SPRINT - 4

Date	13 NOV 2022
Team ID	PNT2022TMID48570
Project Name	Smart Waste Management System for Metropolitan Cities

1, Simulate python code in Python IDE software to transmit data to IBM Watson IOT platform

Python code:

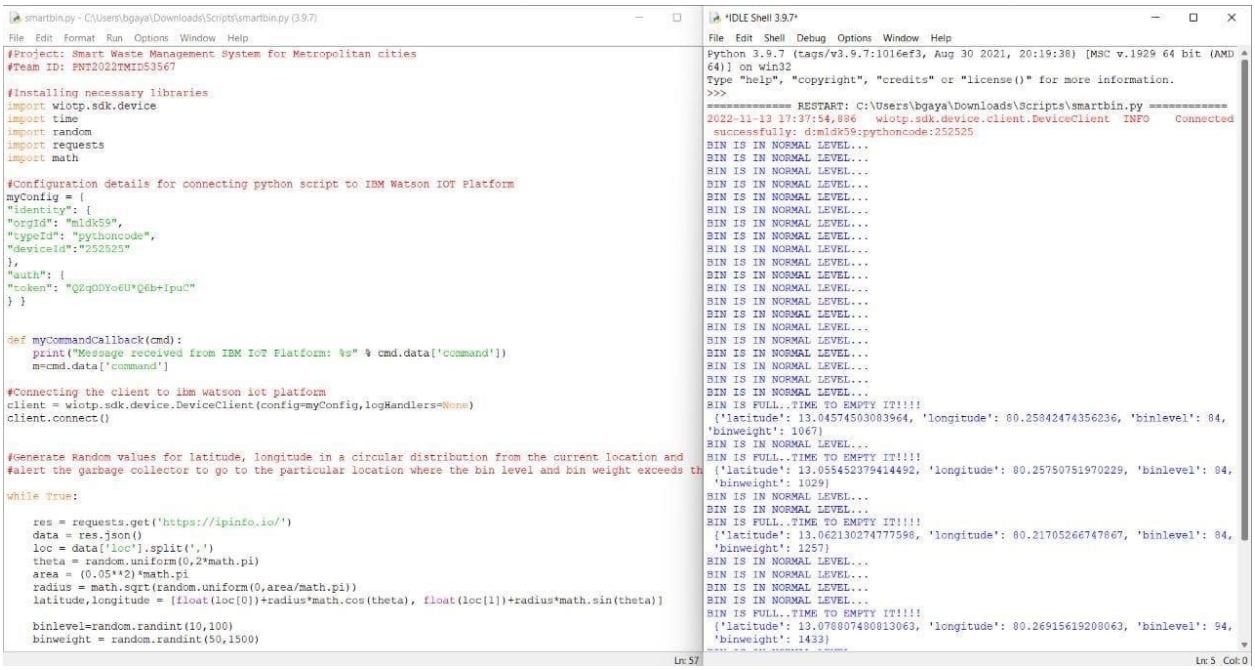
smartbin.py:

```
#Project: Smart Waste Management System for Metropolitan cities
#Team ID: PNT2022TMID53567

#Installing necessary libraries
import wiotp.sdk.device
import time import random
import requests
import math

#Configuration details for connecting python script to IBM Watson IOT Platform
myConfig = {
"identity": {
"orgId": "mldk59",
"typeId": "pythoncode",
"deviceId":"252525"
},
"auth": {
"token": "QZqODYo6U*Q6b+IpuC"
} }


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

```
#Connecting the client to ibm watson iot platform
client = wiotp.sdk.device.DeviceClient(config=myConfig,logHandlers=None)
client.connect()


#Generate Random values for latitude, longitude in a circular distribution from the
current location and
#alert the garbage collector to go to the particular location where the bin level and
bin weight exceeds the threshold

while True:

   res = requests.get('https://ipinfo.io/')
data = res.json()     loc =
data['loc'].split(',')     theta =
random.uniform(0,2*math.pi)     area =
(0.05**2)*math.pi
   radius = math.sqrt(random.uniform(0,area/math.pi))
   latitude,longitude = [float(loc[0])+radius*math.cos(theta), float(loc[1])+radius*
math.sin(theta)]

   binlevel=random.randint(10,100)
   binweight = random.randint(50,1500)

