SCHOOL OF DIGITAL MEDIA AND INFOCOMM TECHNOLOGY (DMIT)

IOT CA2 Step-by-step Tutorial

DIPLOMA IN BUSINESS INFORMATION TECHNOLOGY DIPLOMA IN INFORMATION TECHNOLOGY DIPLOMA IN INFOCOMM SECURITY MANAGEMENT

ST0324 Internet of Things (IOT)

Date of Submission: 20/02/2018

Prepared for: Lin Zhao

Class: DISM/FT/3A/33

Submitted by:

Student ID Name

1551312 Mark Wong

1543742 Sukundhen Jayamoorthi

Table of Contents

Sec	tion 1	Overview of project	3
	A.	A. Where we have uploaded our tutorial	
	В.	Why have we chosen to upload to this site	3
	C.	What have we uploaded	
	D.	What is the application about?	3
	E.	Summary of the steps that will be described	3
	F.	How does the final RPI set-up looks like?	4
	G.	How does the web or mobile application look like?	5
Section 2 Project requirements			5
Hardware checklist			5
	Softv	vare checklist	6
Section 3 Setup overview			6
	Fritzi	ng Diagram	6
Sec	tion 4	Configure softwares	6
	MQTT		6
A.	Insta	ll Mosquitto	6
B.	Configure Mosquitto		7
	C.	Test Mosquitto	8
	Node	Red	10
	Node Red – Picam Directory settings		10
Section 5 A. Node Red Diagram Overview		11	
B. Individual node explanation		ual node explanation	12
Sec	tion 6		13
	A.	Create Node-RED flow on RPi	13
	В.	RPI_GPIO Node for Motion Sensor	14
	C.	Function node	14
	D.	MQTT Node	15
	E.	Debug node	15
	F.	RPI_RPIO Node for LED	16
	G.	Dashboard/Button node	16
	Н.	Dashboard/node	18
	I. C	amerapi Takephoto node	19

J. Dashboard/Template node	19
Section 7	20
IBM Image Recognition Web Service	20
Section 8	22
IBM Image Recognition on NodeRED	22
NodeRED Diagram	22
Building the diagram	22

Section 1 Overview of project

A. Where we have uploaded our tutorial

https://github.com/users/sopossible33/

B. Why have we chosen to upload to this site

Github is a place where people can collaborate together and share source codes.

C. What have we uploaded

We have uploaded our source code for the node-red diagram. This will reduce the time cost for the user to set up our application.

D. What is the application about?

This application is about notifying you when there is a visitor. When a motion is sensed, a red LED will light up and the pi camera will take a picture. This image is then showed on the dashboard. Then we have included a random image reconigtion.

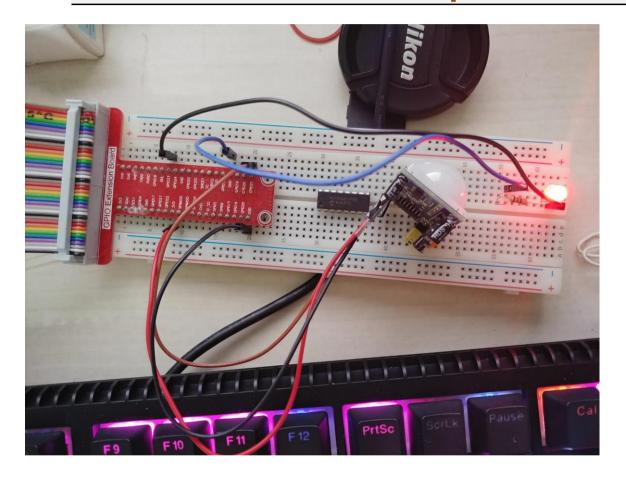
E. Summary of the steps that will be described

Provide a bullet list of the steps that will be covered in the other parts of this tutorial

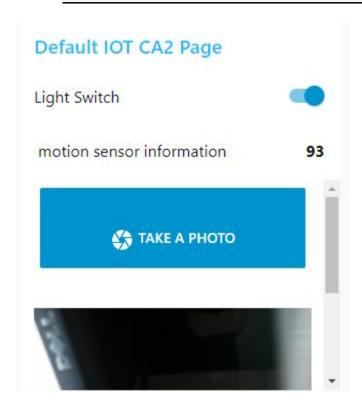
Section		Description		
1)	Overview			
Sect	Sections 2 to 7 provides the step-by-step instructions to set up the application			
2)	Prject requirements	Provides overview of hardware required		
3)	Setup Overview	Overview of the diagram		
4)	Configure Software	Install MQTT, NodeRED palettes and configure Mosquitto,		
		subscribe to topic		
5)	Create NodeRED flows	Create the diagrams		

