Smart Farmer-IOT Enabled Smart Farming Application

Project Development Phase

Sprint Delivery -1

Date	15 November 2022
Team ID	PNT2022TMID09801
Project Name	Smart Farmer-IOT Enabled Smart Farming Application
Leader Name	Bitu kumar
Team Members Name	Rahul patel Sanny kumar Ashish Anand

1. Introduction:

The main aim of this project is to help farmers automate their farms by providing them with a Web App through which they can monitor the parameters of the field like Temperature,soil moisture, humidity etc and control the equipment like water motor and other devices remotely via the internet without their actual presence in the field.

2. Problem Statement:

Farmers need to deal with many problems like coping with climate change, soil erosion and Biodiversity loss. Farmers are to be present at farm for its maintenance irrespective of the weather conditions.They have to ensure that the crops are well watered and the farm status is monitored by them physically.Farmers have to stay most of the time in field in order to get a good yield.In difficult times like in the presence of pandemic also they have to work hard in their fields risking their lives to provide food for the country.

3. Proposed Solution:

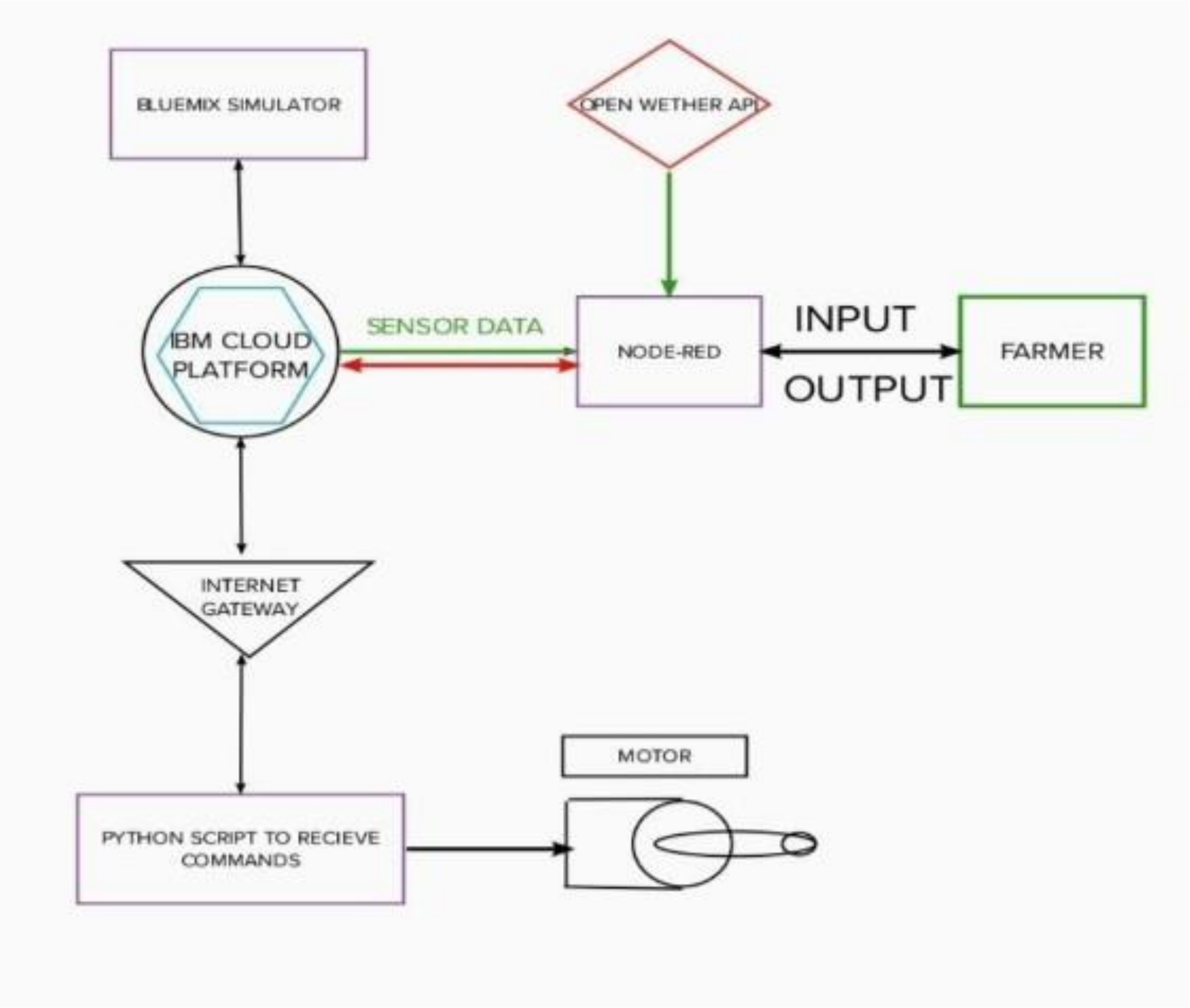
To provide an efficient decision support system using wireless sensor networks which handle different activities of the farm and give useful information related to soil moisture, Temperature and

