

No More Boring Days



WEBIR

25/11/2022

Grupo 11

Martina Font	4942128
Cecilia Toledo	4856277
Facundo Torterola	5592045
Valentina Sediey	5003801

Docente:
Libertad Tansini

Introducción

Hoy en día, existe una gran variedad de películas y de juegos en la web. Nuevas películas y juegos se estrenan constantemente y debido a que existe una gran cantidad de ambos, elegir qué película ver o qué juegos jugar en momentos de ocio, puede ser una tarea que lleve más tiempo del que quisiéramos. Es por esto que surgen los sistemas de recomendación. Nuestro proyecto “No more boring days” consiste en un programa recomendador de juegos y películas, a partir de información obtenida de películas y juegos previamente puntuados por otros usuarios.

Motivación y descripción del problema

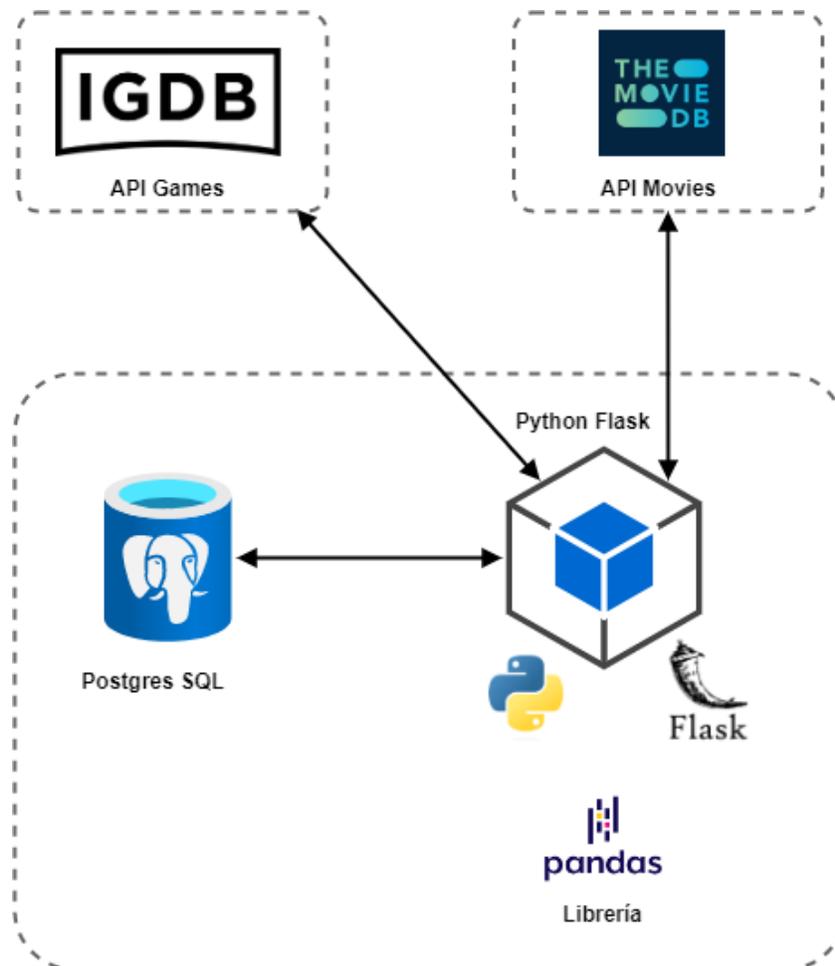
La motivación de este proyecto es crear un programa recomendador de juegos y películas basado en un filtrado colaborativo que utiliza como información la puntuación de películas y de juegos previamente puntuados por otros usuarios. Para desarrollar esto, el filtrado colaborativo utiliza dos fuentes de datos: un conjunto de juegos obtenidos de una API de juegos, “Internet Game Database” (IGDB)[1] y también películas y puntuaciones obtenidas de la API de “The movie database” (TMDB)[2]. La información de películas y juegos puntuadas por otros usuarios, se utilizará para luego recomendar juegos y películas.

