

# RECUPERACIÓN DE INFORMACIÓN Y RECOMENDACIONES EN LA WEB

CURSO 2023

GRUPO 10

---

## *Fing Watch*

---

*Autores:*

Agustín Núñez  
Roman Alvarez  
Mathias Ramilo  
Nicolas Adrien

*Supervisora:*

Libertad Tansini

November 22, 2023

# Contents

<b>1</b>	<b>Introducción</b>	<b>2</b>
<b>2</b>	<b>FingWatch</b>	<b>2</b>
<b>3</b>	<b>Arquitectura</b>	<b>4</b>
3.1	Scrapping . . . . .	5
3.2	Motor de Búsqueda . . . . .	6
3.3	Sistema de Recomendaciones . . . . .	6
3.4	Media . . . . .	8
<b>4</b>	<b>Trabajo a Futuro</b>	<b>9</b>
<b>5</b>	<b>Conclusión</b>	<b>9</b>

# 1 Introducción

El mundo del entretenimiento cinematográfico ofrece una inmensa cantidad de opciones, lo que puede ser abrumador para los amantes del cine que buscan encontrar películas que se adapten a sus gustos personales. Además, la falta de orientación a menudo conduce a una experiencia de visualización insatisfactoria, donde los usuarios pueden perder tiempo y dinero en películas que no les interesan realmente.

La sobreabundancia de películas, géneros, actores y directores, junto con la falta de un sistema de recomendación efectivo, es un problema que enfrentan tanto los cinéfilos casuales como los apasionados. Además, los algoritmos de recomendación tradicionales a menudo tienen dificultades para comprender las preferencias cambiantes y únicas de cada usuario, lo que limita su capacidad para brindar recomendaciones precisas y personalizadas.

## 2 FingWatch

Nuestra solución es una aplicación web de recomendación de películas que utiliza un enfoque personalizado para ayudar a los usuarios a descubrir películas que realmente les gusten. Aquí están los aspectos clave de nuestra solución:

- **Sistema de Recomendación:** Basándonos en las preferencias de los usuarios, vemos la necesidad de crear un sistema de recomendaciones basado en películas populares y recientes. Además de ofrecer recomendaciones basadas en contenido, basadas en características del usuario y basadas en el historial del usuario. Logrando generar recomendaciones precisas y personalizadas para nuestros usuarios como se aprecia en la figura 1.
- **Base de Datos Actualizada:** Uno de los objetivos fundamentales consiste en desarrollar un mecanismo que permita obtener en tiempo real la información correspondiente a los nuevos lanzamientos. De esta manera, conseguiremos mantener actualizada nuestra base de datos con las películas más recientes.
- **Actualizaciones en Tiempo Real:** Nuestra plataforma recopila información sobre los *feedbacks* de los usuarios, permitiéndole al usuario modificar dichos

*feedbacks* en cualquier instante a través de su lista de favoritos. Esto nos permite adaptar continuamente nuestras recomendaciones a medida que los gustos del usuario evolucionan.

- Exploración de Contenido: Además de las recomendaciones personalizadas, ofrecemos a los usuarios la capacidad de explorar películas ya sea por géneros o por similitudes entre una película determinada.
- Disponibilidad en Plataformas: Junto con la información de una película nuestra plataforma mostrará a los usuarios en qué plataformas de streaming se encuentra disponible para mirarla.
- Tolerancia a Errores: Al momento de realizar la búsqueda una película determinada, se aceptara cierto nivel de tolerancia a errores al momento de procesar la misma.

