



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Base de datos cinematográfica con sistema de recomendaciones por perfiles de usuarios

Trabajo Fin de Grado

Grado en Ingeniería Informática

Autor: Pablo Sornosa Pérez

Tutor: Sergio Sáez Barona

Curso 2020/2021

Resumen

Proyecto basado en la idea de convertir los datos y opiniones sobre toda la colección de obras hechas tanto para cine como la televisión en una interacción entre diversas personas. La mayoría de estas aplicaciones o páginas se basan en gran parte en la opinión de expertos o en personas generales, la idea para intentar diferenciar a esta aplicación es poder diferenciar a los usuarios y relacionarlos entre ellos, es decir, mostrar recomendaciones basadas en películas vistas por personas con perfiles similares. Se pueden hacer recomendaciones basadas en los gustos propios del usuario, por ejemplo, películas dentro de su género principal, o recomendaciones basadas en películas de otros géneros que le han gustado a otros usuarios cuyo género principal es similar al del usuario.

Palabras clave: base de datos, películas, series, actores, web, recomendaciones.

Abstract

Project based on the idea of converting data and opinions on the entire collection of works made for both film and television into an interaction between different people. Most of these applications or pages are based largely on the opinion of experts or general people, the idea to try to differentiate this application is to be able to differentiate users and relate them to each other, that is, to show recommendations based on movies seen by people with similar profiles. Recommendations can be made based on the user's own tastes, for example, movies within their main genre, or recommendations based on movies from other genres that other users have liked whose main genre is similar to that of the user.

Keywords: database, films, series, actors, web, recommendations.



Tabla de contenidos

1.	Introducción.....	9
1.1.	Motivación.....	9
1.2.	Objetivos.....	9
1.3.	Metodología.....	10
1.4.	Impacto esperado.....	10
1.5.	Estructura del documento.....	10
2.	Estado del arte.....	12
2.1.	Aplicaciones actuales.....	12
2.1.1.	Internet Movie Database.....	12
2.1.2.	FilmAffinity.....	12
2.1.3.	Rotten Tomatoes.....	13
2.2.	Propuesta.....	13
3.	Análisis del problema.....	15
3.1.	Análisis de requisitos.....	15
3.1.1.	Requisitos funcionales.....	15
3.1.2.	Requisitos no funcionales.....	18
3.2.	Análisis de casos de uso.....	18
3.2.1.	Usuario no registrado.....	18
3.2.2.	Usuario registrado.....	22
3.2.3.	Usuario editor.....	24
3.2.4.	Usuario administrador.....	27
3.3.	Análisis de seguridad.....	31
3.4.	Análisis del marco legal.....	31
3.4.1.	Análisis de protección de datos personales.....	32
3.4.2.	Análisis de las leyes de propiedad intelectual.....	34
3.5.	Identificación y análisis de distintas soluciones.....	36
3.5.1.	Aplicación de escritorio.....	36
3.5.2.	Aplicación web.....	37
3.6.	Solución propuesta.....	39
3.7.	Plan de trabajo.....	40
4.	Diseño de la solución.....	43
4.1.	Tecnologías utilizadas.....	43
4.1.1.	Tecnologías del lado del cliente.....	43

4.1.2.	Tecnologías del lado del servidor	45
4.1.3.	Administración de datos	46
4.2.	Arquitectura del sistema	46
4.3.	Diseño detallado	48
4.3.1.	Capa de presentación	49
4.3.2.	Capa de negocio	60
4.3.3.	Capa de datos	62
5.	Desarrollo de la solución.....	65
5.1.	Herramientas utilizadas	65
5.1.1.	XAMPP	65
5.1.2.	Microsoft Visual Studio Code	68
5.1.3.	Herramientas de diagramas	68
5.1.4.	GIMP	70
5.1.5.	Lenguajes y añadidos utilizados	71
5.2.	Implementación de algoritmos de recomendación	75
5.2.1.	Algoritmo de géneros	75
5.2.2.	Algoritmo de Películas del género principal.....	77
5.2.3.	Algoritmo de series de otros géneros	78
5.3.	Implementación de la Base de Datos.....	79
5.3.1.	Usuarios.....	79
5.3.2.	Tablas de registros	79
5.3.3.	Tablas de cambios	79
5.3.4.	Tablas de apoyo	80
6.	Pruebas	81
6.1.	Pruebas en navegadores.....	81
6.1.1.	Google Chrome	82
6.1.2.	Mozilla Firefox.....	83
6.1.3.	Opera	83
6.2.	Pruebas de funcionamiento.....	84
6.2.1.	Pruebas de seguridad	84
6.2.2.	Administración de errores	92
7.	Conclusiones.....	96
7.1.	Objetivos iniciales	96
7.2.	Relación con los estudios cursados	97
8.	Bibliografía.....	98



Tabla de figuras

Figura 1- Diagrama de casos de uso para usuario no registrado	22
Figura 2- Diagrama de casos de uso para usuario registrado	23
Figura 3- Diagrama de casos de uso para usuario editor	26
Figura 4- Diagrama de casos de uso para usuario administrador	31
Figura 5- Planificación temporal.....	41
Figura 6- Planificación temporal.....	41
Figura 7- Planificación temporal.....	42
Figura 8- Arquitectura del Sistema.....	47
Figura 9- Arquitectura de capas	48
Figura 10- Esquema menú principal	49
Figura 11- Esquema de configuración del perfil	50
Figura 12- Esquema de configuración del perfil	51
Figura 13- Esquema de configuración del perfil	51
Figura 14- Esquema de listas.....	52
Figura 15- Esquema de listas de series.....	53
Figura 16- Esquema de listas de actores	53
Figura 17- Esquema de listas de directores	54
Figura 18- Esquema de lista de noticias.....	54
Figura 19- Esquema de listas de administración de series	55
Figura 20- Esquema de página de visualización de película	55
Figura 21- Esquema de página de visualización de serie.....	56
Figura 22- Esquema de página de visualización de actores.....	57
Figura 23- Esquema de página de visualización de director.....	58
Figura 24- Esquema de página de visualización de noticias	58
Figura 25- Esquema de página de visualización de noticias	59
Figura 26- Esquema de página de edición de película, apartado actores.....	59
Figura 27- Esquema de página de visualización de cambios	60
Figura 28- Diagrama de la capa de negocio.....	61
Figura 29- Diagrama de la capa de datos.....	63
Figura 30- Panel de control de XAMPP v3.2.4	65
Figura 31- Configuración SMTP en el archivo php.ini	66
Figura 32- Configuración SMTP en el archivo sendmail.ini	67
Figura 33- Panel de control de una tabla en PHPMyAdmin.....	67
Figura 34- Visualización de un archivo HTML en VS Code	68
Figura 35- Panel principal de Diagrams.net	69
Figura 36- Panel principal de Creately.....	69
Figura 37- Menú de Balsamiq	70
Figura 38- Menú de herramientas de GIMP	70
Figura 39- Ejemplo de funcionamiento del sistema Grid.....	71
Figura 40- Muestra de funcionamiento de Carousel en el proyecto	72
Figura 41- Ejemplo de 2 tipos de botones usados en el proyecto	72
Figura 42- Ejemplo de modal usado en el proyecto	73
Figura 43- Ejemplo de DataTable	74
Figura 44- Select Picker con búsqueda usado en el proyecto.....	74
Figura 45- Fragmento de código utilizando métodos jQuery	75



Figura 46- Fragmento de código del sistema de selección de género	76
Figura 47- Código del algoritmo recomendador de género principal.....	77
Figura 48- Código de algoritmo recomendador de otros géneros	78
Figura 49- Código fuente de Google.es validado por los validadores del W3C dando errores.....	81
Figura 50- Prueba de la página principal en el navegador Google Chrome.....	82
Figura 51- Prueba de la página principal en el navegador Mozilla Firefox.....	83
Figura 52- Prueba de la página principal en el navegador Opera	83
Figura 53- Menú principal sin recomendaciones para usuario no registrado	84
Figura 54- Modal de Inicio de sesión o registro para usuario no registrado	85
Figura 55- Código redireccionador para URL sin permisos	85
Figura 56- Muestra de menú sin opciones en usuario no registrado	85
Figura 57- Recomendaciones aparecen con el rol usuario con sesión iniciada.....	86
Figura 58- Muestra de menú para cerrar sesión o mostrar datos con usuario con sesión iniciada	86
Figura 59- Página de administrar cuenta accesible para usuarios con sesión iniciada	87
Figura 60- Administración inaccesible para usuarios sin permisos	87
Figura 61- Fragmento de código de seguridad.....	88
Figura 62- Muestra de que se permite administrar todo salvo cambios con usuario editor	88
Figura 63- Muestra de la imposibilidad de añadir o eliminar para el usuario editor	89
Figura 64- Código para redirigir los cambios a una tabla secundaria para usuarios editores	89
Figura 65- Página de cambios para usuarios editores	90
Figura 66- Muestra de que el usuario administrador puede borrar y añadir datos.....	90
Figura 67- Página de creación de películas	91
Figura 68- Lista de cambios	91
Figura 69- Página de administración de cambios.....	92
Figura 70- Código de seguridad para administradores	92
Figura 71- Error de campos vacíos en edición de películas	93
Figura 72- Error de campo incorrecto en la edición de películas.....	93
Figura 73- Error de campos vacíos en creación de películas	94
Figura 74- Código de eliminación de registros incompletos.....	94

1. Introducción

En este apartado se describen las ideas iniciales del proyecto, tales como la motivación para hacer el proyecto, los objetivos, el impacto esperado que produciría el proyecto, la estructura del documento y la metodología usada para desarrollarlo.

1.1. Motivación

Inicialmente, al haber cursado la rama de Tecnologías de la Información, tuve contacto con el desarrollo de páginas y aplicaciones web por medio de varias asignaturas de esta especialidad.

A partir de este primer contacto en la asignatura de Desarrollo Web del 3er curso del grado, también obtuve experiencia laboral en este campo tanto dentro de prácticas como fuera de ellas, mejorando mis capacidades en esta área, además de aumentar considerablemente mi interés cuanto más profundizaba en otras tecnologías complementarias.

Por lo tanto, habiendo obtenido más experiencia y curiosidad por esta rama de la informática, me decidí a hacer este proyecto uniendo esta rama y mi gusto por el cine y la televisión.

1.2. Objetivos

El principal objetivo de este proyecto es ofrecer una optativa a las páginas de colecciones de películas y series tales como *IMDb* o *Rotten Tomatoes* que, en vez de centrar las recomendaciones y valoraciones en base a las opiniones de expertos, darle a la aplicación otro enfoque, basado en los sistemas de recomendaciones de las redes sociales o los servicios de *Streaming* que están siendo muy utilizados en los últimos años. Relacionado con esto, un objetivo secundario es la posibilidad de orientar las recomendaciones mediante una configuración inicial por género, edad o servicios de *Streaming* que el usuario tenga contratados.

Otro objetivo, desarrollando la idea antes mencionada de los sistemas de recomendaciones, es intentar ofrecer variedad, es decir, no repetir una y otra vez unas recomendaciones que ya se han mostrado, se han visto, y se han descartado.

Por último, un objetivo secundario es la colaboración, en la que cualquier usuario previamente aceptado por un administrador pueda ayudar en la edición y el mantenimiento de los datos de la página, en un sistema en el que los administradores se encargan de revisar y confirmar los cambios para evitar contenido no deseado o inventado.



1.3. Metodología

Principalmente, la metodología de desarrollo a utilizar será un desarrollo en cascada, ya que se realizarán paso por paso, de forma secuencial, las fases del desarrollo de la aplicación. Es decir, primero el análisis del problema, después el diseño de la solución a desarrollar, seguido del desarrollo de una solución y por último unas pruebas, pudiendo ser opcional en un futuro el mantenimiento si fuera necesario.

A pesar de seguir un modelo en cascada, es posible que en un futuro convenga revisar los requisitos que se realicen, por lo cual los costes de desarrollo se podrían incrementar, pero es seguramente la mejor alternativa ya que cada fase del desarrollo deberá estar bien comentada y justificada, todo esto para no complicar el desarrollo si se decide retomar el proyecto más adelante.

1.4. Impacto esperado

Se espera que el proyecto suponga varias mejoras respecto a los sistemas actuales de bases de datos de películas y series como *IMDb*, relacionadas especialmente con el usuario, aunque también que haya mejoras respecto al sistema de edición.

Desde el punto de vista del usuario, se esperan varias mejoras en el campo de las recomendaciones de los usuarios, ya que, como se ha comentado en el apartado de los objetivos, una de las principales funcionalidades que se plantea implementar son las recomendaciones basadas en otros sistemas usados en otros servicios, especialmente de *Streaming*, siempre con la posibilidad de configurar la cuenta para obtener recomendaciones genéricas y no usar los datos del propio usuario.

También, la idea de que las recomendaciones se puedan centrar en servicios que el usuario tenga contratados podrá proporcionar una ventaja respecto a otros sistemas, ya que permitirá al usuario reducir las búsquedas a películas/series que pueda ver sin necesidad de invertir dinero o recurrir a la piratería.

Por parte de los administradores de la página, una de las ventajas que se espera obtener con el sistema de edición es que, los usuarios con rol administrador puedan nombrar a editores que puedan editar sin consentimiento previo de este, y a usuarios registrados se les permita editar con un consentimiento previo para confirmar la notificación, como funcionan las herramientas como *GitHub*.

1.5. Estructura del documento

La memoria se estructurará de la siguiente forma, dividiendo el contenido en apartados relacionados con cada una de las fases del proyecto:

- **Introducción:** en este apartado se explica tanto la motivación como los objetivos para el desarrollo del proyecto, junto la metodología a seguir en la realización del proyecto. Además, se especifica el impacto esperado sobre el proyecto.

- **Estado del arte:** en este punto se explican las distintas alternativas que se pueden encontrar en el mercado hoy en día similares al proyecto que se desarrollará en el trabajo, junto con la propuesta de proyecto que se va a desarrollar.
- **Análisis del problema:** aquí se especifican todos los requisitos de la aplicación, tanto funcionales como no funcionales, además de los casos de uso de la aplicación, con sus respectivos diagramas según sus características. Por otra parte, también se encuentra el análisis de seguridad, donde se estudian las distintas acciones a llevar a cabo con la seguridad de la aplicación. También se incluye un análisis del marco legal, tanto en la vertiente de la protección de datos personales como en las leyes de propiedad intelectual. Una vez analizado todo esto, se analizan distintas soluciones y se selecciona un tipo de desarrollo según sus puntos fuertes y sus puntos débiles y a partir de esto, se decide un plan de trabajo.
- **Diseño de la solución:** en este apartado se diseña la solución propuesta anteriormente, donde se investigan y especifican las distintas tecnologías a utilizar, divididas por sus características y funcionalidades. Aparte de esto, se desarrolla la arquitectura del sistema junto con un diseño detallado de la misma, incluyendo todas sus capas.
- **Desarrollo de la solución:** en este punto se especifican todas las tecnologías utilizadas finalmente en el desarrollo, al igual que las herramientas. También se explican las distintas implementaciones del proyecto que sean necesarias, así como las implementaciones de la Base de Datos que se hayan llevado a cabo.
- **Pruebas:** se realizan pruebas tanto de funcionalidad en distintos sistemas como del correcto funcionamiento de la aplicación tanto a nivel de seguridad como tratamiento de los distintos errores que puedan surgir en esta.
- **Conclusiones:** en este apartado se analizan las conclusiones, tanto relacionadas con los objetivos iniciales desarrollados en la introducción como una valoración de dificultades técnicas relacionadas con los estudios cursados.
- **Bibliografía:** en este apartado se hace referencia a todos documentos o páginas utilizados para el desarrollo de la memoria, así como sus respectivos números enlazando con las referencias del texto. Además de esto, también se incluye una lista de trabajos consultados.



2. Estado del arte

2.1. Aplicaciones actuales

Hay muchos sitios web que se encargan de recopilar información sobre películas y series. Con el creciente uso tanto de los dispositivos conectados a internet como de los servicios de *streaming* como Netflix, HBO o Amazon Prime Video, estas plataformas están siendo cada día más conocidas, ya que, al permitirnos los servicios antes mencionados una gran cantidad de películas y series de televisión para elegir, la gente tiende a informarse cada vez más del producto que va a ver antes de verlo.

En España, posiblemente las 3 más conocidas son *IMDb*, *FilmAffinity* y *Rotten Tomatoes*.

2.1.1. Internet Movie Database

IMDb, acrónimo de *Internet Movie Database*, es un sitio web fundado el 17 de octubre de 1990 por Col Needham, un ingeniero de software británico (1). El origen de esta página se encuentra en un grupo de noticias de *usenet*, donde algunos usuarios comenzaron a publicar nombres de actores y actrices con las películas en las que habían participado. A partir de ahí, Needham comenzó *IMDb* como una lista de las películas que había visto desde 1980, desarrollando posteriormente el primer software que permitía la consulta, basada en scripts de Unix (2).

Entre 1992 y 1993, se fue adaptando un sistema de consultas basado en el correo electrónico, mientras que también, en 1993, se publicó la base de datos en los inicios de *World Wide Web*. En 1998 fue comprada por Amazon.com (2).

Hoy en día, *IMDb* permite la consulta de gran cantidad de datos relacionados tanto con la producción como el desempeño de una película en el cine o televisión, así como datos de la propia película como premios, fechas de estreno o restricciones de edad, sin perder sus características iniciales como base de datos de actores, actrices y películas.

2.1.2. FilmAffinity

FilmAffinity es un sitio web español dedicado al cine y las series. Al igual que *IMDb*, está clasificada como Base de datos cinematográfica. Fue creada en 2002 por Pablo Kurt y Daniel Nicolás, siendo el primero un crítico de cine y el segundo un programador. (3)

FilmAffinity se hizo conocida por un sistema de recomendación de películas basado en afinidades entre usuarios llamado "Almas gemelas". Aparte de este sistema también posee un apartado donde los usuarios pueden hacer críticas sobre las películas.

Hoy en día, este sitio web ofrece versiones para otros países como EE. UU, México, Argentina o Chile, así como una versión en inglés, pero sin tantas funcionalidades como la versión española.

2.1.3. Rotten Tomatoes

Rotten Tomatoes es un sitio web de origen estadounidense creado en 1998 (4) por Senh Duong que se dedica a la recopilación de críticas de películas y series de televisión, teniendo como característica única un sistema de valoraciones del cual le viene el nombre, ya que se basa en la acción que cometían algunas audiencias en actuaciones en directo, tirar tomates podridos si la actuación era mala.

La idea original del creador era recopilar las películas de Jackie Chan, ya que Duong ya había recopilado todas las reseñas de todas las películas que había hecho este actor en Hong Kong, poniendo en marcha la idea cuando se iba a estrenar la primera película de Jackie Chan en Hollywood, pero al retrasarse el estreno, Duong decidió comenzar a incluir otras películas en su página. (4)

El famoso *TOMATOMETER*® funciona de la siguiente manera: se recopilan varias críticas tanto de audiencia como de críticos profesionales (ya que hay 2 medidores, uno para crítica y otro para usuarios), luego se cuentan cuántas de las valoraciones son positivas del total, si supera el 60% de valoraciones positivas, se le otorga el rango de tomate fresco, si no, se le considera tomate podrido, también hay una tercera categoría, que se utiliza cuando una película no se ha estrenado o no hay suficientes críticas. También hay una distinción especial llamada *Certified Fresh*, que consiste en que la puntuación debe de estar por encima del 75%, haber sido reseñada por 5 críticos de gran renombre y deben tener un mínimo de reseñas según el tipo de producto y estreno que siga. (5)

2.2. Propuesta

La propuesta del proyecto es la siguiente, como se ha podido ver en el apartado de estado del arte, los sitios web dedicados a las reseñas y noticias sobre películas son muy diversos, ya que cada uno implementa unas funcionalidades únicas, lo cual hace que las características más interesantes estén repartidas entre las distintas plataformas.

Por lo tanto, la propuesta del proyecto es añadir nuevas opciones interesantes, como las recomendaciones basadas en una funcionalidad similar a los servicios de *streaming*, pero también combinar y mejorar otras funcionalidades que se encuentran en algunos servicios ya existentes, como puede ser el sistema de edición de *IMDb* o el sistema de recomendaciones de *FilmAffinity* llamado “almas gemelas”.

Otras opciones interesantes para añadir son relacionar estas películas con los servicios de *streaming*, como, por ejemplo, a la hora de hacer recomendaciones, poder configurar la cuenta de la aplicación para que solo se recomienden películas que estén en ese momento en los servicios que tenga seleccionados en la cuenta dicho usuario. También es interesante añadir un Chat Bot para simplificar la búsqueda de un baremo amplio de



Base de datos cinematográfica con sistema de recomendaciones por perfiles de usuarios

opciones, como pueden ser todas las películas de un género a partir de un año en concreto.

3. Análisis del problema

3.1. Análisis de requisitos

Los requisitos del sistema se pueden diferenciar en dos bloques, los requisitos funcionales y los requisitos no funcionales.

Los requisitos funcionales son aquellos que especifican las funciones que puede realizar el sistema, formado por un conjunto de entradas, procesos y salidas que ejecuta este sistema, pudiendo diferenciar las funciones según los distintos tipos de usuarios que puedan hacer uso de esta aplicación

Los requisitos no funcionales complementan a los anteriores, ya que, en vez de especificar las funcionalidades del sistema, se refiere a las propiedades del propio sistema. Esto incluye tanto a propiedades como limitaciones del sistema.

3.1.1. Requisitos funcionales

Los requisitos funcionales van a variar según el tipo de usuario que use la aplicación, ya que podemos diferenciar distintos tipos: usuario no registrado o anónimo, usuario registrado, editor y administrador.

3.1.1.1. Usuario no registrado

El usuario no registrado es aquel que no tiene unas credenciales guardadas en la aplicación, o que simplemente no tiene una sesión iniciada, por lo que solo tiene las funciones básicas, sin ningún tipo de recomendación personalizada ni ningún tipo de dato del usuario guardado.

Id	Nombre	Descripción
RF-01	Registro	Permite al usuario registrarse en la aplicación.
RF-02	Buscar películas	Permite al usuario buscar películas en una lista con una serie de filtros
RF-03	Ver películas	Permite al usuario visualizar los datos de una película después de hacer una búsqueda
RF-04	Buscar series	Permite al usuario buscar series en una lista con una serie de filtros
RF-05	Ver series	El usuario puede ver los datos de una serie de la cual se ha hecho una búsqueda
RF-06	Buscar actores/actrices	Permite al usuario buscar actores o actrices en una lista con una serie de filtros
RF-07	Ver actores/actrices	El usuario puede ver los datos de un actor o actriz del cual se ha hecho una búsqueda



RF-08	Buscar directores	El usuario puede hacer una búsqueda de directores en una lista con varios filtros
RF-09	Ver directores	El usuario puede ver los datos de un director del cual se ha hecho una búsqueda
RF-10	Buscar noticias	El usuario puede buscar en una lista de las noticias guardadas en el sistema
RF-11	Ver noticias	El usuario puede visualizar las noticias después de una búsqueda
RF-12	Búsqueda general	El usuario puede hacer una búsqueda general sobre cualquiera de las clases que hemos mencionado (películas, series y actores o actrices)

3.1.1.2. *Usuario registrado*

El usuario registrado o usuario con sesión iniciada (se le llamará usuario con sesión iniciada por simplificación, pero al ser el inicio de sesión una acción no permitida al usuario no registrado se debe especificar) es aquel que tiene credenciales creadas en la base de datos del sistema, y todos los requisitos funcionales especificados anteriormente están implícitos en este usuario (Salvo el registro, ya que no se pueden repetir datos de inicio de sesión). Los siguientes requisitos funcionales son para el usuario que ya ha iniciado sesión, salvo el primero que es la propia función de inicio de sesión.

Id	Nombre	Descripción
RF-13	Iniciar sesión	El usuario puede usar sus credenciales para iniciar sesión en el sistema
RF-14	Configurar cuenta	El usuario puede acceder a configuración y cambiar o añadir sus datos personales o los servicios de <i>streaming</i> de los que dispone, entre otras cosas

3.1.1.3. *Usuario con permisos de edición o editor*

El usuario con permisos de editor es aquel que ha solicitado permisos de edición al administrador y este ha configurado los permisos de su cuenta para que pueda hacer ediciones de los registros públicos (Datos de películas, series y actores o actrices) y crear nuevos registros del mismo tipo, siempre todo esto con autorización de un usuario administrador, por tanto, hereda los requisitos anteriores y añade otros nuevos. Nunca tiene permiso para borrar ningún registro completo, es decir, puede borrar texto de una noticia de una película, pero no puede borrar la noticia en sí, solo puede consultarle al administrador, al igual que tampoco puede crear un registro de 0.

Id	Nombre	Descripción
RF-15	Editar películas	Permite al usuario editar los datos de las películas

RF-16	Editar series	Permite al usuario editar los datos de las series
RF-17	Editar actores o actrices	Permite al usuario editar los datos de los actores
RF-18	Editar directores	El usuario puede editar los datos de los directores
RF-19	Editar noticias	El usuario puede editar los datos de las noticias

3.1.1.4. *Usuario administrador*

El usuario administrador es aquel que tiene permisos para administrar todos los datos de la aplicación (siempre teniendo en cuenta las leyes de protección de datos de los usuarios, por lo que no puede ver los datos personales). Los usuarios administradores pueden crear y eliminar registros completos, dar de baja a usuarios, dar permisos de edición a los usuarios y dar permisos de administrador. Como es de esperar, hereda todos los requisitos funcionales de los grupos especificados anteriormente.

Id	Nombre	Descripción
RF-20	Añadir películas	Permite al usuario crear un nuevo registro de película en la base de datos.
RF-21	Borrar películas	Permite al usuario borrar películas de la base de datos.
RF-22	Añadir series	Permite al usuario añadir una nueva serie en la base de datos
RF-23	Borrar series	Permite al usuario borrar registros de series de la base de datos.
RF-24	Añadir actores y actrices	Permite al usuario añadir un registro de actor o actriz a la base de datos
RF-25	Borrar actores y actrices	Permite al usuario borrar registros de actores o actrices de la base de datos.
RF-26	Añadir directores	El usuario puede añadir registros de nuevos directores a la base de datos
RF-27	Borrar directores	Permite al usuario eliminar registros de directores de la base de datos
RF-28	Añadir noticias	Permite al usuario crear una nueva noticia en la base de datos
RF-29	Borrar noticias	Permite al usuario borrar una noticia de la base de datos
RF-30	Administrar cambios	Permite al usuario administrar los cambios que han hecho los usuarios editores



3.1.2. Requisitos no funcionales

Los requisitos no funcionales, como ya se ha explicado antes, no se corresponden con funcionalidades del sistema, si no con características y/o restricciones de este.

Id	Descripción
RNF-01	El sistema debe estar disponible continuamente, es decir, se deben intentar evitar caídas del servicio para no perjudicar al usuario
RNF-02	El sistema debe asegurar la consistencia de los datos, que no se borren perfiles, se cambien configuraciones sin que el usuario lo permita o que se pierdan datos sobre las películas o las series.
RNF-03	La seguridad debe estar garantizada en el sistema, ningún usuario debe poder acceder a la contraseña o al correo de otro usuario.
RNF-04	La protección de los datos debe cumplirse según las especificaciones legales, por lo cual no se pueden mostrar los datos personales del usuario. Ningún usuario debe poder acceder a los datos de otro, aunque tenga máximos privilegios en la aplicación.
RNF-05	Se debe garantizar el sencillo uso de la aplicación, para que cualquier usuario, independientemente del nivel técnico que posea, pueda utilizar de manera rápida y fácil la aplicación.
RNF-06	El sistema debe garantizar tiempo de respuesta lo más bajo posible, es decir, asegurar un nivel de rendimiento que permita una experiencia satisfactoria para el usuario.

3.2. Análisis de casos de uso

Después de analizar los requisitos funcionales y no funcionales, se deben analizar los casos de uso de la aplicación, los cuales se dividirán según los distintos usuarios. Dentro de los casos de uso nos encontramos también unos diagramas que servirán para entender de forma más clara y concisa los casos de uso.

3.2.1. Usuario no registrado

Nombre	Registro												
Precondición	El usuario no debe estar registrado ni tener una sesión iniciada												
Descripción	El usuario puede registrarse en la aplicación												
Secuencia	<table border="1"> <tbody> <tr> <td>1</td> <td>En el menú principal, el usuario pulsa el botón de usuarios</td> </tr> <tr> <td>2</td> <td>El sistema muestra una opción para registrarse y una para iniciar sesión</td> </tr> <tr> <td>3</td> <td>El usuario pulsa el botón de registro</td> </tr> <tr> <td>4</td> <td>El sistema muestra unos campos de registro</td> </tr> <tr> <td>5</td> <td>El usuario rellena dichos datos y pulsa el botón de registro</td> </tr> <tr> <td>6</td> <td>El sistema comprueba los datos y si son correctos el registro se completa con éxito</td> </tr> </tbody> </table>	1	En el menú principal, el usuario pulsa el botón de usuarios	2	El sistema muestra una opción para registrarse y una para iniciar sesión	3	El usuario pulsa el botón de registro	4	El sistema muestra unos campos de registro	5	El usuario rellena dichos datos y pulsa el botón de registro	6	El sistema comprueba los datos y si son correctos el registro se completa con éxito
1	En el menú principal, el usuario pulsa el botón de usuarios												
2	El sistema muestra una opción para registrarse y una para iniciar sesión												
3	El usuario pulsa el botón de registro												
4	El sistema muestra unos campos de registro												
5	El usuario rellena dichos datos y pulsa el botón de registro												
6	El sistema comprueba los datos y si son correctos el registro se completa con éxito												
Postcondición	El usuario ya está registrado en la aplicación.												
Excepciones	Los datos están repetidos y el registro no ha podido completarse												

Nombre	Búsqueda de películas	
Precondición	Ninguna	
Descripción	El usuario puede hacer una búsqueda de películas en la aplicación	
Secuencia	1	En el menú principal, el usuario pulsa el botón de películas
	2	El sistema abre una página, hace una búsqueda de todos los registros de películas y los muestra, así como un conjunto de filtros.
Postcondición	El usuario puede buscar películas utilizando los filtros	
Excepciones	Ninguna	

Nombre	Ver datos de una película	
Precondición	Haber hecho una búsqueda, ya sea mediante la lista o el buscador	
Descripción	El usuario puede ver todos los datos disponibles sobre una película: título, actores, directores, recaudación, etc.	
Secuencia	1	Una vez hecha la búsqueda, el usuario pulsa en una de las opciones disponibles
	2	El sistema abre una página, recupera los datos relacionados con dicha película y los muestra
Postcondición	El usuario puede ver los datos de la película seleccionada	
Excepciones	Si se introducen los identificadores mediante URL, la página puede dar errores dado que algunos de estos pueden no estar siendo usados aún.	

Nombre	Búsqueda de series	
Precondición	Ninguna	
Descripción	El usuario puede hacer una búsqueda de series en la aplicación	
Secuencia	1	En el menú principal, el usuario pulsa el botón de series
	2	El sistema abre una página, hace una búsqueda de todos los registros de series y los muestra, así como un conjunto de filtros.
Postcondición	El usuario puede navegar en la lista de las series utilizando los filtros	
Excepciones	Ninguna	

Nombre	Ver datos de una serie	
Precondición	Haber hecho una búsqueda de series, ya sea mediante la lista o el buscador	
Descripción	El usuario puede ver todos los datos disponibles sobre una serie: título, actores, directores, temporadas y capítulos, etc.	
Secuencia	1	Una vez hecha la búsqueda, el usuario pulsa en una de las opciones disponibles
	2	El sistema abre una página, recupera los datos relacionados con dicha serie y los muestra
Postcondición	El usuario puede ver los datos de la película seleccionada	
Excepciones	Si se introducen los identificadores mediante URL, la página puede dar errores dado que algunos de estos pueden no estar siendo usados aún.	

Nombre	Búsqueda de actores o actrices	
Precondición	Ninguna	
Descripción	El usuario puede hacer una búsqueda de los actores o actrices en la aplicación	
Secuencia	1	En el menú principal, el usuario pulsa el botón de actores
	2	El sistema abre una página, hace una búsqueda de todos los registros de los actores y las actrices, y los muestra, así como un conjunto de filtros.
Postcondición	El usuario puede navegar en la lista de los actores y actrices utilizando los filtros	
Excepciones	Ninguna	

Nombre	Ver datos de un actor o una actriz	
Precondición	Haber hecho una búsqueda de actores y actrices, ya sea mediante la lista o el buscador	
Descripción	El usuario puede ver todos los datos disponibles sobre un actor o una actriz: nombre, fecha de nacimiento, películas en las que ha participado, etc.	
Secuencia	1	Una vez hecha la búsqueda, el usuario pulsa en una de las opciones disponibles
	2	El sistema abre una página, recupera los datos relacionados con dicha serie y los muestra
Postcondición	El usuario puede ver los datos de la película seleccionada	
Excepciones	Si se introducen los identificadores mediante URL, la página puede dar errores dado que algunos de estos pueden no estar siendo usados aún.	

Nombre	Búsqueda de directores	
Precondición	Ninguna	
Descripción	El usuario puede hacer una búsqueda de los directores en la aplicación	
Secuencia	1	En el menú principal, el usuario pulsa el botón de directores
	2	El sistema abre una página, hace una búsqueda de todos los registros de los directores, y los muestra, así como un conjunto de filtros.
Postcondición	El usuario puede navegar en la lista de los directores utilizando los filtros	
Excepciones	Ninguna	

Nombre	Ver datos de un director o una directora	
Precondición	Haber hecho una búsqueda de directores, ya sea mediante la lista o el buscador	
Descripción	El usuario puede ver todos los datos disponibles sobre un director o una directora: nombre, fecha de nacimiento, películas en las que ha participado, etc.	
Secuencia	1	Una vez hecha la búsqueda, el usuario pulsa en una de las opciones disponibles
	2	El sistema abre una página, recupera los datos relacionados con dicho director y los muestra

Postcondición	El usuario puede ver los datos del director o directora seleccionado
Excepciones	Si se introducen los identificadores mediante URL, la página puede dar errores dado que algunos de estos pueden no estar siendo usados aún.

Nombre	Búsqueda de noticias				
Precondición	Ninguna				
Descripción	El usuario puede hacer una búsqueda de las noticias sobre películas o series en la aplicación				
Secuencia	<table border="1"> <tr> <td>1</td> <td>En el menú principal, el usuario pulsa el botón de noticias</td> </tr> <tr> <td>2</td> <td>El sistema abre una página, hace una búsqueda de todos los registros de las noticias, y los muestra, así como un conjunto de filtros.</td> </tr> </table>	1	En el menú principal, el usuario pulsa el botón de noticias	2	El sistema abre una página, hace una búsqueda de todos los registros de las noticias, y los muestra, así como un conjunto de filtros.
1	En el menú principal, el usuario pulsa el botón de noticias				
2	El sistema abre una página, hace una búsqueda de todos los registros de las noticias, y los muestra, así como un conjunto de filtros.				
Postcondición	El usuario puede navegar en la lista de las noticias utilizando los filtros				
Excepciones	Ninguna				

Nombre	Ver noticias				
Precondición	Haber hecho una búsqueda de noticias, ya sea mediante la lista o el buscador				
Descripción	El usuario puede ver la noticia junto con un conjunto de imágenes, una principal y tres secundarias				
Secuencia	<table border="1"> <tr> <td>1</td> <td>Una vez hecha la búsqueda, el usuario pulsa en una de las opciones disponibles</td> </tr> <tr> <td>2</td> <td>El sistema abre una página, recupera los datos relacionados con dicho director y los muestra</td> </tr> </table>	1	Una vez hecha la búsqueda, el usuario pulsa en una de las opciones disponibles	2	El sistema abre una página, recupera los datos relacionados con dicho director y los muestra
1	Una vez hecha la búsqueda, el usuario pulsa en una de las opciones disponibles				
2	El sistema abre una página, recupera los datos relacionados con dicho director y los muestra				
Postcondición	El usuario puede ver la noticia seleccionada				
Excepciones	Si se introducen los identificadores mediante URL, la página puede dar errores dado que algunos de estos pueden no estar siendo usados aún.				

