

Frameworks de diseño. Bootstrap



Además de todas las ventajas que se han ido incorporando en el diseño web con las sucesivas versiones de HTML y CSS (especialmente con HTML5 y CSS3), ciertas comunidades de desarrolladores han elaborado una serie de herramientas que permiten diseñar sitios web de una forma más integral y sencilla. Estas herramientas, denominadas comúnmente *frameworks de diseño web*, son un conjunto compuesto normalmente de:

- Estilos CSS predefinidos que podemos añadir a nuestras webs utilizando, normalmente, una serie de selectores de clase o de id predefinidos.
- Plugins JavaScript que dotan de cierta funcionalidad adicional a la web y sus componentes, a la hora de interactuar con formularios, arrastrar componentes, etc.
- Otras opciones. Por ejemplo, algunos incorporan funcionalidad en el servidor mediante páginas PHP o similares, otros traen una serie de imágenes o iconos predefinidos que utilizar en las aplicaciones, etc.

Algunos ejemplos de Frameworks populares son:

- **Bootstrap**: quizá sea el framework más popular para desarrollo web. Sus nuevas versiones se centran cada vez más en el diseño web adaptativo, y cada vez incluye más componentes que añadir a las páginas (elementos de formularios, migas de pan de navegación, etc.). Su web oficial es getbootstrap.com.
- **Ionic**: es un framework que, unido a otros frameworks de desarrollo en la parte cliente como Angular o React, permite desarrollar la apariencia de la aplicación, e incluso hacerla exportable a otras plataformas, como los dispositivos móviles, a través de aplicaciones híbridas (desarrolladas con tecnologías web, pero exportadas a otros soportes). Su web oficial es ionicframework.com.
- **Tailwind**: con una filosofía similar a Bootstrap, se utiliza en el diseño de plantillas de algunos frameworks en servidor, como Laravel. Podemos leer sobre él en su [web oficial](#).
- ...

1. Primeros pasos con Bootstrap

Bootstrap puede utilizarse tanto descargando el framework desde su web, como enlazándolo directamente desde algún repositorio online. Veamos aquí las dos opciones de enlazar con el framework, y un primer vistazo a sus posibilidades.

1.1. Descargar Bootstrap

Si queremos descargar Bootstrap, tenemos varias opciones para ello. Una opción bastante sencilla consiste en descargarlo de su web oficial (sección de [descargas](#)). Podemos descargar la versión ya compilada y preparada para usar (es menos modificable pero podemos empezar a trabajar con ella) o el código fuente de CSS y Javascript (es menos eficiente, pero lo podemos modificar mejor). En estos apuntes optaremos por la versión compilada, ya que no tenemos previsto modificar nada.

Una vez descargado el archivo comprimido, veremos dentro varios archivos y carpetas, divididos en dos grandes bloques:

- **css:** con los diferentes archivos CSS de estilos que podemos utilizar
- **js:** con los distintos archivos JavaScript necesarios para algunas funcionalidades

En nuestro caso, basta con que copiemos estas dos carpetas en nuestra web (o bien en la carpeta inicial, o bien en alguna subcarpeta como *bootstrap*, por ejemplo). Después, simplemente tenemos que enlazar con el archivo `bootstrap.min.css` de la subcarpeta *css*. Por ejemplo:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Ejemplo Bootstrap</title>
    <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">
  </head>
  ...
```

Opcionalmente (aunque suele ser habitual) podemos enlazar también con el archivo JavaScript al final del cuerpo del documento, para cargar las funcionalidades añadidas:

```
...
  <script type="text/javascript" src="bootstrap/js/bootstrap.min.js"></script>
</body>
</html>
```

1.2. Enlazar con Bootstrap desde una fuente externa

Alternativamente, podemos no descargar Bootstrap y enlazar con alguna versión oficial en Internet. En este caso, los enlaces al CSS y al JavaScript pueden quedar como sigue (modificando la URL en función de la versión de Bootstrap que queramos usar), tal y como puede verse en la [documentación oficial](#).

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Ejemplo Bootstrap</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
      integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2Qv
      rel="stylesheet" crossorigin="anonymous">
  </head>
  <body>
    ...
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
      integrity="sha384-ka7Sk0Gln4gmtz2MlQnik11wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTc
      crossorigin="anonymous"></script>
  </body>
</html>
```

