



## BOOTSTRAP

© 2017, ACTIBYTI PROJECT SLU, Barcelona  
Autor: Ricardo Ahumada



MINISTERIO  
DE ENERGÍA, TURISMO  
Y AGENDA DIGITAL

red.es



ESTRATEGIA DE  
EMPRENDIMIENTO Y  
EMPLEO JUVENIL  
*garantía juvenil*



UNIÓN EUROPEA

Fondo Social Europeo  
“El FSE invierte en tu futuro”

# ÍNDICE DE CONTENIDOS

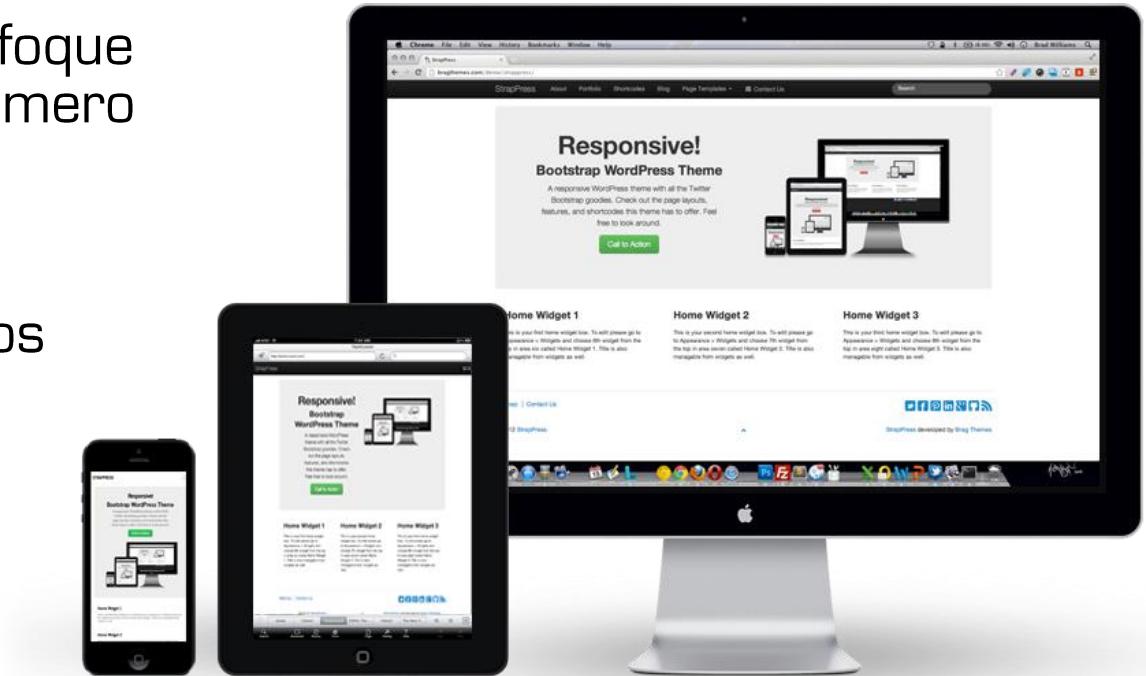
1. Implantaciones y configuración
2. Contenedores y sistemas de cuadrícula
3. Bootstrap y la interfaz de usuario: LESS, texto,  
Clases de transformación
4. Clases, abreviaturas, direcciones
5. Menús, botones y formularios
6. Jumbotron, objetos multimedia y plugins JavaScript

1

# Implantaciones y configuración

# Bootstrap

- Bootstrap es un framework HTML, CSS y JavaScript para crear sitios web responsive.
- Bootstrap proporciona clases CSS responsive, componentes CSS y componentes JavaScript.
- Bootstrap utiliza el enfoque **mobile first**: crear primero el web site para los dispositivos móviles y después mejorar y enriquecerlo para otros dispositivos grandes.



# Introducción

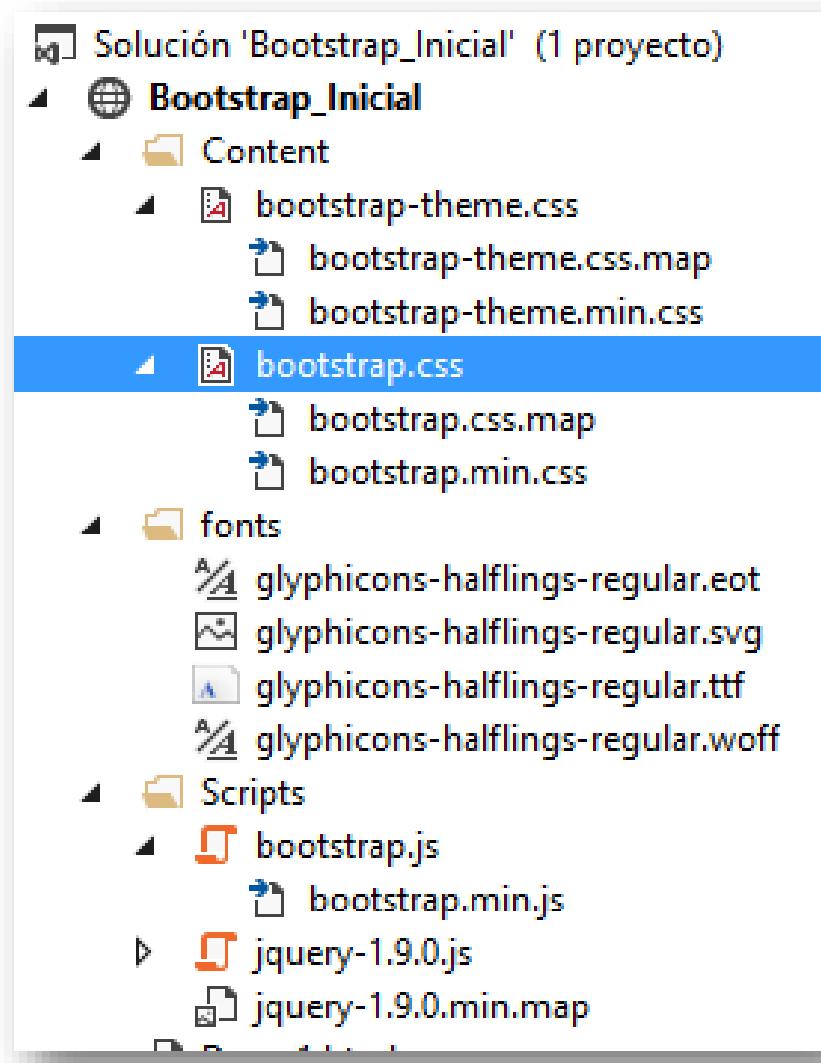
- ¿Qué es BootStrap?
  - Un conjunto de herramientas para simplificar el desarrollo Web
- ¿Quién está detrás de las librerías (soporte)?
  - Autores: Mark Otto y Jacob Thornton (Twitter)
  - Ahora cuenta con un equipo más largo de desarrollo (Nick Gallagher, entre otros)
- Versiones actuales
  - Versión 3.3.2 estable (Febrero/2015)
- Sitios de distribución de contenido (CDN)
  - <http://maxcdn.bootstrapcdn.com/bootstrap/3.2.0/css/bootstrap.min.css>
  - <http://maxcdn.bootstrapcdn.com/bootstrap/3.2.0/js/bootstrap.min.js>
  - <http://getbootstrap.com/getting-started/>
  - <http://cdnjs.com/libraries/twitter-bootstrap> (desglosado)

