

Smart Farmer - IoT Enabled Smart Farming Application

SPRINT - 2

TEAM ID	PNT2022TMID04018
DATE	14 TH NOVEMBER 2022

IBM Watson IoT platform, Workflows for IoT scenarios using Node-red:

Building Project:

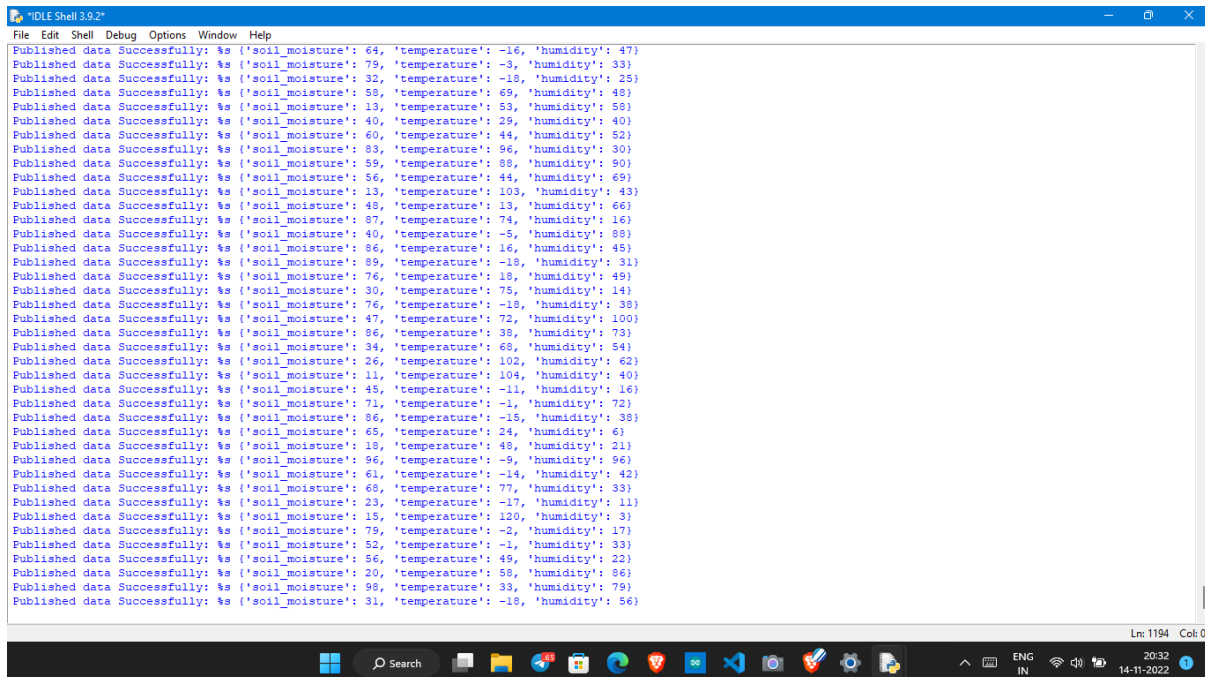
Connecting IOT Simulator to IBM Watson IOTPlatform

