

WIKICINE



MEMORIA DEL PROYECTO

Wikicine

C.I.F.P. "JUAN DE COLONIA"

Departamento de Informática

2º DAW

Autor Luis G. Contreras Contreras

Tutor Lara manjon

Fecha 10/5/2025

Contenido

Contenido	2
Índice de figuras	3
Índice de tablas	5
1. Introducción.....	6
1.1. Descripción.....	6
1.2. Justificación.....	6
2. Planificación	7
2.1. Requisitos	7
2.1.1. Requisitos funcionales.....	7
2.1.2. Requisitos no funcionales.....	8
2.2. Recursos	9
2.2.1. Recursos hardware.....	9
2.2.2. Recursos software.....	10
2.3. Planificación temporal.....	12
2.4. Planificación económica.....	14
3. Tecnologías.....	15
4. Desarrollo y secuenciación temporal.....	16
4.1. Diseño.....	16
Diagrama de aplicación	16
Diagrama E/R (entidad/relación) y esquema de BBDD.....	23
Diseño de interfaces.....	25
4.2. Pruebas	27
5. Conclusiones finales	31
5.1. Conclusiones y grado de cumplimiento de los requisitos:	31
5.2. Propuestas de mejora o ampliaciones futuras.....	34
6. Guías.....	35
7. Referencias	49

Índice de figuras

Fig. 1-1-1-1- Pantalla inicial de la aplicación.....	6
Fig. 2-1-2-1- Alertas (Añadir película) que indican como rellenar correctamente el formulario.....	8
Fig. 2-1-2-2 - Alertas (Login) que indican como rellenar correctamente el formulario.....	9
Fig. 2-2-2-1- Visual Studio Code.....	10
Fig. 2-2-2-2- W.A.M.P.	10
Fig. 2-2-2-3- DbSchema.....	10
Fig. 2-2-2-4- PhpMyAdmin.....	11
Fig. 2-2-2-5- Draw.io.....	11
Fig. 2-2-2-6- Adobe Illustrator.....	11
Fig. 2-2-2-7- Google.....	11
Fig. 2-2-2-8- Hostinger.....	12
Fig. 2-3-1- Diagrama de Gantt.....	13
Fig. 4-1-1- Diagrama MVC genérico.....	16
Fig. 4-1-2- Archivos de Modelo en árbol de directorios local.....	17
Fig. 4-1-3- Archivos de Vista en árbol de directorios local.....	18
Fig. 4-1-4- Archivos de Controlador en árbol de directorios local.....	19
Fig. 4-1-5- Diagrama de secuencia básico.....	21
Fig. 4-1-6- Diagrama del sitio web.....	22
Fig. 4-1-7- Diagrama Entidad-Relación.....	23
Fig. 4-1-8- Diagrama Relacional.....	24
Fig. 4-1-9- Paleta de colores.....	25
Fig. 6-1- Pagina de portada o index.....	35
Fig. 6-2- Barra del menú versión Pc.....	36
Fig. 6-3- Barra del menú versión móvil con menú contraído.....	36

Fig. 6-4- Barra del menú versión móvil con menú desplegado.....	36
Fig. 6-5- Sección Acción de Página principal.....	37
Fig. 6-6- Sección Terror de Página principal.....	37
Fig. 6-7- Página de película seleccionada (Terminator) parte superior.....	38
Fig. 6-8- Página de película seleccionada (Terminator) parte inferior.....	38
Fig. 6-9- Botón de login en menú principal.....	39
Fig. 6-10- Ventana modal de login.....	39
Fig. 6-11- Barra de menú tras login.....	39
Fig. 6-12- Pagina de película seleccionada tras login.....	40
Fig. 6-13- Botón de puntuar.....	40
Fig. 6-14- Ventana modal de voto antes de seleccionar puntuación.....	41
Fig. 6-15- Ventana modal de voto después de seleccionar puntuación.....	41
Fig. 6-16- Ventana modal de voto emitido exitosamente.....	41
Fig. 6-17- Botón de añadir película en barra de menú.....	42
Fig. 6-18- Pagina de formulario para añadir película.....	42
Fig. 6-19- Ventana modal de error al añadir película.....	43
Fig. 6-20- Botones para añadir nuevos campos de actor y guionista.....	43
Fig. 6-21- Formulario con nuevos campos añadidos de actor y guionista.....	44
Fig. 6-22- Botones para eliminar los nuevos campos actor y guionista.....	44
Fig. 6-23- Pagina de película añadida exitosamente.....	45
Fig. 6-24- Pagina de película añadida erróneamente.....	45
Fig. 6-25- Botón para crear nuevo usuario en la ventana modal de login.....	46
Fig. 6-26- Formulario para crear nuevo usuario.....	46
Fig. 6-27- Ventana modal de campo vacío.....	47
Fig. 6-28- Ventana modal de correo ya registrado.....	47
Fig. 6-29- Pagina de usuario registrado exitosamente.....	48
Fig. 6-30- Pagina de usuario creado erróneamente.....	48

Índice de tablas

Tabla 2-3-1: Planificación temporal del proyecto	12
Tabla 2-4-2: Presupuesto del proyecto	15

1.Introducción

1.1. Descripción

Este proyecto plantea un sitio web que almacena y organiza información sobre películas de distintos géneros. Desde el trailer, director, poster etc, etc, etc hasta el reparto de actores.

Se usarán herramientas de desarrollo web como HTML[1], PHP[3] y SQL[5] principalmente con algunos complementos para front-end como [6]Bootstrap, JavaScript[4] y [7]SweetAlert2.

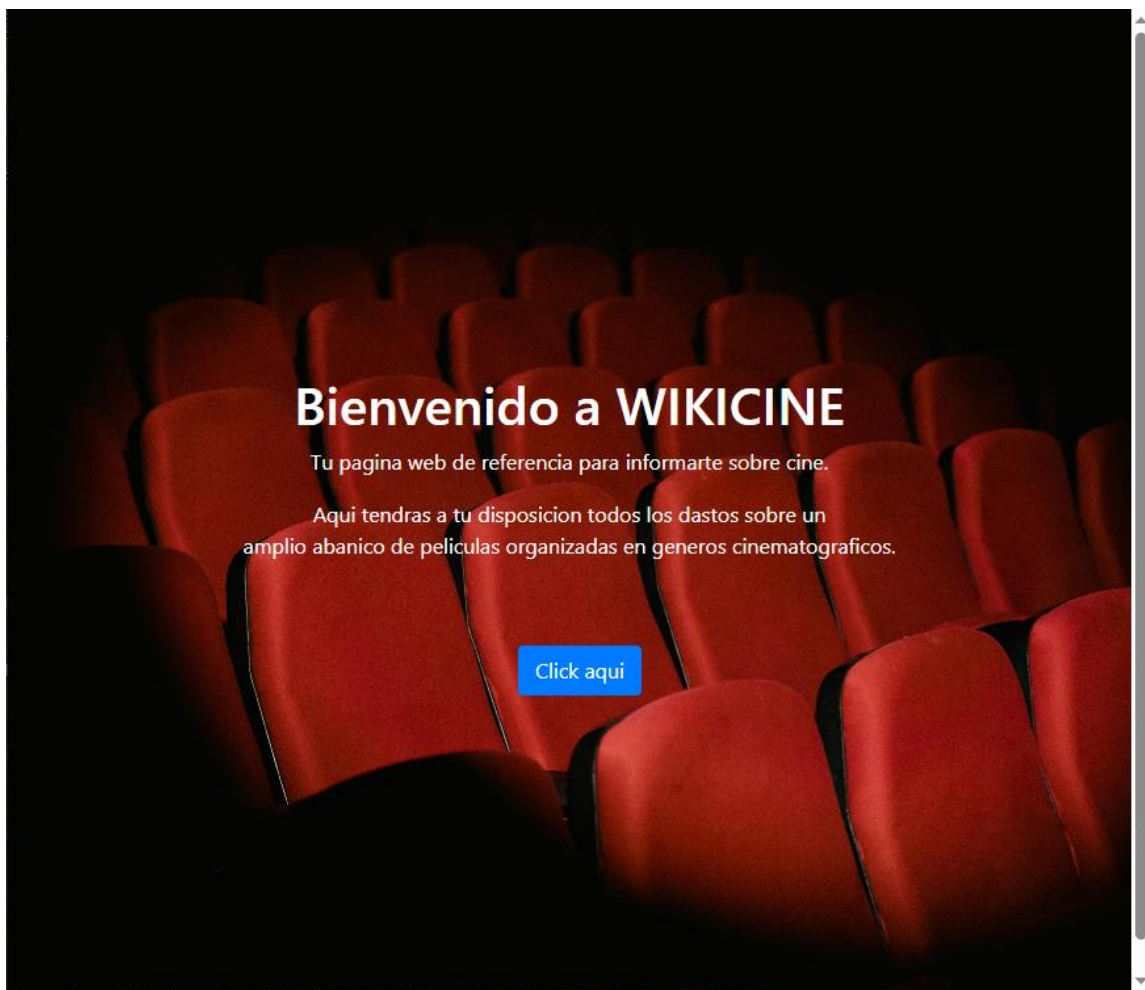


Fig. 1-1-1-1- Pantalla inicial de la aplicación

1.2. Justificación

Elegí esta temática (cine y películas) por que la conozco bien y entiendo como integrarla en diversos elementos que estudie en clase.

2. Planificación

2.1. Requisitos

2.1.1. Requisitos funcionales

✓ **Registro y login de usuarios:**

Se accede a un formulario de registro con tres campos: nombre de usuario, contraseña y correo electrónico. La información ingresada se almacena en una base de datos para su posterior consulta.

También se accede a un formulario de inicio de sesión con dos campos (nombre de usuario y contraseña), que valida los datos ingresados mediante una consulta a la base de datos.

✓ **Pastillas de película:**

La página principal se divide en secciones por género cinematográfico. Cada sección contiene una serie de pastillas, cada una correspondiente a una película almacenada en la base de datos y asociada a ese género. Cada pastilla incluye una imagen del póster, el título y algunos datos básicos de la película.

Al seleccionar una de las pastillas se accederá a una página con información extendida sobre dicha película.

✓ **Página con de película seleccionada:**

Seleccionar una pastilla se dirige a una página plantilla que se rellena procedualmente con la información detallada de la película seleccionada.

✓ **Sistema de puntuación:**

Los usuarios registrados pueden puntuar películas de 1 a 10. Esta puntuación se muestra en la página de la película seleccionada.

✓ **Sesiones:**

Existen 2 tipos de sesiones, una para los usuarios “No-registrado” y otra para los “Registrados”. Cada tipo de sesión ofrece distintas funcionalidades.

Los no registrados pueden navegar libremente por la página, pero solo una vez registrados pueden puntuar o introducir nuevas películas.

2.1.2. Requisitos no funcionales

El sitio web está diseñado para que la navegación resulte intuitiva y no requiera explicaciones al usuario. Ante posibles errores de este, se muestran avisos específicos según el caso: por ejemplo, al intentar registrar un título ya existente o al subir una imagen con un formato no admitido.

El tiempo de espera para la carga de recursos sigue el valor por defecto, con un máximo de 60 segundos. No se modifican los parámetros `max_execution_time` en PHP[3] ni `setTimeout` en JavaScript[4].

Las ventanas emergentes no inhabilitan nada. Puedes pulsar en el ok o pulsar fuera, esta se cierra automáticamente. En algunos casos estas incluso se cierran automáticamente tras 5 segundos, como por ejemplo a la hora de puntuar exitosamente.

Funciona en los navegadores Chrome, Edge y Firefox. Esta en idioma español.

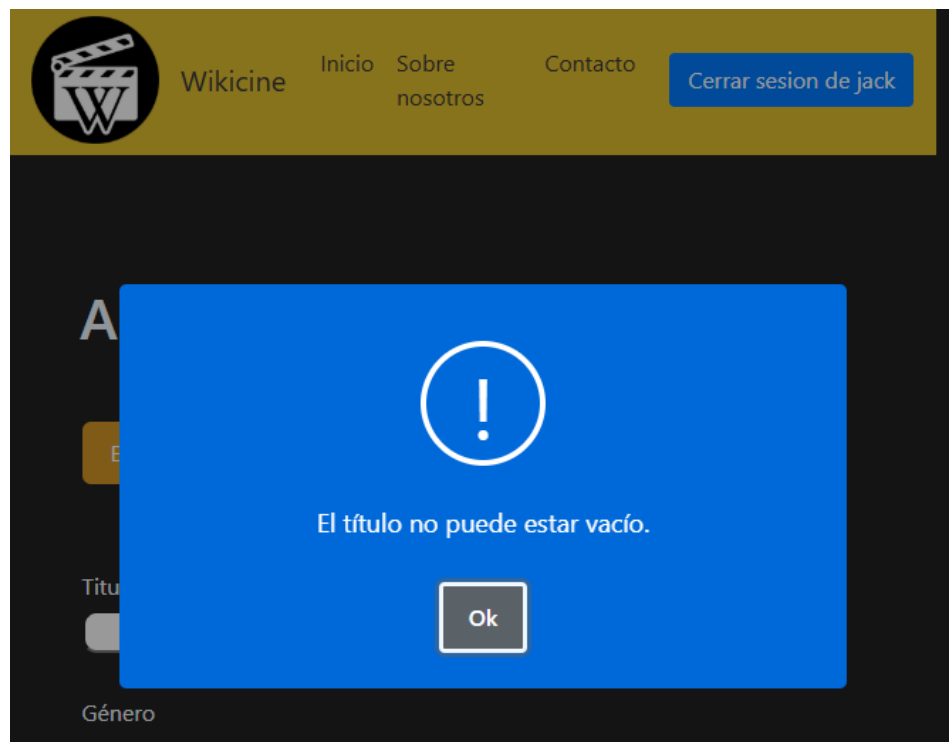


Fig. 2-1-2-1- Alertas (Añadir película) que indican como rellenar correctamente el formulario

