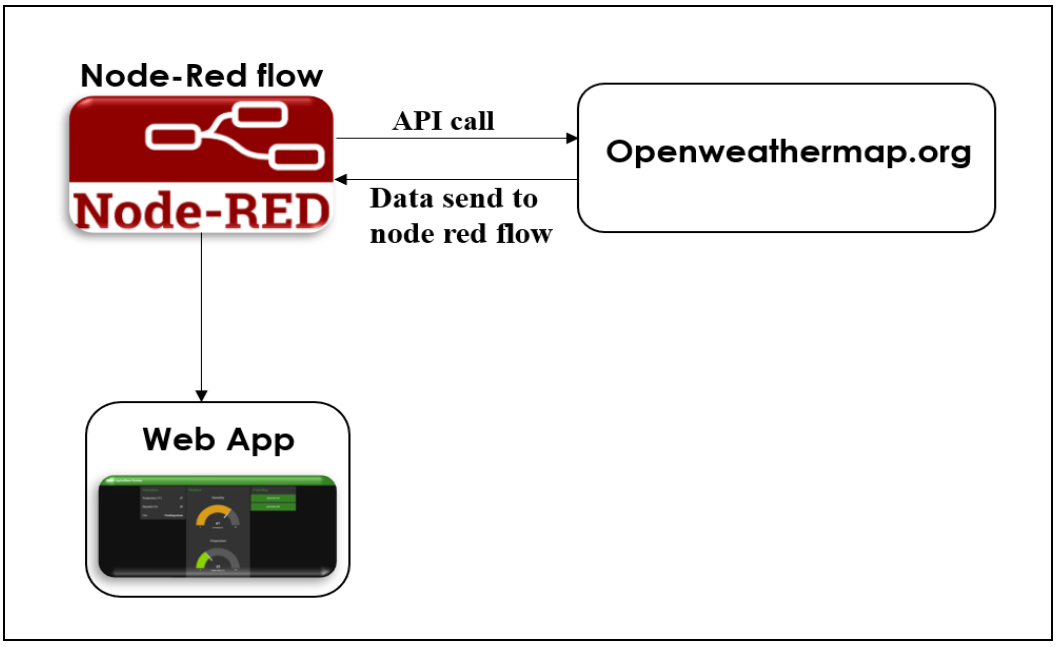
- Now our device is connected to the IBM IoT simulator.
- We create a flow to get the data from simulator to the node red and it is displayed in Web app.



3.2.2 Flow of data from Openweathermap.org to Web app

The flow of data from Openweathermap.org to Web app

- Place the API call that obtained from the Openweathermap.org in the URL of the http request node.
- Now inject the timestamp and we get the data from the Openweathermap.org through an API call that we placed in the http request node.
- These values are sent to the web app and are displayed there.



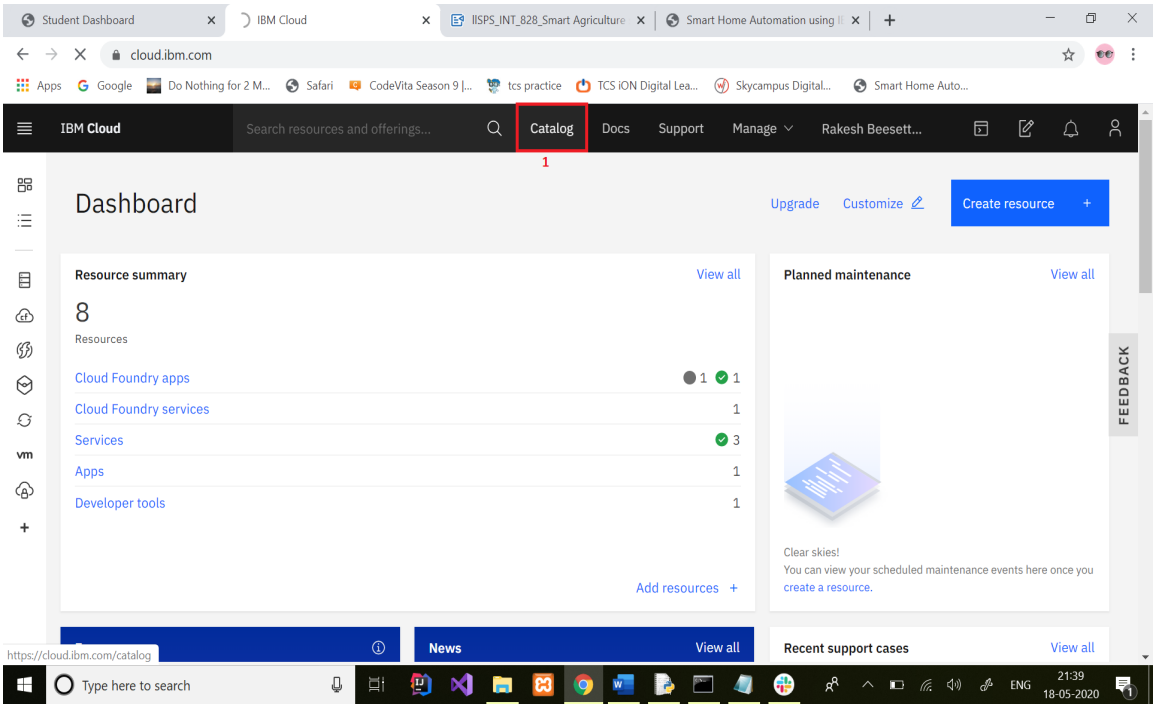
4. EXPERIMENTAL INVESTIGATIONS

- Creating IBM Watson IoT platform
- Creating a device on the IoT platform
- API creation for Node-red application
- Configuring the connection Security for the device created

