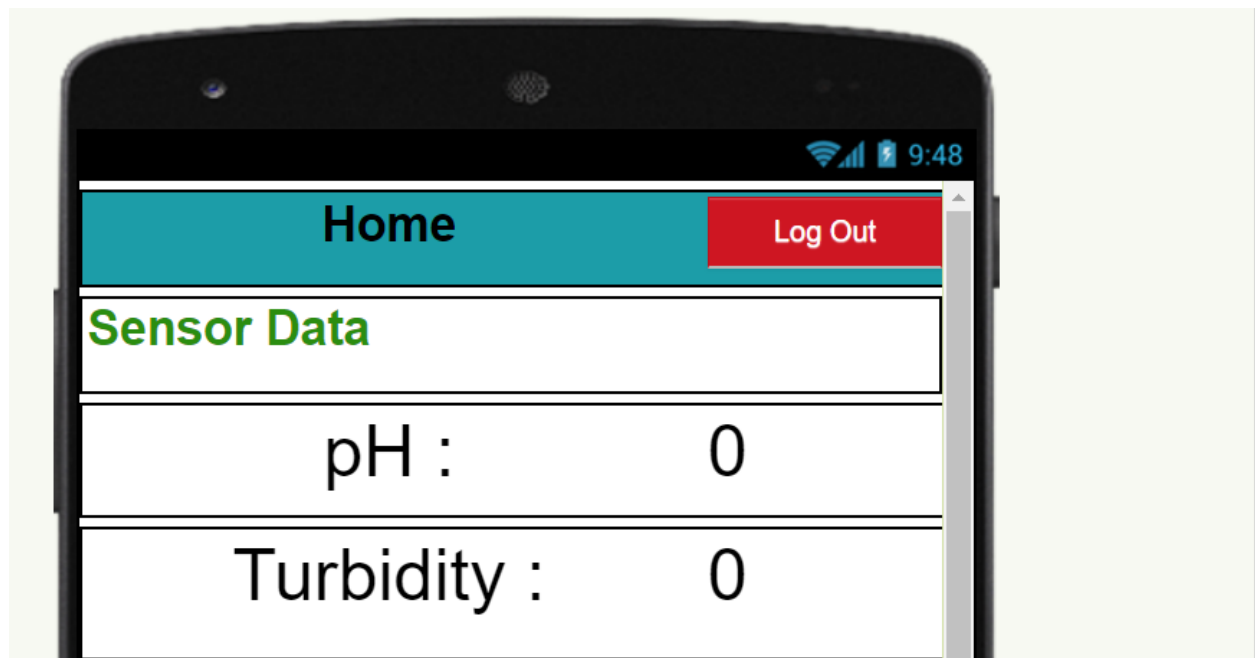


Building Mobile App

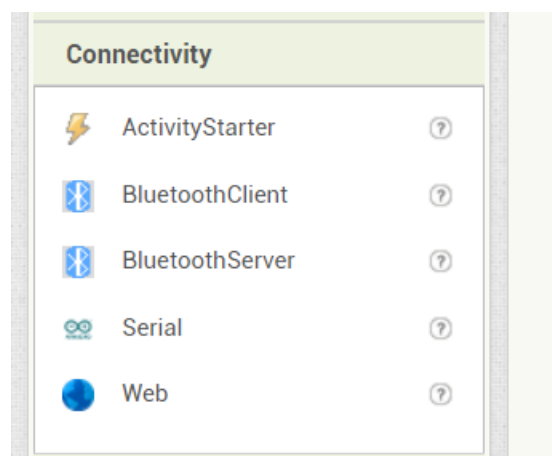
TEAM ID	PNT2022TMID48692
PROJECT TITLE	REAL TIME RIVER WATER QUALITY MONITORING AND CONTROL SYSTEM

Configure The Application To Receive The Data From Cloud

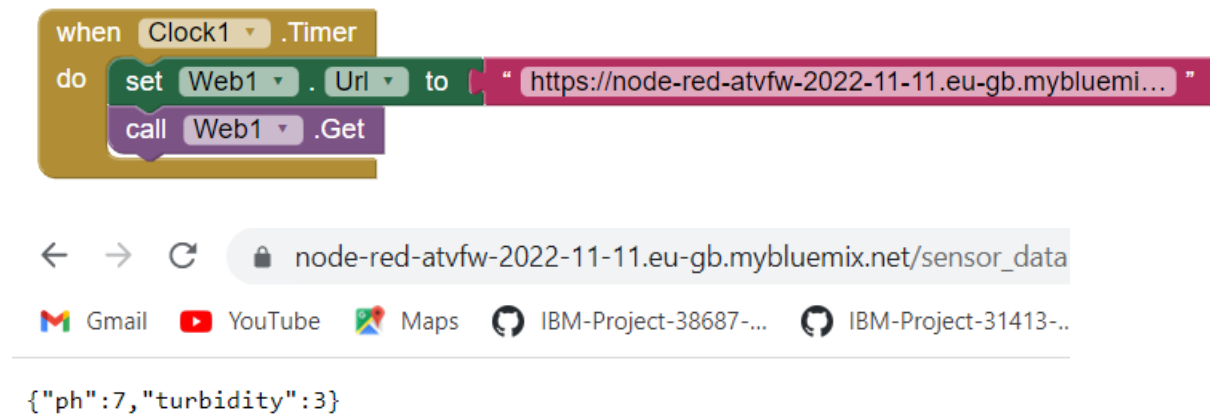
Step 1: After designing the UI and now Configure the application to receive the data from the Cloud



