LAB - 2

Face Recognition Using Watson Services

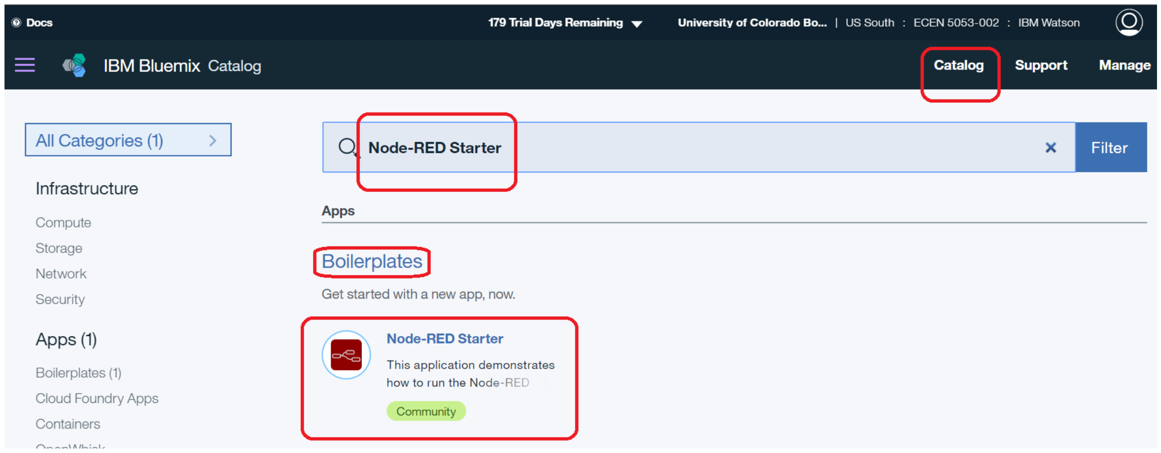
ECEN 5053-002

Developing Internet of Things

Professor Dave Sluiter

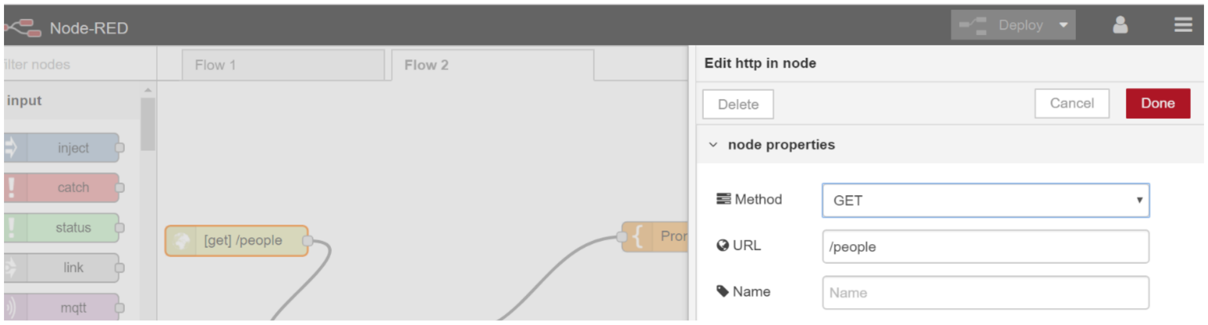
In this document, the steps to generate the Face Recognition Data from an image URL using Watson services. The flow will present a simple Web Page, where the user will input the Image's URL and submit it. Watson services will be used to determine the age range, gender, and name.

- First Sign into Bluemix
- Create a boilerplate app in Bluemix by accessing the **Catalog** tab.
- From Bluemix console, access the Catalog tab (Top Right) and search for Node-RED Starter.



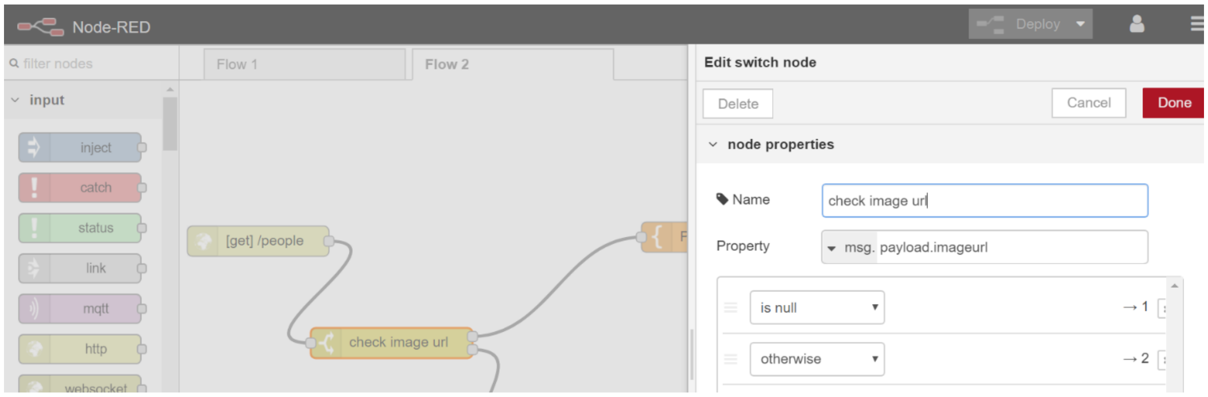
- Specify the name for your app and Click Create [App will take few minutes to stage]
- Click Visit App. URL
- Configure Node-RED authentication (as per instructions are given in the previous document)
- Go to your Node-RED Editor
- This open the editor where you begin to drag and drop the nodes and enter the code (given below)