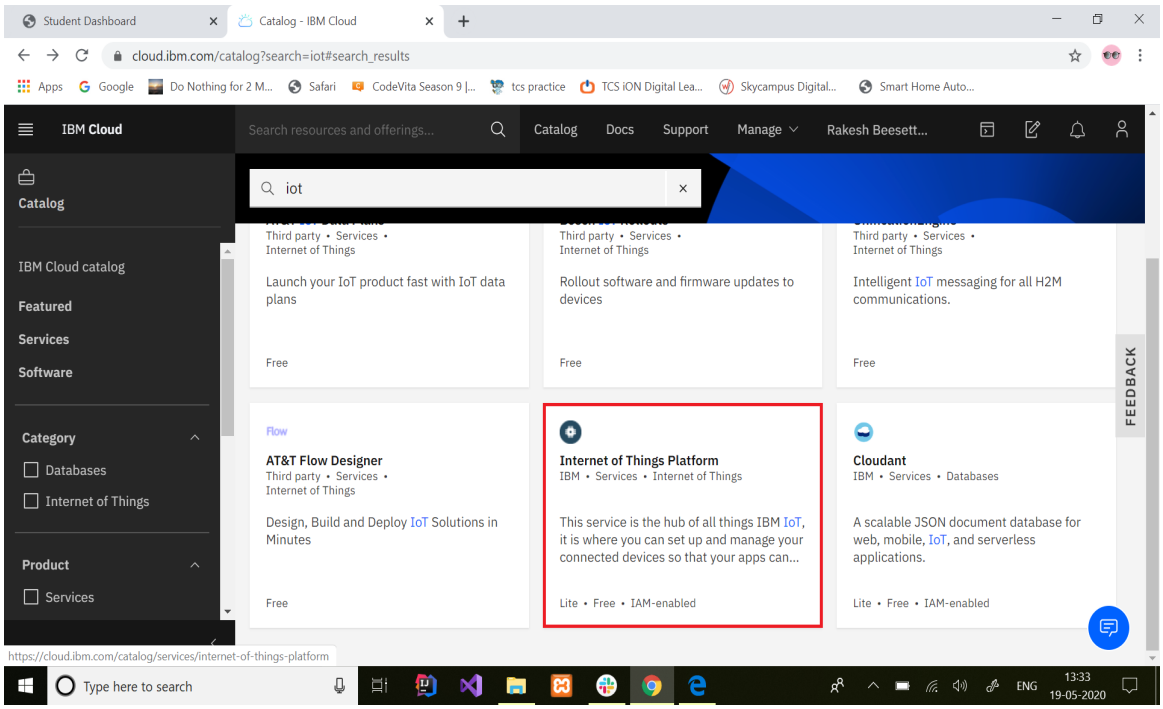
4.1 Creating IBM Watson IoT platform

To create a IBM Watson IoT platform, first we need to have an account in IBM cloud. After logging in successfully you will be redirected to the dashboard page. If you are a new user you will not have any resources listed. For a returning user, you will have your previous resources listed.

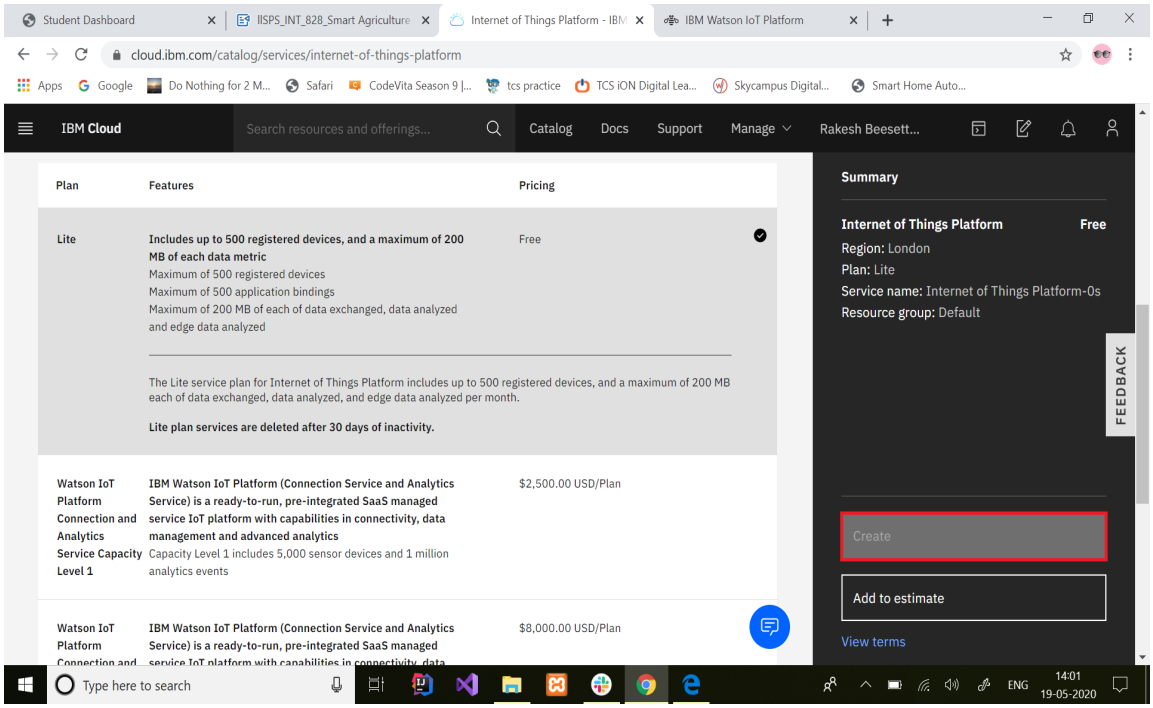
1. Click on the catalog on the dashboard.



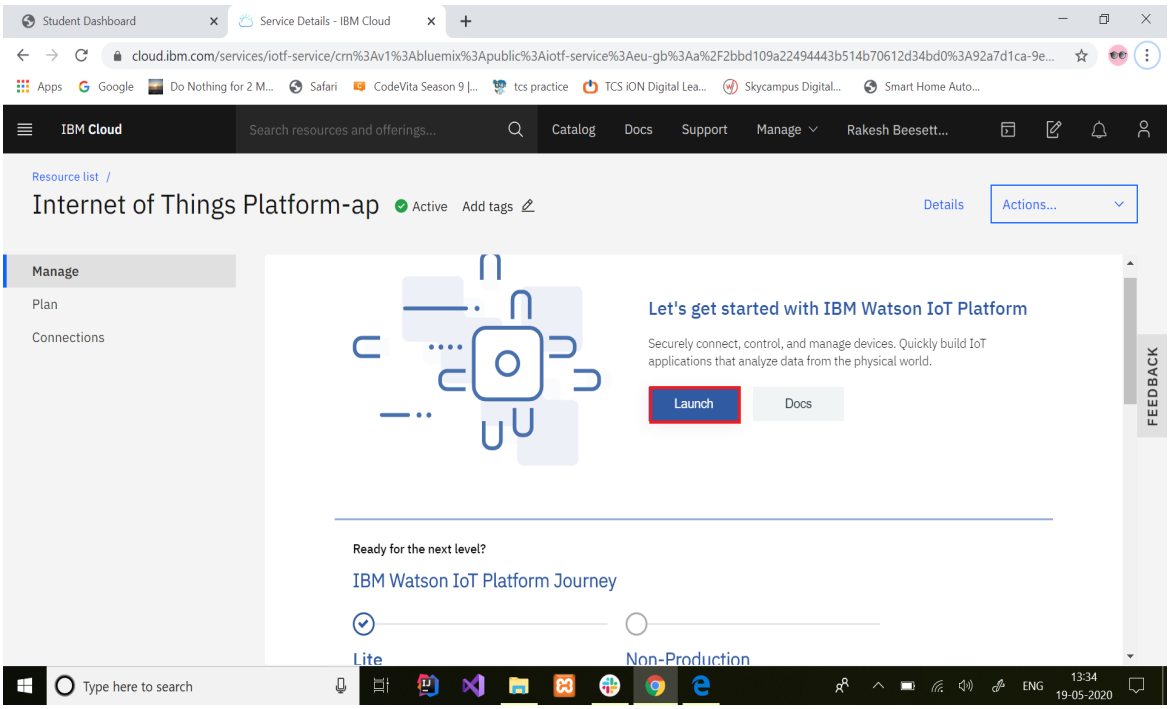
2. Search for the Internet of Things in the search bar provided.
3. Click on the Internet of Things Platform to launch the IoT platform.