Step 2: In the Connectivity add the Web component in the screen



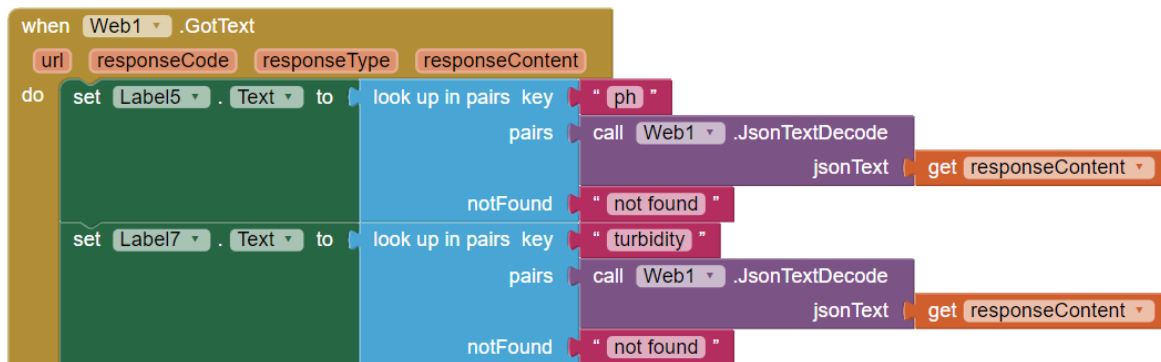
Step 3: Add the Clock Timer in the block, now set the Web Url as the api in the Node-red after setting the Web and now call the Get method



The Node-RED flow diagram shows a 'when' block with a 'Clock1' timer. Inside the 'do' loop, there is a 'set' block for 'Web1.Url' with the value 'https://node-red-atvfw-2022-11-11.eu-gb.mybluemix.net/sensor_data', followed by a 'call' block for 'Web1.Get'.

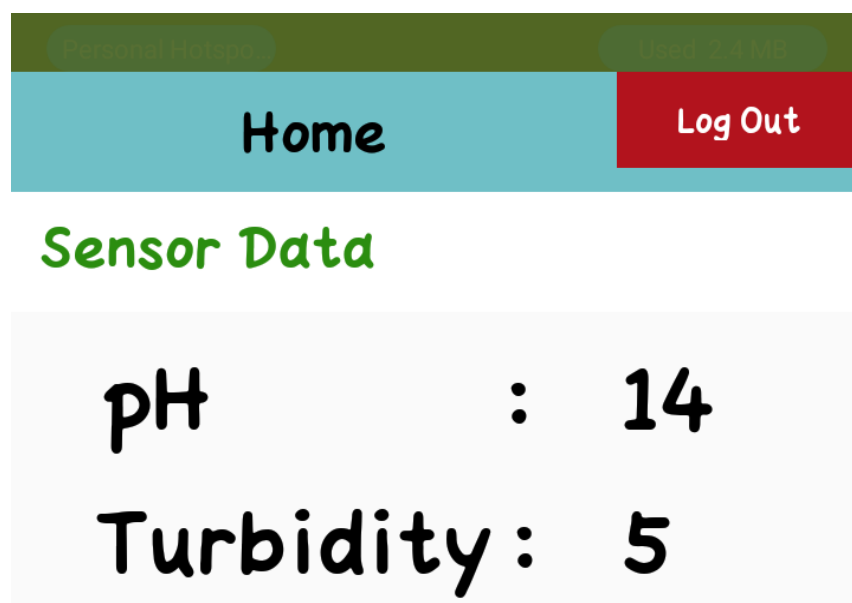
Below the flow diagram is a screenshot of a web browser showing the URL 'node-red-atvfw-2022-11-11.eu-gb.mybluemix.net/sensor_data'. The browser's address bar and navigation icons are visible. The page content displays a JSON object: `{"ph":7,"turbidity":3}`.

Step 4: Add Web GotText block and change the label when the values of the API changes



The Node-RED flow diagram shows a 'when' block with a 'Web1.GotText' trigger. The 'do' loop contains two 'set' blocks for 'Label5.Text' and 'Label7.Text'. Each 'set' block is followed by a 'look up in pairs' block with a key of 'ph' and 'turbidity' respectively. The 'pairs' block is connected to a 'call' block for 'Web1.JsonTextDecode' with a 'jsonText' input from 'get responseContent'. The 'notFound' output of the 'look up in pairs' block is set to 'not found'.

Step 5: Now connect to the AI companion in the App now we can able to view the Sensor values



The app interface shows a 'Home' screen with a 'Log Out' button. Below the 'Home' screen, the 'Sensor Data' is displayed. The sensor data shows 'pH : 14' and 'Turbidity: 5'.