1.3. Uso de Bootstrap junto con estilos propios

Tras el paso anterior, ya podríamos ver que el contenido de nuestra página tiene un aspecto diferente al que tiene por defecto en un navegador, gracias a los estilos de Bootstrap. Por ejemplo, se habrá cambiado el tipo de fuente por defecto. Sin embargo, en ocasiones puede que nos interese modificar ese estilo predefinido, y aportar algo propio: colores de fondo, bordes a algunos componentes, etc. En este caso, podemos incluir nuestra propia hoja de estilos CSS, que deberá ubicarse siempre después de haber incorporado los estilos de Bootstrap, para evitar problemas si hubiera algún conflicto con algún estilo:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Ejemplo Bootstrap</title>
    <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">
    <link rel="stylesheet" href="css/estilos.css">
  </head>
  ...
```

NOTA: en el ejemplo anterior, se asume que todo el contenido de Bootstrap está en una subcarpeta *bootstrap* de nuestro proyecto, y los estilos propios están en una carpeta *css*, externa a Bootstrap.

2. Diseño *responsive* con Bootstrap. La rejilla

Una vez enlazada la hoja de estilos CSS, ya podremos utilizar algunos selectores de clase de Bootstrap que nos permiten colocar diferentes elementos de contenido. Veremos que Bootstrap estructura el área de contenido de la página en una especie de rejilla, de forma que cada pieza de contenido puede ocupar las casillas que queramos. Aquí tenemos un ejemplo, que podemos probar en el navegador y cambiar el tamaño de la ventana para ver cómo se reajusta el contenido:

```
<!DOCTYPE html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Ejemplo Bootstrap</title>
  <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">
</head>
<body>
  <div class="container">
    <div class="row">
      <div class="col-sm-4">
        <h2>Columna 1</h2>
        <p>Texto de prueba para esta primera columna.</p>
      </div>
      <div class="col-sm-4">
        <h2>Columna 2</h2>
        <p>Texto de prueba para esta segunda columna.</p>
        <p>Un segundo párrafo para esta columna de prueba.</p>
      </div>
      <div class="col-sm-4">
        <h2>Columna 3</h2>
        <p>Tercera y última columna.</p>
      </div>
    </div>
  </div>
</body>
</html>
```

La apariencia que tendrá esta página en el navegador será aproximadamente ésta:

Columna 1

Texto de prueba para esta primera columna.

Columna 2

Texto de prueba para esta segunda columna.

Columna 3

Tercera y última columna.

Un segundo párrafo para esta columna de prueba.

Hemos utilizado algunos selectores de clase de los muchos que ofrece el CSS de Bootstrap.

El selector `container` permite definir una caja que contenga a todo nuestro contenido, estableciendo unos márgenes de separación con el borde de la página. Existe una alternativa, que es el selector `container-fluid`, que no pone márgenes laterales para que el contenido ocupe todo lo que pueda.

El selector `row` sirve para estructurar el contenido de la página en filas, de forma que cada fila estará compuesta por una serie de columnas, y esas columnas se autoajustarán según el tamaño de la pantalla. Pondremos un selector `row` para cada fila que queramos añadir de contenido en nuestra página.

En cuanto a las columnas, hay diferentes estilos que empiezan por la palabra `col`, seguida de un tamaño (*xs*, *sm*, *md*, *lg*, *xl* o *xxl*, según hasta qué resolución queramos mantener las columnas, aunque en la versión 5 de Bootstrap el sufijo *xs* se omite), y un número, que indica cuántas celdas (de un total de 12 por fila) va a ocupar esa columna. Así, en el ejemplo anterior, con el selector `col-sm-4`, estamos utilizando columnas para dispositivos relativamente pequeños (hasta unos 768px), de forma que cada columna ocupará 4 de 12 celdas (por tanto, caben 3 columnas). En el momento en que el tamaño de la ventana baje de 768px, se deshacen las columnas y se apilan una debajo de otra, ocupando cada una el 100% del ancho de pantalla disponible. Podemos hacer que una caja se comporte como varios de estos tipos, según la resolución disponible en cada momento:

```
<div class="col-sm-4 col-6">...</div>
```

Esta caja ocupa 4 rejillas para dispositivos pequeños (< 768px) o superiores, y 6 rejillas para dispositivos extra pequeños (*xs*, es decir, < 576px). Es decir, se verá a 3 columnas en pantallas pequeñas o superiores, y a 2 columnas en pantallas extra-pequeñas.

4.1. Más opciones de posicionamiento de elementos

Los selectores `col` vistos sirven para distribuir las diferentes columnas de cada fila. Sin embargo, el contenido de estas celdas se ve "pegado", es decir, no hay separación o *gap* por defecto entre las celdas de la rejilla. Para definir espaciados entre los elementos (ya sean celdas de rejilla u otros componentes), Bootstrap dispone de otros selectores de clase que podemos utilizar.

Por ejemplo, el selector `p` sirve para establecer un *padding* interno en el componente que elijamos. Detrás de este prefijo `p` podemos añadir una serie de elementos que definen las características de dicho *padding*. Así, por ejemplo, si especificamos `pt-0` estamos diciendo que no queremos *padding* superior (*t* = *top*). Bootstrap tiene unos tamaños predefinidos numerados, de forma que `p-1` es el *padding* más pequeño que tiene definido, y `p-5` el más grande.

Del mismo modo, el selector `m` sirve para establecer un margen externo en el componente sobre el que se aplica, con los mismos sufijos disponibles que para el *padding*.

Echemos un vistazo a este ejemplo:

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css">
    <link rel="stylesheet" href="estilos.css">
    <title>Ejemplo Bootstrap</title>
  </head>
  <body>
    <div class="container">
      <h1>Prueba de rejilla Bootstrap</h1>
      <div class="row m-2">
        <div class="col-md-3 m-1 p-2 caja">
          <h2>Caja 1</h2>
          <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit.
            Dolores eligendi dolorem iure. Expedita, pariatur in.</p>
        </div>
        <div class="col-md-5 m-1 p-2 caja">
          <h2>Caja 2</h2>
          <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
            Dolorem totam voluptate repellat sunt rem neque quia
            distinctio quo nulla esse.</p>
        </div>
        <div class="col-md-3 m-1 p-2 caja">
          <h2>Caja 3</h2>
          <p>Lorem ipsum dolor sit amet consectetur adipisicing elit.
            Earum accusantium quod nesciunt porro reiciendis architecto
            nobis dolorem maxime! Ipsam accusamus nisi dolorem
            consequuntur sit nihil voluptatibus, libero animi autem eius.</p>
        </div>
      </div>
    </div>
    <script src="bootstrap/js/bootstrap.min.js"></script>
  </body>
</html>
```

Si definimos en el selector *caja* un color de fondo (por ejemplo, naranja) podemos ver cómo queda delimitada cada rejilla. En este caso, hemos añadido un margen externo de tipo 1 (por todos lados) y un padding interno de tipo 2, obteniendo este resultado:

Prueba de rejilla Bootstrap

Caja 1

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Dolores eligendi dolorem iure. Expedita, pariatur in.

Caja 2

Lorem ipsum dolor sit amet consectetur adipisicing elit. Dolorem totam voluptate repellat sunt rem neque quia distinctio quo nulla esse.

Caja 3

Lorem ipsum dolor sit amet consectetur adipisicing elit. Earum accusantium quod nesciunt porro reiciendis architecto nobis dolorem maxime! Ipsam accusamus nisi dolorem consequuntur sit nihil voluptatibus, libero animi autem eius.

Notar que, al definir un margen, la rejilla ya no puede albergar 12 filas (el margen ocupa un espacio extra adicional), y la suma de columnas de los elementos `col` en el ejemplo anterior es 11 para que quepan en la misma fila.

Podéis leer más sobre Bootstrap y los usos de la rejilla en su [documentación oficial](#). También podéis investigar las opciones de espaciado de elementos [aquí](#).

Ejercicio 1:

Descarga [este ejemplo](#) que ya hemos utilizado en secciones anteriores. Añade una hoja propia `estilos.css` simplemente para dar un color de fondo a las cajas. Ahora vamos a utilizar la rejilla de Bootstrap para que:

- Las cuatro cajas se muestren una junto a otra en dispositivos tamaño *lg* o superior
- Las cajas se muestren de dos en dos en tamaño *sm* o superior
- Las cajas se muestren una bajo la otra en dispositivos más pequeños

Define un margen entre las cajas para separarlas, y asigna a cada caja una anchura en celdas apropiada para su contenido, a tu gusto.