- Drag and drop an **HTTP IN** node.
- For the Method, select GET and for the URL, specify any context name, in this example /people (refer the snapshot below).
- Click Done



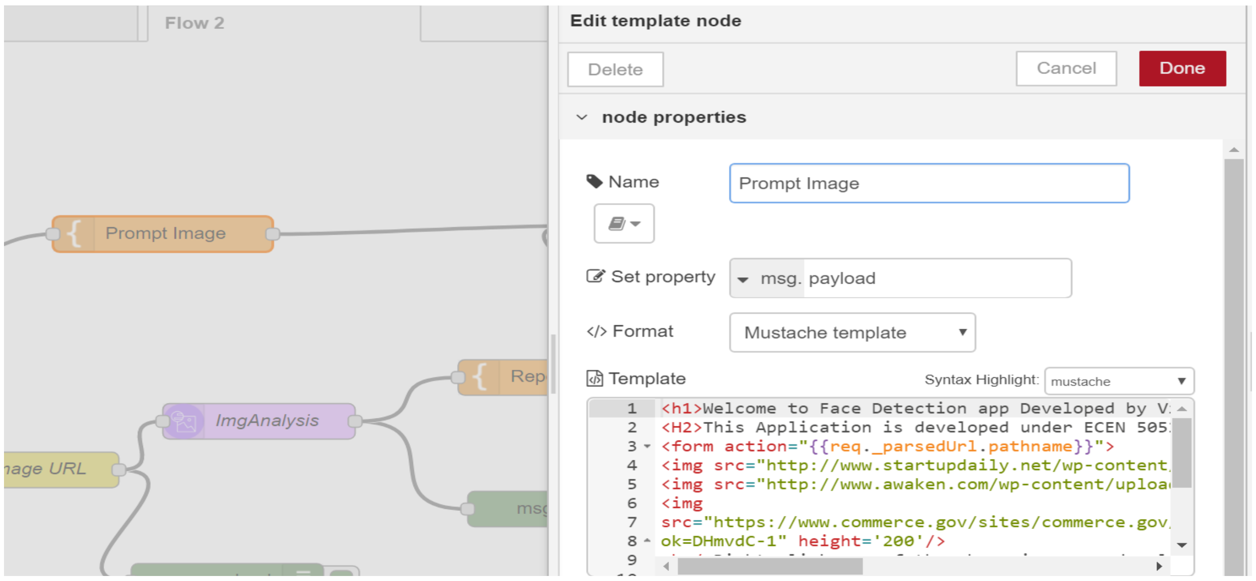
Method: GET
URL: /people
Name: Name

- Drag and drop a **Switch** node, which will test for the presence of the imageurl query parameter. (You can use search nodes in the Top Left)
- Use any name in the given tab.
- In the property box, append imageurl after payload.
- Set two conditions from the drop-down list, first is null and second is otherwise (refer the snapshot below).
- Click Done.



Name: check image url
Property: msg. payload.imageurl
is null → 1
otherwise → 2

- Drag and drop the **Template** node, configure to output an HTML input field and suggest a few selected images taken from official sources.
- Specify a name
- Set the property to msg.payload (Use the drop down list)
- Type payload after the msg [The Function node allows JavaScript code to run against the messages that are passed in and then return zero or more messages to continue the flow. The message is passed in as an object called msg. By convention it will have a msg.payload property containing the body of the message]