Nombre	Búsqueda general				
Precondición	Ninguna				
Descripción	El usuario puede hacer una búsqueda general de cualquier elemento, ya sean películas, series, actores, directores o noticias				
Secuencia	<table border="1"> <tr> <td>1</td> <td>El usuario escribe en el cuadro de búsqueda ubicado a la derecha del menú</td> </tr> <tr> <td>2</td> <td>El sistema hace una búsqueda de máximo 2 elementos por categoría a partir de 3 caracteres, que se actualiza conforme el usuario escribe</td> </tr> </table>	1	El usuario escribe en el cuadro de búsqueda ubicado a la derecha del menú	2	El sistema hace una búsqueda de máximo 2 elementos por categoría a partir de 3 caracteres, que se actualiza conforme el usuario escribe
1	El usuario escribe en el cuadro de búsqueda ubicado a la derecha del menú				
2	El sistema hace una búsqueda de máximo 2 elementos por categoría a partir de 3 caracteres, que se actualiza conforme el usuario escribe				
Postcondición	El usuario puede acceder a los datos que ha buscado				
Excepciones	Si no se introducen mínimo 3 caracteres la búsqueda no funciona				



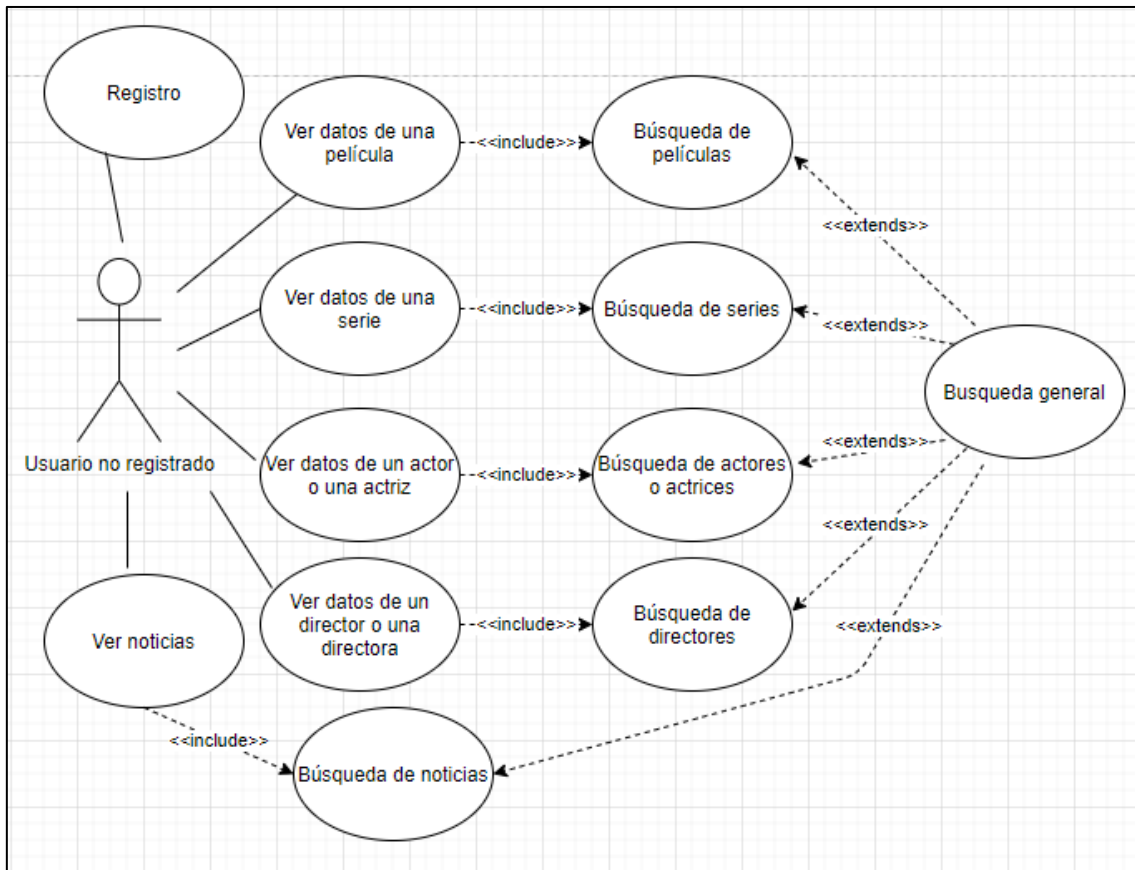


Figura 1- Diagrama de casos de uso para usuario no registrado

3.2.2. Usuario registrado

Nombre	Iniciar sesión								
Precondición	Haberse registrado con anterioridad en la aplicación mediante el apartado de registro								
Descripción	El usuario puede iniciar sesión, esto le permite recomendaciones personalizadas y cambiar algunas configuraciones								
Secuencia	<table border="1"> <tr> <td>1</td> <td>El usuario pulsa el botón de usuarios en el menú, arriba a la derecha</td> </tr> <tr> <td>2</td> <td>El sistema abre un <i>modal</i> donde el usuario puede introducir sus datos de sesión.</td> </tr> <tr> <td>3</td> <td>El usuario rellena los datos de sesión y pulsa el botón de iniciar sesión</td> </tr> <tr> <td>4</td> <td>El sistema busca en el registro el nombre de usuario y si existe comprueba si los datos son correctos. Una vez comprobados, se cargan unas variables con los datos del usuario y se redirige a la página principal.</td> </tr> </table>	1	El usuario pulsa el botón de usuarios en el menú, arriba a la derecha	2	El sistema abre un <i>modal</i> donde el usuario puede introducir sus datos de sesión.	3	El usuario rellena los datos de sesión y pulsa el botón de iniciar sesión	4	El sistema busca en el registro el nombre de usuario y si existe comprueba si los datos son correctos. Una vez comprobados, se cargan unas variables con los datos del usuario y se redirige a la página principal.
1	El usuario pulsa el botón de usuarios en el menú, arriba a la derecha								
2	El sistema abre un <i>modal</i> donde el usuario puede introducir sus datos de sesión.								
3	El usuario rellena los datos de sesión y pulsa el botón de iniciar sesión								
4	El sistema busca en el registro el nombre de usuario y si existe comprueba si los datos son correctos. Una vez comprobados, se cargan unas variables con los datos del usuario y se redirige a la página principal.								
Postcondición	El usuario ha iniciado sesión en la aplicación								
Excepciones	Si los datos introducidos son incorrectos, saltará una alerta indicando que no se ha podido iniciar sesión								

Nombre	Configurar cuenta	
Precondición	El usuario debe haber iniciado sesión y tener la sesión abierta	
Descripción	El usuario puede configurar varios parámetros, como sus datos personales, algunos ajustes de privacidad, las credenciales de inicio de sesión, etc.	
Secuencia	1	El usuario pulsa el botón de usuarios en el menú, arriba a la derecha
	2	Se despliega una pequeña ventana que muestra el correo del usuario, la foto y dos botones, siendo uno un botón en el que pone "Mas Datos"
	3	El usuario pulsa el botón de "Más Datos"
	4	El sistema redirige a una página con una serie de submenús donde se muestran varios datos y configuraciones del usuario
	5	El usuario hace los cambios pertinentes y pulsa el botón guardar cambios
	6	El sistema pide una verificación por correo electrónico, si está configurado, si no directamente guarda los cambios y recarga la página
Postcondición	El usuario ha cambiado la configuración de la cuenta	
Excepciones	Ninguna	

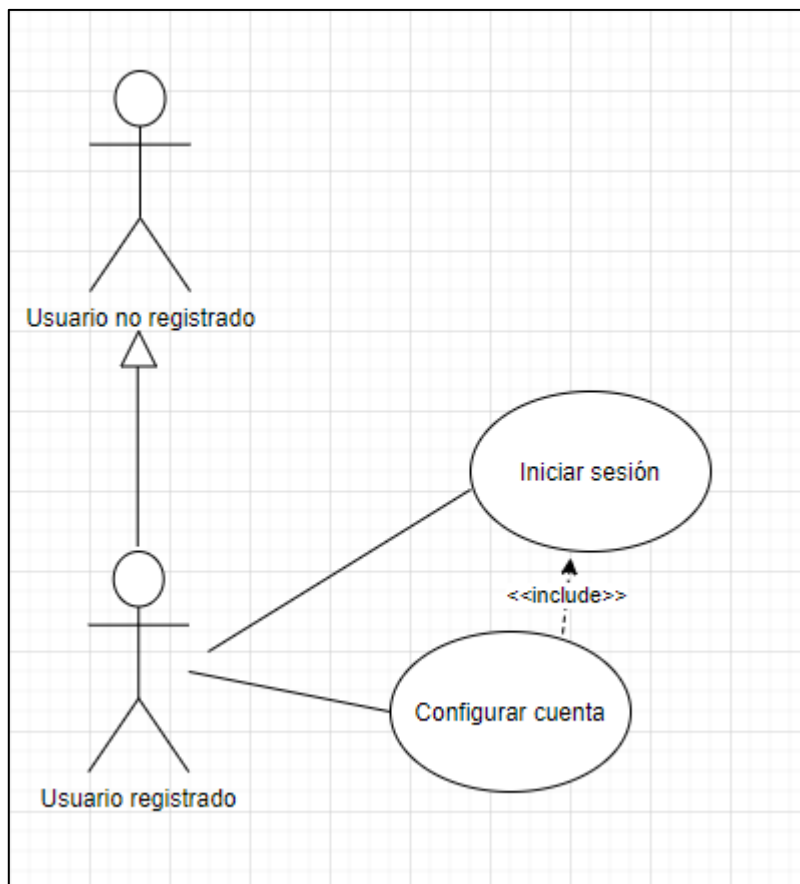


Figura 2- Diagrama de casos de uso para usuario registrado

3.2.3. Usuario editor

Nombre	Editar datos de una película	
Precondición	Ninguna	
Descripción	El usuario puede editar los datos de una película, siendo permanentes si el usuario es administrador o siendo necesaria una revisión si el usuario es editor	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de películas en el menú
	2	El sistema carga la lista de películas, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón editar
	4	El sistema carga una página con los datos de la película, similar a la página para ver los datos, salvo que los campos son editables
	5	El usuario cambia los datos necesarios y pulsa el botón para guardar
	6	El sistema comprueba los datos y el rol del usuario, si el usuario es editor, los cambios van a una tabla secundaria para ser revisados sin que cambien los datos que se ven, si es administrador se aplican directamente. Después se redirige a la página para ver los datos de la película
Postcondición	El usuario puede ver los cambios aplicados	
Excepciones	Si los datos son incorrectos, no se aplicarán cambios	

Nombre	Editar datos de una serie	
Precondición	Ninguna	
Descripción	El usuario puede editar los datos de una serie, siendo permanentes si el usuario es administrador o siendo necesaria una revisión si el usuario es editor	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de películas en el menú
	2	El sistema carga la lista de serie, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón editar
	4	El sistema carga una página con los datos de la serie, similar a la página para ver los datos, salvo que los campos son editables
	5	El usuario cambia los datos necesarios y pulsa el botón para guardar
	6	El sistema comprueba los datos y el rol del usuario, si el usuario es editor, los cambios van a una tabla secundaria para ser revisados sin que cambien los datos que se ven, si es administrador se aplican directamente. Después se redirige a la página para ver los datos de la serie
Postcondición	El usuario puede ver los cambios aplicados	
Excepciones	Si los datos son incorrectos, no se aplicarán cambios	

Nombre	Editar datos de un actor o actriz	
Precondición	Ninguna	
Descripción	El usuario puede editar los datos de un actor o actriz, siendo permanentes si el usuario es administrador o siendo necesaria una revisión si el usuario es editor	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de actores en el menú
	2	El sistema carga la lista de películas, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón editar
	4	El sistema carga una página con los datos del actor o la actriz, similar a la página para ver los datos, salvo que los campos son editables
	5	El usuario cambia los datos necesarios y pulsa el botón para guardar
	6	El sistema comprueba los datos y el rol del usuario, si el usuario es editor, los cambios van a una tabla secundaria para ser revisados sin que cambien los datos que se ven, si es administrador se aplican directamente. Después se redirige a la página para ver los datos del actor o la actriz
Postcondición	El usuario puede ver los cambios aplicados	
Excepciones	Si los datos son incorrectos, no se aplicarán cambios	

Nombre	Editar datos de un director o directora	
Precondición	Ninguna	
Descripción	El usuario puede editar los datos de un director o directora, siendo permanentes si el usuario es administrador o siendo necesaria una revisión si el usuario es editor	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de películas en el menú
	2	El sistema carga la lista de películas, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón editar
	4	El sistema carga una página con los datos del director o directora, similar a la página para ver los datos, salvo que los campos son editables
	5	El usuario cambia los datos necesarios y pulsa el botón para guardar
	6	El sistema comprueba los datos y el rol del usuario, si el usuario es editor, los cambios van a una tabla secundaria para ser revisados sin que cambien los datos que se ven, si es administrador se aplican directamente. Después se redirige a la página para ver los datos del director o directora
Postcondición	El usuario puede ver los cambios aplicados	
Excepciones	Si los datos son incorrectos, no se aplicarán cambios	



Nombre	Editar noticias	
Precondición	Ninguna	
Descripción	El usuario puede editar una noticia, siendo los cambios permanentes si el usuario es administrador o siendo necesaria una revisión si el usuario es editor	
Secuencia	1	El usuario pulsa el botón "Administrar" del apartado de noticias en el menú
	2	El sistema carga la lista de noticias, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón editar
	4	El sistema carga una página con la noticia, similar a la página para ver la noticia, salvo que los campos son editables
	5	El usuario cambia los datos necesarios y pulsa el botón para guardar
	6	El sistema comprueba los datos y el rol del usuario, si el usuario es editor, los cambios van a una tabla secundaria para ser revisados sin que cambien los datos que se ven, si es administrador se aplican directamente. Después se redirige a la página para ver la noticia
Postcondición	El usuario puede ver los cambios aplicados	
Excepciones	Si los datos son incorrectos, no se aplicarán cambios	

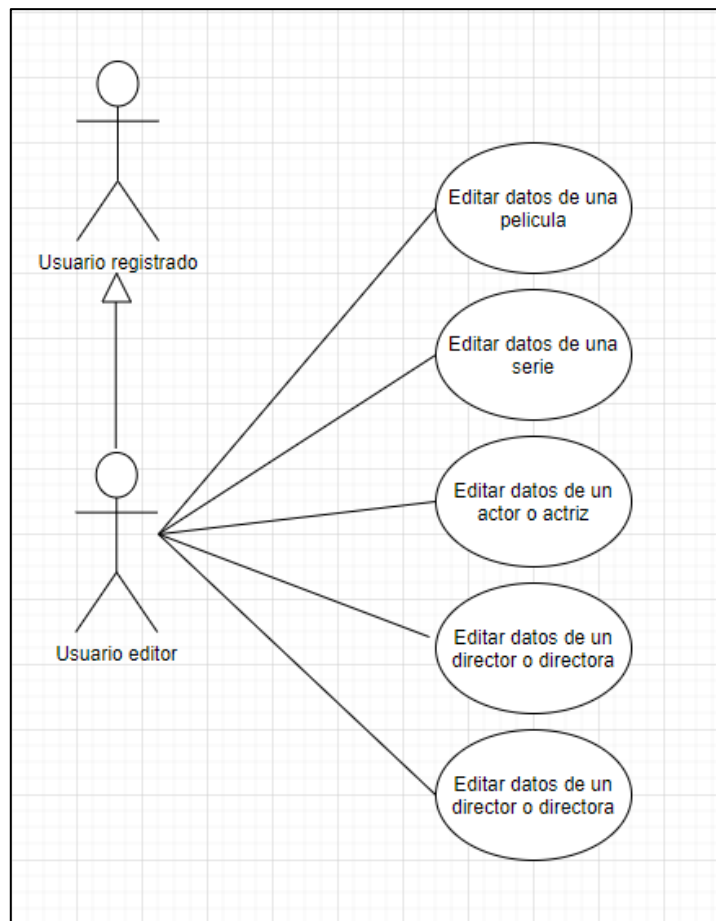


Figura 3- Diagrama de casos de uso para usuario editor

3.2.4. Usuario administrador

Nombre	Eliminar película	
Precondición	Ninguna	
Descripción	El usuario puede eliminar una película de la base de datos	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de películas en el menú
	2	El sistema carga la lista de películas, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón eliminar
	4	El sistema elimina el registro mediante el identificador en la base de datos, posteriormente, se elimina de la tabla y se recarga la página para aplicar los cambios.
Postcondición	El usuario puede eliminar, añadir o editar otras películas	
Excepciones	Ninguna	

Nombre	Eliminar serie	
Precondición	Ninguna	
Descripción	El usuario puede eliminar una serie de la base de datos	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de series en el menú
	2	El sistema carga la lista de series, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón eliminar
	4	El sistema elimina el registro mediante el identificador en la base de datos, posteriormente, se elimina de la tabla y se recarga la página para aplicar los cambios.
Postcondición	El usuario puede eliminar, añadir o editar otras series	
Excepciones	Ninguna	

Nombre	Eliminar actores o actrices	
Precondición	Ninguna	
Descripción	El usuario puede eliminar un actor o actriz de la base de datos	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de actores en el menú
	2	El sistema carga la lista de actores y actrices, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón eliminar
	4	El sistema elimina el registro mediante el identificador en la base de datos, posteriormente, se elimina de la tabla y se recarga la página para aplicar los cambios.
Postcondición	El usuario puede eliminar, añadir o editar otras películas	
Excepciones	Ninguna	



Nombre	Eliminar directores	
Precondición	Ninguna	
Descripción	El usuario puede eliminar un director o directora de la base de datos	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de directores en el menú
	2	El sistema carga la lista de directores, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón eliminar
	4	El sistema elimina el registro mediante el identificador en la base de datos, posteriormente, se elimina de la tabla y se recarga la página para aplicar los cambios.
Postcondición	El usuario puede eliminar, añadir o editar otros directores	
Excepciones	Ninguna	

Nombre	Eliminar noticias	
Precondición	Ninguna	
Descripción	El usuario puede eliminar una noticia de la base de datos	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de noticias en el menú
	2	El sistema carga la lista de noticias, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón eliminar
	4	El sistema elimina el registro mediante el identificador en la base de datos, posteriormente, se elimina de la tabla y se recarga la página para aplicar los cambios.
Postcondición	El usuario puede eliminar, añadir o editar otras noticias	
Excepciones	Ninguna	

Nombre	Añadir película	
Precondición	Ninguna	
Descripción	El usuario puede añadir una película a la base de datos	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de películas en el menú
	2	El sistema carga la lista de películas, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón “Añadir Película” que se encuentra debajo de la tabla
	4	El sistema abre una página similar a la que se carga para editar una película, pero con los datos vacíos
	5	El usuario rellena los datos y pulsa en el botón “Guardar”
	6	El sistema comprueba los datos y si están correctos añade los registros a la base de datos. Una vez hecho esto, se redirige a la página para ver los datos creados

Postcondición	El usuario puede eliminar, añadir o editar otras películas	
Excepciones	Ninguna	
Nombre	Añadir serie	
Precondición	Ninguna	
Descripción	El usuario puede añadir una serie a la base de datos	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de series en el menú
	2	El sistema carga la lista de series, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón “Añadir Serie” que se encuentra debajo de la tabla
	4	El sistema abre una página similar a la que se carga para editar una serie, pero con los datos vacíos
	5	El usuario rellena los datos y pulsa en el botón “Guardar”
	6	El sistema comprueba los datos y si están correctos añade los registros a la base de datos. Una vez hecho esto, se redirige a la página para ver los datos creados
Postcondición	El usuario puede eliminar, añadir o editar otras series	
Excepciones	Ninguna	

Nombre	Añadir actor o actriz	
Precondición	Ninguna	
Descripción	El usuario puede añadir un nuevo actor o actriz a la base de datos	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de actores en el menú
	2	El sistema carga la lista de actores, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón “Añadir Actor” que se encuentra debajo de la tabla
	4	El sistema abre una página similar a la que se carga para editar un actor o una actriz, pero con los datos vacíos
	5	El usuario rellena los datos y pulsa en el botón “Guardar”
	6	El sistema comprueba los datos y si están correctos añade los registros a la base de datos. Una vez hecho esto, se redirige a la página para ver los datos creados
Postcondición	El usuario puede eliminar, añadir o editar otro actor o actriz	
Excepciones	Ninguna	

Nombre	Añadir director o directora	
Precondición	Ninguna	
Descripción	El usuario puede añadir un director o directora a la base de datos	
Secuencia	1	El usuario pulsa el botón “Administrar” del apartado de directores en el menú
	2	El sistema carga la lista de directores, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.

	3	El usuario pulsa el botón "Añadir Director" que se encuentra debajo de la tabla
	4	El sistema abre una página similar a la que se carga para editar un director o directora, pero con los datos vacíos
	5	El usuario rellena los datos y pulsa en el botón "Guardar"
	6	El sistema comprueba los datos y si están correctos añade los registros a la base de datos. Una vez hecho esto, se redirige a la página para ver los datos creados
Postcondición	El usuario puede eliminar, añadir o editar otros directores	
Excepciones	Ninguna	

Nombre	Añadir noticia	
Precondición	Ninguna	
Descripción	El usuario puede añadir una noticia a la base de datos	
Secuencia	1	El usuario pulsa el botón "Administrar" del apartado de noticias en el menú
	2	El sistema carga la lista de noticias, así como un CRUD para editar o eliminar, este último solo disponible para usuarios administradores.
	3	El usuario pulsa el botón "Añadir Noticia" que se encuentra debajo de la tabla
	4	El sistema abre una página similar a la que se carga para editar una noticia, pero con los datos vacíos
	5	El usuario rellena los datos y pulsa en el botón "Guardar"
	6	El sistema comprueba los datos y si están correctos añade los registros a la base de datos. Una vez hecho esto, se redirige a la página para ver la noticia creada
Postcondición	El usuario puede eliminar, añadir o editar otras noticias	
Excepciones	Ninguna	

Nombre	Administrar cambios	
Precondición	Tienen que haber cambios hechos por usuarios editores	
Descripción	El usuario puede administrar los cambios hechos por usuarios editores para películas, series, directores, actores y noticias	
Secuencia	1	El usuario pulsa el botón "Administrar" en el menú
	2	El sistema carga la lista de cambios, así como un CRUD para administrar o eliminar.
	3	El usuario pulsa el botón "Administrar"
	4	El sistema abre una página en la que se muestran los datos actuales y los datos en el cambio concreto
	5	El usuario puede pulsar el botón "Aplicar" para cada uno de los cambios
	6	El sistema recoge los datos de la tabla de cambios y los aplica en la tabla principal
Postcondición	El usuario puede administrar otros cambios	
Excepciones	Ninguna	

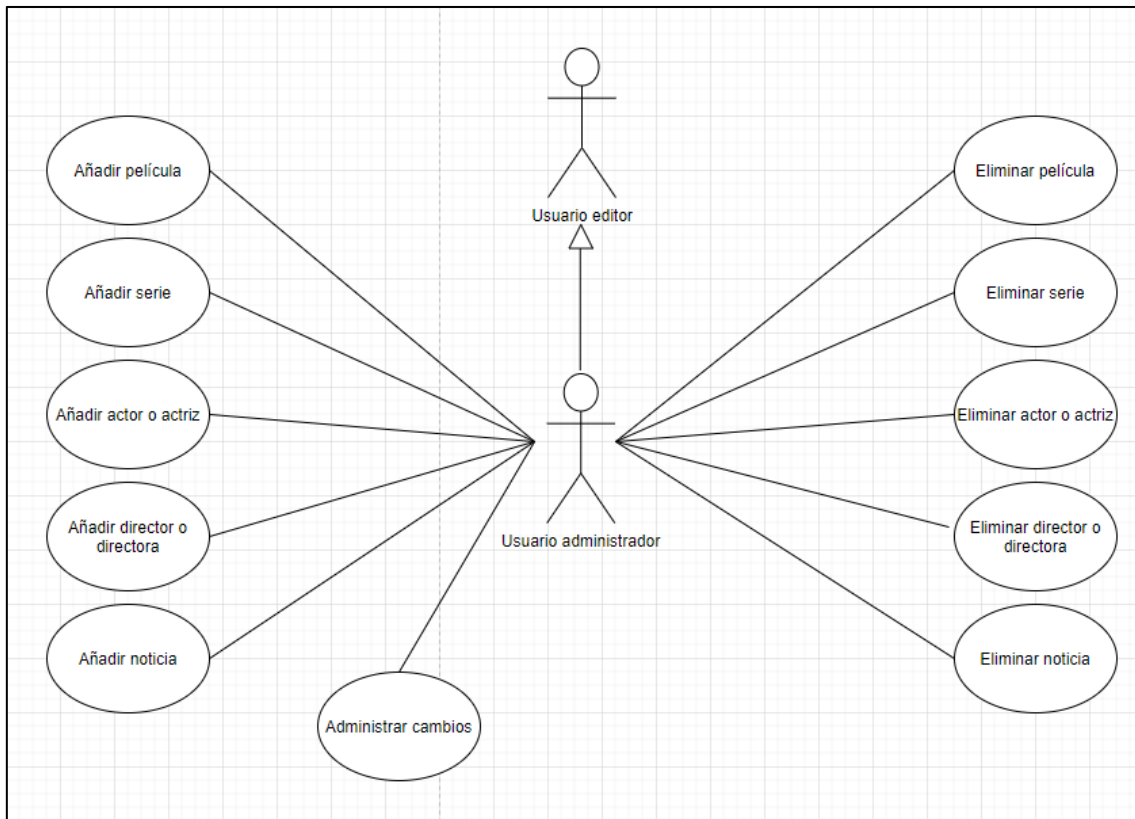


Figura 4- Diagrama de casos de uso para usuario administrador

3.3. Análisis de seguridad

La seguridad en este proyecto es un tema crucial, ya que, si hubiera una filtración de datos, el conjunto de usuario, correo electrónico, contraseña y datos personales puede producir una gran fuga de seguridad en otros sistemas, ya que, según un estudio de la empresa española de seguridad Panda Security, más del 52% de los usuarios reutilizan sus contraseñas (6). Esto puede provocar, por ejemplo, que, si un usuario puede acceder a las contraseñas y correos guardados en la base de datos, pueda acceder aproximadamente a la mitad de las cuentas de los correos electrónicos de los usuarios registrados en la aplicación.

Por lo mencionado anteriormente, los sistemas de seguridad deben ser revisados y ser tenidos en cuenta a la hora del desarrollo de esta aplicación. Dependiendo de la implementación que se lleve a cabo, se pueden optar por diversas opciones de seguridad, por lo que este dato lo comentaremos posteriormente.

3.4. Análisis del marco legal

Como en este proyecto se trabaja tanto con datos personales de usuarios como con datos sobre algunos medios que están sujetos a las leyes de propiedad intelectual, analizar las medidas legales que se deben tomar en cuenta es necesario, ya que entramos en un tema con una gran controversia hoy en día, el tratamiento de datos personales.

Por tanto, se va a dividir este análisis en dos secciones, una analizando la protección de datos de los usuarios y otra analizando las leyes de propiedad intelectual.

3.4.1. Análisis de protección de datos personales

Para el análisis de este tema, hay que dirigirse al Boletín Oficial del Estado, más concretamente a la **Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales** (7). Esta ley, como bien dice su nombre, trata sobre los derechos de las personas sobre sus datos personales. Dadas las necesidades del proyecto sobre los datos personales, se van a especificar los puntos más interesantes en este marco, que corresponden a los títulos II y III de esta ley, titulados *Principios de protección de datos* y *Derechos de las personas*, respectivamente.

3.4.1.1. Principios de protección de datos

Este apartado está formado por los artículos del número 4 al número 10, de los cuales nos interesan el artículo 4, el artículo 5, el artículo 6 y el artículo 7. (7)

El artículo 4 (7) hace referencia a la exactitud de los datos del usuario, lo cual no debe ser un problema, ya que es el propio usuario el que proporciona dichos datos, aunque se puede dar el remoto caso de un error en el sistema que tergiversen esos datos y acaben siendo totalmente inexactos. Para cumplir correctamente con este artículo, se debe asegurar que los datos se guardan correctamente en el sistema. Otro apartado de este artículo hace referencia a que el responsable debe permitir que estos datos sean rectificadas o suprimidos.

El artículo 5 (7) trata sobre el deber de confidencialidad de los datos, lo cual es algo crucial, ya que, en definitiva, la aplicación almacena datos, lo cual podría provocar una filtración, por lo que la confidencialidad, aun tratándose de un sistema, y no de una persona física, es de vital importancia, por lo que los datos que no correspondan a ese usuario deben ser lo más inaccesibles posible para las personas.

El artículo 6 (7) especifica que el tratamiento de los datos debe de ser consentido por el usuario, lo cual deriva en una cláusula de conformidad a la hora de la creación de la cuenta del usuario, ya que del usuario anónimo no se debe guardar ningún tipo de dato.

Por último, el artículo 7 (7) amplía lo tratado en el artículo 6, en referencia a los menores de edad. Se especifica que solo los mayores de 14 años pueden aceptar el consentimiento del tratamiento de sus propios datos, o, en caso de ser menor de esta edad, ser requerido el consentimiento de los titulares de la patria potestad o tutela. Pero debido al carácter de la red, en la que no es posible verificar si una persona tutela a dicho menor, la única limitación posible es pedir al usuario una verificación de que es mayor de 14 años para poder registrarse, ya que, en caso de no serlo, el responsable de la aplicación no está cometiendo ninguna ilegalidad respecto a la protección de datos.

Por tanto, después de analizar este apartado, se llega a la conclusión de que, para cumplir con la ley de protección de datos, hay que solicitar una conformidad con el

tratamiento de dichos datos, y en caso de que se acepte, solicitar una confirmación de que el usuario es una persona mayor de 14 años. En caso de que alguna de estas situaciones no se cumpla, no se puede crear una cuenta para dicho usuario.

Por otra parte, también hay que asegurar la confidencialidad de dichos datos, por lo cual se debe tener mucho cuidado sobre no mostrar ningún dato correspondiente a otras personas a dicho usuario. También es necesario cifrar ciertos datos personales más críticos, como puede ser la contraseña, ya que pueden darse muchos tipos de inconvenientes a la hora del almacenamiento de datos.

3.4.1.2. Derechos de las personas

En este apartado se especifican los derechos de las personas sobre sus datos, así como el ejercicio de dichos derechos por estas personas. (7)

Primero, en el capítulo uno se encuentra el artículo 11 (7), que trata sobre la transparencia e información al afectado. Aquí se separa en dos casos distintos, siendo el primero de ellos cuando los datos personales son obtenidos directamente del afectado, y el segundo cuando los datos personales son obtenidos de otras fuentes. Dado el funcionamiento de la aplicación, solo es interesante el primer caso, ya que no se obtienen datos de fuentes externas.

En este caso, el afectado tiene derecho a que le faciliten la información básica (responsable del tratamiento y la finalidad) sobre el tratamiento de sus datos, dándole opción mediante un correo electrónico u otros medios para acceder al resto de la información.

En el segundo capítulo, se encuentra una especificación de los derechos del afectado. Los derechos son: derecho de acceso, rectificación, supresión, limitación del tratamiento, portabilidad y oposición.

El derecho de acceso (Artículo 13) (7), hace referencia a la posibilidad que tiene el usuario afectado a poder visualizar los datos relativos a este. Aquí se especifican distintos casos en los cuales se puede cumplir este derecho. Primero, si el responsable trata una gran cantidad de datos del afectado y el afectado solicita acceso sin especificar si a todos o a una parte, el responsable puede pedirle al afectado una especificación sobre los datos. Otro caso, el cual es el más interesante para este proyecto, especifica que, si el responsable proporciona un sistema para acceder permanentemente a dichos datos de una forma remota y segura, se considera cumplido este derecho de acceso.

El derecho de rectificación (Artículo 14) (7), hace referencia a que el usuario afectado debe tener la posibilidad de que se rectifiquen diversos datos, debiendo acompañar, en caso de rectificación, la documentación pertinente si es necesario.

El derecho de supresión (Artículo 15) (7), indica que el usuario afectado tiene derecho a la eliminación de los datos personales a los que el responsable tenga acceso, estando ligado a diversas circunstancias. (8)



El derecho a la limitación del tratamiento (Artículo 16) (7), indica que el usuario tiene derecho a solicitar al responsable aplicar medidas para que no se puedan modificar o borrar los datos. (9)

El derecho a la portabilidad (Artículo 17) (7), hace referencia al derecho del usuario de poder recibir los datos personales de un responsable de tratamiento y transmitirlos a otro responsable.

El derecho de oposición (Artículo 18), hace referencia al derecho del usuario afectado para oponerse a que el responsable del tratamiento de los datos realice el tratamiento de dichos datos, salvo casos excepcionales.

Dados los derechos de las personas afectadas que se han especificado, se pueden sacar diversas conclusiones sobre el camino a seguir para poder cumplir con los derechos de los usuarios. Para cumplir los derechos de acceso, rectificación, limitación del tratamiento y de portabilidad, se proporcionará al usuario un sistema para poder acceder a sus datos permanentemente, como indica el apartado 2 del artículo 13. Este sistema también permitirá la edición de dichos datos para cumplir el derecho de rectificación, asegurando así el derecho a que el usuario pueda transmitir sus datos a otro responsable de tratamiento (derecho a la portabilidad) y asegurando que no se modifiquen o borren los datos sin el consentimiento del usuario (artículo 16). Aparte de esto, también se cumple el artículo 11 (transparencia e información al afectado).

Por otra parte, en la configuración de la cuenta del usuario, también se deben proporcionar 2 opciones. La primera de dichas opciones es la posibilidad para que los datos del usuario no sean utilizados por la aplicación, respetando así el derecho de oposición.

Por último, la otra opción es la posibilidad de la eliminación de la cuenta, incluyendo todos sus datos, lo que respeta el derecho de supresión.

3.4.2. Análisis de las leyes de propiedad intelectual

La ley de propiedad intelectual es un punto crítico de la aplicación, puesto que se trabaja con datos que pertenecen, normalmente a las productoras o a los directores de una película o una serie.

En este caso concreto, interesa hablar de la definición de los derechos de autor. Según el Real Decreto Legislativo 1/1996 (10), más concretamente al Capítulo II del título II del Libro Primero, las obras cinematográficas y otras obras audiovisuales son objeto de la propiedad intelectual, tratándose de obras y títulos originales (Artículo 10). En cuanto a quién corresponden dichos derechos, varía en cada caso, pero suelen repartirse la autoría entre los diversos procesos de creación, como los guionistas y los directores, aunque también puede haber varios acuerdos con otras entidades relacionadas con la producción de la película.

