360 Project

Preliminary Server-Side Summary

Our server-side implementation utilizes PHP and JavaScript. PHP script is used for most of the files in order to successfully connect to the database and work with the database tables. With most of our files, we first check whether the user is logged in or not and then display a navigation and header bar that corresponds to the state of the user. If the user is logged in, then a “logout” button appears at the top, and they also have a “profile” navigation tab visible to them. If the user is not logged in, then a “login” and “register” button is visible at the top of the page, and they are not able to see a “profile” navigation tab. PHP is used in most of our files because we put every movie that a blog post can be written about in our database, along with additional tables for users, likes, comments, saves, watchlists, advertisements, and reviews. Nearly everything that the users interact with on the site comes from our database.

When a user registers for an account, the image that they upload is inserted into a table in the database as a BLOB. When the user goes to their profile, they can see their profile picture on the right-hand side. The registration form is validated using JavaScript which goes through each form entry box and makes sure that it is not empty. If one of the required boxes are left empty, the form submission does not go through, and an alert pops up that lets the user know they need to fill out every box. If every box is filled out, the validation script makes sure that the password matches the “repeat password” entry box. If they do not match, the form submission does not go through and the user has to re-enter their passwords. If a user already exists in the database with the entered username and email, then an error message pops up telling the user to return to user entry with a link back to the register page. If no user already exists in the database, then the user’s account is created, and they are redirected to the login page.

Similar to the registration form, the login form and corresponding PHP code make sure that there is a good connection to the database before ensuring that the form’s request method is “post.” A prepared SQL statement is used to check if the username and password exist in the database in order to prevent against SQL injection and ensure security for the user’s information. The passwords are encrypted using the md5() method which also protects the user’s information. If the entered username and password are correct and match the ones in the database, then the user is redirected to the home page of the website. If the username is correct but the password is not, or if the username is incorrect, then an error message pops up telling the user to try again and links them back to the login form. If a user forgets their password, password recovery through email has been implemented.

If the user is logged in, they are able to like, comment, add to watchlist, and save different blog posts that they see. The number of likes and saves on the post will increase as users press each button and comment threads are visible to the users. When a user clicks on a movie, they are able to write a review if they are logged in. If they are not logged in, then instead of having a “submit” button for the review entry, they see a “login” button with a message telling them to login in order to leave a review. Each movie’s page information is pulled from the database by specifying the title of the movie that the user clicked on. Through SQL statements, all the reviews, ratings, images, and facts about the movie are displayed on its page.

Admins are given an admin code to enter if they register for an account. They are able to search for users and disable them, as well as edit and remove posts and comments. The number of users that an admin disables is visible in the database which allows everything to be kept track of.