

Cognitive Home - Watson Visual Recognition Lab

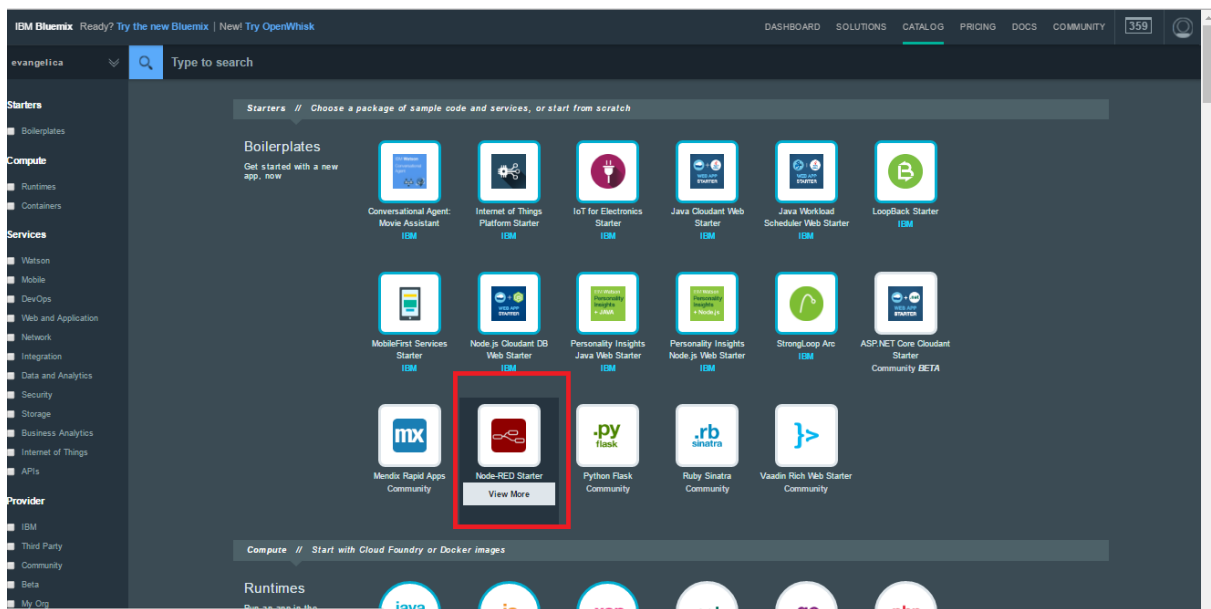
Created by Shubhradeep Nandi (AI, Cognitive and DataScience expert)

