

# Proyecto Final



**Cervantes Hernández Mariano**

**Montaño Garcia Anette**

**Hernández Morales Julio Alberto**

Materia: Programación web

Profesor: Diana Xochitl Ruano Vargas



## CONTENIDO

Área del proyecto .....	3
Descripción del proyecto .....	3
Planteamiento del problema .....	4
Especificación de requisitos.....	4
FrontEnd.....	6
Framework (Bootstrap 3.4).....	6
Wireframes.....	7
MÓVIL .....	7
Pc.....	9
Mockups.....	13
Móvil .....	13
Pc.....	15
Mapa de navegación .....	19
Capturas del sitio .....	20
Modelo relacional.....	24
BackEnd .....	25
Framework (Codeigniter) .....	25
Controladores.....	26
Vistas .....	28
Modelos .....	41
Link del repositorio web .....	42
Link del hosting .....	42

## ÁREA DEL PROYECTO

Entretenimiento

## DESCRIPCIÓN DEL PROYECTO

El proyecto consiste en el desarrollo web de un sitio especializado en la distribución de contenidos audiovisuales a través de una plataforma en línea que también se conoce como servicio de video bajo demanda por streaming. El objetivo de la web a desarrollar es brindar servicio de streaming con una amplia variedad en contenido audiovisual como series, películas, documentales etc. La web es accesible desde cualquier navegador o dispositivo, y permitirá la reproducción de su contenido de manera gratuita y legal para el público en general. El sitio web debe ser seguro por lo que deberá contar con ciertos estándares y lineamientos de seguridad para permitir su acceso al contenido y los tipos suscripción con el fin de evitar la implantación de cualquier tipo de malware en los dispositivos y el uso inadecuado de información personal. Para poder acceder al servicio el usuario debe registrarse en el sitio web y obtendrá un mes gratis de servicio en cualquiera de las modalidades de suscripción con la única condición de renovar el plan seleccionado al terminar dicho mes. Las modalidades de suscripción son las siguientes:

1. OneFreemium: Tendrá acceso a la mayor parte del contenido de la plataforma, pero con la restricción de visualización de anuncios ante cada reproducción y la restricción de contenido exclusivo de la plataforma, películas galardonadas, entre otros.
2. Basic: Tendrá acceso a todo el contenido disponible dentro de la plataforma, pero con la única limitación de visualización de anuncios cortos en la reproducción de cada contenido.
3. Premium: podrá acceder a todo el contenido disponible en la plataforma sin limitaciones del contenido y en este caso se restringen los anuncios de forma definitiva.

Cabe mencionar que los 2 planes iniciales podrán contar con la opción de ingresar de forma anónima con credenciales temporales que defina el cliente, esto con el fin de promover un buen uso de la información personal de los usuarios a las marcas anunciantes dentro del sitio, con el fin de evitar fraudes de información y la recepción de spam y/o elementos no deseados por el usuario.

## PLANTEAMIENTO DEL PROBLEMA

El problema planteado es diseñar y desarrollar un sitio web para ver series y películas de manera gratuita y abierto a todo público, donde el usuario podrá ver películas de manera segura y de fácil acceso, a grandes rasgos, debería ser:

- Mostrar un catálogo general del contenido audiovisual en el sitio.
- Permitir registro de usuarios.
- Reproducir películas o series mediante streaming.
- Mostrar información específica de los contenidos en la plataforma.
- Generar recomendaciones en base al uso y búsqueda de contenido de la página.
- Permitir gestionar información de pago y facturación.
- Poder recomendar títulos de contenido a través de redes sociales o mediante enlaces cortos.

4

## ESPECIFICACIÓN DE REQUISITOS

### ➤ Usuario anónimo:

- Consultar contenido: Se muestra una serie de películas limitadas y tráiler de series las cuales puede visualizar el usuario sin necesidad de ingresar con una cuenta.
- Registro de usuario: Se muestran una serie de cajas de texto para que el usuario ingrese sus datos personales y así poder ingresar de forma normal al sitio web en caso que así se desee.
- Inicio de sesión: Se muestra un par de cajas de texto en las cuales ingresa su correo y contraseña registradas para acceder al contenido de la página en caso que así se desee.

### ➤ Usuario registrado: Usuario común / Usuario premier.

- Inicio de sesión: Se muestra un par de cajas de texto en las cuales ingresa su correo y contraseña registradas para acceder al contenido de la página.
- Cerrar sesión: Se muestra un botón para cerrar sesión y dirigirse a la vista del usuario anónima.
- Búsqueda: Se muestra una barra de búsqueda para que el usuario ingrese el nombre o título de alguna serie o película la cual quiera visualizar, y cuando presiona "Enter" mostrará la película o serie que ésta buscando, en caso de no encontrar el elemento deseado, mostrará más contenidos relacionados con el título ingresado.

- Lo último añadido: Se muestra un repertorio de películas y series que se han agregado últimamente en la página web.
- Estrenos: Se muestra un apartado de todos los estrenos recientes los cuales ya están disponibles en el sitio web.
- Catálogo: Se muestra el catálogo de todas las películas y series de la página web, al seleccionar esta parte mostrará los elementos clasificados en diferentes categorías.
- Inicio: Se muestra un botón de inicio, el cual al ser presionado nos mandará a la página principal del sitio web.
- My profile: Se muestra un botón “My profile” en el que se podrá visualizar los datos personales ingresados al momento de registrarse al igual que podrá hacer la modificación de alguno de ellos.
- Anuncio: Al iniciar una reproducción de contenido se muestra una tarjeta de anuncio o un breve bumper publicitario o masthead, y hará un conteo de tiempo mínimo para mostrar un botón que permita saltar el anuncio y continuar en la reproducción.

➤ Usuario premium:

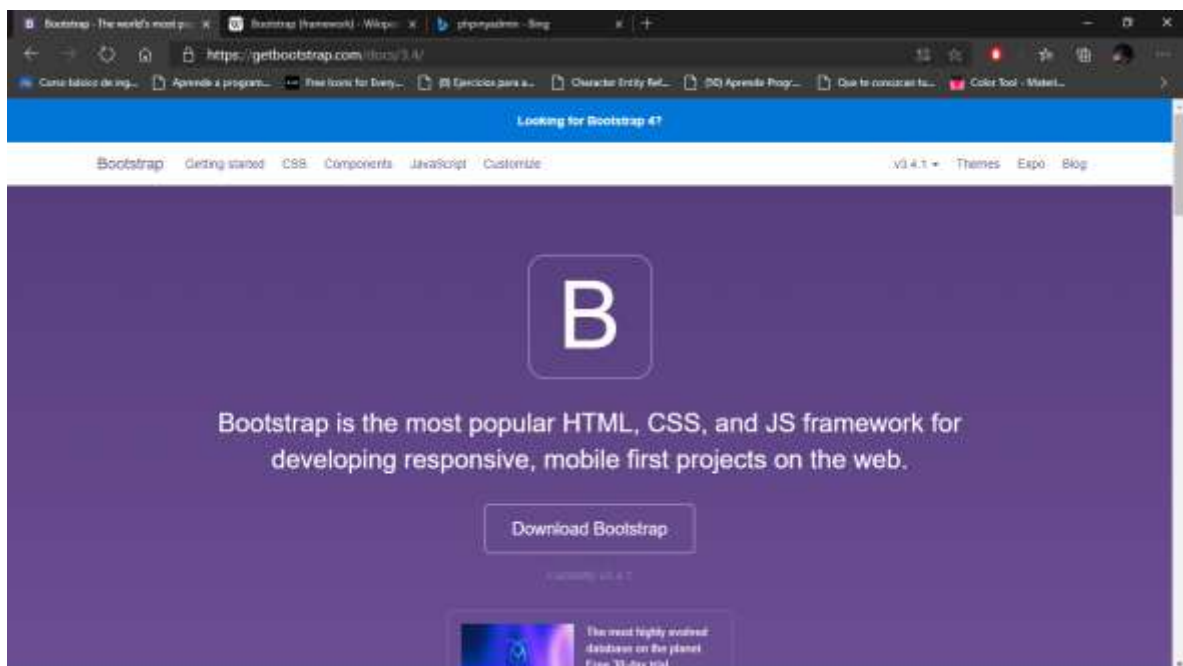
- Información sobre pago y facturación: Se muestra un botón de “Facturación y pago” el cual al ser seleccionando lo dirigirá a una página donde mostrará el estado de la suscripción e información acerca de su forma de pago.
- Nuevo contenido a futuro: Se muestra un apartado en el cual se podrá visualizar títulos futuros a estrenarse en el sitio, y además en cada uno de ellos tendrá un botón que permitirá activar una notificación cuando dicho contenido esté disponible.
- Reproducción offline: Habrá un botón de descarga situado en las secciones de reproducción, o descripción del título, que permitirá la reproducción cuando el usuario se encuentre fuera de alcance de internet.

## FRONTEND

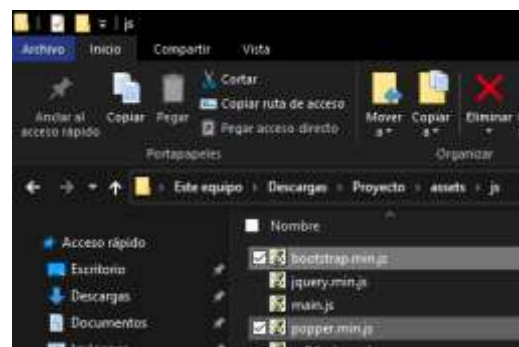
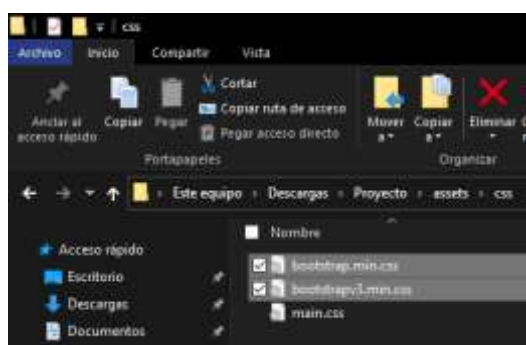
### FRAMEWORK (BOOTSTRAP 3.4)

Bootstrap es un framework de código abierto dedicado al diseño de sitios y aplicaciones web, totalmente enfocado al desarrollo FrontEnd. Consiste en una serie de hojas de estilo que implementan la variedad de componentes de la herramienta, seleccionando los componentes que se desean usar en el proyecto (botones, tamaños, colores etc.). La versión seleccionada se puede descargar desde el repositorio propio de GitHub o bien en la página oficial del Framework.

6



Las hojas de estilo y los del framework fueron insertadas en la ubicación “assets” de la carpeta raíz del proyecto, y al mismo tiempo son invocados a través de las etiquetas correspondientes en la sintaxis HTML de la página principal dentro de la sección <head>.







Gotham

Sinopsis

El detective James Gordon se desmuestra por el mundo peligrosamente corrupto de Gotham City, mientras que un joven Bruce Wayne encuentra su camino para convertirse en un héroe.

👍 138 👎 2



Gotham

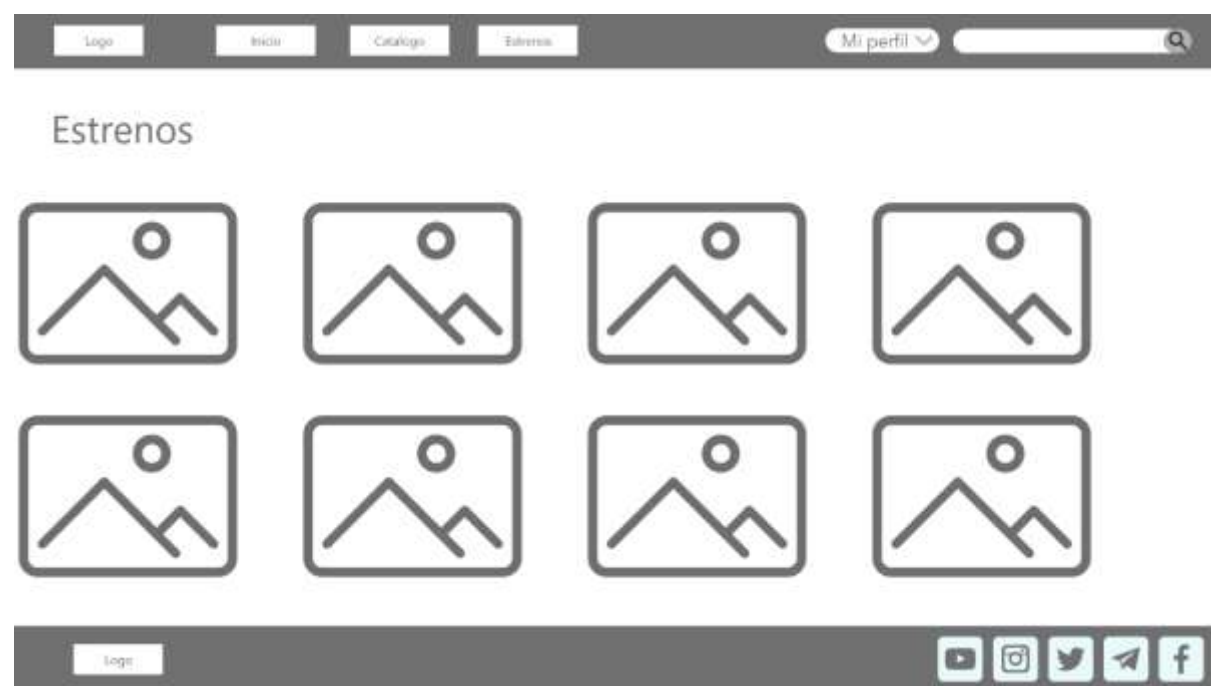
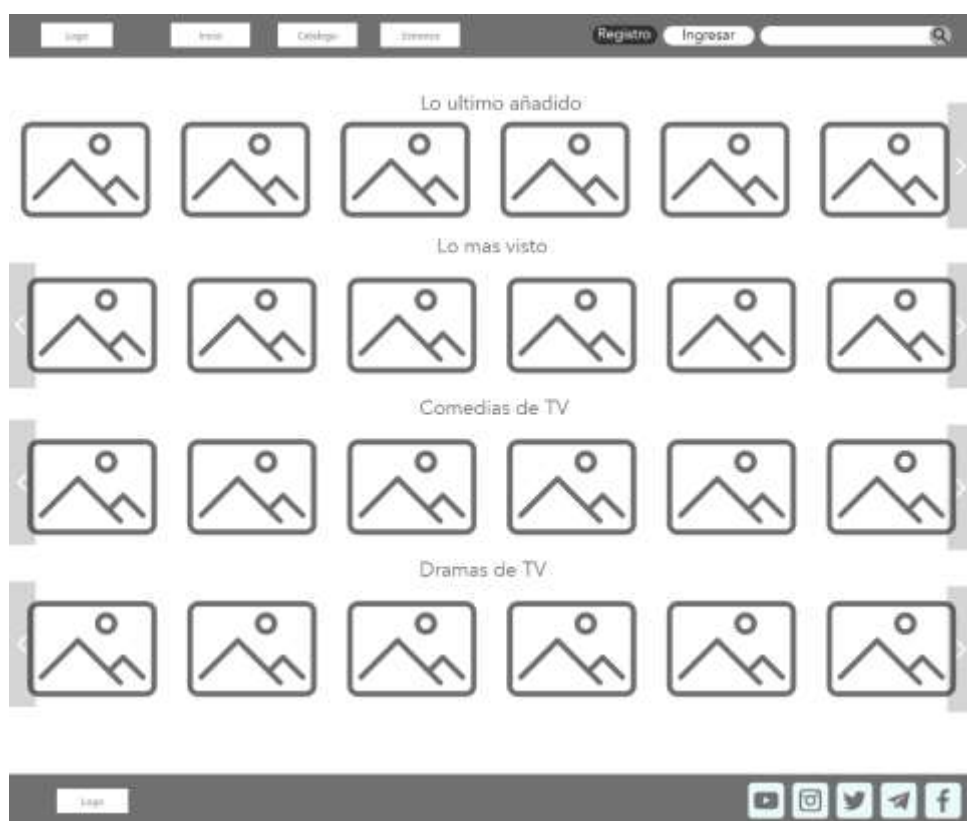
Sinopsis

