



Práctica 1

Diseño en HTML y CSS

Francisco Javier Merchan Martin 31032280J

Índice

Introducción	2
Principio del desarrollo	2
Desarrollo	3
Index.html y gestorbd.html.....	3
Bd1.html.....	5
Altarecurso.html	8
Recursoseccion1.html	10
Recurso1.tml	12
Index.html	13
Gestorbd.html	15
Contacto.html	18
Altagestor.html	19
Editargestor.html	21
Borrargestor.html.....	22
Crearbd.html	23
Editarbd.html	25
Borrarbd.html.....	26
EdicionRecursos.html.....	27
BorrarRecurso.html.....	28
Altaseccion.html.....	28
Editarseccion.html.....	29
Borrarseleccion.html	30

Introducción

Esta página web se hace como resultado a un problema explicado en clase, para su posterior evaluación de la parte práctica de la teoría. El problema es cuestión es crear un gestor de bibliotecas digital, que se compone de varias bibliotecas, y a su vez esta se compone de secciones que agrupan los recursos de esta biblioteca digital.

Nuestro gestor de biblioteca se llama *DigiBiblio Gestor*, hace de la combinación de palabras Digital y Biblioteca. Y nuestro símbolo es un libro negro sobre un fondo azul. Para esta práctica hemos creado una biblioteca digital para películas. Esta a su vez se divide en 4 secciones Comedia, Acción, Suspense y Animación. Es decir, clasificamos la biblioteca Películas a través de su género.

Hemos apostado por un diseño minimalista con colores pasteles, para hacer más amigable y llamativa a la vista. También hemos pensado en las pantallas más reducidas por lo que no está hecha totalmente adaptable a dispositivos móviles (aunque la mayoría de la página se ve perfectamente en estos dispositivos), sí que está pensada para pantallas medianas o “tabletas”. Por lo en pantallas de gran tamaño se puede ver un poco vacía o demasiado dispensados los elementos.

Principio del desarrollo

Como todo desarrollo, primeramente, me leí el guion donde estaba especificado las directrices y algunos diseños del proyecto. Seguidamente me puse a investigar un poco más de html y css a ver cómo se podía hacer los diseños propuesto. Mis primeras fuentes y tomas de contacto son los guiones de prácticas, poco después seguí con una de las páginas que se recomienda para algunos conceptos de los guiones que no me quedaban claro, esta página es *w3schools.com*, y me ayudo a tener algunos conceptos más claros.

Con estos conceptos me puse a empezar a programar la primera página del proyecto. Como no sabía por qué página de todo el gestor empezar, me decidí empezar por la primera que aparece en el guion, *index.html*.

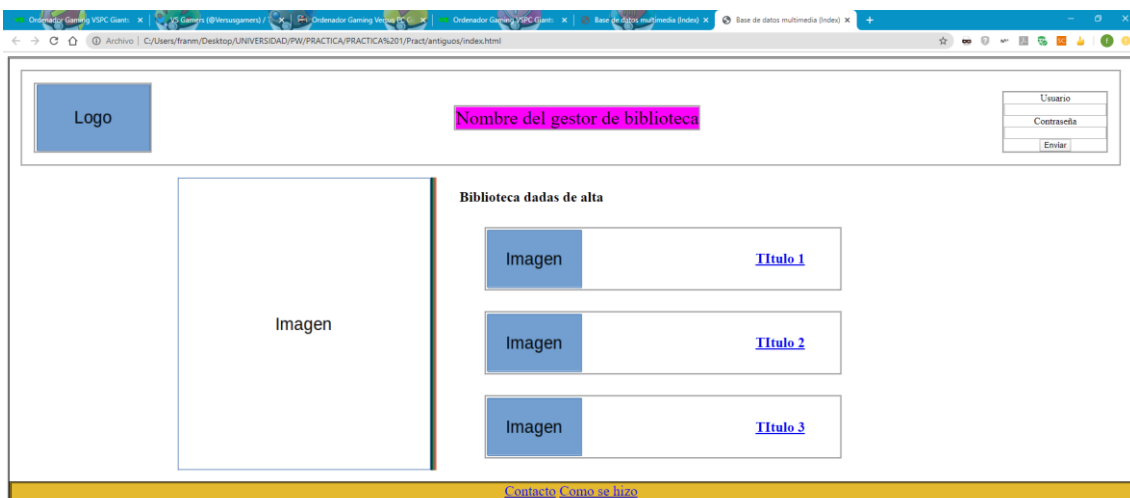
Desarrollo

Index.html y gestorbd.html

Primer gran problema

Empecé programando lo que era la cabeza, el cuerpo, etc. Programe todo el texto html del tirón separándolo por sección y articulos anidados, mi primer error. Al ver que había sido más o menos fácil y rápido, me puse con ánimos conseguir hacer el css y aquí me encontré con un gran problema. Dar formato a ese monstruo de html lleno de sección y articulos era misión imposible.

Con muchas horas, paciencia y algunos 'parches' al código html, conseguí que se veía tal que así:



Especial mención a lo que me costó poner todo en orden, en el cual mezcle el modelo de cajas flex, con el de flexbox y block. Dado la falta de practica y el no tener muy claro algunas funciones de estas, ya que como supe después no todos los atributos funcionan en todas.

También intenté conseguir que siguiera el diseño *responsive design*, adaptándolo a pantallas más pequeñas como de móviles y tablets, algo que conseguí para el segundo dispositivo, pero a medias para los móviles.

Viendo el resultado no muy distinto del propuesto me di por satisfecho con el diseño, aunque apuntándome que tenía que volver para arreglar el diseño para móviles, proseguí con el diseño de la próxima página, *gestorbd.html*

Cuando empecé con *gestorbd.html*, pensé que podía reutilizar el código, ya que la pagina es muy parecida a la anterior solo por algunos detalles como el usuario, la foto y las opciones de las distintas bibliotecas, ese fue mi segundo gran error.

Con la reutilización del código pensé que me ahorraría mucho trabajo que ya había hecho en la página anterior, cosa que no fue así ya que me resulto muy costoso encajar esos cambios. El código html no tiene muchos cambios comparados con la anterior página. El código en css sin embargo, sí tiene muchísimos más cambios, que me llevaron mucho tiempo, mucho más que con la primera página (casi 1 semana de trabajo).

El resultado fue tal que el siguiente:



Después de tanto tiempo dedicado a estas dos páginas, me desespero y me bloqueo tanto la relación del tiempo dedicado al trabajo con el resultado hecho que me tuve que tomar un pequeño descanso de unos días para volver a empezar de nuevo.

Lo que saqué en claro de aquello fueron varias cosas. La primera de ellas que necesitaba más información acerca de cómo ordenar las páginas HTML, por lo que busqué algo más de información acerca de esto, encontré varios manuales, pero la página que más me ayudó fue developer.mozilla.org, aunque la mayoría de la página está en inglés, hay algunas secciones, las más usadas en castellano, por lo que me ayudó a comprender mucho de los elementos que no comprendía. Entre este concepto estaba el de *flex* que se volvería la parte más importante y en lo que está basado gran parte de mi diseño.

La segunda cosa que aprendí fue a que mientras diseñaba ayudaba mucho poner colores de fondo a literalmente a todo lo que estuviera editando, ya que ayudaba a tener claro los márgenes, paddings y como afectaba cada atributo o cambio en estos sobre los elementos en la pantalla.

La tercera cosa que aprendí es a hacer las cosas por partes. Primero una parte de HTML, y después un poco de CSS. Así conseguía que no se volviera tan caótico y que cuando cambiaba algo sabía exactamente por qué atributo era.

La última cosa, aunque esta la descubrí un poco más tarde es que el navegador tiene un "editor" de código que te muestra exactamente todo el código así como su formato y todos los márgenes que tiene, esto me ayudó a ajustar algunos márgenes que no sabía cambiar.

Con todo esto en mente y después de haberme despejado me puse a hacer la siguiente página `bd1.html` dejando el desastre de las otras dos páginas atrás y dejándolas para rehacerlas más tarde.

Bd1.html

Con todo lo aprendido de los problemas de las otras dos páginas, me puse manos a la obra con bd1.html.

Primero me puse a hacer la cabera, la cual me serviría para poder hacerla en las demás paginas ya que no cambia en ninguna de ellas, excepto en las primeras antes de meterte en alguna biblioteca.

Para ello utilicé la etiqueta de *header* y añadí la imagen de la biblioteca el título y el nombre del usuario. En el css, utilicé como dije antes el atributo *display* con el valor de *flex*, así como el atributo *justify-content* con el valor *space-between* para que allá espacio entre ellos. Para la imagen definí el tamaño relativo al espacio que tiene así como un tamaño máximo para que no se viera tan grande pero esta vez en pixeles. Para el texto aparte de lo anterior use *text-align* con valor de *center* el cual hace que se centre el texto en el medio del cajón. Para el nombre del usuario use también el atributo *align-self* con el valor de *flex-start*, que alinea el componente con el inicio del bloque flex.

Para el menú de navegación, tuve que investigar un poco más, encontré varios tutoriales, por lo que mezcle varios de ellos. Primero en el código html utilice la etiqueta *nav* que se utiliza para los menús de navegación. Después dentro de este puse unas listas con las secciones de la correspondiente biblioteca.

