



Página hecha por Nicole Lara Loupias.

# ÍNDICE

1. Inspiración

2. Guía de estilos

3. Secciones de la página

a. Barra de navegación

b. Inicio

c. Carta

d. Quiénes Somos

e. Contacto

- o Formulario de contacto

- o Nuestras tiendas

- o Horario

f. Footer

g. Producto

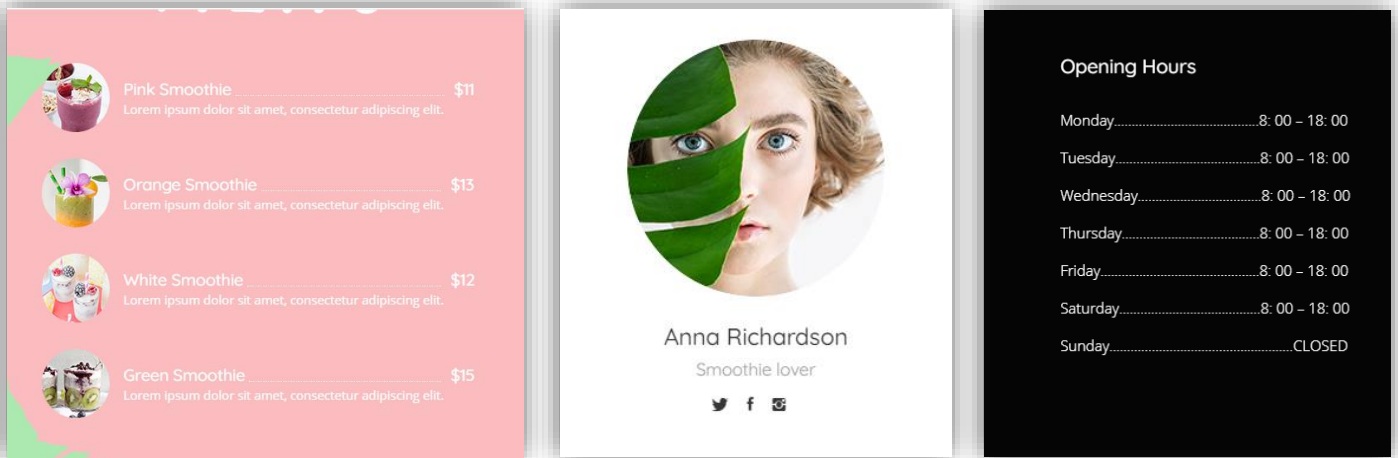
4. Custom.css

5. Listado de componentes de investigación

## 1. Inspiración

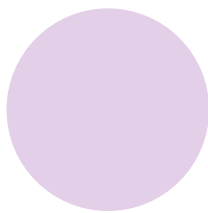
La página que he creado es de la tienda de smoothies llamada **Smoothie Boire**.

Para crear este diseño, me he inspirado de muchas webs que he visto. Algunas capturas de pantalla que hice:

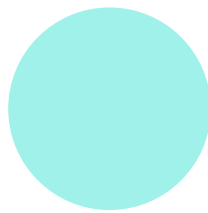


## 2. Guía de estilos

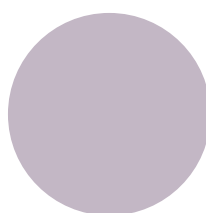
Su diseño se basa en colores morados y azul cyan.



#E3CFE7



#A0F1EA



#C3B7C5

A lo largo de toda la página hay de fondo un degradado que va entre esos dos primeros colores.

La fuente que utilizo para los encabezados y títulos es “**Livvic**” que la he importado desde Google Fonts.

El resto de la página utiliza la fuente “**Muli**”, también importada desde Google Fonts.

Livvic

El veloz murciélago hindú comía feliz cardillo y kiwi.  
El veloz murciélago hindú comía fel  
El veloz murciélago hindú c  
El veloz murciélag  
El veloz murc

Muli

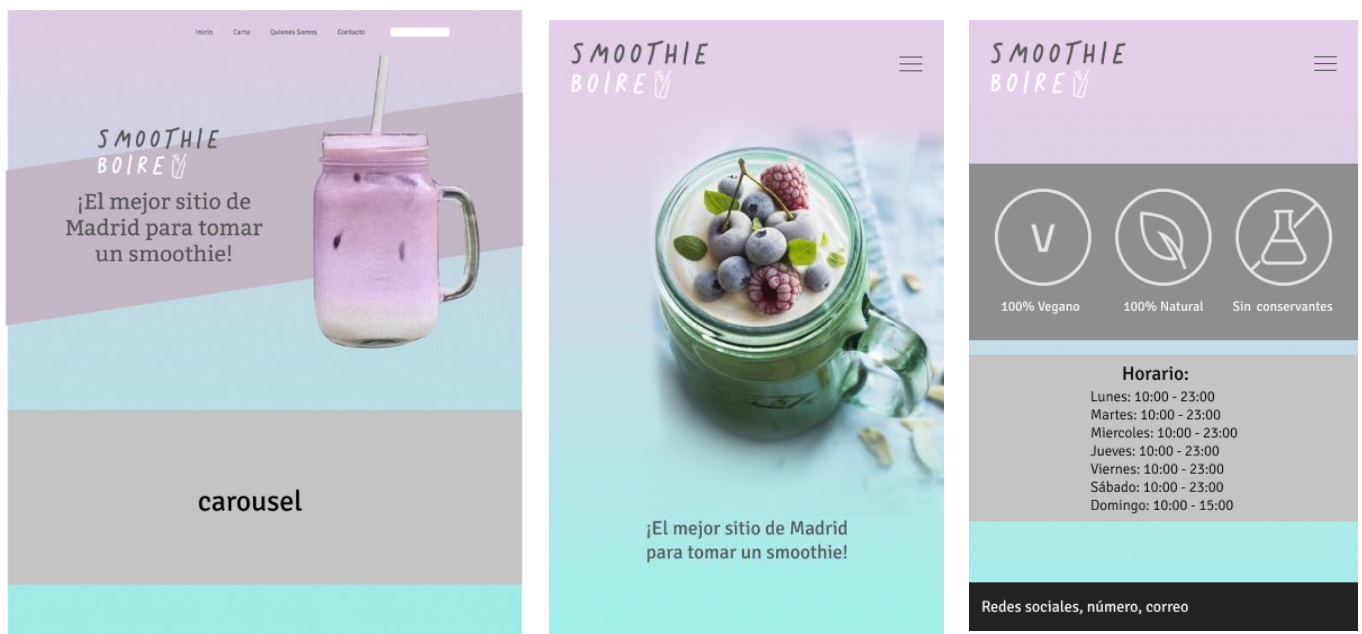
El veloz murciélago hindú comía feliz cardillo y kiwi.  
El veloz murciélago hindú comía fe  
El veloz murciélago hindú  
El veloz murciélag  
El veloz murc

