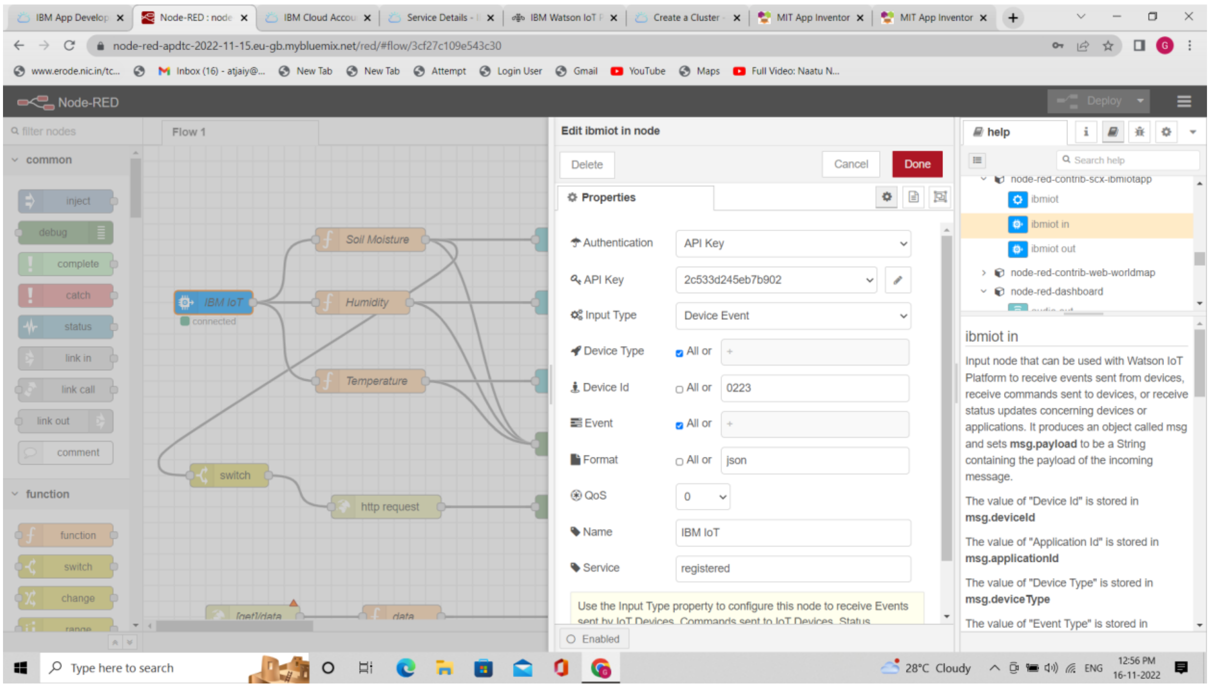
# SPRINT-3

| | |
|---|---|
| Team ID | PNT2022TMID19466 |
| Project Title | Smart Farmer – IoT enabled Smart Farming Application |
| Date | 16-11-2022 |

## Smart Farming Application using NODE RED SERVICE:

### IBM IOT IN NODE:



node-red-apdtc-2022-11-15.eu-gb.mybluemix.net/red/#flow/3cf27c109e543c30

Edit ibmiot in node

Delete  Cancel  Done

Properties

Authentication: API Key

API Key: 2c533d245eb7b902

Input Type: Device Event

Device Type: ☑ All or +

Device Id: ☐ All or 0223

Event: ☑ All or +

Format: ☐ All or json

QoS: 0

Name: IBM IoT

Service: registered

Use the Input Type property to configure this node to receive Events sent by IoT Devices, Commands sent to IoT Devices, Status

○ Enabled

ibmiot in

Input node that can be used with Watson IoT Platform to receive events sent from devices, receive commands sent to devices, or receive status updates concerning devices or applications. It produces an object called msg and sets **msg.payload** to be a String containing the payload of the incoming message.

The value of "Device Id" is stored in **msg.deviceId**

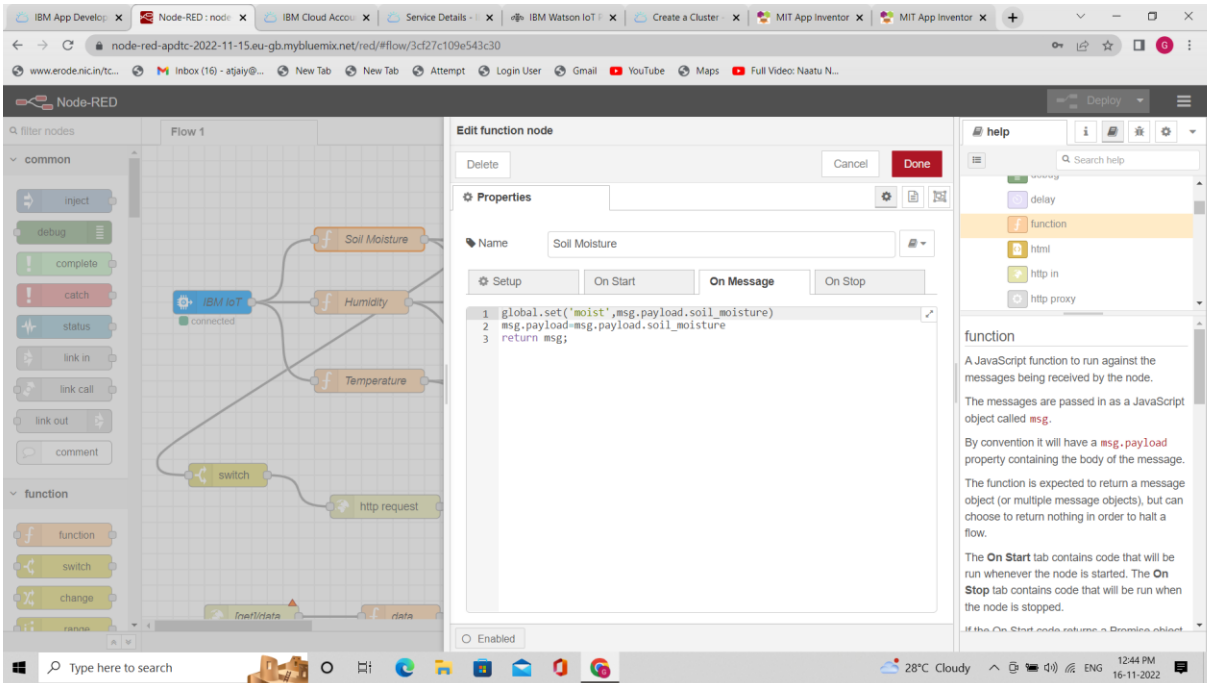
The value of "Application Id" is stored in **msg.applicationId**

The value of "Device Type" is stored in **msg.deviceType**

The value of "Event Type" is stored in

28°C Cloudy 12:56 PM 16-11-2022

# SOIL MOISTURE FUNCTION:



Name: Soil Moisture

Setup | On Start | On Message | On Stop

```
global.set('moist',msg.payload.soil_moisture)
msg.payload=msg.payload.soil_moisture
return msg;
```

function

A JavaScript function to run against the messages being received by the node.

