



Tutorial de Bootstrap

Guía completa paso a paso

Aprender a usar Bootstrap en tus proyectos es una de las mejores ideas que puedes tener, y aquí te voy a explicar no solo los pasos que tienes que seguir para hacerlo, sino cómo funciona y cuáles son los elementos más usados.

Índice de contenidos

1. Qué es Bootstrap
2. Cómo lo agrego a mi proyecto
3. Cómo crear estructuras
 1. Contenedor
 2. Fila
 3. Columna
4. Elementos
 1. Navegación
 2. Botones
 3. Alertas
 4. Etiquetas
 5. Barras de progreso
5. Resumen
6. Conclusión

1. ¿Qué es Bootstrap?

Un marco de trabajo (*framework*) para crear diseños *responsive* de calidad. Como el concepto de marco de trabajo puede resultar difícil de comprender, te lo voy a explicar con otras palabras:

Bootstrap es un conjunto de clases CSS y funcionalidades Javascript que ya están hechas. Y lo único que tienes que hacer es ordenarlas como más te guste para crear algo propio usando los elementos que te ofrece.

Entonces, en lugar de tener que reinventar la rueda cada vez que quieres maquetar algo, pues simplemente echas mano de este maravilloso *framework*.

Así de simple.

Es como tener bloques prefabricados y unirlos a tu gusto para crear algo, simplemente.

La mejor manera de aprender es practicando, pero para ello lo primero que debes hacer es agregar Bootstrap.

2. ¿Cómo lo agrego a mi proyecto?

La respuesta es tan sencilla que va a parecer que estoy de broma:

Para añadirlo tienes que hacer exactamente igual que harías con cualquier otro archivo CSS o Javascript, incorporarlo en tu maqueta usando *link* y *script*.

Las dos formas principales de hacerlo son, o bien descargando los archivos o bien añadirlos directamente desde el CDN oficial. Un CDN (*Content Delivery Network*) es básicamente una red de servidores que se encargan de ofrecer contenido. Suelen usarse para hospedar en ellos imágenes y otros recursos generalmente estáticos, como en este caso ficheros *.css* y *.js* para que se carguen desde ahí.

Lo metes en el `<head>` de tu página, así:

```
<!-- Cargar el CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet" :

<!-- Cargar el JS -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/js/bootstrap.bundle.min.js" integrity=";
```

Si nunca has usado los atributos *integrity* o *crossorigin* te estarás pensando que si sirven para algo.

Sí sirven. El primero es un control para verificar que el archivo en el CDN no ha sido modificado. Básicamente es más seguridad para ti. Si una mala persona va al CDN y cambia el archivo por uno malicioso, a los usuarios de tu proyecto no les pasará nada porque el archivo modificado no se cargará. Y el segundo es, básicamente, para permitir que cargues el fichero sin tener relación con el dominio del CDN.

Pero bueno, eso no tienes que tocarlo. Simplemente haz un copia y pega. Es la parte más aburrida pero es necesaria para que Bootstrap funcione.

3. Cómo crear estructuras

Sin duda esta es la parte más famosa de Bootstrap, ya que nos permite crear impresionantes estructuras:

Hola	Mira	
Qué	Bien	Puedo
Organizar	Mis	Columnas

Y no solo eso, sino que además podemos hacer que estas se adapten a cualquier tamaño de pantalla de forma muy sencilla. Así se vería el ejemplo anterior en una pantalla muy estrecha:

Hola
Mira
Qué
Bien
Puedo
Organizar
Mis
Columnas

Pero vamos a explicarlo bien para que puedas usarlo sabiendo lo que haces.

Tienes tres tipos de bloques:

- Contenedor
- Filas
- Columnas

Y esos nombres no se han elegido al azar, de hecho están muy bien escogidos porque representan a la perfección el uso de cada tipo de bloque.

La idea es tener un contenedor (puedes tener más pero casi siempre se usa un contenedor para toda la página) y dentro meterás filas. Y dentro de las filas meterás columnas.

Digamos que el **contenedor** es el super cajón grande donde va todo, luego tienes **filas**, y en cada fila puedes tener un número determinado de **columnas**.

Vamos a verlo bien uno por uno.