Para el css. A través de los tutoriales le di el formato a cada una de las etiquetas. Para añadir el efecto de seleccionar cuando pasamos el ratón utilizamos la etiqueta: *hover*, que básicamente es para cuando se selecciona elemento. También se tuvo que cambiar la anchura y el padding para adaptarlos a nuestra página.

Para seguir haciendo la página definimos una etiqueta *main* en html en la que desarrollaremos el centro de la página. Que estará dividida en:

Columna Izquierda

En html, definimos un *setcion* con este id, dentro de él introducimos el texto con una breve introducción de la biblioteca a través de la etiqueta *p*. Después introducimos el título de la sección a través de la etiqueta *h1*. Para separarlo de lo otro dentro de un *article* en el que estará la lista de los Recursos destacados.

Para dar forma a cada parte de la lista, dividimos cada recurso en un *setcion*. Dentro de él tenemos una imagen, la del recurso. Después en un *articulo* para ayudarnos a dividir el diseño en css tenemos el título con un enlace a la página de esta (con una etiqueta *a* dentro de una *h2* que define que es un título) y una breve sinopsis del recurso bajo una etiqueta *p*.

Como formato de css, seguimos con los atributos de *display* con valor *flex*, para poder poner las partes de este bloque en forma vertical en vez de horizontal utilizamos el atributo *flex-direction* con el valor de *column*. Y con el atributo *width* ajustamos el ancho de esta parte que es del 65%, así como el borde con el atributo *border-style* con el valor de *doble* para poner el borde doble.

Después le damos formato a las distintas partes con los atributos utilizados antes para definir el padding y margin y así centrarlos. Para poner el scroll en la Lista de Recursos he usado el atributo *overflow-y* y el valor *scroll*, para poder crear la barra para subir y bajar las secciones, también fije que la máxima altura que pueda coger del bloque donde está fuera del 35 em (una medida de pantalla escalable) a través del atributo *max-height*. También he usado los atributos *justify-seft* y *justify-content* para centrar el elemento y el contenido respectivamente dándole el valor *center*.

Información

Al otro lado tenemos un bloque de información. En este caso hemos utilizado la etiqueta *aside* para encapsularlo. Dentro de esta encontramos que está dividido en varias partes. Primero nos encontramos con un *setcion* que contiene el bloque de información acerca de la biblioteca. Cada información está etiquetada con *p* excepto el título que lo contiene bajo la etiqueta *h3*.

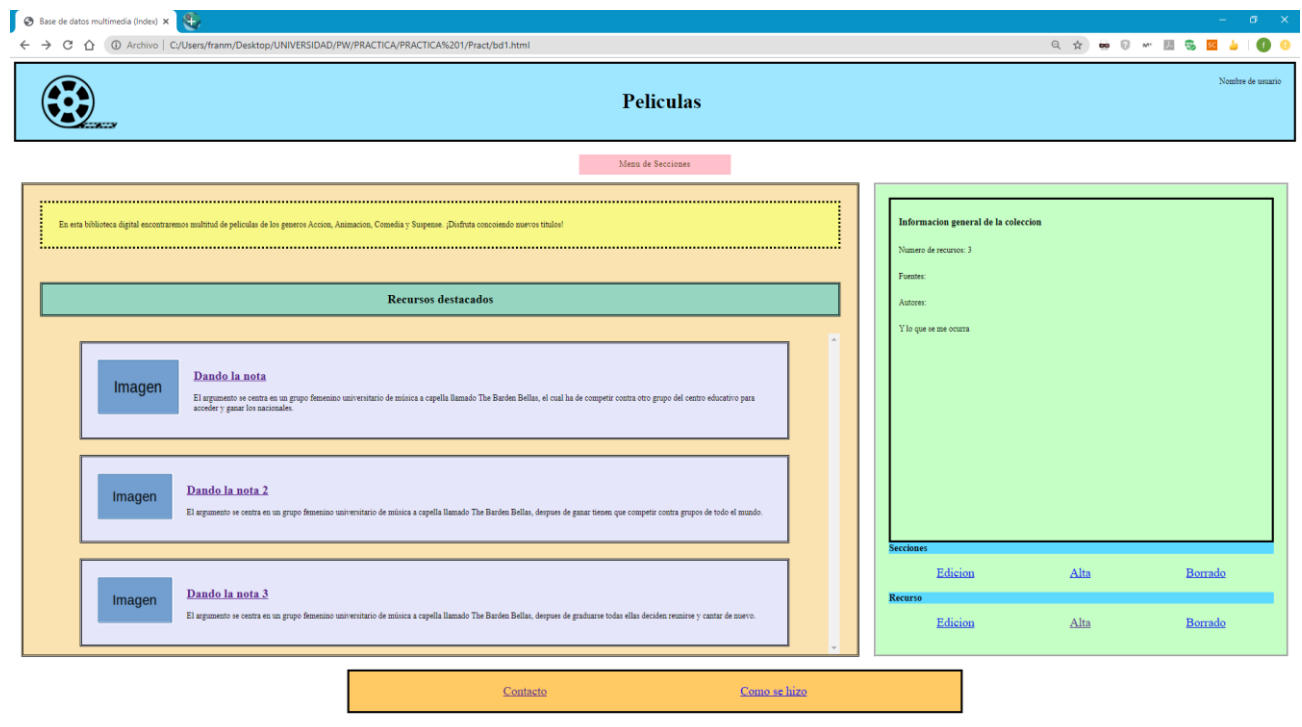
Seguido de esta tenemos un título con una etiqueta *h3* y un *setcion* que contienen los links de las opciones de las secciones. De igual manera encontramos abajo las opciones de los recursos de igual manera distribuida.

Para el css de esta parte he usado los mismos atributos, pero con diferente configuración de padding y margin. Así como otros trucos, como por ejemplo, para poner las opciones de secciones y recursos en fila y no en columna, esto es posible gracias a la división que hemos hecho antes en html y usando el mismo atributo que hemos usado antes para ponerlo en vertical pues esta vez lo cambiamos por el valor *row*, junto con el atributo *justify-content* con el valor de *space-around*, el cual hace que los elementos se separen entre ellos por con espacio alrededor utilizando el máximo de anchura disponible. Otro atributo que hemos utilizado es el de *font-size* para modificar el tamaño de la letra, en este caso el *x-large*.

Para el css general de la parte de main he usado el atributo *display* con el valor *flex*, así como el atributo mencionado anteriormente de *justify-content* con el valor de *space-around*, para que ocupe lo máximo posible, pero guardando unas distancias.

Por último, tenemos el pie de página encapsulado en una etiqueta *footer*. Esta a su vez tiene los dos enlaces requeridos, a una página de contacto y a otra a este documento. Para dar el formato en css, he usado los mismos atributos de siempre. Aunque esta vez me acordé un poco más de centrarlo por lo que he tenido que poner varios atributos. En primer lugar, el de atributo *justify-content* para desplazar ambos elementos a ambos extremos de la caja que ocupan. Seguidamente le fui dando valores al margin por la derecha y por la izquierda para que se fueran acercando y el cuadro se hiciera más pequeño. Cuando el cuadro era del tamaño deseado, al igual que he hecho con el margin lo hice con el padding para juntar los elementos a la distancia deseada.

Para los colores como hemos dicho nos hemos decantado por colores pastel sobre el fondo blanco de la página.