3. Formularios Bootstrap

Bootstrap dispone de unos pocos selectores de clase que nos ayudan a dotar de un estilo diferente a nuestros formularios. Por ejemplo, podemos utilizar los selectores de clase `form-label` para etiquetas y `form-control` para el resto de controles. Observemos la diferencia...

Así definiríamos un formulario normal y corriente sin hacer uso de Bootstrap:

```
<form action="mipagina.php" method="post">
  <label for="nombre">Introduce tu nombre:</label>
  <br>
  <input type="text" name="nombre" id="nombre" size="20">
  <br>

  <label for="email">Introduce tu e-mail:</label>
  <br>
  <input type="email" name="email" id="email" size="20" maxlength="50">
  <br>

  <input type="checkbox" name="casado" id="casado">
  <label for="casado">Casado</label>
  <br>

  <label>Elige tu idioma preferido:</label>
  <br>
  <input type="radio" name="idioma" id="ingles" value="ingles" checked>
  <label for="ingles">Inglés</label>
  <br>
  <input type="radio" name="idioma" id="frances" value="frances">
  <label for="frances">Francés</label>
  <br>

  <input type="submit" value="Enviar datos">
</form>
```

Y su aspecto visual sería éste:

Introduce tu nombre:

Introduce tu e-mail:

Casado

Elige tu idioma preferido:
 Inglés
 Francés

Ahora vamos a añadir la clase *form-label* a las etiquetas y *form-control* a los *inputs* de tipo texto (nombre y email) y al botón de enviar:

```
<form action="mipagina.php" method="post">
  <label class="form-label" for="nombre">Introduce tu nombre:</label>
  <br>
  <input class="form-control" type="text" name="nombre" id="nombre" size="20">
  <br>

  <label class="form-label" for="email">Introduce tu e-mail:</label>
  <br>
  <input class="form-control" type="email" name="email" id="email" size="20"
    maxlength="50">
  <br>

  <input type="checkbox" name="casado" id="casado">
  <label class="form-label" for="casado">Casado</label>
  <br>

  <label class="form-label">Elige tu idioma preferido:</label>
  <br>
  <input type="radio" name="idioma" id="ingles" value="ingles" checked>
  <label class="form-label" for="ingles">Inglés</label>
  <br>
  <input type="radio" name="idioma" id="frances" value="frances">
  <label class="form-label" for="frances">Francés</label>
  <br>

  <input class="form-control" type="submit" value="Enviar datos">
</form>
```

Ahora la apariencia es la siguiente:

Introduce tu nombre:

Introduce tu e-mail:

Casado

Elige tu idioma preferido:

Inglés

Francés

Puedes leer más sobre el diseño de formularios con Bootstrap [aquí](#).

En cuanto a los botones, Bootstrap también ofrece algunos selectores de clase para definir distintos tipos de botones, con distintas apariencias y colores. Por ejemplo, este botón se verá con texto blanco y fondo azul, al

ser de tipo *primary*:

```
<input type="submit" value="Enviar datos" class="form-control btn btn-primary">
```

En general, tenemos disponible la siguiente gama de clases de botones (y veremos a continuación que esos mismos selectores se pueden aplicar también a otros componentes):



Ejercicio 2:

Descarga [este formulario](#) y aplícale los estilos facilitados por Bootstrap para poderlo visualizar de este modo:

Formulario de registro

Nombre:	<input type="text" value="Escribe tu nombre"/>	Provincia:	<input type="text" value="Alicante"/>
E-mail:	<input type="text" value="Escribe tu e-mail"/>	<input type="checkbox"/>	Deseo suscribirme al boletín de noticias
Password:	<input type="text" value="Al menos 8 caracteres entre letras y dígitos"/>	<input type="button" value="Enviar"/>	

4. Algunos componentes de Bootstrap

Para finalizar este breve repaso introductorio al framework Bootstrap, veamos algunos de los componentes predefinidos que podemos añadir en nuestras páginas.

4.1. Barras de navegación

Las barras de navegación son un componente importante en el diseño de cualquier página. Contienen un conjunto de enlaces que nos permiten visitar las páginas más relevantes del sitio. Bootstrap ofrece una amplia gama de selectores de clase que podemos utilizar para definir un estilo personalizado en nuestras barras. Podemos consultar toda la información disponible [aquí](#).

