ETFLX

API TÍTULOS DE NETFLIX

Javier Requejo y Gonzalo Valdez

Descripción de el proyecto

Servicio web para la gestión de una extensa biblioteca de películas y series de Netflix. La API diseñada permite realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sobre una base de datos MongoDb, que incluye películas y series, facilitando así la gestión integral de estos contenidos.

Funcionalidades principales



Operaciones CRUD sobre películas y series: Los usuarios pueden añadir nuevas películas y programas de Netflix, obtener detalles específicos por ID, actualizar la información existente, y eliminar registros de la base de datos.



Gestión de géneros y directores: Además de las operaciones sobre películas y programas de TV individuales, nuestra API también permite la gestión de colecciones relacionadas, como géneros y directores, proporcionando así una forma de organizar mejor el contenido según sus características distintivas.



Integración con API externa: Para enriquecer la información disponible y ofrecer detalles adicionales, nuestra API consume servicios de una API externa de películas, lo que permite obtener datos actualizados y extensivos sobre las películas y programas de TV gestionados.

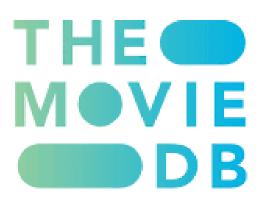
Dataset

• Total de entradas: 8,809

• Columnas:

- • show_id: Identificador único para cada título.
- • type: Categoría de título (Serie o película).
- • title: Nombre de la película o serie.
- • director: Director(es) del título. Puede contener valores nulos.
- • Casting: Lista de actores/actrices principales. Puede estar incompleto.
- • Country: País o países de producción.
- • date_added: Fecha en que se agregó el título a Netflix.
- • release_year: Año de lanzamiento original del título.
- • Rating: Clasificación por edades del título.
- • duration: Duración en minutos (películas) o temporadas (programas de televisión).
- • listaded in: Géneros del título.
- • description: Breve resumen del título.

Api de TMDb



The Movie Database (TMDb) es un servicio comunitario en línea que ofrece una amplia base de datos de información sobre películas, programas de televisión, y actores. La API de TMDb permite acceder a esta vasta cantidad de información de manera programática, facilitando la integración de datos de entretenimiento en aplicaciones y servicios.

Api externa Datos

ID de la película: Un identificador único para cada película.

Título: El título oficial de la película.

Descripción/Resumen: Un breve resumen del argumento de la película. Fecha de estreno: La fecha en que la película fue lanzada en cines o en otros medios. Género: Los géneros asociados con la película (por ejemplo, acción, aventura, comedia, etc.).

Duración: La duración total de la película en minutos.

Idioma original: El idioma en que fue originalmente producida la película.

Presupuesto: El presupuesto de producción de la película.

Ingresos: Los ingresos generados por la película.

País de origen: Los países donde fue producida la película.

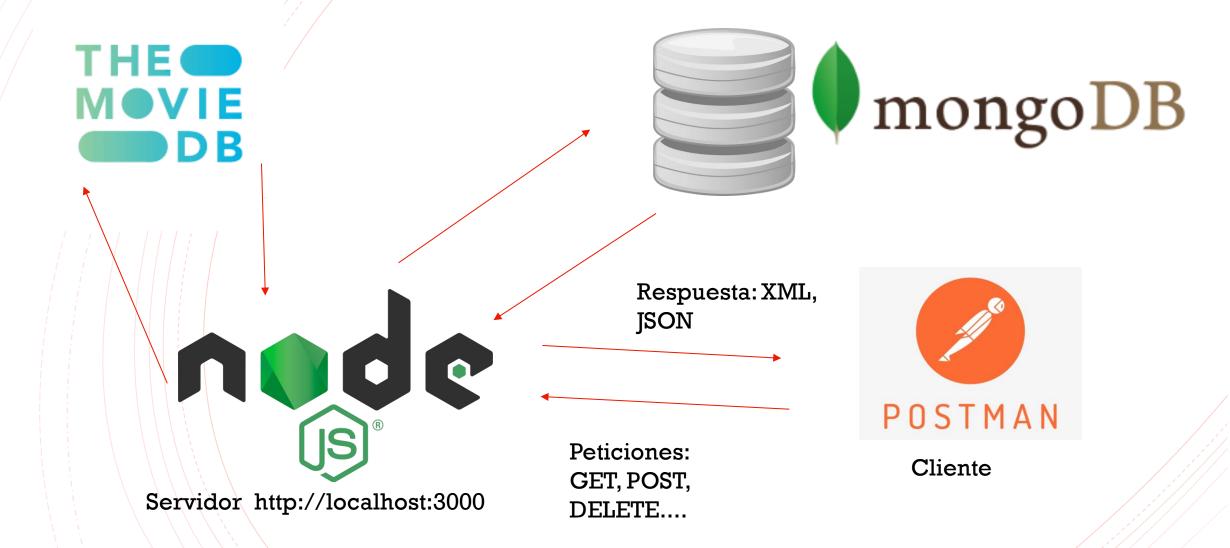
Calificación:

Clasificación de edad o de contenido de la película.

