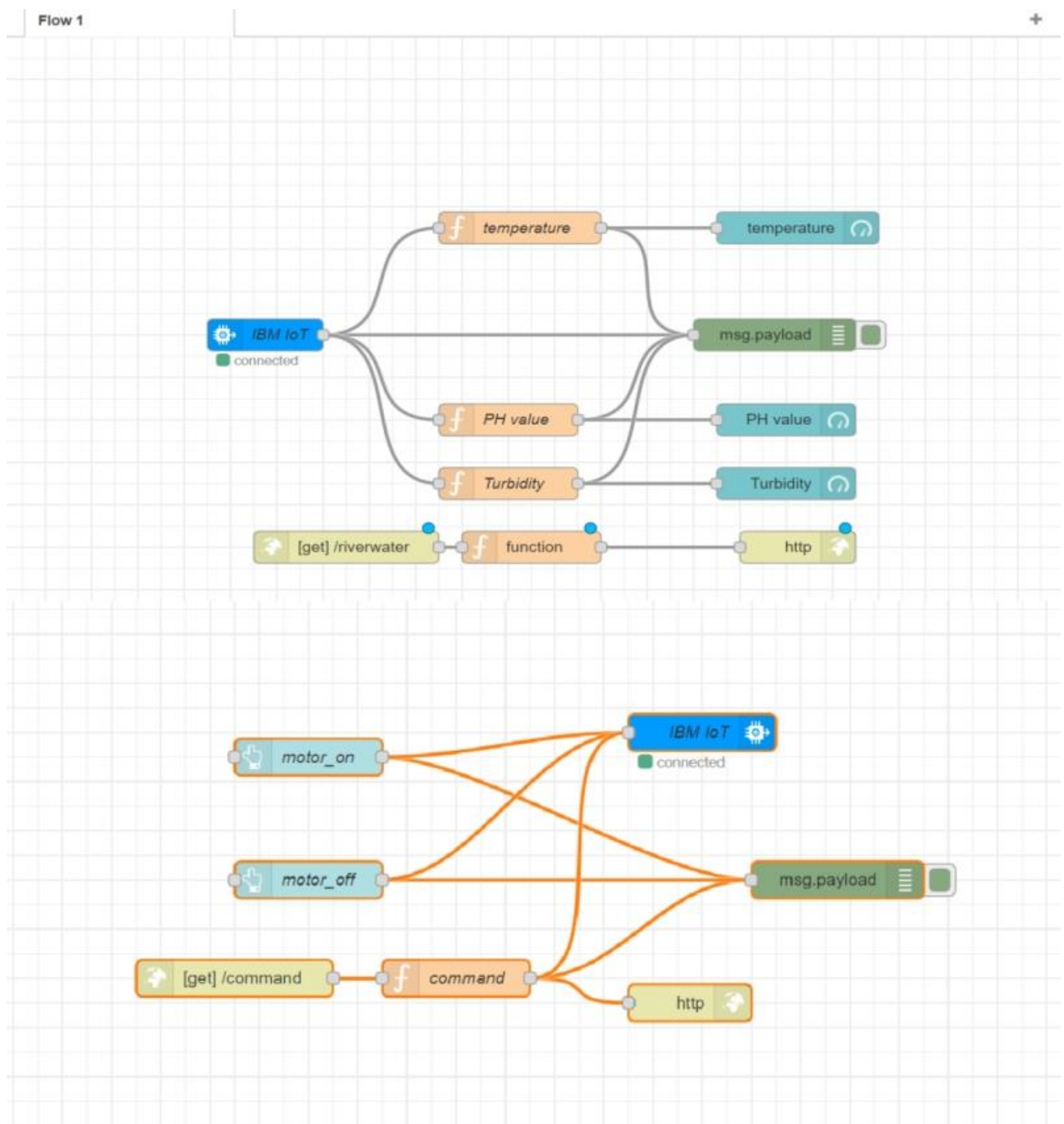


SPRINT 2:

WEB APPLICATION USING NODE RED

Date	8 November 2022
Team ID	PNT2022TMID43374
Project Name	River Water Quality Monitoring and Control System

NODE RED HOME:



IBM IOT NODE:

The image shows the Node-RED web interface. On the left, a sidebar lists various nodes under categories like 'input', 'output', and 'sequence'. The main workspace, 'Flow 1', contains an 'IBM IoT' node (blue with a gear icon) and several function nodes (orange 'f' icons). The 'IBM IoT' node is connected to three function nodes labeled 'tempera', 'PH valu', and 'Turbidity'. Below these, there is a '[get] /riverwater' node connected to a 'funct' node. On the right, the 'Edit ibmiot in node' dialog is open. It has tabs for 'Properties', 'Code', and 'Help'. The 'Properties' tab is active, showing fields for Authentication (API Key), API Key (IBM API), Input Type (Device Event), Device Type (All or kprp), Device Id (All or 2222), Event (All or data), Format (All or json), QoS (0), Name (IBM IoT), and Service (registered). A yellow note at the bottom of the dialog reads: 'Use the Input Type property to configure this node to receive Events sent by IoT Devices, Commands sent to IoT Devices, Status Messages referring to IoT Devices, or Status Messages referring to'. At the bottom left of the dialog is an 'Enabled' checkbox.

