

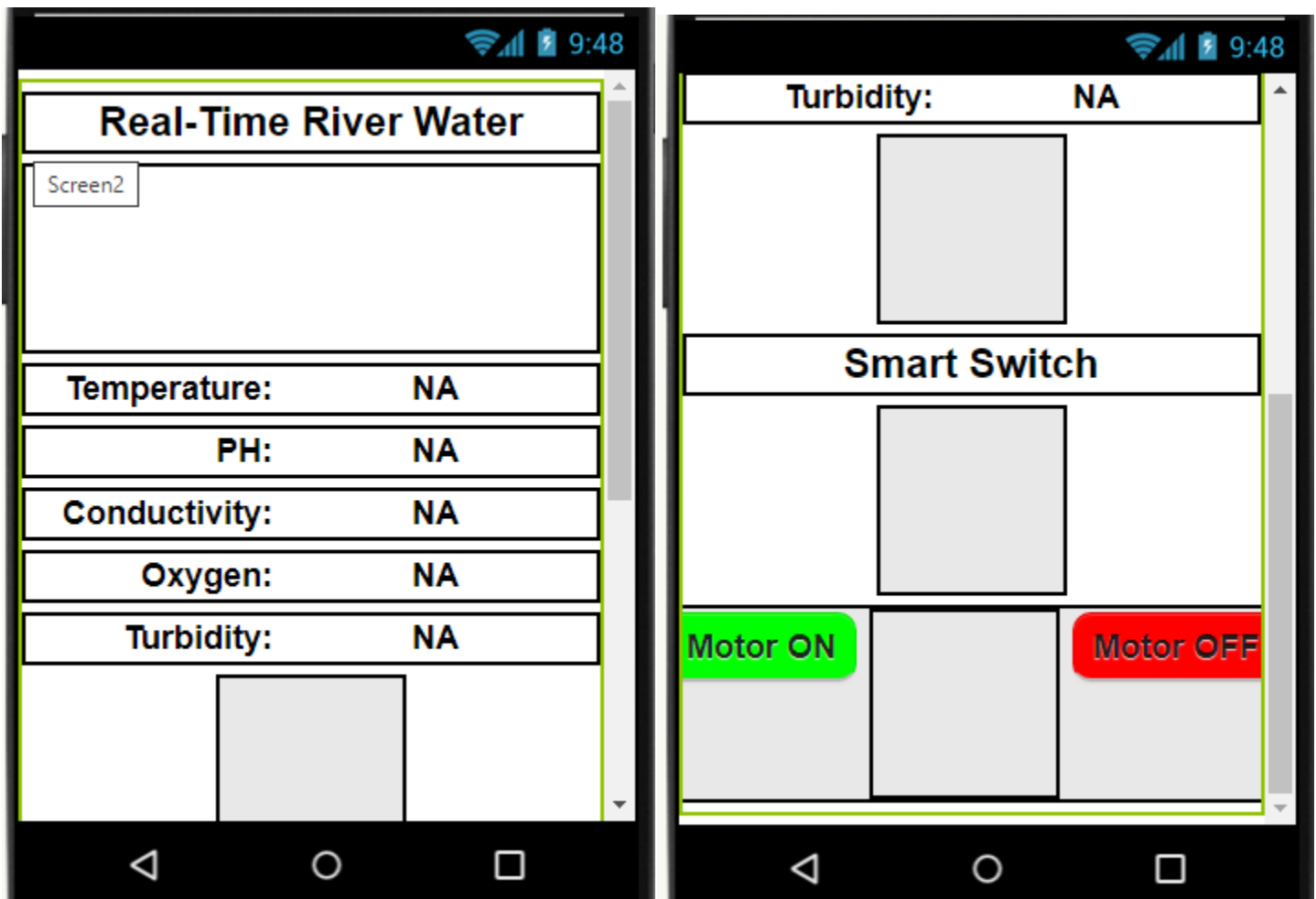
Project Development Phase

Sprint – 3

Date	13 Nov 2022
Team ID	PNT2022TMID06691
Project Name	Real-Time River Water Quality Monitoring and Control System
Maximum Marks	8 Marks