Humidity content. Due to the weather condition, water level increases, Farmers get a lot of distractions which is not good for Agriculture.

4. Theoretical Analysis:

4.1 Block Diagram:

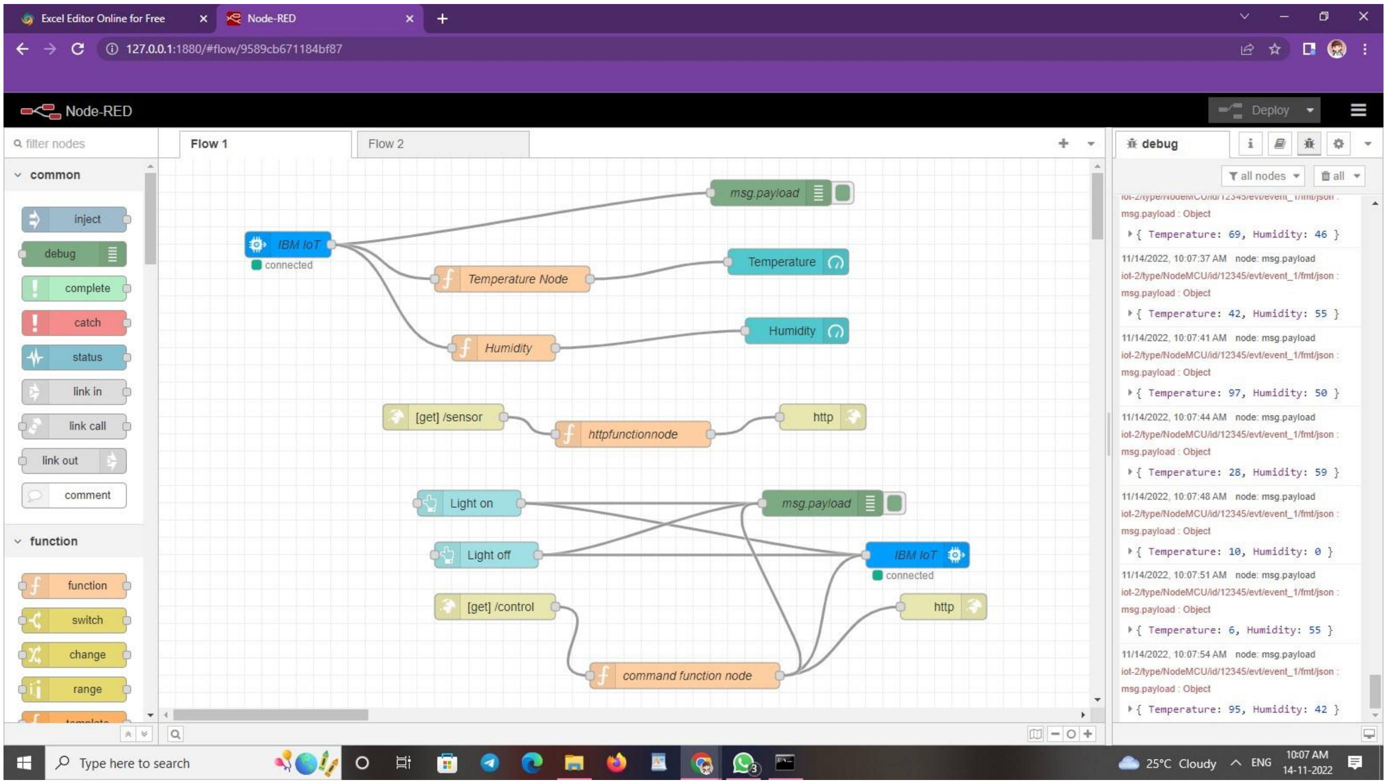
In order to implement the solution , the following approach as shown in the block diagram,is used