Por tanto, dado que las obras cinematográficas están sujetas a la Ley de propiedad intelectual antes mencionada, los beneficiarios de esta propiedad intelectual tienen unos derechos de explotación exclusiva bien especificados en el Capítulo III del título II del

Libro Primero (10). En la sección 2, del artículo 18 al 21 se especifican los derechos de reproducción, distribución, comunicación pública y transformación, de los cuales el que influye en el proyecto en este caso es el derecho de comunicación pública, imposibilitando a cualquier persona que no sea titular de dichos derechos el poder transmitir en un ámbito público ningún tipo de dato que esté sujeto a estas leyes, lo que incluye a los avances y las portadas de las películas o las series.

Por tanto, el uso de imágenes o videos de las películas está limitado por estas leyes, dado que están sujetos a los derechos cinematográficos. Aunque hay varias formas para poder incluir datos sobre estas películas o series sin incumplir los derechos de autor. Aquí entran en juego varios conceptos, uno de ellos es una doctrina legal utilizada en Estados Unidos llamado *fair use* (11), aplicado sobre el copyright. Otro concepto son las licencias *Creative Commons* (12), que permite el uso de los datos de obras protegidas sin solicitar permiso al autor de la obra. Se ha optado por centrarse en el último punto, ya que el *fair use* es una doctrina legal sujeta a interpretación, por lo cual complicada de entender y aplicar correctamente.

En este caso particular, se pueden utilizar los datos de la página *Open Movie Database*, una alternativa a *IMDb* más simple y con una licencia como la que se ha comentado anteriormente. En su sitio web, en el apartado “Acerca de”, especifica lo siguiente: *omdb es un proyecto no comercial. Todos los contenidos del sitio se publican bajo la licencia Creative Commons. Por tanto, el banco de datos representa una contribución al conocimiento gratuito en Internet. La información no pertenece a una empresa o persona, sino al público en general. Omdb conscientemente se mantiene alejado de la información comercial relacionada con el objetivo de proporcionar una base de datos de películas autónoma.* (13)

En resumen, el análisis de la ley de propiedad intelectual es un punto que no es necesario comentar extensamente, al menos en lo relacionado con el Copyright, ya que es de conocimiento general el hecho de que no se pueden usar datos sin el permiso del autor o del propietario de dichos derechos, por lo que se ha optado por obtener los datos de una forma libre y gratuita mediante uno de los muchos proyectos con licencias libres que se encuentran hoy en día en internet.

Por otra parte, lo relacionado con contenido multimedia como puede ser un *trailer*, es de fácil solución, ya que en la plataforma YouTube (Donde prácticamente todos los distribuidores de películas o series suben avances o *trailers*), hay una opción para añadir un vídeo incrustado utilizando la API de YouTube y, por tanto, el video original de este. (14)



3.5. Identificación y análisis de distintas soluciones

Tras analizar los requisitos, se presentan distintas soluciones para llevar a cabo el proyecto, teniendo cada una sus pros y sus contras. Estas soluciones derivan en la utilización de distintas tecnologías para el desarrollo de una aplicación.

Principalmente se encuentran dos soluciones muy distintas entre sí, y es el desarrollo de una aplicación web o una aplicación de software de escritorio.

3.5.1. Aplicación de escritorio

Para el desarrollo de una aplicación de escritorio es muy importante elegir el lenguaje de programación. Dada la naturaleza de la aplicación, es necesaria una interfaz gráfica que haga simple la utilización de dicha aplicación, por lo que pueden ser interesantes lenguajes como Java (o JavaFX) o C#, ya que proporcionan herramientas sencillas para la creación de interfaces. (15)

Otra característica importante de la aplicación es la multiplataforma, ya que la idea es que cualquier usuario, independientemente del sistema que utilice, tenga acceso a la utilización de esta aplicación, por lo que aquí, siguiendo las recomendaciones anteriores, Java es una opción viable, ya que, a pesar de que C# también proporciona sistemas para aplicaciones multiplataforma, en Java esto es implícito, y te asegura la utilización en cualquier sistema, ya que Java se ejecuta en su propia máquina virtual (16), siendo utilizada de intermediario entre el lenguaje y el dispositivo.

Dado que todo el proyecto se sustenta sobre una base de datos relacional de películas, series y actores/actrices, es necesario un sistema para crear y administrar bases de datos. Un sistema que permite generar y administrar una base de datos para una aplicación de escritorio es SQLite, que es compatible con muchos lenguajes de programación, entre los que entran los mencionados anteriormente (17). Pero se encuentra un problema que puede llevar a una implementación complicada, y es la necesidad de mantener actualizada la base de datos, así como la posibilidad de que los usuarios puedan editar o añadir datos. Esto implica un problema ya que es necesario un sistema para que las bases de datos locales de cada usuario se sincronicen, lo que puede provocar grandes inconsistencias.

Una característica que solo puede proporcionar este tipo de implementación es que la disponibilidad del sistema no depende de un sistema central, ya que cada usuario tiene una copia exacta del contenido, por lo que permite que, independientemente de un servidor que puede perder conexión, o incluso sin tener conexión el propio sistema donde se está utilizando la aplicación (no es recomendable, ya que como se ha comentado anteriormente, si una base de datos se desincroniza, a la hora de hacer cambios se puede provocar una inconsistencia en la base de datos).

A la hora del almacenamiento de datos, existen ventajas e inconvenientes. Como se ha comentado anteriormente, el hecho de que sea una aplicación de escritorio permite que todas las fuentes de multimedia sean reproducidas localmente, lo que puede ser una ventaja para aquel que necesite acceder sin conexión, o una desventaja, ya que, al tener

todos los datos almacenados localmente, eso puede producir un uso masivo de la memoria del sistema.

3.5.2. Aplicación web

Para el desarrollo de la aplicación web, al igual que para el desarrollo de una aplicación de escritorio, hay que analizar los requisitos para elegir unas tecnologías u otras.

En el entorno del desarrollo web, hay dos metodologías para realizar un sitio web, que son el desarrollo a medida y el desarrollo mediante un CMS.

3.5.2.1. Content Manager System

Los CMS (siglas de *Content Manager System*), son aplicaciones web que permiten gestionar contenidos digitales, proporcionando un sistema de personalización al gusto del usuario, siendo normalmente editores WYSIWYG, acrónimo de *What You See Is What You Get*, es decir, que permite una edición visual de los datos que administra el sistema. (18)

Estos sistemas tienen una serie de ventajas muy claras, ya que permiten que un usuario sin experiencia en programación o diseño pueda administrar un sitio web, teniendo muchos de estos sistemas para implementar plugin, que añaden diversas funcionalidades. Entre los más famosos podemos encontrar *Blogger* o *Wordpress*, u otros con algunas funcionalidades más avanzadas como Joomla.

En lo que al desarrollo del proyecto se refiere, el CMS es una opción interesante, ya que permite un desarrollo multiplataforma, una disponibilidad bastante alta, un sistema de almacenamiento de datos de una forma simple y segura y la posibilidad de desarrollar una interfaz gráfica dentro de los baremos que permite la aplicación, así como diversos plugin para implementar un sistema de privilegios de usuarios.

Las desventajas que tiene son también varias, siendo una de las principales que, dentro de toda la personalización que puede haber en un sistema CMS, siempre tendrá que seguir los estándares que la aplicación haya marcado, por lo que se pierde un grado de personalización que, dependiendo de cada proyecto, puede ser crucial. Otro inconveniente es el hecho de que, a pesar de que los plugins proporcionan funcionalidades muy interesantes a la hora del desarrollo, el abanico de posibilidades es mucho menor al que se obtiene con una aplicación desarrollada a medida. Y, por último, está el hecho de que, a pesar de que hay CMS gratuitos, muchos de ellos, sobre todo los más avanzados, tienen un coste, ya sea de alojamiento o para poder usar la aplicación.

3.5.2.2. Desarrollo a medida

Esta metodología es la más clásica para el desarrollo de aplicaciones web, ya que permite a los usuarios experimentados desarrollarla como deseen, pudiendo cambiar cualquier tipo de detalle de esta.



Dentro del desarrollo a medida hay 3 tecnologías que son totalmente indispensables, HTML, CSS y JavaScript, incluyendo dentro de este lenguaje bibliotecas como JQuery.

HTML son las siglas de *HyperText Markup Language*, es un lenguaje, además de un estándar, basado en etiquetas que nos permite desarrollar la estructura de una página web. (19)

CSS (*Cascading Style Sheets*) es un lenguaje que se utiliza para definir o cambiar los estilos de los lenguajes basados en marcas como HTML, y, al igual que HTML, el *World Wide Web Consortium* es el encargado de mantener su especificación. (19)

JavaScript es un lenguaje de programación orientado a objetos, siendo utilizado en el ámbito de la programación web para añadir funcionalidades a las páginas web del lado del cliente, es decir, se ejecuta por parte del navegador web del usuario. Una biblioteca muy importante dentro de *JavaScript* es *jQuery*, que se suele utilizar como una implementación más sencilla de JavaScript orientada exclusivamente al desarrollo de páginas web, incluyendo, aparte de la manipulación de DOM, un sistema de interacciones llamado AJAX, que es muy útil para obtener datos mediante la ejecución de código en la parte del servidor. (20)

Por la parte del servidor, hay 2 tecnologías necesarias para el desarrollo de un proyecto de estas características, un lenguaje que ejecute funciones en el servidor y una base de datos relacional. En estos casos podemos usar PHP y MySQL, dos opciones muy utilizadas en este tipo de metodología.

PHP es un lenguaje de programación que, en el ámbito de la programación web, se utiliza para la creación de páginas web dinámicas (21), ya que, al contrario de JavaScript, que se ejecuta en el cliente mientras se cargan los datos, se ejecuta en el lado del servidor, pudiendo crear dinámicamente diversas estructuras HTML o *JavaScript*, y luego el cliente lo interpreta como parte del código de la página. Otra funcionalidad de PHP es, mediante AJAX, poder cambiar dinámicamente el contenido de una página web, ejecutando en un documento aparte un código PHP y luego comunicando el resultado de esa ejecución por HTTP gracias a la petición AJAX. MySQL es un sistema de gestión de bases de datos, muy utilizado dada su naturaleza de base de datos de código abierto. Este sistema es muy popular para el desarrollo web ya que, aparte de ser multiplataforma, lenguajes del lado del servidor como PHP incluyen librerías nativas para ver y administrar los datos de estas bases de datos. (22)

En referencia a la implementación del proyecto, hay una serie de ventajas que proporciona este tipo de metodología. HTML, CSS y *JavaScript* son unas herramientas para el desarrollo de interfaces gráficas muy potentes y versátiles, ya que permiten que un usuario con conocimientos en estas pueda hacer una interfaz completamente a su gusto, pudiendo implementar, gracias a la gran comunidad que hay en la red sobre estas tecnologías, una gran cantidad de elementos con funcionalidades muy diversas.

Otra ventaja que tiene es que, al tratarse de un estándar de los navegadores web, todos los navegadores actuales deben ser compatibles con HTML5, CSS3 y *JavaScript*, por lo que el sistema es completamente multiplataforma, sin necesidad de librerías extras o máquinas virtuales como en el caso de JAVA.

Por otra parte, una problemática encontrada en el desarrollo de una aplicación de escritorio es el hecho de que, al tratarse de una base de datos donde los usuarios pueden hacer cambios de dichos datos, el tener una copia de dicha base de datos de forma local en el sistema se convertía en un problema, ya que había que encontrar una manera de que se sincronizaran las bases de datos cada poco tiempo para no provocar inconsistencias. Al tratarse de una aplicación web, se desarrolla un modelo cliente-servidor, donde la base de datos está ubicada en el servidor, por lo que solo habría una base de datos, lo que simplifica mucho el hecho de la sincronización de los datos.

Un posible inconveniente que se puede encontrar en esta implementación es que la disponibilidad de la aplicación depende exclusivamente de donde elija el administrador que se ubique esta, ya que se puede ubicar tanto en un ordenador personal, haciendo función de servidor, lo cual provocaría que el administrador debiese tener encendido el sistema constantemente, como en un sistema de hosting, que delegaría las funciones de disponibilidad a terceros, minimizando la problemática de la disponibilidad, pero derivando en una serie de gastos.

3.6. Solución propuesta

Expuestas las distintas posibles soluciones para desarrollar el proyecto, y dada la naturaleza de este, deriva en muchas ventajas el desarrollar la aplicación como aplicación web y no una aplicación de escritorio, ya que dado el carácter colaborativo que tiene la aplicación, el tener que desarrollar un servicio web aparte para poder sincronizar la base de datos local con una base de datos centralizada es una complicación que se puede evitar mediante la aplicación web.

Por otra parte, para la implementación de la aplicación web se pueden seguir dos metodologías distintas, los CMS o el desarrollo a medida. Como se ha visto, ambos tienen unas ventajas y unos inconvenientes equilibrados, ya que el CMS permite la administración de contenidos sin necesidad de conocimientos de desarrollo web, pero sacrificando en el proceso la capa de personalización que sólo se puede obtener con un desarrollo a medida, ya que, tal como se ha comentado antes, a pesar de que los CMS incorporan una gran cantidad de plugins, cabe la posibilidad de que para una característica concreta que se quiera implementar como, en el caso de este proyecto, las recomendaciones personalizadas, no haya un plugin concreto.

Por tanto, la solución elegida para el desarrollo del proyecto es un desarrollo web a medida, ya que, con unos conocimientos técnicos medios y gracias a la extensa comunidad de internet, se puede implementar prácticamente cualquier idea que el administrador necesite.

Para el desarrollo, se van a utilizar las tecnologías antes mencionadas, es decir: HTML, CSS, *JavaScript*, PHP y *MySQL*. Dentro de algunas de estas tecnologías se encuentran algunas librerías o implementaciones concretas, como en el caso de CSS y Bootstrap. Bootstrap es un conjunto de clases de estilos que proporcionan ciertas funcionalidades extra a las proporcionadas de forma nativa por CSS, es decir, estas características se pueden implementar nativamente mediante CSS, pero Bootstrap proporciona las clases con estas características ya implementadas. Una de las características más conocidas



es el *Grid*, que es un sistema de organización de los contenidos de una página en una plantilla que ajusta el ancho de las columnas en las que se organiza de manera variable. Otras características que puede implementar son algunos plugins de JavaScript, aunque no son muy recomendables, ya que, al ser implementadas por Bootstrap, pueden interferir con los plugin que utiliza el desarrollador de la aplicación.

Antes de comenzar el desarrollo de la aplicación, se desarrollarán una serie de bocetos de las interfaces del usuario, así como la interacción con estas para realizar acciones concretas, además de la especificación del sistema de administración de los datos mediante un diagrama UML.

Para validar el funcionamiento de la aplicación, se comprobarán todos los requisitos funcionales, especificando unas pruebas concretas que, para forzar un error a la hora de especificar los datos, para comprobar si los errores en la aplicación son tratados o no se ejecuta ninguna acción en respuesta.

3.7. Plan de trabajo

En este punto hay que desarrollar un plan de trabajo para hacer una aproximación del coste de trabajo que puede conllevar el desarrollo del proyecto, pudiendo así visualizar cómo se está desarrollando en temas de efectividad.

Para esto se va a utilizar la técnica del diagrama de Gantt, que consiste en la división de las tareas en subtareas e identificar las dependencias que puede haber entre estas, para así poder aproximar tanto el coste temporal como económico que conlleva desarrollar el proyecto. En este caso en particular se ha decidido centrarse únicamente en el coste temporal, ya que para el coste económico ya hay que suponer cierto grupo de personas, con un coste tanto de hardware y software como del propio trabajo de cada una de las personas.

La planificación temporal está construida en base a los siguientes datos:

- Solo hay un usuario para la realización del proyecto
- El horario es de 15:00 a 20:00
- El proyecto comienza el 1 de junio de 2021
- La jornada es de lunes a domingo

Al utilizar dependencias, hay algunas tareas que no se pueden realizar simultáneamente, ya que para realizarlas necesitas haber realizado anteriormente una tarea anterior, aunque teóricamente, aunque no se pusieran dependencias, al desarrollar únicamente una persona todo el proyecto, no debería ser posible realizar ninguna tarea simultáneamente con otra.

Task name	Start date	End date	Duration
	01.06.20...	09.07.20...	192h
1 <input type="checkbox"/> Base de datos cinematográfica con sistema de recomendaciones por perfiles de usuari...	01.06.20...	09.07.20...	192h
1.1 <input type="checkbox"/> Analisis	01.06.20...	09.07.20...	192h
1.1.1 <input type="checkbox"/> Analisis de requisitos	01.06.20...	09.07.20...	192h
1.1.1.1 Diagramas UML	08.07.20...	09.07.20...	3h
1.1.1.2 Seleccion de requisitos funcionales y no funcionales	08.07.20...	08.07.20...	4h
1.1.1.3 Diagramas de casos de uso	01.06.20...	01.06.20...	3h
Add a task Add a milestone			
1.1.2 <input type="checkbox"/> Validez	07.07.20...	07.07.20...	3h
1.1.2.1 Elección de los requisitos finales de la aplicación	07.07.20...	07.07.20...	3h
Add a task Add a milestone			
Add a task Add a milestone			
1.2 <input type="checkbox"/> Diseño	01.06.20...	07.07.20...	182h
1.2.1 <input type="checkbox"/> Capa de presentación	02.07.20...	07.07.20...	25h
1.2.1.1 Diseño grafico de la aplicación web	04.07.20...	07.07.20...	15h
1.2.1.2 Prototipo	02.07.20...	04.07.20...	10h
Add a task Add a milestone			
1.2.2 <input type="checkbox"/> Capa de negocio	01.06.20...	05.07.20...	171h
1.2.2.1 Diagrama de Clases	04.07.20...	05.07.20...	6h
1.2.2.2 Diseño de comunicaciones	01.06.20...	01.06.20...	2h
Add a task Add a milestone			
1.2.3 <input type="checkbox"/> Capa de datos	02.07.20...	03.07.20...	8h
1.2.3.1 Diseño de la BD	02.07.20...	03.07.20...	8h

Figura 5- Planificación temporal

1.3 <input type="checkbox"/> Implementación	11.06.20...	02.07.20...	103h	0
1.3.1 <input type="checkbox"/> Informe de las herramientas de desarrollo	01.07.20...	02.07.20...	3h	0
1.3.1.1 Creacion del informe de las herramientas	01.07.20...	02.07.20...	3h	0
Add a task Add a milestone				
1.3.2 <input type="checkbox"/> Software desarrollado	11.06.20...	01.07.20...	100h	0
1.3.2.1 Aplicación para clientes	11.06.20...	01.07.20...	100h	0
Add a task Add a milestone				
Add a task Add a milestone				
1.4 <input type="checkbox"/> Verificación y pruebas	06.06.20...	11.06.20...	28h	0
1.4.1 <input type="checkbox"/> Pruebas de la aplicación	07.06.20...	11.06.20...	20h	0
1.4.1.1 Pruebas generales	07.06.20...	11.06.20...	20h	0
Add a task Add a milestone				
1.4.2 <input type="checkbox"/> Pruebas de aceptación	06.06.20...	07.06.20...	8h	0
1.4.2.1 Gestión de pruebas con usuarios	06.06.20...	07.06.20...	6h	0
1.4.2.2 Analisis de opiniones de usuarios reales	06.06.20...	06.06.20...	2h	0
Add a task Add a milestone				
Add a task Add a milestone				
1.5 <input type="checkbox"/> Despliegue e implementación	05.06.20...	06.06.20...	2h	0
1.5.1 Despliegue de la aplicación	05.06.20...	06.06.20...	2h	0

Figura 6- Planificación temporal



Base de datos cinematográfica con sistema de recomendaciones por perfiles de usuarios

1.6	<input type="checkbox"/> Mantenimiento	01.06.20...	05.06.20...	24h	0
1.6.1	<input type="checkbox"/> Corrección de errores	01.06.20...	05.06.20...	24h	0
1.6.1.1	Solución de errores de funcionamiento	01.06.20...	05.06.20...	20h	0
1.6.1.2	Solucion de errores en la interfaz	01.06.20...	02.06.20...	3h	0
1.6.1.3	Pruebas de los arreglos	01.06.20...	01.06.20...	4h	0
Add a task Add a milestone					
1.6.2	<input type="checkbox"/> Mejora del sistema	01.06.20...	05.06.20...	23h	0
1.6.2.1	Mejoras en interfaz	01.06.20...	02.06.20...	7h	0
1.6.2.2	Nuevas funcionalidades	01.06.20...	05.06.20...	20h	0
1.6.2.3	Optimizacion de codigo	01.06.20...	02.06.20...	5h	0
1.6.2.4	Pruebas de las mejoras	01.06.20...	01.06.20...	3h	0
Add a task Add a milestone					
1.6.3	<input type="checkbox"/> Administracion	01.06.20...	02.06.20...	10h	0
1.6.3.1	Administracion de costes	02.06.20...	02.06.20...	5h	0
1.6.3.2	Elección de personal para administrar cambios de los datos	01.06.20...	01.06.20...	5h	0

Figura 7- Planificación temporal

Como se puede ver en el esquema del proyecto, el desarrollo de la aplicación tiene un coste temporal de alrededor de 200 horas, aunque realmente es una aproximación, posteriormente en la memoria se mostrará el coste temporal final del desarrollo del proyecto.

4. Diseño de la solución

Una vez seleccionados los requisitos, la solución elegida según las características del proyecto, la planificación temporal y otros aspectos del análisis, se debe especificar el diseño de la solución, es decir, como se va a implementar la solución escogida anteriormente. Dentro de esta especificación se incluye tanto la arquitectura que va a seguir el sistema como las tecnologías usadas.

4.1. Tecnologías utilizadas

Las tecnologías utilizadas para este proyecto, dada la solución elegida anteriormente, derivan del desarrollo web, por lo que están divididas en varios módulos conectados entre sí, ya que el sistema se ejecuta inicialmente en el navegador web del cliente, que cuando lo necesita, ejecuta comandos en el lado del servidor, que a su vez se conecta a la base de datos guardada dentro de ese servidor.

Por tanto, se diferencian las tecnologías en 3 grandes bloques, tecnologías del lado del cliente, tecnologías del lado del servidor y administración de datos.

4.1.1. Tecnologías del lado del cliente

Las tecnologías del lado del cliente son aquellas que el navegador web desde el que se accede a la aplicación ejecuta en el propio sistema cliente, estas tecnologías son: HTML, JavaScript y CSS.

4.1.1.1. HTML

Como se ha comentado anteriormente, HTML es un lenguaje de marcas que sirve para la estructuración de las páginas web. Las marcas se utilizan para distinguir los distintos elementos HTML, que se escriben como “<etiqueta>”, teniendo cada uno de estos elementos unos atributos y unas funcionalidades propias. (19)

Dentro de HTML existe una API llamada *DOM* (23) (Modelo de Objetos del Documento), que define la estructura de los documentos, pudiendo acceder, crear o editar objetos en el documento HTML. Se utiliza constantemente en JavaScript para añadir funciones extras a las que la propia definición de elementos y atributos en HTML permite. (19)

HTML está estandarizado por la *World Wide Web Consortium* (W3C), una organización que estandariza muchas tecnologías ligadas a la web, y actualmente el estándar ha evolucionado hasta HTML5.



4.1.1.2. CSS

CSS es un lenguaje que se utiliza para definir estilos de lenguajes de marcado como HTML, permitiendo cambiar muchas de las propiedades visuales de los elementos HTML, así como especificar dichas propiedades por clases, identificadores de elementos, nombres o etiquetas.

Al igual que HTML, está estandarizado por la *World Wide Web Consortium* y actualmente se utiliza el estándar CSS3. (19)

Además de permitir cambiar las propiedades visuales de los elementos, también permite crear animaciones para estos, así como estructurarlos de una forma determinada o cambiar su tamaño.

Para estilos CSS existe una biblioteca llamada Bootstrap¹, que se basa en HTML, CSS y *JavaScript*, implementando una serie de plantillas para diversos objetos con estilos determinados, normalmente utilizados mediante clases. (24)

Bootstrap, como reclamo principal, incorpora un sistema llamado *Grid*, que es muy utilizado en el entorno del desarrollo de aplicaciones web para permitir que se puedan redimensionar en anchura para ajustarse a los tamaños de las pantallas de los distintos dispositivos que puedan ser utilizados para acceder a la aplicación.

Otra funcionalidad muy interesante es que incluye algunas clases que implementan automáticamente plugin de *JavaScript*, permitiendo la utilización sencilla de algunos de ellos, pero no suelen ser muy recomendables ya que en ciertos casos pueden tener conflictos con plugin incorporados por el propio usuario, causando la mala visualización de la aplicación.

La versión utilizada para este proyecto es Bootstrap v4.5, que no es la más actualizada, pero hay ciertos proyectos de la comunidad que no son compatibles con algunas de las clases utilizadas en Bootstrap v5.0, por lo que se ha optado por esta versión.

4.1.1.3. JavaScript

JavaScript (20) es un lenguaje muy popular para el desarrollo web, ya que prácticamente cualquier página implementa algún tipo de código de este tipo. Una de las ventajas de este lenguaje es que implementa muchos métodos para interactuar con el DOM de HTML, permitiendo crear funciones para cambiar o añadir elementos del documento HTML en el que están escritos, así como crear eventos interactivos en el documento.

Otra razón por la cual es tan popular es que, a lo largo de la comunidad de internet, se han desarrollado una gran cantidad de plugin basados en *JavaScript* que añaden funcionalidades extra a las páginas web, lo que lo convierte prácticamente en un estándar a la hora de desarrollar aplicaciones.

Una biblioteca muy importante dentro de *JavaScript* es *jQuery*², que se suele utilizar como una implementación orientada exclusivamente al desarrollo de páginas web,

¹ <https://getbootstrap.com/>

² <https://jquery.com/>

incluyendo, aparte de unos métodos más sencillos para la manipulación del DOM, un sistema de interacciones llamado AJAX, que es muy útil para obtener datos mediante la ejecución de código en la parte del servidor.

A día de hoy, *jQuery* se utiliza en muchos ámbitos, por ejemplo, plugins desarrollados para páginas web se ejecutan en los distintos elementos de HTML mediante métodos creados para *jQuery*.

Otra de las razones para que *jQuery* haya sido tan popular es que, hasta la versión 5.0, Bootstrap ha utilizado este lenguaje como base para ejecutar ciertas características en lugar de JavaScript.

4.1.2. Tecnologías del lado del servidor

Las tecnologías del lado del servidor son aquellas que se ejecutan en el servidor en lugar del navegador web del cliente. En este caso en concreto vamos a usar 1, siendo esta una herramienta que implementa un conjunto de herramientas.

4.1.2.1. XAMPP

XAMPP³ es un paquete compuesto por software libre, en concreto un sistema de gestión de bases de datos *MySQL*, un servidor web *Apache* y los elementos necesarios para poder ejecutar código PHP en el servidor. (25)

Para este proyecto se utiliza XAMPP versión 8.0.3, que incluye Apache en su versión 2.4.46 para Windows de 64 bits, PHP versión 8.0.3 y una base de datos *MariaDB* en su versión 10.4.18, siendo *MariaDB*⁴ una ramificación del proyecto original *MySQL*, pero con licencia GPL.

También se incluye *FileZilla Server*, un administrador de archivos FTP; Mercury y Tomcat. Para la administración de la base de datos se utiliza *phpMyAdmin*⁵, un sistema programado en PHP que simplifica la tarea de administrar la base de datos.

Dentro de las herramientas que incluye XAMPP, se deben especificar las más importantes para el proyecto, excluyendo a *MySQL*, que son PHP y Apache.

PHP⁶ es un lenguaje de programación muy utilizado en la programación web, ya que se puede implementar dentro de un documento HTML indicando siempre en la extensión del fichero que contiene código PHP. El código escrito en este lenguaje se ejecuta del lado del servidor, por lo que, antes de que el navegador web del cliente intérprete el código HTML y *JavaScript*, se ejecuta este código, lo que permite crear páginas web dinámicas al igual que JavaScript, aunque sirven para dos cosas totalmente distintas,

³ <https://www.apachefriends.org/es/index.html>

⁴ <https://mariadb.org/>

⁵ <https://www.phpmyadmin.net/>

⁶ <https://www.php.net/>



ya que JavaScript se utiliza para cambiar propiedades de los elementos del DOM utilizando una serie de condiciones o métodos.

Otro de los usos que tiene este lenguaje es que, utilizando las funciones AJAX que implementa *jQuery*, permite cambiar dinámicamente el contenido del documento mediante métodos *JavaScript* sincronizados con acciones en los elementos HTML.

PHP también es útil a la hora de realizar consultas a las bases de datos, ya que nativamente implementa unos métodos para contactar con bases de datos administradas por *MySQL*.

*Apache*⁷ es un servidor HTTP multiplataforma, que permite el acceso a los documentos de una página web mediante dicho protocolo, tanto localmente como remotamente, siempre que tengamos encendido dicho servidor y abiertos los puertos pertinentes en el servidor (26).

Aparte de hacer la función de servidor, también incluye algunas herramientas de configuración, así como un sistema de autenticación.

4.1.3. Administración de datos

Como se indica en el apartado anterior, la base de datos utilizada es gestionada mediante un SGDB *MariaDB* (25), una ramificación de *MySQL* con licencia GNU, pero, al ser una ramificación, posee las mismas órdenes y bibliotecas, lo que la convierte en muy compatible con las órdenes *MySQL* de PHP. La versión de *MariaDB* incluida en la versión utilizada para este proyecto de XAMPP es la versión 10.4.18.

Para administrar las tablas y ciertos contenidos al inicio del proyecto es necesario una aplicación que permita administrar bases de datos *MySQL*, que en este caso ha sido *phpMyAdmin*, una aplicación escrita en PHP que permite la administración de dichas bases mediante un portal web.

4.2. Arquitectura del sistema

La arquitectura del sistema, al tratarse de una aplicación web, será un modelo cliente-servidor con varios componentes, ya que dentro del cliente intervienen varios módulos, incluyendo también la intervención del usuario, al igual que en el servidor, donde también hay incorporados varios módulos.

Los componentes que se encuentran en la arquitectura del sistema son: cliente, navegador web, servidor web, base de datos.

⁷ <https://httpd.apache.org/>

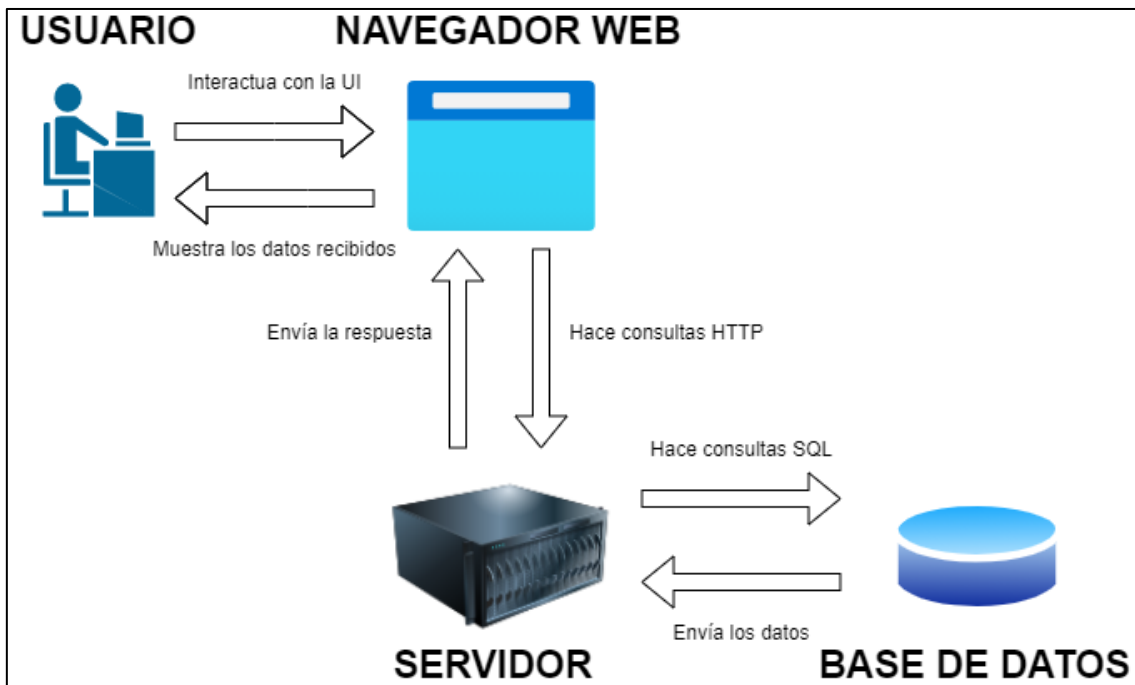


Figura 8- Arquitectura del Sistema

Como se puede ver, la estructura es la siguiente. El usuario interactúa con la interfaz de usuario mostrada en el navegador web, cuando esto ocurre, el navegador web envía una consulta HTTP al servidor, que envía la respuesta. Cuando el navegador web recibe la respuesta, muestra los datos recibidos, volviendo a comenzar el ciclo.

Dado que en la arquitectura del sistema se incluye una base de datos, hay otra posible vía de ejecución. Al igual que en el caso anterior, el usuario interactúa con la interfaz de usuario, haciendo una consulta HTTP al servidor. Cuando el servidor recibe la consulta, hace una consulta SQL a la base de datos, devolviendo esta los datos pertinentes que ha solicitado el servidor. Una vez recibidos, el servidor hace el tratamiento pertinente de los datos. Cuando acaba, envía la respuesta al navegador web, que a su vez también hace el tratamiento pertinente de la respuesta recibida para luego mostrarla en la interfaz web.

En esta arquitectura se puede identificar el modelo de desarrollo llamado programación por capas, también llamado arquitectura multinivel (27). En este caso, se pueden identificar las 3 capas más comunes en el diseño de aplicaciones que siguen la arquitectura cliente servidor.

La capa de presentación es la capa que ve el usuario, es decir, la conocida interfaz gráfica. Esta se encarga de mostrar la información de una forma sencilla para el usuario, además de permitir al usuario interactuar con la capa de negocio de la forma más amigable posible.

La capa de negocio es la capa en la que se encuentran los métodos que se ejecutan en el sistema, se reciben peticiones por parte de la capa de presentación y se envían los resultados una vez han sido ejecutadas. Esta capa se comunica tanto con la capa de presentación como con la capa de datos.

La capa de datos es donde están almacenados los datos, por tanto, está formada por el gestor de las bases de datos. Esta capa interactúa únicamente con la capa de negocio.

En el esquema de la arquitectura del sistema se pueden identificar fácilmente dichas capas.

