

SPA - MELISEARCH

Una aplicación de React responsiva que permite buscar y ver información de un artículo utilizando la API de Mercado Libre.

Link Vercel: [Haga clic aquí](#) para acceder al proyecto.

Descripción

La aplicación es una especie de catálogo de productos que debe contener una lista de productos y una pantalla de detalles para cada uno. Además, la aplicación debe permitir hacer filtros por búsqueda, categoría y precio.

Start de la aplicación

Para iniciar las dependencias del proyecto, después de clonar el SSH, abrir la terminal y usar el comando yarn, y después de eso usar el comando yarn start. Enlace de Vercel: [Haga clic aquí](#) para acceder al proyecto.

Bibliotecas utilizadas

ReactJs

Styled components

Context API

React toastify

React Router DOM

Axios

Redux

React-js-pagination

React-responsive-carousel

Tecnologías utilizadas

VS Code

Github

Javascript

Node 16.18.0

Funcionalidades

Generales

- Consumo de la API de Mercado Libre
- Lista de productos
- Modal que muestra los detalles del producto seleccionado
- Filtrado por título, precio y categorías de los productos
- Agregar y quitar el producto seleccionado al carrito

Informe de errores

Por hacer:

Corregir error en la estilización del modal.

Corregir error en la paginación, que debe poder hacer clic en cualquier página y renderizar los productos de esa página.

Documentación de la API de Mercado Libre

[Haga clic](#) aquí para acceder a la API de Mercado Libre.

Requisitos detallados para las funcionalidades:

Crear una aplicación de React que permita buscar y ver información del artículo utilizando la API (<https://api.mercadolibre.com/sites/MLA/search?q=:query>), SIN usar los colores de Mercado Libre.

La aplicación debe implementar Server-Side Rendering (SSR) utilizando una tecnología de tu elección, como Next.js, Nuxt.js, o cualquier otro marco o biblioteca de tu preferencia.

La aplicación debe tener una página de búsqueda y un Modal o Pop up de detalles del artículo.

En la página de búsqueda, el usuario debe poder ingresar el nombre del artículo y buscarlo.

La página de búsqueda debe mostrar los resultados de la búsqueda en tarjetas (imagen, nombre, precio etc) y permitir al usuario hacer clic en un resultado para ver el Modal o Pop up de detalles del artículo.

En el Modal o Pop up de detalles del artículo se podrá consultar desde la API <https://api.mercadolibre.com/items/:id>, se debe mostrar toda la información disponible, incluyendo la imagen, nombre, precio, descripción, tipo etc y un botón de compra.

Para ver la descripción <https://api.mercadolibre.com/items/:id/description>.

La aplicación debe ser responsive y tener un diseño agradable (Tablets $\geq 768\text{px}$, Laptops $\geq 992\text{px}$ y Desktop $\geq 1200\text{px}$).

Bonus

Implementa la funcionalidad de paginación en la página de resultados

Implementa una funcionalidad de búsqueda avanzada que permita al usuario filtrar los resultados por precio, tipo, descripción etc

Utiliza Redux o cualquier otra biblioteca de gestión de estado para manejar el estado de la aplicación

Crea pruebas unitarias para cada componente funcional que verifiquen su comportamiento en diferentes escenarios (por ejemplo, componentes que renderizan correctamente, componentes que reciben las props adecuadas, componentes que manejan los errores correctamente, etc)

Crea pruebas de integración para verificar el comportamiento de la aplicación en diferentes navegadores y dispositivos, y para asegurarte de que los componentes interactúen correctamente entre sí

Implementar React testing-library ESLint + Prettier

Implementa manejo adecuado de errores y mensajes de error para una experiencia de usuario mejorada