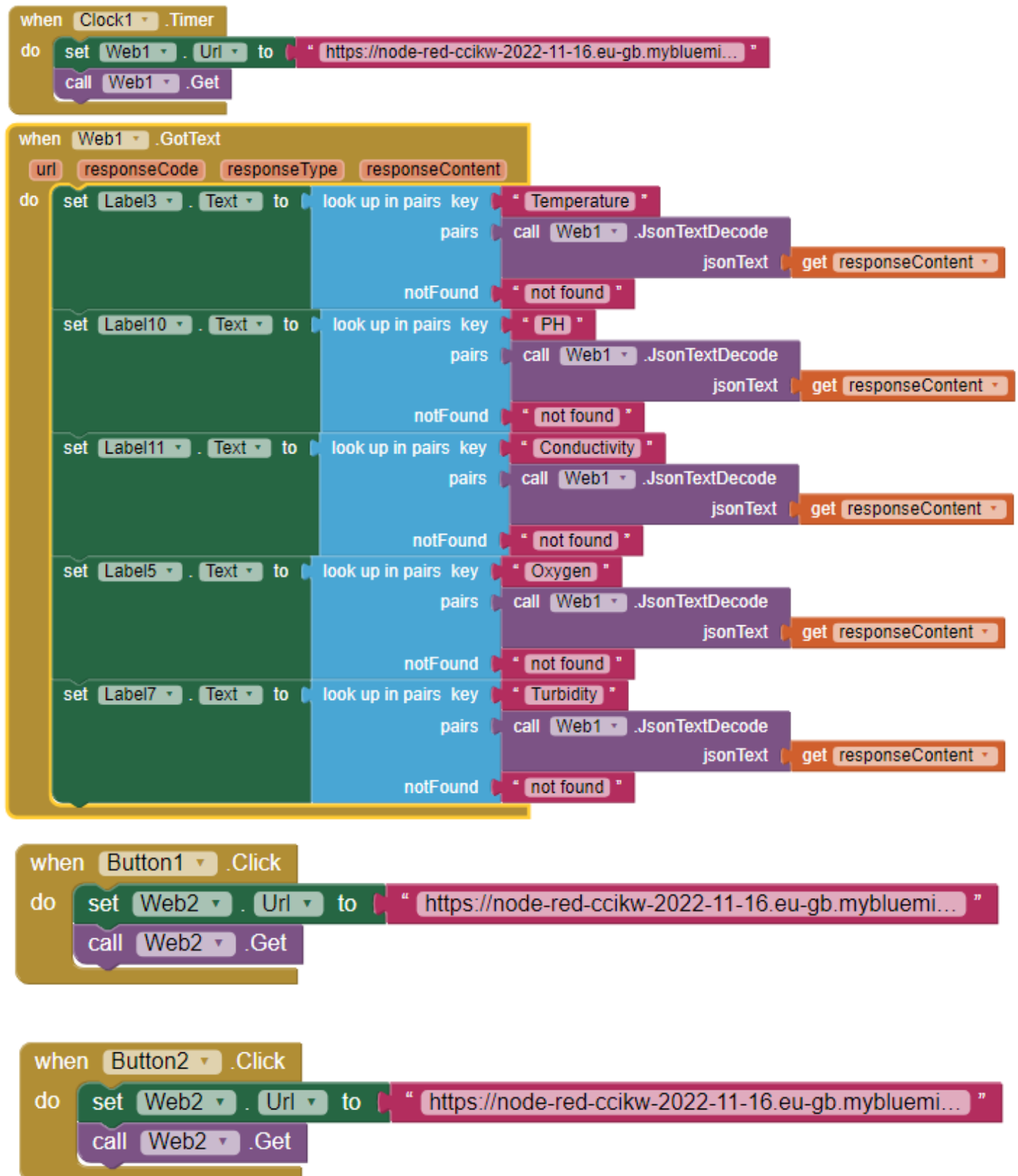
USN – 8: MIT app inventor (Front end Design)

As a user, I can create the front end design for the application using MIT app Inventor



USN – 9: MIT app inventor (Back end Design)