# Implantaciones y configuración

- Podemos ver algunos modelos de aplicaciones/sitios Web implantados en la página <http://wrapbootstrap.com>
- Y la documentación oficial se encuentra en:  
<http://getbootstrap.com/getting-started/>
- Y hay una versión en español:  
<http://www.oneskyapp.com/es/docs/bootstrap>
- Y se dispone de un conjunto de demos en:  
<http://getbootstrap.com/components>
- Y existe un editor “online” que genera código para Bootstrap:  
<http://twitter.github.io/typeahead.js/>
- Si queremos utilizar temas predeterminados, podemos encontrar varios temas "extra" en la dirección: <http://bootswatch.com/>

# Configuración predeterminada

- El resultado de la instalación básica produce (en V. Studio) una estructura de contenidos de Bootstrap como la que vemos en la figura adjunta:



# Plantilla inicial recomendada

- La propia documentación de Bootstrap recomienda una plantilla inicial similar a la siguiente para evitar algunos problemas con navegadores antiguos y algún que otro tipo de incompatibilidad
- Las librerías HTML5 Shim y Respond.js suministran soporte para los elementos HTML5 y de “media queries” de CSS3 en IE8
- Téngase en cuenta que Bootstrap es “responsive” de manera predeterminada, y que adopta la filosofía “mobile-first”
- De hecho, si no se desea esta funcionalidad, debe ser deshabilitada “ex-profeso”
- Tener en cuenta que muchas aplicaciones no utilizan los “plugins”, de forma que esto no sería necesario más que en esos casos.
- Por supuesto esto es solo el punto de partida. A partir de aquí configuraremos todo lo que necesite la aplicación y modificaremos Bootstrap a nuestro gusto.

# Plantilla inicial recomendada

- › <http://getbootstrap.com/getting-started/#template>

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- The above 3 meta tags *must* come first in the head; any other head content must come *after* these tags -->
    <title>Bootstrap 101 Template</title>

    <!-- Bootstrap -->
    <link href="css/bootstrap.min.css" rel="stylesheet">

    <!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
    <!--[if lt IE 9]>
      <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
      <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
    <![endif]-->
  </head>
  <body>
    <h1>Hello, world!</h1>

    <!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
    <!-- Include all compiled plugins (below), or include individual files as needed -->
    <script src="js/bootstrap.min.js"></script>
  </body>
</html>
```

# Guía de migración de la versión 2 => 3

- La versión 3 es la más ampliamente utilizada en la actualidad, e incluye importantes mejoras y características respecto a la anterior.
- En el caso de que se tenga que realizar un mantenimiento que implique actualización o directamente una migración completa para el soporte de la versión 3, la documentación oficial de migración se encuentra en la dirección:  
<http://www.oneskyapp.com/es/docs/bootstrap/migration/>
- Hay que tener presente que existen muchas características retiradas de la versión anterior (ver documento anterior).

# Los fundamentos

- Bootstrap asume desde el comienzo (v.3) la filosofía “mobile-first”
- En vez de agregar estilos opcionales de móviles, están incorporados en la parte nuclear. Los estilos móviles pueden verse en toda la librería en vez de incluirse en archivos separados.
- Debemos recordar que para garantizar la generación de gráficos correcta y un funcionamiento correcto de zoom en pantallas “touch”, agregamos la etiqueta meta:

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

- Se puede desactivar las opciones de zoom en dispositivos móviles al agregar **user-scalable=no** a esa etiqueta.
- Esto desactiva zoom, lo que significa que los usuarios solo pueden desplazarse:

```
<meta name="viewport" content="width=device-width, initial-scale=1,  
maximum-scale=1, user-scalable=no">
```

# La Tipografía y los gráficos básicos

- Bootstrap establece la pantalla global básica, tipografía y estilos de enlace iniciales. Específicamente, hace lo siguiente:
  - Configura el background-color: #fff; (blanco) en el body.
  - Utilizar los atributos @font-family-base, @font-size-base y @line-height-base como base tipográfica.
  - Configura el color global de enlaces mediante @link-color y aplica los subrayados de enlace solamente en :hover. (\* -> estados)
- Estos estilos se encuentran en scaffolding.less, si descargamos el proyecto de Github.
- Los gráficos “generados” pueden ofrecer problemas en algunos navegadores, por lo que se recomienda el uso de [Normalize.css](#), para esos casos (un proyecto desarrollado por [Nicolas Gallagher](#) y [Jonathan Neal](#))

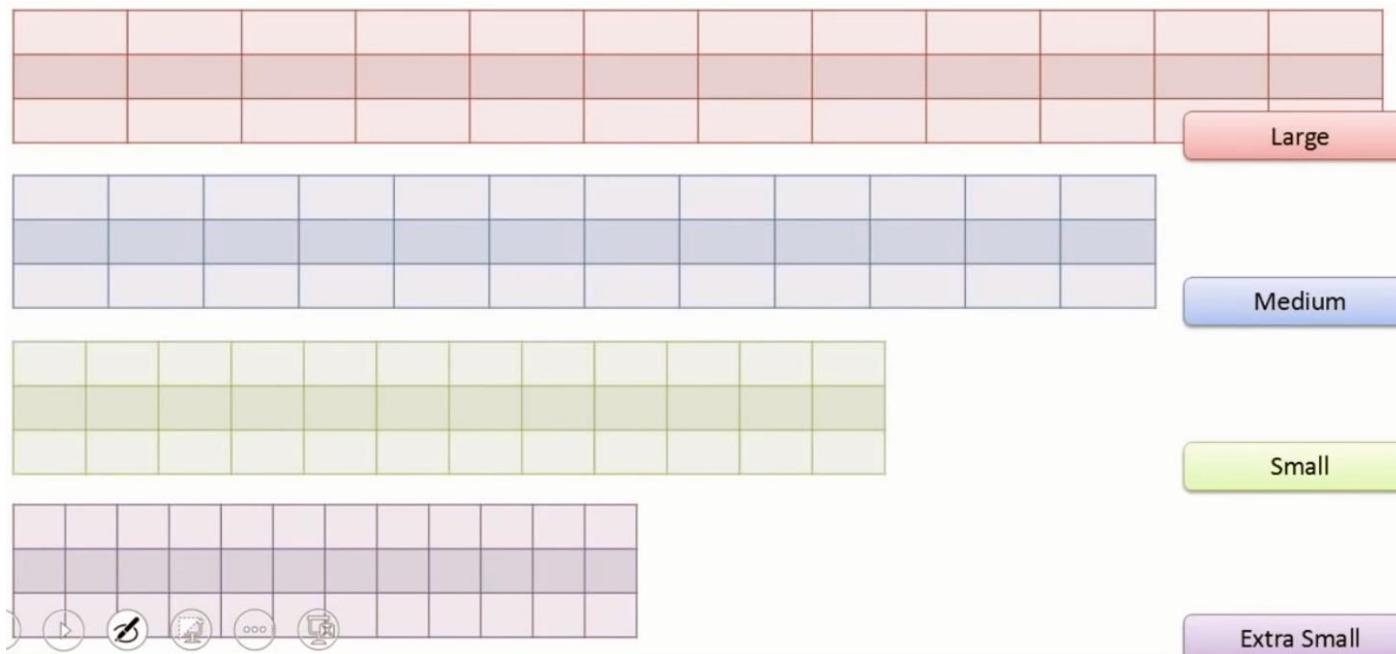
2

## Contenedores y sistemas de cuadrícula

# Los contenedores (“Layout”)

- Bootstrap define una estructura visual de contenedores tipo cuadrícula (Grid), permitiendo envolver sus contenidos y organizarlos de forma sencilla (Eso no significa que vaya a usar tablas)
- El contenedor puede ser cualquier elemento que habitualmente se utilice como “wrapper” (envoltorio) de otros elementos.
- Básicamente, existen dos tipos, aunque ninguno es “anidable” (por temas de “padding”)
  - Podemos utilizar la clase .container, si queremos un contenedor “responsive”, pero de ancho fijo.
  - O podemos utilizar .container-fluid, para un contenedor “responsive”, de anchura completa, que ocupe todo el espacio disponible en el “viewport”.

# Los contenedores (“Layout”)



- Bootstrap define una cuadrícula (Grid), con 4 modos de visualización
- Cada modo es un tamaño predeterminado (que depende del "Viewport") y se divide en 12 columnas.
- De esa forma, el usuario programa un solo modo lógico, pero el contexto se adapta a cualquiera de los cuatro (en modo físico).

# Breakpoints bootstrap

Name	Symbol	Viewport range	Example
Extra small	xs	<768px	Mobile devices
Small	sm	>=768px and <992px	Smart Phones and tablet devices
Medium	md	>=992px and <1200px	Laptops
Large	lg	>=1200px	Large desktop devices

- Dispositivos con una anchura <768px caen en la categoría de dispositivos extra pequeños como dispositivos móviles. Bootstrap proporciona "xs" para representar estos dispositivos.
- Dispositivos con una anchura > = 768px y <992px caen en la categoría de dispositivos pequeños; como los dispositivos de tableta. Bootstrap proporciona "sm" para representar estos dispositivos.
- Dispositivos que tienen una anchura > = 992px y <1200px caen en la categoría de dispositivos medianos; como dispositivos portátiles. Bootstrap proporciona "md" para representar estos dispositivos.
- Dispositivos que tienen una anchura > = 1200px caen en la categoría de dispositivos grandes; como dispositivos de escritorio grandes. Bootstrap proporciona el símbolo "lg" para representar estos dispositivos.

# Estructura de un layout Bootstrap

.container or .container-fluid

.row

.col-\* -12

.col-\* -6

.col-\* -6

.col-\* -4

.col-\* -4

.col-\* -4

.col-\* -6

.col-\* -3

.col-\* -3

.col-\* -2

.col-\* -2

.col-\* -2

.col-\* -2

.col-\* -2

.col-\* -2

.col-\* -1

# Ejemplo

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Bootstrap 101 Template</title>
    <link href="css/bootstrap.min.css" rel="stylesheet">
  </head>
  <body>
    <header class="container">
      <h2>Header</h2>
    </header>
    <section class="container" >
      <div class="row">
        <div class="col-md-3 leftNav">Left Navigation</div>
        <div class="col-md-6 content">Content Area</div>
        <div class="col-md-3 relatedContent">Related content</div>
      </div>
    </section>
    <footer class="container">
      <h2>Footer</h2>
    </footer>

    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
    <script src="js/bootstrap.min.js"></script>
  </body>
</html>
```