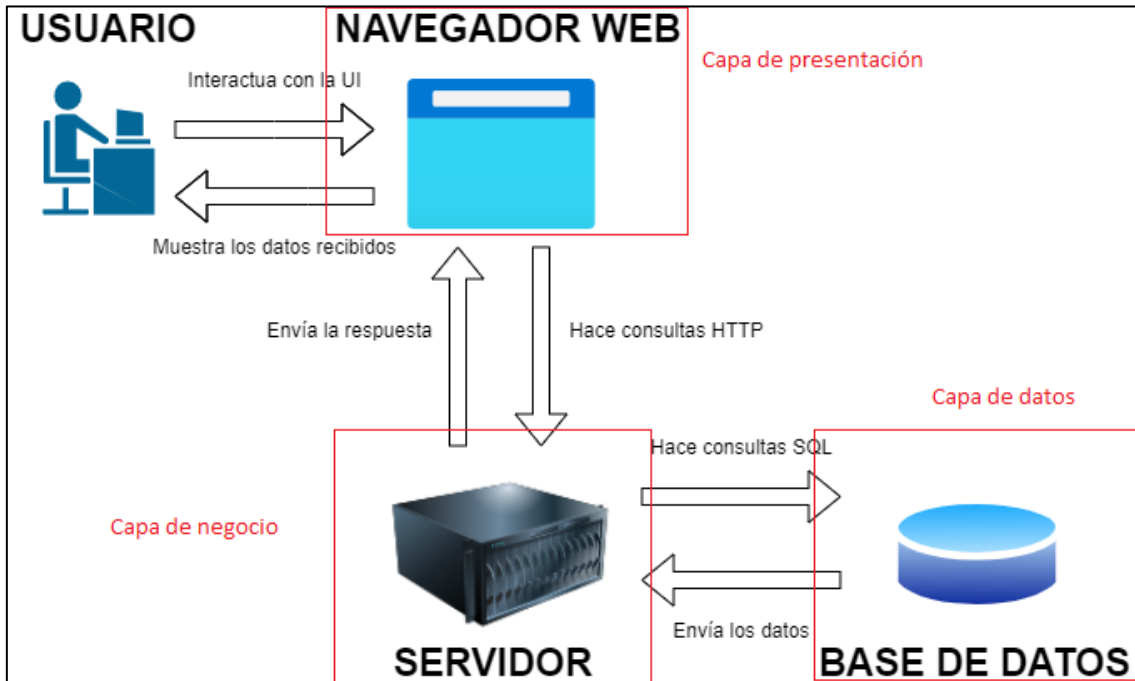


Figura 9- Arquitectura de capas

Como se puede ver, cada una de las capas concuerda con cada uno de los elementos de la arquitectura del sistema, ya que es una arquitectura simplificada, se basa exclusivamente en el modelo cliente-servidor, por lo que no intervienen más módulos.

4.3. Diseño detallado

Dentro de cada uno de los módulos vistos anteriormente, cada uno tiene una arquitectura interna. La capa de presentación (Navegador Web) tiene un conjunto de interfaces de usuario que permiten navegar e interactuar con los datos del servidor.

La capa de negocio tiene un esquema de consultas y respuestas que se pueden realizar al servidor, indicando las distintas acciones que se pueden realizar en la aplicación web. (27)

Por último, la capa de datos tiene las estructuras de tablas y atributos que utiliza la base de datos, relacionando unas tablas con otras mediante un esquema UML.

4.3.1. Capa de presentación

Como ya se ha mencionado anteriormente, la capa de presentación va muy ligada a las interfaces de usuario, las cuales son muy variables, ya que, dependiendo de la acción que se quiera realizar, se pueden mostrar unas interfaces u otras. (27)

En estos diseños, al ser exclusivamente relacionados con la capa de presentación, los datos que se muestran son irrelevantes, por lo que se puede realizar un esquema aproximado de la organización de los datos, sin importar cuales sean estos, indicando en cada caso en que tipo de acción se encuentra el usuario.

4.3.1.1. Menú Principal

Todas las páginas incluyen un menú en la parte superior, que facilita el acceso a cualquier página desde cualquier página.

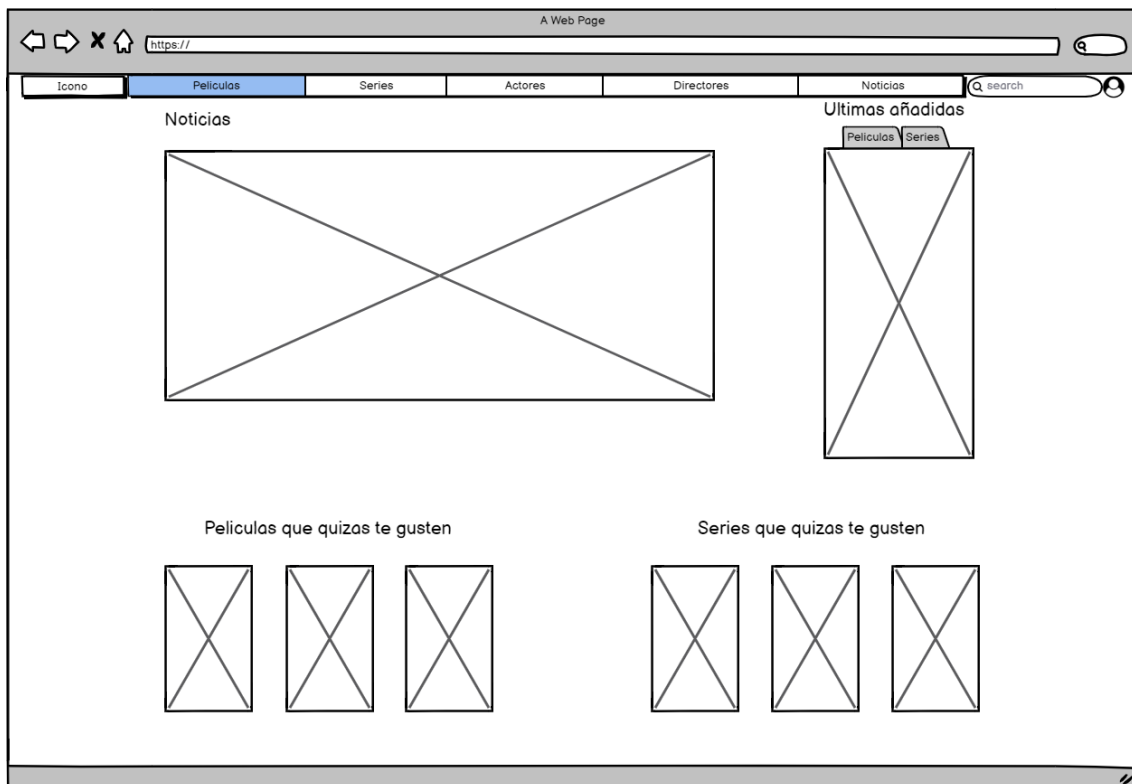


Figura 10- Esquema menú principal

En la página principal se encuentra un apartado de últimas noticias, que es un marco deslizable con las 3 últimas noticias añadidas en el sistema, y en la parte de abajo se incluyen las miniaturas para poder acceder a la noticia deseada.

A la derecha se puede ver un apartado de últimas añadidas, que está compuesto por un grupo de pestañas que permite elegir entre películas y series, y dentro de cada una de estas pestañas se incluye otro marco deslizable con las 3 últimas películas o series añadidas.

Debajo se encuentra un sistema de recomendaciones básico para cualquier usuario, recomienda películas al azar de las más votadas del sistema, son 3 imágenes con el título y un enlace directo a la página de la película o la serie.

Si se pulsa el botón del usuario, se ofrece una opción para cerrar sesión y otra opción para más datos. Si se accede a la opción de más datos, se carga una página con un submenú con 4 opciones a la izquierda, y el contenido de esa opción a la derecha.

4.3.1.2. Configuración de perfil

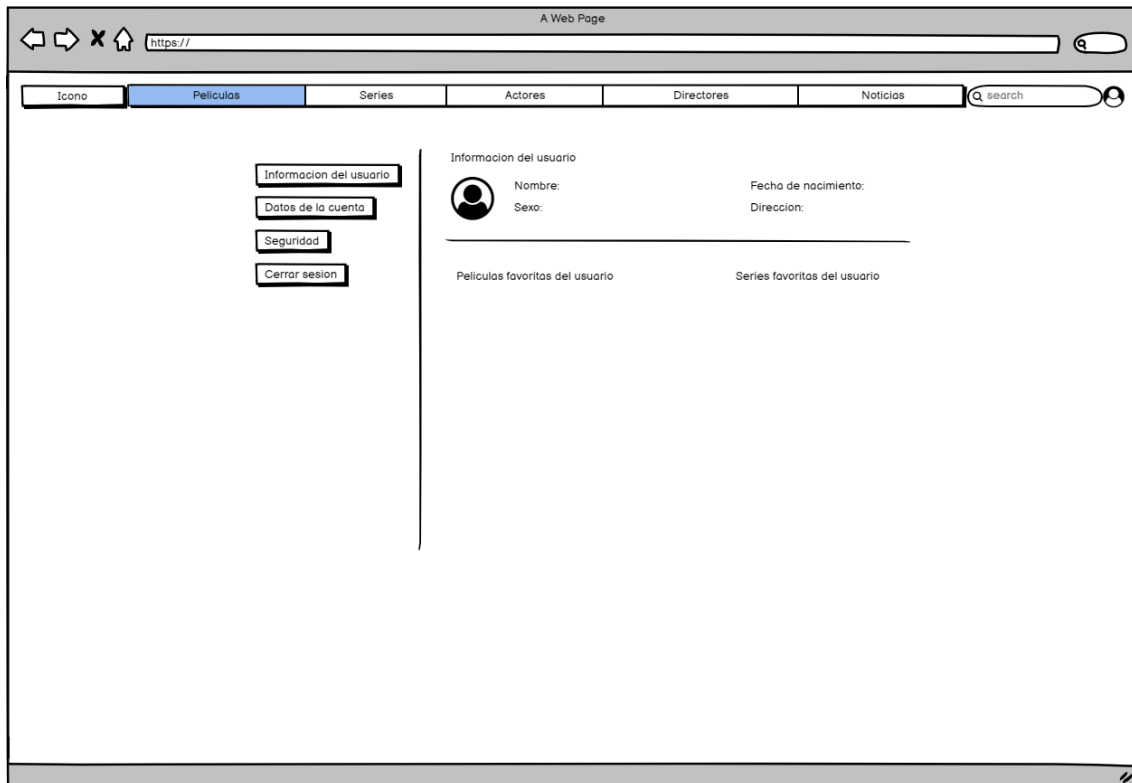


Figura 11- Esquema de configuración del perfil

La primera opción es la información del usuario, aquí se muestra un pequeño resumen de los datos del usuario, tales como su nombre, sexo, fecha de nacimiento, dirección, imagen de perfil, y una lista con sus películas y series favoritas. Dentro de esta opción se permite ver las películas que le han gustado al usuario, así como una opción para cambiar su imagen de perfil.

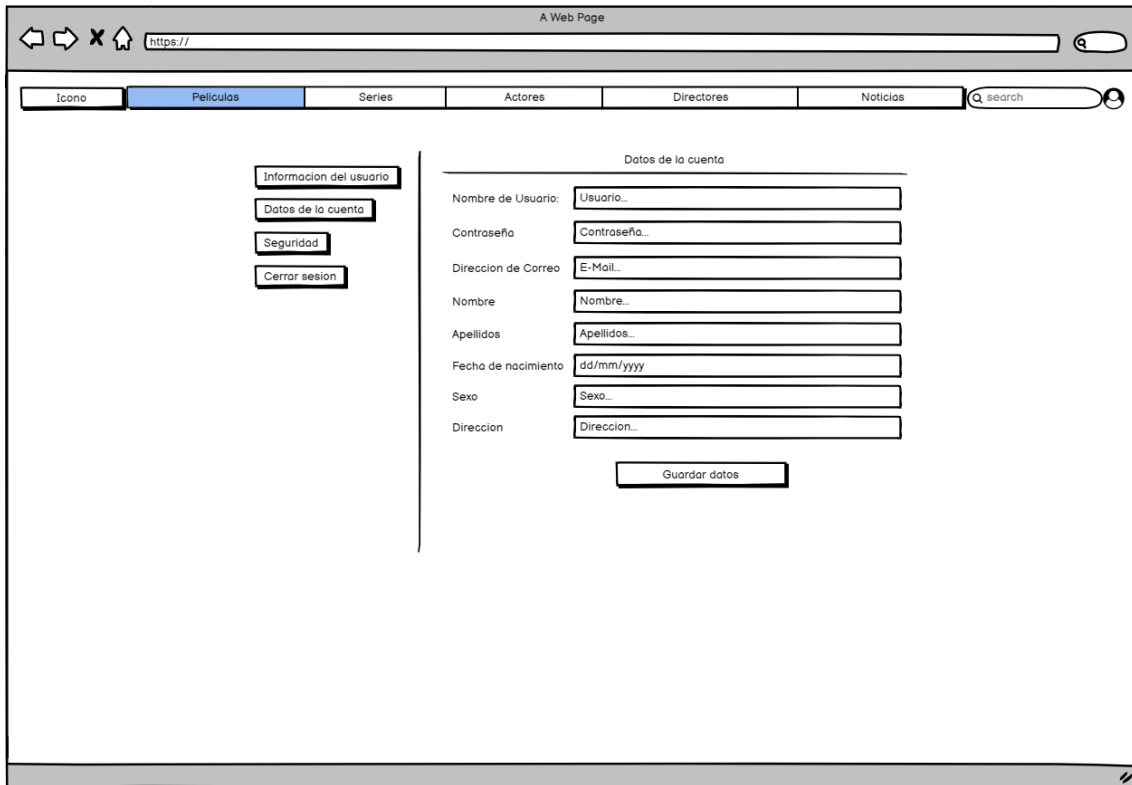


Figura 12- Esquema de configuración del perfil

Si se accede a la segunda opción, se muestra un conjunto de cuadros de texto con distintos datos del usuario. Los datos de sesión se muestran siempre que se accede al menú, pero a los datos personales sólo se accede si el usuario pulsa un botón que permite cambiarlos.

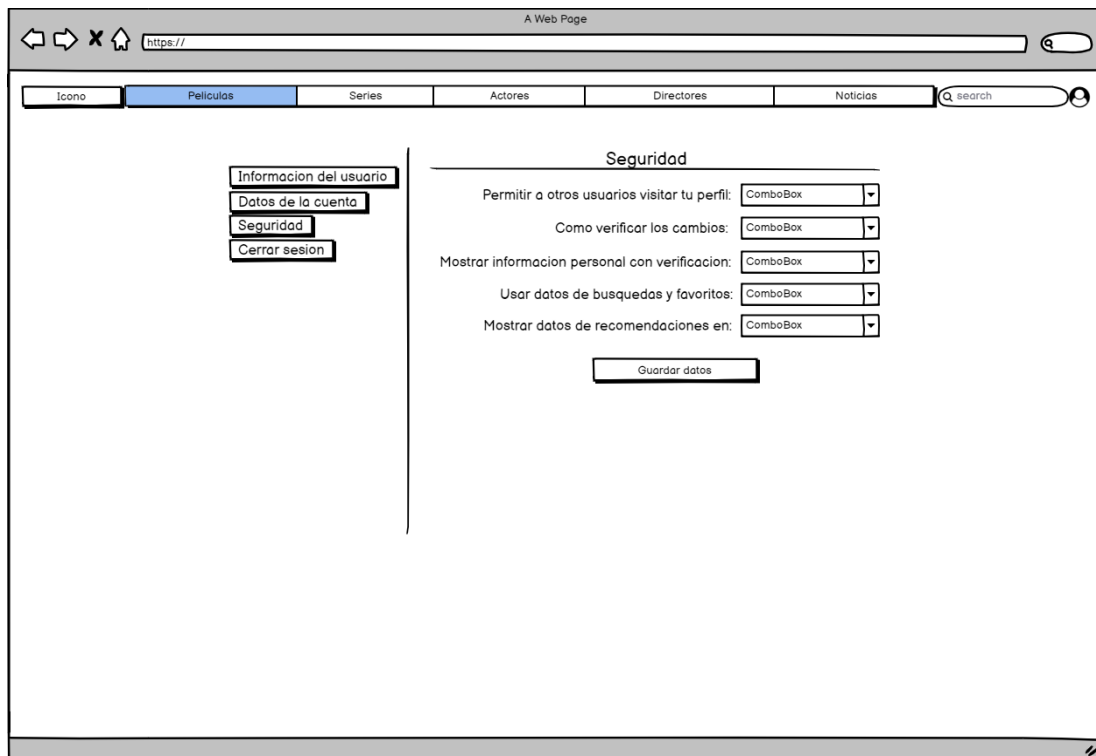


Figura 13- Esquema de configuración del perfil

En la pestaña de seguridad, se ofrecen diversas opciones para la configuración de la seguridad, tales como la opción para verificar los cambios, mostrar información personal solo con verificación, si usar los datos para las recomendaciones, o mostrar los datos de las recomendaciones.

El último apartado es un botón de cerrar sesión, por lo que no tiene una pestaña de interfaz gráfica concreta.

4.3.1.3. Listas de elementos

En los menús de películas, hay una opción para buscar por la primera letra, además de unos filtros de año, género, valoración mínima y buscar por nombre.

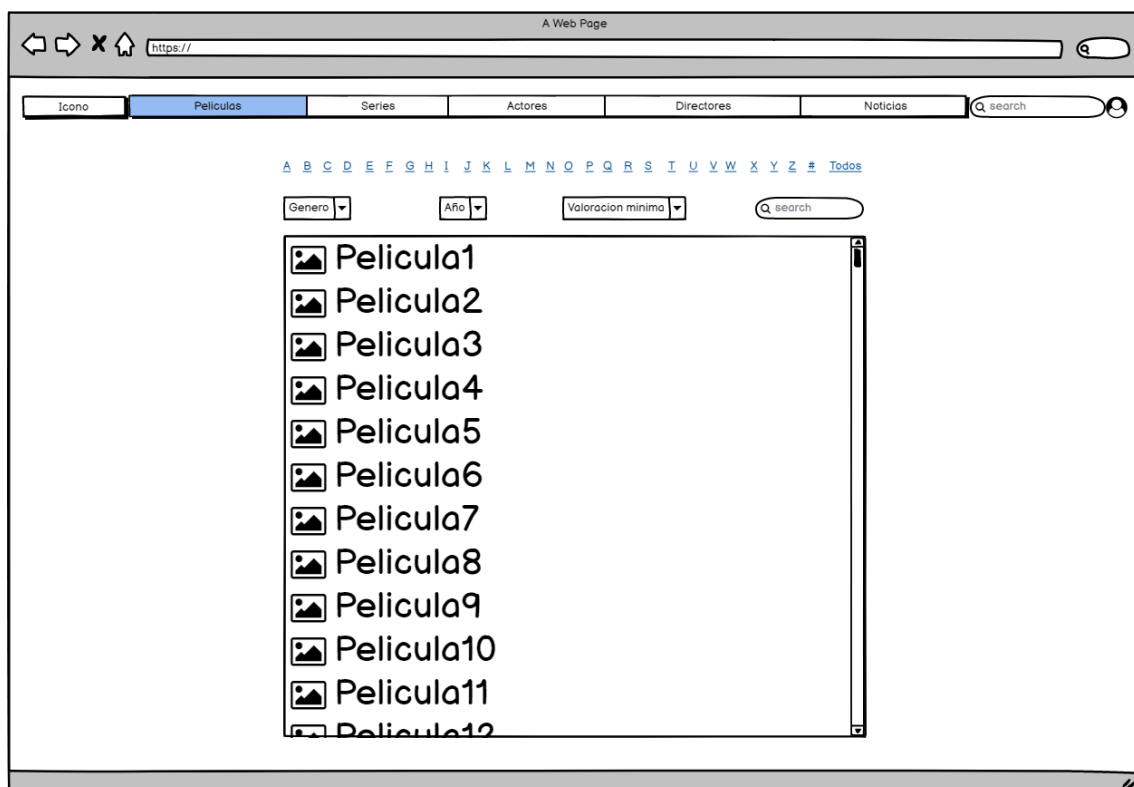


Figura 14- Esquema de listas

Para las series, el menú es exactamente igual, salvo que en vez de una lista de películas es una lista de series.

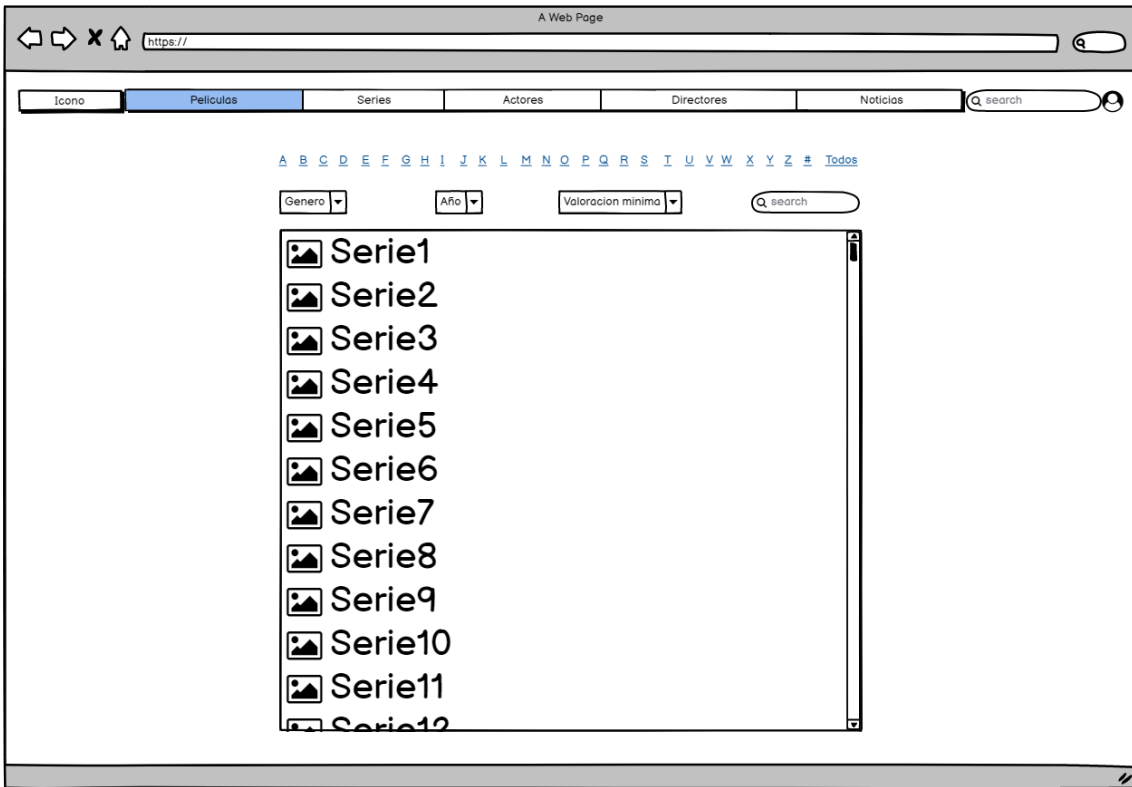


Figura 15- Esquema de listas de series

Para la lista de actores actrices, se encuentra una lista similar, salvo que hay menos filtros de búsqueda, ya que, a la hora de las búsquedas, la edad del actor o el género son indiferentes.

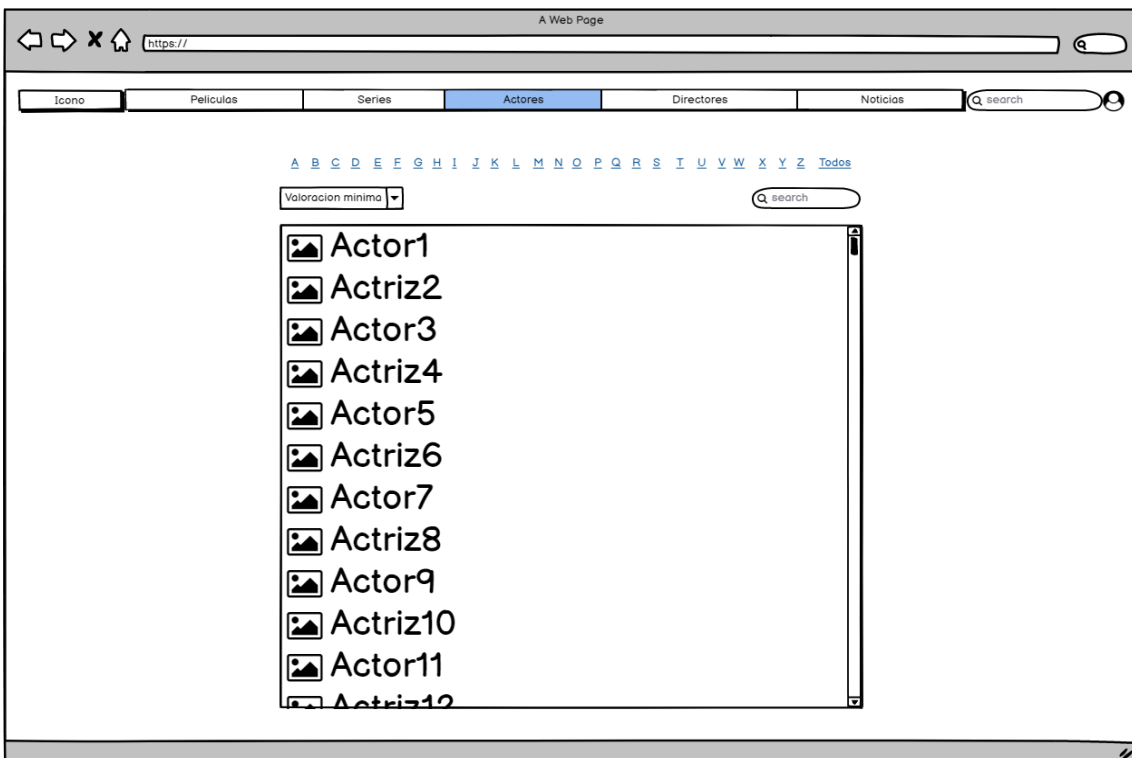


Figura 16- Esquema de listas de actores



La búsqueda de directores es similar a la de actores/actrices.

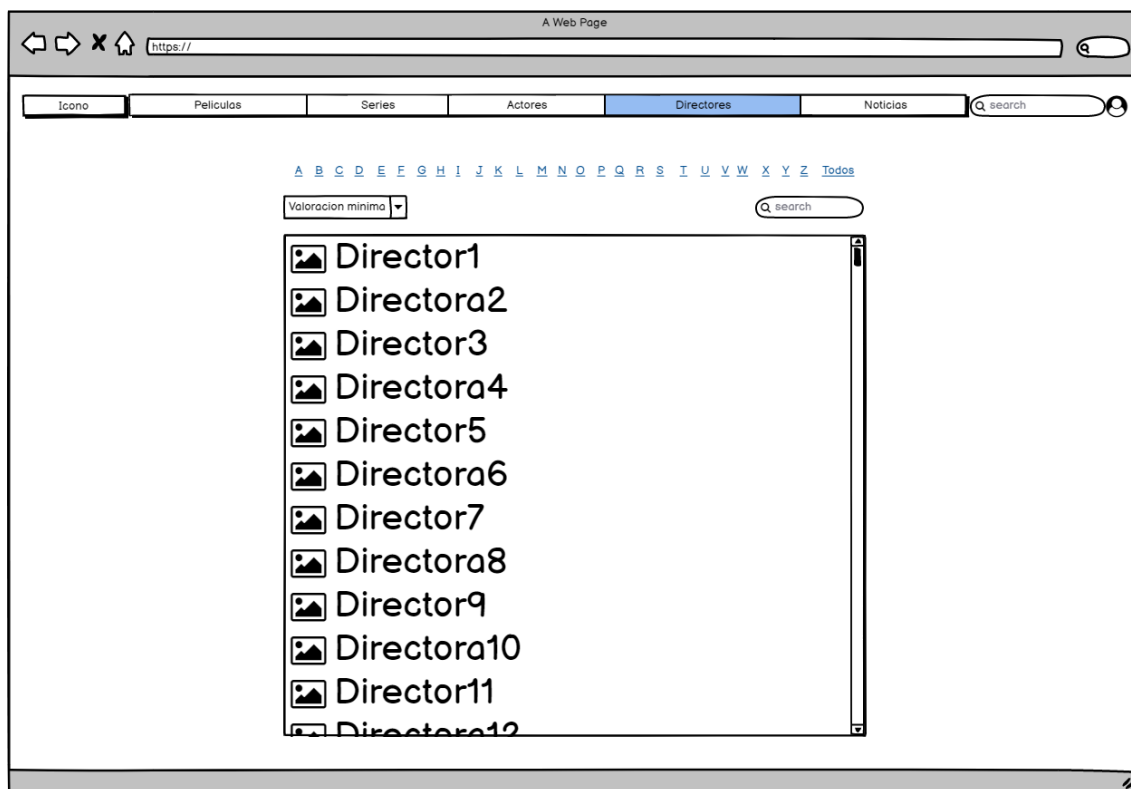


Figura 17- Esquema de listas de directores

Por último, la búsqueda de noticias está compuesta por una lista y un conjunto de filtros, siendo estos el año, el género y la búsqueda de la noticia. Si se pulsa en la lista el enlace de la película, se filtran todas las noticias de dicha película.

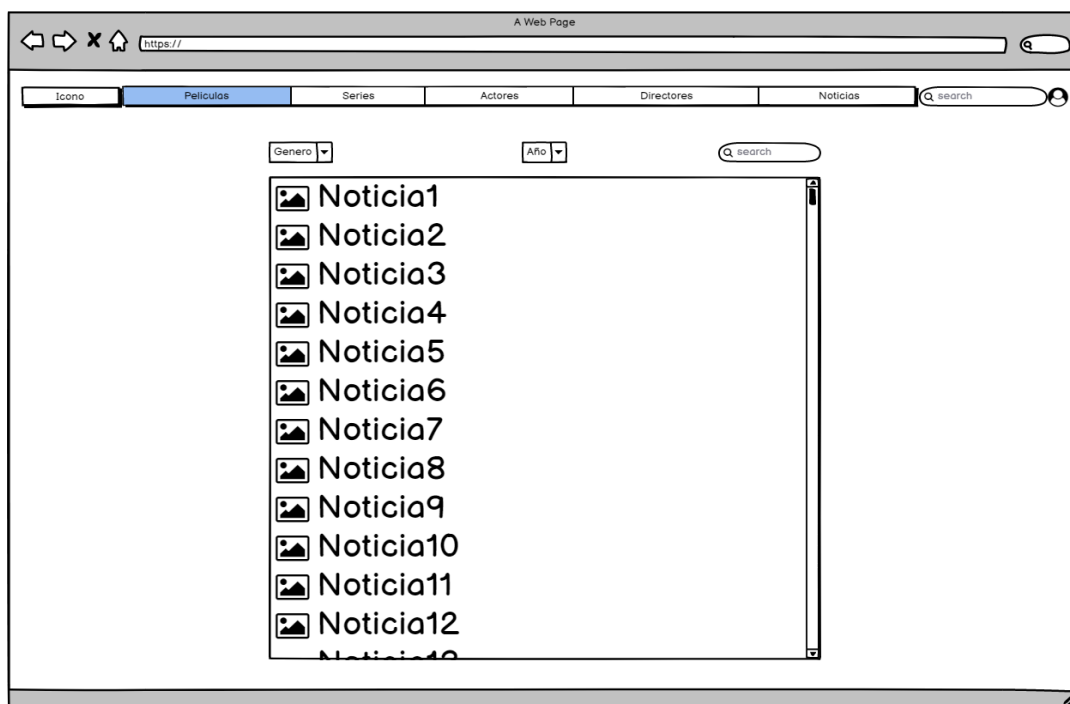


Figura 18- Esquema de lista de noticias

Aparte, para cada uno de estos apartados hay un menú de administración habilitado para los administradores. Es igual que el visto anteriormente, pero con un menú CRUD.

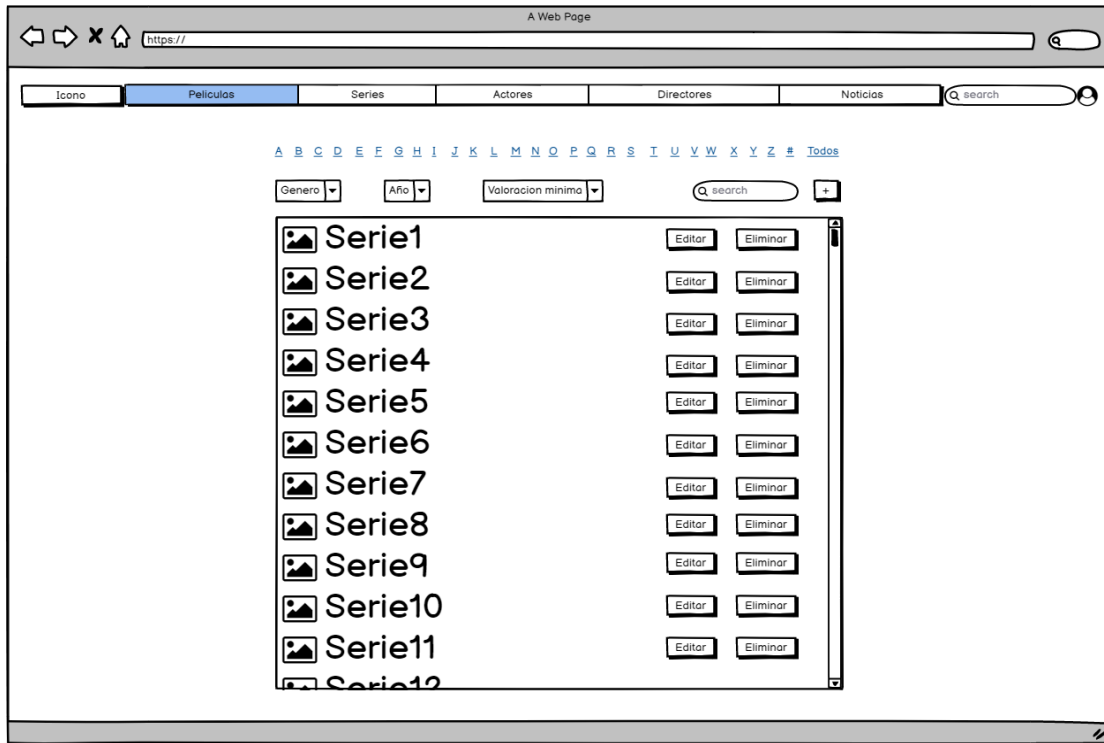


Figura 19- Esquema de listas de administración de series

4.3.1.4. Presentación de elementos

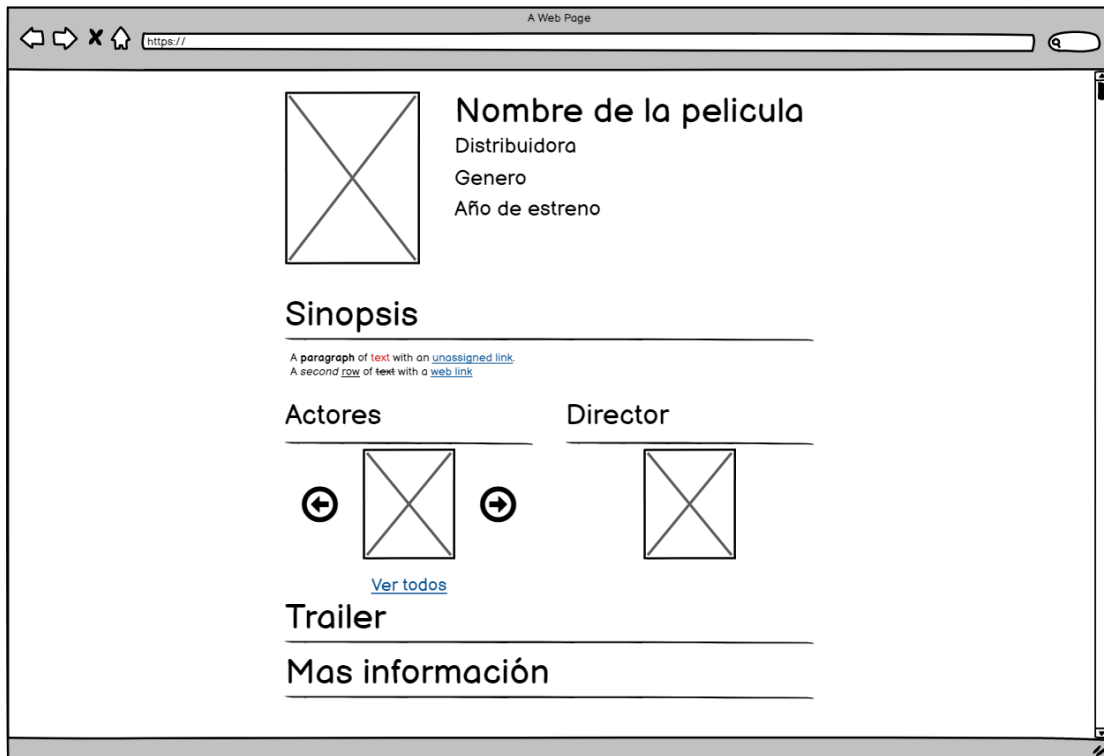


Figura 20- Esquema de página de visualización de película

El apartado individual de una película muestra el póster de dicha película, junto con datos como distribuidor, género y año de estreno. Aparte también hay una sinopsis, un sistema de imagen deslizante con los actores y director, un tráiler (Que será un incrustado de YouTube para no infringir las leyes de Copyright) y otra información, que puede ser la duración, recaudación en taquilla o nominaciones a premios.

