

# MEMORIA DEL PROYECTO WEB YUMEANIME



Creado, diseñado y desplegado por **Raúl Fernández Arce**

2º Ciclo Formativo de Grado Superior

**Desarrollo de Aplicaciones Web**

PARA VER LA MEMORIA EN SU MÁXIMO ESPLENDOR:

<https://github.com/raulfarce01/yumeanime/blob/main/README.md>



## yumeanime

Repositorio para el proyecto del TFG, aquí se guardará toda la información, avances y ficheros para la resolución del proyecto.

---

1. [Nombre del Proyecto](#)
  2. [Investigación y Comparación del Proyecto con similares](#)
  3. [Explicación de las especificaciones del proyecto](#)
  4. [Planning del Proyecto](#)
  5. [Diseño del proyecto y sus páginas](#)
  6. [Esquemas y especificaciones de la base de datos](#)
  7. [Diagrama de Casos de Uso](#)
  8. [Anotaciones y Aclaraciones](#)
- 

### Nombre del Proyecto

Yumeanime

### Explicación del nombre escogido

El nombre procede de la unión de dos palabras:

- Yume (del japonés), que traducido al español significa "sueño".
- Anime (del japonés), que significa "animado".

### Investigación y Comparación del Mercado

- somoskudasai:  
Puntos a favor:
  - Está constantemente en actualización
  - Tiene un gran equipo de desarrollo que cada día escribe entre 3 y 5 noticias nuevas
  - Gran variedad de información sobre anime, cultura japones, etc...  
Puntos en contra:
  - No tiene una indexación que te permita buscar fácilmente las noticias antiguas
  - En la página principal solo aparecen las noticias más recientes, por lo que las antiguas las pierdes.
- animeflv:  
Puntos a favor:
  - Gran cantidad de animes
  - Gran comunidad

- Muchas formas de filtrar la búsqueda de animes
- Guardado de animes favoritos, en espera y viendo



Puntos en contra:

- No puedes crear tus propias listas de anime
- No puedes compartir una lista con varios animes
- Faltan algunos animes por añadir

- jkanime:

Puntos a favor:

- Bonita interfaz y atractiva
- Tiene todos los animes del mercado

Puntos en contra:

- Tarda un par de días en subir el capítulo
- No tiene página de noticias
- No puedes hacer lista personalizada de animes

## Explicación del Proyecto

Voy a crear una página principalmente de noticias y contenido anime, donde puedas investigar los animes del mercado y puedas crear tu propia lista personalizada de los animes que quieras almacenar en la misma, ya sea por su categoría, o por la atracción que te han creado, o por si sería un anime que recomendarías, etc... (opcionalmente también de las noticias). La página tendrá tres tipos de usuarios: "Usuario normal", es decir, el que se registra en la página, al que se le permite crear sus propias listas, comentar, dar "me gusta"; "Usuario sin registrar", al que no se le permite realizar ninguna acción exceptuando la de navegar por la aplicación, leer las páginas dedicadas de cada anime, y visualizar las listas de otros usuarios; "Usuario administrador", al que se le permite realizar cualquier acción en la página, como añadir animes y noticias. Opcionalmente, también intentaré crear una "mini" red social donde los otakus puedan interactuar entre ellos y compartir sus listas personalizadas, noticias, y hablar de otras cosas (versión extendida). En el caso de que sea rentable y tenga tiempo, seguiré desarrollando la página para que permita la visualización de los animes (versión extendida extendida).

## Planificación

Grant en el siguiente enlace:

<https://docs.google.com/spreadsheets/d/1-KXS5N--oKPiSEBPsgFjpQpuVVrh9BLd4Oyl7MlccDY/edit?usp=sharing>

---

## Diseño

En este apartado se recogerán todas las características del diseño, es decir, tipografía, color, y el diseño de las páginas será publicado al github una vez finalizado.