Name: Prompt Image

Set property: msg. payload

Format: Mustache template

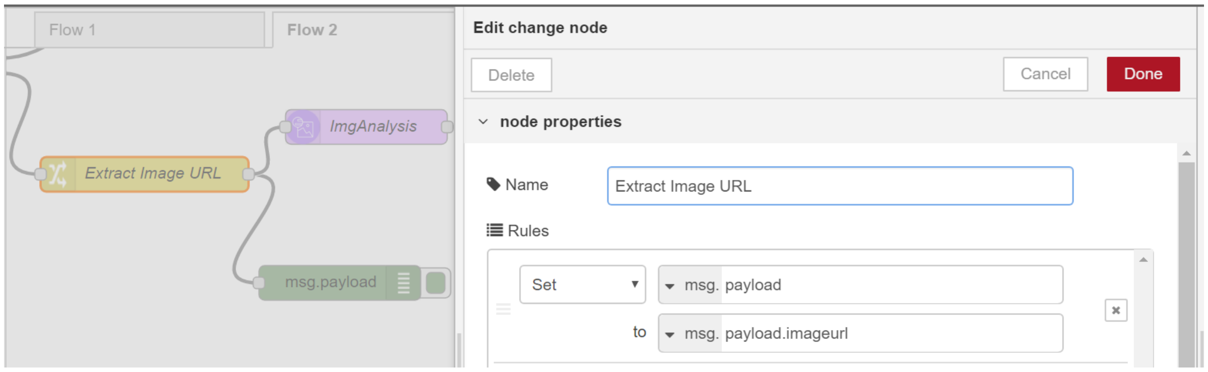
Template — Syntax Highlight: mustache

```
<h1>Welcome to Face Detection app Developed by V
<H2>This Application is developed under ECEN 505
<form action="{{req._parsedUrl.pathname}}">
<img src="http://www.startupdaily.net/wp-content
<img src="http://www.awaken.com/wp-content/uploa
<img
src="https://www.commerce.gov/sites/commerce.gov
ok=DHmvdC-1" height='200'/>
```

- Here is the code which needs to be written in the Template (copy paste)

```
<h1>Welcome to Face Detection app Developed by INSERT YOUR NAME HERE
</h1>

<H2>This Application is developed under ECEN 5053-002 DDIOT Course</
H2>

<form action="{{req._parsedUrl.pathname}}">

<img src="http://www.startupdaily.net/wp-content/uploads/2012/04/
Successful-Business-Proposal.jpg" height='200'/>

<img src="http://www.awaken.com/wp-content/uploads/2015/05/
forbes.jpg" height='200'/>

<img

src="https://www.commerce.gov/sites/commerce.gov/files/styles/
scale_250w/public/media/images/profile/elizabethholmes.jpg?it

ok=DHmvdC-1" height='200'/>

<br/>Right-click one of the above images and select Copy image
location and paste the URL in the box below.<br>Do

an image search for faces, try multiple faces. After you click on an
image, to the right it usually says "View image" click

that to get the URL.<br/>

<br>Image URL: <input type="text" name="imageurl"/>

<input type="submit" value="Analyze"/>

</form>
```

- Drag and Drop a **CHANGE** node to extract the imageurl query parameter from the web request
- Specify a name
- Select Set for rule [The first rule is msg.payload, and the second rule is msg.payload.imageurl (use the drop down-list to select the msg part and type the rest)]

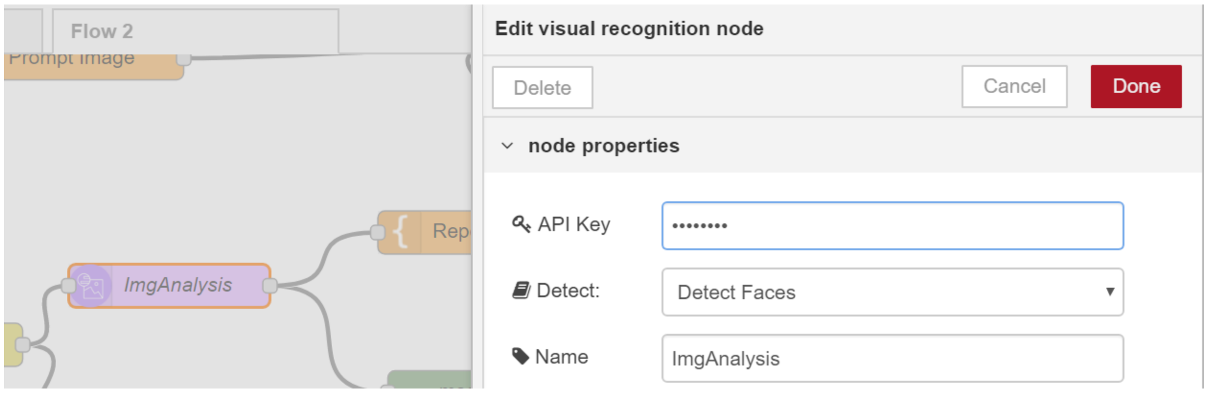


Edit change node

Name: Extract Image URL

Rules: Set msg. payload to msg. payload.imageurl

- Drag and drop **Visual Recognition** Node
- Go to Bluemix, search for Visual Recognition service.
- Give the service a unique name
- Click Create
- Back to the Node-RED canvas, for Detect annotators, select "Detect Faces"
- Click Ok



