

| | |
|---------------------|----------|
| Introduction | 1 |
| Files | 1 |
| Overview | 2 |
| Components | 4 |
| Conclusion | 5 |

Introduction

MovieCapital is a react-based application that allows the user to enter the name of a movie which then is entered as a search that fetches inquired data from the site OMBD and their API.

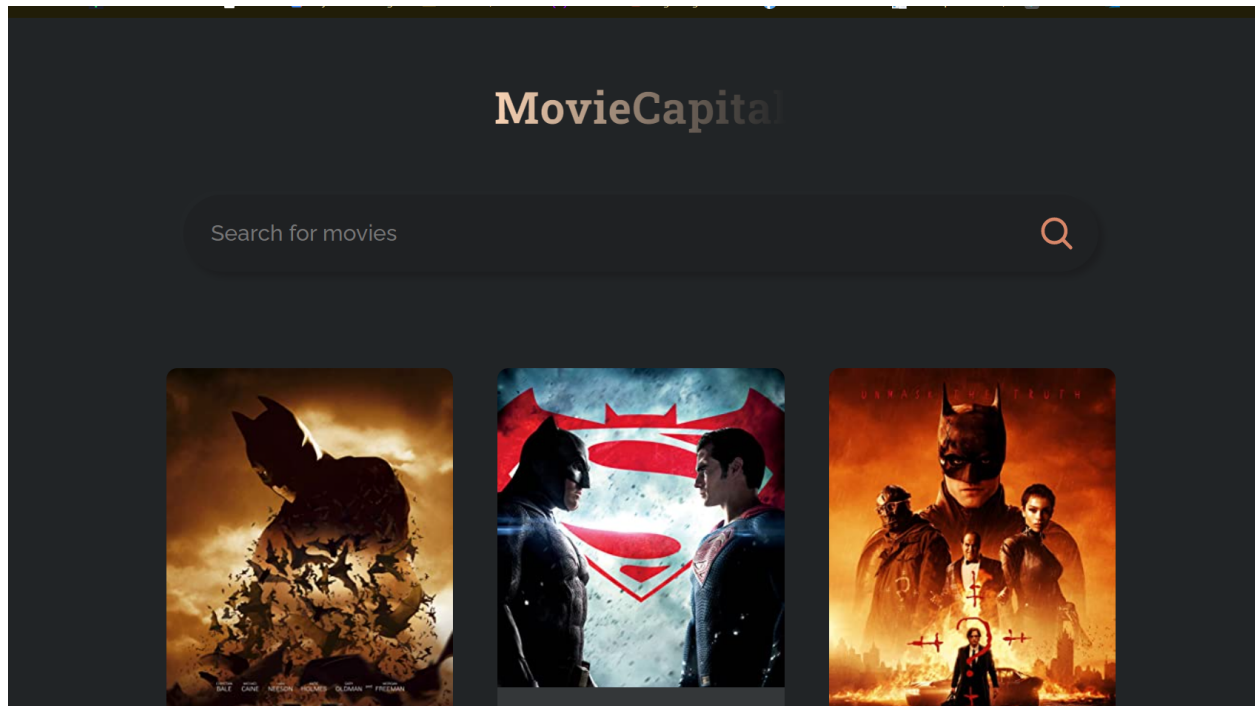
The following application was coded with react to build a modernized web platform that supports displaying movie information through REST API methods, incorporating REACT, JS, and Node.JS.