3.1. El contenedor

Es una clase CSS que sirve como caja grande donde meterás filas y columnas. Para usarlo simplemente coge tu elemento y añade la clase *container*.

```
<div class="container">
  Aquí dentro irán filas y columnas
</div>
```

Tendrás con ello tu cajón grande, que en principio no sirve para nada más que para meter las cosas.

Aquí dentro irán filas y columnas

Nota: el borde lo he puesto yo simplemente para que se vea.

¿Para qué sirve exactamente este cajón?

Sirve para que puedas especificar un ancho máximo en el que quieres meter tus contenidos. Porque hay pantallas que son exageradamente anchas, pero no quieres que tu web se estire hasta los filos, sino que quieres tener un contenedor centrado con todo el contenido, y márgenes a los lados.

Además, el contenedor tiene propiedades que afectan a las filas y columnas. Por eso, es importante que "respetes" este convenio de meter tu contenido en un contenedor.

3.2. Fila

Una fila es otra especie de cajón pero que se usa dentro del contenedor principal y como su nombre indica, sirve para organizar una fila de columnas.

```
<div class="container">
  <div class="row">
    Aquí irá una fila de columnas
  </div>
  <div class="row">
    Aquí irá otra fila de columnas
  </div>
</div>
```

Aquí irá una fila de columnas

Aquí irá otra fila de columnas

3.3. Columna

La columna es el elemento con el que más jugaremos, ya que es el más versátil. Vamos a verlo directamente con un ejemplo:

```
<div class="container">
  <div class="row">
    <div class="col">Una columna</div>
    <div class="col">Otra columna</div>
  </div>
</div>
```

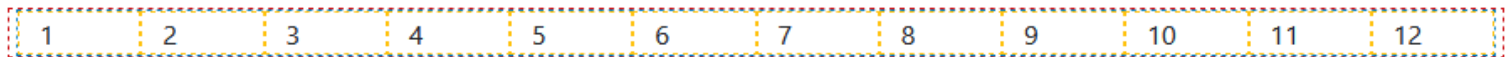


Ya has visto que si ponemos *col* sin nada más, el espacio se reparte por igual ocupando todo el ancho.

Pero... ¿qué pasa por ejemplo si queremos una columna lateral y otra más grande?

Bien. Ahí es donde entra en juego la magia de Bootstrap.

Tienes que imaginar el ancho completo como un espacio dividido en doce partes.

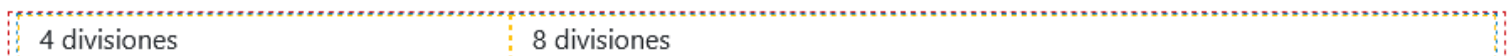


¿Por qué doce?

Pues porque así es como funciona Bootstrap, ese es el número de divisiones que han elegido sus creadores.

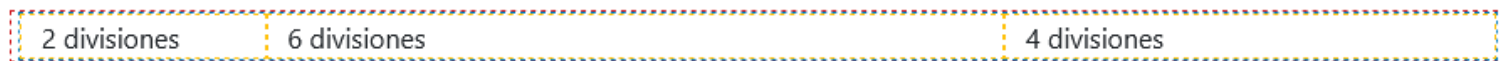
Y a la hora de meter las columnas, le agregas el número de divisiones que quieres que ocupen.

```
<div class="container">
  <div class="row">
    <div class="col-4">4 divisiones</div>
    <div class="col-8">8 divisiones</div>
  </div>
</div>
```



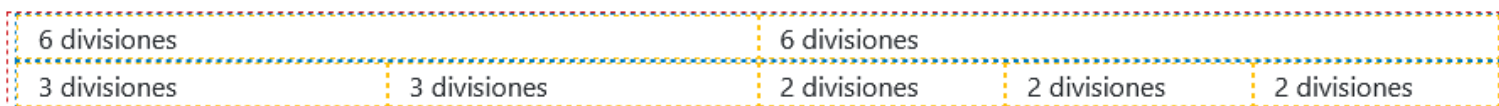
Y puedes meter las columnas que quieras, teniendo en cuenta que el ancho total es doce.

```
<div class="container">
  <div class="row">
    <div class="col-2">2 divisiones</div>
    <div class="col-6">6 divisiones</div>
    <div class="col-4">4 divisiones</div>
  </div>
</div>
```