The messages are passed in as a JavaScript object called `msg`.

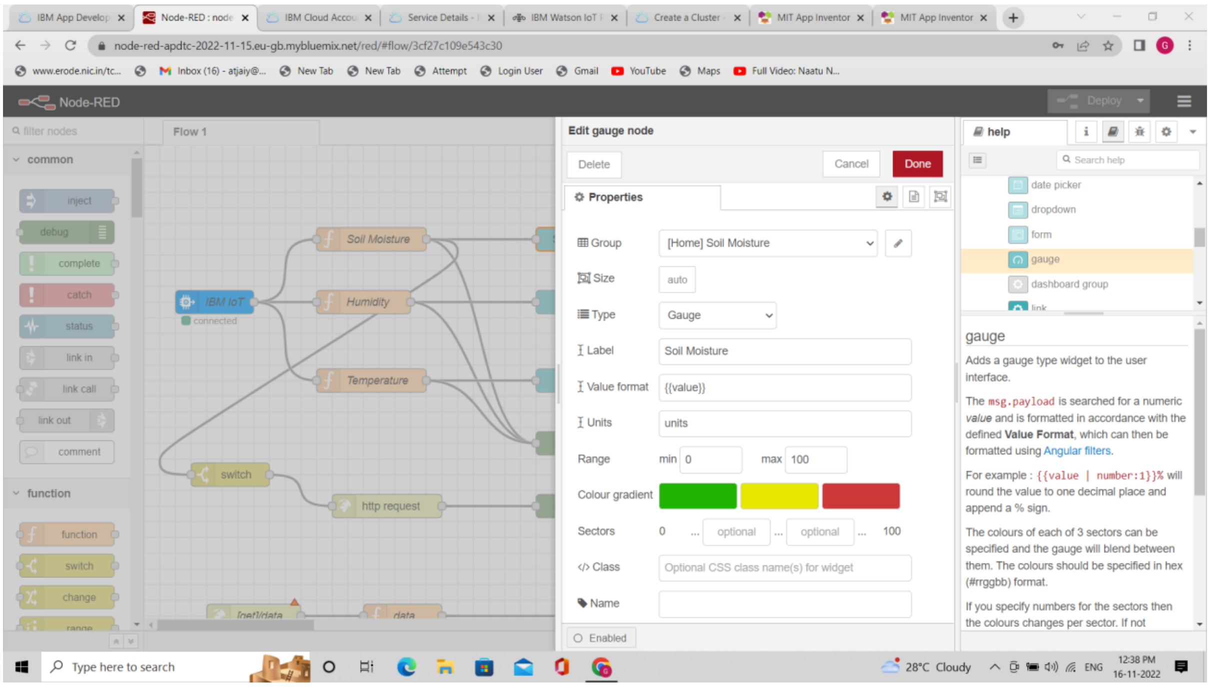
By convention it will have a `msg.payload` property containing the body of the message.

The function is expected to return a message object (or multiple message objects), but can choose to return nothing in order to halt a flow.

The **On Start** tab contains code that will be run whenever the node is started. The **On Stop** tab contains code that will be run when the node is stopped.

If the On Start code returns a Promise object

# SOIL MOISTURE GAUGE:



Group: [Home] Soil Moisture

Size: auto

Type: Gauge

Label: Soil Moisture

Value format: {{value}}

Units: units

Range: min 0 max 100

Colour gradient: 🟩 🟨 🟥

Sectors: 0 ... optional ... optional ... 100

Class: Optional CSS class name(s) for widget

Name:

gauge

Adds a gauge type widget to the user interface.

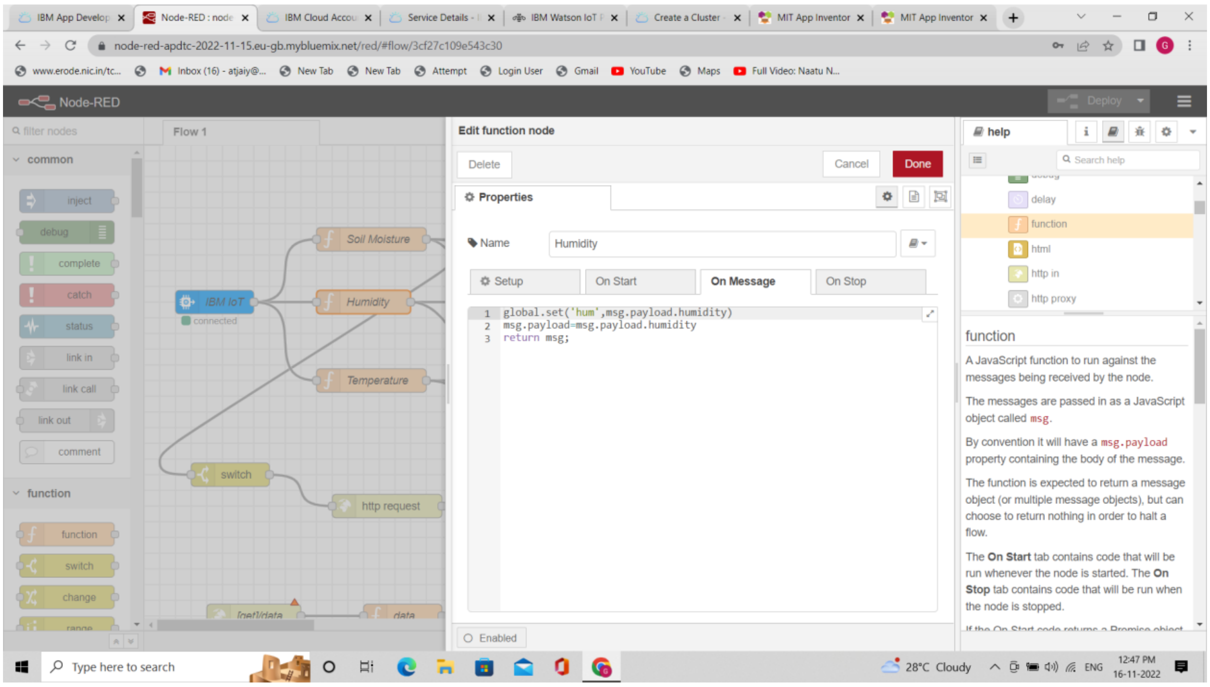
The `msg.payload` is searched for a numeric *value* and is formatted in accordance with the defined **Value Format**, which can then be formatted using Angular filters.

For example : `{{value | number:1}}%` will round the value to one decimal place and append a % sign.

The colours of each of 3 sectors can be specified and the gauge will blend between them. The colours should be specified in hex (#rrggbb) format.

If you specify numbers for the sectors then the colours changes per sector. If not

# HUMIDITY FUNCTION:



Edit function node

Delete Cancel Done

Properties

Name: Humidity

Setup | On Start | On Message | On Stop

```
global.set('hum',msg.payload.humidity)
msg.payload=msg.payload.humidity
return msg;
```

○ Enabled

help

function

A JavaScript function to run against the messages being received by the node.