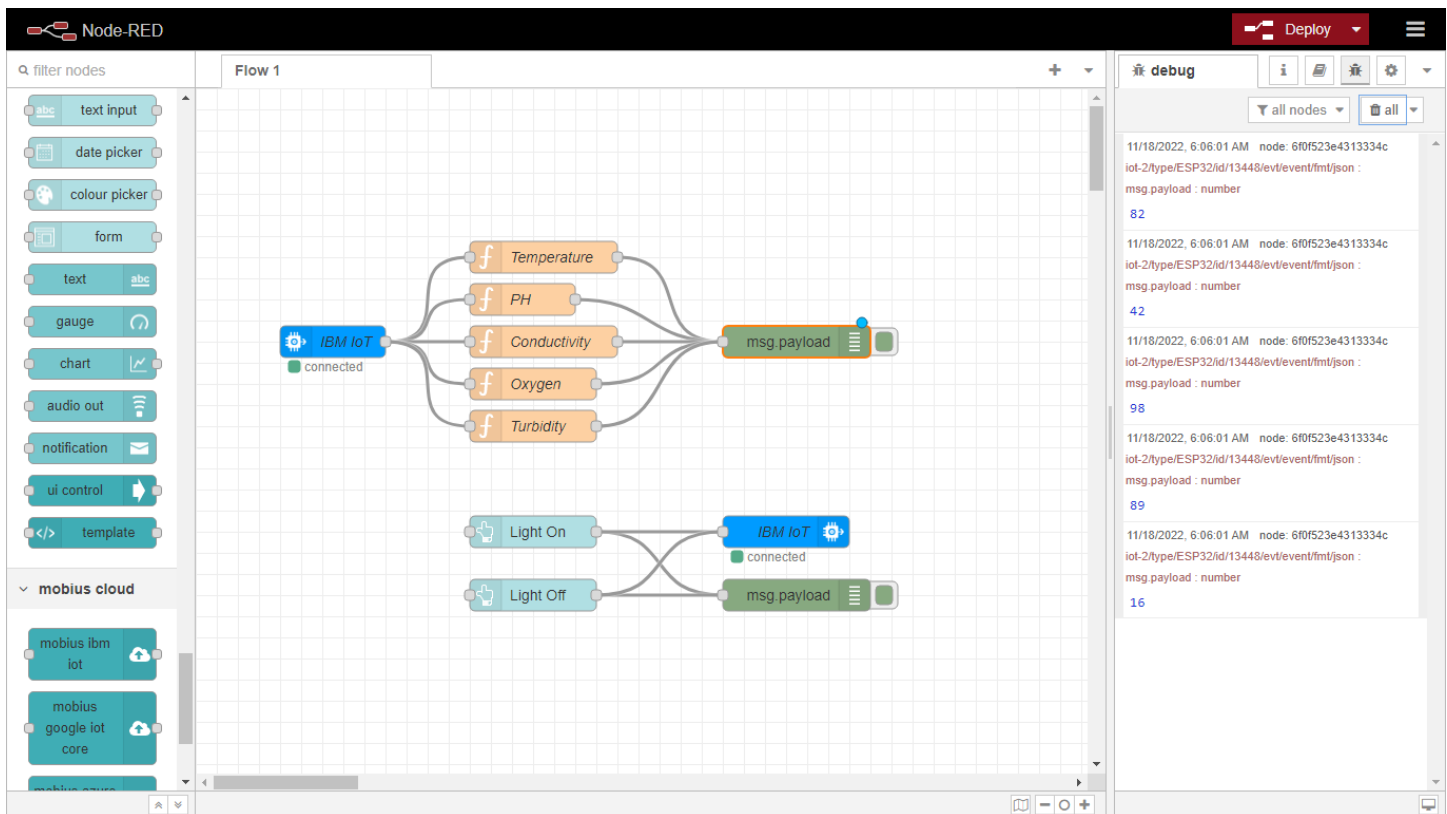
As a user, I can create the back end design for the application using MIT app Inventor