SENSORS: TEMPERATURE

The image shows the Node-RED web interface with the 'Edit function node' dialog open. The dialog has tabs for 'Properties', 'Code', and 'Help'. The 'Properties' tab is active, showing the 'Name' field set to 'temperature'. Below the properties, there are four tabs: 'Setup', 'On Start', 'On Message', and 'On Stop'. The 'On Message' tab is selected, and it contains a code editor with the following JavaScript code:

```
1 global.set('Temperature',msg.payload.Temperature)
2 msg.payload=msg.payload.Temperature
3 return msg;
```

The background shows the same 'Flow 1' workspace as the previous image, with the 'IBM IoT' node and its connections.

PH SENSOR:

The screenshot displays the IBM IoT Platform interface. On the left, a canvas shows a flow named 'Flow 1' with an 'IBM IoT' node (status: connected) and a '[get] /riverwater' node. On the right, the 'Edit function node' dialog is open, showing the configuration for the 'PH value' node.

Edit function node

Delete Cancel Done

Properties

Name PH value

Setup On Start On Message On Stop

```
1 global.set('pH',msg.payload.pH)
2 msg.payload=msg.payload.pH
3 return msg;
```

TURBIDITY:

The screenshot displays the IBM IoT Platform interface. On the left, a canvas shows a flow named 'Flow 1' with an 'IBM IoT' node (status: connected) and a '[get] /riverwater' node. On the right, the 'Edit function node' dialog is open, showing the configuration for the 'Turbidity' node.

