

## **MINI MANUAL DE BOOTSTRAP 3**

## La base fundamental

Bootstrap es un framework JS/CSS de Twitter que facilita el desarrollo de sitios web responsive, que se adaptan al dispositivo donde se muestran. Para ello hace uso de 3 componentes fundamentales que van incluidos en todos los HTML que lo usan:

1.- La librería JQuery de Javascript

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
```

2.- La librería de estilos CSS de Bootstrap

```
<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
```

3.- La librería Javascript de Bootstrap

```
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
```

Toda página que use este framework, por tanto siempre incluirá en su código estos 3 componentes. A parte de esto, otras 3 partes importantes deben de aparecer en el código HTML:

1.- El Doctype de HTML5 UTF-8:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
  </head>
</html>
```

2.- El meta tag específico para adaptación de dispositivos móviles, en <head>:

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

3.- El elemento contenedor que envolverá al resto de componentes Bootstrap (el contenedor .container)

```
<div class="container">
</div>
```

### Esqueleto base de una web Bootstrap:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.0/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
</head>
<body>
<div class="container">
  </div>
</body>
</html>
```

## Particularidades del elemento container

- Hay 2 tipos diferentes de containers en bootstrap:

- a) .container => contenedor responsive de anchura fija para cada tamaño de pantalla
  - b) .container-fluid => contenedor de ancho elástico que se expande ocupando todo el viewport
- Los contenedores no son anidables, es decir no puede haber uno dentro de otro.

## El grid de Bootstrap

Bootstrap permite un sistema de grid de hasta 12 columnas a lo largo de la página.

Las columnas pueden juntarse para ocupar varias columnas, siempre que sumen hasta 12 por fila.

El sistema de grid es responsive y se ajusta al tamaño de la pantalla del dispositivo.

Hay 1 clase de grid para cada tamaño/dispositivo:

- xs (para móviles < 768 px)
- sm (para tabletas >= 768 px)
- md (para portátiles >= 992 px)
- lg (para pc sobremesa >= 1200 px)

Pueden combinarse para crear disposiciones más dinámicas.

Este es un ejemplo de grid:

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

Es una fila, con 3 celdas de 4 columnas cada una, sumando 12 en total.

¿Qué significa que sea de la clase -sm- ?. Pues que se mantendrá en la misma fila si el tamaño de la pantalla es de tabletas para arriba, y solamente colapsará a mostrarse en 3 filas si el tamaño es menor ( o sea, de móvil).

## Reglas del sistema de Grid

- Las filas (row) deben ir dentro de un container o container-fluid para su correcta alineación
- Use filas (row) para crear grupos de columnas horizontales
- El contenido debe de estar dentro de las columnas, y solamente las columnas deben ser los hijos inmediatos de las filas (nada debe haber entre ambos nunca)
- Las columnas crean canalones (pequeños espacios entre ellas) via padding. 30 px

## Ejemplo de mezcla de clases de diferente tamaño

```
<div class="container-fluid">
  <h1>Hola mundo!</h1>
  <div class="row">
    <div class="col-sm-3 col-md-6" style="background-color:yellow;">
      <p>Lorem ipsum...</p>
    </div>
    <div class="col-sm-9 col-md-6" style="background-color:pink;">
      <p>Sed ut perspiciatis...</p>
    </div>
  </div>
</div>
```

Usando las reglas del Grid, podemos entender en este ejemplo la combinación de clases de diferente tamaño dentro de una misma columna. Cuando la pantalla es -md- o superior, ambas columnas tienen el mismo tamaño de 6. Cuando la pantalla es inferior hasta -sm- la segunda es el triple que la primera y se mantienen en la misma fija. Cuando la pantalla es inferior a -sm- entonces pasan a apilarse una sobre la otra, y ocuparan ambas todo el ancho por lo que serán igual de anchas.

## **Tipografía**

Son las clases y tags que tratan todo lo referente al texto y la forma en como mostrarlo.

### **Tags**

- h1 ... h6 => 36 px, 30 px, 24 px, 18 px, 14 px, 12 px (respectivamente)
- small => tamaño inferior para texto secundario
- mark => para remarcar un texto
- abbr => abreviatura
- blockquote => bloque cita
- dl => lista de descripciones (dt – item / dd – descripción)
- code => código
- kbd => teclas
- pre => preserva el tamaño, saltos de líneas y espacios del texto original

### **Clases**

- text-primary, text-success, text-info, text-warning, text-danger => color del texto
- bg-primary, bg-success, bg-info, bg-warning, bg-danger => color del fondo
- text-left, text-center, text-right, text-justify => alineación
- text-lowercase, text-uppercase, text-capitalize
- list-unstyled => lista sin estilo, no aplicable a los hijos de estos (ul, ol)
- list-inline => lista con todos los elementos en una sola línea
- dl-horizontal => dl con dt/dd en horizontal una al lado de la otra

## **Tablas**

Son las clases que definen el comportamiento de las tablas. Las tablas de bootstrap tienen un ligero padding y solo líneas divisoras horizontales.

### **Clases**

- table => añade el estilo básico para tablas
- table-stripped => tabla con líneas de cebra
- table-bordered => tabla con bordes en todas las celdas <td>
- table-hover => tabla que resalta filas con el ratón
- table-condensed => tabla con la mitad de padding que lo normal
- active, success, info, warning, danger => contextuales para <tr> o <td>
- table-responsive => genera una barra scroll para -sm- e inferiores (ponerlo en un <div> contenedor de la tabla a convertir en responsive, nunca como subclase de <table>)

## **Visibilidad de elementos**

Son clases que hacen al elemento visible o invisible dependiendo del tamaño de la pantalla

- visible-xs/-sm/-md/-lg => dice con que tamaño el elemento es visible
- hidden-xs/-sm/-md/-lg => dice con que tamaño un elemento es ocultado

## Imágenes

Son las clases que definen la forma y comportamiento de las imágenes.

- img-rounded => con esquinas redondeadas
- img-circle => en forma de círculo
- img-thumbnail => miniatura
- img-responsive => aplica display:block; max-width:100%; height: auto por tanto se ajustará al tamaño del elemento padre

### **Embeds responsive para video**

```
<!-- 16:9 aspect ratio -->
<div class="embed-responsive embed-responsive-16by9">
  <iframe class="embed-responsive-item" src="..."></iframe>
</div>
<!-- 4:3 aspect ratio -->
<div class="embed-responsive embed-responsive-4by3">
  <iframe class="embed-responsive-item" src="..."></iframe>
</div>
```

Estas clases son aplicables a elementos: <iframe> <embed> <video> <object>

## Alertas

Son clases añadidas a la clase .alert que definen el tipo de cartel de alerta a generar.

- alert-success/-info/-warning/-danger => éxito, información, aviso, peligro

### **Alertas descartable**

Para que una alerta se pueda cerrar como descartable, se requiere definirla como .alert-dismissible y dentro en un link definir la x (&times;) de la clase close tal como esto:

```
<div class="alert alert-success alert-dismissible">
  <a href="#" class="close" data-dismiss="alert" aria-label="close">&times;</a>
  <strong>Success!</strong> Indicates a successful or positive action.
</div>
```

## Botones

Clases que definen propiedades de la clase .btn genérica aplicables a los tags <a> <button> <input>

### **Estilos**

- btn-primary/-success/-info/-warning/-danger/-link

### **Tamaños de botón**

- btn-xs/-sm/-md/-lg => no tiene nada que ver con el tamaño de pantalla
- btn-block => ocupan el ancho del elemento padre
- active/disabled => activo o deshabilitado

## Grupos de botones

Al meter varios botones en un <div> de clase .btn-group, se ponen juntos en una fila.

El propio grupo puede tener un tamaño .btn-group-xs/-sm/-md/-lg

La clase .btn-group-vertical crea un grupo vertical de botones

## Barras de navegación

La barra de navegación es una cabecera situada en lo alto de la web usada para navegar por el contenido de la misma.

### Forma standard de la barra

```
<nav class="navbar navbar-default">
  <div class="container-fluid">
    <div class="navbar-header">
      <a class="navbar-brand" href="#">WebSiteName</a>
    </div>
    <ul class="nav navbar-nav">
      <li class="active"><a href="#">Home</a></li>
      <li><a href="#">Page 1</a></li>
      <li><a href="#">Page 2</a></li>
      <li><a href="#">Page 3</a></li>
    </ul>
  </div>
</nav>
```

Vamos a ver de qué partes se compone:

- Un tag <nav> de la clase .navbar y la sub-clase navbar-default
- Un <div> container que contiene todo el contenido de la barra
- Una lista <ul> de la clase .nav y sub-clase .navbar-nav con las opciones como elementos hijo
- Los elementos <li> activos deben ser de la clase .active
- Cada elemento <li> es un link <a> para llevarnos a una parte de la web concreta
- Antes de las opciones listadas, se pueden poner otros elementos, como en este caso, un .nav-header con un link .nav-brand.

### Barras de navegación más avanzadas

```
<nav class="navbar navbar-inverse">
  <div class="container-fluid">
    <div class="navbar-header">
      <a class="navbar-brand" href="#">WebSiteName</a>
    </div>
    <ul class="nav navbar-nav">
      <li class="active"><a href="#">Home</a></li>
      <li><a href="#">Page 1</a></li>
      <li><a href="#">Page 2</a></li>
    </ul>
    <ul class="nav navbar-nav navbar-right">
      <li><a href="#"><span class="glyphicon glyphicon-user"></span> Sign Up</a></li>
      <li><a href="#"><span class="glyphicon glyphicon-log-in"></span> Login</a></li>
    </ul>
  </div>
</nav>
```

Este ejemplo lleva algunos elementos nuevos:

- La clase de la <nav> es .navbar-inverse que invierte el color de la barra y sus elementos
- Se ha añadido otro bloque <ul> de sub-clase .navbar-right alineado a la derecha con glyphsicons para darle mas vistosidad

### Barras de navegación colapsables

Cuando las barras de navegación se muestran en dispositivos pequeños, deben de poder colapsar en un botón que contenga todos sus elementos.

```
<nav class="navbar navbar-inverse navbar-fixed-top">
  <div class="container-fluid">
    <div class="navbar-header">
      <button type="button" class="navbar-toggle" data-toggle="collapse" data-
target="#myNavbar">
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
      <a class="navbar-brand" href="#">WebSiteName</a>
    </div>
    <div class="collapse navbar-collapse" id="myNavbar">
      <ul class="nav navbar-nav">
        <li class="active"><a href="#">Home</a></li>
        <li><a href="#">Page 1</a></li>
        <li><a href="#">Page 2</a></li>
        <li><a href="#">Page 3</a></li>
      </ul>
      <ul class="nav navbar-nav navbar-right">
        <li><a href="#"><span class="glyphicon glyphicon-user"></span> Sign Up</a></li>
        <li><a href="#"><span class="glyphicon glyphicon-log-in"></span> Login</a></li>
      </ul>
    </div>
  </div>
</nav>
```

Los añadidos son:

- En la cabecera .nav-header se ha introducido un botón de clase .navbar-toogle para que solo se vea cuando por tamaño de pantalla colapsa. La acción que lleva es data-toggle = collapse y el target es el id de la barra completa, que se ha metido en un <div> del tipo .collapse navbar-collapse
- Al <nav> le hemos añadido la clase .navbar-fixed-top para que siempre se mantenga arriba, aunque el contenido de la página se haga scroll.
- Podemos cambiar la linea (<li><a href="#">Page 1</a></li>) por un menu dropdown

```
<li class="dropdown">
  <a class="dropdown-toggle" data-toggle="dropdown" href="#">Page 1 <span
class="caret"></span></a>
  <ul class="dropdown-menu">
    <li><a href="#">Page 1-1</a></li>
    <li><a href="#">Page 1-2</a></li>
    <li><a href="#">Page 1-3</a></li>
  </ul>
</li>
```

- Al lado del título Page 1 ponemos un `<span>` de clase `.caret` para dibujar el triángulo hacia abajo que indica la existencia de un submenú.
- Los 3 `<span>` `.icon-bar` del botón collapse muestras 3 rayas horizontales superpuestas que indican que es un botón de menú para móviles.

## Formularios

- `<input>` `<textarea>` `<select>` con `.form-control` ocupan un ancho del 100%
- Envuelve los labels y controles con un `<div>` `.form-group`

El `<form>` por defecto es de **disposición vertical**:

```
<form>
  <div class="form-group">
    <label for="email">Email address:</label>
    <input type="email" class="form-control" id="email">
  </div>
  <div class="form-group">
    <label for="pwd">Password:</label>
    <input type="password" class="form-control" id="pwd">
  </div>
  <div class="checkbox">
    <label><input type="checkbox"> Remember me</label>
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```

Para dispositivos de al menos 768px se puede usar la **disposición inline** añadiendo a `<form>` la clase `.form-inline`.

```
<form class="form-inline">
  <div class="form-group">
    <label for="email">Email address:</label>
    <input type="email" class="form-control" id="email">
  </div>
  <div class="form-group">
    <label for="pwd">Password:</label>
    <input type="password" class="form-control" id="pwd">
  </div>
  <div class="checkbox">
    <label><input type="checkbox"> Remember me</label>
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```



Si queremos que los labels vayan al lado de su control y alineados, se usa la **disposición horizontal**.

```
<form class="form-horizontal">
  <div class="form-group">
    <label class="control-label col-sm-2" for="email">Email:</label>
    <div class="col-sm-10">
      <input type="email" class="form-control" id="email" placeholder="Enter email">
    </div>
  </div>
  <div class="form-group">
    <label class="control-label col-sm-2" for="pwd">Password:</label>
    <div class="col-sm-10">
      <input type="password" class="form-control" id="pwd" placeholder="Enter password">
    </div>
  </div>
  <div class="form-group">
    <div class="col-sm-offset-2 col-sm-10">
      <div class="checkbox">
        <label><input type="checkbox"> Remember me</label>
      </div>
    </div>
  </div>
  <div class="form-group">
    <div class="col-sm-offset-2 col-sm-10">
      <button type="submit" class="btn btn-default">Submit</button>
    </div>
  </div>
</form>
```

Para ellos usamos .form-horizontal en <form> y .control-label en los <label>

Para alinear los checkbox y botones que no llevan labels, usamos col-\*-offset-\* para saltarnos columnas en la alineación horizontal.

.input-group, es un contenedor que añade un icono, texto o boton al form-control

.input-group-addon, adjunta el icono o texto al control

.input-group-btn, adjunta un botón a un input (p.ej un search)

```
<form>
  <div class="input-group">
    <span class="input-group-addon"><i class="glyphicon glyphicon-user"></i></span>
    <input id="email" type="text" class="form-control" name="email" placeholder="Email">
  </div>
  <div class="input-group">
    <span class="input-group-addon"><i class="glyphicon glyphicon-lock"></i></span>
    <input id="password" type="password" class="form-control" name="password"
placeholder="Password">
  </div>
  <div class="input-group">
    <span class="input-group-addon">Text</span>
    <input id="msg" type="text" class="form-control" name="msg" placeholder="Additional Info">
  </div>
</form>
```

## Estados de los controles

- Entradas deshabilitadas => añadir **disabled** al final del tag <input .... **disabled**>
- Campos de solo lectura => añadir **readonly** al final del tag <input .... **readonly**>

## Tamaños de las entradas

- .input-xs/-sm/-md/-lg => tamaño vertical de <input>
- .form-group- xs/-sm/-md/-lg => tamaño de labels y controles a la vez

Se pueden meter en una fila con un tamaño de columnas concreto, usando form-group row, y usando col-\*-\*:

```
<div class="form-group row">
  <div class="col-xs-2">
    <label for="ex1">col-xs-2</label>
    <input class="form-control" id="ex1" type="text">
  </div>
  <div class="col-xs-3">
    <label for="ex2">col-xs-3</label>
    <input class="form-control" id="ex2" type="text">
  </div>
  <div class="col-xs-4">
    <label for="ex3">col-xs-4</label>
    <input class="form-control" id="ex3" type="text">
  </div>
</div>
```

## Mas elementos de Bootstrap

- Jumbotron => Caja grande con elementos a resaltar
- Wells => Añade un borde redondeado al elemento
- Glyphicons => Iconos especiales <span> e <i>
- Badges => Indicadores numéricos adjuntos
- ProgressBars => Barras de progreso
- Pagination => Paginadores numéricos
- Pager => Next y Previous
- Panels => Caja bordeada con padding
- Dropdowns => Es un menu toggle
- Collapse => Elementos de colapsan ante acciones
- Tabs/Pills => Pestañas y tickets
- Carousel => Carrusel de fotos con controles

## Sitios interesantes:

- <https://hackerthemes.com/bootstrap-cheatsheet/> => chuleta con todos los términos
- <https://www.w3schools.com/bootstrap/default.asp> => toda la teoría explicada con ejemplos