

Final Project Documentation

Komal Patel

(801215979)

Index

Title	Page No.
Introduction to computer vision API v1.0	2
Why to use v1.0 instead of original azure API?	2
Dependencies used	2
Computer vision API v1.0 Endpoints:	3
To Run the project:	24

Introduction to Computer Vision API v1.0:

Computer Vision API provides facility to process images and returns information contained in the images. It can be used to determine if image contains landmarks, any celebrity faces, it can provide list of tags related to image provided and it also can describe the image based on objects available in the image. It also provides flexibility with the language. User can specify the language when requesting tag image API of and get the response of tags in the specified language. APIs are developed in using Node.js programming language.

Why to use v1.0 instead of original azure API?

- No need to sign up with Microsoft Azure
- No need to pay subscription charges
- Easy to use using swagger playground
- Can be used with different programming languages

Dependencies Used:

- **Express:** To create REST-like API, express is used to communicate as a client to the server. Version used is 4.17.1.
- **Axios:** It is a java script library that is used to make HTTP request from the node.js application. (Version used 0.24.0)
- **Cors:** It stands for Cross-origin resource sharing, which is used to perform a cross-domain request. (Version used 2.8.5)
- **Dotenv:** This dependency is used to load environment variables from *.env file to process.env. This make sures that secure data is kept in the secure location. The example of this data is uploaded to version control and not the actual data. (Version used 10.0.0)
- **Swagger-ui:** This is used to make visual documentation for the OpenAPI specification. (Version used 4.1.3)
- **Swagger-ui-express:** This is allowing to server auto-generated swagger-ui API docs from express framework. (Version used 4.2.0)
- **Swagger-jsdoc:** This dependency reads source code of the API generated and produces an OpenAPI specification. (Version used 6.1.0)
- **Npm:** nodejs package manager.

Computer vision API v1.0 Endpoints:

1. 137.184.155.171:3000/api/v1/analyzeImage?details=landmarks

Description:

Analyze image API endpoint extracts image details and provides content of the image. It allows specifying image URL from the web to analyze the image. To utilize the maximum features of the API, query parameters with details needs to use, which are optional yet very useful parameters for this API endpoint. This endpoint analyzes landmarks in the picture if any and returns name of the landmark in the response.

Method: Post

Headers: 'Content-Type': 'application/json',

'Ocp-Apim-Subscription-Key': key

Request Header: 'Content-Type': 'application/json'

Query parameters: details: landmarks

Request body:

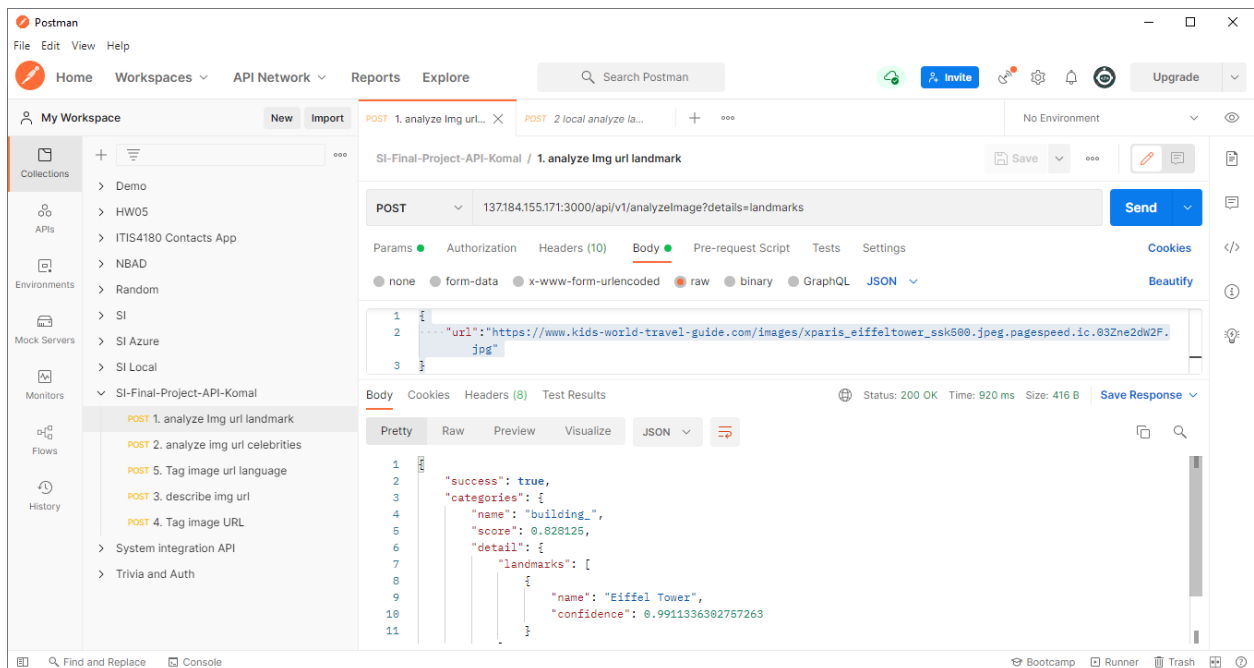
```
{
  "url": "https://www.kids-world-travel-
guide.com/images/xparis_eiffeltower_ssk500.jpeg.pagespeed.ic.03Zne2dW2F.jpg"
}
```

