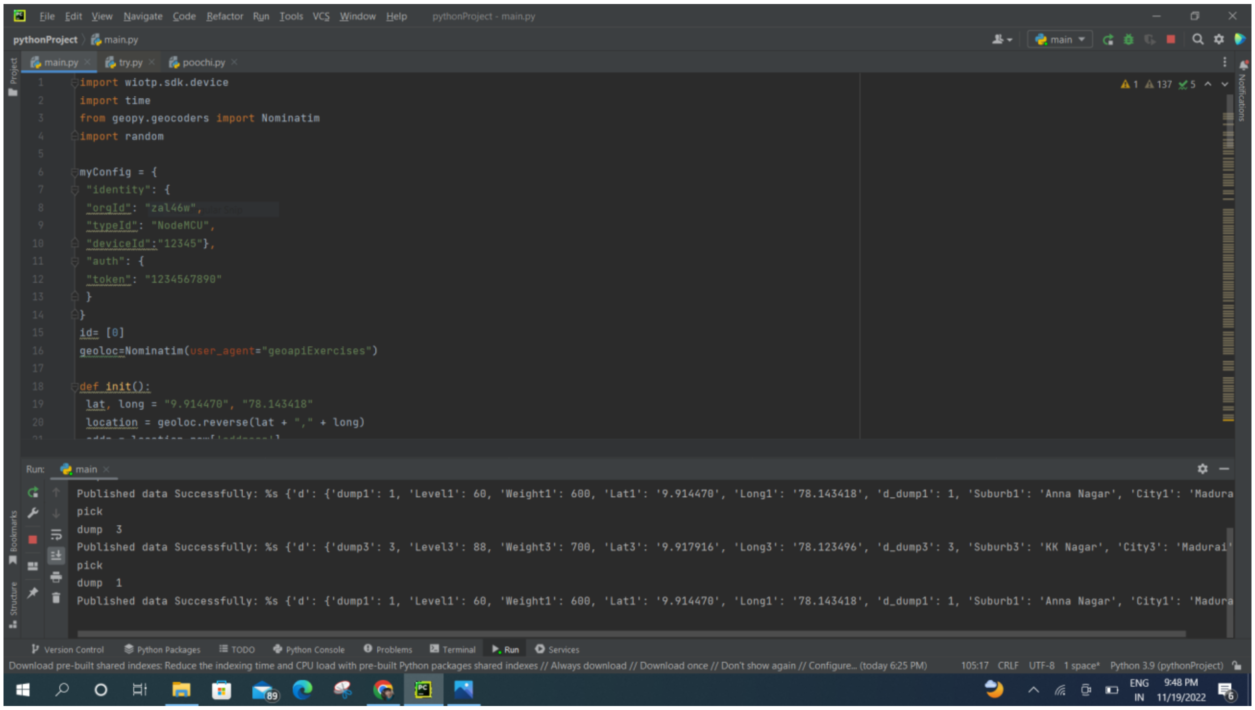
SPRINT-3

Date	19 November 2022
Team ID	PNT2022TMID11549
Project Name	Smart Waste Management System For Metropolitan Cities

*Creating web ui using node-red and publishing sensor values and location to ui.



```
import wiotp.sdk.device
import time
from geopy.geocoders import Nominatim
import random

myConfig = {
 "identity": {
 "orgId": "zal46w",
 "typeId": "NodeMCU",
 "deviceId":"12345"},
 "auth": {
 "token": "1234567890"
 }
}
id= [0]
geoloc=Nominatim(user_agent="geoapiExercises")

def init():
 lat, long = "9.914470", "78.143418"
 location = geoloc.reverse(lat + "," + long)
```

```
Published data Successfully: %s {'d': {'dump1': 1, 'Level1': 60, 'Weight1': 600, 'Lat1': '9.914470', 'Long1': '78.143418', 'd_dump1': 1, 'Suburb1': 'Anna Nagar', 'City1': 'Madura
pick
dump  3
Published data Successfully: %s {'d': {'dump3': 3, 'Level3': 88, 'Weight3': 700, 'Lat3': '9.917916', 'Long3': '78.123496', 'd_dump3': 3, 'Suburb3': 'KK Nagar', 'City3': 'Madurai'
pick
dump  1
Published data Successfully: %s {'d': {'dump1': 1, 'Level1': 60, 'Weight1': 600, 'Lat1': '9.914470', 'Long1': '78.143418', 'd_dump1': 1, 'Suburb1': 'Anna Nagar', 'City1': 'Madura
```

