



Connecting IoT Devices to Watson IoT with Node Red

DS'17 Hands-On IoT Lab

Manasa Sutapalli

Lead Researcher Cloud/Enterprise IoT
Miracle Software Systems, Inc.

December 7th, 2017

Connecting IoT Devices to Watson IoT with Node Red

Goal

In this lab the user will create a Bluemix Watson IoT Service and will add Logistic Simulator to Watson IoT Platform. It will send events like – Speed, Fuel, Tyre Pressure, Location to the IBM IoT Cloud.

Pre-Requisites

You will need the following to complete this lab successfully,

- Active email ID for registering with Bluemix
- Download and Install NodeJS
- Test Editor such as Sublime Text (or) Notepad ++

Technology Involved

- IBM Bluemix(PaaS)
- Watson IoT Platform
- Node-Red
- Miracle's Logistic Simulator

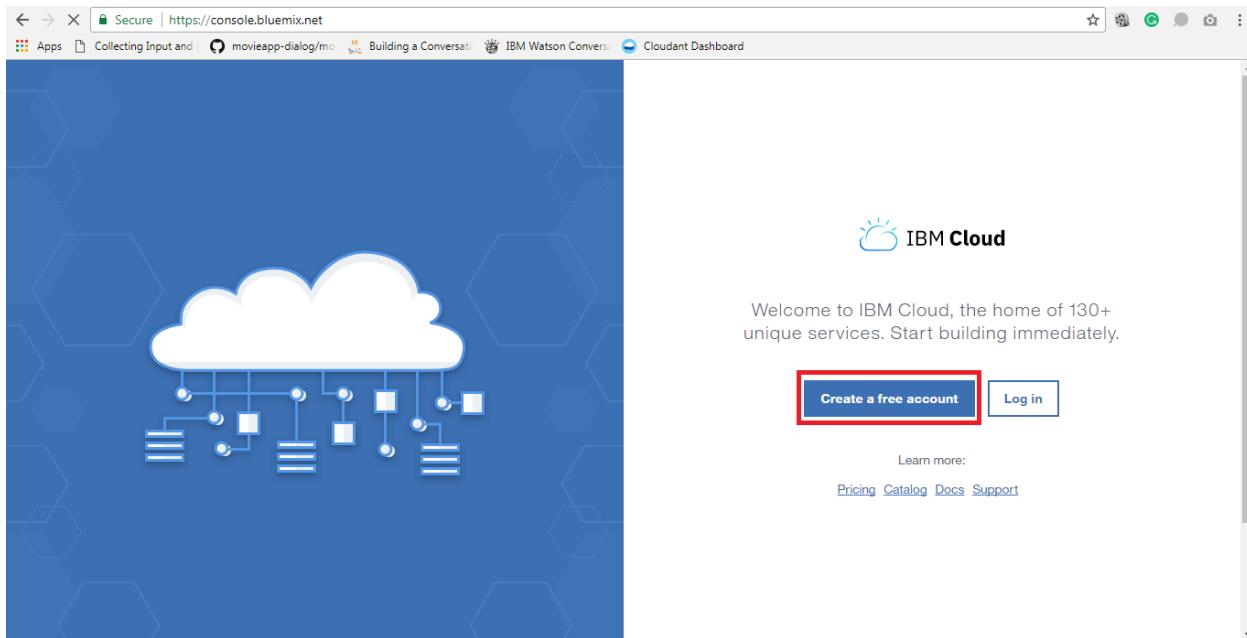
Lab Steps

So, let us get started with the lab!

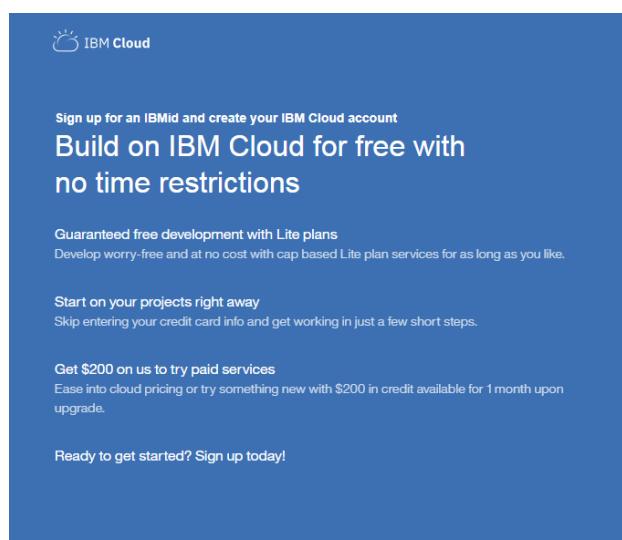
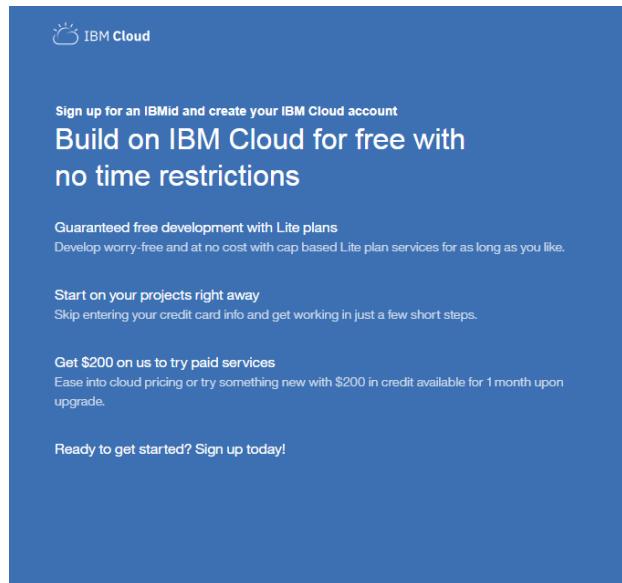
#1 | Create IBM Bluemix account

The first step will be to make sure that we have access to the IBM Bluemix Console with either the free trial option (or) the paid subscription option.

Login to Bluemix at <http://bluemix.net> (or) Register today at <https://console.ng.bluemix.net/registration/>



Click on Create a free account, and the fields as required.



After Clicking on "**Create Account**", confirmation mail will be sent to the registered mail id. Click on Confirm account and then Login to your Bluemix account.

Log into IBM Cloud

Enter Email or IBMid: [Forgot your IBMid?](#)

[Continue](#)

New? [Create an IBM Cloud account](#)

After your login, you can see the dashboard where you can take a look at your applications and services.

The screenshot shows the IBM Cloud Dashboard. At the top, there's a navigation bar with 'Docs', 'IBM Cloud Dashboard' (highlighted), 'Catalog', 'Support', and 'Manage'. Below the navigation is a large circular icon containing a 3x3 grid of squares. To its right, the word 'Dashboard' is displayed. A message below the icon says: 'We noticed you don't have anything on your platform yet, get started with one of the options below, or go to the catalog to create a new application or service.' A blue button labeled 'Explore our Offerings' is present. On the right side of the dashboard, there are several service icons with their names: 'Liberty for Java™', 'SDK for Node.js™', 'ASP.NET Core', 'Runtime for Swift', 'XPages', 'Go', 'PHP', 'Python', 'Ruby', and 'Tomcat'. Each icon consists of a colored circle with a white file-like symbol and the extension name below it.

The next step will be to take your application and deploy it back to Bluemix so that you can share it with your friends.

#2 | Create Application and Watson IoT Service

Click on **Catalog**, for creating the application

The screenshot shows the IBM Cloud Catalog interface. At the top, there's a navigation bar with links for Catalog, Docs, Support, and Manage. A search bar contains the text "label:lite". Below the search bar, a message says "Try the best of the Catalog for free with no time restrictions with Lite plans. The Lite filter is enabled. Remove the filter to see the full Catalog." On the left, a sidebar lists categories like All Categories (44), Infrastructure (3), Platform (41), and Boilerplates (5). The main area shows results for "label:lite". It includes sections for Infrastructure (Storage, Object Storage), Containers (Kubernetes Cluster, Container Registry), and Platform (Boilerplates). The "Internet of Things Platform Starter" service is visible under the Boilerplates section.

