

Aplicaciones Distribuídas en Internet

Herramientas de personalización de Front-End

Luis Miguel Panadero Cerdá
Sergio Ruiz Francés
Joaquín Cerdán López

Índice

Índice	2
Introducción a las herramientas de personalización de Front-End	3
Bootstrap	4
Introducción	4
Instalación	4
Conceptos básicos	4
Sass	5
Colores	5
Breakpoints	5
Contenedores	6
Grid	6
Filas	6
Columnas	6
Tipografía	7
Headings	7
Display	7
Lead	8
Inline text elements	8
Images	8
Tables	8
Componentes	8
Navbar	8
Card	9
Button	9
Pagination	9
Alerts	10
Accordion	10
Otras herramientas	12
Tailwind	12
Material(ize)/(UI)	12
Bulma	12
Pros y contras del uso de las herramientas	14
Pros	14
Contras	14
Conclusión	15
Bibliografía	16

Introducción a las herramientas de personalización de Front-End

Desde los inicios de las webs accesibles a todo el mundo, en el auge de Internet, se ha ido llegando a la idea de que la interfaz de usuario es un elemento clave en que nuestras páginas web sean accesibles y fáciles de usar por quienes las visitan. Para ello el lenguaje HTML se apoyó en el lenguaje CSS para poder editar a su gusto todos los elementos que pudieran componer sus páginas.

A medida que el desarrollo de páginas web se fue extendiendo y popularizando, los equipos de desarrollo quedaban a la orden del día y la división del trabajo hacía muy difícil la consistencia visual de los elementos en una misma página o dominio web. Un equipo podría estar compuesto por personas de varios lugares del mundo haciendo una parte de un proyecto común, y como la edición y personalización de elementos en HTML es muy variada gracias a CSS era inviable pensar que el trabajo de uno fuera a coincidir visualmente con el trabajo de sus compañeros.

Los frameworks de personalización de Front-End se encargan de solucionar estos problemas de consistencia gracias a la creación de clases predeterminadas que modifican de una manera los elementos de las páginas web que conocemos, además de incluir fuentes comunes e incluso iconos.

Utilizando estas herramientas los desarrolladores solo tenían que ponerse de acuerdo en que clases querían utilizar en su proyecto y así todo quedaría consistente.

Bootstrap

Introducción

Bootstrap es un framework de front-end gratuito que facilita el desarrollo web, gracias a sus plantillas creadas con HTML, CSS y JavaScript los desarrolladores de software podemos dar estilo a nuestros proyectos de manera rápida y sencilla. En esta documentación nos centraremos en Bootstrap v5.1.

Instalación

Al igual que todo lo que vamos a explicar la instalación de Bootstrap es muy simple, no es necesario descargar nada ni instalar ningún software. Implementando los CDN (Content Delivery Network) con jsDelivr en nuestra cabecera <head>, tendremos integrado bootstrap.

```
<!-- Bootstrap CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
      rel="stylesheet"
      integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3"
      crossorigin="anonymous">
<!-- JavaScript: Bootstrap Bundle with Popper -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
        integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
        crossorigin="anonymous">
</script>
```

Debemos distinguir entre el primer CDN, el link, el cual implementa todos los elementos CSS del framework y el segundo CDN, el script, que implementa el JavaScript necesario para poder interaccionar con la página web.

Conceptos básicos

Para el correcto aprendizaje del framework es necesario explicar diferentes conceptos comunes para todos los objetos y documentos.

Al empezar un documento, una buena práctica sería realizar algo similar al ejemplo. Encontramos la etiqueta de HTML5 doctype, con el idioma especificado y dentro de la cabecera además de los CDN ya comentados debemos crear un título, indicar la codificación y debido a que bootstrap está desarrollado centrándose en el diseño de un móvil primero para luego escalar al tamaño necesario, añadimos la etiqueta responsive meta.