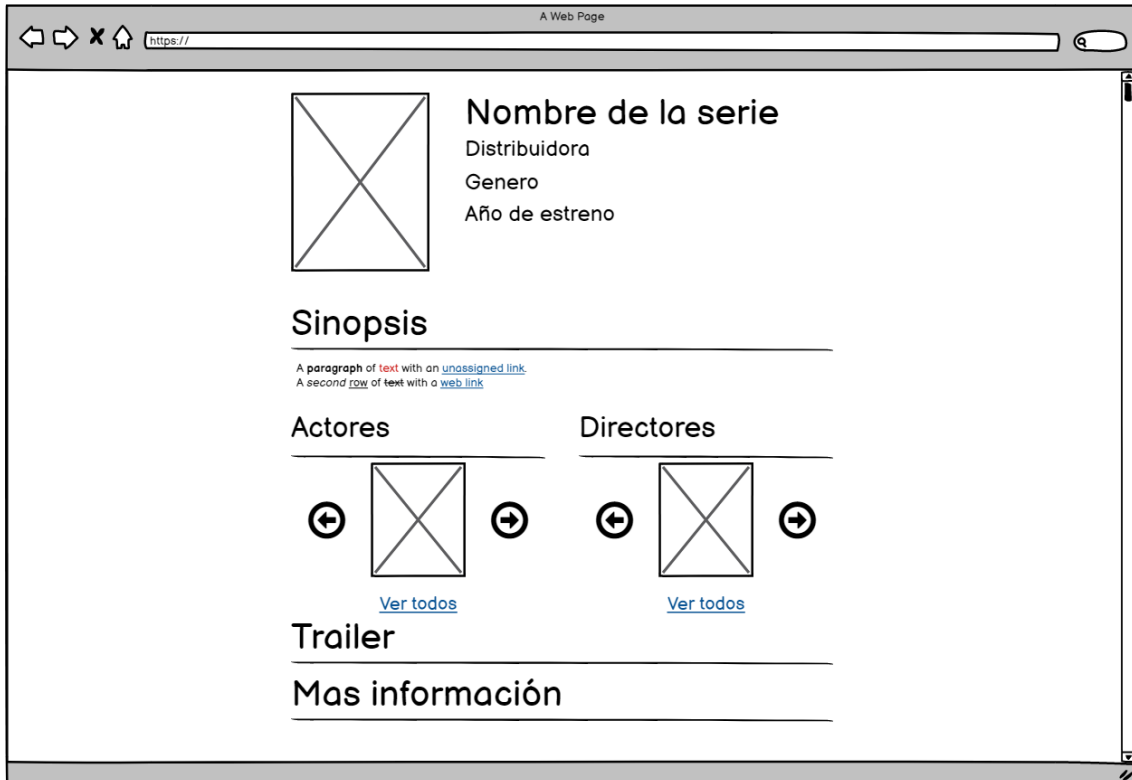


Figura 21- Esquema de página de visualización de serie

En cuanto al apartado de películas es similar, salvo que, en lugar de haber solo un director, hay otro sistema de imagen deslizante, ya que normalmente los capítulos suelen ser rodados por directores diferentes. En más información se encuentra las temporadas emitidas y los premios a los que ha sido nominada. Así como también se encuentra un apartado de noticias.

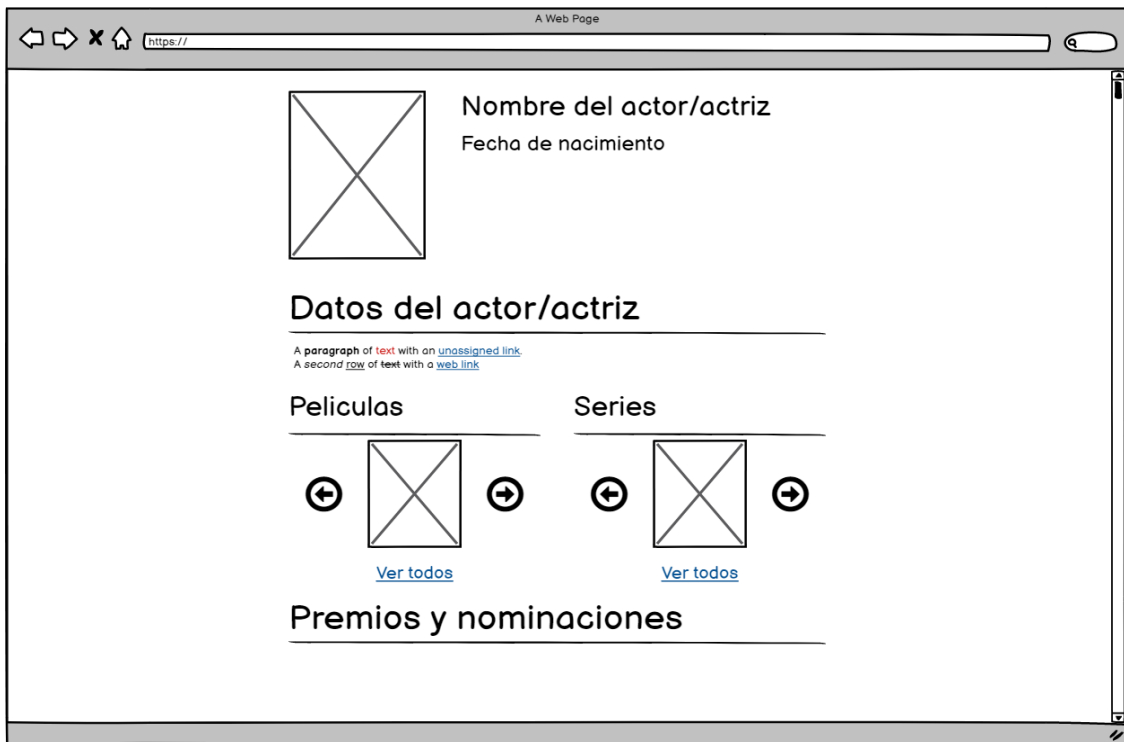


Figura 22- Esquema de página de visualización de actores

El apartado de actores o actrices incluye el nombre de dicho actor junto a su fecha de nacimiento. Luego incluye un pequeño resumen de su vida, así como las películas en las que ha participado dicho actor y los premios y nominaciones que ha recibido por ello.

El apartado de directores es similar, ya que dirigen películas y series y están nominados a diversos premios, por lo que las categorías de la página serán similares.

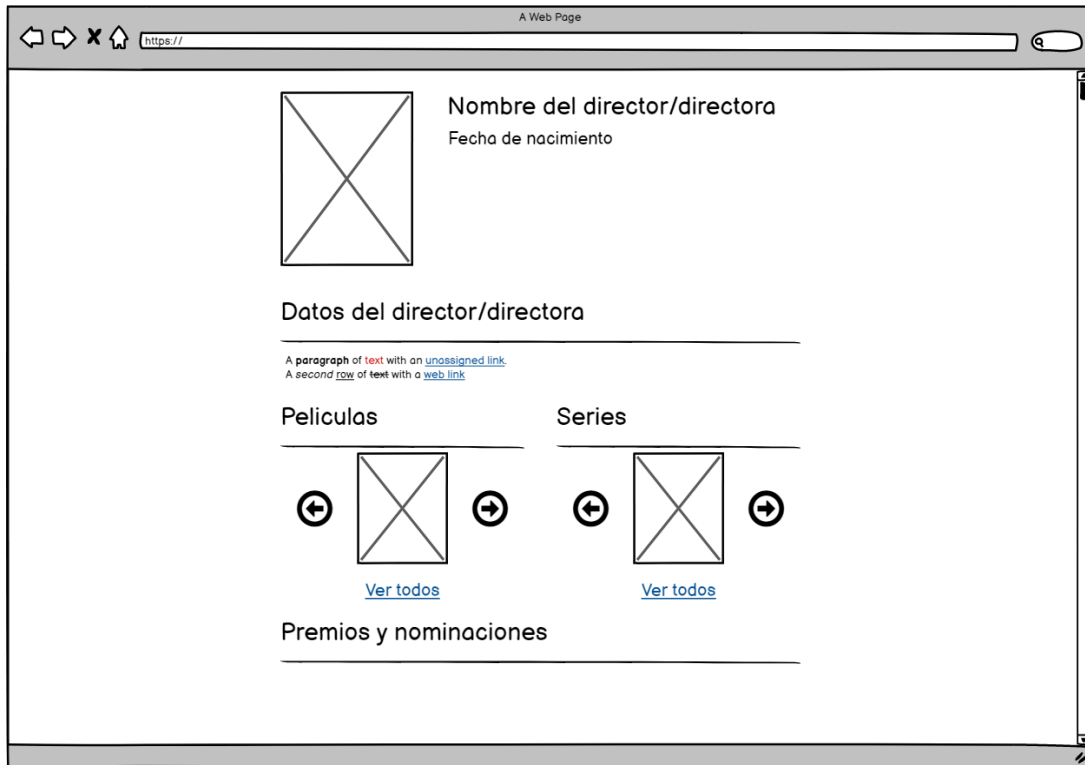


Figura 23- Esquema de página de visualización de director

Las noticias se representan mediante una imagen panorámica como la mostrada en el menú de inicio, junto con el texto de la noticia y unas imágenes relacionadas con un texto explicativo debajo.

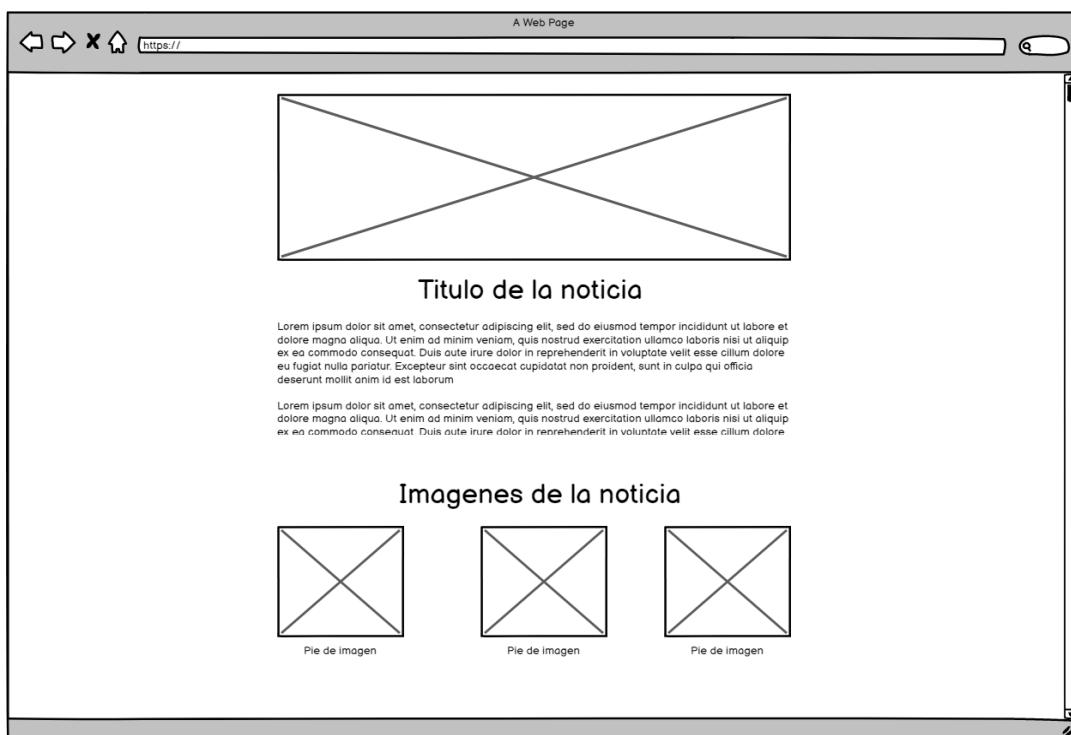


Figura 24- Esquema de página de visualización de noticias

4.3.1.5. Edición de elementos

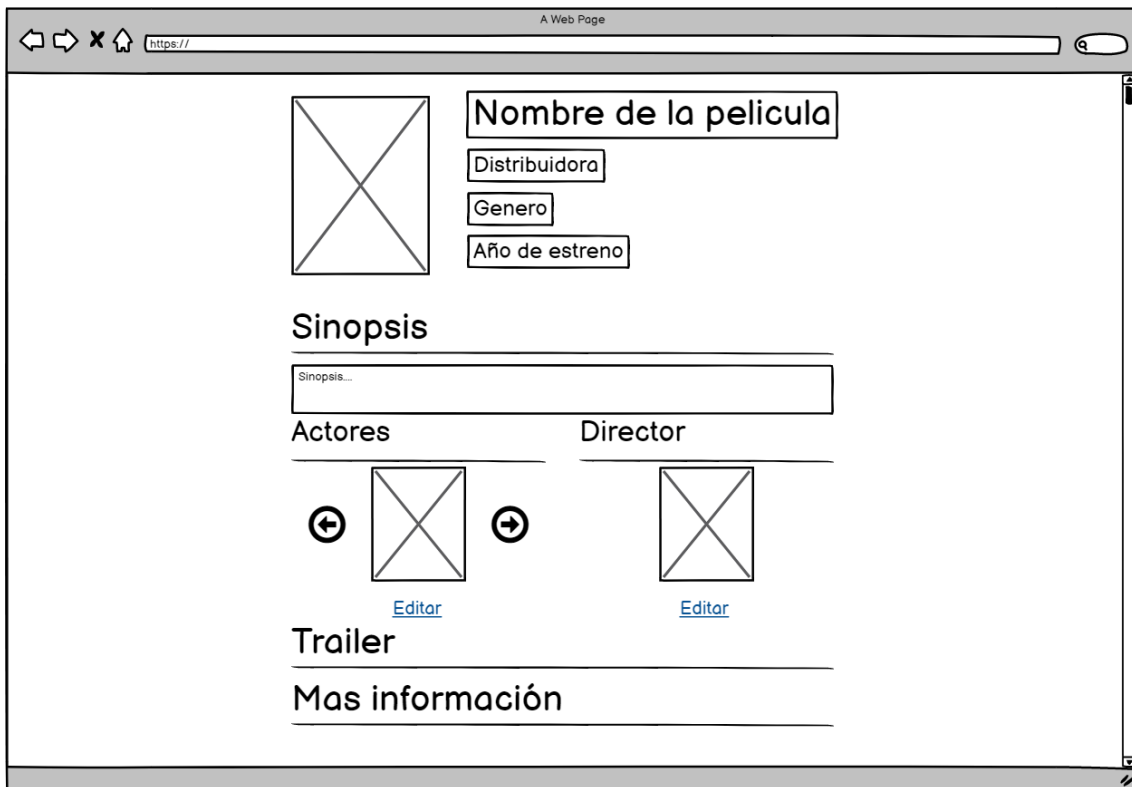


Figura 25- Esquema de página de visualización de noticias

El sistema de edición es exactamente igual que la visualización, por lo que se podría considerar un sistema WYGIWYS. A la hora de añadir o eliminar actores o actrices y directores, se utiliza un sistema de *modals* que añaden o eliminan a estos de una lista.



Figura 26- Esquema de página de edición de película, apartado actores

El resto de los elementos son texto o insertar enlaces incrustados de YouTube, como el caso de los *trailers*. Para el resto de los elementos la edición es exactamente igual, por lo que no es necesario mostrar cada una de las interfaces, ya que solo cambian los inputs de texto y la opción de añadir datos.

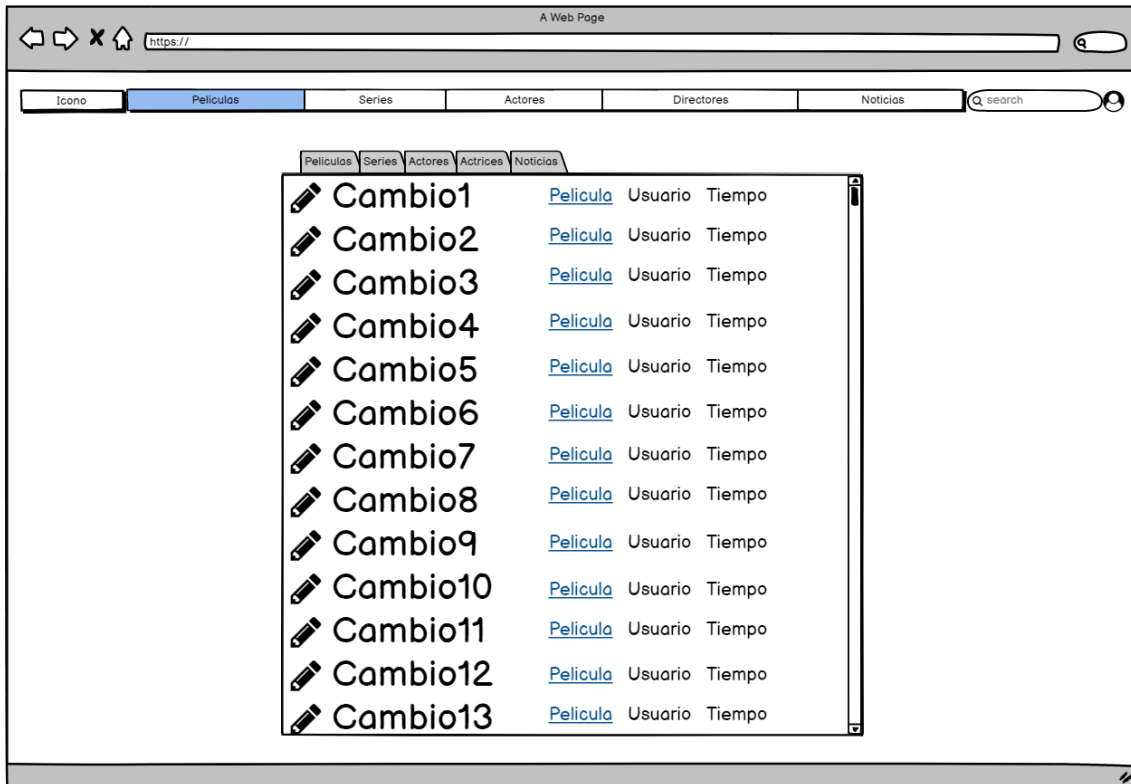


Figura 27- Esquema de página de visualización de cambios

El panel de cambios de los administradores permite ver al administrador los cambios que proponen los usuarios, se puede ver a qué película pertenece, qué usuario ha hecho el cambio y cuánto tiempo hace que lo ha hecho.

4.3.2. Capa de negocio

La capa de negocio es la capa en la que se encuentran los métodos que se ejecutan en el sistema, se reciben peticiones por parte de la capa de presentación y se envían los resultados una vez han sido ejecutadas. (27)

La arquitectura de esta capa es un conjunto de funciones que se pueden ejecutar en el sistema y el orden en el que pueden ser ejecutadas.

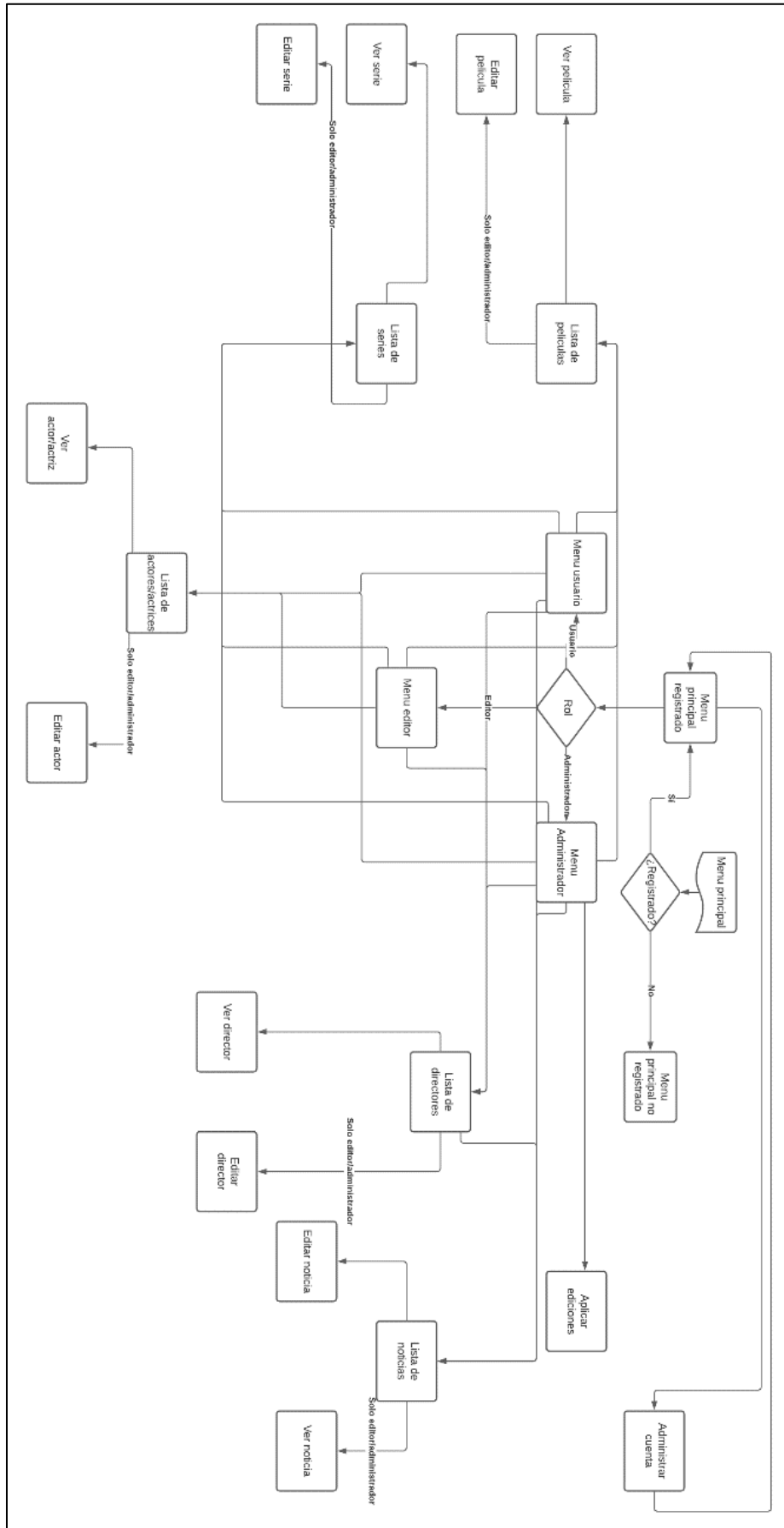


Figura 28- Diagrama de la capa de negocio



El diagrama de ejecución es el siguiente, se divide en usuario registrado y no registrado (El usuario registrado tiene todas las funciones de visualización, pero no ha sido posible representarlo en el diagrama). Además de esto, es un sistema cíclico, ya que, dada la funcionalidad que proporciona el menú, se puede redirigir de cualquier punto a cualquier punto.

4.3.3. Capa de datos

Como se ha comentado anteriormente, la capa de datos se corresponde con la Base de Datos, tanto estructuras como los datos en sí. La estructura de la capa de datos se puede representar con un diagrama UML. (27)

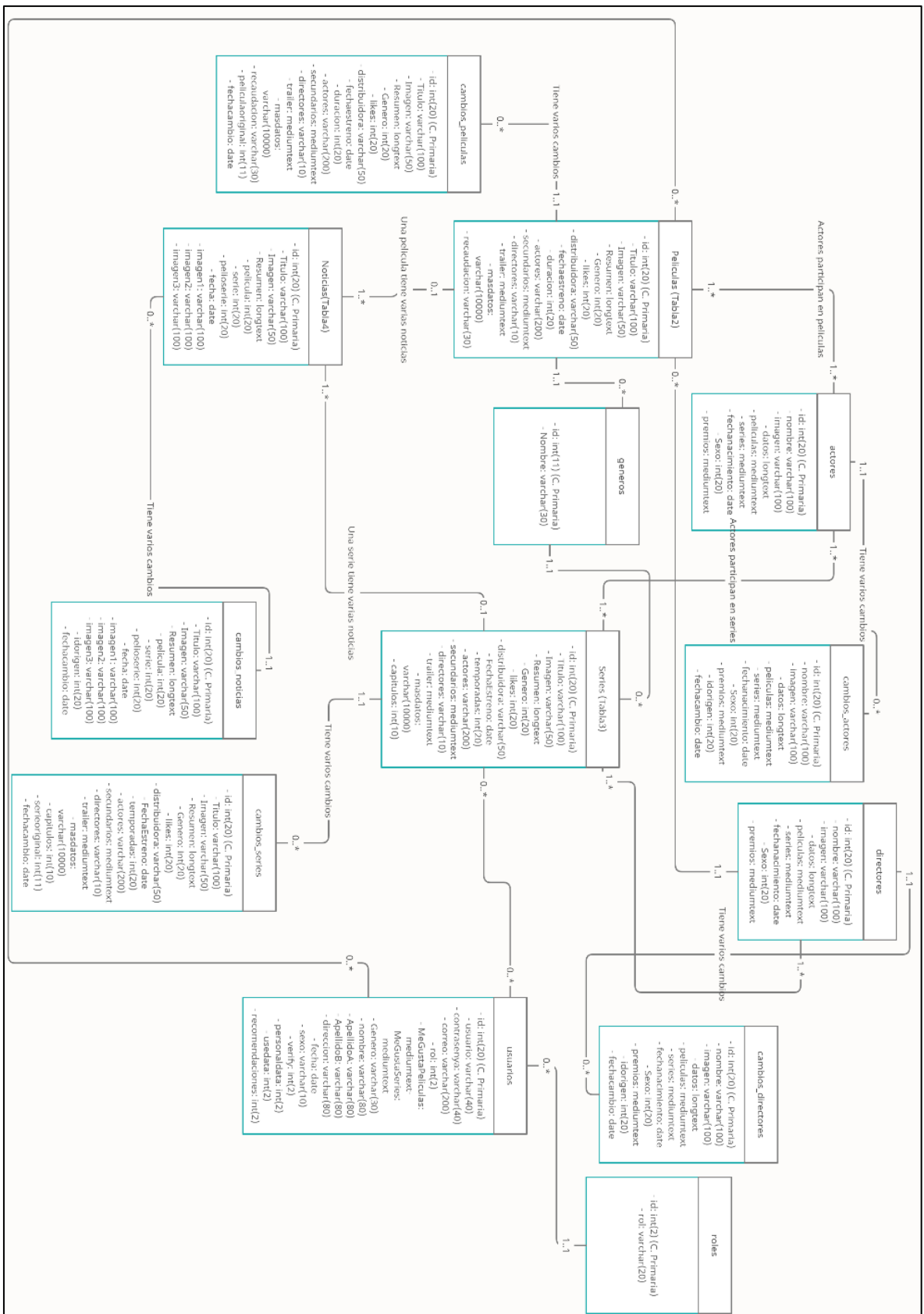


Figura 29- Diagrama de la capa de datos



Como se puede ver en el diagrama, la Base de Datos está formada por distintas tablas, siendo las principales las tablas de películas, series, actores, directores y noticias. Estas tablas tienen copias utilizadas para almacenar cambios hechos por usuarios editores para mantener la integridad de las filas originales hasta que el administrador acepte y administre dichos cambios. Por otra parte, se encuentra la tabla de usuario donde se almacenan los usuarios registrados en la aplicación. Por último, existen 2 tablas de apoyo, que son la tabla de género y la tabla de rol, donde se indica el id y la especificación tanto de los géneros de las películas y series y el rol que cumplen los usuarios.

5. Desarrollo de la solución

La solución, como se ha explicado anteriormente, se ha hecho mediante el desarrollo de una aplicación web a medida, es decir, programando directamente los documentos, estilos y utilidades del sistema.

La implementación de las diversas páginas se ha basado en los diseños de los *wireframes*, por lo que no es necesario mostrar todas las distintas páginas de nuevo, ya que son bastante similares. Dado esto, es recomendable centrarse en algunas implementaciones de funcionalidades del sistema, como los sistemas de recomendaciones que se utilizan. También se especifican las herramientas utilizadas para el desarrollo del proyecto.

5.1. Herramientas utilizadas

Como se ha comentado en el apartado de diseño de la solución, las tecnologías utilizadas están relacionadas con el diseño de aplicaciones web a medida. Como ya se han explicado anteriormente las tecnologías que se utilizan en relación con la programación y diseño, no se hará hincapié para no repetir lo dicho en apartados anteriores.

5.1.1. XAMPP

XAMPP es un paquete de software libre que contiene varias herramientas necesarias para la creación de una aplicación web, como el servidor *Apache*, PHP o *MySQL* (*MariaDB*). (25)

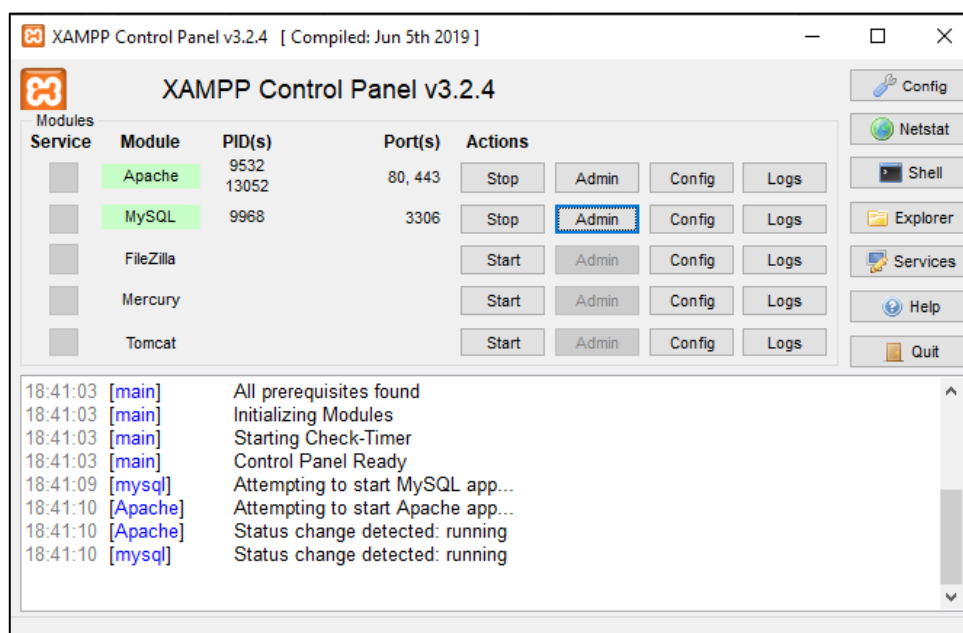


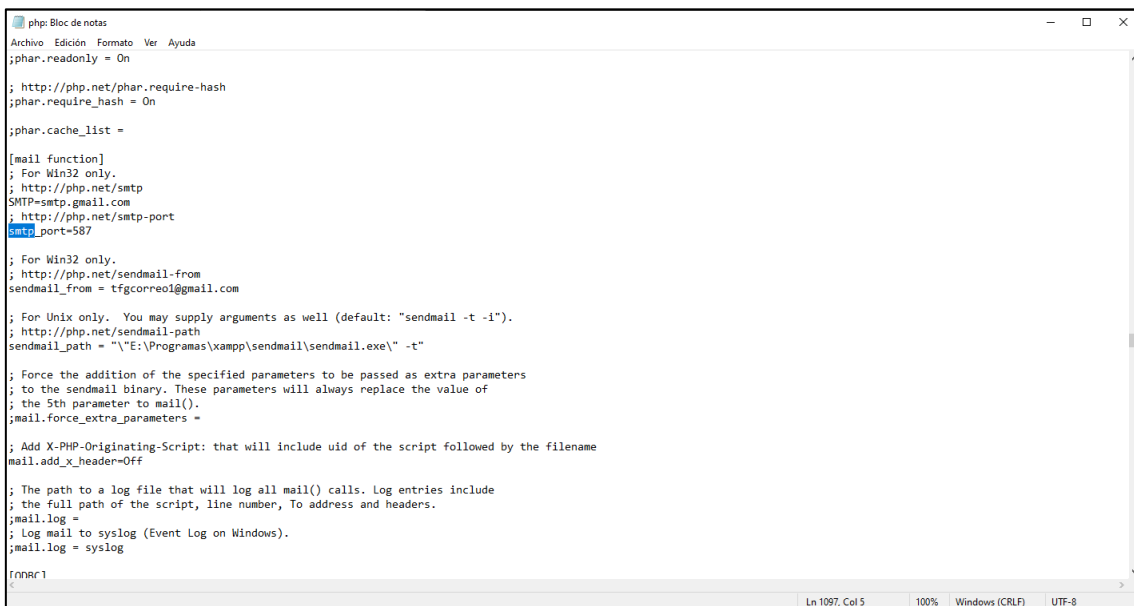
Figura 30- Panel de control de XAMPP v3.2.4

Como se puede ver en la imagen, las dos herramientas utilizadas en este proyecto son *Apache* y *MySQL*. XAMPP tiene un panel de control que permite iniciar las herramientas necesarias, así como configurar algunas de estas.

5.1.1.1. Apache

Apache (26), como se ha comentado, es un servidor HTTP que permite que la aplicación web sea accesible mediante el protocolo mencionado. Apache se utiliza por defecto en la mayoría de los casos, salvo para indicar la ruta en la que se ubican los datos de la aplicación web, para esto se tiene que indicar la ruta en el archivo *httpd.conf*, pero en este caso concreto se deja por defecto dentro de la carpeta *htdocs* dentro de la instalación de XAMPP.

