

UNIVERSITY OF THE WEST INDIES, CAVE HILL CAMPUS

Department of Computer Science, Mathematics & Physics

COMP2245 – Web Development Concepts, Tools and Practices

Assignment 2

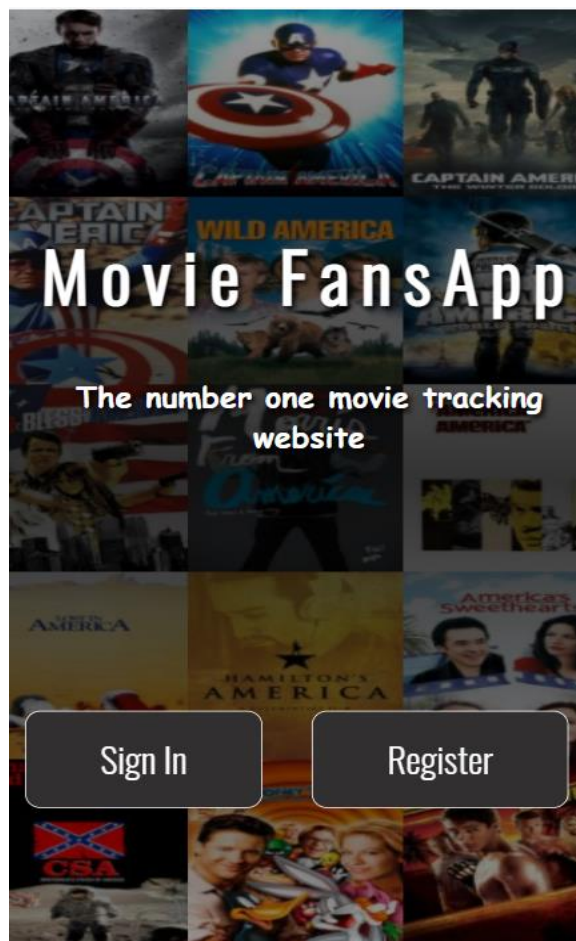
Due Date: **December 15th 2021 - Midnight**

This assignment is worth 20% of your coursework

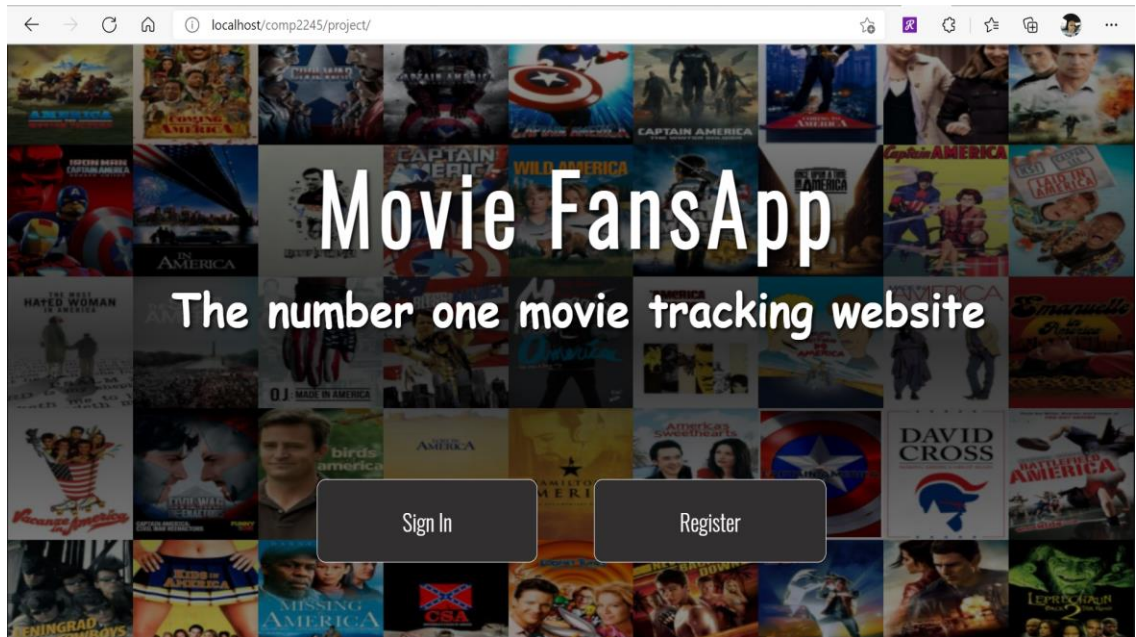
1. The first page that should be displayed in this project will be a page that allows the user to **sign in** or to **register** as a user of the site. Below is an image of this page. If one tries to go directly to any other page via the address bar without signing in this first page will be displayed.

