

Frontend Path: Movie and TV Series Tracking Web Application

This document outlines the steps to build a movie and TV series tracking web application, similar to The Movie Database (TMDB). The project will utilize Vue 3, Tailwind CSS, and be deployed using Docker.

# Learning Resources

* [Vue.js](https://vuejs.org/guide/introduction.html)
* [Tailwind CSS - Rapidly build modern websites without ever leaving your HTML.](https://tailwindcss.com/)

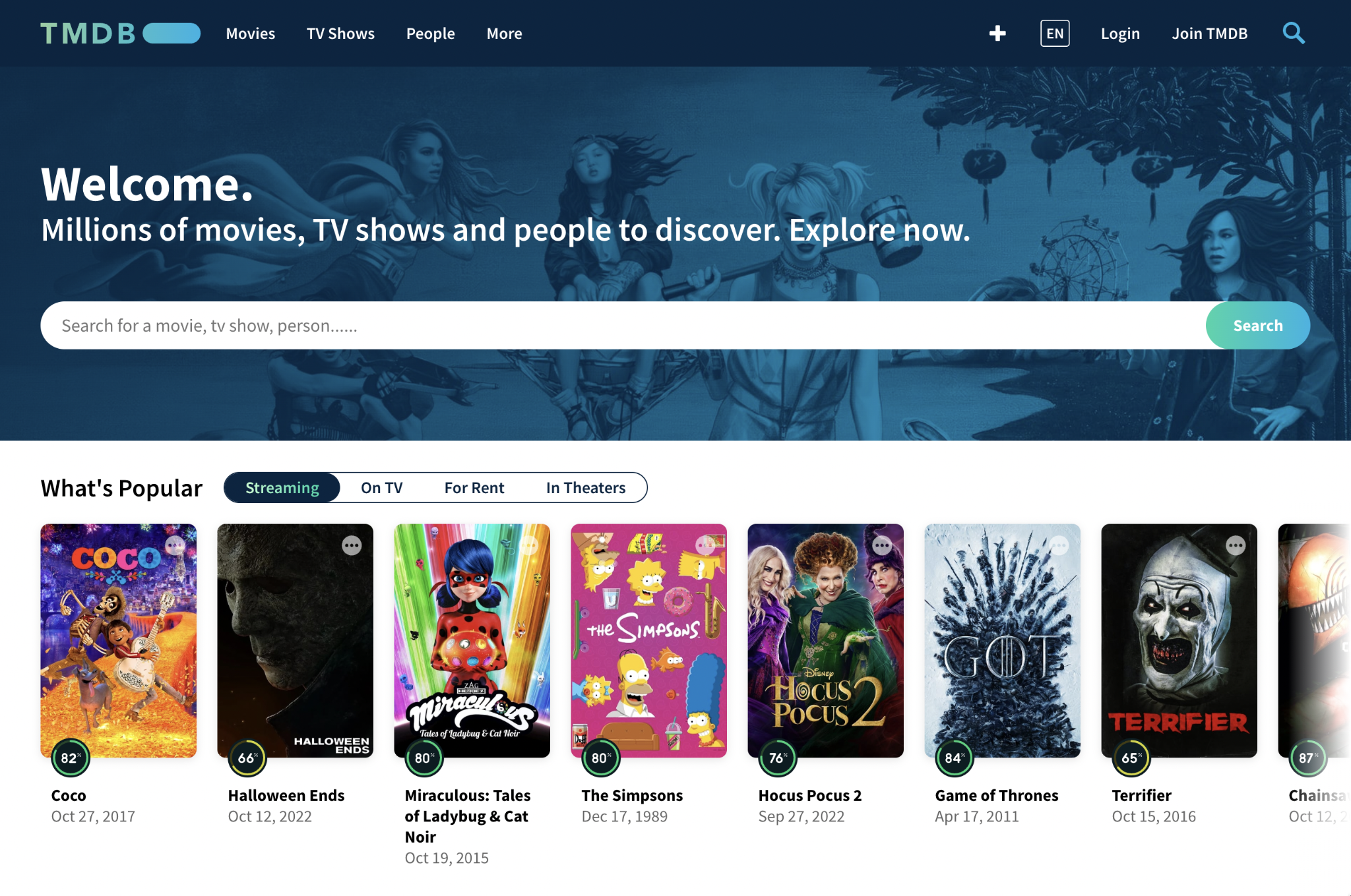
# Project Overview

Build a movie and TV series tracking web application similar to [The Movie Database (TMDB)](https://www.themoviedb.org/).