Edit visual recognition node

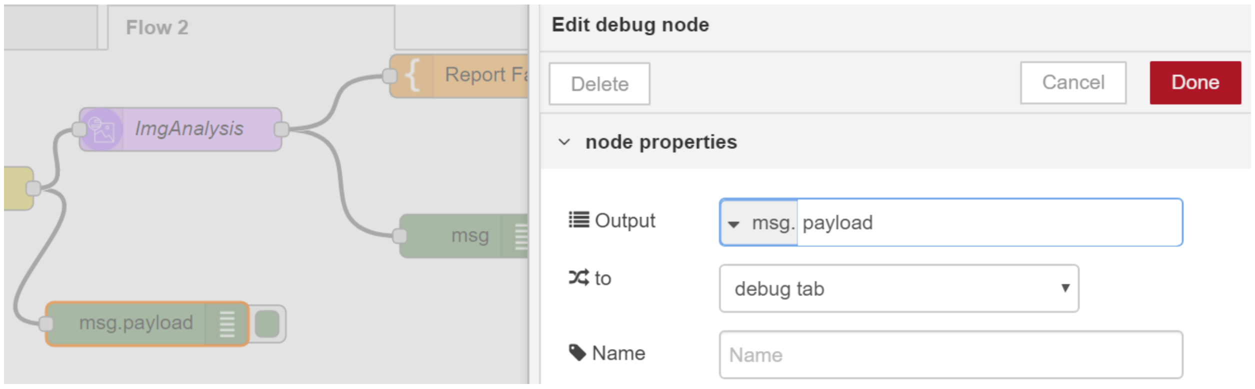
API Key: ••••••••

Detect: Detect Faces

Name: ImgAnalysis

- Need to enter API key to get the Visual Recognition node to work.
- The API key will be extract from Bluemix Dashboard-> Visual Recognition

- Connect the **Extract** Image URL with msg.payload
- Drag and drop debug in the output section to make it to msg.payload
- Double Click on the debug, and fill the table contents as shown in the figure below



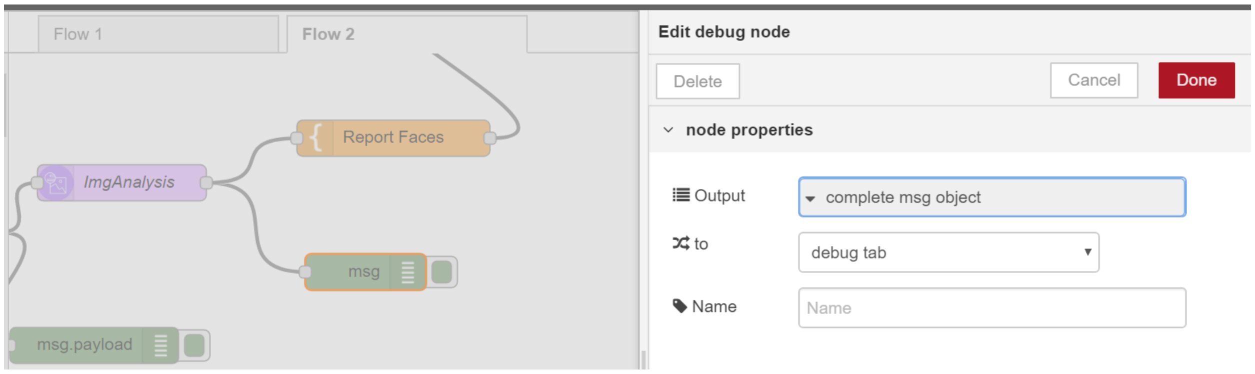
Edit debug node

Output: msg. payload

to: debug tab

Name: Name

- Add another debug node which is connected to ImgAnalysis
- Edit the table content of the debug as shown in the figure below

