**API-ENDPOINT Documentation**

1) /auth/signup/:

Method: POST

Parameters:

username: string

email: string

password1: string

password2: string

Return: None

Description:

1)Creates user with credentials supplied

2)password1 and password2 must match. Should have at least 8characters.

3)Password should not be trivial

4)username and email should not have been registered before

2)/auth/login/:

Method: POST

Parameters:

username: string

password: string

Return: REST Framework Token Object's key

Description:

1)log in the user

2)username should be the one with which you have registered

3)/auth/logout/:

Method: POST

Parameters: None

Return: None

Description: logout the user. Your token is no longer valid

4)/auth/password/change/:

Method: POST

Parameters:

new\_password1: string

new\_password2: string

Return: Returns the success/fail message.

Description:

1)your password will be changed to new\_password1 and you’ll be logged out (if conditions

below are satisfied)

2)password1 and password2 must match. Should have at least 8characters.

3)Password should not be trivial

4)username and email should not have been registered before

5)your old token is no longer valid

5)/auth/refresh-token/:

Method: POST

Parameters: Current REST Framework Token Object's key

Return: Returns a refreshed token (with new expiration) based on existing token

Description: return a new token with new expiration

6)/upload/{func\_code}/:

Method: POST

Parameters: FormData object of the form

{“file”, blob (i.e your file), filename (i.e the name of your file)}

Return: None

Headers: {header=HttpHeaders ({“Authorization”: ‘JWT’ + (your Current Token Object's key)})}

Description:

1)uploads your file to backend and processes it to find the plagiarism percentages between

the constituent files

2)you are expected to upload a zip file here.

3)all your files to be tested should be directly inside this zip. i.e they should not be present

inside a folder in the zip

4)

* **func\_code=5** then the file sent in request data will be saved as **stub file**
* **func\_code = 0 or 1 or 2 or 3** the file sent in request data is saved as **zip file** and will be processed as described below
* **func\_code=0** then files uploaded will be processed as if they are **text files** and **no stub code** given
* **func\_code=1** then files uploaded will be processed as if they are **C++ files** and **no stub code** given
* **func\_code=2** then files uploaded will be processed as if they are **text files** with **stub code** given
* **func\_code=3** then files uploaded will be processed as if they are **C++** with **stub code** given

5)**Caution:** the moment you upload a new zip file all your old results are lost.

7)/download/{filename}/:

Method: GET

Parameters: None

Return: downloads the processed Results of the input file you have given

(Note that the filename in the endpoint should be same as the one you have uploaded)

Headers: {header=HttpHeaders ({“Authorization”: ‘JWT’ + (your Current Token Object's key)}),

observe: ‘body’, responsetype: ’arraybuffer’ as ‘json’}

Description:

1)downloads the results on the zip that you have uploaded

2)downloaded folder consists of a csv file showing the plagiarism percentages between

The files arranged in descending order, Folder also has a image showing the plagiarism

Percentages between files shown as a matrix