Son dos fuentes parecidas pero una es un poco más redondeada, lo cual era el estilo que estaba buscando.

También hay que mencionar la fuente que he utilizado para hacer el logo. Se llama "CHARLESTON".

EL VELOZ MURCIÉLAGO HINDÚ COMIÓ FELIZ CAR.  
EL VELOZ MURCIÉLAGO HINDÚ C  
EL VELOZ MURCIÉLAGO  
EL VELOZ MURC  
EL VELOZ M  
EL VELOZ

Estas son algunas maquetas que hice antes de hacer la web:



### 3. Secciones de la página

La página está dividida en las siguientes páginas/secciones:

- Inicio
- Carta
- Quiénes somos
- Contacto

En este apartado explicaré la estructura HTML de cada página habiendo usado las clases de bootstrap.

## Barra de navegación

Es un elemento común que está en todas las páginas y las conecta entre sí mediante enlaces.

Se hace poniendo la clase `navbar` y `fixed-top` para que se mantenga arriba de la web en todo momento. También se añade la clase `navbar-expand-md` para que salga entera a partir del tamaño md. Así, conseguimos que cuando se vea en móvil, aparezca el menú al darle click a un botón. Le añadí la clase `navbar-light` y `bg-violet` para darle ciertos estilos a la barra de navegación.

Antes del botón, puse dos links (a) que tienen la clase `navbar-brand` y cada uno tienen una imagen en su interior. Quería variar la imagen dependiendo si estaba en móvil o no, por lo que use las clases `d-md-none` y `d-md-block`. Así, en móvil se ve la imagen más grande que en pantallas de un tamaño mayor o igual que una tablet.

El botón se hace poniendo un button con la clase `navbar-toggler`. Ese botón tiene un data-target llamado `"#navbarSupportedContentMD"`. Dentro del button, hay un span con la clase `navbar-toggler-icon` que hace que se vean tres líneas.

A parte del botón, hay un div (que será expandido al darle al botón si está en modo móvil).

Para ello tiene el id `"navbarSupportedContentMD"` y dos clases `collapse` y `navbar-collapse`.

En este div, creas la lista (ul) en la que yo le puse dos clases `navbar-nav` (esto lo requiere bootstrap) y `ml-lg-auto` para centrarlo en ordenadores.

Dentro de la lista, hay cuatro elementos (li) con las clases `nav-item` (requerida) y `px-lg-4` para poner un padding a los lados cuando esté en ordenador. En cada elemento hay un link (a) que tiene la clase `nav-link` (requerida por bootstrap) y un href que conecta los ids de cada sección (inicio, carta, quienes somos, contacto).

También tiene una barra de búsqueda. Se crea poniendo un form con la clase `form-inline` para que sea en línea y yo además le puse algunas clases (`my-2`, `my-lg-0`, `ml-md-auto` y `px-lg-4`) con las que controlo los paddings y los márgenes dependiendo del tamaño de la pantalla.

Dentro del form, he puesto un input que tiene las clases `form-control` y `search-bar` para que tenga estilos de barra de búsqueda. Además le puse las clases `col-10`, `col-md-8`, `col-lg-9`. Tiene un placeholder que pone "Buscar". También en el form puse un botón (button) para realizar la supuesta búsqueda con la clase `btn` para que tenga un estilo de botón.

Esta barra de navegación incluye un componente de bootstrap que se llama **scroll-spy**. Hace que a medida que vas bajando por la web, vayas cambiando de enlace en la barra y si haces click en un enlace de la barra, te lleva a dicha sección de la página. Es decir, al llegar a cada sección al hacer scroll, añade la clase `active` al link de la barra. Por lo que todo el contenido de la web está en una misma página. Esto se consigue añadiendo en el body un `data-spy="scroll"` para que tenga en cuenta el scroll que haces, un `data-target=".navbar"` para que afecte a la barra de navegación y un `data-offset="50"` que es el número de píxeles desde el que empieza al calcular la posición del scroll.

A parte de eso, hay que unir las secciones con los links del nav. Eso se consigue con ids. En las secciones pones `id="section1"` por ejemplo, y en el link `href="#section1"`.

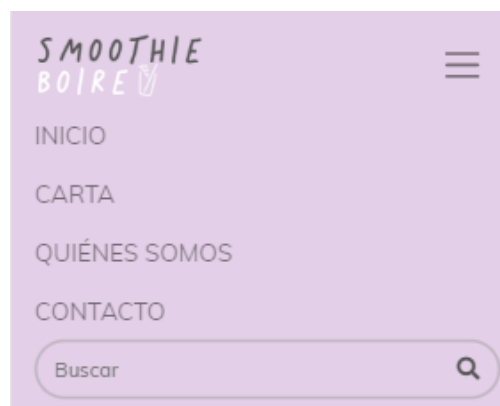
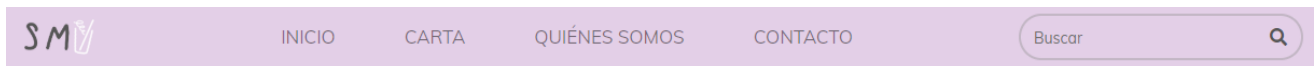
Además, teniendo en cuenta que mi navbar ocupa un espacio desde arriba, al darle click en la barra de navegación me llevaba a la sección correcta pero la barra me solapaba un poco de información de dicha sección. Buscando en internet, encontré como se solucionaba eso:

- En vez de unir cada enlace del navbar con la sección en sí, cree un div antes de cada sección y le asigne una clase que cree que se llama `scroll-nav`. A esos divs son a los que les asignaba el id correspondiente al enlace del navbar. Además, a los divs que contenían toda la sección tuve que asignarles la clase `position-relative` para que empiece desde un punto en concreto que yo quería.