# Sistema de cuadrícula

- La cuadrícula es de tipo “mobile-first” y fluida, escalable hasta 12 columnas a medida que el “viewport” crece.
- Incluye clases predefinidas para facilitar la disposición visual de los contenidos, así como “mixins” para generar más contenidos semánticos.

El funcionamiento, podemos resumirlo en las siguientes **reglas**:

- Se deben colocar filas dentro de un `.container` (ancho fijo) o `.container-fluid` (ancho completo) para una alineación y relleno adecuados.
- Utilizar filas para crear grupos horizontales de columnas.
- Se debe colocar el contenido entre columnas y solamente éstas pueden ser los hijos inmediatos de las filas. (`row > col`)
- Las clases predefinidas de cuadrícula tales como `.row` y `.col-xs-4` están disponibles para crear distribuciones de cuadrícula rápidamente.

## Sistema de cuadrícula (cont.)

- Las columnas crean medianiles (espacios entre el contenido de columnas) mediante padding.
- Ese relleno se compensa en filas para la primera y última columna mediante un margen negativo en .row.
- El margen negativo es la causa por la que muchos ejemplos predeterminados aparecen indentados incorrectamente. Es así de manera que el contenido entre las columnas de la cuadrícula se alinee con el contenido externo a ella.
- Se crean las columnas de cuadrícula al especificar la cantidad de columnas que se desee generar.
- Por ejemplo, tres columnas iguales para el modo "Extra-Small" se definirían mediante .col-xs-4.
  - Técnicamente eso significa (columns-extra small-4).
  - Lo que traducido viene a ser:

**Columnas - Tamaño máximo – Espacio Disponible/4**

## Sistema de cuadrícula (cont.)

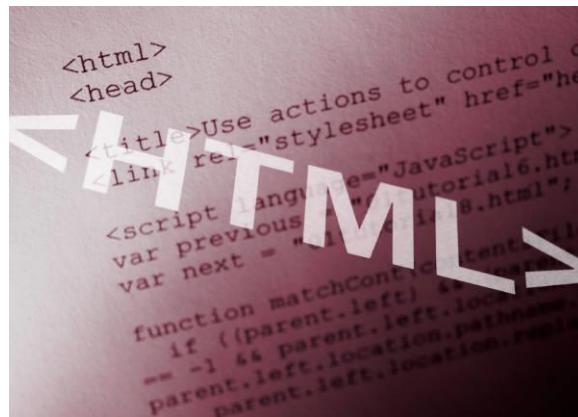
- Si se colocan más de 12 columnas en una sola fila, cada grupo extra de columnas se desplazará –como un solo grupo- a una nueva línea.
- Las clases de cuadrícula se aplican a dispositivos con anchos de pantalla mayores o iguales a los del punto de ruptura y reemplazan a las clases de cuadrícula que se usan en los dispositivos más pequeños.
- Por tanto, al aplicar cualquier clase .col-md-\* a un elemento, no solo afectará a su estilo en dispositivos medianos, sino también en grandes, si no hay una clase .col-lg-\* presente.

# Opciones de cuadrícula

	Dispositivos muy pequeños Teléfonos (<768px)	Dispositivos pequeños Tabletas (≥768px)	Dispositivos medianos Escritorios (≥992px)	Dispositivos grandes Escritorios(≥1200px)
Comportamiento de la cuadrícula	Horizontal en todo momento	Comprimido al inicio, horizontal encima de los puntos de corte		
Ancho de contenedor	Ninguno (automático)	750px	970px	1170px
Prefijo de la clase	.col-xs-	.col-sm-	.col-md-	.col-lg-
# de columnas	12			
Ancho de columna	Automático	~62px	~81px	~97px
Ancho de separación	30px (15px en cada lado de columna)			
Permite anidado	Sí			
Desplazamientos	Sí			
Cambio de orden de columnas	Sí			

# DEMO

- Ver las demos
  - 000 -> 008



# Media queries definidas

- Las “media queries” se utilizan para definir los puntos de ruptura claves del sistema de cuadrículas.
- Como se utiliza “Less” en la definición interna, existe un conjunto de valores predeterminado basado en las medidas indicadas en la tabla anterior.

```
/* Dispositivos muy pequeños (tel., y menores de 768px) */  
/* No hay media query, ya que es el predeterminado de Bootstrap */  
  
/* Dispositivos pequeños (tabletas, 768px +) */  
@media (min-width: @screen-sm-min) { ... }  
  
/* Medios (Escritorios, 992px+) */  
@media (min-width: @screen-md-min) { ... }  
  
/* Grandes (Escritorios grandes, 1200px+) */  
@media (min-width: @screen-lg-min) { ... }
```

# Media queries definidas

- Ocasionalmente, ampliamos estas media queries para incluir max-width con el fin de limitarnos a un conjunto menor de dispositivos.
- Para ello, Bootstrap define otros valores límite en el extremo superior, del tipo **@screen-xx-max**