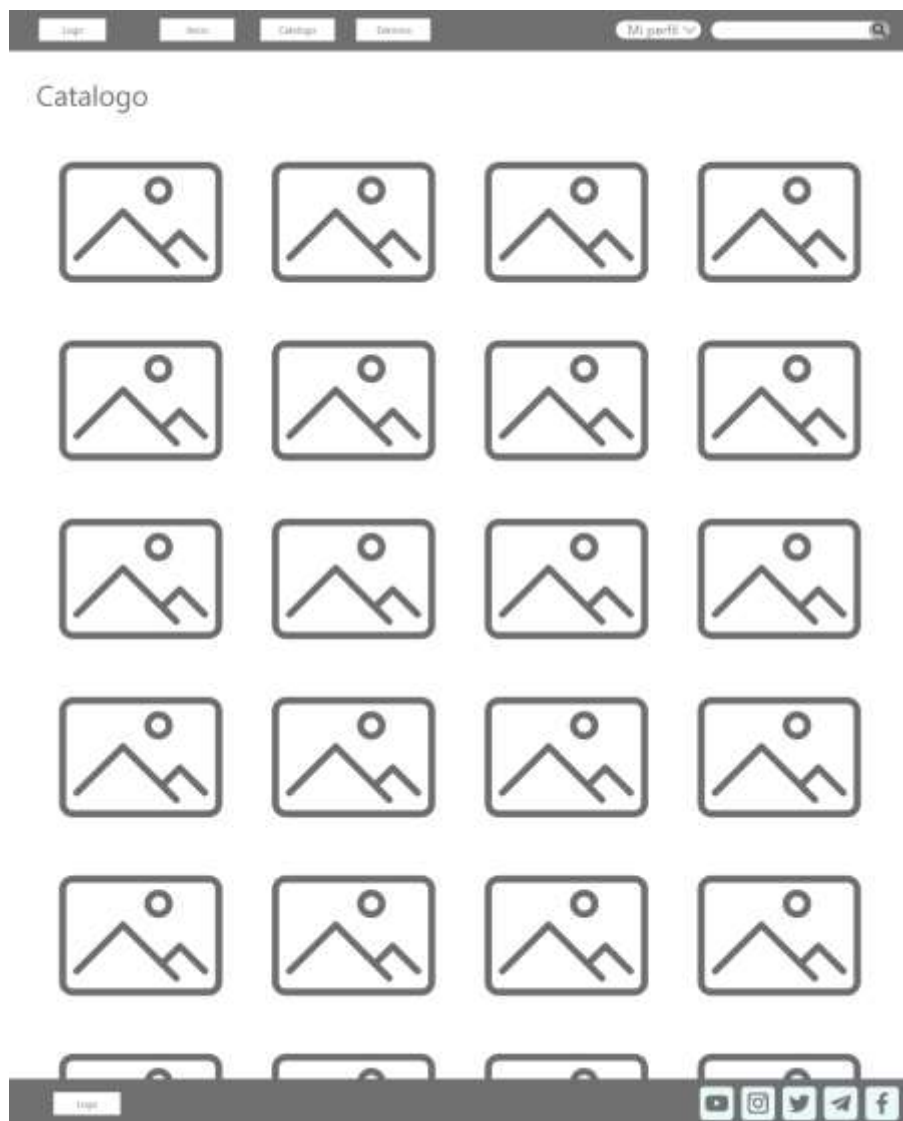
El detective James Gordon se desmuestra por el mundo peligrosamente corrupto de Gotham City, mientras que un joven Bruce Wayne encuentra su camino para convertirse en un héroe.

👍 138 👎 2

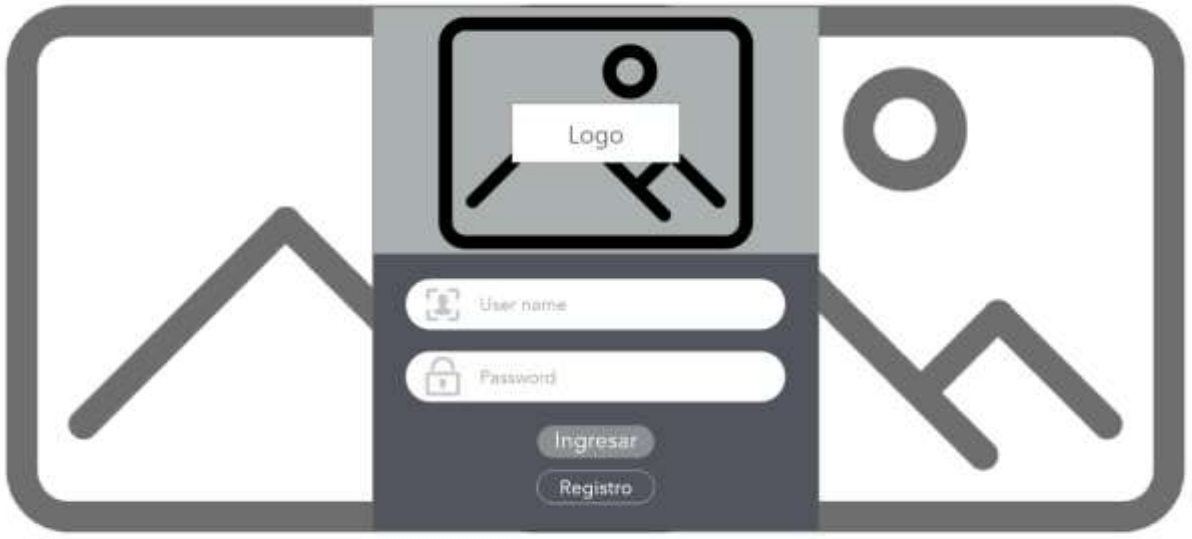
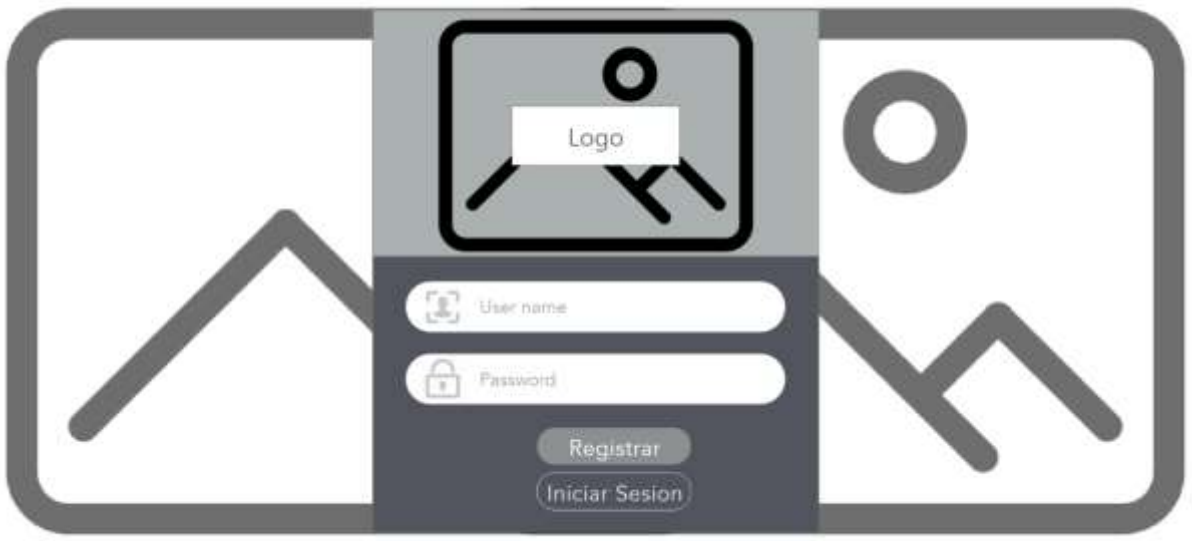