- A la clase scroll-nav, le tuve que poner los siguientes estilos:

```
position: absolute;
background-color: transparent;
width: 1px;
height: 1px;
top: -90px;
```

De esta forma, conseguí que al darle click al enlace, llevara justo donde yo quería (cambiando el número del "top"). Los divs que contienen la clase scroll-nav no se ven ya que no tienen contenido, son transparentes.



Después de la barra de navegación, hay un gran div que tiene las clases `container-fluid` para que ocupe el 100% del ancho y `bg-gradient` que es una clase que creé yo.

## Inicio

La pagina de inicio tiene el id section1.

Está compuesta por tres grandes rows.

1. El primer div con la clase `row` tiene además otra clase `align-items-md-center`, que hace que usando las propiedades de flexbox, se alineen en el centro verticalmente a partir del tamaño mediano.  
Dentro del row, tiene tres divs:
  - o Un div que es una línea diagonal, para crearla hice una clase llamada `diagonal`, le puse la clase `bg-violet-secondary` para que tuviese el color de fondo que yo quería.
  - o Otro div que varía en tamaño de columnas dependiendo del tamaño de la pantalla (con la clase `col`). Además, use la clase `order` para que en ordenador saliese un div que va después antes que este (`order-md-2` y `order-1`).  
Dentro de este div, hice otra `row` y otro `col` porque así me permitía alinear la imagen como yo quisiese. Las `cols` y `offset` varían de un tamaño a otro.  
Dentro del col está el contenido, que es una imagen de un smoothie con la clase `img-fluid` para que siempre tenga el tamaño al 100%.

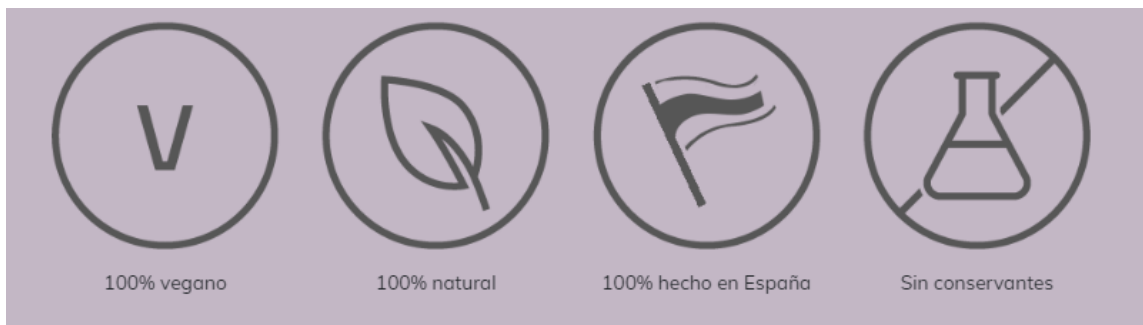
- Otro div que también varía dependiendo del tamaño de la pantalla (clase `col`). En este div también utilicé la ayuda de la clase order (`order-md-1` y `order-2`). Dentro de este div, hice otra `row` y otros dos `cols` porque así me permitía alinear los contenidos mejor. Yo quería que cuando estuviera en tamaño tablet y ordenadores, saliese la imagen del logo por lo que utilicé las clases `d-none` y `d-md-block` para lograrlo. Lo centré con `m-auto` y dentro metí la imagen con la clase `img-fluid`. En la otra columna escribí las clases `mx-auto` para centrarlo y `text-center`. Puse un margen arriba con la clase `mt-3`. Dentro del div, puse un texto con `h2`.



- El segundo div con la clase `row` tiene además un `margin-top` con la clase `mt-5`. Dentro del `row`, tiene dos divs:
  - Un div con las clases `col-md-8`, `col-xl-7` y `mx-auto` para centrarlo. Dentro de este div, está el carousel. Para hacer un carousel en bootstrap, se hace un div con el `id="carouselExampleControls"`, añades las clases `carousel` (crea el carousel) y `slide` que hace un efecto al cambiar las imágenes y el `data-ride="carousel"`. Para hacer este carousel, se hace un div con la clase `carousel-inner` en el que dentro van los divs con las clases `carousel-item`. Hay que poner la clase `active` al que quieras que salga primero. Dentro de ese div es cuando pones la imagen. A la imagen yo le puse las clases `d-block` y `w-100`. Para controlar el carousel, hay que poner dos links (a) con la clase `carousel-control-prev` o `carousel-control-next`, depende si es la de delante o la de atrás. Estos links, conectan con el div del carousel por el `id` que se mete en el `href`. Los links tienen otra propiedad llamada `role="button"` y `data-slide="{prev/next}"`. Dentro de los links, se mete un `span` con la clase correspondiente al icono que quieras darle, `carousel-control-{prev/next}-icon` y otro `span` con el texto, pero al llevar la clase `sr-only`, solo se verá si es un dispositivo de lectura.
  - Un div que cambia de tamaño dependiendo el tamaño de la pantalla (`col`) y que esta centrado en el centro con `mx-auto`. Dentro de él tiene un texto escrito con `h5` y centrado con la clase `text-center`.
- El último `row` del inicio lleva algunas características de los productos de la tienda. Dentro de este `row`, hay un div con las clases `col-12`, `bg-violet-secondary` (para que tenga un color específico de fondo), `mx-auto` y `padding (p-4)`. He puesto que en móvil tenga la clase `d-block` para que saliesen los elementos uno debajo del otro y la clase `d-md-flex` para que actúe como un `row` normal en tamaño mediano.