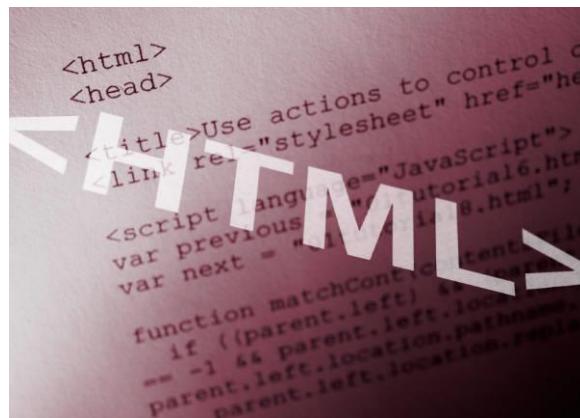
```
@media (max-width: @screen-xs-max) { ... }  
@media (min-width: @screen-sm-min) and (max-width: @screen-sm-max) {..}  
@media (min-width: @screen-md-min) and (max-width: @screen-md-max) {..}  
@media (min-width: @screen-lg-min) { ... }
```

# Contenedores y distribución visual (“Layout”)

- Apilado a horizontal
- Al utilizar un conjunto de clases de cuadrícula .col-md-\* , se puede crear un sistema básico de cuadrícula que inicialmente se apila en dispositivos móviles y tabletas (desde las más pequeñas a pequeñas), antes de volverse horizontales en dispositivos de escritorio (medianos).
- Los valores indicados en \* no pueden sobrepasar (en total) las 12 columnas establecidas por el esquema visual
- Las columnas de cuadrícula deben ir en cualquier .row.
- También disponemos de un conjunto de utilidades adicionales para casos especiales (clearfix, offset, pushes y pulls) que permiten gestionar desplazamientos forzados, anular configuraciones, etc.

# Opciones de cuadrícula

- DEMO de Contenedores Básicos.
  - Opciones de columna
  - Variantes para móviles/tabletas
  - Distribuciones irregulares
  - Columnas fuera de cuadrícula
  - Reconfigurar columnas cuando una tiene una altura distinta.



# Desplazamiento y anidación de columnas

- Se pueden mover las columnas hacia la derecha, usando clases
  - .col-md-offset-\*.
- Estas clases aumentan el margen izquierdo de una columna por el número de columnas indicadas (\*).
- Por ejemplo, .col-md-offset-4 mueve .col-md-4 en lo que corresponde a cuatro columnas.
  - Para anidar columnas, incluimos el contenido en la cuadrícula predeterminada, añadimos una nueva .row y configuramos las col-sm-\* dentro de una columna col-sm-\* existente.
  - Las filas anidadas deben incluir un conjunto de columnas no mayor que 12 (aunque no se requiere que lleguen a 12)

# Cambio de orden de las columnas

- Podemos modificar el orden de las columnas pre-configuradas con las clases (tipo modificador): .col-md-push-\* y .col-md-pull-\*.

```
<div class="row">
  <div class="col-md-9 col-md-push-3">.col-md-9 .col-md-push-3</div>
  <div class="col-md-3 col-md-pull-9">.col-md-3 .col-md-pull-9</div>
</div>
```

# 3

## Bootstrap y la interfaz de usuario: LESS, texto, Clases de transformación

# Bootstrap y la Interfaz de Usuario

- La librería define algunos estilos con los nombres de las etiquetas correspondientes para que resulte más fácil operar
- Coherencia desde el punto de vista operativo y visual
- De manera predeterminada, todas las etiquetas utilizan las definiciones base, por lo que solo con incluir la librería ya se aprecian cambios en la salida
- Además, se define toda la secuencia de estilos que corresponde a las etiquetas de cabecera (**h1...h6**), simplemente como (.h1 .... .h6), para que puedan usarse cuando se desee que coincida el estilo pero se despliegue en línea (sin salto)
- En cualquiera de estas opciones se puede crear un texto secundario (más ligero), con solo añadir una etiqueta <**small**>

```
<h1>h1. Bootstrap heading <small>Secondary text</small></h1>
```

# Bootstrap y la Interfaz de Usuario

- Esto genera la siguiente salida (utilizamos todas las etiquetas para ver el efecto)

**h1. en Bootstrap** Texto secundario

**h2. en Bootstrap** Texto secundario

**h3. en Bootstrap** Texto secundario

**h4. en Bootstrap** Texto secundario

**h5. en Bootstrap** Texto secundario

**h6. en Bootstrap** Texto secundario

# Bootstrap y texto

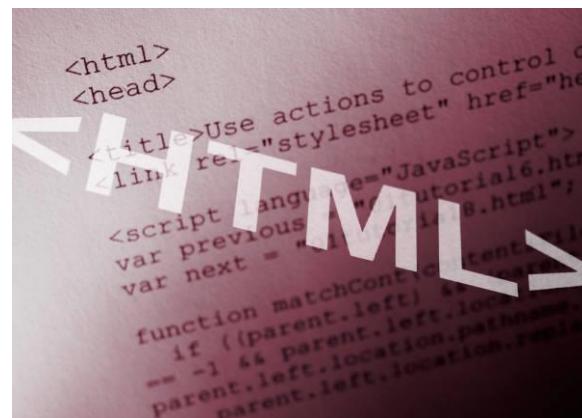
- El font-size global predeterminado de Bootstrap es 14px, con una altura de línea line-height de 1.428.
- Esto se aplica a <body> y a todos los párrafos.
- Además, los elementos <p> (párrafos) reciben un margen inferior de la mitad de su altura de línea calculada (10px de forma predeterminada)
- Recordemos que la escala de tipografía se basa en dos variables Less:
  - @font-size-base: el tamaño de fuente básico usado en todos lados
  - @line-height-base: la altura de línea básica.
- Bootstrap usa estas variables y alguna matemática sencilla para crear márgenes, rellenos y alturas de línea de todas las fuentes y mucho más.
- Si se personalizan, Bootstrap se adapta.

