

Web App Design with React Week 2 Coding Assignment

Points possible: 70

Category	Criteria	% of Grade
Functionality	Does the code work?	25
Organization	Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear.	25
Creativity	Student solved the problems presented in the assignment using creativity and out of the box thinking.	25
Completeness	All requirements of the assignment are complete.	25

Instructions: In VS Code, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document, with your JavaScript project code, to the repository. Add the URL for this week's repository to this document where instructed and submit this document to your instructor when complete.

Coding Steps:

- Using what you've learned this week, create a page of an application that enables users to vote and leave reviews on movies.
- You should include at least the following components:
 - MovieList – a container for all the Movie components and their data.
 - Movie – a component that represents movie data (i.e. image, synopsis, rating, etc...)
 - Stars – a one to five-star rating component that allows users to rate something (movies in this case, but remember that components are reusable, so you could use it elsewhere!)
 - ReviewList – a container inside of a Movie that houses Review components.

- Review – A text review a user can leave on a movie.
- ReviewForm – a form at the bottom of a Movie component that allows users to leave reviews. When submitted, the review should be added to the movie. All this data can be stored in an array, no networking or database needed for this assignment.

Screenshots of Code:

```

JS indexjs M X JS Appjs M JS moviejs U JS movie-listjs U JS reviewjs U JS review-formjs U
src > JS indexjs
1 import React from 'react';
2 import ReactDOM from 'react-dom';
3 import './node_modules/bootstrap/dist/css/bootstrap.css';
4 import './index.css';
5 import App from './App';
6 import reportWebVitals from './reportWebVitals';
7 import Movie from './components/movie';
8
9 ReactDOM.render(
10   <Movie />,
11   document.getElementById('root')
12 );
13
14 // If you want to start measuring performance in your app, pass a function
15 // to log results (for example: reportWebVitals(console.log))
16 // or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
17 reportWebVitals();
18

```

```

JS indexjs M JS Appjs M X JS moviejs U JS movie-listjs U JS reviewjs U JS review-formjs U
src > JS Appjs > App
1 import logo from './logo.svg';
2 import './App.css';
3 import Movie from './components/movie';
4 import ReviewForm from './components/review-form';
5
6 function App() {
7   return (
8     <div className="App">
9       <Movie />
10      <ReviewForm />
11    </div>
12   );
13 }
14
15 export default App;
16

```

```
JS index.js M JS App.js M JS movie.js U X JS movie-list.js U JS reviews.js U JS review-form.js U
src > components > JS movie.js > Movie > render > etReviews
1 import React from 'react';
2 import MovieList from './movie-list';
3 import ReviewForm from './review-form';
4
5 export default class Movie extends React.Component{
6   render(){
7
8     let backToTheFutureReviews = [
9
10      {
11        user:'Matt',
12        review:'Fun film!',
13        stars:'4'
14      },
15      {
16        user:'Sally',
17        review:'Brings back memories',
18        stars:'3'
19      }
20    ];
21
22    let homeAloneReviews = [
23      {
24        user:'Christina',
25        review:'Never gets old!',
26        stars:'4'
27      },
28      {
29        user:'Jerry',
30        review:'A bit dated, but still fun.',
31        stars:'3'
32      }
33    ];
34
35    let etReviews = [
36      {
37        user:'Mary',
38        review:'A moving, nice film about friendship.',
39        stars:'4'
40      },
41      {
42        user:'Jack',
43        review:'I watch it every year!',
44        stars:'5'
45      }
46    ];
```