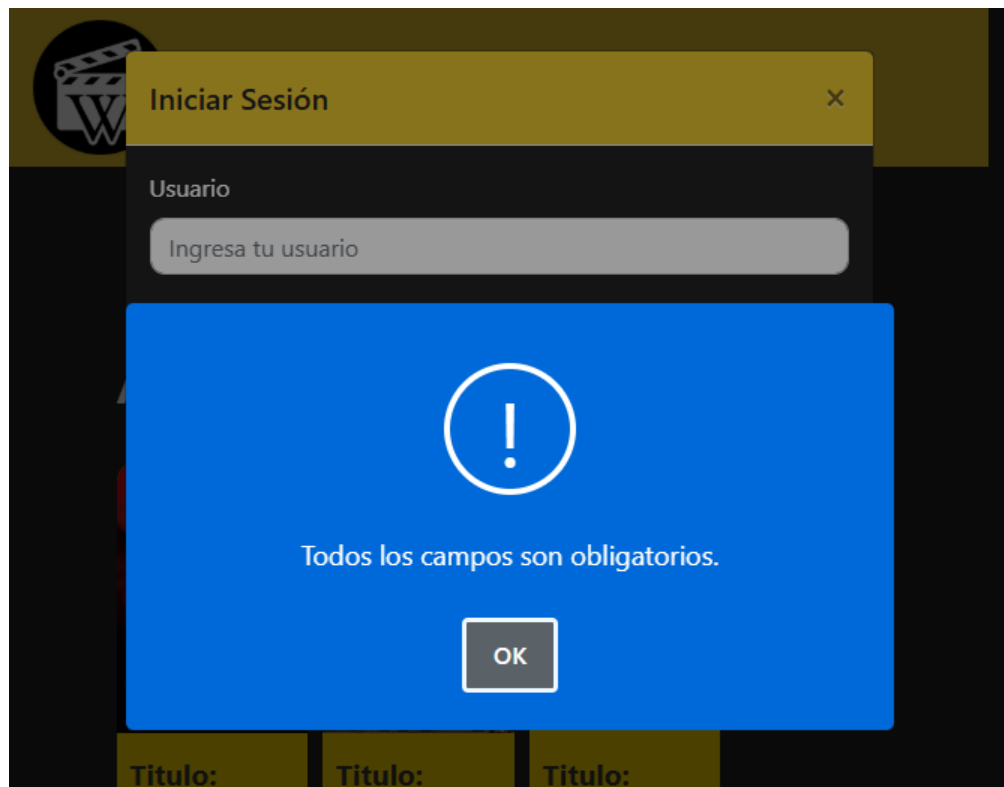


Fig. 2-1-2-2 - Alertas (Login) que indican como rellenar correctamente el formulario

2.2. Recursos

2.2.1. Recursos hardware

Lista de recursos hardware

- ✓ PC Windows
- ✓ AMD Ryzen 73700X 8-Core Processor
- ✓ Ram 16 GB
- ✓ Sistema Operativo de 64bits

2.2.2. Recursos software

Lista de recursos software:

- ✓ **Visual Studio Code:** Para generar código de todo tipo.



Fig. 2-2-2-1-Visual Studio Code

- ✓ **W.A.M.P.:** En la etapa inicial, antes de poder desplegarlo en un servidor online lo use para generar un servidor local en el que probar el código php[3] de mi web.



Fig. 2-2-2-2-W.A.M.P.

- ✓ **DbSchema:** Es un software para representar diagramas relacionales. Inicialmente el diagrama relacional lo hice con él.



Fig. 2-2-2-3- DbSchema

- ✓ **PhpMyAdmin:** Principalmente para gestionar la base de datos, pero también hacer el diagrama relacional.



Fig. 2-2-2-4- PhpMyAdmin

- ✓ **Draw.io:** Para generar diagramas varios. Como el diagrama Entidad-Relación o el del sitio web.



Fig. 2-2-2-5- Draw.io

- ✓ **Illustrator:** Lo uso para generar diagramas y generar imágenes. Como por ejemplo la paleta de colores y el diagrama de Gantt.



Fig. 2-2-2-6- Adobe Illustrator

- ✓ **Google:** Búsqueda de información primaria



Fig. 2-2-2-7- Google

- ✓ **Hostinger:** Es un servicio de hosting y dominio, entre otras varias funcionalidades. En la etapa final de desarrollo del proyecto, una vez desplegada la página web, el código lo genero y modifíco desde el editor de código integrado.



Fig. 2-2-2-8- Hostinger

2.3. Planificación temporal

Empecé el proyecto el 1 de Octubre. Lo terminé el 10 de Abril. Dado que la entrega final al tutor del 05-09 de Mayo(05-09 del 5).

Tabla 2-3-1: Planificación temporal del proyecto

Descripción de la tarea	Nº de horas
Análisis y búsqueda de ideas	2-3 Oct / 3h
Planteamiento de requisitos de la aplicación	3-4 Oct / 6h
Diseño de Base de datos	5-9 Oct / 15h
Desarrollo de principal-page y páginas del sitio (front-end y estructura iniciales)	10 Oct / 3h
Desarrollo de modelo y métodos	10 Oct - 10 Nov / 90h
Desarrollo Controladores	10-14 Nov / 15h
Pruebas unitarias y de integración	14-16 Nov / 6h
Desarrollo de mejoras técnicas y estéticas	16 Nov-17 Dic / 90h
Retoques y ajustes finales Front-end y Back-end	17 Dic-17 Mar / 90h
	318h

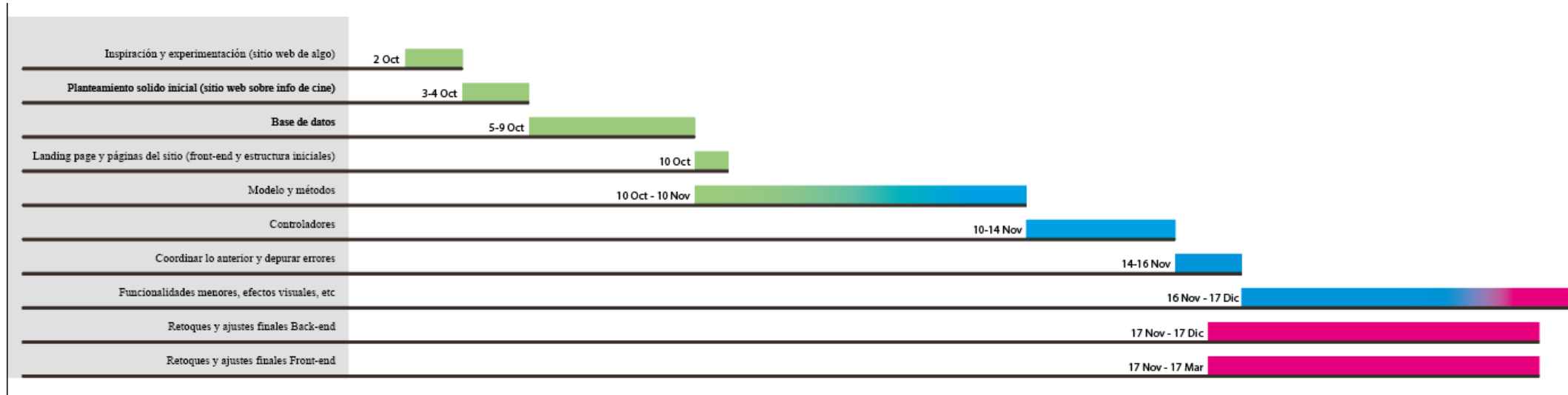


Fig. 2-3-1- Diagrama de Gantt

2.4. Planificación económica

El gasto de luz son **60€** mensuales.

El proyecto A durado de **Octubre a Marzo** son 6 meses
6 meses de desarrollo: **360€**

El hosting son **10€** mensuales.

3 meses desde que lo desplegué en el servidor online: **30€**

“El salario programador web promedio en España es de € 25.000 al año

o € 12,82 por hora.

*Los cargos de **nivel inicial** comienzan con un ingreso de € 21.000 al año, mientras que profesionales más **experimentados** perciben hasta € 34.000 al año.”*

[8]-*Talen.com*

*“El **Salario junior** es de 22.000 euros brutos anuales”*

[9]-*Ceac.es*

Obtener el salario por horas según [10]-*global66.com*:

Dado que yo pertenezco al nivel inicial cobro 21.000€ anuales.

Divide el salario anual por el número de semanas al año:

$21.000 \div 52 = 403,8€$ por semana.

Divide el resultado semanal por el número de horas trabajadas a la semana:

$403,8 \div 40 = 10€$ por hora.

Este presupuesto se basa en que cobro **10€ por hora**.

Tabla 2-4-2: Presupuesto del proyecto

Descripción de la tarea	Precio (€)
Análisis y búsqueda de ideas	30
Planteamiento de requisitos de la aplicación	60
Diseño de Base de datos	150
Desarrollo de principal-page y páginas del sitio (front-end y estructura iniciales)	30
Desarrollo de modelo y métodos	900
Desarrollo Controladores	150
Pruebas unitarias y de integración	60
Desarrollo de mejoras técnicas y estéticas	900
Retoques y ajustes finales Back-end y Back-end	900
Luz	360
Hosting	30
TOTAL (€)	3.540

3. Tecnologías

- ✓ **HTML[1]:** Para crear la landing page, el formulario de registro, páginas de que notifican el resultado del formulario, así como la plantilla para crear las pastillas y paginas concretas de cada película.
- ✓ **CSS[2]:** Para dar estilos a las páginas creadas en HTML[1].
- ✓ **PHP[3]:** Para la modelo y controlador. También para renderizar algunos de los archivos de la vista.
- ✓ **JAVASCRIPT[4]:** Para la vista. Aporta funcionalidades de validación y efectos visuales (algunas bastante cruciales pese a ser front-end).
- ✓ **SQL[5]:** Para crear las tablas de la base de datos.
- ✓ **BOOTSTRAP[6]:** Como complemento del estilo de las páginas. Principalmente la organización de contendores en columnas, el espaciado y parte del comportamiento responsive.
- ✓ **SWEETALERT2[7]:** Para ventanas modales y toast.

4. Desarrollo y secuenciación temporal

4.1. Diseño

Diagrama de aplicación

El sitio web sigue un patrón **MVC: *Modelo Vista Controlador***.

Diagrama MVC genérico: El patrón Modelo-Vista-Controlador se resume así. La vista presenta la aplicación al usuario y le permite interactuar con ella de forma simplificada. Los métodos gestionan la interacción con la base de datos, el acceso a la información almacenada en ella y demás funcionalidades necesarias para el correcto funcionamiento de la aplicación. El controlador gestiona el flujo de ejecución de los distintos métodos, operaciones y resto de código.

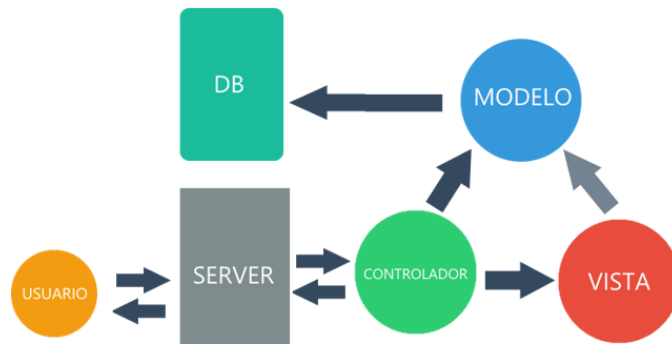


Fig. 4-1-1- Diagrama MVC genérico

Esta es la distribución por carpetas del Wikicine. Aquí se puede observar los archivos dentro de cada carpeta. Los controladores y los archivos necesarios para estos, lo mismo para el modelo y la vista.

Aparte de las carpetas del patrón MVC, tengo carpetas auxiliares, como **Css[2]**, que contiene todos los estilos. **Img** contiene todos los recursos gráficos.

Index es el único archivo que está fuera y actúa como landing page y tiene incluida mediante include la página principal.

La carpeta de **Modelo** contiene los archivos para acceder a la base de datos.

El archivo **Conexion.php** contiene únicamente un método estático para obtener la instancia de la conexión a la base de datos y lo tengo como archivo separado para incluirlo con `include` fácilmente en cualquier otro archivo para simplificar el proceso. El archivo de **Modelo_DatosPeliculas.php** contiene métodos para acceder a los datos de las tablas referentes a las películas como **películas, géneros, guionistas, directores, actores, directores**, etc.

El archivo de **Usuario.php** contiene métodos para acceder a los datos de las tablas referentes a los usuarios.

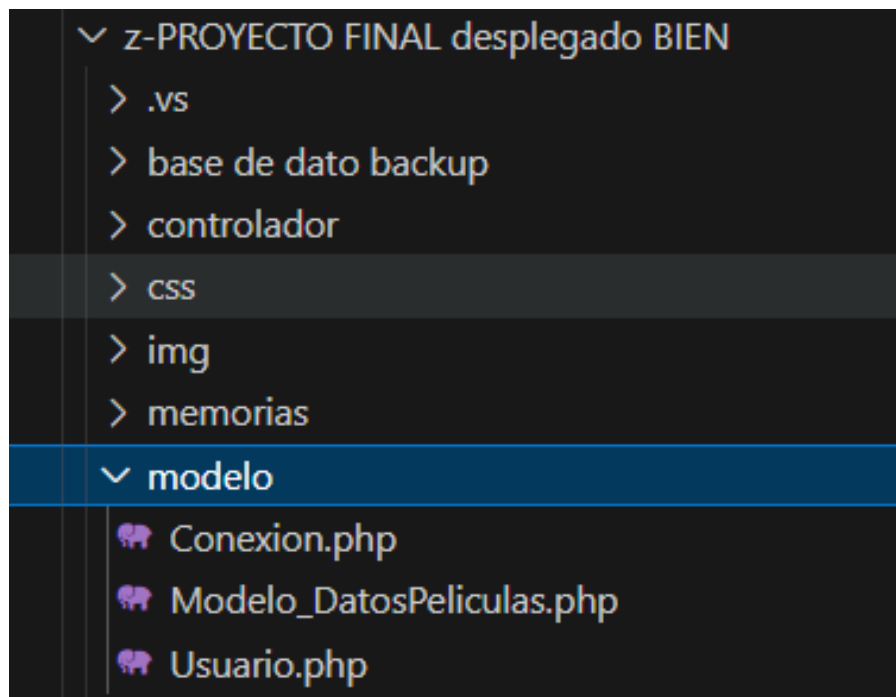


Fig. 4-1-2- Archivos de Modelo en árbol de directorios local

La carpeta **Vista** contiene los archivos para gestionar el front-end. Esto abarca desde páginas enteras como **formulario_añadir_pelicula.php** (a pensar de la extensión funcionan como un html[1]), scripts de javascript[4] que añaden funcionalidades de validación y de efectos visuales como **subir_archivo_dragable.js** y **validar_antes_de_fetch.js**, hasta bloques comunes a varias paginas como **header.php** o **principal_page.php** que se incluyen con include.