# Bootstrap y texto

- Las opciones fundamentales sobre el texto las soporta modificando la presentación de las etiquetas básicas habituales de tipo “in-line”, como **<b>**, **<i>**, **<s>**, **<u>**, **<del>**, etc.
- Para el uso de **negritas** y **cursivas**, se prefieren respectivamente, los elementos **<strong>** y **<em>**.
- Además, añade otras de “motu propio” para proporcionar opciones que ni siquiera están disponibles de la otra forma.
  - Esto es lo que sucede con estilos predefinidos como “**lead**”, que se utiliza para resaltar un párrafo.
  - La librería también incluye clases para el tratamiento de cómo se debe alinear un texto dentro (o fuera) de un contenedor.
  - Para ello, disponemos de un conjunto de clases: **text-left**, **text-right**, **text-center**, **text-justify** y **text nowrap**, similares a las que se encuentran disponibles en CSS tradicional, pero compatibles con todo el marco de trabajo adaptado de Bootstrap.

# Opciones básicas con el texto

- DEMO de opciones de texto
- (01, 02, 03)
  - Cabeceras
  - Etiquetas estándar
  - Clases propias para uso con texto



# Efectos avanzados sobre el texto

- Aparte de las funciones básicas que hemos visto, disponemos de todo un conjunto de efectos que facilitan la presentación de contenidos de texto.
- Entre los más interesantes, cabe destacar:
  - Clases de transformación
  - Abreviaturas
  - Direcciones
  - Bloques de citas
  - Listas
  - Código
  - Cuadros
  - Formularios
  - Áreas de texto
  - CheckBoxes y OptionButtons
  - Selección de opciones
  - Focos y desactivaciones
  - Estados de validación
  - Iconos opcionales

# Clases de transformación

Clases que modifican el contenido de texto, afectando el formato:

- Mayúsculas, Minúsculas y Tipo Oración. Clases:
  - `.text-uppercase`, `.text-lowercase` y `.capitalize`
- Abreviaturas:
  - Añaden subrayado suave a etiquetas `<abbr>`
  - Abreviaturas pequeñas: clase `.initialism` cambia la letra un 10% menor
- Direcciones:
  - formatea la salida de forma similar a una dirección personal
  - Usa la etiqueta `<address>`
  - Requiere que se incluyan saltos de línea (`<br/>`) después de cada línea lógica

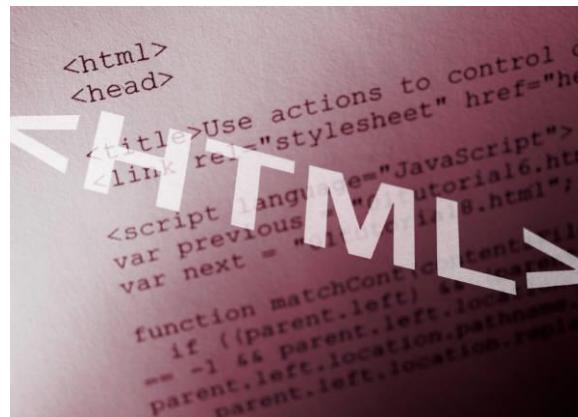
# Clases de transformación

## ➤ Citas y referencias

- Actúa sobre las etiquetas **<blockquote>**.
- Produce un efecto visual de separación
- Mediante la clase **.blockquote-reverse** la salida se alinea por la derecha
- Si se añade un elemento **<footer>**, puede usarse como “firma”

# Trabajo con textos

- # ➤ DEMO: Clases de transformación



4

# Clases, abreviaturas, direcciones

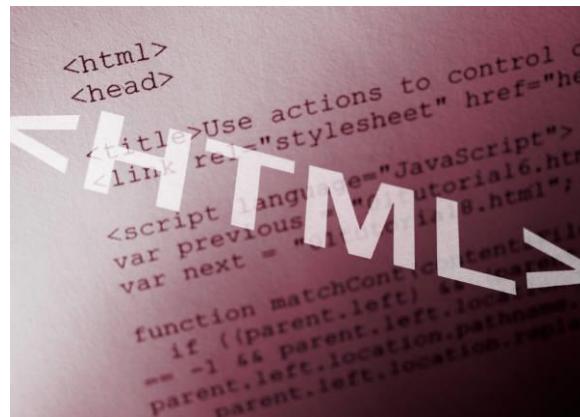
# Listas

- Las listas (ordenadas o no) también son objetivo de la librería
- Utiliza sus definiciones predeterminadas, pero también dispone de un conjunto de clases para ellas:
- Por ejemplo, se pueden anular los iconos predeterminados mediante **.list-unstyled**
- También podemos forzar la salida para que sea “in-line” (horizontal), usando **.list-inline** especialmente útil con elementos de menú.
- Igualmente, contamos con varias clases para agrupar los contenidos de las listas: **.list-group-\***

- **list-group-item**
- **list-group-item-danger**
- **list-group-item-heading**
- **list-group-item-info**
- **list-group-item-success**
- **list-group-item-text**
- **list-group-item-warning**
- **list-inline**
- **list-unstyled**

# Trabajo con textos

## ➤ DEMO: Listas



# Listas de definiciones

- Otros elementos manejados son las estructuras HTML para listas de definiciones
- Maneja estructuras del tipo

```
<dl>
  <dt>...</dt>
  <dd>...</dd>
</dl>
```

- La salida se formatea coherentemente con el conjunto de salidas de texto
- También podemos generar descripciones horizontales mediante .dl-horizontal

<b>Listas de descripción</b>	Una lista de descripción es perfecta para definir los términos.
<b>Euismod</b>	Vestibulum id ligula porta felis euismod semper eget lacinia odio sem nec elit. Donec id elit non mi porta gravida at eget metus.
<b>Malesuada porta</b>	Etiam porta sem malesuada magna mollis euismod.
<b>Felis euismod sempe...</b>	Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut sit amet risus.