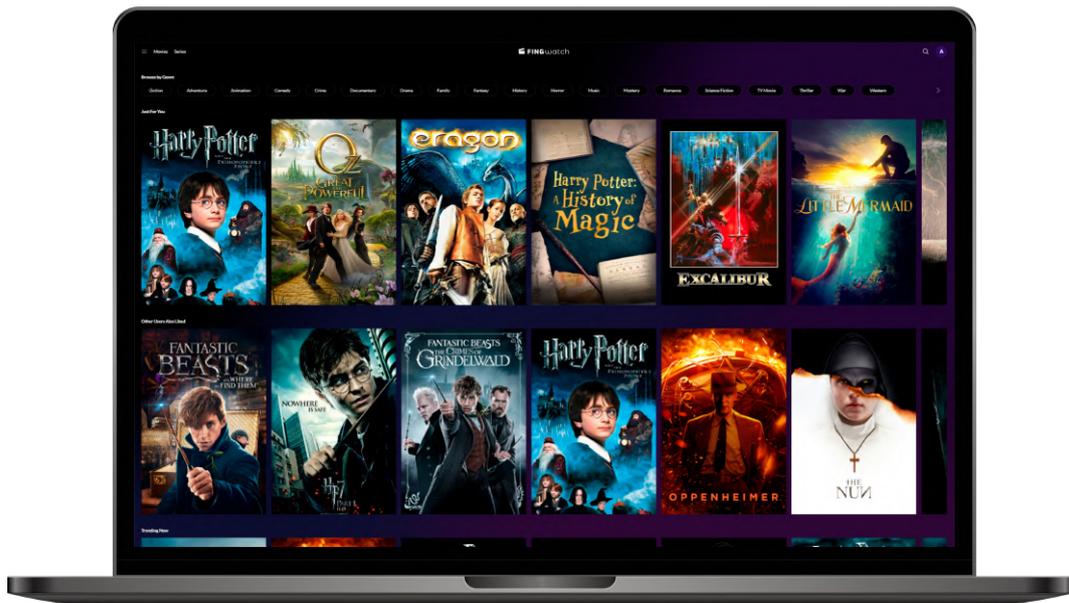
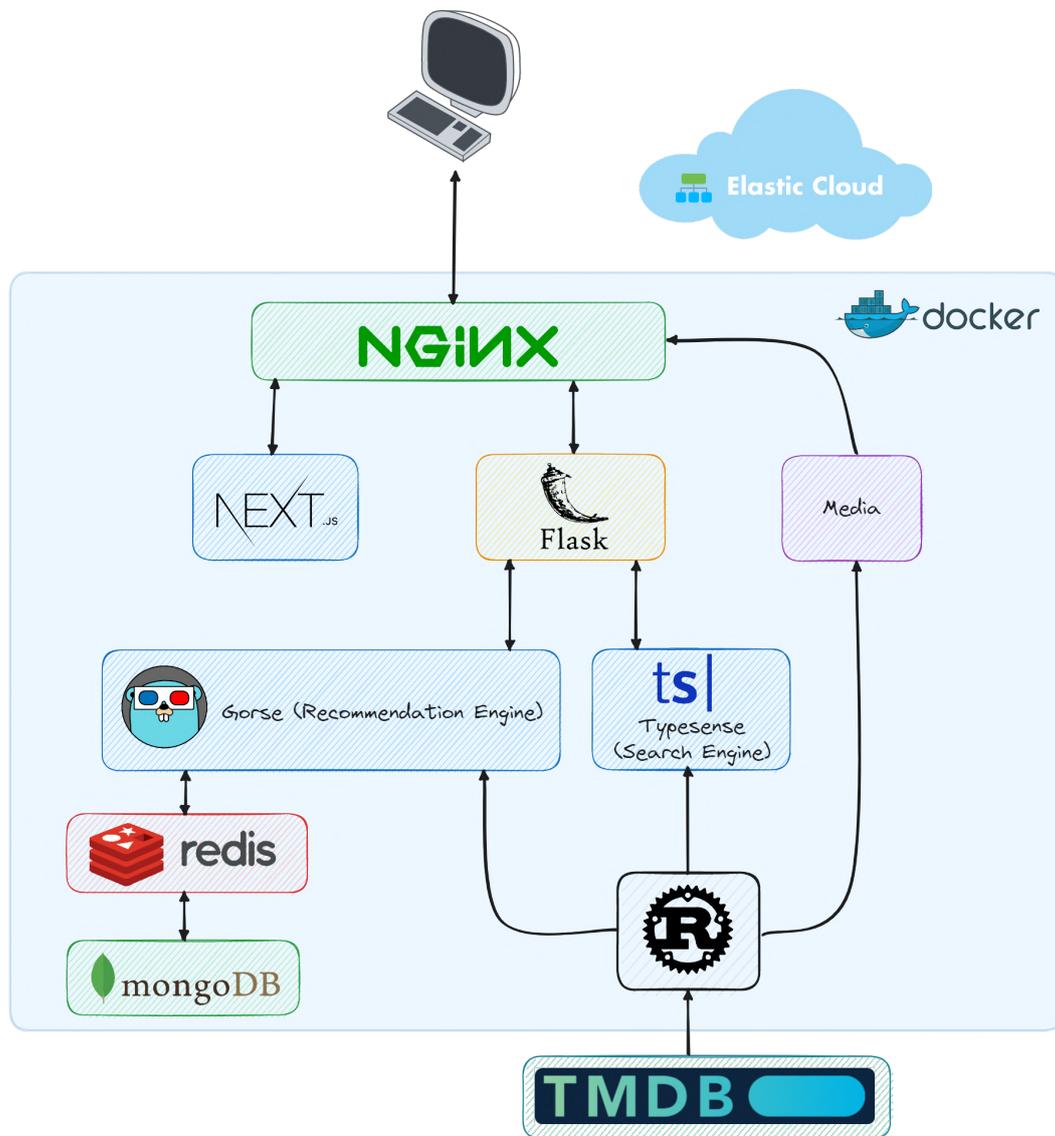


Figure 1: Recomendaciones para un usuario en específico.