4. Click on to create an IoT platform for your project. Leave the default region provided.



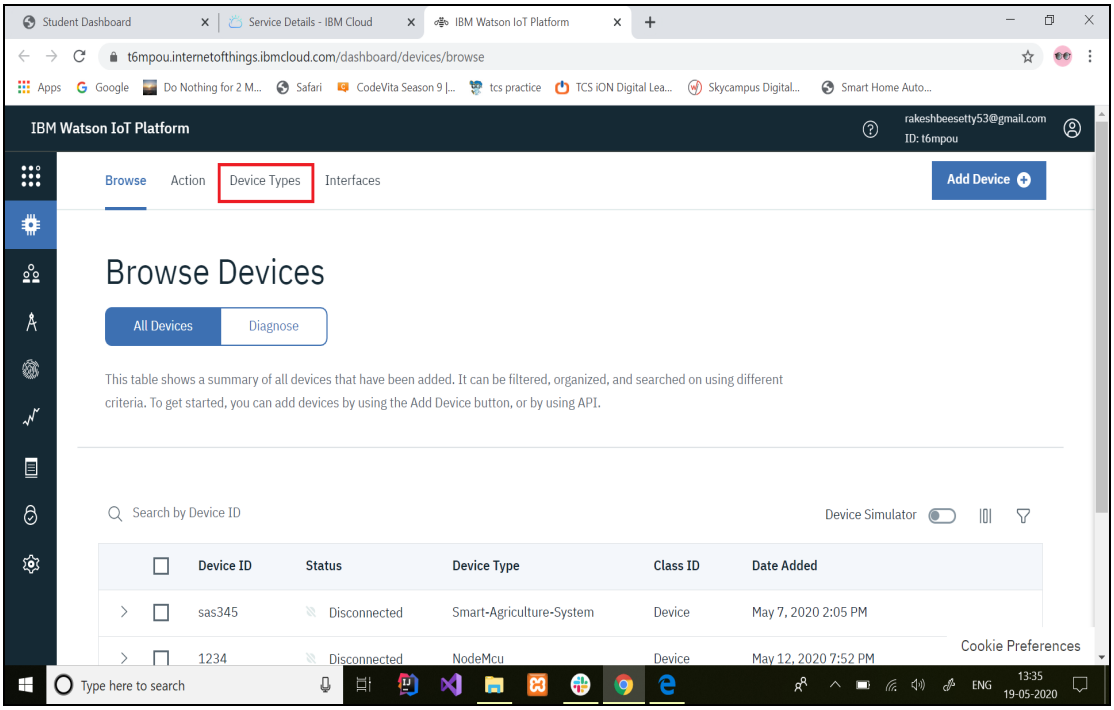
5. After creating the IoT platform you are redirected to the IBM WatsonIoT platform launch page. Click on the launch button to launch your IoT platform.



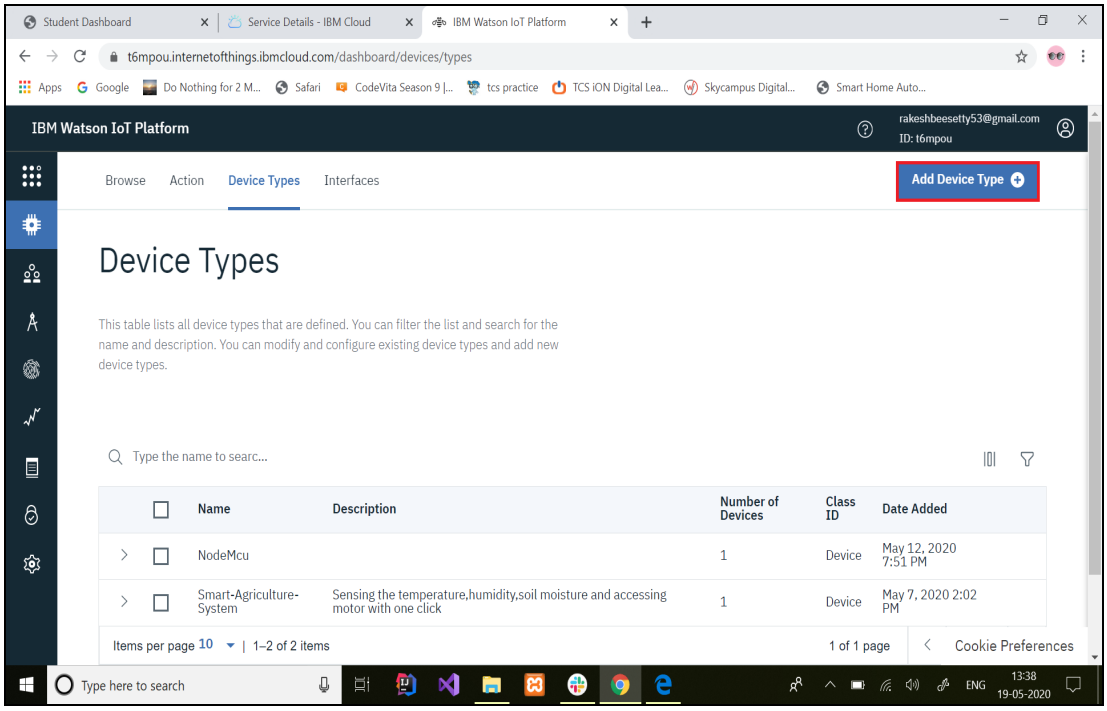
4.2 Creating a device on the IoT platform

After launching the IBM IoT platform you will be redirected to the IoT platform where you must create a device. Before creating a device we must create device type.

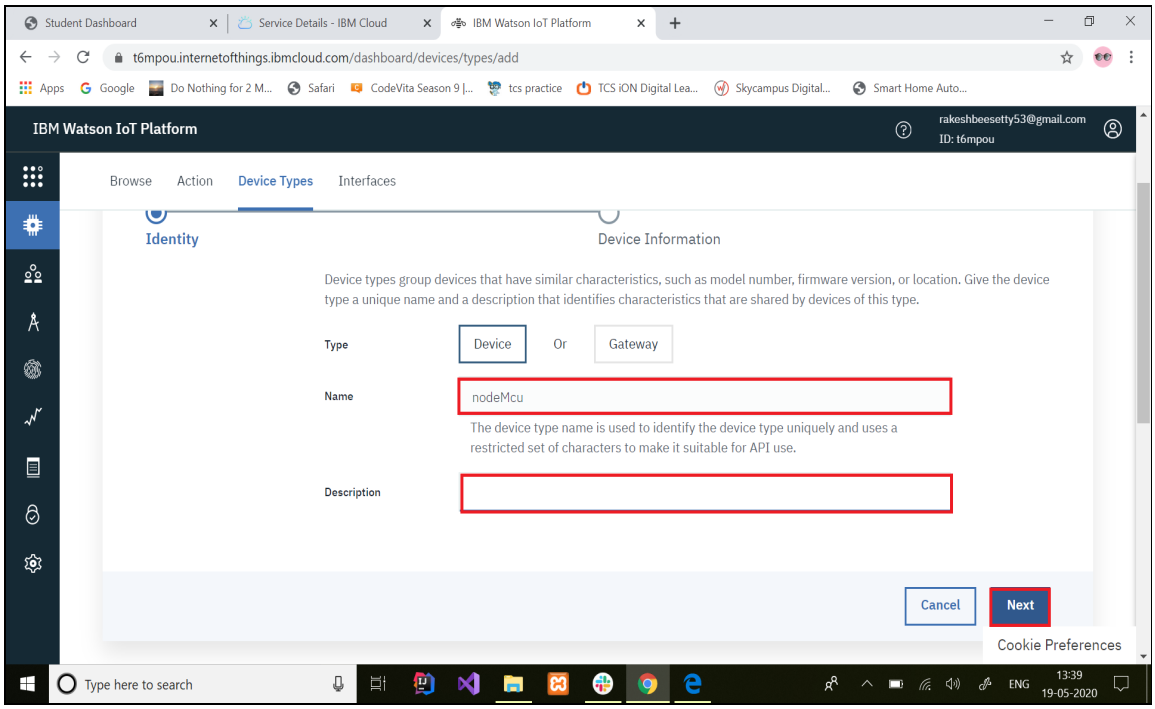
1. Click on the device type section to create a device type.