File Edit View Navigate Code Refactor Run Tools VCS Window Help
pythonProject - main.py
pythonProject
main.py
try.py
poochi.py
Run: main
Version Control
Python Packages
TODO
Python Console
Problems
Terminal
Run
Services
Download pre-built shared indexes: Reduce the indexing time and CPU load with pre-built Python packages shared indexes // Always download // Download once // Don't show again // Configure... (today 6:25 PM)
105:17 CRLF UTF-8 1 space* Python 3.9 (pythonProject)
ENG IN
9:48 PM
11/19/2022

*__python script__

import wiotp.sdk.device

import time

from geopy.geocoders import Nominatim

import random

myConfig = {

"identity": {

"orgId": "zal46w",

"typeId": "NodeMCU",

"deviceId":"12345"},

"auth": {

"token": "1234567890"

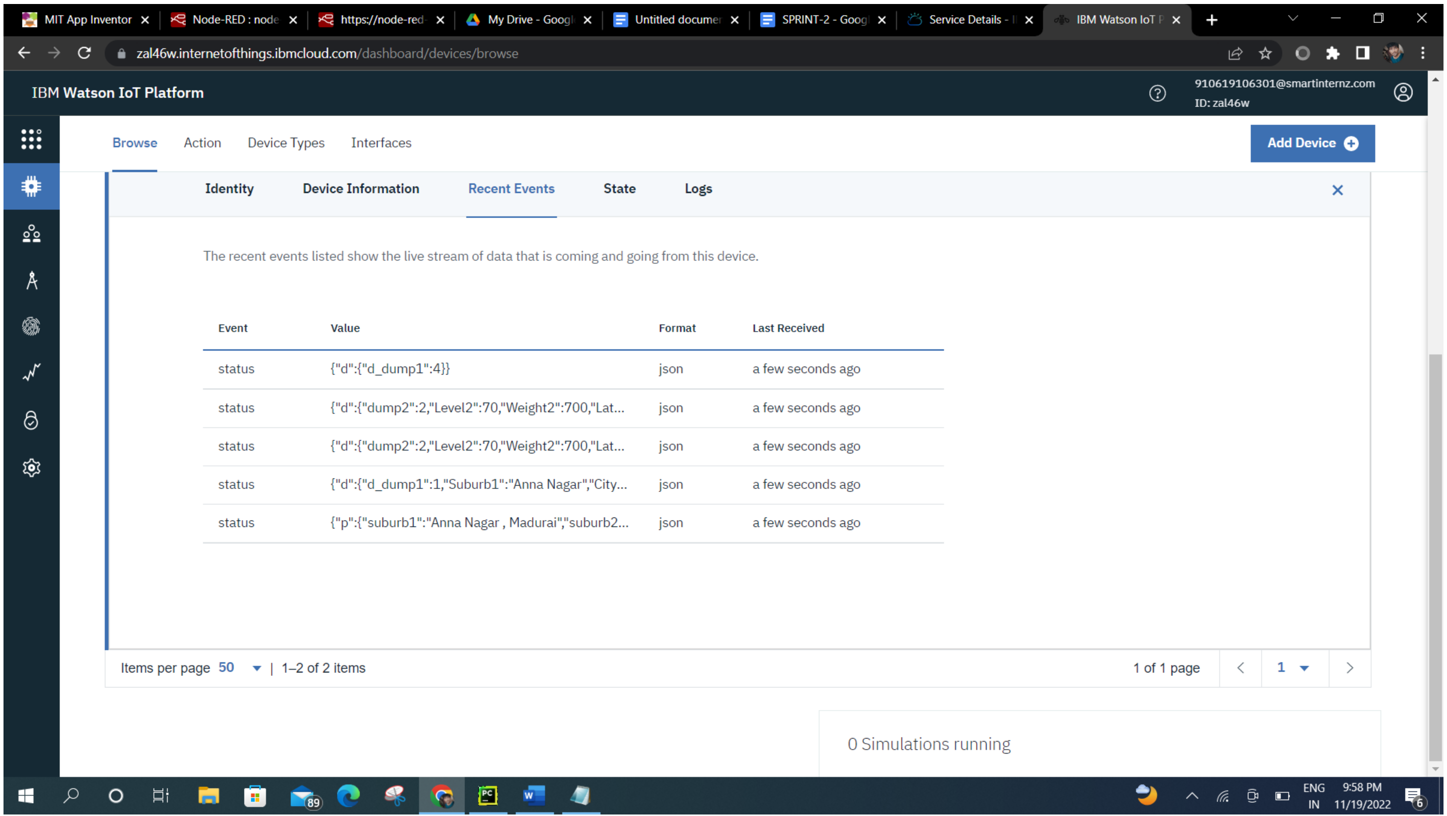
}

}

```
id= [0]

geoloc=Nominatim(user_agent="geoapiExercises")

def init():

        lat, long = "9.914470", "78.143418"

        location = geoloc.reverse(lat + "," + long)

        addr = location.raw['address']

        suburb1 = addr.get('suburb', '')

        city1 = addr.get('city', '')


        lat, long = "9.9933491", "78.127579"

        location = geoloc.reverse(lat + "," + long)

        addr = location.raw['address']

        suburb2 = "Tepakulam"

        city2 = addr.get('city', '')


        lat, long = "9.917916", "78.123496"

        location = geoloc.reverse(lat + "," + long)

        addr = location.raw['address']

        suburb3 = "KK Nagar"

        city3 = addr.get('city', '')


        mydata = {

         'd': {'d_dump1': 1, 'Suburb1': suburb1, 'City1': city1, 'd_dump2': 2, 'Suburb2': suburb2, 'City2': city2,
        'd_dump3': 3,

             'Suburb3': suburb3, 'City3': city3}}