# Código fuente y texto de usuario

- Si incluimos código fuente, debemos utilizar la etiqueta <code>
- Si el usuario introduce texto, y, especialmente, si se trata de pulsaciones especiales de teclado, debemos utilizar <kbd>
- Bootstrap formateará las pulsaciones de forma que resulte evidente la combinación de teclas a usar:
- Utilizar <pre> para varias líneas de código. Hay que desactivar cualquier corchete cuadrado del código, para lograr un despliegue correcto.

Por ejemplo, se debe usar la etiqueta <section> como "in-line"

Para cambiar de directorio, pulse **cd** seguido del nombre del directorio.

Para editar la configuración, pulse **ctrl + ,**

<p>Texto de código cualquiera...</p>

# Código fuente y texto de usuario

- Para el caso de las variables, podemos utilizar la etiqueta `<var>` para las declaraciones, lo que modifica la salida visual.

```
<!--Declaración de variables-->
<var>y</var> = <var>m</var><var>x</var> + <var>b</var>
```

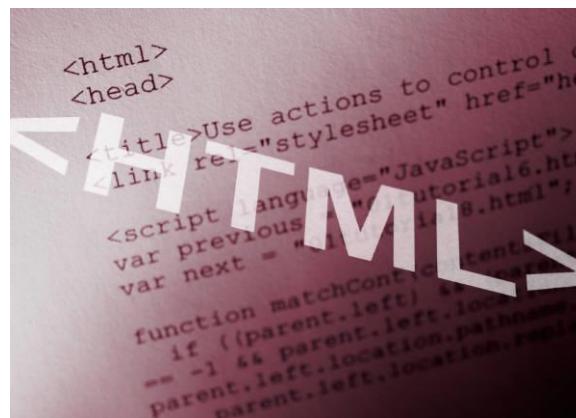
- Y para las salidas del código por pantalla (salida de la ejecución del código), debe utilizarse la etiqueta `<samp>`, lo que genera salidas similares a la siguiente:

```
y = mx + b
```

Este texto se trata como la salida correspondiente a al ejecución de un código

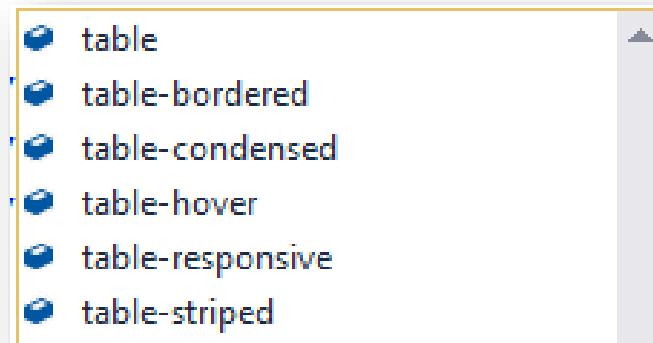
# Trabajo con textos

- › **DEMO:**
- › **Definiciones**
- › **Combinaciones de teclado**
- › **Manejo de código fuente**
- › **Variables y salidas de texto**



# Cuadrículas de texto

- Para estilos básicos o relleno ligero y solamente divisores horizontales, se puede agregar la clase base .table a cualquier elemento <table>.
- Puede parecer muy redundante, pero debido al uso generalizado de cuadros para otros **plug-in** como calendarios y selectores de fecha, decidieron aislar los estilos de cuadro personalizados.
- Existen clases variantes de **.table-\*** para modificar los cuadros de formas diversas:



# Cuadrículas de texto

- Por ejemplo, con la clase `.table-striped` se consiguen salidas de tablas tipo “papel-pijama” (con franjas):

Elemento1	Elemento2	Elemento3
Elemento1	Elemento2	Elemento3
Elemento1	Elemento2	Elemento3

- Nota:** *Los cuadros con franjas logran su estilo mediante el seleccionador de CSS `:nth-child`, que no está disponible en Internet Explorer 8*
- También podemos activar un cambio en el fondo de una fila, mediante la clase `.table-hover`, aplicada secuencialmente junto al resto de clases activas de la tabla

# Cuadrículas de texto

- Para modificar el aspecto de esta salida (y de otras similares), siempre es posible establecer nuestros propios estilos, que sobrescriban los de Bootstrap.
- Por ejemplo, si queremos que al pasar el cursor por encima de una fila cambie el color predeterminado, podemos incluir un estilo a nivel de página, con una regla similar a la siguiente:

```
.table tbody tr:hover td, .table tbody tr:hover th {  
    background-color: #4cff00;  
}
```

- Lo que generaría una salida como la que vemos a continuación

Elemento1	Elemento2	Elemento3
Elemento1	Elemento2	Elemento3
Elemento1	Elemento2	Elemento3



# Cuadrículas de texto : clases contextuales

- También podemos usar clases contextuales para hacer un tratamiento específico de las filas según las necesidades.
- Disponemos de

.active	Aplica el color de desplazamiento del ratón a una fila o celda específicas.
.success	Indica una acción exitosa o positiva.
.info	Indica un cambio o acción informativos neutrales.
.warning	Muestra una advertencia que puede hacer falta solucionar.
.danger	Indica un acción peligrosa o potencialmente negativa

- Cada uno de estos estilos formatea la propiedad **background-color**.