BLUEMIX SIMULATOR
OPEN WETHER API
IBM CLOUD PLATFORM
SENSOR DATA
NODE-RED
INPUT
OUTPUT
FARMER
INTERNET GATEWAY
MOTOR
PYTHON SCRIPT TO RECIEVE COMMANDS

4.2 Required Software Installation:

4.2.A Node-Red:

Node-RED is a flow-based development tool for visual programming developed originally by IBM for wiring together hardware devices,APIs and online services as part of the Internet of Things.Node-RED provides a web browser-based flow editor, which can be used to create JavaScript functions.



```
iot-2/type/NodeMCU/id/12345/evt/event_1/fmt/json :
msg.payload : Object
▸{ Temperature: 69, Humidity: 46 }
11/14/2022, 10:07:37 AM   node: msg.payload
iot-2/type/NodeMCU/id/12345/evt/event_1/fmt/json :
msg.payload : Object
▸{ Temperature: 42, Humidity: 55 }
11/14/2022, 10:07:41 AM   node: msg.payload
iot-2/type/NodeMCU/id/12345/evt/event_1/fmt/json :
msg.payload : Object
▸{ Temperature: 97, Humidity: 50 }
11/14/2022, 10:07:44 AM   node: msg.payload
iot-2/type/NodeMCU/id/12345/evt/event_1/fmt/json :
msg.payload : Object
▸{ Temperature: 28, Humidity: 59 }
11/14/2022, 10:07:48 AM   node: msg.payload
iot-2/type/NodeMCU/id/12345/evt/event_1/fmt/json :
msg.payload : Object
▸{ Temperature: 10, Humidity: 0 }
11/14/2022, 10:07:51 AM   node: msg.payload
iot-2/type/NodeMCU/id/12345/evt/event_1/fmt/json :
msg.payload : Object
▸{ Temperature: 6, Humidity: 55 }
11/14/2022, 10:07:54 AM   node: msg.payload
iot-2/type/NodeMCU/id/12345/evt/event_1/fmt/json :
msg.payload : Object
▸{ Temperature: 95, Humidity: 42 }
```

Installation :

- First install npm/node.js.
- Open cmd prompt.
- Type => npm install node-red.

To run the application :

- Open cmd prompt.
- Type=>Node-RED.
- Then open http://127.0.0.1:1880/ on your browser.

Installation of IBM IoT and Dashboard nodes for Node RED:

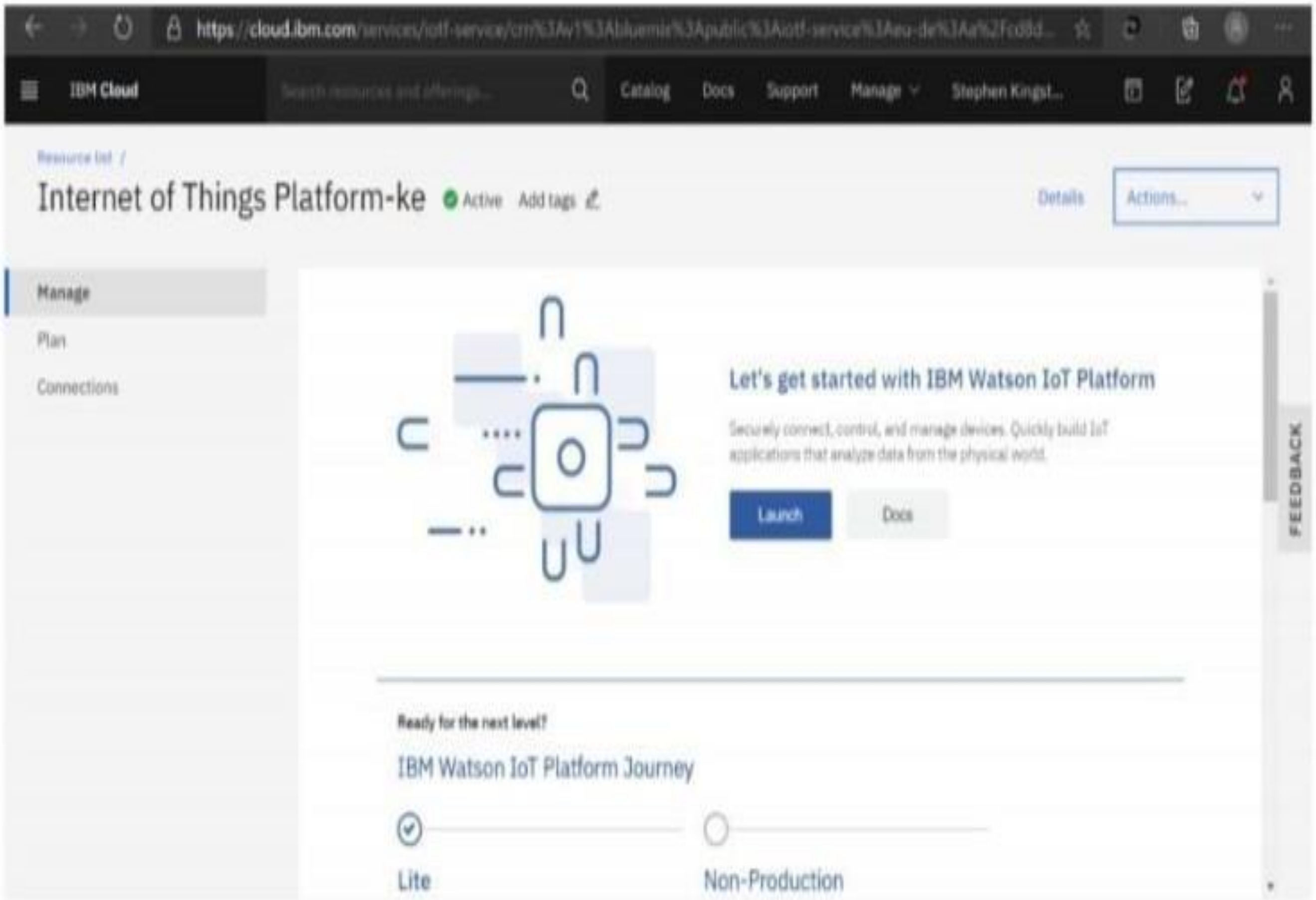
In order to connect to IBM Watson IoT platform and create the WEB UI these nodes are required.

1. IBM IoT node.

2. Dashboard node.

4.2.B IBM Watson IoT Platform:

A fully managed, cloud-hosted service with capabilities for device registration,connectivity,control, rapid visualization and data storage.IBM Watson IoT Platform is a managed, cloud-hosted service designed to make it simple to derive value from your IoT devices.



https://cloud.ibm.com/services/iotf-service/crn%3Av1%3Abluemix%3Apublic%3Aiotf-service%3Aeu-de%3Aa%2Fcd8d...