   if binweight>=1000 and binlevel>80:
      myData={'latitude':latitude, 'longitude':longitude,'binlevel':binlevel,
            'binweight':binweight}
      client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=
0, onPublish=None)
      ##print("Published data Successfully: %s", myData)
print("BIN IS FULL..TIME TO EMPTY IT!!!!\n",myData)
client.commandCallback = myCommandCallback        time.sleep(2)
      #break

else :
      print("BIN IS IN NORMAL LEVEL...")
      time.sleep(2)
```

#Disconnect the client connection client.disconnect()

Python IDE output:

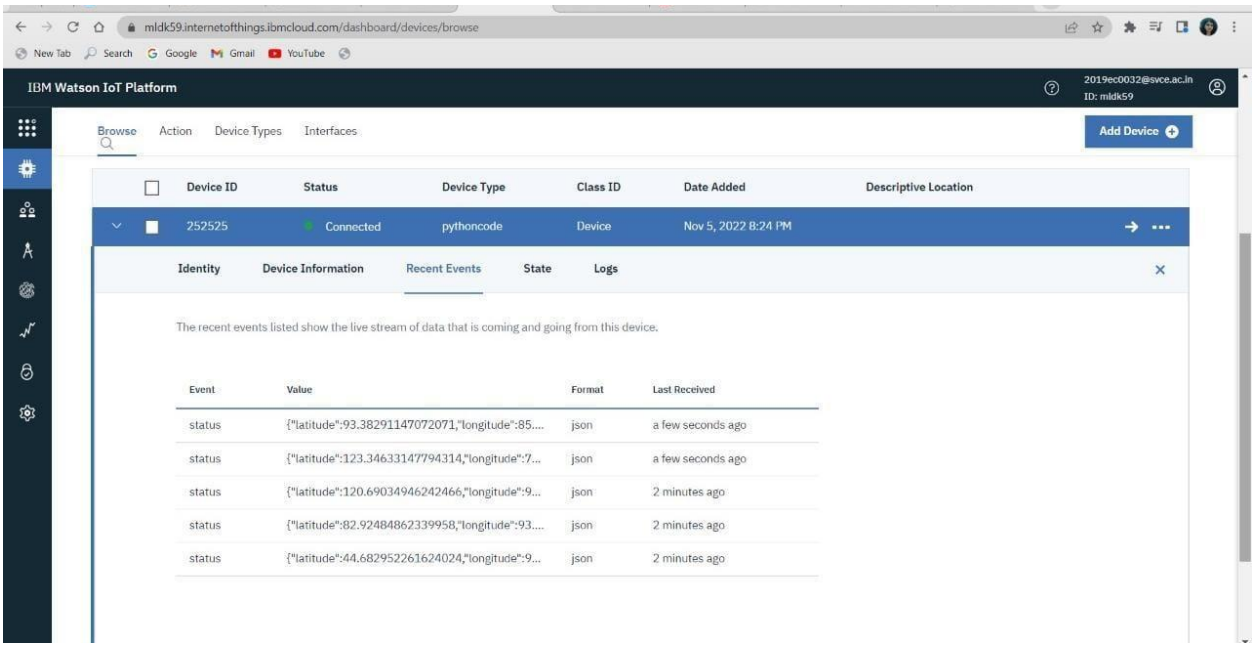


```
#Project: Smart Waste Management System for Metropolitan cities
#Team ID: PNT2022TMID53567

#Installing necessary libraries
import wiotp.sdk.device
import time
import random
import requests
import math

#Configuration details for connecting python script to IBM Watson IOT Platform
myConfig = {
"identity": {
"orgId": "mldk59",
"typeId": "pythoncode",
"deviceId":"252525"
},
"auth": {
"token": "Q2qODYo6U*Q6b+IpuC"
} }


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

#Connecting the client to ibm watson iot platform
client = wiotp.sdk.device.DeviceClient(config=myConfig,logHandlers=None)
client.connect()