Por otra parte, en esta aplicación se ofrece una opción de verificación de titularidad de la cuenta mediante una verificación por email, para lo que es necesario configurar tanto en PHP como en la herramienta *sendmail* incluidas en XAMPP. Los archivos de configuración son *php.ini* y *sendmail.ini*



```
php: Bloc de notas
Archivo Edición Formato Ver Ayuda
;phar.readonly = On

; http://php.net/phar.require-hash
;phar.require_hash = On

;phar.cache_list =

[mail function]
; For Win32 only.
; http://php.net/smtp
SMTP=smtp.gmail.com
; http://php.net/smtp-port
smtp_port=587

; For Win32 only.
; http://php.net/sendmail-from
sendmail_from = tfgcorreol@gmail.com

; For Unix only. You may supply arguments as well (default: "sendmail -t -i").
; http://php.net/sendmail-path
sendmail_path = "\"E:\Programas\xampp\sendmail\sendmail.exe\" -t"

; Force the addition of the specified parameters to be passed as extra parameters
; to the sendmail binary. These parameters will always replace the value of
; the 5th parameter to mail().
;mail.force_extra_parameters =

; Add X-PHP-Originating-Script: that will include uid of the script followed by the filename
mail.add_x_header=Off

; The path to a log file that will log all mail() calls. Log entries include
; the full path of the script, line number, To address and headers.
;mail.log =
; Log mail to syslog (Event Log on Windows).
;mail.log = syslog

rnnrr 1
Ln 1097, Col 5 100% Windows (CRLF) UTF-8
```

Figura 31- Configuración SMTP en el archivo *php.ini*

```

sendmail: Bloc de notas
Archivo Edición Formato Ver Ayuda
smtp_server=smtp.gmail.com

; smtp port (normally 25)
smtp_port=587

; SMTPS (SSL) support
; auto = use SSL for port 465, otherwise try to use TLS
; ssl = always use SSL
; tls = always use TLS
; none = never try to use SSL

smtp_ssl=auto

; the default domain for this server will be read from the registry
; this will be appended to email addresses when one isn't provided
; if you want to override the value in the registry, uncomment and modify
;default_domain=mydomain.com

; log smtp errors to error.log (defaults to same directory as sendmail.exe)
; uncomment to enable logging

error_logfile=error.log

; create debug log as debug.log (defaults to same directory as sendmail.exe)
; uncomment to enable debugging

debug_logfile=debug.log

; if your smtp server requires authentication, modify the following two lines

auth_username=tfccorreo1@gmail.com
auth_password=elcorreodelfg1

; if your smtp server uses pop3 before smtp authentication, modify the
; following three lines do not enable unless it is required

```

Figura 32- Configuración SMTP en el archivo sendmail.ini

Como se puede ver, en estos archivos se configura tanto el puerto SMTP como el servidor SMTP de *Gmail*, así como la dirección y la contraseña que se usan para enviar los correos a los usuarios.

5.1.1.2. MySQL

MySQL es el gestor de la Base de Datos de la aplicación, más concretamente se usa una versión de *MariaDB* (25). Para administrar los contenidos que guarda esta base de datos, a pesar de que muchos de estos datos pueden ser modificados desde la propia aplicación, otros solo pueden ser accedidos desde un administrador de *MySQL*, siendo en este caso *PHPMysqlAdmin*, ya que es la herramienta por defecto de *XAMPP*.

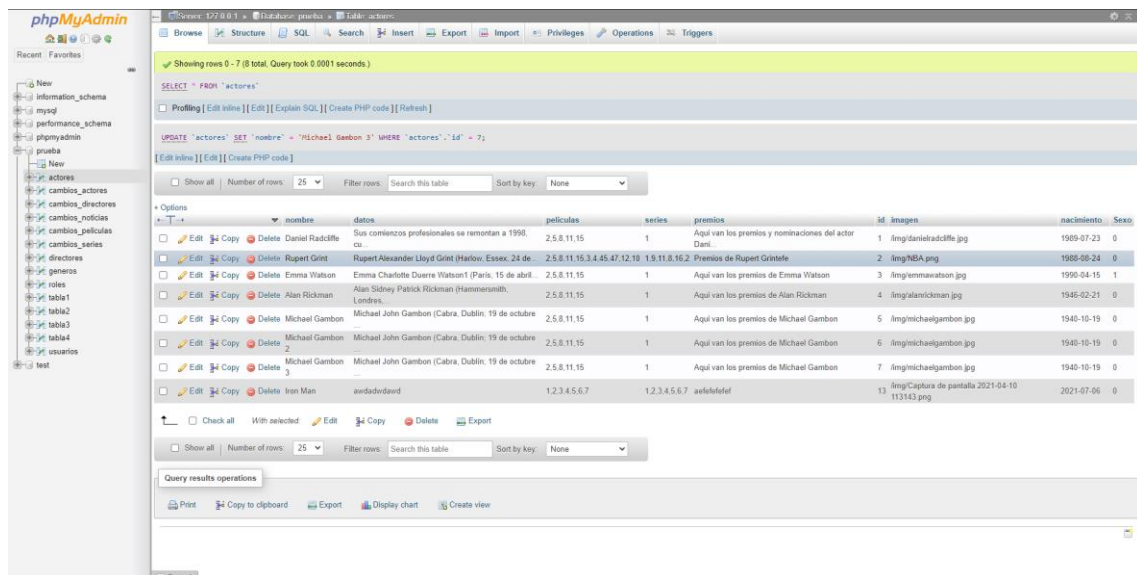


Figura 33- Panel de control de una tabla en PHPMysqlAdmin

Base de datos cinematográfica con sistema de recomendaciones por perfiles de usuarios

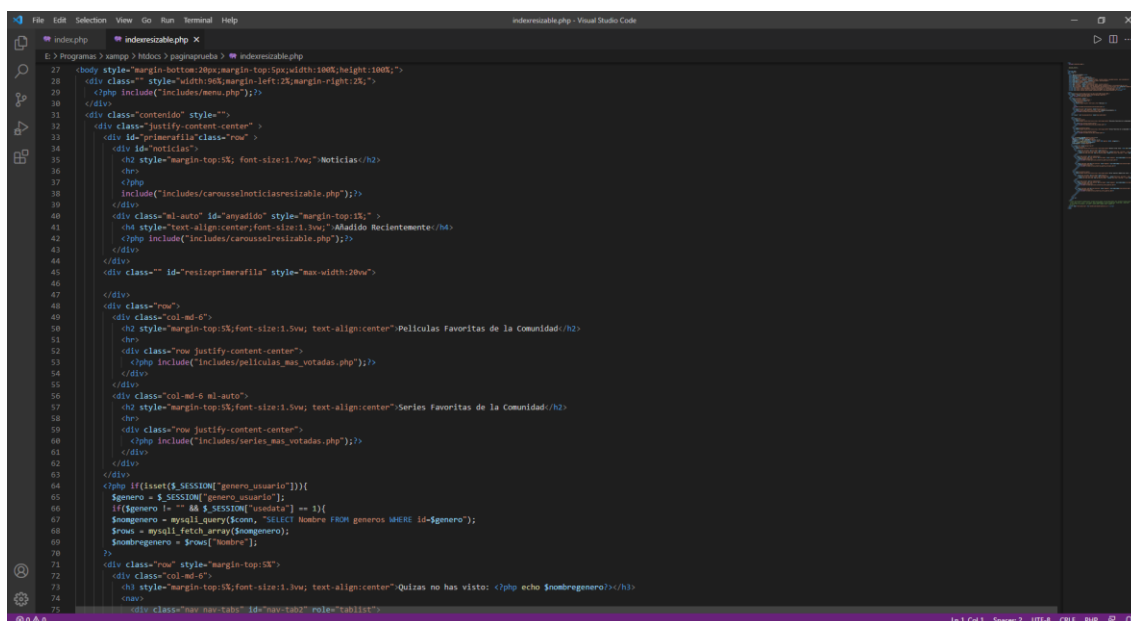
En *PHPMyAdmin* se pueden administrar todos los contenidos y estructuras de forma sencilla, cómo se puede ver en la imagen anterior, donde se muestra un ejemplo de los contenidos de una tabla de la Base de Datos.

Además de ver las filas de las diversas tablas, se pueden administrar las estructuras, insertar o buscar filas o hacer consultas SQL.

5.1.2. Microsoft Visual Studio Code

*Visual Studio Code*⁸ (28) es un editor de código fuente con soporte para la gran mayoría de lenguajes de programación, además de permitir ejecutar código mediante diversos plugin que se pueden descargar, aparte de todo esto, incluye soporte para *git*.

En este proyecto se utiliza ya que permite funciones de resaltar los distintos tipos de código, además de funciones de autocompletado. Es compatible tanto con HTML, CSS, *JavaScript* y PHP.



```
index.php
E:\Programas\xampp\htdocs\ejemplo2 > index.php
27 <body style="margin-bottom:20px;margin-top:20px;width:100%;height:100%;">
28 <div class="" style="width:90%;margin-left:5%;margin-right:5%;">
29 <php include("includes/menu.php");>
30 </div>
31 <div class="contenido" style="">
32 <div class="justify-content-center">
33 <div id="primerafila" class="row">
34 <div id="noticias">
35 <h2 style="margin-top:5%; font-size:1.7em;">Noticias</h2>
36 <hr>
37 <hr>
38 <php include("includes/carouselnoticiasresizable.php");>
39 </div>
40 <div class="nl-auto" id="anuncio" style="margin-top:5%;">
41 <h2 style="text-align:center;font-size:1.5em;">¡¡¡¡¡Ahadido Recientemente!!!</h2>
42 <php include("includes/carouselresizable.php");>
43 </div>
44 </div>
45 <div class="" id="resizprimerafila" style="max-width:200px">
46 </div>
47 </div>
48 <div class="row">
49 <div class="col-md-6">
50 <h2 style="margin-top:5%;font-size:1.5em; text-align:center">Películas Favoritas de la Comunidad</h2>
51 <hr>
52 <div class="row justify-content-center">
53 <php include("includes/peliculas_mas_votadas.php");>
54 </div>
55 </div>
56 <div class="col-md-6 nl-auto">
57 <h2 style="margin-top:5%;font-size:1.5em; text-align:center">Series Favoritas de la Comunidad</h2>
58 <hr>
59 <div class="row justify-content-center">
60 <php include("includes/series_mas_votadas.php");>
61 </div>
62 </div>
63 </div>
64 <php if(isset($_SESSION["genero_usuario"])){
65 $genero = $_SESSION["genero_usuario"];
66 if($genero != "" && $_SESSION["usadata"] == 1){
67 $sqlgenero = mysql_query($con, "SELECT Nombre FROM generos WHERE id=$genero");
68 $rows = mysql_fetch_array($sqlgenero);
69 $nombregenero = $rows["Nombre"];
70 }
71 <div class="row" style="margin-top:5%;">
72 <div class="col-md-6">
73 <h2 style="margin-top:5%;font-size:1.5em; text-align:center">Quizás no has visto:</h2><php echo $nombregenero;></h2>
74 <hr>
75 <div class="nav" style="margin-top:5%;font-size:1.5em; text-align:center">Quizás no has visto:</h2><php echo $nombregenero;></h2>
```

Figura 34- Visualización de un archivo HTML en VS Code

5.1.3. Herramientas de diagramas

Para los distintos tipos de diagramas que se incluyen en la memoria del proyecto, se han utilizado distintos programas.

5.1.3.1. Diagrams.net

Esta aplicación web se utiliza para hacer muchos tipos de diagramas distintos, ya que proporciona herramientas para hacer diagramas de clases y diagramas personalizados

⁸ <https://code.visualstudio.com/>



entre otros. Ha sido usada para desarrollar la arquitectura del sistema, ya que permite la utilización de imágenes para representar Bases de Datos o servidores

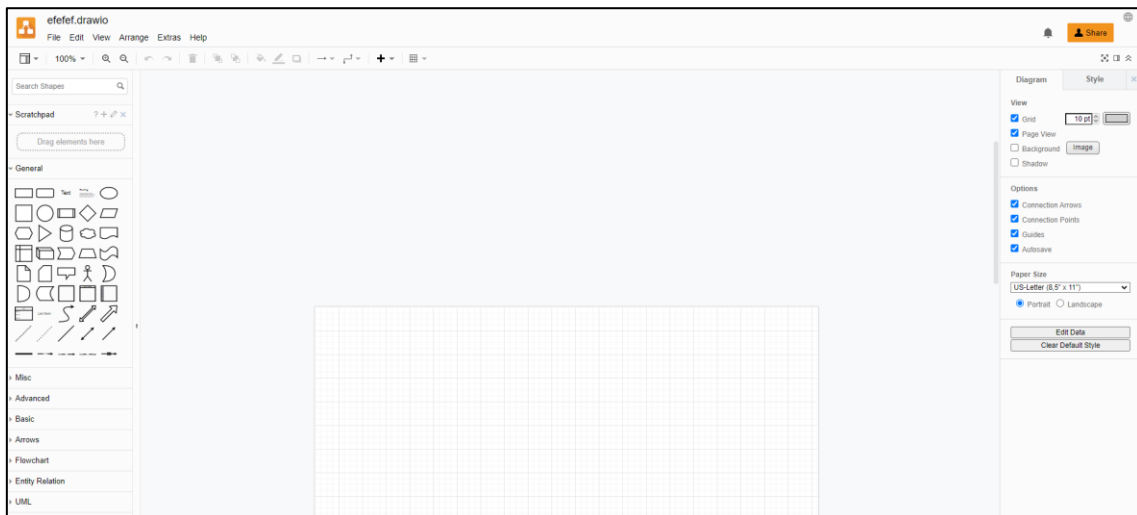


Figura 35- Panel principal de Diagrams.net

5.1.3.2. Creately

Es otra aplicación web que permite crear diagramas al igual que Diagrams.net, proporcionando esquemas predefinidos de diagramas UML, por lo que ha sido utilizado para desarrollar el diagrama UML de las clases de la Base de Datos junto con sus atributos.

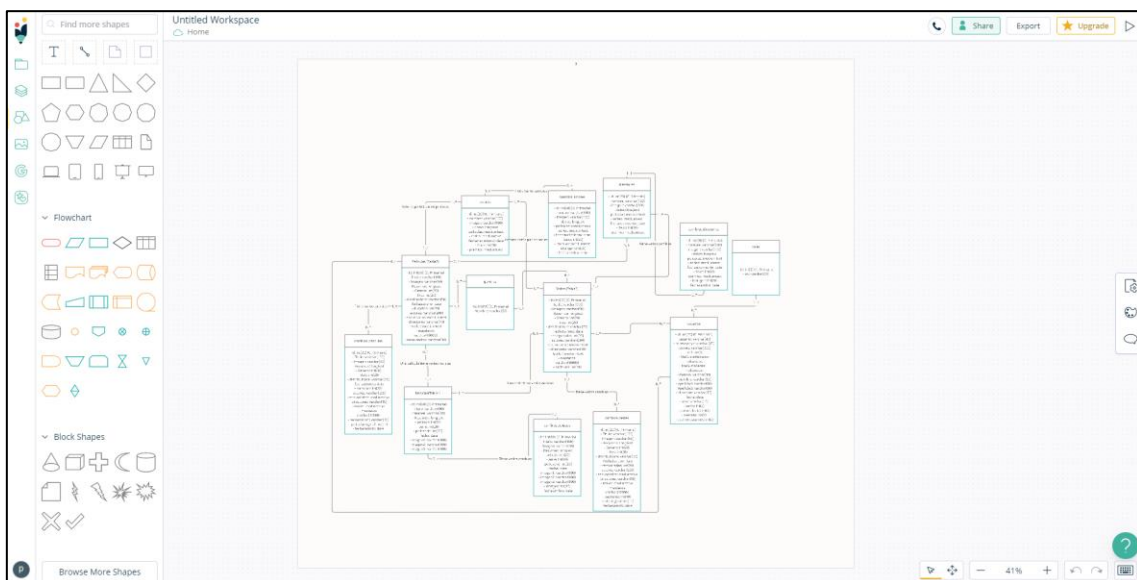


Figura 36- Panel principal de Creately

5.1.3.3. Balsamiq

Balsamiq es una herramienta que permite la realización de *wireframes*, ya que proporciona muchos elementos que simulan un navegador web, además de incluir



Base de datos cinematográfica con sistema de recomendaciones por perfiles de usuarios

muchos elementos que se utilizan en el lenguaje HTML. Esta herramienta se ha utilizado para realizar los esquemas de la capa de presentación.

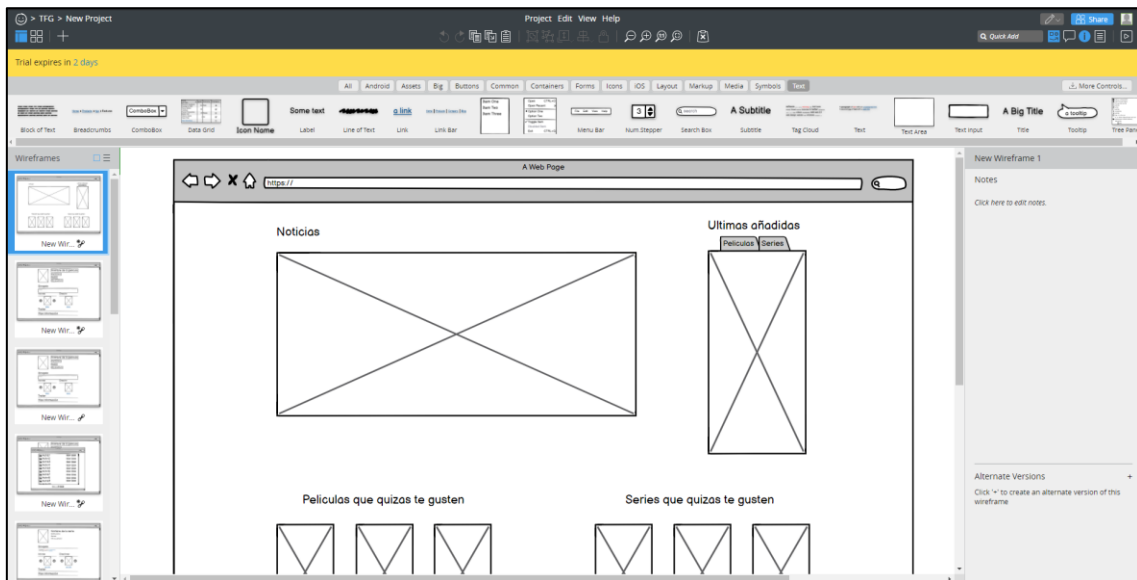


Figura 37- Menú de Balsamiq

5.1.4. GIMP

GIMP⁹ es una aplicación de software libre para la creación y edición de imágenes, siendo utilizada en este caso para el desarrollo del logo de la aplicación, así como de ciertos elementos artísticos que ha sido necesario crear.

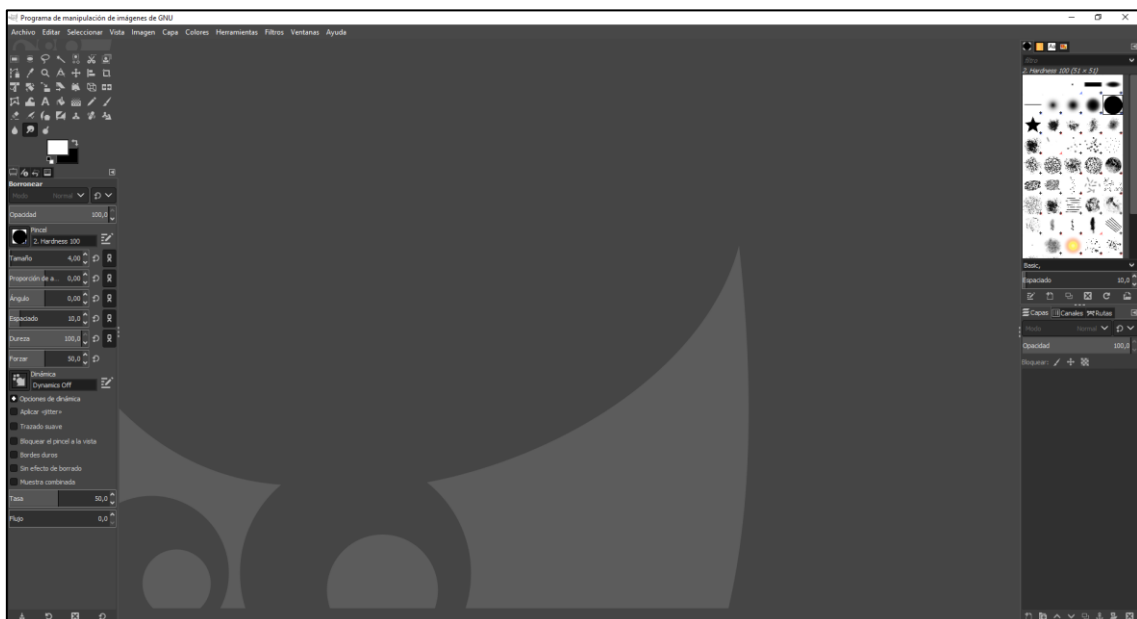


Figura 38- Menú de herramientas de GIMP

⁹ <http://www.gimp.org/es/>

5.1.5. Lenguajes y añadidos utilizados

Aparte de las distintas herramientas utilizadas, al haber sido explicados ya los lenguajes de programación utilizados, este apartado está dirigido a las distintas funcionalidades implementadas mediante diversos plugins

5.1.5.1. *Bootstrap*

Como se ha explicado anteriormente, *Bootstrap* es una biblioteca utilizada en HTML, CSS y JavaScript. Se basa principalmente en un conjunto de clases CSS y extensiones JavaScript en forma de plantillas para añadir algunas funcionalidades extras a HTML. Dentro de Bootstrap, hay varios elementos que han sido de gran utilidad:

- *Grid System*:

El sistema *Grid* (29) de Bootstrap es un sistema de alineación basado en filas y columnas que permite organizar los elementos HTML a lo largo de los documentos. Este sistema es llamado *Grid* ya que tiene forma de red dadas las intersecciones entre filas y columnas. El ancho del documento se divide en 12 columnas, permitiendo organizar distintos elementos en una misma fila, dejando que Bootstrap decida el número de columnas asignadas o asignándolas manualmente. Aparte de esto, las clases se dividen en 5 tamaños que permiten hacer a la interfaz “*responsive*”, es decir, que se pueda redimensionar.

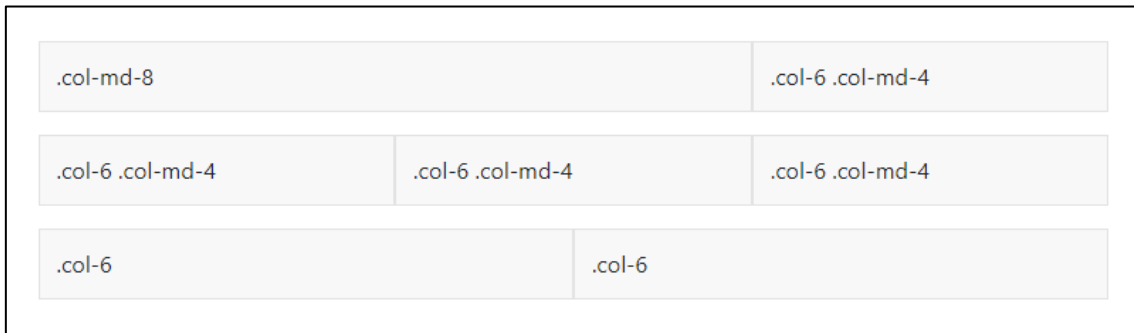


Figura 39- Ejemplo de funcionamiento del sistema Grid

- *Carousel:*

Carousel (30) es el sistema de marco de fotos deslizante, que en el proyecto se utiliza para mostrar distintas películas, series, actores, directores o noticias de una forma sencilla, incluyendo unos controles para movernos por las distintas posiciones del *Carousel* y un pie de imagen para indicar un texto, normalmente un nombre o un título.

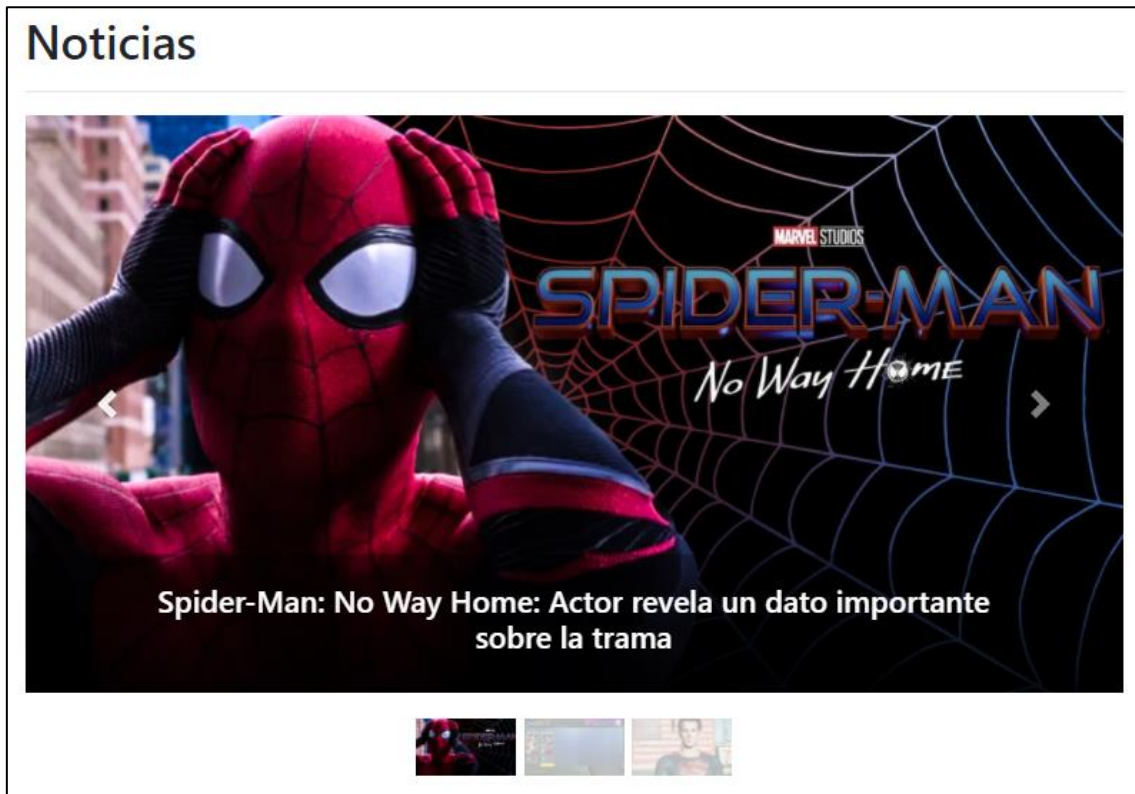


Figura 40- Muestra de funcionamiento de Carousel en el proyecto

Como se puede ver en la imagen, con una pequeña edición del código original, se puede añadir una previsualización de las imágenes en el control manual de las imágenes, además de poder formatear la imagen de la forma que se necesite.

- Botones:

Bootstrap incluye un conjunto de botones con distintos colores o iconos para distintas funcionalidades, como pueden ser botones de emergencia, botones de confirmación o menús desplegados mediante un botón, lo que permite una gran personalización en el sistema.

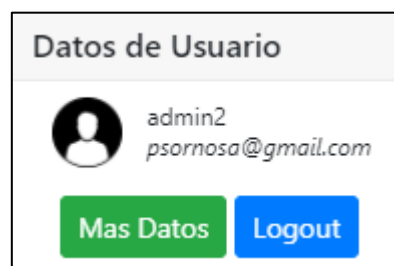


Figura 41- Ejemplo de 2 tipos de botones usados en el proyecto

- Modals

Los *modals* son ventanas emergentes dentro del propio documento HTML, que permiten mostrar contenido auxiliar sin necesidad de perder los datos que se están mostrando en ese momento, ya que en el momento que se cierra la ventana emergente, todo sigue en su lugar. Es muy útil en el proyecto para el inicio de sesión o el registro, así como para mostrar listas con más datos sobre una película o para mostrar ciertas imágenes en pantalla completa.

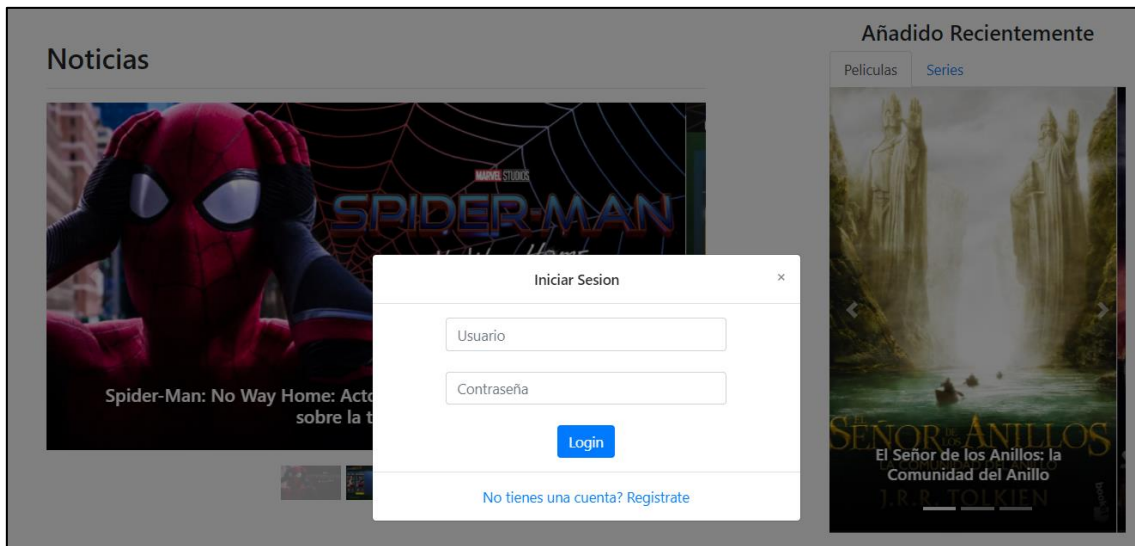


Figura 42- Ejemplo de modal usado en el proyecto

Como se puede ver en la figura, todo el contenido continúa en su lugar, el modal solo se superpone con una animación, pudiendo cerrarlo en cualquier momento.

5.1.5.2. DataTables

*DataTables*¹⁰ es un plugin implementado para jQuery que permite transformar los elementos de tipo `<table>`, añadiendo funcionalidades como paginación, ordenación por las distintas columnas de la tabla o búsqueda, aparte de darle un estilo más cuidado.

¹⁰ <https://datatables.net/>

Show entries Search:

Nombre	Año de estreno	Genero	Distribuidora
Iron Man (Seis)	2001	Superheroes	Warner Bros. Pictures
El Señor de los Anillos: la Comunidad del Anillo	2001	Aventura	New Line Cinema
El Señor de los Anillos: la Comunidad del Anillo (Cinco)	2001	Aventura	Warner Bros. Pictures
El Señor de los Anillos: la Comunidad del Anillo (Cuatro)	2001	Aventura	Warner Bros. Pictures
El Señor de los Anillos: la Comunidad del Anillo (Par)	2001	Superheroes	New Line Cinema
El Señor de los Anillos: la Comunidad del Anillo (Tres)	2001	Superheroes	Warner Bros. Pictures
Harry Potter y la Piedra Filosofal (Cinco)	2001	Fantastico	Warner Bros. Pictures
Harry Potter y la Piedra Filosofal (Cuatro)	2001	Fantastico	Warner Bros. Pictures
Harry Potter y la Piedra Filosofal (Par)	2001	Superheroes	Warner Bros. Pictures
Harry Potter y la Piedra Filosofal (Tres)	2001	Superheroes	Warner Bros. Pictures
Harry Potter y la Piedra Filosofal ha surgido el cambio	2001	Fantastico	Warner Bros. Pictures
Iron Man	2001	Superheroes	Warner Bros. Pictures
Iron Man	2021	Superheroes	sefsefef
Iron Man (Cinco)	2001	Superheroes	Warner Bros. Pictures
Iron Man (Cuatro)	2001	Superheroes	Warner Bros. Pictures

Showing 1 to 15 of 17 entries Previous Next

Figura 43- Ejemplo de DataTable

5.1.5.3. Select Picker

Select Picker¹¹ es un plugin para jQuery que se utiliza para añadir distintas funcionalidades a los elementos `<select>` como pueden ser multi-selección o un cuadro de búsqueda para `selects` con muchas opciones, para hacer la búsqueda más sencilla.

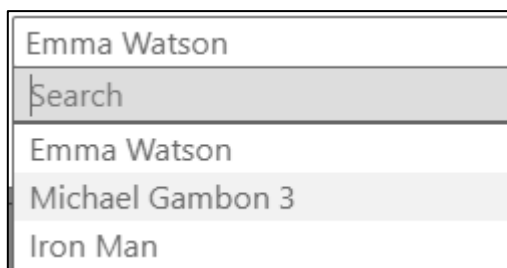


Figura 44- Select Picker con búsqueda usado en el proyecto

5.1.5.4. jQuery

Como se ha explicado anteriormente, es una implementación orientada exclusivamente al desarrollo de páginas web, incluyendo, aparte de unos métodos más sencillos para la manipulación del DOM, un sistema de interacciones llamado AJAX, que es muy útil para obtener datos mediante la ejecución de código en la parte del servidor.

¹¹ <https://picker.uhlir.dev/#about>

```

$("#cambiardirector").click(function(){
    var id = $("#selectdirector").val();
    var indexpelicula = $("#indexpelicula").val();
    var datos = {"id":id,"indexpelicula":indexpelicula};
    $.ajax({
        type: "post",
        url: 'ajax/editardirectorpelicula.php',
        data:datos,
        success: function(respuesta) {
            actualizarSelectDirectores();
            actualizarCarouselDirectores();
        }
    });
})

```

Figura 45- Fragmento de código utilizando métodos jQuery

En la imagen se puede ver una implementación de AJAX y de métodos jQuery para obtener datos de los elementos HTML y mandarlos a un archivo PHP mediante un evento iniciado por un clic en un botón HTML.

5.2. Implementación de algoritmos de recomendación

Dentro del proyecto, se ha especificado como un punto clave la implementación de un sistema de recomendaciones para usuarios. En referencia a esto, se han desarrollado dos sistemas distintos.

5.2.1. Algoritmo de géneros

Ambos métodos están basados en la idea de la tecnología Almas Gemelas de *FilmAffinity* que se ha comentado anteriormente en la memoria.

Esta idea es juntar a los usuarios según su género recomendado, que se genera dinámicamente cada vez que da un "Me Gusta" a una serie o una película, es decir, cada vez que ocurre esto, se recalcula el género mayoritario según el registro de Me Gusta que tenga el usuario.

```

$arraylistaseries = $row["MeGustaSeries"];
$arraylistaseries = explode(",", $arraylistaseries);
$arraylistapelículas = $row["MeGustaPelículas"];
$arraylistapelículas = explode(",", $arraylistapelículas);
mysqli_query($conn, $sql);
$GenerosLikes = array();
if(count($arraylistaseries)>0){
    foreach ($arraylistapelículas as $valor){
        $result2 = mysqli_query($conn, "SELECT * FROM `tabla2` WHERE `id`=$valor");
        $row2 = mysqli_fetch_array($result2);
        $genero = $row2["Genero"];
        if(isset($GenerosLikes[$genero])){
            $valor = $GenerosLikes[$genero];
            $valor++;
            $GenerosLikes[$genero] = $valor;
        }
        else{$GenerosLikes[$genero] = 1;}
    }
}
if(count($arraylistaseries)>0){
    foreach ($arraylistaseries as $valor){
        $result3 = mysqli_query($conn, "SELECT * FROM `tabla3` WHERE `id`=$valor");
        $row3 = mysqli_fetch_array($result3);
        $genero = $row3["Genero"];
        if(isset($GenerosLikes[$genero])){
            $valor = $GenerosLikes[$genero];
            $valor++;
            $GenerosLikes[$genero] = $valor;
        }
        else{$GenerosLikes[$genero] = 1;}
    }
}
$generoPrincipal = 0;
$numerogenero = 0;
foreach ($GenerosLikes as $clave => $valor){
    if($valor > $numerogenero){
        $generoPrincipal = $clave;
    }
}
$sql = "UPDATE `usuarios` SET `Genero`='$generoPrincipal' WHERE `id`=$usuario";
mysqli_query($conn, $sql);

```

Figura 46- Fragmento de código del sistema de selección de género

Como se puede ver en la imagen, el sistema de género recomendado funciona de la siguiente forma: se recogen los datos de “Me Gusta” tanto de series como de películas, que se encuentra en formato de *string* y se convierte en *array*.