Está compuesta por 4 cols que contienen una imagen y un texto cada una. Cada col tiene distintas clases (col y offset) para que esté centrado en cada tamaño y que en ordenador se vean las 4 columnas en una fila, en tablet se vean en 2 columnas y 2 filas y en móvil se vea una debajo de la otra.



## Carta

La página de carta tiene el id section2.

Esta página está contenida en una fila que tiene 2 columnas.

1. Una columna es el título, que tiene la clase `col-12` para que ocupe el 100% del tamaño de la fila y dentro tiene un `h2`.
2. La otra col es el div con la carta en sí. La columna tiene las clases `col-11`, `col-md-8` para que varíe el tamaño en cada pantalla, `mx-auto`, `bg-light`, y le he quitado el padding de los lados con la clase `px-0`. En esta columna he creado una barra de navegación en **tabs**. Para ello, se crea una lista (ul) y le añades las clases `nav` y `nav-tabs`. Para que en móvil cada tab se ponga en vertical en vez de horizontal he usado las clases `flex-column` y `flex-lg-row`. Cada elemento (li) tiene que llevar la clase `nav-item` y dentro un link (a) que tenga la clase `nav-link`. Estas dos clases las requiere bootstrap para crear los navtabs. Si tiene la clase `active` es la que saldrá abierta primero. Hay que enlazar los hrefs de los links con los ids de los divs que quieres que aparezcan al darle click. Los links también llevan los atributos `data-toggle="tab"`, `role="tab"`, `aria-controls="(nombre que quieras darle)"` y `aria-selected="{true/false}"`. Estos atributos son usados por bootstrap con js para guardar estados de elementos de la página web. Yo en este caso he puesto tres links (Smoothie Frutas, Smoothie Verduras y Otros smoothies). Para tener un contenedor de los divs de las pestañas, hay que añadir una clase llamada `tab-content` y a los divs que contienen la información de cada pestaña, también hay que poner las clases `tab-pane`, `fade` (crea una animación), `show` (para que se vea la primera) y `active` (si quieres que empiece por ella). Además hay que enlazar los atributos con los links, por lo que se escribe el id que coincida con el href, `role="tabpanel"` y `aria-labelledby` con el nombre que le quieras dar.

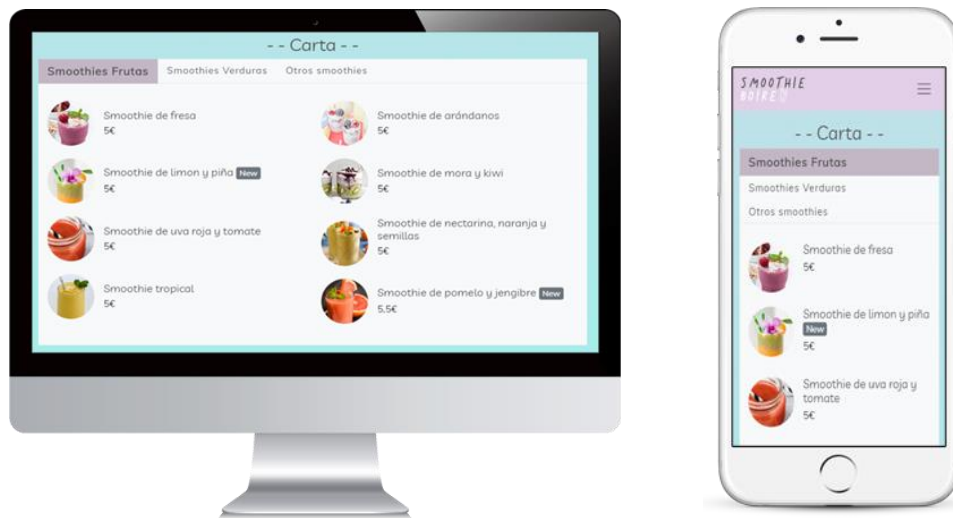
Cada div contiene una fila con dos columnas. En pantallas de ordenador, hace efecto la clase `col-lg-6`, en cambio en móvil utilizan `col-12`. Por lo que en ordenador hay dos columnas pero en móvil se ve una debajo de la otra.

Dentro de las columnas, he ido poniendo los artículos. Los artículos los he metido en una lista con la clase `list-unstyled` para que no tuviera ningún estilo.

Cada smoothie es un elemento (li) en el que he puesto el título, una foto y la descripción con un layout que tiene bootstrap que se llama **"Media object"**.



Para hacer un media object, hay que añadir al elemento la clase `media`. Dentro de cada elemento, hay una imagen y un div que tiene la clase `media-body`, en la que se pone el contenido del media object. En ese div metí un h5 con un enlace que lleva a la página del producto.

