

# **AnimeSho**

Realizado por:

- Vicente Cambrón Tocados 20622913D

# Objetivos

La finalidad del proyecto es proporcionar un sitio web donde se encuentre una colección de los 1000 animes tops de MyAnimeList, que se encuentran en este [enlace](#). El sitio web nos proporciona una interfaz con la que podemos visualizar los animes, sus detalles, añadirlos a nuestra lista de 'Vistos', hacer búsquedas; y además nos proporciona recomendaciones personalizadas.

## Proyecto, herramientas y su uso

En el desarrollo del proyecto se han usado principalmente las siguientes herramientas: Django, BeautifulSoup, Whoosh y un sistema de recomendaciones basado en contenido.

### Django

El proyecto está realizado sobre Django, proporcionándonos un sitio web sobre Python. Accediendo a la ruta [raíz](#) podemos observar como se nos listan todos los animes paginados de 36 en 36.

En el menú superior nos encontramos con el nombre del sitio web, AnimeSho, que haciendo click nos lleva a la página principal de la web. Si estuviéramos logueados nos encontraríamos al lado un botón que nos permite cargar los animes. Seguidamente, en la página principal, nos encontramos con un menú desplegable para ordenar los animes, de forma ascendente y descendente, por ranking, popularidad, puntuación y número de episodios. Además, nos encontramos con un botón que nos lleva a otra página para realizar búsquedas algo más complejas y un input para realizar cualquier búsqueda, estos dos últimos casos los veremos más adelante. Por último, si no estás logueado aparecerán dos botones para loguearse o registrarse que abren un modal con un formulario dependiendo de a cuál le demos.

Si seguimos en la página principal, antes del listado de los animes nos encontramos con los filtros que se pueden aplicar sobre el listado de los animes, por género, estudio, tipo o estado.

En listado de los animes si hacemos click sobre alguna imagen, entramos en la vista de detalles del mismo donde se nos muestra toda la información del mismo. También, nos proporciona enlaces para ordenar por ranking y popularidad y filtrar por género y estudio. Por último, esta ventana también vemos que nos recomienda animes relacionados. Si estuviéramos logueados nos muestra si lo hemos visto o no, y la posibilidad de añadirlo/quitarlo de la lista de vistos.

## Beautifulsoup

He realizado el scraping de los 1000 animes tops de MyAnimeList ([enlace](#)).

Para esto, inicialmente, he obtenido el link de la vista de detalle de los 1000 animes de las imágenes del listado. Después he ido por cada link recuperando la información necesaria; como esta tarea consumía mucho tiempo, gracias a librería 'multiprocessing' y sus 'Pool' he conseguido paralelizar este proceso y reducir considerablemente los tiempos. De entre todas las pruebas que he realizado para encontrar la configuración óptima, he paralelizado 'multiprocessing.cpu\_count()\*8' procesos, aunque en mi caso me ha ido bien hasta multiplicando por 16 en vez de por 8.

El proceso del scraping, gracias a la paralelización, no debería de superar los 2 o 3 minutos si se cuenta con un internet decente.

Problemas que pueden surgir (casos muy raros):

- Debido a la paralización en alguna ocasión puede dar error y no cargarse los animes. Me ha ocurrido solo un par de veces y no he conseguido encontrar ni el problema ni la solución, y debido a que la paralelización reduce bastante el tiempo del scraping he decidido mantenerla.
- Debido a que entramos algo más de 1000 veces al sitio de MyAnimeList, en ocasiones puede dar error el scraping. La solución sería entrar a [MyAnimeList](#), le indicará que ha entrado muchas veces a la página y que confirmes que no eres un bot. Una vez solucionado volver a iniciar la carga.

## Whoosh

La carga con Whoosh se realiza a la par del almacenamiento en la base de datos, y se guarda casi la misma información de cada anime en el índice.

Como he explicado en el apartado de [Django](#), en la navbar nos encontramos con un enlace para hacer búsquedas algo más personalizadas y un buscador más generalizado.

Para la búsqueda generalizada se hace uso del 'MultifieldParser' de la librería whoosh para buscar en el título, sinopsis, estudios y géneros a la vez.

La búsqueda personaliza nos ofrece tres formas distintas de búsqueda: en la sinopsis, por fechas y por número de episodios. El primero, hace una búsqueda de palabras y frases en la sinopsis de los animes. En la búsqueda por fechas, podemos hacer una búsqueda a partir de la fecha de inicio, hasta la fecha de fin, o entre la fecha de inicio y fecha de fin. Por último, la búsqueda de número de episodios, se puede buscar los animes con mayor o menor número de episodios del indicado; si se ha seleccionado mayor, se ordenan de menor a mayor los animes resultantes, y si se ha seleccionado menor, se ordenan de mayor a menor.

## Sistema de recomendación

El sistema de recomendación está basado en contenido.

En la vista de detalle de un anime, como he comentado se recomienda animes según el anime que se muestra por su géneros haciendo uso del Coeficiente de Dice.

Si nos encontramos logueados, al final de la navbar nos encontramos con un menú con el nombre de nuestro usuario. En el menú tenemos dos links, uno nos lleva a nuestra lista de vistos y el otro a una lista de 36 animes recomendados según los animes que tengamos marcados como vistos

## Anotaciones extra

El proyecto ha sido realizado:

- SO: Pop!\_OS 22.04 LTS
- Python: 3.10.6 (virtualenv)
- IDE: PyCharm 2022.3.1 (Professional Edition)

No ha sido probado fuera de lo descrito.

El proyecto hace uso de alguna librería que no hemos visto en la asignatura, se deberían de instalar todas las encontradas en el archivo 'requirements.txt'.

Link github: <https://github.com/vicato2000/animeSho>

Link git (https): <https://github.com/vicato2000/animeSho.git>