Assignment Category: Orchid

MOVIE PORTAL

Project Theme

You must create a user-friendly Movie Portal to simplify the process of exploring movies, viewing movie details, adding movies, and deleting movies while managing user favorites. This project must include a dynamic user interface, robust functionality, and seamless user experience.

Key Rules:

- **GitHub Commits:**
  - Include a minimum of 15 notable GitHub commits on the client side.
  - Include a minimum of 8 notable GitHub commits on the server side
- **Readme.md:** Add a meaningful readme.md file with the name of your website and a live site URL. Include a minimum of five bullet points to feature your website.
- **Responsiveness:** Make it responsive for all devices, including mobile, tablet, and desktop views.
- **Environment Variables:** Use the Environment variable to hide the Firebase config keys and Mongodb credentials.
- **Lorem Text:** Don't use any Lorem ipsum text; you can not use the default alert to show any error or success message.
- **Unique Design:** First, decide what kind of website you want to make. Then, search online or check out websites like ThemeForest to get ideas for the design. But remember, your website idea shouldn't be similar to any projects you've done before or to any examples in our modules or conceptual sessions.

★ You can also look for free resources on blogs to help with your website.

- **Host your Application:** You can choose deployment systems like Netlify, Surge, and Firebase for client-side hosting and vercel for server-side hosting. As you develop a single-page application

- Ensure that the page doesn't throw any error on reloading from any routes.
- Add your domain for authorization to Firebase if you use Netlify / surge
- Logged in User must not redirect to Login on reloading any private route

---

Main Requirements:

data format:

- Movie Poster (string url).
- Movie Title (string)
- Genre (array)
- Duration (number)
- Release Year (number)
- Rating (number)
- Summary (string)

1. Layout Structure

Ensure that your design is eye-catching. Create a roadmap of your website to ensure a better development. Look for ideas for the design on the internet or by visiting sites such as ThemeForest or Dribble. But keep in mind that your idea for a website should look different from any past projects of yours or any of the samples from our modules or conceptual sessions. Take inspiration from other websites. You can mix several things to your own website.

🎯You can also look for free resources on blogs to help with your website. https://bootcamp.uxdesign.cc/free-images-and-resources-collection-for-website-c77f2fc46ce5

2. **Navbar:** The navbar will contain the

- Website name/logo,
- Home,
- All Movies,
- Add Movie **(It will be a private route)**,
- My Favorites **(It will be a private route)**,
- Add One extra route **(It will be a Public route,Show anything relevant in this route)**,
- “Login” and “Register”.

a. The “Login” and “Register” buttons is conditional, if the user is not logged in it will show the “Login” and “Register” Buttons. If a user clicks on “Login” it will redirect to the login page. If a user clicks on “Register” it will redirect to the Register page.

b. But If the user is logged in here you will show the user photoURL, when you hover over the photoURL it will show the displayName. And it will show the “Logout” button.

3. **Authentication**: Users will show a Login page with a form, so that the user can Log in this application.
   - Show a Title for Login. & Form with the following fields (Email, Password, Forget Password, Login button)

If the user logs in successfully then

- navigate him to his desired Route / Home page.
- If not, show him an error with a toast/error message anywhere in the form.

There will be some other options like

- Show the user a Link for Register so that he can go to the register page.
- Show users a Social Login Button (Google only) . on Clicking it
  - user authenticates with Google
  - Navigate him to his desired Route / Home page.

User Registration

Create a register page with a form, so that the user can register himself in this application.

- Show a Title for registration and a Form with the following fields (Name, Email, Photo-URL, Password & Register Button)

If the user registers successfully then

- navigate him to his desired Route / Home page.
- If not, show him an error with a toast/error message anywhere in the form.

Implement password validation

For password validation, you need to follow the below criteria. Show a password error in the form, and don't Register for an invalid password

- Must have an Uppercase letter in the password
- Must have a Lowercase letter in the password
- Length must be at least 6 character

There will be some other options like

- Show the user a Link for Login so that he can go to the Login page.
- Show users a Social Login Button (Google only) . on Clicking it
  - user authenticates with Google
  - Navigate the user to the Home page.

💡Don't implement email verification or forget password method as it will inconvenience the examiner. If you want, you can add these after receiving the assignment result.