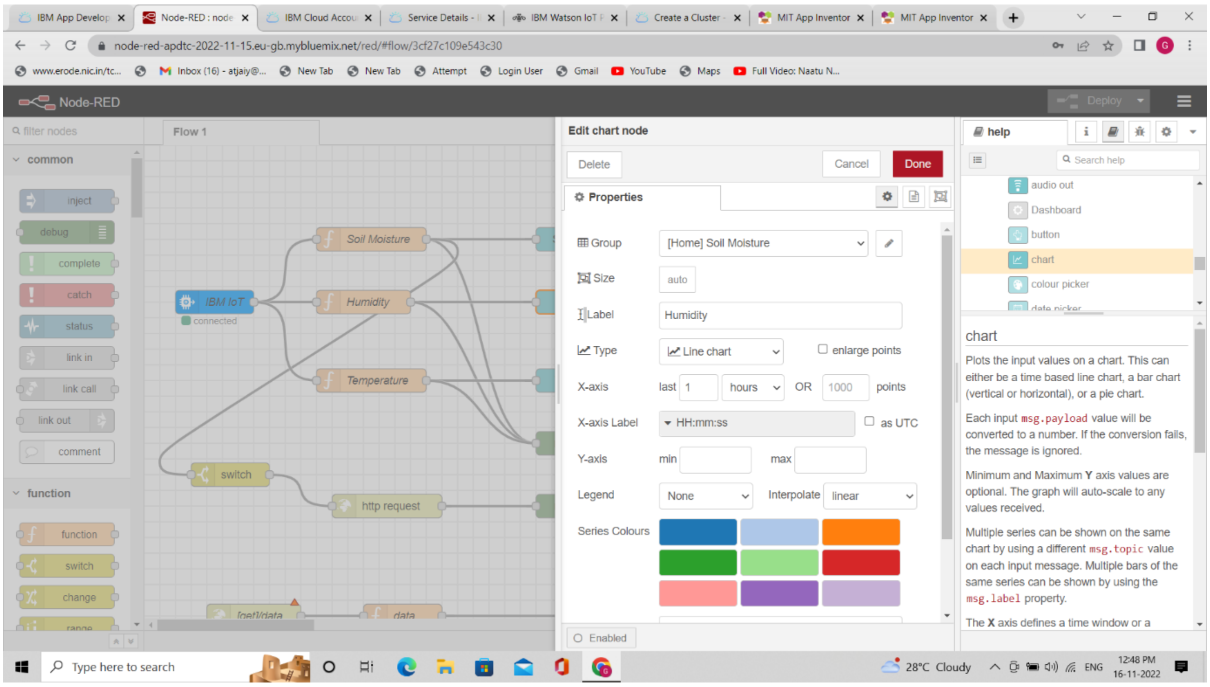
The messages are passed in as a JavaScript object called `msg`.

By convention it will have a `msg.payload` property containing the body of the message.

The function is expected to return a message object (or multiple message objects), but can choose to return nothing in order to halt a flow.

The **On Start** tab contains code that will be run whenever the node is started. The **On Stop** tab contains code that will be run when the node is stopped.

# HUMIDITY CHART:



Edit chart node

Delete Cancel Done

Properties

Group: [Home] Soil Moisture

Size: auto

Label: Humidity

Type: Line chart □ enlarge points

X-axis: last 1 hours OR 1000 points

X-axis Label: HH:mm:ss □ as UTC

Y-axis: min max

Legend: None Interpolate: linear

Series Colours

○ Enabled

help

chart

Plots the input values on a chart. This can either be a time based line chart, a bar chart (vertical or horizontal), or a pie chart.

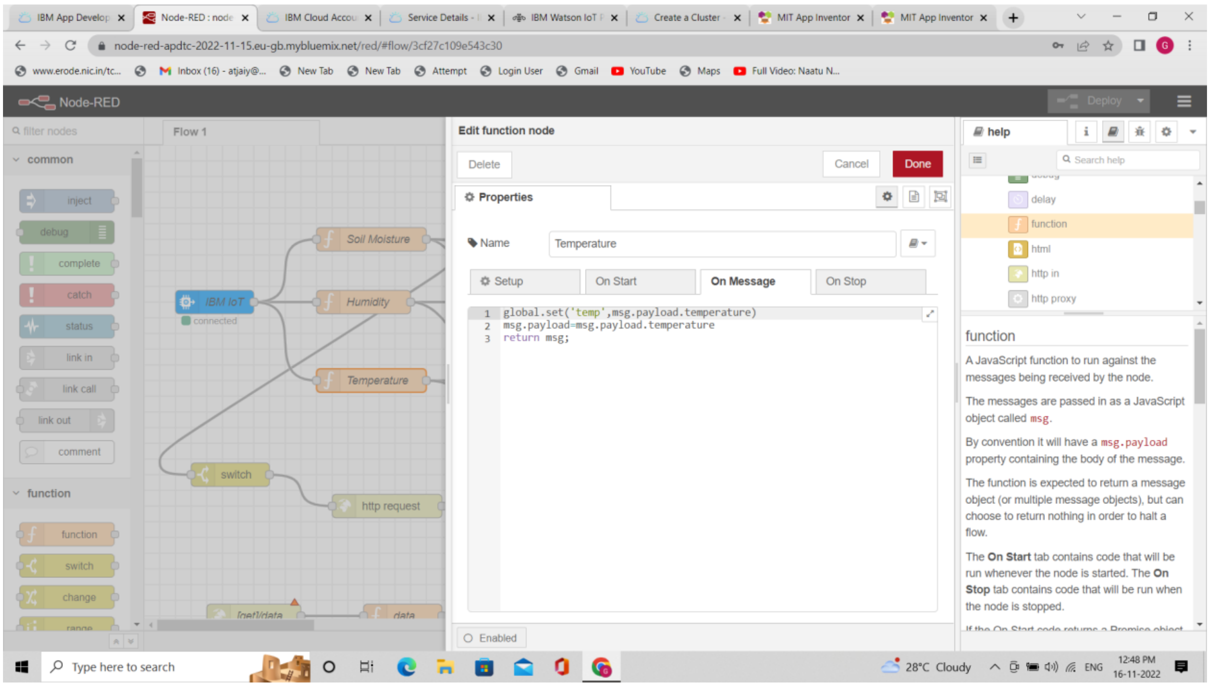
Each input `msg.payload` value will be converted to a number. If the conversion fails, the message is ignored.

Minimum and Maximum **Y** axis values are optional. The graph will auto-scale to any values received.

Multiple series can be shown on the same chart by using a different `msg.topic` value on each input message. Multiple bars of the same series can be shown by using the `msg.label` property.

The **X** axis defines a time window or a

# TEMPERATURE FUNCTION:



Edit function node

Delete | Cancel | Done

Properties

Name: Temperature

Setup | On Start | **On Message** | On Stop

```
global.set('temp',msg.payload.temperature)
msg.payload=msg.payload.temperature
return msg;
```

○ Enabled

help

function

A JavaScript function to run against the messages being received by the node.

