

**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

**FACULTAD DE ESTUDIOS SUPERIORES**

**“ARAGÓN”**

**ALUMNOS.**

**GONZÁLEZ MEDINA FERNANDO ISAAC.**

**JIMÉNEZ BOLIO JULIO CÉSAR.**

**MIRAMONTES RAMÍREZ ANDRÉS MAURICIO.**

**SÁNCHEZ SÁNCHEZ VÍCTOR MANUEL.**

SISTEMAS DE INFORMACIÓN

GRUPO. 2860

**PROYECTO FINAL.**

14 DE JUNIO DE 2020

# PROYECTO FINAL.

**INTEGRANTES.**

González Medina Fernando Isaac.

Jiménez Bolio Julio César.

Miramontes Ramírez Andrés Mauricio.

Sánchez Sánchez Víctor Manuel.

**INTRODUCCIÓN.**

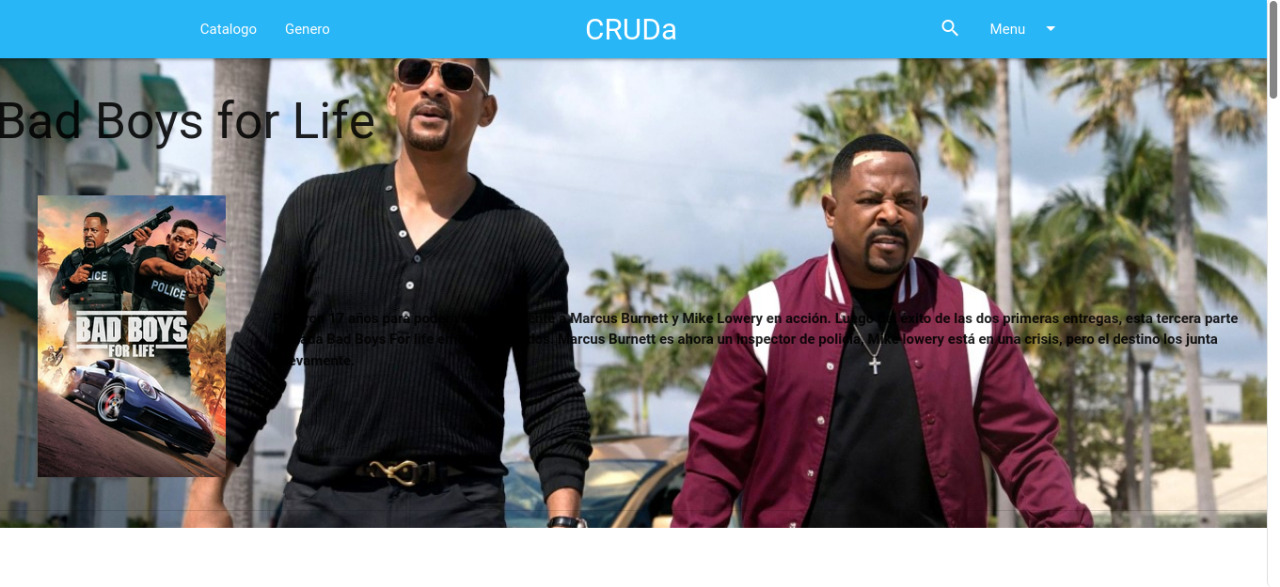
Se decidió crear un catálogo de películas. Éste deberá cumplir con los requerimientos listados a continuación para ser accesible y brindarle una experiencia grata al usuario.

Requerimientos:

* Que la lista esté actualizada y organizada.
* Que la interfaz sea sencilla y amigable con los usuarios.
* Que las descripciones cumplan con la expectativa siendo sencillas y concretas, guardando por supuesto cualquier detalle importante de la película.

**FRONT END.**

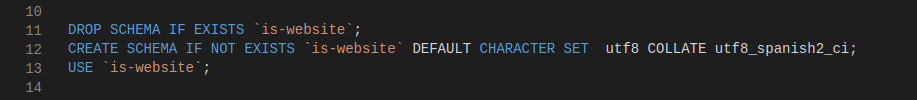
En este módulo se utilizó el *framework* de materialize css para facilitar el trabjo de estilo.



**CONEXIÓN-BASES DE DATOS.**

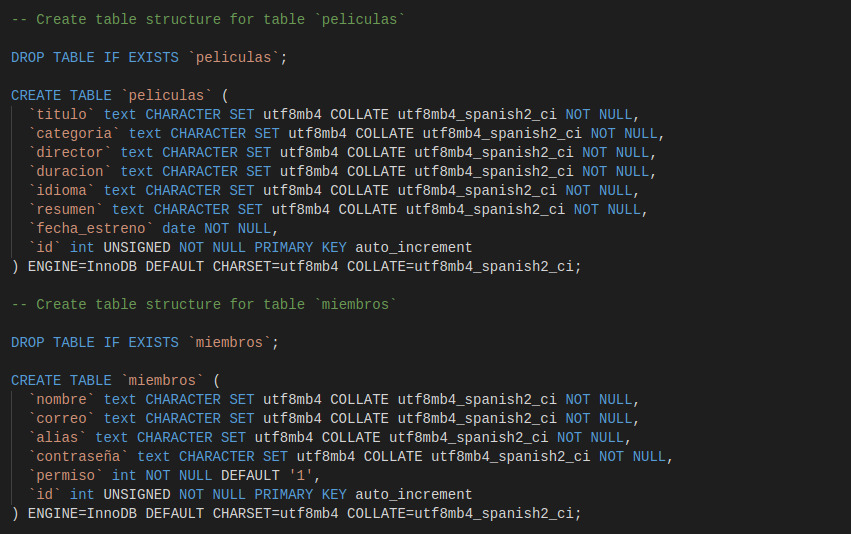
Primero se creó la base de datos “is-website”. (*Imagen 1.1)*

Imagen .1



En esta base se crearon únicamente dos tablas; una para las películas y otra para los usuarios registrados. (*Imagen 1.2)*

Imagen 1.



Después se realizó la conexión a la base de datos con un usuario con permisos para poder realizar cambios en las bases. (*Imagen 1.3)*

También hay que aclarar que es necesario que se tenga conexión a la base de datos en todo momento. (*Imagen 1,4)*

Imagen 1.

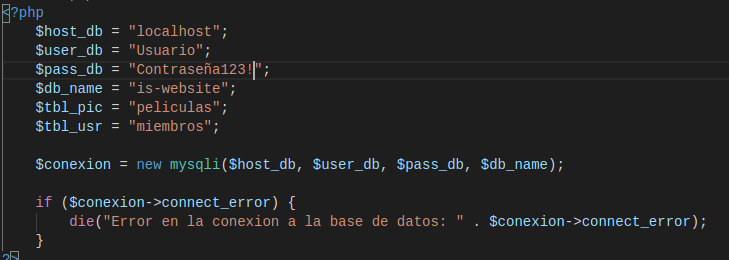


Imagen 1.

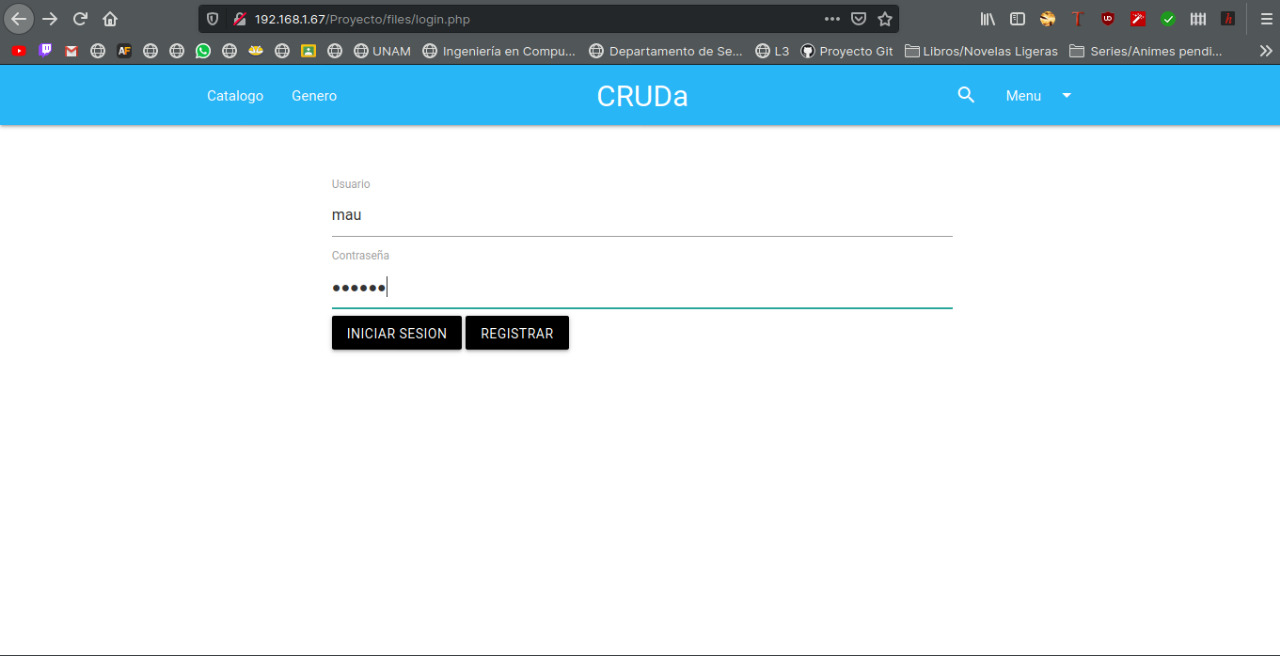


**BACK-END.**

Para poder realizar cambios en la base de datos es necesario ingresar al módulo admin.php con un usuario que tenga = 2.

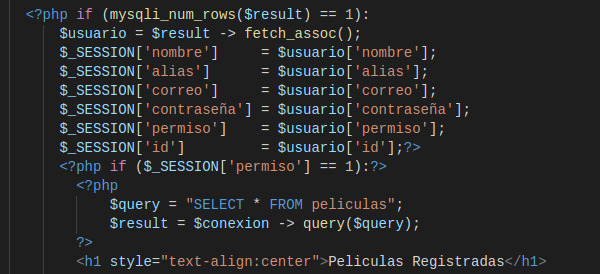
Se ingresa a admin.php a través de login.php. (*Imagen 2.1-login.php).*

Imagen 2.1-login.php



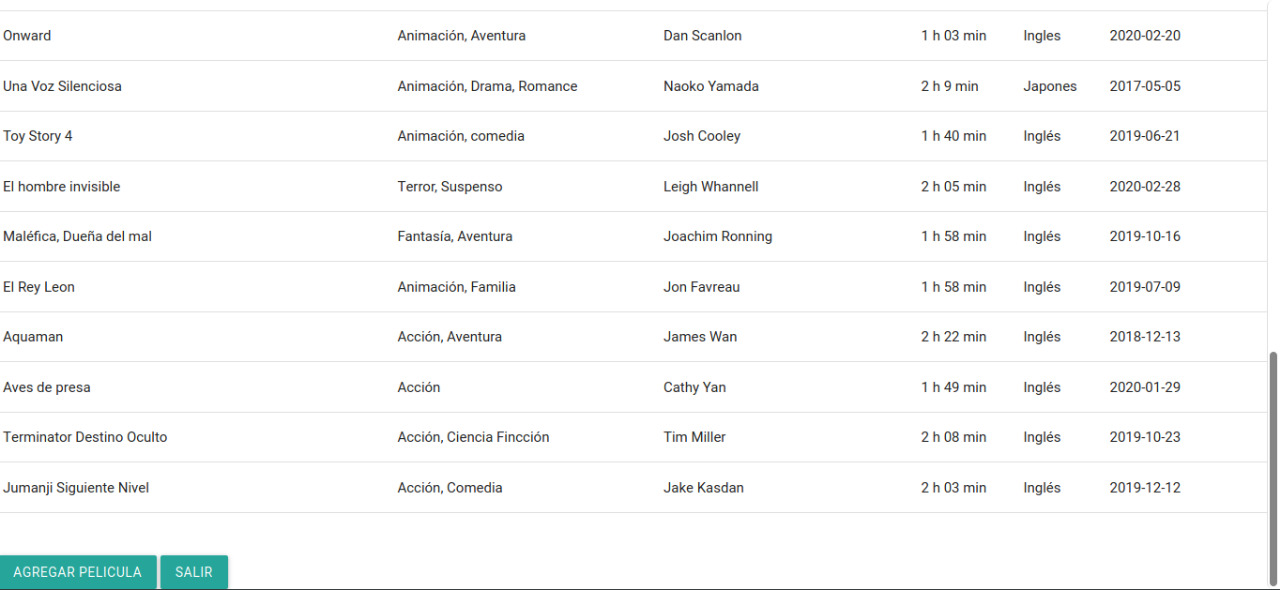
En admin.php se revisará al usuario que se registró y si tiene un permiso de 1 solo se desplegara una tabla con las películas registradas y un botón para agregar películas o salir de la sesión, de igual manera esto explicará un posible **SUBMODULO READ**. (*Imagen 2.2).*

Imagen 2.2



admin.php en caso de que permiso == 1 (*Imagen 2.3)*

Imagen 2.3

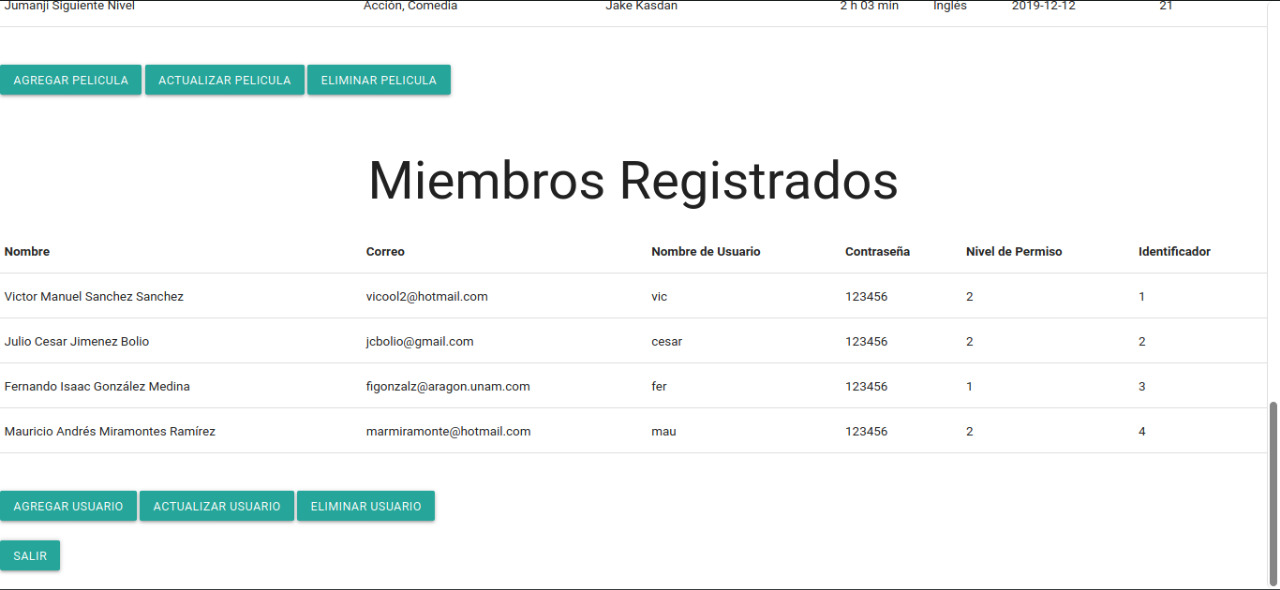


En caso de que el usuario tenga un permiso igual a 2 se desplegaran 2 tablas, una tabla con todas las películas registradas y 3 botones para:

-Agregar película.  
-Actualizar película.  
-Borrar película.

Y una tabla con los usuarios registrados de la página con los mismos 3 botones para realizar cambios. (*Imagen 2.4)*

Imagen 2.4



**CREATE.**

Al seleccionar el botón "Agregar" se hace llamar a create.php.

También se manda una variable con identificador "v", el objetivo de esta variable es poder determinar si se va a agregar una película o un nuevo miembro. Así como se muestra en las imágenes 3.1 y 3.2

Imagen 3.1



Imagen 3.2



create.php tomara el valor de "v" y en base a eso desplegara un formulario para películas o para miembros con los datos requeridos por la base de datos.

Imagen 3.3-m = miembros

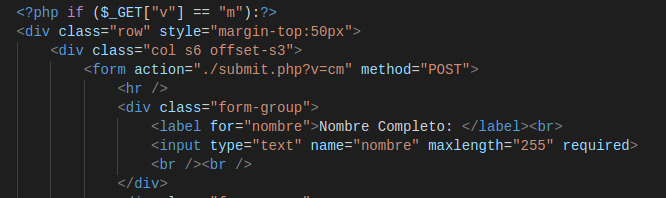


Imagen 3.4-p = peliculas

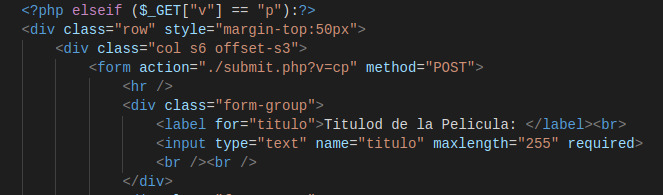
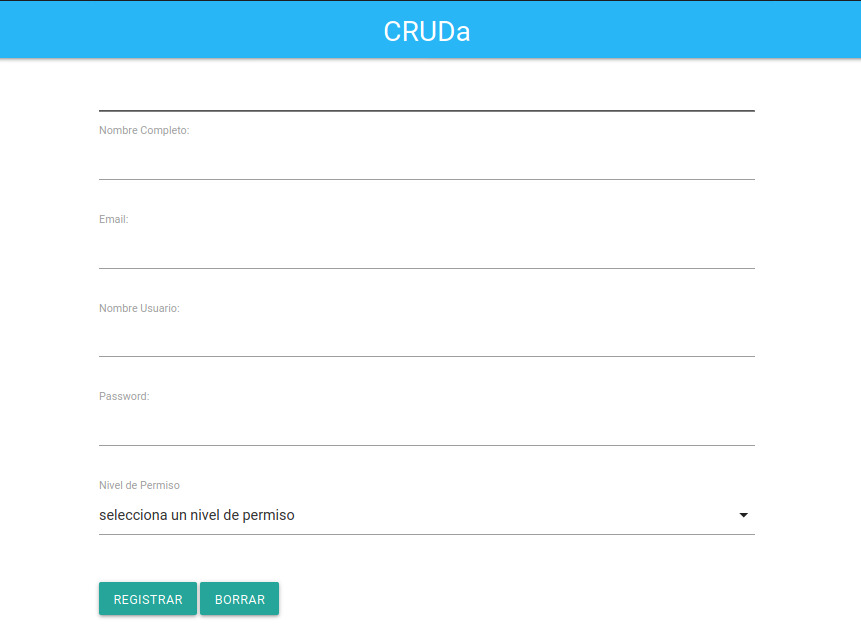


Imagen 3.5-form de pelucula



Imagen 3.6-form de usuario.



create.php tomará los datos ingresados en el formulario (*Imágenes 3.5-form de pelicula y 3.6-form de usuario)* y al hacer clic en el botón "Registrar" mandará a llamar a submit.php, después le dará los datos registrados mediante un POST (*Imágenes 3.4-m = miembros y 3.4-p = peliculas)*

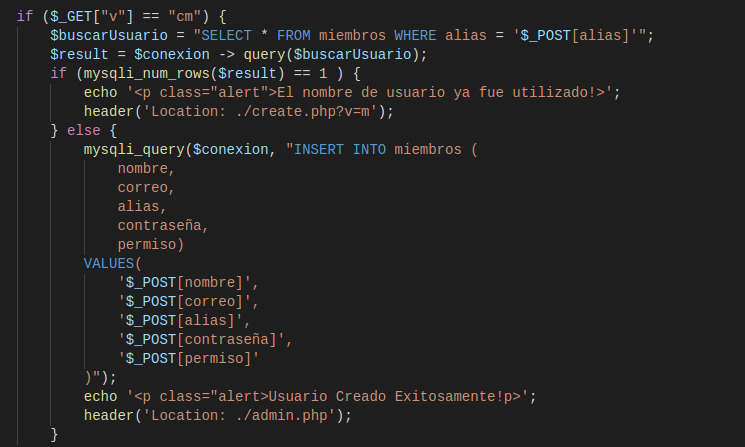
Cabe aclarar que submit.php es en donde se realizaran todos los cambios a la base de datos, además se utilizara el valor de "V" para determinar qué tipo de accion se tomara en la base de datos.

Create.php manda "v" con el valor de "cp" o "cm" donde:

- "cp" = create película  
- "cm" = create miembro

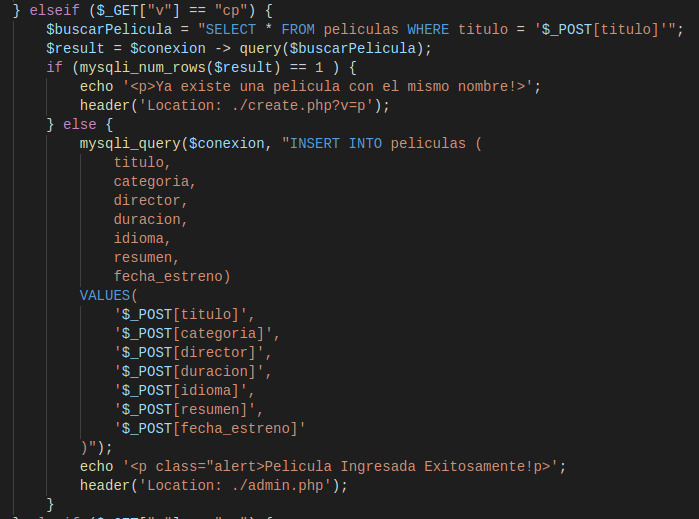
En caso de ser "cm" se hará un INSERT INTO miembros con los valores que mando create.php (*Imagen 3.7)*

Imagen 3.7



En caso de ser "cp" se hará un INSERT INTO películas con los datos recibidos por create.php (*Imagen 3.8)*

Imagen 3.8

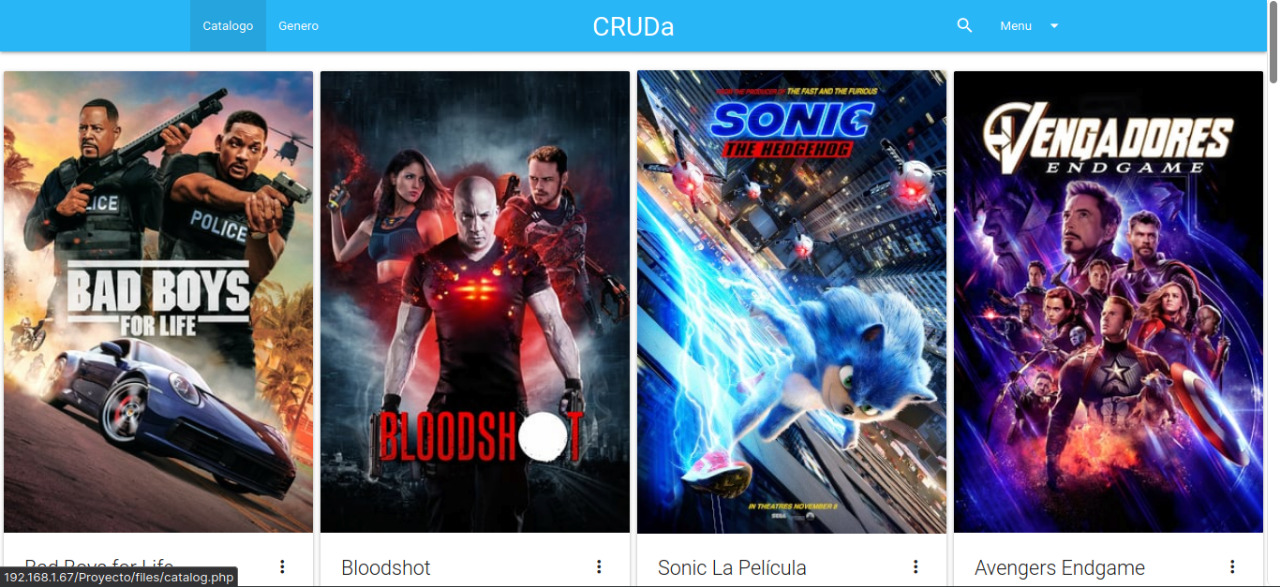


Y se dará un mensaje para hacer saber al usuario que se ha agregado exitosamente la película/miembro nuevo.

**READ.**

Cuando se le da clic en "catalogo" en la parte superior de la página se mandará a llamar a catalogo.php *Imagen catálogo)*

Imagen catálogo



catalogo.php desplegara una tabla donde se mostaran los posters de las peliculas registradas en la base de datos junto con su nombre y genero en la parte inferior. Cabe mencionar que aquí se utilizó la clase "card" del materialize css.

La imagen del poster se toma de la carpeta "images" con la siguiente nomenclatura:

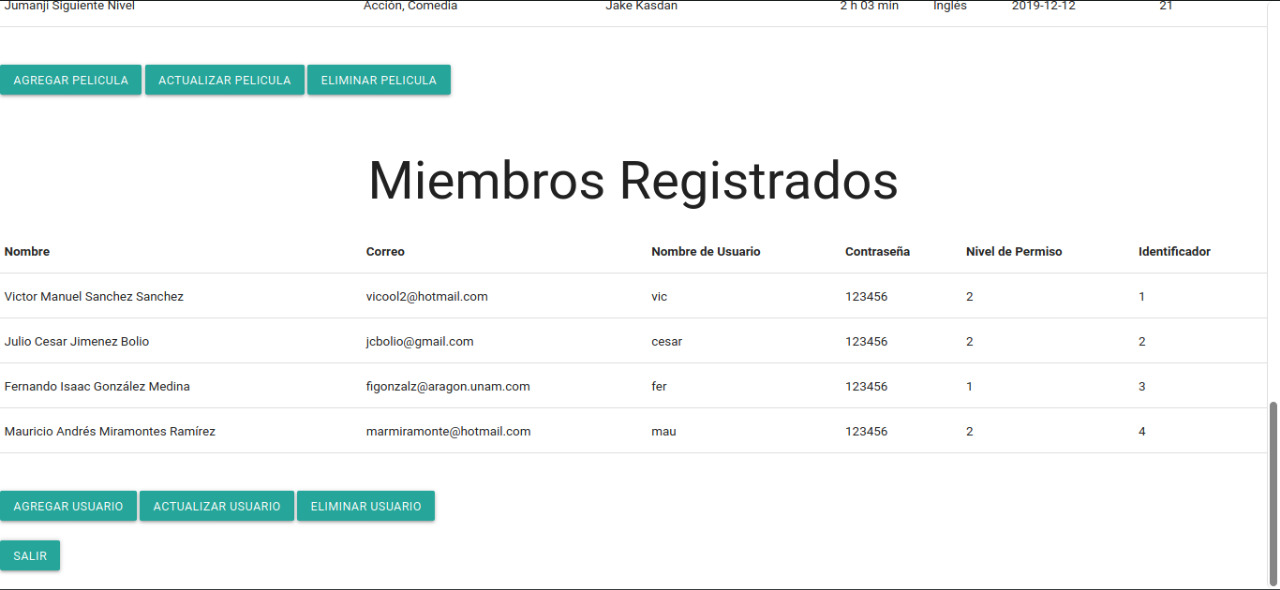
NombrePelicula-poster,jpg.

Al momento de dale clic en una de las imágenes se despliega mediante la clase "card-reveal" de materialize toda la información que se tiene en la base de datos de la película

**UPDATE/DELETE.**

Para estar en estos submódulos es necesarios dar clic en actualizar o eliminar usuario/película (*Imagen 4.1)* Se mandará a llamar a select\_item.php y se le mandará una variable llamada "s"

Imagen 4.1



select\_item.php nos desplegara una lista de todas(os) las(os) Peliculas/Miembros, según se haya elegido, que se tengan registrados en este momento y permitirá el usuario elegir cual de ellos será al que se le aplicara la opción elegida. (*Imagen 4.2-peliculas, Imagen 4.3-usuarios).* EL valor de "s" nos dirá cuál fue el cambio elegido por el usuario

Imagen 4.2-peliculas



Imagen 4.3-usuarios



NOTA:

- "ep" = Eliminar Película  
- "ap" = Actualizar Película   
- "em" = Eliminar miembro  
- "am" = Actualizar miembro  
- "dp" = Delete Pelicula  
- "dm" = Delete Miembro"

En los casos de eliminar película o eliminar miembro solamente se tomará el "id" de la pelicula/miembro al cual se le aplicará el cambio

Y se mandara a llamar a submit.php con un valor en "v" de "dp" o "dm” Junto con el valor del ID de la pelicula/miembro a eliminar (*Imagen 4.4, Imagen 4.5)*

Imagen 4.4

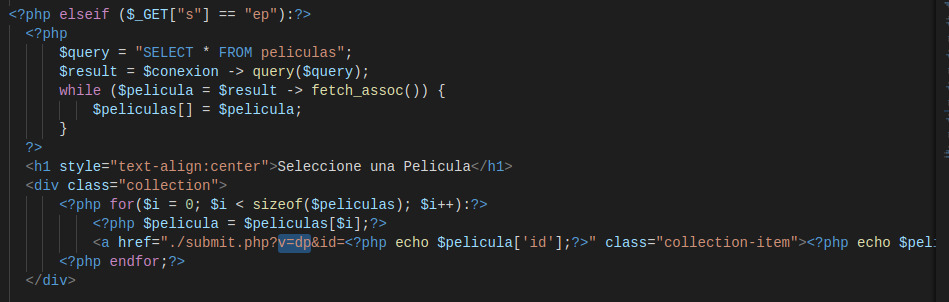
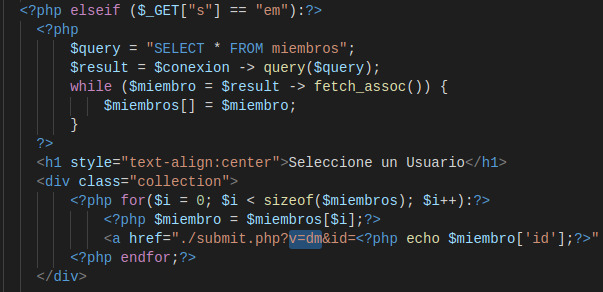
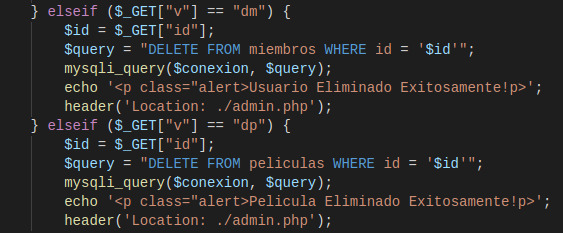


Imagen 4.5



En submit.php se tomará el valor del ID de la pelicula/miembro seleccionad@ y se usara un DELETE FROM miembros donde el id coincida. (*Imagen 4.6)*

Imagen 4.6



En caso de haber seleccionado actualizar pelicula/miembro se hará llamar a update.php (*Imagen 4.7)*

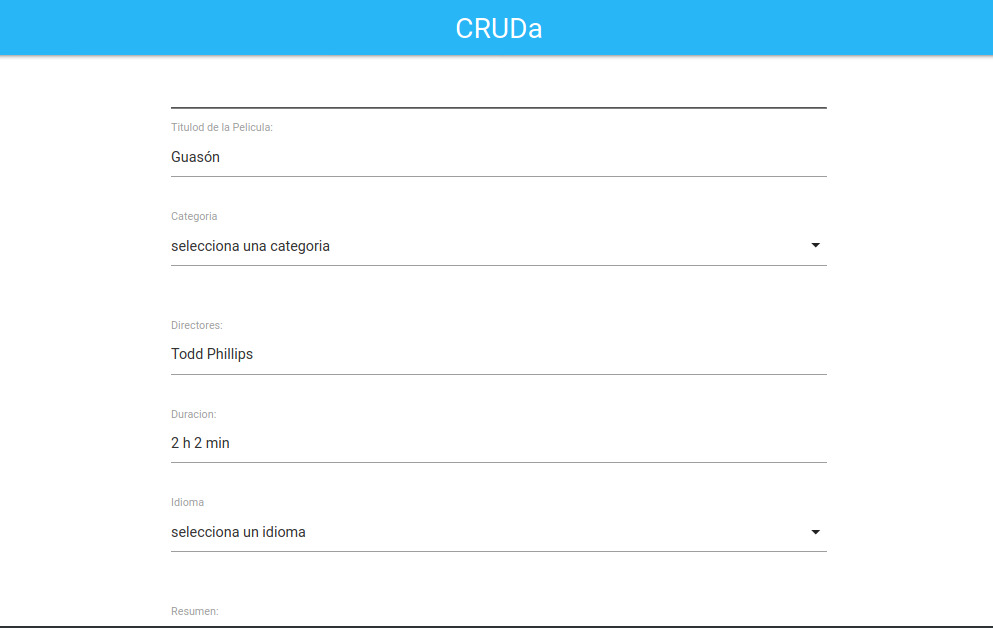
Imagen 4.7



update.php recibira el ID de la pelicula/miembro que se haya seleccionado para la operación.

update.php desplegara un formulario con algunos de los datos que contiene la base de datos de la pelicula/miembro (*Imagen 4.8)* En este formulario se ingresaran los nuevos datos de la pelicula/miembro

Imagen 4.8



Y al darle clic en el botón actualizar se hará llamar a submit.php con un valor de "v" de "up" o "um" junto con los valores de los campos por medio de un POST.

NOTA:

- "up" = update película  
- "um" = update miembro.

submit.php tomara los datos que le mandaron y hará un UPDATE de la tabla peliculas/miembros donde el id sea el mismo que el de la película/miembro seleccionada (*Imagen 4.9)*

Imagen 4.9