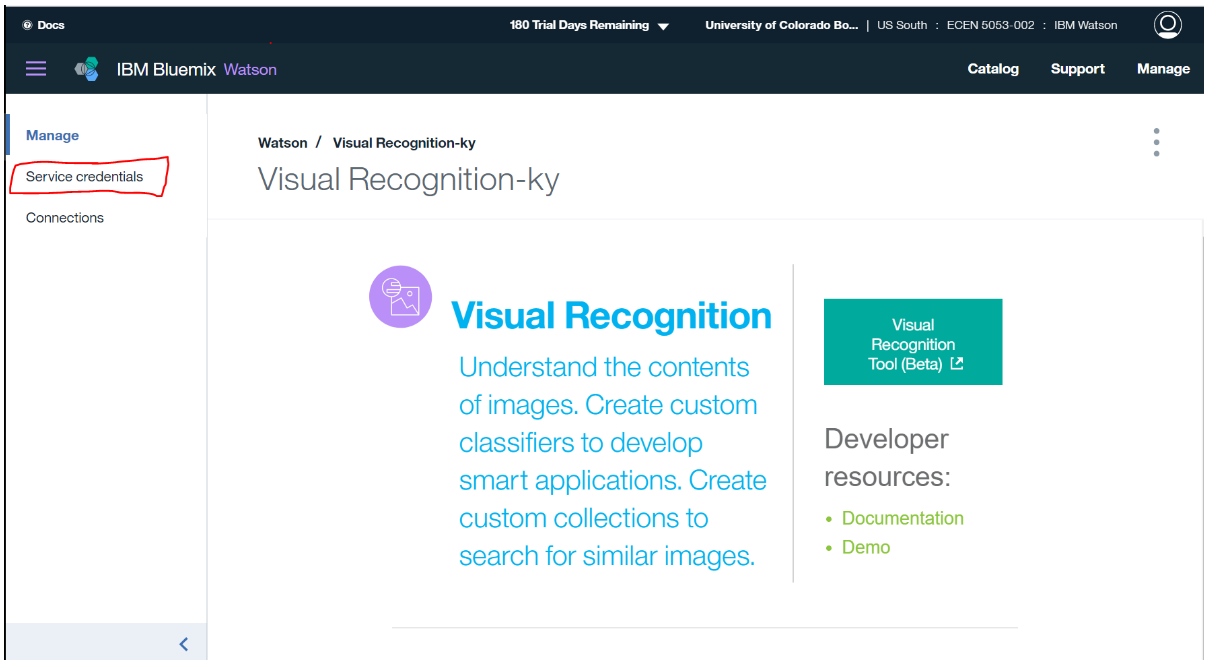


Edit debug node

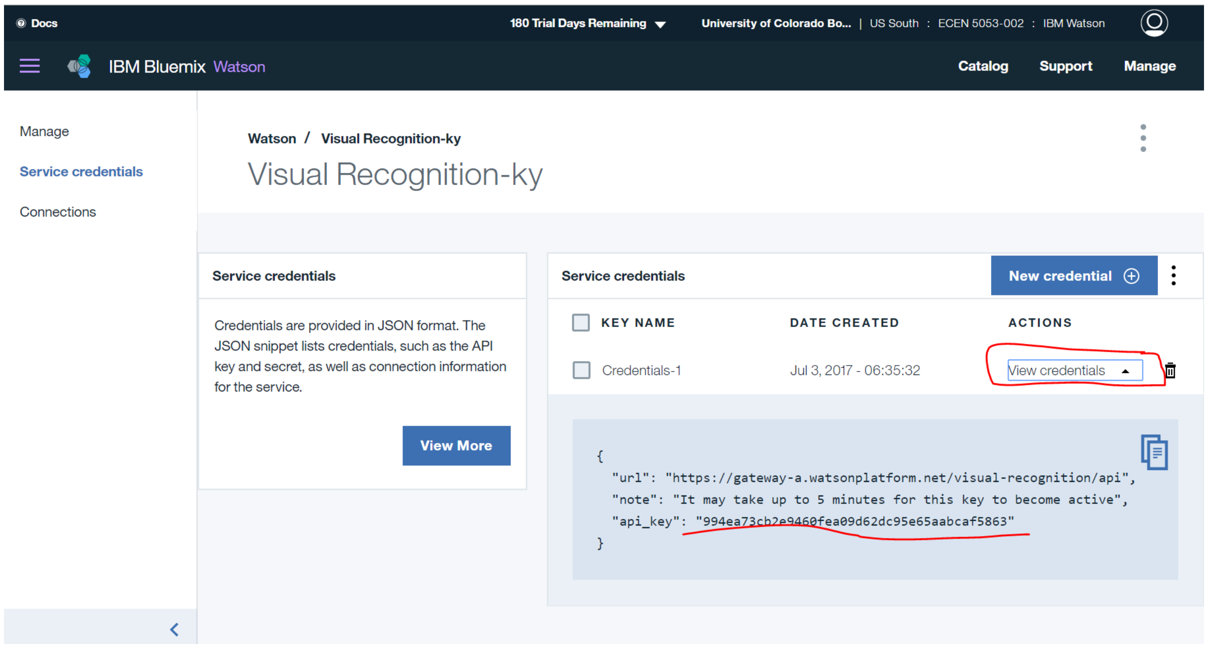
Output: complete msg object

to: debug tab

Name: Name



Understand the contents of images. Create custom classifiers to develop smart applications. Create custom collections to search for similar images.

