Ejercicio Práctica Final

Segundo Año Grado Superior DAW

Asignatura Diseño de Interfaces Web

09/02/2023

Mario Montalvillo Herrezuelo, Victor Puerta García

Índice

1. Introducción.
   1. Objeto del sitio web.
   2. Lenguaje y recursos empleados.
   3. Distribución.
   4. Requisitos.
   5. Licenciamiento.
2. Recursos
   1. Hardware.
   2. Software.
   3. Humanos.
3. Descripción del sitio.
   1. Estructura.
   2. Usabilidad.
   3. Accesibilidad.
   4. Pruebas
4. Conclusiones.
   1. Competencias profesionales, personales y sociales adquiridas.
   2. Reflexión sobre el trabajo individual y en equipo de forma cooperativa.
   3. Objeto del sitio web.

El objeto del sitio web, era crear una página con una interfaz que pudiera simular a la de una tienda genérica, haciéndola accesible, intuitiva y fácil de usar por los clientes

* 1. Lenguajes y recursos utilizados.

Hemos utilizado BootStrap junto a HTML para crear la página de index.html, para la creación de las páginas de detalles se ha realizado a través de HTML y CSS

* 1. Distribución.

La página contiene tres elementos, una es la página de inicio (index.html), luego tenemos la página de detalle (card.html) y por último tenemos la página de formulario(formulario.html) que es un formulario para que el cliente se registre en nuestra página web.

* 1. Requisitos.

Página de inicio: tiene que ser elaborada a través del framework BootStrap 5, el menú de navegación debe incluir un logotipo, junto a enlaces para navegar por el sitio web, el footer se debe mostrar cuando se llegue al final de la página no debe de estar anclado, deberá contener enlaces(el enlace de Contacto deberá de llevarnos a la página del formulario) e iconos de redes sociales. El cuerpo de la página contendrá al menos dos secciones compuestas por un mínimo de tres elementos con el tipo “card” las cuáles contendrán una imagen, un título, un texto descriptivo y un botón o texto de enlace que nos llevará a su página de detalle. Los elementos contenidos en las secciones se dispondrán de la siguiente forma:

* Dispositivos móviles en sentido vertical formando una única columna.
* Dispositivos con pantalla entre 768px y menores de 992px se dispondrán en dos columnas.
* Dispositivos de 992px y mayores en tres columnas.

Página de detalle:

Se deberá de poder acceder a ella desde alguno de los enlaces de la página de inicio, el nav y el footer deberán de ser los definidos por BootStrap, se deberá de realizar esta página a través de HTML y CSS, la definición de las filas y columnas deberá de ser de la siguiente manera:

* Columnas (columns) mediante fracciones restantes fr.
* Filas (rows) mediante la combinación de fr y span.

Se deberá de un incluir un BreadCrum, los textos deben de ser responsivos, la tipografía y la paleta de colores deberá de coincidir con nuestra página principal, todos los estilos aplicados estarán en la página styles.css

Página de Formulario de contacto:

Debe ser accesible desde el menú principal como desde el footer, como desde el BreadCrum de la página detalle. Su diseño debe ser a través de BootStrap, en concreto el layaout deberá ser clases row-cols-\*. Los textos utilizados deberán de ser responsivos, el diseño deberá de adaptarse dependiendo del tamaño del dispositivo.

* 1. Licenciamiento.

Las fotos que hemos elegido eran libres de derecho y además las hemos editado a mano para conseguir la foto que deseábamos para nuestra página web.

Las de los ordenadores juntando en paint3d esta imagen del ordenador (1) sacada de stock con diversos memes sobre mapaches encontrados en internet (2).

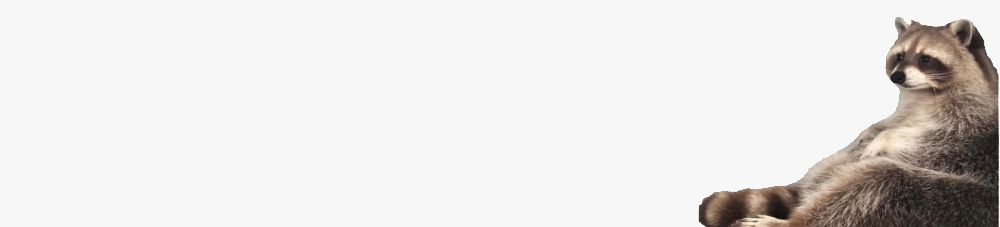




La de la cabecera tomando uno de estos memes y añadiendo el texto “MaPeChe” a su lado.



La de las cards son imágenes en las que se ha modificado el ancho también con paint3d para ser más anchas de manera que no se deformen tanto en grandes tamaños:



2.1 Hardware.

Ordenadores del instituto y personales.

2.2 Software.

VisualStudioCode con la extensión Windows Live server (entre otras), Word.

2.3 Humanos.

Trabajo realizado por los alumnos Mario Montalvillo Herrezuelo y Victor Puerta García, con la ayuda puntual del profesor del módulo de DIW Antonio Matachana Rodríguez.

3.1 Estructura.

La estructura de nuestro sitio web, sigue los requisitos iniciales, por lo tanto, nuestro sitio está estructurado de la siguiente manera. Una carpeta proyecto, la cual contiene una carpeta assets que a su vez contiene una carpeta imágenes que contiene todas las imágenes que utilizamos en nuestro sitio web, junto al logo y nuestro icono.

A continuación, tenemos la carpeta CSS, la cual contiene un fichero denominado style.css en él se encuentra todo lo necesario para la realización de las diferentes páginas de detalle.

Ahora pasamos a la carpeta lib, la cual contiene tres carpetas más la primera que encontramos es la carpeta css, que es la que contiene todos los CSS para poder utilizar BootStrap en nuestro proyecto. Otra carpeta que tenemos es la de fonts, la cual contiene las diferentes fuentes de BootStrap. Por último, tenemos la carpeta js la cual contiene los archivos js necesarios para la utilización de BootStrap.

El último elemento que tenemos en nuestra carpeta proyecto es nuestra página principal que es un archivo html, en concreto es index.html, tenemos la página de detalle (card.html) y la página de formulario (formulario.html).

Todas las páginas están comunicadas entre ellas gracias al header y al footer excepto card.html, a la que solo se puede acceder a través de index.html.

**3.1.2 Descripción de la interfaz.**

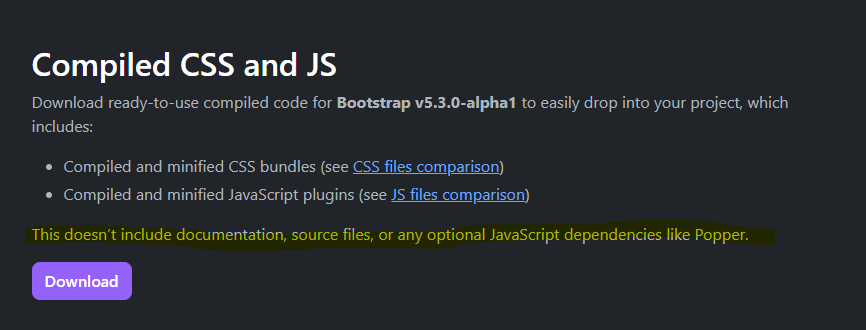
**3.1.2.1 Comportamiento del diseño.**

**Header→ index.html**

Comenzamos colocando en un <div class=”row”> dos columnas, una que fuera la imagen del logo y otra una lista normal (un <ul> con varios <li> que sólo contenían texto). Para que el menú quedara también en una línea dimos la clase row también al ul, pero investigando BootStrap encontramos la clase list-group y otras similares que existen para dar formato a menús de este estilo. Para el menú desplegable usamos también la página de BootStrap basándonos en el siguiente ejemplo para realizarlo.



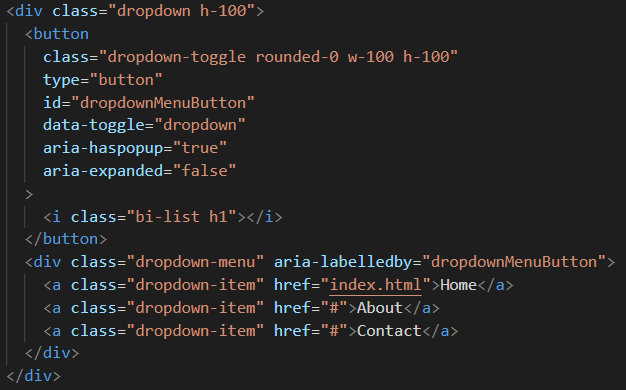
Sin embargo, al pasarlo a nuestra página descubrimos que el botón no funcionaba y encontramos que era a causa de la falta de un determinado script (popper, que debería encontrarse en bundle), la versión descargada no lo incluía.



Añadiéndole al final de la página de la siguiente manera se soluciona el problema:

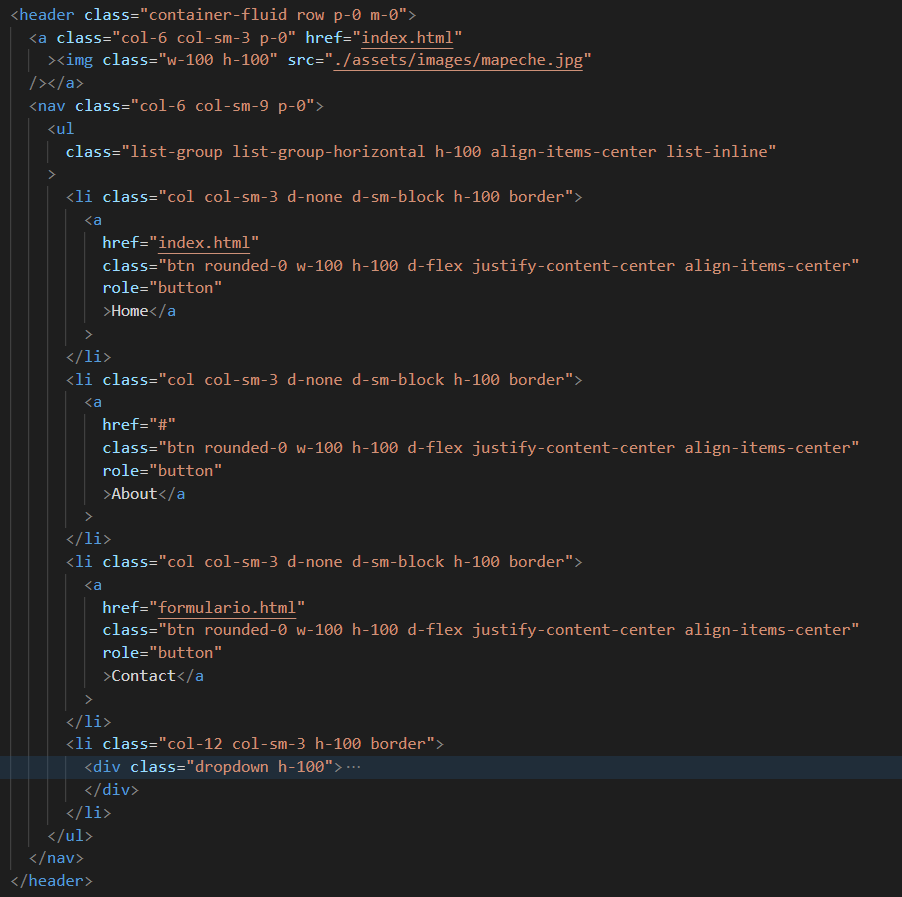


Ya funcionando, sustituimos los datos con los nuestros para darle el aspecto que buscábamos:



Dimos formato al resto de elementos, con columnas concretas según el tamaño de la pantalla para que tuvieran el aspecto requerido en la práctica y enlaces de modo que llevan a las demás páginas, y acabó de la siguiente manera (El div de la línea azul sería el mostrado el del menú desplegable de la imagen anterior):

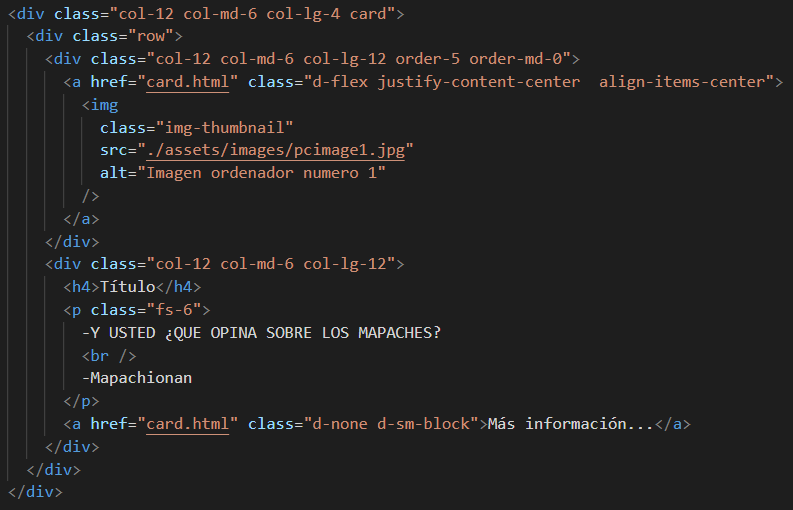
**Main→index.html**



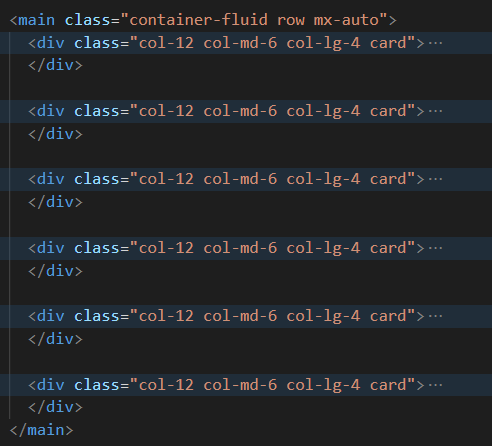
Este el código de las (6) cards. De base ocupan 12 (toda una línea) para los casos en el que el ancho es 576px, 6, para los que el ancho es mayor que 576 y menor que 992 (2 por fila), y 4 ara los que el ancho es mayor que 992 (3 por fila).

Dentro hay otra row con un elemento que incluye la imagen, que está dentro de un a para que funcione de enlace y otro que incluye todo el texto con las columnas diseñadas para adaptarse según el tamaño anteriormente mencionado.

El enlace cuenta con la clase d-none y d-sm-block para desaparecer pasado el breakpoint sm.

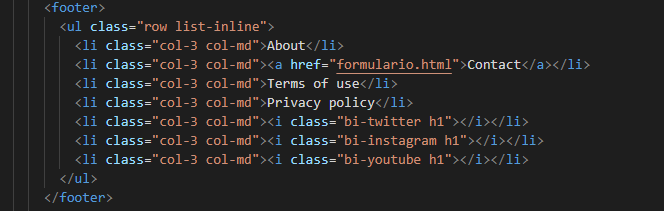


Cada div azul es una card de las descritas anteriormente:

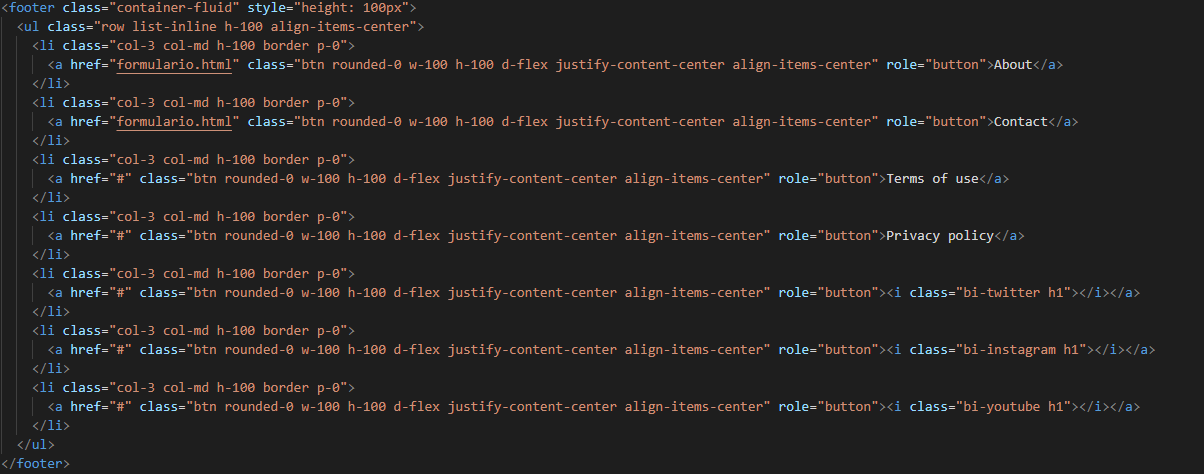


**Footer → index.html**

La estructura del footer era esta, en la cual solo aplicábamos la clase row list-inline para quitarle los estilos a la lista, además a cada li, le hemos indicado que en caso de que sea md(l≥768px ) ocupe una sola columna y ello solo se ajuste al tamaño y en caso de no ser medio, le indicamos que sean 3 columnas.



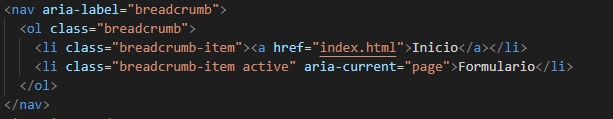
Una vez acabado lo básico del footer le aplicamos estilos adicionales para que quedará más visual.



Con el h-100 es que ocupe todo el alto, con el align-ítems-center para centrar el ul. Luego a los li les hemos aplicado un h-100 para que también ocupen todo el alto, le hemos un border y el p-0 es para darle un padding de 0, además a la etiqueta a hemos hecho que sea un botón, para que así donde haga clic el usuario le lleve a la página deseada.

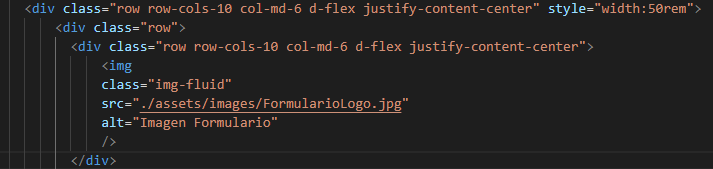
**Página Formulario.html**

Nota: El nav y el footer se ha utilizado el mismo que en el index.html explicado anteriormente.

Parte BreadCrum:

Hemos creado un BreadCrum, es una lista ordenada, en este caso compuesta por dos etiquetas “li” el primero es un enlace al index.html y el segundo indica en la página que se encuentra el usuario.

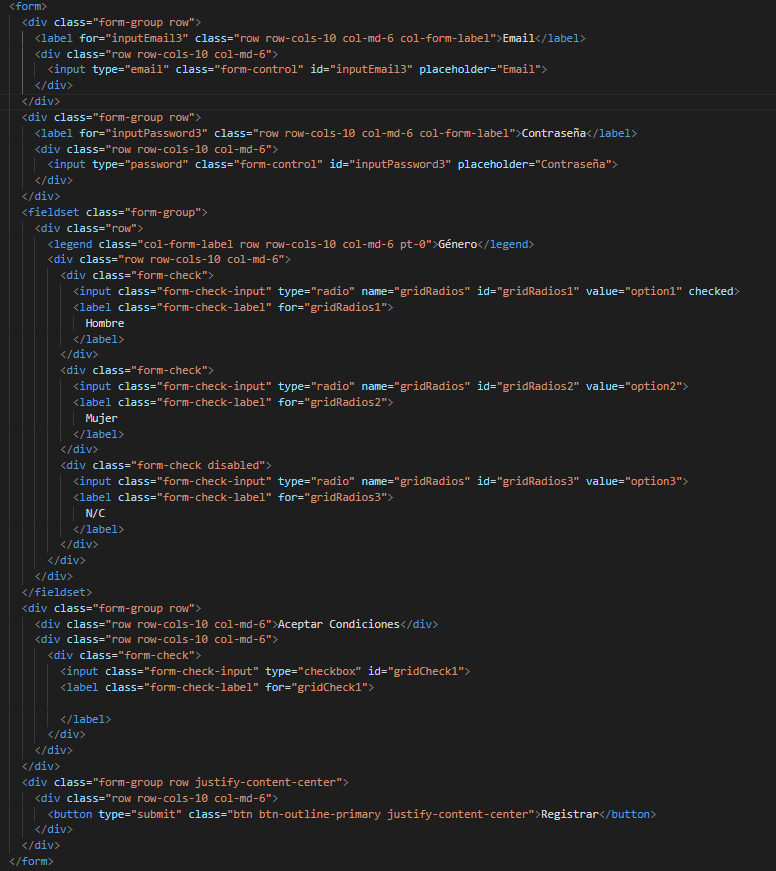
División de la página:



Hemos utilizado la clase indicada en las especificaciones que era row-cols, se ha indicado que sea contenido Flex, además hemos alineado los contenidos en el centro y hemos puesto un acho de 50rem.

En la imagen del formulario hemos utilizado la class img-fluid, además se ha indicado un alt describiendo la imagen para hacerlo más accesible.

Parte formulario:



3.2.2 Usabilidad.

¿?

3.2.3 Accesibilidad.

¿?

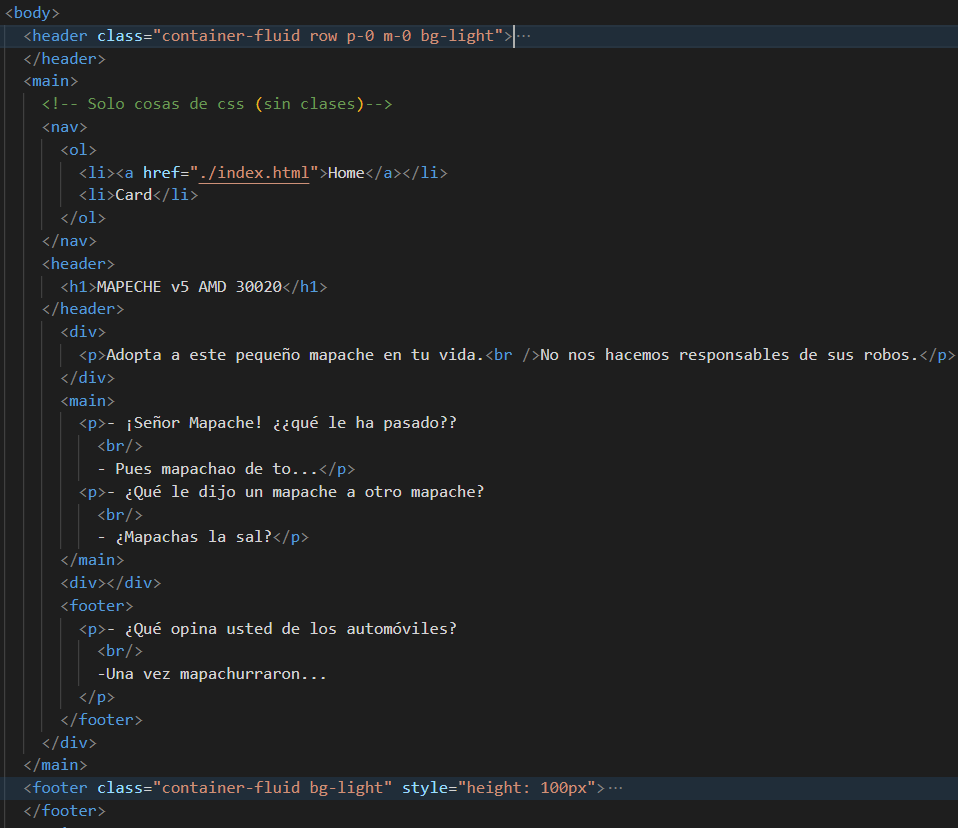
3.2.4. Pruebas Realizadas

¿?

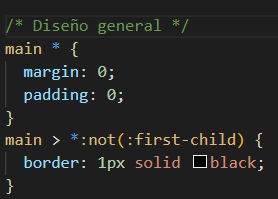
Hemos cogido un formulario ya hecho de BootStrap, pero le hemos adaptado a nuestras necesidades, además se ha añadido a mayores un estilo al botón para hacerlo más dinámico.

**Card.html.**

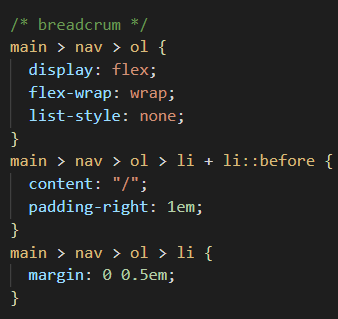
Como había que realizar esta página solo con css clásico, sin clases, creamos un archivo css específico.

El html tiene el siguiente aspecto:

Diseño general:

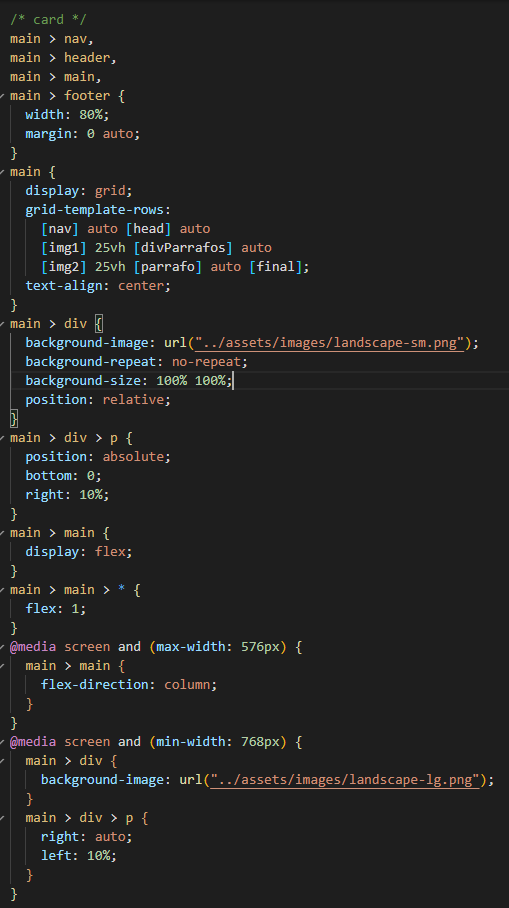


BreadCrum



Diseño de card:

Para que las imágenes se ajusten al tamaño del div y no al contrario hemos puesto que sean background-image, esto permite que, además, podamos insertar el texto encima. Para dar un mejor aspecto hemos hecho que haya dos imágenes, una para sm y otra para tamaños a partir de md y el párrafo dentro de la primera imagen se mueve a izquierda y derecha para ocupar el espacio vacío. (ver img página 15)



* 1. Usabilidad.

Nuestra página gracias al tamaño de sus elementos y lo intuitivo que es el diseño hemos conseguido esa usabilidad, ya que los enlaces se ven claramente que son enlaces y los botones del header y del footer son enlaces en sí, ya que da igual en que parte haga clic el usuario que le llevará a la página que desee el usuario.

* 1. Accesibilidad.

En el tema de accesibilidad lo que hemos tenido en cuenta ha sido dar detalle a las imágenes para las personas con problemas de visión tenga una descripción detallada de las imágenes que hemos utilizado.

Hemos elegido colores de alto contraste (blanco y negro) para facilitar la experiencia a las personas con problemas de visión.

* 1. Pruebas.

Las pruebas que hemos realizado se han centrado sobre todo en el comportamiento de la página ante los cambios de resolución y de navegador.

* 1. Competencias profesionales, personales y sociales adquiridas.

Las competencias que hemos adquirido han sido en varios aspectos, ya que lo primero que hemos tenido que hacer ha sido adaptarnos a trabajar con otra persona y ver cómo nos repartíamos el trabajo equitativamente, además de estar abiertos a los cambios que nos propusiera el compañero en nuestra parte del trabajo, ya que ambos debemos de estar contentos con el resultado final.

En cuanto a las profesionales hemos adquirido conocimiento de CSS, concretamente de la parte FLEX y GRID y del framework BootStrap, ya que gracias a ello nos hemos dado cuenta que nos reduce considerablemente el trabajo, aunque en ocasiones específicas requiere código externo, como a la hora de las hacer fonts responsive.

* 1. Reflexión sobre el trabajo individual y en equipo de forma cooperativa.

El trabajo individual nos ha servido a lo que hemos comentado anteriormente, como el conocer y aplicar en un proyecto “real” una aplicación tan potente como BootStrap, a repartirnos el trabajo con nuestro compañero y a estar abierto a los cambios que nos propusiera la otra persona en nuestra parte especifica del trabajo, ya que ambos debemos quedar satisfechos con el resultado de la página de nada vale hacer una web funcional si todos los trabajadores de esa página no estamos satisfechos con el trabajo final.

# Bibliografía

Vinculación a BootStrap 5

* BootStrap Page Builder, recuperado 2023-02-09 17:16:23 de 🡪 [Bootstrap · The most popular HTML, CSS, and JS library in the world. (getbootstrap.com)](https://getbootstrap.com/)

Página index.html:

* BootStrap Page Builder, recuperado 2023-02-10 11:20:34 de 🡪 <https://getbootstrap.com/docs/4.0/components/list-group/>
* BootStrap Page Builder, recuperado 2023-02-12 13:08:34 de 🡪 <https://getbootstrap.com/docs/4.0/components/dropdowns/>
* BootStrap Page Builder, recuperado 2023-02-10 11:28:34 de 🡪 <https://bootstrap.com>
* HABRA QUE PONER DE DONDE HAS SACADO LOS SCRIPTS PARA EL DESPLEGABLE

Página formulario.html

* BootStrap Page Builder, 2023-02-14 23:06:46 🡪[Forms · Bootstrap (getbootstrap.com)](https://getbootstrap.com/docs/4.0/components/forms/)
* BootStrap Page Builder, 2023-02-14 23:12:22 🡪[Breadcrumb · Bootstrap (getbootstrap.com)](https://getbootstrap.com/docs/4.0/components/breadcrumb/)
* BootStrap Page Builder, 2023-02-14 23:20:02 🡪 [Grid system · Bootstrap (getbootstrap.com)](https://getbootstrap.com/docs/4.0/layout/grid/)
* BootStrap Page Builder, 2023-02-15 00:03:02 🡪 [Images · Bootstrap (getbootstrap.com)](https://getbootstrap.com/docs/4.0/content/images/#responsive-images)
* BootStrap Page Builder, 2023-02-16 12:33:25 🡪 [Buttons · Bootstrap (getbootstrap.com)](https://getbootstrap.com/docs/4.0/components/buttons/)

Página card.html

BootStrap Page Builder, 2023-02-18 20:12:22 🡪 https://getbootstrap.com/docs/4.0/components/breadcrumb/)