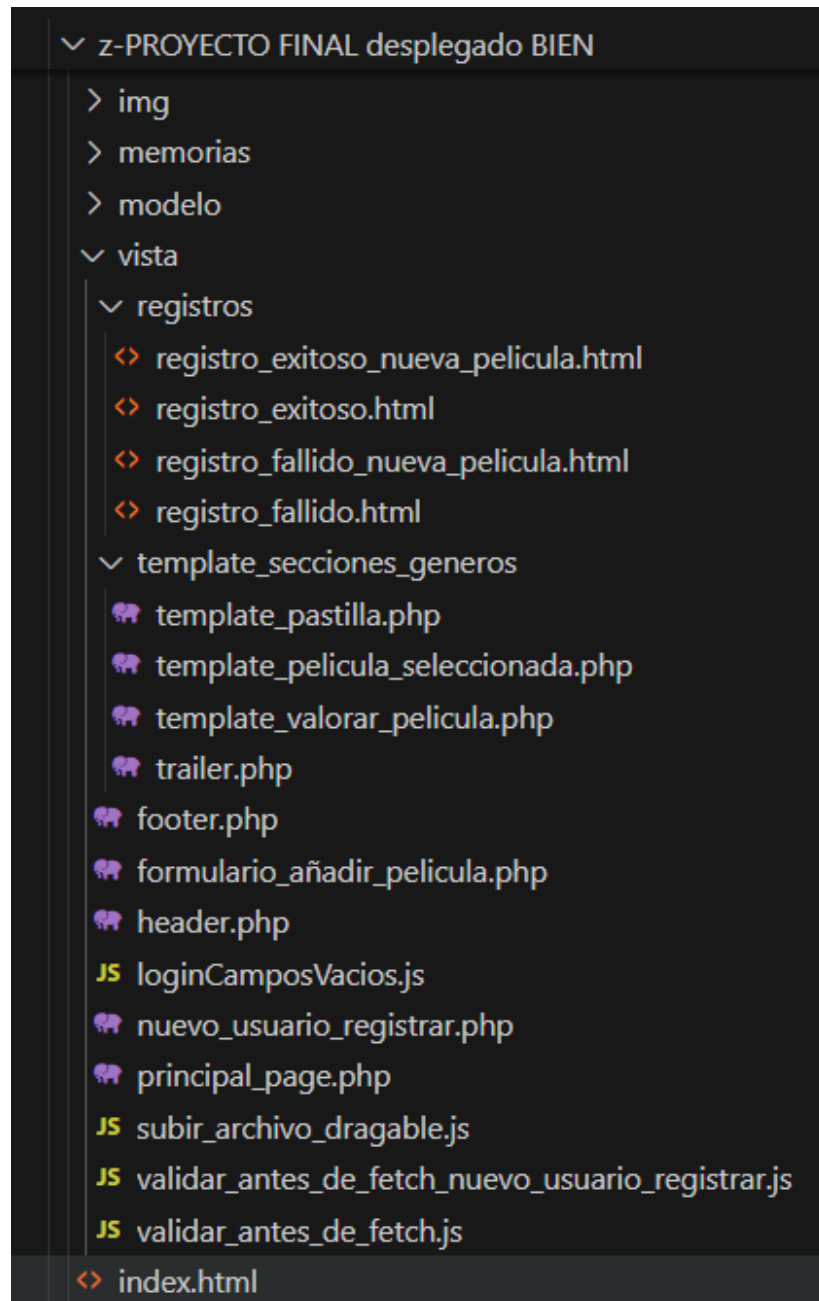


Fig. 4-1-3- Archivos de Vista en árbol de directorios local

También contiene las carpetas **registros** y de **template_secciones_generos**.

Registros contiene paginas a las que redirigir cuando el resultado es **fallido** o **exitoso** tras rellenar un formulario de registro de usuario o al agregar una nueva película.

Template_secciones_generos son paginas-plantilla. Combinan una estructura básica de html[1] pero gracias a que también contienen código php[3] y a un archivo del controlador que se ejecuta previamente, se rellenan de contenido sacado de la base de datos. Así un solo archivo genera multitud de instancias distintas. Abarca los archivos **template_pastilla.php**, **template_pelicula_seleccionada.php**, **template_valorar_pelicula.php** y **trailer.php**.

En la carpeta **Controlador** tenemos archivos que gestionan el flujo de ejecución de operaciones y llamadas a clases y a sus métodos correspondientes.

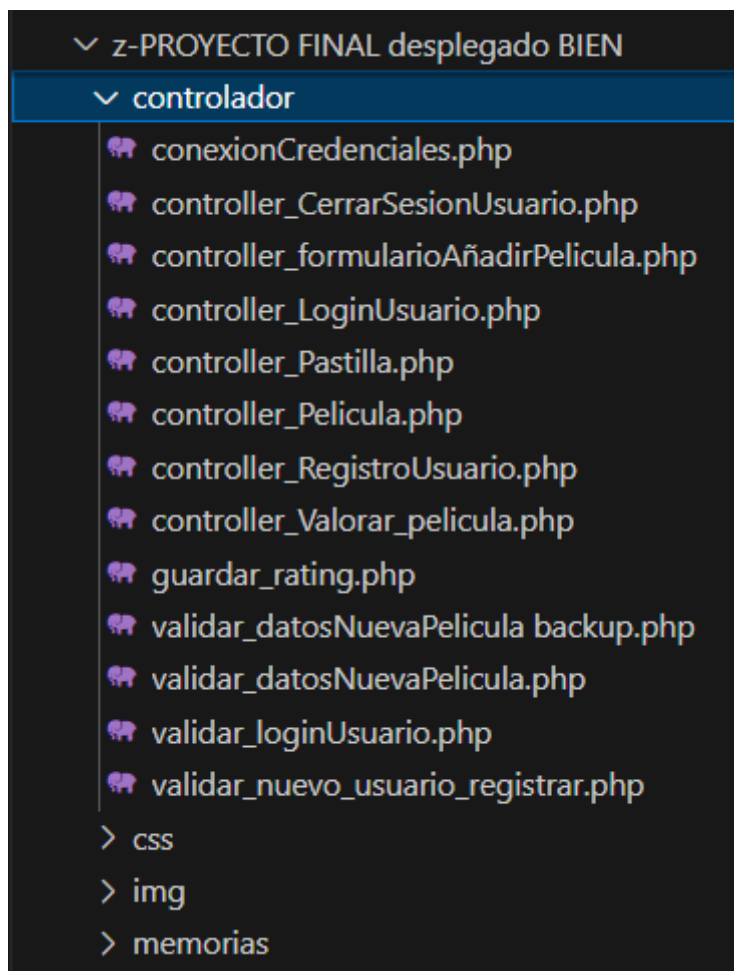


Fig. 4-1-4- Archivos de Controlador en árbol de directorios local

Algunos archivos sirven para generar las paginas del sitio que va a poder visitar en algún momento, como **Controller_Pastilla.php** y otros archivos sirven para posibilitar funcionalidades, como **Controller_Login_Usuario.php** o **guardar_rating.php**.

Aquí comento algunos controladores cuya función es menos obvia.

ConexionCredenciales.php es un archivo que contiene variables inicializadas con las credenciales para la conexión a la base de datos e incluso una instancia de la clase usuario.

Este archivo se incluye con incluye en distintos archivos para simplificar el proceso, que se repetirá en numerosas ocasiones.

Controller_Pastilla.php genera cada una de las pastillas en la página principal, con la información básica de cada película.

Controller_LoginUsuario.php gestiona el login accediendo a la base de datos y comparando los datos de entrada con los almacenados en la base de datos.

Controller_Pelicula.php gestiona que se la página de **película_seleccionada.php**, se rellene correctamente para crear la instancias de la página de la película seleccionada para rellenarla con los datos de la película correspondiente en cada caso.

Los controladores tienes nombres auto explicativos. Los que contienen las palabras “controller_formulario” enviar la información de los formularios para ingresarla en la base de datos o para ejecutar la funcionalidad correspondiente.

El controlador **controller_Valorar_pelicula.php** gestiona el funcionamiento tanto en términos de front-end como de back-end de la ventana modal mediante la cual se pueden valorar de 1 a 10 las películas.

El controlador “**guardar_rating.php**” guarda en la base de datos la puntuación que un usuario a asignado a una película.

Los que contienen la palabra “validar”, se encargan de validar la información antes de enviar dicha información.

Diagramas de secuencia: Esto es un diagrama de una de las numerosas secuencias, no de toda la aplicación. Representa el flujo de secuencia de la página principal (Una de las más importantes) y como genera las pastillas de cada película almacenada en la base de datos y ordenadas por el genero al que pertenecen.

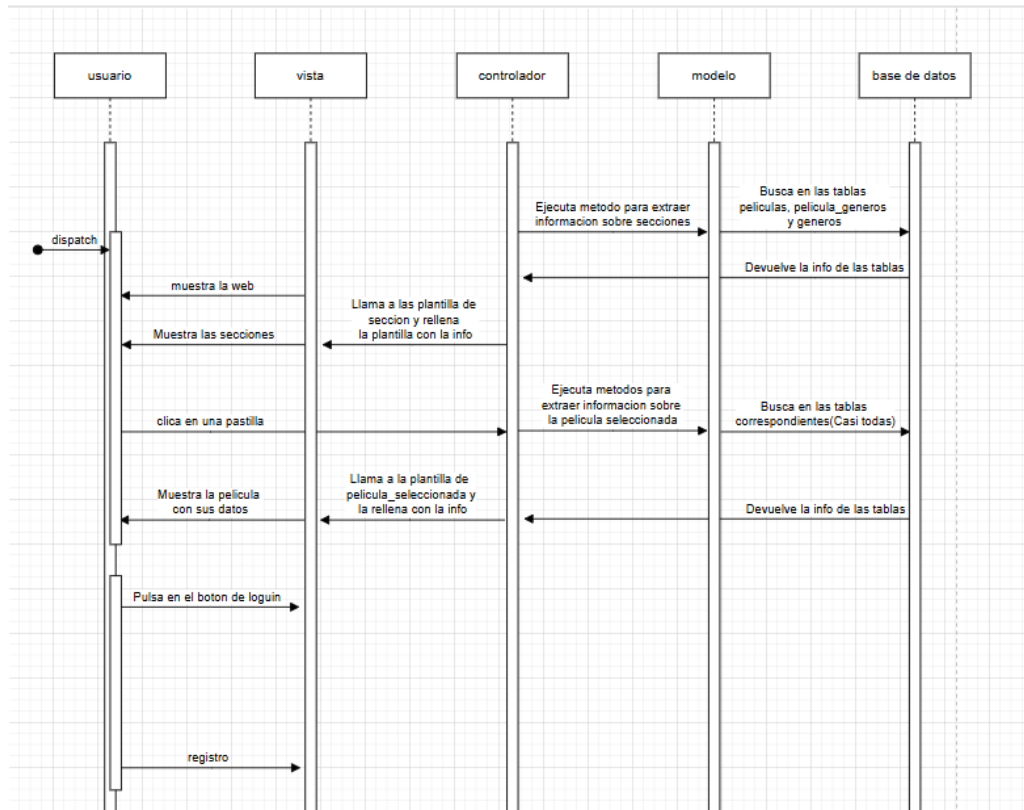


Fig. 4-1-5- Diagrama de secuencia básico

Diagrama E/R (entidad/relación) y esquema de BBDD

La base de datos almacena toda la información referente a películas, usuarios y a las puntuaciones de las películas asignadas por los usuarios.

Hay 2 tipos de tablas, las tablas principales como “actores” y las tablas intermedias como “película_actor”.

En el esquema las segundas no se representan cuando la relación es N:M. A no ser que se indique desde la relación. Como por ejemplo “película_género” en el caso de las tablas películas y género por ser de especial importancia.

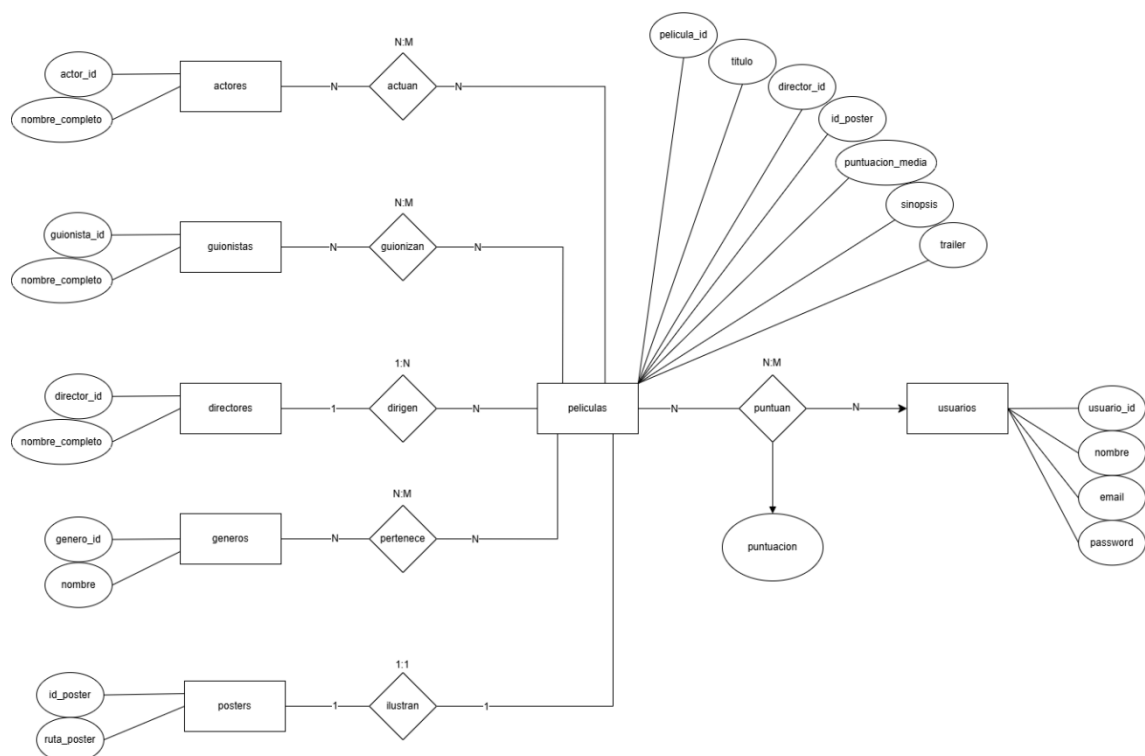


Fig. 4-1-7- Diagrama Entidad-Relación

Este es el esquema de base de datos. Aquí se muestra la relación de las tablas entre si principalmente a través de sus claves primarias y foráneas.

