# USER INTERFACE CREATION WITH Node-RED

## Prerequisites

- 1) Have an account on Bluemix
- 2) Have created an Internet of Things Platform on Bluemix
- 3) Have created a device on Internet of Things Platform, and have its Device ID and its Authentication Token
- 4) Have generated an API KEY on Internet of Things Platform, and have its API Authentication Token
- 5) Have configured SS10680 gateway correctly

#### Procedure

- 1) Login on Bluemix (http://www.bluemix.net/) with your credentials
- 2) Go to 'Catalog' following this link: <u>https://console.ng.bluemix.net/catalog/</u>
- 3) From left menu, select 'Platform' and then click on 'Node-RED Starter'

☰ ਁ IBM Cloud			Catalog Docs Support Manag	, Q
Try the best of the Catalog for free with no The Lite filter is enabled. Remove the filter to :				×
All Categories (45)	Q labebilte			Filter
Infrastructure (3) Compute Storage (1) Network	Boilerplates Get started with a new app, now.			
Security Containers (2) VMware	Internet of Things Platform Starter Get started with IBM Watson IoT platform using the Node-RED Nodeja sample application. With the Ser Use IBM	Node.js Cloudant DB Web Starter Use the Cloudant NoSQL DB service with the 'SDK for Node Ja <sup>101</sup> runtum.	Node-RED Starter This application demonstrates how to run the Node-RED open-source project within IBM Bluems. (b)	
Platform (42) > Boilerplates (5) APIs (1) Application Services Blockchain Cloud Foundry Apps (10)	Python Flask A simple Python Flask application that will get you up and running quickly.	Ruby Sinatra Develop a Ruby web application using the Sinatra framework.		
Data & Analytics (8) DevOps (4) Finance Functions	APIs APIs published in your org or shared from API Management.			
Integrate (2) Internet of Things (1) Mobile (2) Network Security (1) Watson (8)	API Connoct Crea, pestisce, applica ed esegue le API.			

4) Fill the requested fields and then click on 'Create' button for create a Node-RED instance

Node-RE	ED Starter	App name:					
This application	demonstrates how to run the Node-RED open-source	sielcoiotapp					
project within IB		Host name:		Domain:			
Lite Commu	unity	sielcoiotapp		eu-gb.mybluemix.net			•
View Docs		Choose a region/location to deploy in:	Choose an organization:		Choose a space:		
VERSION	0.7.0	United Kingdom	- sielcoloT		SSIoT		
TYPE	Boilerplate						
REGION	United Kingdom, Germany, US South, Sydney	Selected Plan:					
		SDK for Node.js™		Cloudant NoSQL DB			
		Lite	÷	Lite			-
		.js 🔾					
		SDK for Nodejs™ Cloudant NoSQL					
		DB					

So, now, Node-RED instance creation starts.

At the end of the procedure, the following page is shown to you

☰ Ö IBM Cloud		Catalogo Documenti Supporto Gestisci 🔘
Introduzione Panoramica Runtime	Applicazioni Cloud Foundry       /         SielColotapp       L'applicazione è operativa ma verrà disattivata se l'inattività di sviluppo si protrae per altri 10 giorni.       Visita URL applicazione         Organizzazione: sielcolo7       Ubicazione: United Kingdom       Spazio: SSIoT	Potte - C O
Connessioni Log Monitoraggio Gestione API	Start coding with Node-RED         Witting aggiornaments: 2017-08-15         Image: Comparison and started, click on the Routes URL or enter the following URL in a browser:         Image: Comparison and the Routes URL or enter the following URL in a browser:         Image: Comparison and the Routes URL or enter the following URL in a browser:         Image: Comparison and the Routes URL or enter the following URL in a browser:         Image: Comparison and the Routes URL or enter the following URL in a browser:         Image: Comparison and the Routes URL or enter the following URL in a browser:         Image: Comparison and the Routes URL or enter the following URL in a browser:         Image: Comparison and the Routes Comparison and the routes or enter the following URL in a browser:         Constraining your Node-RED Instance         Image: Comparison and the Route Comparison and the interface.         Image: Comparison and the Route Comparison and the interface.         Image: Comparison and the Route Comparison and the interface.	
۲	<ul> <li>Download and extract your starter code to set up your development environment.</li> <li>i communato stratter code</li> <li>Change to your new directory:         <ul> <li>d directory_name</li> <li>common to Bluemio@.</li> <li>common to Bluemio@.</li> <li>f dpi https://dpi.eu-gb.bluemin.set</li> <li>Log in to Bluemix.</li> <li>i of login ~u</li></ul></li></ul>	

5) Click in 'Visit application URL'

6) The first time that you execute the Node-RED instance, you must define some properties for it. On first page, click on 'Next' button

Welcome to your new Node-RED instance on IBM Bluemix
We know you're eager to start wiring up your flows, but first there are a couple of tasks you should do:
<ul><li>Secure your Node-RED editor</li><li>Browse available IBM Bluemix nodes</li></ul>
Previous     Next

٦

7) In case you want to allow access only authorized person to Node-RED instance select 'Secure your editor so only authorised users can access it' and enter a username and a password for it.

In case you want to allow access to any user but do not allow them to do any changes select 'Allow anyone to view the editor, but not make any changes'.

In case you want to allow access to any user and allow them to do changes select 'Allow anyone to access the editor and make changes'.

N.B. Last choice is not recommended.

At the end, click on 'Next' button

Secure your ed	itor so only authorised users can access it
Username	admin
Password	••••••
○ Not recommend	ded: Allow anyone to access the editor and make changes

# 8) Read general information and, then, click on 'Next' button

Browse available IBM Bluemix	nodes	
There are lots of nodes available from the commu application. The list below is just a small selection	unity that can be used to add more capabilities to your h.	
You can find many more nodes on the Flow Librar	ſ <b>y</b> .	
You can use the Palette Manager built into editor also edit your application's package.json file ar	to search for and install nodes. Alternatively, you can ad adding them to the dependencies section.	
node-red-dashboard	node-red-contrib-ibm-wiotp-device-ops	
Quickly create dashboards driven by Node-RED	Perform device and gateway operations using the Watson IoT Platform	
node-red-contrib-iot-virtual-device	node-red-contrib-objectstore	
Simulate device behavior and use it to run many device instances	Store, delete and restore objects in the ObjectStore service	
	node-red-contrib-ibmpush	

Finish the install
You have made the following selections:
Secure your editor so only authorised users can access it
You can change these settings at any time by setting the following environment variables via the Bluemix console:
<ul> <li>NODE_RED_USERNAME - the username</li> <li>NODE_RED_PASSWORD - the password</li> <li>NODE_RED_GUEST_ACCESS - if set to `true`, allows anyone read-only access to the editor</li> </ul>
Previous     Finish

10) When the install is finished, click on 'Go to Node-RED flow editor' to access the Workflow Editor

11) Enter Username and Password (if you made the first choice at point 7) and click on 'Login'

Node-RED			
	Node-RED	Username: Password: Login	

If login went well, you can see the following page

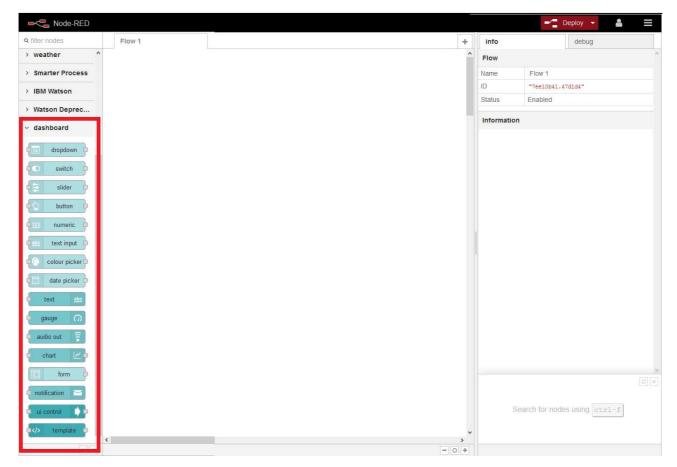


12) Install 'node-red-dashboard' packet, that we will use to build the user interface

Click on click on 'Manage Palette' label. On the panel that opens click on 'Install' tab (1), on the search bar enter the keyword 'dashboard' (2), find 'node-red-dashboard' packet in the list and click on 'install' to install it.

Node-RED	- 1910 -		105	🛫 Deploy 🝷 🛔 🚍
Q filler nodes Flow 1	User Settings		info	debug
v input ^		Close	Flow	~
⇒ inject		Nodes 1 Install	Name	Flow 1
- inject	View		ID	"7ee10b41.47d1d4"
catch D	Keyboard	sort: a-z recent	Status	Enabled
status	2	Q dashboard 7/124	× Informa	tion
	Palette	P node-red-contrib-freeboard	^	
		Freeboard Dashboard Node for Node-RED		
mqtt O		💊 0.0.7 🏥 1 year, 6 months ago install		
http		🗑 node-red-contrib-netatmo-dashboard 🗷		
websocket		Get JSON payload for NetAtmo dashboard. See http://netatmo.com, https://dev.netatmo.com/doc		
WEDSOCKET C		0.2.3      11 months ago     install		
() top ()				
🗢 malight 👂		node-red-contrib-polymer      Polymer based dashboard UI for Node-RED		
Co torviot		🗣 0.0.21 🋗 3 days ago install		
		🗑 node-red-contrib-uibuilder 🖪		
~ output		Easily create web UI's for Node-RED using any (or no) front-end library. JQuery and		
debug E		normalize.css included by default but change as desired.		
uccog =		• 1.0.2 ш 1 бау адо	-	
tink 👘		📦 node-red-dashboard 🗭		
( mątt	3	A set of dashboard nodes for Node-RED 2.6.2		
Chttp response				
		node-red-dashboard-es     C		
websocket		A set of dashboard nodes for Node-RED 2.6.0 mm 2 months ago install		
c top				
udo A		node-shri-dashboard      A set of dashboard nodes for Node-RED		
		2.3.11-beta      11 9 months ago		
e mqlight			-	
tuitio 🚗 👻 <			17	
A ¥	100		0	

If the installation of the package went well, on the left bar in the main window, you can find the '**dashboard**' menu with all its nodes.