Después se clasifican los “Me Gusta” por géneros mediante un array en el que cada género se identifica porque su posición en el *array*, es decir, su clave, coincide con el identificador de dicho género en la Base de Datos. Después, una vez recorridas ambas listas y clasificados sus Me Gusta, se recorre la nueva lista para encontrar el género con mayor número de Me Gusta para ese usuario, para después insertar en la Base de Datos el identificador del género en el registro de dicho usuario.

5.2.2. Algoritmo de Películas del género principal.

El primer método para obtener recomendaciones se basa en la búsqueda de películas y series de usuarios con el mismo género principal que el del usuario actual (de ahí la idea base del sistema “Almas Gemelas”), intentando la selección de elementos que no haya marcado el usuario como “Me Gusta”.

Aquí se ha encontrado una problemática, y es que, ante la posibilidad de que un usuario haya dado una cantidad de “Me Gusta” que abarque la mayoría de los registros de la Base de Datos, se ha optado por añadir un mecanismo para que el algoritmo no se congele, ya que continúa una búsqueda continua de películas o series no marcadas como “Me Gusta”.

```
function repetido($numero, $arraynumero){
    $res = false;
    foreach ($arraynumero as $valor) {
        if($numero == $valor){$res = true;return $res;}
    }
    return $res;
}
function meGusta($id,$arraylikes){
    $res = false;
    foreach ($arraylikes as $valor) {
        if($id == $valor){$res=true; return $res;}
    }
    $res = false;
}
echo '<div class="row justify-content-center">';
$numerodefilas = mysqli_query($conn, "SELECT COUNT(*) as numero FROM `tabla2` WHERE `genero`="._$SESSION["genero_usuario"]."");
$row = mysqli_fetch_array($numerodefilas);
$count = $row["numero"];
$porcentaje = round($count/4);
$result = mysqli_query($conn, "SELECT * FROM `tabla2` ORDER BY likes DESC LIMIT $porcentaje");
$contador = 0;
$arraynumero = array();
while($row = mysqli_fetch_array($result)){
    $json[] = $row;
}
$random = $porcentaje - 1;
while($contador < 3) {
    $numero = random_int(0, $random);
    $id =
    $repetido = true;
    $likespeliculas = explode(",",$SESSION["likesPeliculas"]);
    $repeticiones = 0;
    while($repetido==true && $repeticiones < 100){
        $numero = random_int(0, $random);
        $repetido = repetido($numero,$arraynumero);
        if($repetido==false){
            $id = $json[$numero]["id"];
            $repetido = meGusta($id,$likespeliculas);
        }
        $repeticiones++;
    }
}
```

Figura 47- Código del algoritmo recomendador de género principal

Como se puede ver en la figura, en primer lugar se obtiene el número de películas (hay otro sistema exactamente igual, pero para las series) del género principal del usuario, para luego coger el 25% que más “Me Gusta” tiene, para luego escoger 3 películas aleatorias, siempre comprobando que dicha película no la tiene ese usuario en la lista de “Me Gusta” y que no se ha añadido ya a esa lista de 3 películas.



5.2.3. Algoritmo de series de otros géneros

El otro método de recomendaciones se basa en películas de géneros distintos al género principal que les han gustado a usuarios con un género principal igual al del usuario.

```

$numerodefilas = mysqli_query($conn, "SELECT * FROM `usuarios` WHERE
`Genero`=" . $_SESSION['genero_usuario'] . " AND `id` != '" . $_SESSION["usuario"] . "'");
$arrayrows = array();
while($row = mysqli_fetch_array($numerodefilas)) {
    $arrayrows[] = $row;
}
$cuentasaleatorias = array_rand($arrayrows, 3);
$arrayserieslikes = array();
$contador = 0;
while($contador < 3){
    $valor = $arrayrows[$cuentasaleatorias[$contador]];
    $series = $valor["MeGustaSeries"];
    $arrayseries = explode(",",$series);
    foreach($arrayseries as $valor){
        $numerodefilas2 = mysqli_query($conn, "SELECT * FROM `tabla3` WHERE `id`=$valor");
        $row2 = mysqli_fetch_array($numerodefilas2);
        if($row2["Genero"] != 1){
            if(isset($arrayserieslikes[$valor])){
                $numero = $arrayserieslikes[$valor];
                $numero++;
                $arrayserieslikes[$valor] = $numero;
            }
            else{
                $arrayserieslikes[$valor] = 1;
            }
        }
    }
    $contador++;
}
arsort($arrayserieslikes);

```

Figura 48- Código de algoritmo recomendador de otros géneros

Como se puede ver en la figura, se buscan todos los usuarios que tienen el género principal igual al del usuario con sesión iniciada, pero que no sean dicho usuario. Luego, se obtienen 3 claves aleatorias del array que contiene dichos usuarios (pueden ser todos los usuarios que se necesiten, actualmente está inicializado a 3 ya que solo hay 4 usuarios creados en la Base de Datos) y se obtiene la lista de “Me Gusta” de dicho usuario, y se trata guardando la cantidad de “Me Gusta” como un par clave-valor de un array, siendo el valor el número de “Me Gusta” que tiene esa serie y la clave el identificador de la serie en la Base de Datos. Luego se reordena la lista, ocupando los primeros lugares las series con más “Me Gusta”.

5.3. Implementación de la Base de Datos

Para la implementación de la base de datos se ha optado por no incluir restricciones de integridad más que un identificador como clave primaria para cada una de las filas de todas las tablas, ya que pueden llegar a complicar en gran medida a la hora de crear o editar ciertos elementos, optando por hacer las asociaciones manualmente, lo que lo hace más sencillo.

5.3.1. Usuarios

La tabla usuarios ha sido creada con columnas para los datos personales, como nombre y apellidos, fecha de nacimiento o sexo. Por otra parte, el usuario y el correo se guardan sin ningún tipo de cifrado en la base de datos, pero la contraseña se guarda utilizando un cifrado MD5, y cuando el usuario quiere iniciar sesión, se comprueban los MD5 y se inicia sesión o no dependiendo si es correcto.

También hay dos columnas en las que se guardan los registros de las películas y las series a las que el usuario le ha dado “Me Gusta”, así como unas columnas para diversas opciones de seguridad y otra que indica el rol que tiene el usuario en la Base de Datos.

5.3.2. Tablas de registros

Las tablas de registros hacen referencia a las tablas de películas, series, actores, directores y noticias. Estas tablas guardan toda la información que se muestra cuando se accede a la página de visualización, así como las páginas de edición siempre y cuando se acceda con un usuario que tenga el rol de Administrador.

Estas tablas también relacionan las películas y las series con actores y editores y viceversa, las películas y las series tienen una lista de los actores y directores que participan en dicha película, y los actores y directores una lista de películas y series. Por otra parte, las noticias se relacionan con las películas y las series, además de incluir un valor binario para indicar si hace referencia a una serie o una película.

5.3.3. Tablas de cambios

Desde las etapas iniciales del proyecto, se optó por un sistema de roles, que incluía un mecanismo de cambios basado en *git*. Los usuarios editores pueden editar los elementos ya creados en la base de datos, pero para evitar posibles errores en los datos, los cambios hechos por estos usuarios no se aplican directamente, se guardan en una tabla de cambios, en los que se relaciona el cambio con la película a la cual hace referencia, así como una fecha (Los cambios son anónimos, no hay ningún registro que indique el nombre del usuario que ha hecho el cambio).

Luego se pueden administrar estos cambios pudiendo editar uno por uno todos los campos del registro, así como descartar el registro de edición cuando ya se hayan hecho los cambios pertinentes.



5.3.4. Tablas de apoyo

Estas tablas se utilizan como registros de valores prefijados, como pueden ser los roles de los usuarios o los géneros de las películas y las series. También se utilizan para poder crear selectores para estos valores, para luego añadir los identificadores dentro del campo en el que deba incluirse en las tablas de películas, series o usuarios.

6. Pruebas

Antes de finalizar el desarrollo del proyecto, es necesario hacer diversas pruebas para poder asegurar el correcto funcionamiento de la aplicación. Dado que la solución escogida ha sido una aplicación web, además de la necesidad de comprobar el propio funcionamiento, se debe comprobar que pueda funcionar el cualquier tipo de navegador (Siempre ciñéndose a un ordenador personal, ya que al tratarse de una aplicación que contiene gran cantidad de información e imágenes, se ha optado por no adaptarla a los teléfonos móviles).

Por otra parte, el lenguaje HTML es un lenguaje con mucho margen de errores, es decir, aunque el código tenga errores, los navegadores lo interpretarán de una manera correcta en muchos casos a no ser que se haya cometido un error importante, por lo cual a veces es recomendable utilizar un validador de HTML como el que proporciona el W3C, pero en este caso, al trabajar con *plugins* externos, PHP, jQuery y Bootstrap, las validaciones se complican, ya que hay ciertas estructuras que no se reconocen en el validador. Esto no supone un gran problema, ya que, por una parte, se puede comprobar que está todo correcto en las validaciones de los navegadores, y por otra, muchas páginas, como puede ser la página de inicio de Google, tampoco pasan los validadores del W3C, lo cual no supone ningún tipo de riesgo.

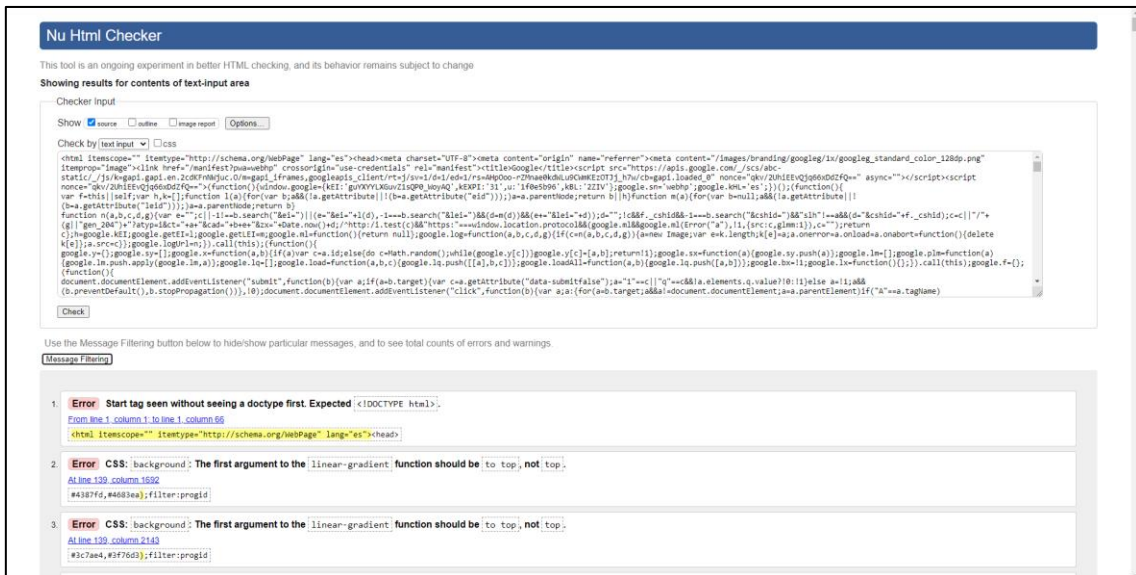


Figura 49- Código fuente de Google.es validado por los validadores del W3C dando errores

6.1. Pruebas en navegadores

Para las pruebas, se van a usar los siguientes navegadores: Google Chrome, Mozilla Firefox y Opera. Se han elegido estos 3 navegadores por diversas razones, la primera es que, según la página *statcounter* (31), los 5 navegadores más utilizados del mundo son Google Chrome, Safari, Microsoft Edge, Firefox, Opera e Internet Explorer, pero no se ha podido acceder a un dispositivo con Safari ya que solo funciona en sistemas



MacOS; para Microsoft Edge aún coexisten 2 versiones, la basada en *Chromium* y la anterior, aunque Microsoft ya ha empezado a sustituir las versiones antiguas por la versión basada en *Chromium*, por lo que los resultados son similares a Google Chrome; y por último, Internet Explorer ya está completamente inutilizado a día de hoy, con solo un 1% de cuota de mercado.

Para comprobar las diferencias, ya que en estas pruebas solo se valora que los elementos HTML se muestren correctamente, se basará en una muestra del menú principal de la aplicación.

6.1.1. Google Chrome

Google Chrome ha sido el navegador principal utilizado para el desarrollo del proyecto, por lo que, en cuanto a diseño, el más similar a los bocetos será este.

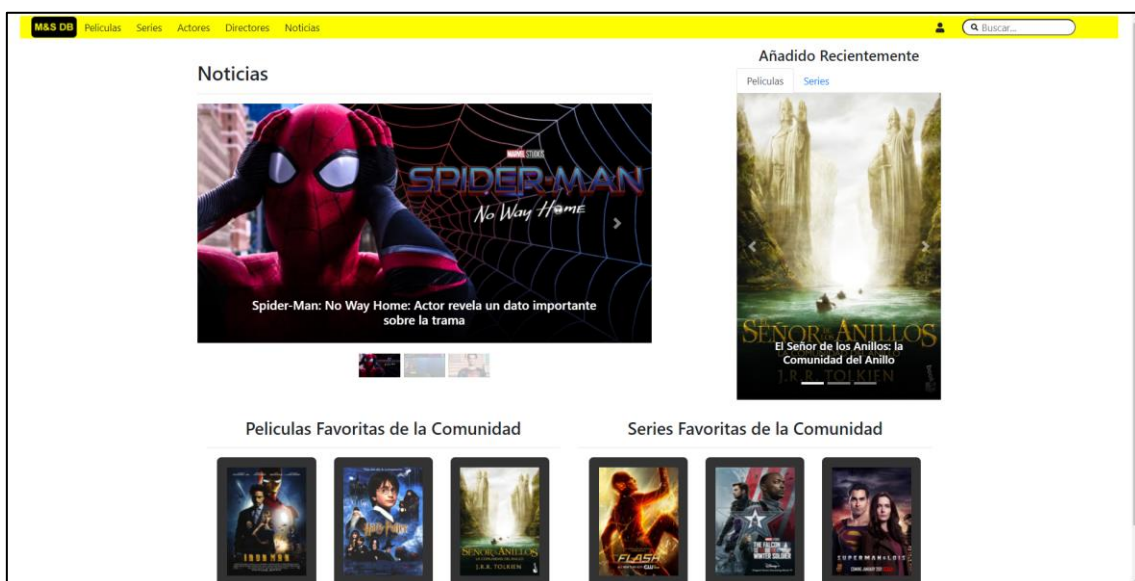


Figura 50- Prueba de la página principal en el navegador Google Chrome

Como se puede ver, todos los elementos se muestran correctamente, y al ser el sistema base, el resto de las pruebas se compararán con esta primera.

6.1.2. Mozilla Firefox

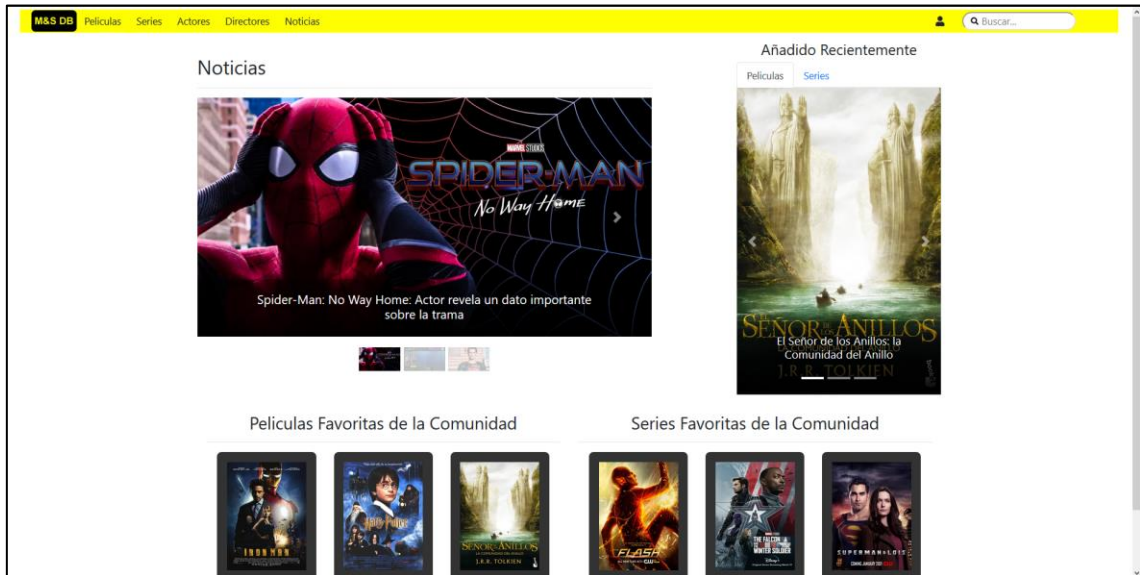


Figura 51- Prueba de la página principal en el navegador Mozilla Firefox

Como se puede ver en la figura, el diseño es exactamente igual al mostrado en la prueba con el navegador Google Chrome, solo cambian algunas decisiones de diseño en cuanto al sombreado de los inputs.

6.1.3. Opera

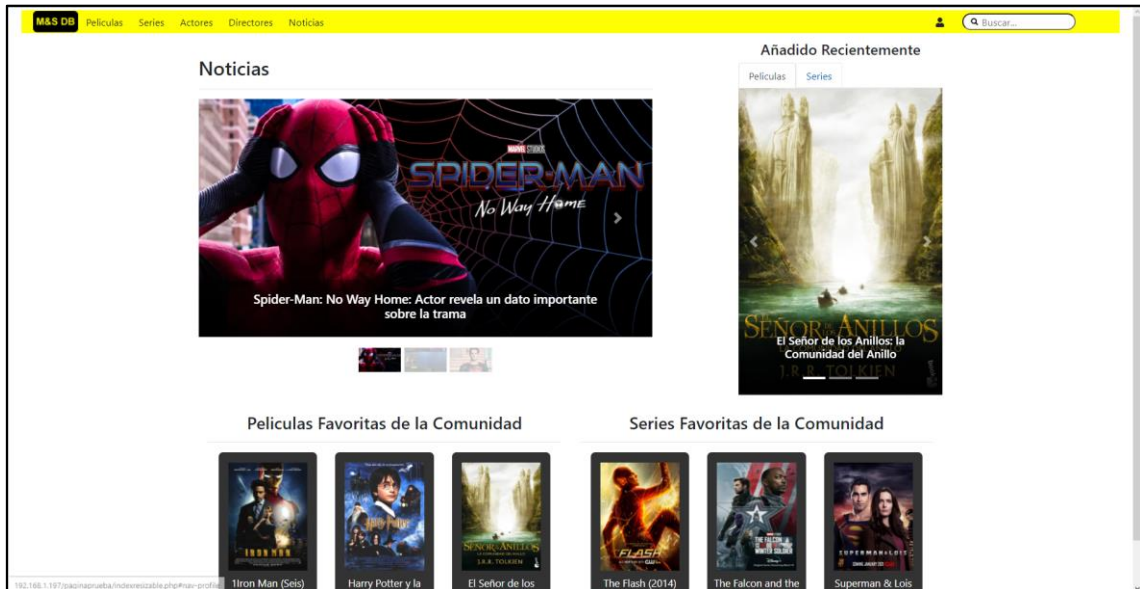


Figura 52- Prueba de la página principal en el navegador Opera

Por último, en el navegador opera todo se muestra exactamente igual que en Google Chrome, por lo que se puede llegar a la conclusión de que no hay ninguna incompatibilidad entre los estilos y los navegadores

6.2. Pruebas de funcionamiento

Para asegurar que la experiencia de uso sea la esperada, se deben hacer pruebas siguiendo un uso común de los usuarios, además de intentar forzar errores y comprobar si la aplicación es capaz de administrar dichos errores o funciona erróneamente.

Por una parte, se van a probar los 4 tipos de usuarios para comprobar si un usuario puede acceder a un sistema en el que no le está permitido acceder. Por otra, se va a probar a hacer tanto inserciones como ediciones de distintos elementos para impedir errores.

6.2.1. Pruebas de seguridad

Como se ha comentado anteriormente, en la aplicación existen 4 tipos de usuarios: el usuario no registrado, el usuario registrado, el usuario editor y el usuario administrador, y cada uno tiene unos permisos distintos, que les impide acceder a ciertos elementos a los que no deberían poder entrar.

6.2.1.1. Usuario no registrado

El usuario no registrado no puede acceder a ediciones de elementos ni a los datos de usuario, y se le debe permitir tanto registrarse como iniciar sesión. No debe mostrar recomendaciones personalizadas

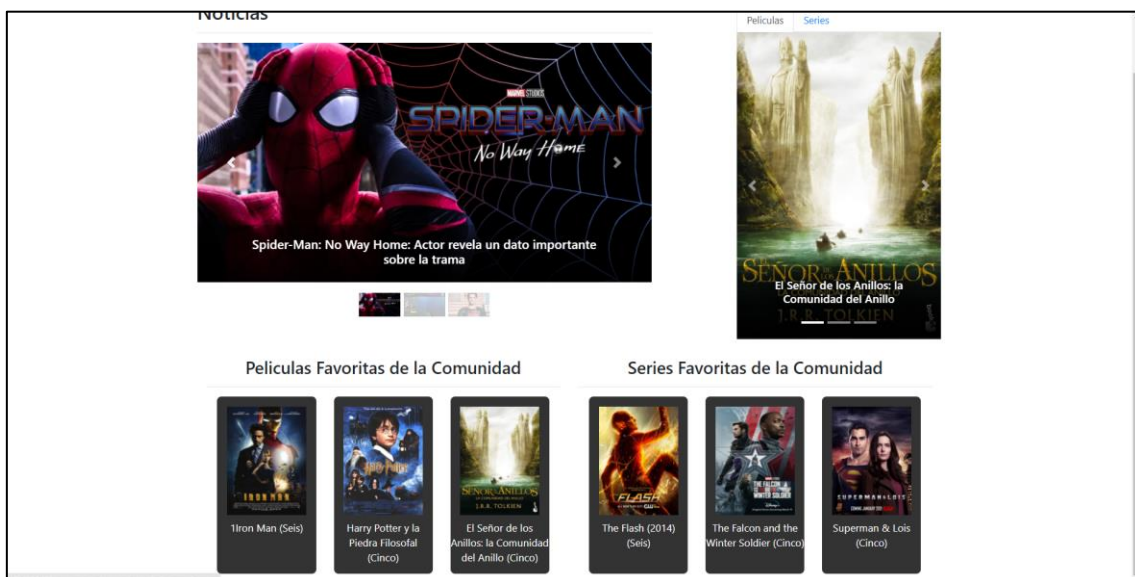


Figura 53- Menú principal sin recomendaciones para usuario no registrado

Como se puede ver en la barra de deslizamiento vertical, el apartado “Películas Favoritas de la Comunidad” se encuentra al fondo de la página, por lo que no se encuentra el apartado de recomendaciones personalizadas.

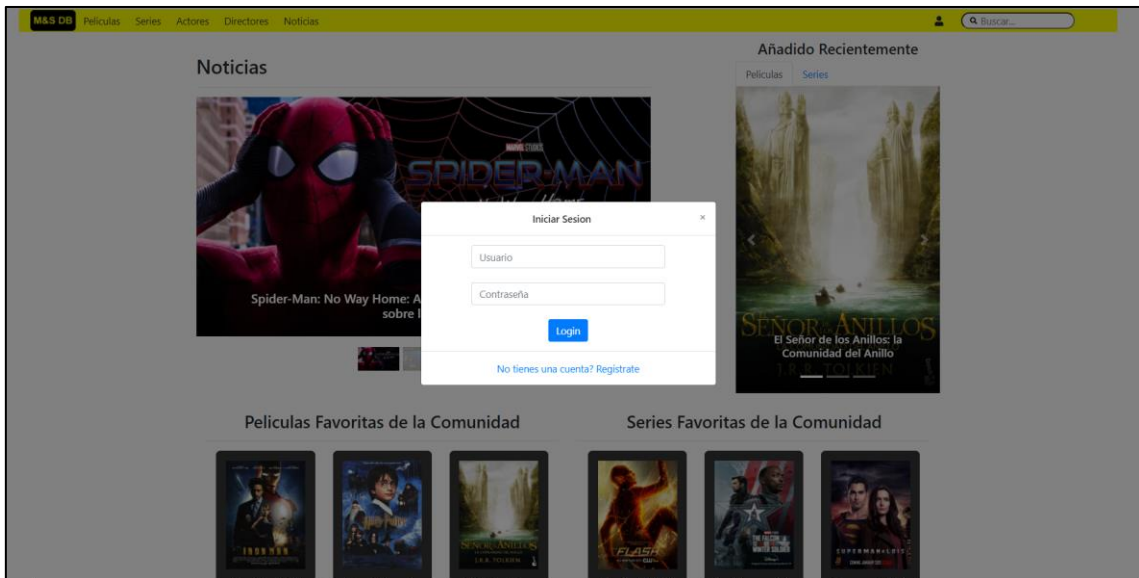


Figura 54- Modal de Inicio de sesión o registro para usuario no registrado

Cuando se pulsa el botón del usuario que se encuentra al lado de la barra de búsqueda se muestra la opción de inicio de sesión o registro, por lo que es imposible para este usuario acceder al apartado de datos del usuario.

```

<?php
if(isset($_SESSION["usuario"])){
?>
<body style="margin-bottom:20px;margin-top:5px;width:100%;height:100%;">...
</body>
<?php
else{echo '<script>>window.open("indexresizable.php","_self");</script>';}}?>

```

Figura 55- Código redireccionador para URL sin permisos

Para evitar que se pueda acceder mediante URL, todas las páginas que tienen algún tipo de restricción de permisos incluyen un código PHP que comprueba unas variables de sesión que se crean cuando el usuario inicia sesión (Y al ser un código que se ejecuta en el servidor, no se pueden editar como pasa con los datos JavaScript y HTML en la herramienta "Inspeccionar" que incluyen muchos navegadores).

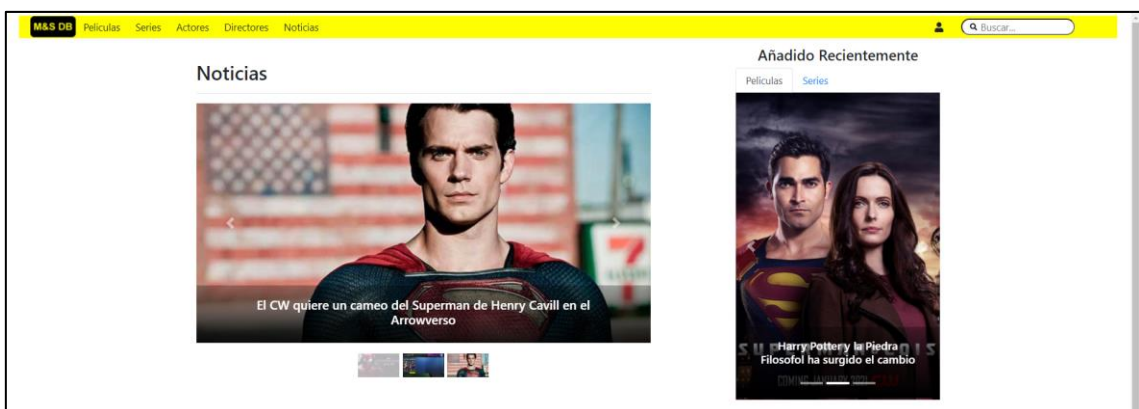


Figura 56- Muestra de menú sin opciones en usuario no registrado



Como se puede ver en la imagen, solo se puede acceder a las listas para ver, no para administrar como si se puede con los roles de editor y administrador.

6.2.1.2. *Usuario con sesión iniciada*

El usuario con sesión iniciada puede acceder a sus datos personales, así como a unas recomendaciones personalizadas.

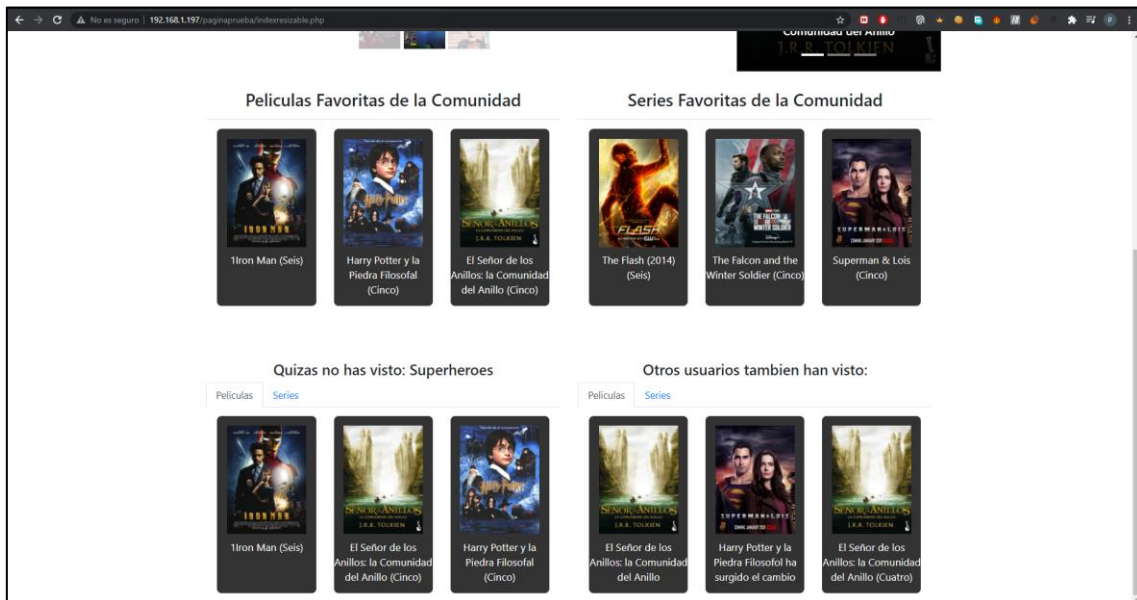


Figura 57- Recomendaciones aparecen con el rol usuario con sesión iniciada

Como se puede ver en la imagen, al contrario que con el usuario no registrado, sí que aparecen las recomendaciones personalizadas para el usuario.

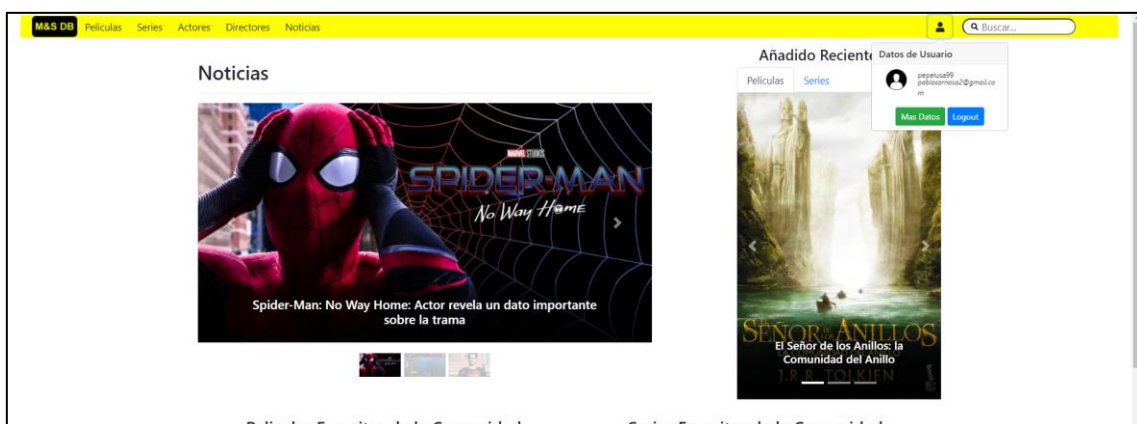


Figura 58- Muestra de menú para cerrar sesión o mostrar datos con usuario con sesión iniciada

Cuando se pulsa el botón de usuario, se muestra un pequeño *pop-up* que muestra una imagen de usuario junto con el nombre de usuario y el correo electrónico, además de 2 botones, uno para ir al apartado de datos del usuario y otro para cerrar sesión.

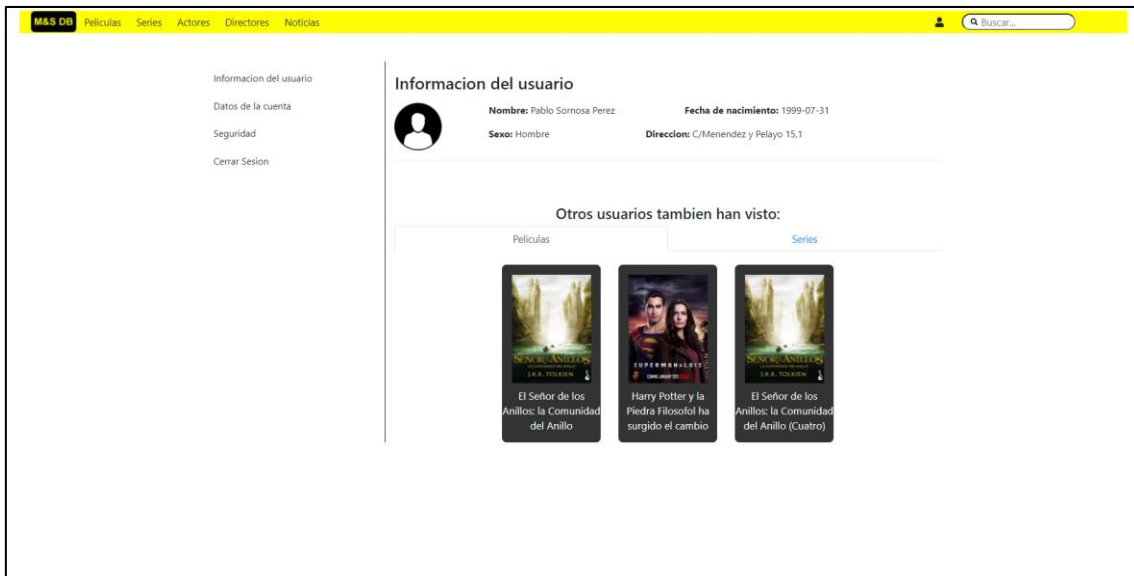


Figura 59- Página de administrar cuenta accesible para usuarios con sesión iniciada

Cuando se pulsa el botón de “Más Datos”, se redirige a la página que se puede ver en la figura, donde se muestran todos los datos del usuario, así como recomendaciones.

Por último, se comprueba el acceso únicamente a las listas para visualización, no para edición. Una muestra de ello es que los menús no incluyen un desplegable, lugar donde se encuentra la opción de administrar.

