



COM661 - Full Stack Strategies and Development

This is documentation for COM661 - Full Stack Strategies and Development Submission

GET ENDPOINT_ReadAllPosts

```
{{localhost}}/api/v1.0/posts?pn=1&ps=10
```

This endpoint will return all the posts in the MongoDB. Page Numbers can be provided using the "PN" and "PS" meaning Page Number and Page Size.

PARAMS

pn

1

ps

10

Example Request

ENDPOINT_ReadAllPosts

```
curl --location -g --request GET '{{localhost}}/api/v1.0/posts?pn=1&ps=10'
```

GET ENDPOINT_ReadOnePost

```
{{localhost}}/api/v1.0/posts/93
```

This end point will return one post given a post ID provided after the / : POST_UUID



Example Request

ENDPOINT_ReadOnePost

```
curl --location -g --request GET '{{localhost}}/api/v1.0/posts/93'
```

DEL ENDPOINT_DeleteOnePost

{{localhost}}/api/v1.0/posts/101

This end point will delete one post given a post ID provided after the / : POST_UUID

Example Request

ENDPOINT_DeleteOnePost

```
curl --location -g --request DELETE '{{localhost}}/api/v1.0/posts/101'
```

PUT ENDPOINT_UpdateOnePost

{{localhost}}/api/v1.0/posts/93

This end point will update one post given a post ID provided after the / : POST_UUID

BODY formdata

Post_Author

Gerry

Post_Title

wd

Post_Body

dwdwd

Post Upvotes

91
==

Documentation Settings ▼

Example Request

ENDPOINT_UpdateOnePost

```
curl --location -g --request PUT '{{localhost}}/api/v1.0/posts/93' \  
--form 'Post_Author="Gerry"' \  
--form 'Post_Title="wd"' \  
--form 'Post_Body="dwdwd"' \  
--form 'Post_Upvotes="91"'
```

POST ENDPOINT_CreateOnePost

{{localhost}}/api/v1.0/posts

This end point will create one post provided its passed with a given form data.

BODY formdata

Post_Author

Toby

Post_Title

Angular is shite craic

Post_Body

Its me

Example Request

ENDPOINT_CreateOnePost

```
curl --location -g --request POST '{{localhost}}/api/v1.0/posts' \  
--form 'Post_Author="Toby"' \  
--form 'Post_Title="Angular is shite craic"' \  
--form 'Post_Body="Its me"'
```

GET ENDPOINT_GetAllComments

Documentation Settings ▼

```
{{localhost}}/api/v1.0/posts/101/comments
```

This end point will retrieve all the comments on a given post provided its passed with a Post_UUID after the /: Post_UUID

Example Request

ENDPOINT_GetAllComments

```
curl --location -g --request GET '{{localhost}}/api/v1.0/posts/101/comments'
```

GET ENDPOINT_ReadOneComment

```
{{localhost}}/api/v1.0/posts/56/comments/4
```

This end point will retrieve one comment on a given post provided its passed with a Post_UUID and Comment_UUID after the Posts/: Post_UUID/comments/:Comment_UUID

Example Request

ENDPOINT_ReadOneComment

```
curl --location -g --request GET '{{localhost}}/api/v1.0/posts/56/comments/4'
```

POST ENDPOINT_CreateOneComment

```
{{localhost}}/api/v1.0/posts/103/comments
```

This endpoint will create one comment on a given post provided its passed with a Post_UUID after the /:Post_UUID

BODY formdata

Comment_Author

Alusbs

Documentation Settings ▼

Comment_Body

This is shite

Example Request

ENDPOINT_CreateOneComment

```
curl --location -g --request POST '{{localhost}}/api/v1.0/posts/103/comments' \
--form 'Comment_Author="Alusbs"' \
--form 'Comment_Body="This is shite"'
```

DEL ENDPOINT_DeleteOneComment

```
{{localhost}}/api/v1.0/posts/56/comments/2
```

This end point will retrieve delete one comment on a given post provided its passed with a Post_UUID and Comment_UUID after the Posts/: Post_UUID/comments/:Comment_UUID

Example Request

ENDPOINT_DeleteOneComment

```
curl --location -g --request DELETE '{{localhost}}/api/v1.0/posts/56/comments/2'
```

PUT ENDPOINT_UpdateOneComment

```
{{localhost}}/api/v1.0/posts/56/comments/4
```

This end point will retrieve update one comment on a given post provided its passed with a Post_UUID and Comment_UUID after the Posts/: Post_UUID/comments/:Comment_UUID and it's provided with the correct form data.

BODY formdata

Comment_Author Toby Killen

Documentation Settings ▼

Comment_Body

Example Request

ENDPOINT_UpdateOneComment

```
curl --location -g --request PUT '{{localhost}}/api/v1.0/posts/56/comments/4' \
--form 'Comment_Author="Toby Killen"' \
--form 'Comment_Body=""'
```

POST ENDPOINT_Register

`{{localhost}}/api/v1.0/register`

This end point allows users to create an account on the platform providing the user provides correct form data and it passes server side validation. Once registered the user will be able to use this and head to the login endpoint to then login.

BODY formdata

Username

Toby

First_Name

Toby

Last_Name

Killen

Email_1

itoby24@gmail.com

Email_2

itoby24@gmail.com

Password_1

Rubikscube24

Password_2

Rubikscube24



Documentation Settings ▼

Example Request

ENDPOINT_Register

```
curl --location -g --request POST '{{localhost}}/api/v1.0/register' \
--form 'Username="Toby"' \
--form 'First_Name="Toby"' \
--form 'Last_Name="Killen"' \
--form 'Email_1="itoby24@gmail.com"' \
--form 'Email_2="itoby24@gmail.com"' \
--form 'Password_1="Rubikscube24"' \
--form 'Password_2="Rubikscube24"'
```

POST ENDPOINT_Login

{{localhost}}/api/v1.0/login

This endpoint will allow the user to login using the two fields "Username" & "Password". If the user has successfully authenticated. They will be provided with a JSON Web Token which will then provide them with authentication protected endpoints.

AUTHORIZATION

Basic Auth

Username

<username>

Password


<password>

Example Request

ENDPOINT_Login

```
curl --location -g --request POST '{{localhost}}/api/v1.0/login'
```

GET ENDPOINT_Logout

 `{{localhost}}/api/v1.0/logout`Documentation Settings 

The JWT (JSON Web Token) is provided with a TTL (Time to live) which is a predefined time from which the token was created to the token to be rendered unusable. For users who wish to render their existing token unusable before the token expires, this end point will enable the user to add their token to a blacklist, rendering their token unusable forever. They would then have to reauthenticate with the server using the login endpoint.

HEADERS

x-access-token

`{{JSON_Web_Token}}`

Example Request

ENDPOINT_Logout

```
curl --location -g --request GET '{{localhost}}/api/v1.0/logout' \
--header 'x-access-token: {{JSON_Web_Token}}'
```

GET ENDPOINT_Upvote

`{{localhost}}/api/v1.0/post/upvote/105`

This endpoint will provide the user with the feature of upvoting a post, similarly to Reddit and it's upvoting system. This route will take in a Post_UUID. For example `post/upvote/:Post_UUID` and the Endpoint will increase the post upvote counter by one.

Example Request

ENDPOINT_Upvote