Click on Boilerplates in left side pane, then you will be provided with available services. Select "Internet of Things Platform Starter"

This screenshot shows the same IBM Cloud Catalog interface as above, but the focus is on the "Boilerplates" section. The sidebar on the left has a red box around the "Boilerplates (5)" link. In the main content area, the "Internet of Things Platform Starter" service is highlighted with a red box. Other services listed include Node.js Cloudant DB Web Starter, Node-RED Starter, Python Flask, and Ruby Sinatra.

The Boilerplate will have "**SDK for Node.js**" and "**Cloudant NoSQL DB**" services by default for us to use as services. It will also have **Node-RED** pre-installed for you.

Give a unique name to your application here and click on "**Create**". Application Names must be unique as they will be on a public domain.

App name: IoT Logistics Simulator

Host name: IoT-Logistics-Simulator

Domain: mybluemix.net

Choose a region/location to deploy in: US South

Choose an organization: Krishnakannepalli1994@gmail.com

Choose a space: dev

Selected Plan: SDK for Node.js™

Cloudant NoSQL DB

Internet of Things Platform

Need Help? Contact IBM Cloud Sales

Estimate Monthly Cost Cost Calculator

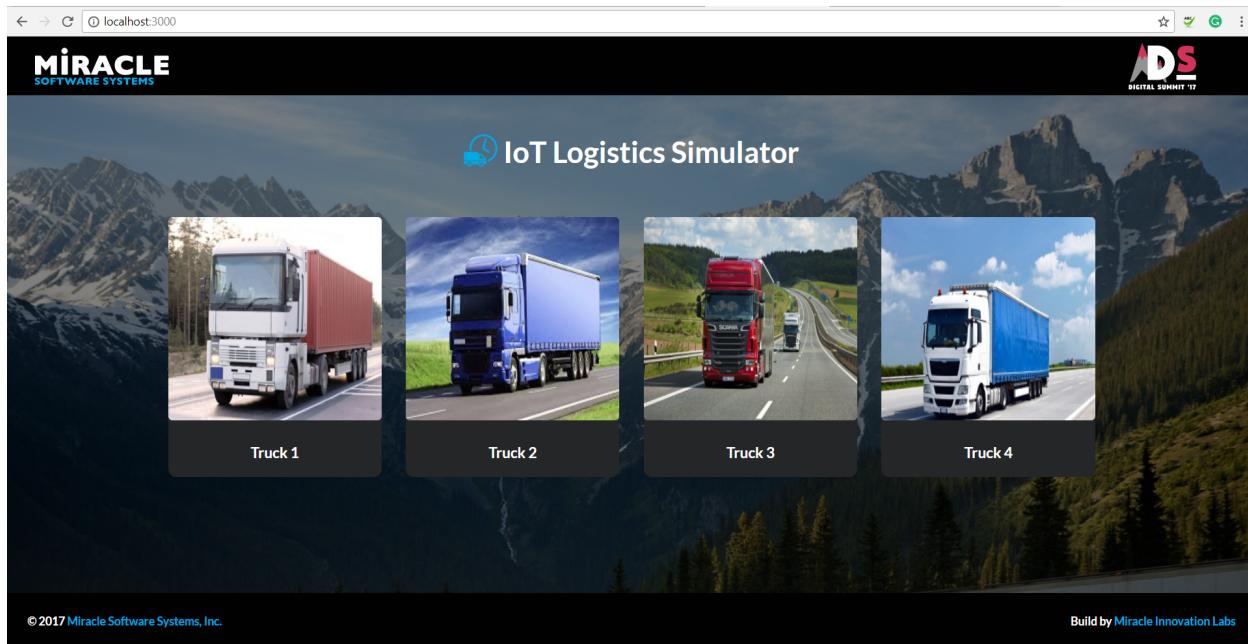
Create

Note : Once created, the application will take about 2 minutes for staging and start running.

#3 | Run the Logistic Simulator

You can download the Simulator file from the lab repository on GitHub and send the vehicle data to the IBM Watson IoT Platform.

Run the **index.html** file to go to the Logistic Simulator Dashboard and it can be used to simulate Truck events, for example - Speed, Fuel, Tyre Pressure and Location



#4 | Register your Truck (Simulator)

Go back to your Bluemix account and click on your application.

Name	Route	Memory (MB)	State
IoT Logistics Simulator	IoT-Logistics-Simulator.mybluemix.net	256	Awake (1/1)

Name	Service Offering	Plan
IoT Logistics Simulator-cloudantNoSQLDB	Cloudant NoSQL DB	Lite
IoT Logistics Simulator-iotf-service	Internet of Things Platform	Lite

Click on “Internet of Things Service”

Click on “Launch Dashboard” button. This will take you to your IoT Platform Organization space.

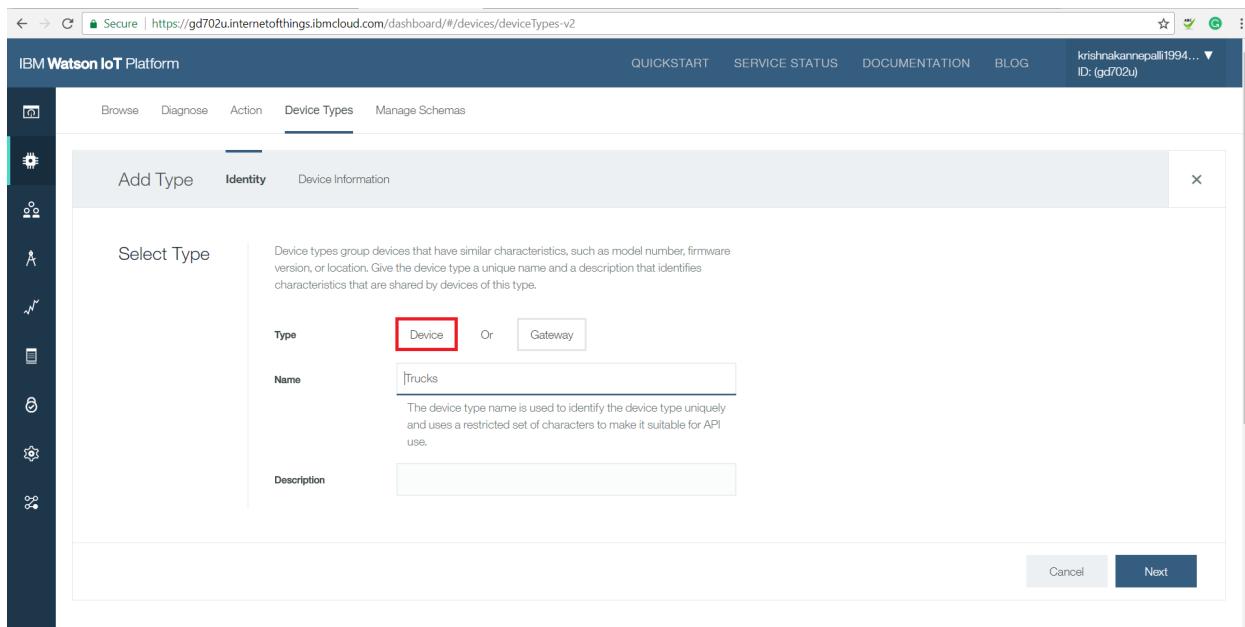
The screenshot shows the IBM Cloud IoT service interface. On the left, there's a sidebar with 'Manage', 'Plan', and 'Connections' options. The main area displays the organization details: 'IoT Logistics Simulator-iotf-service', 'Location: US South', 'Org: Krishnakannepali1994@gmail.com', and 'Space: dev'. Below this is a large icon representing a device connected to a network. A call-to-action section follows with the text 'Let's get started with Watson IoT Platform' and a 'Launch' button, which is highlighted with a red box. At the bottom right of the main area is a 'Docs' button.

