

Installation Instructions

Team Cautions Spoon

Web-based Social Networking Platform Installation Instructions:

To run the program on the local host server:

**We recommend using a Virtual Machine with Ubuntu (Linux)*

- 1) Clone the repository from git and navigate to the 'SEG-major-project' folder or unzip the submitted 'Cautions Spoon' folder and navigate to the 'source_code' directory.
- 2) To view the project on the local host you will need to create a .env file. To do this open a new file and copy the text below into the file, naming it **.env**:

```
# S3 upload settings
```

```
DEFAULT_FILE_STORAGE = storages.backends.s3boto3.S3Boto3Storage
```

```
AWS_S3_ACCESS_KEY_ID = AKIA4AMLNUUEYG65NFC7
```

```
AWS_S3_SECRET_ACCESS_KEY = 5G/73JWJF1FvO5LFbIS9jeOQhfBxyd9vEcXLTQaa
```

```
AWS_STORAGE_BUCKET_NAME = bookclubimages
```

```
os.environ.setdefault('S3_USE_SIGV4', 'True')
```

- 3) From within the folder/directory, you will need to run the following commands:

- a) **virtualenv venv**
- b) **source venv/bin/activate**
- c) **pip3 install -r requirements.txt**
- d) **python3 manage.py migrate**
- e) **python3 manage.py seed**
- f) **python3 manage.py runserver**

If using a Mac, first run:
pip install setuptools --upgrade

- 4) You will now need to open your browser and visit the following site: <http://localhost:8000/> and you should now be able to see the book club landing page.

Recommended Browsers: Firefox & Chrome

Avoid: Safari

[Please have JavaScript enabled]

- 5) To create a super user, you can enter the following commands:

A super user is used to view and manage the site, if the admin would like to use the site as a book club member, they would need to make their own 'normal' account.

- a) **python3 manage.py createsuperuser**
- b) Enter a username, email and password when prompted

Recommender System Evaluator Installation and Result Replication

Instructions:

**We recommend using a Mac if Virtual Machine with Ubuntu (Linux) is giving a 'killed' error*

Content Based Recommender System:

NOTE: The evaluator will use a much smaller test set consisting of 31,000 ratings and 500 books which can be found in the csv files named 'BX_Book_test_set.csv' and 'BX-Book-Ratings_test_set.csv'.

To Replicate the results found in the Recommender Systems report you will need to run the 'content_based_evaluator.py' file from within the 'content_based_recommend' folder.

1. Before doing this, you will need to enter the following commands from within the 'source_code' directory (or the 'SEG-major-project' folder if using the git repo). (This assumes that you have a fresh folder which you have not run any commands on. If you have already run the following commands, skip to step 2).
 - c) **virtualenv venv**
 - d) **source venv/bin/activate**
 - e) **pip3 install -r requirements.txt**
 - f) **python3 manage.py migrate**
 - g) **python3 manage.py seed**
2. You can now enter the following commands to run the evaluator
 - a. **cd clubs/content_based_recommender/**
 - b. **python3 content_based_evaluator.py**

```
(venv) ~ -VirtualBox:~/SEG-major-project$ cd clubs/content_based_recommender/
(venv) ~ -VirtualBox:~/SEG-major-project/clubs/content_based_recommender$ python3 content_based_evaluator.py
```

3. To evaluate the performance of the combined property, the summary and the algorithm using both you will need to comment out the following lines in the content_based_KNN file and run the 'content_based_evaluator.py' file each time. **python3 content_based_evaluator.py**

- To evaluate the algorithm using the combined property:

```
43 # self.similarities[thisRating, otherRating] = summarySimilarity * combinedSimilarity
44 # self.similarities[thisRating, otherRating] = summarySimilarity
45 self.similarities[thisRating, otherRating] = combinedSimilarity
```

- To evaluate the algorithm using the Summary:

```
43 # self.similarities[thisRating, otherRating] = summarySimilarity * combinedSimilarity
44 self.similarities[thisRating, otherRating] = summarySimilarity
45 # self.similarities[thisRating, otherRating] = combinedSimilarity
```

- To evaluate the algorithm using both the previously mentioned properties:

```
43 self.similarities[thisRating, otherRating] = summarySimilarity * combinedSimilarity
44 # self.similarities[thisRating, otherRating] = summarySimilarity
45 # self.similarities[thisRating, otherRating] = combinedSimilarity
```

Neighborhood Based Recommender System:

NOTE: The evaluator will use a much smaller dataset consisting of 30,000 ratings which can be found in the csv files named 'BX-Book-Ratings_formatted_evaluation.csv' and 'BX_Books_formatted_evaluation.csv'.

To Replicate the results found in the Recommender Systems report you will need to run the 'N_based_KNN_bakeoff.py' file from within the 'N_based_RecSys_evaluation' folder found in the 'clubs' of the 'source_code' directory (or the 'SEG-major-project' folder if using the git repo).

(In the Recommender Systems report, we are using 40,000 ratings, however it would give "Killed: 9" error sometimes. In order to show the stable output and results, we are using 30,000 instead, so the result will have a slight difference.)

1. Before doing this, you will need to enter the following commands from within the SEG-major-project' folder. (This assumes that you have a fresh folder which you have not run any commands on. If you have already run the following commands, skip to step 2).
 - a. **virtualenv venv**
 - b. **source venv/bin/activate**
 - c. **pip3 install -r requirements.txt**
 - d. **python3 manage.py migrate**
 - e. **python3 manage.py seed**
2. You can now enter the following commands to run the evaluator (It might take more than 10 minutes to get the results. As we are using a smaller book dataset, if the evaluator cannot find the title of book, it will print the isbn instead.)
 - a. **cd clubs/N_based_RecSys_evaluation**
 - b. **python3 N_based_KNN_bakeoff.py**

```
[(venv) ~/Documents/SEG/SEG-major-project (main) $ cd clubs/N_based_RecSys_evaluation  
[(venv) ~/Documents/SEG/SEG-major-project/clubs/N_based_RecSys_evaluation (main) $ python3 N_based_KNN_bakeoff.py
```