Edit function node

Delete Cancel Done

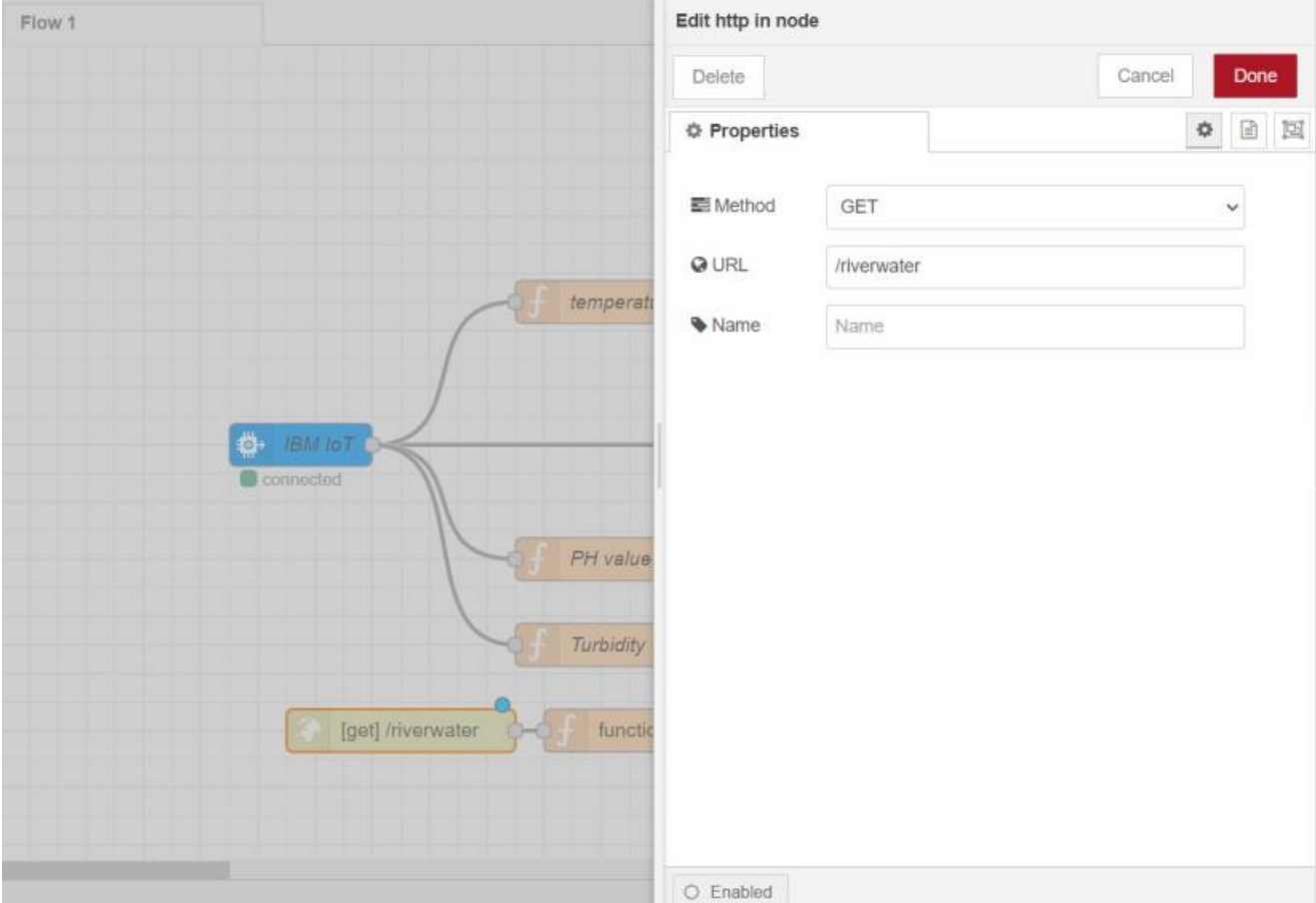
Properties

Name Turbidity

Setup On Start On Message On Stop

```
1 global.set('Turbidity',msg.payload.Turbidity)
2 msg.payload=msg.payload.Turbidity
3 return msg;
```

HTTP NODE:



The screenshot shows a Node-RED workspace with a flow named "Flow 1". The flow starts with an "IBM IoT" node (blue) which is connected. It branches into three function nodes (orange) labeled "temperature", "PH value", and "Turbidity". Below these, there is an HTTP node (orange) labeled "[get] /riverwater" connected to a function node labeled "function". The HTTP node is selected, and its configuration panel is open on the right.