As the Organization is new, there will be no registered devices so Click on “Device Type”

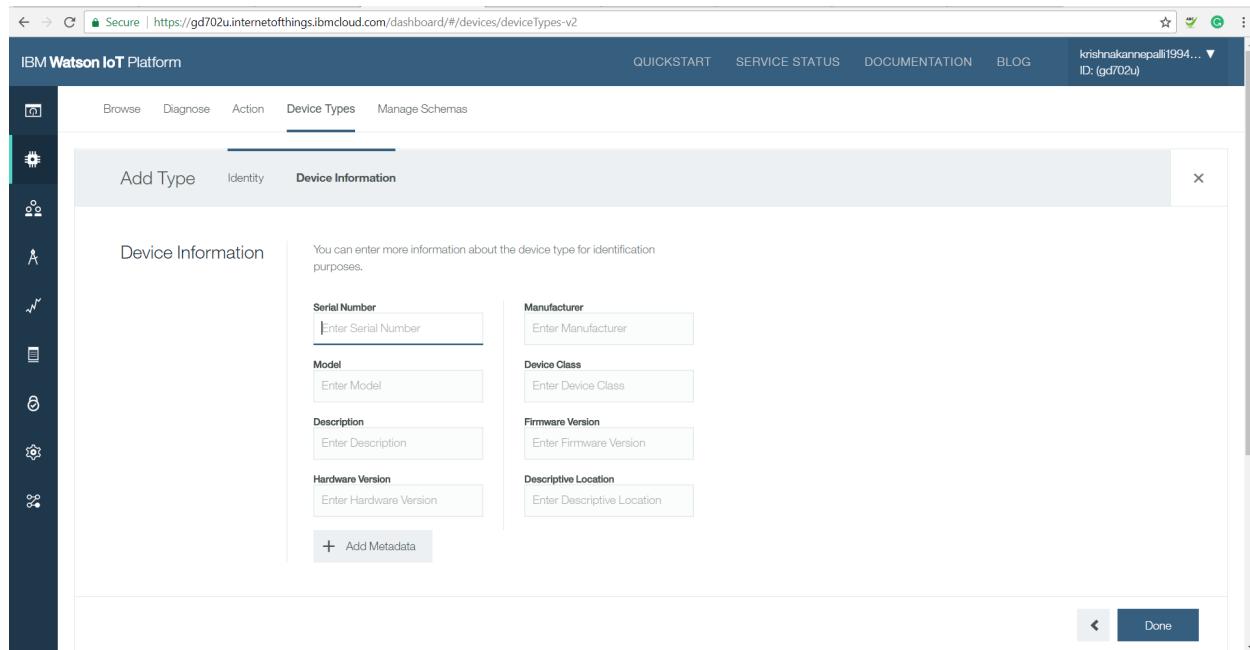
The screenshot shows the 'IBM Watson IoT Platform' dashboard. On the left, there's a sidebar with various icons. The main navigation bar includes 'QUICKSTART', 'SERVICE STATUS', 'DOCUMENTATION', 'BLOG', and a user profile section. Below the navigation is a menu bar with 'Browse', 'Diagnose', 'Action', 'Device Types' (which is highlighted with a red box), and 'Manage Schemas'. A '+ Add Device' button is located on the right side of the menu bar. The main content area is titled 'Browse Devices' and contains a table with the following columns: Device ID, Device Type, Class ID, Date Added, and Descriptive Location. The table currently shows '0 results'. At the bottom of the page, there's a message stating 'You don't have any devices.' followed by a 'Create a device.' button.

Observe that we have 2 options. As we want to create a Device Type and not a Gateway, click "**Device**".

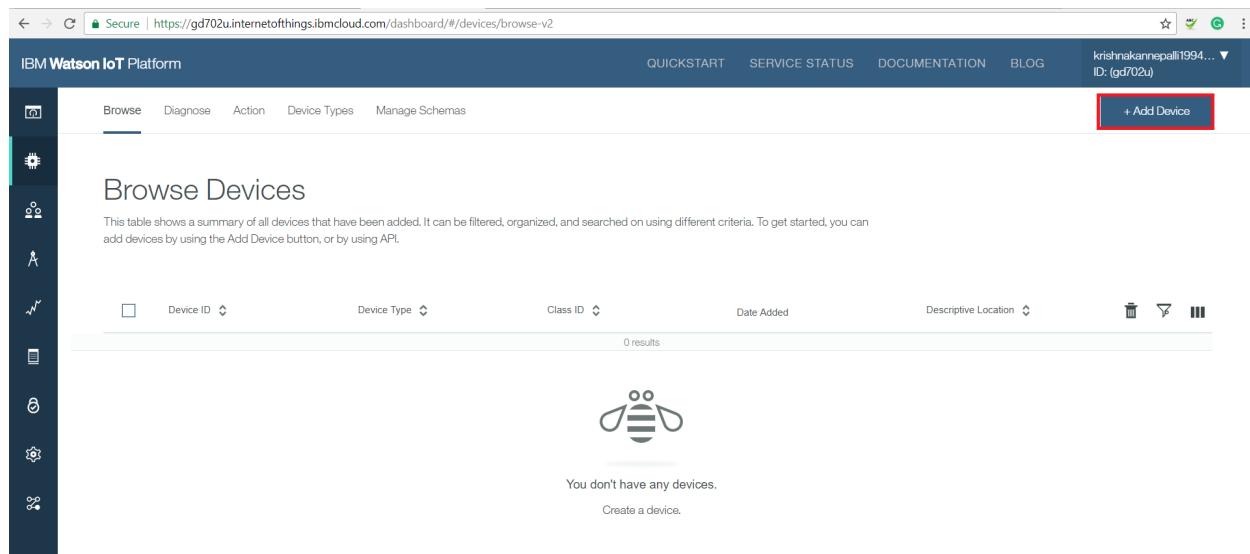
Specify a name for the device type to be added and give description(Optional). For example you can give "**Trucks**" for "Name" field. Click on "Next".



If you want to insert any "**Device Information**", you can insert it here(This is optional). Click on "**Done**"



Click on "Add Device" on top right corner.



When prompted, "Select the Existing Device Type" as "Trucks" and provide the Device ID, then click on Next.

Secure | https://gd702u.internetofthings.ibmcloud.com/dashboard/#/devices/browse-v2?add=Trucks

IBM Watson IoT Platform

QUICKSTART SERVICE STATUS DOCUMENTATION BLOG krishnakannepalli1994... ID: (gd702u)

Browse Diagnose Action Device Types Manage Schemas

Add Device Identity Device Information Security Summary

Identity

Select a device type for the device that you are adding and give the device a unique ID.

Select Existing Device Type: Trucks

Device ID: Truck1

Cancel Next

If you want to insert any "**Device Information**", you can insert it here(This is optional).

Secure | https://gd702u.internetofthings.ibmcloud.com/dashboard/#/devices/browse-v2?add=Trucks

IBM Watson IoT Platform

QUICKSTART SERVICE STATUS DOCUMENTATION BLOG krishnakannepalli1994... ID: (gd702u)

Browse Diagnose Action Device Types Manage Schemas

Add Device Identity Device Information Security Summary

Device Information

You can modify the default device information and enter more information about the device for identification purposes.

Serial Number	Enter Serial Number	Manufacturer	Enter Manufacturer
Model	Enter Model	Device Class	Enter Device Class
Description	Enter Description	Firmware Version	Enter Firmware Version
Hardware Version	Enter Hardware Version	Descriptive Location	Enter Descriptive Location

+ Add Metadata

Back Next

In the Next page, you can either add your own authentication token, or allow the IoT Platform to generate a token for you. If you want to add your own token, enter the token. Otherwise leave the field empty. Click on "**Next**".