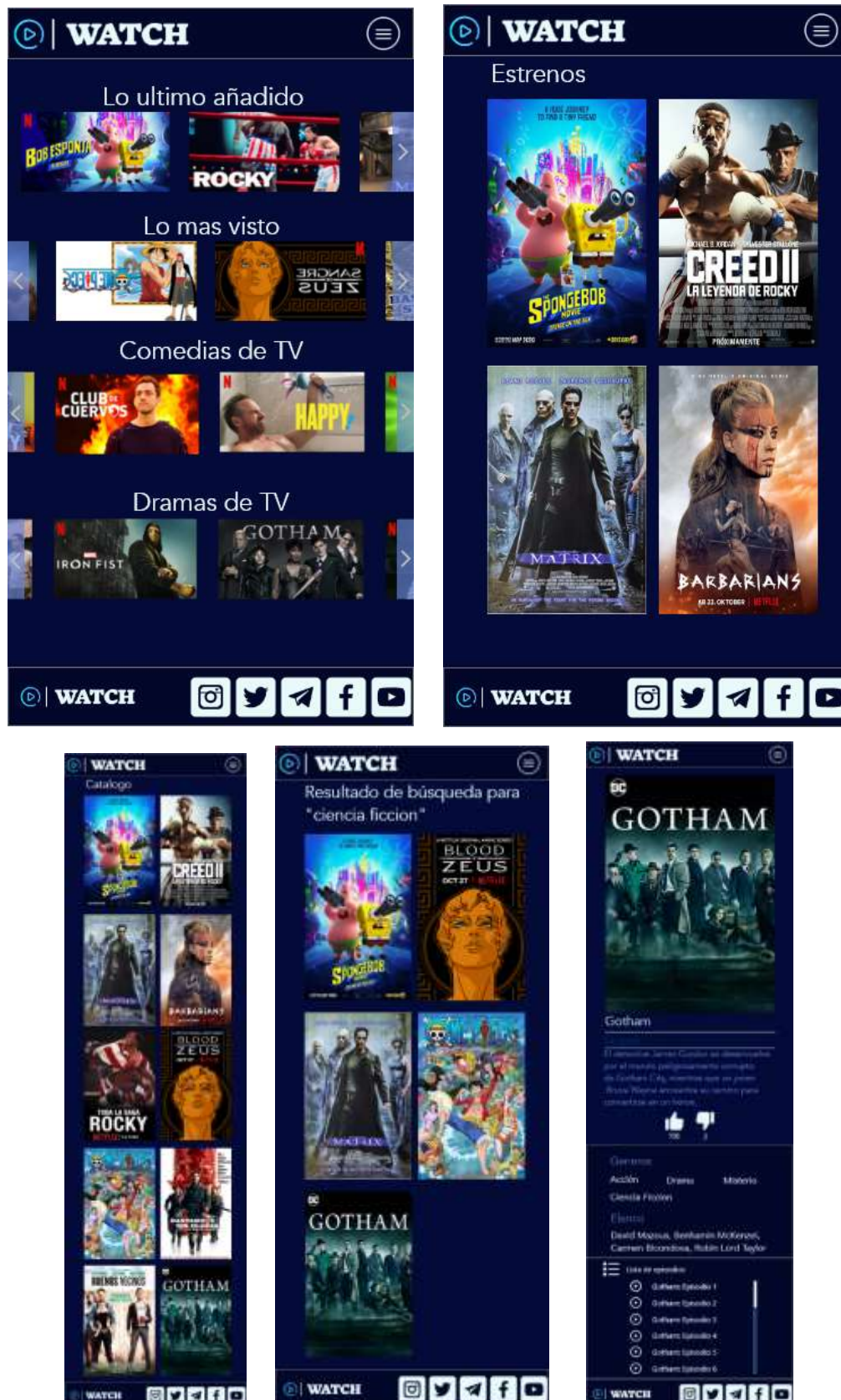




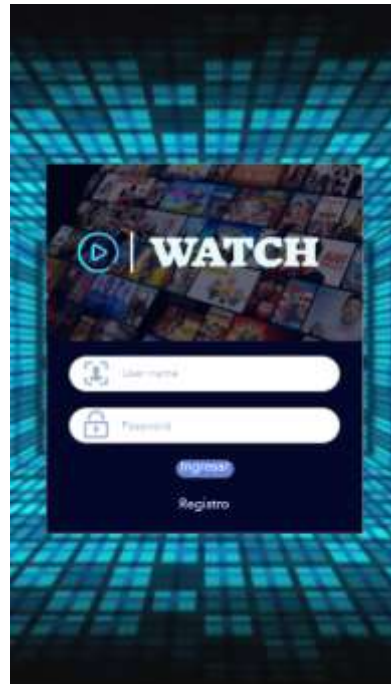


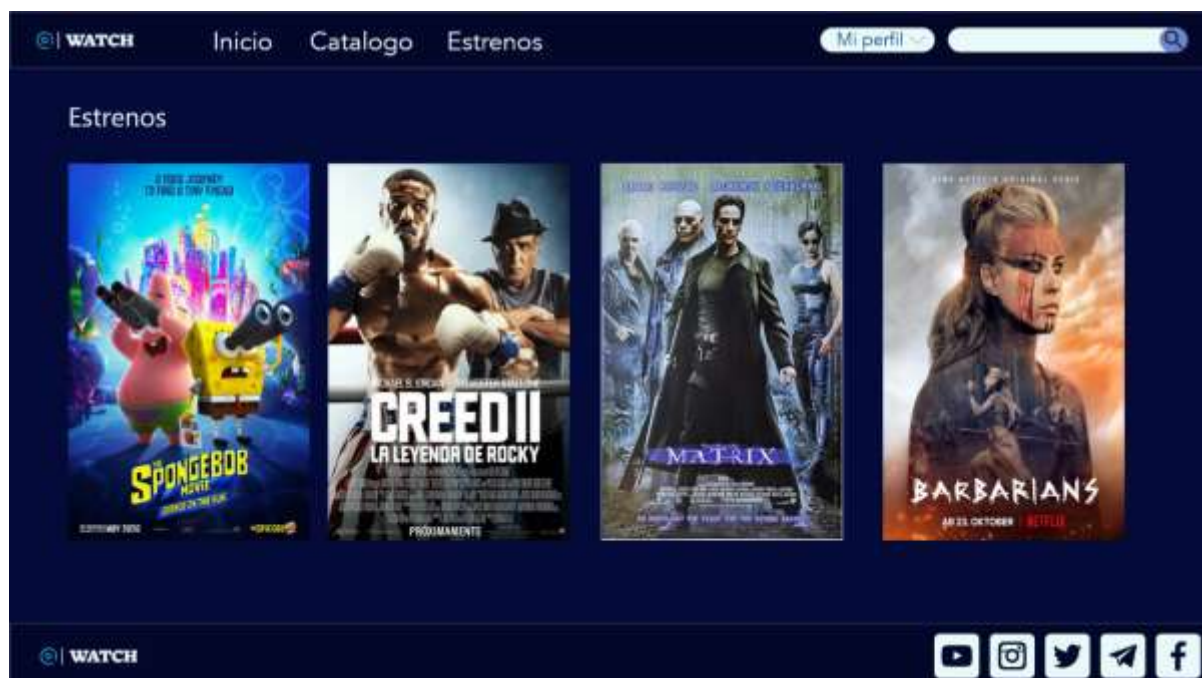
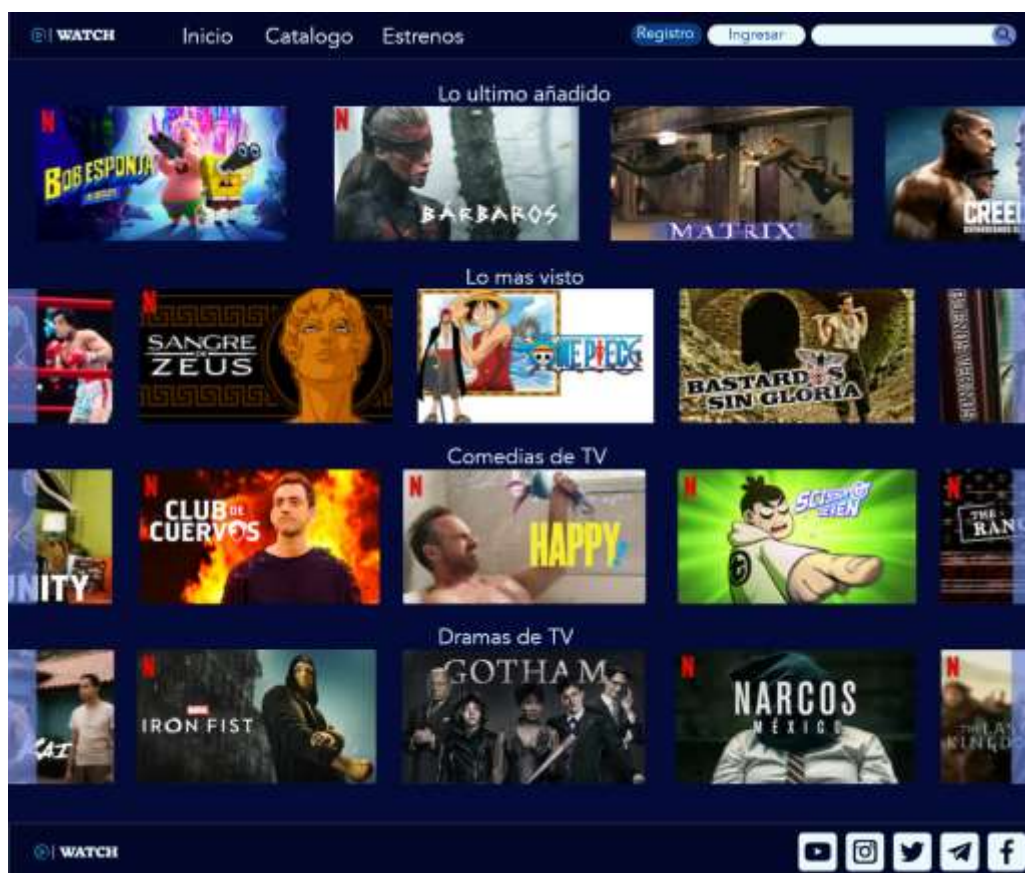
## MOCKUPS

### MÓVIL

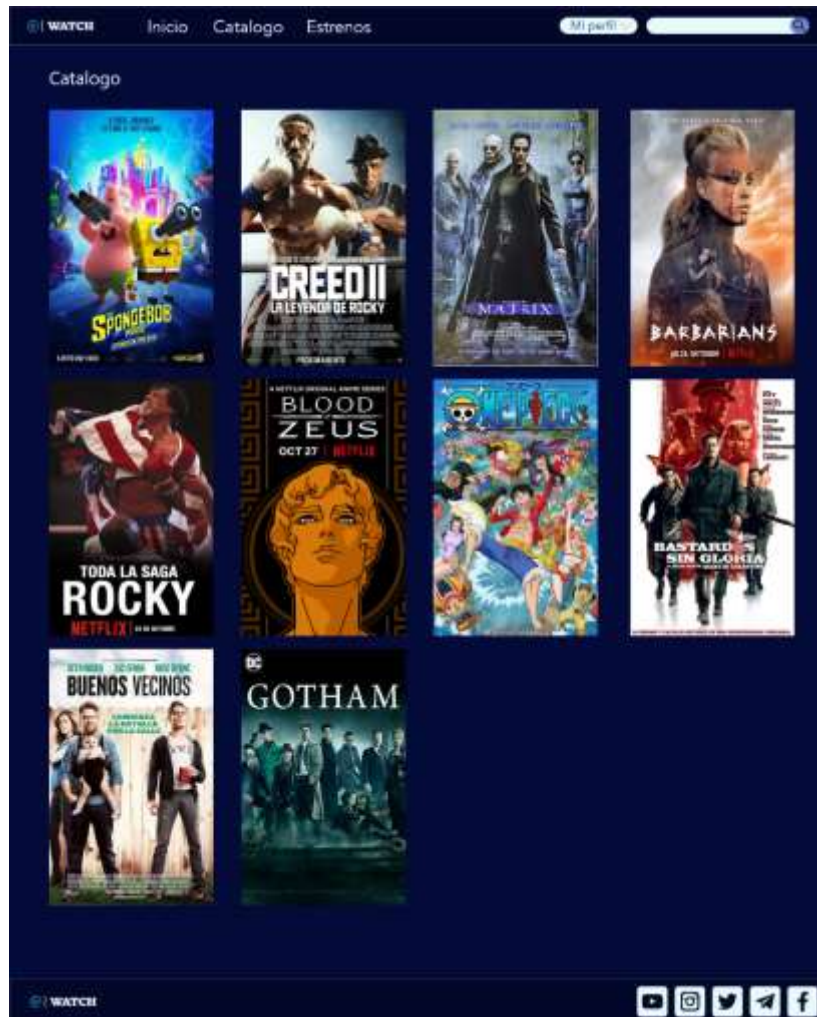
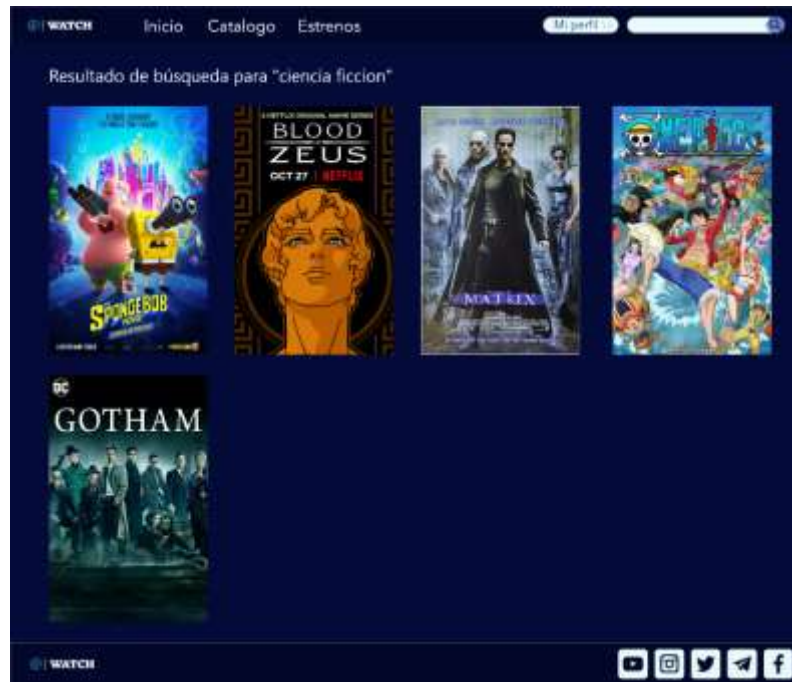










