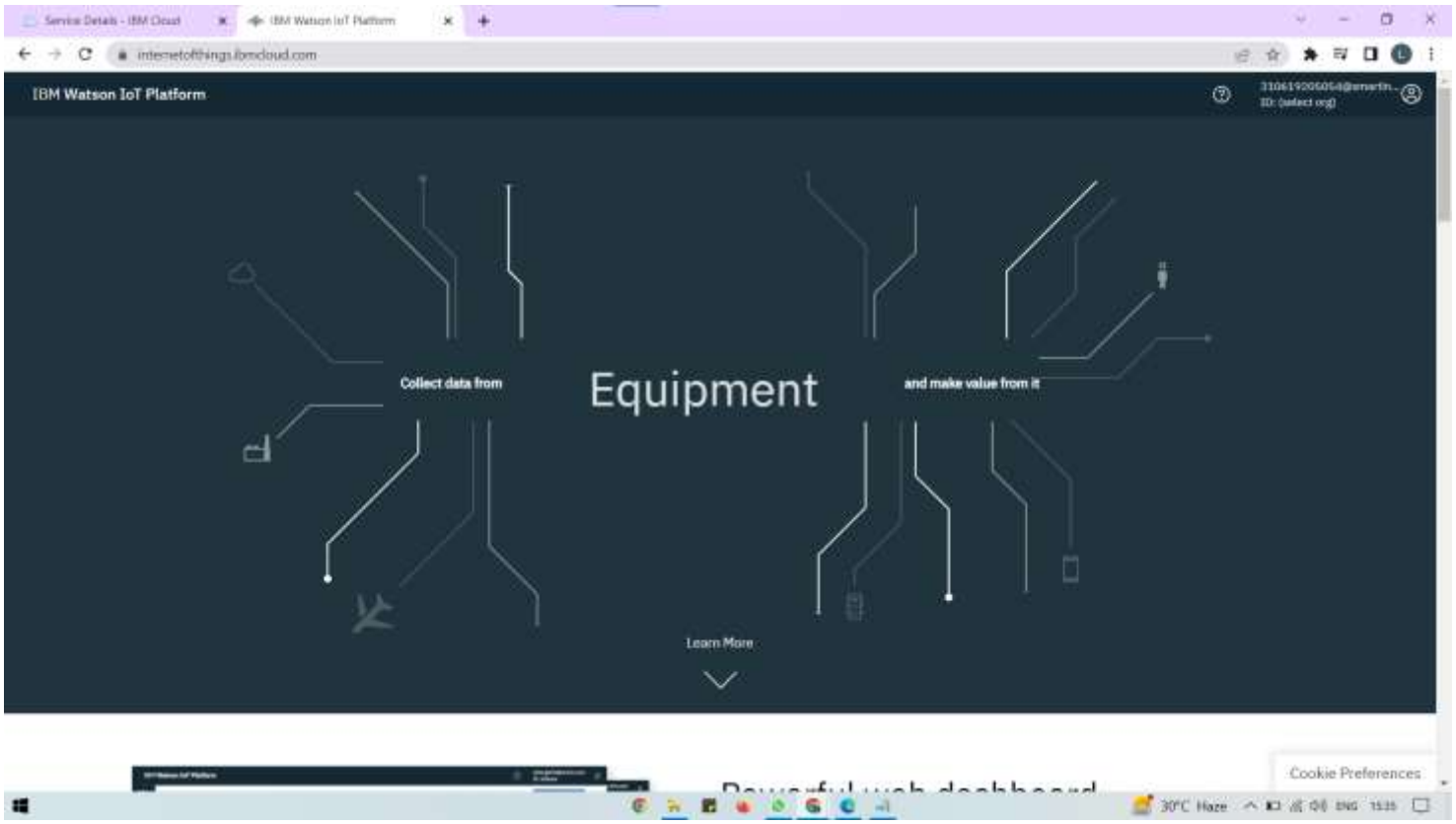
IOT Based Safety Gadget for Child Safety Monitoring and Notification