### 3 Arquitectura



Para el despliegue de este sistema utilizamos principalmente la tecnología **Docker**<sup>1</sup> junto a los servicios de hosting brindados por la nube de Antel **Elastic Cloud**<sup>2</sup>. De esta forma, el cliente final accede mediante un servidor **Nginx** a los distintos servicios que ofrece la plataforma.

<sup>1</sup>Docker Engine: <https://www.docker.com/>

<sup>2</sup>Mi Nube ANTEL: <https://minubeantel.uy/>

## 3.1 Scrapping

En primer lugar, hacemos uso de TMDB API<sup>3</sup> para tener acceso a la metadata que utilizaremos de las películas. Una de las ventajas de esta API es que además de brindar información básica sobre las películas, también brinda otra información de interés como es el caso de las plataformas de streaming en las que se encuentra disponible cierta película en la actualidad. Además, TMDB ofrece la posibilidad de obtener las *'keywords'* asociadas a una película en específico, lo que será de utilidad más adelante (3.3).

Gracias a lo anteriormente mencionado, se implementa un servicio de scrap-ping que se ejecuta en su propio contenedor docker y su principal funcionalidad es consumir la API de TMDB para obtener ciertos campos que son de interés así como las imágenes asociadas al film. Además, este servicio se encuentra siempre en ejecución, lo que nos permite consumir y procesar películas a medida que son agregadas en la API de TMDB. Entre los campos de interés destacamos los siguientes:

- `adult: bool`
- `genres: []String`
- `id: String`
- `keywords: []String`
- `original_language: String`
- `overview: String`
- `release_date: String`
- `runtime: u32`
- `status: String`
- `title: String`
- `vote_average: f64`
- `providers: []String`

---

<sup>3</sup>The Movie Database API: <https://developer.themoviedb.org/>

## Problema con NULLs

En este punto nos encontramos con el problema de que no todas las películas que retorna dicha API contienen los campos esperados. Por ende, se decidió descartar aquellas películas que no puedan ser deserializadas utilizando el schema propuesto.

## 3.2 Motor de Búsqueda

A medida que las películas son procesadas por el servicio de scrapping (3.1), estas son agregadas a una colección `movies` dentro del motor de búsqueda `typesense`<sup>4</sup>.

Esto nos permite hacer consultas sobre la colección `movies` aceptando cierta tolerancia a errores de tipeo así como también asignarle pesos a los distintos campos al momento de realizar la búsqueda para luego ser retornados al usuario final en un orden según el *matching score* de la *query* realizada. Por ejemplo, al momento de realizar la *query* 'Hary Potter' se espera que se muestren los resultados deseados por el usuario como se muestra en la figura 2. *Notar que además de contener errores de tipeo, no es necesario especificar el título completo de la cinta para que se muestre el resultado 'Harry Potter and the Philosopher's Stone'.*

## Mejoras a Largo Plazo

Cabe destacar que `typesense` ofrece una amplia cantidad de servicios además de los anteriormente mencionados, lo que pueden ser de gran utilidad en futuras actualizaciones de la plataforma como se menciona en (4).

## 3.3 Sistema de Recomendaciones

A medida que las películas son procesadas por el servicio de scrapping (3.1), estas son agregadas también al sistema de recomendaciones `Gorse`<sup>5</sup>. Los únicos campos que se utilizan en este punto son 'id' para la identificación de la película y 'keywords' que son utilizadas para "describir" las características de la película dentro del sistema de recomendaciones. A modo de ejemplo, la película *Harry Potter and the Philosopher's Stone* contiene las siguientes `keywords`:

---

<sup>4</sup>Typesense Search Engine: <https://typesense.org/>

<sup>5</sup>Gorse Recommender System Service: <https://gorse.io/>



Figure 2: Resultados retornados al procesar la *query* 'HARY POTER'.

```
"keywords": [  
  "witch",  
  "school friend",  
  "friendship",  
  "london, england",  
  "based on novel or book",  
  "magic",  
  "boarding school",  
  "child hero",  
  "school of witchcraft",  
  "chosen one",  
  "school",  
  "shopping",  
  "fantasy world",  
  "wizard",  
  "christmas",  
  "based on young adult novel",
```

```
"owl"  
]
```

Gracias a estos campos **Gorse** es capaz de obtener las películas "vecinas" de una película dada como se muestra en la figura 3 al igual que generar recomendaciones a los usuarios ya sean 'Content Based', 'Collaborative Filtering' o 'User Based' como se muestra en la figura 1. Mencionar que la diferencia entre estos dos últimos tipos de recomendaciones es que **Collaborative Filtering** se basa en el historial de *feedbacks* de los usuario mientras que **User Based** se basa en características del usuario (por ejemplo ubicación actual o edad).