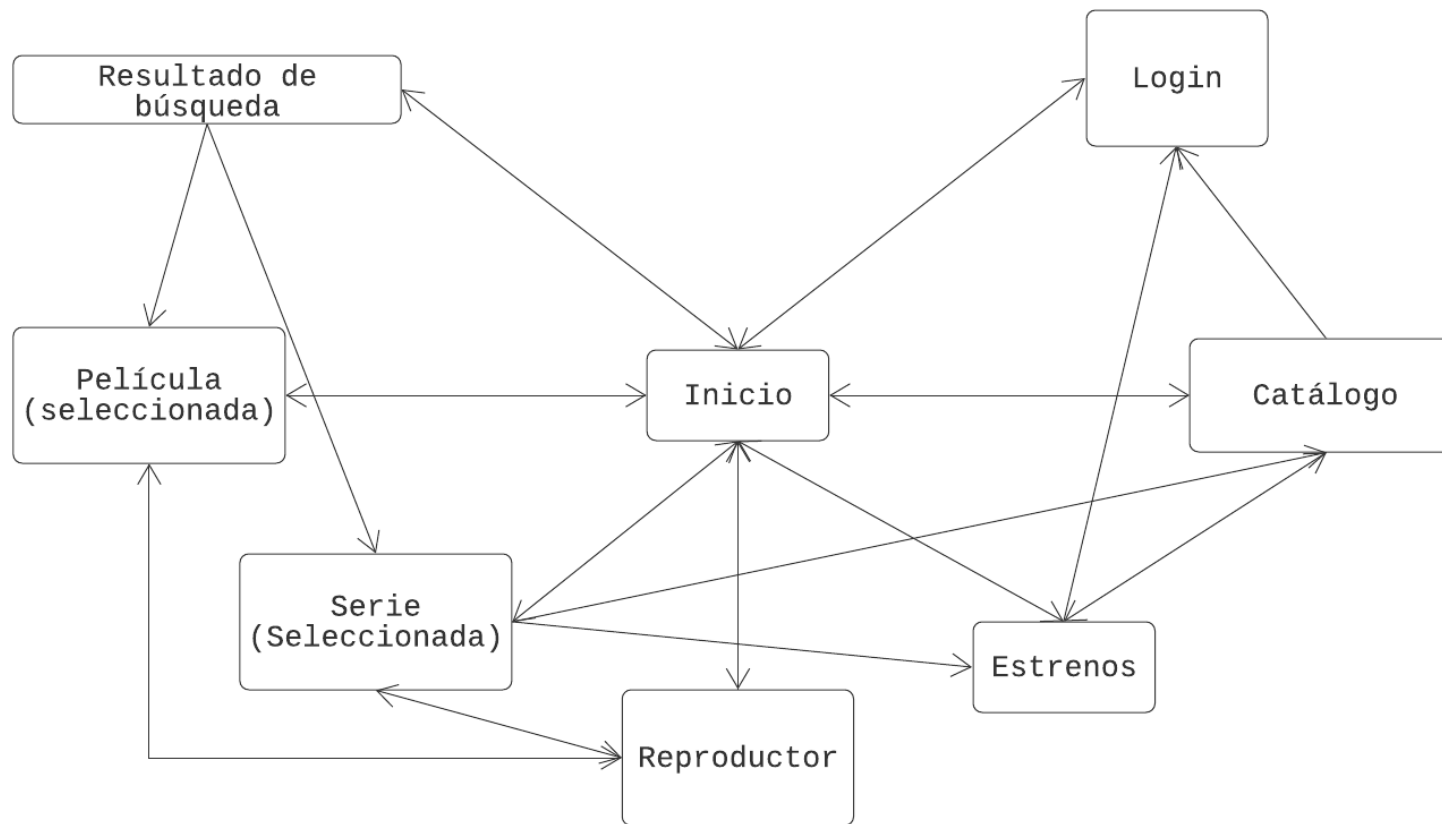






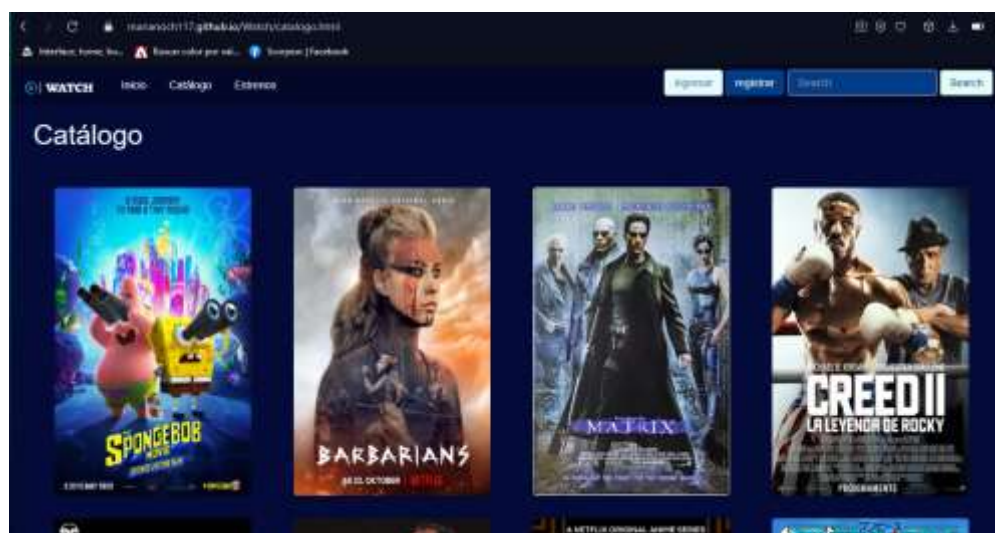
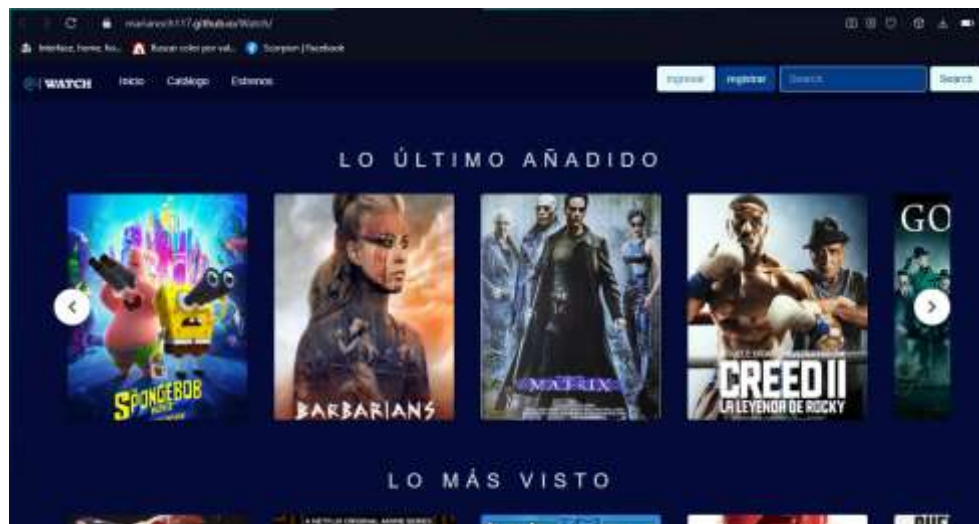


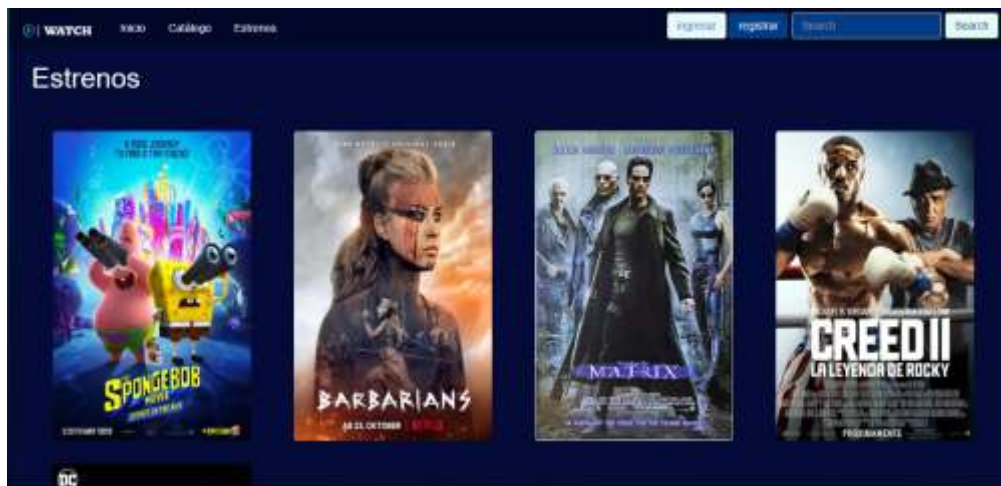
## Mapa de watch



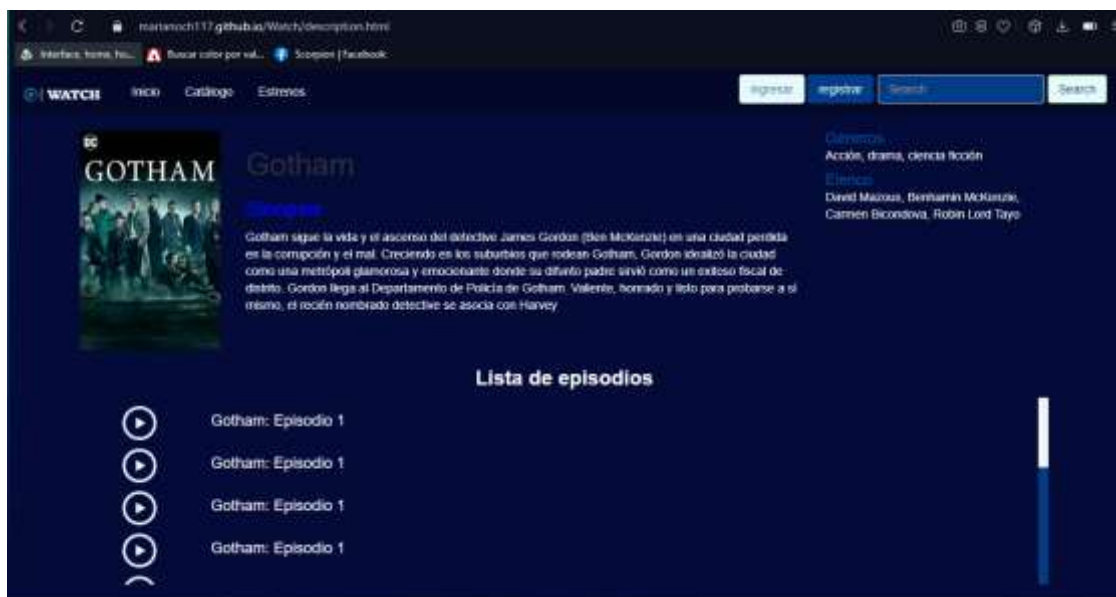


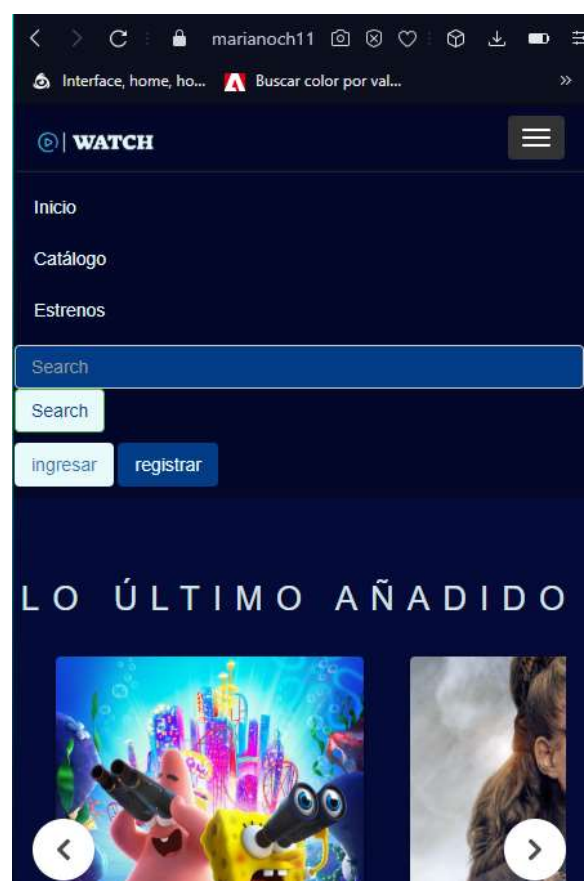
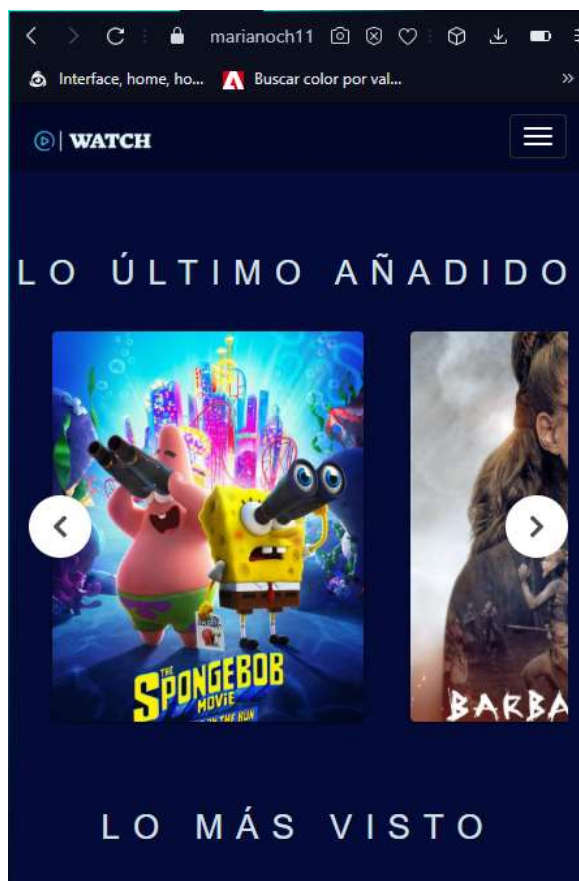
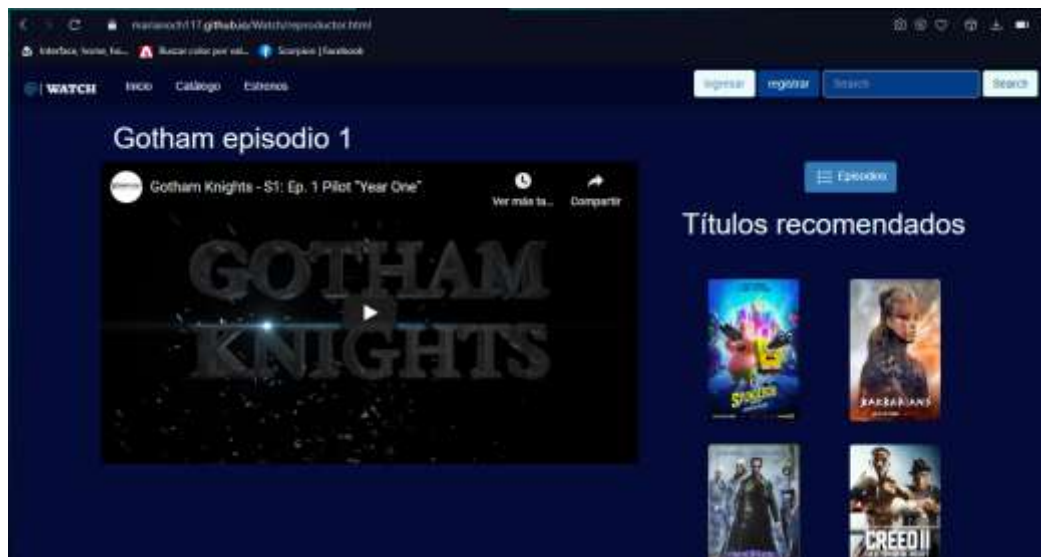
## CAPTURAS DEL SITIO



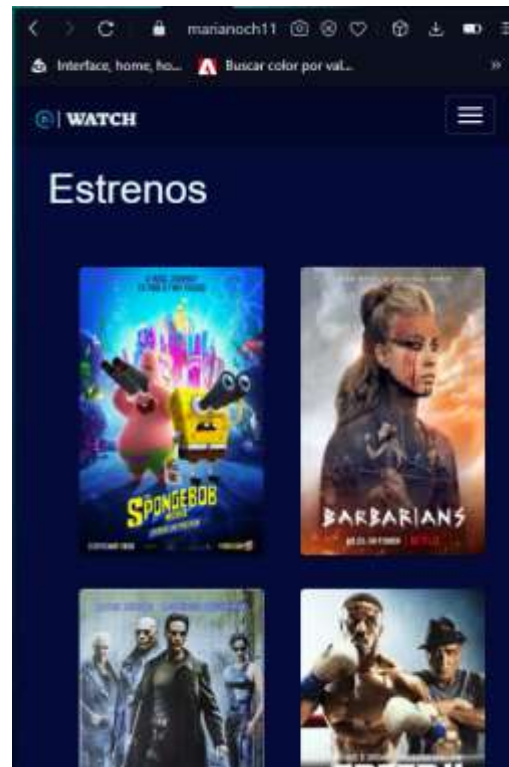
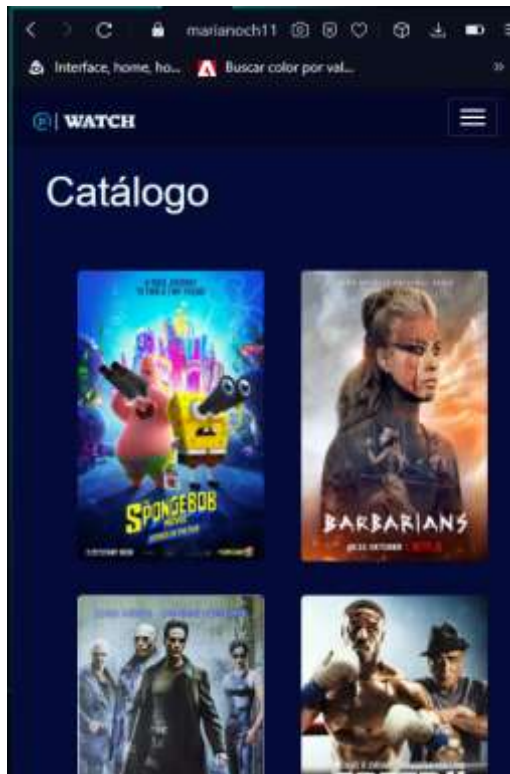