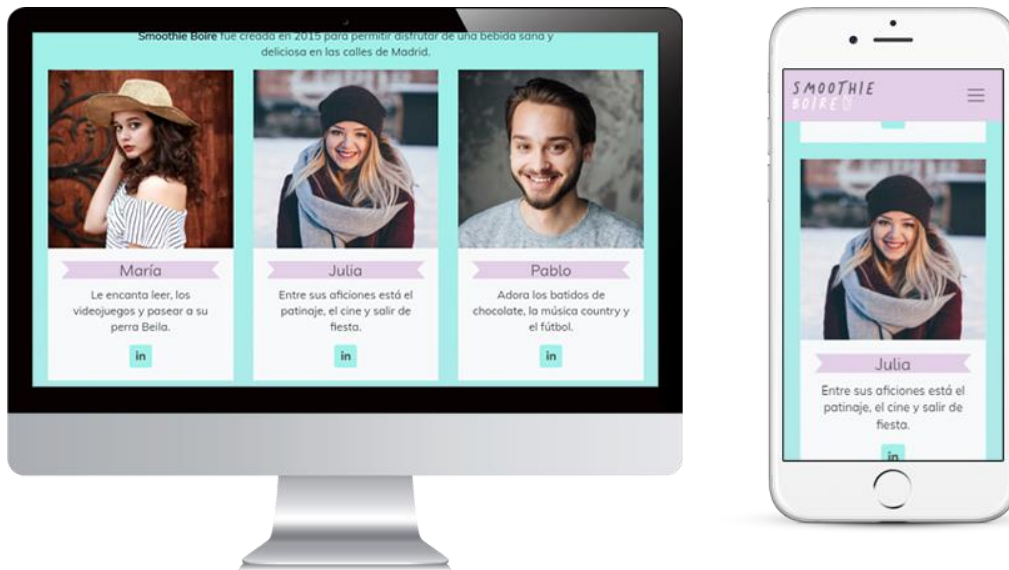


## Quiénes somos

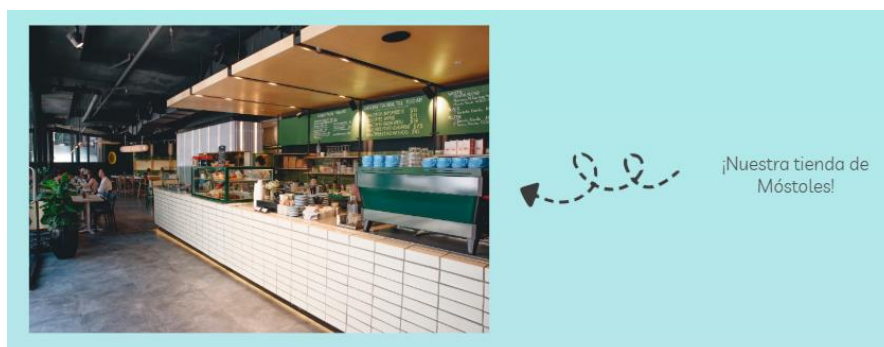
La página de Quiénes Somos tiene el id section3.

Esta página está contenida en una fila que tiene 4 columnas:

1. La primera columna es el título, que tiene la clase `col-12` para que ocupe el 100% del tamaño de la fila y dentro tiene un h2 que solo se ve a partir del tamaño de pantalla md (`d-none` y `d-md-block`) y un h3 que se ve solo en móvil (`d-block` y `d-md-none`). Así, el texto se veía mejor dependiendo del tamaño de cada pantalla.
2. La segunda columna contiene un pequeño párrafo con una descripción del equipo de la empresa. En móvil ocupa el máximo del tamaño y a partir de md se va centrando en la pantalla. (`col-12`, `col-md-10...`)
3. La tercera columna es la que contiene a los integrantes del equipo. Para eso, he creado un **card** para cada uno de ellos. Para crear un card, tienes que asignar la clase `card` a un div. Yo también le he puesto la clase `bg-light` y `text-center`. Dentro de ese div, he metido una imagen con la clase `img-fluid` y `card-img-top` para que la imagen estuviera arriba en el card y también metí un div que tiene la clase `card-body` en la que hay que meter el contenido del card. Dentro de card-body he puesto un h4 que tiene la clase `card-title` que lleva el nombre del miembro del equipo y un párrafo con la clase `card-text` en la que explico un poco cada uno. En card-text también metí un link que tiene la apariencia de un botón porque tienes las clases `btn` y `btn-primary`. Dentro del link he puesto el icono de linkedin de FontAwesome.



4. La última columna solo se ve en tamaños md, lg y xl por lo que tiene las clases `d-none` y `d-md-block`. Tiene una fila en la que luego se divide por tres columnas:
  - La primera columna (`col-5`) tiene dentro una imagen con la clase `img-fluid`.
  - La segunda columna (`col-2`) tiene otra imagen. He utilizado las clases `d-md-flex` y `align-items-center` para centrar la imagen verticalmente.
  - Y la tercera columna (`col-2`) tiene un texto que esta centrado también verticalmente igual que en la segunda columna.



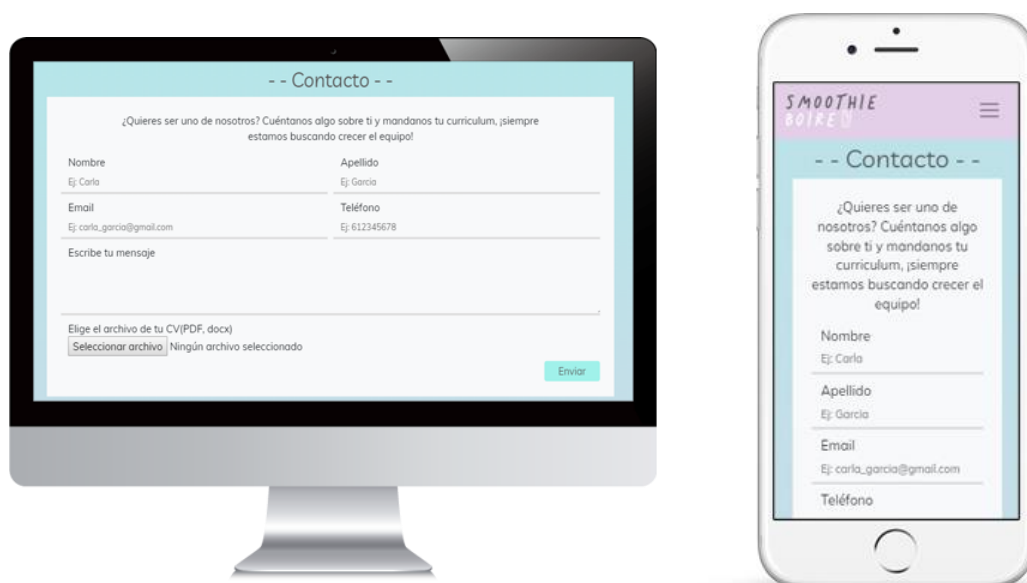
## Contacto

La página de Contacto tiene el id section4.

