

SmartFarmer - IoT Enabled Smart Farming Application

Project Development Phase – Sprint 3

Configuration of Node-Red to send commands to IBM cloud

ibmiot out node is used to send data from Node-Red to IBM Watsondevice. So, after adding it to the flow we need to configure it with credentials of our Watson device.

The screenshot shows the Node-RED web interface in a browser. The flow 'Flow 1' contains an 'IBM IoT' node (green) connected to two function nodes: 'temperature' and 'humidity'. These function nodes are connected to 'msg.payload' and then to 'Temperature' and 'Humidity' nodes (blue). Below this, there are 'Motor ON' and 'Motor OFF' nodes (blue) connected to a 'function' node (orange). This 'function' node is connected to the 'IBM IoT' node. At the bottom, there are '[get]/sensordata' and '[get]/control' nodes (green) connected to a 'function' node (orange). This 'function' node is connected to two 'http' nodes (green) and a 'msg.payload' node (green). The right sidebar shows the 'debug' console with a log of messages sent to the IBM IoT node, including temperature and humidity data.

The screenshot shows the Node-RED web interface with the 'Edit http in node' dialog box open. The dialog box has a 'Delete' button, a 'Cancel' button, and a 'Done' button. The 'Properties' section shows the 'Method' set to 'GET', the 'URL' set to '/sensordata', and the 'Name' set to 'Name'. The right sidebar shows the 'debug' console with a log of messages sent to the IBM IoT node, including temperature and humidity data.

Node-RED interface showing a flow with an IBM IoT node connected to a function node. The function node is configured to send data to a URL. The debug console shows the output of the function node, displaying temperature and humidity data.

Edit http in node

Method: GET
URL: /control
Name: Name

debug

```
11/18/2022, 4:34:55 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : Object  
{ temp : 91, Humid : 99 }  
11/18/2022, 4:34:55 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : number  
91  
11/18/2022, 4:34:56 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : number  
99  
11/18/2022, 4:35:05 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : Object  
{ temp : 96, Humid : 75 }  
11/18/2022, 4:35:05 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : number  
96  
11/18/2022, 4:35:06 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : number  
75
```

Node-RED interface showing a flow with an IBM IoT node connected to a function node. The function node is configured to send data to a URL. The debug console shows the output of the function node, displaying temperature and humidity data.