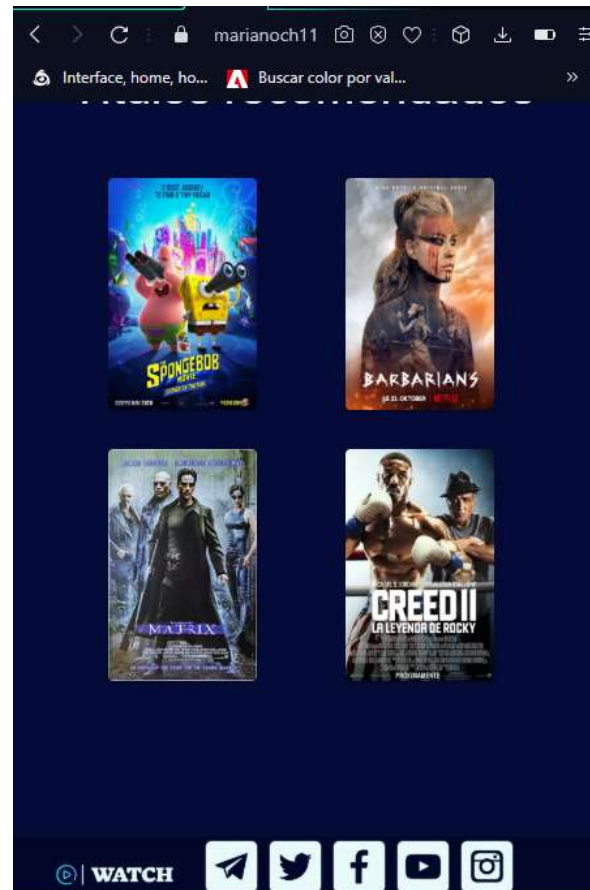
21







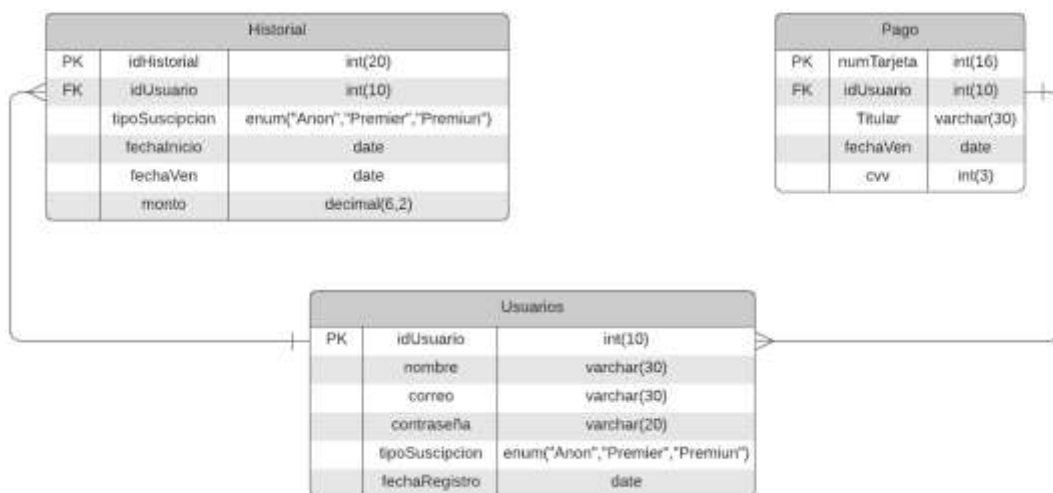




## MODELO RELACIONAL

### Modelo relacional-Wach

December 1, 2019

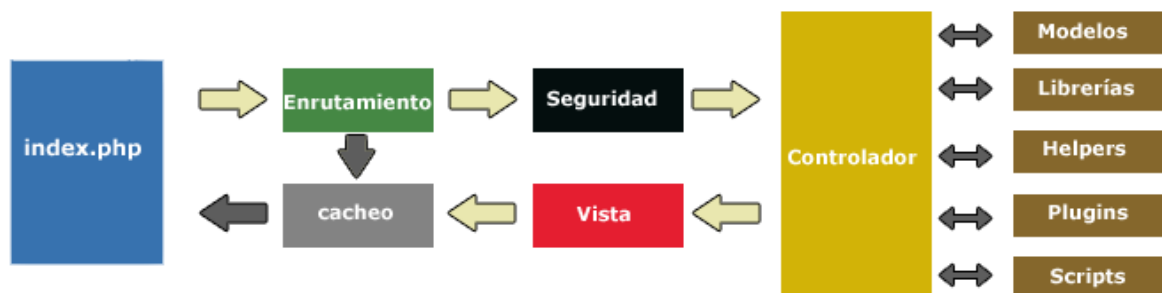




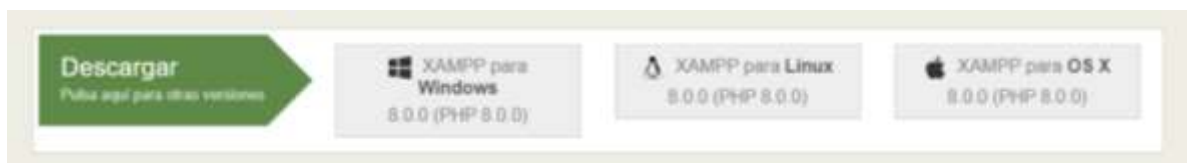
## BACKEND

### FRAMEWORK (CODEIGNITER)

CodeIgniter es un framework para aplicaciones y sitios web de código abierto que permite aprovechar el máximo rendimiento y funcionalidad con una longitud de código altamente reducida, dentro de la creación cualquier aplicación web bajo PHP. Esta herramienta se basa principalmente en el proceso de desarrollo llamado Modelo Vista-Controlador (MVC), que es un estándar de programación, utilizado tanto para hacer sitios web como programas tradicionales.



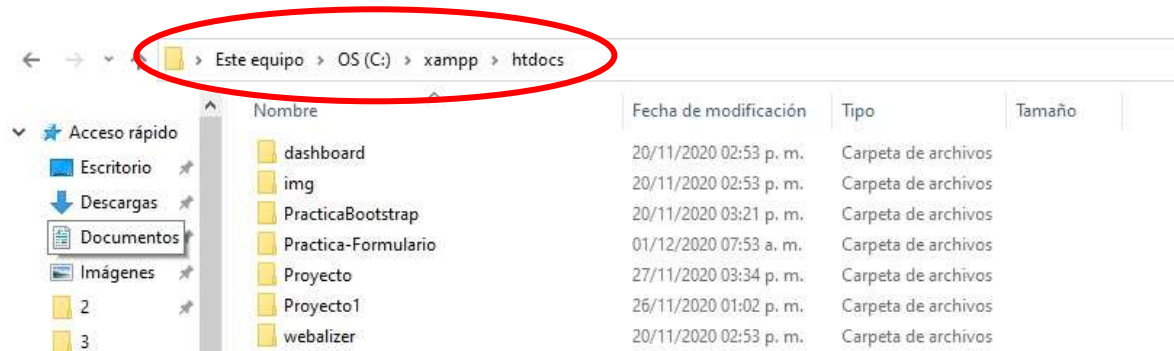
Este framework se puede descargar de manera gratuita desde su página oficial y a diferencia de Bootstrap en donde solo debemos invocar hojas de estilo dentro del código HTML, en CodeIgniter se debe disponer de un servidor PHP 4 o PHP 5 para insertar las carpetas correspondientes del framework y sean ejecutadas por el servidor que tengamos.



En este caso se usó XAMPP, que es una distribución de Apache completamente gratuita y fácil de instalar que contiene MariaDB, PHP y Perl y esta disponible para cualquier sistema operativo.



Debe seleccionar la siguiente ubicación del servidor para poder instalar el framework y ejecutar el sitio web realizado.



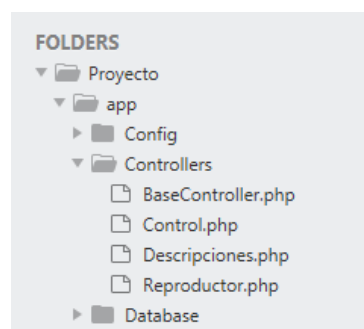
CodeIgniter varias de las más habituales bases de datos en el desarrollo de webs: MySQL (4.1 o posterior), MySQLi, MS SQL, Postgress, Oracle, SQLite, y acceso a cualquier base de datos en entornos Windows por ODBC.

## CONTROLADORES

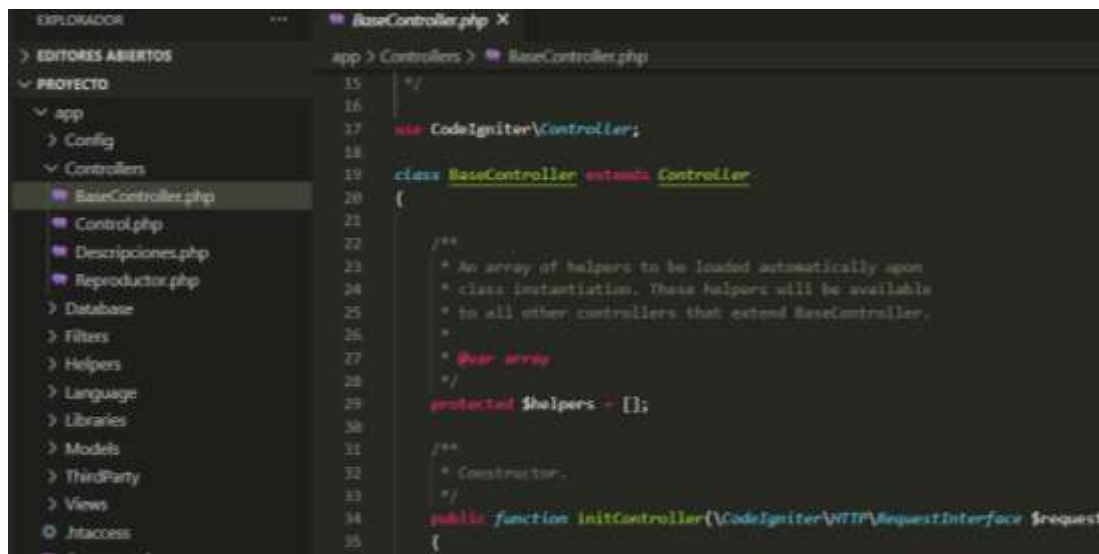
El Controlador acepta solicitudes que hace el usuario a través del navegador, contacta al Modelo correspondiente para cualquier dato que pueda necesitar, y luego toma la Vista adecuada para mostrarle esos datos a el usuario.

El módulo de enrutamiento (routing) se encarga esencialmente de que cualquier URL que se solicite al servidor se ejecute en el controlador adecuado. La URL se analiza y los datos se procesan y aseguran antes de enviarlos al controlador adecuado, en el que simplemente tendremos que codificar sus diversos métodos. El controlador adecuado realiza el procesamiento de la solicitud. CodeIgniter decide el controlador que debe procesar la solicitud en función de la URL solicitada.

Los ficheros correspondientes a los controladores del sitio web son creados desde la ubicación “Proyecto\app\Controllers” en la raíz de la carpeta de CodeIgniter (En la imagen podemos observar los controladores Creados).



## ➤ BaseController



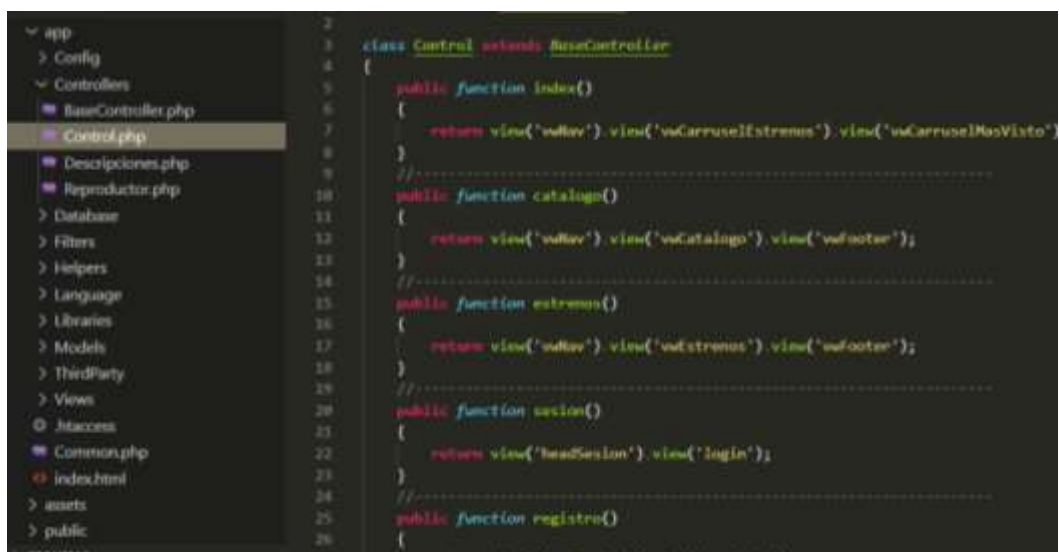
```
15  */
16
17  use CodeIgniter\Controller;
18
19  class BaseController extends Controller
20  {
21
22      /**
23       * An array of helpers to be loaded automatically upon
24       * class instantiation. These helpers will be available
25       * to all other controllers that extend BaseController.
26       *
27       * @var array
28       */
29      protected $helpers = [];
30
31      /**
32       * Constructor.
33       */
34      public function initController(CodeIgniter\HTTP\RequestInterface $request)
35      {
```

El archivo “BaseController.php” contiene lo que se puede considerar como controlador general (clase) para todos los apartados y elementos de nuestro sitio web. Es un lugar conveniente para la carga de componentes que tengamos creados en otros archivos brindándonos más seguridad en el sitio declarando métodos como protegidos o privados.