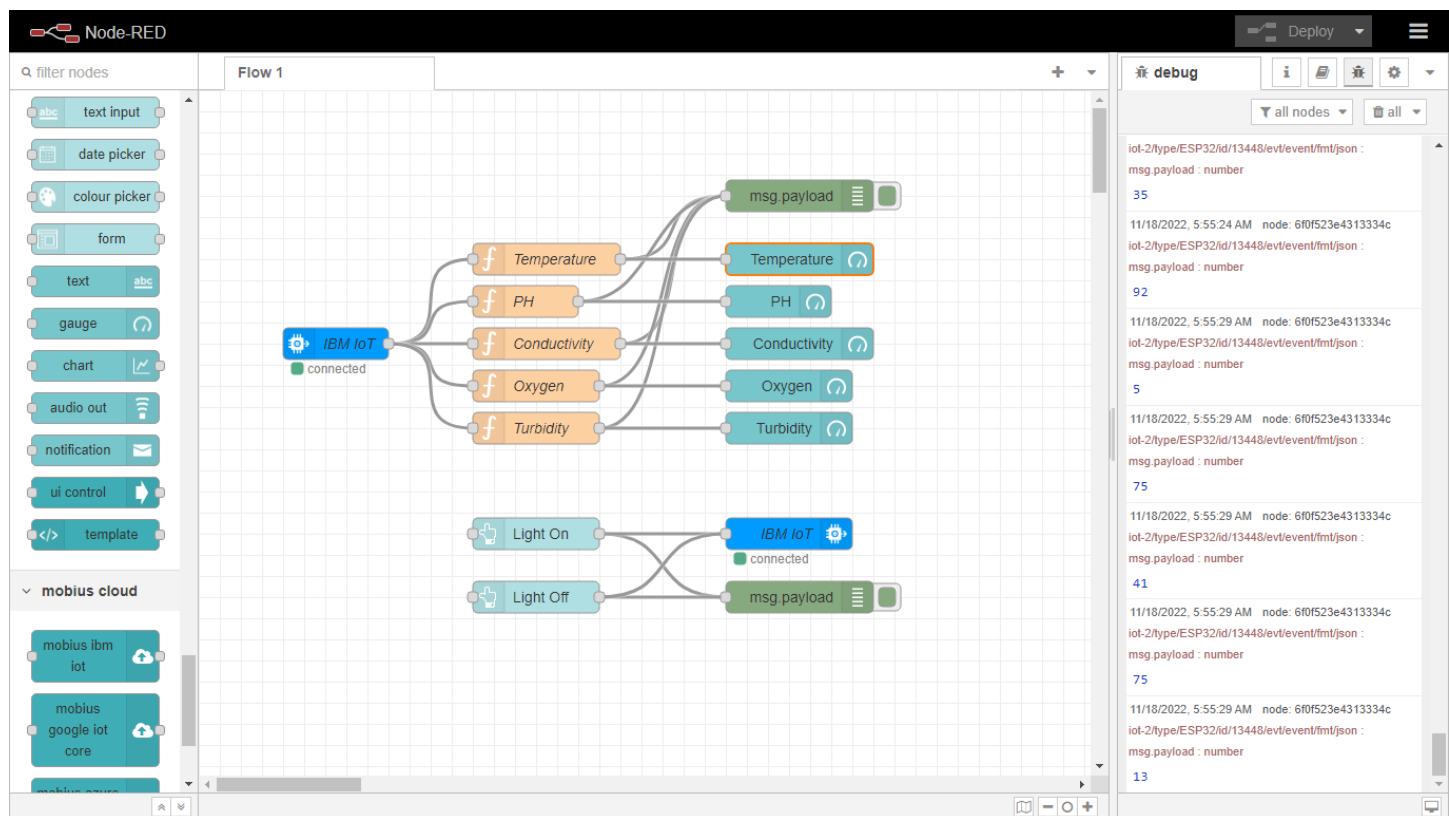
USN – 10: Create a Web UI using Node-red

As a user, I can create a Web UI, to access the data from the cloud and display all parameters.