The screenshot shows the 'Add Device' process in the IBM Watson IoT Platform. The 'Security' tab is selected. It provides two options for device authentication tokens: 'Auto-generated authentication token (default)' and 'Self-provided authentication token'. An 'Authentication Token' field contains 'miraclelabs'. A note below states that tokens are encrypted before storage.

Click on "Done".

The screenshot shows the 'Summary' tab of the 'Add Device' process. It displays the device information entered previously: 'Device Type' is 'Trucks' and 'Device ID' is 'Truck1'. A 'View Metadata' button is visible. A 'Done' button is located at the bottom right of the summary section.

Click on "Done", then the Device Type is created successfully. After registering, store the credentials for the registered device.

The screenshot shows the IBM Watson IoT Platform dashboard. The left sidebar has a dark theme with various icons. The main content area is titled 'Device Truck1'. On the left, under 'DEVICE DRILLDOWN', there is a list of options: Device Credentials, Connection Information, Recent Events, State, Device Information, Metadata, Extension Configuration, Diagnostics, Connection Logs, and Device Actions. The 'Device Credentials' section is expanded, showing the following details:

Organization ID	gd702u
Device Type	Trucks
Device ID	Truck1
Authentication Method	use-token-auth
Authentication Token	miraclelabs

Below this, a warning message states: 'Authentication tokens are non-recoverable. If you misplace this token, you will need to re-register the device to generate a new authentication token.' There is also a link: 'Find out how to add these credentials to your device'.

#5 | Configure the Simulator

For your simulator to communicate sensor events via MQTT to the IoT Platform we will need to configure it with the required endpoints and the registrations details. Go to your application and open the **server.js** file.

Update the details of the file with the configuration details as mentioned in the comments. The following are the details that you should have with you to modify the file,

```
var orgId = "<your-Organization-ID>"  
var deviceType = "<your-device-Type>"  
var deviceId = "<your-device-ID>"  
var deviceToken = "<your-device-Token>"
```

```
new 8 server.js
1 var iotf = require("ibmiotf");
2 var express=require('express');
3 var app = express();
4 const bodyParser=require('body-parser');
5 var server = require('http').Server(app);
6 var io = require('socket.io')(server);
7
8 var port=3000;
9 app.get('/',function(req,res){
10   | res.sendFile( __dirname+ '/index.html');
11 });
12 app.use(bodyParser.urlencoded({extended: true}));
13
14 app.use(bodyParser.json());
15
16 app.use(bodyParser.json());
17 app.use(express.static(__dirname + '/'));
18 app.use(function(req, res, next)
19 {
20   | res.header("Access-Control-Allow-Origin", "*");
21   | res.header("Access-Control-Allow-Headers", "*");
22   | next();
23 });
24
25 var config1 = {
26   "org" : "hdlxmj",
27   "id" : "Truck1",
28   "type" : "thing",
29   "auth-method" : "token",
30   "auth-token" : "digitalsummit@2107"
31 }
```

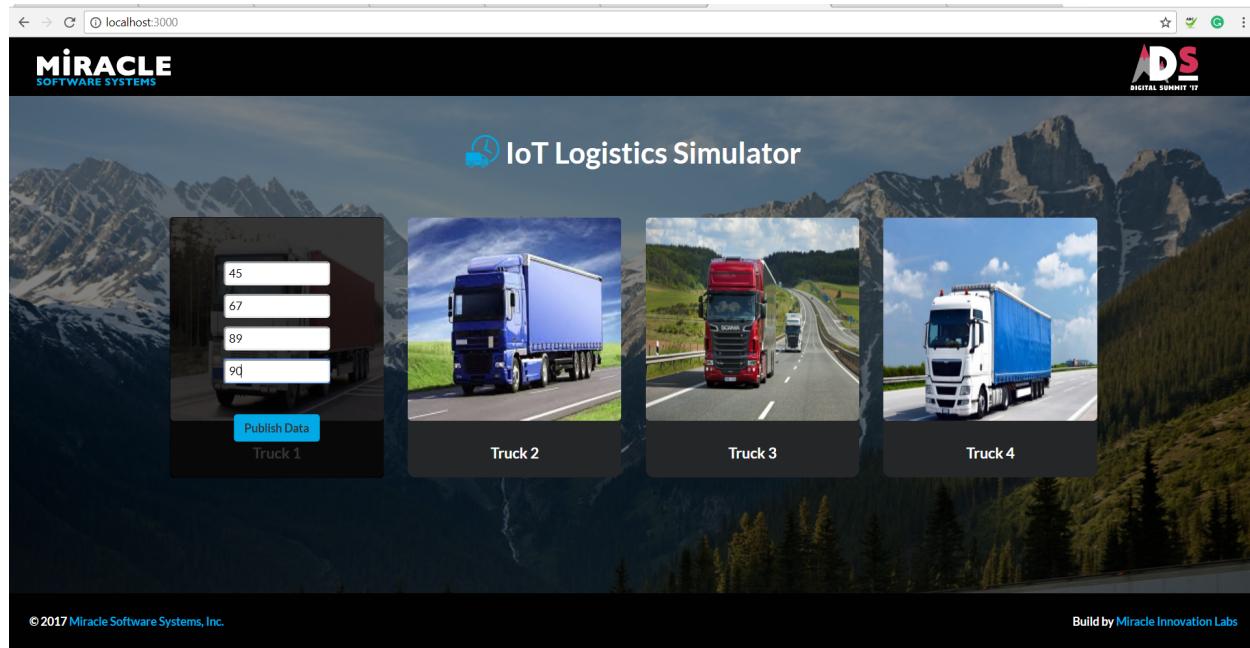
Save the file and re-load the index.html file in a browser. You can now return back to the IoT Dashboard and see that your device is connected to the cloud.

Open Command Prompt, and go to the application directory and run **app.js** file

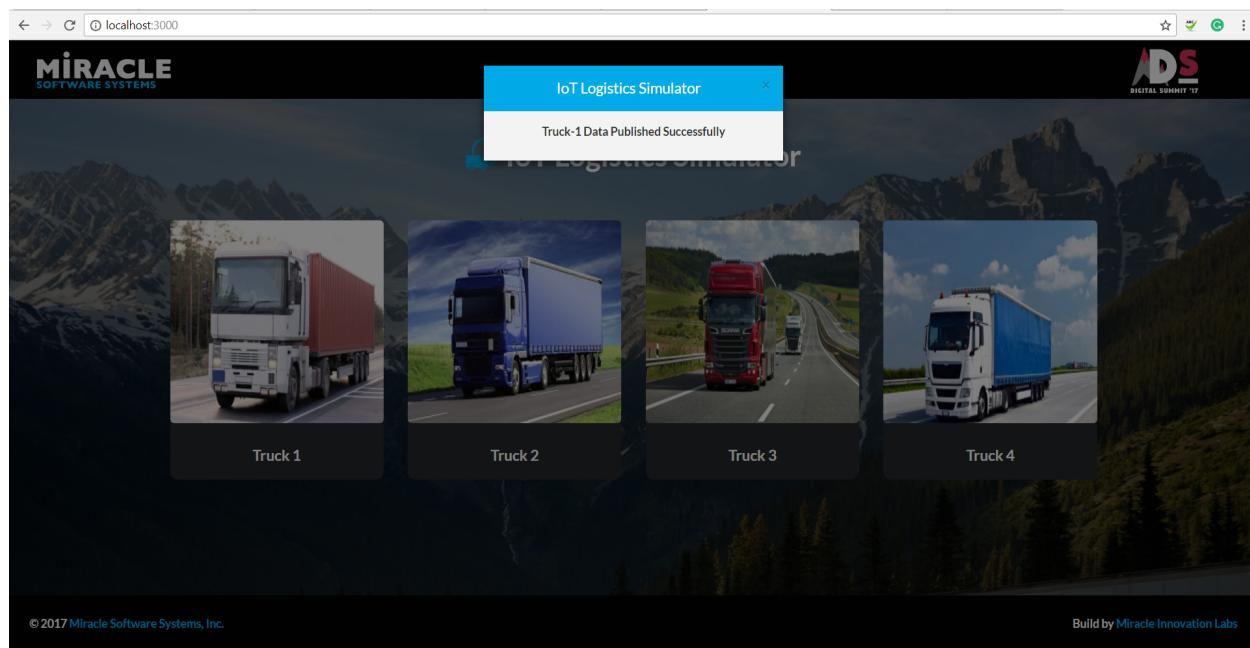
```
C:\Windows\System32\cmd.exe - node server.js
F:\IoT Logistics Simulator\simulator-20171205T144206Z-001\simulator>node server.js
server running at 3000
```

#6 | Publishing the data to Watson IoT platform

Go to Logistic Simulator dashboard. Place your mouse control over **Truck 1**. Fuel details will appear on top right corner. Give any values for the fields and click on “**Publish Data**”



If the data is published successfully to Watson IoT platform, then a pop-up appears.