13) Install 'node-red-contrib-web-worldmap' packet, that will use for create a map in the user interface

Click on click on 'Manage Palette' label. On the panel that opens click on 'Install' tab (1), on the search bar enter the keyword 'worldmap' (2), find 'node-red-contrib-web-worldmap' packet in the list and click on 'install' to install it.

Node-RED		Deploy -
Q siter nodes Flow 1	User Settings	info debug
~ input	Close	T all nodes
inject 0	View Nodes Install	
1 catch o	2 q worldmap	
status 🦻	Palette     1.1.7	
ink o	y 3 nodes install	
(i) mqtt		
http 9		
websocket		
1 top		
malight		
Contract Con		
~ output		
debug 📕		
6 link 9		
C mgt		
Chttp response (		
websocket		
C top		
udp 👔		
mqigit 🤤		
hulin v C		

If the installation of the package went well, on the left bar in the main window, you can find the '**location**' menu with all its nodes.

Rode-RED					■⁄∎ Deploy	- 🌡	Ξ
Q filter nodes	Flow 1	+		info	debug		
^			^			T all nodes	
button p							^
dropdown p							
switch							
slider P							
numeric 🔶							
text input							
date picker							
colour picker							
form							
text 🔤			ľ				
gauge 🕥							
chart 🗠							
audio out							
o notification							
ai control							
template							
✓ location							
🔹 worldmap 📀							
🔷 worldmap							
tracks			~				
L N	<						P

Now, let's start with the user interface building.

14) From '*input*' menu in the left bar, select *ibmiot* node and drag & drop it into 'Flow 1' area. Once the node is in the working area, double click on it and the following panel will be shown

Node-RED					🗾 Deploy 👻 👗 🗮
Q filter nodes Flow 1	Edit ibmiot in node			info	debug
v input ^	Delete		Cancel Done	Node	^
inject	v node properties	5		Name	ibmiot
				Туре	ibmiot in
L catch	Authentication	API Key	1	ID	"987c955c.824358" show more •
status of ibrition	a API Key	Add new ibmiot 🗸 🌶	2	Informatio	
link	📽 Input Type	Device Event	3		that can be used with Watson IoT Platform to receive
) mqtt p http	🖋 Device Type	All or +	4	receive star	t from devices, receive commands sent to devices, or tus updates concerning devices or applications. It
websocket	🛓 Device Id	All or device id e.g. ab12cd231a21	5		n object called msg and sets msg.payload to be a aining the payload of the incoming message.
top	Event	All or +	6	The value of	of "Device Id" is stored in msg.deviceId
	as Lyon	M MI OI T	0	The value of	of "Application Id" is stored in msg.applicationId
mqlight	Format	□ All or json	7	The value of	of "Device Type" is stored in msg.deviceType
🔂 ibmiat 🗣	(B) QoS	0 ~	8	The value of	of "Event Type" is stored in msg.eventType
> output					of "Command Type" is stored in msg.command Type
	Name Name	ibmiot	9		of "Format" is stored in msg.format. This node on, buffer and other types. When the format is set to
> function	Use the Input Type	e property to configure this node to receive Events		json, this n	ode parses the incoming data using JSON parse()
> social		es, Commands sent to IoT Devices, Status Message evices, or Status Messages referring to IoT	es	300000000000000000000000000000000000000	s. When the buffer object is received by this node, it contents as is without any conversion. And for
> storage	Applications				ypes, this node outputs the message in String.
> analysis	Check the info tab	b, to get more information about each of the fields			kstart mode, value for QoS (Quality of Service) and keep alive interval value can be specified.
> advanced					y of Service (QoS) level is an agreement between
> weather				sender and	I receiver of a message regarding the guarantees of Y
> Smarter Process				Move ti	ne selected nodes using the + + + and -+
> IBM Watson					keys. Hold to nudge them further
> Watson Deprec v c	> port labels			_	
A V					

Enter the following parameter:

- 1. Authentication: select 'API KEY'
- 2. API KEY: click on . In the panel that opens, you can configure how the Node-RED application communicates with Watson IoT Platform

💩 API Key		
API Token		
Server-Name	orgid.messaging.intern	etofthings.ibmcloud.com
Scalable [	Application ID	
O Keep Alive	60 Seconds	☑ Use Clean Session

- A. Name: enter a name for the connection or leave it blank
- B. API Key: enter in this field the API Key of the APP created on Watson IoT
- C. API Token: enter in this field the Authentication Token on the APP create on Watson IoT
- D. Server-Name: enter in this field the address of the server used to communicate with Watson IoT. The address has the following syntax ogld.messaging.internetofthings.ibmcloud.com

where *orgId* is the unique organization ID that was generated when you registered the service instance

Click on Update button

- 3. Input Type: select 'Device Event'
- 4. Device Type: if you check 'All', all the device types declared on Watson IoT are allowed. If you want allow only a device type, don't check 'All' and enter into the text field the device type that you want to connect
- 5. Device Id: if you check '*All*', all the devices that belong to the device types chosen are allowed. If you want allow only a device, don't check 'All' and enter into the text field the device id that you want to connect
- 6. Event: if you check '*All*', all the events of the allowed devices are considered. If you want to handle only an event, don't check '*All*' and enter into the text field the event that you want.
- 7. Format: if you check '*All*', all the formats allowed by Watson IoT Platform (json, xml, text) can be used. If you want to use a specific format, don't select '*All*' and enter it into the text field.
- 8. QoS (Quality of Service):
  - 0: the message is delivered at most once, or it is not delivered at all. Its delivery across the network is not acknowledged.

- 1: the message is always delivered at least once. If the sender does not receive an acknowledgment, the message is sent again with the DUP flag set until an acknowledgment is received. As a result, receiver can be sent the same message multiple times, and might process it multiple times.
- 2: the message is always delivered exactly once.
- 9. Name: enter the node's name

At the end of node's configuration, the result is similar to the following

Delete	Cancel	)one
v node properties		one
Authentication	API Key	
🔩 API Key	Connection to Watson IoT	
🕸 Input Type	Device Event	
Y Device Type	All or +	
🛓 Device Id	All or gateway	
Event Event	All or temperature1	
Format	All or json	
🛞 QoS	0 ~	
Name	Reading temperature1	

To continue, click on button

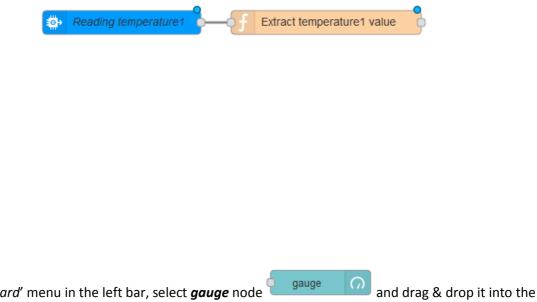
Flow 1	Flow 2		Edit function	node	
			Delete		Cancel
			✓ node prop	erties	
	Read temperature t	function	Name	Extract temperature1 value	
			Function		
			1 msg.p 2 retu	<pre>payload = msg.payload.value; rn msg;</pre>	

function

To continue, click on **Done** button.

16) Link the two nodes

Put the mouse over the grey square of the 'Read temperature1' node (the square becomes orange) and move the mouse over the grey square of 'Extract temperature 1 value' node. The result of the operation is shown below



Node-RED								=	Deploy 🔻	<b>≜</b> ≡
Q filter nodes	Flow 1	Flow 2		Edit gauge nod	2			info	debug	dashboa 🕷
> input ^				Delete		Cancel	Done	Node		^
> output				v node proper	ties			Туре	ul_gauge	
> function						_		ID	"dea3b987.d27	918"
> social				I Group	Add new ui_group	~ /	1			
> storage	Reading temperature t	Extract temperature 1 value	Gauge	Size	auto 2			Information	type widget to th	
> analysis	Connected			🔳 Туре	Gauge - 3			interface.	type woget to th	e user
> advanced				I Label	Gauge		4		load is searche and is formatted	
> weather							6	accordance w	th the defined Va	lue Format,
> Smarter Process				1 Value format	{{value}}		5	which can then filters.	be formatted us	sing Angular
> IBM Watson				<b>I</b> Units	units		6		{{value   num	
				Range	min 0 max 10 <b>7</b>			will round the append a % si	value to one deci gn.	imal place and
> Watson Deprec				Colour gradient		8			each of 3 secto	
<ul> <li>dashboard</li> </ul>						10 9			the gauge will ble ours should be s	
button				Sectors	0 optional optional	10 9	1.401	(#rrggbb) form		
dropdown b				Name Name			10		numbers for the s anges per sector	
switch								specified the o total range.	olours are blend	ed across the
slider o								The gauge ha	s several modes	
numeric b								(a) (a)	compass and wa	
text input									also be set by a tting the field to	the name of Y
date picker									1	
colour picker								ctrl-sp	ace will toggle this sidebar	e the view of
form				> port labels					1010 0100001	
×	<									

Edit the node as below:

1. Group: select 'Add new ui\_group' and click on . In the panel that opens we can define the 'group' and the 'page' to insert the gauge

jauge > <b>Add</b>	new dashboard group con	ifig node
		Cancel Add
Name	SS10014	Α
🆽 Tab	Home	~ Ø B
↔ Width	6 <b>C</b>	
	Display group name	D

- A. Name: enter the name of the group. For this example, insert the name of the device 'SS10014'
- B. Tab: select 'Home'. If you want to add a new 'page', select 'Add new ui\_tab' and then click on .