Así de sencillo, puedes ir creando filas y columnas a tu gusto.

```
<div class="container">
  <div class="row">
    <div class="col-6">6 divisiones</div>
    <div class="col-6">6 divisiones</div>
  </div>
  <div class="row">
    <div class="col-3">3 divisiones</div>
    <div class="col-3">3 divisiones</div>
    <div class="col-2">2 divisiones</div>
    <div class="col-2">2 divisiones</div>
    <div class="col-2">2 divisiones</div>
  </div>
</div>
```



Si haces cosas extrañas, como meter una fila columnas que sumen más de doce divisiones o menos, simplemente se organizará el espacio de forma que se cumpla lo que has escrito.

4. Elementos

Los elementos de Bootstrap son también una gran maravilla y un ahorro de tiempo enorme a la hora de crear interfaces de usuario profesionales con poco esfuerzo. Desde botones hasta carruseles de imágenes, la cantidad de cosas que tienes a tu disposición es enorme.

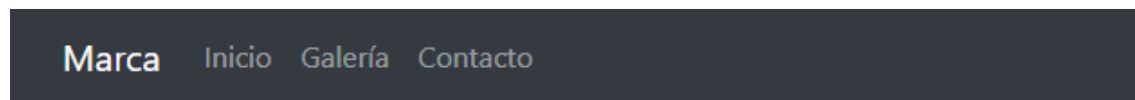
Y sí, vamos a verlas todas una por una, para que sepas qué herramientas tienes y las uses para crear esas cosas ingeniosas que se te pasan por la cabeza.

4.1. Barra de navegación

Es muy raro el caso en el que tu web no lleve una barra de navegación. Y la verdad que este elemento tan básico, cuando queremos implementarlo de forma artesana con código puro, puede llegar a ser un quebradero de cabeza.

Porque cuando el menú tiene varias opciones y queremos que se vea en el móvil, hay que implementar la típica solución de los botones que abren y cierran las opciones del menú.

Gracias a Bootstrap no tienes que preocuparte por nada de eso porque ya viene creado por defecto y funciona perfectamente.



Automáticamente se muestra la hamburguesita para abrir y cerrar el menú:



En este caso necesitamos el siguiente código. Obviamente lo interesante no es simplemente copiar y pegar sino que observes cómo está estructurado para que lo entiendas, ya que además ofrece grandes posibilidades de personalización.

```
<nav class="navbar navbar-dark navbar-expand-md bg-dark">
  <div class="container-fluid"><a class="navbar-brand" href="#">Marca</a><button data-bs-toggle="collapse"
    <div class="collapse navbar-collapse"
      id="navegacion">
        <ul class="nav navbar-nav">
          <li class="nav-item"><a class="nav-link" href="#">Inicio</a></li>
          <li class="nav-item"><a class="nav-link" href="#">Galería</a></li>
          <li class="nav-item"><a class="nav-link" href="#">Contacto</a></li>
        </ul>
      </div>
    </div>
```



```
</div>  
</nav>
```

Así de simple.

4.2. Botones

Sin duda el elemento más característico de este fantástico *framework*.



¿Cómo los creamos?

Bueno en realidad no los creas, los usas. Porque ya están creados.

Aquí tienes la lista completa:

```
<button class="btn btn-primary">Primario</button>  
<button class="btn btn-secondary">Secundario</button>  
<button class="btn btn-success">Éxito</button>  
<button class="btn btn-danger">Peligro</button>  
<button class="btn btn-warning">Alerta</button>  
<button class="btn btn-info">Info</button>  
<button class="btn btn-light">Claro</button>  
<button class="btn btn-dark">Oscuro</button>
```

Fíjate que añadimos la clase *btn* para decir que es un botón, y la clase *btn-xxx* para especificar el estilo concreto.

Nota: también puedes usar estas clases con elementos `<a>` o `<input>`. Dependerá del caso en el que te encuentres. Si estás usando un botón para un enlace normal, usarás el elemento `<a>`. Si es un botón de tipo *submit* en un formulario, usarás la etiqueta `<input>`. Pero eso son aspectos normales de HTML, lo que quiero decir es que *btn* como tal es simplemente una clase, y tú la añades al elemento que te dé la gana.