Básicamente es en este archivo donde se reconoce la ubicación de los archivos PHP donde tengamos métodos y funciones.

## ➤ Control

En este archivo encontramos cada método de todas las secciones que contiene nuestro sitio web agrupadas en una sola clase llamada “control”, donde se manda llamar a la función de vista de cada sección de los apartados del sitio.

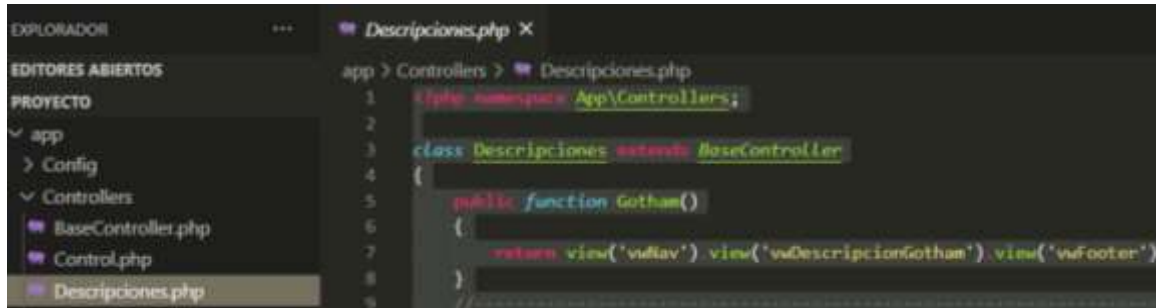


```
1  class Control extends BaseController
2  {
3      public function index()
4      {
5          return view('welcome')->view('vwCarruselExtremos')->view('vwCarruselMasVisto');
6      }
7      //.....
8      public function catalogo()
9      {
10         return view('welcome')->view('vwCatalogo')->view('vwFooter');
11     }
12     //.....
13     public function extremos()
14     {
15         return view('welcome')->view('vwExtremos')->view('vwFooter');
16     }
17     //.....
18     public function sesion()
19     {
20         return view('headSesion')->view('login');
21     }
22     //.....
23     public function registro()
24     {
25         return view('headRegistro')->view('registro');
26     }
27 }
```

## ➤ Descripciones

En este archivo PHP encontramos la clase “Descripciones” que nos sirve para mandar llamar la vista de la sección del sitio donde encontramos la descripción del contenido de la plataforma. Conforme se tengan las descripciones de cada título que presenta la plataforma se van agregando los métodos con llamada a su función de vista correspondiente.

28



## ➤ Reproductor

Al igual que en el archivo anterior donde se manda a llamar la función de vista de esa página, en el archivo “Reproductor.PHP” se encuentra la clase Reproductor en donde llaman las funciones ya mencionadas estableciendo el valor de retorno correspondiente a las secciones y/o otros elementos que contenga o se añadan en un futuro a esta página del sitio.



## VISTAS

Las vistas que creamos fueron las siguientes (siendo un total de 25):



- **consultaAdmin:** Esta vista pertenece a la consulta en la base de datos de los usuarios registrados en el sitio por parte del administrador, en la cual estamos mostrando por medio de una tabla con los campos “Nombre” “Apellidos” y “Correo” los datos correspondientes a los ingresados por el usuario ya registrados en la base de datos.

```

1 <table class="table table-responsive table-bordered" style="background-color: #E8F5E9">
2   <thead>
3     <tr>
4       <th>Nombre</th>
5       <th>Apellidos</th>
6       <th>Correo</th>
7     </tr>
8   </thead>
9   <tbody>
10    <tr>
11      <td>John</td>
12      <td>Doe</td>
13      <td>john@example.com</td>
14    </tr>
15    <tr>
16      <td>Smith</td>
17      <td>Thomas</td>
18      <td>smith@example.com</td>
19    </tr>
20    <tr>
21      <td>Merry</td>
22      <td>Jan</td>
23      <td>merry@example.com</td>
24    </tr>
25  </tbody>
26 </table>

```

Esta vista junto con la vista **navadmin** se puede visualizar de la siguiente manera:

Nombre	Apellidos	Correo
John	Doe	john@example.com
Smith	Thomas	smith@example.com
Merry	Jan	merry@example.com

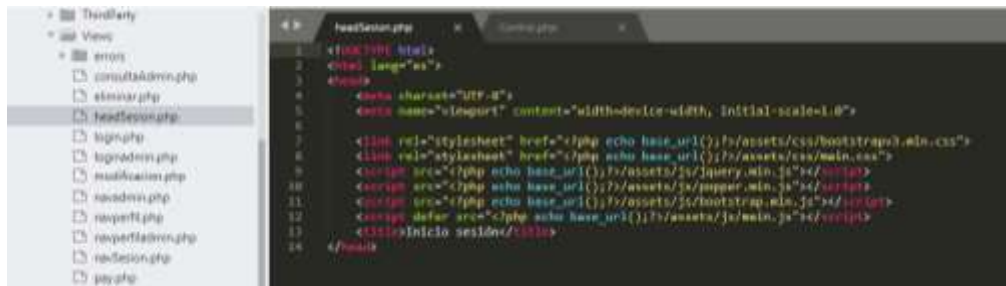
- **eliminar:** Esta vista es para que el usuario o el administrador pueda eliminar su cuenta de watch a través de un formulario similar al de registro.

```

1 <div id="login" background="#f9f9f9" style="border: 1px solid #ccc; padding: 10px; margin: 10px auto; width: 400px;">
2   <div class="card" style="background-color: #fff; border: 1px solid #ddd; padding: 10px;">
3     <div class="text-align: center; margin-bottom: 10px;">
4       <img alt="Watch Logo" style="height: 50px;"/>
5     </div>
6     <div class="form-group" style="margin-bottom: 10px;">
7       <input type="text" class="form-control" id="nombre" placeholder="Nombre" required="">
8     </div>
9     <div class="form-group" style="margin-bottom: 10px;">
10      <input type="text" class="form-control" id="apellidos" placeholder="Apellidos" required="">
11    </div>
12    <div class="form-group" style="margin-bottom: 10px;">
13      <input type="email" class="form-control" id="correo" placeholder="Correo" required="">
14    </div>
15    <div class="form-group" style="margin-bottom: 10px;">
16      <input type="password" class="form-control" id="password" placeholder="Contraseña" required="">
17    </div>
18    <div class="form-group" style="margin-bottom: 10px;">
19      <input type="password" class="form-control" id="confirmar" placeholder="Confirmar" required="">
20    </div>
21    <div class="text-align: center; margin-top: 10px;">
22      <button type="button" class="btn btn-primary" id="eliminar">Eliminar</button>
23    </div>
24  </div>
25 </div>

```

- **headSesion:** Es la vista que llama a los diferentes archivos externos de estilos como el de CSS, el JavaScript, y las carpetas del Bootstrap para darle el diseño al formulario.

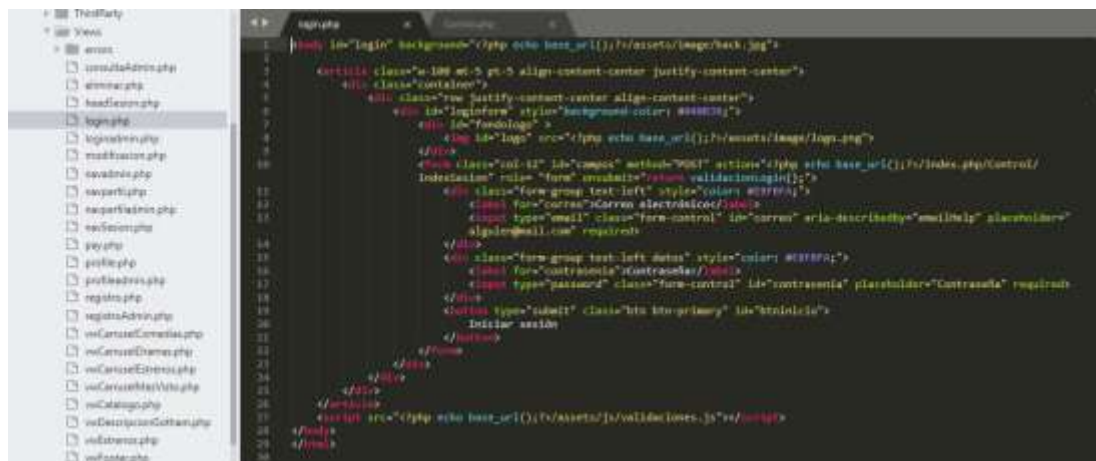


30

Con las 2 vistas anteriores (**eliminar** y **headSesion**) se forma la siguiente página, donde podemos observar que es un formulario para la eliminación de cuentas.



- **login:** Este código es parte del login, con esta vista se le da la estructura del formulario para iniciar sesión en watch.

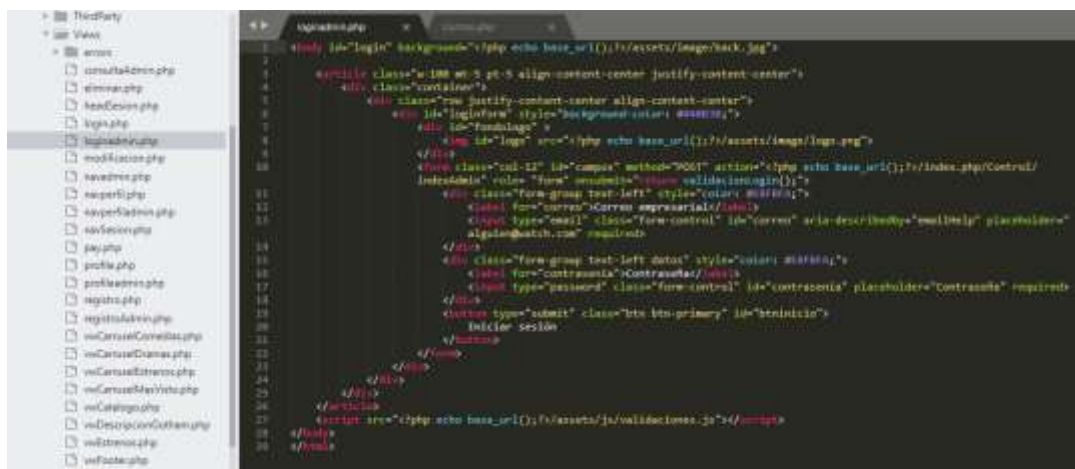




Con la vista [headSesion](#) y [login](#) le dan el diseño del formulario para ingresar con su cuenta, este formulario es solo para iniciar sesión, el de registro es otro.

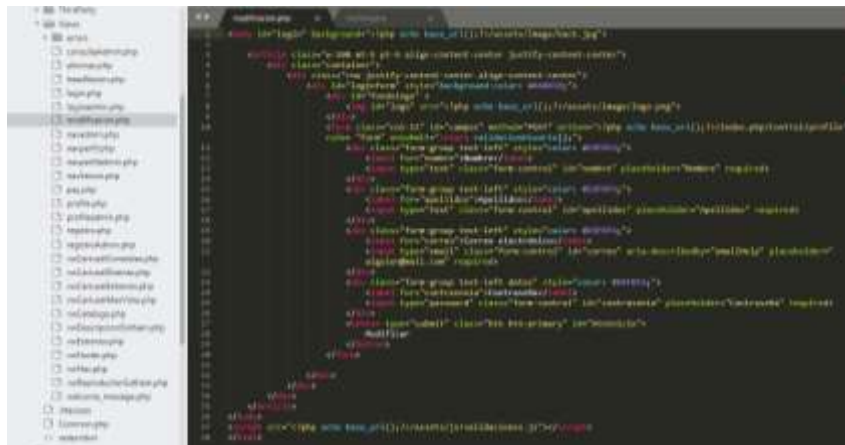


- **loginadmin:** Para que un administrador inicie sesión debe ingresar al login del administrador ya que a comparación del login normal, en este formulario solicita un correo empresarial el cuál es dado de alta desde la empresa, con esta vista se diseñó el formulario para que el administrador pueda iniciar sesión.



Con esta vista ([loginadmin](#)) y la de [headSesion](#) se forma el login del administrador, es similar al del usuario, pero el correo a ingresar es de diferente organización y dominio.

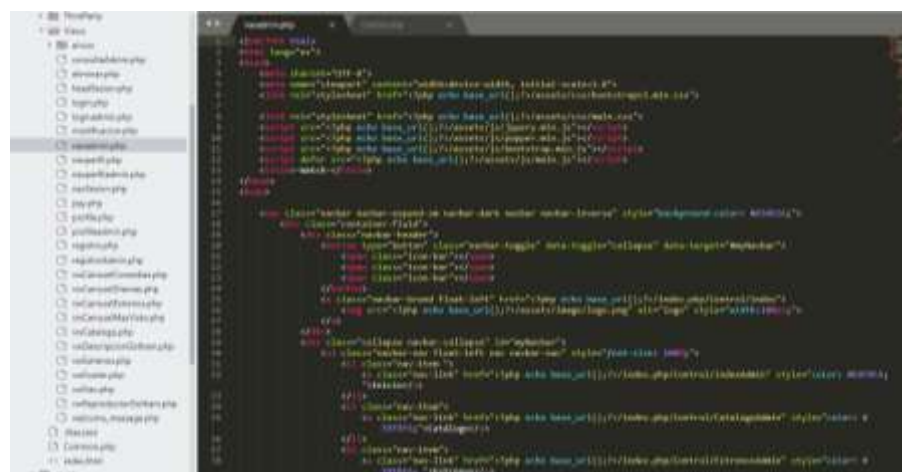
- **modificacion:** Para realizar cambios en tu cuenta (una vez que te hayas registrado en Watch ya sea como usuario o administrador) esta vista es la que te permitirá esos cambios, con esta vista te mandará a un formulario donde se podrán modificar los datos.



Con la vista **headSesion** y **modificacion** muestran un formulario para que tanto un usuario como un administrador puedan modificar los datos registrados en la base de datos del sistema.



- **navadmin:** Esta vista le da el diseño a la barra de navegación del administrador, a diferencia de un usuario de watch, el administrador puede realizar diferentes tareas de edición, al igual que tiene un botón desplegable en “Mi perfil”, el cuál muestra otras interacciones que tiene un administrador.