Body-type: Json data

Body parameter name: url

Status Codes:

1. 200: Success
2. 400: Bad request or invalid input in query parameters or body
3. 500: Internal server error

Image:**Postman API call:****API Response:**

```
{
  "success": true,
  "categories": {
    "name": "building_",
    "score": 0.828125,
    "detail": {
      "landmarks": [
        {
          "name": "Eiffel Tower",
```

```
}
  }
}
}
}
```

"confidence": 0.9911336302757263

Swagger API Call:

newFinalProjectServer - DigitalO x Happy holidays, Niner Nation! - x Swagger UI x +

Not secure | 137.184.155.171:3000/docs/#/Analyze%20Image/post_api_v1_analyzeImage

Analyze Image

POST /api/v1/analyzeImage API will take image url, provides categories based on query parameters as a response

Parameters

Name	Description
details string (query)	Choose any details category to analyze image with more accurate details
Image URL object (body)	Provide URL of image to analyzed (Either URL or file upload is allowed)

Parameter content type: application/json

newFinalProjectServer - DigitalO x Happy holidays, Niner Nation! - x Swagger UI x +

Not secure | 137.184.155.171:3000/docs/#/Analyze%20Image/post_api_v1_analyzeImage

Curl

```
curl -X 'POST' \
  'http://137.184.155.171:3000/api/v1/analyzeImage?details=landmarks' \
  -H 'accept: application/json' \
  -H 'content-type: application/json' \
  -d '{
    "url": "https://www.kids-world-travel-guide.com/images/xparis_eiffeltower_ssk500.jpeg.pagespeed.ic.03zne2d4ZF.jpg"
  }'
```

Request URL

http://137.184.155.171:3000/api/v1/analyzeImage?details=landmarks

Server response

Code	Details
200	<pre>{ "success": true, "categories": { "name": "Building_", "score": 0.928125, "detail": { "landmarks": [{ "name": "Eiffel Tower", "confidence": 0.9911336302757263 }] } } }</pre>

Response headers

```
access-control-allow-origin: *
connection: keep-alive
content-length: 148
content-type: application/json; charset=utf-8
date: Thu, 16 Dec 2021 01:56:23 GMT
etag: W/"94-ql1cV9ahr500vpr0kxprlequ"
keep-alive: timeout=5
x-powered-by: Express
```

2. 137.184.155.171:3000/api/v1/analyzeImage?details=celebrities

Description:

Analyze image API endpoint extracts image details and provides content of the image. It allows specifying image URL from the web to analyze the image. To utilize the maximum features of the API, query parameters with details needs to use, which are optional yet very useful parameters for this API endpoint. This endpoint analyzes celebrities in the picture and returns the array of name of the celebrities as well as other details such as confidence of the response and the dimensions of face in a rectangle.

Method: Post

Headers: 'Content-Type': 'application/json',

'Ocp-Apim-Subscription-Key': key

Request Header: 'Content-Type': 'application/json'

Query parameters: details: celebrities

Request body:

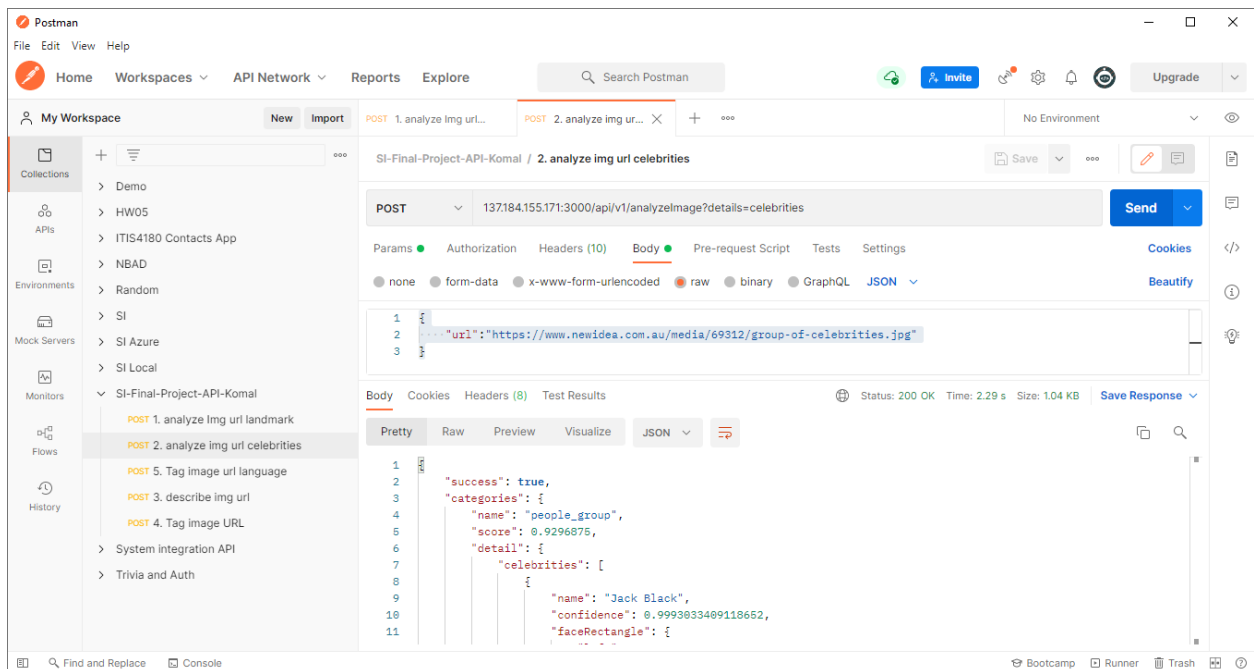
```
{  
  "url": "https://www.newidea.com.au/media/69312/group-of-celebrities.jpg"  
}
```