C. Width: enter the group's width

D. Display group name: check the checkbox if you want to show the name of the group in the user interface

At the end, click on Add button.

- 2. Size: enter the dimensions of the object in the UI. For this example, leave 'auto'
- 3. Type: select 'Gauge'
- 4. Label: enter '*Temperature 1*'
- 5. Value format: indicates the format of the value. Leave {{value}}.
- 6. Units: indicates the unit of measure. Enter °C.

- 7. Range: enter the min. possible value and the max. possible value. Enter -60 as min. value and 60 as max. value
- 8. Colour gradient: for this example, create only one green sector
- 9. Sectors: leave the intermediate values blank
- 10. Name: enter the name of the node. In this case 'Temperature1'

At the end, the result is as below

	Edit gauge node	9		
	Delete		Cancel	Done
	v node proper	ties		
	I Group	SS10014 [Home]	~	
	ច្រាំ Size	auto		
	🔳 Туре	Gauge		
	I Label	Temperature 1		
	∃ Value format	{{value}}		
	1 Units	٦° ا		
	Range	min -60 max 60		
	Colour gradient			
	Sectors	-60 optional optional	. 60	
	Name Name	Temperature 1		
To continue, click on	<sup>Done</sup> buttor	1.		

18) Link the 'Extract temperature1 value' node and the 'Temperature1', as shown at point 16.



## and drag & drop it in the

19) From '*dashboard*' menu in the left bar, select *char*t node working area. Double click on it

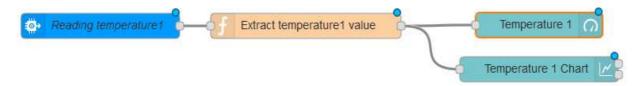
	Flow 1	Flow 2	Edit chart	node		info	debug	dashboard
eather ^			Delete		Cancel Done	Node		
narter Process			∽ node i	ronartias		Туре	ul_chart	
M Watson			. Hotel	loperties		ID	"4a907211.010854	-
atson Deprec			I Group	Add new ui_group	- 1			show m
shboard	Reading temperature	Extract temperature1 value	Temperature 1 Size	auto 2		Information	1	
button 0	Connected		Chart 12	optional chart title	3			This can either be a t ical or horizontal), or a
dropdown			<b>™</b> Type	🗠 Line chart 🚽 🗸 🗹	enlarge points 5	Each input		will be converted to a e message is ignored.
switch 9			X-axis	last 1 hours - OR 1	1000 points 6			ues are optional. The
slider 0			X-axis La	♥ HH:mm:ss	7	S 10	to-scale to any values	
numeric 🖗			Y-axis	min -60 max 60	8	a different n	es can be shown on the shown on the stopic value on the stopic val	each input message.
text input			Legend	None 9 Interpolate	linear v 10		of the same series ca bel property.	an be shown by using
date picker					initedi <b>10</b>		efines a time window (	
colour picker			Series C		11	from the gra	play. Older data will be ph. The axis labels ca ne formatted string.	
form a						Inputting a will clear the	nsg.payload contair chart	ning a blank array
gauge 🕥			Blank lat				ormation for how to p a complete chart.	pre-format data to be
chart 🖉			Name	Use deprecated (pre 2.5.0) data	a format.		abel field can be used alid data is received.	d to display some text
udio out 🔋			• Ivanie		15	-		
dification 🖂						Hold dov	wn ctrl when yo	u click on a not
template			> port la	nels			port to enable qu	lick-wiring

- 1. Group: select 'Add new ui\_group' and click on . On the panel that opens, follow the instruction explained at point 17.1, and insert a new group named '*SS10014 Temperature Charts*'. Leave the 'Tab' field on '*Home*', set '*Width*' field to *12*, and check the checkbox '*Display group name*'
- 2. Size: set '12x6'
- 3. Label: enter 'Temperature 1 Chart'
- 4. Type: select 'Line Chart'
- 5. Enlarge points: check the checkbox, for let value points visible
- 6. X-axis: select 'last 1 hour'
- 7. X-axis Label: select 'HH:mm:ss'
- 8. Y-axis: insert -60 as min. value and 60 as max. value
- 9. Legend: select 'None'
- 10. Interpolate: select 'linear'
- 11. Series Colours: select the color for chart lines
- 12. Blank label: leave the field blank
- 13. Name: enter the name of the node. In this case 'Temperature Chart 1'

## At the end, the result is as below

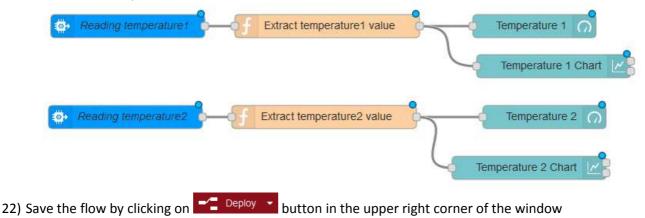
Edit chart node	
Delete	Cancel Done
✓ node proper	rties
I Group	SS10014 Temperature Charts [Home]
D Size	12 x 6
I Label	Temperature 1 Chart
🗠 Туре	Line chart ~ enlarge points
X-axis	last 1 hours V OR 1000 points
X-axis Label	
Y-axis	min -60 max 60
Legend	None v Interpolate linear v
Series Colours	
Blank label	display this text before valid data arrives
	Use deprecated (pre 2.5.0) data format.
Name	Temperature 1 Chart
o continue, click on Done butto	n.

20) Link the '*Extract temperature1 value*' node and the '*Temperature 1 Chart*' as shown at point 16 The result is the following



21) Repeat the points from 14 to 20 for the *temperature2* of the SS10014 device.The procedure is the same followed for the *temperature1*, pay attention at point 14.2, where you have to select '*Connection to Watson IoT*', and at point 14.6, where you have to enter '*temperature2*' as event.

At the end of the procedure, the result must be similar at the one below



Now we can move on to define the user interface relative to the SS10130 device, defined in the SS10680 gateway interface.

23) From 'input' menu in the left bar, select ibmiot node area.

Doub 25) Link t

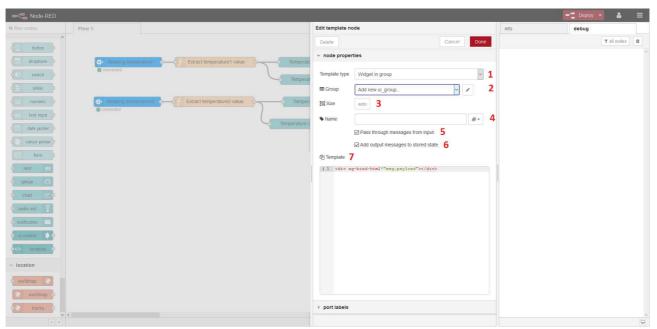
24) From

and drag & drop it into the working

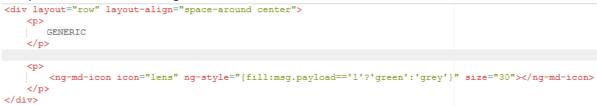
Once the node is in the working area, double click on it and a panel as the one described at point 14 is shown. Select for the field 'API Key' the option 'Connect to Watson IoT'. For the field 'Event', insert the string 'input1'. At the end of node's configuration, the result is as the following one

Edit ibmiot in r	node
Delete	Cancel Done
✓ node prope	erties
	ion API Key 🗸
a API Key	Connection to Watson IoT
ଷ୍ଟ Input Type	Device Event ~
I Device Typ	e All or +
🛓 Device Id	☐ All or gateway
n Event	All or input1
Format	All or json
🛞 QoS	0 ~
Name	Reading Input1
·	
To continue, click on Done b	utton.
From ' <i>function</i> ' menu in the lef working area.	
Double click on it and edit it as Link the previous two nodes ins	

26) From '*dashboard*' menu in the left bar, select *template* node template e drag & drop it into the working area. Double click on it and edit it as shown below



- 1. Template type: select 'Widget in group'
- 2. Group: click on and insert a new group 'SS10130 Input Status' as shown at point 17.1
- 3. Size: set '6x1'
- 4. Name: enter 'Led input 1'
- 5. Don't check 'Pass through messages from input' checkbox
- 6. Don't check 'Add output messages to stored state.' checkbox
- 7. Template: enter the following code



lit template no	de	
Delete		Cancel Don
node proper	ties	
Template type	Widget in group	~
I Group	SS10130 - Inputs Status [Home]	-
🖸 Size	6 x 1	
Name	Led input 1	h -
	Pass through messages from input.	
	Add output messages to stored state.	
2 •	yout="row" layout-align="space-around center"> GENERIC	
4 * <td></td> <td></td>		
6 <b>- </b> 7	<pre><ng-md-icon icon="lens" ng-style="(fill:msg.payload=='1'?'green':'grey')" pre="" siz<=""></ng-md-icon></pre>	e="30">
8 * 9 *		
<		

#### At the end, the result is similar to the one shown below

To continue, click on Done button.