Enfoque de la solución

Para solucionar el problema planteado, se debió establecer una conexión con la API de TMDB, e implementar un inicio de sesión para que los usuarios entren a su cuenta de TMDB, para así poder obtener sus puntuaciones de películas. También se obtuvieron los 50 juegos mejores puntuados, utilizando la API de IGDB (API de juegos).

Con la API de TMDB, podemos obtener una lista de películas con sus respectivas puntuaciones para cada usuario y de esta manera podemos guardar una correspondencia de usuario/película/puntuación para así aplicar el algoritmo de filtrado colaborativo. Luego, se muestra una lista de juegos al azar para que el usuario puntúe y poder establecer posteriormente una relación con otros usuarios (ya que no se encontró una plataforma de puntuación de juegos que funcione de igual manera que TMDB). El algoritmo de filtrado implementado, se explica más adelante en la sección “Filtrado Colaborativo basado en usuarios”.

Diseño e implementación



El sistema desarrollado cuenta con una interfaz web donde el usuario primeramente debe registrarse.

Luego de esto, se le muestra una pantalla donde indicará su usuario de la plataforma de películas. Esto permite que la base de datos se cargue con información de las preferencias del usuario que inició sesión. Se guardará la lista de películas y su calificación del 1 al 5.

Seguidamente, se le brinda una lista de juegos randómica a la cual deberá calificar del 1 al 5 (esto es para que coincida con los valores que traen las películas y evitar tener que normalizar las calificaciones) y se guardará esta información al igual que con las películas. Esta lista de juegos, se obtiene de la base de datos, donde inicialmente se cargaron los 50 más populares según la plataforma de juegos.

Una vez que se obtienen las preferencias de las dos temáticas, se le brinda al usuario una recomendación calculada en base al filtrado colaborativo de usuarios.

Herramientas utilizadas

- Python Flask

Framework Web para Python utilizado para la implementación de lógica del negocio y para las vistas del sistema junto con templating de html.

- Postgresql

Base de datos relacional donde se almacenan los usuarios, los juegos, los ratings de los juegos y los ratings de las películas.

- Pandas

Librería de Python utilizada para manipular los datos y realizar análisis de los mismos.

- IGDB

API de juegos. (1)

- TMDB

API de películas. (2)

Filtrado Colaborativo basado en usuarios

Para la implementación del filtrado colaborativo se utilizó Pandas con el fin de unificar la información en un dataset y calcular la matriz para luego encontrar las similitudes.

Primero se crearon los dataset de películas y de juegos, los cuales contienen las columnas del identificador del usuario, nombre del juego o película y rating.

Para poder concatenar estas dos tablas, se agregó al final de cada nombre una string “__game” o “__movie” para poder identificar de qué tabla proviene cada elemento.

Una vez se concatena, se obtiene un dataset el cual se transforma en una matriz, donde las columnas son las películas y juegos y las filas son los usuarios. El valor de la matriz es el rating, en caso de no tener esa evaluación se ingresa el valor 'NaN'.

Dado que algunas personas cuando dan una calificación tienden a clasificar más alto que otras, es que se normaliza la matriz en base a la calificación promedio de cada usuario. Luego de esto se obtiene una matriz donde, los elementos con un rating menor al promedio obtienen un valor negativo y los que son superior al promedio obtienen un valor positivo.

Para proseguir, se calcula la similitud entre los usuarios, aplicando el Coeficiente de correlación de Pearson (“índice que puede utilizarse para medir el grado de relación de dos variables siempre y cuando ambas sean cuantitativas y continuas” (3)). A partir de este punto se obtiene una matriz con índices de relación entre usuarios. Como lo que se busca son los similares al

usuario que inició sesión, se extrae la matriz de los similares a este. Esta matriz contiene valores de -1 a 1, donde -1 significa que tienen gustos opuestos y 1 que tienen la misma preferencia.