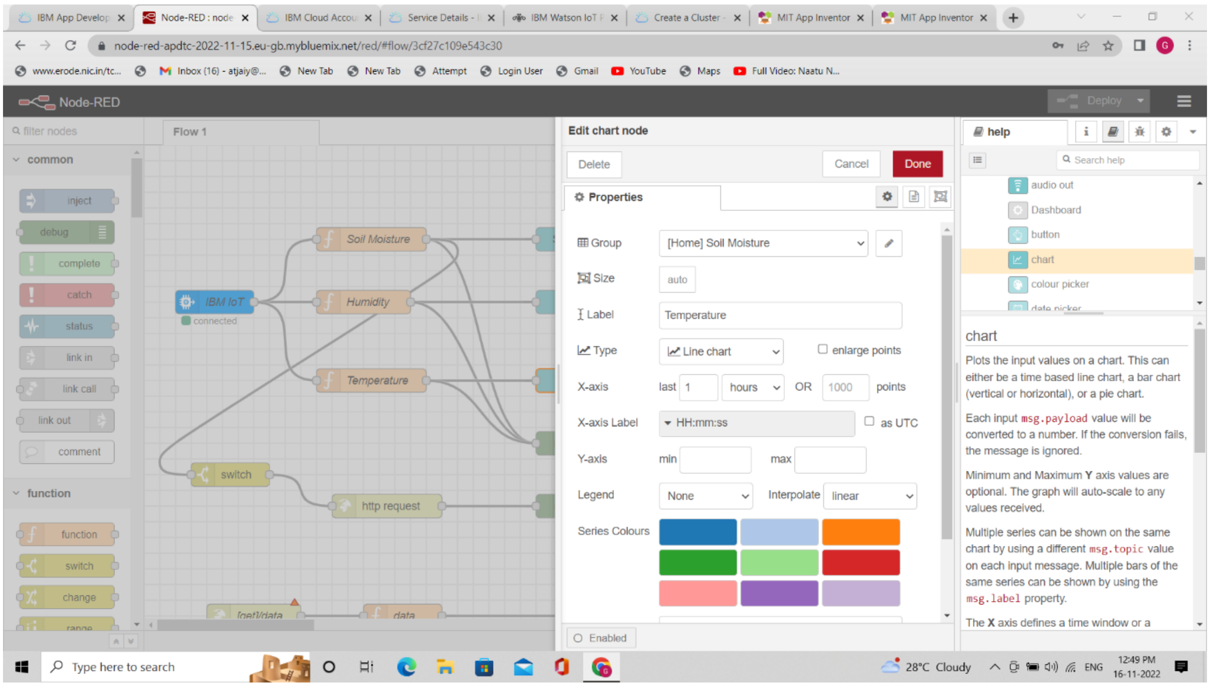
The messages are passed in as a JavaScript object called `msg`.

By convention it will have a `msg.payload` property containing the body of the message.

The function is expected to return a message object (or multiple message objects), but can choose to return nothing in order to halt a flow.

The **On Start** tab contains code that will be run whenever the node is started. The **On Stop** tab contains code that will be run when the node is stopped.

# TEMPERATURE CHART:



Edit chart node

Delete | Cancel | Done

Properties

Group: [Home] Soil Moisture

Size: auto

Label: Temperature

Type: Line chart □ enlarge points

X-axis: last 1 hours OR 1000 points

X-axis Label: HH:mm:ss □ as UTC

Y-axis: min max

Legend: None Interpolate: linear

Series Colours

○ Enabled

help

chart

Plots the input values on a chart. This can either be a time based line chart, a bar chart (vertical or horizontal), or a pie chart.

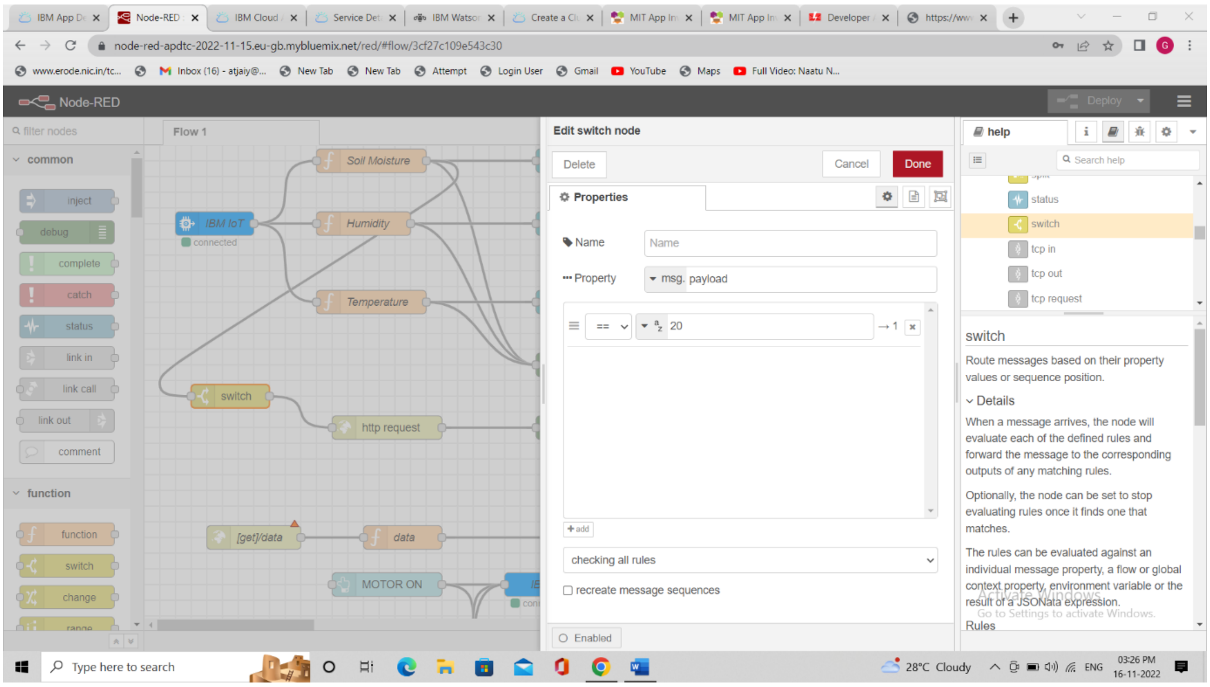
Each input `msg.payload` value will be converted to a number. If the conversion fails, the message is ignored.

Minimum and Maximum **Y** axis values are optional. The graph will auto-scale to any values received.

Multiple series can be shown on the same chart by using a different `msg.topic` value on each input message. Multiple bars of the same series can be shown by using the `msg.label` property.

The **X** axis defines a time window or a

## SWITCH OPTION:



Edit switch node

Name: Name

Property: msg. payload

== 20 → 1

checking all rules

☐ recreate message sequences

Enabled

switch

Route messages based on their property values or sequence position.