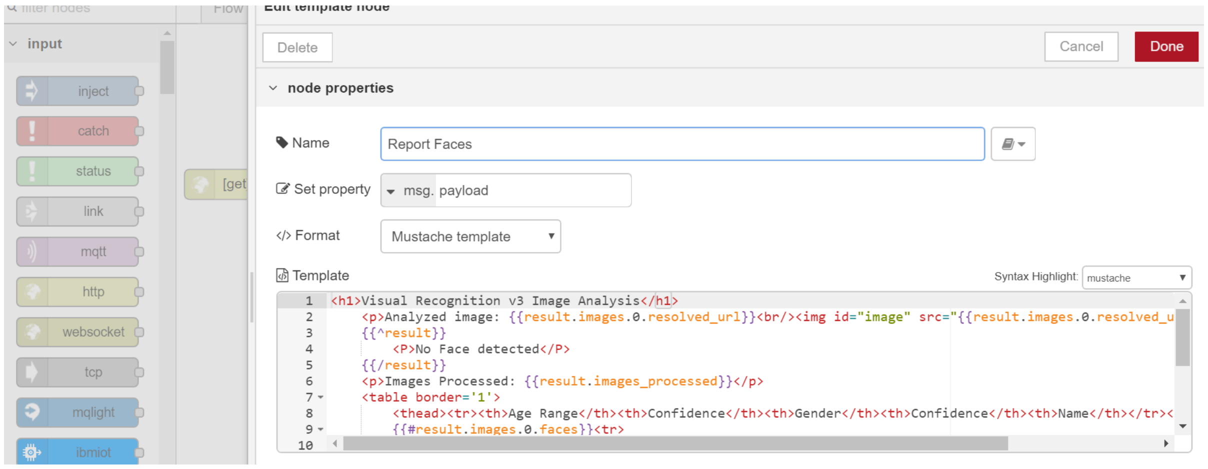


Credentials are provided in JSON format. The JSON snippet lists credentials, such as the API key and secret, as well as connection information for the service.

KEY NAME	DATE CREATED	ACTIONS
Credentials-1	Jul 3, 2017 - 06:35:32	View credentials

```
{
  "url": "https://gateway-a.watsonplatform.net/visual-recognition/api",
  "note": "It may take up to 5 minutes for this key to become active",
  "api_key": "994ea73cb2e9460fea09d62dc95e65aabcaf5863"
}
```

- Drag and drop a **Template** node with the following contents, which will format the output returned from the Image Analysis node into an HTML table for easier reading. Override the existing line of code.



Name: Report Faces

Set property: msg. payload

Format: Mustache template

Template

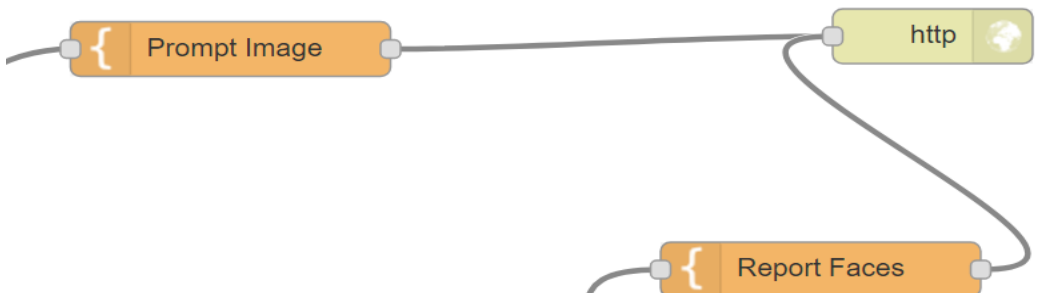
Syntax Highlight: mustache

```
<h1>Visual Recognition v3 Image Analysis</h1>
    <p>Analyzed image: {{result.images.0.resolved_url}}<br/><img id="image" src="{{result.images.0.resolved_u
    {{^result}}
        <P>No Face detected</P>
    {{/result}}
    <p>Images Processed: {{result.images_processed}}</p>
    <table border='1'>
        <thead><tr><th>Age Range</th><th>Confidence</th><th>Gender</th><th>Confidence</th><th>Name</th></tr><
        {{#result.images.0.faces}}<tr>
```