27) Link the node added at the previous point at '*Extract input1 value*' node.

-	Reading tempera	ture1	Extract temperature1 va		Temperature 1 (၇)
				4	Temperature 1 Chart
	Reading tempera	ture2 }{f	Extract temperature2 val		Temperature 2 🕥
				Те	emperature 2 Chart
<b>*</b>	Reading Input1		act input1 value	Led input 1	<b></b>
28) From area.	<i>'input'</i> menu in t	he left bar, seled	tt <b>ibmiot</b> node	miot and drag &	drop it into the working

Once the node is in the working area, double click on it and a panel as the one described at point 14 is shown. Select for the field 'API Key' the option '*Connect to Watson IoT*'. For the field 'Event', insert the string '*fan1*'. At the end of node's configuration, the result is as the following one

Edit ibmiot in node			
Delete		Cancel Done	
✓ node properties	;		
	API Key	~	
a API Key	Connection to Watson IoT	P	
og Input Type	Device Event	~	
area Device Type	All or +		
🌡 Device Id	All or gateway		
≣ Event	□ All or fan1		
Format	All or json		
🛞 QoS	0 ~		
Name	Reading Fan1		
To continue, click on <sup>Done</sup> butto From ' <i>function</i> ' menu in the left bar working area. Double click on it and edit it as show	r, select <i>function</i> node f <sup>func</sup>	<sup>ction</sup> , drag & drop	) it into the
<ul> <li>From 'dashboard' menu in the left l working area. Double click on it and</li> <li>Template type: select 'Widg</li> <li>Group: select 'SS10130 – In</li> <li>Size: set '6x1'</li> <li>Name: enter 'Led fan 1'</li> <li>Don't check 'Pass through n</li> <li>Don't check 'Add output me</li> <li>Template: enter the following</li> </ul>	bar, select <b>template</b> node d edit it as shown below get in group' puts Status' messages from input' checkbox essages to stored state.' checkbox		lrop it into the
<pre><div layout="row" layout-align="sp&lt;/th&gt;&lt;th&gt;ace-around center"></div></pre>			
FAN 1 STATUS			
	-style="{fill:msg.payload=='1'?'	green':'grey'}" size	e="30">

29)

30)

Delete	Cancel
node proper	ties
emplate type	Widget in group
I Group	SS10130 - Inputs Status [Home]
ඩ් Size	6 x 1
Name	Led Fan 1
	Pass through messages from input.
	Add output messages to stored state.
	layout="row" layout-align="space-around center"> >>
i 1 <div 1<br="">2 <p 3</p </div>	FAN 1 STATUS
i 1 • <div ]<br="">2 • <p 3   4 • <!--</td--><td>FAN 1 STATUS</td></p </div>	FAN 1 STATUS
i 1 * <div 1<br="">2 * <p 3 4 * <!--<br-->5 6 * <p< td=""><td><pre>p&gt;   FAN 1 STATUS /p&gt; &gt;&gt;</pre></td></p<></p </div>	<pre>p&gt;   FAN 1 STATUS /p&gt; &gt;&gt;</pre>
i 1 • <div j<br="">2 • <r 3 4 • <!--<br-->5 6 • <r 7</r </r </div>	<pre>p&gt; FAN 1 STATUS /p&gt; p&gt;  cng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey')" size="30"&gt;</pre>
i 1 - <div 1<br="">2 - <pre><pre> <pre> <pre></pre></pre></pre></pre></div>	<pre>p&gt; FAN 1 STATUS /p&gt; cng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey'}" size="30"&gt;</pre>
i 1 - <di∀ 1<br="">2 - <pre><pre><pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre><pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre></pre></pre></pre></pre></di∀>	<pre>p&gt; FAN 1 STATUS /p&gt; cng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey'}" size="30"&gt;</pre>
i 1 - <di∀ ]<br="">2 - <pre><pre><pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre></pre><pre></pre><pre></pre><pre></pre></pre><pre></pre></pre><pre></pre><pre></pre></di∀>	<pre>p&gt; FAN 1 STATUS /p&gt; cng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey'}" size="30"&gt;</pre>
i 1 - <di∀ ]<br="">2 - <pre><pre><pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre><pre></pre></pre><pre></pre><pre></pre><pre></pre></pre><pre></pre></pre><pre></pre><pre></pre></di∀>	<pre>p&gt; FAN 1 STATUS /p&gt; cng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey'}" size="30"&gt;</pre>
i 1 - <div 1<br="">2 - <pre><pre><pre><pre></pre></pre></pre></pre></div>	<pre>p&gt; FAN 1 STATUS /p&gt; cng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey'}" size="30"&gt;</pre>
i 1 - <div ]<br="">2 - <pre><pre><pre><pre></pre></pre></pre></pre></div>	<pre>p&gt; FAN 1 STATUS /p&gt; cng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey'}" size="30"&gt;</pre>
i 1 - <div 1<br="">2 - <pre><pre><pre><pre></pre></pre></pre></pre></div>	<pre>p&gt; FAN 1 STATUS /p&gt; cng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey'}" size="30"&gt;</pre>
i 1 - <div ]<br="">2 - <pre><pre><pre><pre></pre></pre></pre></pre></div>	<pre>p&gt; FAN 1 STATUS /p&gt; cng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey'}" size="30"&gt;</pre>

To continue, click on Done button.

31) Link the node added at the previous point to 'Extract fan1 value node'At the end, the flow in the working area must be similar to the following one

Reading temperature1	Extract temperature1 value	Temperature 1
		Temperature 1 Chart
Reading temperature2	Extract temperature2 value	Temperature 2 (7)
		Temperature 2 Chart
Reading Input1	f Extract input1 value	Led input 1
Reading Fant 0-0	Extract fan1 value	an 1

32) From '*input*' menu in the left bar, select *ibmiot* node area.

Once the node is in the working area, double click on it and a panel as the one described at point 14 is shown. Select for the field 'API Key' the option '*Connect to Watson IoT*'. For the field 'Event', insert the string '*fan2*'. At the end of node's configuration, the result is as the following one

ibmiot

	Edit ibmiot in node			
	Delete		Cancel Done	
	<ul> <li>node properties</li> </ul>			
	Authentication	API Key	·	
	🕰 API Key	Connection to Watson IoT		
	📽 Input Type	Device Event	·	
	I Device Type	All or +		
	🛓 Device Id	☐ All or gateway		
	Event	□ All or fan2		
	Format	☐ All or json		
	🛞 QoS	0 ~		
	Name	Reading Fan2		
To continue, click o 33) From <i>'function</i> ' me working area. Double click on it a	nu in the left bar	, select <i>function</i> node finct	<sup>tion</sup> , drag & droj	p it into the
working area. Doub Template tr Group: sele Size: set '62 Name: ente Don't check Template: e	ole click on it and ype: select ' <i>Widg</i> ect ' <i>SS10130 – In</i> <1' er ' <i>Led fan 2</i> ' < ' <i>Pass through r</i> < ' <i>Add output me</i> enter the followi	par, select <b>template</b> node l edit it as shown below get in group' puts Status' messages from input' checkbox essages to stored state.' checkbox ng code	-	drop it into the
<div i<br="" layout="row">FAN 2 STATU</div>		ace-around center">		
<ng-md-icon< th=""><th>n icon="toys" ng-</th><th>-style="{fill:msg.payload=='1'?'g</th><th>reen':'grey'}" size</th><th>e="30"&gt;</th></ng-md-icon<>	n icon="toys" ng-	-style="{fill:msg.payload=='1'?'g	reen':'grey'}" size	e="30">

<pre>r node properties Template type Widget in group Group SS10130 - Inputs Status [Home] Group SS10130 - Inputs Status [Home] Group SS10130 - Inputs Status [Home] Group Strough messages from input.</pre>	Dor
<pre> Group SS10130 - Inputs Status [Home]  Ssize 6x1  Name Led Fan 2  Pass through messages from input. Add output messages to stored state.  The model of the store state.  The store store state.  The store store state.  The store store state.  Store store store state.  Store store state.  Store store store store store state.  Store st</pre>	
<pre>Size 6 x 1 Name Led Fan 2 Pass through messages from input. Add output messages to stored state. Template  i 1 * <div center"="" layout='align="space-around'> 2*</div></pre>	
Name Led Fan 2 Pass through messages from input. Add output messages to stored state. Template i 1+ <div layout="row" layout-align="space-around center"> 2- 3 FAN 2 STATUS 4-  5 6- 7  5 6- 7  5  6- 7  9- </div>	
<pre>Pass through messages from input.</pre>	
Add output messages to stored state. 2 Template i 1 - <div layout="row" layout-align="space-around center"> 2 - 3 FAN 2 STATUS 4 - 5 6 - 7 <ng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey'}" size="30">8 - 9 - </ng-md-icon></div>	
<pre> # Template  i 1. <div layout="row" layout-align="space-around center"> 2.  3</div></pre>	
<pre> # Template  i 1. <div layout="row" layout-align="space-around center"> 2.  3</div></pre>	
<pre>i 1* <div layout="row" layout-align="space-around center"> 2*</div></pre>	
<pre>2 * 3</pre>	
<pre>2 * 3</pre>	
<pre>4*  5 6*  7</pre>	
5 6 * 7 <ng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey'}" size="30">8 * 9 * </ng-md-icon>	
<pre>6*  7 <ng-md-icon icon="toys" ng-style="{fill:msg.payload=='1'?'green':'grey'}" size="30"> 9* </ng-md-icon></pre>	
<pre>7</pre>	
8^ 9^	
9 -	-md-icon

To continue, click on Done button.