Details

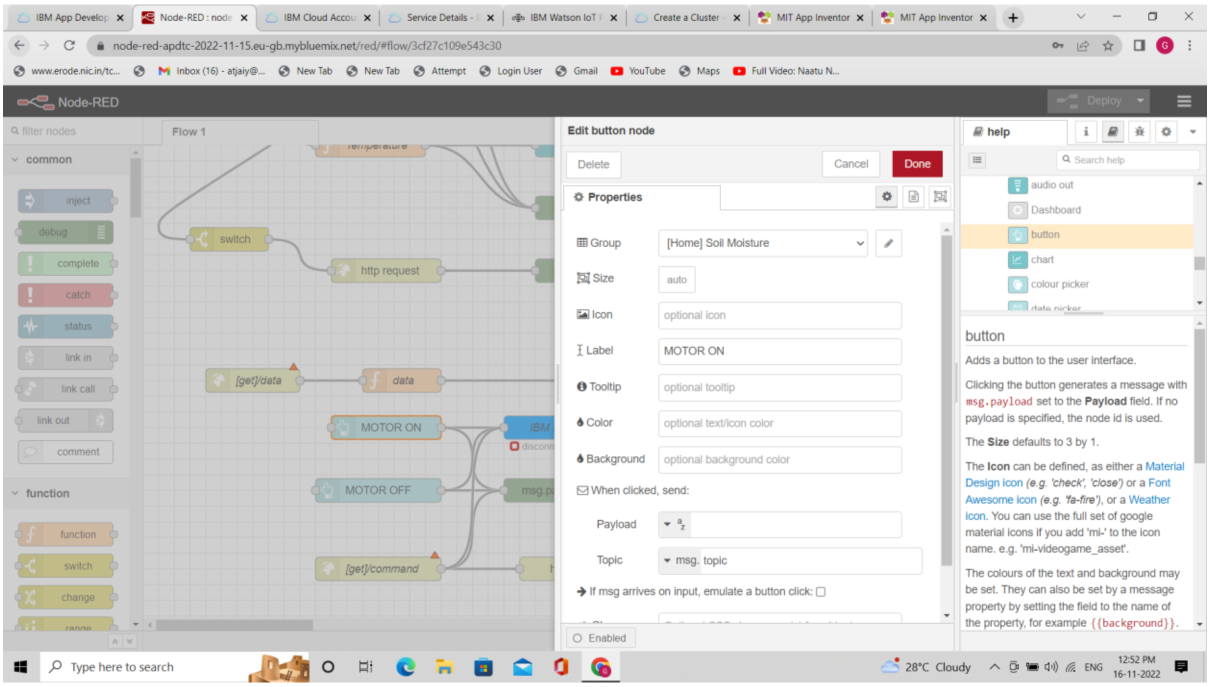
When a message arrives, the node will evaluate each of the defined rules and forward the message to the corresponding outputs of any matching rules.

Optionally, the node can be set to stop evaluating rules once it finds one that matches.

The rules can be evaluated against an individual message property, a flow or global context property, environment variable or the result of a JSONata expression.

Rules

## MOTOR ON BUTTON:



Edit button node

Group: [Home] Soil Moisture

Size: auto

Icon: optional icon

Label: MOTOR ON

Tooltip: optional tooltip

Color: optional text/icon color

Background: optional background color

When clicked, send:

Payload:

Topic: msg. topic

If msg arrives on input, emulate a button click: ☐

Enabled

button

Adds a button to the user interface.

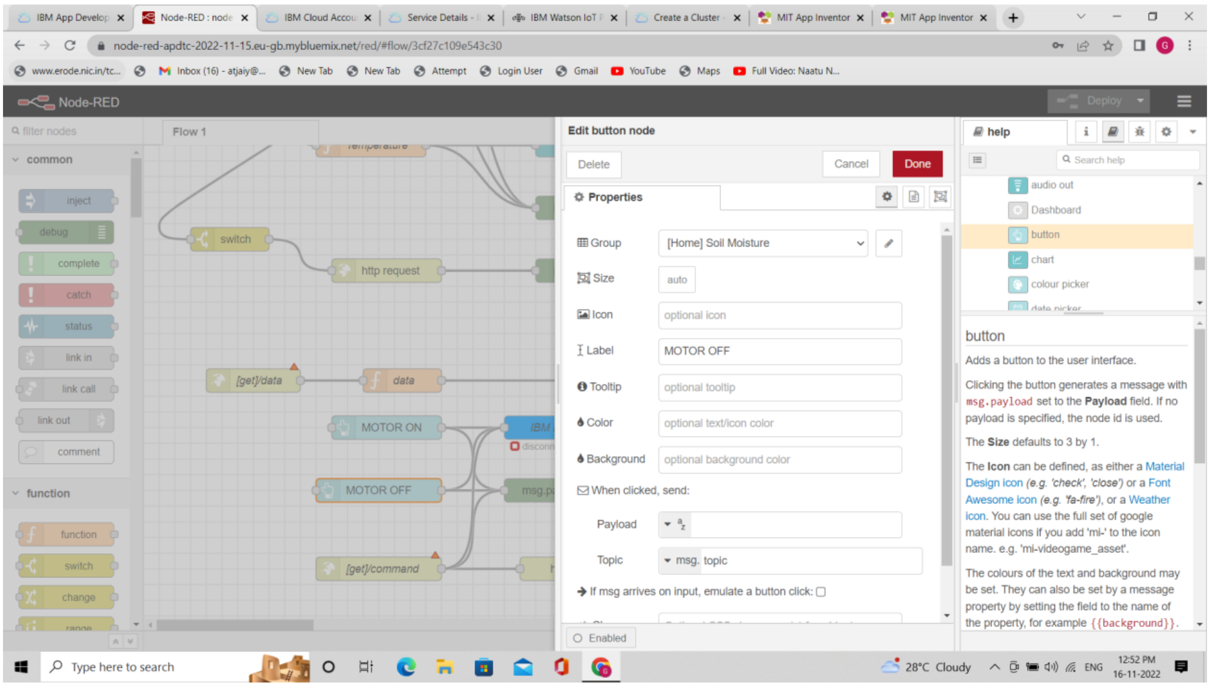
Clicking the button generates a message with `msg.payload` set to the **Payload** field. If no payload is specified, the node id is used.

The **Size** defaults to 3 by 1.

The **Icon** can be defined, as either a Material Design icon *(e.g. 'check', 'close')* or a Font Awesome icon *(e.g. 'fa-fire')*, or a Weather icon. You can use the full set of google material icons if you add 'mi-' to the icon name. e.g. 'mi-videogame_asset'.

The colours of the text and background may be set. They can also be set by a message property by setting the field to the name of the property, for example `{{background}}`.

## MOTOR OFF BUTTON:



node-red-apdtc-2022-11-15.eu-gb.mybluemix.net/red/#flow/3cf27c109e543c30

Edit button node

Properties

Group: [Home] Soil Moisture

Size: auto

Icon: optional icon

Label: MOTOR OFF

Tooltip: optional tooltip

Color: optional text/icon color

Background: optional background color

When clicked, send:

Payload:

Topic: msg. topic

If msg arrives on input, emulate a button click: □

○ Enabled

button

Adds a button to the user interface.

