DELIVERY OF SPRINT – 3

DATE	18- NOVEMBER 2022
TEAM ID	PNT2022TMID48657
PROJECT NAME	SMART WASTE MANAGEENT FOR METROPOLITAN CITIES

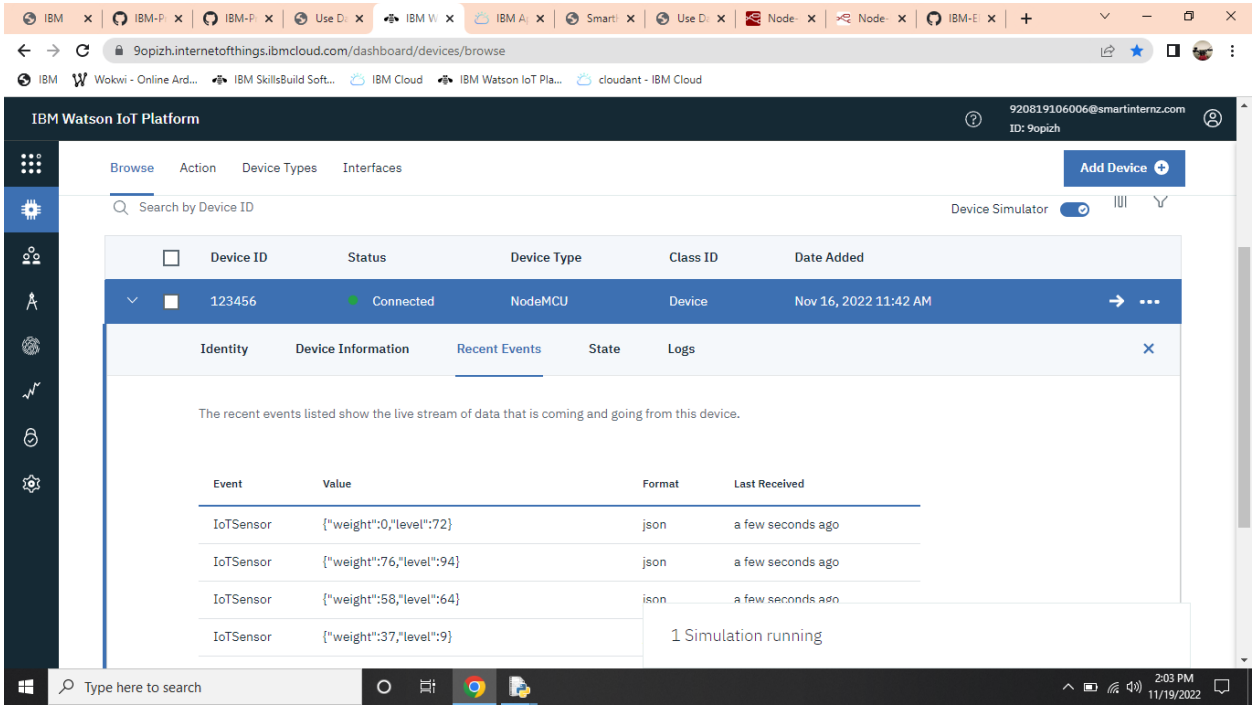
Code :

```
import time

import sys

import ibmiotf.application

import ibmiotf.device

import random



#Provide your IBM Watson Device Credentials

organization = "9opizh"

deviceType = "NodeMCU"

deviceId = "123456"

authMethod = "token"

authToken = "8098439666"


# Initialize GPIO
```

```
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    else :
        print ("led is off")

    #print(cmd)



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:

        #Get Sensor Data from DHT11


        weight=random.randint(0,100)

        level=random.randint(0,100)


        data = { 'weight' : weight, 'level':level  }

        #print data

        def myOnPublishCallback():

            print ("Published Weight = %s Kg" % weight, "level = %s %%" % level, "to IBM Watson")

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)

        if not success:

            print("Not connected to IoTF")

        time.sleep(1)


        deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
```

Creation of device in IBM Watson platform and running simulation



Node Red platform