Modificadores

Hay más clases que puedes usar en conjunto con *btn*, que sirven para cambiar el tamaño o el estilo de los botones.

Tamaño:

- Usa *btn-sm* para hacerlo pequeño
- Usa *btn-lg* para hacerlo grande

Estilo:

- Usa *btn-outline-xxx* para que el estilo sea un borde exterior

Desactivado

- Usa el atributo *disabled* para que el botón esté inactivo

Ejemplos

```
<button class="btn btn-outline-success btn-sm">Éxito outline pequeño</button>
<button class="btn btn-outline-warning">Alerta outline</button>
<button class="btn btn-info btn-lg">Info grande</button>
<button class="btn btn-info btn-lg" disabled>Info grande desactivado</button>
```

Que se ven así:



La gracia es que puedes combinar las clases a tu gusto. Y eso es obvio, porque una clase CSS no es más que un conjunto de propiedades.

4.3. Alertas

A veces puede ser que quieras mostrar un aviso en tu proyecto, como por ejemplo:

- Muy bien, has realizado la operación correctamente.
- Error, te falta un dato para seguir.
- Información: esta guía de Bootstrap es perfecta para principiantes.

Puedes hacerlo usando el elemento de alertas. Por supuesto tienes a tu disposición distintos estilos, tanto para mostrar mensajes de éxito como de peligro o simplemente informativo. Hay de sobra para que elijas el que más se adapte al caso:

Mensaje de aviso con estilo primario

Mensaje de aviso con estilo secundario

Mensaje de aviso con estilo de éxito

Mensaje de aviso con estilo de peligro

Mensaje de aviso con estilo de precaución

Mensaje de aviso con estilo genérico

Mensaje de aviso con estilo claro

Mensaje de aviso con estilo oscuro

Para mostrarlos, usamos la clase `alert` junto con `alert-xxx` para definir el estilo.

Aquí te dejo el código del ejemplo de arriba:

```
<div class="alert alert-primary">
  Mensaje de aviso con estilo primario
</div>
<div class="alert alert-secondary">
  Mensaje de aviso con estilo secundario
```

```
</div>
<div class="alert alert-success">
  Mensaje de aviso con estilo de éxito
</div>
<div class="alert alert-danger">
  Mensaje de aviso con estilo de peligro
</div>
<div class="alert alert-warning">
  Mensaje de aviso con estilo de precaución
</div>
<div class="alert alert-info">
  Mensaje de aviso con estilo genérico
</div>
<div class="alert alert-light">
  Mensaje de aviso con estilo claro
</div>
<div class="alert alert-dark">
  Mensaje de aviso con estilo oscuro
</div>
```

Úsalos como más te guste.

Enlaces

Si quieres meter un enlace dentro de la alerta y hacer que tenga un estilo acorde, puedes usar la clase *alert-link*, así:

```
<div class="alert alert-success">
  Ejemplo de un <a href="#">enlace normal</a> dentro del aviso
</div>
<div class="alert alert-success">
  Ejemplo de un <a href="#" class="alert-link">enlace con estilo</a> dentro del aviso
</div>
<div class="alert alert-danger">
  Otro ejemplo de <a href="#" class="alert-link">enlace con estilo</a> dentro del aviso
</div>
```

Ejemplo de un [enlace normal](#) dentro del aviso

Ejemplo de un [enlace con estilo](#) dentro del aviso

Otro ejemplo de [enlace con estilo](#) dentro del aviso

Fíjate que si no le das ninguna clase al enlace, aparece con el estilo por defecto y no queda tan bien. Sin embargo, usando la clase *alert-link*, de repente tiene un estilo mucho más homogéneo.

4.4. Etiquetas

Las etiquetas o insignias son un elemento que te permite añadir precisamente eso, una etiqueta, en el lugar que consideres oportuno.

Titular de ejemplo **Etiqueta**

La etiqueta se adapta **Al tamaño**

Botón de ejemplo **7**

Mostrarlas es muy sencillo, simplemente usa la clase *badge* y *bg-xxx* para el estilo.