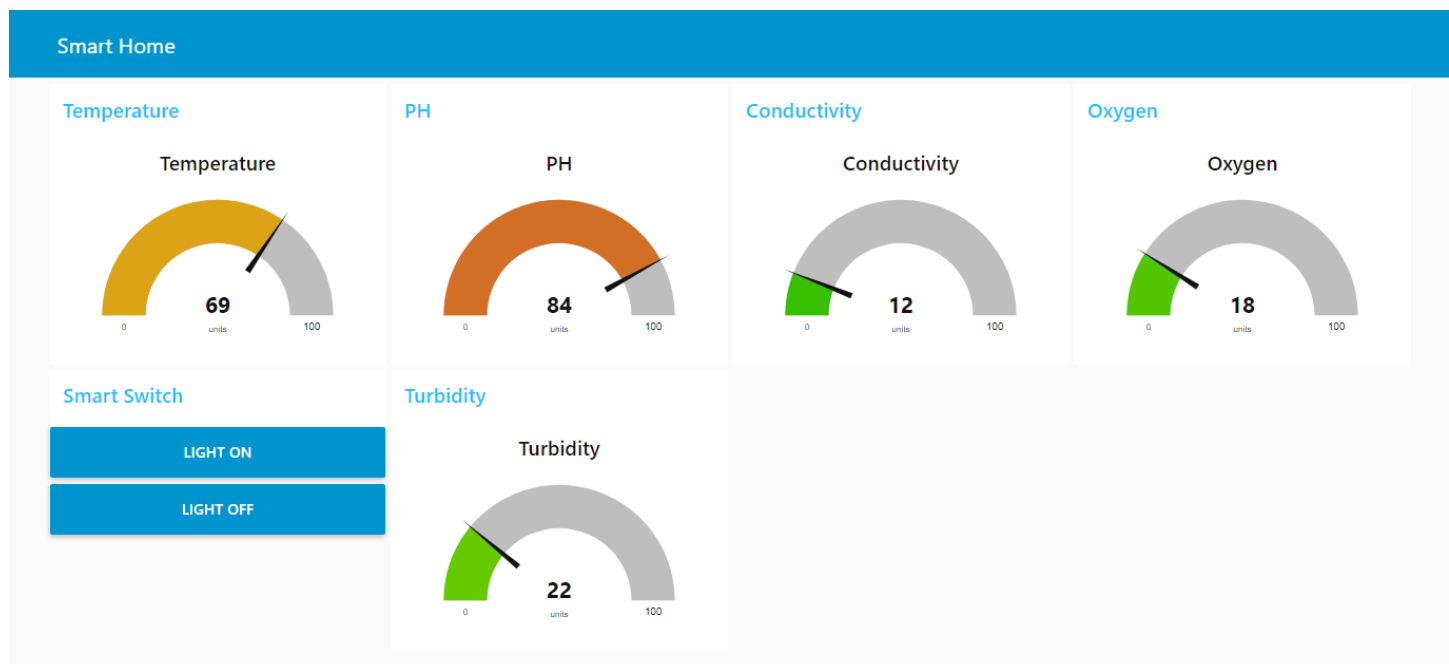
Creating web application in Node-Red:



Creating Web UI:



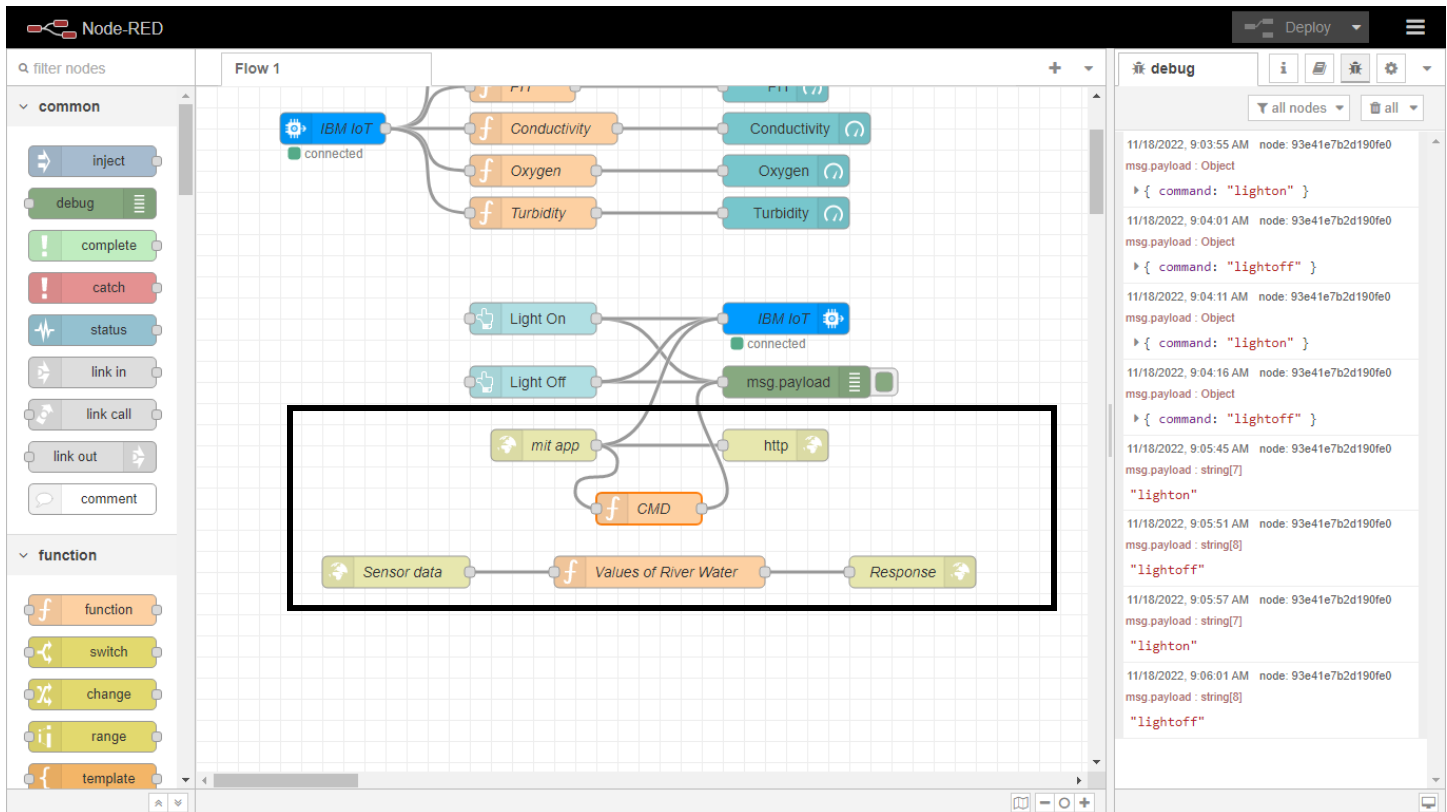
Web UI Dashboard:



USN – 11: Connecting Node-red to Mobile Application

As a user, I can connect the node-red to the mobile application to display all the parameters in the mobile app

Creating HTTP Request:



Data in Mobile App:



Real-Time River Water Quality Monitoring and Control System

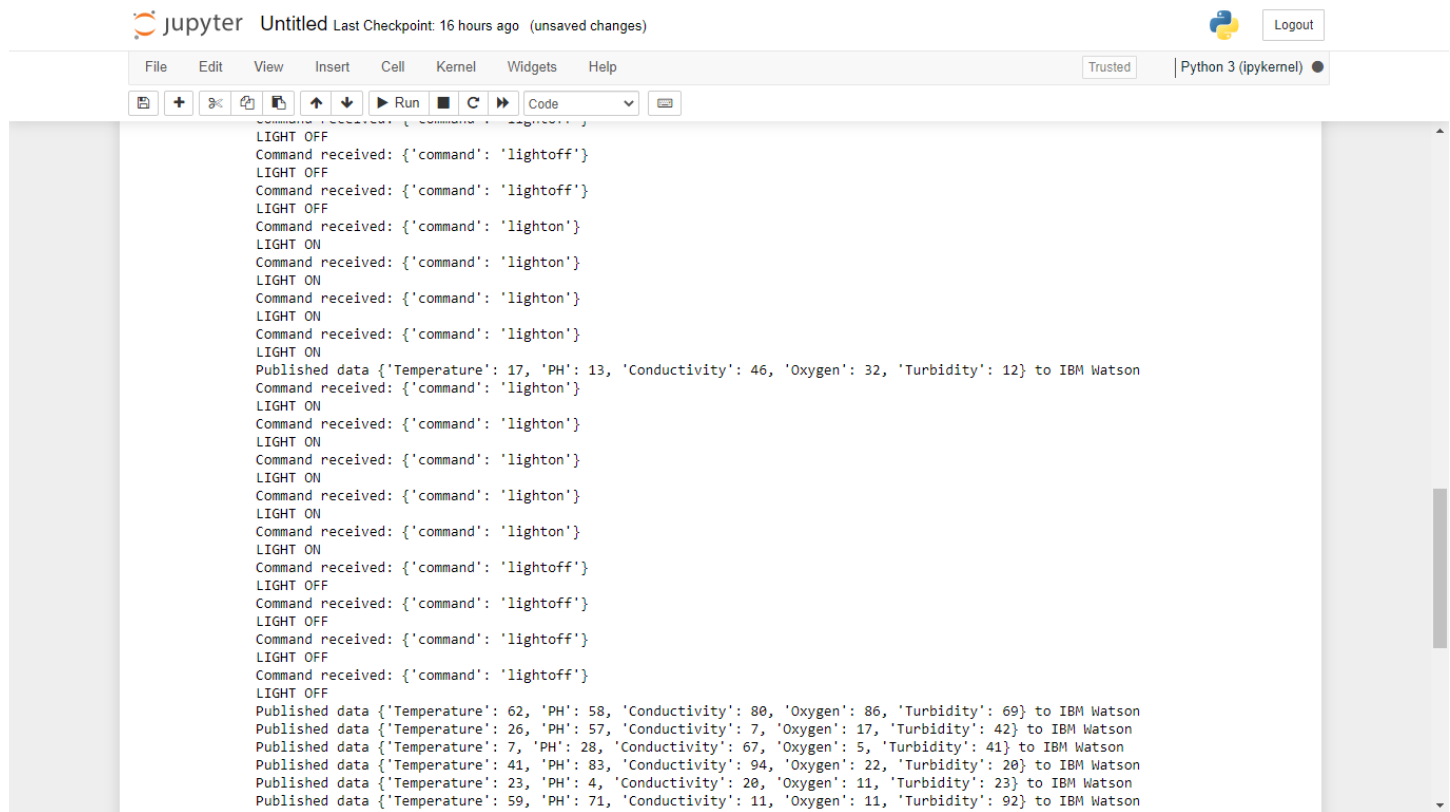
Temperature:	69
PH:	84
Conductivity:	12
Oxygen:	18
Turbidity:	22