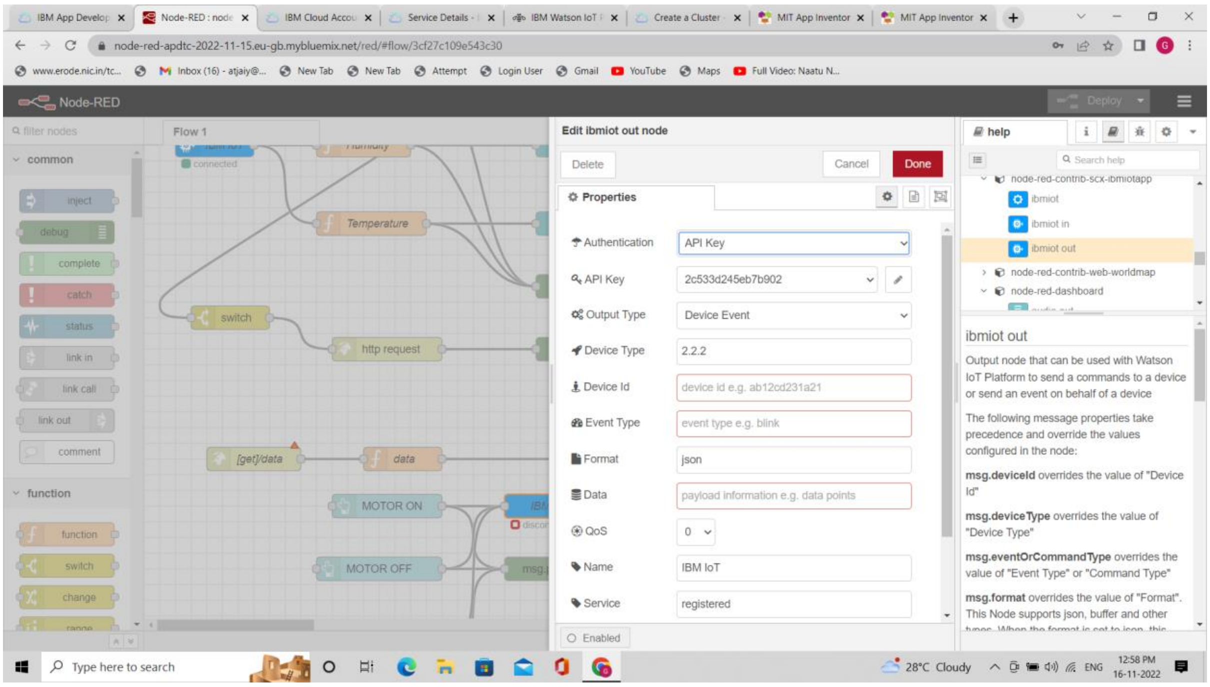
Clicking the button generates a message with `msg.payload` set to the **Payload** field. If no payload is specified, the node id is used.

The **Size** defaults to 3 by 1.

The **Icon** can be defined, as either a Material Design icon *(e.g. 'check', 'close')* or a Font Awesome icon *(e.g. 'fa-fire')*, or a Weather icon. You can use the full set of google material icons if you add 'mi-' to the icon name. e.g. 'mi-videogame_asset'.

The colours of the text and background may be set. They can also be set by a message property by setting the field to the name of the property, for example `{{background}}`.

## IBM IOT OUT NODE:



node-red-apdtc-2022-11-15.eu-gb.mybluemix.net/red/#flow/3cf27c109e543c30

Edit ibmiot out node

Properties

Authentication: API Key

API Key: 2c533d245eb7b902

Output Type: Device Event

Device Type: 2.2.2

Device Id: device id e.g. ab12cd231a21

Event Type: event type e.g. blink

Format: json

Data: payload information e.g. data points

QoS: 0

Name: IBM IoT

Service: registered

○ Enabled

ibmiot out

Output node that can be used with Watson IoT Platform to send a commands to a device or send an event on behalf of a device

The following message properties take precedence and override the values configured in the node:

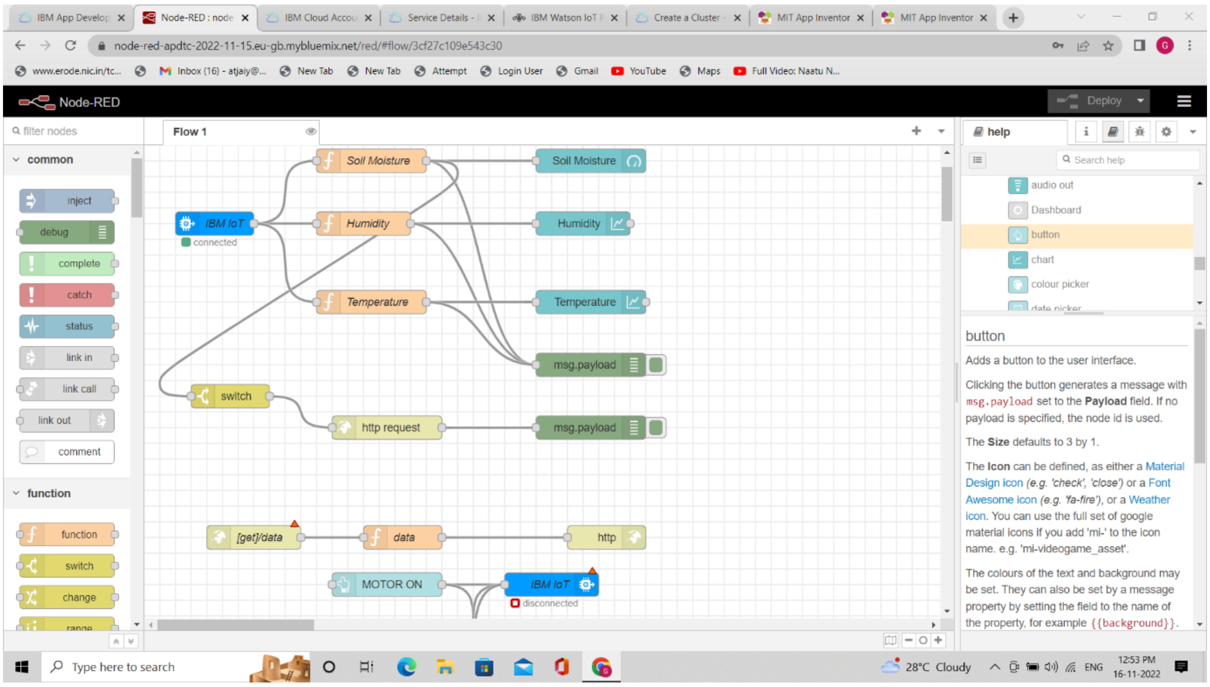
**msg.deviceId** overrides the value of "Device Id"

**msg.deviceType** overrides the value of "Device Type"

**msg.eventOrCommandType** overrides the value of "Event Type" or "Command Type"

**msg.format** overrides the value of "Format". This Node supports json, buffer and other types. When the format is set to json, this