6)	Configure IBM Bluemix (on IBM website)	
7)	Configure IBM Image	
	Recognition on NodeRED	

F. How does the final RPI set-up looks like?



G. How does the web or mobile application look like?



Section 2 Project requirements

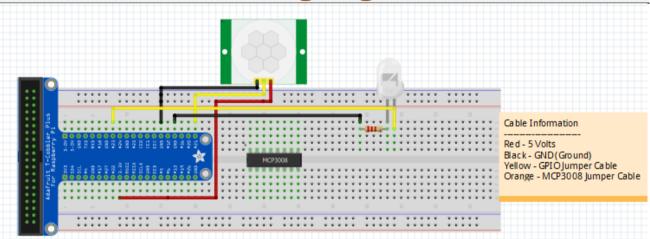


Software checklist

IBM Bluemix Account
Node-red
mosquitto

Section 3 Setup overview

Fritzing Diagram



Section 4 Configure softwares

MQTT

Install Mosquitto A.

- **Task** o Install the Mosquitto broker and clients on your Raspberry Pi with this command
 - sudo apt-get install mosquitto mosquitto-clients
 - b) Note, as Mosquitto has already been installed on the ST0324 Raspbian image, you will see this output instead.

a)

```
N
     Task
o
      pi@raspberrypi-dorachua:~/labs/p9 $ sudo apt-get install mosquitto mosquitto-cli
      Reading package lists... Done
     Building dependency tree
     Reading state information... Done
     mosquitto is already the newest version.
     mosquitto-clients is already the newest version.
     O upgraded, O newly installed, O to remove and 1 not upgraded.
```

Configure Mosquitto

By default, the Mosquitto accepts connections from anonymous clients (i.e. anyone!). We want to limit access to valid devices only, so let's change some settings in Mosquitto in this sub-section.

- N Task 0 Stop the Mosquitto broker first a) sudo /etc/init.d/mosquitto stop b) Mosquitto has been stopped pi@raspberrypi-dorachua:~/labs/p9 \$ sudo /etc/init.d/mosquitto stop ok] Stopping mosquitto (via systemctl): mosquitto.service. Create a config file that will instruct the Mosquitto broker to reject anonymous clients c) cd /etc/mosquitto/conf.d/ sudo nano mosquitto.conf d) Add the following lines in mosquitto.conf File Edit Tabs Help File: mosquitto. GNU nano 2.2.6 allow_anonymous false password_file /etc/mosquitto/conf.d/passwd require_certificate false allow_anonymous false password_file
 - /etc/mosquitto/conf.d/passwd require_certificate false

Press Ctrl-O and Ctrl-X to save the file and exit the nano editor e)

f) From the current /conf.d directory, create an empty password file sudo touch passwd

g) Use the mosquitto_passwd tool to create a password hash for user pi You will be asked to enter your password twice. Enter the password you wish to use for the user you defined, e.g. I entered my password as 12345678

C. Test Mosquitto

sudo mosquitto passwd -c /etc/mosquitto/conf.d/passwd pi

a) Open three Terminal windows

b) In the first Terminal window, make sure the Mosquitto broker is running

mosquitto

mosquitto

Task

o

In the first Terminal window, make sure the Mosquitto broker is running

mosquitto

mosquitto

mosquitto

c) In the second Terminal window, run the command line subscriber:

mosquitto sub -v -t 'sensors/motion'



d) You should see another message in the first terminal window saying another client is connected.

```
1498275338: New connection from ::1 on port 1883.
1498275338: New client connected from ::1 as mosqsub/11841-
raspberry (c1, k60).
```

You should also see this message in the subscriber terminal:

N Task

0

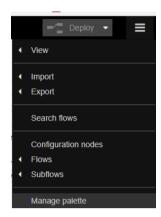
```
sensors/motion 1
sensors/motion /home/pi/Pictures/photo1.JPEG
sensors/motion 0
sensors/motion /home/pi/Pictures/photo1.JPEG
sensors/motion 1
sensors/motion /home/pi/Pictures/photo1.JPEG
sensors/motion 0
```

e) You have tested that Mosquitto is configured correctly and you can subscribe to a topic.