La tabla películas además de ser la que más información aglutina, es la más importante pues alrededor de ella orbitan el resto directa o indirectamente.

Como las tablas directores, posters y tablas intermedias como película_género, película_actor, película_guionista, película_puntuacion_usuarios y de las tablas géneros, actores, actores y usuarios respectivamente.

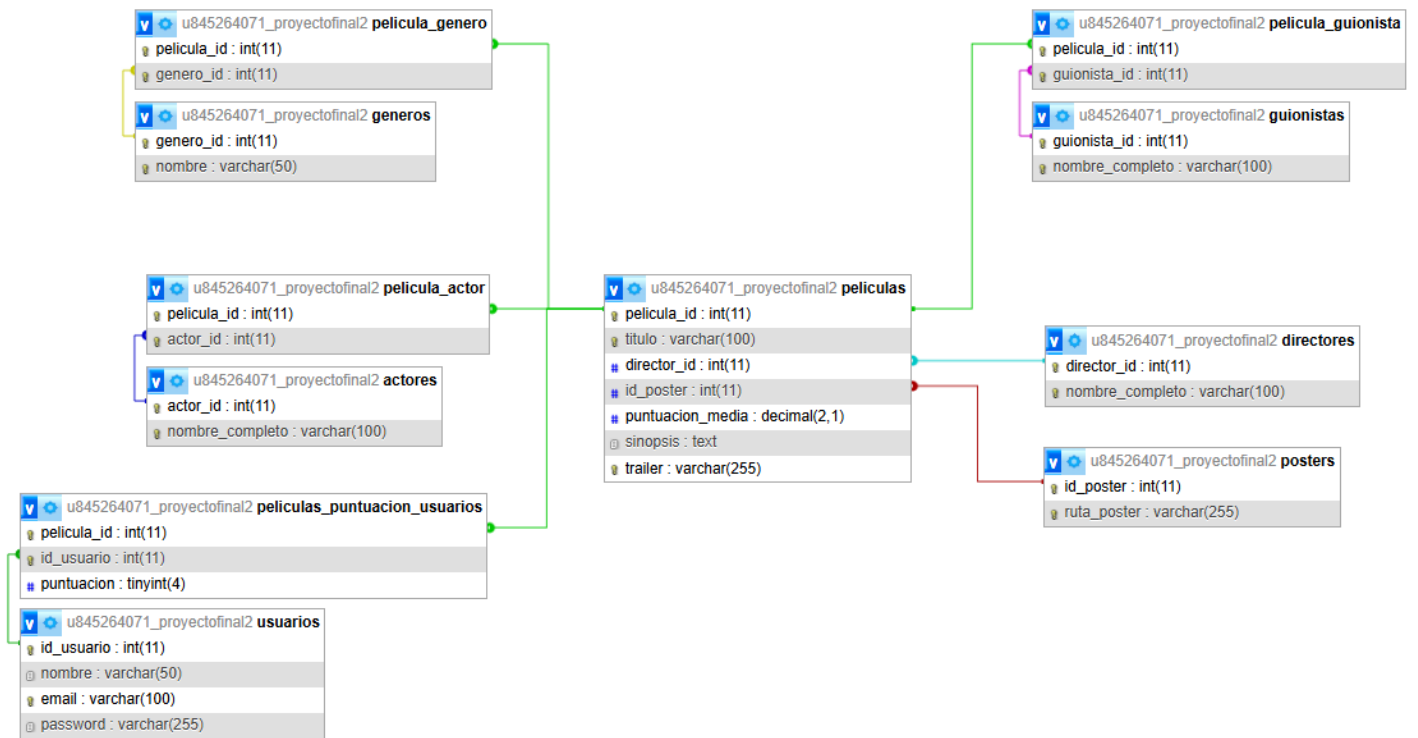


Fig. 4-1-8- Diagrama Relacional

Diseño de interfaces

Elegí un esquema de colores basados en amarillos, grises y azules.

En total 9 colores:

Blanco puro, blanco aliceblue, 4 “dorados”, gris oscuro, gris claro, verde y azul.
Algunos están asociados a muy variados aspectos y otros a elementos muy concretos.



Fig. 4-1-9- Paleta de colores

Como se observa uso variedades de dorados y de blancos para crear jerarquía visual, las combinaciones no se sienten tan repetitivas. En algunos casos como cuando uso “blanco aliceblue” para algunos textos, rebaja el contraste y hace que la sensación al ojo sea menos acida y mas agradable. También me permite crear jerarquía visual sutilmente en el propio texto.

Los más importantes son los dorados (sobre todo el #ffd700) y el gris oscuro. Para generar mediante el color, 2 categorías de importancia muy diferenciadas entre sí: Secciones y elementos importantes en dorados y gris oscuro para lo demás, el fondo mayormente.

El amarillo también funciona para llamar la atención sobre algo seleccionado.

Tuve en cuenta las 3 propiedades del color (tono, saturación e iluminación) para que los dos colores principales funcionaran aun así fuera armónico a la vista. Aunque estos 2 son colores con tonos muy distintos entre sí genera armonía porque tienen una saturación similar e iluminación en similar proporción.

Los colores blancos, gris claro y azul son auxiliares. Su presencia se justifica primero por que todos en general aportan variedad y cada uno tiene propiedades concretas.

Su principal aporte es que solucionan los problemas de jerarquía visual cuando la combinación “dorado-gris oscuro” son repetitivas o confusas. Por ejemplo un botón de color gris oscuro en un fondo dorado, podría dar la sensación de que es un hueco en vez de un botón.

Como aportes muy concretos para casos específicos.

El blanco ayuda a que el diseño respire un poco y no se sienta tan claustrofóbico con colores tan intensos como la dualidad principal. Además genera un contraste casi universal con todo tipo de fondos. Sutil en fondo dorado, intenso en fondo gris oscuro.

El azul. Como el dorado es el color elegido para llamar la atención, en el fondo dorado esto es un problema. Así que es perfecto para destacar un elemento en el fondo dorado, como un botón importante.

El gris claro. De manera similar al anterior, pero para destacar en el gris oscuro sin adoptar demasiado protagonismo como para opacar al azul. Útil para botones secundarios.

En cualquier caso, no me adiero a rajatabla a este pequeño “set de reglas”, si no que lo uso como unas directrices generales para conseguir cierta sorpresa y frescura en el diseño.

4.2. Pruebas

He realizado varios tipos de pruebas a medida que iba desarrollando el proyecto y una vez como puesta a punto final.

Pruebas funcionales:

Pruebas del formulario de logueo:

- Comprobar que la ventana modal con el formulario de login se despliega correctamente.
- Comprobar que el formulario login lanza los avisos de SweetAlert2[7] al dejar los campos vacíos. Ya fuera el nombre de usuario o una contraseña o ambos.
- Comprobar que el formulario login lanza el aviso de SweetAlert2[7] al introducir un nombre de usuario o una contraseña que no están registrado en la base de datos.
- Comprobar que el formulario lanza el aviso de login exitoso al introducir las credenciales correctamente.

Pruebas de Logueo:

- Comprobar que se inicia sesión correctamente al introducir las credenciales de un usuario ya registrado.

Hay cambios en el front que lo indican, como un mensaje con el nombre usuario en la página principal y los botones que se muestran en la barra de menú serán distintos. El botón de login será sustituido por los botones de añadir nueva película y el de cerrar sesión.

- Comprobar que se cierra sesión al pulsar el botón de cerrar sesión.

Pruebas del formulario de registro de nuevo usuario:

- Comprobar que el formulario registro de usuario lanza los avisos de SweetAlert2[7] al dejar algún campo vacío.
- Comprobar que el formulario registro de usuario lanza el aviso correspondiente de SweetAlert2[7] al introducir un email que ya está registrado en la base de datos.

Pruebas de registro de nuevo usuario:

- Comprobar que el nombre de usuario y la contraseña y el email se registran correctamente en la base de datos, la información introducida a través de los campos del formulario.
- Comprobar que el nombre de usuario y la contraseña y el email se registran correctamente en la base de datos, la información introducida a través de los métodos del modelo en el controlador.

Pruebas del formulario para añadir nueva película:

- Comprobar que el formulario registro lanza los avisos correspondientes de SweetAlert2[7] al dejar algún campo vacío.
- Comprobar que el formulario registro lanza el aviso correspondiente de SweetAlert2[7] al introducir un título o un trailer que ya está registrado en la base de datos.
- Comprobar que el formulario registro lanza el aviso correspondiente de SweetAlert2[7] al introducir en un campo el nombre de un actor que ya esté presente en otro campo del bloque de actores.
- Comprobar que el formulario registro lanza el aviso correspondiente de SweetAlert2[7] al introducir en un campo el nombre de un guionista que ya esté presente en otro campo del bloque de guionistas.
- Comprobar que el formulario registro lanza el aviso correspondiente de SweetAlert2[7] al intentar subir una imagen con un formato inadecuado para el poster de la película que ya está registrado en la base de datos.

Pruebas sobre puntuar una película seleccionada:

- Comprobar que se despliega la ventana modal con el formulario para puntuar al pulsar el botón de puntuar.
- Comprobar que el front end del formulario reacciona correctamente al interactuar con él. Por ejemplo, que el botón de enviar cambia de gris a dorado cuando seleccionas una puntuación. Que tras pulsar una de las estrellas, todas las estrellas a la izquierda de suya cambian de blanco a dorado o la interrogación cambia por el valor de puntuación seleccionado al pulsar una estrella correspondiente.

-Comprobar que la puntuación de la película se actualiza correctamente en el front-end de la página de la película seleccionada.

Pruebas del formulario de puntuar película seleccionada:

-Comprobar que salta el aviso de SweetAlert2[7] al pulsar el botón de enviar cuando se enviado correctamente tu puntuación.

-Comprobar que salta el aviso de SweetAlert2[7] al pulsar el botón de enviar no has seleccionado ninguna puntuación.

-Comprobar que la nueva puntuación se ha registrado correctamente en la base de datos en la tabla correspondiente a través del formulario de puntuación.

-Comprobar que la nueva puntuación se ha registrado correctamente en la base de datos en la tabla correspondiente a través del método correspondiente.

Pruebas sobre añadir una nueva película:

-Comprobar que se ha registrado correctamente en la base de datos, la información introducida a través de los campos del formulario.

-Comprobar que se ha registrado correctamente en la base de datos, la información introducida a través de los métodos del modelo en el controlador.

-Comprobar que se ha generado correctamente la pastilla de la nueva película en la página principal.

-Comprobar que se ha generado correctamente la página de la película seleccionada de la nueva película registrada.

Pruebas del página principal:

-Comprobar que la página principal genera las pastillas con la información correspondiente de la base de datos. Como mínimo si hay un registro en la tabla Películas debería generarse la pastilla correspondiente, aunque sea con información incompleta.

Pruebas generales:

- Comprobar que el header se visualiza correctamente desde todas las paginas el sitio.
- Comprobar que el header y sus botones y enlaces funcionan correctamente desde todas las páginas del sitio.

Pruebas de base de datos y triggers:

- Comprobar que al insertarse una puntuación en peliculas_puntuacion_usuarios se registre también en películas.
- Comprobar que al actualizarse una puntuación en peliculas_puntuacion_usuarios se registre otra vez(actualizarse) también en películas.
- Comprobar que al eliminarse un registro en determinadas tablas principales, los triggers eliminan los registros en las tablas intermedias.
Por ejemplo al introducir un registro en peliculas, se debe eliminar también un registro en pelicula_actor o pelicula_guionista.

5. Conclusiones finales

5.1. Conclusiones y grado de cumplimiento de los requisitos:

El grado de cumplimiento es bastante satisfactorio.

Algunos aspectos podrían ser mejores de haber tenido una mejor perspectiva, una visión más amplia antes al planear el proyecto, así como otros aspectos necesitan de más tiempo para ser perfilados.

Este proyecto ha experimentado una gran evolución desde el planteamiento de la idea inicial hasta su finalización.

Entendí también que usar bibliotecas y tecnologías nuevas es una espada de doble filo. Pues por lo general están preparados para resolver un problema concreto de una forma muy específica y en cuanto te sales del rango de soluciones que ellos proveen, te vas a encontrar con errores muy desconcertantes.

Parte de los requisitos incluía experimentar con nuevas tecnologías, recurrí a una biblioteca llamada SweetAlert2[7].

Esta tecnología facilita enormemente hacer y personalizar ventanas modales(hice todas las ventanas modales con ella excepto la de login que la hice con Bootstrap[6]) pero no estaba preparada para usos más rebuscados. Como por ejemplo intentar colocar dentro un bloque para puntuar películas (que requiere JavaScript[4]) y fue un camino muy complicado y lleno de errores e incompatibilidades.

Comparación con los requisitos iniciales:

A continuación, abordo los requisitos iniciales y como han ido evolucionando a la hora de ejecutarlos.

Los requisitos que **Si** he cumplido están marcados con **V**.

Los requisitos que **No** he cumplido están marcados con **X**.

V-Pagina que mostraría una selección de películas: Lo organicé todo alrededor de los géneros cinematográficos y el método que se me ocurrió para estructurar el front-end es a través de pastillas con información resumida.

V-Tendría una página con información expandida: Lo ejecute con una página genérica, una plantilla que se rellenaría con los datos de la película seleccionada con el id de la película seleccionada enviado por get.