# NODE-RED:



button

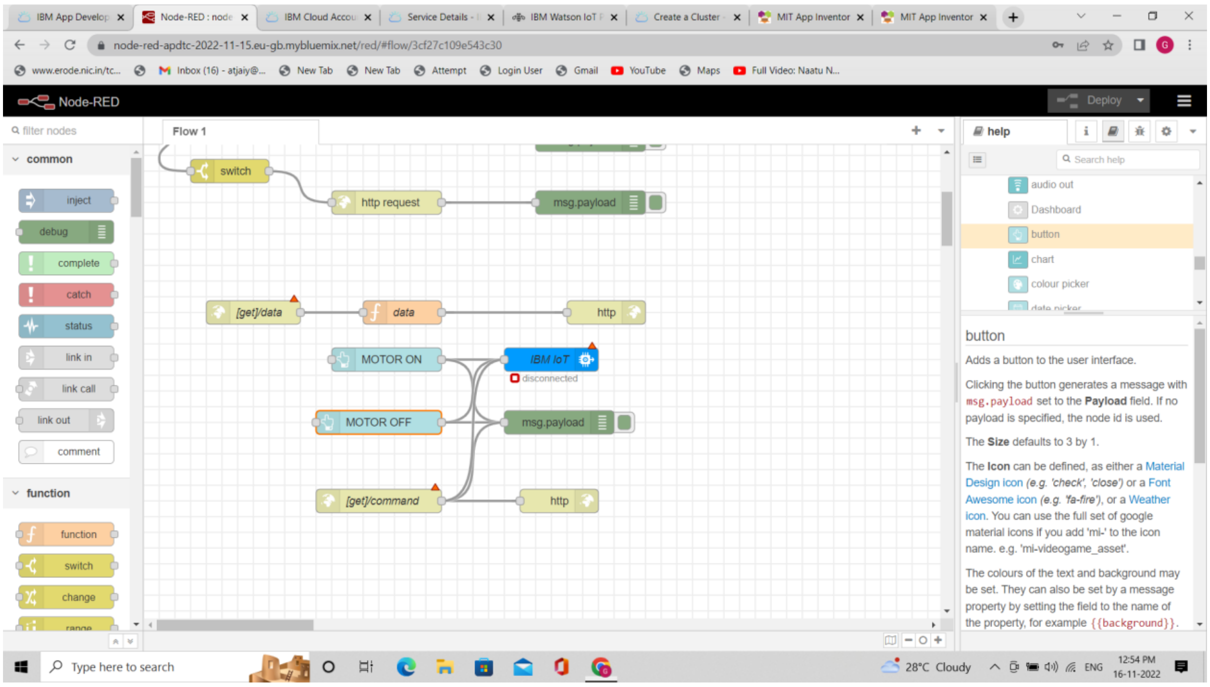
Adds a button to the user interface.

Clicking the button generates a message with msg.payload set to the **Payload** field. If no payload is specified, the node id is used.

The **Size** defaults to 3 by 1.

The **Icon** can be defined, as either a Material Design icon *(e.g. 'check', 'close')* or a Font Awesome icon *(e.g. 'fa-fire')*, or a Weather icon. You can use the full set of google material icons if you add 'mi-' to the icon name. e.g. 'mi-videogame_asset'.

The colours of the text and background may be set. They can also be set by a message property by setting the field to the name of the property, for example {{background}}.



button

Adds a button to the user interface.

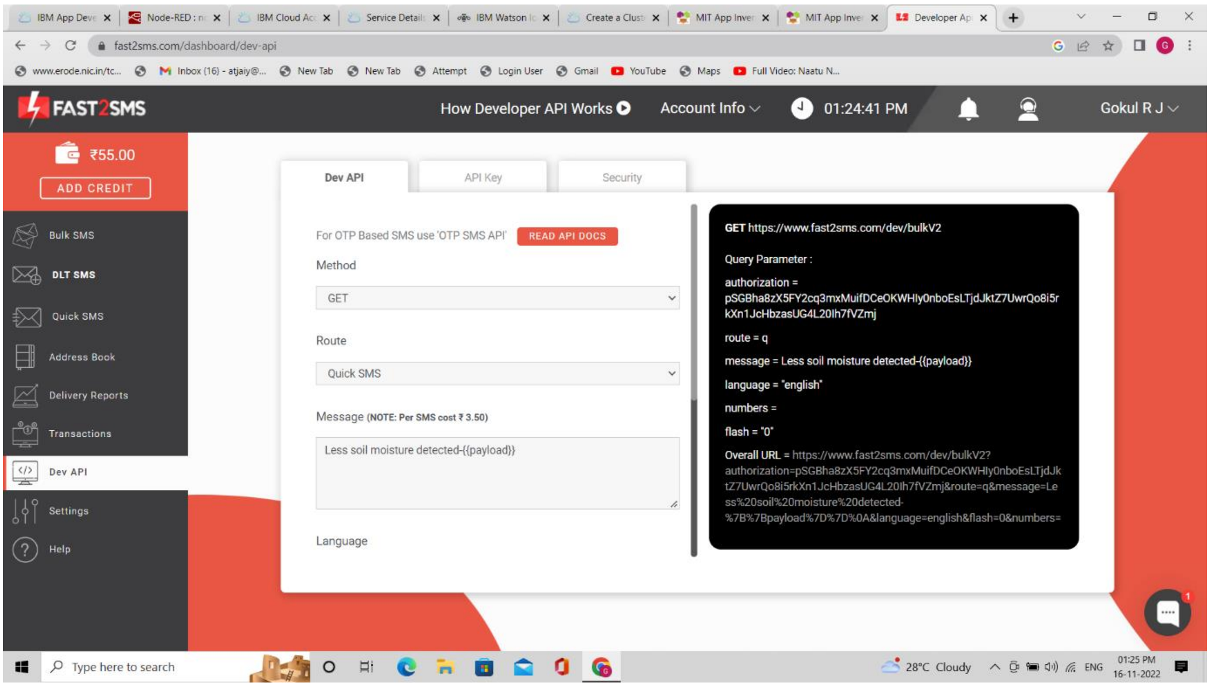
Clicking the button generates a message with msg.payload set to the **Payload** field. If no payload is specified, the node id is used.

The **Size** defaults to 3 by 1.

The **Icon** can be defined, as either a Material Design icon *(e.g. 'check', 'close')* or a Font Awesome icon *(e.g. 'fa-fire')*, or a Weather icon. You can use the full set of google material icons if you add 'mi-' to the icon name. e.g. 'mi-videogame_asset'.

The colours of the text and background may be set. They can also be set by a message property by setting the field to the name of the property, for example {{background}}.