        client.publishEvent(eventId="status", msgFormat="json", data=mydata, qos=0, onPublish=None)




def dumpster_1():

        lat, long = "9.914470", "78.143418"

        location = geoloc.reverse(lat + "," + long)

        addr = location.raw['address']

        suburb = addr.get('suburb', '')

        city = addr.get('city', '')

        level = 60
```

```
    weight = 600
    mydata = {'d': {'Level1': level, 'Weight1': weight, 'Lat1': lat, 'Long1': long,'d_dump1':1,'Suburb1': suburb, 'City1': city}}
    if (level > 50 and weight > 500):
     mydata = {
      'd': {'dump1': dumpid, 'Level1': level, 'Weight1': weight, 'Lat1': lat, 'Long1': long, 'd_dump1':1,'Suburb1': suburb, 'City1': city}}
     client.publishEvent(eventId="status", msgFormat="json", data=mydata, qos=0, onPublish=None)
     print("pick")
     time.sleep(2)
    client.publishEvent(eventId="status", msgFormat="json", data=mydata, qos=0, onPublish=None)
    print("dump ", dumpid)
    print("Published data Successfully: %s", mydata)
def dumpster_2():
    lat, long = "9.9933491", "78.127579"
    location = geoloc.reverse(lat + "," + long)
    addr = location.raw['address']
    suburb = "Tepakulam"
    city = addr.get('city', '')
    level = 70
    weight = 700
    mydata = {'d': {'Level2': level, 'Weight2': weight, 'Lat2': lat, 'Long2': long,'d_dump2':1,'Suburb2': suburb, 'City2': city}}
    if (level > 50 and weight > 500):
     mydata = {
      'd': {'dump2': dumpid, 'Level2': level, 'Weight2': weight, 'Lat2': lat, 'Long2': long,'d_dump2':2,'Suburb2': suburb, 'City2': city}}
     client.publishEvent(eventId="status", msgFormat="json", data=mydata, qos=0, onPublish=None)
     print("pick")
     time.sleep(2)
    client.publishEvent(eventId="status", msgFormat="json", data=mydata, qos=0, onPublish=None)