1 OBJECTIVE

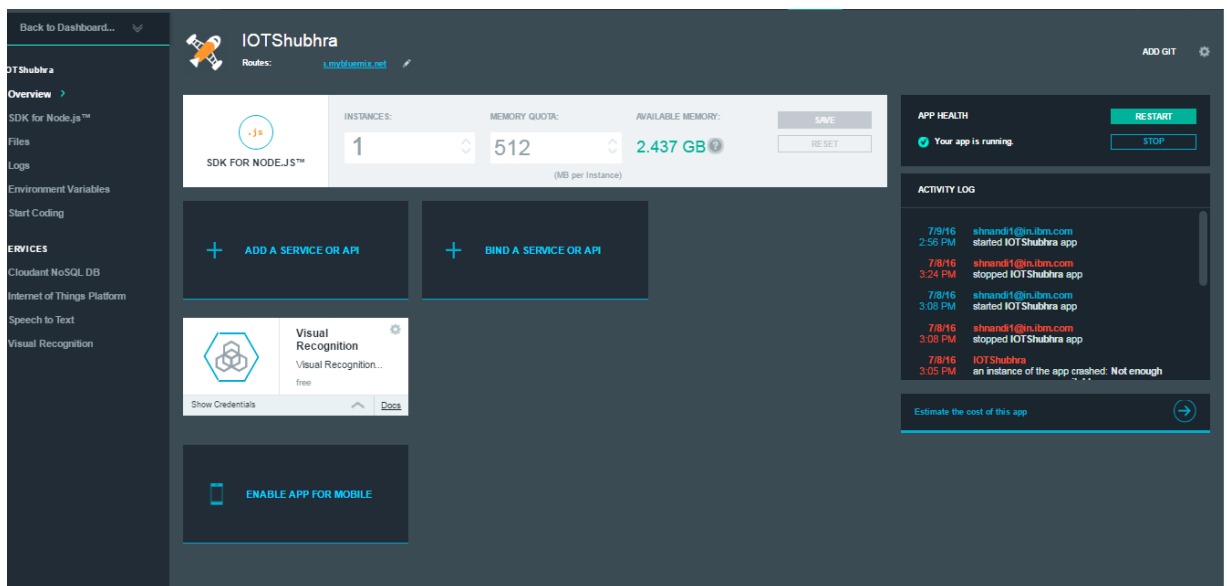
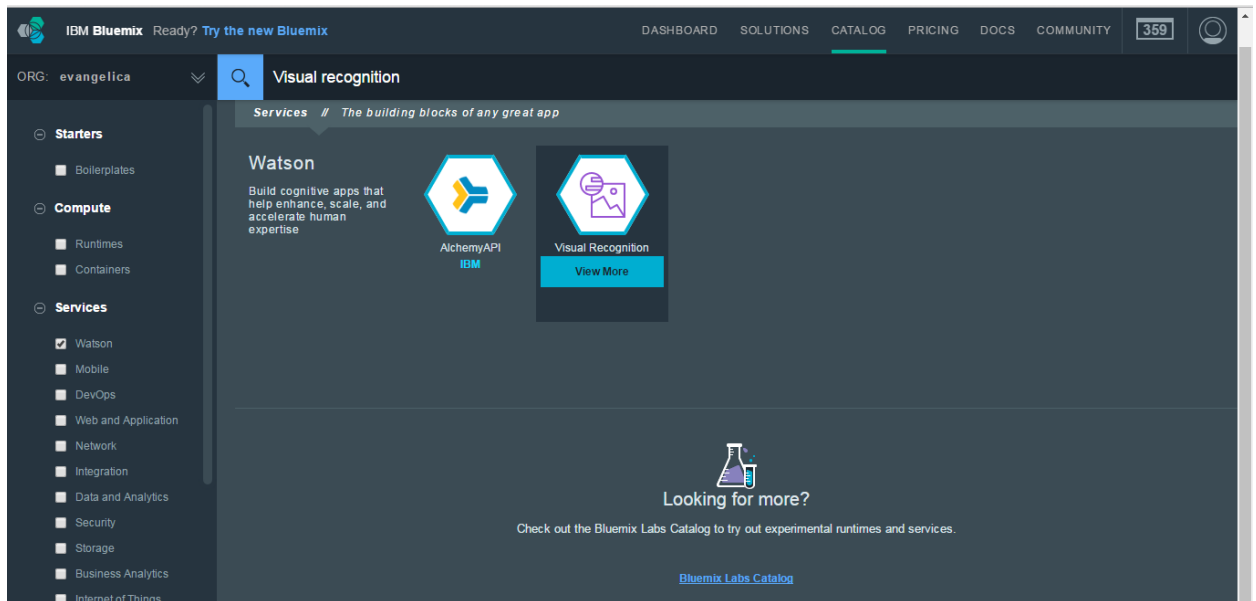
To verify if Watson visual recognition system can distinguish between smile and Anger. This is something that has never been tried before.

2 CREATE THE APPLICATION BASE

1. Click on the <https://console.ng.bluemix.net/catalog/starters/node-red-starter/>
2. Create a Node-Red Application



3. Add a Watson Visual Recognition service to the application.



3 TRAINING THE SYSTEM

1. Take around 20 smiley selfies and 20 angry selfies of a single Individual.
2. Take around 20 images where this individual is neither smiling nor angry.
3. Resize the images to 600X800 pixel in size.

4. Create a zip file each for two set of images name them as smile.zip and anger.zip.
5. Go to <http://visual-recognition-demo.mybluemix.net/train>
 - Try to upload your zip files and create a custom classifier.

visual-recognition-demo.mybluemix.net/train

Try **Train**

Positive Classes

Upload your own images in zipped folders to the spaces below.
Zipfile size must not exceed 5 MB.

Select or drag a zipped folder with at least 50 images

Name of This Class

Select or drag a zipped folder with at least 50 images

Name of This Class

Select or drag a zipped folder with at least 50 images

Name of This Class

Optional Negative Classes

Select or drag a zipped folder with at least 50 images

Use images that may seem like the bundles but are different.

Upload at least two image bundles.

Traindata

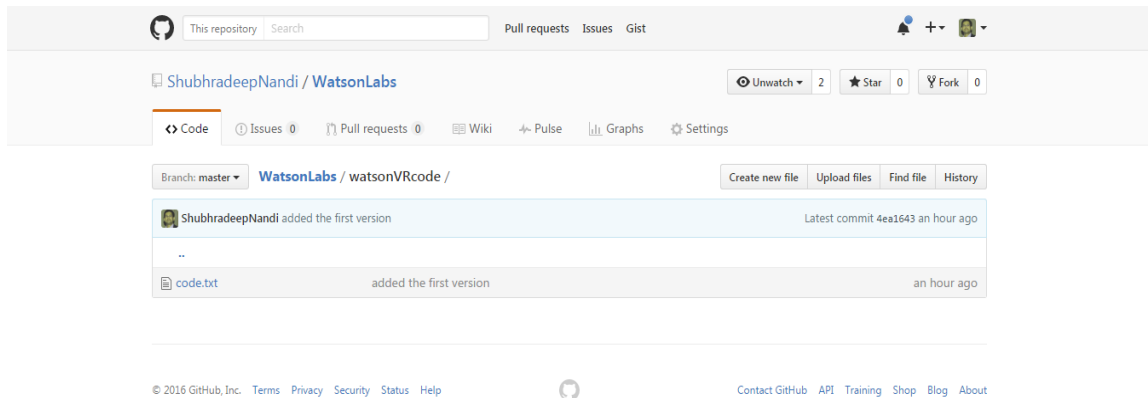
Train your classifier

[Reset](#)