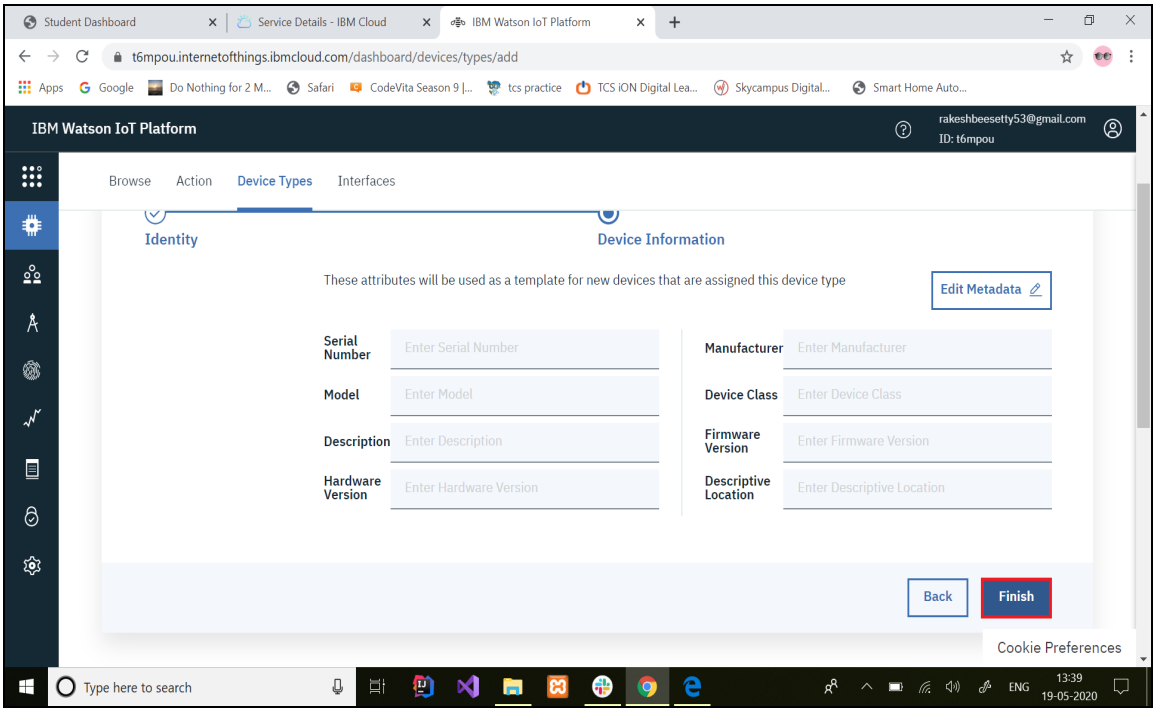
2. Click on Add device type to create a device type.



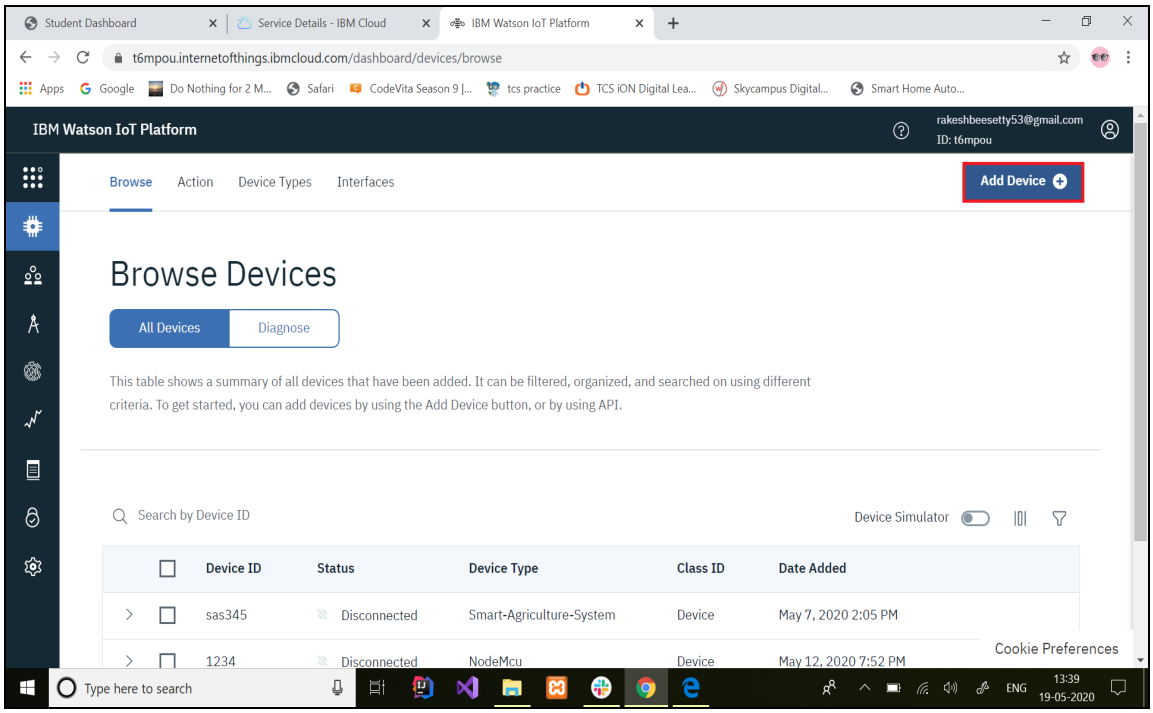
3. Select device and give some related device type name and description is optional and click next.



