

React Movie Critic Review Project

Overview:

For this scenario, you have been asked to write an application which will provide information about movie critic reviews. Approach this problem as if the application will go to production. We don't expect it to be perfect (no production code is) but it should also not be a hacked together script. Spend whatever time you think is reasonable but no more than **20 hours**. Do the best that you can, with the time you have available.

While the end result may not be your best work, it should represent you and your abilities as a developer.

You can use whatever 3rd party libraries you feel are appropriate but the framework used must be React and attempt to follow React best practices. We've got things started for you with a boilerplate and some examples.

Note: Please use functional components (instead of class components).

Requirements:

The following features should be within the site:

1. A reviews list should be displayed by publication date with the most recent first and only display the image, title, date of publication, MPAA Rating and Critics pick. We'd like the amount of reviews displayed to be variable with a default of 20 upon loading the page. The user may increase this but no more than 50 at a time.
 - a. Users should be able to search by title and filter by:
 - i. MPAA Rating
 - ii. Publication date
 - iii. Critic's pick
 - b. The search functionality should apply the filter as the user types
 - c. Clicking a review should display:
 - i. Title
 - ii. image
 - iii. MPAA Rating
 - iv. if it's a critic's pic
 - v. headline
 - vi. summary of the review
 - vii. the critic's name (re: byline)
 - viii. link to the article
 - d. Users search and/or filters should not be lost after viewing a review

2. A critics page with the number of reviews each critic has written and how many are critics pick. Please display the critic's image and bio if they are available in the list.

Data

The data is available in *static/movie-reviews.json* and *static/critics.json*.

Setup:

We recommend using nvm for node version management. The project is set up with webpack and express for the development environment. We used node lts/dubnium (v10.22.1).

Once you have nvm (or the correct version of node) all you should need to run is below.

```
1. If using nvm, otherwise skip to 2
```

```
first time:
```

```
nvm install lts/dubnium
```

```
every other time:
```

```
nvm use lts/dubnium
```

```
2. npm i
```

```
3. npm start
```

```
The project should be running on localhost:8000. Show us what you got!
```

Please note: We are looking for you to demonstrate your knowledge related to common software practices to include reusability, portability, and encapsulation – to name a few. Interface should also be quite responsive with little or no wait times. Work submitted should be in project form and implemented as you were implementing any production solution.