[15 Marks]

Mobile:



Desktop:



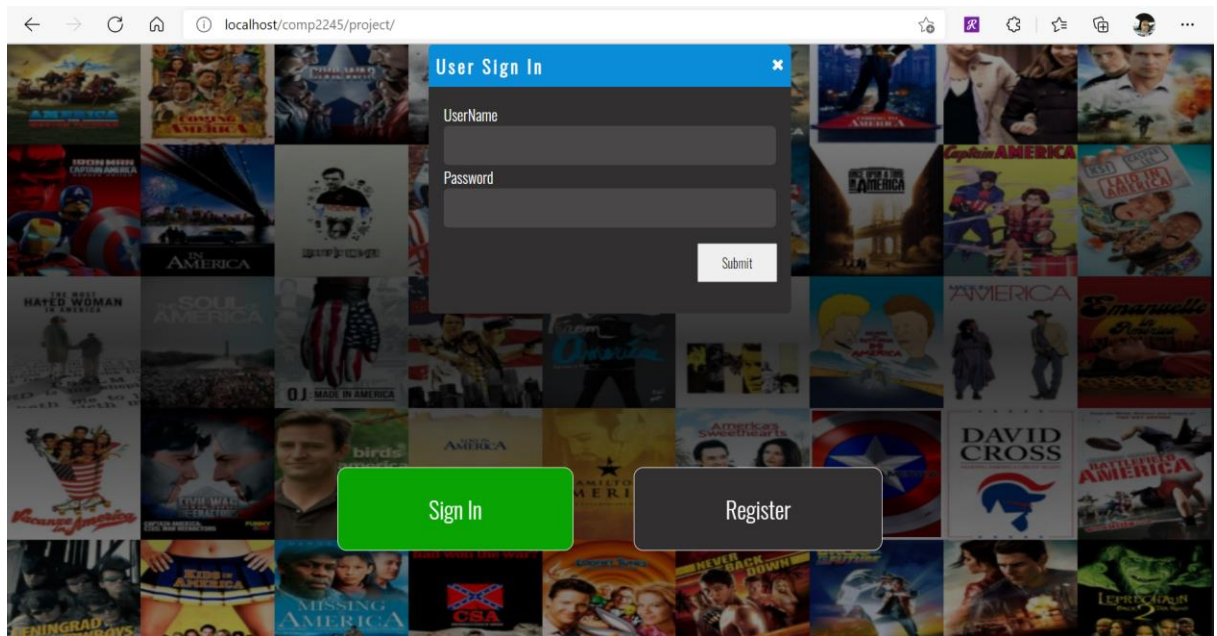
Hints:

- The image on this page is the **landing.jpg** supplied with this assignment
- Use the following code in the css file to get the image added to its containing div:

```
background: linear-gradient(#0006, #000c, #0006),  
url("../img/landing.jpg");
```