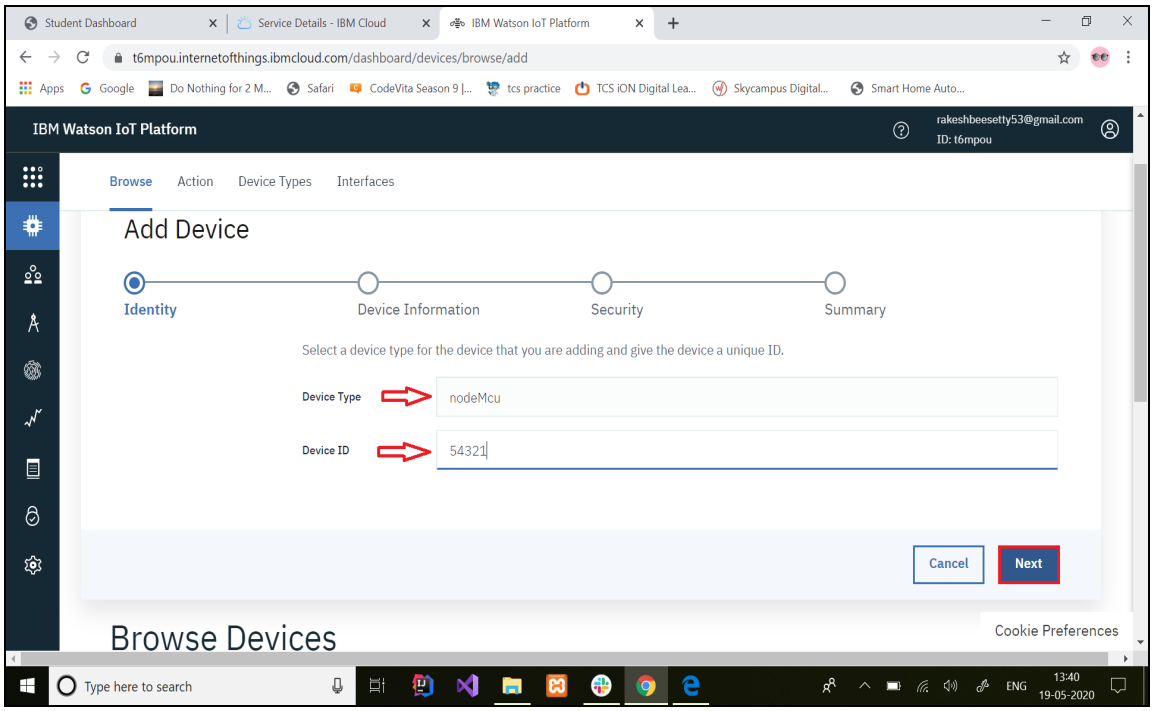
4. Device information is optional for now as we are not adding a hardware device leave everything as it is and click on finish.



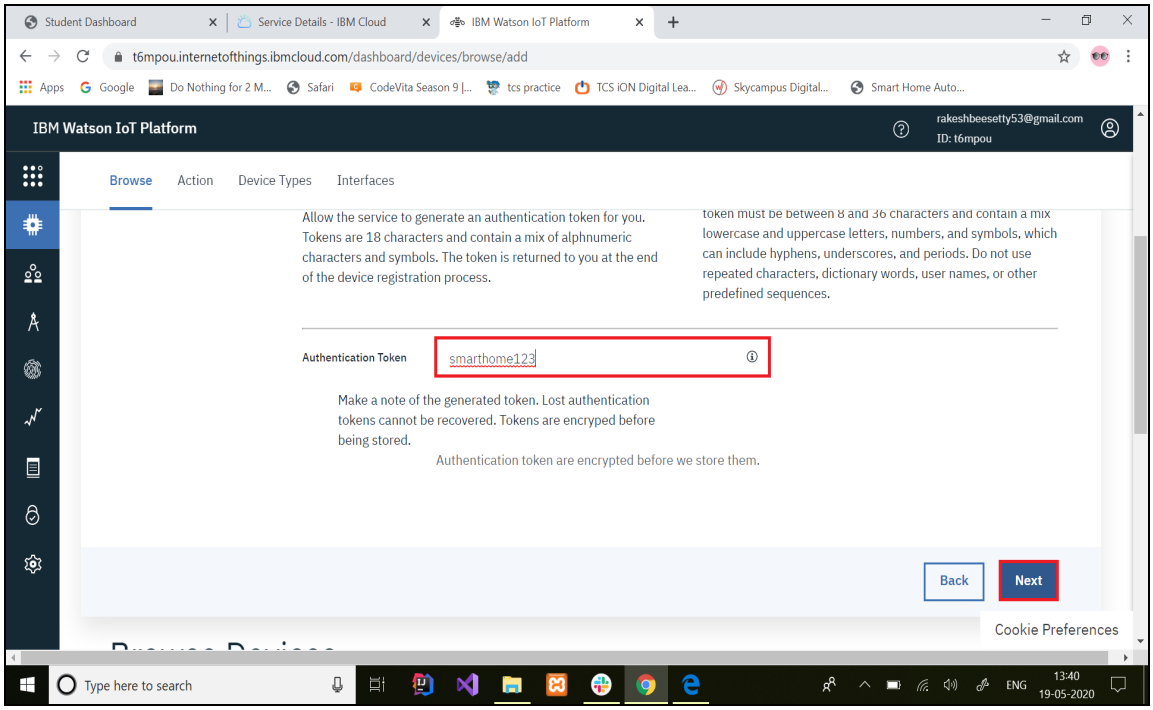
5. After device type is created go to browse section to add a device and click on Add Device.



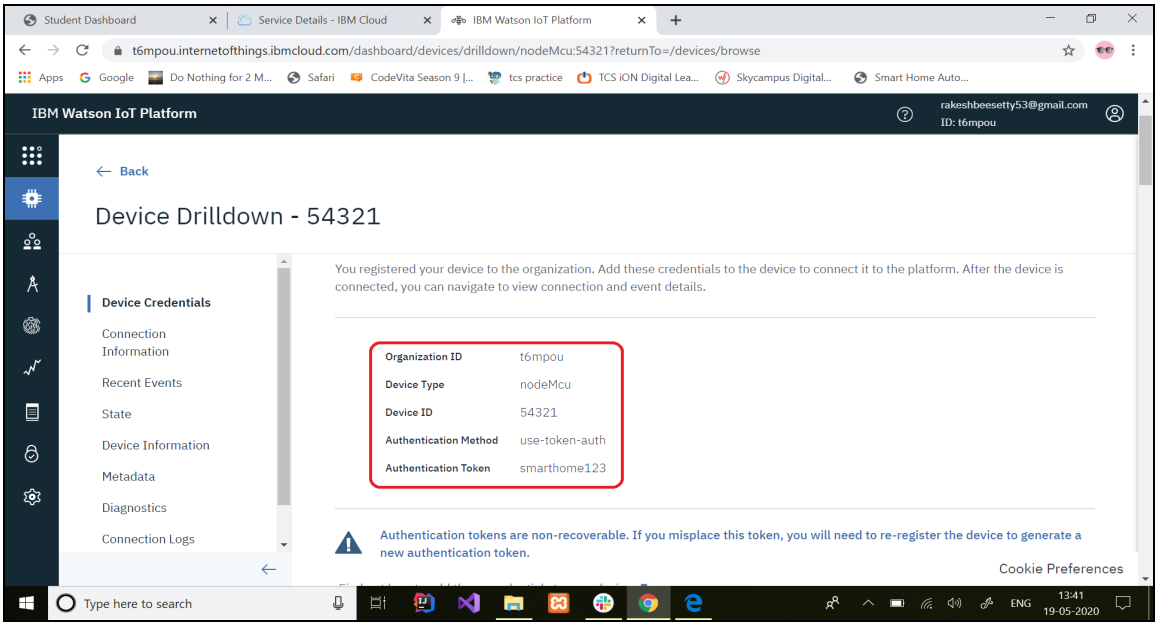
6. Now give device type which you created before and give some ID to the device to identify the device uniquely and click on next.