Esta página se distribuye en cuatro filas:

1. La primera fila es el título y tiene la clase `col-12` para que ocupe el 100% del tamaño de la fila y dentro tiene un `h2`.
2. La segunda fila tiene una columna que está centrada, tiene padding y `bg-light`. En ella he metido dos elementos.
  - a. Un párrafo de introducción al formulario. Está centrado (`mx-auto`) y tiene la clase `p-3`.
  - b. Un formulario de contacto. Para hacer el formulario, después de la etiqueta `form`, he hecho un `div` que tiene la clase `form-row` para que se comporte como la fila de un grid. He ido metiendo los campos del formulario englobados cada uno con la clase `form-group`.

La clase `form-group` es la forma más fácil de agregar algo de estructura a los formularios. Proporciona una clase flexible que fomenta la agrupación adecuada de etiquetas, controles, texto de ayuda opcional y mensajes de validación de formularios. Por defecto solo aplica margen inferior. Dentro de cada `form-group` hay un `label` y un `input`. Todos los `inputs` llevan la clase `form-control` excepto el de elegir un archivo que tiene la clase `form-control-file`. `Form-control` incluye estilos para apariencia general, estado de focus, tamaño y más. Al final del formulario, tengo un botón “Enviar” que hace la función de submit con las clases `btn`, `btn-primary` y `float-right` para que salga a la derecha del formulario.

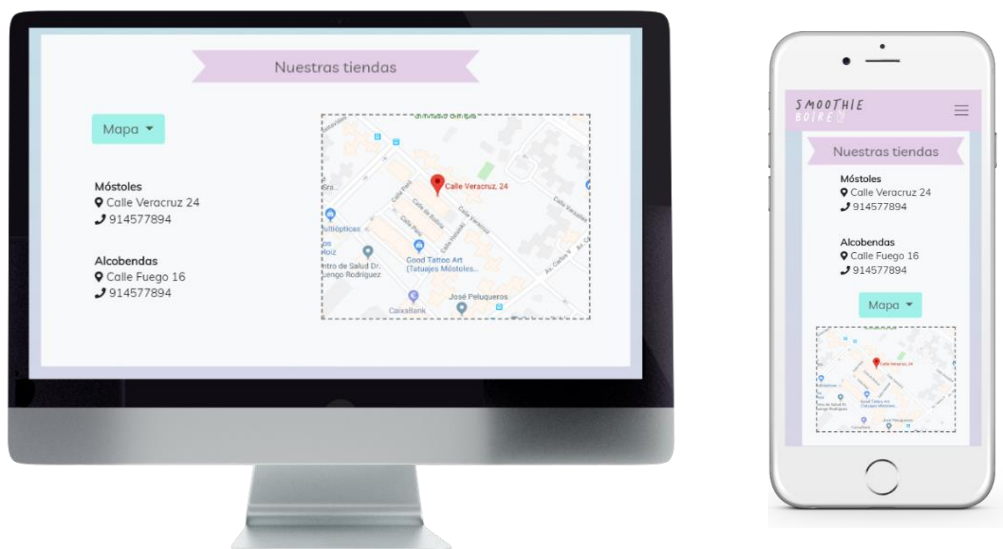


3. La tercera fila de contacto es una sección en la que enseño las tiendas que existen. En ella, he metido un título que lo he centrado con las clases `d-flex`, `align-items-center` y `justify-content-center`. El título también tiene la clase `subtitulos` que explicaré más adelante. En otra fila que tiene las clases `justify-content-around` y `my-lg-5`, he metido dos columnas más. Una columna tiene la clase `col-lg-4` y la otra `col-lg-6`.
  - La primera, quería que tuviera dos elementos: un dropdown y una lista con los detalles de cada tienda, y para posicionarlos como yo quería utilicé la clase `d-flex`. Añadiendo la clase `flex-column` he conseguido que cada elemento se pusiera uno debajo del otro, porque si no, se me ponían al lado. Y como cambia el sentido del eje X y el Y, ahora `justify-content-between` me lo alineaba en respecto al eje Y, `align-items-center` en móvil y `align-items-md-start` para que se viera a partir del tamaño de tablet de esa forma centrado o a la izquierda.
    - Para hacer el dropdown, le he asignado a un `div` la clase `dropdown`. Dentro de ese `div` hay un button que tiene las clases `dropdown-toggle` para que haga la función de toggle y `btn-lg` para que sea un botón más grande. Además de esas clases, el button lleva los atributos `type="button"`, un `id="dropdownMenuButton"`, `data-toggle="dropdown"`, `aria-haspopup="true"` y `aria-expanded="false"`. Los booleans los cambia jquery al darle click al dropdown. Tras el button, hay que crear el menú que sale al activar el dropdown, por lo que se hace un `div` con la clase `dropdown-menu` y el atributo `aria-labelledby="dropdownMenuButton"` (ese tiene que coincidir con el id del button para que funcione). Dentro de ese `div` hay dos links (a) con la clase `dropdown-item` para que salgan en el dropdown como opciones.

- Para la lista, he asignado la clase `list-group`. Esta clase es un componente muy útil y flexible para mostrar listas de elementos de una manera bonita. Dentro tiene dos elementos (li) que tienen la clase `list-group-item`. En cada elemento he escrito el nombre de la ciudad en la que se encuentra la tienda, un icono de teléfono de FontAwesome para poner el teléfono al lado y otro icono de localización para poner la dirección de la tienda.
- La segunda columna simplemente tiene una imagen con la clase `img-fluid` y la clase `img-mapa` que creé para darle un borde específico.