Smart Switch

Motor ON

Motor OFF

Command Received in Python Shell when the button is Clicked:



The image shows a Jupyter Notebook interface with a menu bar (File, Edit, View, Insert, Cell, Kernel, Widgets, Help) and a toolbar with icons for file operations, running, and code execution. The notebook is titled "Untitled" and shows "Last Checkpoint: 16 hours ago (unsaved changes)". The Python 3 (ipykernel) environment is selected. The code cell contains a series of commands and data published to IBM Watson.

```
LIGHT OFF
Command received: {'command': 'lightoff'}
LIGHT OFF
Command received: {'command': 'lightoff'}
LIGHT OFF
Command received: {'command': 'lighton'}
LIGHT ON
Command received: {'command': 'lighton'}
LIGHT ON
Command received: {'command': 'lighton'}
LIGHT ON
Command received: {'command': 'lighton'}
LIGHT ON
Published data {'Temperature': 17, 'PH': 13, 'Conductivity': 46, 'Oxygen': 32, 'Turbidity': 12} to IBM Watson
Command received: {'command': 'lighton'}
LIGHT ON
Command received: {'command': 'lighton'}
LIGHT ON
Command received: {'command': 'lighton'}
LIGHT ON
Command received: {'command': 'lighton'}
LIGHT ON
Command received: {'command': 'lighton'}
LIGHT ON
Command received: {'command': 'lightoff'}
LIGHT OFF
Command received: {'command': 'lightoff'}
LIGHT OFF
Command received: {'command': 'lightoff'}
LIGHT OFF
Command received: {'command': 'lightoff'}
LIGHT OFF
Published data {'Temperature': 62, 'PH': 58, 'Conductivity': 80, 'Oxygen': 86, 'Turbidity': 69} to IBM Watson
Published data {'Temperature': 26, 'PH': 57, 'Conductivity': 7, 'Oxygen': 17, 'Turbidity': 42} to IBM Watson
Published data {'Temperature': 7, 'PH': 28, 'Conductivity': 67, 'Oxygen': 5, 'Turbidity': 41} to IBM Watson
Published data {'Temperature': 41, 'PH': 83, 'Conductivity': 94, 'Oxygen': 22, 'Turbidity': 20} to IBM Watson
Published data {'Temperature': 23, 'PH': 4, 'Conductivity': 20, 'Oxygen': 11, 'Turbidity': 23} to IBM Watson
Published data {'Temperature': 59, 'PH': 71, 'Conductivity': 11, 'Oxygen': 11, 'Turbidity': 92} to IBM Watson
```