Open Command prompt for viewing the json response

```
C:\Windows\System32\cmd.exe - node server.js
F:\IoT Logistics Simulator\simulator-20171205T144206Z-001\simulator>node server.js
server running at 3000
data : {"name":"Track1","speed":45,"fuel":67,"tyrepres":89,"loc":90}
Device1 is connected..
```

Go back to Watson IoT Dashboard for checking the JSON payload

The screenshot shows the IBM Watson IoT Platform dashboard. On the left, there's a sidebar with various icons. The main area has a header with 'IBM Watson IoT Platform' and navigation links like 'QUICKSTART', 'SERVICE STATUS', 'DOCUMENTATION', and 'BLOG'. A user profile is shown on the right. The main content area is titled 'Browse' and includes tabs for 'Diagnose', 'Action', 'Device Types', and 'Manage Schemas'. Below this, a table lists devices. One row for 'Truck1' is selected, showing details: Device ID: Truck1, Device Type: thing, Class ID: Device, Date Added: 4 Dec 2017 18:35. Under the 'Recent Events' tab, it says 'Showing Raw Data' and lists one event: Event: myevt1, Value: {"name": "Track1", "speed": 12, "fuel": 23, "t...}, Format: json, Last Received: a few seconds ago.

The screenshot shows the IBM Watson IoT Platform dashboard. At the top, there are links for QUICKSTART, SERVICE STATUS, DOCUMENTATION, and BLOG, along with a user ID (jp14231@gmail.com). A modal window titled "Event Payload" is open, displaying the JSON data for an event named "myevt1" received on 2017-12-07T18:23:26.099Z. The JSON content is:

```

1: {
2:   "name": "Track1",
3:   "speed": "12",
4:   "fuel": "23",
5:   "tyrepress": "3",
6:   "loc": "3333"
7: }

```

Below the modal, the main dashboard shows a list of devices, with "Truck1" selected. The "Recent Events" section shows a single event named "myevt1" with the same JSON payload, received 3 minutes ago.

#7 | Persisting your data with NodeRed and Cloudant

Go to your Bluemix Application Dashboard and click on **Route**

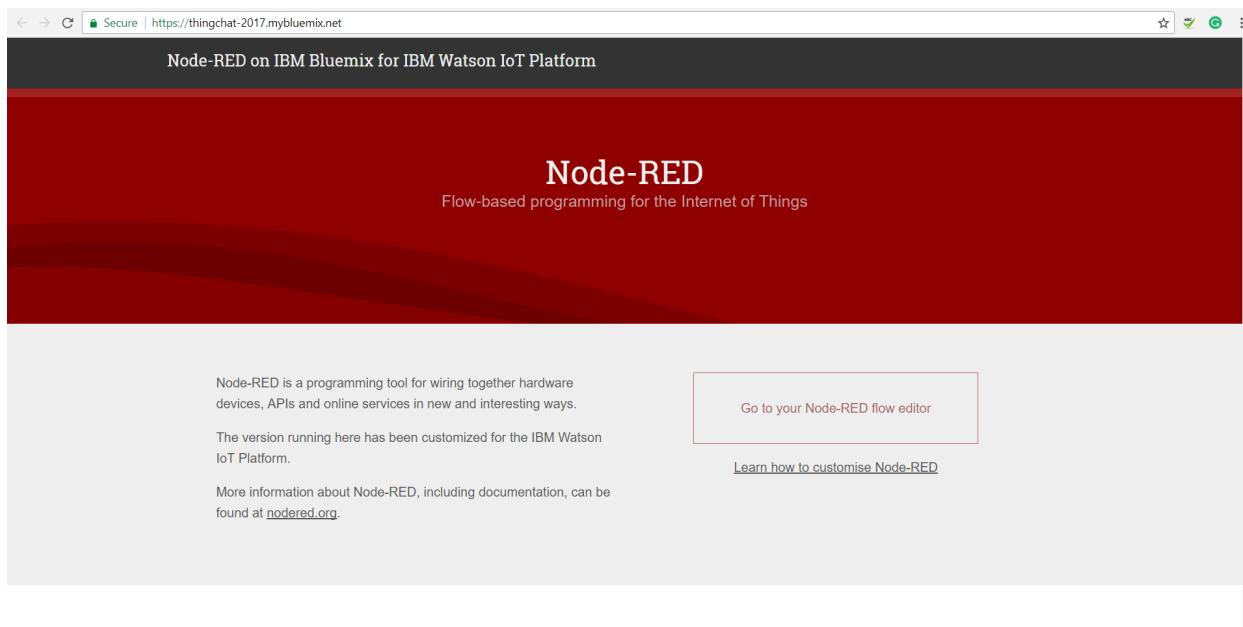
The screenshot shows the IBM Cloud Application Dashboard. The top navigation bar includes links for Catalog, Docs, Support, Manage, and a user icon. The main area is titled "Dashboard" and shows resource filters for REGION (US South), CLOUD FOUNDRY ORG (jp14231@gmail.com), and CLOUD FOUNDRY SPACE (dev). A search bar allows filtering by resource name. Below this, the "Cloud Foundry Apps" section displays two applications:

Name	Route	Memory (MB)	State
nodejs-cloudlab	nodejs-cloudlab.mybluemix.net	256	● Stopped (0/1)
ThingChat-2017	ThingChat-2017.mybluemix.net	256	● Awake (1/1)

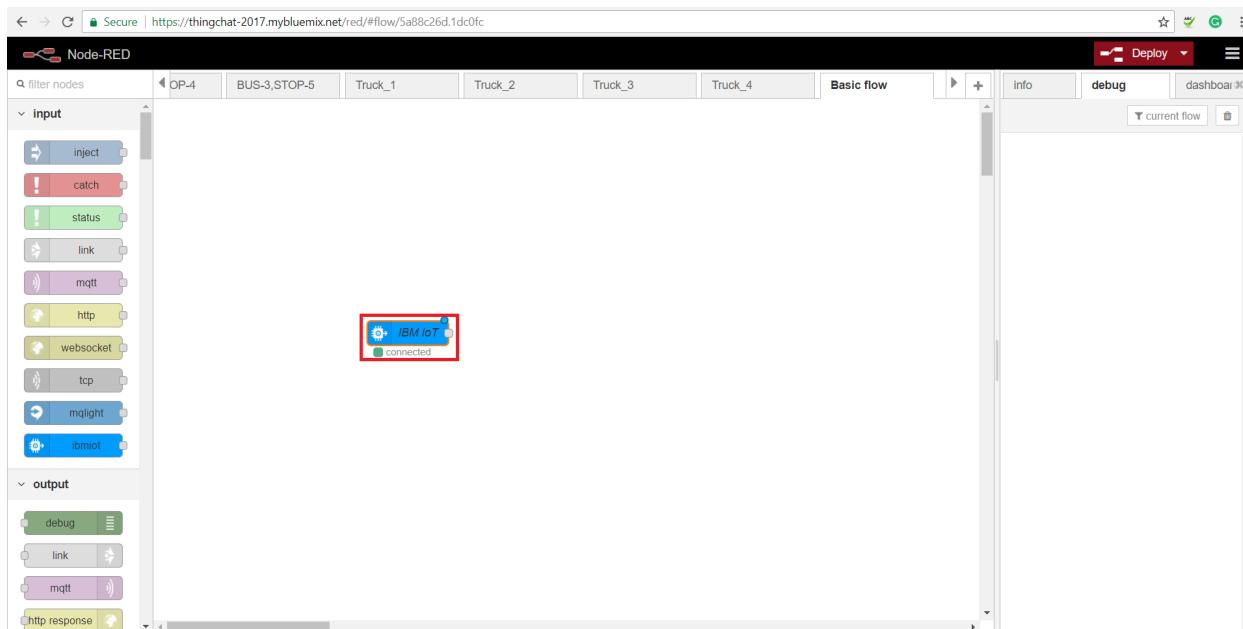
Below the application list is the "Cloud Foundry Services" section, which lists two services:

Name	Service Offering	Plan
ThingChat-2017-cloudantNoSQLDB	Cloudant NoSQL DB	Lite
ThingChat-2017-iotf-service	Internet of Things Platform	Lite

This will redirect you to Node Red tool in Bluemix. Click on "**Go to your Node Red Flow Editor**"

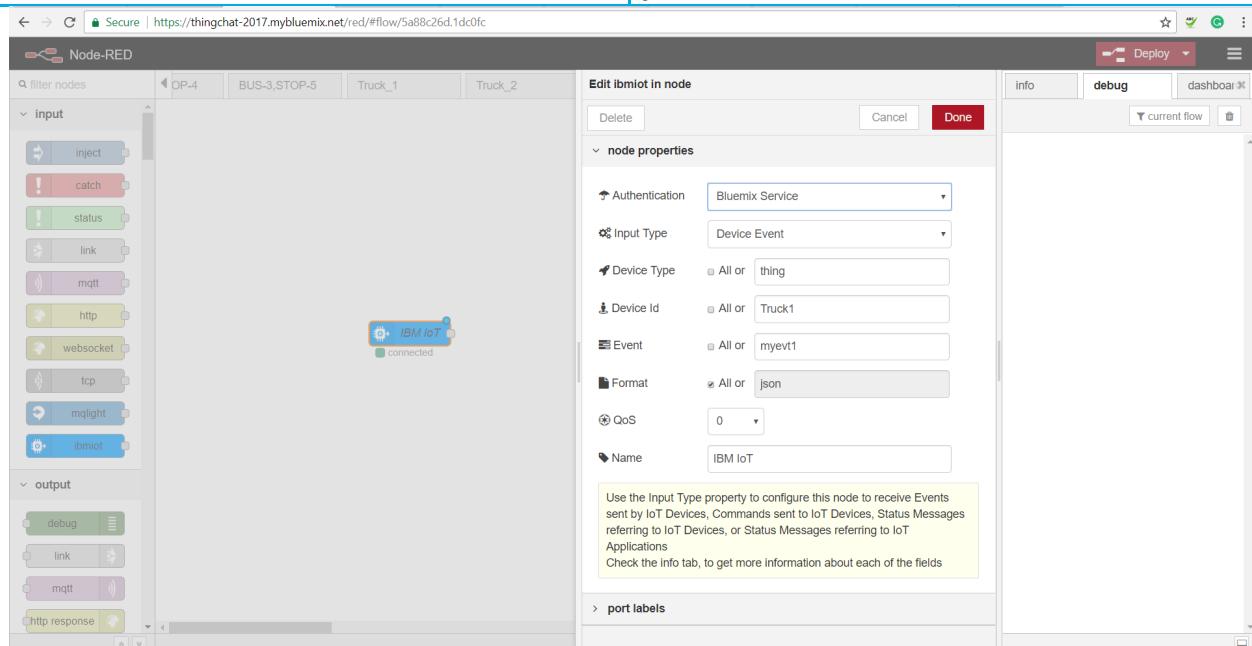


Node Red Flow sheet will be opened with a set of input and output nodes. Drag and drop "**IBM IoT**" node onto the flow sheet.

