



Challenge Frontend Developer

Se requiere realizar una aplicación web con motivo a las películas de Star Wars.

Los requerimientos son los siguientes:

Fuente de datos:

Se deberá hacer uso de SWAPI, una API con toda la información esperada de los 6 Films de Star Wars y completamente abierta <https://swapi.dev>

Landing Page:

Se debe realizar una landing la cual permita ir a la sección de films o a la sección de personajes

Página de films:

Cada film debe tener la siguiente información:

- Nombre
- Imagen genérica (debe ser la misma para todos los films)
- Número de episodio

Página de detalle del film:

El film debe mostrar la siguiente información:

- Nombre
- Imagen genérica
- Número de episodio
- Director
- Personajes: (deben tener nombre, imagen genérica y url que lleve al detalle de cada personaje)



Página de personajes:

Cada Personaje en la lista debe mostrar las siguientes características:

- Nombre
- Imagen genérica (debe ser la misma para todos los personajes)
- Color de Ojos
- Género

Se puede implementar un paginado de 10 personajes por página o también un scroll infinito, eso queda a elección del desarrollador

Filtros:

Como plus se pueden implementar los siguientes filtros para la página de personajes:

- Filtro por color de ojos
- Filtro por género

Nota: Los filtros se deben aplicar en todos los personajes, no en los primeros 10

Página del detalle de personaje:

En la página del personaje se debe mostrar la siguiente información:

- Nombre
- Imagen genérica
- Color de ojos
- Año de cumpleaños
- Color de pelo
- Altura
- Color de piel
- Masa

Aclaración:

Las propiedades que tengan valor “n/a” o “unknown” no deberían mostrarse en la aplicación



Consideraciones generales:

- ★ Los estilos se deben desarrollar con tailwind.
- ★ Se espera que la aplicación web sea desarrollada con next.
- ★ Se debe usar la api que provee Next.
- ★ El lenguaje de programación es a elección del desarrollador (JavaScript o TypeScript).
- ★ Se debe navegar entre páginas haciendo uso del sistema de rutas del framework.
- ★ Se espera que la data que recibe de SWAPI sea mapeada de acuerdo a como la vista va a mostrar.
- ★ No usar la librería que SWAPI ofrece.
- ★ Como plus, la aplicación puede ser Responsive

Nomenclatura de commits

Los commits deberán tener las siguiente nomenclatura:

prefijo(-): mensaje

Los prefijos pueden ser:

- Add: hace referencia a un nuevo archivo/componente/función
- Fix: hace referencia a algún arreglo del código
- Drop: hace referencia a alguna eliminación de código y/o archivo
- Refactor: hace referencia a alguna refactorización del código

Ejemplo: add(-): create card component



Se evaluará

- Arquitectura (forma de manejar las carpetas dentro del proyecto)
- Prácticas de nombramiento de métodos y variables.
- Uso y manejo de datos externos.
- Uso de Client Side Rendering y Server Side Rendering
- Uso de Lazy Load para el routing de la aplicación.

Nota final

Esta prueba busca definir tu nivel en el manejo de Next, no determina que seas buen@ o mal@, recuerda que por no saber todos los conceptos no necesariamente eres mal@, puedes ayudarte de internet, pero se honest@, trata de escribir todo el código que puedas para que nosotros podamos identificar tu nivel.