

PROJECT DEVELOPMENT PHASE

SPRINT 1

Date	16 November 2022
Team ID	PNT2022TMID13542
Project name	Hazardous area monitoring for industrial power plants by IoT
Maximum marks	2 marks

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 = "67yqBw"
deviceType = "Nodemcu"
deviceId = "123"
authMethod ="token"
authToken = "12345678"
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    elif status == "lightoff":
```

```

print ("led 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(90,110)

        Humidity =random.randint(90,110)

        Temperature = random.randint(90,110)

        data = {'gasconcentration' :gasconcentration,'Humidity' : Humidity,'Temperature' :Temperature}

    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")

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

        if not success:

            print("Not connected to IoTF")

            time.sleep(10)

deviceCli.commandCallback=myCommandCallback

deviceCli.disconnect()

```

```

ibmpython.py - C:\Users\USER\Desktop\bm\ibmpython.py (3.7.0)
File Edit Format Run Options Window Help
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
organization = "127fmq"
deviceType = "12345"
deviceId = "123456"
authMethod = "token"
authToken = "123456789"
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    if status=="lighton":
        print ("led is on")
    elif status == "lightoff":
        print ("led 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(90,110)
    humidity = random.randint(90,110)
    temperature = random.randint(90,110)
    data = {'gasconcentration' : gasconcentration, 'Humidity' : Humidity, 'Temperature' : Temperature}
    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")
    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPub
    if not success:
        print("Error connected to IoT")
    time.sleep(10)
deviceCli.commandCallback=myCommandCallback
deviceCli.disconnect()

```

Code runs successfully and random output values are generated

Creation of IBM cloud

Untitled

For you

- Build**
Explore IBM Cloud with this selection of easy starter tutorials and services.
- Get Started with Watson Studio**
Get started with using AI and Cloud Object Storage in 15 minutes.
- Get started with Watson Discovery**
Get up to speed on Watson Discovery with step-by-step tutorials, deep-dive videos, and complete examples of working code.
- Explore IBM Cloud Shell**
Try a command-driven approach for creating, developing, and deploying a web project.
- Get Started with the CLI**
Install the IBM Cloud™ developer tools, which include the latest IBM Cloud CLI, verify the installation, and configure the environment.
- Explore**
Try out to get Cloud complete

User access

News

Planned maintenance

IBM Cloud Satellite New Pricing

Procedure for the creation of IBM IOT watson

The screenshot shows the IBM Watson IoT Platform interface. At the top, there are three tabs: WhatsApp (2), Service Details - IBM Cloud, and IBM Watson IoT Platform. The current view is the 'IBM Watson IoT Platform' tab, which displays the 'Browse Devices' page. The header includes a user profile with the email vikrampraveenavp65@gmail.com and ID 67yq8w. Below the header, there are navigation links: Browse, Action, Device Types, and Interfaces, with 'Add Device' being the active link. The main content area is titled 'Browse Devices' and contains two buttons: 'All Devices' (selected) and 'Diagnose'. A descriptive text states: 'This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.' Below this is a table with columns: Device ID, Status, Device Type, Class ID, and Date Added. One row is present: Device ID 123, Status Disconnected, Device Type Nodemcu, Class ID Device, and Date Added 13 Nov 2022 1:04 PM. At the bottom of the table are pagination controls: 'Items per page' (set to 50), '1 of 1 page', and navigation arrows. The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray.

Device creation

The screenshot shows the 'Device Drilldown' page for device 123. The top navigation bar has tabs for Welcome to Project Delights, IBM, IBM-Project-8377-16589167, WhatsApp, and IBM Watson IoT Platform. The current view is the 'IBM Watson IoT Platform' tab. The header shows the user profile vikrampraveenavp65@gmail.com and ID 67yq8w. The left sidebar has a 'Back' button and a vertical menu with sections: Device Credentials, Connection Information, Recent Events, State, Device Information, Metadata, Diagnostics, and Connection Logs. The main content area is titled 'Device Drilldown - 123' and contains a 'Device Credentials' section. It says: 'You registered your device to the organization. Add these credentials to the device to connect it to the platform. After the device is connected, you can navigate to view connection and event details.' Below this are tables for Organization ID (67yq8w), Device Type (Nodemcu), Device ID (123), Authentication Method (use-token-auth), and Authentication Token (12345678). The bottom of the screen shows the Windows taskbar with multiple pinned icons and the system tray.

Generation of random values in IBM Watson

The screenshot shows the IBM Watson IoT Platform interface. At the top, there are several tabs: 'Welcome to Project! Delights', 'IBM', 'IBM-Project-8377-16589167', 'WhatsApp', and 'IBM Watson IoT Platform'. Below the tabs, the URL is 67yq8w.internetofthings.ibmcloud.com/dashboard/devices/browse. The main area is titled 'IBM Watson IoT Platform' and shows a device named '123' with status 'Disconnected', device type 'Nodemcu', class 'Device', and last updated on '13 Nov 2022 12:23 PM'. There are tabs for 'Identity', 'Device Information', 'Recent Events', 'State', and 'Logs'. Under 'Recent Events', it says 'The recent events listed show the live stream of data that is coming and going from this device.' A table lists events with columns: Event, Value, Format, and Last Received. The events listed are:

Event	Value	Format	Last Received
eventflow	[{"temperature":98}]	json	a few seconds ago
eventflow	[{"temperature":90}]	json	a few seconds ago
eventflow	[{"temperature":92}]	json	a few seconds ago
eventflow	[{"temperature":96}]	json	1 Simulation running
eventflow	[{"temperature":94}]	json	

At the bottom, there is a search bar 'Type here to search' and a taskbar with icons for File, Open, Save, Print, and others. The system tray shows the date and time as 13-11-2022 12:38 ENG.