## Colores

Resalto -> #D4C859

Fondo -> #FAF8F7

Nav, cabecera y destacados -> #0A1940

Botones y contenedores -> #465987

## Fuentes

Título -> {

- Oleo Script Swash Caps :

[https://fonts.google.com/specimen/Oleo+Script+Swash+Caps?category=Display&preview.text=Yumeanime&preview.text\\_type=custom](https://fonts.google.com/specimen/Oleo+Script+Swash+Caps?category=Display&preview.text=Yumeanime&preview.text_type=custom)

- Lilita One : [https://fonts.google.com/specimen/Lilita+One?category=Display&preview.text=Yumeanime&preview.text\\_type=custom](https://fonts.google.com/specimen/Lilita+One?category=Display&preview.text=Yumeanime&preview.text_type=custom)

}

Parrafadas -> Mulish : [https://fonts.google.com/specimen/Mulish?category=Sans+Serif&preview.text=Lorem%20ipsum%20dolor%20sit%20amet,%20consectetur&preview.text\\_type=custom](https://fonts.google.com/specimen/Mulish?category=Sans+Serif&preview.text=Lorem%20ipsum%20dolor%20sit%20amet,%20consectetur&preview.text_type=custom)

## Web Design

La maqueta se puede encontrar en el siguiente enlace:

<https://github.com/raulfarce01/yumeanime/tree/main/Maqueta>

La guía de estilo se encuentra en este:

<https://github.com/raulfarce01/yumeanime/blob/main/Gu%C3%ADa%20de%20Estilo.pdf>

---

## Base de Datos

El modelo entidad - relación se puede encontrar en este enlace:

<https://github.com/raulfarce01/yumeanime/blob/main/YumeanimeE-R.png>

El grafo relacional se encuentra en este otro:

<https://github.com/raulfarce01/yumeanime/blob/main/Grafo%20Relacional%20Yumeanime.pdf>

En el último se explica qué era lo que faltaba para normalizar y el proceso para hacerlo.

---

## Casos de uso

En este esquema podemos ver los distintos usuarios que vamos a poder crear en la página y las funcionalidades de cada uno de ellos:

<https://github.com/raulfarce01/yumeanime/blob/main/DiagramaCasosUso.png>



## Anotaciones y Aclaraciones

### **Despliegue de la aplicación con Laravel y phpMyAdmin**

Para desplegar la aplicación hemos tenido que crear el siguiente virtualHost en la ruta:

*/etc/apache2/sites-available*

```
arce@arce: ~  
GNU nano 4.8 /etc/apache2/sites-available/yumeanime.com.conf  
<VirtualHost *:80>  
  
    ServerName yumeanime.com  
    ServerAlias www.yumeanime.com  
    DocumentRoot /var/www/yumeanime/web/public  
  
</VirtualHost>
```

La ruta del documento está puesta en public ya que ahí se encuentra la *welcome.blade.php* de laravel, la página donde tenemos la documentación y todo lo necesario de laravel. Una vez vayamos avanzando en el código crearemos una redirección al *index* del proyecto.

Y debemos escribir los siguientes comandos:

```
sudo a2ensite yumeanime.com.conf  
sudo systemctl restart apache2.service
```

También hemos tenido que añadir un Host en:

*/etc/hosts*

```
arce@arce: ~  
GNU nano 4.8 /etc/hosts  
127.0.0.1    localhost  
127.0.1.1    arce  
127.0.0.1    mysql.com  
127.0.0.1    yumeanime.com
```

Al hacer esto solamente, si entramos en *yumeanime.com* nos saldrá una pantalla de error típica de Laravel de que no hay permisos para mostrar la página, por eso, debemos escribir las siguientes líneas de comando:

```
sudo chown -R www-data:www-data /var/www/yumeanime/web  
sudo chmod -R 755 /var/www/yumeanime/web
```

Estas líneas darán los permisos necesarios para que, a partir de ahora, podamos ver la página de laravel.