En esta sección, los divs tienen la clase de `col-12` cuando la pantalla es tamaño móvil y así, se disponen uno debajo del otro. Además, el dropdown y la lista invierten el orden. El dropdown tiene las clases `order-2` y `order-lg-1` y la lista tiene las clases `order-1` y `order-lg-2`.

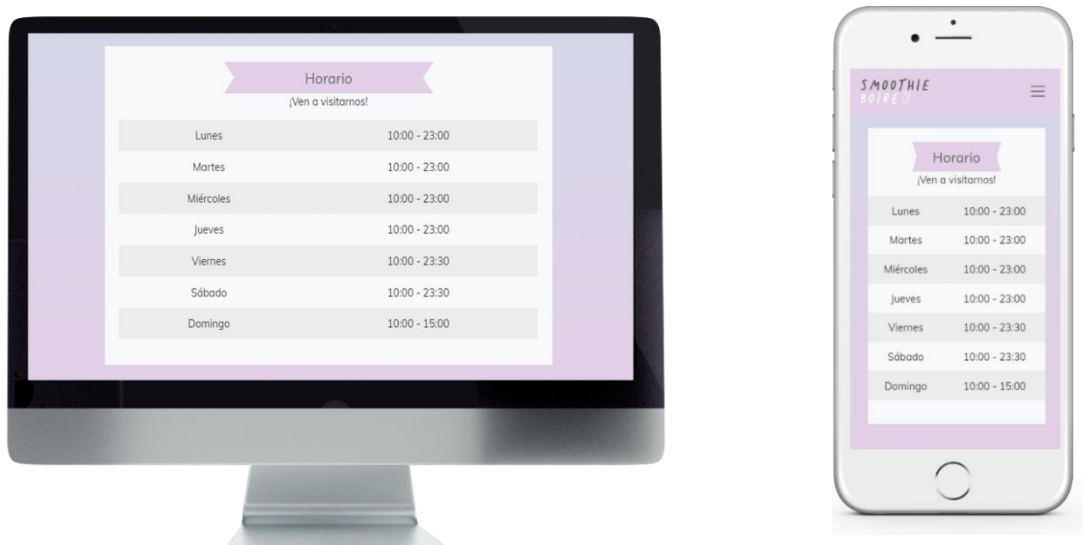
El dropdown tiene algo de JavaScript para que cambie la imagen dependiendo de la elección. Ese efecto lo conseguí añadiendo ids distintos a cada dropdown-item y otro id a la imagen (id=imagenMapa). Lo que hice de script fue añadir un eventListener de click a cada id de los dropdown-items. Si pulsaba la de Móstoles, se llamaba a la función para cambiar el src de imagenMapa a la imagen del mapa de Móstoles, y lo mismo si pulsaba la opción de Alcobendas.



4. La cuarta fila de contacto es la sección del horario que tienen las tiendas físicas. Dentro de esa row tengo una columna que está centrada y que varía su tamaño según el tamaño de la pantalla (`col-10`, `col-md-7` y `col-lg-6`).

En ella he puesto un div para escribir el título que tiene la clase `subtitulos` y está centrado vertical y horizontalmente (`align-items-center` y `justify-content-center`), la clase `w-50` para que ocupara un 50% del ancho total del div contenedor y `mx-auto` para centrarlo.

En esta sección se encuentra la tabla. Para crearla he puesto a la etiqueta `table` las clases `table`, `table-striped` (para que tenga estilo rayado) y `table-borderless` (para que no tenga los bordes que asigna bootstrap por defecto). La tabla es responsive por lo que se adapta automáticamente a la pantalla.



## Footer

Aparece en todas las páginas abajo del todo.

El footer está contenido en un `container-fluid`.

La etiqueta footer hace de row. Para darle el fondo oscuro he añadido la clase `bg-secondary` además de ponerle paddings (`px-3` y `py-5`). También tiene la etiqueta `justify-content-around` para que los elementos que contiene tengan espacio antes, entre medio y después.

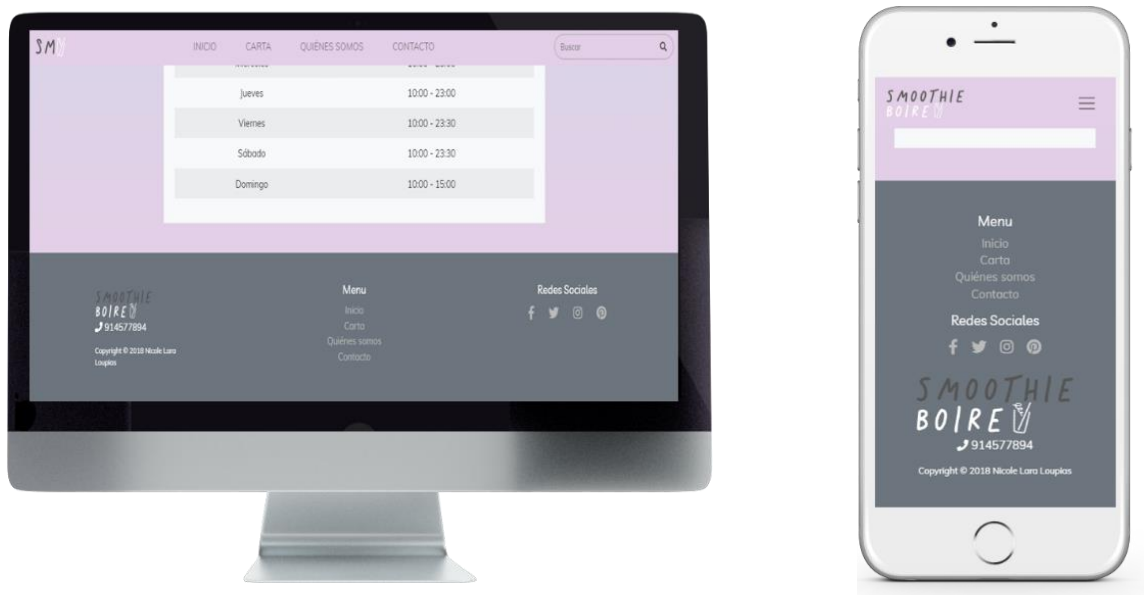
Esta fila tiene tres columnas:

1. La primera tiene las clases `col-md-4`, `col-xl-2` y para que tenga un display flex he puesto `d-flex` y `align-items-center` para centrar los elementos que contiene. Dentro de esta columna, he hecho div que actúa como fila (`row`) para dividirlo en dos columnas más. La primera lleva el logo de la empresa, y en cada tamaño de pantalla va variando el tamaño de la columna para que la imagen no se vea muy grande en ninguno de los casos (`col-10`, `col-md-9`, `col-lg-8`).