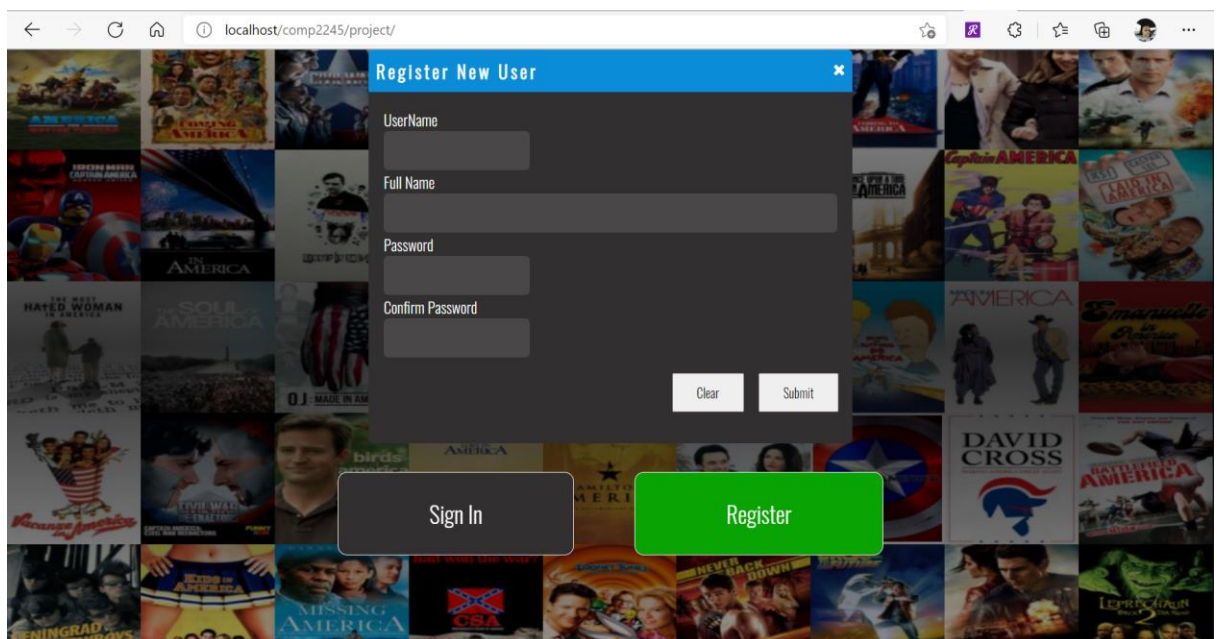
- The **height** of the <div> containing the image is **100vh**
 - The two buttons are position : fixed at 15vh from the bottom
2. When the **Sign In** button is clicked a popup sign in window should be displayed, that allows the user to sign in, if the **UserName** and **Password** exist in the **users** table, the main page of the website will be displayed. Below shows an image of the sign in popup window in desktop view.

[20 Marks]