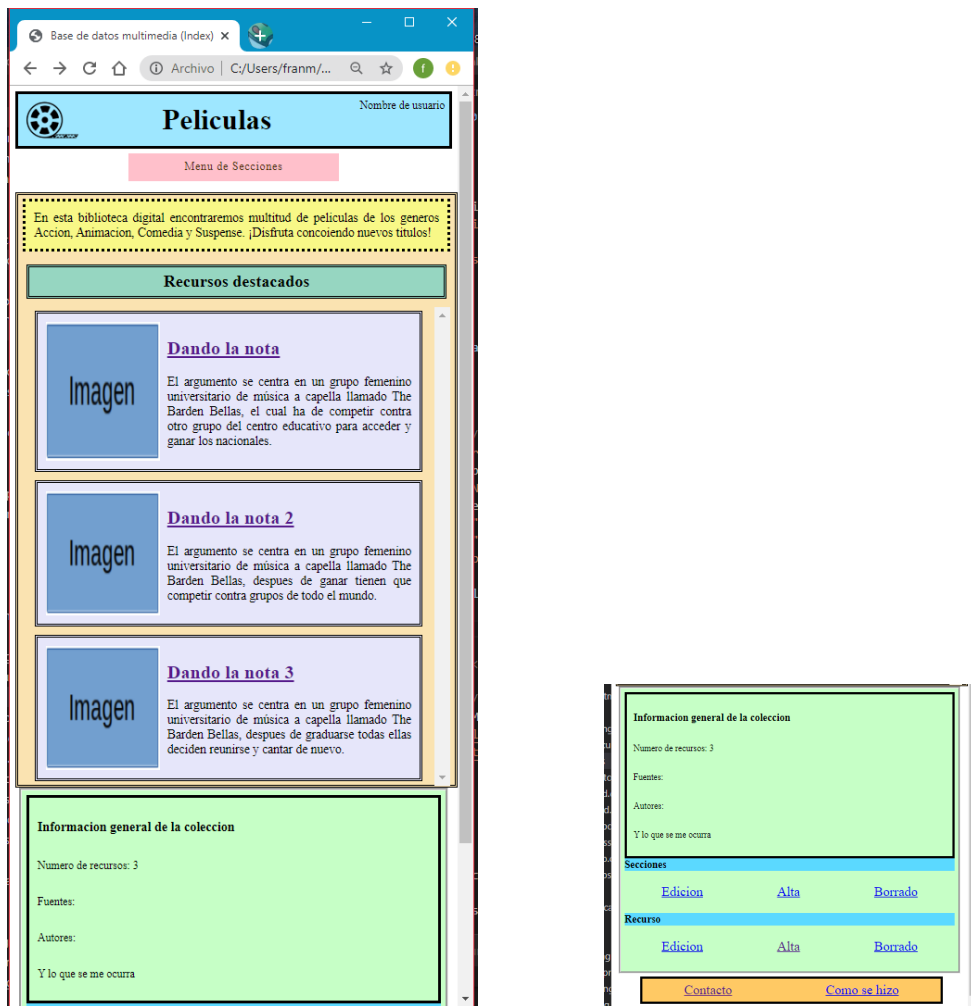


En el guion explica que esta página tiene que estar especialmente dirigida al responsive design, por lo que con ayuda de la sentencia:

```
@media screen and (max-width: 768px){}
```

Todo lo que haya dentro de ella en CSS solo se aplica cuando el ancho de la pantalla no supera los 768px, que es lo que pone en el guion lo correspondiente a las pantallas pequeñas. Para que se adaptara simplemente cambie par que en vez solo hubiera una columna en el centro con *flex-direction* dale valor de *column*, así ponen las dos columnas una debajo de otra. Gracias al uso de porcentaje en la mayoría de las opciones no tuve que retocar gran cosa como los *width* de las columnas y ponerlos al 95% para que ocupara el ancho de pantalla. También he modificado el padding y los márgenes de elementos para que se vean un poco mejor como es el caso del footer que he tenido que reducirles los márgenes establecidos anteriormente.

Con estos cambios la página se ven tan bien tal que así:



Altarecurso.html

Con lo aprendido en la anterior página, proseguí con la siguiente página propuesta por el guion. En este caso *altarecurso.html*, este diseño también me sirvió como plantilla para los diseños de *editarrecurso.html* y el *creaabd.html*.

La cabecera, el menú y el pie de página como hemos dicho antes es el mismo. Para el centro, encapsulado en un section, hemos dividido en varias partes:

Formulario

Que a su vez se divide en varias partes: Centro Arriba (Izquierda y Derecha), Centro Abajo y botonera. Como todo el centro era un gran formulario los article están dentro de la etiqueta *form*. Aunque se podía hacer de muchas ellas, por ejemplo, poniendo a cada *input* al formulario al que pertenece.

En primer lugar, vamos a ver el Centro Arriba. Se compone a su vez por dos section. El de arriba izquierda guarda el enlace para subir el recurso a través de una etiqueta *input* de tipo *file*. En el terreno del *css*, tenemos una de las cosas más complicadas que he hecho en el trabajo, ya que basándome en un artículo que encontré intenté emular una versión más simple de este. Para ello primero puse en su sitio a través de los tributos *display*, *margin* y *align-self* en el medio de la parte izquierda. Después mediante la sentencia *input[type="file"]* fui dándole formato al botón que permitía subir el recurso. Para ello a través de las sentencias *position* y con el valor *absolute*, con ayuda de los valores *left* y *top* pude colocar el elemento dentro del recuadro para que sea interactivo. Aunque tiene el problema que, como son valores aproximados puede que se desvíe cuando se redimensione la pantalla. Aun así el nombre sigue siendo totalmente funcional cuando pinchas en medio, aunque se mueva el recuadro y no entorpece a los otros componentes. Con el atributo *opacity* y el valor *0*, hacemos que desaparezca de la vista, pero sin que se pierda la funcionalidad.

El centro arriba derecho en cambio contiene casi todos los campos de formulario acotados por un *section*. Dentro de ellas encontramos diferentes campos acotados por *article* cada uno, así como diferentes etiquetas *input* y *select* para los distintos datos del recurso dentro de estos. Entre ellos tenemos:

- ***inputs con tipo text*** para el Título, Director y Protagonista.
- ***inputs con tipo date*** para la Fecha de estreno.
- ***inputs con tipo number*** para la Duración.
- ***select*** con opciones para Género, subgénero, País y Tipo de archivo.

Como pone en el guion que todos los campos deben de ser obligatorios añadimos el atributo *required* el cual no nos dejara enviarlo hasta que los campos con este atributo estén rellenos.

Para el *css* de estamos ante las mismas sentencias de siempre. Esta vez aplicamos el atributo *justify-content* con el valor de *column* para poner todos los datos uno encima de otros y *justify-content* con el valor de *space-between* para que ocupen todo el espacio disponible en la separación entre ellos, por eso lo hemos añadido dentro de un *article* a cada campo. Después hemos dado formato a ese *article* así como a los *inputs* y los *select* de dentro con *width* para el ancho, *padding* y *margin* y para ajustar estos parámetros y con *font-size* para ajustar el tamaño de letra.

En segundo lugar, tenemos el centro abajo que incluye el campo con la descripción. Para esto en *html* hemos usado una etiqueta *textarea*, a la que le hemos añadido los tributos de *row* para las filas, *column* para las columnas y el *required* para hacerlo obligatorio. En el *css*, hemos usado los mismos atributos que en el arriba, salvo que esta vez *width* le hemos dado el valor *auto* para tener el máximo del campo de descripción en la pantalla, así como *flex-direction* con el valor *column* para separar la etiqueta y el cuadro de texto por separado.

Por últimos tenemos la botonera, que incluye en html la etiqueta *input* con el tipo de valor *submit* que sirve para enviar el formato por el método declarado en el formulario. Para el formato en css, hemos usado el atributo *float* con el valor de *right* para moverlo a la derecha como en el diseño propuesto así como varios valores que hemos mencionado antes para modificar el margen (*margin*) y tamaño de letra (*font-size*).

Por último, he de decir que el formulario se hace a través del método post.

Como hemos dicho antes el footer tiene lo mismo que en la página anterior, por lo que es igual tanto en html como css.

El formato de colores seguimos usando el formato de colores pastel sobre un fondo blanco. En este caso para diferenciar entre las distintas altas hemos optado por usar distintos colores en este caso hemos decidido usar el naranja para los recursos.

The screenshot shows a web browser window with the address bar displaying a local file path. The page has a light blue header with a film icon and the title 'Películas'. Below the header is a pink navigation bar with the text 'Menú de Secciones'. The main content area has an orange background and contains a form for adding movie data. The form includes a green 'Subir Recurso' button on the left and a series of input fields on the right: 'Titulo', 'Genero' (dropdown), 'Subgenero' (dropdown), 'Fecha estreno' (date picker), 'Duracion', 'Director', 'Pais' (dropdown), 'Protagonista', and 'Tipo de archivo' (dropdown). Below these fields is a large text area for 'Descripcion' and a 'Enviar' button. At the bottom of the page is a yellow footer bar with the text 'Contacto' and 'Como se hizo'.

Recursoseccion1.html

Como siguiente página en el guion es el recursoseccion.html. Pensé en varios diseños, como hacer una tabla, pero con lo aprendido en las anteriores paginas me decanté por un modelo de cajas individuales.

La cabecera, el menú de navegación y el pie de página es igual que las anteriores páginas.

Después del menú de navegación y antes de la parte central tenemos el título. En html esta encapsulado con una etiqueta *h2*. En css, tenemos las típicas sentencias que llevamos utilizando desde el principio como son: *display* con valor de *flex* para convertir el bloque al tipo flex, *align-self* y *justify-content* ambos con valor *center* para centrar tanto el contenido como el bloque, *background-color* para darle color de fondo y *margin-bottom* para ajustar el margen de la parte de abajo.

Para el centro lo hemos encapsulado en un section y lo hemos dividido en filas y a su vez en los distintos recursos. En css, para que el conjunto se vea como nos muestra el ejemplo del guion, hemos utilizado el atributo *display* con valor de *flex* junto con *flex-direccion* con valor de *column* así cada fila se coloca una encima de otra, dando la sensación de una tabla.

En el caso de las filas las hemos encapsulado cada una en un *article* en html. En css le hemos dado el atributo de *display* con valor de *flex* y el atributo *width* para se expanda al 100% del bloque. No hemos usado el atributo *flex-direccion* porque por defecto tiene un valor de *row* que es lo que queremos para las filas que cada recurso se ponga un al lado del otro horizontalmente.

