

CHALLENGE NODE

Objetivo

Desarrollar una API para explorar el mundo de Disney. La idea de la misma es que permita conocer y modificar los personajes que lo componen y entender en qué películas estuvieron. La API creada deberá devolver la información para que cualquier frontend pueda consumirla.

Requerimientos técnicos

El modelado de datos puede implementarse y extenderse según el criterio del aplicante, como mínimo debe contener las siguientes entidades (pueden agregarse otros atributos que consideren correctos):

Personaje: deberá tener al menos una imagen, nombre, edad, peso, historia, y películas o series relacionadas (en las que participó).

Película o Serie: podrá tener muchos personajes asociados, tendrá una imagen, título, fecha de creación, calificación (del 1 al 5) y género.

La API deberá realizar las siguientes operaciones:

Autenticación de Usuarios

Para realizar peticiones a los endpoints subsiguientes el usuario deberá contar con un token que obtendrá al autenticarse. Para ello, deberán desarrollarse los endpoints de registro y login, que permitan obtener el token.

Listado de Personajes

El listado deberá mostrar solamente imagen, nombre.

Creación, Edición y Eliminación de Personajes

Deberán existir las operaciones básicas de creación, edición y eliminación de personajes.

Detalle de Personaje

En el detalle deberán listarse todos los atributos del personaje, como así también sus películas o series relacionadas.

Búsqueda de Personajes

Deberá permitir buscar por nombre, y filtrar por edad, peso o películas/series en las que participó.

Listado de Películas

Deberá mostrar solamente los campos imagen, título y fecha de creación.

Detalle de Película / Serie con sus personajes

Devolverá todos los campos de la película o serie junto a los personajes asociados a la misma

Creación, Edición y Eliminación de Película / Serie

Deberán existir las operaciones básicas de creación, edición y eliminación de películas o series.

Búsqueda de Películas o Series

Deberá permitir buscar por título, y filtrar por género. Además, permitir ordenar los resultados por fecha de creación de forma ascendiente o descendiente.

Requerimientos funcionales

- La API deberá desarrollarse con Nodejs y Express para el servidor y Sequelize como ORM.
- Las rutas deberán seguir el patrón REST.
- No es necesario armar el Frontend.
- Es deseable documentar los endpoints utilizando alguna herramienta como Postman o Swagger.
- El modelado de datos puede ser extendido según el criterio del aplicante en base a los requerimientos solicitados.

Criterios a evaluar

- Conocimientos básicos de Nodejs.
- Validación de datos.
- Buenas prácticas de codificación.
- Buenas prácticas para nombre de rutas.

Tests

De forma opcional, se podrán agregar tests de los diferentes endpoints de la APP, verificando posibles escenarios de error:

- Campos faltantes o con un formato inválido en BODY de las peticiones
- Acceso a recursos inexistentes en endpoints de detalle

Los tests pueden realizarse utilizando Mocha + Chai u otras herramientas que se consideren convenientes.