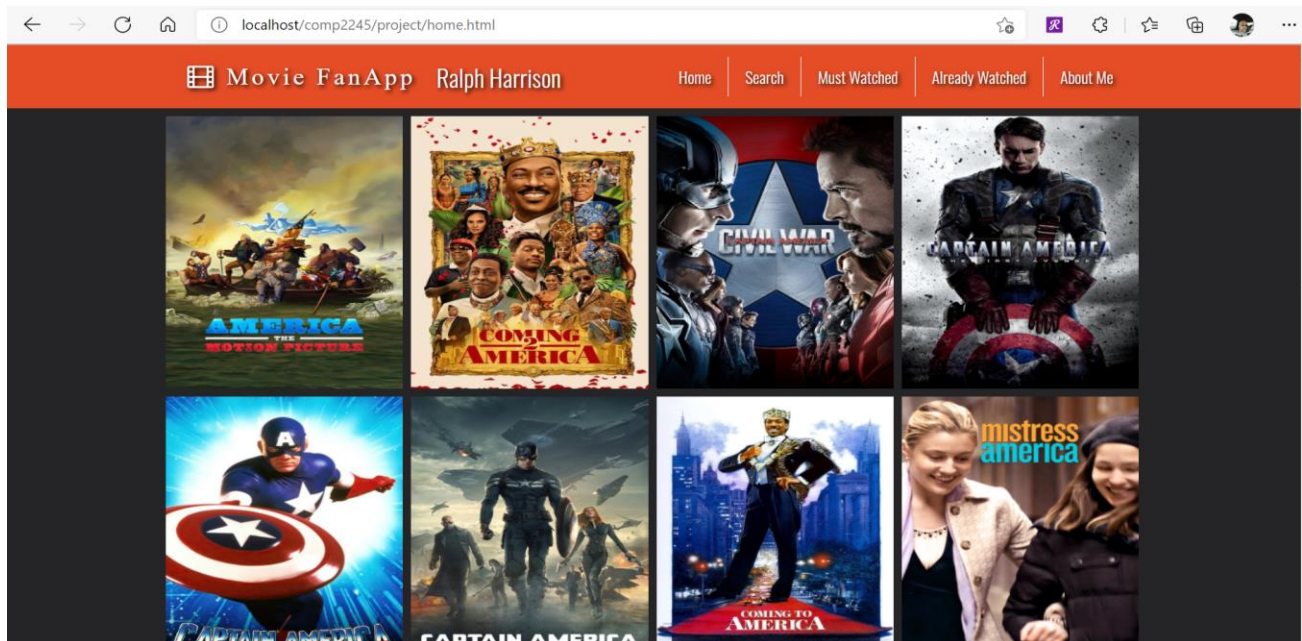


- When the **Register** button is clicked a popup register window should be displayed, that allows the user to register. If the **UserName** has not already been used and the Password adheres to the criteria specified in the lab session video on **validation**, then the data should be entered into the **users** table, and the new user should be able to sign in. Below shows an image of the register popup window in desktop view.

[20 Marks]



4. In assignment 1 there was a **<main>** that has a number of **** tags, those hard-coded image tags are to be removed, and the **<main>** should be populated with the movie posters from the given **movies** table. There should only be twenty posters displayed per page, and a pagination facility implemented to allow the user to navigate to other pages. On examining the movies table you should notice a **poster_path** field that stores the jpg portion of the URL to the movie poster, you will need to append that portion to the following link, to have a complete image URL for your **** tags **src** attribute: **<https://image.tmdb.org/t/p/w500>** Below shows an image of the main page of this website in desktop view. **[30 Marks]**