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Bootstrap Tutorial</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
  </head>
  <body> ...
</body>
</html>

```

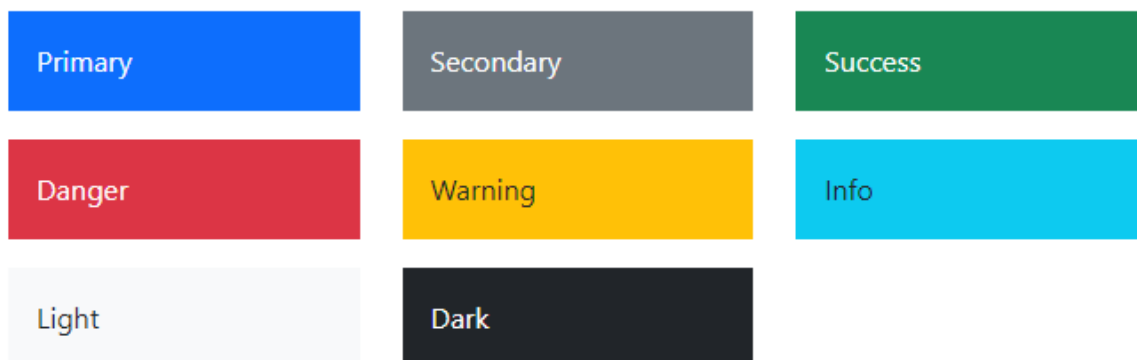
Sass

Todos los estilos, componentes, elementos se pueden personalizar mediante Sass. Dentro de nuestro directorio, encontraremos el directorio scss donde se encuentra el archivo custom.scss. Dentro de él encontramos todas las variables predefinidas, podremos personalizar cualquier variable o crear nuestros propios componentes.

Para más información sobre [Sass](#).

Colores

A un gran porcentaje de los objetos podemos añadirles un tema para personalizar nuestra web, para ellos bootstrap predefine unos estilos:



Breakpoints

Los breakpoints son los bloques de construcción del “responsive design”, bootstrap incluye seis breakpoints predefinidos:

Breakpoint	Clase	Dimensiones
X-Small	-	<576px
Small	sm	≥576px
Medium	md	≥768px
Large	lg	≥992px
Extra large	xl	≥1200px
Extra extra large	xxl	≥1400px

Contenedores

Los contenedores son el objeto de diseño más básico, como su nombre indica sirven contener/delimitar el contenido de nuestra página, encontramos 3 diferentes contenedores:

- `.container`
- `.container-fluid`
- `.container-{breakpoint}`, por ejemplo `.container-sm`

```
<div class="container-sm">
  <h1>Bootstrap tutorial small</h1>
</div>
<div class="container-fluid">
  <h1>Bootstrap tutorial fluid</h1>
</div>
```

Grid

El sistema de malla de bootstrap utiliza una serie de elementos tales como contenedores, filas y columnas para alinear el contenido, todo esto es construido de manera “responsive” mediante flexbox, para aprender flexbox podemos utilizar el juego [Flexbox Froggy](#).

Filas

Las filas son envoltorios para las columnas, es decir, sirven para agrupar un conjunto de columnas con la misma personalización.

Columnas

Las columnas se introducen dentro de una fila, dentro de cada fila caben hasta 12 columnas.

```
<div class="col">1 of 12</div>
<div class="col">2 of 12</div>
<div class="col">3 of 12</div>
<div class="col">4 of 12</div>
<div class="col">5 of 12</div>
<div class="col">6 of 12</div>
<div class="col">7 of 12</div>
<div class="col">8 of 12</div>
<div class="col">9 of 12</div>
<div class="col">10 of 12</div>
<div class="col">11 of 12</div>
<div class="col">12 of 12</div>
```

[illegible]

Tipografía

Bootstrap utiliza la tipografía por defecto del navegador con tamaño 1rem (16px) y con separación entre líneas de 1.5. Podemos cambiar diferentes aspectos de la tipografía con los siguientes atributos:

font-family	Cambiar la fuente
font-size	Cambiar el tamaño
line-height	Cambiar la separación entre líneas
link-color	Cambiar el color de los links
body-bg	Cambiar el color del fondo
text-{themeColor} (text-primary)	Cambiar el color

Headings