Give the credentials of your device in IBM Watson

"orgId": "04gt4e",

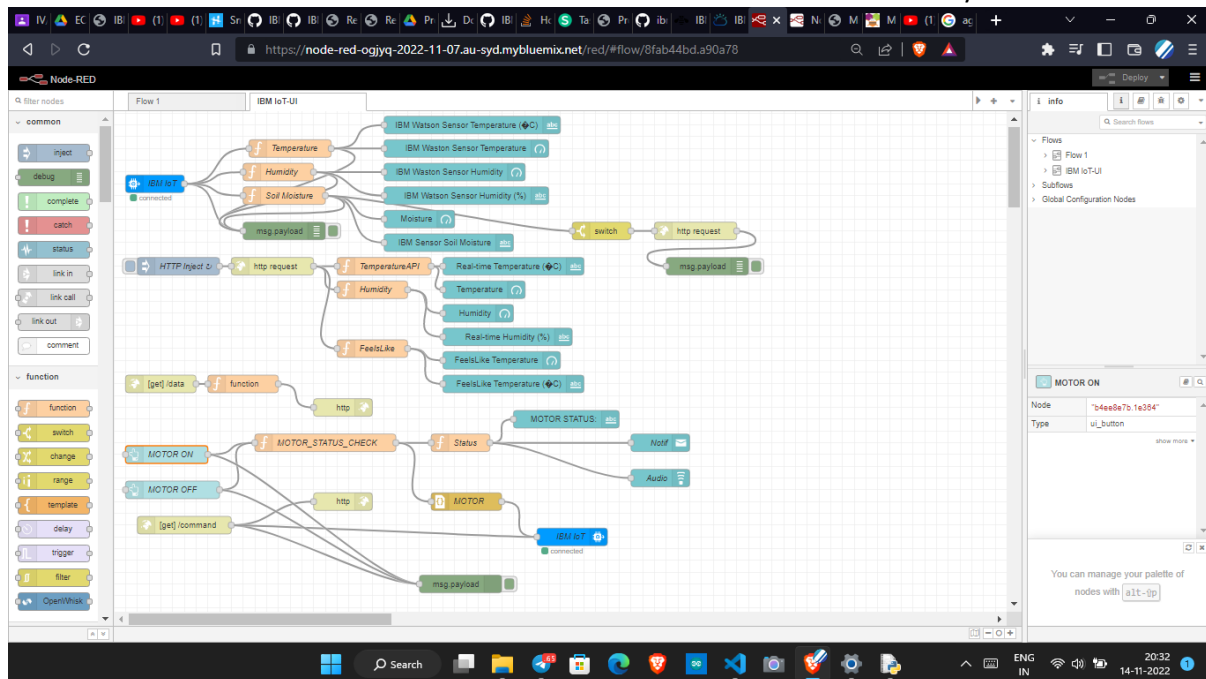
"typeId": "NodeMCU",

"deviceId": "12345"



```
Published data Successfully: {"soil_moisture": 64, "temperature": -16, "humidity": 47}
Published data Successfully: {"soil_moisture": 79, "temperature": -3, "humidity": 33}
Published data Successfully: {"soil_moisture": 32, "temperature": -18, "humidity": 25}
Published data Successfully: {"soil_moisture": 58, "temperature": 69, "humidity": 48}
Published data Successfully: {"soil_moisture": 13, "temperature": 53, "humidity": 58}
Published data Successfully: {"soil_moisture": 40, "temperature": 29, "humidity": 40}
Published data Successfully: {"soil_moisture": 60, "temperature": 44, "humidity": 52}
Published data Successfully: {"soil_moisture": 83, "temperature": 96, "humidity": 30}
Published data Successfully: {"soil_moisture": 59, "temperature": 88, "humidity": 90}
Published data Successfully: {"soil_moisture": 56, "temperature": 44, "humidity": 69}
Published data Successfully: {"soil_moisture": 13, "temperature": 103, "humidity": 43}
Published data Successfully: {"soil_moisture": 48, "temperature": 13, "humidity": 66}
Published data Successfully: {"soil_moisture": 87, "temperature": 74, "humidity": 16}
Published data Successfully: {"soil_moisture": 40, "temperature": -5, "humidity": 88}
Published data Successfully: {"soil_moisture": 86, "temperature": 16, "humidity": 45}
Published data Successfully: {"soil_moisture": 89, "temperature": -18, "humidity": 31}
Published data Successfully: {"soil_moisture": 76, "temperature": 18, "humidity": 49}
Published data Successfully: {"soil_moisture": 30, "temperature": 75, "humidity": 14}
Published data Successfully: {"soil_moisture": 76, "temperature": -18, "humidity": 38}
Published data Successfully: {"soil_moisture": 47, "temperature": 72, "humidity": 100}
Published data Successfully: {"soil_moisture": 86, "temperature": 38, "humidity": 73}
Published data Successfully: {"soil_moisture": 34, "temperature": 68, "humidity": 54}
Published data Successfully: {"soil_moisture": 26, "temperature": 102, "humidity": 62}
Published data Successfully: {"soil_moisture": 11, "temperature": 104, "humidity": 40}
Published data Successfully: {"soil_moisture": 45, "temperature": -11, "humidity": 16}
Published data Successfully: {"soil_moisture": 71, "temperature": -1, "humidity": 72}
Published data Successfully: {"soil_moisture": 86, "temperature": -15, "humidity": 38}
Published data Successfully: {"soil_moisture": 65, "temperature": 24, "humidity": 6}
Published data Successfully: {"soil_moisture": 18, "temperature": 48, "humidity": 21}
Published data Successfully: {"soil_moisture": 96, "temperature": -9, "humidity": 96}
Published data Successfully: {"soil_moisture": 61, "temperature": -14, "humidity": 42}
Published data Successfully: {"soil_moisture": 68, "temperature": 77, "humidity": 33}
Published data Successfully: {"soil_moisture": 23, "temperature": -17, "humidity": 11}
Published data Successfully: {"soil_moisture": 15, "temperature": 120, "humidity": 3}
Published data Successfully: {"soil_moisture": 79, "temperature": -2, "humidity": 17}
Published data Successfully: {"soil_moisture": 52, "temperature": -1, "humidity": 33}
Published data Successfully: {"soil_moisture": 56, "temperature": 49, "humidity": 22}
Published data Successfully: {"soil_moisture": 20, "temperature": 58, "humidity": 86}
Published data Successfully: {"soil_moisture": 98, "temperature": 33, "humidity": 79}
Published data Successfully: {"soil_moisture": 31, "temperature": -18, "humidity": 56}
```

The node IBM IoT App In is added to Node-Red workflow. Then the appropriate device credentials obtained earlier are entered into the node to connect and fetch device telemetry to Node-Red



The Node-Red also receive data from the Open Weather API by HTTP GET request. An inject trigger is added to perform HTTP request for every certain interval. HTTP request node is configured with URL we saved before.

The screenshot shows the IBM Watson IoT Platform dashboard. The 'Browse' tab is selected, displaying a table of device data. The table has columns for 'Event', 'Value', 'Format', and 'Last Received'. The data shows status updates for a device with ID '04gt4e'.

Event	Value	Format	Last Received
status	{"soil_moisture":66,"temperature":82,"humidity"...	json	a few seconds ago
status	{"soil_moisture":62,"temperature":-20,"humidity"...	json	a few seconds ago
status	{"soil_moisture":29,"temperature":43,"humidity"...	json	a few seconds ago
status	{"soil_moisture":80,"temperature":85,"humidity"...	json	a few seconds ago
status	{"soil_moisture":88,"temperature":105,"humidit..."	json	a few seconds ago

Below the table, there is a summary row for the device '23456' (ID: 04gt4e), which is 'Disconnected' and of type 'esp32'. It shows the device was last seen on '6 Nov 2022 15:56'. The dashboard also indicates '0 Simulations running'.

FLOW CHART:

