



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.

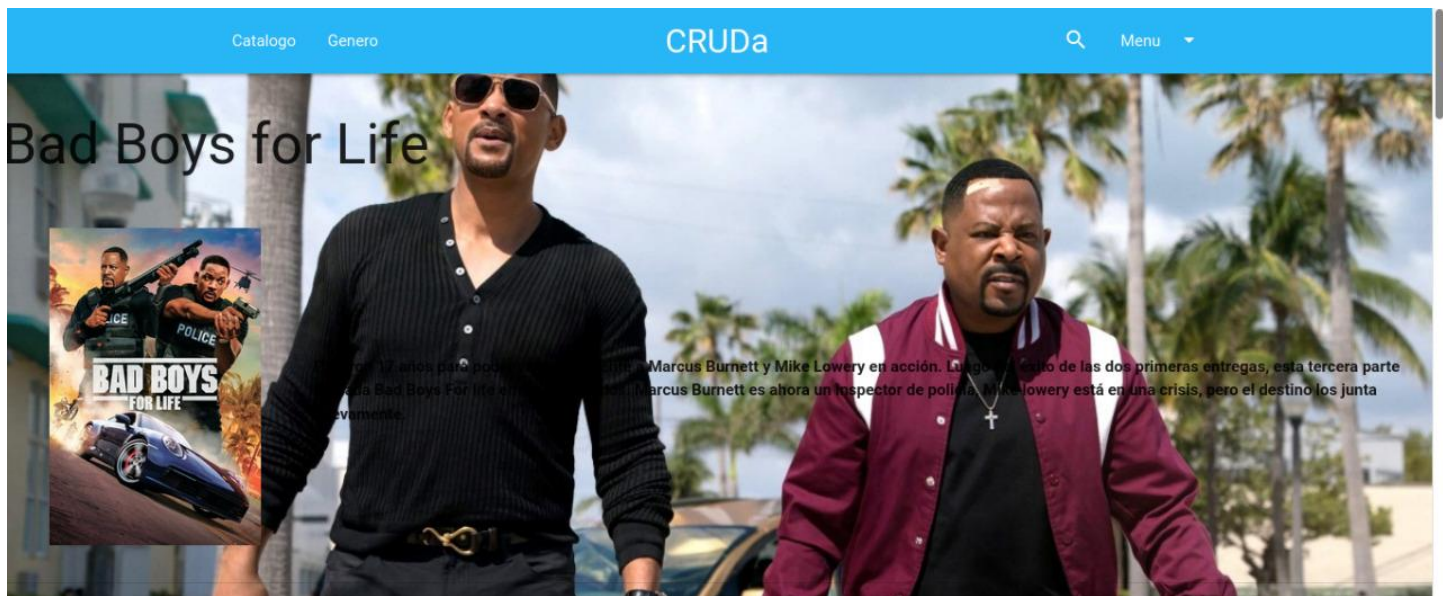
Se decidió crear un catálogo de películas. Éste deberá cumplir con los requerimientos listados a continuación para ser accesible y brindarle una experiencia grata al usuario.

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 trabajo de estilo.



CONEXIÓN-BASES DE DATOS.

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

Imagen 1.1

```
10
11 DROP SCHEMA IF EXISTS `is-website`;
12 CREATE SCHEMA IF NOT EXISTS `is-website` DEFAULT CHARACTER SET utf8 COLLATE utf8_spanish2_ci;
13 USE `is-website`;
14
```

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

Imagen 1.2

```
-- Create table structure for table `peliculas`

DROP TABLE IF EXISTS `peliculas`;

CREATE TABLE `peliculas` (
  `titulo` text CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci NOT NULL,
  `categoria` text CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci NOT NULL,
  `director` text CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci NOT NULL,
  `duracion` text CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci NOT NULL,
  `idioma` text CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci NOT NULL,
  `resumen` text CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci NOT NULL,
  `fecha_estreno` date NOT NULL,
  `id` int UNSIGNED NOT NULL PRIMARY KEY auto_increment
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_spanish2_ci;

-- Create table structure for table `miembros`

DROP TABLE IF EXISTS `miembros`;

CREATE TABLE `miembros` (
  `nombre` text CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci NOT NULL,
  `correo` text CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci NOT NULL,
  `alias` text CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci NOT NULL,
  `contraseña` text CHARACTER SET utf8mb4 COLLATE utf8mb4_spanish2_ci NOT NULL,
  `permiso` int NOT NULL DEFAULT '1',
  `id` int UNSIGNED NOT NULL PRIMARY KEY auto_increment
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_spanish2_ci;
```

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 1.3

```
<?php
$host_db = "localhost";
$user_db = "Usuario";
$pass_db = "Contraseña123!";
$db_name = "is-website";
$table_pic = "peliculas";
$table_usr = "miembros";

$conexion = new mysqli($host_db, $user_db, $pass_db, $db_name);

if ($conexion->connect_error) {
    die("Error en la conexión a la base de datos: " . $conexion->connect_error);
}
```

Imagen 1.4

```
<?php
require "conexion.php";
```

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



The image shows a web application interface. At the top is a blue navigation bar containing the text 'Catalogo', 'Genero', 'CRUDa', a magnifying glass icon, and a 'Menu' dropdown arrow. Below this is a login form. It features two input fields: the first is labeled 'Usuario' and contains the text 'mau'; the second is labeled 'Contraseña' and contains masked characters represented by dots. Below these fields are two black buttons with white text: 'INICIAR SESION' and 'REGISTRAR'.

En admin.php se revisará al usuario que se registró y si tiene un permiso de 1 solo se desplegará 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

```
<?php if (mysqli_num_rows($result) == 1):  
    $usuario = $result -> fetch_assoc();  
    $_SESSION['nombre'] = $usuario['nombre'];  
    $_SESSION['alias'] = $usuario['alias'];  
    $_SESSION['correo'] = $usuario['correo'];  
    $_SESSION['contraseña'] = $usuario['contraseña'];  
    $_SESSION['permiso'] = $usuario['permiso'];  
    $_SESSION['id'] = $usuario['id'];?>  
<?php if ($_SESSION['permiso'] == 1):?>  
    <?php  
        $query = "SELECT * FROM peliculas";  
        $result = $conexion -> query($query);  
    ?>  
    <h1 style="text-align:center">Peliculas Registradas</h1>
```

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

Imagen 2.3

Onward	Animación, Aventura	Dan Scanlon	1 h 03 min	Ingles	2020-02-20
Una Voz Silenciosa	Animación, Drama, Romance	Naoko Yamada	2 h 9 min	Japones	2017-05-05
Toy Story 4	Animación, comedia	Josh Cooley	1 h 40 min	Inglés	2019-06-21
El hombre invisible	Terror, Suspenso	Leigh Whannell	2 h 05 min	Inglés	2020-02-28
Maléfica, Dueña del mal	Fantasía, Aventura	Joachim Ronning	1 h 58 min	Inglés	2019-10-16
El Rey Leon	Animación, Familia	Jon Favreau	1 h 58 min	Inglés	2019-07-09
Aquaman	Acción, Aventura	James Wan	2 h 22 min	Inglés	2018-12-13
Aves de presa	Acción	Cathy Yan	1 h 49 min	Inglés	2020-01-29
Terminator Destino Oscuro	Acción, Ciencia Ficción	Tim Miller	2 h 08 min	Inglés	2019-10-23
Jumanji Siguiente Nivel	Acción, Comedia	Jake Kasdan	2 h 03 min	Inglés	2019-12-12

AGREGAR PELICULA

SALIR

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

Jumanji Siguiente Nivel	Acción, Comedia	Jake Kasdan	2 h 03 min	Ingles	2019-12-12	21
-------------------------	-----------------	-------------	------------	--------	------------	----

AGREGAR PELICULA

ACTUALIZAR PELICULA

ELIMINAR PELICULA

Miembros Registrados

Nombre	Correo	Nombre de Usuario	Contraseña	Nivel de Permiso	Identificador
Victor Manuel Sanchez Sanchez	vicool2@hotmail.com	vic	123456	2	1
Julio Cesar Jimenez Bolio	jcbolio@gmail.com	cesar	123456	2	2
Fernando Isaac González Medina	figonzalz@aragon.unam.com	fer	123456	1	3
Mauricio Andrés Miramontes Ramírez	marmiramonte@hotmail.com	mau	123456	2	4

AGREGAR USUARIO

ACTUALIZAR USUARIO

ELIMINAR USUARIO

SALIR

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

```
<a style="right:inherit" class="waves-effect waves-light btn" href="/create.php?v=p">Agregar Pelicula</a>
```

Imagen 3.2

```
<a style="right:inherit" class="waves-effect waves-light btn" href="/create.php?v=m">Agregar Usuario</a>
```

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

```
<?php if ($_GET["v"] == "m"):?>
<div class="row" style="margin-top:50px">
  <div class="col s6 offset-s3">
    <form action="./submit.php?v=cm" method="POST">
      <hr />
      <div class="form-group">
        <label for="nombre">Nombre Completo: </label><br>
        <input type="text" name="nombre" maxlength="255" required>
        <br /><br />
      </div>
    </form>
  </div>
</div>
```

Imagen 3.4-p = peliculas

```
<?php elseif ($_GET["v"] == "p"):?>
<div class="row" style="margin-top:50px">
  <div class="col s6 offset-s3">
    <form action="./submit.php?v=cp" method="POST">
      <hr />
      <div class="form-group">
        <label for="titulo">Titulod de la Pelicula: </label><br>
        <input type="text" name="titulo" maxlength="255" required>
        <br /><br />
      </div>
    </form>
  </div>
</div>
```

Imagen 3.5-form de pelucula

CRUDa

Titulod de la Pelicula:

Categoria

selecciona una categoria



Directores:

Duracion:

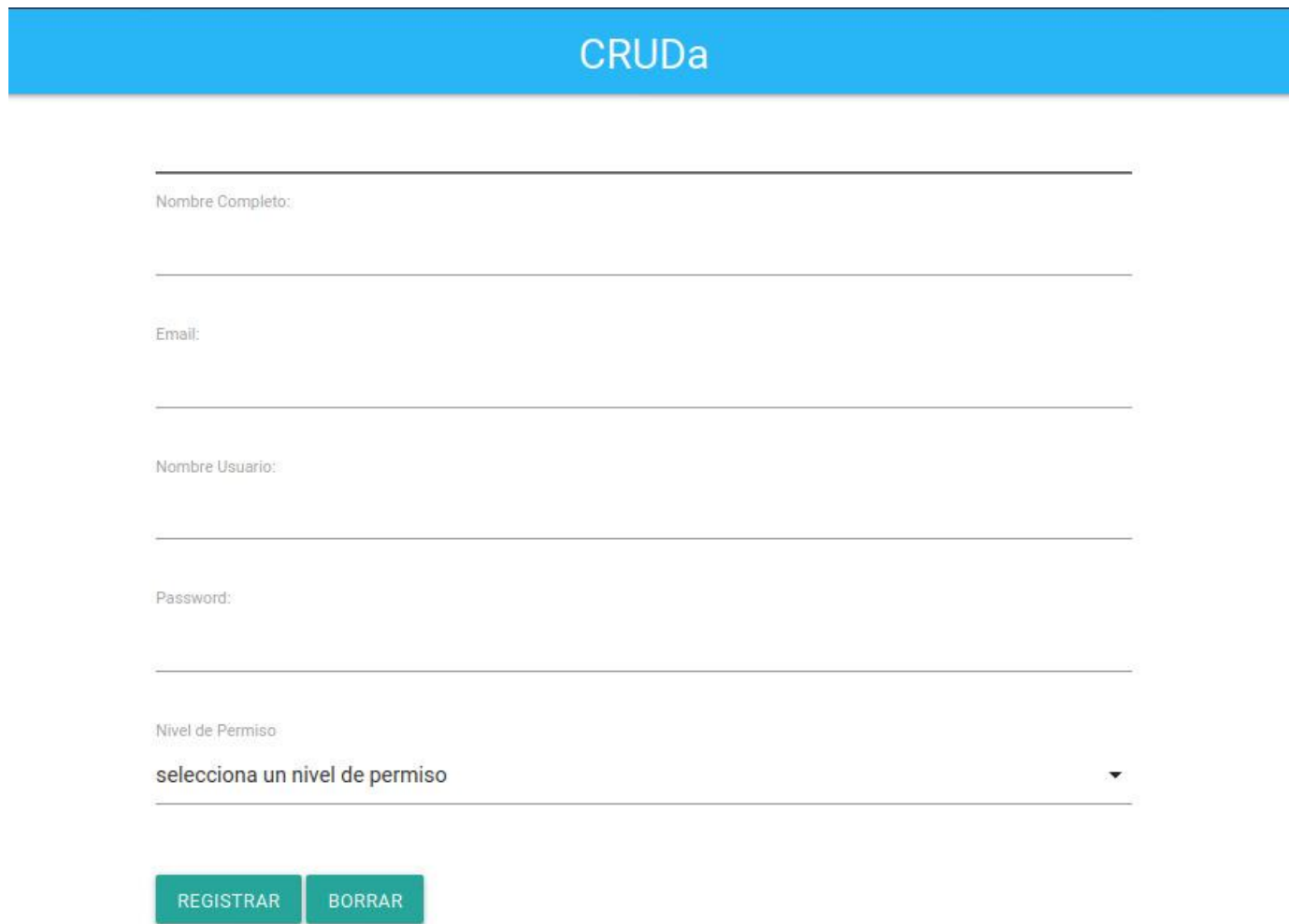
Idioma:

selecciona un idioma



Resumen:

Imagen 3.6-form de usuario.



The image shows a web form titled "CRUDa" in a blue header. The form contains several input fields with labels: "Nombre Completo:", "Email:", "Nombre Usuario:", "Password:", and "Nivel de Permiso:". The "Nivel de Permiso:" field is a dropdown menu with the text "selecciona un nivel de permiso" and a downward arrow. At the bottom of the form are two green buttons labeled "REGISTRAR" and "BORRAR".

create.php tomará los datos ingresados en el formulario (*Imágenes 3.5-form de película 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 = películas*)

Cabe aclarar que submit.php es en donde se realizarán todos los cambios a la base de datos, además se utilizará el valor de "V" para determinar qué tipo de acción se tomará 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 mandó create.php (*Imagen 3.7*)

Imagen 3.7

```
if ($_GET["v"] == "cm") {
    $buscarUsuario = "SELECT * FROM miembros WHERE alias = '$_POST[alias]'";
    $result = $conexion -> query($buscarUsuario);
    if (mysqli_num_rows($result) == 1 ) {
        echo '<p class="alert">El nombre de usuario ya fue utilizado!>';
        header('Location: ./create.php?v=m');
    } else {
        mysqli_query($conexion, "INSERT INTO miembros (
            nombre,
            correo,
            alias,
            contraseña,
            permiso)
        VALUES(
            '$_POST[nombre]',
            '$_POST[correo]',
            '$_POST[alias]',
            '$_POST[contraseña]',
            '$_POST[permiso]'
        )");
        echo '<p class="alert">Usuario Creado Exitosamente!p>';
        header('Location: ./admin.php');
    }
}
```

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

Imagen 3.8

```
} elseif ($_GET["v"] == "cp") {
    $buscarPelicula = "SELECT * FROM peliculas WHERE titulo = '$_POST[titulo]'";
    $result = $conexion -> query($buscarPelicula);
    if (mysqli_num_rows($result) == 1 ) {
        echo '<p>Ya existe una pelicula con el mismo nombre!>';
        header('Location: ./create.php?v=p');
    } else {
        mysqli_query($conexion, "INSERT INTO peliculas (
            titulo,
            categoria,
            director,
            duracion,
            idioma,
            resumen,
            fecha_estreno)
        VALUES(
            '$_POST[titulo]',
            '$_POST[categoria]',
            '$_POST[director]',
            '$_POST[duracion]',
            '$_POST[idioma]',
            '$_POST[resumen]',
            '$_POST[fecha_estreno]'
        )");
        echo '<p class="alert">Pelicula Ingresada Exitosamente!p>';
        header('Location: ./admin.php');
    }
}
```

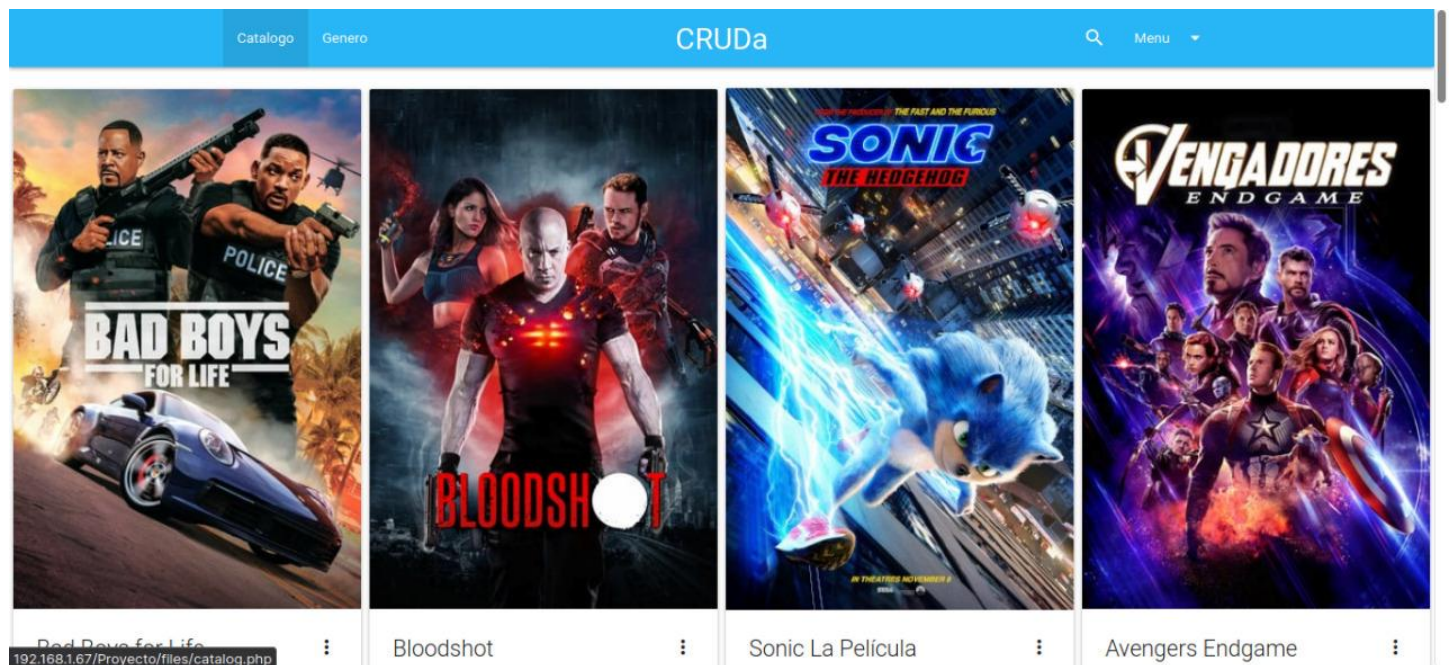
Y se dará un mensaje para hacer saber al usuario que se ha agregado exitosamente la película/miembro nuevo.

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

Jumanji Siguiente Nivel	Acción, Comedia	Jake Kasdan	2 h 03 min	Ingles	2019-12-12	21
-------------------------	-----------------	-------------	------------	--------	------------	----

AGREGAR PELICULA

ACTUALIZAR PELICULA

ELIMINAR PELICULA

Miembros Registrados

Nombre	Correo	Nombre de Usuario	Contraseña	Nivel de Permiso	Identificador
Victor Manuel Sanchez Sanchez	vicoool2@hotmail.com	vic	123456	2	1
Julio Cesar Jimenez Bolio	jcbolio@gmail.com	cesar	123456	2	2
Fernando Isaac González Medina	figonzalz@aragon.unam.com	fer	123456	1	3
Mauricio Andrés Miramontes Ramírez	marmiramonte@hotmail.com	mau	123456	2	4

AGREGAR USUARIO

ACTUALIZAR USUARIO

ELIMINAR USUARIO

SALIR

`select_item.php` nos desplegara una lista de todas(os) las(os) Películas/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

```
<a style="right:inherit" class="waves-effect waves-light btn" href="./select_item.php?s=ap">Actualizar Pelicula</a>  
<a style="right:inherit" class="waves-effect waves-light btn" href="./select_item.php?s=ep">Eliminar Pelicula</a>
```

Imagen 4.3-usuarios

```
<a style="right:inherit" class="waves-effect waves-light btn" href="./select_item.php?s=am">Actualizar Usuario</a>  
<a style="right:inherit" class="waves-effect waves-light btn" href="./select_item.php?s=em">Eliminar Usuario</a>
```

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 película/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 película/miembro a eliminar (*Imagen 4.4, Imagen 4.5*)

Imagen 4.4

```
<?php elseif ($_GET["s"] == "ep"):?>
<?php
    $query = "SELECT * FROM peliculas";
    $result = $conexion -> query($query);
    while ($pelicula = $result -> fetch_assoc()) {
        $peliculas[] = $pelicula;
    }
?>
<h1 style="text-align:center">Seleccione una Pelicula</h1>
<div class="collection">
    <?php for($i = 0; $i < sizeof($peliculas); $i++):?>
        <?php $pelicula = $peliculas[$i];?>
        <a href="./submit.php?v=dp&id=<?php echo $pelicula['id'];?>" class="collection-item"><?php echo $pelicula['titulo'];?>
    <?php endfor;?>
</div>
```

Imagen 4.5

```
<?php elseif ($_GET["s"] == "em"):?>
<?php
    $query = "SELECT * FROM miembros";
    $result = $conexion -> query($query);
    while ($miembro = $result -> fetch_assoc()) {
        $miembros[] = $miembro;
    }
?>
<h1 style="text-align:center">Seleccione un Usuario</h1>
<div class="collection">
    <?php for($i = 0; $i < sizeof($miembros); $i++):?>
        <?php $miembro = $miembros[$i];?>
        <a href="./submit.php?v=dm&id=<?php echo $miembro['id'];?>" class="collection-item"><?php echo $miembro['nombre'];?>
    <?php endfor;?>
</div>
```

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

Imagen 4.6

```
} elseif ($_GET["v"] == "dm") {
    $id = $_GET["id"];
    $query = "DELETE FROM miembros WHERE id = '$id'";
    mysqli_query($conexion, $query);
    echo '<p class="alert">Usuario Eliminado Exitosamente!p>';
    header('Location: ../admin.php');
} elseif ($_GET["v"] == "dp") {
    $id = $_GET["id"];
    $query = "DELETE FROM peliculas WHERE id = '$id'";
    mysqli_query($conexion, $query);
    echo '<p class="alert">Pelicula Eliminado Exitosamente!p>';
    header('Location: ../admin.php');
```

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

Imagen 4.7

```
<?php if ($_GET["s"] == "ap"):?>
    <?php
        $query = "SELECT * FROM peliculas";
        $result = $conexion -> query($query);
        while ($pelicula = $result -> fetch_assoc()) {
            $peliculas[] = $pelicula;
        }
    ?>
    <h1 style="text-align:center">Seleccione una Pelicula</h1>
    <div class="collection">
        <?php for($i = 0; $i < sizeof($peliculas); $i++):?>
            <?php $pelicula = $peliculas[$i];?>
            <a href="../update.php?u=p&id=<?php echo $pelicula['id'];?>"
            <?php endfor;?>
        </div>

    <?php elseif ($_GET["s"] == "am"):?>
        <?php
            $query = "SELECT * FROM miembros";
            $result = $conexion -> query($query);
            while ($miembro = $result -> fetch_assoc()) {
                $miembros[] = $miembro;
            }
        ?>
        <h1 style="text-align:center">Seleccione un Usuario</h1>
        <div class="collection">
            <?php for($i = 0; $i < sizeof($miembros); $i++):?>
                <?php $miembro = $miembros[$i];?>
                <a href="../update.php?u=m&id=<?php echo $miembro['id'];?>"
                <?php endfor;?>
            </div>
        </div>
```

update.php recibirá el ID de la película/miembro que se haya seleccionado para la operación.

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

Imagen 4.8

CRUDa

Título de la Película:

Guasón

Categoría

selecciona una categoría ▼

Directores:

Todd Phillips

Duración:

2 h 2 min

Idioma

selecciona un idioma ▼

Resumen:

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 tomará los datos que le mandaron y hará un UPDATE de la tabla películas/miembros donde el id sea el mismo que el de la película/miembro seleccionada (*Imagen 4.9*)

Imagen 4.9

```
} elseif ($_GET["v"] == "um") {
    $id = $_GET["id"];
    $query = "UPDATE miembros SET
        nombre      = '$_POST[nombre]',
        correo      = '$_POST[correo]',
        alias       = '$_POST[alias]',
        contraseña  = '$_POST[contraseña]',
        permiso     = '$_POST[permiso]'
    WHERE id = '$id'";
    mysqli_query($conexion, $query);
    echo '<p class="alert">Usuario Actualizado Exitosamente!p>';
    header('Location: ./admin.php');
} elseif ($_GET["v"] == "up") {
    $id = $_GET["id"];
    $query = "UPDATE peliculas SET
        titulo      = '$_POST[titulo]',
        categoria   = '$_POST[categoria]',
        director    = '$_POST[director]',
        duracion    = '$_POST[duracion]',
        idioma      = '$_POST[idioma]',
        resumen     = '$_POST[resumen]',
        fecha_estreno = '$_POST[fecha_estreno]'
    WHERE id = '$id'";
    mysqli_query($conexion, $query);
    echo '<p class="alert">Pelicula Actualizada Exitosamente!p>';
    header('Location: ./admin.php');
```