

Oluwademilade Edward Akapo
101095403

Instruction on how to run Openstacks

Username : student

Password : t@milore

Instance Name : OluwademiladeAkapo

Public IP : 134.117.128.56

To start the Movie-server.js , go to the '/home/student/project' directory then all the files should be there. Type 'node .\Movie-server.js' to run server.

Project Report

Things I Implemented;

User Accounts

- users can create new accounts by specifying a username and password. The usernames must be unique.
- A user can log in and log out repeatedly without fail.
- only one user can be logged in at one time
- newly created users are considered regular. Then by using a button in the userprofile page they can upgrade to contributing user.
- When a user changes to a contributing account they can then create movies and people from their user profile by clicking on buttons.
- Everything the user has created will still be there after they change back to normal
- A user can view the reviews they have made as well as people and users they follow and a list of recommended movies. They are able to view any of these things by clicking on them on the webpage.

- The recommended movies list updates based on who you're following. If a user your following has made a review on a movie, the movie will be in your recommended list. Moreover, if a person you are following has worked in a movie, then that movie will also be added to your list of recommended movies. (*has little bug see below*)
- In the headers there is a movies link that sends the user to a page where they can search for movies by genre, name, title and keyword.

Viewing Movies

- Users are able to see a clickable image of the movie poster when they are viewing a movies data page.(*extension*)
- Users are able to see all basic movie information such as, Title,Initial Release year, Genres, Actors ,Director,Writers,Average Rating, Runtime,Short and Long reviews and Plot. Furthermore, users can see a list of similar movies
- The Director, Actors, Writers in the movie page all have links that will take you to the page of the person when clicked.
- The Genres all have links on them so that when clicked the user will be able to see a list of movies with similar genre
- The user will be able to see all short and long reviews made on that movie by all users and will be able to view the users page by clicking on the link in the name.
- The user will be able to add a short review by giving a score out of 10 and the user will be able to add a long review by giving summary, review and score.

Viewing People

- When viewing people, the user is able to see all the movies the person has worked in, with clickable links in all of them that send the user to that specific movie page.
- User will be able to see a list of people that person has worked, with clickable links to take the user to that specific persons page
- The user will be able to see a follow/unfollow button when clicked will either follow or unfollow the person. (*because i did not use databases i could not implement the notification functionality)

Viewing other Users

- When viewing other users, the user can see a list of all short and long reviews made by that user and can click on the movie title to go to the movies data page
- User will be able to see a list of all people the other user is following and is able to navigate to the persons page by clicking the link in the person's name
- Users will be able to add another user as a friend by clicking the follow/unfollow button in another user's page. (*because i did not use databases i could not implement the notification functionality)

Contributing Users

- A contributing user can add a new person or movie to the movie website data by clicking links in their user page, they will be redirected to either an add person page or an add movie page and will be asked to add information. (*my implementation was supposed to be that the user can add any person or genre they want, then that person will then be automatically added to the movie data. But adding person to movie data from movie creation gives some bugs)
- When viewing a movie as a contributing user you can add a person to either the actors or writers lists.

Bugs and things that i couldn't implement in time;

- I didn't implement notifications because i didn't implement databases
- When logged in and trying to add only ' John Lassenter' as a friend it takes you to an error page that i set up that tells the user they have to be logged in to do that specific action. But this only happens with john , i couldnt figure it out in time. Moreover, i think that the same bug affects my recommended movies. If you follow users and people their work and review add into your recommendation, now 9/10 times this works fine even when adding people who worked in the same movies , or users who have reviewed the same movies. But sometimes the recommended list will start holding duplicates of the same movie after following people or users with movies or reviews in common.

- When creating a person, creating a movie, adding a person to the movie i didn't implement a worked with function to update the people involved in the specified processes. Therefore, people worked with list will remain empty if not hardcoded into movie data

Design decisions

- Most of My routes call on redirects to other routes, this design can help with future additions of other routes to my server as it expands. For example my movies search route redirect to my parseMovieQuery route. I feel like the reusability of the routes and code is useful in reducing errors.
- Most of my pug pages restrict what the user is allowed to enter so as to not let possible bad inputs into the server that could cause some kind of error. Although i still put a lot of checks in my server to avoid it crashing i wanted to reduce that risk even more. for example some pages require you to enter data whilst other require you to enter only numbers at certain inputs. When thinking about handling a lot of users reducing the risk of bad data by any means is necessary
- My website can handle alot of bad inputs from users for example trying to do some user specific functionality when not logged in sends you to a page that tells you to either log in or sign up to complete action. I feel like this is useful in many ways, firstly it allows anyone to view the people, users and movies on the website but you will be redirected to sign up or log in if you want to do more than that. This is similar to what most websites like Instagram or IMBD do, because it's not every user that might want to log in to do basic things on the site.

- Only 1 type of review is allowed to exist on a specific movie, if the user adds to the site again it updates the review. This is to make the avg rating more valid.

Improvements to server

- Firstly to implement the work with update functionality and fix the bugs with 'John Lassenter'. Or I could just delete him entirely.
- I would like to improve the search functionality for all my listing and searching pages, most of the work on for each loops or nested for loops which have an $O(n)$ or $O(n^2)$ expected time, if we think about scalability and user experience on the website then the more users that join the more time the search function will take to get movies, people and users. This could also slow down many other functions in my server as well. I would like to improve the searching functionality and cut the expected time down to $O(n \log(n))$ so that I can reduce the time it would take for my server to respond to a user.
- Add little things to improve user experience, like instead of showing you an error bad input page, the form will tell the user that the input is bad, like when trying to create a new user with a non unique username.
- For my website I'd like to improve the follow button so that it shows unfollow if a user is already following. And I'd like to improve add review so that it shows update/ change your review if the user has already made one.
- One obvious improvement is the pug pages. The css could be much better.

Best features

- I feel like the user experience is decent. I allow non users to see the website and see all the functionality of it, like add review, add person etc.. before they decide to become a user.
- I like the background color but it looks a little bit too yellow. Nevertheless it is easy on the eyes.
- Honestly seeing as I only slept for 12 hours total in the last 4days to make this deadline, I am pretty proud of everything I've done. Also this 4 day journey has greatly increased my interest in web designing.