

## PROJECT DEVELOPMENT PHASE

### SPRINT 1

Date	19 November 2022
Team ID	PNT2022TMID13514
Project name	Gas Leakage Monitoring & Alerting System for Industries

#### ANALYZE THE PREREQUISITES

Needed prerequisites for real time river water quality monitoring and control system using Internet Of Things (IoT) were

- ❖ IBM Watson IoT Platform
- ❖ Node-RED Service
- ❖ Cloudant DB

#### Python code:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
organization = "l27fmg"
deviceType = "12345"
deviceId = "123456"
authMethod = "token"
authToken = "123456789"
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="alarmon":
        print ("Alarm is on")
    elif status == "alarmoff":
        print ("Alarm is off")
    else:
        print ("Please send proper command")
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()
deviceCli.connect()
while True:
    gasconcentration = random.randint(50,110)
    Humidity = random.randint(90,110)
```

```

Temperature = random.randint(90,110)
if gasconcentration>80:
    gas_status=" Hurry gas is leaking \n Alert!!!!!"

else:
    gas_status="gas is not leaking"
data = {'gasconcentration' : gasconcentration,'Humidity' :
Humidity,'Temperature' :Temperature,'gas_status':gas_status}

def myOnPublishCallback():
    print(" GasConcentration = %s PPM" % gasconcentration, "to IBM Watson")
    print(" Humidity = %s%%" % Humidity, "to IBM Watson")
    print(" Temperature = %s C" % Temperature, "to IBM Watson")
    print(gas_status)
    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoT")
        time.sleep(10)
deviceCli.commandCallback=myCommandCallback
deviceCli.disconnect()

```

The screenshot shows a Python script running in a shell window. The script generates random values for gas concentration, humidity, and temperature, and prints them along with a status message. The output is as follows:

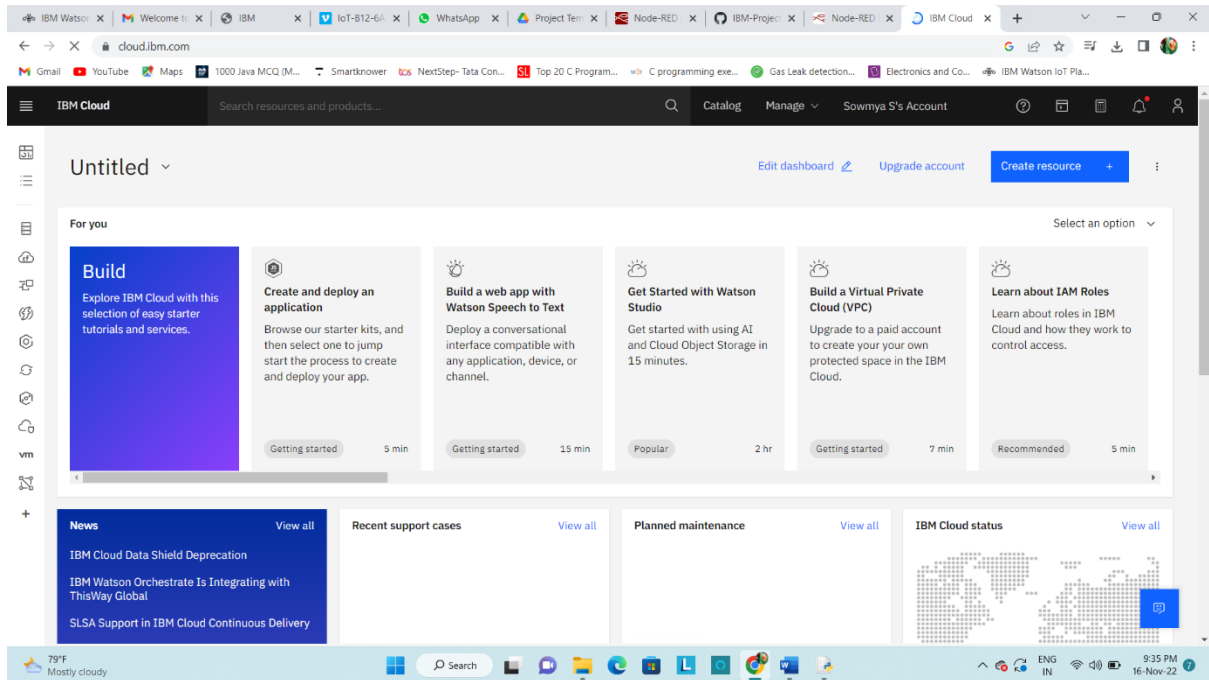
```

Temperature = 92 C to IBM Watson
gas is not leaking
GasConcentration = 72 PPM to IBM Watson
Humidity = 92% to IBM Watson
Temperature = 100 C to IBM Watson
gas is not leaking
GasConcentration = 65 PPM to IBM Watson
Humidity = 103% to IBM Watson
Temperature = 103 C to IBM Watson
gas is not leaking
GasConcentration = 58 PPM to IBM Watson
Humidity = 97% to IBM Watson
Temperature = 99 C to IBM Watson
gas is not leaking
GasConcentration = 71 PPM to IBM Watson
Humidity = 100% to IBM Watson
Temperature = 103 C to IBM Watson
gas is not leaking
GasConcentration = 107 PPM to IBM Watson
Humidity = 103% to IBM Watson
Temperature = 91 C to IBM Watson
Hurry gas is leaking
Alert!!!!
GasConcentration = 98 PPM to IBM Watson
Humidity = 106% to IBM Watson
Temperature = 101 C to IBM Watson
Hurry gas is leaking
Alert!!!!
GasConcentration = 55 PPM to IBM Watson
Humidity = 95% to IBM Watson
Temperature = 95 C to IBM Watson
gas is not leaking
GasConcentration = 87 PPM to IBM Watson
Humidity = 108% to IBM Watson
Temperature = 104 C to IBM Watson
Hurry gas is leaking
Alert!!!!
GasConcentration = 66 PPM to IBM Watson
Humidity = 102% to IBM Watson
Temperature = 110 C to IBM Watson
gas is not leaking
GasConcentration = 78 PPM to IBM Watson
Humidity = 104% to IBM Watson
Temperature = 104 C to IBM Watson
gas is not leaking