2. La segunda columna tiene las clases `col-md-2`. Dentro de ella, he puesto un `h5` que tiene las clases `text-white` y `text-center` para poner "Menu". También hay una lista con las clases `list-unstyled` (para que bootstrap le quite todo su estilo por defecto) y `text-center`. En esta lista he metido los mismos links que hay en el navbar.
3. A la tercera columna le he asignado las clases `col-md-3` y `col-xl-2` para variar el tamaño según la pantalla y una clase que yo he creado que se llama `links-footer` para ponerle un color exacto a estos links ya que si no, no se veían por el fondo oscuro que tiene el footer. Dentro de esta columna, he puesto también un título con la etiqueta `h5` y las clases `text-white` y `text-center`. También he puesto un `div` que engloba los links de las redes sociales. Este `div` lleva las clases `d-flex`, `justify-content-md-around` y `justify-content-center`. Esto lo he hecho porque quería que a partir del tamaño `md`, se vieran con un poco más de espacio entre una y otra, en cambio en móvil quedaba mejor todo centrado.

Los enlaces que contiene este `div` llevan cada uno un ícono de FontAwesome.

Estas tres columnas cambian de orden en móvil, gracias a que la primera columna tiene las clases `order-3` y `order-md-1` (sale la última en móvil), la segunda tiene `order-1` y `order-md-2` (sale la primera en móvil) y la tercera tiene las clases `order-2` y `order-md-3` (sale la segunda en móvil).



## Producto

En la página de producto, está el mismo nav y footer que en la página principal.

Todo lo demás está contenido en un `div` con las clases `container-fluid` y `bg-gradient`.

Dentro de ese `div`, tengo una `row` con una columna dentro con las clases `col-10`, `bg-light`, `mx-auto` y `my-5`. En esa columna, hay una fila que está dividida por tres columnas importantes:

1. La primera, tiene la clase `col-md-5`. Esta compuesta por dos `divs`. El primero, tiene la clase `d-flex` y dentro tiene un `h2` con una clase que yo creé llamada `titulo-producto` para darle un estilo en concreto y un `button` con la clase `btn-primary` que lleva el precio del producto. El segundo `div` contiene dos párrafos y una lista (`ul`) con varios elementos. Nada de esto último tiene asignada ninguna clase en particular de bootstrap.

2. La segunda columna, tiene la clase `col-md-2` y solo se ve en pantallas lg (tiene las clases `d-none` y `d-lg-block`). Dentro de ella, solamente lleva una imagen con la clase `img-fluid`.
3. La tercera columna, tiene las clases `col-md-7`, `col-lg-5` para que dependiendo del tamaño de la pantalla sea de un u otro tamaño de ancho y también tiene la clase `d-flex` y `align-items-center` para centrarla verticalmente. Dentro de ella, hay una imagen con las clases `img-fluid` y `img-producto` (que explicaré más adelante).



#### 4. Custom.css

La página había que hacerla con bootstrap, por lo que tampoco he utilizado mucho css propio.

La mayoría de lo que hay en custom.css son clases de bootstrap en las que he cambiado sobretodo el color del background, el color de la letra, y quitado los bordes por defecto que asigna bootstrap a esas clases. También, a algunas clases les he quitado el border-radius o el box-shadow que ya venía.

Algunos selectores que tengo que destacar y comentar son:

**.img-mapa** -> A esta clase le he puesto `border: 2px dashed #707070;`

**.img-producto** -> A esta clase le he puesto:

- `border-radius: 50%; transition: all .7s ease-in-out;`
- `border: 2px dashed #707070; padding: 7px;`

Para que la imagen fuese un círculo, tuviese el borde característico de la web y la transición es para que se haga más suave el cambio de redondo a cuadrado.

**.img-producto:hover** -> A esta clase le he puesto `border-radius: 0%;` para que se hiciera la transición de imagen redonda a cuadrado.

**.li.media img** -> A esta clase le he puesto algo parecido,

`border-radius: 50%; border: 2px dashed #707070; padding: 4px;`

**.li.media img:hover** -> Al hacer hover se pone `opacity: 0.6;`



**.carousel-control-prev-icon y .carousel-control-next-icon** -> A estas dos clases, le cambie una parte de la url del background-image, donde ponía `fill='%23707070'`, puse el código hexadecimal que yo quise.

**.subtitulos** -> A esta clase le he puesto:

- `background-color: #e3cfe7;`
- `clip-path: polygon(100% 0%, 95% 50%, 100% 100%, 0 100%, 5% 50%, 0 0);`
- `height: 40px;`

Para que tenga un color violeta clarito y una forma como de un lazo (clip-path). Height para que tuviera el alto que yo quería.

**.diagonal** -> Esta clase es la que lleva el div del inicio que hace una forma diagonal. Para conseguirlo, le asigne esto:

- `width: 100%;`
- `height: 500px;`
- `position: absolute;`
- `clip-path: polygon(0 30%, 100% 0%, 100% 73%, 0% 100%);`

**.bg-violet** -> Esta clase simplemente asigna un background-color determinado que yo quería.

## 5. Listado de componentes de investigación

Los componentes que he investigado han sido:

- Scroll-spy
- Dropdown
- Media object
- Carousel
- Nav tabs
- Cards