4. **Home Page:** The home page will have the following sections- Navbar,Static Slider,Featured Movies, 2 Extra Sections, and Footer.

🎯 **Make sure to keep the navbar and footer on all the pages except the 404 page.**

- **Banner:** Add a slider (you can use any type of static slider/carousel) with a minimum of 3 slides and meaningful information

- **Featured Movies:** In this section 6 highest rated movies cards.Each card will have: (highest rated means u have to sort the movies based on rating and show 6 movie.)
    - Movie Poster.
    - Movie Title.
    - Genre.
    - Duration.
    - Release Year.
    - Rating.
    - "See Details" button.

Clicking the "See Details" button will navigate the user to the movie details page. Below this section there will be a button called "See all movies".Clicking this button will take the user to the All Movies page.

For showing 6 data you can use the limit operator of MongoDB

- **Extra Section:** Add 2 relevant and meaningful extra sections which are mentioned above on the Home page.

5. **Add movie page: (Private route)** Create an Add movie page where there will be a form for the user to add a movie. The form will have:
    - Movie Poster. (you can upload the country image to imgbb and provide the image link here.) [validation: you must check if the given input is must be a link]
    - Movie Title. [validation: you must check if the given input is not empty and have at least 2 characters]
    - Genre. It will be a dropdown. The values are like ("comedy","drama","horror", etc…)
    - Duration. (It will be a number input and users will give input the duration as minutes). [validation: you must check if the given input is not empty and must be greater than 60]

  - Release Year. (It will be a dropdown of Year and you can take any year value here like 2024,2023,2022,2021 etc) [validation: you must check if the given input is not empty]
  - Rating (Here you will use this package https://www.npmjs.com/package/react-simple-star-rating to integrate rating.) [validation: user must select a rating to add a movie]
  - Summary (Here will be a textarea and user will give a short summary of that movie) [validation: you must check if the given input is not empty and have at least 10 characters]

If any validation fails you will have to show a toast or an error message and block the form submission. **Also add the users email with movie data** when any movie is added. So you can filter movies on the “Favorite movie” page.

- When you fill in the data and click the “Add Movie” button, this data will be stored in your database and you will show a success message through toast/sweet alert. Also this newly added moviewill be shown on the “All Movies” page.

6. **All movies page:** On this page, a user can see all the movies added by all users. Create a 3 column grid layout using cards. Each card will contain this information-
    - Movie Poster.
    - Movie Title.
    - Genre.
    - Duration.
    - Release Year.
    - Rating.
    - “See Details” button.

Clicking the “See Details” button will navigate the user to the movie details page.

Below this section there will be a button called “See all movies”.Clicking this button will take the user to the All Movies page

7. **Movie details page:** **(Private route)** Here you will show all the data which you took through the form on the “Add movie” page. In the details page there will be 2 Buttons. 1) Delete Movie 2) Add to Favorite
   Delete Movie: Clicking the button will delete the movie and navigate to “All Movies” Page
   Add to Favorite: Clicking the button will add that movie to users favorite list

8. **Favorite movie page:** **(Private route)** Here you will show all the movies which an individual added as their favorite. It will also be 3 column layout where you will show -
   - Movie Poster.
   - Movie Title.
   - Genre.
   - Duration.
   - Release Year.
   - Rating.
   - “Delete Favorite” button.

Clicking the Delete favorite button will delete the movie from users favorite list. (Hint: create a database for favorite movie and filter the movies with currently logged in user’s email)

9. **Additional Requirements:**
   - **Footer:** Create a reasonable and meaningful footer. (including website name, copyright, contact information, social media links)
   - **404 page:** Add a 404 page/Not Found Page
   - **Spinner:** Show a loading spinner when the data is in a loading state.
   - **Toast:** For all the CRUD operations, show relevant toast/ notification/ sweet alert with a meaningful message

Challenges

1. Add one more button on the movie details which will be "Update Movie" On Clicking the button users will navigate to a new route (private route) where they can update the movie. There will be a form which will be exactly the same as "Add Movie". You also have to keep the validation the same as "Add Movie".
2. Implement Search functionality on the "All Movies" page based on the Movie Title.you can do it from client side / server side as you want.
3. Use "React hook form" for at least one of your forms.
4. Implement a dark/light theme toggle for the home page.

What to submit:

- Your client-side code GitHub repository
- Your server-side code GitHub repository
- Your live website link