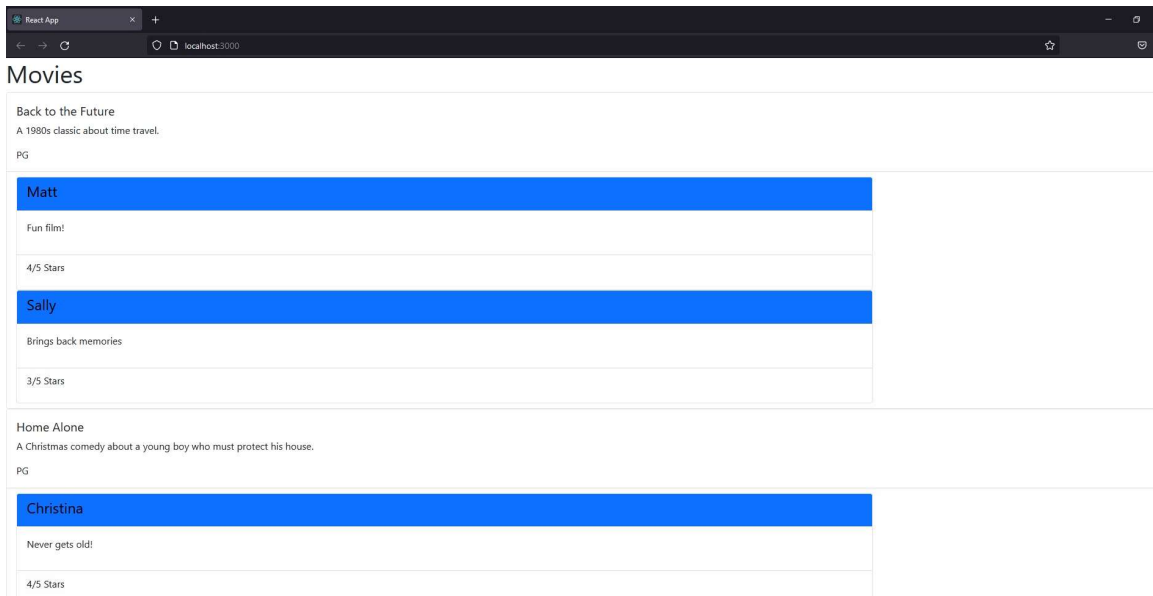
```
JS index.js M JS App.js M JS movie.js U X JS movie-list.js U JS reviews.js U JS review-form.js U
src > components > JS movie.js > Movie > render > etReviews
43      stars:'5'
44    }
45  ];
46
47  return <div>
48    <div>
49      <h1>Movies</h1>
50      <MovieList {...{reviews: backToTheFutureReviews, name:'Back to the Future', synopsis: 'A 1980s classic about time travel.', rating: 'PG'}}/>
51      <MovieList {...{reviews: homeAloneReviews, name:'Home Alone', synopsis: 'A Christmas comedy about a young boy who must protect his house.', rating: 'PG'}}/>
52      <MovieList {...{reviews: etReviews, name:'E.T. the Extra-Terrestrial', synopsis: 'A sci-fi film about a child helping a friendly alien back home.', rating: 'PG'}}/>
53    </div>
54  </div>
55
56  }
57 }
```