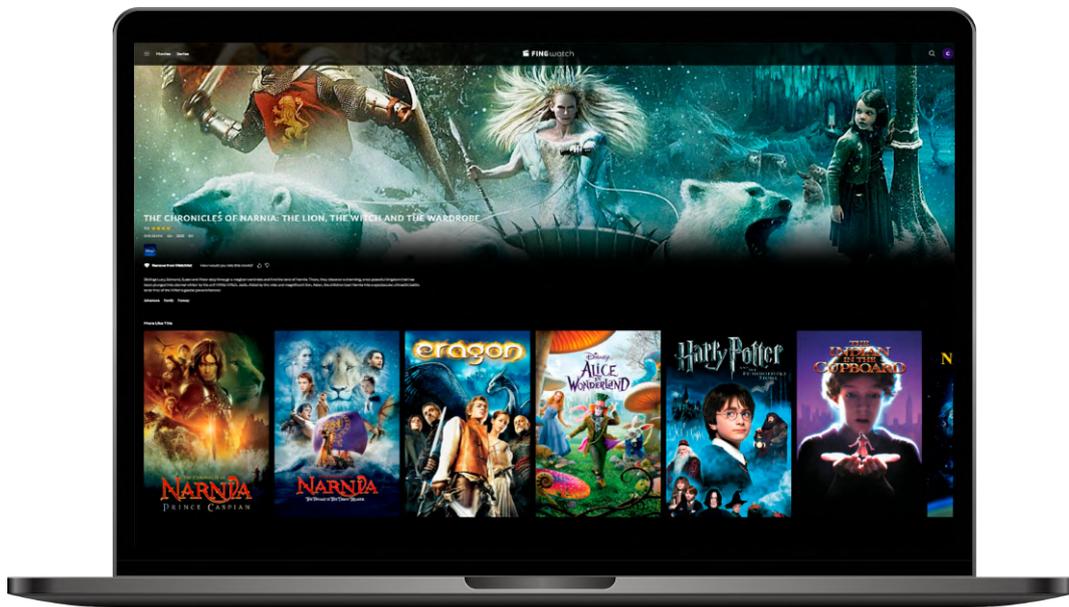


Figure 3: Pagina principal para el film 'THE CHRONICLES OF NARNIA: THE LION, THE WITCH AND THE WARDROBE'.

Por otra parte, **Gorse** ofrece una amplia configuración como es el caso de agregar un campo TTL para los *feedbacks* realizados por los usuarios.

### 3.4 Media

Se utiliza un volumen 'images' para almacenar las imágenes (posters y banners) de las películas procesadas. Luego, estas son consumidas por el cliente mediante

un servidor web que brinda acceso a dicho volumen. *Al día de hoy 21/11/2023 este volumen cuenta con 9.26GiB correspondientes a 11.219 películas procesadas y sigue en crecimiento.*

## 4 Trabajo a Futuro

Como se menciona en el documento, varias de las herramientas usadas poseen una gran variedad de funcionalidades que pueden ser útil en futuras actualizaciones de la plataforma. Por ejemplo, al momento de consumir los datos de la API de TMDB, se podría consultar por otros datos de utilidad como es el caso del elenco participante de la película. Esto nos permitiría tener en cuenta esta información al momento de procesar búsquedas. Además, el sistema de búsqueda podría pulirse agregando un sistema de filtro intuitivo para el usuario.

Por otro lado, otro punto que creemos importante agregar a futuro es la transparencia de las recomendaciones ofrecidas. Ya que se debería proporcionar a los usuarios información sobre cómo se generan las recomendaciones y cómo funcionan los algoritmos que utiliza la plataforma, de esta forma, se brinda transparencia y confianza sobre el funcionamiento y en como se utilizan realmente los datos que aportan los usuarios.

Por último pero no menos importante, creemos que la integración social podía ser un gran aporte en la plataforma incorporando funciones que permitan a los usuarios compartir sus listas de favoritos y recomendaciones con amigos. Además, se podría implementar un sistemas de comentarios para fomentar la interacción entre usuarios.

## 5 Conclusión

El desarrollo de la plataforma *finwatch* resultó sumamente enriquecedora, ya que logramos satisfactoriamente alcanzar los objetivos planteados inicialmente. Durante el proceso, exploramos y aplicamos una variedad de tecnologías interesantes, entre las que se destacan el motor de búsqueda **Typesense** y el sistema de recomendaciones **Gorse**. Estas herramientas no solo nos permitieron implementar de manera efectiva los conceptos aprendidos a lo largo del curso, sino que también

enriquecieron significativamente nuestra comprensión y nuestra visión sobre los temas abordados en el curso.