At this point, keep all the two terminal windows open and running. You will need them in the next section.

Node Red

At the top right hand corner, click on the 3 lines and select "Manage Palette"



Click on the tab "Install" and install the following items.



node-red-contrib-aws-iot-hub node-red-contrib-ibm-watson-iot node-red dashboard node-red-node-watson node-red-contrib-camerapi

Node Red - Picam Directory settings

In terminal, type sudo nano /root/.node-red/settings.js

Find for httpStatic and append httpStatic: '/home/pi/Pictures/',

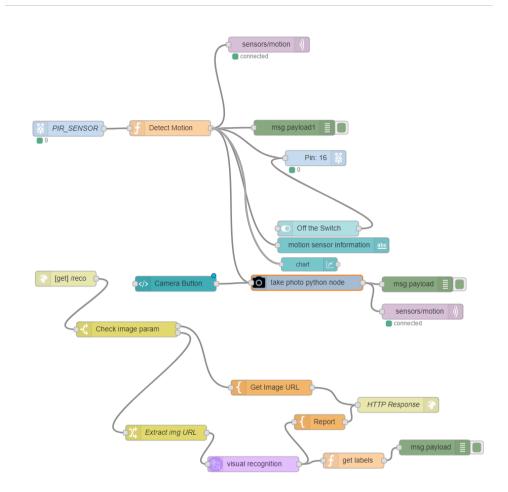
```
// The following property can be used
// to apply the same root to both par
//httpRoot: '/red',

// When httpAdminRoot is used to move
// following property can be used to
// that should be served at http://loo
httpStatic: '/home/pi/Pictures/',

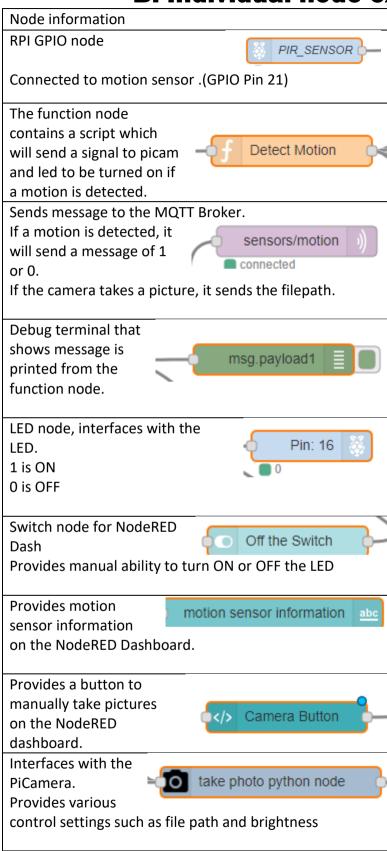
// The maximum size of HTTP request to
// Default: 5mb
//apiMaxLength: '5mb',
```