#Generate Random values for latitude, longitude in a circular distribution from the current location and
#alert the garbage collector to go to the particular location where the bin level and bin weight exceeds th

while True:

    res = requests.get('https://ipinfo.io/')
    data = res.json()
    loc = data['loc'].split(',')
    theta = random.uniform(0,2*math.pi)
    area = (0.05**2)*math.pi
    radius = math.sqrt(random.uniform(0,area/math.pi))
    latitude,longitude = [float(loc[0])+radius*math.cos(theta), float(loc[1])+radius*math.sin(theta)]

    binlevel=random.randint(10,100)
    binweight = random.randint(50,1500)
```

```
Python 3.9.7 (tags/v3.9.7:1016ef3, Aug 30 2021, 20:19:38) [MSC v.1929 64 bit (AMD
64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:\Users\bgaya\Downloads\Scripts\smartbin.py ==========
2022-11-13 17:37:54,886  wiotp.sdk.device.client.DeviceClient  INFO    Connected
 successfully: d:mldk59:pythoncode:252525
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS FULL..TIME TO EMPTY IT!!!!
 {'latitude': 13.04574503083964, 'longitude': 80.25842474356236, 'binlevel': 84,
'binweight': 1067}
BIN IS IN NORMAL LEVEL...
BIN IS FULL..TIME TO EMPTY IT!!!!
 {'latitude': 13.055452379414492, 'longitude': 80.25750751970229, 'binlevel': 84,
 'binweight': 1029}
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS FULL..TIME TO EMPTY IT!!!!
