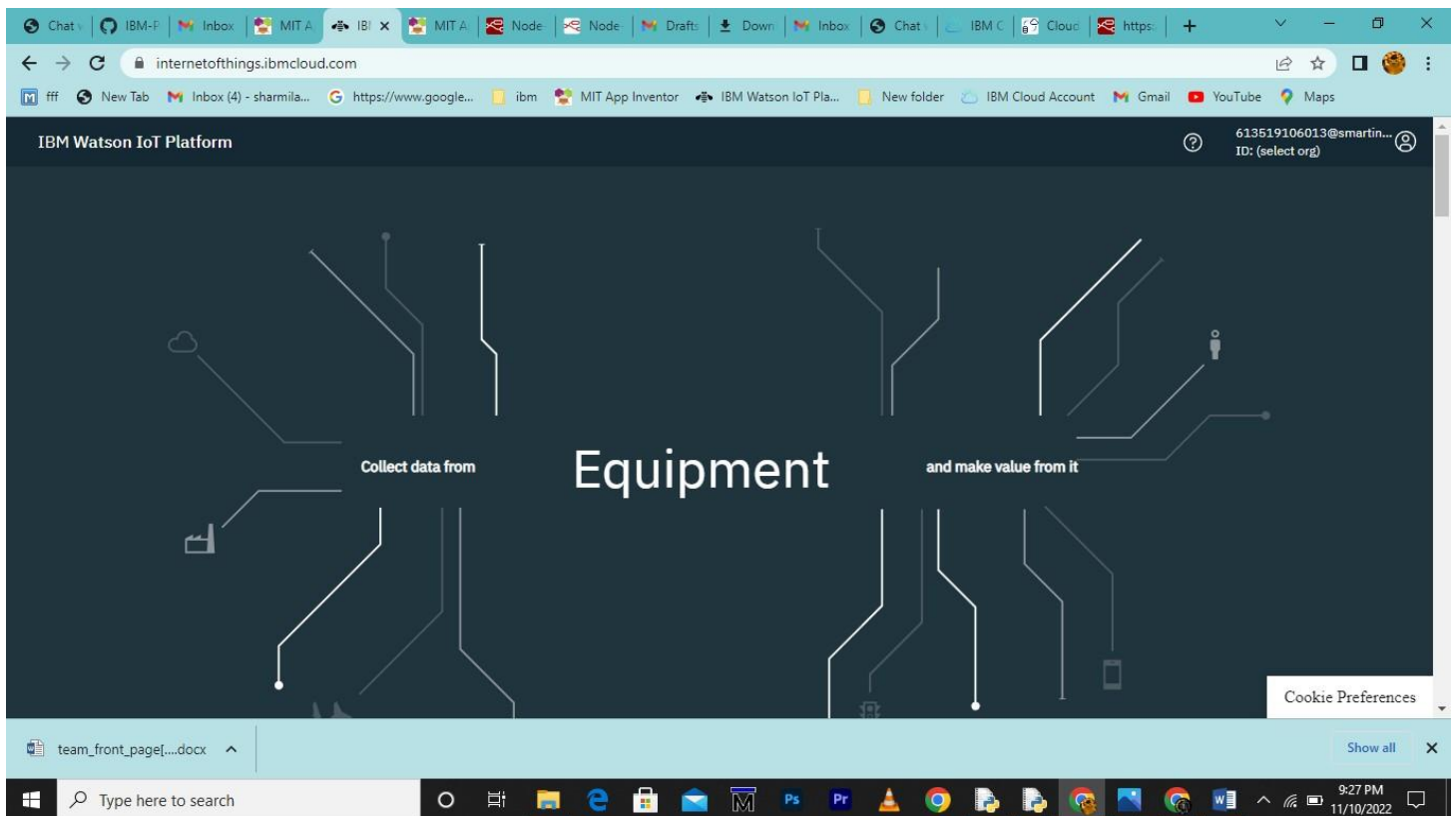


Project Development –Delivery of Sprint 1

Creating and Connecting IBM cloud for Project and Python Code

TITLE	IOT based child safety gadget for child safety monitoring and notification
TEAM ID	PNT2022TMID44407

Creating IBM Cloud Service and creating the device:



The screenshot displays the IBM Watson IoT Platform dashboard. The top navigation bar includes tabs for Chat, IBM-F, Inbox, MIT A, IBI, MIT A, Node, Node, Drafts, Down, Inbox, Chat, IBM C, Cloud, and https. The main header shows the user's email (613519106013@smartinternz.com) and ID (zw6lb). The dashboard is divided into sections: Browse, Action, Device Types, and Interfaces. A table lists devices, with one device (ID 13, Type ABCD) highlighted. A modal window shows details for this device, including its ID, type, date added, and connection status (Disconnected). A simulation running indicator is visible at the bottom right of the modal.

