## ITIS - 6177 - SYSYTEM INTEGRATION

#### FINAL PROJECT - FACE API

#### HARI VENKATESHA THONUPUNOORI - 801214665

# **FACE API**

#### AN API SERVICE THAT ANALYZES FACES IN IMAGES

This project is an embedded facial recognition application with a seamless and highly secured user experience. Features include face detection that perceives facial features and attributes – such as a face mask, glasses or face location in an image and identification of a person by a match to your private repository or via photo ID.

#### **API ENDPOINTS:**

#### **DETECT FACE CO-ORDINATES**

137.184.50.131:3000/detectface: Detects face coordinates for the given face present in image

Method: POST

**Headers:** 

Content-type: application/json
Ocp-Apim-Subscription-Key

Body Param: should be a JSON:

**Error Codes**:

200: Successful

400: Bad request or invalid request

500: Internal Server Error

### Sample:

### Input:

```
Detects the faces in image

Parameters

Description

ImageUri * required (hody)

Edit Value | Model

Cancel

Parameter content type |

application/json | v
```

## Output:

```
Responses

Response Coult

Curl x "0051" \

"http://37.184.59.131:3800/detectface" \

"http://37.184.59.131:3800/detectface" \

"http://37.184.59.131:3800/detectface" \

"http://17.184.59.131:3800/detectface" \

"scept: application/joon" \

"http://37.184.59.131:3800/detectface" \

"scept: application/joon" \

"scept: application/joon or application/j
```

**137.184.50.131:3000/detectfacelandmarks : D**etects face coordinates and landmarks like nose, eye, mouth coordinates if any face present in provided image

**Method: POST** 

#### **Headers:**

Content-type: application/json Ocp-Apim-Subscription-Key

## Body Param: should be a JSON:

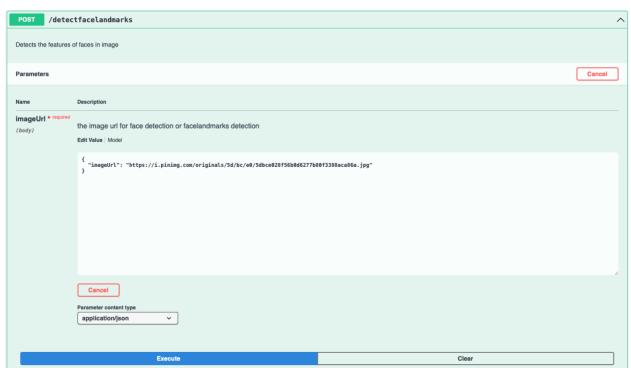
#### **Error Codes**:

200: Successful

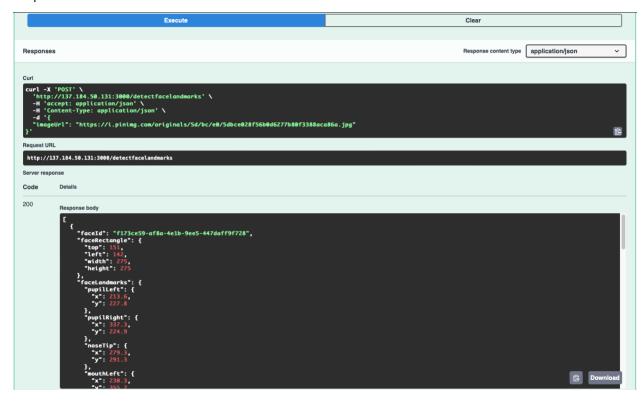
400: Bad request or invalid request

500: Internal Server Error

#### Sample:



## Output:



## **DETECT FACE LAND LANDMARKS**

**137.184.50.131:3000/detectfaceemotions :** This endpoint detects the face coordinates and emotions of face if any face present in image

Method: POST

#### **Headers:**

Content-type: application/json Ocp-Apim-Subscription-Key

## Body Param: should be a JSON:

{
 "imageUrl": "<provide image URL>"
}

#### **Error Codes:**

200: Successful

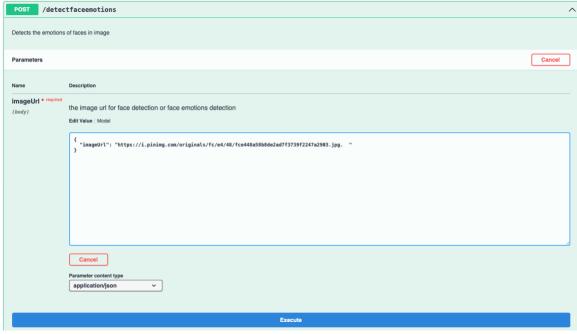
400: Bad request or invalid request

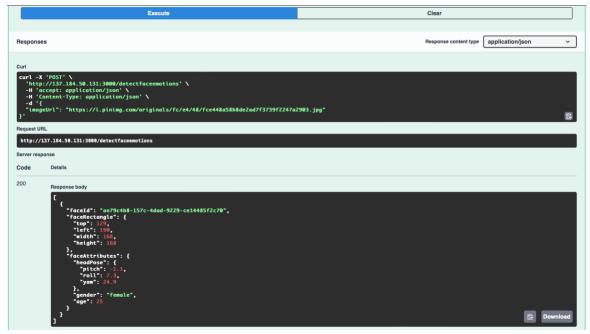
500: Internal Server Error

## Sample:

## Uploaded Image:







**137.184.50.131:3000/findsimilar:** This endpoint detects face coordinates and landmarks like nose, eye, mouth coordinates if any face present in provided image