Edit function node

Name: Name

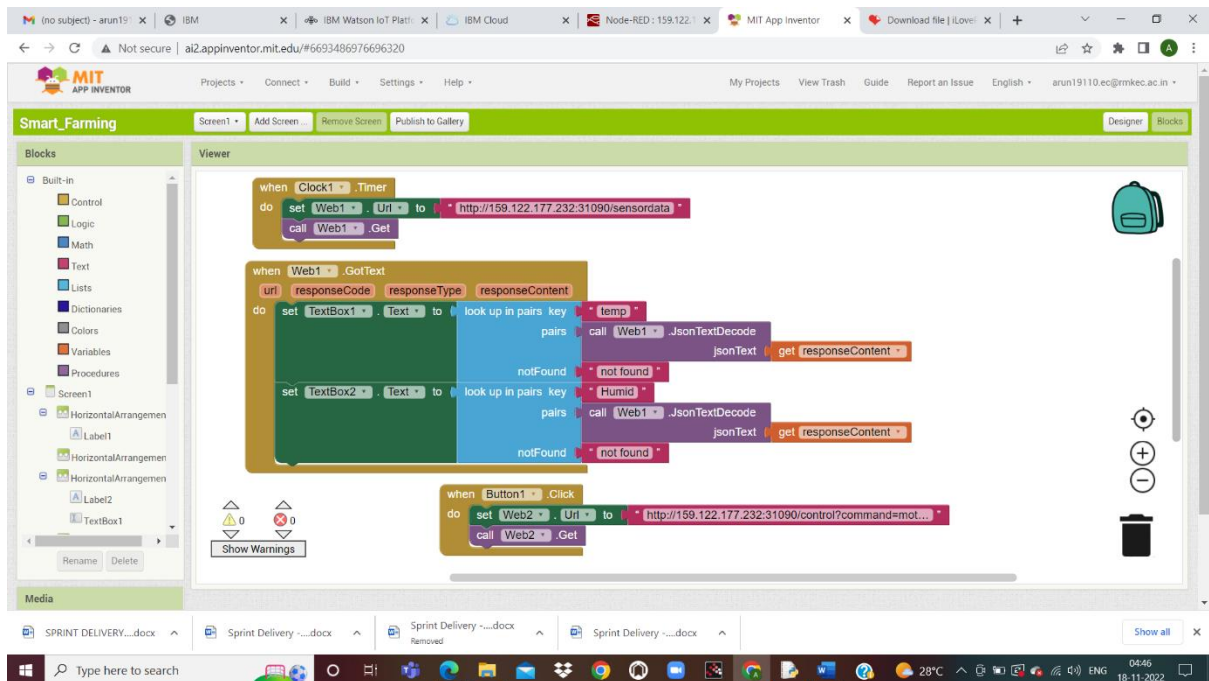
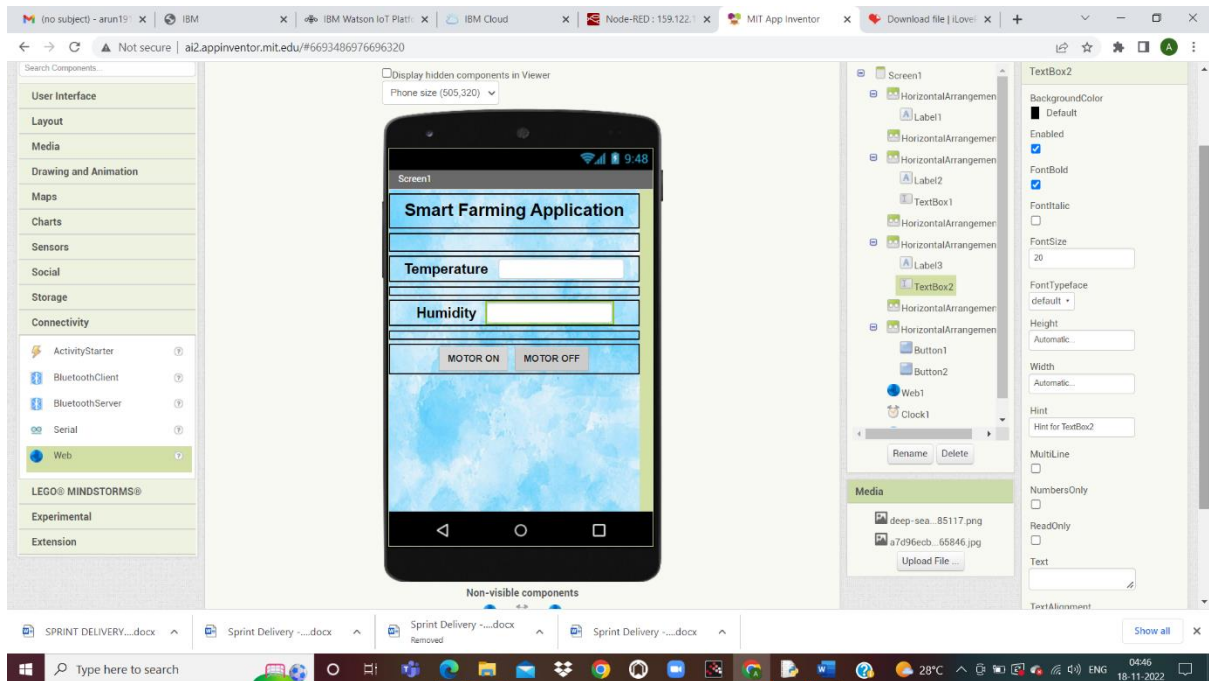
Setup On Start On Message On Stop

```
1 = msg.payload = {  
2   "temp": global.get('t'),  
3   "Humid": global.get('h')  
4 }  
5 return msg;
```

debug

```
11/18/2022, 4:35:05 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : Object  
{ temp : 96, Humid : 75 }  
11/18/2022, 4:35:05 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : number  
96  
11/18/2022, 4:35:06 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : number  
75  
11/18/2022, 4:35:15 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : Object  
{ temp : 106, Humid : 92 }  
11/18/2022, 4:35:15 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : number  
106  
11/18/2022, 4:35:16 AM node: 0ea8da4c58208d38  
iot-2/type/abcdid/12345evn/iotSensor/rmt/json :  
msg.payload : number  
92
```

Create an application in MIT Inventor App



Build an android apk and download it an android mobile phone