35) Link the node added at the previous point to '*Extract fan2 value*' node.At the end, the flow in the working area must be similar to the following one

Reading temperature 1 0	Extract temperature1 value	Temperature 1 🥥
		Temperature 1 Chart
Reading temperature2	Extract temperature2 value	Temperature 2 🥥
		Temperature 2 Chart
Reading Input1 0-0 f Extrac	t input1 value	Led input 1
Reading Fan1 - f Extract fan	t value Led Fan 1	
Reading Fan2	2 value company Led Fan :	2

36) From '*input*' menu in the left bar, select *ibmiot* node area.

Once the node is in the working area, double click on it and a panel as the one described at point 14 is shown. Select for the field 'API Key' the option '*Connect to Watson IoT*'. For the field 'Event', insert the string '*bulb1*'. At the end of node's configuration, the result is as the following one

ibmiot

	Edit ibmiot in node		
	Delete	Cancel Done	
	<ul> <li>node properties</li> </ul>		
	Authentication		
	🕰 API Key	Connection to Watson IoT	
	📽 Input Type	Device Event ~	
	🖌 Device Type	₩ All or +	
	i Device Id	☐All or gateway	
	Event	All or bulb1	
	Format	☐ All or json	
	🛞 QoS	0 ~	
	Name	Reading Bulb1	
working area. Doub Template t Group: sele Size: set '62 Name: ente Don't chect	nu in the left bar nd edit it as show nenu in the left k ble click on it and ype: select ' <i>Widg</i> ect ' <i>SS10130 – Inj</i> x1' er ' <i>Led bulb 1</i> ' k ' <i>Pass through n</i> k ' <i>Add output me</i> enter the followi	, select <b>function</b> node <b>function</b> , drag & d wn at point 15. par, select <b>template</b> node <b>()</b> template e drag edit it as shown below the tin group' pouts Status' messages from input' checkbox essages to stored state.' checkbox ing code	drop it into the & drop it into the
<rp><rg-md-icon :<="" r=""></rg-md-icon></rp>	icon="wb_incandesc	ent" ng-style="{fill:msg.payload=='1'?'green':'grey'}	" size="30">

dit template no	de			
Delete			Cancel	Don
node proper	ies			
Template type	Widget in group		~	
🖽 Group	SS10130 - Inputs Status [Home]	8		
🕒 Size	6 x 1			
Name	Led Bulb1	-		
2 Template       i     1 + <div< td="">       2 + <p< td="">       3       4 +        5</p<></div<>	BULB 1 STATUS			
6 <b>* <p< b=""> 7</p<></b>	<pre><ng-md-icon icon="wb_incandescent" ng-style="{fill:msg.payload=='1'?'green':'grey p&gt;&lt;/pre&gt;&lt;/td&gt;&lt;td&gt;(')" siz<="" td=""><td>e="30"×/ng-m</td><td>d-icon</td></ng-md-icon></pre>	e="30"×/ng-m	d-icon	

To continue, click on Done button.

39) Link the node added at the previous point to '*Extract bulb1 value*' node.

At the end, the flow in the working area must be similar to the following one

Reading temperature 1 Competitive 1 Competit
Temperature 1 Chart
Reading temperature2 p
Temperature 2 Chart
Reading Input1
Reading Fan1 - of Extract fan1 value - of Led Fan 1
Reading Fan2 - of Extract fan2 value Led Fan 2
Reading Bulb1     Composition for Extract bulb1 value     Composition for the second sec
40) Save the flow by clicking on T Deploy to button in the upper right corner of the window

41) From '*input*' menu in the left bar, select *ibmiot* node area.

Once the node is in the working area, double click on it and a panel as the one described at point 14 is shown. Select for the field 'API Key' the option '*Connect to Watson IoT*'. For the field 'Event', insert the string '*alarm\_input1*'. At the end of node's configuration, the result is as the following one

٠

ibmiot

Edit it	omiot in node		
Dele	ete	Cancel	ne
~ no	de properties		
÷ A	uthentication API Key	<i>,</i>	
<b>Q</b> A	PI Key Connec	tion to Watson IoT	
¢ë in	put Type Device	Event ~	
<b>₽</b> D	evice Type Zall or	+	
<u>ڈ</u> D	evice ld All or	gateway	
≣E	vent All or	alarm_input1	
Fi Fi	ormat All or	json	
) (*)	0	~	
Na Na	ame Reading	ı alarm_input1	
	t as shown at point the left bar, select on it and edit it as ect ' <i>Widget in grou</i> and add a new gro <i>larm input 1</i> ' <i>hrough messages fo</i> <i>utput messages to</i> e following code	t 15. <b>template</b> node shown below p' oup 'SS10130 – Alarm Status' from input' checkbox stored state.' checkbox	drag & drop it into the
<rg-md-icon icon="w&lt;br&gt;&lt;/div&gt;&lt;/th&gt;&lt;th&gt;arning" ng-style="&lt;/th&gt;&lt;th&gt;{fill:msg.payload=='1'?'red':'g&lt;/th&gt;&lt;th&gt;green'}" size="20"></rg-md-icon>			

Delete		Cancel Don
node proper	rties	
emplate type	Widget in group	~
∎ Group	SS10130 - Alarms Status [Home]	
ਤੋਂ Size	auto	
Name	Led alarm_input1	
	Pass through messages from input.	
	Add output messages to stored state.	
Tomplato		
ት Template		
i 1• <div la<="" th=""><th>ayout="row" layout-align="space-around center"&gt;</th><th></th></div>	ayout="row" layout-align="space-around center">	
i 1 • <div la<br="">2 • <p; 3   4 • <td>S GENERIC</td><td></td></p; </div>	S GENERIC	
i 1 • <div 1a<br="">2 • <p: 3   4 • <td>&gt; GENERIC p&gt;</td><td></td></p: </div>	> GENERIC p>	
i 1 + <div 1:<br="">2 + <p; 3 4 + 5 6 + <p; 7</p; </p; </div>	<pre>&gt; GENERIC p&gt; &gt;  </pre>	0"×/ng-md-icon)
i 1 - <div 1:<br="">2 - <p: 3 - 4 - 5 - <p: 7 - 8 - <td><pre>Severic p&gt; </pre> <pre> </pre> </td><td>0"&gt;</td></p: </p: </div>	<pre>Severic p&gt; </pre> <pre> </pre>	0">
i 1 + <div 1:<br="">2 + <p; 3 4 + 5 6 + <p; 7</p; </p; </div>	<pre>Severic p&gt; </pre> <pre> </pre>	0">
i 1 + <div 1:<br="">2 - <p: 3 4 + 5 6 - <p: 7 8 + <td><pre>Severic p&gt; </pre> <pre> </pre> </td><td>0"×/ng-md-icon)</td></p: </p: </div>	<pre>Severic p&gt; </pre> <pre> </pre>	0"×/ng-md-icon)
i 1 + <div 1:<br="">2 - <p: 3 4 + 5 6 - <p: 7 8 + <td><pre>Severic p&gt; </pre> <pre> </pre> </td><td>0"&gt;</td></p: </p: </div>	<pre>Severic p&gt; </pre> <pre> </pre>	0">
i 1 + <div 1:<br="">2 - <p: 3 4 + 5 6 - <p: 7 8 + <td><pre>Severic p&gt; </pre> <pre> </pre> </td><td>0"&gt;</td></p: </p: </div>	<pre>Severic p&gt; </pre> <pre> </pre>	0">
i 1 + <div 1a<br="">2 + <p: 3 + 5 - <p: 7 + 8 + <td><pre>Severic p&gt; </pre> <pre> </pre> </td><td>0"&gt;</td></p: </p: </div>	<pre>Severic p&gt; </pre> <pre> </pre>	0">
i 1 + <div 1a<br="">2 + <p: 3 + 5 - <p: 7 + 8 + <td><pre>Severic p&gt; </pre> <pre> </pre> </td><td>0"&gt;</td></p: </p: </div>	<pre>Severic p&gt; </pre> <pre> </pre>	0">
i 1 + <div 1a<br="">2 + <p: 3 + 5 - <p: 7 + 8 + <td><pre>Severic p&gt; </pre> <pre> </pre> </td><td>0"≫/ng-md-icon)</td></p: </p: </div>	<pre>Severic p&gt; </pre> <pre> </pre>	0"≫/ng-md-icon)
i 1 + <div 1a<br="">2 + <p: 3 + 5 - <p: 7 + 8 + <td><pre>Severic p&gt; </pre> <pre> </pre> </td><td>0"≫/ng-md-icon)</td></p: </p: </div>	<pre>Severic p&gt; </pre> <pre> </pre>	0"≫/ng-md-icon)
i 1 + <div 1:<br="">2 - <p: 3 4 + 5 6 - <p: 7 8 + <td><pre>Severic p&gt; </pre> <pre> </pre> </td><td>0"≫/ng-md-icon)</td></p: </p: </div>	<pre>Severic p&gt; </pre> <pre> </pre>	0"≫/ng-md-icon)

To continue, click on button.

44) Link the node added at the previous point to '*Extract alarm\_input1 value*' node. At the end, the flow in the working area must be similar to the following one

Reading temperature1     Comperature1     Comperatur
Temperature 1 Chart
Reading temperature2 Extract temperature2 value Temperature 2
Temperature 2 Chart
Reading Input 1 Extract input 1 value
Reading Fant - F Extract fan1 value - Led Fan 1
Reading Fan2     Extract fan2 value     Led Fan 2
Reading Builb 1 Company Extract bulb1 value Company Led Bulb1
Reading alarm_input1 C Extract alarm_input1 value C Led alarm_input1
45) From ' <i>input</i> ' menu in the left bar, select <b><i>ibmiot</i></b> node and drag & drop it into the working

area.