# BULK SMS:



For OTP Based SMS use 'OTP SMS API'

Method: GET

Route: Quick SMS

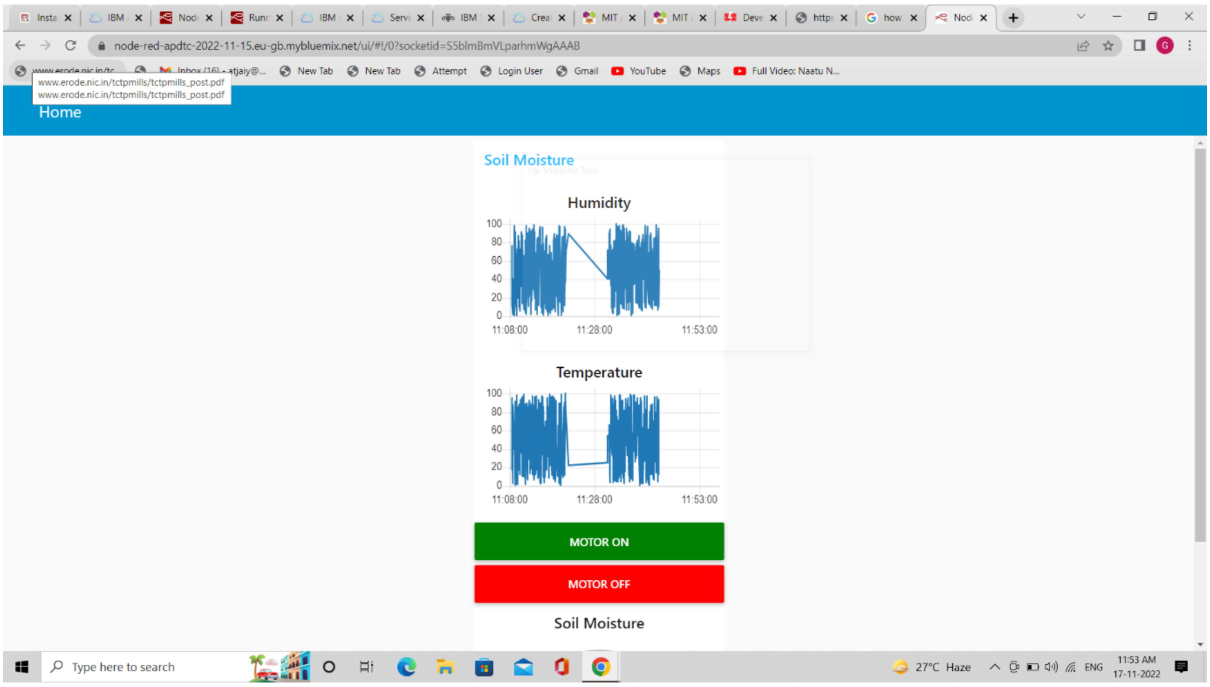
Message (NOTE: Per SMS cost ₹ 3.50)

Less soil moisture detected-{{payload}}

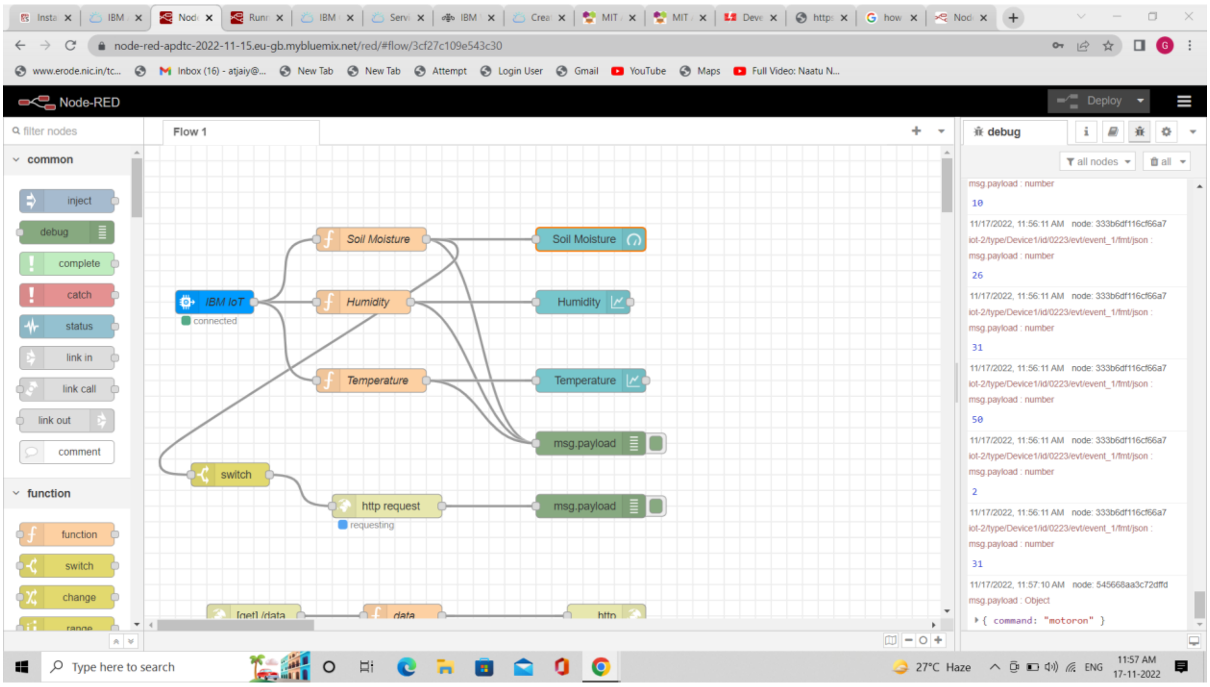
Language

```
GET https://www.fast2sms.com/dev/bulkV2

Query Parameter :

authorization =
pSGBha8zX5FY2cq3mxMuifDCeOKWHIy0nboEsLTjdJktZ7UwrQo8i5r
kXn1JcHbzasUG4L20Ih7fVZmj

route = q

message = Less soil moisture detected-{{payload}}

language = "english"

numbers =

flash = "0"

Overall URL = https://www.fast2sms.com/dev/bulkV2?
authorization=pSGBha8zX5FY2cq3mxMuifDCeOKWHIy0nboEsLTjdJk
tZ7UwrQo8i5rkXn1JcHbzasUG4L20Ih7fVZmj&route=q&message=Le
ss%20soil%20moisture%20detected-
%7B%7Bpayload%7D%7D%0A&language=english&flash=0&numbers=
```

# NODE RED DASHBOARD HOME:



# NODE RED DEBUG CONSOLE:



```
msg.payload : number
 10
11/17/2022, 11:56:11 AM   node: 333b6df116cf66a7
iot-2/type/Device1/id/0223/evt/event_1/fmt/json :
msg.payload : number
 26
11/17/2022, 11:56:11 AM   node: 333b6df116cf66a7
iot-2/type/Device1/id/0223/evt/event_1/fmt/json :
msg.payload : number
 31
11/17/2022, 11:56:11 AM   node: 333b6df116cf66a7
iot-2/type/Device1/id/0223/evt/event_1/fmt/json :
msg.payload : number
 50
11/17/2022, 11:56:11 AM   node: 333b6df116cf66a7
iot-2/type/Device1/id/0223/evt/event_1/fmt/json :
msg.payload : number
 2
11/17/2022, 11:56:11 AM   node: 333b6df116cf66a7
iot-2/type/Device1/id/0223/evt/event_1/fmt/json :
msg.payload : number
 31
11/17/2022, 11:57:10 AM   node: 545668aa3c72dffd
msg.payload : Object
 ▸{ command: "motoron" }
```