 {'latitude': 13.062130274777598, 'longitude': 80.21705266747867, 'binlevel': 84,
 'binweight': 1257}
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS IN NORMAL LEVEL...
BIN IS FULL..TIME TO EMPTY IT!!!!
 {'latitude': 13.078007480013063, 'longitude': 80.26915619208063, 'binlevel': 94,
 'binweight': 1433}
```

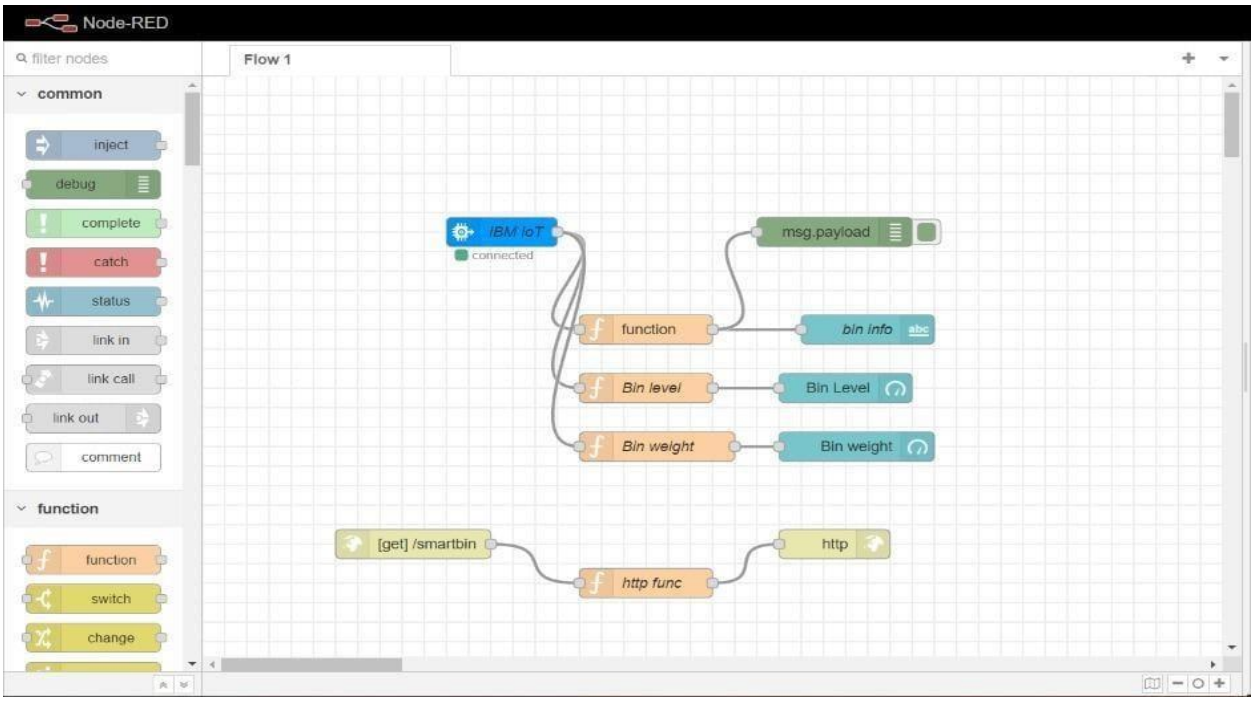
2. Data is transferred to IBM Watson IoT platform.