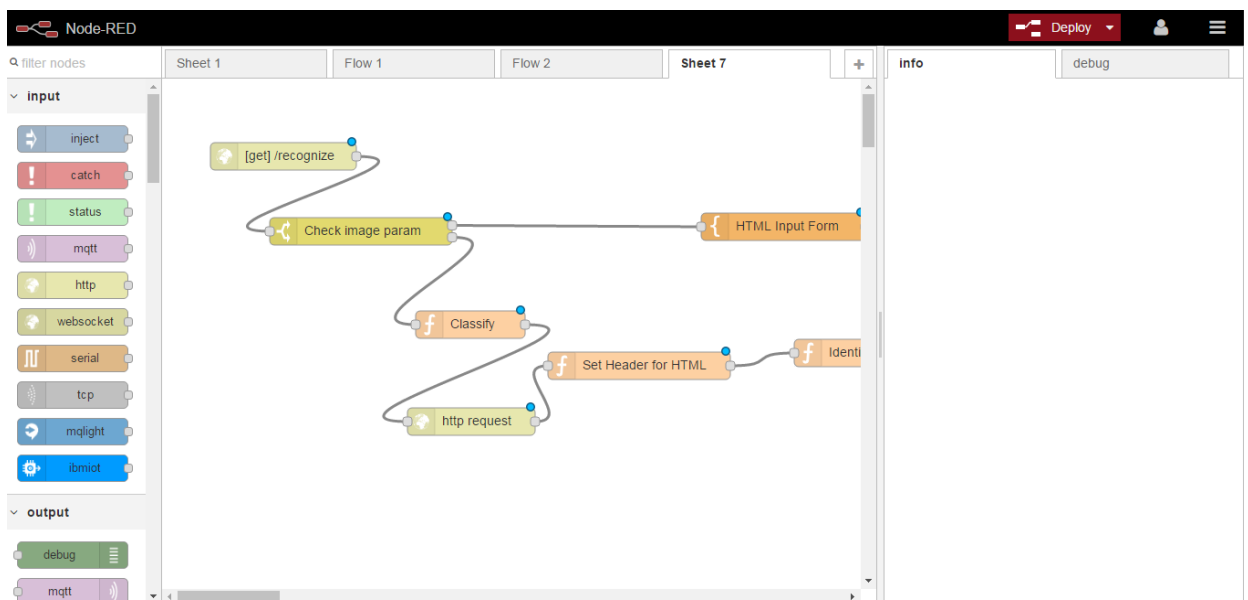
6. If step 3 does not work for you then follow step 5-7
7. (Optional) Alternatively, open up the command prompt.
8. (Optional) Go to your folder where the selfie zips are stored.
9. (Optional) Run the below curl command to train.
 - `curl -X POST -F "<smile-name>_positive_examples=@smile.zip" -F "<anger_positive_examples=@anger.zip" -F "negative_examples=@negative.zip" -F "name=selfiesec" "https://gateway-a.watsonplatform.net/visual-recognition/api/v3/classifiers?api_key=<Your_API_KEY>&version=19-05-2016"`
10. This will take up to a maximum 15 minutes for the system to get trained with your customer data.
11. Note the classifier id for future references.

4 CREATING THE YOUR VISUAL AUTHENTICATION SYSTEM.

1. Go to GitHub repo <https://github.com/ShubhradeepNandi/WatsonLabs.git> and click on watch + star + fork



2. Get the code.txt from **watsonVRcode**
3. Import the code text in to Node Red application Sheet space.



4. Modify the below items :-
 - In the **Classify** function node modify this line `"msg.url="https://gateway-a.watsonplatform.net/visual-recognition/api/v3/classify?&api_key=1234myapikey&url="+encodeURIComponent(msg.payload.url)+"&classifier_ids=<ADD YOUR OWN CLASSIFIER ID>&owners=me,IBM&threshold=0.2&version=19-05-2016";"`
 - In the **Identification** function node modify this line `"if(msg.class === '<Add your first class>' || msg.class === '<Add your second class>'){ "`
5. Deploy the code and you are done.

5 TEST IT OUT

1. Make a GET call to the below API :-

- **<Your Node Red App url> /recognize?url=<Image url>**
- A parameter '**url**' should be appended to the api call where you need to send an Image URL
[How and where to upload an Image will be informed during the Lab]

2. This application will send a JSON response in the below format :-

```
{"custom_classes":2,"images":[{"classifiers":[{"classes":[{"class":"smile","score":0.237042}],"classifier_id":"myclassifier","name":"selfiesec"},"resolved_url":"https://***/$$$$/shubhra2.jpg","source_url":"https://***/$$$$/shubhra2.jpg"}],"images_processed":1,"authorize":1}
```

'class' is the parameter to look for.