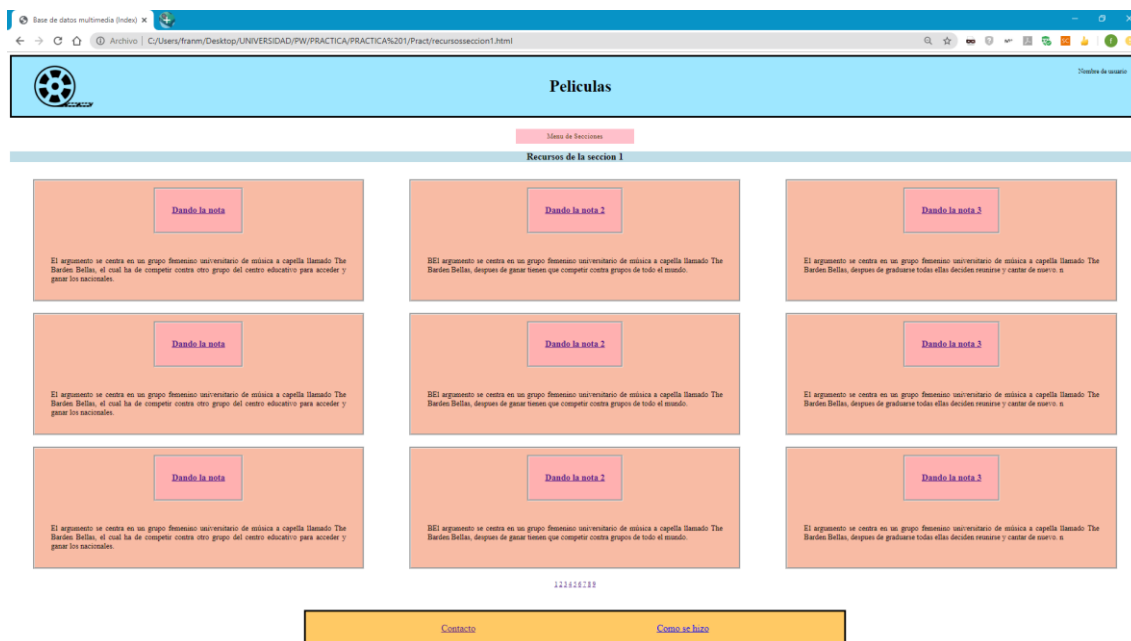
Para los recursos seguimos con lo mismo establecido anteriormente, esta vez siguiendo la jerarquía están encapsulados en etiquetas *section*. Dentro contienen una etiqueta de título *h3* que a su vez contiene una etiqueta *a* que guarda el enlace al recurso descrito. Después de la etiqueta del título tenemos una etiqueta *p* que contiene una pequeña reseña del recurso. En css, tenemos varios atributos para cada uno. Para el bloque de recurso tenemos los tributos: *display* con valor de *flex* y el atributo *width* para se expanda al 33% del bloque, *flex-direccion* para que sus componentes se ordenen de manera horizontal, *border* con el valor *groove* para darle un formato al borde y así como *margin* para darle un margen. Para el título hemos usado varias cosas del anterior como: *display*, *border*, *margin*; pero también hemos añadido: *max-width* para darle un ancho máximo, *font-size* para darle un tamaño más grande a la letra, *text-align* con valor de *center* para centrar el contenido y con ayuda de los valores de *margin* también la caja, así como *padding* para darle un valor a este. Para las sinopsis del recurso que se etiqueta con *p*, hemos usado algunas de las anterior como *margin*, *font-size*, *padding* y *text-align* pero esta vez con valor de *justify* para justificar el texto.

Hay que decir que el centro tiene tres *article*, uno por fila, y que dentro de estos hay 3 *section*, que representa los 3 recursos por fila.

Por último, debajo del centro y antes del *footer* tenemos un *section* que encapsula a los número de “páginas” que listan los recursos. Para ello en html, dentro del *section* tenemos un párrafo que se encapsula con la etiqueta *p*, que contienen los enlaces identificados por la etiqueta *a*. Para css, solo hemos usado los atributos de *text-align* con valor de *center* para centrar el texto y *margin-bottom* para definir el margen inferior del elemento.

Por último, está el *footer* que como hemos dicho antes es igual tanto en html como en css a los anteriores.

El formato elegido es el más parecido al del guion, con los colores elegidos de la paleta de tono pastel.



Recurso1.html

Para esta página hemos usado de referencia la de *altarecruso.html* ya que es igual exceptuando que esta rellena y que no tiene el botón de enviar si no dos enlaces al recurso anterior y al siguiente. Por lo que voy a comentar lo cambios ya que el código es igual exceptuando algunas partes o algunos parámetros como el *margin* o el *padding* de algunos elementos.

La primera diferencia en el html la encontramos en que en el centro arriba izquida ya no tenemos un input de tipo *file*, si no un enlace al recurso, por ahora vacio ya que no tenemos manera de enlazarlo.

La segunda diferencia la encontramos que los campos del formulario en html es que ahora están rellenos y disponen de un atributo nuevo, *readonly*, el cual hace que solo se pueda leer en los *inputs* y *textarea*. Después en los select como no podemos usar este atributo, lo hemos subsanado con el atributo *disabled* que deshabilita la interacción y el atributo *selected* para seleccionar la opción elegida. En css tenemos todo igual excepto que añadimos para los select que están estado *disabled*, a través de *select:disabled* como hicimos en el menú, que el *color* tome el valor de *black* para asi aparezca en negro y no gris.

La tercera cosa que cambia es la botonera que hay debajo del formulario, donde antes había un botón de enviar, ahora hay dos enlaces al recurso anterior y siguiente. Estos dos elementos *a*, están encapsulado por una etiqueta *article* pero esta vez no están dentro del bloque de formulario, si no que están fuera pero centro del section del centro. En css, al ser esta parte distinta sí que cambia, aunque tenemos los mismos atributos de siempre. Los atributos que usamos son: *clear* con el valor *both* para limpiar los formatos heredados, *display* con valor de *flex* para hacer el elemento sea una caja flex, *justify-content* y *align-self* con valor *space-between* y *flex-end* para llevar los elemento a los extremos de la caja y alinearlos con el final, *margin* para alterar los márgenes, *width* para que no ocupara todo el bloque y *font-size* para cambiar el tamaño de la letra. También añadimos algo más de padding a los elementos *a* de la botonera con el atributo *padding*.

Siguiendo las directrices de diseño que hemos seguido en las anteriores eligiendo colores suaves y pasteles. La página queda tal que así:

The screenshot shows a web browser window with a URL bar indicating a local file path. The web page has a light blue header with a film icon and the title 'Películas'. Below the header is a pink button labeled 'Añadir la película'. The main content area is orange and contains a form with fields for Title, Genre, Subgenre, Release date, Duration, Director, Year, Protagonist, and Type of movie. A green button labeled 'Guardar' is on the left. Below the form is a description field and a pink button labeled 'Guardar'. At the bottom are two orange buttons labeled 'Cancelar' and 'Crear el dato'.

Index.html

2º intento

Con lo que había hecho me sentía lo suficientemente preparado como para intentar hacer las dos páginas que al principio me dieron tantos problemas, por lo que me puse manos a la obra.

Decir de ejemplo cojí la cabecera que he estado haciendo durante todas las páginas.

El documento html está dividido en Sección de Arriba, Centro (Izquierda y derecha) y parte abajo.

Sección arriba

La sección de arriba esta dividida en 3 componentes en html: La imagen del icono del gestor bajo una etiqueta *img*, el nombre del gestor con la etiqueta *h1* y por último el formulario de inicio sesión encapsulado por la etiqueta *form*.

Este formulario tiene 3 partes: el usuario con *input* de tipo *text*, el de contraseña un *input* de tipo *password* que oculta la que hay escrito y por último un botón envía el formulario, aunque en este caso es un botón que lleva un enlace para ir al gestorbd.html.

En css, es bastante parecido al hecho en las anteriores páginas. En la etiqueta de estilo de esta parte es igual salvo que esta vez no tenemos el atributo *margin-bottom* para modificar el margen. En la imagen lo tenemos igual salvo que modificamos el color de fondo con *background-color* a un tono de azul, y aumentamos el padding a 1%. Para el texto, *h1*, tenemos los atributos: *display* con valor *flex*, *justify-self* con valor de *center* con margin y padding automáticos para mantener el elemento en el centro (como podemos ver es otra forma de mantener en el centro un elemento), y *font-size* para modificar el tamaño de la letra. Para el formulario hemos usado: *display* con valor *flex*, *align-self* con valor *center* para centrar la caja

flex en el medio de la sección, *flex-direction* con valor *column* para poner los elementos del formulario de forma horizontal.

Centro

Para el centro nos encontramos ante una división en dos partes, izquierda y derecha, ambas divisiones encerradas en un *section*. En *css*, tenemos los atributos de *display* con valor de *flex* y *background-color* para el color de fondo.

Para el centro izquierda encapsulado en *html* con un *article*, solo encontramos un único elemento, que es una imagen bajo la etiqueta *img*. En *css*, para el bloque tenemos los atributos: de *display* con valor *flex*, *width* para señalar el ancho que ocupa este bloque en este caso el 43% y el atributo *border-right* para dar formato al borde derecho, en este caso de *groove* (tipo de línea) y un color en formato *rgb*, es decir, cuanto de rojo verde y azul tiene ese color.

Para el centro derecha en *html* lo tenemos encapsulado en otro *article*, que a su vez contiene varios elementos, como es el título de la sección, es decir, una etiqueta *h2* y un conjunto de *sections* que representas a cada fila, es decir a las bibliotecas dadas de alta en el gesto. Cada *section* de la fila contiene dos elementos, una imagen de la película (elemento *img*) y un párrafo con el título (elemento *p*).