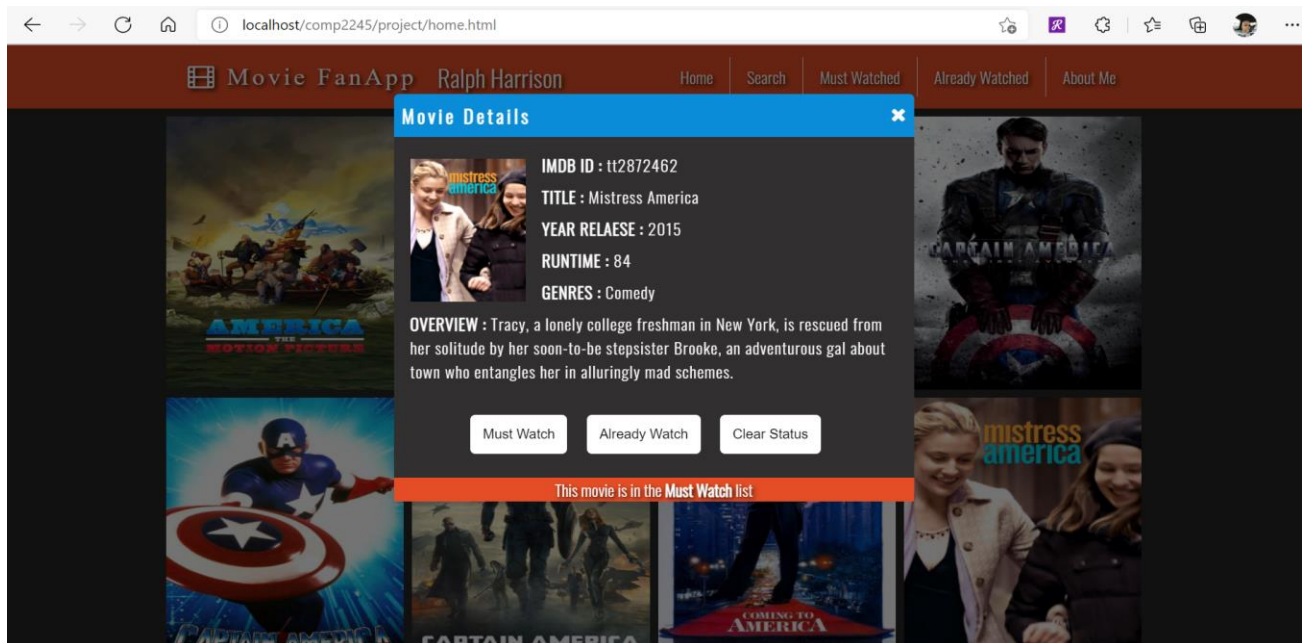


Hints:

- You can use the following css code to get the images in the above format in its containing div without the need for media queries:

```
display: grid;
grid-template-columns: repeat(auto-fill, minmax(200px, 1fr));
gap: 0.5rem;
```
- The fetched data from the **movies** table should be placed into an array in JavaScript.
- The height of the images should be set to 275px.

- The user that is currently logged in, full name should be displayed in the page header for this screen.
5. For all the images added to the <main> tag they should be a click event added so that a popup with the movie details should be shown. Below is an example of such a popup in desktop view. **[20 Marks]**



6. Notice the Details popup window from above, there are three buttons, these are used to update the **user_status** table. There is a status field in the table that is to be updated with "MW", "AW" and "", when the **Must Watch**, **Already Watch** and **Clear Status** buttons are pressed. In addition to this field, the user number, movie number, poster path, movie title and movie overview are all to be updated when the buttons are clicked.

A message should be shown at the bottom of the popup window letting the user know the status of the movie. Below shows two examples of the messages that should be displayed.

[20 Marks]

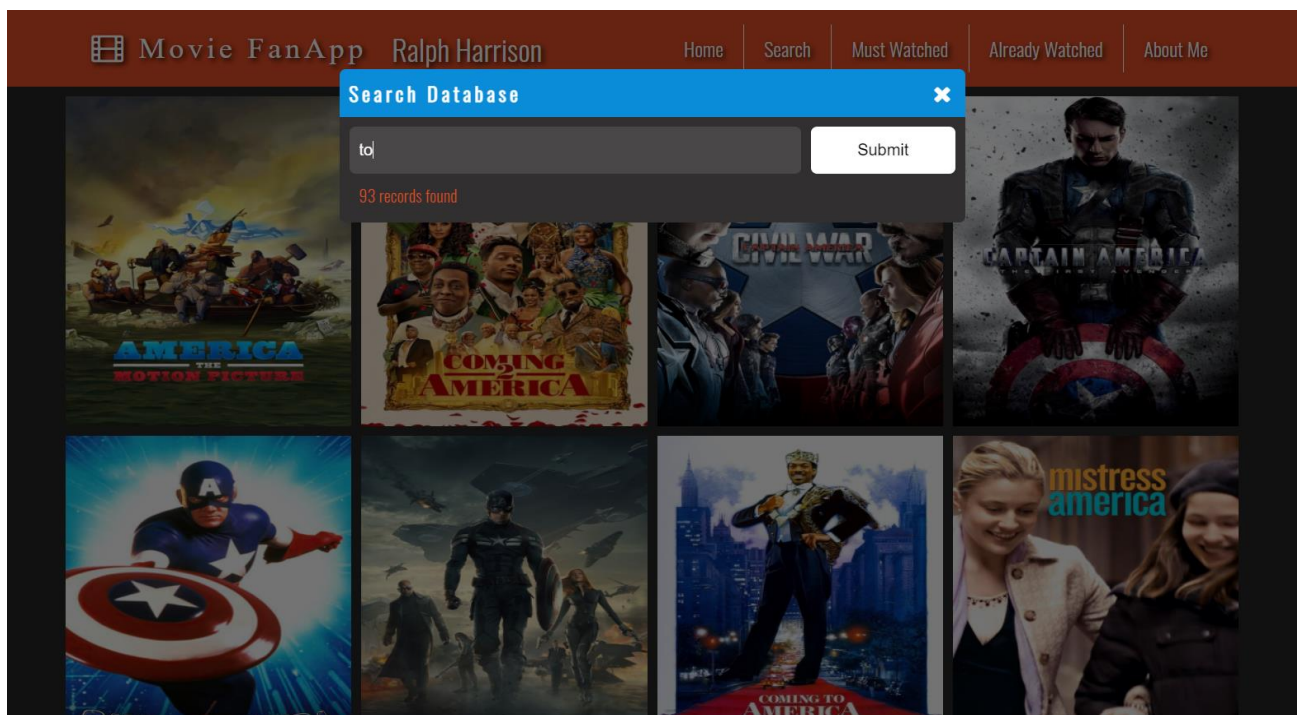
This movie is in the **Already Watch** list