Body-type: Json data

Body parameter name: url

Status Codes:

1. 200: Success
2. 400: Bad request or invalid input in query parameters or body
3. 500: Internal server error

Image:**Postman Api call:****API Response:**

```
{
  "success": true,
  "categories": {
    "name": "people_group",
    "score": 0.9296875,
    "detail": {
```

```
"celebrities": [  
  {  
    "name": "Jack Black",  
    "confidence": 0.9993033409118652,  
    "faceRectangle": {  
      "left": 822,  
      "top": 127,  
      "width": 62,  
      "height": 62  
    }  
  },  
  {  
    "name": "Kevin Hart",  
    "confidence": 0.9870463609695435,  
    "faceRectangle": {  
      "left": 654,  
      "top": 178,  
      "width": 61,  
      "height": 61  
    }  
  },  
  {  
    "name": "Dwayne Johnson",  
    "confidence": 0.9999815225601196,  
    "faceRectangle": {  
      "left": 468,  
      "top": 65,  
      "width": 60,  
      "height": 60  
    }  
  },  
  {  
    "name": "Jake Kasdan",  
    "confidence": 0.9986875653266907,  
    "faceRectangle": {  
      "left": 128,  
      "top": 164,  
      "width": 57,  
      "height": 57  
    }  
  },  
  {  
    "name": "Karen Gillan",  
    "confidence": 0.997129499912262,  
  }  
]
```



```
"faceRectangle": {
  "left": 592,
  "top": 80,
  "width": 54,
  "height": 54
},
{
  "name": "Nick Jonas",
  "confidence": 0.9989489912986755,
  "faceRectangle": {
    "left": 268,
    "top": 140,
    "width": 53,
    "height": 53
  }
}
]
}
}
```

Swagger API Call:

newFinalProjectServer - DigitalO... Happy holidays, Niner Nation! - Swagger UI

Not secure | 137.184.155.171:3000/docs/#/Analyze%20Image/post_api_v1_analyzeImage