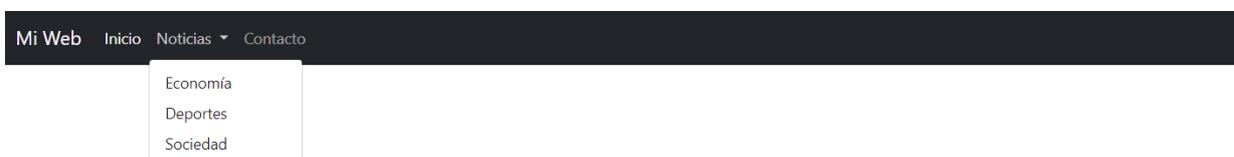
Por ejemplo, así podemos definir una barra de navegación con tres apartados: *Inicio*, *Noticias* y *Contacto*. Dentro de *Noticias* definimos un *dropdown menu* (menú desplegable) con las diferentes categorías de noticias que podemos consultar.

```

<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
  <div class="container-fluid">
    <a class="navbar-brand" href="#">Mi Web</a>
    <button class="navbar-toggler" type="button"
      data-bs-toggle="collapse" data-bs-target="#navbarScroll"
      aria-controls="navbarScroll" aria-expanded="false"
      aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarScroll">
      <ul class="navbar-nav me-auto my-2 my-lg-0 navbar-nav-scroll"
        style="--bs-scroll-height: 100px;">
        <li class="nav-item">
          <a class="nav-link active" aria-current="page"
            href="inicio.html">Inicio</a>
        </li>
        <li class="nav-item dropdown">
          <a class="nav-link dropdown-toggle" href="#"
            id="navbarScrollingDropdown" role="button"
            data-bs-toggle="dropdown" aria-expanded="false">
            Noticias
          </a>
          <ul class="dropdown-menu"
            aria-labelledby="navbarScrollingDropdown">
            <li><a class="dropdown-item"
              href="economia.html">Economía</a></li>
            <li><a class="dropdown-item"
              href="deportes.html">Deportes</a></li>
            <li><a class="dropdown-item"
              href="sociedad.html">Sociedad</a></li>
          </ul>
        </li>
        <li class="nav-item">
          <a class="nav-link" aria-current="page"
            href="contacto.html">Contacto</a>
        </li>
      </ul>
    </div>
  </div>
</nav>

```

El aspecto visual sería éste, ya que hemos elegido el tema *dark*:



NOTA: Es **IMPORTANTE** tener en cuenta que, para que algunas funcionalidades como el *dropdown* funcionen, será necesario cargar el componente JavaScript de Bootstrap en la página, como se ha visto en los ejemplos anteriores (normalmente al final de la página).

4.2. Alertas e insignias

Dos componentes muy sencillos de utilizar son las alertas (*alert*) y las insignias (*badges*). Los primeros permiten mostrar mensajes con un color de fondo particular en ciertas partes de la web. Pueden resultar útiles para mostrar mensajes de error o de advertencia en algunos puntos. Por ejemplo, un error en un formulario:

```
<form action="mipagina.php" method="post">
  <label class="form-label" for="nombre">Introduce tu nombre:</label>
  <input class="form-control" type="text" name="nombre" id="nombre" size="20">
  <div class="alert alert-danger">Error: el nombre no puede estar vacío</div>
```

que se vería de este modo:

Introduce tu nombre:

Error: el nombre no puede estar vacío

Existen diferentes colores para las alertas, similares a los colores para los botones vistos antes (*primary*, *secondary*, *success*, *warning*, *info*...).

En cuanto a las insignias o *badges*, permiten mostrar un pequeño texto resaltado en ciertas partes de la web. Podemos utilizarlo, por ejemplo, para indicar contenido nuevo:

```
<p>
  Ya a la venta nuestro nuevo curso de diseño web para principiantes
  <span class="badge badge-warning">Nuevo!</span>
</p>
```

Ya a la venta nuestro nuevo curso de diseño web para principiantes **Nuevo!**

4.3. Más componentes

Existe una amplia variedad de componentes ofrecidos por Bootstrap. Desde elementos simples como migas de pan o listas, hasta elementos más complejos como carruseles de diapositivas o ventanas modales. Puedes consultar información sobre todos ellos en la [documentación oficial](#).