Created by Mark Wong, Sukun Jayamoorthi

Section 5 A. Node Red Diagram Overview



B. Individual node explanation



Section 6

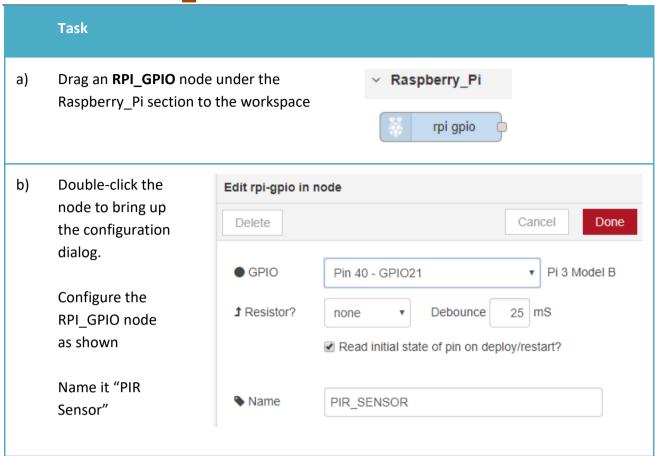
A. Create Node-RED flow on RPi

```
No
         Task
         Open a Terminal window and start Node-RED on your RPi
a)
         node-red start
b)
        pi@raspberrypi-dorachua:~ $ node-red start
        2 Apr 14:23:31 - [info]
        Welcome to Node-RED
        2 Apr 14:23:31 - [info] Node-RED version: v0.16.2
        2 Apr 14:23:31 - [info] Node.js version: v7.8.0
2 Apr 14:23:31 - [info] Linux 4.4.50-v7+ arm LE
2 Apr 14:23:32 - [info] Loading palette nodes
           MODE: { MODE_0: 0, MODE_1: 1, MODE_2: 2, MODE_3: 3 },
           CS: { none: 64, high: 4, low: 0 },
           ORDER: { msb: false, lsb: true },
           Spi: [Function: Spi] }
           Apr 14:23:37 - [info] Settings file : /home/pi/.node-red/settings.js
Apr 14:23:37 - [info] User directory : /home/pi/.node-red
Apr 14:23:37 - [info] Flows file : /home/pi/.node-red/start
Apr 14:23:37 - [info] Creating new flow file
Apr 14:23:37 - [info] Starting flows
Apr 14:23:37 - [info] Started flows
Apr 14:23:37 - [info] Started flows
                 14:23:37 - [info] Server now running at http://127.0.0.1:1880/
c)

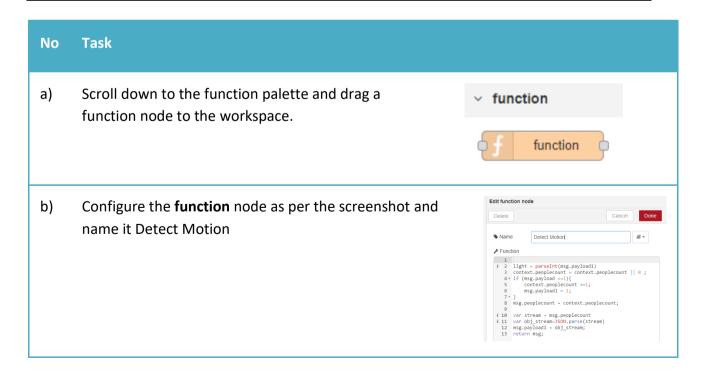
    Next, open the Node-RED webpage on your laptop's browser

        • The URL would be the IP address of your Raspberry Pi, followed by the 1880 port e.g.
            http://192.168.0.111:1880
```