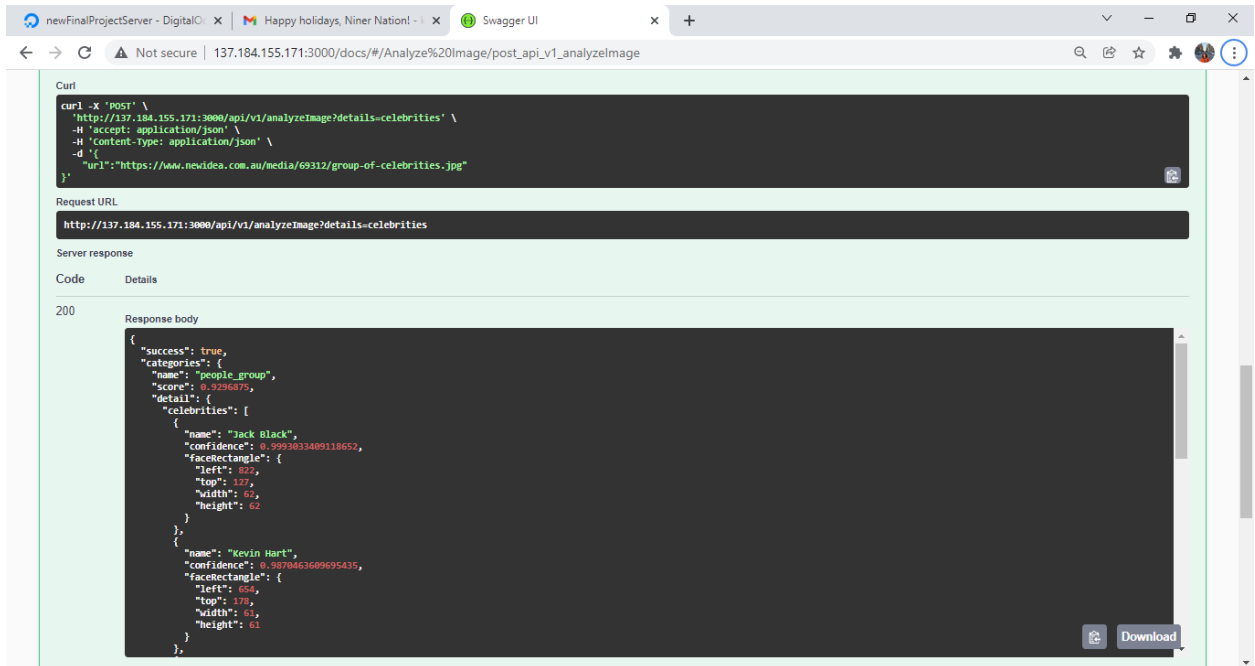
Analyze Image

POST /api/v1/analyzeImage API will take image url, provides categories based on query parameters as a response

Parameters

Name	Description
details string (query)	Choose any details category to analyze image with more accurate details <input type="text" value="celebrities"/>
Image URL object (body)	Provide URL of image to analyzed (Either URL or file upload is allowed) <div>Edt Value Model <pre>{ "url": "https://www.newidea.com.au/media/69312/group-of-celebrities.jpg" }</pre></div>

Parameter content type



3. 137.184.155.171:3000/api/v1/describeImage

Description:

Describe image API endpoint provides a description of an image in human readable language with complete sentences. It allows specifying image URL from the web to get the description of the image. It returns an array of tags and a sentence of description of an image.

Method: Post

Headers: 'Content-Type': 'application/json',

'Ocp-Apim-Subscription-Key': key

Request Header: 'Content-Type': 'application/json'

Request body:

```
{  
  "url": "https://metro.co.uk/wp-content/uploads/2015/11/dog2.jpg?quality=90&strip=all"  
}
```

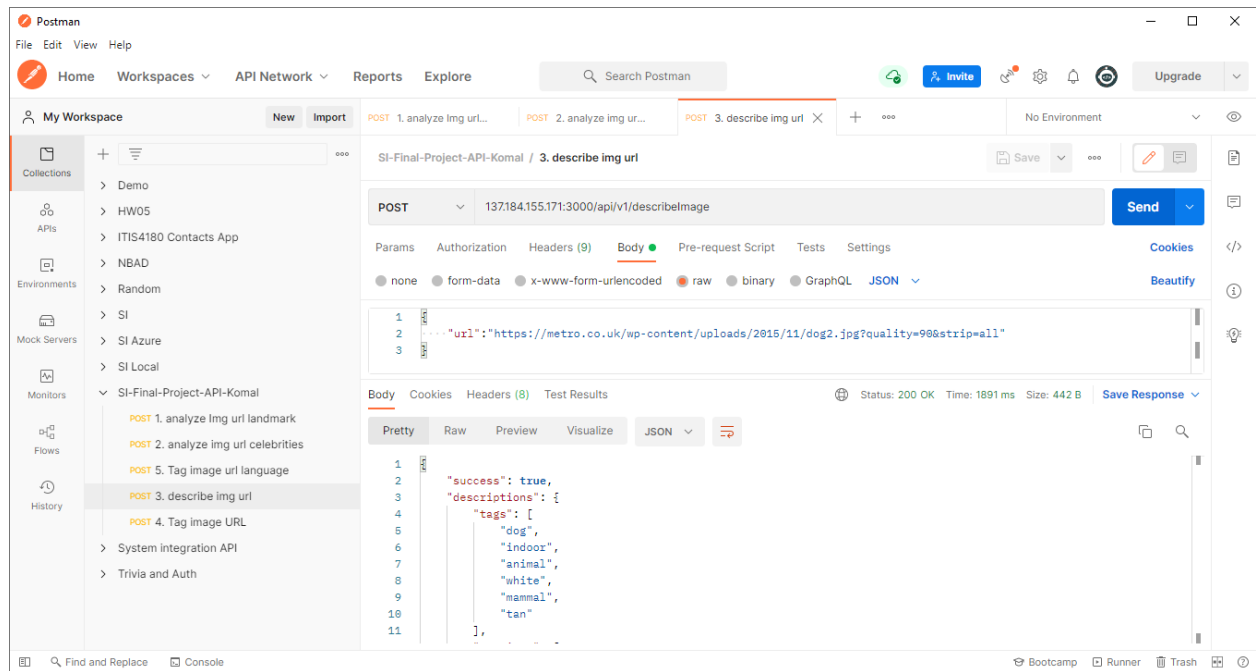
Body-type: Json data

Body parameter name: url

Status Codes:

1. 200: Success
2. 400: Bad request or invalid input in query parameters or body
3. 500: Internal server error

Image:

Postman API call:**API Response:**

```

{
  "success": true,
  "descriptions": {
    "tags": [
      "dog",
      "indoor",
      "animal",
      "white",
      "mammal",
      "tan"
    ],
    "captions": [
      {
        "text": "a puppy sitting on the floor",
        "confidence": 0.5006706118583679
      }
    ]
  }
}

```

Swagger API Call:

The image shows two screenshots of the Swagger UI interface. The top screenshot displays the configuration for a POST request to the endpoint `/api/v1/describeImage`. The request body is a JSON object with a single key `url` pointing to a dog image. The content type is set to `application/json`. The bottom screenshot shows the result of executing this request, displaying the request URL, the 200 status code, and the JSON response body which includes image tags, a caption, and a confidence score.