```
<h1>Heading 1</h1>  
<h2>Heading 2</h2>  
<h3>Heading 3</h3>  
<h4>Heading 4</h4>  
<h5>Heading 5</h5>  
<h6>Heading 6</h6>
```

Heading 1

Heading 2

Heading 3

Heading 4

Heading 5

Heading 6

Display

```
<p class="display-1">Display 1</p>  
<p class="display-2">Display 2</p>  
<p class="display-3">Display 3</p>  
<p class="display-4">Display 4</p>  
<p class="display-5">Display 5</p>  
<p class="display-6">Display 6</p>
```

Display 1

Display 2

Display 3

Display 4

Display 5

Display 6

Lead

```
<p class="lead">
  Esto es un párrafo lead.
</p>
<p>
  Esto es un párrafo normal.
</p>
```

Esto es un párrafo lead.

Esto es un párrafo normal.

Inline text elements

```
<p><mark>Podemos seleccionar un párrafo</mark></p>
<p><del>Podemos tachar un párrafo</del></p>
<p><u>Podemos subrayar un párrafo</u></p>
<p><small>Podemos hacer más pequeño un párrafo</small></p>
<p><strong>Podemos ponerlo en negrita</strong></p>
<p><em>Podemos ponerlo en cursiva</em></p>
```

Podemos seleccionar un párrafo

~~Podemos tachar un párrafo~~

Podemos subrayar un párrafo

Podemos hacer más pequeño un párrafo

Podemos ponerlo en negrita

Podemos ponerlo en cursiva

Imágenes

Las imágenes en Bootstrap son "responsive" usando `.img-fluid`, podemos hacer miniaturas con `-img-thumbnail`, alinear imágenes con `float-start`, `float-end`,...