Para calcular los juegos y películas a recomendar, se optó por realizarlo en base al usuario con mayor similitud de la matriz calculada anteriormente. Primero, a partir de la primera matriz calculada, se quitan aquellos juegos y películas que el usuario ya calificó y por tanto ya los jugó o las miró. Luego, extraemos todos los usuarios que no sean el que se eligió con mayor similitud y además se quitan todos los elementos que tienen valor NaN, es decir que no los ha calificado. Para realizar la recomendación, se recorre la matriz y se ordenan los elementos según el score. Se decidió obtener de ser posible los primeros 10 y separar según películas o juegos para desplegarlos luego en la pantalla.

Funcionalidades y uso

A continuación se muestran algunas imágenes del programa desarrollado: inicio de sesión y registro, inicio de sesión con la cuenta de TMDB (para obtener las puntuaciones de películas). También se muestran la pantalla donde el usuario debe puntuar juegos y finalmente la recomendación arrojada por el sistema.

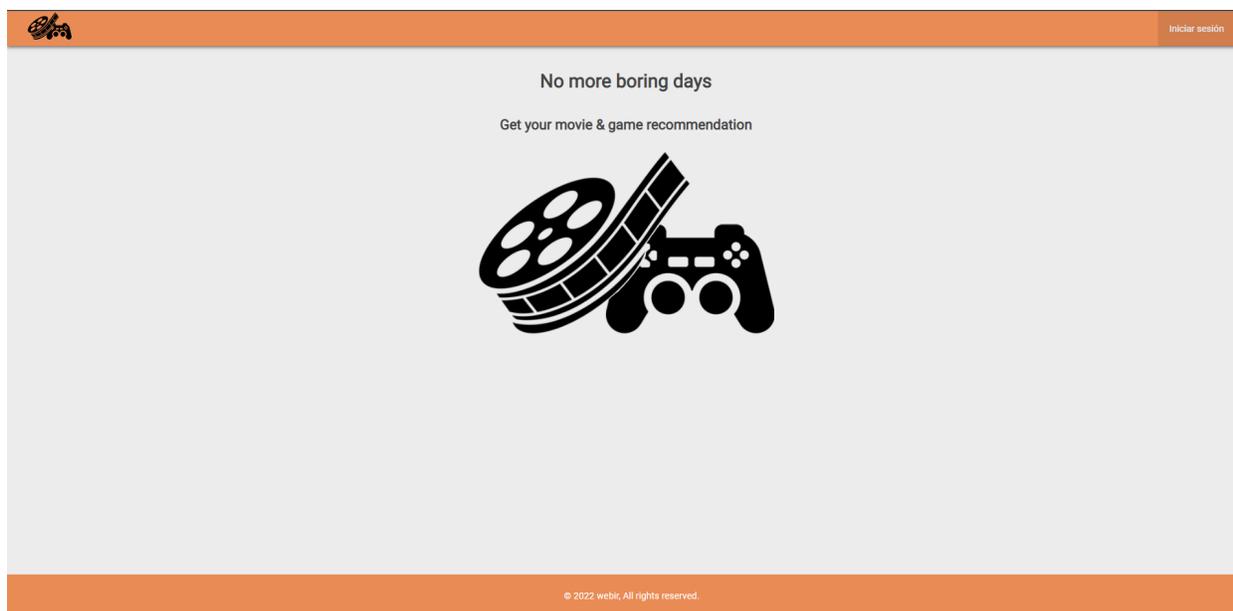
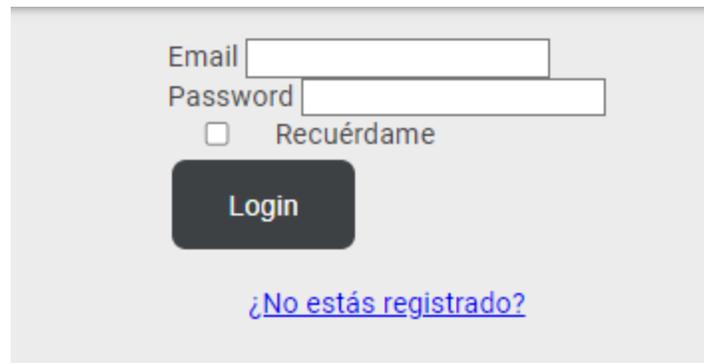