V-Tendría 2 sesiones: Una como visitante y otra como miembro registrado: Esto implicaba un formulario de login. Aunque es central al concepto del proyecto no pude implementarlo hasta completar los puntos anteriores. Generar una sesión distinta para cada tipo de usuario pensé que sería el punto más complicado, incluso cuando ni siquiera me planteaba las varias diferencias que implica la experiencia de sesiones de ambos tipos. Sin embargo, una vez implementado con relativa facilidad, de forma bastante orgánica fui entendiendo que distintas necesidades tenían cada uno y como tenía que acomodarlo todo. No fue tan complicado como laborioso, así como prestar atención a los pequeños detalles.

V-Los usuarios podrían incluir nuevas películas en la página: En este punto ocurrió justo lo contrario al anterior. Pensé que sería sencillo, pero fue el apartado más elaborado y complejo.

Validaciones de todo tipo, consultas a la base de datos, conflictos constantes entre varias funciones de JavaScript[4], interminables comprobaciones, limpieza constante de inserciones en la base de datos que estaban incompletas. Aprendí la utilidad de la palabra reservada “exit” en la línea adecuada y evité tener que limpiar muchas películas con información incompleta, aunque al final siempre tienes que.

No es hasta estas pruebas donde entendí la cantidad de funcionalidades y validaciones que me faltaban y tuve que volver a modificar controladores y el modelo.

V-Se podrán puntuar las películas: Este requisito enlaza con el núcleo de la idea del proyecto. La ejecución implica varios bloques. Requiere front-end, consultas a la base de datos, funciones de java script.

En primer lugar, muestra un botón que solo aparece cuando estas registrado, este despliega una ventana modal que muestras un formulario de puntuación, el formulario muestra el número de estrellas y el valor que representa cada una cambia dependiendo de cual pulses y tiene sus propias validaciones. Además, al enviar la puntuación a la base de datos actualiza automáticamente la puntuación en la página de la película seleccionada.

V-Tendrá una base de datos de más de 4 tablas relacionadas: El objetivo se cumple correctamente. Es su desarrollo aprendí que una base de datos es el pilar fundamental de este tipo de proyectos y si está bien planteada hace todo el proceso mucho más fluido.

Cometí el error de no plantearla correctamente desde un inicio, pues no tenía del todo claro que necesitaba almacenarlo ni el cómo necesitaba almacenarlo.

Por ejemplo uno de los varios errores de planteamiento que cometí fue no pensar en que cada película podría pertenecer a varios géneros. Cambiar eso cuando estaba todo casi terminado fue un bastante engorroso.

Volvía a la base de datos continuamente por que a medida que iba desarrollando el

proyecto en otros aspectos iba viendo carencias en la base de datos que tenía que ir solventando.

V-Estará desplegado en un servidor: Este requisito se cumple correctamente. No obstante, aprendí que no basta con hacer un copia y pega del código. Cada hosting tiene su propia idiosincrasia. En Hostinger por ejemplo la base de datos la creas desde su propio PhpMyAdmin y te obliga a nombrar la base de datos con un prefijo pre creado de forma automática por Hostinger, así que tuve que cambiar el nombre de todos los sitios donde estuviera.

V-Tendrá un diseño responsive: Este requisito se cumple correctamente.

V-Se registrá por el patrón modelo-vista-controlador: Este requisito se cumple correctamente.

V-Utilización de tecnologías no vistas en clase: Este requisito se cumple correctamente porque uso SweetAlert2[7].

V-Se podrán puntuar las películas: Este requisito enlaza con el núcleo de la idea del proyecto. La ejecución implica varios bloques. Requiere front-end, consultas a la base de datos, funciones de java script.

X-Tendría un buscador: No me dio tiempo a implementar esa funcionalidad. Sobreestime lo rápido que solucionaría el resto de requisitos.

X-Tendría una página dedicada a cada género: No me dio tiempo a implementar esa funcionalidad. Sobreestime lo rápido que solucionaría el resto de requisitos. En cierto modo esta implementado en la página principal pues hay secciones dedicadas a cada género, pero están todas juntas.

X-Tendría una sección de comentarios: No me dio tiempo a implementar esa funcionalidad. Sobreestime lo rápido que solucionaría el resto de requisitos y subestime lo amplio que este bloque podría ser. Además, tendría por necesidad que implementarse siempre después de todos los anteriores.

X-Tendría un buscador: No me dio tiempo a implementar esa funcionalidad. Sobreestime lo rápido que solucionaría el resto de requisitos.

Como conclusión general, reconozco que hasta que no pude plantear una estructura correcta (en varios ámbitos) desde el principio y tuve que rehacer varios bloques varias veces.

Por una parte, fue no tener los objetivos claros y por otra, no saber cual era el camino más eficiente y esto requirió necesariamente, ir descubriéndolo por el camino, ensayo y error.

5.2. Propuestas de mejora o ampliaciones futuras

- ✓ *Depurar el apartado gráfico. Estoy contento con el resultado final, pero me gustaría rehacerlo de cero con la perspectiva más global que tengo ahora, aunque conservaría bastantes elementos.*
- ✓ *Mejorar tipografías. Añadiría una selección más personal o al menos más variada.*
- ✓ *Agregar opción para que el usuario pueda corregir información de películas ya insertadas.*
- ✓ *Añadir una sección de comentarios de usuarios sobre cada película.*
- ✓ *Agregar un sistema de recuperación de contraseñas o nombres de usuarios.*
- ✓ *Expandir información sobre las películas.*
- ✓ *Añadir un buscador de películas por título, director, etc.*
- ✓ *Añadir una página dedicada a cada género.*

6. Guías

Bienvenido a la guía de uso de Wikicine.

Puedes disfrutar del sitio web de 2 formas distintas. Como usuario visitante o como miembro registrado.

Comencemos con la experiencia como usuario visitante.

En la Landing page (index) clicas en el botón y saltas a la página principal.

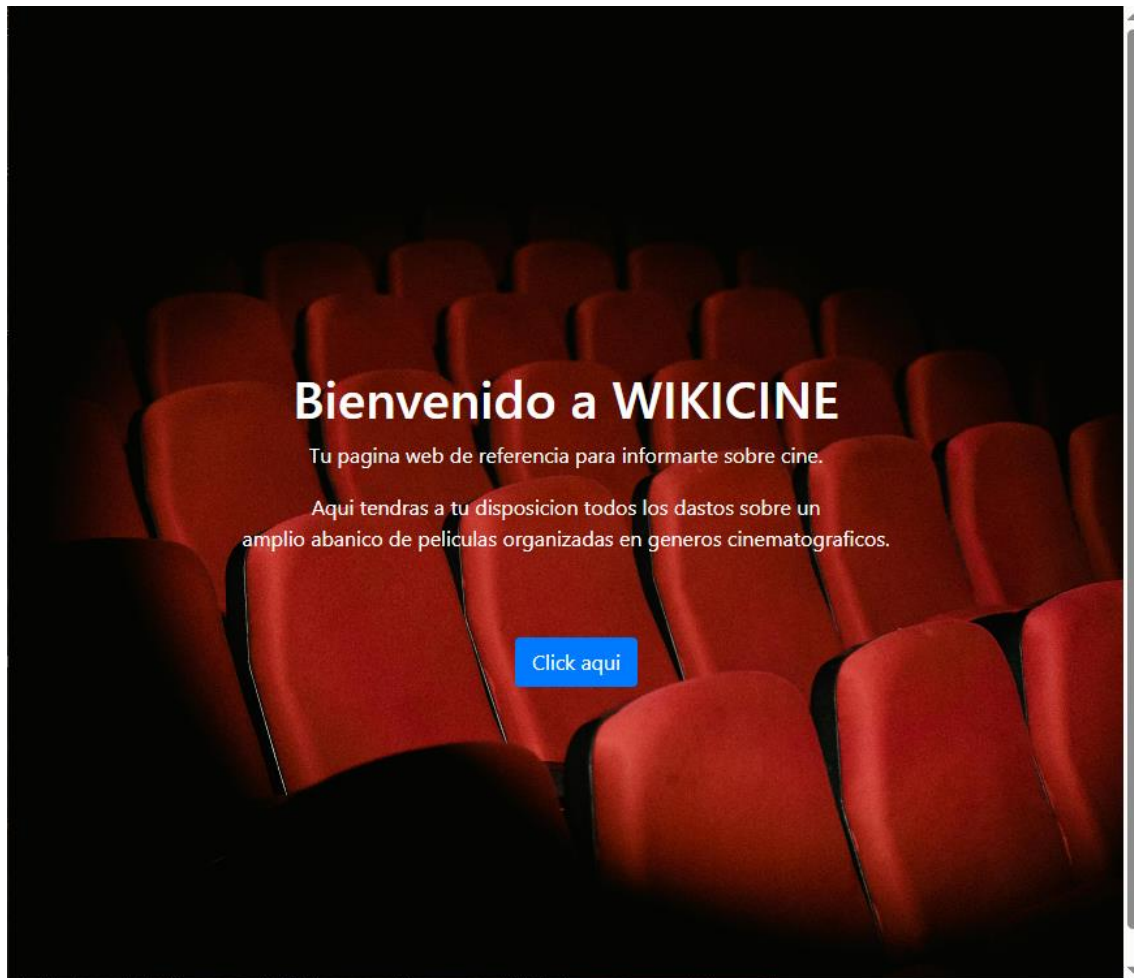


Fig. 6-1- Pagina de portada o index

Una vez en la página principal podrás visualizar 2 cosas. Las pastillas y la barra de menú principal.

En Pc la barra del menú se vería así:

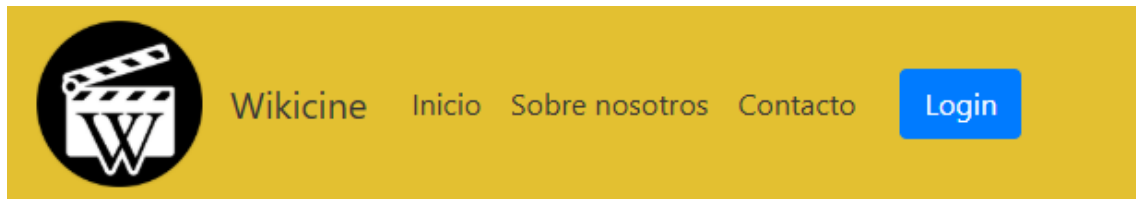


Fig. 6-2- Barra del menú versión Pc

En dispositivos móviles la barra del menú se vería en forma comprimida como un menú de hamburguesa.

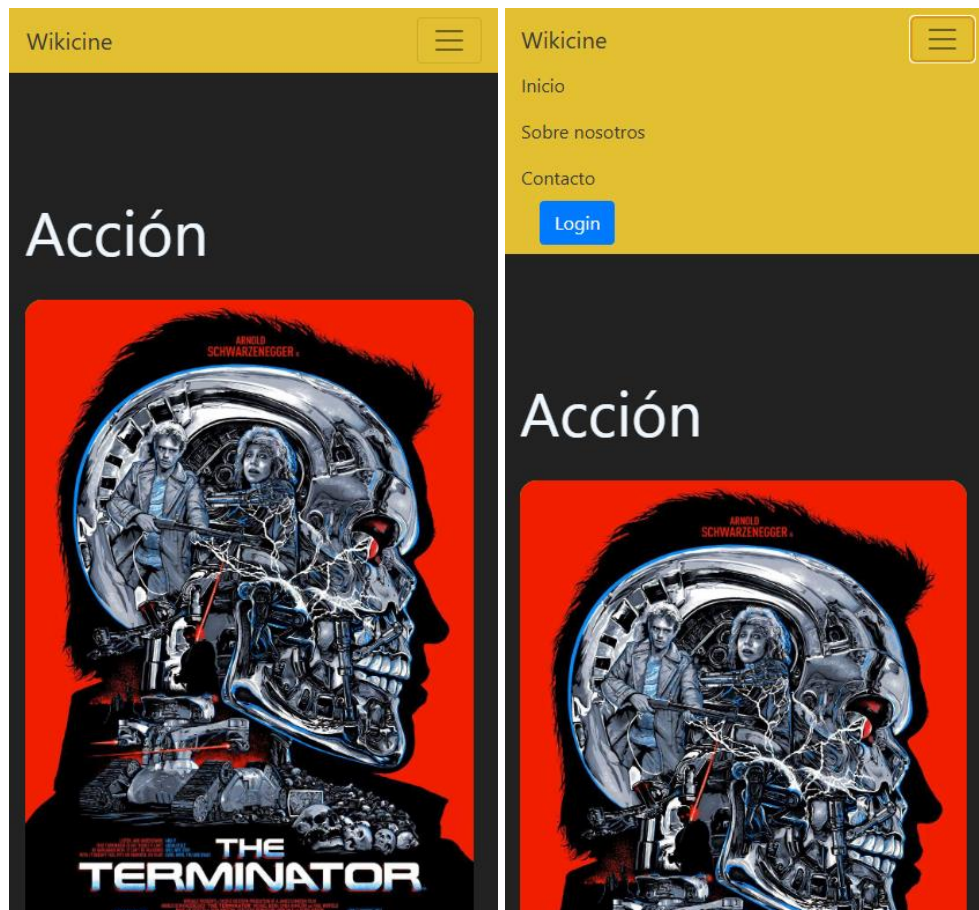


Fig. 6-3- Barra del menú versión móvil con menú contraído

Fig. 6-4- Barra del menú versión móvil con menú desplegado

Varias pastillas de películas, formadas por el poster y unos pocos datos, agrupadas por género.