Internet of Things Platform-ke

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

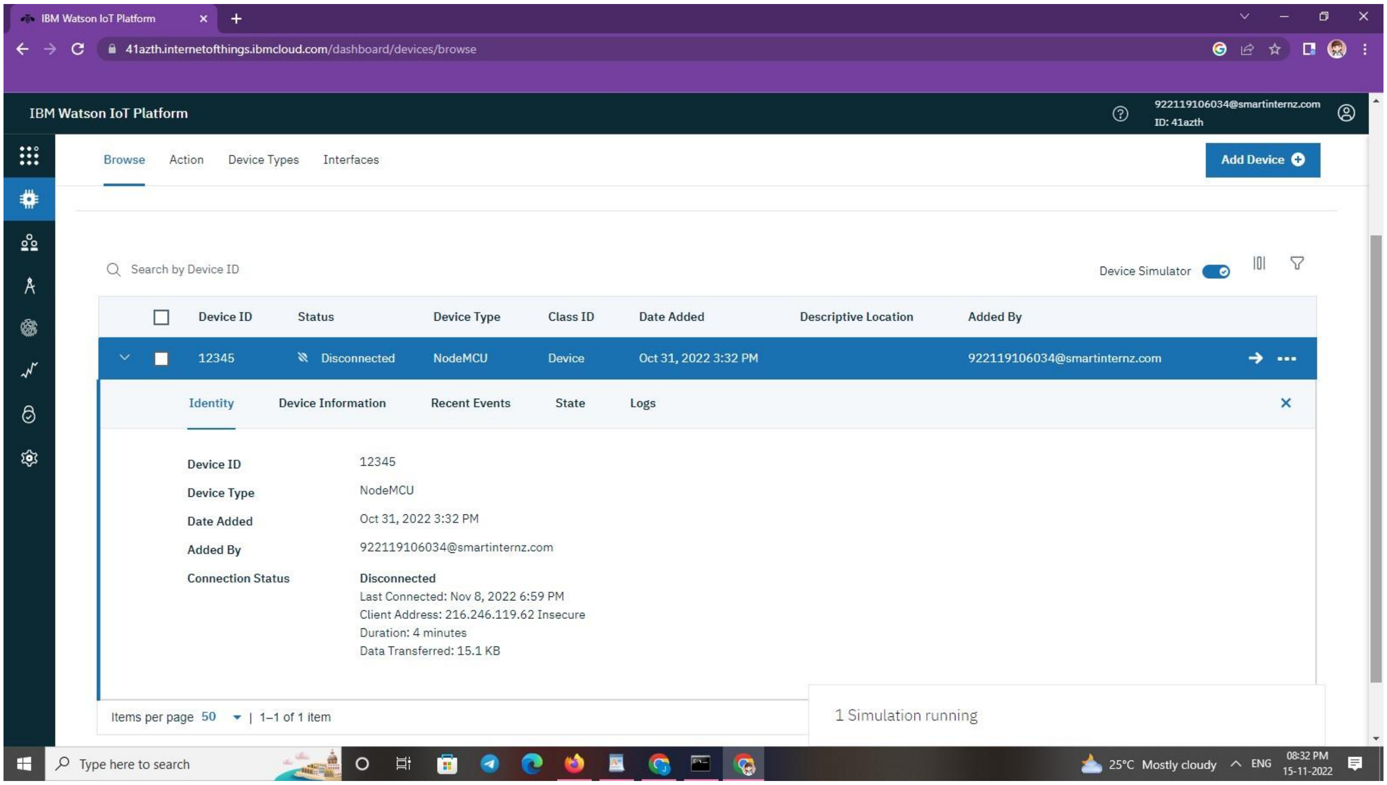
Steps to configure:

- Create an account in IBM cloud using your email ID
- Create IBM Watson Platform in services in your IBM cloudAccount.
- Launch the IBM Watson IoT Platform.