Fig 1. Pantalla principal de la web (home)



Email

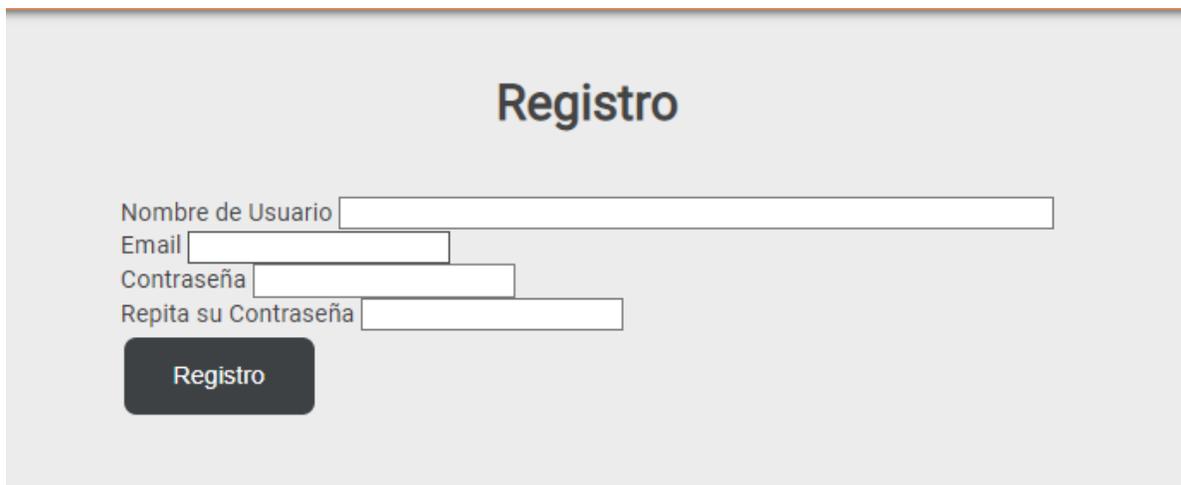
Password

Recuérdame

Login

[¿No estás registrado?](#)

Fig 2. Formulario de Log In



Registro

Nombre de Usuario

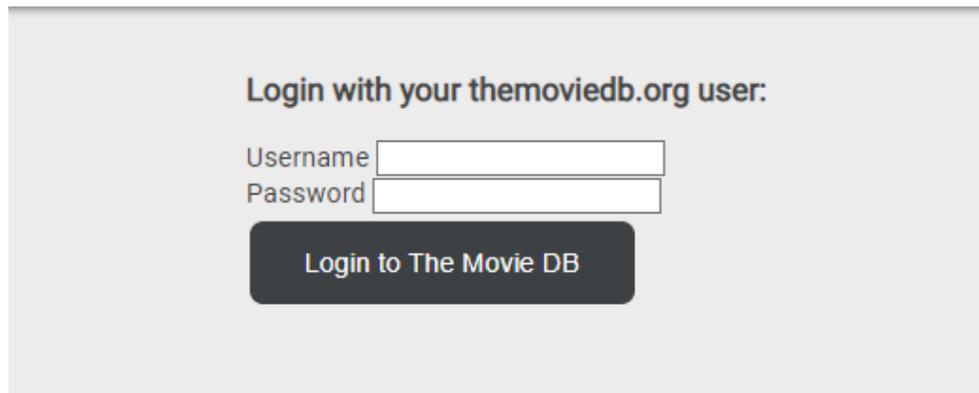
Email

Contraseña

Repita su Contraseña

Registro

Fig 3. Formulario de Registro



Login with your themoviedb.org user:

Username

Password

Login to The Movie DB

Fig 4. Formulario de Log In a TMDB

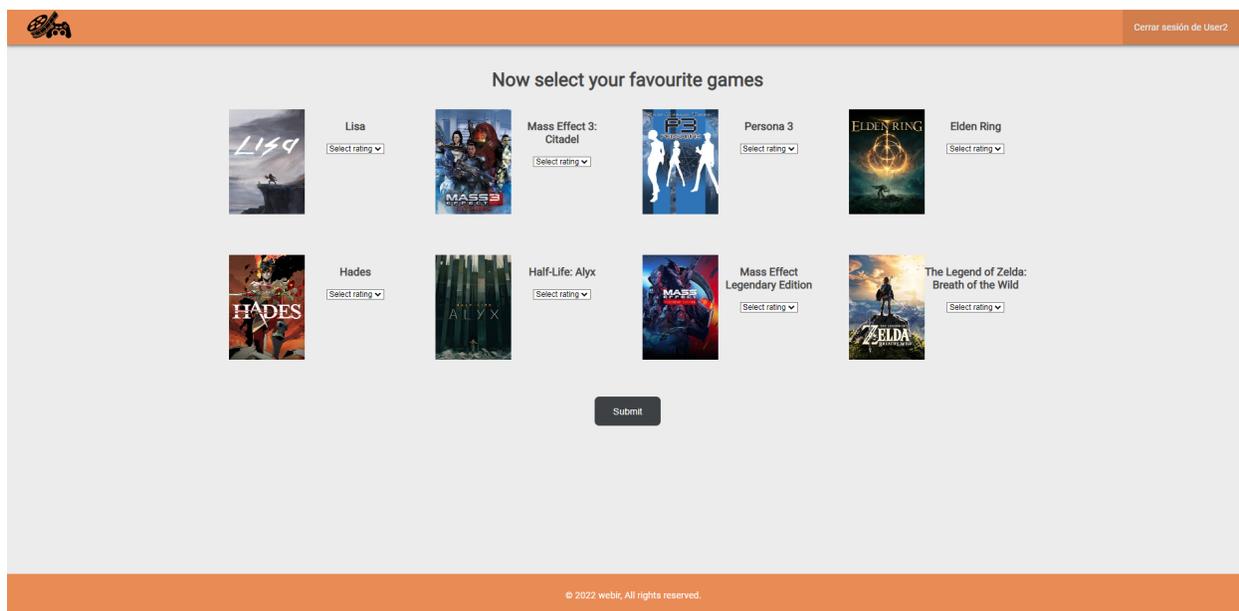


Fig 5. Pantalla para que el usuario puntúe juegos del 1 al 5

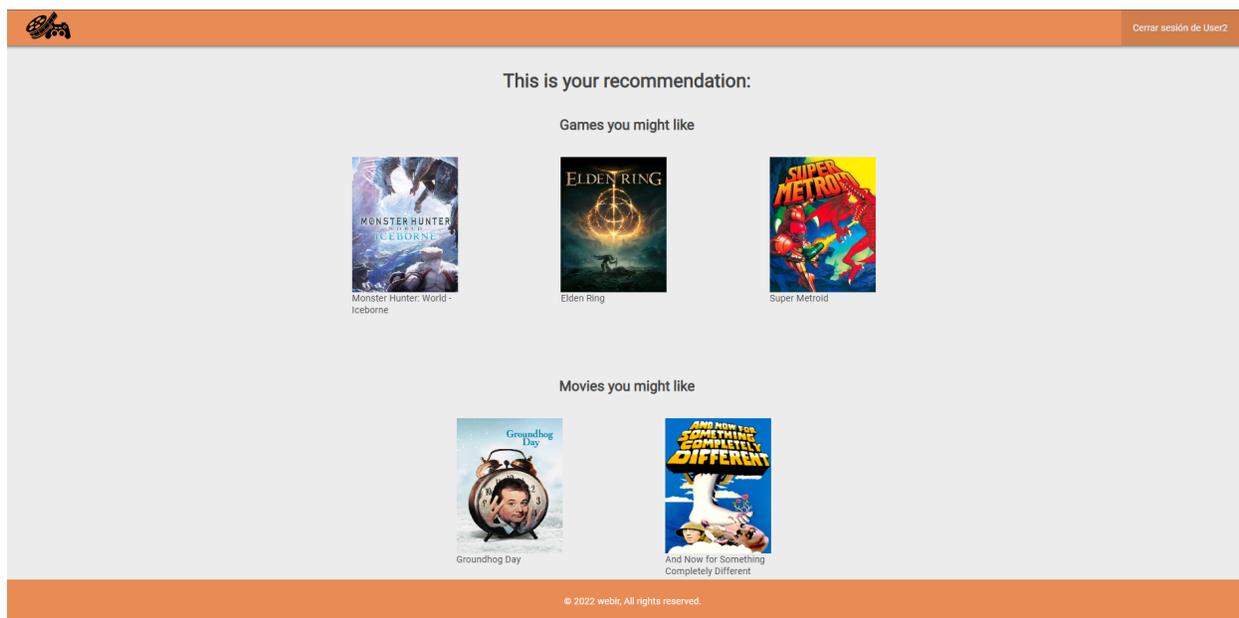


Fig 6. Pantalla final con recomendaciones de juegos y películas

Evaluación y resultados

Para evaluar el sistema registramos una variedad de usuarios de prueba en TMDB, con los cuales puntuamos películas, tratando de que algunas de estas puntuaciones se repitieran entre varios usuarios, manteniendo la idea de que cada usuario tuviera alguna puntuación particular distinta.

Luego, simulamos manualmente el uso del recomendador con dichos usuarios. Accedimos con un usuario A al sistema de recomendación y puntuamos juegos. A continuación ingresamos con un usuario B que puntuara juegos de forma parecida, y obtuvimos recomendaciones coherentes. Este proceso lo repetimos, cambiando algunas variables en cada caso, concluyendo que los resultados son coherentes con los gustos de los usuarios creados.

Conclusiones

Implementar un filtrado colaborativo fue muy valioso para entender su funcionamiento con mayor profundidad. Asimismo, concluimos que a mayor cantidad de usuarios usando el sistema, mejores serán los resultados arrojados por este. El filtrado colaborativo basado en usuarios es una