```
<div class="text-center">
  
</div>
```

200x200

Tablas

Las tablas se construyen mediante filas y columnas. La etiqueta `table` marca el contenido de la tabla, la etiqueta `'thead'` marca los títulos de las columnas, `'tbody'` marca el cuerpo de la tabla, `'tr'` marca las filas y `'th'` las columnas.

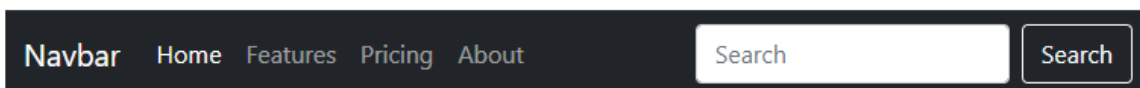
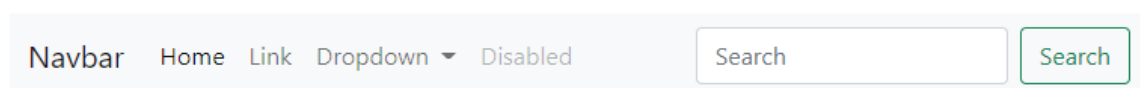
```
<table class="table">
  <thead>
    <tr>
      <th scope="col">#</th>
      <th scope="col">First</th>
      <th scope="col">Last</th>
      <th scope="col">Handle</th>
    </tr>
  </thead>
```

#	First	Last	Handle
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry the Bird		@twitter

Componentes

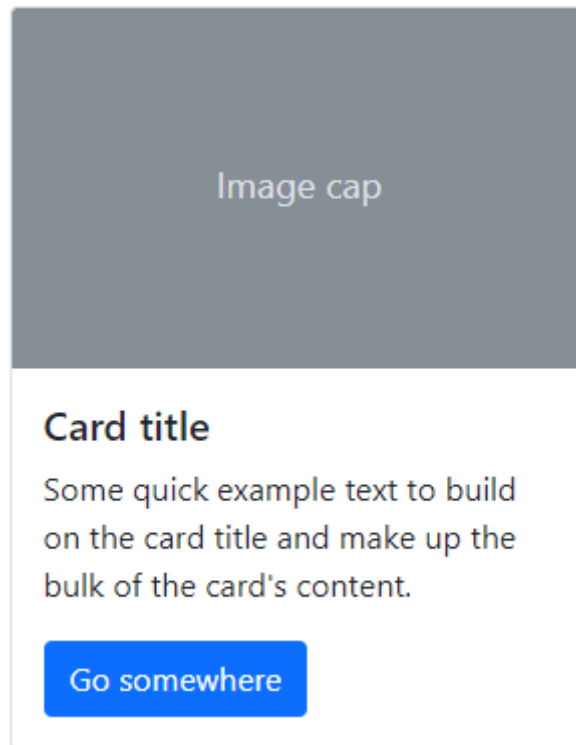
Navbar

Barra de navegación con las principales funciones de las páginas, barras de búsqueda y otras funciones como login, perfil...



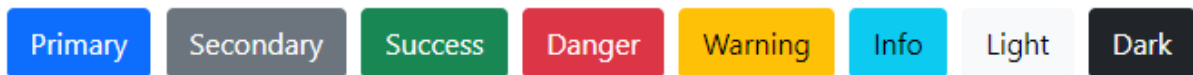
Card

Es una tarjeta que muestra información compacta y contenida junto a un título, imagen y enlace, todos ellos opcionales.



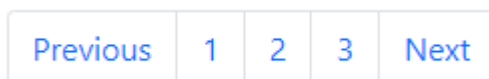
Button

Es un botón que puede accionar diferentes funciones.



Pagination

Es una numeración que ofrece navegar entre las diferentes páginas que muestran un contenido dividido entre estas.



Alerts

Son alertas que se muestran por encima de la interfaz de usuario para asegurar la lectura de estas.

A simple primary alert—check it out!

A simple secondary alert—check it out!

A simple success alert—check it out!

Accordion

Este componente ofrece mucha información de manera compacta dividida en subgrupos con un título el cual al pulsarlo te permite ver un párrafo de texto asociado.

Accordion Item #1



This is the first item's accordion body. It is shown by default, until the collapse plugin adds the appropriate classes that we use to style each element. These classes control the overall appearance, as well as the showing and hiding via CSS transitions. You can modify any of this with custom CSS or overriding our default variables. It's also worth noting that just about any HTML can go within the `.accordion-body`, though the transition does limit overflow.

Accordion Item #2



Accordion Item #3



Formularios

Los formularios son otra de las funcionalidades más populares y permiten recoger datos de todo tipo desde cajas de texto, checkbox, desplegables entre otros.

Email address

We'll never share your email with anyone else.

Password

☐ Check me out

Submit

Otras herramientas

Como ya hemos podido comprobar, bootstrap cuenta con un gran número de características que lo convierten en el framework de front-end más popular actualmente. No obstante, existen muchas otras alternativas que pueden ser igualmente útiles o incluso mejores a la hora de adecuarse a las necesidades de los programadores en ese momento. Por ello, a continuación revisaremos tres de las alternativas más populares que existen actualmente.