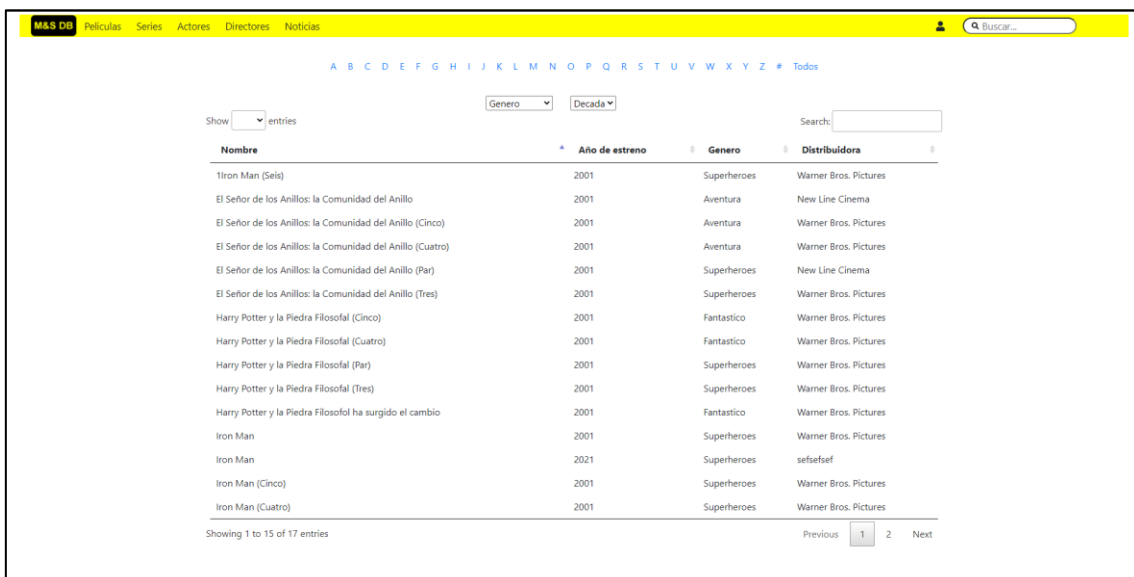


Figura 60- Administración inaccesible para usuarios sin permisos

Y, al igual que en el apartado datos del usuario, en todos los apartados de administrar se encuentra el siguiente código.

```
<?php
    if(isset($_SESSION["usuario"])){
        if($rolsesion == 1 || $rolsesion == 2){
?>
<body style="margin-bottom:20px;margin-top:5px;width:100%;height:100%;">...
</body>
<?php }
else{echo '<script>window.open("indexresizable.php","_self");</script>';}}?>
```

Figura 61- Fragmento de código de seguridad

6.2.1.3. Usuario editor

El usuario editor tiene las mismas opciones que el usuario básico, además de habilitar los sistemas de administración, pero con la condición de que no puede añadir ni eliminar datos y que las ediciones deben ser supervisadas por un usuario administrador.

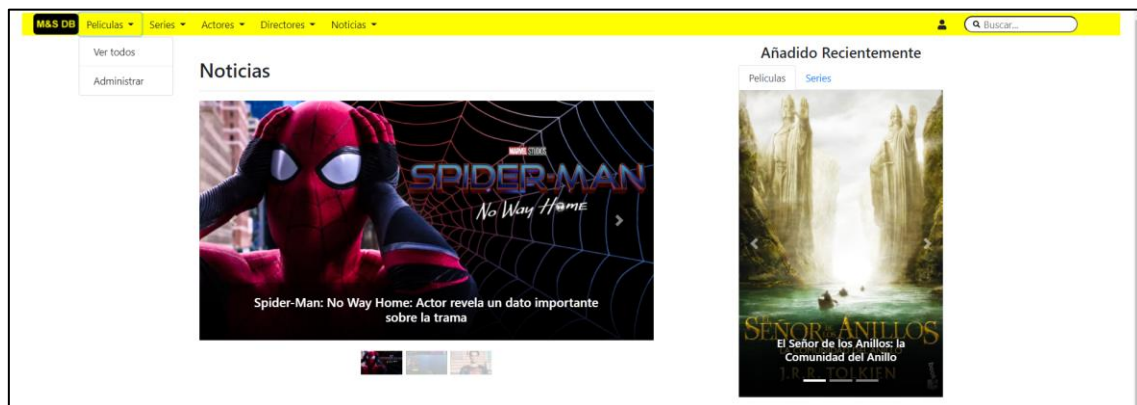


Figura 62- Muestra de que se permite administrar todo salvo cambios con usuario editor

Como se puede ver en la imagen, con el rol editor se muestran los desplegables de administración para cada tipo de registro, pero no se muestra el botón administrar ya que este usuario no puede administrar los cambios que han hecho otros usuarios editores.

Cuando se pulsa el botón administrar, se muestra lo siguiente:

Nombre	Año de estreno	Genero	Distribuidora	Administrar
Iron Man (Seis)	2001	Superheroes	Warner Bros. Pictures	Editar
El Señor de los Anillos: la Comunidad del Anillo	2001	Aventura	New Line Cinema	Editar
El Señor de los Anillos: la Comunidad del Anillo (Cinco)	2001	Aventura	Warner Bros. Pictures	Editar
El Señor de los Anillos: la Comunidad del Anillo (Cuatro)	2001	Aventura	Warner Bros. Pictures	Editar
El Señor de los Anillos: la Comunidad del Anillo (Par)	2001	Superheroes	New Line Cinema	Editar
El Señor de los Anillos: la Comunidad del Anillo (Tres)	2001	Superheroes	Warner Bros. Pictures	Editar
Harry Potter y la Piedra Filosofal (Cinco)	2001	Fantastico	Warner Bros. Pictures	Editar
Harry Potter y la Piedra Filosofal (Cuatro)	2001	Fantastico	Warner Bros. Pictures	Editar
Harry Potter y la Piedra Filosofal (Par)	2001	Superheroes	Warner Bros. Pictures	Editar
Harry Potter y la Piedra Filosofal (Tres)	2001	Superheroes	Warner Bros. Pictures	Editar
Harry Potter y la Piedra Filosofal ha surgido el cambio	2001	Fantastico	Warner Bros. Pictures	Editar
Iron Man	2021	Superheroes	sefsef	Editar
Iron Man	2001	Superheroes	Warner Bros. Pictures	Editar
Iron Man (Cinco)	2001	Superheroes	Warner Bros. Pictures	Editar
Iron Man (Cuatro)	2001	Superheroes	Warner Bros. Pictures	Editar

Figura 63- Muestra de la imposibilidad de añadir o eliminar para el usuario editor

Como se puede ver, no se muestran opciones para eliminar ni añadir nuevas películas, ya que se incluye un código como los vistos anteriormente pero con el usuario de rol con identificador 1.

Cuando se pulsa el botón de editar, se abre la página de edición, pero al ser el usuario editor, se crea un nuevo registro en las tablas de cambios y se redirige a dicha página mediante el siguiente código.

```

<?php
    if(isset($_SESSION["usuario"])){
        if($rolsesion == 1 || $rolsesion ==2){
            if($rolsesion == 2){
                $id = $_GET["id"];
                $result = mysqli_query($conn, "INSERT INTO `cambios_peliculas` (`Titulo`, `Imagen`, `URL`,
                $id = mysqli_insert_id($conn);
                echo '<script>>window.open("editar pelicula editor.php?id='.$id.'","_self");</script>';
            }
        }
    }
?>
<body style="margin-bottom:20px;margin-top:5px;width:100%;height:100%;"> ...
</body>
<?php
else{echo '<script>>window.open("indexresizable.php","_self");</script>';}
}
else{echo '<script>>window.open("indexresizable.php","_self");</script>';}
?>

```

Figura 64- Código para redirigir los cambios a una tabla secundaria para usuarios editores



Una vez hecho eso, se redirige a la página:

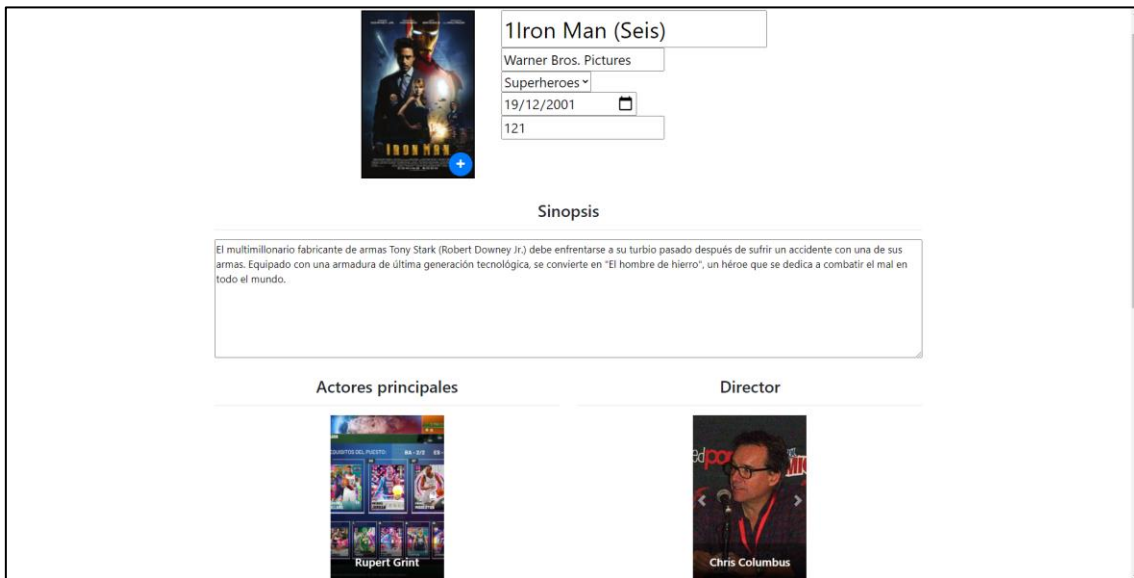


Figura 65- Página de cambios para usuarios editores

6.2.1.4. Usuario administrador

Al contrario que el usuario editor, las listas de administración se ven de la siguiente forma:

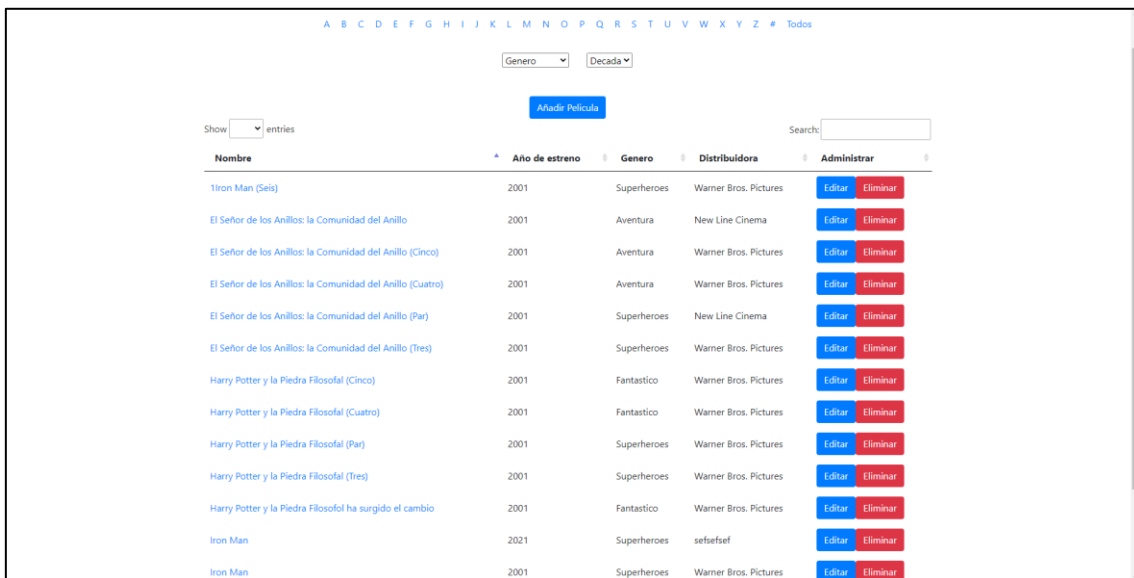


Figura 66- Muestra de que el usuario administrador puede borrar y añadir datos

Se genera un botón de añadir película y un botón de eliminar cada una de las películas, cuando se pulsa el botón de eliminar, se recarga la página para mostrar los datos correctamente. Si se pulsa el botón de añadir, se redirige a la página de crear película.

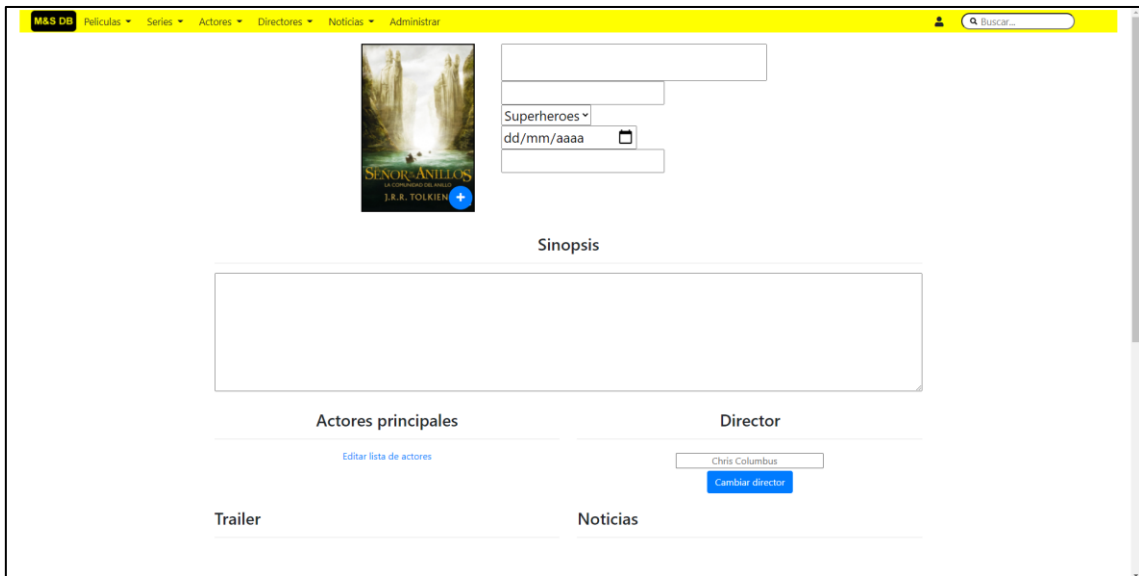


Figura 67- Página de creación de películas

Por último, se puede acceder a la opción de administrar, que permite controlar los cambios que han hecho usuarios editores en diversos elementos.



Figura 68- Lista de cambios

Cuando se pulsa en el enlace del nombre, se accede a una página en la que se puede administrar cada uno de los cambios.

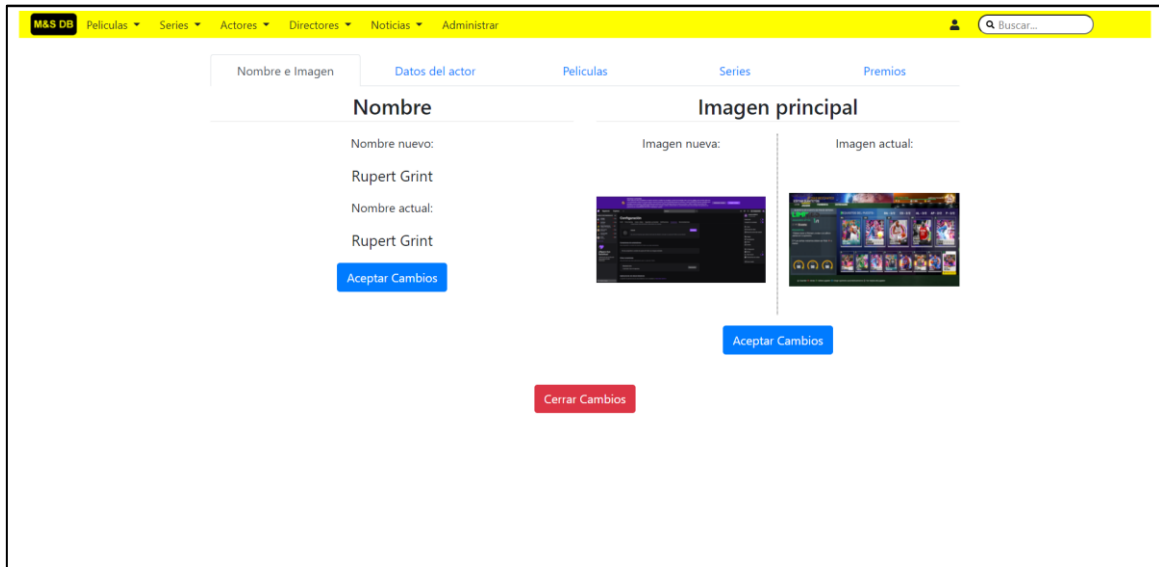


Figura 69- Página de administración de cambios

La seguridad de estas páginas se controla de la siguiente manera, al igual que el resto de los sistemas de seguridad, que se basan en las variables de sistema que se crean al iniciar sesión, pero en este caso se administra simplemente al rol con identificador 1.

```
<?php if(isset($rolsesion) && $rolsesion == 1){ ?>  
<body style="margin-bottom:20px;margin-top:5px;width:100%;height:100%;">...  
</body>  
<?php else{echo '<script>window.open("indexresizable.php","_self");</script>';} ?>
```

Figura 70- Código de seguridad para administradores

6.2.2. Administración de errores

Hay 3 casos en los que puede dar errores la aplicación, cuando se edita un elemento y se dejan campos vacíos, cuando se crea un elemento y no se rellenan todos los datos y cuando se cierra una creación de elemento antes de guardar los datos.

6.2.2.1. Edición errónea de elementos

Para comprobar cómo se administran los errores, si se deja un campo en blanco y se pulsa el botón de guardar, se muestra una alerta de rellenar los datos para continuar.

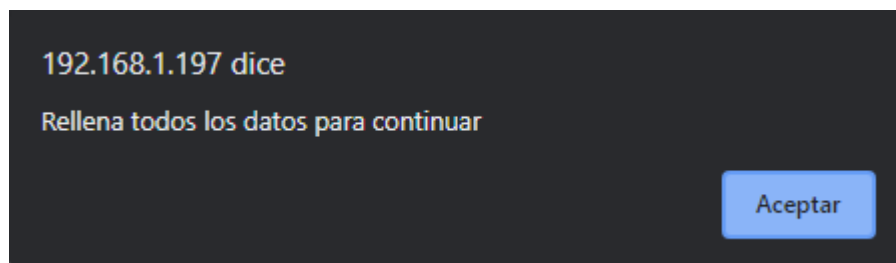
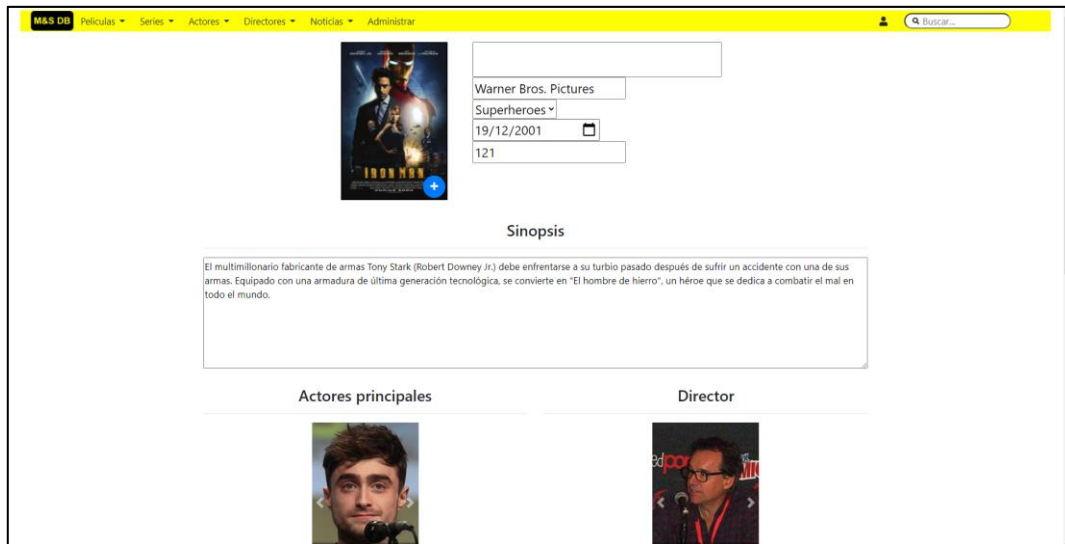


Figura 71- Error de campos vacíos en edición de películas

En el caso de que, por ejemplo, se añadieran letras al apartado de duración, también se mostraría una alerta.

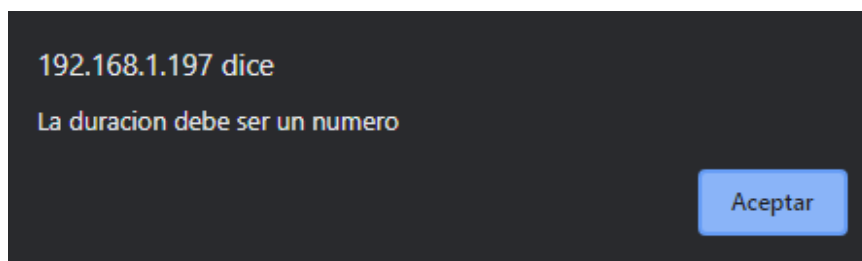


Figura 72- Error de campo incorrecto en la edición de películas

6.2.2.2. Creación errónea de elementos

Si se intentan añadir elementos sin todos los datos completos, aparece el mismo error que con la edición. Lo mismo ocurre con los datos con valores erróneos.

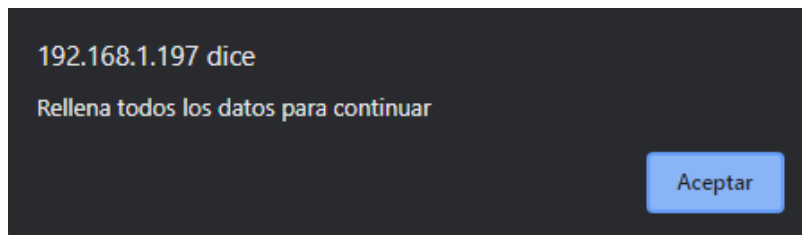


Figura 73- Error de campos vacíos en creación de películas

6.2.2.3. Eliminación de tablas incorrectas

Dado que ciertos elementos como los actores o los directores no están sujetos a guardar los cambios mediante el botón de guardar cambios, un mal funcionamiento podría generar que la aplicación tuviera muchos fallos. Por esto, se optó por añadir un método que se ejecuta cada vez que se carga una nueva página que busca en cada una de las tablas registros incompletos.

```
#!/usr/bin/php
$arraytablas = array("actores","cambios_actores","cambios_directores","cambios_noticias","cambios_peliculas","cambios_series","directores","tabla2","tabla3","tabla4");
foreach ($arraytablas as $tabla){
    $sql = "SELECT * FROM '$tabla' ";
    $result = mysqli_query($conn, $sql);
    while($row = mysqli_fetch_array($result)){
        $id = $row["id"];
        if(str_contains($tabla, 'tabla2') || str_contains($tabla, 'tabla3') || str_contains($tabla, 'series') || str_contains($tabla, 'peliculas')){
            if($row["actores"] == "" || $row["secundarios"] == "" || $row["directores"] == "" || $row["imagen"] == "" || $row["titulo"] == ""){
                mysqli_query($conn, "DELETE FROM '$tabla' WHERE 'id' = '$id'");
            }
        }
        else if(str_contains($tabla, 'actores') || str_contains($tabla, 'directores')){
            if($row["peliculas"] == "" || $row["series"] == "" || $row["imagen"] == "" || $row["nombre"] == ""){
                mysqli_query($conn, "DELETE FROM '$tabla' WHERE 'id' = '$id'");
            }
        }
        else{
            if($row["imagen1"] == "" || $row["imagen2"] == "" || $row["imagen3"] == "" || $row["imagen"] == "" || $row["titulo"] == ""){
                mysqli_query($conn, "DELETE FROM '$tabla' WHERE 'id' = '$id'");
            }
        }
    }
}
```

Figura 74- Código de eliminación de registros incompletos

Se comprueban los distintos aspectos que no están relacionados entre sí, ya que, como se ha podido comprobar, no se puede guardar el atributo de distribuidora si el género es vacío, por lo que hacer una comprobación de todas las columnas sería un coste de computación innecesario.



7. Conclusiones

Tras la realización de este proyecto, se han alcanzado unas conclusiones, tanto relacionadas con los objetivos iniciales que se han podido o no llevar a cabo, con las dificultades técnicas que han ocurrido, como conclusiones relacionadas con los estudios cursados, centrándose en este caso en las dificultades que conllevan los conocimientos adquiridos durante los 4 años de carrera.

7.1. Objetivos iniciales

Al principio del proyecto se especificaron unos objetivos iniciales a cumplir durante la realización de proyectos. El principal objetivo era, textualmente: *“ofrecer una optativa a las páginas de colecciones de películas y series tales como IMDb o Rotten Tomatoes que, en vez de centrar las recomendaciones y valoraciones en base a las opiniones de expertos, darle a la aplicación otro enfoque, basado en los sistemas de recomendaciones de las redes sociales o los servicios de Streaming”*. Este objetivo era el objetivo básico, y como se ha podido ver, se ha conseguido en un alcance moderado, es decir; ofrecer un rival que haga sombra a páginas tan conocidas como *IMDb* o *Rotten Tomatoes* es muy complicado, ya que son servicios que llevan en funcionamiento muchos años y son páginas muy conocidas, por lo que no es fácil competir con estas. Pero, si se considera que, al ser una página web funcional, aunque aún no tenga una base de datos completa, es una optativa, aunque sea minoritaria, de estas opciones, entonces el objetivo se ha cumplido, aparte también por el hecho de haber implementado un sistema de recomendaciones propio y basado en aplicaciones de *Streaming*.

Relacionado con esto, un objetivo secundario era la posibilidad de orientar las recomendaciones mediante una configuración inicial por género, edad o servicios de *Streaming* que el usuario tenga contratados. Estos objetivos, al ser secundarios, se han implementado a medias por una razón, se tenía la idea inicial de realizar un *ChatBot*, que recomendara automáticamente mediante los servicios de *streaming* contratados por el usuario, pero en su momento se le vio poca utilidad y se descartó. Por otra parte, las recomendaciones por género sí que se han llevado a cabo, aunque esto forma parte del objetivo principal.

Otro objetivo, desarrollando la idea antes mencionada de los sistemas de recomendaciones, era intentar ofrecer variedad, es decir, no repetir una y otra vez unas recomendaciones que ya se han mostrado, se han visto, y se han descartado. Esto se ha implementado en cierta medida mediante el algoritmo de recomendaciones, que elimina las opciones a las que el usuario ya ha dado “Me Gusta”. Eliminar las opciones que el usuario ya ha visto era algo complicado, por una parte, porque la base de datos solo es una implementación básica de lo que debe ser en una aplicación final, por lo que a la 3 o 4 películas o series visitadas se van acabando las opciones de recomendación, pero en un futuro sería interesante hacer un registro de películas visitadas por el usuario.

Por último, un objetivo secundario era la colaboración, en la que cualquier usuario previamente aceptado por un administrador pudiera ayudar en la edición y el mantenimiento de los datos de la página, en un sistema en el que los administradores se encargaran de revisar y confirmar los cambios para evitar contenido no deseado o inventado. Esto se ha llevado a cabo completamente, ya que el sistema basado en *git* se ha implementado de forma correcta utilizando tablas de apoyo para los cambios y permitiendo a los administradores aplicar uno a uno los cambios para ser lo más personalizable posible.

7.2. Relación con los estudios cursados

Al haber cursado la rama de Tecnologías de la información, donde se ofrece una pequeña introducción a algunos aspectos del desarrollo web, como pueden ser HTML, CSS, *JavaScript* o *jQuery* por parte del cliente y *Servlets* por parte del servidor, el desarrollo del proyecto no ha conllevado muchas complicaciones técnicas, ya que tengo experiencia en este tipo de desarrollos dada mi experiencia laboral en este campo.

Basándome únicamente en los conceptos estudiados en la rama, este proyecto está desarrollado de una forma alternativa en ciertos aspectos a lo cursado, ya que, por ejemplo, se enseña a utilizar AJAX en la asignatura de Desarrollo Web para contactar con un Servlet Java, sin embargo, en este proyecto se han sustituido los *Servlets* Java por páginas PHP, que permiten una mayor versatilidad en ciertos aspectos por su gran compatibilidad con HTML, lo contrario que pasa con *Servlets*, ya que son una tecnología algo específica y no demasiado utilizada.



8. Bibliografía

• Páginas y documentos consultados:

1. IMDb. IMDb | Help. [En línea] [Citado el: 19 de mayo de 2021.] https://help.imdb.com/article/imdb/general-information/what-is-imdb/G836CY29Z4SGNMK5?ref=__seemr.
2. Encyclopedia Britannica. *IMDb | History, Features, & Facts*. [En línea] [Citado el: 19 de mayo de 2021.] <https://www.britannica.com/topic/IMDb>.
3. FILMAFFINITY. FilmAffinity. [En línea] [Citado el: 19 de mayo de 2021.] <https://www.filmaffinity.com/es/meetus.php>.
4. Duong, Senh. Rotten-Tomatoes. [En línea] 19 de agosto de 1999. [Citado el: 28 de junio de 2021.] https://web.archive.org/web/20090927130308/http://www.asianconnections.com/entertainment/rotten_tomatoes.php.
5. Rotten Tomatoes. *Rotten Tomatoes: About*. [En línea] [Citado el: 28 de junio de 2021.] <https://www.rottentomatoes.com/about>.
6. Panda Security Mediacyber. *El 52% de los usuarios reutiliza sus contraseñas en distintos servicios*. [En línea] [Citado el: 5 de julio de 2021.] <https://www.pandasecurity.com/es/mediacyber/seguridad/reutilizacion-contrasenas/>.
7. BOE.es - Agencia Estatal Boletín Oficial del Estado. *Disposición 16673 del BOE núm. 294 de 2018*. [En línea] 6 de diciembre de 2018. [Citado el: 6 de agosto de 2021.] <https://www.boe.es/boe/dias/2018/12/06/pdfs/BOE-A-2018-16673.pdf>.
8. Portada MPTFP:: Ministerio de Política Territorial y Función Pública :: *Derecho de Supresión o Derecho al Olvido*. [En línea] [Citado el: 8 de julio de 2021.] <http://www.mptfp.es/portal/ministerio/proteccion-datos/ejercicio-derechos/Derecho-de-supresion.html>.
9. Ayuda Ley Protección Datos. *Derecho a la limitación del tratamiento | Ayuda Ley Protección Datos*. [En línea] 25 de junio de 2021. [Citado el: 8 de julio de 2021.] <https://ayudaleyprotecciondatos.es/derecho-limitacion/>.
10. Ministerio de Cultura. BOE.es. *Real Decreto Legislativo 1/1996*. [En línea] 22 de abril de 1996. [Citado el: 9 de julio de 2021.] <https://www.boe.es/buscar/pdf/1996/BOE-A-1996-8930-consolidado.pdf>.
11. RAE Diccionario Jurídico. *Fair Use*. [En línea] [Citado el: 9 de julio de 2021.] <https://dpej.rae.es/lema/fair-use>.
12. Creative Commons. Creative Commons. *About CC Licenses*. [En línea] [Citado el: 9 de julio de 2021.] <https://creativecommons.org/about/cclicenses/>.

13. OMDb. Open Media Database. *Help:FAQ*. [En línea] [Citado el: 9 de julio de 2021.] <https://en.omdb.org/content/Help:FAQ>.
14. Shibashake. TurboFuture. *Embedded YouTube Videos and Copyright Infringement*. [En línea] 23 de junio de 2009. [Citado el: 5 de agosto de 2021.]
15. FP Online. *Qué lenguajes de programación desarrollan interfaces gráficas*. [En línea] [Citado el: 9 de julio de 2021.] <https://fp.uoc.fje.edu/blog/que-lenguajes-de-programacion-desarrollan-interfaces-graficas/>.
16. Java. *¿Que es Java y para qué es necesario?* [En línea] [Citado el: 10 de julio de 2021.] https://www.java.com/es/download/help/whatis_java_es.html.
17. Developer tools, technical documentation and coding examples | Microsoft Docs. *Uso de Visual Studio con .NET y C# para realizar consultas - Azure SQL Database & SQL Managed Instance*. [En línea] [Citado el: 1 de agosto de 2021.] <https://docs.microsoft.com/es-es/azure/azure-sql/database/connect-query-dotnet-visual-studio>.
18. Medium Multimedia. *Que es un CMS*. [En línea] [Citado el: 1 de agosto de 2021.] <https://www.mediummultimedia.com/que-es/cms/>.
19. World Wide Web Consortium (W3C). *WORLD WIDE WEB CONSORTIUM. HTML & CSS - W3C*. [En línea] [Citado el: 1 de agosto de 2021.] <https://www.w3.org/standards/webdesign/htmlcss>.
20. MDN Web Docs. MDN Web Docs. *JavaScript | MDN*. [En línea] [Citado el: 3 de agosto de 2021.] <https://developer.mozilla.org/es/docs/Web/JavaScript>.
21. PHP: *¿qué es, para qué sirve y cuáles son sus características?* Rock Content - ES. [En línea] [Citado el: 3 de agosto de 2021.]
22. PHP Group. PHP: Hypertext Preprocessor. *PHP: General Information - Manual*. [En línea] [Citado el: 3 de agosto de 2021.] <https://www.php.net/manual/en/faq.general.php>.
23. *¿Qué es el Modelo de Objetos del Documento? HTML Con Clase | Diseño web*. [En línea] [Citado el: 5 de agosto de 2021.] <http://html.conclase.net/w3c/dom1-es/introduction.html>.
24. Bootstrap Team. Bootstrap · The most popular HTML, CSS, and JS library in the world. *Introduction*. [En línea] [Citado el: 6 de agosto de 2021.] <https://getbootstrap.com/docs/5.1/getting-started/introduction/>.
25. APACHE FRIENDS. XAMPP Installers and Downloads for Apache Friends. *About the XAMPP project*. [En línea] [Citado el: 6 de agosto de 2021.]
26. *¡Bienvenido! - El proyecto del servidor HTTP apache*. [En línea] [Citado el: 7 de agosto de 2021.] <https://httpd.apache.org/>.
27. Visual FoxPro. Técnicas avanzadas. *Entendiendo la programación en 3 (o más) capas*. [En línea] [Citado el: 6 de agosto de 2021.]



<https://vfpavanzado.wordpress.com/2017/10/31/entendiendo-la-programacion-en-3-o-mas-capas/>.

28. Visual Studio Code. *Preguntas Frecuentes de Visual Studio Code*. [En línea] [Citado el: 7 de agosto de 2021.] <https://code.visualstudio.com/docs/supporting/faq>.

29. Bootstrap v5.1. *Grid System*. [En línea] [Citado el: 8 de agosto de 2021.] <https://getbootstrap.com/docs/5.1/layout/grid/>.

30. Bootstrap v5.1. *Carousel*. [En línea] [Citado el: 8 de agosto de 2021.] <https://getbootstrap.com/docs/5.1/components/carousel/>.

31. *Statcounter Global Stats*. [En línea] [Citado el: 9 de agosto de 2021.] <https://gs.statcounter.com/>.

• **Trabajos consultados:**

1. GARCÍA MOCHALES, Sonia. *Plataforma web para intercambio de material educativo*. Trabajo de Fin de Grado, Universitat Politècnica de València, 2018.
2. NOVELLA ESCRIBANO, M^a Dolores. *Desarrollo de una página Web para una empresa de restauraciones en pintura*. Trabajo de Fin de Grado, Universitat Politècnica de València, 2017.
3. SAIZ GARCÍA, Francisco Javier. *Portal web dedicado a la difusión de la figura de Luis Sánchez Polack "TIP"*. Proyecto Final de Carrera, Universitat Politècnica de València, 2016.