

**Project Author:** Nirmith Victor D'Almeida

**ID Number:** 101160124

**Site functionality (working):**

1. **Login/ registration** -> they work like a dime although you wouldn't be able to login once you shut down the server and restart it but you can look that user up
2. **Home /search Page** -> You will get a Recommended movie list whose algo will be explained at a later on stage and a list of all the movies stored in the database below it you will be able to search just by exact movie name since I had no time to implement more dynamic ability of the search function.
3. **Movie Info Page** -> You will get the information of the movie and will be offered a chance to leave a review on it but due to lack of knowledge of mongoose I didn't implement the stay functionality of the site such as append the review to each movie  
-> you can add this movie to your watch list and as well as leave a review but sadly no rating since this functionality caused a lot of crashes with the main mongodb  
-> you will get a list of similar movies which algo will be explained at a later on stage
4. **The Writer/ Director/ Actors info page** -> these users have a clickable link which when clicked will take you to their respective page which will show you their movies that they worked on and their 4-ish collaborators in one of their many movies you can add these "creators" to your subscribed list as well whether you are a contributing user or a normal user  
-> you will be offered a chance as well to search up for other actors (case sensitive as well)
5. **Profile page** -> in this page I display the user's name password and account level and a upgrade button any user that clicks on it will be able to upgrade from a regular to a contributing user  
-> separate section to view and store the movie reviews of this particular user when you log out and log back in you will still be able to view this  
-> I then have a section for Watch later list say that a user has taken a liking to a movie and then saves it in his store later he can do so and logout and login and it will be there  
-> Then I have a section for Following which is for content creators and other users ( just viewed for users though)  
-> then I have a contributing history which doesn't work because I haven't been able to store the data to a database successfully without crashing the entire program  
-> then there a section to search for other users in the site say they are already existent users in the program you can search them and view their history for reviews, following history (creators and users viewed), and their watch history there is a follow button as well which when clicked will also update the user with a new follower
6. **Upgrade button on profile page** -> You can click on this button and update the user to a contributing user and then the contributing history will be made available to the user but atm it just says that it is a contributing user which is an easy fix but sadly due to lack of time and other exams I had to reduce the amount of its functionality to updating the user type and showing the buttons for adding a movie which loads up another page

7. **Adding movies** -> sadly this will just be stored locally and I do not know how to make the function work with storing it to the database without creating a lot of errors in both the server and database side

#### **Site functionality (not working):-**

1. **Home / search page** -> in this page I am not able to make it dynamic since it was causing a lot of issues although I have a way of working round the issue I have a few exams starting soon and also another project which was a do or die scenario and also time difference made it rather hard to attend any love workshops and any TA office hour but my -> solution was to create a separate function that loaded the page with the array of movies that matched the given said description
2. **Use of Mongoose** -> I have not really understood how any of this worked so I really just left this since I didn't want to worry about a small functionality
3. **Writer / actors / Creator and User's page** -> I didn't understand how unsubscribe properly worked at the time and left it for last and the fix is rather simple check if the user is subbed in the array if he is un sub
4. **Adding movie** -> adding this to the user's database rather than storing it locally thereby making the search function a bit useless in the regard of searching for it you can view it being updated in the console.log() sadly fix is rather simple just a insertOne would have worked but sadle no time to implement this
5. **Adding a person** -> Never really understood what this did so I left it out

#### **Additional features:**

1. Since prof did say he wouldn't grade the session data being stored in db I somewhat managed to store whatever the user did if the user is in the db that is so I would say the update function within the DB was my add on for adding in more data and updating their user's list

#### **Critique of design:**

For starters, even though I did apply for an extension I will not be able to use it since I completely forgot my exams started on the 18<sup>th</sup>

But enough of excuses... onto the critique

##### **1. What I am proud of:**

- a. The Home page and my recommendation and similarity algorithm although simple was one of the most challenging aspects that I faced till now in this course so I am happy with what I came up with
- b. I am proud of the way I fixed up my profile page to make it much more segregated and easier to differentiate it between each section of the user
- c. The search user Kinda was inspired by discord's case sensitive lookup search
- d. Similar to the creator's lookup and the subscribe function that worked in the end was a major success to me

##### **2. What I could improve:**

- a. If I had not procrastinated and only focused on tutorials and other courses I would have finished this entire project by atleast the beginning of the month instead of rushing all in the last few days
- b. The search page being more dynamic

- c. A bit more advanced recommendation and similar movie algo would have been a nice addition to have
- d. Maybe a bit more use of Mongoose would have been helpful to have since that would make my code much more cleaner than the way it is at the moment
- e. Ability to unfollow and unsubscribe would have also been a nice addition to my program
- f. Understand what notifications mean in this context

#### Recommendation Algorithm And Similar Movie Algorithm

-> Both of them are based on the same principle of creating a recommended movie based on the similarities of the either a User's search history for instance the user's last searched up movie it looks to see if they have any common genre

-> The Recommendation although sharing a similarity in algorithm is just based on random 4 number generator that way I can recommend random 4 movies to the user when he/she visits the search movie page (would have been better to have it in the home page but it was very confusing with the additional containers and I didn't want the user's profile page to be sooo big.

Overall: I would have done better if all my courses had projects instead of final exams XD