estrategia que resulta muy útil hoy en día, ya que hay muchísimo contenido disponible, que puede ser muy difícil de manejar para los usuarios. Es por esto, que es de gran utilidad construir un sistema de “inteligencia colectiva” que ayude a facilitar el proceso de elección ante la creciente oferta de entretenimiento.

Trabajo Futuro

Algunas posibles mejoras para tener en cuenta a futuro podrían ser:

- Al momento de puntuar juegos se lista una cantidad de juegos al azar: es posible que un usuario no conozca un juego de los que se muestran, podría agregarse un botón para cambiar de juego, para así obtener un juego nuevo para puntuar y finalmente lograr una mayor exactitud.
- Se podría agregar al filtrado, el género de las películas y juegos para obtener recomendaciones en base a ítems similares, además de usuarios similares.
- Se podría agregar el concepto de “amigos” dentro de la aplicación y modificar el algoritmo de modo de que los gustos de los amigos de un usuario influyan con más fuerza en sus recomendaciones.

Referencias

1. "IGDB API",

<https://api-docs.igdb.com/#about>.

2. "The Movie Database API."

<https://developers.themoviedb.org/3/getting-started/introduction>.

3. "Coeficiente de correlación de Pearson." *Wikipedia*,

https://es.wikipedia.org/wiki/Coeficiente_de_correlaci%C3%B3n_de_Pearson.