En *css* en el centro derecha para el bloque completo los atributos de *display* con el valor *flex* para convertir el bloque, el atributo *width* para decir que lo máximo que se puede expandir el bloque es el 65%, *background* con el valor de *lightskyblue* para colorear el fondo de ese color, *flex-direction* con el valor de *column* para desarrollar los elemento de forma vertical y *border-left* con los valores de *groove* y *rgb(0, 140, 255)* que son los mismos que para el centro izquierda pero esta vez puestos en el bode izquierdo en vez de en el derecho. Para el elemento *h2*, es decir el título de esta parte, tenemos *margin* para controlar el margen, *font-size* para cambiar el estado de la letra y *display* con el valor *flex* y *justify-self* con el valor de *center* para centrar el elemento.

Para el elemento fila tenemos algunos de los anteriores iguales como es *display*, *margin* (aunque con distintos valores numéricos), *width* (aunque con distintos valores numéricos) y *background-color* (con un color distinto). Añadimos a la lista *flex-direction* con el valor *row* para que los elemento se coloquen de manera horizontal. Dentro de fila el formato de imgane contiene distintos atributos para controlar la altura y ancho como hemos dicho anteriormente, como son *height* y *widht* pero añadiendo esta vez su versión *max-height* y *max-width* estableciendo un máximo en lo que se puede expandir. También tenemos un texto, *p*, para el tenemos también otros atributos que hemos usado anteriormente como son *display*, *margin*, *font-size*. Pero también otros como *align-self* con valor de *flex-start* para que se alinee con el principio del bloque o *justify-content* con el valor *end* para llevar el contenido hasta el final del bloque.

Footer

En *html* el footer no es muy diferente al pospuesto en las otras páginas salvo que esta vez hay una etiqueta *p* entre las dos con un guion para seguir el esquema propuesto por el guion.

En *css* sí que tiene algunas diferencias con el anterior. Para el bloque entero tenemos loa atributos típicos ya de *display* (*flex*), *background.color* (*rgb(255, 201, 100)*), *font-size* (*x-large*), *paddin* (*auto*) y *margin* (*auto*), con las mismas funciones que expusimos anteriormente. Incluimos los atributos *justify-content* con el valor *center* para centrar el contenido en mitad del

bloque y *border-style* con el valor *solid* para definir el estilo del borde que rodea a la caja. Para los enlaces, *a*, y los párrafos, *p*, que están dentro definimos los márgenes con *margin* cada uno por separado, aunque se puede hacer todos en una sola sentencia.

Con un formato lo más parecido al dado como ejemplo, pero añadiendo las directrices decorativas que hemos dictado al principio queda tal que así:



Gestorbd.html

Cogiendo de referencia la página anterior empecé a hacer la siguiente página, gestorbd.html. En este caso he cogido bastantes cosas de la página anterior como es su esqueleto de ordenación en html, ya que la pagina es igual salvo algunas modificaciones.

Sección de arriba

Para la parte de arriba no hemos cambiado nada en el css ni el html salvo en la parte del formulario, que se sustituye por el nombre de usuario y las opciones de usuarios debajo. En html, para ello hemos hecho un *section* que en capsula a esta parte. Dentro encontramos 3 elementos. El primero es un *article* que tiene dentro un párrafo, *p*, con el nombre de usuario. En segundo lugar, tenemos un párrafo, *p*, con el título “opciones de usuario”. Y en tercer y último lugar, un *article* que incluye dentro 3 párrafos, *p*, con enlaces, *a*, a las distintas opciones.

En css, como hemos dicho los demás elementos no cambian, se introducen cambios en los elementos que hemos añadido. Para el bloque *section* que encierra todo, le hemos dado los siguientes atributos: *display* con el valor *flex*, *margin* para controlar el margen, *padding* para controlar el padding, *flex-direction* con el valor de *column* para desarrollar los elementos de forma vertical, *width* y *max-width* para definir el ancho y el ancho máximo que se puede ocupar.

Para el *article* que contiene el nombre tenemos el ya recurrente *display* con valor *flex* así como el *margin* y el *padding* ambos a 0%, después con *margin-left* y *margin-right* redefinimos los márgenes a un 20%, con *border-style* definimos el estilo del borde, esta vez a *dotted* (punteado) y por último ponemos un color de fondo con *background-color*. También hemos aplicado algo de formato sobre la etiqueta *p* que hay dentro con los atributos de *font-size* para aumentar el tamaño de la letra y con *margin*, *margin-left* y *margin-right* para modificar los márgenes.

Después en el formato de la etiqueta *p* que se refiere al título de las “opciones de usuario” le hemos aplicado los atributos de *display* con el valor de *flex* y las etiquetas *margin-top*, *margin-left* y *margin-bottom* para modificar los márgenes.

A continuación, tenemos el *article* cuyo formato no tiene más que el atributo *display* con valor *flex* y *margin* con valor *auto* para que el margen se calculó automáticamente. El formato de los párrafos, *p*, que hay dentro tienen los mismos atributos, pero en este caso el valor de *margin* es de 2%.

Centro

El centro no hay cambiado mucho, se sigue dividiendo en dos *articles* izquierda y derecha.

El centro izquierda, en html cambiamos la etiqueta *img* que hay dentro del *article*, por la etiqueta *a* que contiene a su vez otra etiqueta *h2*, es decir, le ponemos un enlace al título. En css, no cambia mucho, los atributos del *article* son los mismos y lo que cambia son los atributos del *img* que ahora son los del *a* y son: *display* con valor *flex*, *margin* con el valor *auto* para hacer automáticos los márgenes ya si hacer que se centre automáticamente y *font-size* que cambia el tamaño de letra por uno más grande.

En el centro derecha, la mayoría de html y css es igual salvo que ahora tenemos algunas cosillas más y modificamos algunas cosas más en css. Para html, seguimos guardando todo en un *article* y el estilo de este en su conjunto, el css de esa parte, no ha cambiado. Dentro del *article* vemos que ahora todas las filas están dentro de un mismo *section*. Antes de este tenemos el título de esta parte bajo la misma etiqueta que en la página de *index.html*, la etiqueta *h2*. En css este ha sufrido varias modificaciones y nos quedamos con los atributos *display* con el mismo valor, cambiamos el *justify-self* por *justify-content* con el mismo valor ya que esta vez centramos el contenido no el bloque, *font-size* con el mismo valor y *margin* para dar formato solo al *top* y a *bottom*.

Dentro del *section* que encapsula a las filas, tenemos tres *articles* para agrupar a cada una de las filas. En su interior tenemos una etiqueta *img* al igual que la otra página, debajo tenemos un *section*, que agrupa al párrafo, *p*, con el título de la película que a su vez es un enlace, *a*, y a un *article* que contiene las opciones para esa biblioteca de igual forma que el título, es decir, un párrafo *p* y en su interior un enlace *a*.

Para el css, primero para el estilo de esa lista de filas que se han agrupado en el *section*, tenemos los siguientes atributos: *display* con el valor *flex*, *flex-direction* con el valor de *column* para desarrollar los elementos de forma vertical, *overflow-y* con el valor *scroll* para añadir la barra de movimiento lateral como hay el guion, *margin-top* para controlar el margen superior y por último *max-height* que dice lo máximo en altura que se puede expandir. El estilo de la fila, es decir, el artículo que hay dentro no ha cambiado es el mismo al igual que el de la imagen que solo ha sufrido un cambio en el que se le ha añadido el atributo *align-self* con el valor *center* para poder centrar la imagen, también ha sufrido cambios los valores de *height* y *width*. El formato del *section* que agrupa la parte derecha ha cambiado ya que ahora contamos con más elementos. Para el bloque nos encontramos que tiene los atributos: *display* con el valor de *flex*, *flex-direction* con el valor de *column* para desarrollar los elementos de forma vertical y por último *margin-right* y *margin-left* para definir los márgenes de los lados. Para la etiqueta *p* que hay dentro tenemos los atributos de: *display* con el valor de *flex*, *font-size* para modificar el tamaño de la letra y último *margin-bottom* y *margin-top* para definir los márgenes de los abajo y arriba respectivamente. Debajo tenemos el artículo que contiene a las etiquetas *p* con las opciones. Ese bloque tiene las características de *display* con el valor de *flex* y *margin* en la que definimos todos los márgenes uno detrás de otro en una sola sentencia. La etiqueta *p* de dentro también tiene sus atributos que son: *display* con el valor de *flex*, *font-size* para modificar el tamaño de la letra y como en bloque un atributo *margin* para modificar todos los márgenes.

Footer

Como hemos dicho antes el footer es el mismo en html y en css que en la página anterior, *index.html*.

Siguiendo con el formato de guion y el que hemos establecido al principio la página tiene el siguiente aspecto:



Contacto.html

Para crear esta página me he basado en las dos anteriores. En el guion solo pone los datos de contacto, pero yo he añadido también un formulario para contactar por email.

La página la hemos dividido en 3 secciones Arriba, centro (izquierda y derecha) y pie.