Once the node is in the working area, double click on it and a panel as the one described at point 14 is shown. Select for the field 'API Key' the option '*Connect to Watson IoT*'. For the field 'Event', insert the string '*alarm\_fan1*'. At the end of node's configuration, the result is as the following one

	Edit ibmiot in node				
	Delete			Cancel	Done
	✓ node properties	;			
	Authentication	API Key	/	~	
	🕰 API Key	Connec	ction to Watson IoT	~	
	📽 Input Type	Device	Event	~	
	I Device Type	All or	+		
	🛓 Device Id	All or	gateway		
	Event	All or	alarm_fan1		
	Format	All or	json		
	🛞 QoS	0	~		
	Name	Reading	g alarm_fan1		
To continue, click on	<sup>one</sup> button.				

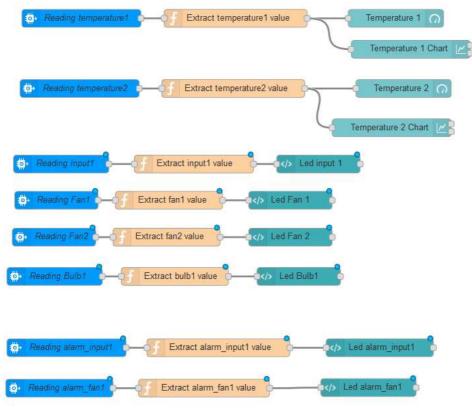
<ul> <li>46) From 'function' menu in the left bar, select function node function, drag &amp; drop it into the working area.</li> <li>Double click on it and edit it as shown at point 15.</li> </ul>
<ul> <li>47) From 'dashboard' menu in the left bar, select template node working area. Double click on it and edit it as shown below</li> <li>Template type: select 'Widget in group'</li> <li>Group: select 'SS10130 – Alarm Status'</li> <li>Size: set 'auto'</li> <li>Name: enter 'Led alarm fan 1'</li> <li>Don't check 'Pass through messages from input' checkbox</li> <li>Don't check 'Add output messages to stored state.' checkbox</li> <li>Template: enter the following code</li> </ul>
<pre><div layout="row" layout-align="space-around center">          FAN 1</div></pre>

```
<ng-md-icon icon="warning" ng-style="{fill:msg.payload=='1'?'red':'green'}" size="20"></ng-md-icon></div>
```

Delete			Cancel	Do
node proper	rties			
Template type	Widget in group		~	
I Group	SS10130 - Alarms Status [Home]	ø		
편 Size	auto			
Name	Led alarm_fan1	₽ -		
	Pass through messages from input.			
	Add output messages to stored state.			
ካ Template				
2 Template				
	ayout="row" layout-align="space-around center">			
i 1 • <div 1a<br="">2 • <p: 3</p: </div>	ayout="row" layout-align="space-around center"> > FAN 1			
i 1 • <div la<br="">2 • <p:< td=""><td>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1</td><td></td><td></td><td></td></p:<></div>	ayout="row" layout-align="space-around center"> > FAN 1			
i 1 • <div la<br="">2 • <p: 3 4 • 6 •</p: </div>	ayout="row" layout-align="space-around center"> > FAN 1 p>			
<pre>i 1 - <div -="" 2="" 3="" 4="" 5="" 6="" 7<="" <="" <p:="" la="" p="" pre=""></div></pre>	<pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt; &gt; </pre> <pre>      <pre>      <pre>      <pre>     <pre>     <pre>     <pre>    <pre>    <pre>    <pre>     &lt;</pre></pre></pre></pre></pre></pre></pre></pre></pre></pre>	size=	"20"> <td>l-ice</td>	l-ice
i 1 • <div la<br="">2 • <p: 3 4 • 6 •</p: </div>	<pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>."20"≻<td>l-ico</td></td></pre>	size=	."20"≻ <td>l-ico</td>	l-ico
i 1 - <div la<br="">2 - <p: 3 4 - 6 - <p: 7 8 - <td><pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"×/ng-md</td><td>l-ico</td></pre></td></p: </p: </div>	<pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"×/ng-md</td><td>l-ico</td></pre>	size=	"20"×/ng-md	l-ico
i 1 - <div la<br="">2 - <p: 3 4 - 6 - <p: 7 8 - <td><pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"×/ng-md</td><td>l-ic</td></pre></td></p: </p: </div>	<pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"×/ng-md</td><td>l-ic</td></pre>	size=	"20"×/ng-md	l-ic
i 1 - <div la<br="">2 - <p: 3 4 - 6 - <p: 7 8 - <td><pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"≫/ng-md</td><td>l-ic</td></pre></td></p: </p: </div>	<pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"≫/ng-md</td><td>l-ic</td></pre>	size=	"20"≫/ng-md	l-ic
i 1 - <div la<br="">2 - <p: 3 4 - 6 - <p: 7 8 - <td><pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"≻⁄ng-md</td><td>l-ic</td></pre></td></p: </p: </div>	<pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"≻⁄ng-md</td><td>l-ic</td></pre>	size=	"20"≻⁄ng-md	l-ic
i 1- <div la<br="">2- <p: 3 4- 6- <p: 7 8- <td><pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"&gt;<td>l-ice</td></td></pre></td></p: </p: </div>	<pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"&gt;<td>l-ice</td></td></pre>	size=	"20"> <td>l-ice</td>	l-ice
i 1- <div la<br="">2- <p: 3 4- 6- <p: 7 8- <td><pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"&gt;<td>l-icc</td></td></pre></td></p: </p: </div>	<pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"&gt;<td>l-icc</td></td></pre>	size=	"20"> <td>l-icc</td>	l-icc
i 1 - <div la<br="">2 - <p: 3 4 - 6 - <p: 7 8 - <td><pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"&gt;<td>l-ic</td></td></pre></td></p: </p: </div>	<pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"&gt;<td>l-ic</td></td></pre>	size=	"20"> <td>l-ic</td>	l-ic
i 1 - <div la<br="">2 - <p: 3 4 - 6 - <p: 7 8 - <td><pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"&gt;<td>l-ic</td></td></pre></td></p: </p: </div>	<pre>ayout="row" layout-align="space-around center"&gt; &gt; FAN 1 p&gt;   <!--</td--><td>size=</td><td>"20"&gt;<td>l-ic</td></td></pre>	size=	"20"> <td>l-ic</td>	l-ic

To continue, click on button.

48) Link the node added at the previous point to '*Extract alarm\_fan1 value*' node.At the end, the flow in the working area must be similar to the following one



49) From '*input*' menu in the left bar, select *ibmiot* node area.

Once the node is in the working area, double click on it and a panel as the one described at point 14 is shown. Select for the field 'API Key' the option '*Connect to Watson IoT*'. For the field 'Event', insert the string '*alarm\_fan2*'. At the end of node's configuration, the result is as the following one

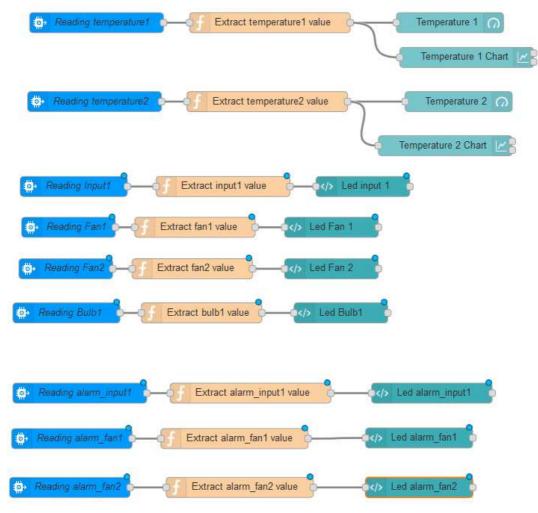
ibmiot

	Edit ibmiot in node			
	Delete		Cancel Done	
	<ul> <li>node properties</li> </ul>	i		
	The Authentication	API Key	~	
	🕰 API Key	Connection to Watson IoT	1	
	😋 Input Type	Device Event	~	
	🖋 Device Type	All or +		
	🌡 Device Id	☐ All or gateway		
	n Event	☐ All or alarm fan2		
	Format			
	€ QoS			
	Name	Reading alarm_fan2		
To continue, click on	Done buttor	I.		
50) From <i>'function</i> ' men working area. Double click on it an		, select <i>function</i> node	<sup>ction</sup> , drag & droj	p it into the
working area. Doubl Template ty Group: select Size: set 'au Name: enter Don't check	le click on it and pe: select ' <i>Widg</i> ct ' <i>SS10130 – Ald</i> to' r 'Led alarm fan 'Pass through n	arm Status' 2' nessages from input' checkbox ssages to stored state.' checkbo	-	drop it into the
<div la<="" layout="row" th=""><th>yout-align="spac</th><th>e-around center"&gt;</th><th></th><th></th></div>	yout-align="spac	e-around center">		
FAN 2				
<rp><rg-md-icon< p=""></rg-md-icon<></rp>	icon="warning" n	ng-style="{fill:msg.payload=='1'	?'red':'green'}" siz	ze="20">

es	Cancel Done
25	
Widget in group	~
SS10130 - Alarms Status [Home]	1
auto	
Led alarm_fan2	<b>—</b>
Pass through messages from input.	
FAN 2	
	ize="20">
	SS10130 - Alarms Status [Home]

To continue, click on button.