    print("dump ", dumpid)
    print("Published data Successfully: %s", mydata)
def dumpster_3():
    lat, long = "9.917916", "78.123496"
    location = geoloc.reverse(lat + "," + long)
    addr = location.raw['address']

    suburb = "KK Nagar"

    city = addr.get('city', '')

    level = 88

    weight = 700

    mydata = {'d': {'Level3': level, 'Weight3': weight, 'Lat3': lat, 'Long3': long,'d_dump3':3,'Suburb3': suburb, 'City3': city}}


    if (level > 50 and weight > 500):

     mydata = {

      'd': {'dump3': dumpid, 'Level3': level, 'Weight3': weight, 'Lat3': lat, 'Long3': long,'d_dump3':3,'Suburb3': suburb, 'City3': city}}

     client.publishEvent(eventId="status", msgFormat="json", data=mydata, qos=0, onPublish=None)

     print("pick")

     time.sleep(2)

    client.publishEvent(eventId="status", msgFormat="json", data=mydata, qos=0, onPublish=None)

    print("dump ", dumpid)

    print("Published data Successfully: %s", mydata)

def myCommandCallback(cmd):

    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])

    m=cmd.data['command']

    client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)

    client.connect()

while True:

    mydata = {'p': {'suburb1':"Anna Nagar , Madurai",'suburb2':"Tepakulam, Madurai",'suburb3':"KK Nagar, Madurai"}}

    client.publishEvent(eventId="status", msgFormat="json", data=mydata, qos=0, onPublish=None)

    dumpid = random.randint(1,3)

    init()

    if dumpid == 1:

     dumpster_1()

    elif dumpid == 2:

     dumpster_2()

    elif dumpid==3:

     dumpster_3()

mydata = {'d': {'d_dump1': 4}}
```

```
client.publishEvent(eventId="status", msgFormat="json", data=mydata, qos=0, onPublish=None)
client.commandCallback = myCommandCallback
time.sleep(2)
client.disconnect()
```