Here is the code for the template frame

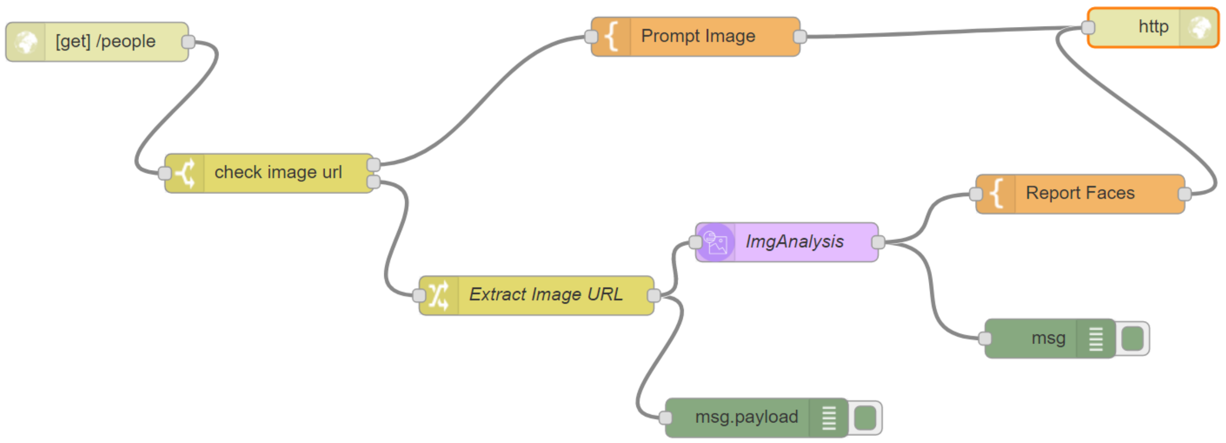
```
<h1>Visual Recognition v3 Image Analysis</h1>

    <p>Analyzed image: {{result.images.0.resolved_url}}<br/><img
id="image" src="{{result.images.0.resolved_url}}" height="200"/></p>

    {{^result}}

        <P>No Face detected</P>

    {{/result}}

    <p>Images Processed: {{result.images_processed}}</p>

    <table border='1'>

        <thead><tr><th>Age Range</th><th>Confidence</th><th>Gender</
th><th>Confidence</th><th>Name</th></tr></thead>

        {{#result.images.0.faces}}<tr>

            <td><b>{{age.min}} - {{age.max}}</b></
td><td><i>{{age.score}}</i></td>

            <td>{{gender.gender}}</td><td>{{gender.score}}</td>

            <td>{{identity.name}} ({{identity.score}})</td>

        </tr>{{/result.images.0.faces}}

    </table>

    <form  action="{{req._parsedUrl.pathname}}">

        <br><input type="submit" value="Try again or go back to the
home page"/>

    </form>
```

- End the flow with the HTTP response node and connect the nodes as shown in the figure below.