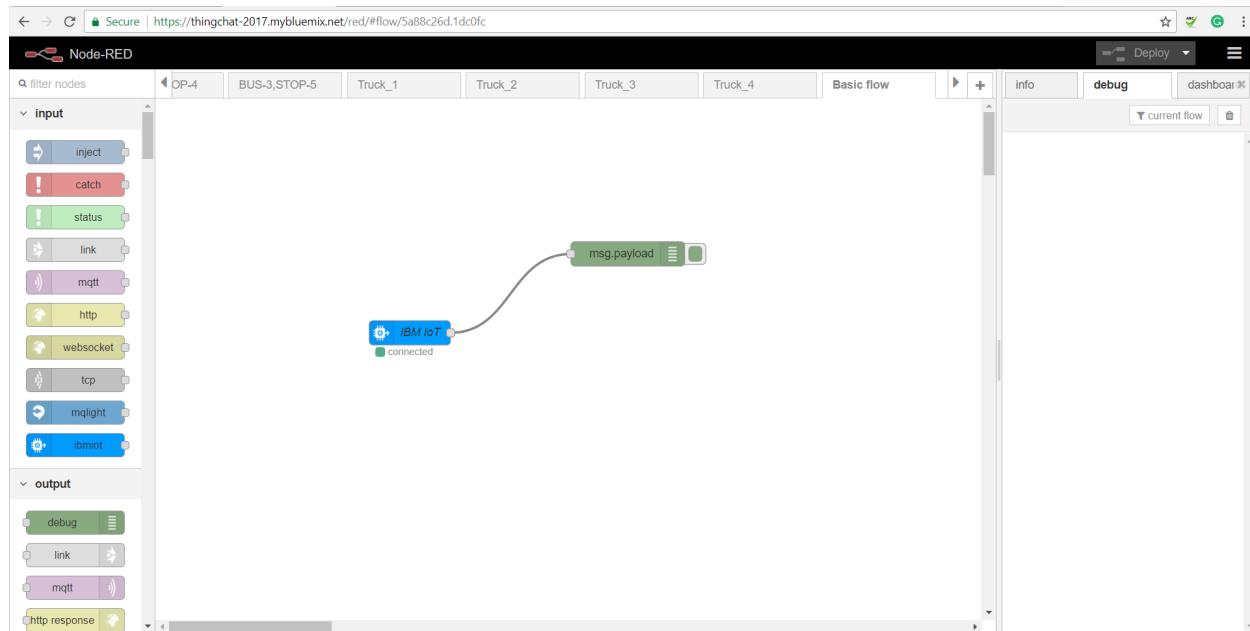


Double click on the "IBM IoT" node. Give the following for prompted fields,

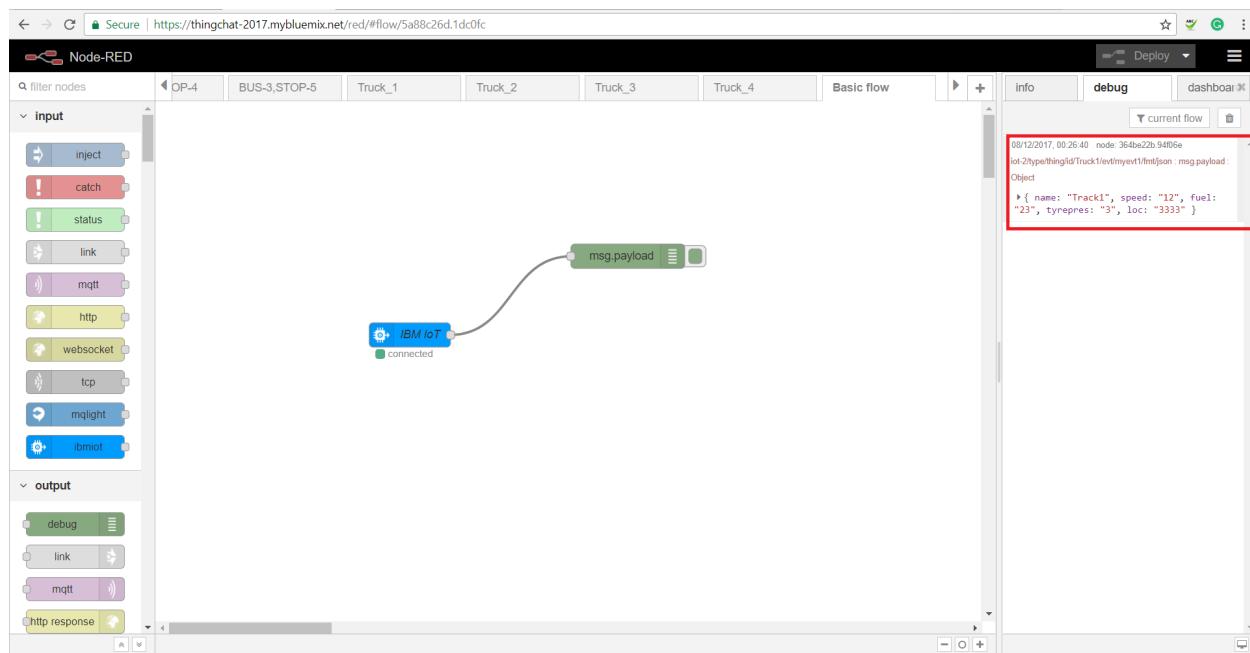
Field	Value
Authentication	Bluemix Service
Input Type	Device Event
Device Id	<Your-Device-ID(MAC ID)>
Format	json



Drag and drop a "debug" node from the list of Output nodes present in the left side menu. Give connection to both the nodes and click on "Deploy".



Go to Logistic Simulator dashboard. Send the events and click **Publish Data**, then the data will be sent to Node-RED



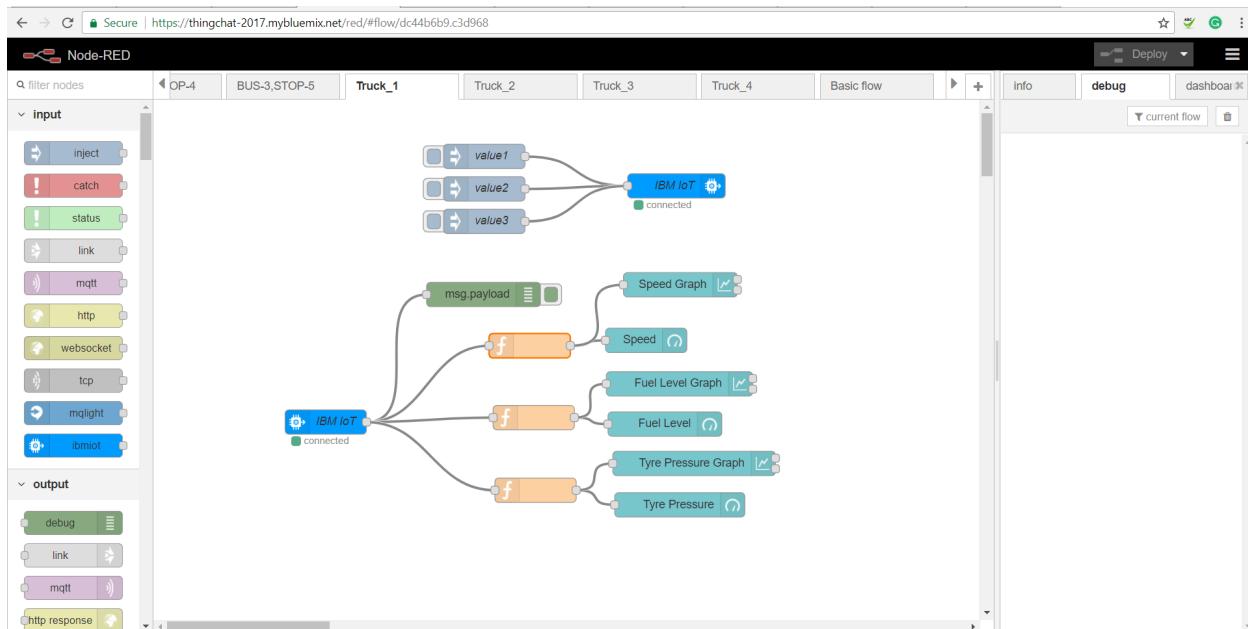
#8 | Create Dashboard using Node-RED

Dashboard module provides a set of nodes in Node-RED to quickly create a live data dashboard. Install Dashboard node using,

npm i node-red-dashboard

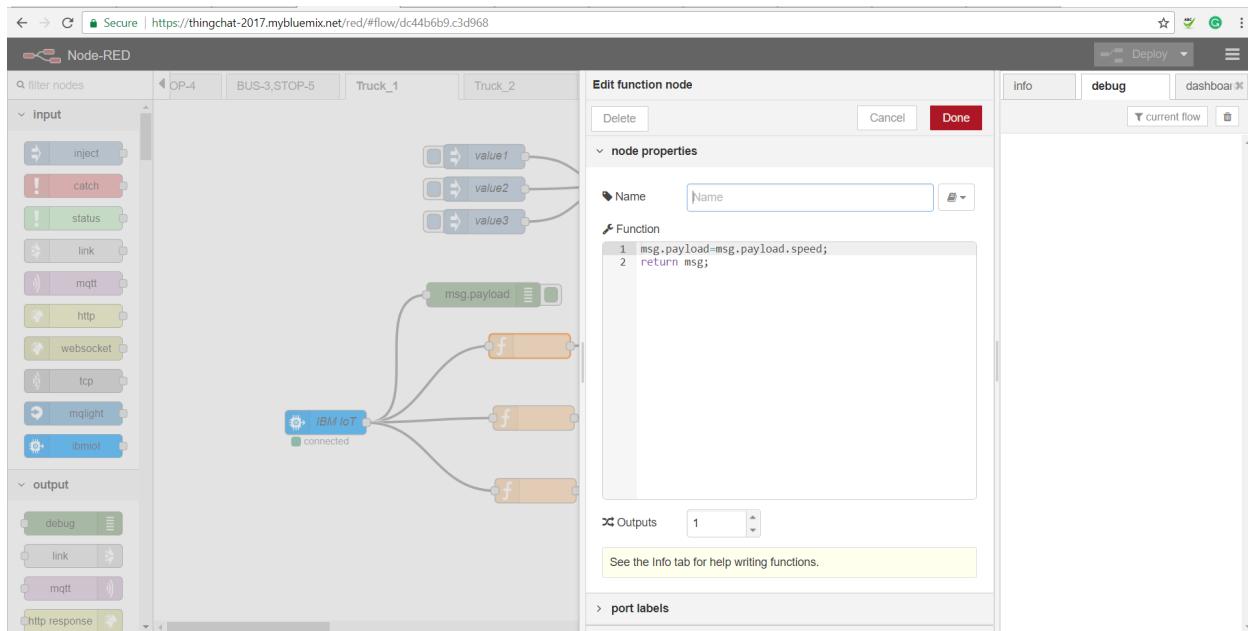
The dashboard layout should be considered as a grid. It contains all the elements which are used for creating Dashboard - offering graphs, gauges, basic text as well as sliders and inputs.

We'll create a simple dashboard element that displays the values of the truck in the form of graph and gauges. Drag and drop the gauges and graphs along with one function as below for 3 different parameters.