Aquí te dejo el código que he usado para el ejemplo de la imagen anterior:

```
<h1>Titular de ejemplo <span class="badge bg-secondary">Etiqueta</span></h1>
<h3>La etiqueta se adapta <span class="badge bg-secondary">Al tamaño</span></h3>
<button class="btn btn-primary">
  Botón de ejemplo <span class="badge bg-secondary">7</span>
</button>
```

Suelen usarse casi siempre para mostrar el numerito de las notificaciones o algún aviso de novedad. Pero es un elemento genérico y puedes darle el uso que creas conveniente.

Bordes redondeados

Si lo prefieres, puedes usar la clase *rounded-pill* para que se muestren los bordes redondeados.

Primario **Secundario** **Éxito** **Peligro** **Alerta** **Info** **Claro** **Oscuro**

```
<span class="badge rounded-pill bg-primary">Primario</span>
<span class="badge rounded-pill bg-secondary">Secundario</span>
<span class="badge rounded-pill bg-success">Éxito</span>
<span class="badge rounded-pill bg-danger">Peligro</span>
<span class="badge rounded-pill bg-warning text-dark">Alerta</span>
```

```
<span class="badge rounded-pill bg-info">Info</span>  
<span class="badge rounded-pill bg-light text-dark">Claro</span>  
<span class="badge rounded-pill bg-dark">Oscuro</span>
```

Así de sencillo.

4.5. Barras de progreso

Este elemento no se usa tanto, pero también te puede venir muy bien para algunos casos específicos, como por ejemplo para el porcentaje completado de un proceso.



¿Y cómo las pongo?

```
<div class="progress">  
  <div class="progress-bar" role="progressbar" aria-valuenow="0" aria-valuemin="0" aria-valuemax="100"></div>  
</div>  
<div class="progress">  
  <div class="progress-bar" role="progressbar" style="width: 25%" aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>  
</div>  
<div class="progress">  
  <div class="progress-bar" role="progressbar" style="width: 50%" aria-valuenow="50" aria-valuemin="0" aria-valuemax="100"></div>  
</div>  
<div class="progress">  
  <div class="progress-bar" role="progressbar" style="width: 75%" aria-valuenow="75" aria-valuemin="0" aria-valuemax="100"></div>  
</div>  
<div class="progress">  
  <div class="progress-bar" role="progressbar" style="width: 100%" aria-valuenow="100" aria-valuemin="0" aria-valuemax="100"></div>  
</div>
```

Fíjate que cada barra tiene por un lado el elemento padre, con la clase *progress*, y por otro lado la parte que se va rellenando, con la clase *progress-bar*.

Para darle el estilo de un *x %* completado, en realidad lo que le damos es un ancho del *x %* a la barrita de relleno.

Los otros atributos son para asignar valores:

- *aria-valuenow*: la cantidad completada
- *aria-valuemin*: la cantidad mínima
- *aria-valuemax*: la cantidad máxima

Estos se usan para lectores especiales. Por ejemplo un dispositivo que lee el contenido a un ciego, le podrá dictar la cantidad por voz.

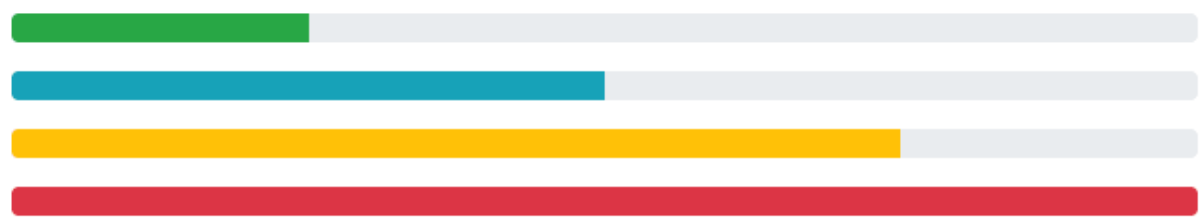
Estilos de barra

¿Se puede cambiar el estilo?