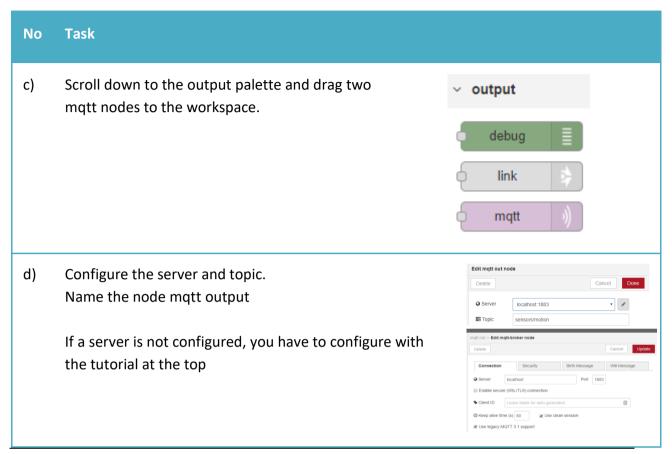
B. RPI_GPIO Node for Motion Sensor



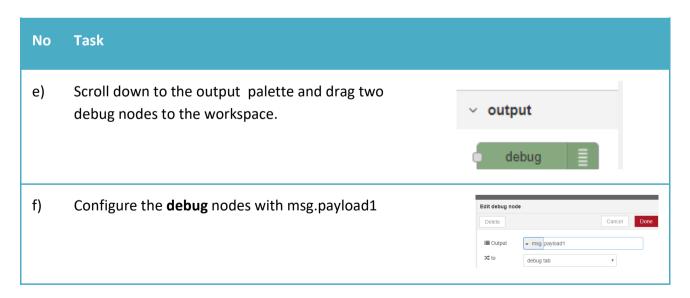
C. Function node



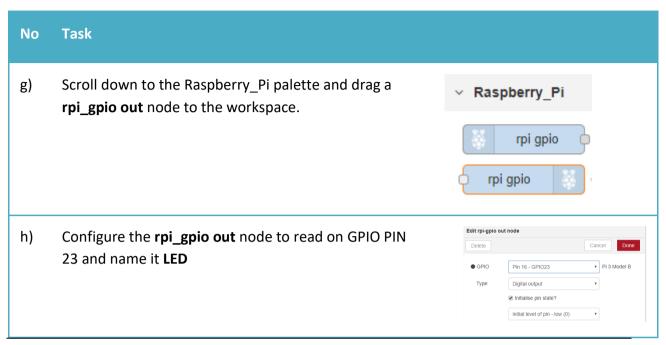
D. MQTT Node



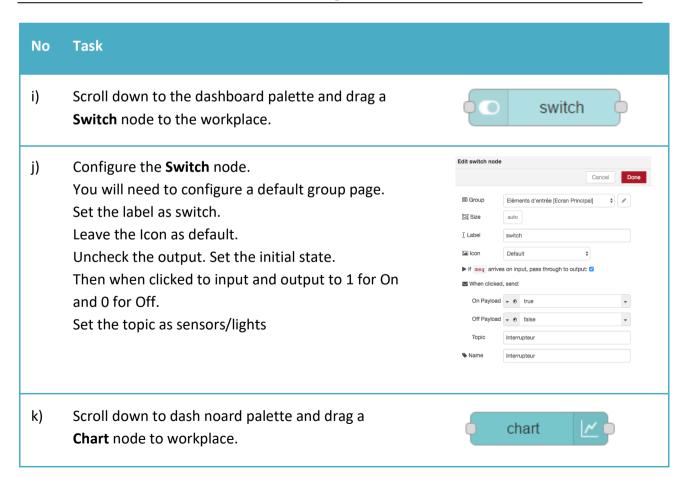
E. Debug node

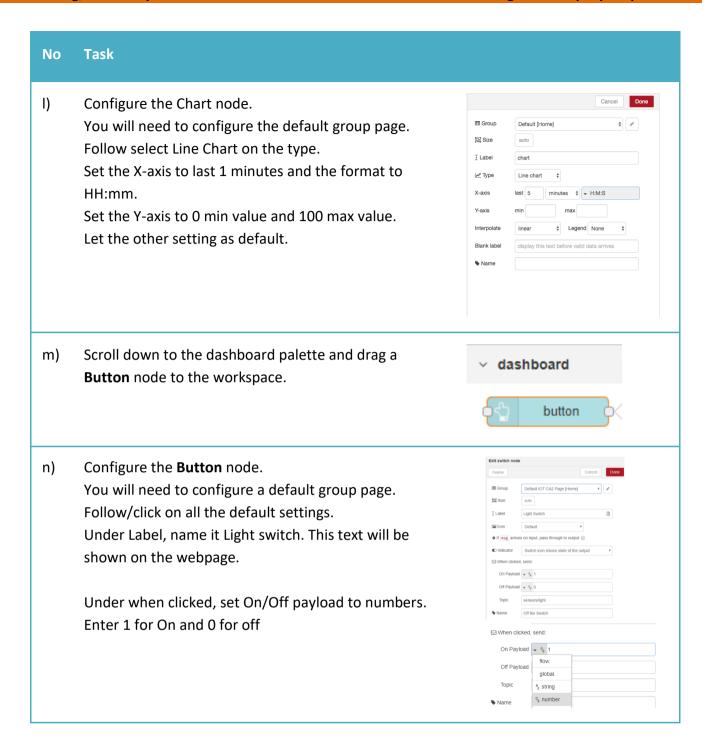


F. RPI_RPIO Node for LED

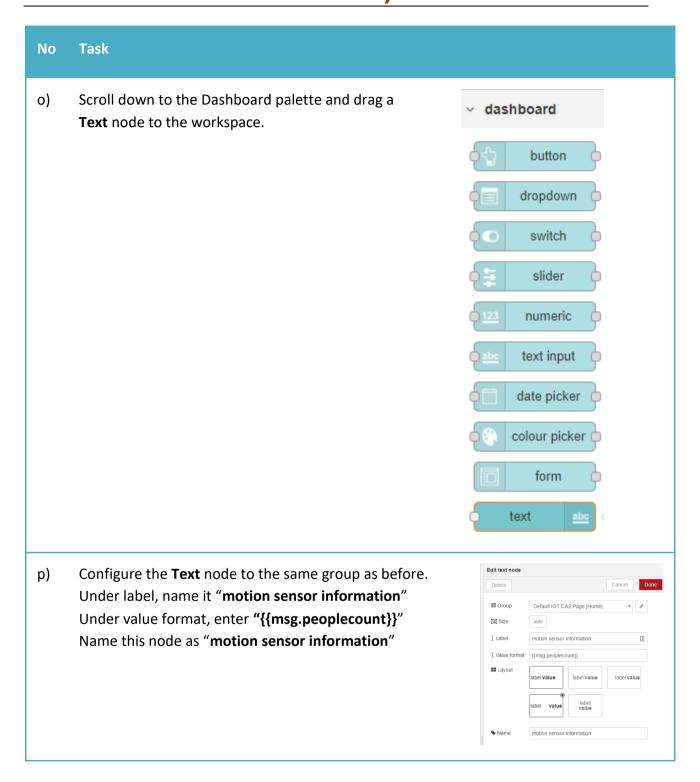


G. Dashboard/Button node