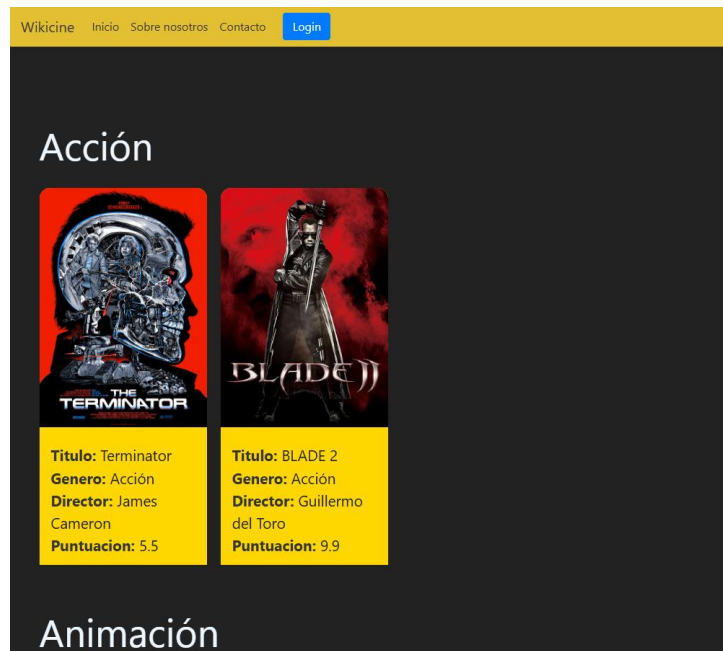


Fig. 6-5- Sección Acción de Página principal

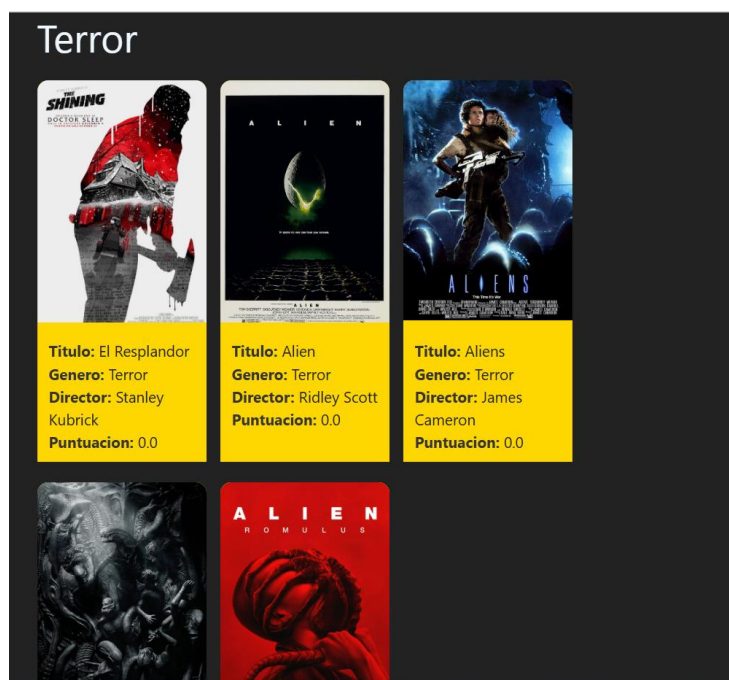


Fig. 6-6- Sección Terror de Página principal

Si clicas en cualquiera de estas miniaturas, accederás a la página de la película seleccionada, donde obtendrás más datos sobre la película, incluido un trailer (de youtube) sobre esta.

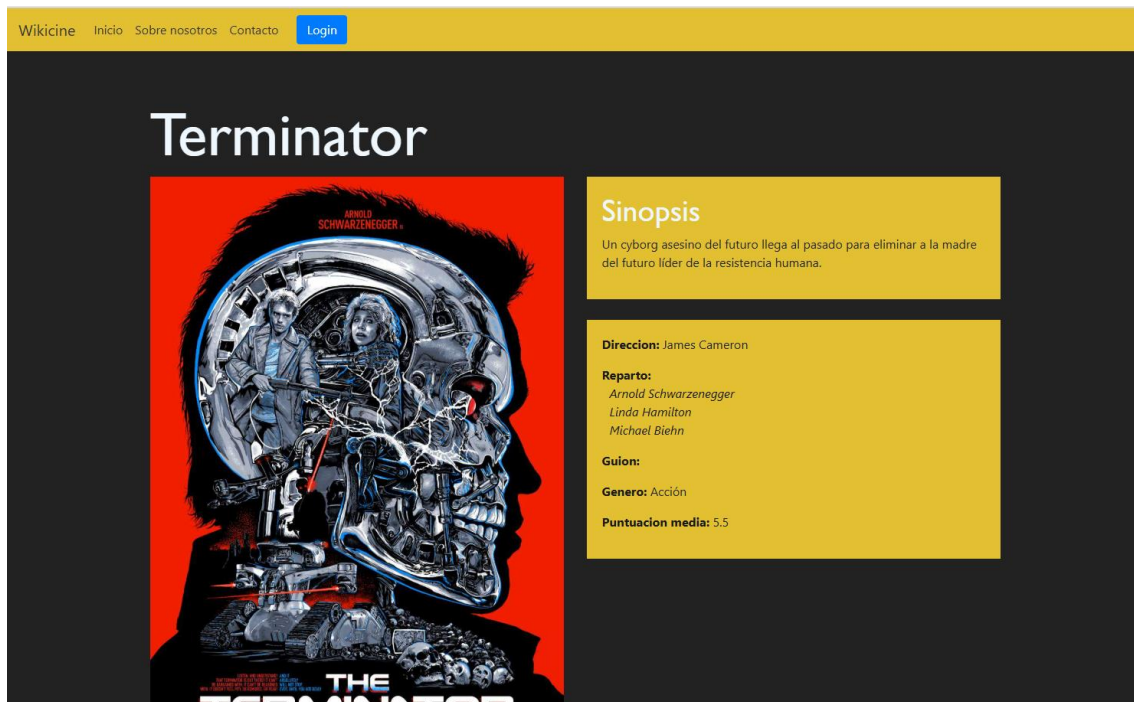


Fig. 6-7- Página de película seleccionada (Terminator) parte superior

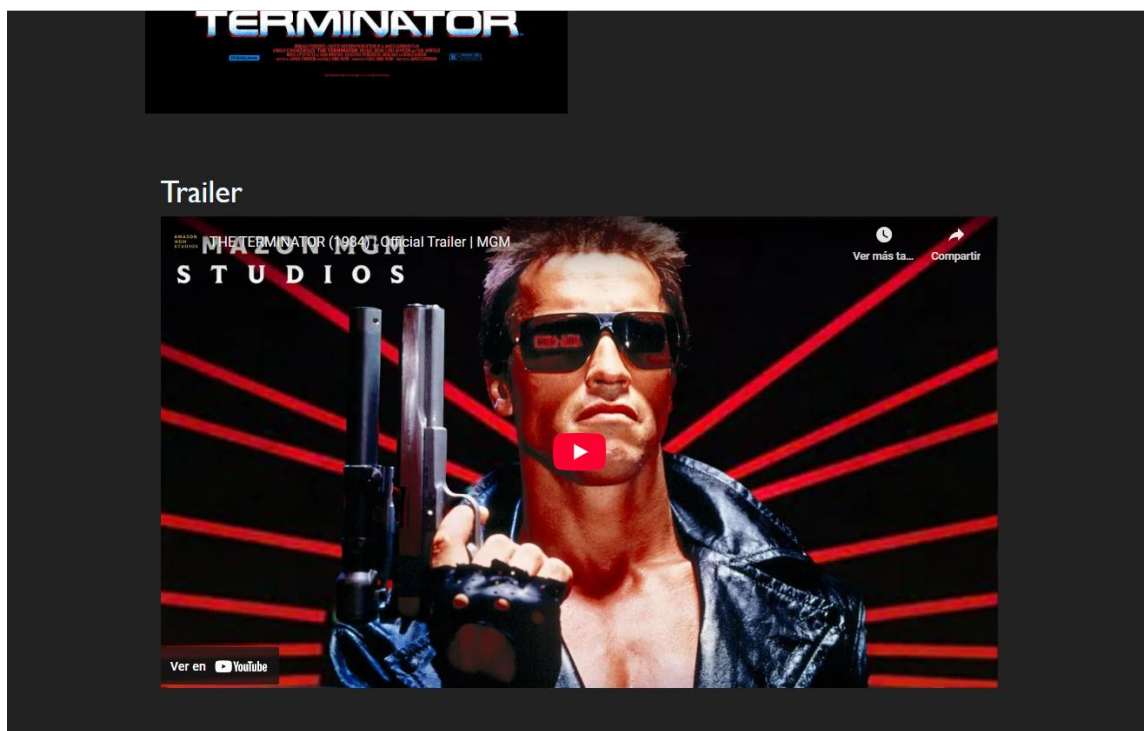


Fig. 6-8- Página de película seleccionada (Terminator) parte inferior

Para iniciar sesión basta con pulsar el botón de login de la barra del menú principal en cualquiera de las páginas del sitio.

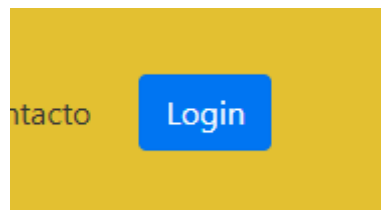


Fig. 6-9- Botón de login en menú principal

Al pulsarlo se desplegará el formulario de login que constará de 2 campos: Usuario y Contraseña. Para acceder bastara con rellenar dichos campos y pulsar el botón de login.

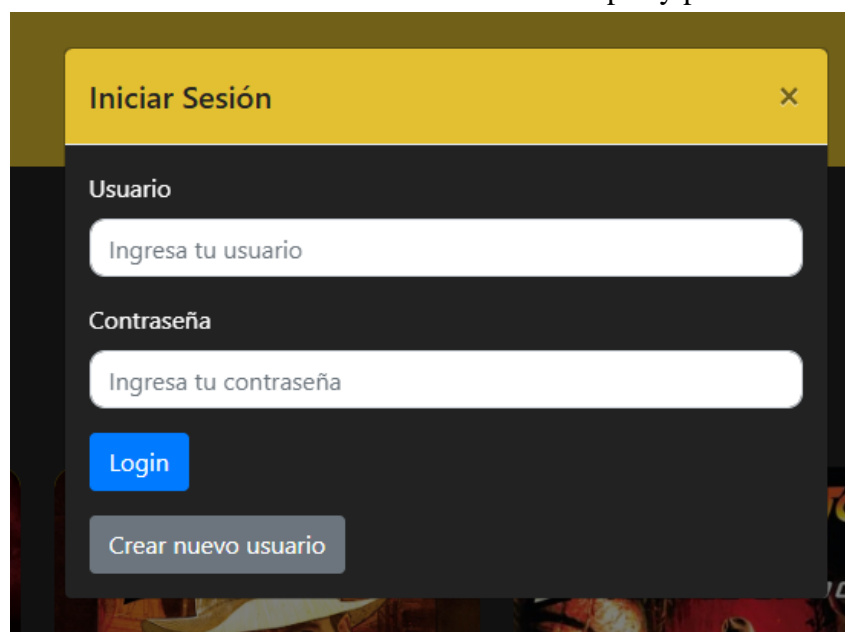


Fig. 6-10- Ventana modal de login

Tras loguearte, independientemente de donde te encuentres se te redirigirá a la página principal. Desde allí puedes seguir navegando como normalmente, pero habrá algunos cambios.

Como por ejemplo la barra de menú, que ahora tiene 2 botones nuevos. Uno para añadir nueva película y otro para cerrar la sesión.



Fig. 6-11- Barra de menú tras login

Si tras loguearte, accedes a la página específica de una película, ahora además de los botones nuevos el menú principal también habrá un nuevo botón llamado “puntuar” con el que podrás asignar.

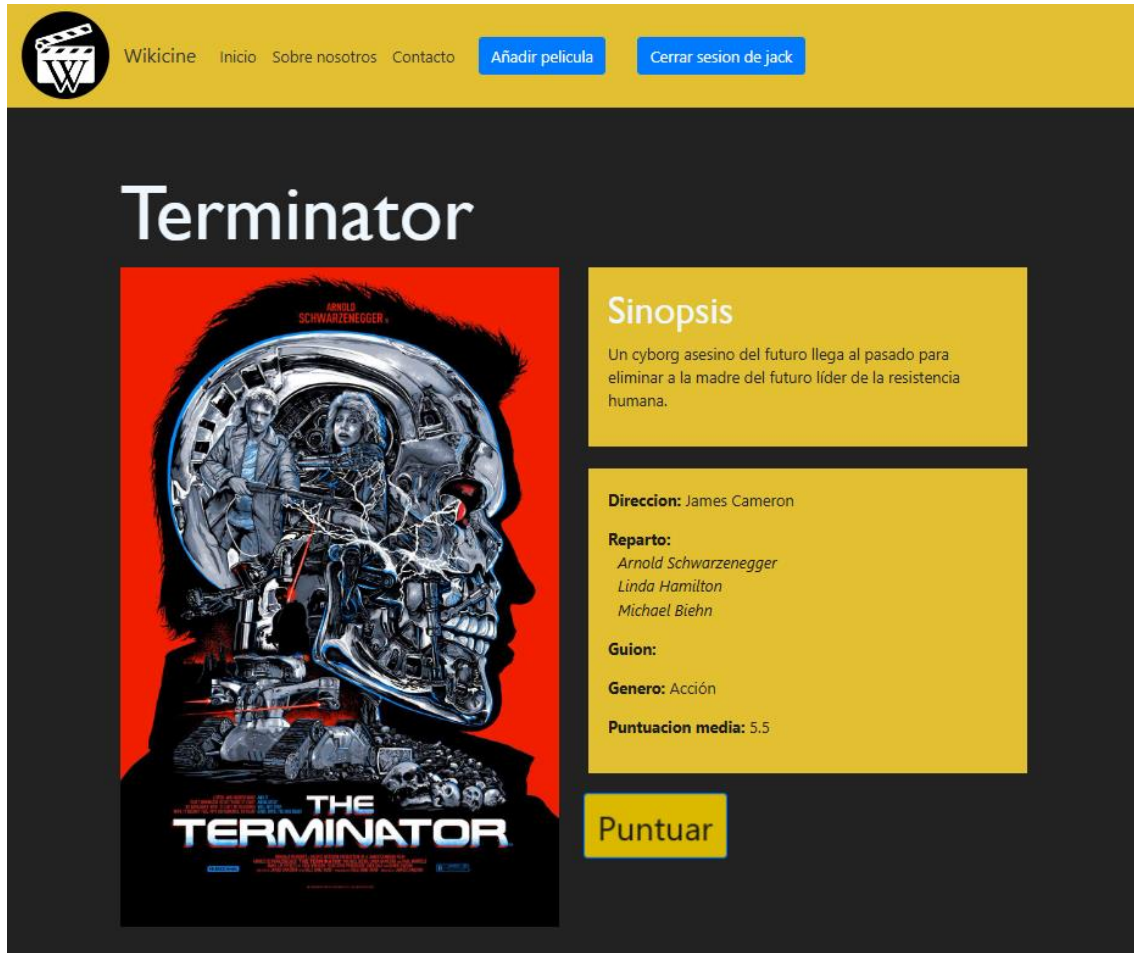


Fig. 6-12- Página de película seleccionada tras login



Fig. 6-13- Botón de puntuar

Si clicas en el botón de puntuar emergerá una ventana modal para votar. Podrás observar como la puntuación media se actualiza en tiempo real.