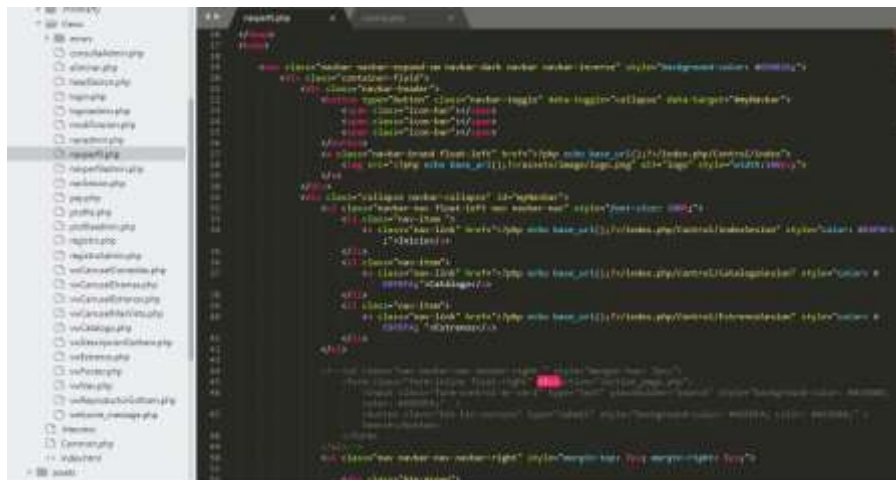


Esta es la imagen de la vista **navadmin**



33

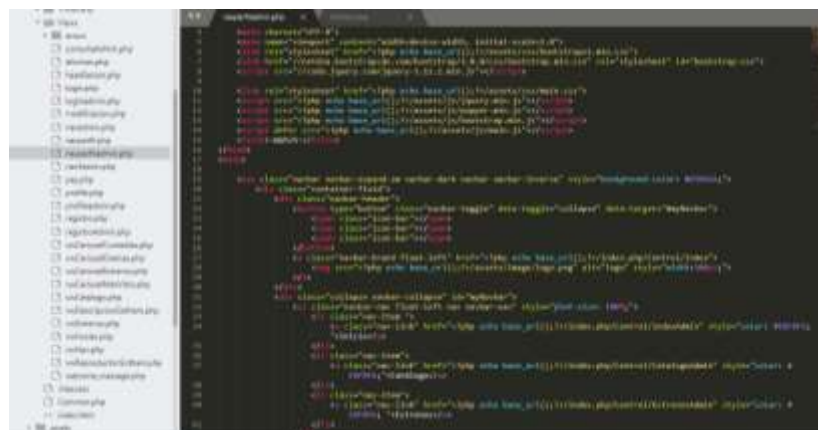
- **navperfil:** Con esta vista se personaliza la barra de navegación al momento de ingresar a la información del perfil de un usuario normal.



En la imagen siguiente se puede visualizar como se ve la vista en el sitio.



- **navperfiladmin:** En esta vista se configura la barra de navegación del administrador cuando éste se encuentra en su perfil.

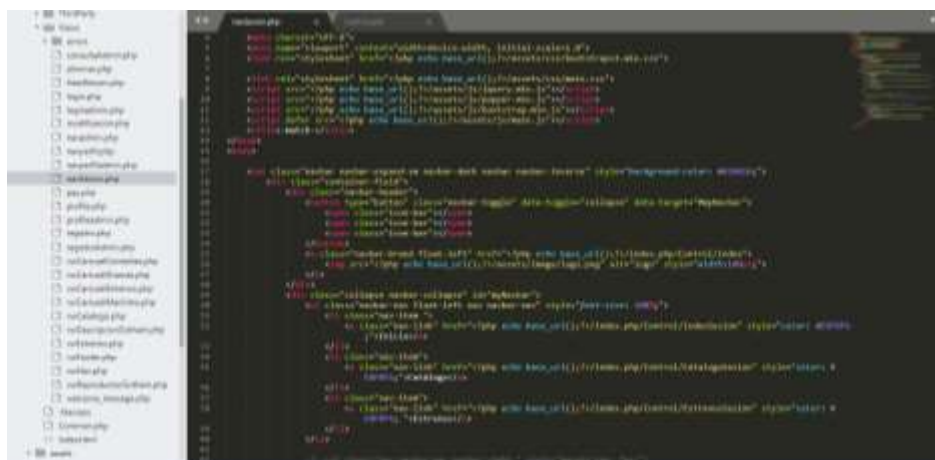


En la imagen se observa la vista desde el sitio web.



34

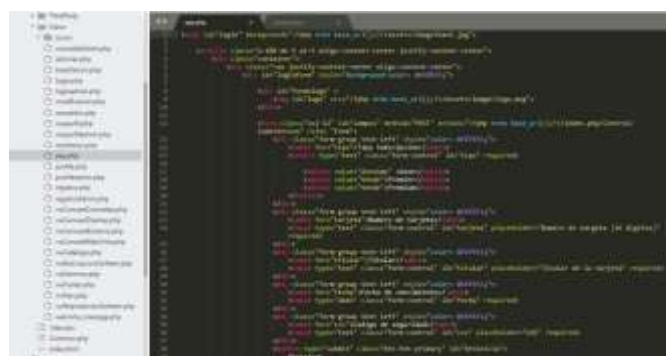
- **navsesion**: Cuando un usuario ya se registró cambia la barra de navegación, por lo que con esta vista se personaliza esa barra y es utilizada en todas las páginas una vez que el usuario inicio sesión.



Así es como se ve la vista, en la primera imagen muestra el botón “Mi perfil” y en la segunda el despliegue de éste mismo botón.



- **pay**: Con esta vista se diseña el formulario para realizar los pagos con tarjeta de crédito del usuario.

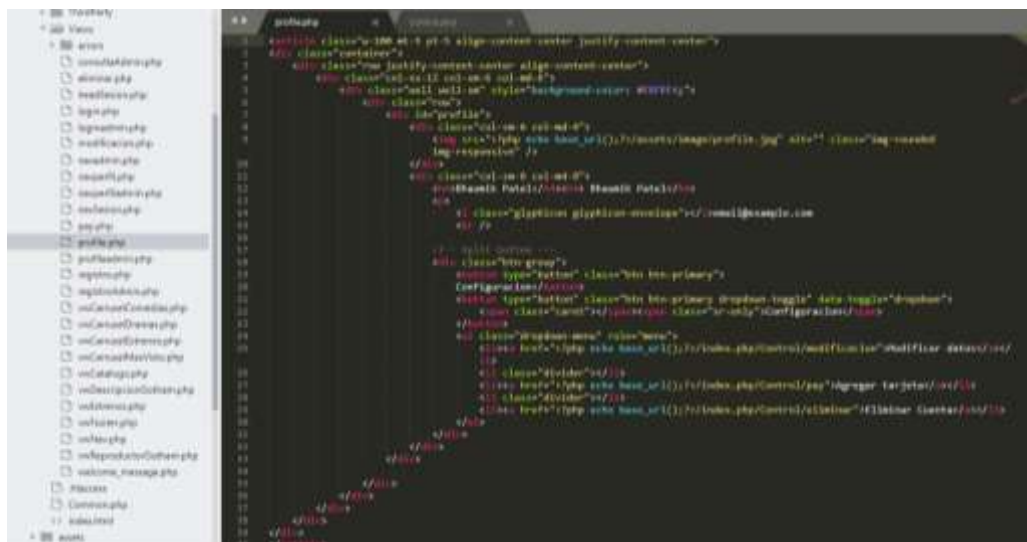


En esta imagen podemos ver la página para los pagos del usuario (vista [pay](#)).



35

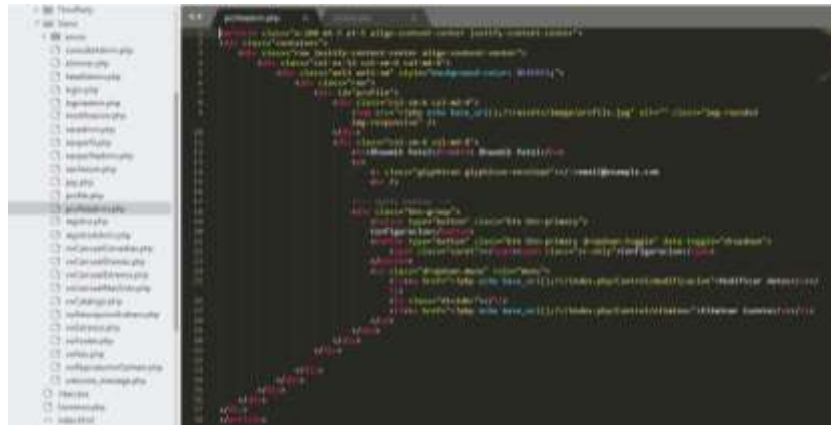
- [profile](#): Esta vista es para que el usuario pueda ver su perfil en watch con los datos registrados.



En la imagen se puede ver como el usuario estaría visualizando su perfil desde la vista [profile](#).



- **profileadmin**: Esta vista es similar a la del usuario, con la diferencia de que tanto en el botón de “Mi perfil” se despliegan otras opciones y el contenido es de un administrador

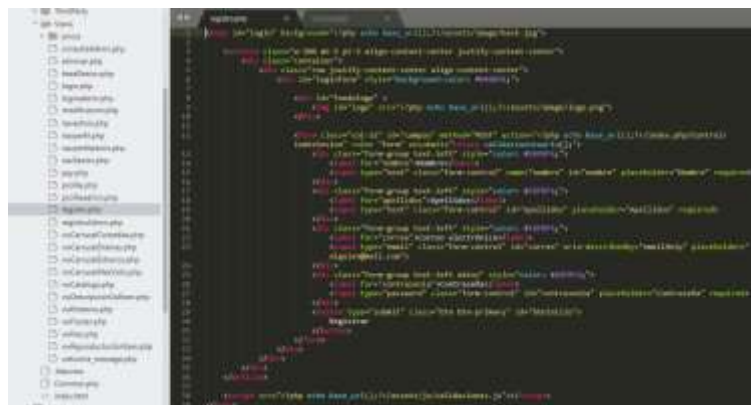


36

De esta manera se ven los perfiles de los administradores.



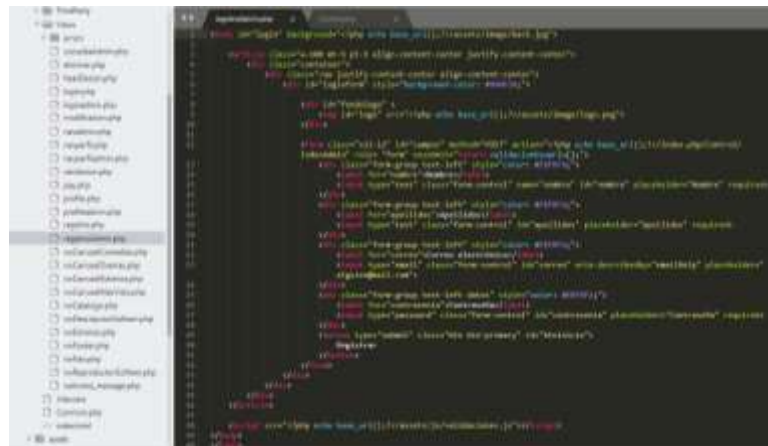
- **registro**: Esta vista crea el formulario del registro de un usuario, con este formulario las personas que no estén registradas en el sitio podrán darse de alta dentro de watch.



En la imagen del costado derecho se puede visualizar el formulario para el registro de los usuarios, es similar al de modificación de datos, pero cambia el botón y la acción es diferente ya que en esta vista se guardan datos nuevos y en la de modificación de guardan datos de un usuario ya registrado.



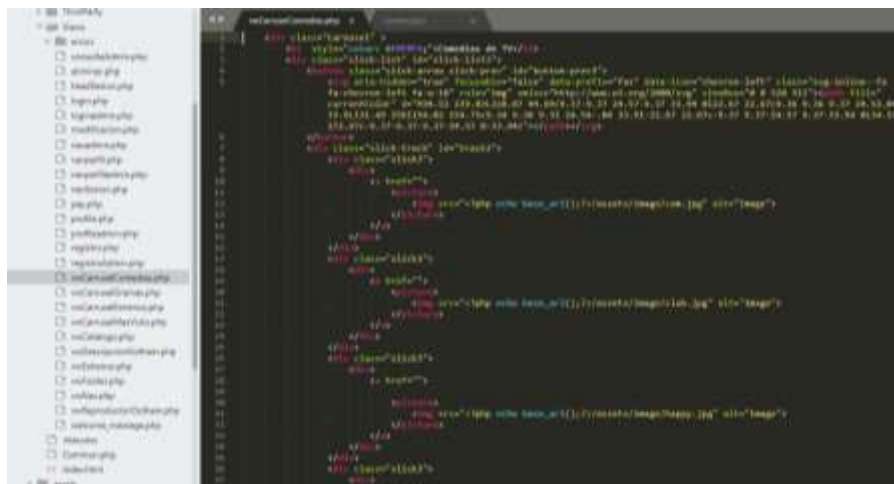
- **registroAdmin:** Esta vista es similar a la anterior, solo que esta es para los administradores del sitio.



El formulario de esta vista pareciera ser el mismo solo que este tiene la funcionalidad de dar de alta o bien registrar a los administradores del sitio.



- **vwCarruselComedias, vwCarruselDramas, vwCarruselEstrenos, VwCarruselMasVistos:** Estas 4 vistas son similares, solo cambian el contenido (imágenes de las portadas de las películas y series y títulos); las 4 forman parte del index.