Tailwind

Tailwind es un framework de front-end desarrollado por Adam Wathan en noviembre de 2017.

Una de las características que lo diferencia de bootstrap es la importancia que le da a la utilidad y a la customización de elementos HTML. Al contrario que en bootstrap donde el framework nos ofrece una serie de componentes predefinidos con los que podemos trabajar y modificar solo parcialmente, tailwind nos permite crear diseños customizables sin demasiado esfuerzo por parte de los programadores utilizando una serie de clases proporcionadas por el framework. Además, tailwind sigue ofreciendo una serie de widgets preestablecidos además de los diseñados por el usuario.

A continuación podemos ver el ejemplo de una barra de navegación diseñada con Tailwind. Este es uno de los muchos componentes que ofrece la página web del framework.

Material(ize)/(UI)

Materialize es un framework de front-end basado en Material Design de Google. Material Design es un sistema de diseño creado por Google en 2014 para la confección de aplicaciones y páginas web en Android por lo que es en esta plataforma donde más podemos apreciar su potencial. Actualmente también tiene compatibilidad con iOS, Flutter y web.

Con lo que respecta a las características principales de Materialize es muy similar al sistema de componentes de bootstrap. Todos los componentes tienen un diseño más ligero y pueden llegar a ser un poco más rígidos que los de sus competidores.

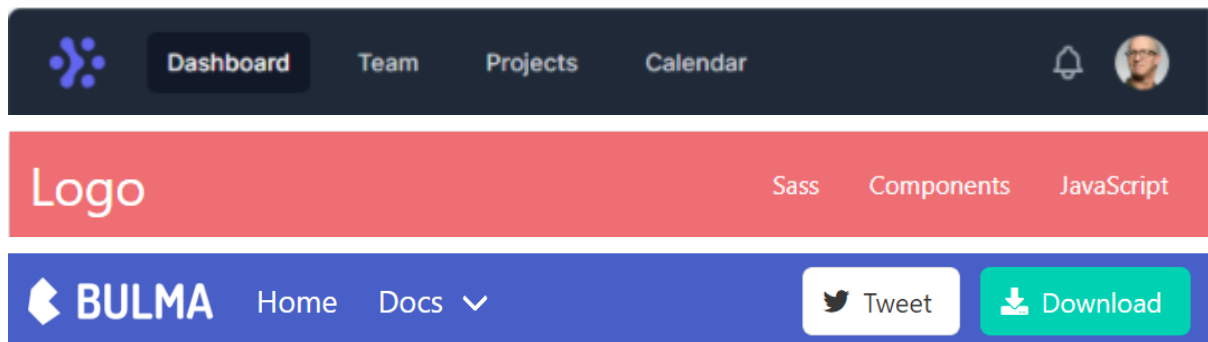
Bulma

Bulma es un framework de front-end desarrollado por Jeremy Thomas con un gran diseño y una sintaxis fácil de aprender. Cuenta con la mayoría de los componentes más importantes de bootstrap y comparte muchas similitudes con este. Sin embargo, Bulma es un framework de CSS puro y por lo tanto no cuenta con ninguna de las funcionalidades de los plugins de javascript de los que hace uso varios de los componentes dinámicos de bootstrap.

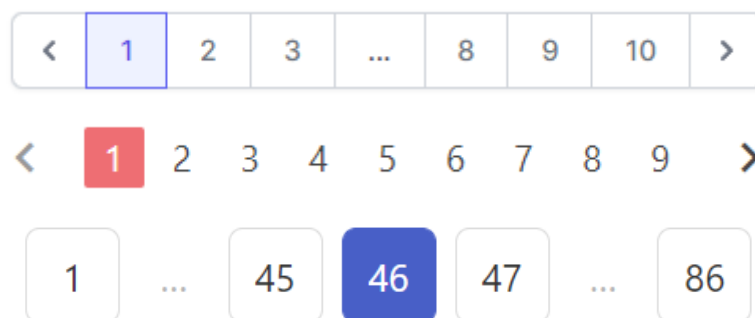
Componentes

Ahora que ya sabemos cuales son las principales características de cada uno de estos tres frameworks vamos a comparar algunos de sus componentes más utilizados para poder apreciar las diferencias a la hora del diseño.

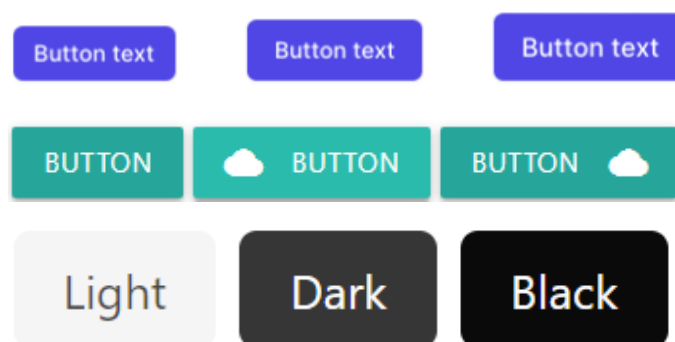
Estas son las diferentes **barras de navegación** para Tailwind, Materialize y Bulma respectivamente.



Estas son las diferentes **paginaciones** para Tailwind, Materialize y Bulma respectivamente.



Estas son las diferentes **botones** para Tailwind, Materialize y Bulma respectivamente.



Como podemos ver, los tres tienen unos diseños muy diferenciados que pueden ser modificados más o menos según el proyecto escogido, a partir de ahí queda a nuestra elección cuál elegir teniendo en cuenta también la existencia de su competidor bootstrap.

Pros y contras del uso de las herramientas

Pros