**API Key:** 348088421ad3fb3a9d6e56bb6a9a8f80  
**Getting Started with TMDB API:** <https://developers.themoviedb.org/3/getting-started/introduction>

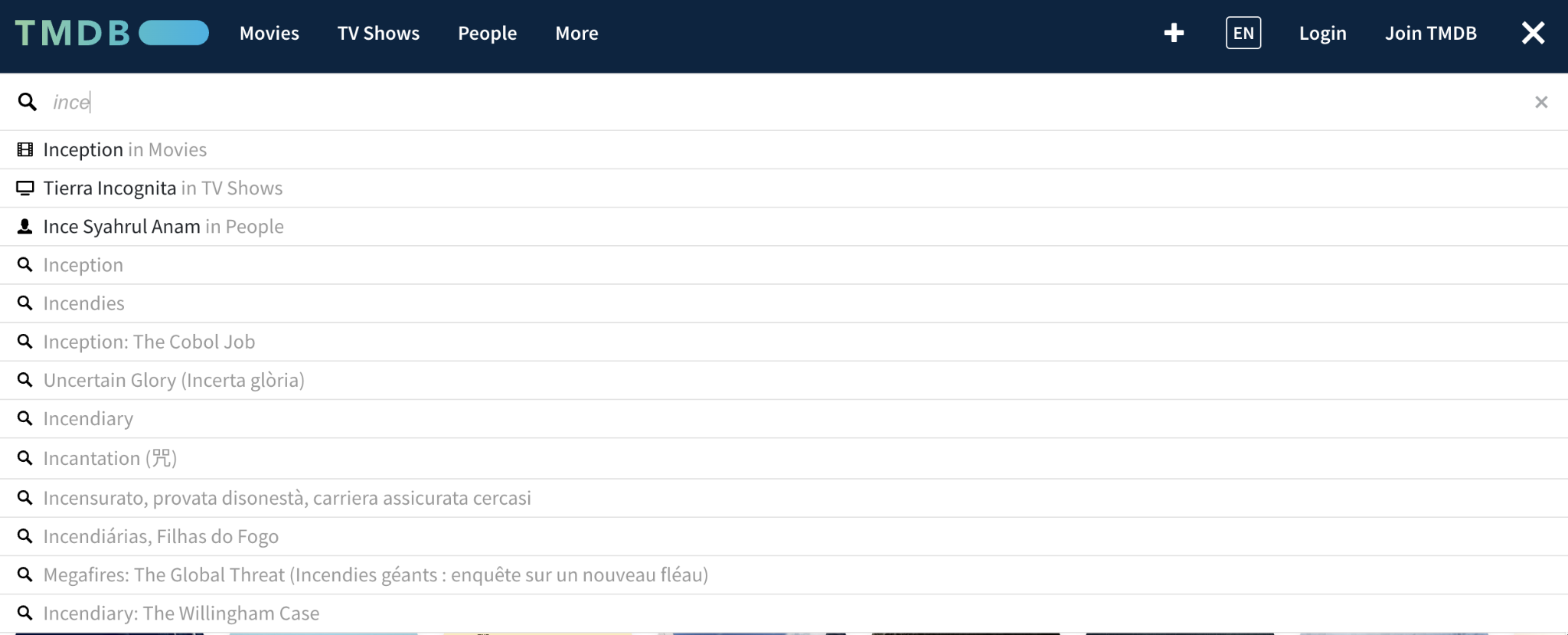
# Phase 1: Initial Setup and Core Components

* Create a Vue 3 application locally and add Tailwind CSS.
* Create a layout component, add “Movies” and “TV Shows” navigation.
* Add a big search input component with background similar to the screenshot.
* Create a selectable pagination that shows popular and top rated Movies, TV Shows. Only fetch the related data when user clicked on the link.
* Create a movie poster component along with a rating component to show fetched movies with their ratings.



# Phase 2: Enhanced Functionality

* Implement a search functionality that list related results after the third letter. Throttle the seach so that it only send request to backend if user didn’t type after 500ms.
* Add a star button to posters so user can favourite the movie. When user favourited a movie make the star icon filled. Implement store functionality (composable, pinia or vuex) to keep favourite data across the application.

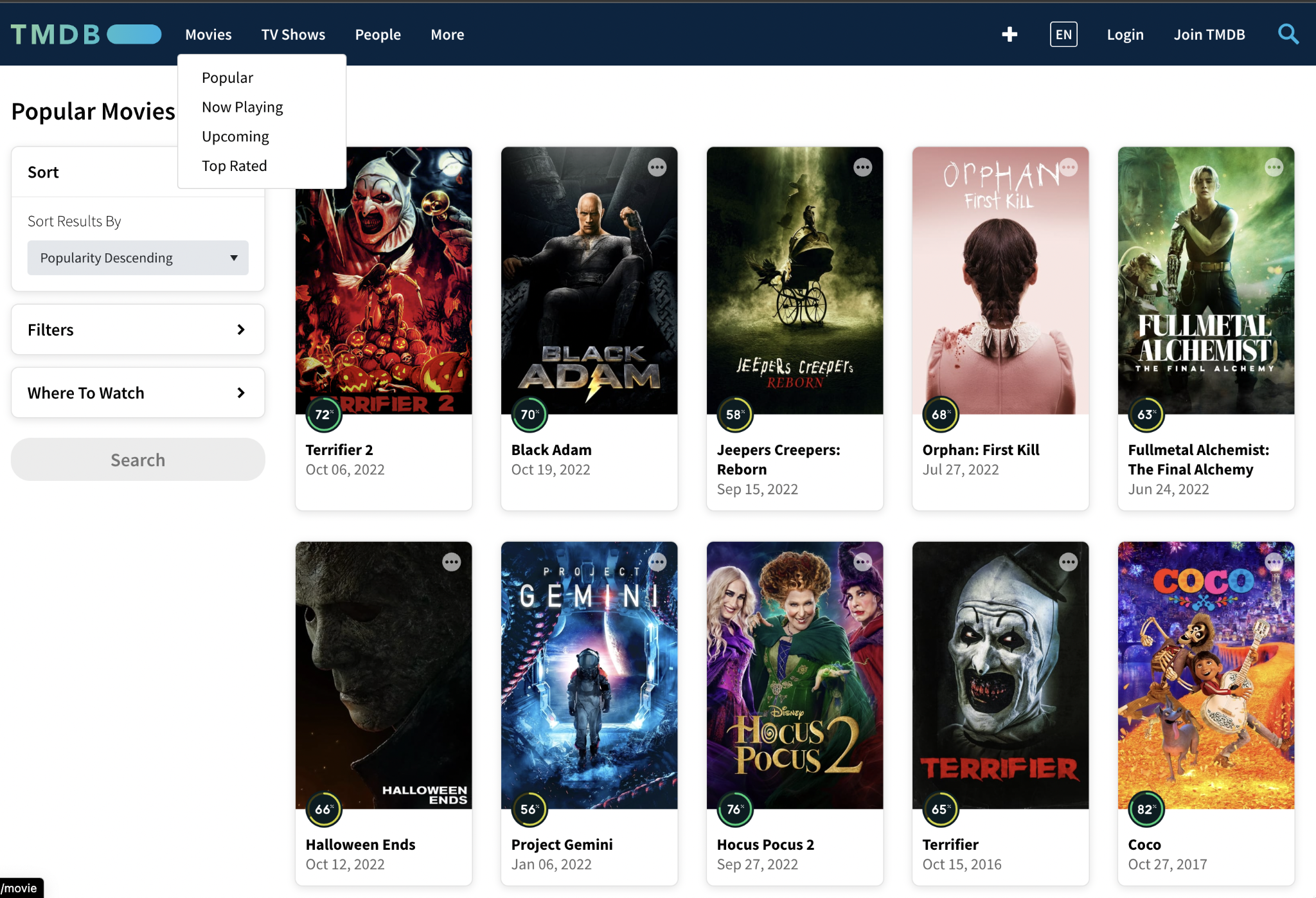


# Phase 3: Server-side Rendering

* Understand the difference between Server-Side Rendering and Client-Side Rendering. Move your project to Nuxt3. It should be fairly easy since structure is pretty much the same.
* [Server-Side Rendering (SSR) Vs Client-Side Rendering (CSR)](https://dev.to/codewithtee/server-side-rendering-ssr-vs-client-side-rendering-csr-3m24)
* [Web Apps: Server-side (SSR) vs. Client-side Rendering (CSR) | Toptal®](https://www.toptal.com/front-end/client-side-vs-server-side-pre-rendering#:~:text=Client%2Dside%20rendering%20manages%20the,displays%20a%20blank%20page%20first.)

# Phase 4: Filter and Sort

* Make a popover on navigation menu which shows popular, upcoming and top rated for both movies and tv shows.
* Create folder based routes for this 6 pages and list all the movies associated.
* Implement a filter and sort functionality with avaliable information.



# Phase 5: Dynamic Pages

* Create dynamically created movie pages that have different URLs.
* Movie page should have poster, user score, overview, actors and other informations like in the examples.
* User should be able to favourite the movie in this page as well.

# Phase 6: Dockerize

[Docker from Zero to Hero | Dockerizing Full Stack Web App REACTJS & NODEJS w/docker-compose](https://www.youtube.com/watch?v=IDVUy34vlSE)

* Dockerize your project after watching the tutorial.
* You need to create Dockerfile for your project.
* Your need to use docker-compose.