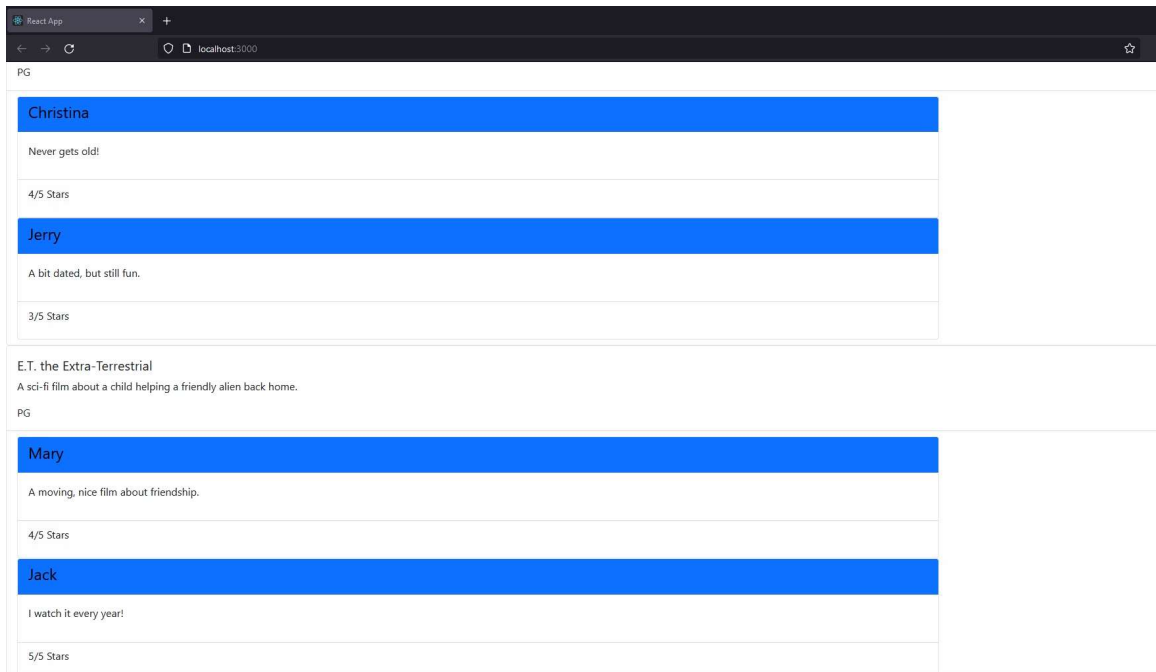
```
JS indexjs M JS Appjs M JS moviejs U JS movie-listjs U X JS reviewsjs U JS review-formjs U
src > components > JS movie-listjs > MovieList > render
1 import React from 'react';
2 import Reviews from './reviews';
3
4 export default class MovieList extends React.Component{
5   constructor(props){
6     super(props);
7     this.state = {
8       reviews: props.reviews,
9       name: props.name,
10      name: props.synopsis,
11      name: props.rating
12    };
13  }
14
15  render(){
16    let reviews;
17
18    if(this.state.reviews){
19      reviews = this.state.reviews.map((review, index)=> <Reviews key={index} {...review}/>);
20    }
21
22    return (
23      <div className="card">
24        <div className="card-body">
25          <h5 className="card-title">{this.props.name}</h5>
26          <p className="card-text">{this.props.synopsis}</p>
27          <p className="card-text">{this.props.rating}</p>
28        </div>
29        <div className="card-footer">
30          {reviews}
31        </div>
32      </div>
33    );
34  }
35 }
```

```
JS indexjs M JS Appjs M JS moviejs U JS movie-listjs U JS reviewsjs U X JS review-formjs U
src > components > JS reviewsjs > Reviews > render
1 import React from 'react';
2
3 export default class Reviews extends React.Component{
4   render(){
5     return (
6       <div className="card w-75">
7         <div className="card-header bg-primary text-black">
8           <h4>{this.props.user}</h4>
9         </div>
10        <div className="card-body">
11          <p>{this.props.review}</p>
12        </div>
13        <div className="card-footer">
14          <p>{this.props.stars}/5 Stars</p>
15        </div>
16      </div>
17    );
18  }
19 }
20 }
```

```
JS index.js M JS App.js M JS movie.js U JS movie-list.js U JS reviews.js U JS review-form.js U X
src > components > JS review-form.js > ReviewForm > render
1 import React from 'react';
2
3 export default class ReviewForm extends React.Component{
4   render(){
5     return (
6       <div class="container form-control border border-primary">
7         <form class="card-body">
8           <div className="form-group">
9             <label>What do you think about this film?</label>
10            <input type="text" className="form-control" placeholder="Enter review"/>
11          </div>
12          <button type="submit" className="btn btn-primary btn-block">Submit</button>
13        </form>
14      </div>
15    );
16  }
17 }
```

Screenshots of Running Application:





URL to GitHub Repository:

<https://github.com/MonicaHunter0812/week2react.git>