52) Link the node added at the previous point to '*Extract alarm\_fan2 value*' node. At the end, the flow in the working area must be similar to the following one



53) From '*input*' menu in the left bar, select *ibmiot* node area.

Once the node is in the working area, double click on it and a panel as the one described at point 14 is shown. Select for the field 'API Key' the option '*Connect to Watson IoT*'. For the field 'Event', insert the string '*alarm\_bulb1*'. At the end of node's configuration, the result is as the following one

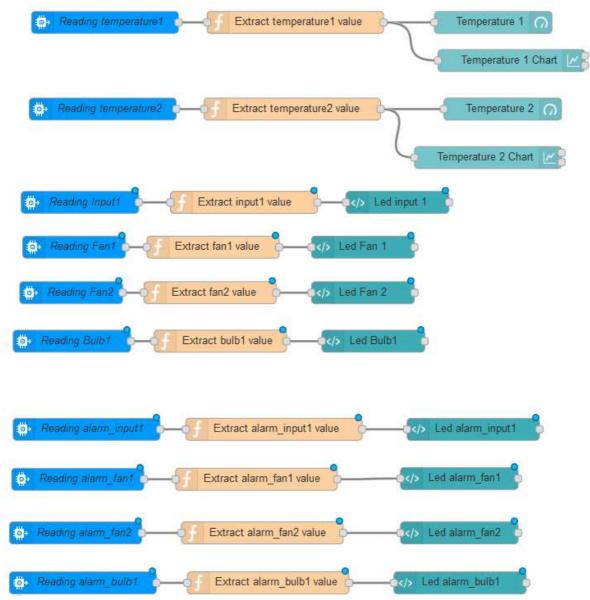
ibmiot

	Edit ibmiot in node	
	Delete	Cancel Done
	✓ node properties	
	Authentication	API Key 🗸
	🕰 API Key	Connection to Watson IoT
	🕸 Input Type	Device Event ~
	🖋 Device Type	All or +
	a Device Id	All or gateway
	Event	All or alarm_bulb1
	Format	All or json
	⊛ QoS	0 ~
	Name	Reading alarm_bulb1
To continue, click or	Done button.	
54) From <i>'function</i> ' mer working area. Double click on it an		
working area. Doubl Template ty Group: selec Size: set 'au Name: ente Don't check Don't check	le click on it and pe: select 'Widge ct 'SS10130 – Ala to' r 'Led alarm bulb 'Pass through m	erm Status' 1' essages from input' checkbox ssages to stored state.' checkbox
<div la<="" layout="row" th=""><th>ayout-align="spac</th><th>e-around center"&gt;</th></div>	ayout-align="spac	e-around center">
BULB 1		
<rp><rg-md-icon< r=""></rg-md-icon<></rp>	icon="warning" n	g-style="{fill:msg.payload=='1'?'red':'green'}" size="20">

de	
	Cancel Done
ties	
Widget in group	~
SS10130 - Alarms Status [Home]	1
auto	
Led alarm_bulb1	<i></i>
Pass through messages from input.	
BULB 1 > <ng-md-icon icon="warning" ng-style="(fill:msg.payload=='1'?'red':'green')" s<="" th=""><th>ize="20"&gt;</th></ng-md-icon>	ize="20">
	SS10130 - Alarms Status [Home]          auto          Led alarm_bulb1          Pass through messages from input.          Add output messages to stored state.          wyout="row" layout-align="space-around center">

To continue, click on button.

56) Link the node added at the previous point and *'Extract alarm\_bulb1 value'* node. At the end, the flow in the working area must be similar to the following one



- and drag & drop it into the
- 57) From '*dashboard*' menu in the left bar, select *chart* node working area. Double click on it and edit it as shown below

Edit chart node	
Delete	Cancel Done
v node proper	ies
I Group	SS10130 - Alarms Status [Home]
📴 Size	auto
1 Label	Alarms Chart
🗠 Туре	Line chart – enlarge points
X-axis	last 1 hours V OR 1000 points
X-axis Label	
Y-axis	min -2 max 2
Legend	Show v Interpolate step v
Series Colours	
Blank label	display this text before valid data arrives
	Use deprecated (pre 2.5.0) data format.
Name	Alarms Chart

chart

To continue, click on button.

58) Link '*Extract alarm\_input1 value*', '*Extract alarm\_fan1 value*', '*Extract alarm\_fan2*', '*Extract alarm\_bulb1*' nodes to the node added at the previous point. The result must be similar to the one shown below

Reading alarm_input1      Extract alarm_input1 value     Led alarm_input1	
Reading alarm_fant Extract alarm_fan1 value Led alarm_fan1	
Reading alarm_fan2 Extract alarm_fan2 value	Alarms Chart
Reading alarm_bulb1 Contract alarm_bulb1 value	

- 59) From '*dashboard*' menu in the left bar, select *switch* node and drag & drop it into the working area. Double click on it and edit it as described below
  - Group: click on and add a new group 'SS10130 Simulate Alarms'
  - Size: set '6x1'
  - Name: enter 'INPUT1 ALARM'
  - Icon: select 'Default'
  - On payload: from menu select JSON and enter '{"value":1}'
  - Off payload: from menu select JSON and enter '{"value":0}'
  - Topic: leave blank
  - Name: enter 'activate input1 alarm'

Edit switch nod	e	
Delete	Cancel	Dor
v node proper	ties	
I Group	SS10130 - Simulate Alarms [Home]	<i>.</i>
ច្រាំ Size	6 x 1	
1 Label	INPUT1 ALARM	
🖾 Icon	Default	
→ If msg arrive	es on input, pass through to output: 🗹	
When clicked	d, send:	
On Payload	<pre>     {} {"value":1} </pre>	
Off Payload	<pre>~ {} {"value":0}</pre>	
Торіс		
Name	activate input1 alarm	

To continue, click on button.

60) From '*output*' menu in the left bar, select **ibmiot** node area.

ibmiot 🔅 and drag & drop it into the working

Double click on it and edit it as shown below

Edit ibmiot out node	2		
Delete		Cancel Done	
✓ node properties			
	API Key	~	
a API Key	Connection to Watson IoT	~ Ø	
😋 Output Type	Device Command	~	
I Device Type	SS10680		
ا Device Id	gateway		
Command Type	activateAlarm_input1		
Format	json		
S Data	msg.payload		
⊛ QoS	0 ~		
Name Name	Write input1 alarm value		
To continue, click on button. 61) Link the nodes added at the previous two activate input1 alar		∾ te input1 alarm v	value
activate input i ala			
<ul> <li>62) From 'dashboard' menu in the left bar, se working area. Double click on it and edit</li> <li>Group: select 'SS10130 – Simulat</li> </ul>	it as described below	and dra	g & drop it into the

Size: set '6x1' •

- Name: enter 'FAN 1 ALARM' •
- Icon: select 'Default'
- On payload: from menu select JSON and enter '{"value":1}'
- Off payload: from menu select JSON and enter '{"value":0}' ٠
- Topic: leave blank
- Name: enter 'activate fan1 alarm' •

Delete		Cancel	Done
node proper	ties		
I Group	SS10130 - Simulate Alarms [Home]	~	
🔄 Size	6 x 1		
1 Label	FAN1 ALARM		
🖾 lcon	Default		
→ If msg arrive	es on input, pass through to output: 🗹		
When clicked	I, send:		
On Payload	{} {"value":1}		
Off Payload	<pre>- {} {"value":0}</pre>		
Торіс			]
	activate fan1 alarm		

To continue, click on Done button.

63) From '*output*' menu in the left bar, select *ibmiot* node area.

Double click on it and edit it as shown below

Delete		Cancel
<ul> <li>node properties</li> </ul>		
Authentication	API Key	~
🕰 API Key	Connection to Watson IoT	~
🕸 Output Type	Device Command	~
Device Type	SS10680	
a Device Id	gateway	
Command Type	activateAlarm_fan1	
Format	json	
E Data	msg.payload	
🛞 QoS	0 ~	
Name	Write fan1 alarm value	

To continue, click on button.

- 64) Link the nodes added at the previous two points, as shown at point 61
- 65) From '*dashboard*' menu in the left bar, select *switch* node working area. Double click on it and edit it as described below
  - Group: select 'SS10130 Simulate Alarms'
  - Size: set '6x1'
  - Name: enter 'FAN 2 ALARM'
  - Icon: select 'Default'
  - On payload: from menu select JSON and enter '{"value":1}'
  - Off payload: from menu select JSON and enter '{"value":0}'
  - Topic: leave blank
  - Name: enter 'activate fan2 alarm'

Edit switch no	de	
Delete		Cancel Done
✓ node prope	erties	
I Group	SS10130 - Simulate Alarms [Home]	~
回 Size	6 x 1	
1 Label	FAN2 ALARM	
🖾 lcon	Default	
→ If msg arriv	ves on input, pass through to output: 🗹	
☑ When click	ed, send:	
On Payloa	ud - {} {"value":1}	
Off Payloa	id - {} {"value":0}	
Торіс		
Name	activate fan2 alarm	
To continue, click on Done but	ton.	

switch and drag & drop it into the

66) From '*output*' menu in the left bar, select *ibmiot* node area.

and drag & drop it into the working

Double click on it and edit it as shown below

Delete		Cancel Do
<ul> <li>node properties</li> </ul>		
Authentication	API Key	~
API Key	Connection to Watson IoT	~
🕸 Output Type	Device Command	~
4 Device Type	SS10680	
🛓 Device Id	gateway	
Command Type	activateAlarm_fan2	
Format	json	
Se Data	msg.payload	
🛞 QoS	0 ~	
Name	Write fan2 alarm value	

ibmiot

To continue, click on Done button.