POST `/api/v1/describeImage` API will take image url, provides description of the image as a response

Parameters

Name	Description
Image URL	provide URL of image to get the description

object (body)

Edit Value | Model

```
{  "url": "https://metro.co.uk/wp-content/uploads/2015/11/dog2.jpg?quality=90&strip=all"}
```

Parameter content type: `application/json`

Execute **Clear**

Responses Response content type: `application/json`

Curl

```
curl -X 'POST' \
  'http://137.184.155.171:3000/api/v1/describeImage' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "url": "https://metro.co.uk/wp-content/uploads/2015/11/dog2.jpg?quality=90&strip=all"
  }'
```

Request URL

`http://137.184.155.171:3000/api/v1/describeImage`

Server response

Code **Details**

200

Response body

```
{  "success": true,  "descriptions": {    "tags": [      "dog",      "indoor",      "animal",      "white",      "mammal",      "tan"    ],    "captions": [      {        "text": "a puppy sitting on the floor",        "confidence": 0.5806786118383679      }    ]  } }
```

Response headers

```
access-control-allow-origin: *
connection: keep-alive
content-length: 174
content-type: application/json; charset=utf-8
date: Thu, 18 Dec 2021 01:59:30 GMT
etag: W/"9751e8vui6uXQ58ZAn1s2geu/E"
keep-alive: timeout=5
x-powered-by: Express
```

Responses

Code	Description
200	

4. 137.184.155.171:3000/api/v1/tagImage

Description:

Tag image API endpoint generates tags and/or list of words related to image. It allows specifying image URL from the web to analyze the image. This is not same as describe image endpoints' tags as it does not organize to hierarchical classification system but correspond to image content. Default language is English for this API endpoint.

Method: Post

Headers: 'Content-Type': 'application/json',

'Ocp-Apim-Subscription-Key': key

Request Header: 'Content-Type': 'application/json'

Request body:

```
{  
  "url": "https://upload.wikimedia.org/wikipedia/commons/c/c8/Taj_Mahal_in_March_2004.jpg"  
}
```

Body-type: Json data

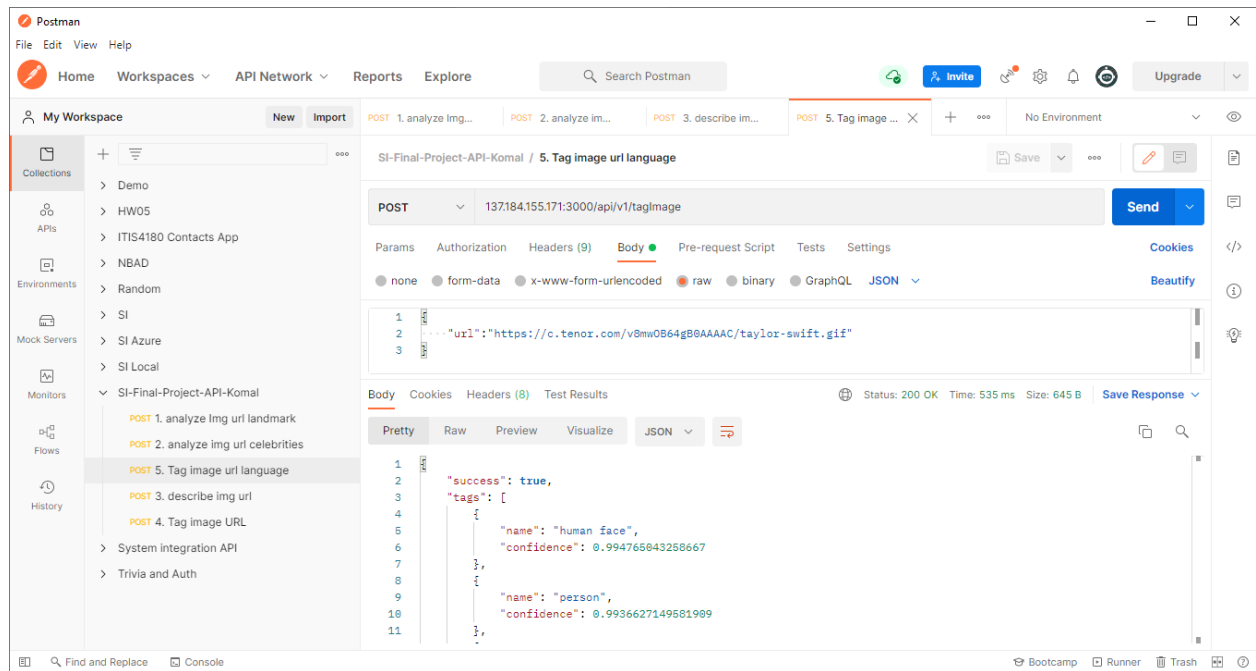
Body parameter name: url

Status Codes:

1. 200: Success
2. 400: Bad request or invalid input in query parameters or body
3. 500: Internal server error

Image:



Postman Api call:**API Response:**

```

{
  "success": true,
  "tags": [
    {
      "name": "human face",
      "confidence": 0.994765043258667
    },
    {
      "name": "person",
      "confidence": 0.9936627149581909
    },
    {
      "name": "clothing",
      "confidence": 0.9665981531143188
    },
    {
      "name": "smile",
      "confidence": 0.9225829839706421
    },
    {
      "name": "woman",
      "confidence": 0.9091215133666992
    }
  ]
}

```

```
    },  
    {  
      "name": "lady",  
      "confidence": 0.8437873125076294  
    },  
    {  
      "name": "indoor",  
      "confidence": 0.7944323420524597  
    }  
  ]  
}
```

Swagger API Call:

The screenshot shows the Swagger UI for a POST endpoint: `/api/v1/tagImage`. The description states: "API will take image url, provides tags in specified language based on query parameters".

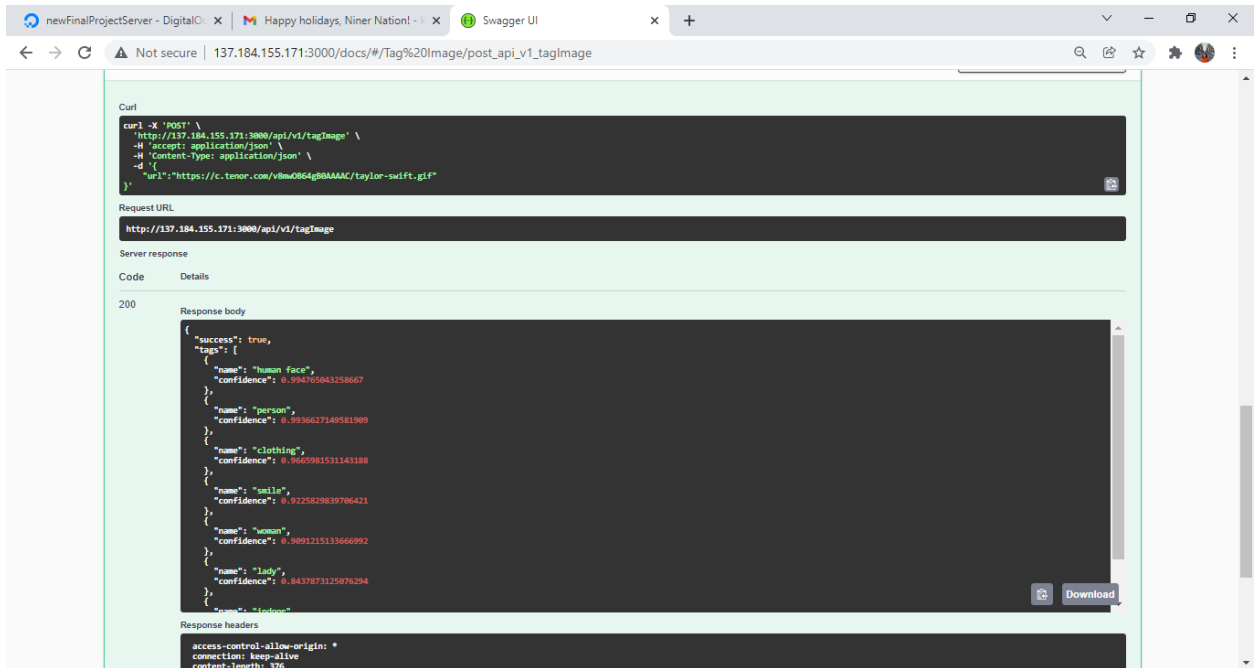
Parameters:

Name	Description
language string (query)	Choose any language from the list and response will be displayed in that language
Image URL object (body)	provide URL of image to get the tags

The **Image URL** parameter is expanded, showing a JSON body:

```
{  
  "url": "https://c.tenor.com/v8mw0864g80AAAC/taylor-swift.gif"  
}
```

At the bottom, the **Parameter content type** is set to `application/json`. The interface includes **Cancel**, **Execute**, and **Clear** buttons.