7. Device information is optional, you can leave them empty and click on next. Now give Authentication token which will be like a password for your device and click next.



8. Now the summary of your device information will be displayed, check them once click on finish. Copy all the credential details on the notepad and save it for the further purpose.

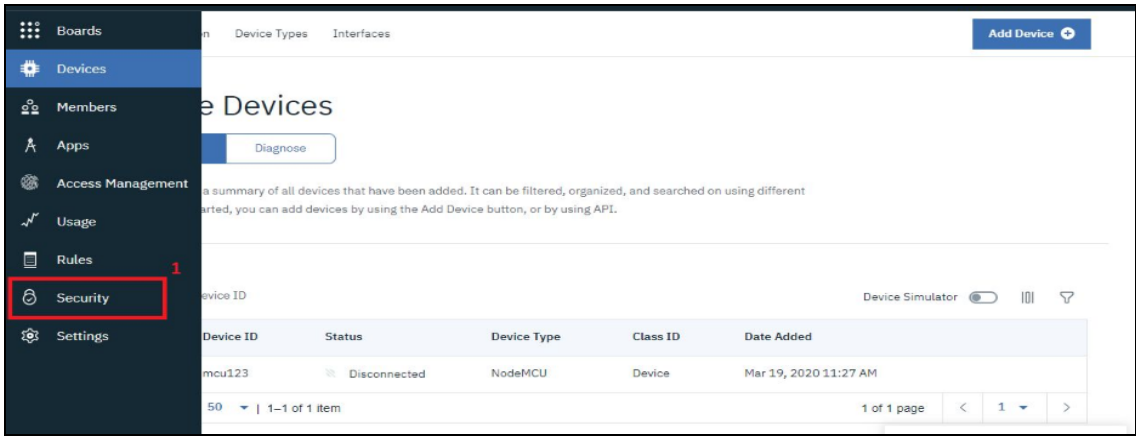


4.3 API creation for Node-red application

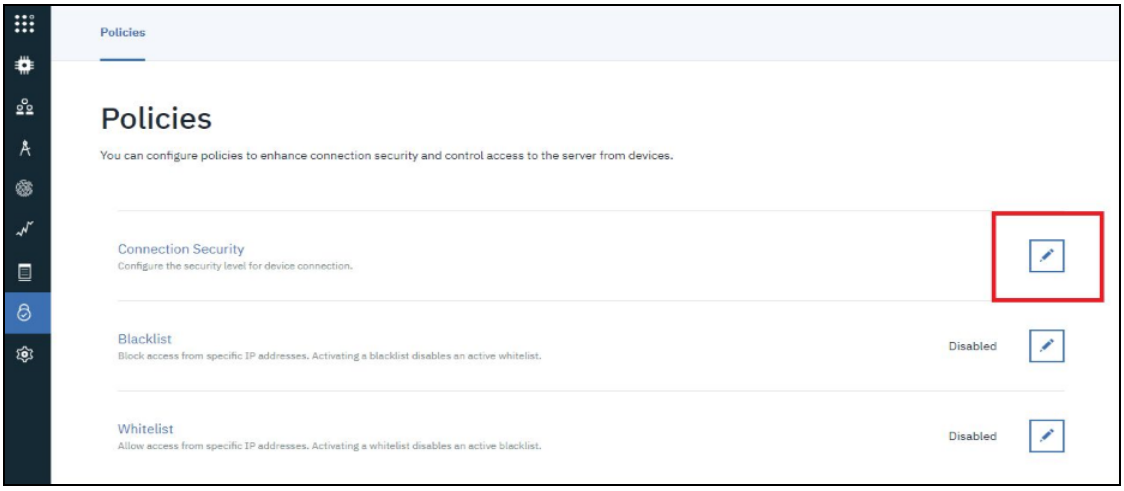
1. Hover your mouse towards the device icon there you will be displayed with different sections in the IoT platform. Click on the Apps for creating an API.
2. After clicking on the Apps you will be redirected to the Browse API keys page. For a new user or returning user click on the Generate API key button.
3. In the information tab, provide the description of the API key you are generating for. The description may be about the project you are developing. Though it is not mandatory practice to identify the API key we have generated out of multiple.
4. Do not turn on the API expires unless you are aware of it. This will lead to expire your API key generated after the time mentioned.
5. Click on next for selecting the role of your API key.
6. In the roles tab, select the role as a standard application. Click on the generate key button to complete the API key generation.
7. Copy the generated API credentials into the text file you have created previously or the device credentials. Click on close.

4.4 Configuring the connection Security for the device created

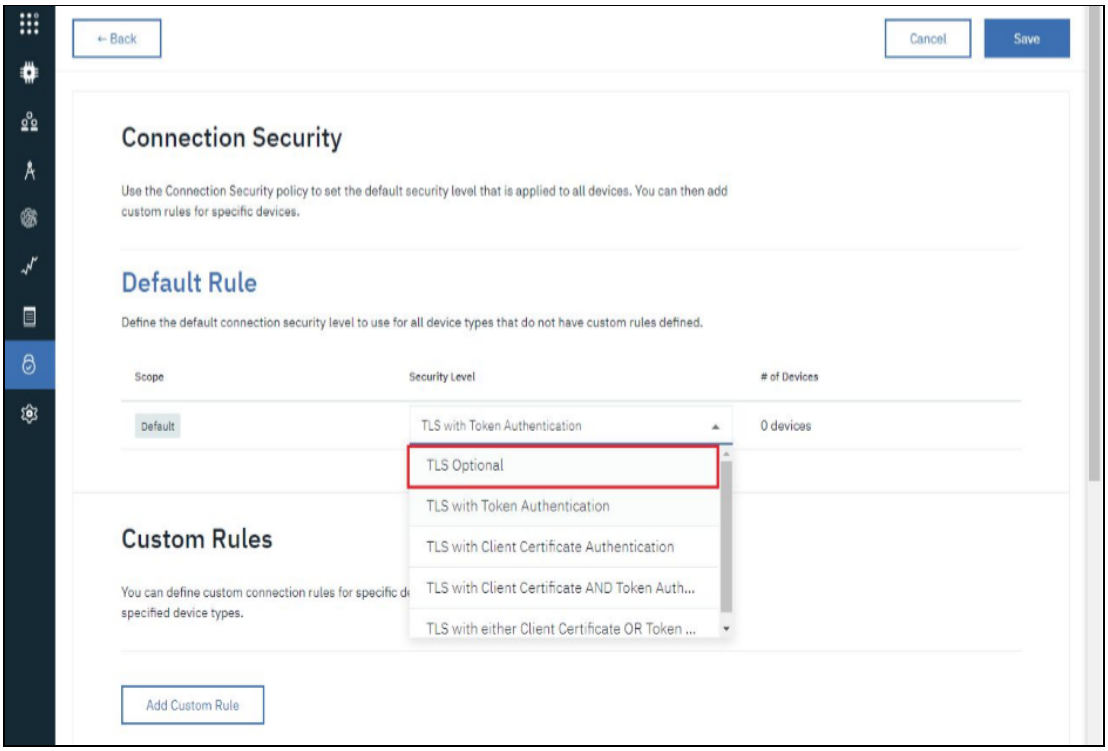
1. Hover your mouse towards the navigation pane and click on Security for configuring the connection security.