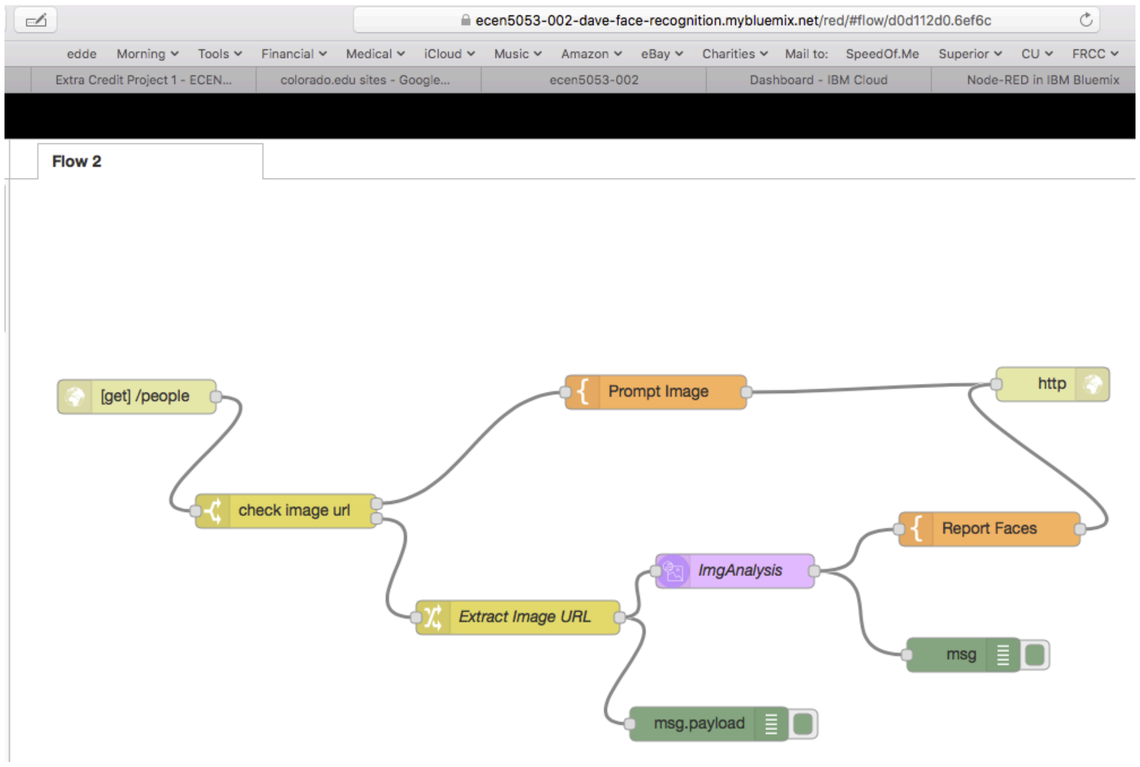
- Click **Deploy**

Before deploying, just make sure that the flow of your face recognition should look like this (the figure is shown below).



- Your application is now active.

Note your "host name". In my NODE RED editor it is the character string prior to mybluemix.net. Mine was:


ecen5053-002-dave-face-recognition.mybluemix.net/red/#flow/d0d112d0.6ef6c

So the path to your site is:

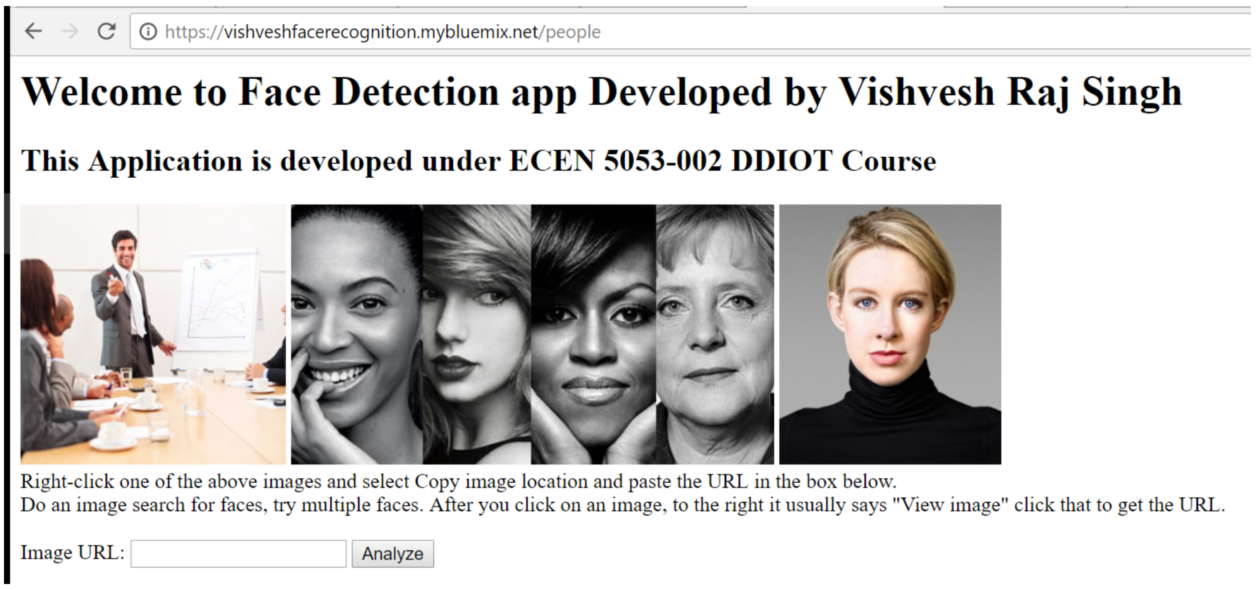
http://[yourhost].mybluemix.net/people

Mine is: https://ecen5053-002-dave-face-recognition.mybluemix.net/people

Expected Results

Find an image of a person on the web. Obtain its URL. Paste that URL into the "Image URL" field on your web page.

[**Note**: Click View Image to obtain the exact link to that image, and copy that link into the Webpage's text editor.]



https://vishveshfacerecognition.mybluemix.net/people

Welcome to Face Detection app Developed by Vishvesh Raj Singh

This Application is developed under ECEN 5053-002 DDIOT Course

Right-click one of the above images and select Copy image location and paste the URL in the box below.
Do an image search for faces, try multiple faces. After you click on an image, to the right it usually says "View image" click that to get the URL.

The output of the above steps will not be the same as shown above, but it would be very similar.

IMPORTANT NOTE: If in any case, you do not get the desired output, then you must be doing something wrong in any of the given steps. So, you need to make sure that you follow all the steps.

If in case of any difficulty, please contact Professor or TA for more help