Files

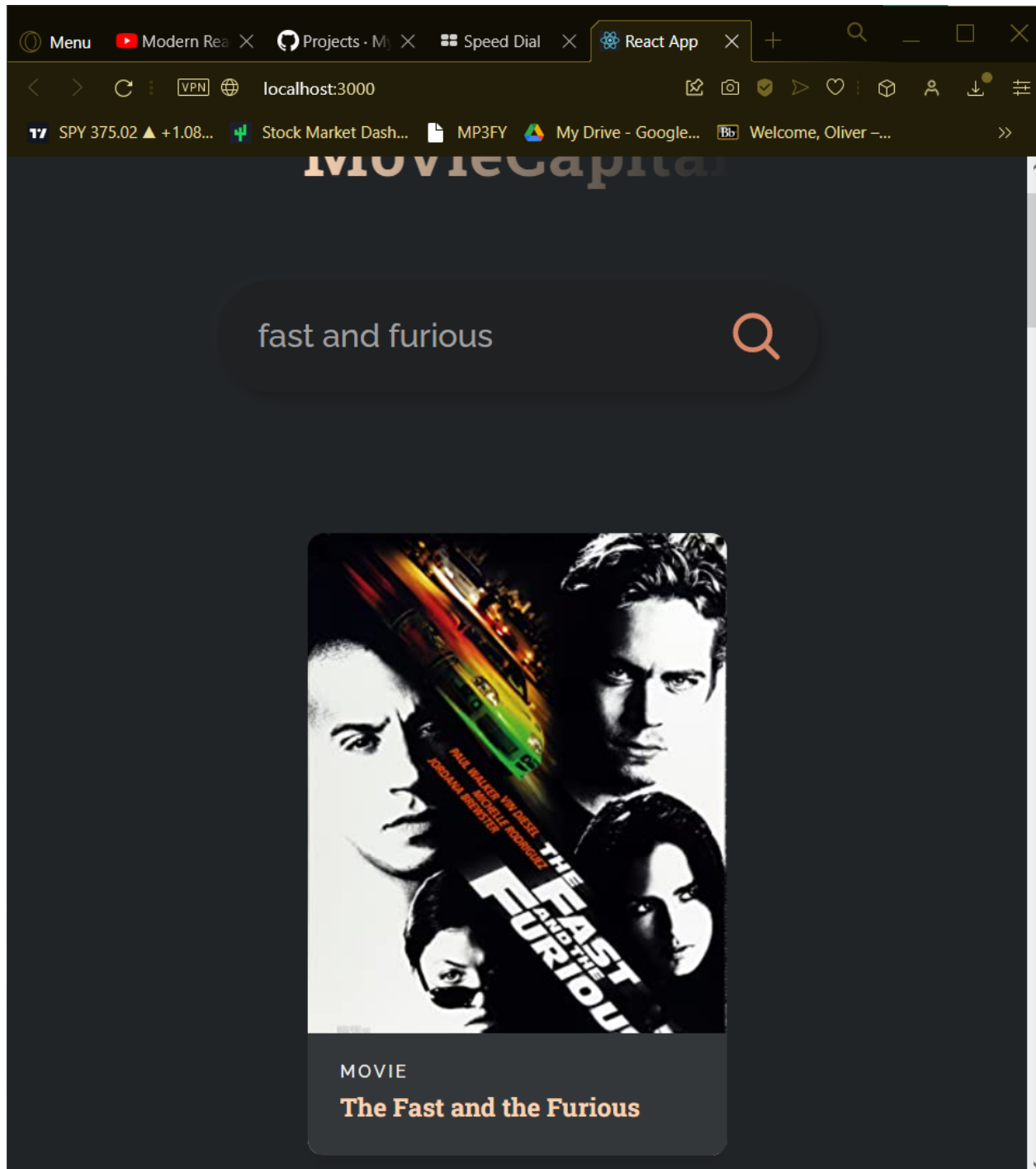
Public: Contains all files that are in the root of the directory when published through a server/local host. This folder contains the main index.html file which is displayed and manipulated through the React framework.

SRC: Within the repository, the files that are provided are the original src file that contains the index.js which allows the user to run the React JSX application.

Overview



The main features of the web application allow the user to search for a certain movie and get results returned to them in a clear and concise format that contains the movie's images, titles, year release date, etc. The entire web app uses Css for its design implementation that works seamlessly with React JSX framework.



There is click functionality, That being said the enter key functionality was never within the project scope and therefore, it was never implemented.

Components

There are two main components that are used within the project. The first is the App() component which has the purpose of displaying the total functionality of the web application. It uses sub-components as well as other methods/techniques such as hooks, and call back functions to render the final output.

```
import MovieCard from "./MovieCard";
import SearchIcon from './search.svg';

/*
  Author: Faisal Chowdhury (MD A Faisal)

  this is a demo application so normally the api keys will be hidden on the backend and
  this
*/

//7164c63e
const api_ulr = 'http://www.omdbapi.com?apikey=7164c63e'
/* const movie1 = {
  "Title": "Italian Spiderman",
  "Year": "2007",
  "imdbID": "tt2705436",
  "Type": "movie",
  "Poster": "https://m.media-amazon.com/images/M/
  MV5BYjFhN2RjZTctMzA2Ni00NzE2LWJmYjMtNDAYYTLlOTkyMmY3XkEyXkFqcGdeQXVyNTA0OTU0OTQ@._V1_S
} */

const App = () =>{

  const [movies, setMovies] = useState([]); //useState default value is an empty array
  const [searchTerm, setSearchTerm] = useState('');

  const searchMovies = async (title) =>{

    // template string uses `` and not ''
    const response = await fetch(`${api_ulr}&s=${title}`);
    const data = await response.json();

    setMovies(data.Search);
  }
}
```

The movie card component allows us to display the movie object based on the movie prop that is passed through its parameters within the main App() component. This is to display an individual movie card that displays all movie information that is retrieved. We use this component to map it to all the search results returned and use this to templatize and fill the webpage with many movie cards based on the search results.

```
1 import React from "react";
2
3
4
5 const MovieCard = ({movie}) =>{
6
7   return(
8
9     <div className="movie">
10       <div>
11         <p>
12           {movie.Year}
13         </p>
14       </div>
15
16       <div>
17         <img src={movie.Poster !== 'N/A' ? movie.Poster : 'https://via.placeholder.com/400'} alt={movie.
18           Title} />
19       </div>
20
21       <div>
22         <span>{movie.Type}</span>
23         <h3>{movie.Title}</h3>
24       </div>
25     </div>
26   );
27
28 }
29
30
31
32
33 export default MovieCard;
```

Conclusion

This app is designed for the purpose to display a certain series of movies through the request made by the user. This shows one of the main features of React which allows the developers to potentially reuse components. The JSX(React) language allows the developer to integrate a seamless experience for the end user through the JSX and the virtual DOM.

