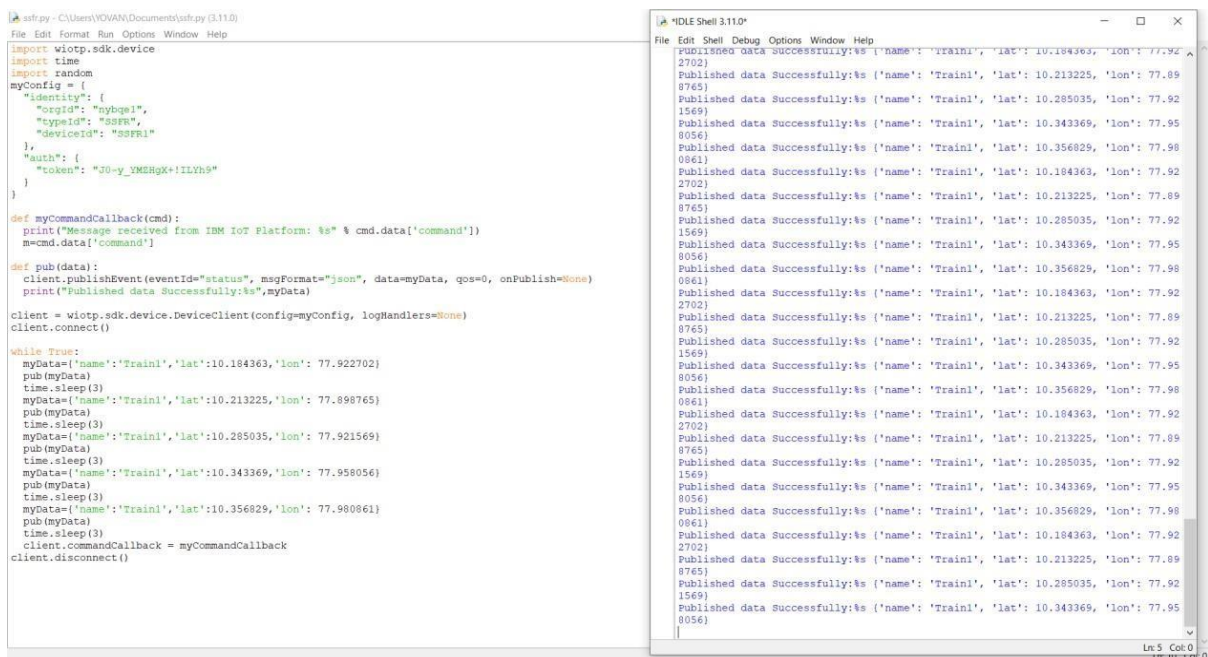


TESTING OF WEB UI

| | |
|--------------|---------------------------------------|
| Date | 10 November 2022 |
| Team ID | PNT2022TMID01176 |
| Project Name | Project – Smart Solution For Railways |

Location Tracking:

- The python code for detecting the location of the train is made to Run.
- The live status of the train is updated in the IBM Watson IoT Platform and it is further used by the node red application.
- The location is shown in the map via the Node red UI.



The image displays two side-by-side screenshots of a code editor and its output console.

The left screenshot shows a Python script in a file named `test.py`. The script imports `wiotp.sdk.device` and `time`, and uses `random` for generating IDs. It defines a `myConfig` dictionary with fields like `orgId`, `typeId`, `deviceId`, `auth`, and `token`. A `myCommandCallback` function is defined to handle incoming commands from the IBM IoT Platform. The main logic is in a `while True:` loop that generates random location data (name, lat, lon) and publishes it to the IoT Platform using `client.publishEvent`. It also includes a `client.disconnect()` call at the end of the loop.

The right screenshot shows the output of the script in a terminal window titled "IDLE Shell 3.11.0". The output consists of multiple lines of log messages, each indicating a successful publish event with the generated location data, such as: `Published data Successfully: { 'name': 'Train1', 'lat': 10.184363, 'lon': 77.922702 }`.

Device Simulator

Search by Device ID

| Device ID | Status | Device Type | Class ID | Date Added | Descriptive Location |
|-------------|--------------|-------------|----------|-------------------|----------------------|
| Jeeva_Yovan | Disconnected | Watson | Device | 28 Oct 2022 19:58 | |
| SSFR1 | Connected | SSFR | Device | 16 Nov 2022 10:32 | |

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 |
|--------|---|--------|-------------------|
| status | {"name":"Train1","lat":10.285035,"lon":77.9215... | json | a few seconds ago |
| status | {"name":"Train1","lat":10.213225,"lon":77.8987... | json | a few seconds ago |
| status | {"name":"Train1","lat":10.184363,"lon":77.9227... | json | a few seconds ago |
| status | {"name":"Train1","lat":10.356829,"lon":77.9808... | json | a few seconds ago |
| status | {"name":"Train1","lat":10.343369,"lon":77.9580... | json | a few seconds ago |

0 Simulations running

Node-RED interface showing a flow with an IBM IoT node connected to a debug node and a worldmap node. The debug console displays messages from the IoT node, including location data (name, lat, lon) and status (msg.payload). The messages are as follows:

```

16/11/2022, 11:06:53 am node: debug 1
iot-2/type/SSFRid/SSFR1/ev/status/rtm/json :
msg.payload : Object
  > { name: "Train1", lat: 10.184363, lon: 77.922702 }

16/11/2022, 11:06:55 am node: debug 1
iot-2/type/SSFRid/SSFR1/ev/status/rtm/json :
msg.payload : Object
  > { name: "Train1", lat: 10.213225, lon: 77.898765 }

16/11/2022, 11:06:57 am node: debug 1
iot-2/type/SSFRid/SSFR1/ev/status/rtm/json :
msg.payload : Object
  > { name: "Train1", lat: 10.285035, lon: 77.921569 }

16/11/2022, 11:07:00 am node: debug 1
iot-2/type/SSFRid/SSFR1/ev/status/rtm/json :
msg.payload : Object
  > { name: "Train1", lat: 10.343369, lon: 77.958056 }

16/11/2022, 11:07:03 am node: debug 1
iot-2/type/SSFRid/SSFR1/ev/status/rtm/json :
msg.payload : Object
  > { name: "Train1", lat: 10.356829, lon: 77.980861 }

16/11/2022, 11:07:05 am node: debug 1
iot-2/type/SSFRid/SSFR1/ev/status/rtm/json :
msg.payload : Object
  > { name: "Train1", lat: 10.184363, lon: 77.922702 }

```

Live Status

Tracking