5. 137.184.155.171:3000/api/v1/tagImage?language=hi**Description:**

This is the tag image endpoint with extension of optional query parameter. With query parameter language, this API facilitates to receive response in specified language, and provides the tags in that language. This supports various language response, however, in swagger few of them are included. The list of language it supports are as below:

Language	language code
Arabic	ar
Azeri (Azerbaijani)	az
Bulgarian	bg
Bosnian Latin	bs
Catalan	ca
Czech	cs
Welsh	cy
Danish	da
German	de
Greek	el
English	en
Spanish	es
Estonian	et
Basque	eu
Finnish	fi
French	fr
Irish	ga
Galician	gl
Hebrew	he
Hindi	hi
Croatian	hr
Hungarian	hu
Indonesian	id
Italian	it
Japanese	ja
Kazakh	kk
Korean	ko
Lithuanian	lt
Latvian	lv
Macedonian	mk
Malay Malaysia	ms
Norwegian (Bokmal)	nb

Dutch	nl
Polish	pl
Dari	prs
Portuguese-Brazil	pt-BR
Portuguese-Portugal	pt/pt-PT
Romanian	ro
Russian	ru
Slovak	sk
Slovenian	sl
Serbian - Cyrillic RS	sr-Cryl
Serbian - Latin RS	sr-Latn
Swedish	sv
Thai	th
Turkish	tr
Ukrainian	uk
Vietnamese	vi
Chinese Simplified	zh/zh-Hans
Chinese Traditional	zh-Hant

Method: Post

Headers: 'Content-Type': 'application/json',

'Ocp-Apim-Subscription-Key': key

Request Header: 'Content-Type': 'application/json'

Query parameters: language=LANGUAGE_CODE

Request body:

```
{  
  "url": "https://www.fredmiranda.com/forum/ufiles/51/966851.jpg"  
}
```

Body-type: Json data

Body parameter name: url

Status Codes:

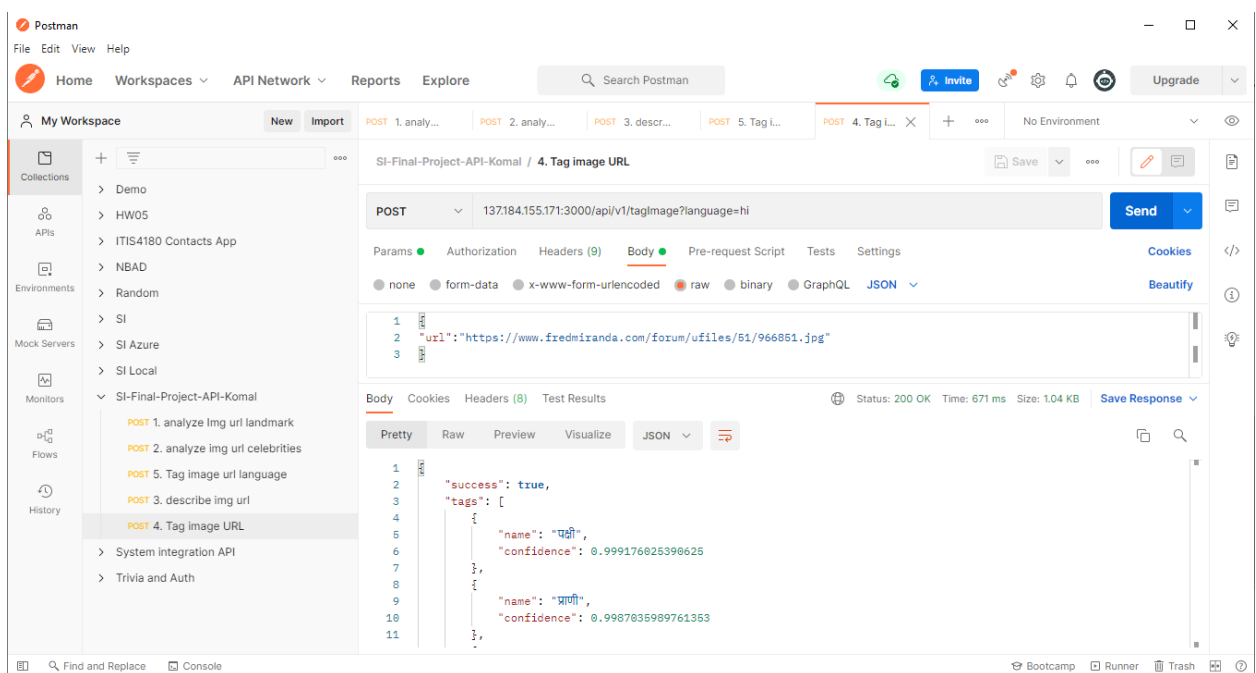
1. 200: Success
2. 400: Bad request or invalid input in query parameters or body

3. 500: Internal server error

Image:



Postman Api call:



API Response:

```
{
  "success": true,
  "tags": [
    {
      "name": "पक्षी",
      "confidence": 0.9991760849952698
    },
    {
      "name": "प्राणी",
      "confidence": 0.9987035989761353
    },
    {
      "name": "पेड़",
      "confidence": 0.9849432706832886
    },
    {
      "name": "बाहर",
      "confidence": 0.9817177653312683
    },
    {
      "name": "लाल",
      "confidence": 0.9631857872009277
    },
    {
      "name": "उत्तरी कार्डिनल (एक प्रकार का पक्षी)",
      "confidence": 0.9378284811973572
    },
    {
      "name": "मौलिक",
      "confidence": 0.9174913763999939
    },
    {
      "name": "चोंच",
      "confidence": 0.8912600874900818
    },
    {
      "name": "शाखा",
      "confidence": 0.8376626372337341
    },
    {

```

```
"name": "पकड़ कर बैठने वाली चिड़िया",  
"confidence": 0.645004391670227  
},  
{  
  "name": "वन्यजीव",  
  "confidence": 0.48880407214164734  
}  
]  
}
```

Swagger API call:

The screenshot shows the Swagger UI for a POST endpoint `/api/v1/tagImage`. The description states: "API will take image url, provides tags in specified language based on query parameters".

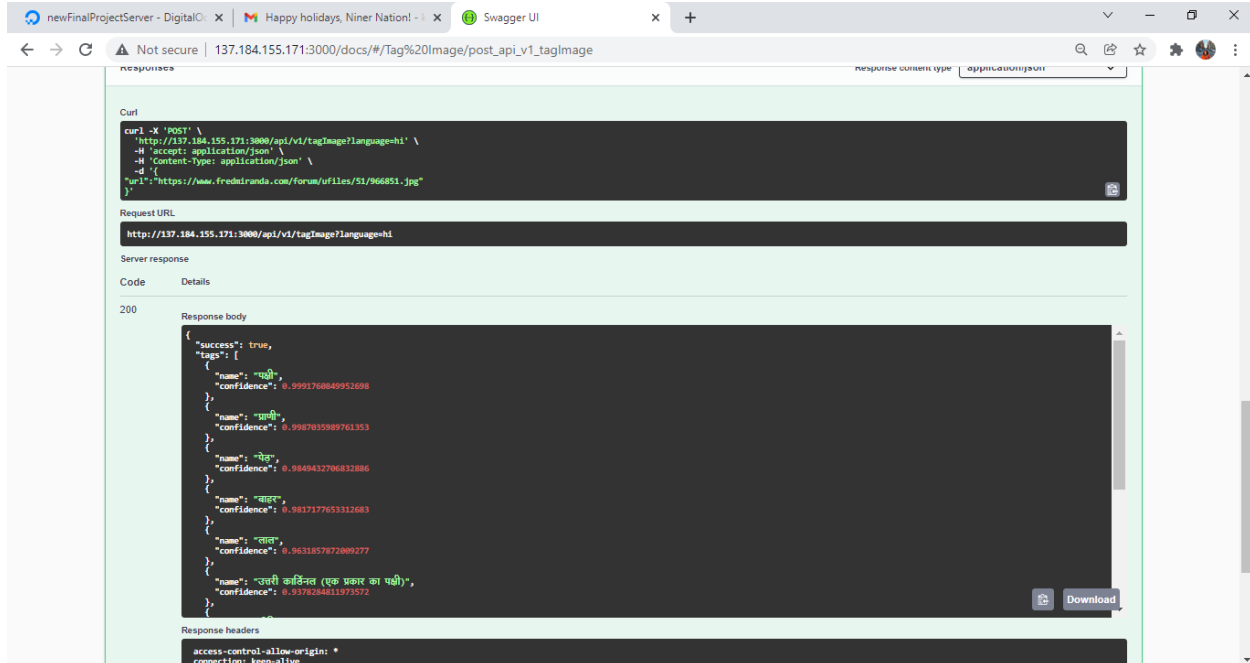
Parameters:

Name	Description
language string (query)	Choose any language from the list and response will be displayed in that language
Image URL object (body)	provide URL of image to get the tags

The **Image URL** parameter is expanded, showing a JSON body:

```
{  
  "url": "https://www.fredstiranda.com/forum/uf1les/51/966851.jpg"  
}
```

At the bottom, the **Parameter content type** is set to `application/json`. The interface includes **Execute** and **Clear** buttons.



To Run the project:

Host address: 137.184.155.171

Port: 3000

API endpoints:

1. 137.184.155.171:3000/api/v1/analyzeImage?details=landmarks
2. 137.184.155.171:3000/api/v1/analyzeImage?details=celebrities
3. 137.184.155.171:3000/api/v1/describeImage
4. 137.184.155.171:3000/api/v1/tagImage
5. 137.184.155.171:3000/api/v1/tagImage?language=hi

Github URL:

<https://github.com/Kpate2012/ComputerVisionApi>

Documentation playground:

<http://137.184.155.171:3000/docs/>

Import file to run in postman:

https://github.com/Kpate2012/ComputerVisionApi/blob/889114fd1b35cc16c35ac8ea0aa7c558b3b9dba6/SI-Final-Project-API-Komal.postman_collection.json

In postman, go to “file” and then “import”. Upload a file that you have downloaded from the above link.