Fig. 6-14- Ventana modal de voto antes de seleccionar puntuación



Fig. 6-15- Ventana modal de voto después de seleccionar puntuación

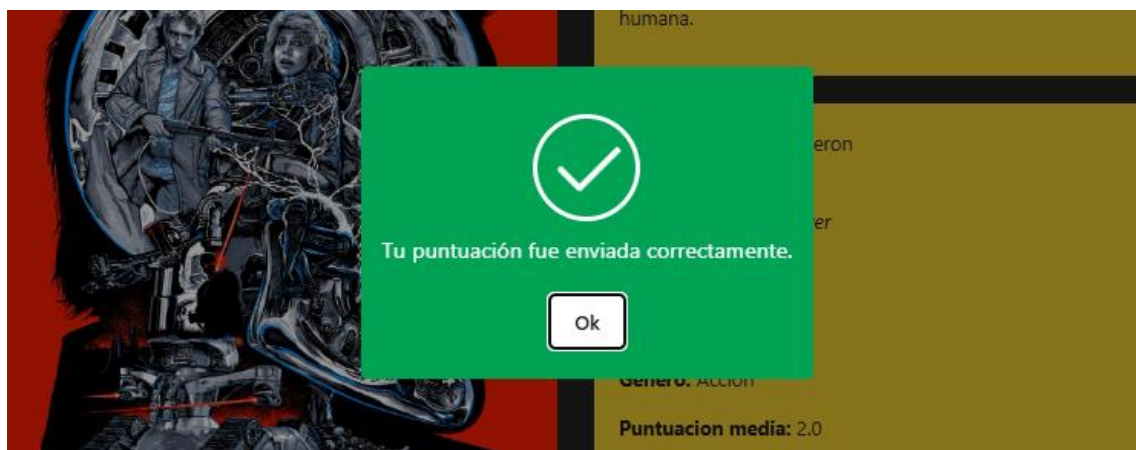


Fig. 6-16- Ventana modal de voto emitido exitosamente

Volviendo a la barra del menú principal, a través del botón de añadir nueva película, puedes acceder a una página del sitio web para añadir una nueva película.



Fig. 6-17- Botón de añadir película en barra de menú

Una vez aquí podrás añadir los datos de la nueva película a través de este formulario.
*Como se pude observar a en la barra de menú solo contiene el botón de cerrar sesión.

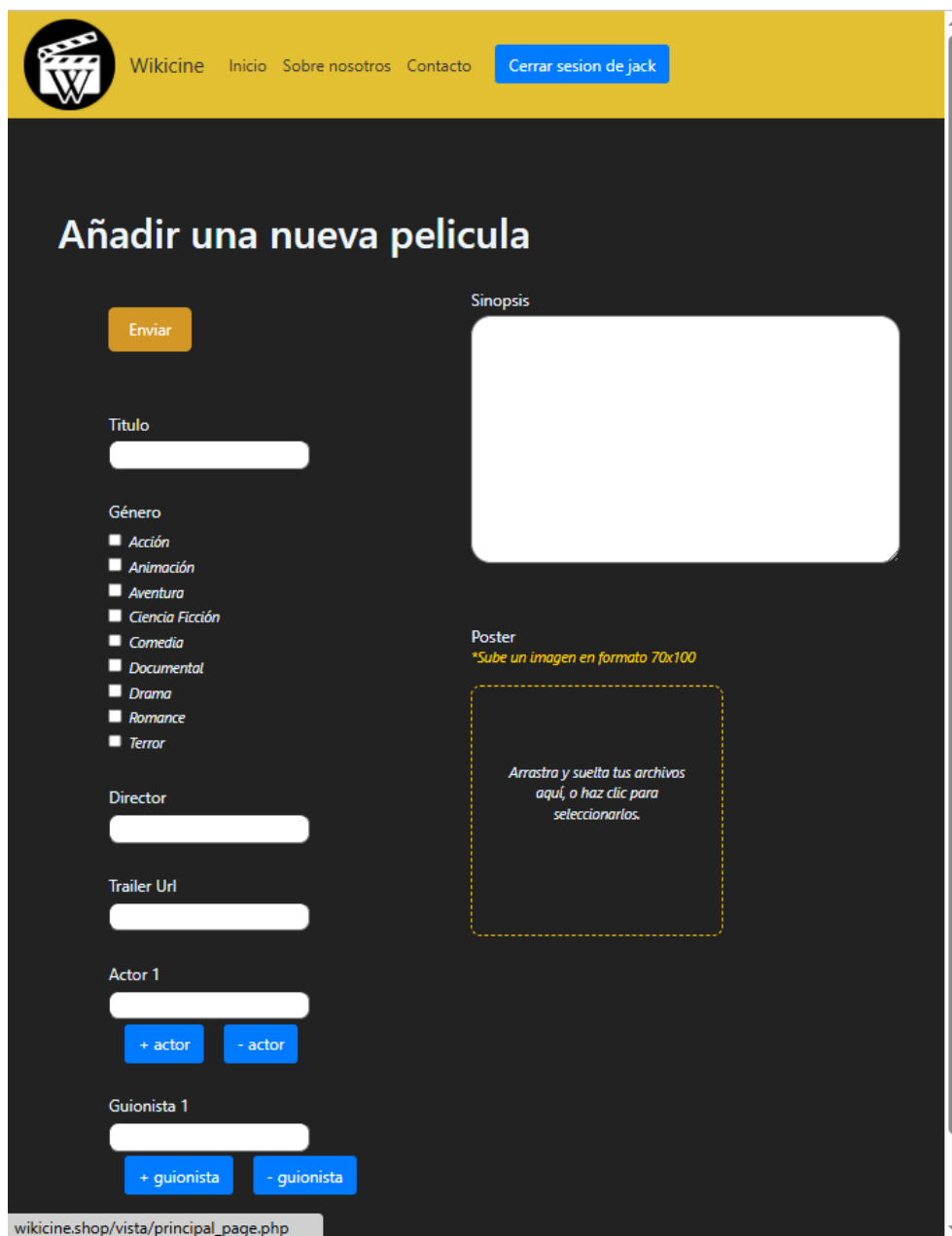


Fig. 6-18- Pagina de formulario para añadir película

Al rellenar alguno de los campos de forma incorrecta mostrara una ventana modal que te avisara de ello. Estos avisos evitaran que puedas enviar el formulario hasta que corrijas los errores.

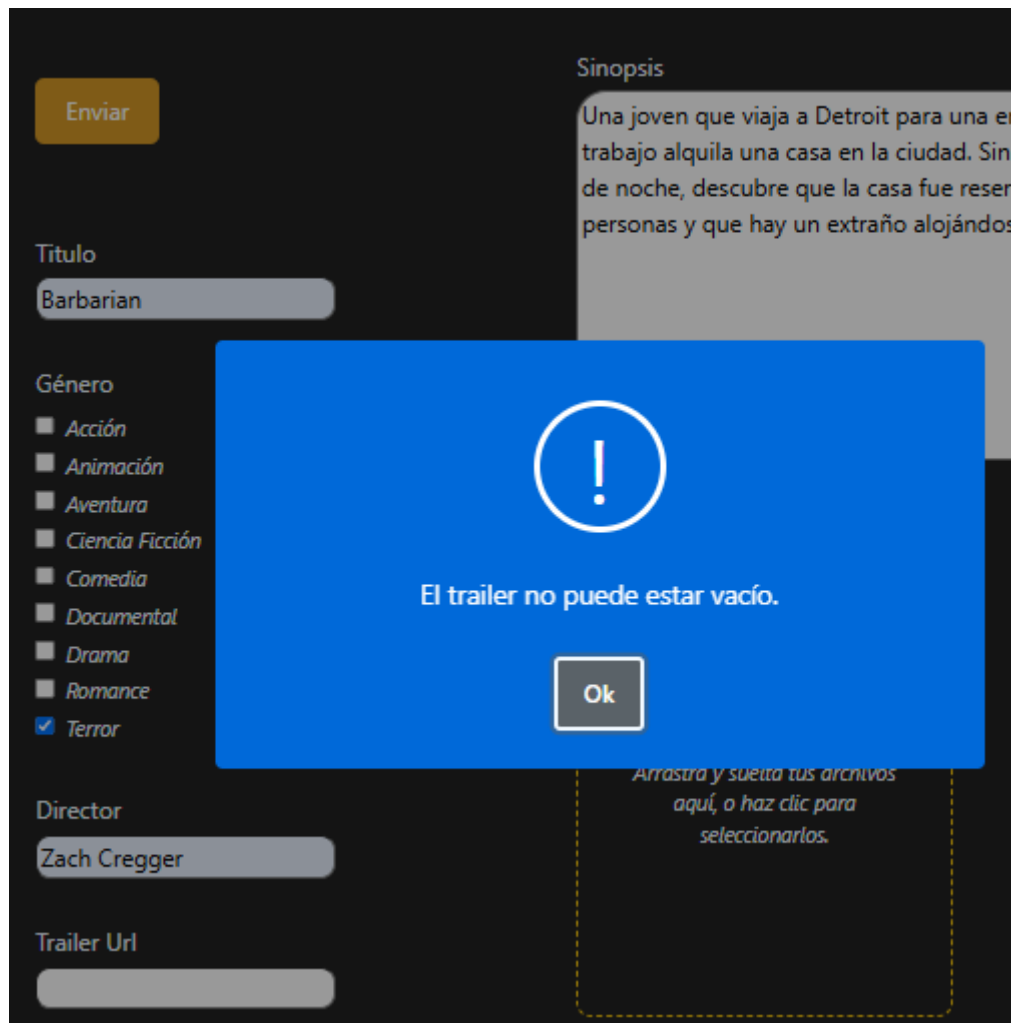
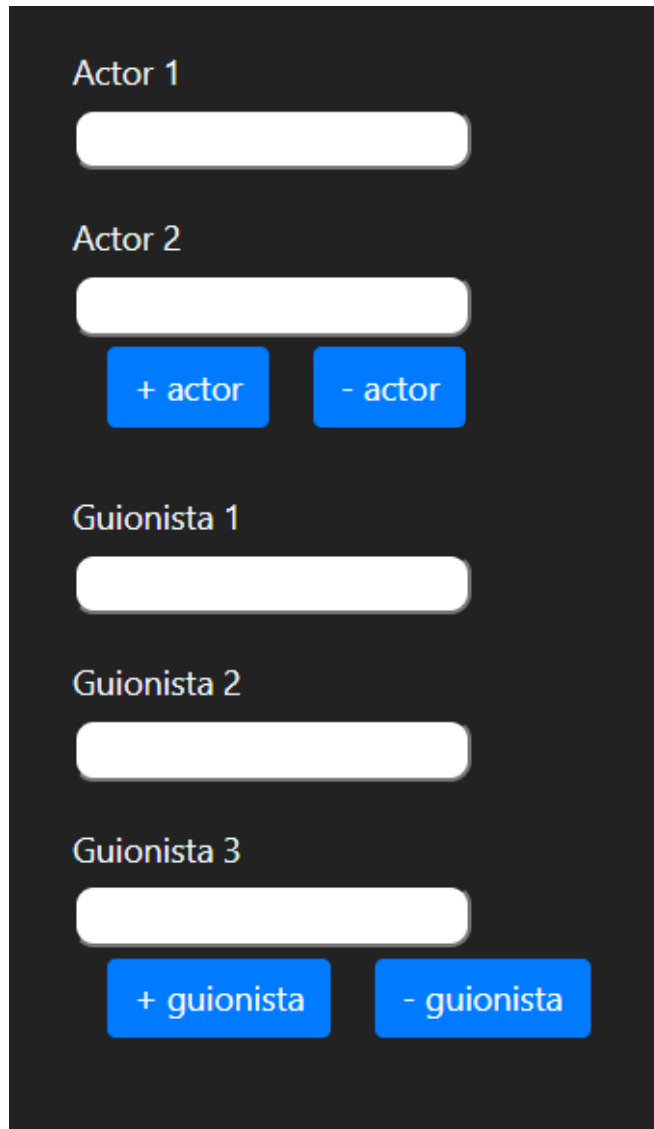


Fig. 6-19- Ventana modal de error al añadir película

Como cada película tiene un numero de actores y de guionistas variables, puedes añadir un campo para un nuevo actor o guionista a través de los botones correspondientes.



Fig. 6-20- Botones para añadir nuevos campos de actor y guionista



Formulario con nuevos campos añadidos de actor y guionista. El formulario tiene un fondo negro y contiene los siguientes elementos:

- Actor 1:** Un campo de texto blanco.
- Actor 2:** Un campo de texto blanco, con dos botones azules debajo: "+ actor" y "- actor".
- Guionista 1:** Un campo de texto blanco.
- Guionista 2:** Un campo de texto blanco.
- Guionista 3:** Un campo de texto blanco, con dos botones azules debajo: "+ guionista" y "- guionista".

Fig. 6-21- Formulario con nuevos campos añadidos de actor y guionista

De la misma forma puedes eliminar cada uno de estos campos (excepto el primero) con los botones correspondientes.



Fig. 6-22- Botones para eliminar los nuevos campos actor y guionista

Si has rellenado todos los campos correspondientes te llevara a la página de confirmación de éxito con un enlace para ir a la página principal.