Naturalmente, ya te habrás dado cuenta de que puedes cambiar el color de fondo usando la clase *bg-xxx*. Donde *xxx* es el estilo concreto.

```
<div class="progress">
  <div class="progress-bar bg-success" role="progressbar" style="width: 25%" aria-valuenow="25" aria-valuemin="0" aria-valuemax="100"></div>
</div>
<div class="progress">
  <div class="progress-bar bg-info" role="progressbar" style="width: 50%" aria-valuenow="50" aria-valuemin="0" aria-valuemax="100"></div>
</div>
<div class="progress">
  <div class="progress-bar bg-warning" role="progressbar" style="width: 75%" aria-valuenow="75" aria-valuemin="0" aria-valuemax="100"></div>
</div>
<div class="progress">
  <div class="progress-bar bg-danger" role="progressbar" style="width: 100%" aria-valuenow="100" aria-valuemin="0" aria-valuemax="100"></div>
</div>
```

Y quedaría así:



Otra cosa que puedes hacer es agregar la clase *progress-bar-striped* para hacer que salga así:

```
<div class="progress">
  <div class="progress-bar progress-bar-striped bg-info" role="progressbar" style="width: 50%" aria-valuenow="50" aria-valuemin="0" aria-valuemax="100"></div>
</div>
```

```
</div>  
<div class="progress">  
  <div class="progress-bar progress-bar-striped bg-warning" role="progressbar" style="width: 75%" aria-valuer  
</div>
```



Fantástico, ¿verdad?

Resumen

Recapitulando sobre todo lo que hemos visto, Bootstrap no es más que un puñado de clases CSS y funcionalidades Javascript que puedes usar a tu manera, combinándolos para crear tus proyectos.

Y la forma de usarlo es asignando clases CSS a los elementos nativos de HTML. Dichas clases están muy bien nombradas, en inglés, según su utilidad.

Puedes combinar las clases como quieras pero por lo general, tendrás contenido metido en columnas, que a su vez están metidas en filas y que a su vez están metidas en un contenedor.

Es literalmente muy fácil aprender a usarlo si ya tienes unas nociones de HTML y CSS, porque en realidad no estás haciendo nada nuevo.

Conclusión

Bootstrap es una herramienta muy potente que puedes sumar a tus utilidades de desarrollo web.

Sin duda es una ayuda increíble a la hora de elaborar proyectos de forma rápida sin dejar de lado la profesionalidad.

Pero eso no significa que tengas que dejar de lado tus habilidades en CSS. De hecho, solo dominando CSS podrás sacarle realmente todo el provecho a Bootstrap, ya que esto te hará ser capaz de modificar algunas propiedades de manera puntual para darles el toque exacto que estés buscando (o el que te hayan pedido).

Y aquí se acaba esta guía, espero que te haya servido para adquirir una visión global. Mientras te aprendes y memorizas las clases más comunes puedes tener esta página abierta en otra pestaña y copiar los elementos que vayas necesitando.

APRENDE GRATIS

[Curso de Desarrollo Web](#)

CONSEJOS

[¿Por qué aprender idiomas y cómo hacerlo gratis?](#)

UTILIDADES

[Generador de CV](#)

Programas gratis (Windows, Mac, Linux)

- 1 [De diseño 3D](#)
- 2 [Para optimizar tu PC](#)
- 3 [Para comprimir vídeos](#)
- 4 [Para crear juegos](#)
- 5 [Para recuperar archivos](#)
- 6 [Para hacer esquemas](#)
- 7 [Para hacer facturas](#)
- 8 [Para comprimir archivos](#)
- 9 [Para ver fotos](#)
- 10 [Para hacer música](#)
- 11 [Para dibujar](#)

- 12 Para escribir
- 13 Para grabar la pantalla
- 14 Para editar PDF
- 15 Para editar fotos
- 16 Para editar audio
- 17 Para editar vídeos

APPS gratis (Android y iOS)

- 1 Para controlar gastos
- 2 Para organizarse
- 3 De karaoke
- 4 Para leer
- 5 Para hacer ejercicio en casa
- 6 Para editar videos
- 7 Para iPhone
- 8 Para hacer collages
- 9 Para hablar inglés
- 10 Para jugar
- 11 Para adelgazar
- 12 Para encontrar trabajo
- 13 Para viajar
- 14 Para tomar apuntes
- 15 Para controlar la regla
- 16 Para editar fotos
- 17 Para ligar