Must Watch list is updated

- There should be a search facility that allows the user to search for a movie title in the **movie** table. As the user is typing in their search criteria there should be an indicator letting them know the number of records found.

On clicking the submit button the images that are currently in the <main> tag should be removed and the images for the results of the search should be displayed. Below shows what the search popup looks like.

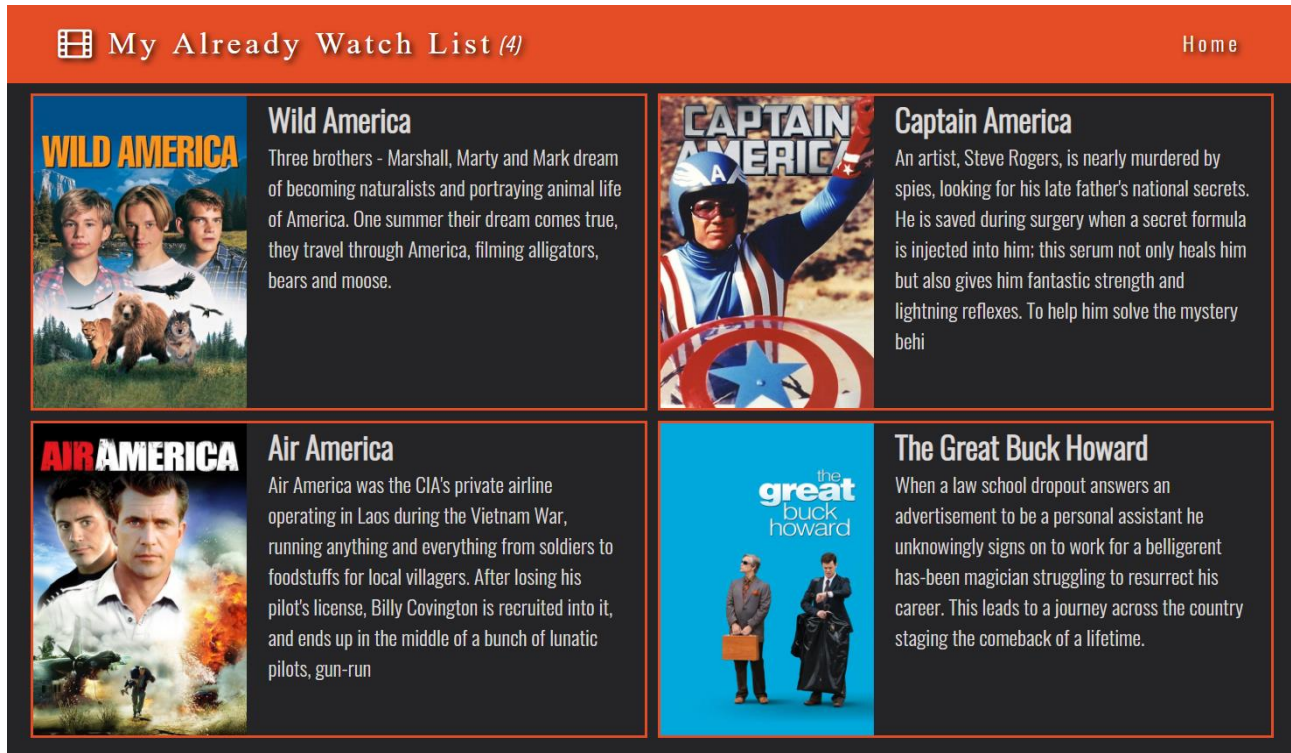
[30 Marks]



