 This documentation is not published yet, this is only a preview. Changes to the publishing setting...

# Image Repository

Created an Image Repository API which uploads and deletes files from GCP Storage Bucket.

## AUTHORIZATION

Bearer Token

### Token

{{authToken}}

## POST Upload

http://localhost:9001/uploads

'/uploads' route uploads a single file to the GCP bucket. If a file with that name already exists within the users bucket, the user will receive an error saying the image already exists. If the image is uploaded successfully the user will receive a url for their image. JWT authentication is required to upload the image.

## AUTHORIZATION

Bearer Token

This request is using Bearer Token from collection Image Repository

## HEADERS

### Authorization

Bearer

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfaWQiOiI2MDk0NWE3MmViMGM3ZDUwNmM5ZGRlZjEiLCJpYXQiOiJlMjMjA0MTQxNDI9.0j4YgOumv88SA1cCIXBBsjvkCjOrDAct90ewYRB8iUc

BODY formdata

file

### Example Request

Upload

```
curl --location --request POST 'http://localhost:9001/uploads' \
--header 'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfawQioiI2MDk0NWE3MmViMGM3ZDUwNmM5ZGRlZ' \
--form 'file=@"/C:/Users/vikra/Downloads/vikram.jpg"'
```

**DEL** Delete Image

http://localhost:9001/delete/6096e01436e5de3f8ccb3112

The delete route will delete the user's image from the GCP bucket. If the image does not exist within the bucket, the user will receive an error. JWT authentication is required for the image to be deleted. When deleting an image, the image id has to be passed into the url.

## AUTHORIZATION

## Bearer Token

This request is using Bearer Token from collection Image Repository

BODY formdata

**file**

vikram.jpg

### Example Request

Delete Image

```
curl --location --request DELETE 'http://localhost:9001/delete/6096e01436e5de3f8ccb3112'
```

## POST Create User

http://localhost:9001/users

Documentation Settings

This will create a new user, and store it in the MongoDB database. When a user is created, a jwt token is created for security purposes. To create a user provide the email, password, and the users name.

BODY raw

```
{
  "name": "VikramS",
  "email" : "vsud@gmail.com",
  "password" : "pencilcase1300"
}
```

Example Request

Create User

```
curl --location --request POST 'http://localhost:9001/users' \
--data-raw '{
  "name": "VikramS",
  "email" : "vsud@gmail.com",
  "password" : "pencilcase1300"
}'
```

## POST Logout User


No request URL found. It will show up here once added.

The logout request will remove the token from this request for the current user. JWT Authentication is required for logging out

AUTHORIZATION

Bearer Token

This request is using Bearer Token from collection Image Repository

[Example Request](#)[Documentation Settings](#) [Logout User](#)

```
curl --location --request POST ''
```

## POST Login

```
http://localhost:9001/login
```

When the user logs in, an authentication token is provided to the user. The user has to provide their email and password which they used when initially creating a user.

BODY raw

```
{
  "email" : "vsud@gmail.com",
  "password" : "pencilcase1300"
}
```

[Example Request](#)[Login](#)

```
curl --location --request POST 'http://localhost:9001/login' \
--data-raw '{
  "email" : "vsud@gmail.com",
  "password" : "pencilcase1300"
}'
```