zal46w.internetofthings.ibmcloud.com/dashboard/devices/browse

IBM Watson IoT Platform

910619106301@smartinternz.com
ID: zal46w

Browse Action Device Types Interfaces Add Device

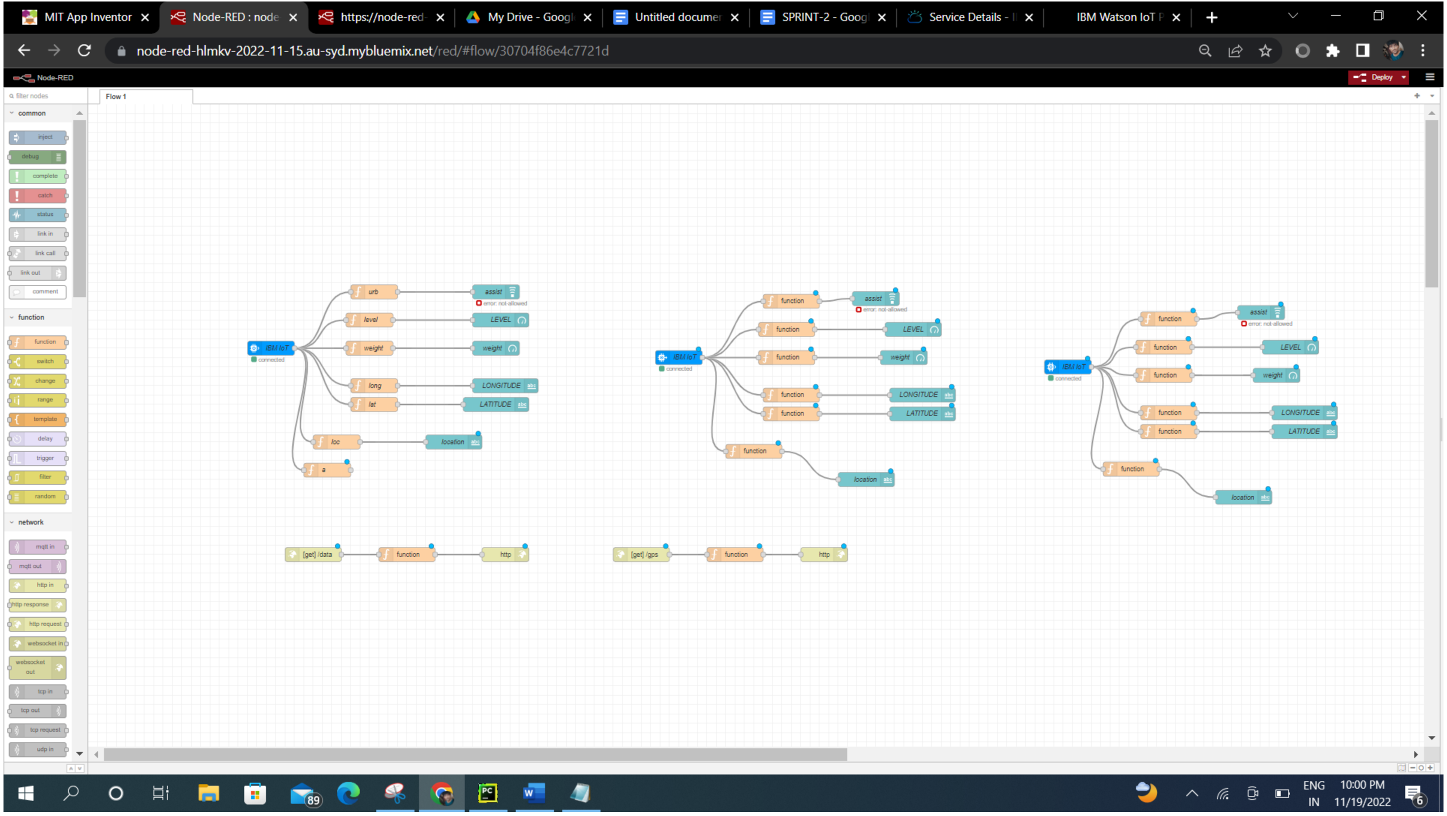
Identity Device Information Recent Events State Logs

The recent events listed show the live stream of data that is coming and going from this device.