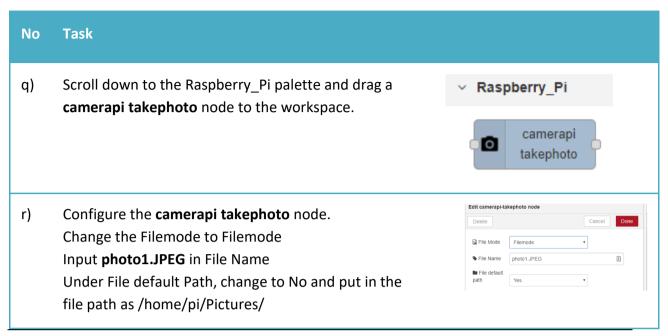




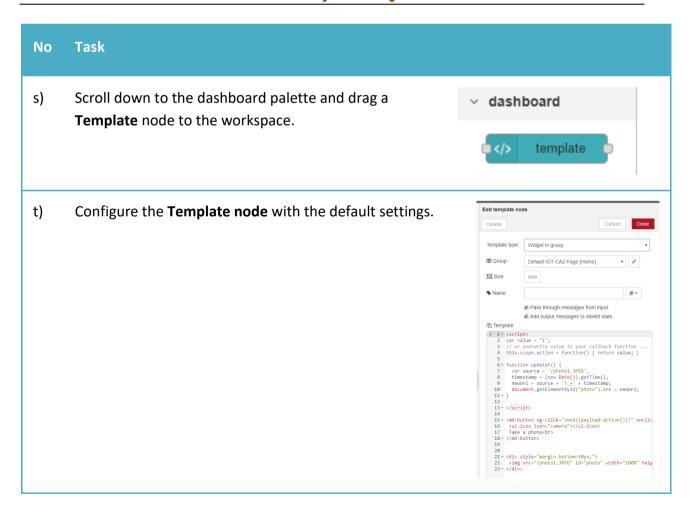
H. Dashboard/node



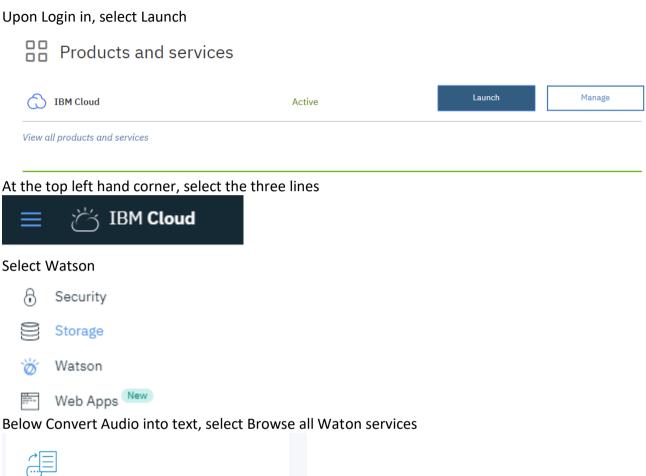
I. Camerapi Takephoto node

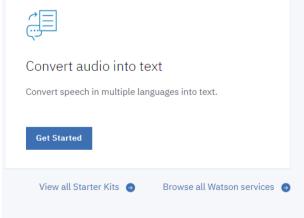


J. Dashboard/Template node

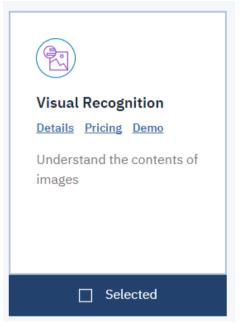


Section 7 IBM Image Recognition Web Service





Select Visual Recognition



On the right pane, you will see the service that you have added.