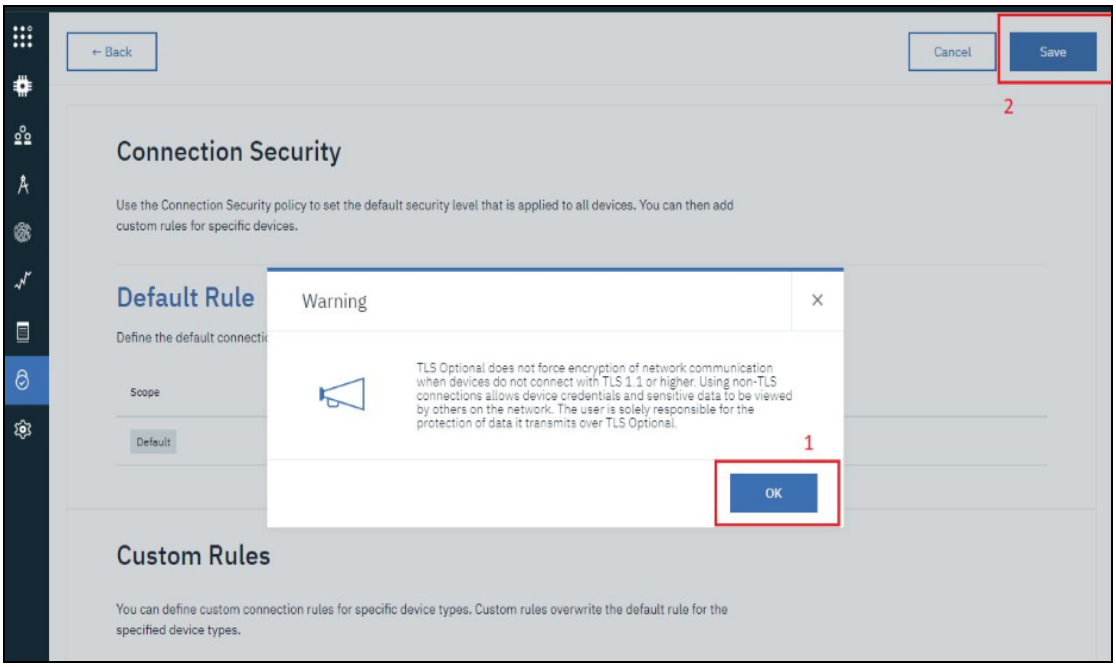
2. In the Security tab, click on the edit icon for connection security.



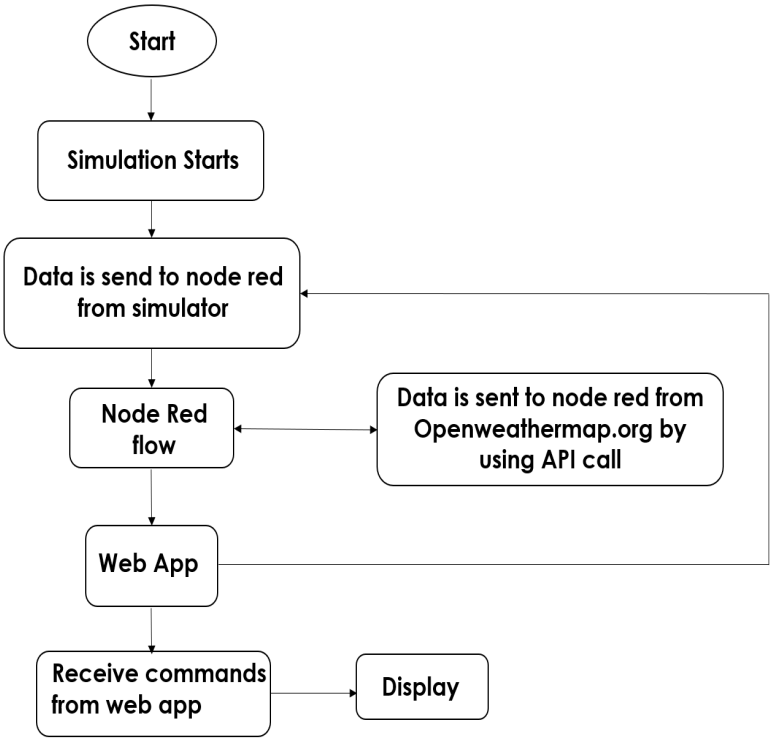
3. In the connection security level tab, select the default rule as TLS Optional.



4. After selecting the TLS optional rule you are prompted with a warning. Click on OK and then click on save for saving the rule created. If you do not save it, you have you to repeat the procedure of connection security again.