- Create a new device

.

- Give credentials like device type, device ID, Auth. Token
- Create API key and store API key and token elsewhere.



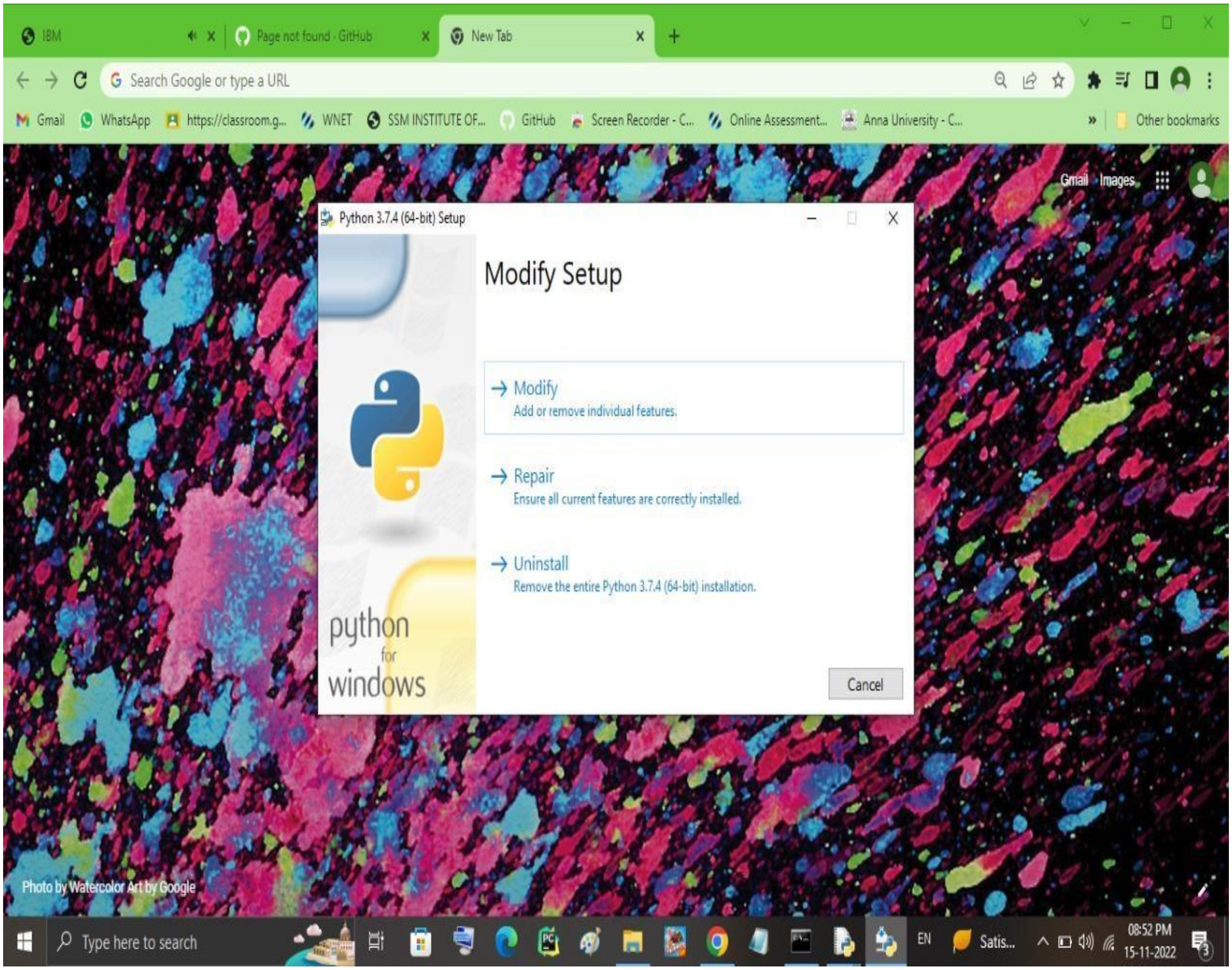
Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By
12345	Disconnected	NodeMCU	Device	Oct 31, 2022 3:32 PM		922119106034@smartinternz.com