IBM Platform output:



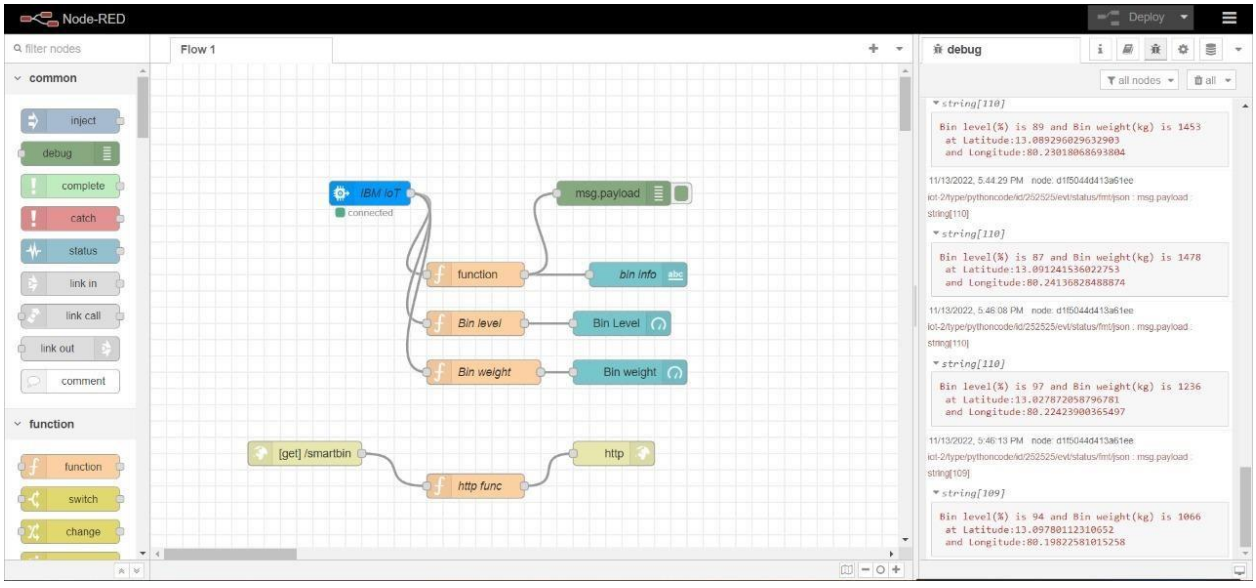
3. Data transfer from IBM Watson IOT platform and Python IDE to Node RED.

Node-RED:

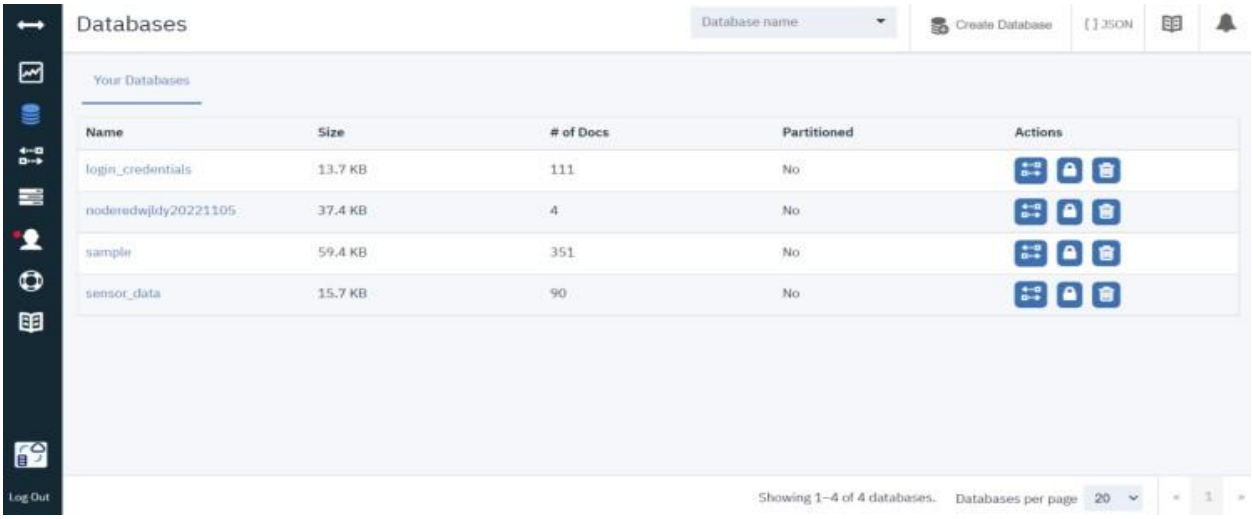


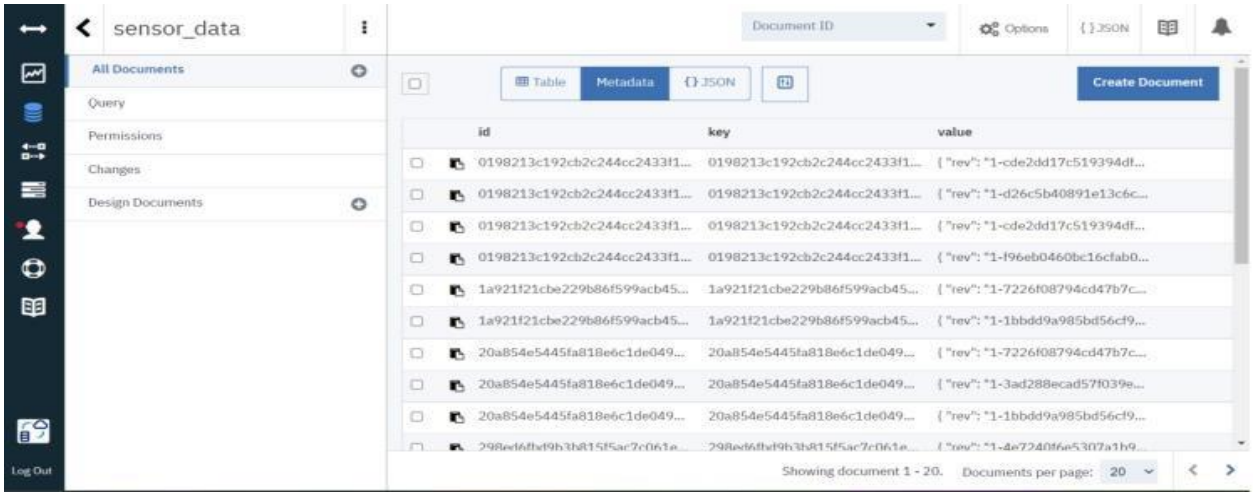
4. Node-RED Connection setup for data transmission from IBM Watson IoT platform to Node-RED dashboard and viewing in Web UI .

Node-RED:

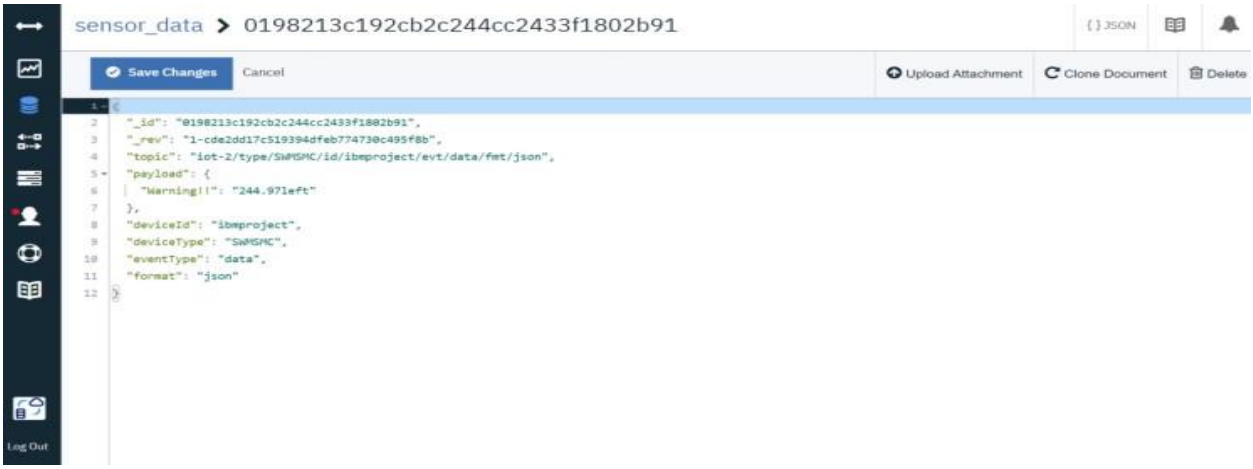


5. Storing database in IBM Cloudant DB

