React Movie List Project

DEMO: <https://wizardly-carson-f4d936.netlify.app/>

Requirements:

**Header:**

1. There should be a header showing icon and "HOME", "Favorite" and “Rated”
2. There should be a login button in the header
3. Clicking home will navigate to root url. Clicking “Favorite” to “/favorite”, Clicking “Rated” to “/rated”
4. Clicking “login” to “/login”
5. If user has login, we should show user name
6. Clicking the username should allow people to logout.

**Login Page**

1. “/login” url should show login page
2. Page should show a “Login” title
3. Page should show a two input box for “Username” and “Password”
4. Page should show a “submit” button
5. Clicking “submit” button should show a loading icon while logging in
6. If success, it should navigate to home page
7. If failed, it should show error message
8. Page should show error message if no username or password is typed.

**Home page**

1. Page should load the first page of “now playing” by default
2. Page should show movie in a Grid format with 4 movies cards in a row
3. it should have a pagination controller allowing user to navigate between pages for current category
4. It should have a category dropdown selector with “Now playing”, “Top rated”, “Popular” and “Upcoming” options.
5. Select category should load the first page of the selected category
6. App should cache the data that is already viewed. It means the app shouldn’t make api call for a viewed category page. Eg, if I view the second page of the now playing category from the first page, and then I turn back to the first page, the app shouldn’t call api for the first page data again because it is already cached in client side.

**Movie Card**

1. it should show movie poster on the top, title below the poster and average rating on the button left and heart icon on the button right.
2. If the movie is user favorite movie, the heart icon should be filled with red, otherwise empty.
3. Clicking on the title will navigate to the movie details page “/movies/:movieId”
4. It should show user’s own rating as wellin “Rate” page
5. If user is login, clicking on the heart should toggle if user likes the movie
6. If user is not login, clicking on the heart shouldn’t do anything.

**Favorite & Rated Page:**

1. It should load user’s favorites and rated movies for “Favorite” and “Rated” page.
2. The movie should be displayed in the Grid style
3. The page shouldn’t show pagination controller
4. If user is not login, both pages shouldn’t not show anything.

**Movie details Page:**

1. Should show details for the movie with id in the url ( demo as reference).
2. The page should show user’s rate if user rated the movie, otherwise should “not yet”
3. The page should have a rate selector from 1- 10 and a “Rate it” button.
4. Clicking on the “Rate it” button should rate the movie with the score user selected.

**API usage:**

1. Load movie list for categories:
2. Now playing movies: <https://developers.themoviedb.org/3/movies/get-now-playing>
3. Popular movies: <https://developers.themoviedb.org/3/movies/get-popular-movies>
4. Top rated movies: <https://developers.themoviedb.org/3/movies/get-top-rated-movies>
5. Up coming movies: <https://developers.themoviedb.org/3/movies/get-upcoming>
6. Load movie details:

<https://developers.themoviedb.org/3/movies/get-movie-details>

1. User login:

It is a little bit completed.

Reference documentation:

<https://developers.themoviedb.org/3/authentication/how-do-i-generate-a-session-id>

You will need to

1. Call <https://developers.themoviedb.org/3/authentication/create-request-token> to get the “request\_token”
2. Then call https://developers.themoviedb.org/3/authentication/validate-request-token with username, password and the “request\_token” you just got, in order to validate the request\_token
3. Then call the “<https://developers.themoviedb.org/3/authentication/create-session>” with the “request\_token”, then get the “session\_id” from the response.
4. Then call the <https://developers.themoviedb.org/3/account/get-account-details> to get the user account details and get the account ID (will be used for favorite movie, get related movies and favorited movies)

Sample code:



1. Rate movie:

POST call to <https://developers.themoviedb.org/3/movies/rate-movie>

1. Mark movie as Favorite:

POST call to <https://developers.themoviedb.org/3/account/mark-as-favorite>

1. Get user Favorite movies:

<https://developers.themoviedb.org/3/account/get-favorite-movies>

1. GET user Rated movies:

<https://developers.themoviedb.org/3/account/get-rated-movies>

Library used for DEMO project:

1. **UI component Library**

@material-ui/core, @material-ui/icons, @material-ui/lab

1. **HTTP client**

axios

1. **Font**

fontsource-roboto

1. **form handing**

formik & yup

1. **state management**

Redux

1. **Routing**

react-router-dom

1. **Utils functions**

lodash