Documentation for node js project assigned by CashFlo

* Public End point (<http://localhost:3000/login>)

Code snippet for ‘GET’ request. It takes any username/password as request body ( req.body) and then returns the JWT token using ‘Bearer’ Authentication algorithm’

![A screenshot of a cell phone

Description automatically generated]()

Fig : 1

Here, ‘secretkey’ is a passphrase provided to tokenized the request.

Postman Screenschot :

Using mock username/password as value and send the ‘POST’ request to <http://localhost:3000/login>

![A screenshot of a cell phone

Description automatically generated]()

Fig : 2

![A screenshot of a social media post

Description automatically generated]()

Fig : 3

After requesting to <http://localhost:3000/login> it returns a jwt token which looks like above in picture fig: 3 .

* Private End Point <http://localhost:3000/login/image>

Code snippet for private end point.

![A screenshot of a cell phone

Description automatically generated]()

Fig : 4

Here, we have used ‘auth’ middleware to authorize the token provided by user :

Code snippet:

![A screenshot of a cell phone

Description automatically generated]()

Fig: 5

![A screenshot of a social media post

Description automatically generated]()

Fig: 6

After getting the token from step 1, we passes that token into “Header Section” of postman : ( in fig : 6)

And providing the “image url” using Key-Value pair in postman:

It will look like as

<http://localhost:3000/login/image?url=imageUrl>

As we insert the key-value, the nodejs server downloads that image and resize it to

50px \*50px and return the same to the response body (res.body) as shown in fig: 7 below

![A screenshot of a cell phone

Description automatically generated]()

Fig : 7

Passing the public image url can be done in two ways :

1. Passing into the req.body.url
2. Passing into req.query.url