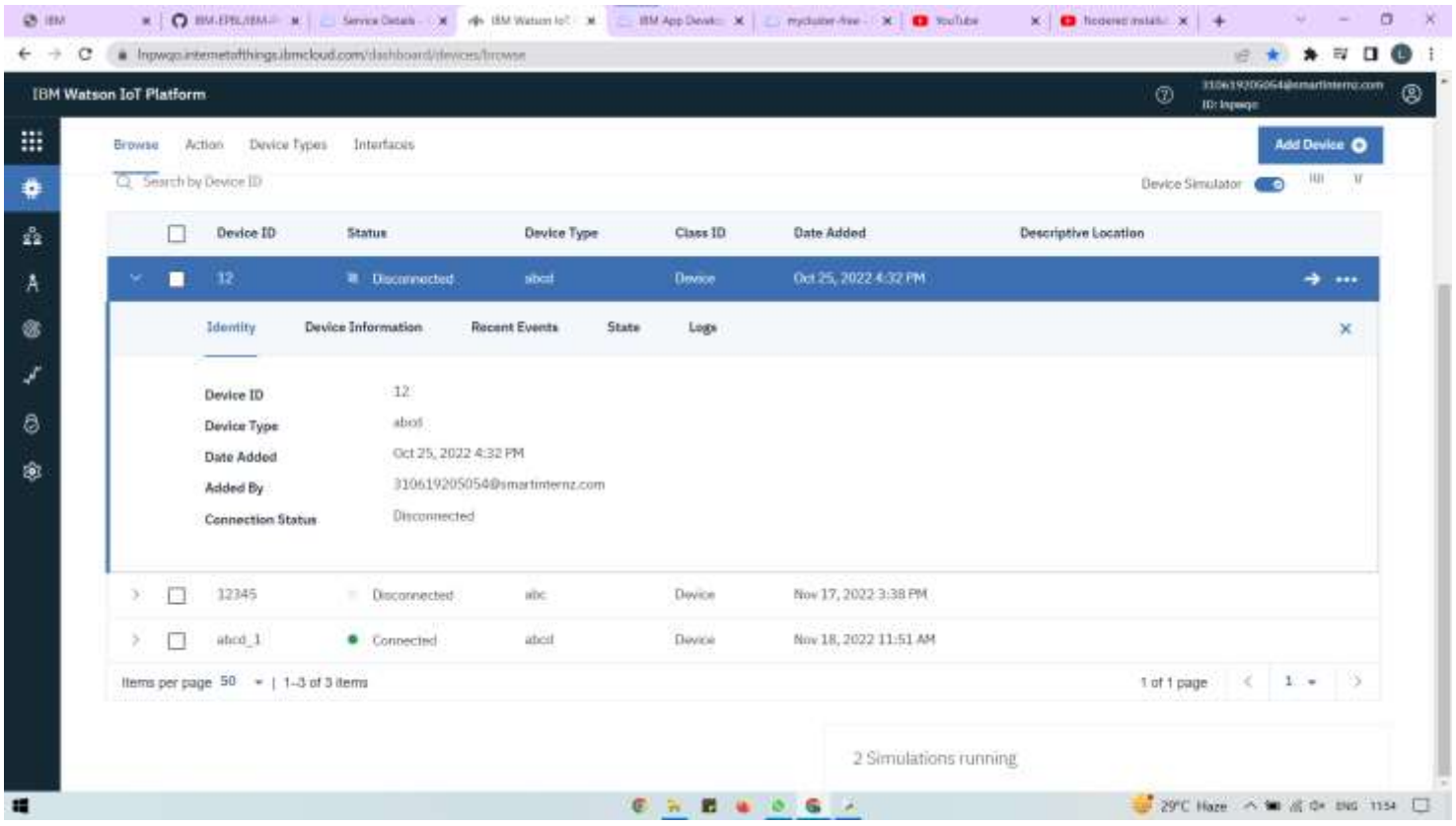
Project Development –Delivery of Sprint 1

Creating and Connecting IBM cloud for Project and Python Code

TITLE	IOT based child safety gadget for child safety monitoring and notification
DOMAIN NAME	INTERNET OF THINGS
TEAM ID	PNT2022TMID09639
TEAM LEADERNAME	A E S Loghapriya
TEAM MEMBER NAME	Aishwarya S Deepa Harshini C A Deepak M
MENTOR NAME	Mrs. K Johny Elma

Creating IBM Cloud Service and creating the device:





Creating Python Code:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "lnpwqo"
deviceType = "abcd"
deviceId = "12"
authMethod = "use-token-auth"
authToken = "12345678"
#api key {a-lnpwqo-623qb5z8ny}
#api token {tJZoLa3sq5judZGuaw}

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
child
    a="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=myO
nPublishCallback)
        print(x)
        print("\n")
     else:


deviceCli.publishEvent("IoTSensorgpsdata","json",data=y,qos=0,on_publish=myO
nPublishCallback)
        print(y)
        print("\n")

     if (temperature>35):


deviceCli.publishEvent("IoTSensorgpsdata","json",data=z,qos=0,on_publish=myO
nPublishCallback)
        print(c)
        print("\n")
     else:


deviceCli.publishEvent("IoTSensorgpsdata","json",data=w,qos=0,on_publish=myO
nPublishCallback)
        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: