

Descripción General:

Este proyecto es una página web de reseñas de películas que permite a los usuarios buscar, filtrar y visualizar información sobre diferentes películas. También incluye funcionalidades como la calificación con estrellas, la visualización de trailers y agregar a favoritos.

Estructura del Proyecto:

El proyecto está dividido en varios módulos y componentes que trabajan juntos para proporcionar una experiencia completa al usuario. A continuación, se detallan los componentes principales y las decisiones técnicas importantes tomadas durante el desarrollo.

Componentes del Proyecto:

App.js

Funcionalidad: Es el componente principal que maneja la lógica de la aplicación, incluyendo la búsqueda y el filtrado de películas, la visualización de detalles y trailers, y la gestión de favoritos.

Decisiones Técnicas:

Uso de `useState` y `useEffect` para manejar el estado y los efectos secundarios.

Integración con la API de The Movie Database (TMDb) para obtener datos de películas.

Implementación de funciones de búsqueda y filtrado dinámicas.

Main.js

Funcionalidad: Contiene el componente principal de la página que muestra el carrusel de películas y la lista de películas.

Decisiones Técnicas:

Uso de componentes secundarios como `MovieCarousel` y `MovieList` para dividir la lógica y mejorar la reutilización del código.

StarRating.js

Funcionalidad: Componente encargado de mostrar la calificación de una película en forma de estrellas.

Decisiones Técnicas:

Uso de la función `Math.floor` para determinar el número de estrellas llenas y la lógica condicional para manejar medias estrellas.

Uso de clases CSS para el estilo visual de las estrellas.

Footer.js

Funcionalidad: Muestra el pie de página de la aplicación.

Decisiones Técnicas:

Sencillo componente funcional para mantener el pie de página consistente en toda la aplicación.

MovieCarousel.js

Funcionalidad: Muestra un carrusel de películas destacadas.

Decisiones Técnicas:

Uso de la librería react-slick para la funcionalidad del carrusel.

Personalización de las flechas del carrusel mediante componentes CustomPrevArrow y CustomNextArrow.

Estilos CSS

Los estilos CSS se usan para mejorar la presentación visual de la aplicación y asegurar una buena experiencia de usuario.

App.css

Estilos generales para la aplicación, incluyendo ajustes de fuentes y colores de fondo.

MovieList.css

Funcionalidad: Estilos específicos para la lista de películas.

Decisiones Técnicas:

Flexbox para la disposición de las películas en un formato de cuadrícula.

Ajustes de márgenes y anchos máximos para una mejor visualización.

MovieDetail.css

Funcionalidad: Estilos específicos para la visualización de los detalles de una película.

Decisiones Técnicas:

Estilo responsivo para ajustarse a diferentes tamaños de pantalla.

Logocss.css

Funcionalidad: Estilos específicos para el logo y el menú de navegación.

Decisiones Técnicas:

Uso de flexbox para alinear los elementos del menú y el logo.

Ajustes de márgenes y paddings para un mejor espaciado visual.

StarRating.css

Funcionalidad: Estilos específicos para el componente de calificación por estrellas.

Decisiones Técnicas:

Ajuste del tamaño y color de las estrellas para una mejor visualización.

Decisiones de Diseño y Funcionalidad

Integración de la API de TMDb:

Se decidió usar la API de TMDb para obtener datos actualizados y precisos sobre películas.

Se implementaron varias funciones para buscar, filtrar y obtener detalles adicionales como trailers y reparto.

Manejo de Estados y Efectos:

Uso extensivo de useState y useEffect para manejar el estado de la aplicación y los efectos secundarios como llamadas a la API.

Componentes Reutilizables:

Dividir la aplicación en componentes reutilizables como MovieCarousel, MovieList y StarRating para mejorar la mantenibilidad y la reutilización del código.

Estilos CSS Personalizados:

Creación de varios archivos CSS para manejar estilos específicos y mejorar la presentación visual de la aplicación.

Uso de flexbox y otros métodos de diseño CSS modernos para una disposición responsiva y atractiva.

Gestión de Favoritos:

Implementación de una funcionalidad para agregar y quitar películas de una lista de favoritos.

Uso de estados para manejar la lista de favoritos y su visualización.

Lista de archivos y Directorios Principales:

src/

App.js

Main.js

StarRating.js

Footer.js

MovieCarousel.js

MovieList.js

styles/

App.css

MovieList.css

MovieDetail.css

Logocss.css

StarRating.css