- When the **Must Watched** or the **Already Watched** menu items are clicked, they should load an associated page which displays the movie poster which have “MW” or “AW” in the status field of the records, these should be displayed six per page. This data will come from the **user_status** table, and

the displayed will be based on the currently logged in user. Below shows an image of what these pages should look like.

[20 Marks]



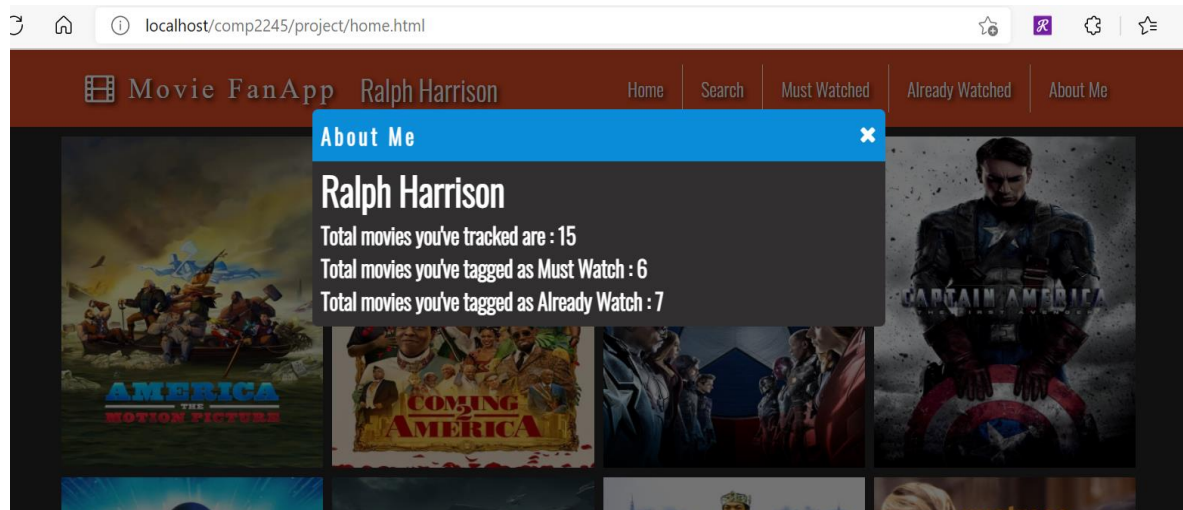
9. Each page displayed should have pagination at the bottom of the page; this is to limit the amount of data that is displayed per page. Below shows what the pagination should look like.



[10 Marks]

10. When the About Me menu item is clicked the **user_status** table will be queried to obtain, the total records with the current user number stored, the total number of movies with a status of AW and MW for the current user. Below shows what the About Me popup should look like.

[15 Marks]



Database Details:

The following information is about the database that will be provided for this project, this information is just for completeness, there should be no need for you to make any changes to the structure of the database or tables, and the following is just for information purposes:

Database Name:

comp2245 - This is the database that holds the three (3) tables that this project is working with.

Table Names:

movies – stores that details of the movies in this project.

users – stores the fullname, username and password of the registered users of this site.

user_status – stores the user number, movie number, poster path, movie title, movie overview and finally the status of the user which are AW for already watched and MW for must watch.

THE END