Fuente: [Cómo instalar Laravel en Ubuntu Linux con Apache](#)



## **Implementación y Codificación**

Se ha optado por no utilizar laravel por la falta de tiempo, ya que no es algo que se puede aprender a utilizar en dos semanas, se necesita dedicar MUCHO tiempo para aprender a usar laravel y ser capaz de cerrar las brechas que tiene, por ejemplo, las bases de datos al crearlas con Laravel, entre muchas otras brechas que tiene.

*Esta decisión se ha tomado el día 19/04/2022*

---

Se ha optado por eliminar los modelos "UserComentaAnime" y "UserGustaAnime" porque los animes son recogidos de una API, por lo que no los tenemos en la base de datos, así que sería muy complicado guardar los comentarios y Likes. En este día también se han finalizado todos los modelos de la web en PHP.

*Esta decisión se ha tomado el día 06/05/2022*

---

Se ha eliminado el select de orden de la página de directorio debido a que la API no tenía forma de ordenar los animes según la categoría y fecha al mismo tiempo.

*Esta decisión se ha tomado el día 12/05/2022*

---

Se ha decidido realizar el AJAX en cambio de datos por input de la ventana perfil del usuario, ha habido problemas ya que al principio intenté sacar los inputs por PHP, pero después de mucho Stack Overflow y foros me enteré de que AJAX no puede interactuar con inputs del lado servidor, por lo que opté a usar el servidor solo para sacar la información.

Ah, y por alguna razón todos los días que cambio algo, si lo copio y pego en otro sitio, el JS dice: "Ah, no me gusta, no quiero funcionar" así que tengo que dedicar un ratito a cada página porque el JS es especialito :), es el niño tonto de la clase

*Esta decisión se ha tomado el día 16/05/2022*

---

Se ha finalizado el proyecto, pasándolo a la fase de testing en el entorno propio y solucionando problemas con el diseño de algunas páginas que no estaban al 100% completas. La funcionalidad de las listas al final no se ha llevado a cabo porque no tenemos JS en el servidor, y al imprimir etiquetas por PHP e intentar recogerlas por JS, según muchos foros y consulta con compañeros de fuera, se ha llegado a la conclusión de que para realizar esa funcionalidad es necesario el uso de herramientas semejantes a NodeJS.



La funcionalidad de las noticias funciona, pero no hay forma de crearlas desde la web, sino que deben hacerse desde la base de datos (para ello tenemos un documento tanto en PDF como en Markdown llamado CrearNoticiasDocu dentro de esta misma carpeta). Las noticias no se crean desde la página puesto que no conozco forma de distinguir las imágenes del texto y separar cada uno de los párrafos (ver la Base de Datos para entender qué son los párrafos) dentro de un mismo cuadro de texto y saber posicionar cada cosa en el sitio donde se han insertado a la hora de mostrarla.

Lo mismo sucede con las listas, al no poder crearlas desde la página podéis encontrar la documentación para la creación desde la base de datos en CrearListasDocu en PDF y en Markdown.

Las documentaciones se pueden encontrar en:

- [Documentación para la creación de listas en Base de Datos](#)
- [Documentación para la creación de noticias en Base de Datos](#)

*Esto ha sucedido el día 18/05/2022*

---

Se ha decidido desplegar la aplicación con Ngrok, una plataforma que usa una nube para redirigir una URL a tu Apache, que debe estar configurado para que te redirija a la carpeta donde se encuentra la web.

Se ha repartido la URL entre unos pocos usuarios y gracias a ellos he encontrado fallos, de los cuales he sido capaz de solucionar todos menos uno. En el móvil, al deslizar hacia la izquierda, aparece el menú desplegable que está escondido, el problema es que si bajas se ve un espacio en blanco porque la altura del menú hamburguesa está calculada para que sea la misma que la de la pantalla.