At the bottom, Select add services.

Enter a project name such as IOTCA2

Under resources, click on Visual recognition



You will be directed to another page. On the left, select Service credentials. Under key name, you will see your project name. Select view credentials

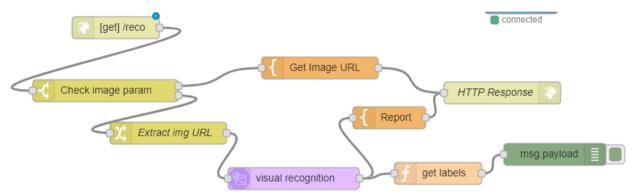


Please note down the api key as we will need to in the following section

Section 8 IBM Image Recognition on NodeRED

(Guide adopted from ibm-offical-github-page)

NodeRED Diagram



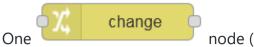
Building the diagram

One	http	node configured with a frage LIDI
One \		node, configured with a /reco URL
One C	switch (node which will test for the presence of the imageurl query
Offic		Hode Which will test for the presence of the imaged I query
parameter:		

One template node, configured to output an HTML input field and suggest a few selected images taken from the main Watson Visual Recognition demo web page:

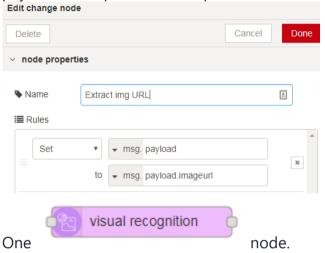


Created by Mark Wong, Sukun Jayamoorthi

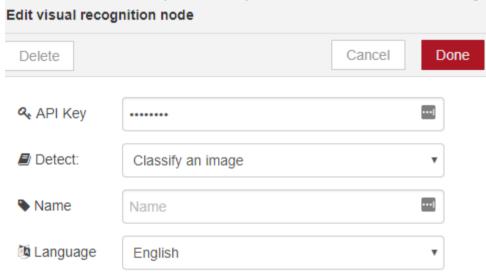


node (named Extract img URL)

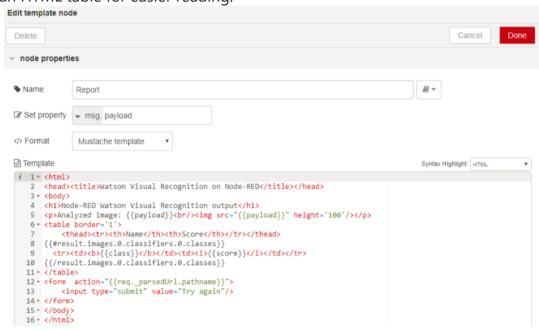
This is to extract the imageurl query parameter from the web request and assign it to the payload to be provided as input to the Visual Recognition node:



You will need to enter your APi Key and leave the rest of the settings at default.



Finally, add a node linked to the node linked to the node. The template will format the output returned from the Visual Recognition node into an HTML table for easier reading:



Testing the flow, goto your http://yourrpiip:1880/reco

Node-RED Watson Visual Recognition output

Analyzed image: https://ii.worldmarket.com/fcgi-bin/iipsrv.fcgi?FIF=/images/worldmarket/source/1267_XXX_v1.tif&wid=650&cvt=jpeg



Name	Score
rummer (drinking glass)	0.805
drinking glass	0.964
tableware	0.964
utensil	0.964
flute	0.635
liqueur glass	0.528
goblet	0.5
beverage	0.794
ash grey color	0.693
bottle green color	0.619
Try again	

ST0324 Internet of	Things CA2 Ste	p-by-step Tutoria	1

-- End of CA2 Step-by-step tutorial --