Póster: URL del póster oficial de la película.

Compañías productoras: Listado de las compañías que produjeron la película. Puntuación de los usuarios (Rating): La calificación promedio otorgada por los usuarios de TMDB.

Arquitectura de la api



Petición HTTP	URL	Descripcíon	Ejemplo
GET	/api/movies	Obtiene una lista de todas las películas.	[{"show_id": "550", "title": "Fight Club", "director": "David Fincher", "rating": "R", " duration": "139 min", " description": "An insomniac office worker and a devil-may-care soapmaker form an underground fight club."}]
POST	/api/movies	Añade una nueva película.	{"show_id": "551", "title": "Pulp Fiction", "director": "Quentin Tarantino", "rating": "R" "duration": "154 min", "description": "The lives of two mob hitmen, a boxer, a gangster's wife, and a pair of diner bandits intertwine in four tales of violence and redemption."}
PUT	/api/movies/{movie_id}	Actualiza una película existente.	{"message": "Movie updated successfully"}
DELETE	/api/movies/{movie_id}	Elimina una película.	{"message": "Movie deleted successfully"}
GET	/api/movies/{movie_id}	Obtiene los detalles de una película específica.	••••
	/api/mo	vies	

Petición HTTP	URL	Descripcíon	Ejemplo
GET	<pre>/api/movies/genres/{genre }</pre>	Obtiene una lista de películas por género.	[{"show_id": "550", "title": "Fight Club", "genre": "Drama", "director": "David Fincher"}]
GET	/api/movies/year/{year}	Obtiene una lista de películas por año de lanzamiento.	{[{"show_id": "550", "title": "Fight Club", "release_year": 1999, "director": "David Fincher"}]
GET	/api/movies/tmdb/{string}	Obtiene una lista de películas de tmdb que contenga el string.	[{"show_id": "550", "title": "Fight Club", "director": "David Fincher", "rating": "R", " duration": "139 min", " description": "An insomniac office worker and a devil-may- care soapmaker form an underground fight cl b."}]
GET	/api/movues/tmbd/{movie Name}?format=xml	Obtiene peliculas por nombre de la api tmbd en formato xml	<pre><?xml version="1.0" encoding="UTF-8"?> <movies></movies></pre>
	/api/mo	vies	

Petición HTTP	URL	Descripcion	Ejemplo
GET	/api/shows	Obtiene una lista de todas las series.	[{"show_id": "550", "title": "Fight Club", "director": "David Fincher", "rating": "R", " duration": "139 min", " description": "An insomniac office worker and a devil-may-care soapmaker form an underground fight club."}]
POST	/api/shows	Añade una nueva serie.	{"show_id": "551", "title": "Pulp Fiction", "director": "Quentin Tarantino", "rating": "R" "duration": "154 min", "description": "The lives of two mob hitmen, a boxer, a gangster's wife, and a pair of diner bandits intertwine in four tales of violence and redemption."}
PUT	/api/shows/{show_id}	Actualiza una serie existente.	{"message": "Show updated successfully"}
DELETE	/api/shows/{show_id}	Elimina una serie.	{"message": "Show deleted successfully"}
GET	/api/shows/{show_id}	Obtiene los detalles de una serie específica.	
	/api/snc	DWS / / /	

Petición HTTP	URL	Descripcíon	Ejemplo
POST	/api/reviews/{movie_id}	Añade una nueva reseña a una película.	{"message": "Review adde d successfully"}
POST	/api/reviews/{show_id}	Añade una nueva reseña a una serie.	{"message": "Review adde d successfully"}
GET	/movies?page=number&pa geSize=number	Obtener una lista paginada de películas.	{ "currentPage": 2, "pageSize": 5, "totalPages": 20, "totalResults": 100, "movies": [{ "id": 6, "title": "Inception", "releaseDate": "2010-07-16", "rating": "8.8" }, { "id": 7, "title": "The Matrix", "releaseDate": "1999-03-31", "rating": "8.7" },
	/api/revi	ews	

Instrucciones de uso



Instalar dependencias: npm install



Cargar los datos en la base de datos con el Script db.js



Iniciar el servidor : npm start



Conectarse al servidor: http://localhost:3010



Realizamos la consultas: GET, POST, DELETE....