- 67) Link the nodes added at the previous two points, as shown at point 61
- 68) From '*dashboard*' menu in the left bar, select **switch** node working area. Double click on it and edit it as described below
  - Group: select 'SS10130 Simulate Alarms'
  - Size: set '6x1'
  - Name: enter 'BULB 1 ALARM'
  - Icon: select 'Default'
  - On payload: from menu select JSON and enter '{"value":1}'
  - Off payload: from menu select JSON and enter '{"value":0}'
  - Topic: leave blank
  - Name: enter 'activate bulb1 alarm'

Edit	switch node		
De	elete		Cancel Done
~ 1	node properti	es	
	Group	SS10130 - Simulate Alarms [Home]	~
ju	Size	6 x 1	
I	Label	BULB1 ALARM	
	lcon	Default ~	
÷	If msg arrives	on input, pass through to output: 🗹	
	When clicked,	send:	
	On Payload	{} {"value":1}	
	Off Payload	<pre>- {} {"value":0}</pre>	
	Topic		
•	Name	activate bulb1 alarm	
ontinue, click on	one butto	n.	

69) From '*output*' menu in the left bar, select *ibmiot* node area.

Double click on it and edit it as shown below

Delete		Cancel Do
node properties		
Authentication	АРІ Кеу	~
🕰 API Key	Connection to Watson IoT	~
📽 Output Type	Device Command	~
🖌 Device Type	SS10680	
🤹 Device Id	gateway	
Command Type	activateAlarm_bulb1	
Format	json	
🛢 Data	value	
🛞 QoS	0 ~	
Name	Write bulb1 alarm value	

To continue, click on Done button.

70) Link the nodes added at the previous two points, as shown at point 61The result must be similar at the following one

Reading temperature1	Extract temperature1 value	Reading Input
	Temperature 1 Chart	Co. Reading Fanl - F Extract fan 1 value - Cy Led Fan 1
ereding temperature2	Extract temperature 2 (7)	C Reading Fan2 - f Extract fan2 value
	Temperature 2 Chart	C. Reading Build Extract bub1 value
Ex Reading siarm_input1	tract alarm_input1 value	💽 activate input1 alarm 🗳 🗸 Write input1 alarm value 🚭
Reading alarm_fan1	act alarm_fan1 value	activate fant alarm
to Reading alarm_fan2	ract alarm_fan2 value	co activate fan2 alarm
🔅 Reading alarm_buib1 💡 — 🥤 E	xtract alarm_bulb1 value	e activate bulb1 alarm 😽 Write bulb1 alarm value 🔅
	ing on Control of the upper right in the upper righ	-
	i in the left bar, select <b>worlmap in</b> node 🎑	worldmap and drag & drop it into the
working area.	s de la ferrar a la servici de la f	function
	I in the left bar, select <b>function</b> node <b>U</b>	and drag & drop it into the
-	Edit function node	
	Delete	Cancel Done
	<ul> <li>v node properties</li> </ul>	
	Name filter	
	<pre> Function 1 if (msg.payload.action === "connected") { return : </pre>	mag; }
	2 3 return null;	
	Cutputs 1 ↓	
		1
To continue, click on	Done button.	



and drag & drop it into the

## 74) From *'function'* menu in the left bar, select *function* node working area. Double click on it and edit it as shown below

✓ node prope	iles	
Name	Set gateway position	
🖋 Function		
2 var 1 3 * msg.p 4 1 5 1 6 n	t:lat, on:lon, ame:"Gateway SS10680 - Guanzate", con:"fa-map-marker fa-3x"	
<b>&gt;⊄</b> Outputs	1	
Guipuis	T T	
Done	hutten	
ue, click on	button.	

76) Lin filter world map Set gateway position world map

77) From '*input*' menu in the left bar, select *inject* node solution and drag & drop it into the working area. Double click on it and edit it as described below:

- Payload: from menu select 'string' and in the text field enter '/worldmap' •
- Topic: leave blank ٠

- Repeat: select 'None' .
- Check 'Inject once a start?' checkbox •

Edit inject noc	le	Cancel Do
v node prop	erties	
Payload		
nterior Topic		
C Repeat	none	~
	☑ Inject once at start?	
Name	Name	

To continue, click on Done button.

78) From *'function'* menu in the left bar, select *template* node from the left bar, select template and drag & drop it into the working area. Double click on it and edit it as show below

Delete			Cancel	
<ul> <li>node propert</li> </ul>	ies			
Name	Name			
▼ Name	Name			
Set property	<ul> <li>msg. payload</li> </ul>			
Format	Mustache template	~		
			Syntax Highlight: mus	stache
1 <iframe< td=""><td><pre>src={{{payload}}}</pre></td><td>height=600px;</td><td>; width=600px;</td><td>≻</td></iframe<>	<pre>src={{{payload}}}</pre>	height=600px;	; width=600px;	≻

To continue, click on button.

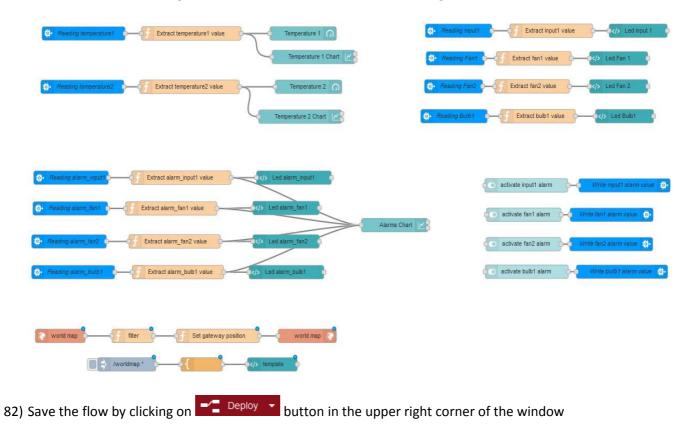
- 79) From '*dashboard*' menu in the left bar, select **template** node <sup>(<)</sup> template and drag & drop it into the working area. Double click on it and edit it as described below
  - Template type: select 'Widget in group'
  - Group: click on and add a new group '*Map*'
  - Size: select '12x12'
  - Check 'Pass through messages from input' checkbox
  - Check 'Add output messages to stored state.' checkbox
  - Template: enter the following code

```
<div ng-bind-html="msg.payload | trusted"></div>
```

At the end of the configuration, the result must be similar at the following one

1	Edit template no	de	
	Delete		Cancel Done
	v node proper	ties	
	Template type	Widget in group	~
	I Group	Map [Home]	<ul> <li>✓</li> </ul>
	ច្រាំ Size	12 x 12	
	Name		
		Pass through messages from inpu	ıt.
		Add output messages to stored st	ate.
	C Template		
	_		
To continue, click on Don			
0) Link the last three nodes a	is shown bel	ow	
	/worldmap 1		

## 81) At the end, in the working area, there is a flow similar at the following one



83) To have an idea on how the user interface is structured, click on in the upper right corner of the window, select '*View*' and then click on '*Dashboard*'

	🚽 Deploy 👻 💄 🗮
Show sidebar	✓ View
Dashboard Debug messages	<ul> <li>Import</li> <li>▲ Export</li> </ul>
	Search flows
	Configuration nodes <ul> <li>Flows</li> <li>Subflows</li> </ul>
	Manage palette
	Settings
	Keyboard shortcuts Node-RED website v0.17.5

In the left bar, a new tab 'dashboard' opens and it has three sub-tabs:

• Layout Tab: from this tab is it possible to see the structure of the user interface

info	debug	dashboard X
Layout Theme S	ite	Ľ
Tabs & Links		* ¥ +tab +link
v 🗋 Home 🔶 Tab		^
> 🆽 Map 🔶 Gr	aphic group	
∽ ⊞ SS10014 ◀	<ul> <li>Graphic group</li> </ul>	
🖾 Temperatur	e 1	
E Temperatur	re 2	<i>∎</i> edit
✓	perature Charts	
🔚 Temperatur	e 1 Chart 🔶 Gra	aphic elements
🔚 Temperatur	re 2 Chart 🔶	
> 🆽 SS10130 - Inp	outs Status	
> 🆽 SS10130 - Ala	arms Status	
> 🆽 SS10130 - Sir	nulate Alarms	~

• Themes Tab: from this tab is it possible to change the appearance of the user interface window

info		debug		dashboard	×	1
Layout	Theme Site				ľ	
Style						
Light (defau	ult)				~	
Base Settings					-	
Colour					C	ł
Font	System Font (de	efault)			~	

• Site Tab: from this tab is it possible to change some properties of the site

info	debug		dashboard	×	
Layout Theme Site				Ľ	
Title					
Node-RED Dashboard					
Options					
Show the title bar				~	
No swipe between tabs					
Date Format					
DD/MM/YYYY					
Sizes	Horizontal		Vertical	Ċ	
1x1 Widget Size	48	÷	48	+	
Widget Spacing	6	÷	6	-	
Group Padding	0	-	0	+	
Group Spacing	6	÷	6	+	

84) To see the user interface, you can click on <sup>C</sup> button in dashboard tab, or enter in the address bar of the browser the route of your application assigned by Watson IoT followed by '/ui'. The user interface realized for this example appears like the following one