Parte de arriba

Para la parte de arriba hemos cogido la misma cabecera que las dos primeras páginas, *index.html* y *gestodb.html* y le hemos quitado la parte de la derecha para que sea independiente de si estas registrado o no, por lo que es en html y css igual que estas, pero quitando lo anteriormente dicho.

Centro

Para el centro hemo seguido con la misma división de siempre, un section que contiene dos *articles* uno para la parte izquierda y otro para la parte derecha. Como css, el de section tiene los siguientes atributos: *display* con el valor *flex*, *flex-direction* con el valor de *row* para desarrollar los elementos de forma horizontal y por último ponemos un color de fondo con *background-color*.

La parte izquierda esta encapsulada en un *article*, dentro tenemos un título de sección, es decir, una etiqueta *h2*, seguido de este tenemos una etiqueta para definir un formulario. Este contiene varios *inputs* y un *textarea* así como sus *label* correspondientes. Los inputs lo tenemos de tipo *text* para el nombre, *email* para el correo electrónico y *submit* para el botón de enviar. En css, el bloque tiene los siguientes atributos: *display* con el valor *flex*, *flex-direction* con el valor de *column* para desarrollar los elementos de forma vertical y *width* para marcar el ancho puede ocupar esta parte, en este caso el 50%. Para la etiqueta *h2* que hay dentro le definimos los atributos de *display* con el valor *flex* y *margin* con los valores de cada uno de los márgenes. El formulario como bloque tiene los mismos atributos que *h2*, pero añadiéndole *flex-direction* con el valor de *column* para desarrollar los elementos de forma vertical y *background-color* para darle color al fondo. Dentro del formulario le damos formato a los inputs, textareas y label que contienen los atributos: *display* con el valor *flex*, *margin* con los valores de cada uno de los márgenes y *font-size* con valor *large* para definir el tamaño de la letra. También damos formato al botón, lo hacemos de la misma manera que lo hacíamos en la página de *altarecurso.html* para poner el botón en posición de subir recurso, pero esta vez cambiamos el *type* por *submit* y le ponemos los atributos de *display* con el valor *flex*, *background-color* para darle color al fondo y *justify-content* con el valor de *center* para centrar el contenido, en este caso la palabra dentro del botón.

Al igual que el centro izquierda, el centro derecha esta encapsulado en un *article*. Dentro encontramos un título etiquetado con *h2* y después una lista formada primero una etiqueta *ul* y dentro cada elemento de la lista entre una etiqueta *li*. En css, tenemos un formato para *article* que forma el bloque derecho tiene los atributos: *display* con el valor *flex*, *width* para marcar el ancho ocupa esta parte al igual que ocurre con la parte derecha esta ocupa el 50% y *flex-direction* con el valor de *column* para desarrollar los elementos de forma vertical. Después para la lista, etiqueta *ul*, añadimos el atributo *font-size* con valor *x-large* para poder poner la letra al tamaño deseado.

Footer

Como hemos dicho anteriormente el pie de página utilizado es el mismo que en las paginas anteriores para mantener el estilo, por lo que no cambia su código html ni css con respecto a las anteriores.

Siguiendo los requisitos del guion y los estilos definidos, la página tiene el siguiente aspecto:



Altagestor.html

Para hacer esta página, reciclé varias partes de código para la apariencia salvo ciertos cambios como el color o el estilo de la parte superior e inferior del *altarecurso.html*. La parte superior e inferior junto con sus estilos fueron cogidos tal cuales de *gestordb.html* ya que son iguales para todas las páginas. Para no repetir la misma explicación recurrentemente vamos a explicar lo que ha cambiado. La estructura no ha cambiado, tenemos la división de parte de arriba(*header*), centro(*section*) y pie de página(*footer*).

Sección de arriba

Como hemos dicho al principio, esta página lleva el mismo código html y css que la página *gestordb.html* ya que son exactamente iguales.

Centro

Para el Centro nos fijamos y cogemos como plantilla en la estructura de *altarecurso.html* y su correspondiente estilo de *altarecurso.html*. Dividimos la parte central en 3 partes: Título, Centro arriba (Izquierda y derecha) y Botonera. Para el css del bloque tenemos las mismas sentencias de la página que usamos como plantilla(*altarecurso.html*) salvo que cambiamos el valor de *background-color* por otro color para diferenciarlo de otras altas.

Lo primero que nos encontramos después del *section* del centro es una etiqueta *h2*, que contiene el título de la página. En css, esta etiqueta no estaba contemplada por lo que le damos los siguientes atributos: *display* con el valor de *flex* y *margin* con valor de *auto* para poner el título centrado en el centro del contenedor.

Seguidamente tenemos el formulario encapsulado con la etiqueta *form*, se envía a través del método POST. Dentro tenemos una etiqueta *article* para encapsular lo que sería el centro arriba. A continuación, tenemos un *section* que delimita el centro arriba izquierdo y que contiene dos etiquetas, *label* para etiquetar el campo y el *input* correspondiente con tipo *file*, para subir una imagen del usuario. En el css, tenemos exactamente las mismas etiquetas que la plantilla que cogimos de ejemplo, lo único que como la página ha cambiado de tamaño hemos tenido que modificar el “truco” que usamos para el botón para subir el archivo y posicionarlo en el sitio requerido, para ello aumentamos el valor del atributo *top* de la marca *input[type="file"]*.

Después del centro arriba izquierda tenemos otro *section* que encapsulando el centro arriba derecho. Este contiene los campos correspondientes a los datos del usuario. Los conjuntos de *label* e *input* está dentro de una etiqueta *article* para poder manejarlo fácilmente en el css. Tenemos 3 tipos de *input*:

- ***inputs con tipo text*** para el nombre y apellidos.
- ***inputs con tipo password*** para la contraseña.
- ***inputs con tipo email*** para el correo electrónico.
- ***inputs con tipo file*** para el archivo. Aunque este está en la section de arriba.

En css, no cambia absolutamente nada con respecto a la referencia.

Todos los campos de formulario llevan el atributo *required* para que no se pueda enviar el formulario sin que estén todos rellenos como se exige en las especificaciones del proyecto.

Antes de acabar el formulario y cuando termina el section de centro arriba. Tenemos un *article* que en su interior guarda el *input* de tipo *submit*, que es el botón para enviar el formulario. En el css no cambia nada para el estilo de estos dos elementos.

Footer

El footer como hemos dicho al principio junto al section se cogen de *gestordb.html* tanto como el código como el estilo (de su correspondiente css).

Como apunte, pese a que la mitad del código se ha cogido de otras partes se han eliminado las partes no necesarias como el estilo del “centro de abajo” o de los *textarea* ya que estas etiquetas no existen en nuestro código html.

Para la apariencia para que fuera en conjunto hemos elegido el rojo pastel para lo que tiene que ver con las opciones de los usuarios. Por lo demás sigue las reglas de estilo que llevamos usando todo el proyecto. La página se tal que así:

Alta Gestor - Base de datos multi...

Archivo | C:/Users/fran/Desktop/UNIVERSIDAD/PW/PRACTICA/PRACTICAS201/Pract/altagestor.html

DigiBliblo Gestor

usuario

Opciones Usuarios

Edición Alta Borrado

Datos alta Usuario

Subir Imagen

Nombre:

Apellidos:

Email:

Contraseña:

Enviar

Contacto Como se hizo

Editargestor.html

Para esta página seguimos como ejemplo la página anterior. Es decir, es igual salvo por que ahora los campos están rellenos. Para rellenarlos (los *inputs*) se ha usado el campo *value* junto con el valor.

Como hemos dicho anteriormente el campo *input* de tipo *file* no puede tomar valor por medidas de seguridad, por lo que si el usuario quiere cambiarlo pues se rellenara, si no se dejara vacío y par que eso ocurra se le ha quitado el atributo *required* haciéndolo así no obligatorio. Una cosa parecida ocurre para la contraseña por lo que hemos optado por la misma solución que el caso anterior.

También se ha cambiado la etiqueta *h2* que había dentro del *section* del centro, para que quedase reflejado de que eso era la página de editar

En el archivo *css*, se optado por tomar el mismo que para la página anterior ya que contiene básicamente lo mismo. Como sigue siendo gestión de usuario el centro tiene el color rojo, siguiendo así el estilo de colores de las páginas. La página se tal que así:

Editar Gestor - Base de datos multi...

Archivo | C:/Users/fran/Desktop/UNIVERSIDAD/PW/PRACTICA/PRACTICAS201/Pract/editargestor.html

DigiBliblo Gestor

usuario

Opciones Usuarios

Edición Alta Borrado

Datos Editar Usuario

Subir Imagen

Nombre: Fco Javier

Apellidos: Merchan

Email: franmerchan6@correo.ugr.es

Contraseña:

Enviar

Contacto Como se hizo

Borrargestor.html

Siguiendo las recomendaciones que dio el profesor, diseñe esa página con una lista central para seleccionar un elemento y enviarlo a través de un formulario, dando lugar a que ese elemento se borrado.

Siguiendo la fórmula de las anteriores páginas, he cogido varias cosas de otras páginas y las he puesto tal cual en esta, como es la sección de arriba(header), el footes y un poco el esquema y los estilos de centro. Para las dos primeras las cogí de *gestordb.html* y de su correspondiente hoja de estilos. Para el centro lo cogí de *altarecurso.html* y de su respectiva hoja de estilos.

