Project 2 CPSC 476-03

Project Name: Blog Post Application

Team:
1) Shekhar Palit
2) Rohan Bedarkar
3) Shruti Taware

Step 1: Running Nginx

```
cd /etc/nginx/sites-available
systemctl start nginx
```

Authentication to start nginx.service (Password:student)

Step 2: Running procfile

Change directory to the project folder

```
foreman start --formation user=3,article=3,comment=3,tag=3,feedgen=1
```

Step 3: Running test cases (tavern.yaml files)

Change directory to the project folder

```
py.test
```

Backend for a frontend

RSS Feeds

RSS allows people to receive updates to web-based content of interest. With RSS feed – we can subscribe and then receive new content automatically in our feed reader.

Web syndication is a form of syndication in which content is made available from one website to other sites. It is an effective way of adding greater depth and immediacy of information to the pages, making them more attractive to users. This generates new traffic for the providing site- making syndication an easy and relatively cheap, or even free, form of advertisement. It can be done by following major steps:

A summary feed listing the title, author, date, and link for the 10 most recent articles.
http://localhost:5400/rssfeed

A full feed containing the full text for each article, its tags as RSS categories, and a comment count.
http://localhost:5400/rssfullfeed

A comment feed for each article.

Micro-service: User - user.py

Functionality: Create new user
Url:http://localhost/user
Method:POST

Curl command example:curl -POST http://localhost/user -d '{"user_name":"Shekhar__", "hashed_password":"ab", "full_name":"Shekhar Palit", "email_id":"asjdhkjas@csu.fullerton.edu"}'
Input variables :

user_name :user name will be unique key
hashed_password :Provide password for user
full_name:Full name of user
email_id:Email of user

All the variables mentioned above are in lower case.

Functionality: Delete existing user
Url: http://localhost/user
Method: DELETE
Login by entering the user id and password.
Curl command example: curl -X DELETE http://localhost/article -d '{" user_name ":"Amey"," hashed_password ":"passbhor"}' –u
Input variables :

user_name: Provide user name
hashed_password: Password for authentication

All variables mentioned above are in lower case.

Functionality: Change existing user’s password

Url: http://localhost/user

Method: PATCH

Login by entering the user id and password.

Curl command example:

```
curl -X PATCH http://localhost/user -d '{"hashed_password":"xyz"}' -u Shekhar__:ab
```

Input variables :

hashed_password: Update current password

All variables mentioned above are in lower case.

Micro-service: Articles - article.py

Functionality: Post a new article
Url:http://localhost/article
Method: POST
Curl command example: curl -GET "http://localhost/article"
Input variables:
title: Provide title of the article
content: Provide the content of the article.

All the variables mentioned above are in lower case.

Functionality: Retrieve an individual article
Url: http://localhost/article
Method: GET
Curl command example:curl -GET "http:// localhost /article?limit=1"
Login by entering the user id and password.
Input variables:
article_id : Provide article id for retrieving article

All the variables mentioned above are in lower case.

Functionality: Edit an individual article. The last-modified timestamp should be updated.
Url: http://localhost/article
Method: PUT
Curl command example: curl http:// localhost /article -d '{"title" : "Edited Title","content" : "Edited content"}' –u Shekhar__:ab

Login by entering the user id and password. Input variables :
article_id: Provide article id to edit
content: Will replace existing content of given article id

All the variables mentioned above are in lower case.

Functionality: Delete a specific existing article
Url:http://localhost/article
Method: DELETE
Curl command example: curl -X DELETE http:// localhost /article -d '{"article_id":8}' -u Shekhar__:ab
Login by entering the user id and password.
Input variables :
article_id: provide the article id.

All the variables mentioned above are in lower case.

Functionality: Retrieve the entire contents (including article text) for the *n* most recent articles
Url: http://localhost/article
Method: GET
Curl command example: curl -GET "http:// localhost /article?limit=2"
Input variables:
limit : Retrieve all data of article.

All the variables mentioned above are in lower case.

Functionality: Retrieve metadata for the n most recent articles, including title, author, date, and URL
Url: http://localhost/article
Method: GET
Curl command example: curl -GET "http:// localhost /article?metadata=2"
Input variables:
metadata: Retrieve metadata of given number

All the variables mentioned above are in lower case.

Micro-service: Comments - comment.py

Functionality: Add a new comment.
Url: http://localhost/comment
Method: POST
You can post a comment by your login credentials, if you want your username to appear in the comment section.
Curl command example:
Input variables:
comment: comment Text.
article_id: ID of the article
variables mentioned above are in lower case.

Functionality: Delete a comment. (Authentication required for this operation).
Url: http://localhost/comment?comment_id=
Method: DELETE
Sign in by your login credentials
Curl command example:
Input variables:
comment_id: comment id to delete the comment.
variables mentioned above are in lower case.

Functionality: Get n number of comments of an article.
Url: http://localhost/comment?article_id=
Method: GET
Authentication not required for this operation
Curl command example:
Input variables:
article_id: article ID.
variables mentioned above are in lower case.

Functionality: Get n recent comments of an article.
Url: "http:// localhost/comment?artile_id=&number="
Method: GET
Authentication not required for this operation
Input variables:
article_id: article ID.
number: provide the number of comments to be retrieved.
variables mentioned above are in lower case.

Functionality: UPDATE comment.
Url: http://localhost/comment
Method: PUT
Authentication not required of updating comments posted by Anonymous Coward
Curl command example:
Input variables:

comment: New comment
commnet_id: ID of the specific comment

Micro-service: Tags - tags.py

Functionality: Add tags for a new URL
Url: http:// localhost /tags
Method: POST
You may login by entering the user id and password.
Curl command example :curl -POST http://localhost/tags -d '{"article_id":3,"tag_name":"#great1"}'
Input variables:
artid: provide the article id.
tag: provide tag to be added.
All the variables mentioned above are in lower case.

Functionality: Delete the link of tag and article.
Url: http:// localhost /tags
Method: DELETE
You may login by entering the user id and password.
Curl command example:curl -X DELETE http:// localhost /tags –d '{"article_id":1,"tag_name":"#fullerton"}'
Input variables:
article_id: Provide the article id.
tag_name: Provide tag name
All the variables mentioned above are in lower case.

Functionality: Retrieve the tags for an individual URL
Url: http:// localhost /tags
Method: GET
You may login by entering the user id and password.
Curl command example: curl -GET http://localhost/tags/7
Input variables:
article_id: Provide the article id.
All the variables mentioned above are in lower case.

Functionality: Retrieve a list of URLs with a given tag.
Url: http://localhost/tags
Method: GET
You may login by entering the user id and password.
Curl command example:curl -GET http://localhost/tags?tag=%23fullerton
Input variables:
tag: provide tag name

All the variables mentioned above are in lower case.