Device ID	Status	Device Type	Class ID	Date Added
13	Disconnected	ABCD	Device	Nov 2, 2022 10:55 PM

Identity	Device Information	Recent Events	State	Logs
Device ID	13			
Device Type	ABCD			
Date Added	Nov 2, 2022 10:55 PM			
Added By	613519106013@smartinternz.com			
Connection Status	Disconnected			
	Last Connected: Nov 10, 2022 7:48 PM			
	Client Address: 106.211.215.236 SecureToken			
	Duration: a few seconds			
	Data Transferred: 18.9 KB			

1 Simulation running

Creating Python Code:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "zw6lb"
deviceType = "ABCD"
deviceId = "13"
authMethod = "token"
authToken = "12345678"
#api key {a-illza1-mbdxqo6z0s}
#api token
{zSYzISuAWF&F_x7GkT}
```

```

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
print("power on ")
    print("checking connection to waston iot...")
    time.sleep(2)
deviceCli.connect()
print("dear user ... welcome to IBM-IOT ")
print("i can provide your children live location and temperature ")print()
name=str(input("enter your child name:"))
while True:

    temperature=random.randint(20,50)#random temperature for your child
    latitude=random.uniform(10.781377,10.78643)#random latitude for your child
    longitude=random.uniform(79.129113,79.134014)#random longitude for your childa="Child
inside the geofence"
    b=" Child outside the geofence"
    c="High temperature"
    d="Low temperature"

```

```

x={'your_child_Zone':a
}
y={'your_child_Zone':b
}
z={'temp_condition':c}
w={'temp_condition':d}
data = { 'temp' :
temperature, 'lat':
latitude,'lon':longitude,'
name':name }#print
data

def myOnPublishCallback():

print ("Published Temperature = %s C" % temperature, "latitude = %s %% " % latitude, "longitude
= %s %% " % longitude, "to IBM Watson")

    print("\n")

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

    if latitude>=10.78200 and latitude<=10.786000 and longitude >=79.130000 and longitude
<=79.133000:

deviceCli.publishEvent("IoTSensorgpsdata","json",data=x,qos=0,on_publish=myOnPublishCallb
ack)

    print(x)
    print("\n")
else:

deviceCli.publishEvent("IoTSensorgpsdata","json",data=y,qos=0,on_publish=myOnPublishCallb
ack)

    print(y)
    print("\n")

```

```
if (temperature>35):
```

```
deviceCli.publishEvent("IoTSensorgpsdata","json",data=z,qos=0,on_publish=myOnPublishCallb  
ack)
```

```
    print(c)
```

```
    print("\n")
```

```
else:
```

```
deviceCli.publishEvent("IoTSensorgpsdata","json",data=w,qos=0,on_publish=myOnPublishCall  
back)
```

```
    print(d)
```

```
print("\n")
```

if not success:

```
print("Not connected to IoTf")
```

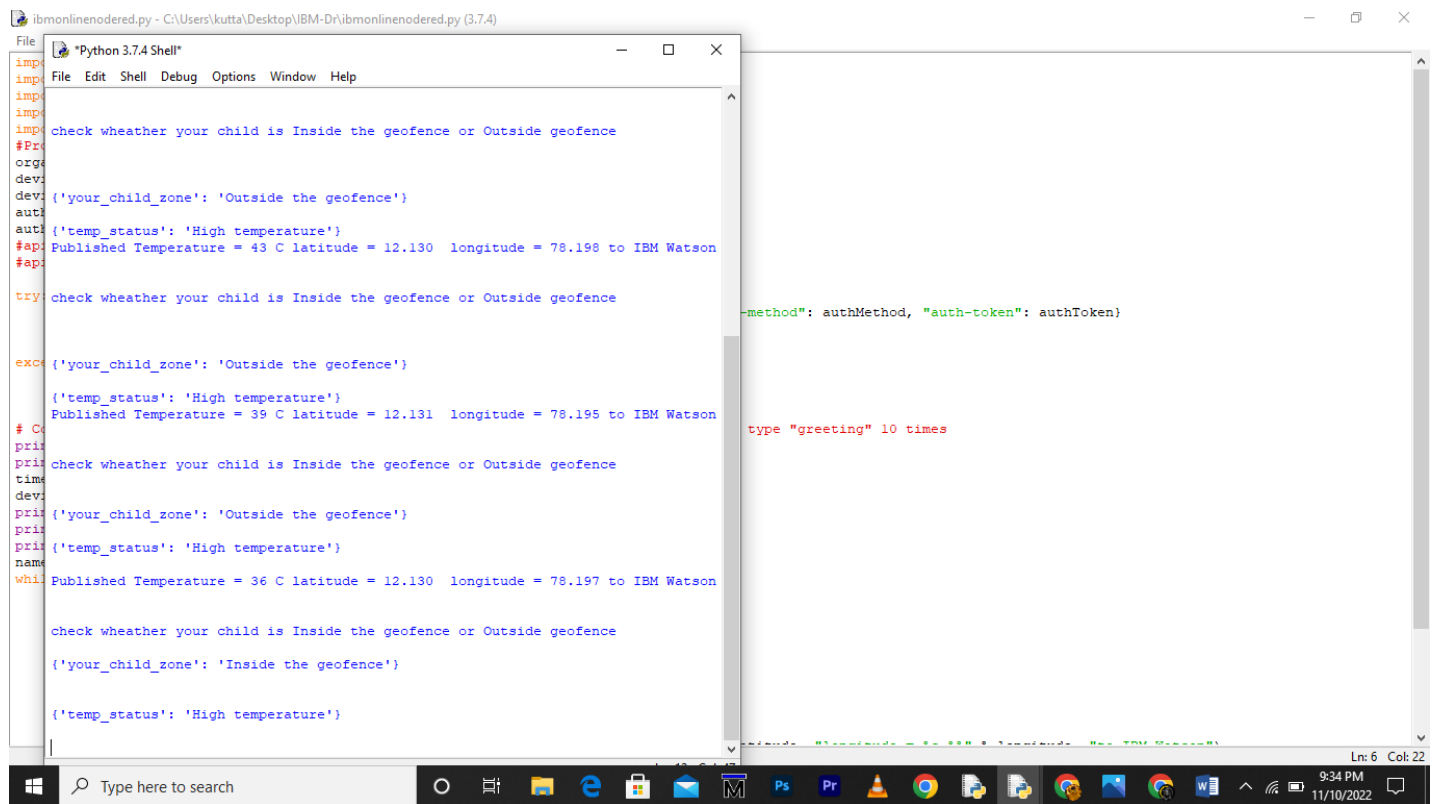
```
print("\n")
```

```
time.sleep(3)
```

Disconnect the device and application from the cloud

```
deviceCli.disconnect()
```

Connecting IBM Watson and python Code:



Chat v IBM-P Inbox MIT A IBI x MIT A Node Node Draft Down Inbox Chat IBM C Cloud https +

zwx6lb.internetofthings.ibmcloud.com/dashboard/devices/browse

M fff New Tab Inbox (4) - sharmila... https://www.google... ibm MIT App Inventor IBM Watson IoT Pla... New folder IBM Cloud Account Gmail YouTube Maps

IBM Watson IoT Platform 613519106013@smartinternz.com ID: zwx6lb

Browse Action Device Types Interfaces

Add Device +

13 Connected ABCD Device Nov 2, 2022 10:55 PM

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
IoTSensorgp...	{"temp_status":"High temperature"}	json	a few seconds ago
IoTSensorgp...	{"your_child_zone":"Outside the geofence"}	json	a few seconds ago
IoTSensorgp...	{"temp":50,"lat":12.132819998043411,"lon":78...	json	a few seconds ago
IoTSensorgp...	{"temp_status":"Low temperature"}		
IoTSensorgp...	{"your_child_zone":"Outside the geofence"}		

1 Simulation running

team_front_page{....docx Show all

Type here to search

9:35 PM 11/10/2022