Como vi que se repetían muchas cosas me decanté para hacer todas las páginas de borrado de elementos con el mismo css como experimento y ya que todas se parecían muchísimo. Por lo que el css de los códigos de *html* de las páginas de borrar es el mismo. Como las secciones de arriba y los colores de la parte central cambian le di dos soluciones. La primera para la sección es darle otro id distinto para las dos secciones (añadí un numero dos al final) y así conseguimos que las dos estuvieran en el mismo archivo. Para el segundo problema, saque el *background-color* de los atributos de centro y cree una clase para cada archivo (usuario, secciones, recursos y bibliotecas) y le el valor de *background-color* a cada una de ellas, pudiendo seguir los estilos de colores distintos exigidos por el diseño.

Sección de arriba

Como hemos dicho anteriormente, esta página lleva el mismo código html y css que la página *gestordb.html* ya que son exactamente iguales.

Centro

Para el centro esta vez tenemos una fórmula más sencilla que antes pero muy del estilo. Primero tenemos un tenemos el *section* que agrupa todo el centro, al cual aparte el *id* que ponemos siempre, le añadimos como hemos dicho antes una clase. Dentro del *section*, no encontramos con un título, *h1*, y seguidamente un formulario, *form* (se utiliza el método POST para enviar la información). En el estilo del bloque de centro es igual al de la plantilla(*formatoAltaRecurso.css*) pero con la diferencia que hemos explicado arriba de sacar el atributo *background-color* a una clase aparte. La etiqueta *h1* también tiene un estilo añadido: *display* con el valor de *flex* y *margin* con valor de *auto* para poner el título centrado en el centro del contenedor.

Dentro del formulario, nos encontramos con una etiqueta *select* a la cual le hemos añadido los atributos de *size* para aumentar el tamaño de opciones que muestra y *required* para que sea obligatorio rellenarlo antes de enviarlo. Dentro tenemos las etiquetas *option* con las distintas opciones, en este caso usuarios. Para el css del bloque del formulario hemos usado: *display* con el valor de *flex*, *width* para acotar el ancho que ocupa el formulario dentro del bloque central, *flex-direction* con el valor *column* para colocar los componentes dentro del formulario de manera vertical y *margin* con valor de *auto* y *2%* para que se centre y tenga margen arriba y abajo.

A continuación, antes del final del formulario tenemos un *article*, que en su interior guarda el *input* de tipo *submit*, que es el botón para enviar el formulario. En el *ccs*, como bloque *article* tiene los tributos: *display* con el valor de *flex* y *margin* con valor de *auto*, 2% y 1% para que se centre y tenga margen arriba y abajo. El *input* del botón tiene un atributo mas que es *font-size* que sirve para poner un tamaño específico a la letra.

Footer

El footer como hemos dicho al principio junto al *header* se cogen de *gestordb.html* tanto como el código como el estilo (de su correspondiente *css*).

Con la compartición del archivo de *css*, nos ahorramos tener más archivos a la vez que reutilizamos y no hacemos tanta redundancia de código, en contra complicamos un poco el código.

El estilo de colores y aspecto general de la página sigue con lo descrito anteriormente. La página se tal que así:



Crearbd.html

Para esta página hemos seguido el mismo estilo que hemos seguido en las páginas de alta de los elementos. Como punto de partida hemos cogido esta vez *formatoAltaUsuario.css* para el estilo, excepto el estilo de un *article*, y para el código hemos cogido *altagestor.html*.

La página siguiendo el habitual estilo, la hemos dividido en sección de arriba, Centro (Arriba y abajo) y parte de abajo.

Sección de arriba

Para la sección de arriba hemos cogido, tal cual, el *header* de la página que hemos seguido de referencia así como su estilo de su homónima.

Centro

Para el centro hemos escogido una etiqueta *section*. Dentro de ellas encontramos varios elementos. Primero encontramos el título de la sección con la etiqueta *h2*, el estilo de esta etiqueta es el mismo que el de la plantilla anteriormente mencionada al igual que el estilo del bloque de *section*, lo único que cambia es el color de fondo dándole un valor distinto al atributo *background-color*.

Después de la etiqueta *h2*, nos encontramos ante el formulario, que envía los atributos mediante el atributo *POST*, además todos los campos tienen el atributo *required* para que sean obligatorio rellenarlo siguiendo las exigencias del guion dado. Dentro del formulario, tenemos la división en varios *artículos*, estos son Centro arriba, Centro abajo y botonera.

Para el *article* de centro arriba, a su vez se encuentra dividido por dos *section* centro arriba izquierda, que contiene la pareja *input* (de tipo *file*) y *label* para subir el archivo, y centro arriba derecho, le cual contiene el *input* (de tipo *text*) y el *label* para introducir el título. En *css* sus atributos de estilo son iguales que la plantilla.

En el *article* de centro abajo tenemos un campo para escribir la descripción de la biblioteca. Esto está etiquetado como el dúo de etiquetas *label* y *textarea*. En el terreno de *css*, esta parte no está incluida en el documento de referencia que usamos (*formatoAltaUsuario.css*) por ello lo cogemos de otro sitio ya que ya lo hemos usado antes, del archivo *formatoAltaRecurso*.

Para el último *article*, que en su interior guarda el *input* de tipo *submit*, que es el botón para enviar el formulario. En el *css* no cambia nada para el estilo de estos dos elementos con respecto a la plantilla.

Footer

Al igual que ocurre con el *header*, lo cogemos tal cual de la plantilla porque ni su código ni su estilo cambian con respecto al anterior.

Salvo los cambios mencionados y alguno como cambiar los parámetros del botón del *input* para subir la imagen para centrarlo, es igual al de la plantilla. En relación con el estilo y *formato* usamos los que llevamos usando en todo el proyecto, pero esta vez cambiamos el color ya que cambiamos de elemento. En este caso hemos asociado el azul a la biblioteca, como el logo es un libro sobre fondo azul, así le damos sentido y continuidad. La página se tal que así:

Editarbd.html

Como hemos dicho en el anterior de editar gestor, la pagina es igual con el mismo estilo que el alta del mismo elemento, en este caso de bd. Aunque en este caso si que hay varios añadidos que diremos a continuación. Para no hacerlo repetitivo solo diremos las diferencias, las demás son iguales que *crearbd.html*.

Sección de arriba

Como hemos dicho antes la sección de arriba es la misma que *crearbd.html* al igual que su estilo.

Centro

Aquí ya si hay diferencias con la plantilla a seguir, *crearbd.html*. Lo primero es que hemos quitado por las razones que dimos cuando explicamos *editargestor.html* el *required* de subir la imagen. Al igual que hemos añadido valores a los campos del formulario con el atributo *value*. Hemos añadido un campo nuevo al formulario debajo del campo para poner el titulo de la biblioteca, en el cual pondrá escrito el propietario de la biblioteca. El estilo es el mismo que en la plantilla.

Dentro también del formulario, pero dentro del *article* de centro abajo, hemos añadido otro campo *textarea* el cual se introduce información adicional que tiene la biblioteca con los mismos atributos que el *textarea* de la descripción. Le añadimos también el atributo de *readonly* ya que este campo solo está de información y se podrá modificar. En el terreno del css, no tenemos muchas diferencias solo se ha sacado el color, *background-color*, del bloque de centro abajo y puesto solo sobre la etiqueta *label* para asi colorear solo la línea superior al *textarea*.

Footer

Al igual que la sección de arriba, no hay ningún cambio con respecto a la plantilla.

Al darme cuenta de que estas partes comunes son iguales y cogerlas, ha hecho que sea más rápido y fácil seguir un estilo común. Al igual que formato de *crearbd.html* sigue lo dictado para todo el proyecto, cogiendo el color de fondo para el centro el azul para diferenciarlo de las opciones de otros recursos. La página se tal que así:

The screenshot shows a web browser window with the address bar displaying a file path. The page has a green header with a book icon and the title 'DigiBiblio Gestor'. On the right side of the header, there are links for 'Opciones Usuarios' and 'Edicion Alta Borrado'. The main content area has a light blue background and is titled 'Datos edicion Biblioteca Digital'. It contains a 'Subir Imagen' button, a form for 'Titulo' (with 'Películas' entered) and 'Propietario' (with 'Administrador' entered), a 'Descripcion' field with a placeholder text, and an 'Informacion adicional' field with a placeholder text. At the bottom right of the form is an 'Enviar' button. Below the form is a yellow footer bar with links for 'Contacto' and 'Como se hizo'.

Borrarbd.html

Para elaborar esta pagina hemos copiado todo el código html de *borrargestor.html* ya que es igualito, pero sustituyendo en el titulo usuario por biblioteca y en el centro poniendo las bibliotecas disponibles en vez de los usuarios del sistema. También como dijimos antes utiliza un formato común con las otras páginas de borrar, pero cambiando los colores del centro de la manera que describimos anteriormente en *borrargestor.html*. La página se tal que así:

The screenshot shows a web browser window with the address bar displaying a file path. The page has a green header with a book icon and the title 'DigiBiblio Gestor'. On the right side of the header, there are links for 'Opciones Usuarios' and 'Edicion Alta Borrado'. The main content area has a light blue background and is titled 'Borrar Base de datos'. It contains a form with a dropdown menu showing 'Películas', 'Series', and 'Documentales'. Below the form is a 'Borrar' button. At the bottom of the page is a yellow footer bar with links for 'Contacto' and 'Como se hizo'.

EdicionRecursos.html

Para hacer esta página al igual que con las anteriores he cogido como base otra página, en este caso *altarecurso.html*. Utiliza la misma hoja de estilos que la pagina usada como base. Solo hay unos detalles los cuales no son iguales a la base, pero lo demás es el mismo código por lo que solo diré los cambios que he hecho sobre ella.

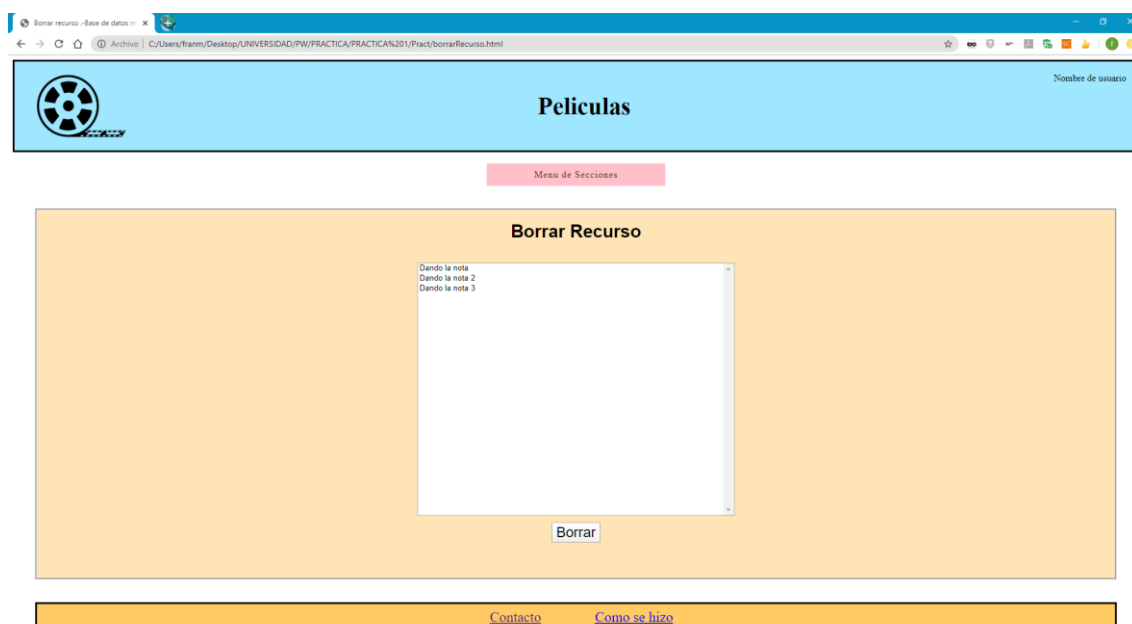
Primero como dijimos también en las otras páginas de edición de elementos, el elemento con la etiqueta *input* de tipo *file* quitamos el atributo *required* para que sea opcional subir el elemento ya que por cuestiones de seguridad del lenguaje no nos permite seleccionar uno de manera predeterminada. Los demás valores llevan un atributo extra que es *value* el cual le da un valor al campo. En los select utilizamos sobre la opción el atributo *selected* ya que no es posible seleccionar lo de otra manera. La página se tal que así:

The screenshot shows a web browser window with the address bar displaying the file path: `C:/Users/franm/Desktop/UNIVERSIDAD/PW/PRACTICA/PRACTICA%201/Pract/edicionRecursos.html`. The page has a light blue header with a film reel icon on the left, the title "Películas" in the center, and a "Nombre de usuario" field on the right. Below the header is a pink "Menu de Secciones" button. The main content area has an orange background and contains a form for editing movie details. On the left of the form is a green "Subir Recurso" button. The form fields are: "Titulo:" with the value "Dando la nota"; "Genero:" with a dropdown menu showing "Comedia"; "Subgenero:" with a dropdown menu showing "Musical"; "Fecha estreno:" with the value "23/06/2018"; "Duracion:" with the value "112"; "Director:" with the value "Jason Moore"; "Pais:" with a dropdown menu showing "EEUU"; "Protagonista:" with an empty text field; and "Tipo de archivo:" with a dropdown menu showing "Video". Below these fields is a "Descripcion:" section with a text area containing the text: "El argumento se centra en un grupo femenino universitario de música a capella llamado The Barden Bellas, el cual ha de competir contra otro grupo del centro educativo para acceder y ganar los nacionales." At the bottom right of the form is an "Enviar" button. Below the form is a yellow footer bar with two links: "Contacto" and "Como se hizo".

BorrarRecurso.html

Para hacer esta página copiamos el código de *borrargestor.html* ya como hemos dicho anteriormente para las paginas de borrado por que el código así como su contenido es el mismo, salvo el color del fondo, que en vez de los usuarios están los recursos de la biblioteca y que en el título en vez de usuarios pone recursos.

Aunque esta vez sí hay un cambio significativo ya que la cabecera (header) y el footer se han sustituido por los encontrados en *bd1.html* para mantener la concordancia de la página. También de esa misma referencia, *bd1.html* hemos obtenido la barra de navegación de las secciones. El código de estilo de esta se ha añadido y como dijimos anteriormente para solucionar los dos tipos de secciones de arriba se cambiaron los id, en este caso es el id normal, ya que el otro le añadimos un dos al final. La página se tal que así:



AltaSeccion.html

Para elaborar esta página me base en dos páginas, *crearbd.html* y *altarecurso.html*. De la primera, use la parte central ya que no tiene mucho más que añadir para crear una sección que el título, una imagen y una breve descripción, todo ello ya estaba hecho por lo que lo reutilice haciendo los cambios pertinentes como cambiar biblioteca o bd por secciones. Por lo que los atributos de esta parte central son los mismos que la que hemos usado en *crearbd.html*. Aunque hemos tenido que hacer los correspondientes ajustes sobre el botón de subir imagen como dijimos anteriormente con los atributos de *top* y *left* en *input[type="file"]*.

Para las secciones de arriba y el pie de página, incluido la barra de navegación las hemos cogido tal cual de la segunda referencia (*altarecurso.html*), tanto el código *html* como el código *css*, por lo que es igual.

Para el diseño de la página seguimos los estándares que hemos seguido durante todo el proyecto. Uno de los cambios no dicho arriba es que el color de fondo del centro lo hemos cambiado por un tono de morado siguiendo la regla de un color para las opciones de los distintos recursos. La página se tal que así:

EdicionSeccion.html

Para hacer esta página al igual que con las anteriores he cogido como base otra página, en este caso *altaSeccion.html*. Utiliza la misma hoja de estilos que la pagina usada como base. Solo hay unos detalles que no son iguales, pero lo demás es el mismo código y al igual que hemos hecho en las anteriores diremos los cambios que hemos hecho.

Primeramente, siguiendo lo hecho también en las otras páginas de edición, el elemento con la etiqueta *input* de tipo *file* quitamos el atributo *required* para que sea opcional, por cuestiones de seguridad del lenguaje, como ya dijimos múltiples veces y no nos permite seleccionar uno de manera predeterminada. Para los demás valores, llevan un atributo extra que es *value* el cual le da un valor al campo. En los textarea simplemente rellenamos entre las dos etiquetas.

Como diseño general seguimos los establecidos para todo el proyecto. Y como es para las secciones el centro es de color morado. La página se tal que así:

The screenshot shows a web browser window with the title 'Editar Seccion-Base de datos'. The URL bar shows a local file path. The page has a light blue header with a film reel icon and the text 'Peliculas'. Below the header is a pink button labeled 'Menu de Secciones'. The main content area has a purple background and is titled 'Datos edicion Seccion'. It contains a green button 'Subir Imagen', a text input field for 'Titulo:' with the value 'Accion', a large text area for 'Descripcion:' with the value 'Peliculas de accion', and a small 'Enviar' button. At the bottom, there is an orange bar with two links: 'Contacto' and 'Como se hizo'.

BorrarSeccion.html

Para esta pagina al igual que ocurría con las Paginas anteriores de borrar el elemento, repetimos la técnica igual que hicimos con *borrarrecurso.html*. Ya que en ese y este caso tenemos la misma sección de arriba y parte abajo, por lo que hacemos lo mismo. Esta vez la clase que cogemos para el fondo es la de *secciones* para ponerlo de color morado y seguir con el estilo visual. También modificamos el interior de la lista con las secciones en vez de los recursos de la biblioteca. La página se tal que así:

The screenshot shows a web browser window with the title 'Borrar Seccion-Base de datos'. The URL bar shows a local file path. The page has a light blue header with a film reel icon and the text 'Peliculas'. Below the header is a pink button labeled 'Menu de Secciones'. The main content area has a purple background and is titled 'Borrar Seccion'. It contains a list box with the following options: 'Accion', 'Comedia', 'Suspense', and 'Animacion'. Below the list box is a small 'Borrar' button. At the bottom, there is an orange bar with two links: 'Contacto' and 'Como se hizo'.