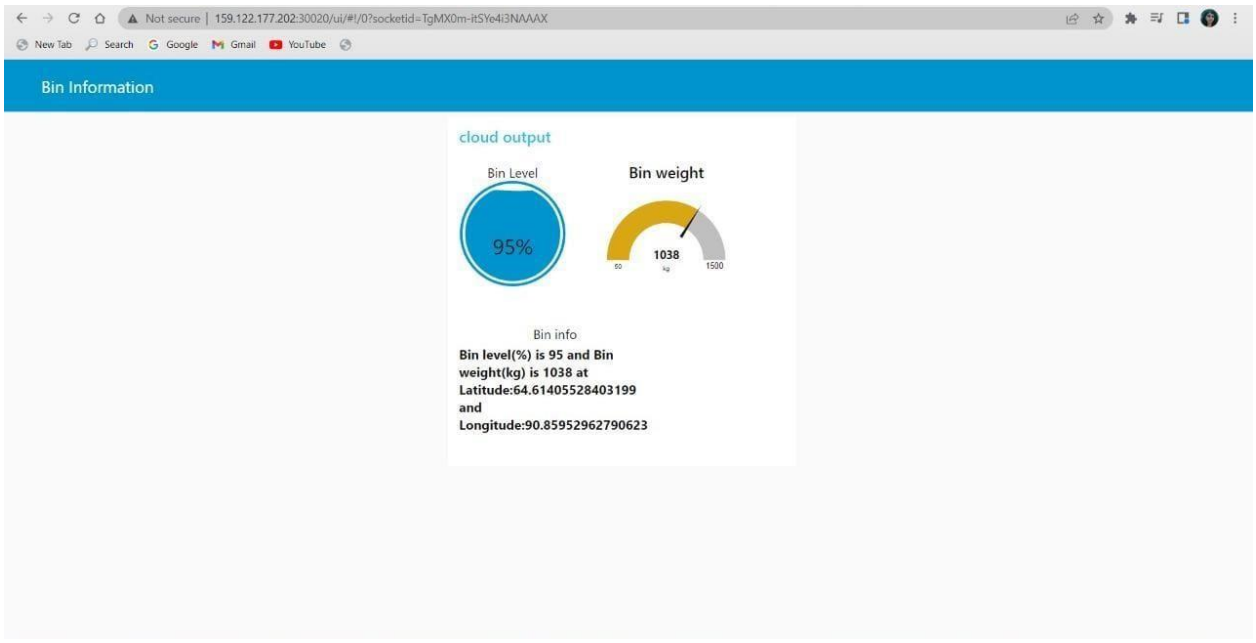




6. Data is stored in JSON format

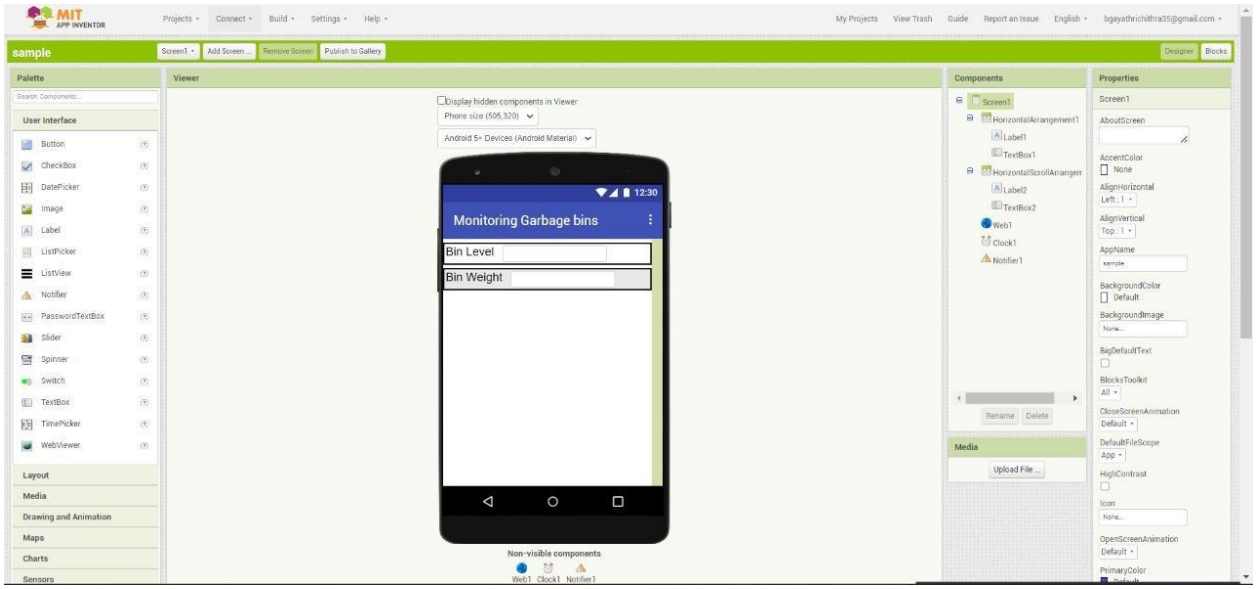


Web UI:

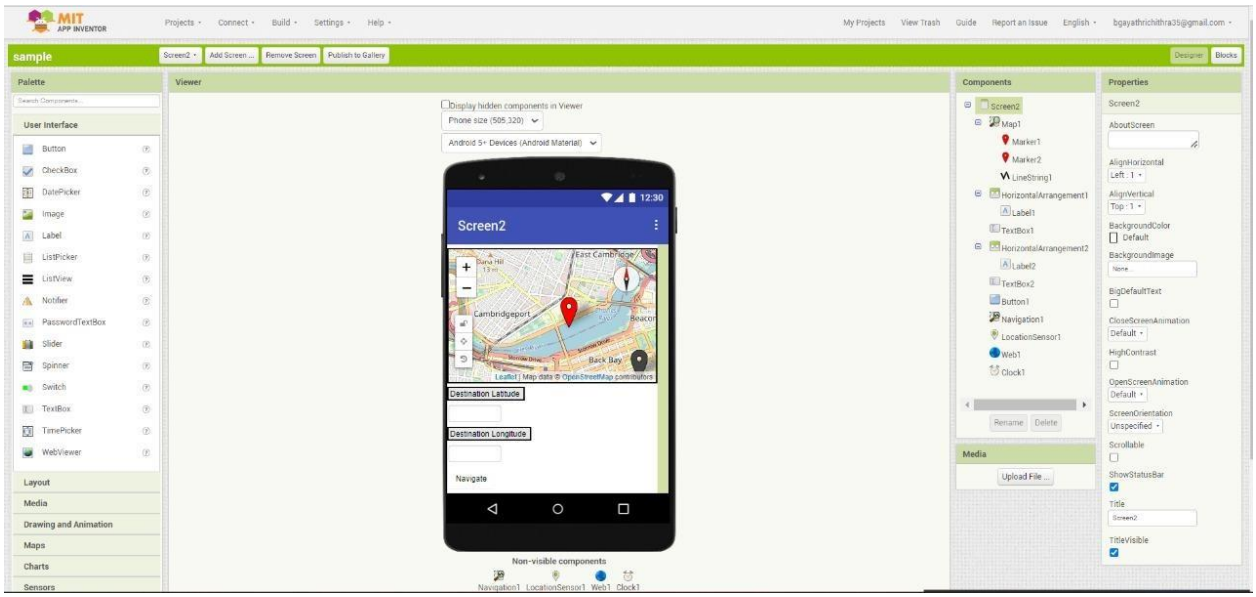


7. App is created using MIT App inventer

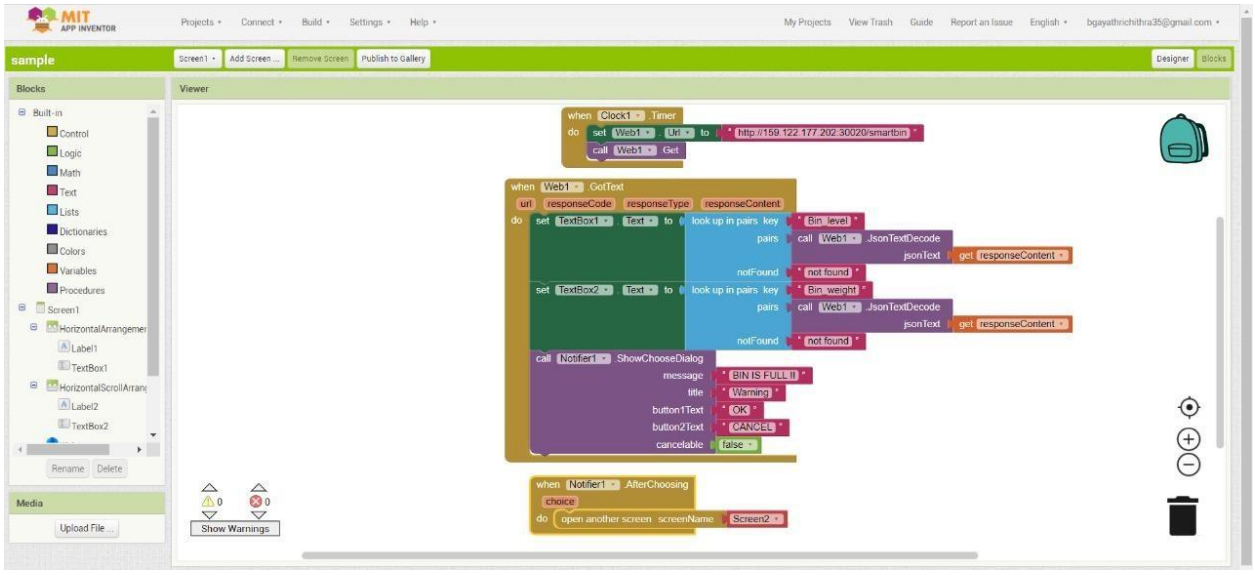
Screen 1:



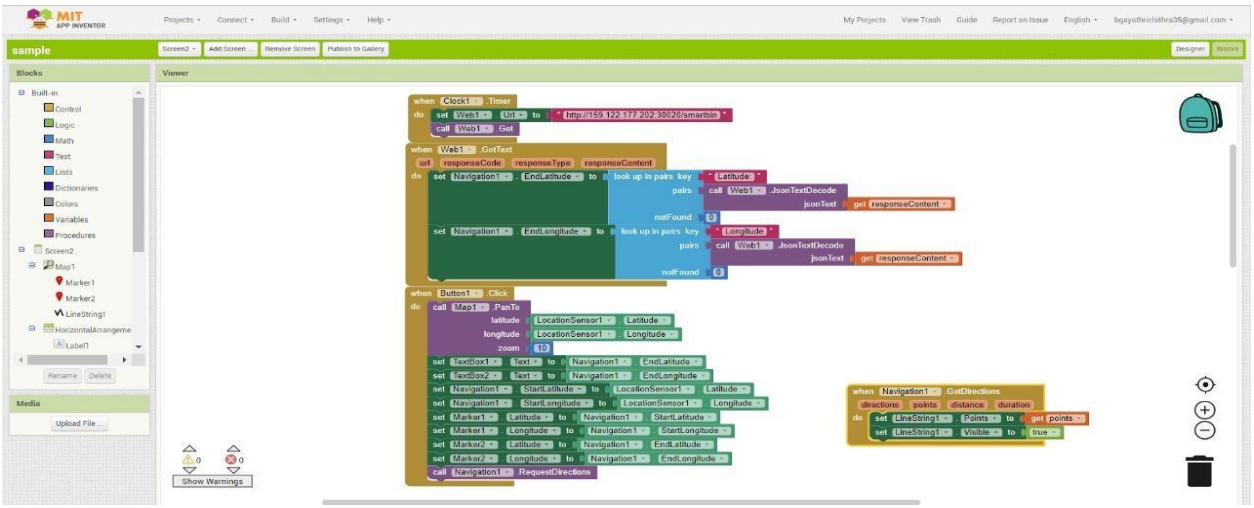
Screen 2:



Screen 1 blocks:



Screen 2 blocks:



8. Install MIT AI2 Companion in phone and scan the QR code showed in AI connect