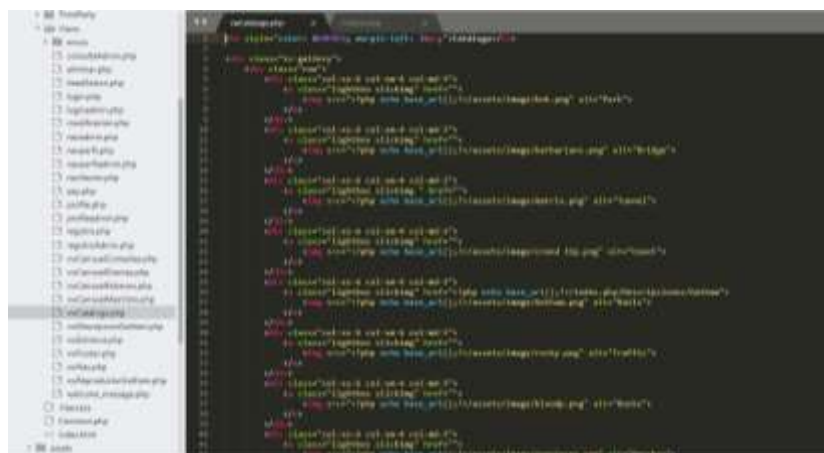




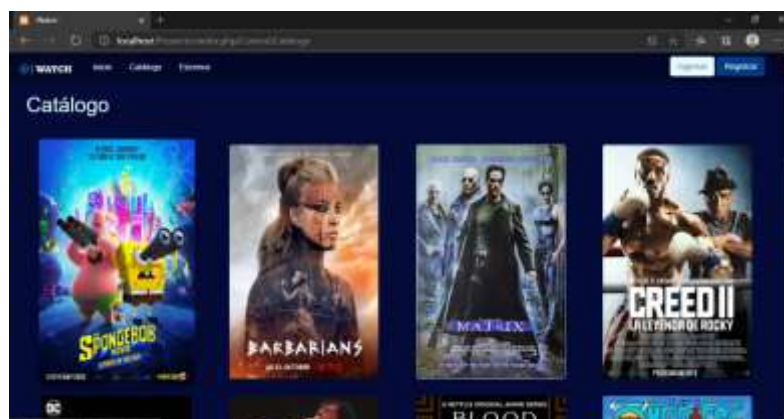
En la imagen se pueden ver 2 carruseles pertenecientes a las vistas [vwCarruselComedias](#) y [VwCarruselMasVistos](#), de esta misma forma se ven los 4 cambiando las portadas de las respectivas películas y series.



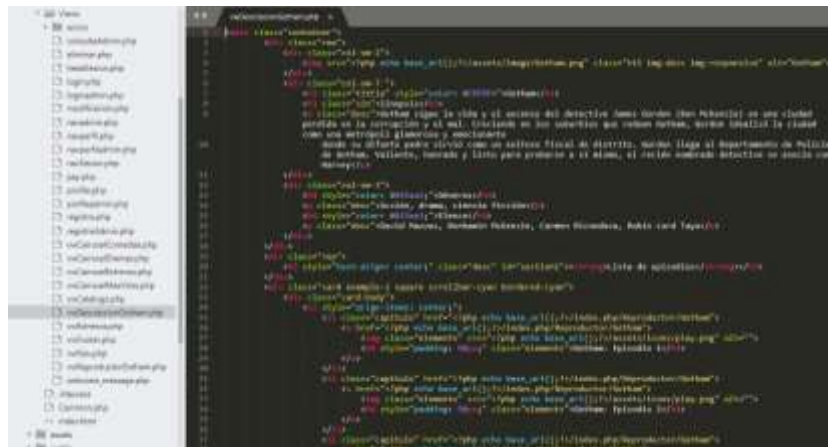
- [vwCatalogo](#): Con esta vista podemos ver en el sitio el catálogo de las películas existentes en el sitio.



La imagen es una visualización de la vista, [vwCatalogo](#) en esta se pueden apreciar las portadas del contenido audiovisual disponible.



- **vwDescripcionGotham**: Esta vista es la página donde da una descripción de la serie seleccionada, esta vista es utilizada en los usuarios registrados y los que no, lo que cambia es la vista nav", que es la que muestra la barra de navegación dependiendo el usuario.

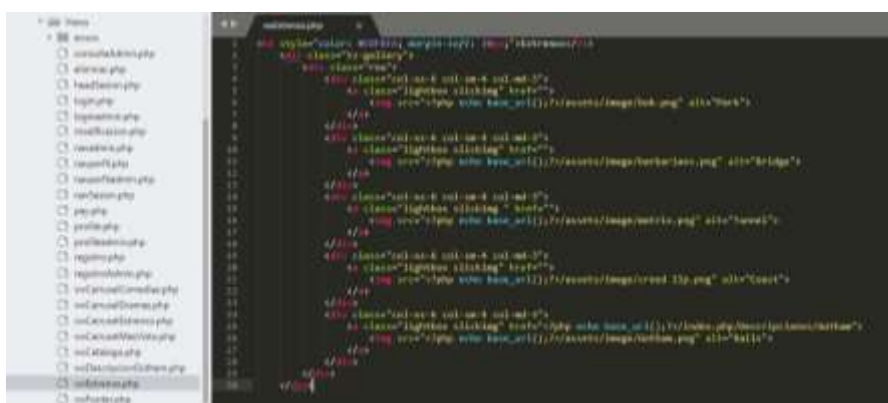


39

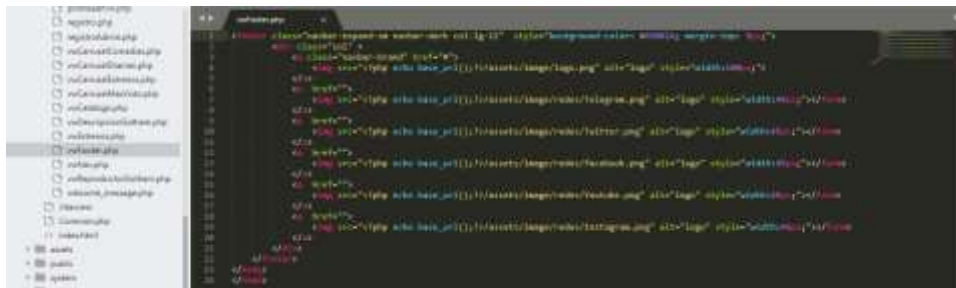
En la imagen se puede apreciar la vista **vwDescripcionGotham** donde se hace descripción de la serie y se ven otros datos como los géneros, el elenco, la lista de episodios, entre otros.



- **vwEstrenos**: Esta vista es similar al catálogo, lo que cambia es la clasificación, en esta vista se muestran las películas y series que son recién añadidas a la plataforma y son estrenos.



- **vwFooter**: Con esta vista se diseñó el pie de página del sitio, esta vista es visualizada en la mayoría de las páginas ya que su diseño no cambia, se mantiene el mismo en todos lados.

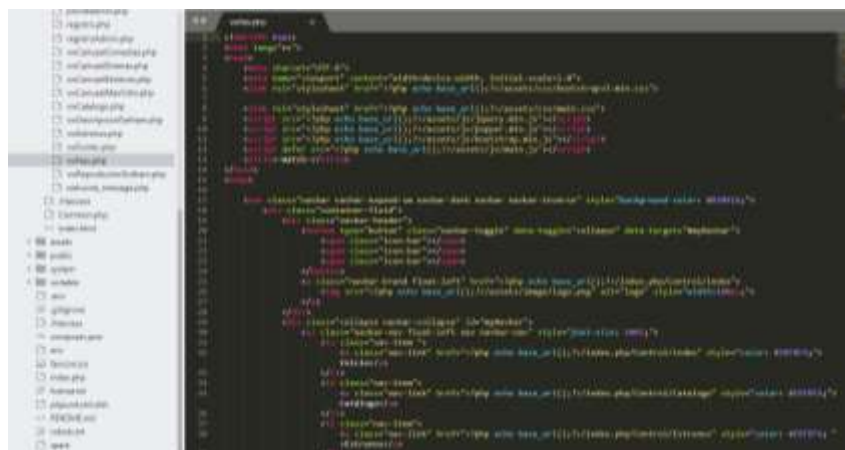


40

En la imagen se puede ver la vista **vwFooter** en la cual se ven los logos de las redes sociales y el logo del sitio.



- **vwnav**: Esta vista es la que muestra la barra de navegación a un usuario que aún no se ha registrado dentro del sitio de watch.

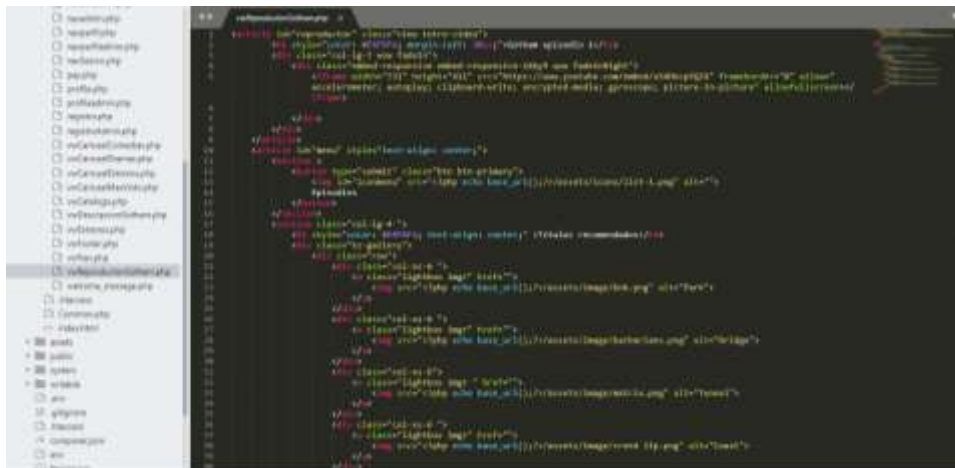


Como se ve en esta vista se observan 2 botones, éstos 2 botones salen cuando un usuario no se ha registrado, puede ver el contenido, pero son limitadas sus opciones.

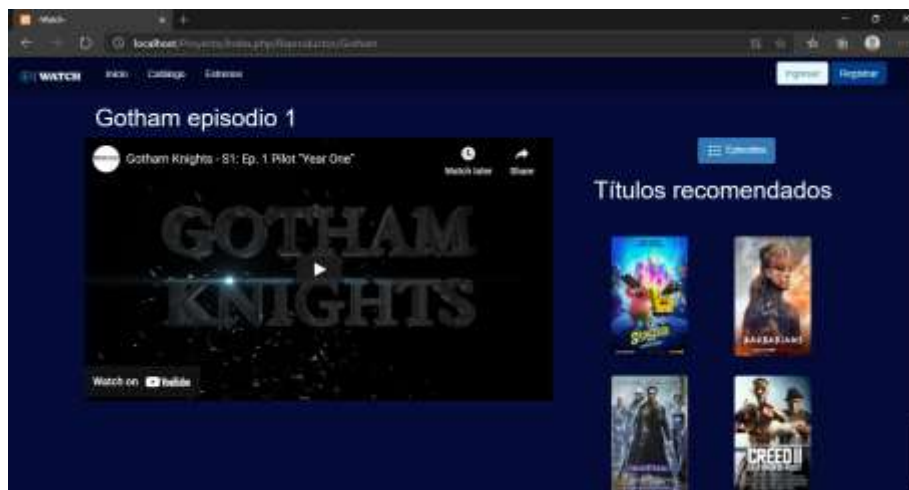


- **vwReproductorGotham**: Con esta vista se puede reproducir el contenido seleccionado, ya sea una película o un episodio de una serie, pero hay que cambiar algunas líneas de código, pero en esta vista se ve el episodio 1 de la serie de Gotham.





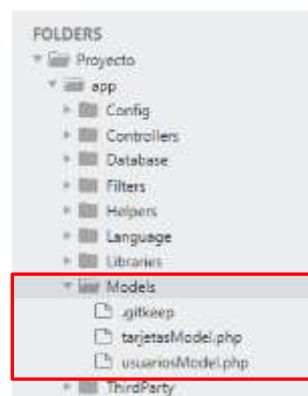
De esta forma se muestra la vista en el sitio, en este caso es el reproductor del contenido.



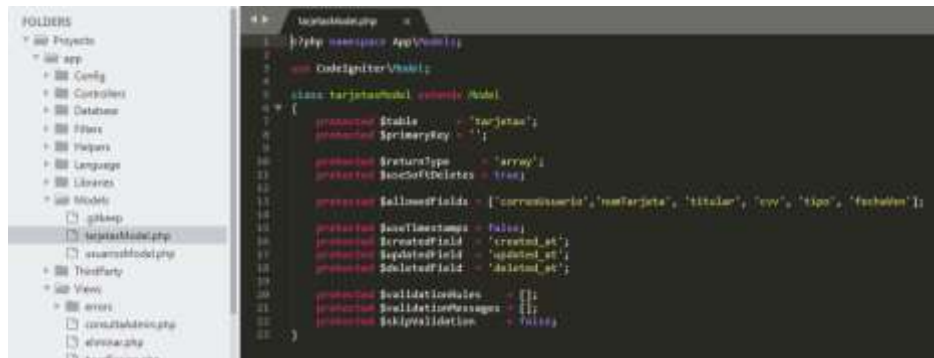
## MODELOS

Se crearon 2 modelos:

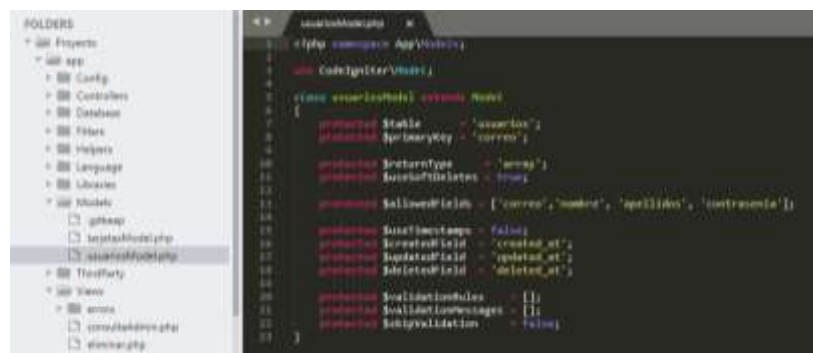
- 1) tarjetasModel
- 2) usuariosModel



- 1) **tarjetasModel**: Este modelo es utilizado para hacer la conexión del sitio web con la base de datos donde se guardan los datos tarjetas de crédito de los usuarios, los datos que se guardan son: Correo del usuario, el número de tarjeta, el titular, el cvv, el tipo de suscripción y la fecha de vencimiento.



- 2) **usuariosModel**: Este modelo es utilizado para hacer la conexión del sitio web con la base de datos donde se guardan los datos de los usuarios que se registran, cuando se registran se solicitan datos como: el correo, nombre, apellidos y la contraseña, para alojarlos en la base de datos y llevar un control de usuarios.



## LINK DEL REPOSITORIO WEB

<https://github.com/MarianoCH17/Watch.git>

## LINK DEL HOSTING

<https://watch17.000webhostapp.com>