Para desplegarlo, hemos tenido que usar el comando:

```
ngrok http yumeanime.com:80
```

Ya que en el hosts marcamos que al poner eso nos redirigía a la página de la aplicación, como vimos en la configuración de Apache dentro de esta misma memoria.



También cabe destacar la creación de un pequeño script en bash porque mi PHP Server, a mitad del TFG dejó de funcionar, así que decidí crear una script para ejecutarlo cada vez que hiciera un cambio y así poder también acceder desde el host configurado. El script es el siguiente:

```
arce@arce: /etc/apache2/sites-available
GNU nano 4.8 /home/arce/scriptYumeanime.sh
#!/bin/bash
cd /var/www
sudo cp -R /home/arce/yumeanime ./
```

*Esto ha sucedido el 19/05/2022*

---

Se ha considerado ampliar el proyecto y se ha realizado una investigación para poder subir las noticias desde la página en lugar de la base de datos. Tras la investigación se ha observado que para esto necesitamos conocimientos de React, por lo que es otra funcionalidad que queda como extensión del proyecto.

*Esto ha sucedido el 26/05/2022*

---

El día 01/06/2022 he intentado arreglar la caída de la API del dominio público usando la API en mi propio entorno, así que he tenido que aprender a utilizar mongoDB. He tenido que clonar el repositorio de [AniAPI Github](#)

Esta imagen muestra las bases de datos, nosotros hemos creado api\_dotnet

```
> show dbs
admin          0.000GB
aniapi_dotnet  0.000GB
config         0.000GB
local          0.000GB
>
```



Aquí mostramos las colecciones, que equivalen a las tablas de mysql

```
> use aniapi_dotnet
switched to db aniapi_dotnet
> show collections
app_settings
website
>
```



Podemos ver en la siguiente imagen el valor de la colección "app\_settings", que contiene los valores para configurar el endpoint y la clave para que podamos usar la API en nuestro entorno

```
> db.app_settings.find()
{ "_id" : 0, "jwt_secret" : "XQAWglMRCodMO1Zsdvzyxful187oPLDghHKlPiuie4H24BNL0MVnR7YoBgqLoWP", "recaptcha_secret" : "", "proxy_host" : "", "proxy_port" : "", "proxy_username" : "", "proxy_password" : "", "proxy_count" : "", "resources_version" : "1.0", "api_endpoint" : "localhost/WebAPI", "smtp" : { "host" : "", "port" : "", "username" : "", "password" : "", "address" : "" }, "mal" : { "client_id" : "9725a48b60481eb7d26d57a07e1eb85f", "client_secret" : "749bd6dc67dda1b9eeddfd7546185872c7af0b7d98a156a72acc74efd81eeeb6" } }
```

En la siguiente y última imagen vemos los valores de los endpoints de los que la API que utilizamos saca los valores, todos correspondientes a endpoints de la API de MyAnimeList

```
> db.website.find()
{ "_id" : 1, "name" : "dreamsub", "active" : true, "official" : true, "site_url" : "https://dreamsub.cc", "can_block_requests" : true, "localization" : "it", "creation_date" : { "$date" : "2021-10-07T00:00:00.000Z" } }
{ "_id" : 2, "name" : "animeworld", "active" : true, "official" : false, "site_url" : "https://www.animeworld.tv/", "can_block_requests" : true, "localization" : "it", "creation_date" : { "$date" : "2021-10-07T00:00:00.000Z" } }
{ "_id" : 3, "name" : "gogoanime", "active" : true, "official" : true, "site_url" : "https://gogoanime.pe/", "can_block_requests" : true, "localization" : "en", "creation_date" : { "$date" : "2021-10-07T00:00:00.000Z" } }
```

La vida es una tómbola, tom, tom tómbola.  
La vida es una tómbola, tom, tom, tómbola.  
De luz y de color, de luz y de color.

Para el que valoró mis esfuerzos hasta el fin de los días: Basis, te quiero mucho <3  
Para el resto: empatía, por favor, gracias <3