

# Memoria SI Práctica 1

Sergio Hidalgo y Miguel Ibañez

|                     |          |
|---------------------|----------|
| <b>Introducción</b> | <b>1</b> |
| <b>Html</b>         | <b>2</b> |
| base.html           | 2        |
| cart.html           | 2        |
| details.html        | 2        |
| history.html        | 2        |
| index.html          | 2        |
| login.html          | 2        |
| register.html       | 3        |
| <b>Python</b>       | <b>3</b> |
| __init__.py         | 3        |
| __main__.py         | 3        |
| routes.py           | 3        |
| /index              | 3        |
| /login              | 3        |
| /logout             | 4        |
| /cart               | 4        |
| /register           | 4        |
| /history            | 4        |
| <b>Css</b>          | <b>4</b> |
| si1.css             | 4        |

# Introducción

En esta memoria se explicaran los distintos archivos que forman la aplicación web de un video club que se pedía implementar. Se irán explicando según su extensión y su funcionalidad.

## Html

### base.html

Este html ya venía dado en la web de ejemplo, pero se le han añadido etiquetas de salto de línea y los enlaces hacia el carrito y el historial que se pedían implementar.

### cart.html

En él se incluyen un bucle utilizando jinja 2 para mostrar los elementos del carrito, un botón para añadir más unidades de esa misma película y otro botón para borrar una unidad de esa misma película. Además incluye también el código necesario para poder mostrar mensajes flash (para cuando el carrito está vacío y se intenta comprar) y para que no deje pulsar el botón si el usuario no se encuentra en una sesión.

### details.html

Se muestran los detalles de una película (fecha, categoría, actores...) cuando es pulsada (tanto en el index como en el historial de compras). Se muestra también la valoración y se permite al usuario puntuar la película.

### history.html

Necesario para el historial de compras, a partir de una fecha y el total de la compra se despliegan las películas que compró el usuario al pasar el ratón por encima. Esto se consigue creando una clase y organizando en etiquetas de listas elementos, que después mediante css ayudarán a formar el menú.

### index.html

Aquí se forma el buscador con una request de tipo select con los géneros y un input de texto para el título de la película. A su vez tiene un botón con imagen que mostrará los detalles de la película.

## login.html

Mediante requests se obtiene el usuario y la contraseña. Además incluye también el código necesario para poder mostrar mensajes flash (que avisará si se introduce un nombre inválido o incorrecto o si la contraseña no es correcta).

## register.html

Mediante requests se obtiene el usuario, la contraseña, el correo, la dirección y la tarjeta de crédito. Si se incluye mal alguno de los campos anteriores saltará un flash message. Además hay un script de javascript que mostrará la fuerza de la contraseña, y una barra implementada mediante jquery que mostrará el progreso acorde a la fuerza de la contraseña.

# Python

## \_\_init\_\_.py

Se cargan las sesiones de flask y si no se puede saltará un error.

## \_\_main\_\_.py

Main donde se carga la página en el puerto 5001.

## routes.py

Archivo de rutas que realizan la lógica detrás de los html.

## /index

Primero se comprueban los requests del buscador en el que se comprobará si la categoría es 'all', si es así solo se realizará una búsqueda en lo referente al título, si se ha introducido algo, sino se mostrarán todas las películas. Si la categoría es otra se comprobará si se ha introducido o no texto, si se ha introducido texto se buscará una película que coincida en título y categoría, y sino se mostrarán todas las películas de esa categoría.

También se comprueba el request en el que se añade al carrito desde los detalles de la película, añadiendolo al carrito si no está y sumando uno ya se encuentra esa película en el carrito.

La puntuación de las películas también aparece en el index, recogiendo en un request la puntuación dada por el usuario y añadiendo al archivo inventario.json.

## /login

Se comprueba si el nombre de usuario está vacío y si lo está saltará un mensaje flash, sino se comprobará que existe, si no existe saltará un mensaje flash, y si existe se comprobará la contraseña dentro del archivo userdata, si no coincide saltará un mensaje flash y si sí lo hace iniciará sesión con el usuario.

## /logout

Salida de la sesión

## /cart

Si no existe una sesión con el carrito se creará un diccionario donde se guardará la información del carrito.

Se comprueba si hay un request en delete y si es así se borra (una unidad si hay mas de una o la película entera si solo hay una)

Se comprueba si hay un request en add y si es así se añade (una unidad si hay mas de una o la película entera si solo hay una)

Se comprueba si hay un request en purchase se calculará el total y se comprobará si tiene saldo, si no lo tiene saltará un mensaje flash, y si sí lo tiene se restará al saldo del usuario y se añadirá al archivo de compras del usuario.

## /register

Se comprueba si existe el directorio si1users y si no es así, se crea, se comprueban todos los campos necesarios y si falla alguno saltará su flash message correspondiente y una vez comprobados todos los campos se crearán los ficheros userdata e inventario.json.

## /history

Se comprobará el request de los detalles de la película al darle al botón en el menú desplegable, pero este menú no se mostrará si no ha iniciado sesión, si si que se ha iniciado sesión se mostrarán las compras del usuario.

# Css

## si1.css

En el css se aplican los cambios estéticos para dar forma a la página web.

- body: se cambia el color del fondo de la página
- t1: se cambia el tamaño del título página
- t2: colocación de la barra de búsqueda
- t3: tamaño del título de la película
- t4: colocación de los autores en los detalles
- t5: colocación de los detalles de la película
- stars: color de las estrellas
- contra: colocación de la contraseña en el registro y de la barra de fuerza de la misma
- segment: colocación de las películas en el index
- segmenth: colocación del menú del historial de compras
- segmentin: colocación del inicio de sesión
- segmentr: colocación del registro
- body\_menu: tipo de letra y fuente del menú
- .navigation: elimina los puntos de cada compra
- .navigation li: define el tamaño del menú
- .navigation li a: define la colocación de las compras
- .navigation ul: define la colocación de las películas de cada compra
- .navigation ul li: define el tamaño de los detalles de las compras
- .navigation ul a: define el largo del bloque de los detalles de las compras
- barra: define la colocación de la barra de progreso de la contraseña