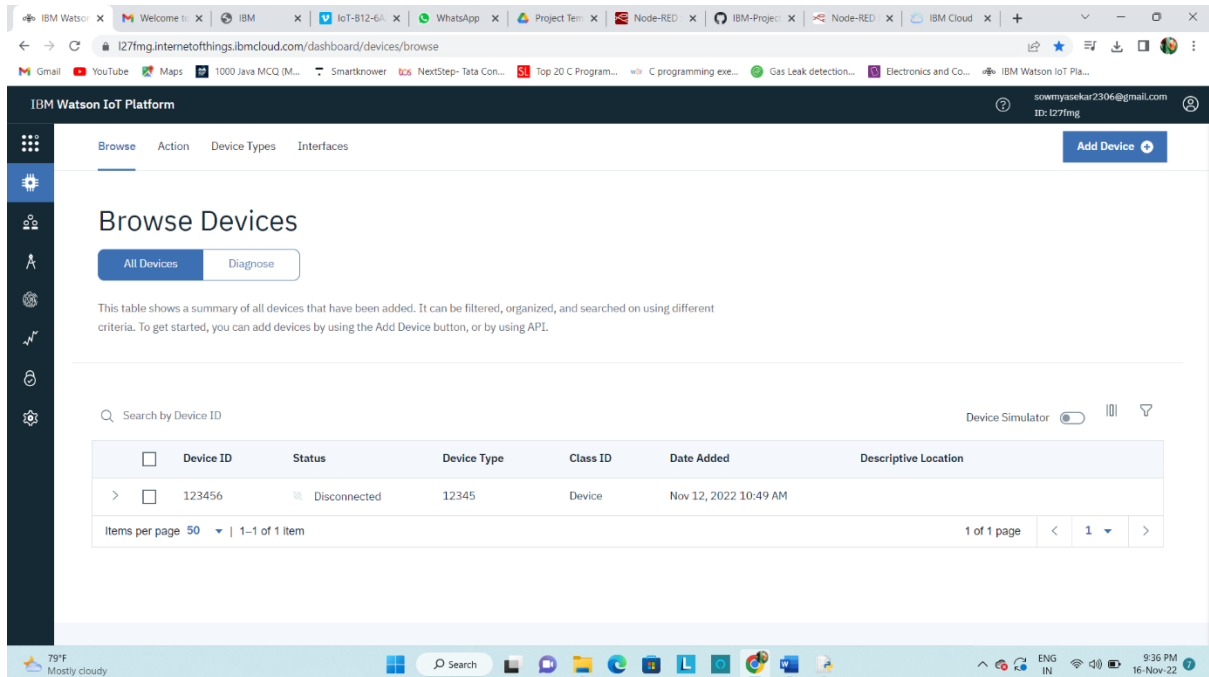
```

Code runs successfully and random output values are generated

## Creation of IBM cloud



## Procedure for the creation of IBM IOT watson



## Device creation

The screenshot shows the IBM Watson IoT Platform dashboard. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A search bar is present with the text 'Search by Device ID'. The main content area displays a table with the following columns: Device ID, Status, Device Type, Class ID, Date Added, and Descriptive Location. A single device is listed with ID 123456, Status Disconnected, Device Type 12345, Class ID Device, and Date Added Nov 12, 2022 10:49 AM. Below the table, a detailed view of the device is shown, including fields for Device ID, Device Type, Date Added, Added By, and Connection Status.

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
123456	Disconnected	12345	Device	Nov 12, 2022 10:49 AM	

Device ID	Device Information	Recent Events	State	Logs
Device ID	123456			
Device Type	12345			
Date Added	Nov 12, 2022 10:49 AM			
Added By	sowmyasekar2306@gmail.com			
Connection Status	Disconnected			

## Generation of random values in IBM Watson

The screenshot shows the IBM Watson IoT Platform dashboard with the 'Recent Events' tab selected. The main content area displays a table with the following columns: Event, Value, Format, and Last Received. The table contains four rows of data, each representing a sensor reading. Below the table, a detailed view of the device is shown, including fields for Device ID, Device Type, Date Added, Added By, and Connection Status.

Event	Value	Format	Last Received
IoT Sensor	{"gasconcentration":100,"humidity":109,"tempe...	json	a few seconds ago
IoT Sensor	{"gasconcentration":94,"humidity":104,"temper...	json	a few seconds ago
IoT Sensor	{"gasconcentration":104,"humidity":108,"tempe...	json	a few seconds ago
IoT Sensor	{"gasconcentration":98,"humidity":109,"temper...	json	a few seconds ago