- **Dan cohesión:** Estas herramientas permiten a los desarrolladores mantener el mismo estilo a lo largo de su aplicación y no romper la armonía visual que nos ofrecen las clases predefinidas de los frameworks.
- **Dan calidad:** Su diseño especializado y simple garantiza que un acabado que sobrepasa con creces los estándares calidad.
- **Ayudan a los novatos:** La mayoría de las herramientas de diseño cuentan con una clara documentación que explica y ejemplifica todos los usos posibles y opciones de sus componentes.
- **Gran customización para expertos:** Aunque estas herramientas sean accesibles para cualquiera que trabaje en el ámbito de las habilidades de diseño, los frameworks front-end ofrecen muchas utilidades para todos aquellos que cuentan con un dominio superior de estas herramientas, entre ellos la customización y alta flexibilidad de sus componentes.

Contras2

- **No aprendes CSS:** El uso de frameworks de diseño requiere un mínimo de conocimientos sobre CSS y HTML. Sin embargo, los programadores que comienzan su aprendizaje sobre el desarrollo de webs con estas herramientas pueden llegar a acomodarse y no llegar nunca a comprender las bases de lo que pensamos que es algo esencial en el conocimiento de un desarrollador web.
- **Se pierde originalidad:** El uso masificado de estas herramientas nos lleva a ver siempre los mismos componentes HTML en distintas páginas ya que no siempre se van a personalizar las clases que ofrecen y se usarán los esquemas por defecto.

Conclusión

Los frameworks de front-end son a día de hoy herramientas clave en el desarrollo de todo tipo de proyectos y están presentes en el aprendizaje de nuevos programadores que se inicien en el desarrollo de aplicaciones web.

Las herramientas de este tipo son utilizadas como base de diseño en muchos equipos de desarrollo, ya que son tan populares, y gracias a ellas estos equipos mantienen un diseño de interfaz compacto que aporta al usuario coherencia visual.

Como también hemos hablado anteriormente, creemos que es esencial en el desarrollo del aprendizaje de un programador descubrir los fundamentos de HTML y CSS antes de comenzar a utilizar estas herramientas.

Para concluir, creemos que las herramientas de personalización de front-end aportan una consistencia y una base sólida en la creación de páginas y proyectos web con una curva de aprendizaje asequible para cualquier desarrollador ya sea junior o más experimentado. Ofreciendo a estos últimos una gran flexibilidad a la hora de customizar el aspecto de los elementos de HTML con un acabado más profesional.

Bibliografía

<https://classpert.com/es/blog/top-bootstrap-alternatives?&isredir>

<https://getbootstrap.com>

<https://www.w3schools.com/bootstrap5/index.php>

<https://getbootstrap.com/docs/5.1/customize/sass/>

<https://flexboxfroggy.com/#es>

<https://blog.coursesity.com/top-bootstrap-alternatives/>

[https://es.wikipedia.org/wiki/Bootstrap_\(framework\)](https://es.wikipedia.org/wiki/Bootstrap_(framework))