**Method: POST** 

**Headers:** 

Content-type: application/json Ocp-Apim-Subscription-Key

## Body Param: should be a JSON:

```
{
    "imageUrl1": "<provide image URL>"
    "imageUrl2": "<provide image URL>"
}
```

## **Error Codes**:

200: Successful

400: Bad request or invalid request

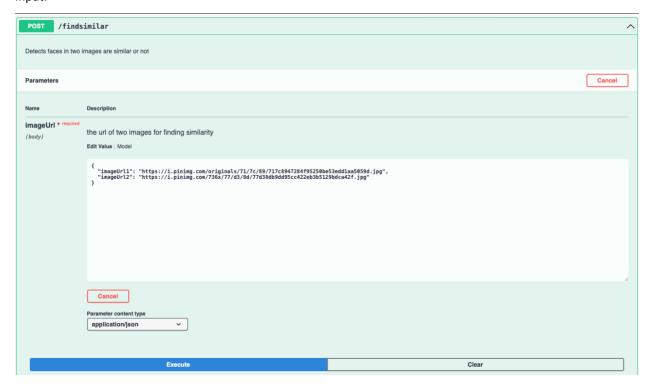
500: Internal Server Error

Sample Images to be compared:

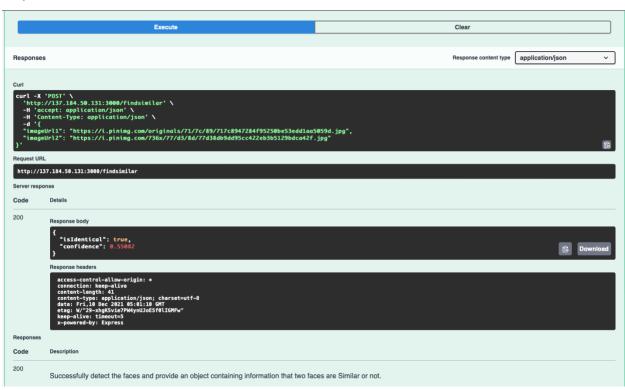




#### Input:

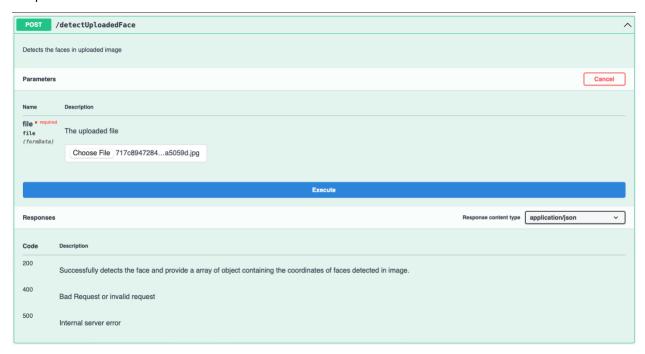


### Output:

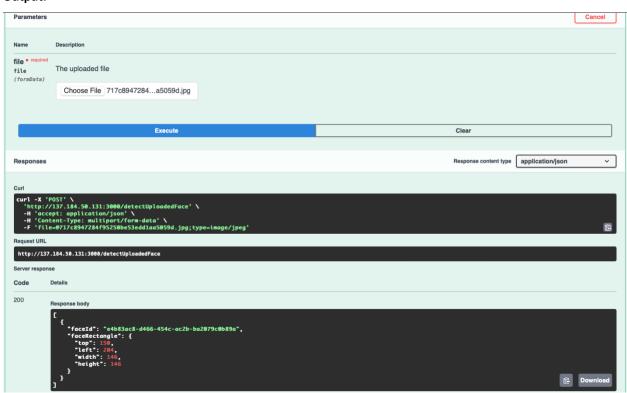


**137.184.50.131:3000/detectUploadedFace:** This endpoint accepts the image file and detects face coordinates and landmarks like nose, eye, mouth coordinates if any face present in provided image.

#### Sample:



#### **Output:**



# DELIVERABLES:

GITHUB URL : <a href="https://github.com/Harith009/SI-FINAL-PROJECT">https://github.com/Harith009/SI-FINAL-PROJECT</a>

Base swagger URL: <u>http://137.184.50.131:3000/docs</u>

All Endpoints:

Detection of Face : 137.184.50.131:3000/detectface

Detection of landmarks : 137.184.50.131:3000/detectfacelandmarks

Detection of emotions : 137.184.50.131:3000/detectfaceemotions

Compare similar photos : 137.184.50.131:3000/findsimilar

Detect uploaded file : 137.184.50.131:3000/detectUploadedFace