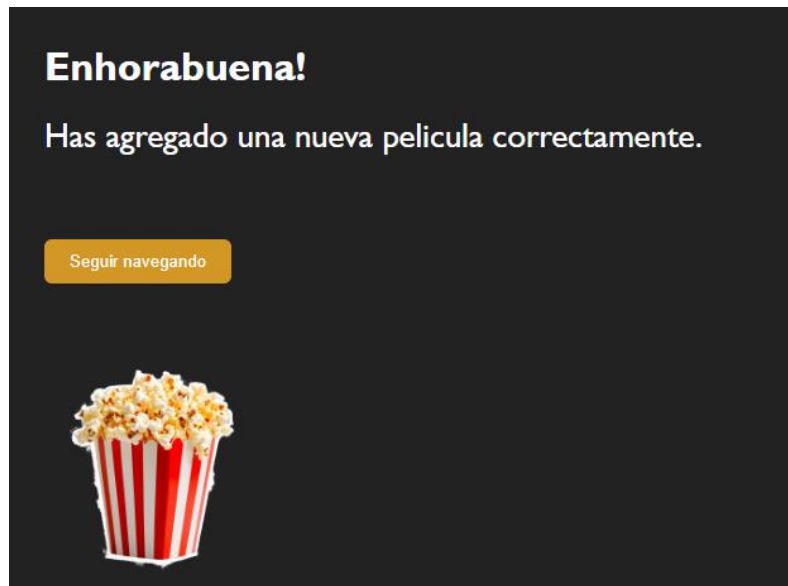


Fig. 6-23- Pagina de película añadida exitosamente

Si hubiera ocurrido algún problema a la hora de enviar el formulario se te redirigiría a esta otra página. También contiene un enlace para volver a la página principal.



Fig. 6-24- Pagina de película añadida erróneamente

Si no estas registrado en Wikicine, puedes acceder al formulario de registro a través del botón de “Crear nuevo usuario”, que se encuentra en la misma ventana modal de login.

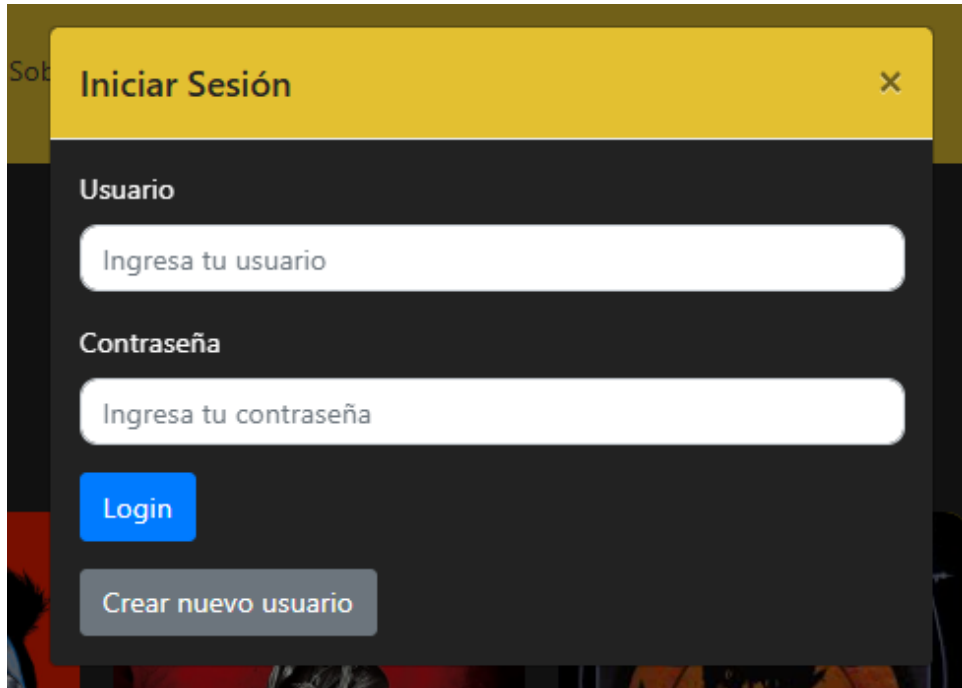
A modal window titled "Iniciar Sesión" (Login) with a close button (X) in the top right corner. It contains two input fields: "Usuario" (User) with placeholder text "Ingresa tu usuario" and "Contraseña" (Password) with placeholder text "Ingresa tu contraseña". Below the fields are two buttons: a blue "Login" button and a grey "Crear nuevo usuario" (Create new user) button.

Fig. 6-25- Botón para crear nuevo usuario en la ventana modal de login

Una vez en la página de registro tienes acceso un sencillo formulario de 3 campos: Nombre de usuario, un correo electrónico y una contraseña.

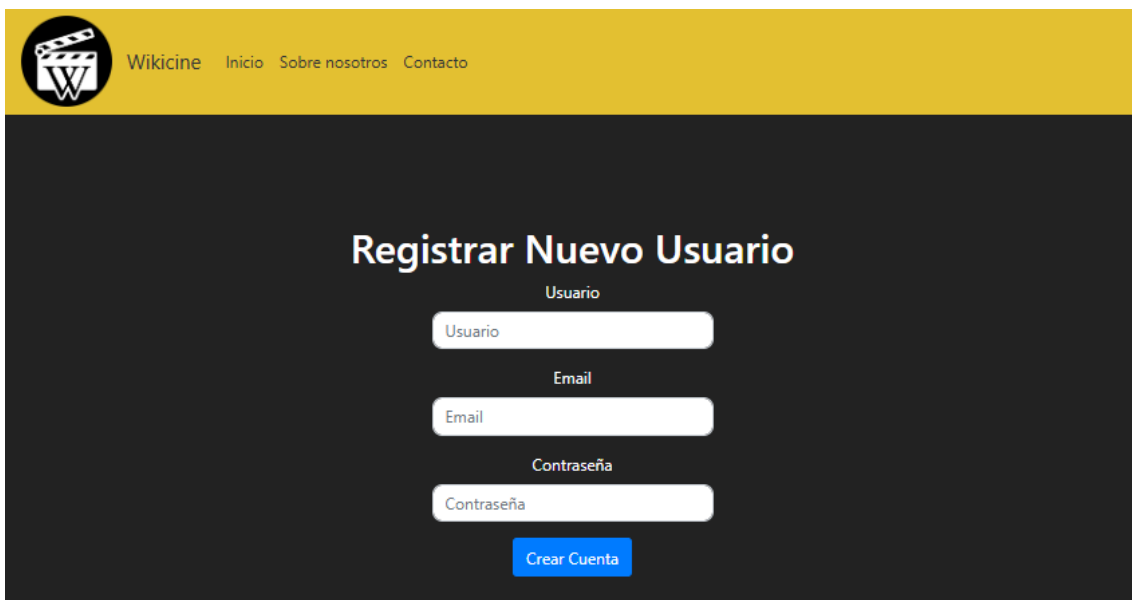
A registration form titled "Registrar Nuevo Usuario" (Register New User). It features three input fields: "Usuario" (User) with placeholder text "Usuario", "Email" with placeholder text "Email", and "Contraseña" (Password) with placeholder text "Contraseña". Below the fields is a blue "Crear Cuenta" (Create Account) button. The form is set against a dark background with a yellow header containing the Wikicine logo and navigation links: "Wikicine", "Inicio", "Sobre nosotros", and "Contacto".

Fig. 6-26- Formulario para crear nuevo usuario

Al intentar enviar el formulario cuando has dejado vacío o has rellenado erróneamente alguno de los campos, salta una ventana modal que te avisa de ello y aporta las instrucciones correspondientes.

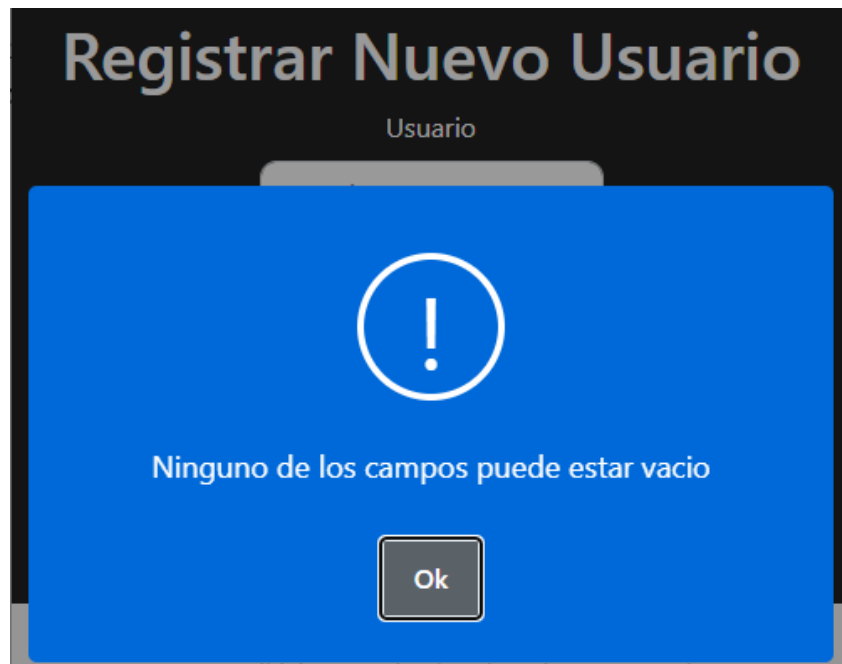


Fig. 6-27- Ventana modal de campo vacío

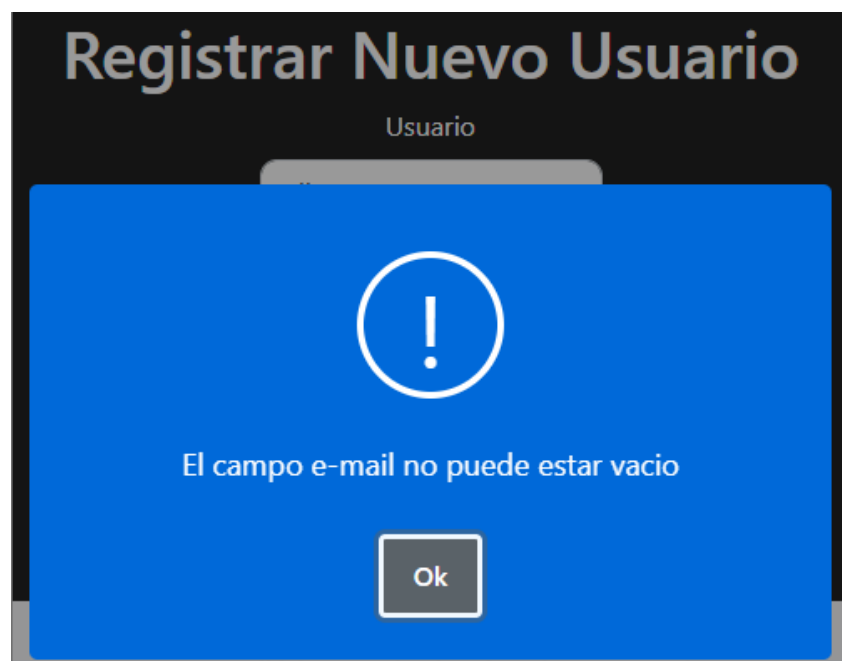


Fig. 6-28- Ventana modal de correo ya registrado

Cuando has rellenado todos los campos correctamente puedes enviar sin problema el formulario pulsando el botón “enviar” y se te redirigirá a la página correspondiente.

Si a no ha ocurrido ningún problema al enviar tu formulario de registro se te redirigirá a esta página la cual contiene un botón para ir a la página principal.

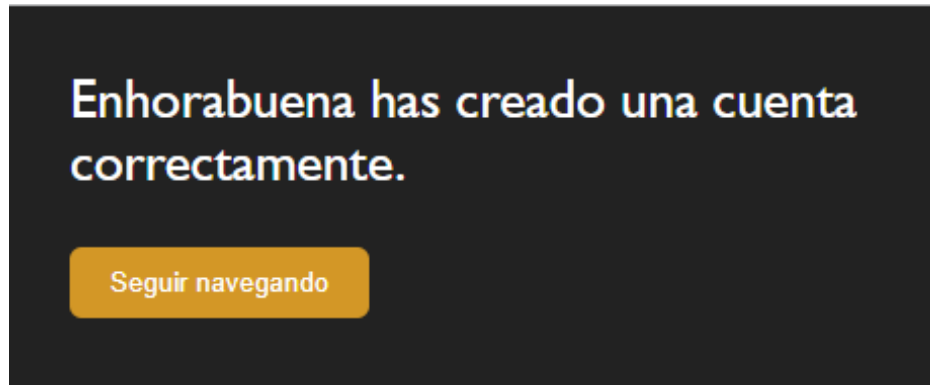


Fig. 6-29- Pagina de usuario registrado exitosamente

Si a ocurrido algún problema al enviar tu formulario de registro se te redirigirá a esta que también contiene un botón para ir a la página principal.

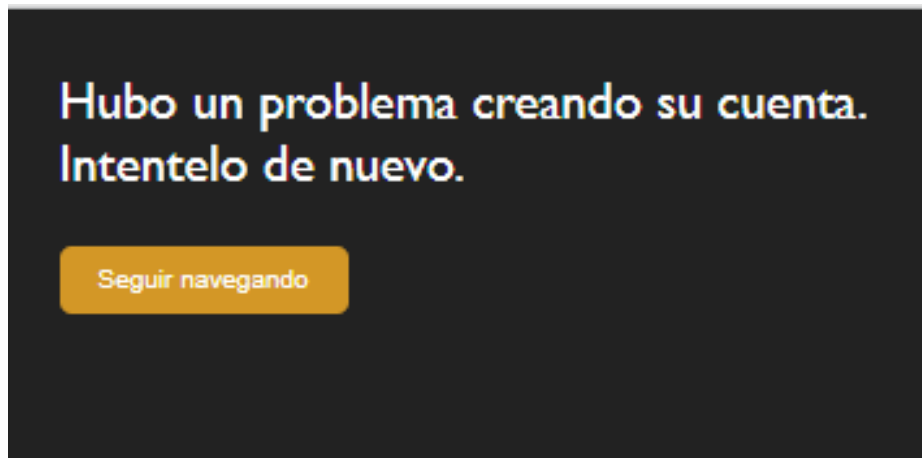


Fig. 6-30- Pagina de usuario creado erróneamente

7.Referencias

Desarrollo web en entorno servidor. DAW. Curso 2º - **Apuntes del Profesor:** María Esther González Lorenzo

Desarrollo Web en Entorno Cliente 2º - **Apuntes del Profesor:** Ana Rosa Álvarez Rojo

Html[1]

Css[2]

Php[3]

JavaScript[4]

Sql[5]

BootStrap[6]

SweetAlert2[7]

Talen.com[8]

Ceac.es[9]

Global66.com[10]