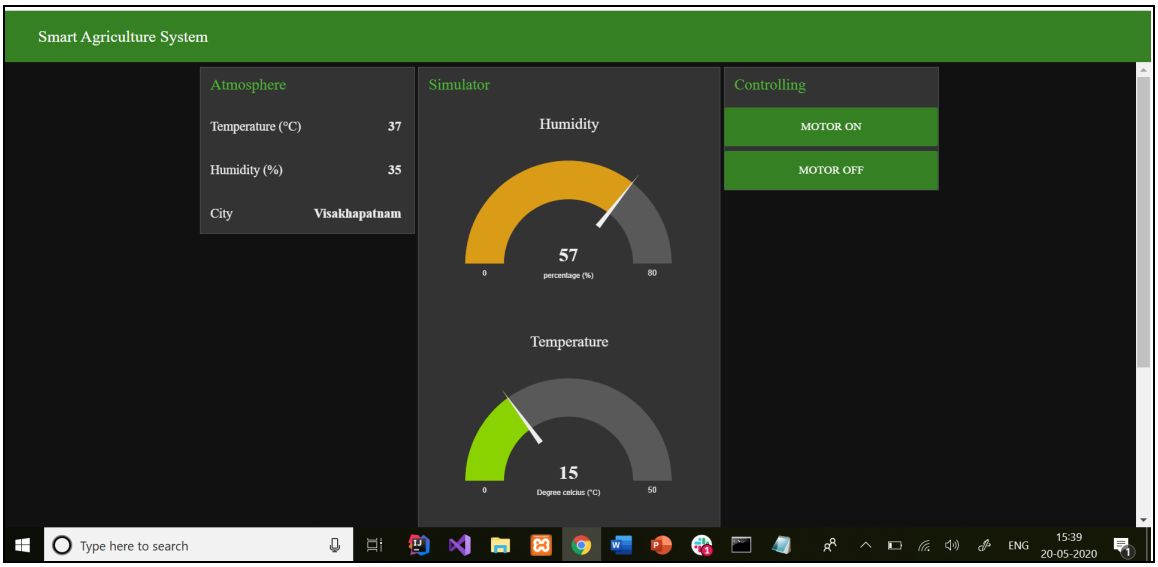


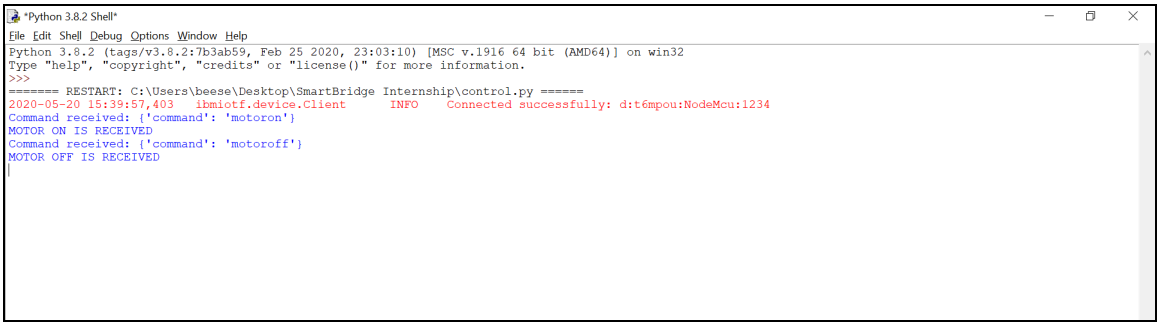
5. FLOWCHART



6. RESULT

- Through this Smart Agriculture System, we can get the real time weather forecasting data by using external platforms like Open Weather API.
- Users are provided with a mobile app using which they can monitor the temperature, humidity and soil moisture parameters along with weather forecasting details.
- Based on all the parameters they can water his crop by controlling the motors using the mobile application.
- Even if the user is not present near his crop he can water his crop by controlling the motors using the mobile application from anywhere.





7. ADVANTAGES & DISADVANTAGES

7.1 Advantages of Smart Agriculture System

- The farmer can get the real time weather forecasting data.
- He can sow the seeds according to the weather conditions.
- Based on the soil moisture the farmer can water his crop.
- Based on all the parameters he can water his crop by controlling the motors using the mobile application.
- The farmer can water his land by controlling the motors using the mobile application from anywhere.

- Time is reduced because the farmer can control the motor from his house only.

7.2 Disadvantages of Smart Agriculture System

- The location cannot be updated dynamically.
- The language may not be understandable to the farmers.
- If the farmer forgot to switch off the motor after switching on, there might be a chance that the crop will get damaged by access of watering.
- If there are faulty data processing equipment or sensors, then it will lead to the situation where the wrong decisions are taken.
- Since these pieces of equipment are already costly, repairing it or replacing it will again cost numerous money.
- To control all these equipment, the farmer needs an internet connection in his local area which is a bit costly when compared to his earnings.

8. APPLICATIONS

The global population is predicted to touch 9.6 billion by 2050 – this poses a big problem for the agriculture industry. Despite combating challenges like extreme weather conditions, rising climate change, and farming's environmental impact, the demand for more food has to be met. To meet these increasing needs, agriculture has to turn to new technology. New smart farming applications based on IoT technologies will enable the agriculture industry to reduce waste and enhance productivity from optimizing fertilizer use to increase the efficiency of farm vehicles' routes.

Table 1.1 Various applications are integrated in Agricultural fields leading to efficient management and controlling of various activities.

Application Name	Description
Precision Farming	High accuracy is required is required in terms of weather information which reduces the chances of crop damage. Agriculture IoT ensures timely delivery of real time data in terms of weather forecasting, quality of soil, cost of labor and much more to farmers.
Agricultural Drones	PrecisionHawk is an organization that uses drones for gathering valuable data via a series of sensors that are used for imaging, mapping, and surveying of agricultural land. These drones perform in-flight monitoring and observations. The farmers enter the details of what field to survey and select an altitude or ground resolution.

Application Name	Description
Crop Water Management	In order to perform agriculture activities in inefficient manner, adequate water is essential. Agriculture IoT is integrated with Web Map Service (WMS) and Sensor Observation Service (SOS) to ensure proper water management for irrigation and in turn reduces water wastage.
Smart Greenhouses	Greenhouse farming is a technique that enhances the yield of crops, vegetables, fruits etc. Greenhouses control environmental parameters in two ways; either through manual intervention or a proportional control mechanism.Different sensors that measure the environmental parameters according to the plant requirement are used for controlling the environment in a smart greenhouse. Then, a cloud server created for remotely accessing the system when it connects using IoT.

9. CONCLUSION

The IoT based smart agriculture System that is proposed via this report will assist farmers in increasing the agriculture yield and take efficient care of food production as the System will always provide helping hand to farmers for getting accurate live feed of environmental temperature and humidity with more than 99 percent accurate results. Also, with this smart agriculture system, the farmers can easily control their motors which are there in their lands without any problem.

10. FUTURE SCOPE

The future work would be focused more on integrating the dynamic location by using a GPS module to get the temperature and humidity values of any location that the farmer wants to know and also sets a timer to the motor so that the motor can turn off after the specific time the farmer had mentioned in the interface and can also set a specific time when the motor should On automatically.

11. APPENDIX

- https://www.ibm.com/support/knowledgecenter/SSQP8H/iot/platform/ts_index.html

Source Code:

```
import time
import sys
import ibmiotf.application  # to install pip install ibmiotf
import ibmiotf.device

# Provide your IBM Watson Device Credentials
organization = "t6mpou"  # replace the ORG ID
deviceType = "NodeMcu"  # replace the Device type wi
deviceId = "1234"  # replace Device ID
authMethod = "token"
authToken = "smartinternz123"  # Replace the authtoken

def myCommandCallback(cmd):  # function for Callback
    print("Command received: %s" % cmd.data)
    if cmd.data['command'] == 'motoron':
        print("MOTOR ON IS RECEIVED")

    elif cmd.data['command'] == 'motoroff':
        print("MOTOR OFF IS RECEIVED")

    if cmd.command == "setInterval":

        if 'interval' not in cmd.data:
            print("Error - command is missing required
```

```
  information: 'interval'")
25          else:
26              interval = cmd.data['interval']
27      elif cmd.command == "print":
28          if 'message' not in cmd.data:
29              print("Error - command is missing required
  information: 'message'")
30          else:
31              output = cmd.data['message']
32              print(output)
33
34  try:
35      deviceOptions = {"org": organization, "type":
  deviceType, "id": deviceId, "auth-method": authMethod,
36                       "auth-token": authToken}
37      deviceCli = ibmiotf.device.Client(deviceOptions)
38  # ..........................................
39
40  except Exception as e:
41      print("Caught exception connecting device: %s" %
  str(e))
42      sys.exit()
43
44  # Connect and send a datapoint "hello" with value "world"
  into the cloud as an event of type "greeting" 10 times
45  deviceCli.connect()
46
47  while True:
```

```
48      deviceCli.commandCallback = myCommandCallback
49
50  # Disconnect the device and application from the cloud
51  deviceCli.disconnect()
```