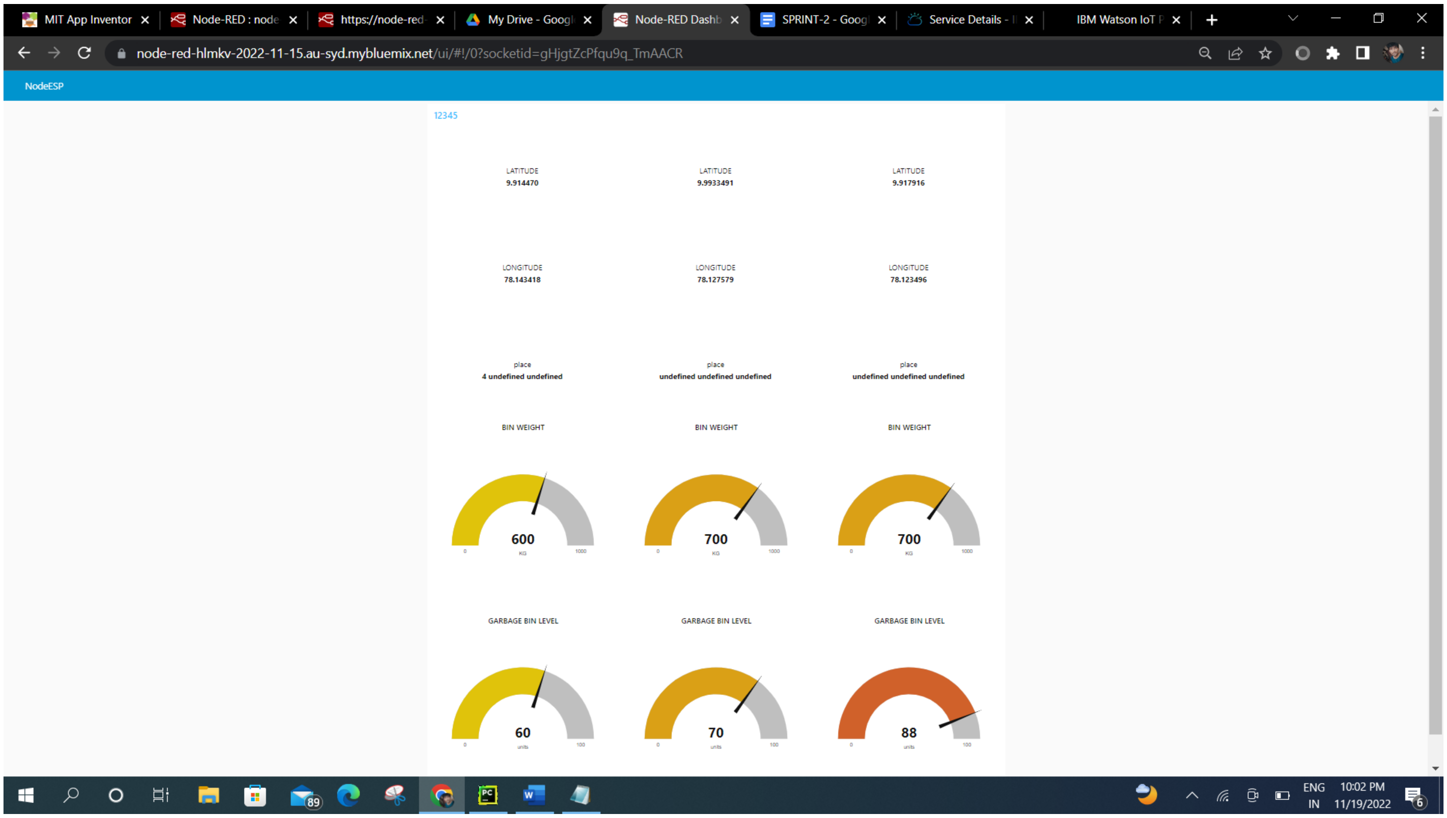
Event	Value	Format	Last Received
status	{"d":{"d_dump1":4}}	json	a few seconds ago
status	{"d":{"dump2":2,"Level2":70,"Weight2":700,"Lat...	json	a few seconds ago
status	{"d":{"dump2":2,"Level2":70,"Weight2":700,"Lat...	json	a few seconds ago
status	{"d":{"d_dump1":1,"Suburb1":"Anna Nagar","City...	json	a few seconds ago
status	{"p":{"suburb1":"Anna Nagar , Madurai","suburb2...	json	a few seconds ago

Items per page 50 | 1–2 of 2 items 1 of 1 page 1

0 Simulations running



node-red-hlmkv-2022-11-15.au-syd.mybluemix.net/red/#flow/30704f86e4c7721d



node-red-hlmkv-2022-11-15.au-syd.mybluemix.net/ui/#!/0?socketid=gHjgtZcPfqu9q_TmAACR

NodeESP

12345

LATITUDE **9.914470**	LATITUDE **9.9933491**	LATITUDE **9.917916**
LONGITUDE **78.143418**	LONGITUDE **78.127579**	LONGITUDE **78.123496**
place **4 undefined undefined**	place **undefined undefined undefined**	place **undefined undefined undefined**
BIN WEIGHT **600** 0 KG 1000	BIN WEIGHT **700** 0 KG 1000	BIN WEIGHT **700** 0 KG 1000
GARBAGE BIN LEVEL **60** 0 units 100	GARBAGE BIN LEVEL **70** 0 units 100	GARBAGE BIN LEVEL **88** 0 units 100