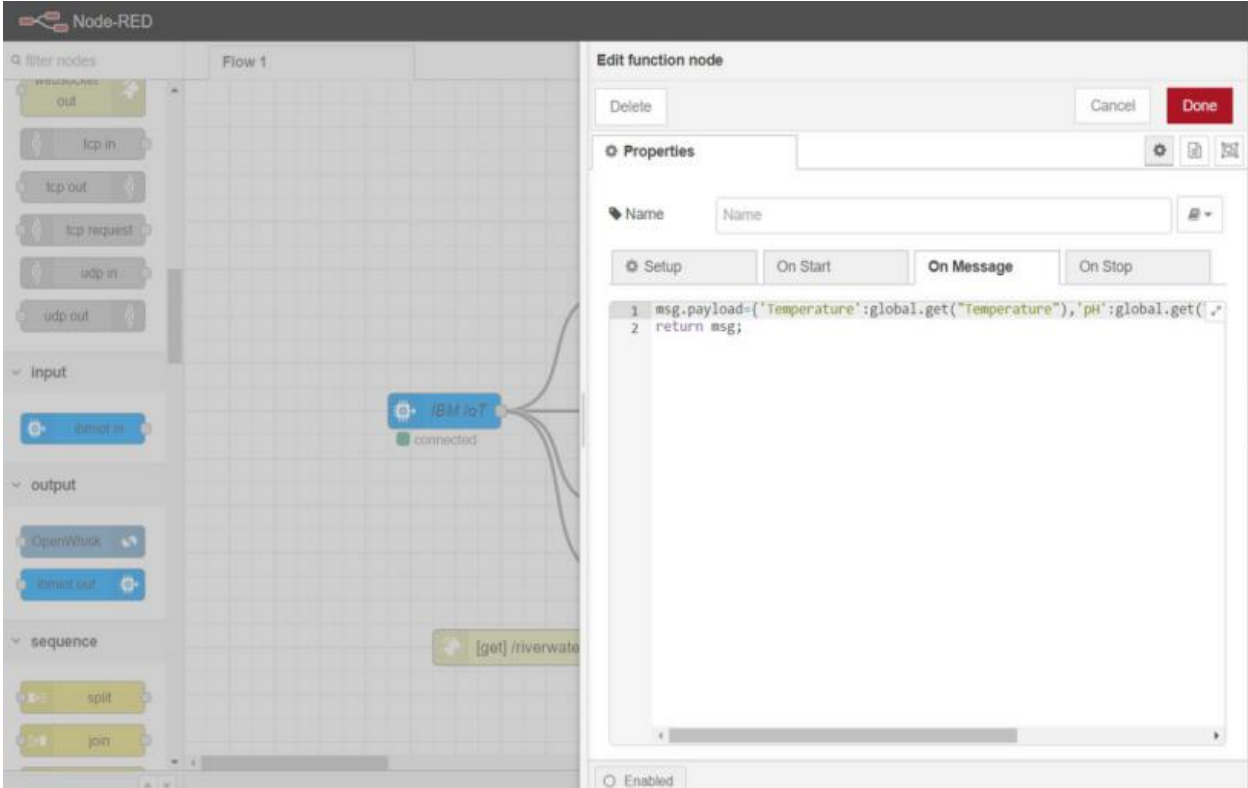
Edit http in node

Buttons: Delete, Cancel, Done

Properties

- Method: GET
- URL: /riverwater
- Name: Name

☐ Enabled



The screenshot shows the same Node-RED workspace, but the function node is selected instead of the HTTP node. The configuration panel for the function node is open on the right.

Edit function node

Buttons: Delete, Cancel, Done

Properties

- Name: Name

Tabs: Setup, On Start, **On Message**, On Stop

```
1 msg.payload={ 'temperature':global.get("temperature"),'ph':global.get('ph') }
2 return msg;
```

☐ Enabled

DASHBOARD NODES: TEMPERATURE:

Flow 1

IBM IoT

connected

f

temperatu

f

PH value

f

Turbidity

[get] /riverwater

f

funcio

Edit gauge node

Delete

Cancel

Done

Properties

Group

[riverwater] DISPLAY

Size

auto

Type

Gauge

Label

temperature

Value format

{{value}}

Units

units

Range

min 0

max 40

Colour gradient

Sectors

0

...

optional

...

optional

...

40

</> Class

Optional CSS class name(s) for widget

Name

Enabled

PH SENSOR:

Flow 1

IBM IoT

connected

f

temperatu

f

PH value

f

Turbidity

[get] /riverwater

f

funcio

Edit gauge node

Delete

Cancel

Done

Properties

Group

[riverwater] DISPLAY

Size

auto

Type

Gauge

Label

PH value

Value format

{{value}}

Units

units

Range

min 0

max 14

Colour gradient

Sectors

0

...

optional

...

optional

...

14

</> Class

Optional CSS class name(s) for widget

Name

Enabled

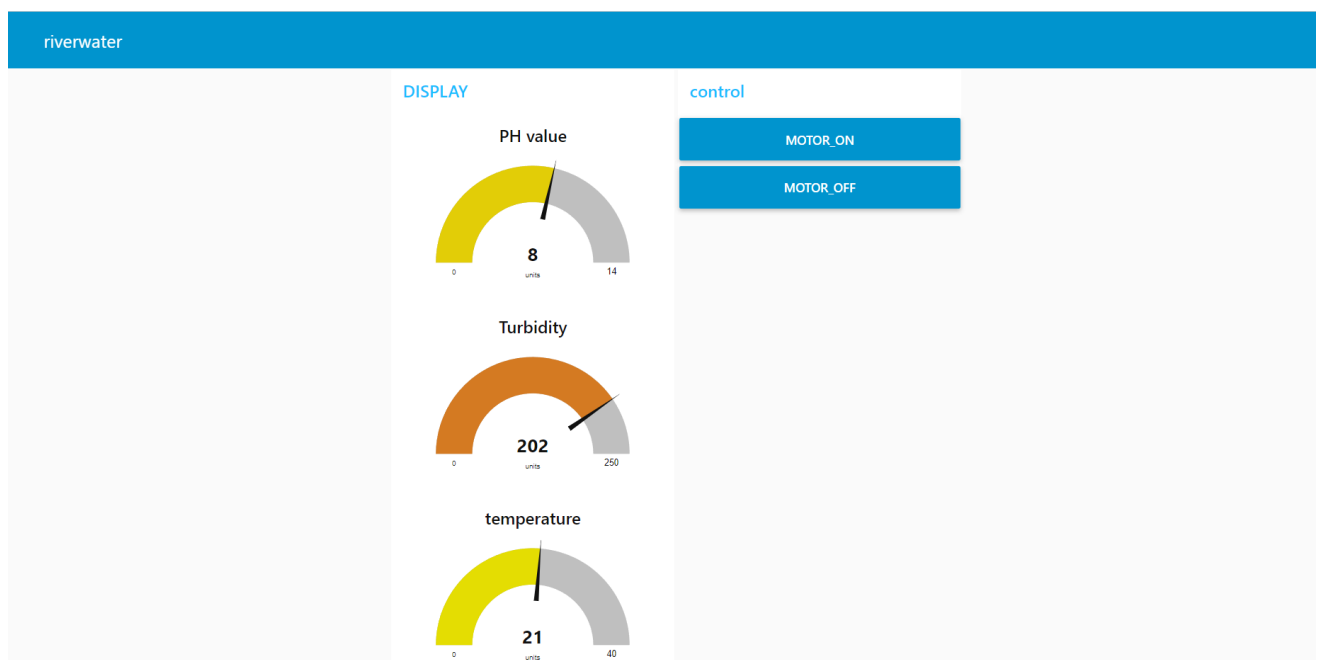
TURBIDITY:

The screenshot shows the IBM IoT Flow editor interface. On the left, a workspace titled 'Flow 1' contains a flow diagram. It starts with an 'IBM IoT' node (green circle, 'connected' status) which branches into three function nodes: 'temperature', 'PH value', and 'Turbidity'. These are connected to a '[get] /riverwater' node, which then connects to a 'function' node. On the right, the 'Edit gauge node' properties panel is open. It includes buttons for 'Delete', 'Cancel', and 'Done'. The 'Properties' section contains the following settings:

- Group: [riverwater] DISPLAY
- Size: auto
- Type: Gauge
- Label: Turbidity
- Value format: {{value}}
- Units: units
- Range: min 0, max 250
- Colour gradient: A gradient bar with green, yellow, and red segments.
- Sectors: 0, optional, optional, 250
- </> Class: Optional CSS class name(s) for widget
- Name: (empty field)

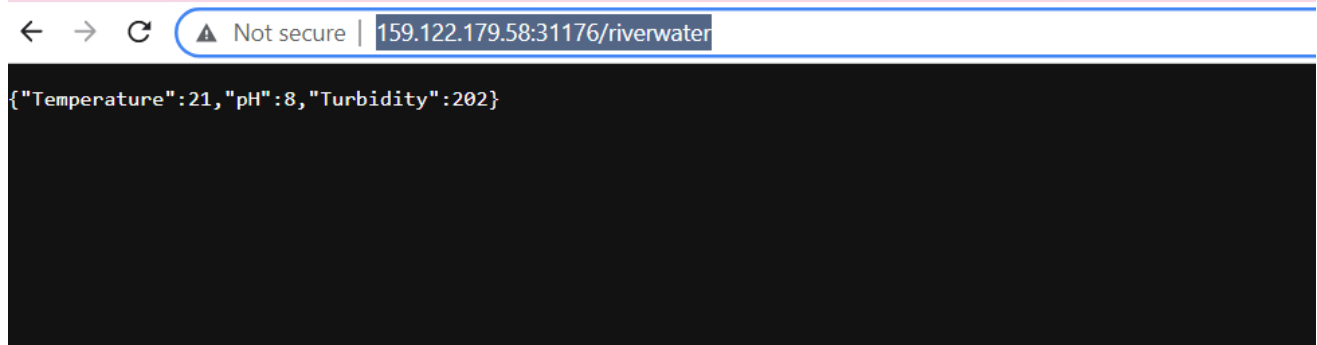
At the bottom of the panel, there is an 'Enabled' checkbox.

WEB APPLICATION: DASHBOARD:



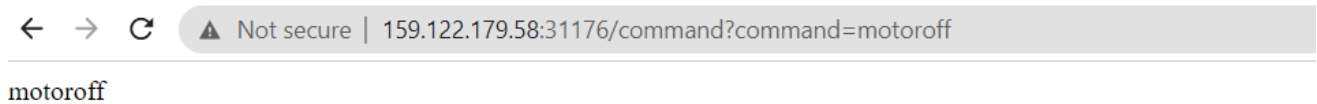
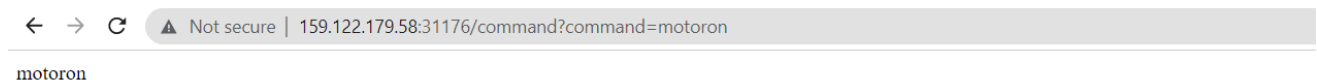
DASHBOARD LINK: <http://159.122.179.58:31176/ui/#!/0?socketid=xpzwLq-likptr6qnAAAr>

WEB PAGE: MONITORING:



WEB PAGE LINK: <http://159.122.179.58:31176/riverwater>

CONTROL :



LINKS:

- <http://159.122.179.58:31176/command?command=motoroff>
- <http://159.122.179.58:31176/command?command=motoroff>