Device ID	12345
Device Type	NodeMCU
Date Added	Oct 31, 2022 3:32 PM
Added By	922119106034@smartinternz.com
Connection Status	Disconnected Last Connected: Nov 8, 2022 6:59 PM Client Address: 216.246.119.62 Insecure Duration: 4 minutes Data Transferred: 15.1 KB

4.2.C Python IDE:

Install Python3 compiler

Install any python IDE to execute python scripts, in my case I used Command Prompt to execute.



Python 3.7.4 (64-bit) Setup

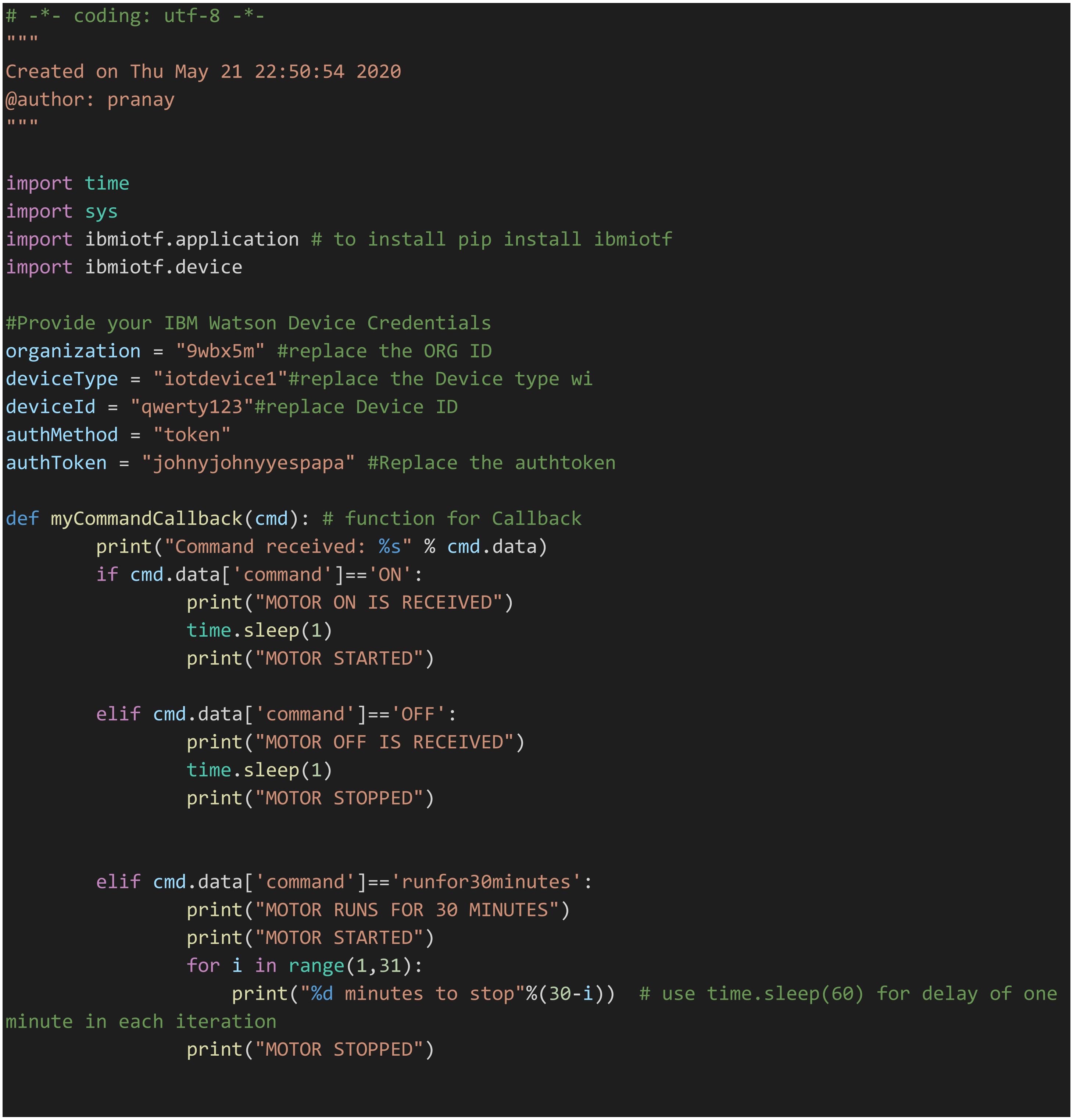
Modify Setup

→ Modify
Add or remove individual features.

→ Repair
Ensure all current features are correctly installed.

→ Uninstall
Remove the entire Python 3.7.4 (64-bit) installation.

Code:



```
# -*- coding: utf-8 -*-
"""
Created on Thu May 21 22:50:54 2020
@author: pranay
"""

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

#Provide your IBM Watson Device Credentials
organization = "9wbx5m" #replace the ORG ID
deviceType = "iotdevice1"#replace the Device type wi
deviceId = "qwerty123"#replace Device ID
authMethod = "token"
authToken = "johnyjohnyyespapa" #Replace the authtoken

def myCommandCallback(cmd): # function for Callback
        print("Command received: %s" % cmd.data)
        if cmd.data['command']=='ON':
                print("MOTOR ON IS RECEIVED")
                time.sleep(1)
                print("MOTOR STARTED")

        elif cmd.data['command']=='OFF':
                print("MOTOR OFF IS RECEIVED")
                time.sleep(1)
                print("MOTOR STOPPED")


        elif cmd.data['command']=='runfor30minutes':
                print("MOTOR RUNS FOR 30 MINUTES")
                print("MOTOR STARTED")
                for i in range(1,31):
                    print("%d minutes to stop"%(30-i))  # use time.sleep(60) for delay of one
minute in each iteration
                print("MOTOR STOPPED")
        if cmd.command == "setInterval":

                if 'interval' not in cmd.data:
                        print("Error - command is missing required information: 'interval'")
                else:
                        interval = cmd.data['interval']
        elif cmd.command == "print":
                if 'message' not in cmd.data:
                        print("Error - command is missing required information: 'message'")
                else:
                        output=cmd.data['message']
                        print(output)

try:
        deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken}
        deviceCli = ibmiotf.device.Client(deviceOptions)
        #..............................................

except Exception as e:
        print("Caught exception connecting device: %s" % str(e))
        sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times
deviceCli.connect()

while True:
        deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

4.3 IoT Simulator:

In our project in the place of sensors we are going to use IoT sensor simulator which give random readings to the connected Cloud.

The link to simulator:

https://41azth.internetofthings.ibmcloud.com/dashboard/devices/br owse

We need to give the credentials of the created device in IBM Watson IoT Platform to connect cloud to simulator.