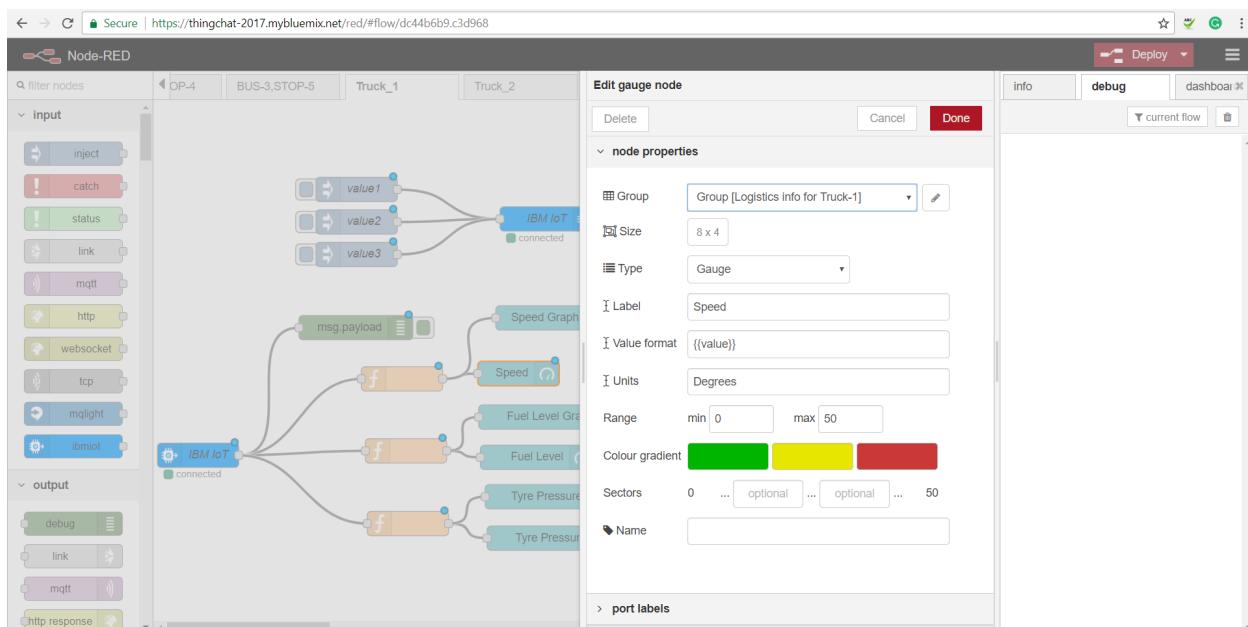


Click on the Function node and copy the below snippet,

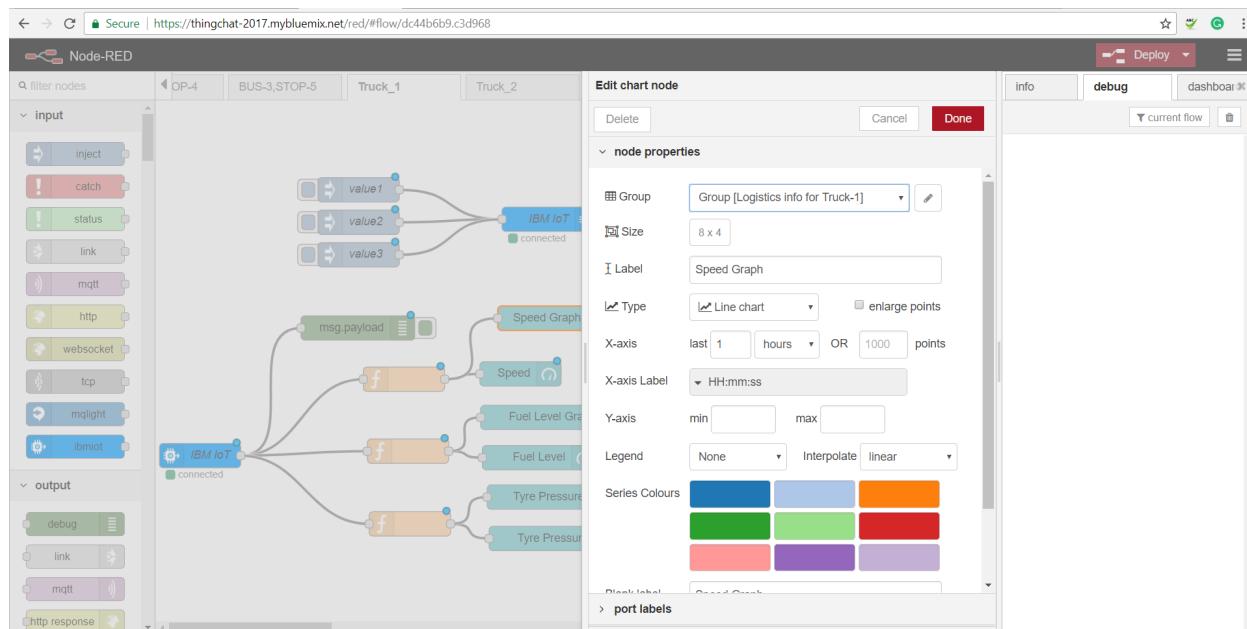
```
msg.payload=msg.payload.speed;  
return msg;
```



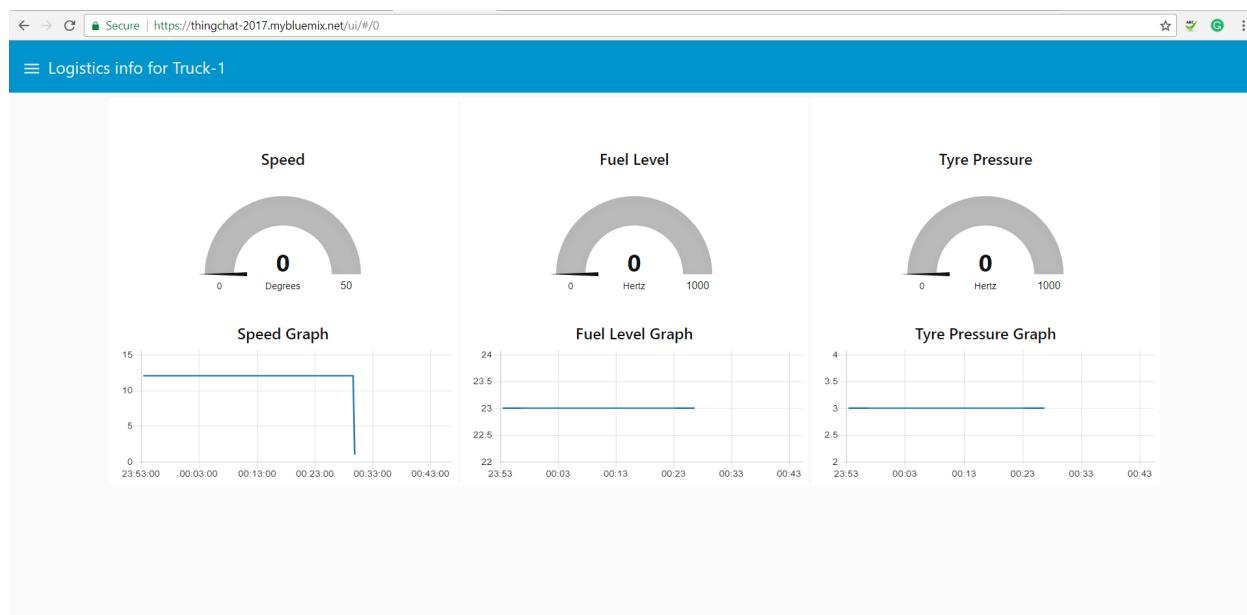
Click on Graph node and add the below details as shown



Click on Gauge node and add the below details as shown



Open <http://<application-name>/ui> , for viewing the dashboard



Now, send the data from the Logistic Dashboard as above, you can see the change in Gauges and Graphs.