# Cuadrículas “responsive”

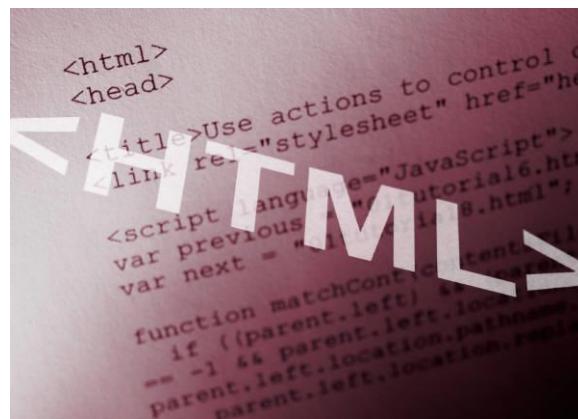
- Se pueden crear cuadrículas “*responsive*” insertándolas en cualquier elemento de clase **.table** en **.table-responsive** para hacer **scroll** horizontal en dispositivos pequeños (menores de 768px)
- Cuando se ven en dispositivos mayores, no existe diferencia.
- Podemos ver el efecto en la figura adjunta

#	Cabecera	Cabecera	Cabe	
1	Celda interior	Celda interior	Celd	
2	Celda interior	Celda interior	Celda inter	Celda int
3	Celda interior	Celda interior	Celda int	Celda interior

«Cuando se visualiza en dispositivos pequeños aparece la barra de scroll»

# Cuadrículas

- › DEMO:
- › Básicas
- › Formatos fundamentales
- › Personalizables
- › *Responsive*



5

# Menús, botones y formularios

# Añadiendo componentes a la página

- Los componentes se pueden obtener de  
<http://getbootstrap.com/components/>
- Proceso:
  - Identificar el componente a usar
  - Copiar el código de ejemplo que interese
  - Personalizar
- Herramientas:
  - Generador de componentes: <http://bootsnipp.com/>

# Formularios

- En primer lugar, todos los formularios se formatean parcialmente con los estilos globales. (Y soportan atributos de accesibilidad, como role)
- Los elementos textuales **<input>**, **<textarea>** y **<select>**, al aplicarles el estilo **.form-control** se configuran como width: 100%; de forma predeterminada.
- Para conseguir un espaciado óptimo, debemos envolver los rótulos y controles en un **.form-group**
- Se pueden conseguir efectos adicionales con otras clases tipo “helper”, como **.help-block** (aporta información complementaria para la introducción de datos)
- Las clases especializadas **.checkbox** y **.optionbutton** también aportan funcionalidad extra.
- Finalmente, las clases para los botones formatean su salida de forma coherente con todo el marco visual.

# Formularios

- La salida formateada puede tener el siguiente aspecto

**Email personal**

**Password**

**Seleccion de Fichero**

Examinar...

Ejemplo de texto a nivel de bloque.

Puedes activar este checkbox

# Formularios

- Podemos crear formularios alineados, mediante .form-inline para disponer de controles alineados a la izquierda y bloques alineados.
- Esto solo se aplica a los formularios con un ancho de salida de por lo menos 768px.
- Se requieren anchos personalizados.
  - Las áreas de entrada, selecciones y textos usan el 100% del ancho del formulario predeterminado en Bootstrap.
  - Para usar formularios alineados, hay que fijar un ancho en los controles utilizados internamente.
  - Se deben añadir siempre elemento <label>, o no se mostrará correctamente en lectores de pantalla (accesibilidad)
  - En los formularios alineados, se pueden ocultar esos rótulos si utilizamos la clase **.sr-only**
- En las demos que siguen vamos a ver el funcionamiento

# Formularios horizontales

- Al agregar **.form-horizontal** al formulario, se utilizan las clases predefinidas de cuadrícula de Bootstrap para alinear los rótulos y grupos de controles en una distribución horizontal.
- Al hacerlo, se cambia **.form-group** para que se comporte como filas de cuadrícula. En este caso, no se necesita **.row**.
- El aspecto que adoptan es el siguiente:

The image shows a screenshot of a web application's login form. The form is styled with Bootstrap's horizontal layout. It consists of two rows. The first row contains a label "Email" followed by an input field. The second row contains a label "Password" followed by another input field. Below these input fields is a checkbox labeled "Remember me". At the bottom of the form is a large, prominent "Sign in" button.

# Controles compatibles

- Los campos más comunes de digitación basados en texto para controlar formularios incluyen compatibilidad para todos los tipos de HTML5:
  - text, password, datetime, datetime-local, date, month, time, week, number, email, url, search, tel y color
- El estilo solo se verá correctamente si se declara el atributo type
- En los <textarea> podemos establecer las filas:

```
<textarea class="form-control" rows="3"></textarea>
```

- Las casillas de verificación y botones de radio con el atributo **disabled** se muestran adecuadamente.
- Para que los controles aparezcan en la misma línea podemos usar las clases **.checkbox-inline** o **.radio-inline**

# Controles compatibles

- › Los elementos **select** admiten el atributo **multiple** para seleccionar más de un ítem.
- › **Controles estáticos:** Cuando necesites colocar texto normal al lado de un rótulo en una formulario horizontal, utiliza la clase **.form-control-static** en un **<p>**

The screenshot shows a portion of a web form. On the left, there is a label 'Correo electrónico' followed by a static text element 'correo@example.com'. Below this, there is another label 'Contraseña' followed by a static text element 'Contraseña'. Both static text elements are enclosed in a light gray box with rounded corners, which is a common way to group related form controls.

- › Otras características (presentes en HTML5), permiten tomar el foco a un elemento, crear entradas de solo lectura, y desactivar campos de entrada.

# Validaciones de formularios

- › Bootstrap incluye estados de validación para estados de error, advertencia y éxito de los controles de un formulario.
- › Para usarlos, se agrega **.has-warning**, **.has-error** o **.has-success** al elemento padre.
- › Cualquier **.control-label**, **.form-control** y **.help-block** dentro de ese elemento recibirá los estilos de validación

The screenshot shows a form with three input fields and a set of radio buttons at the bottom:

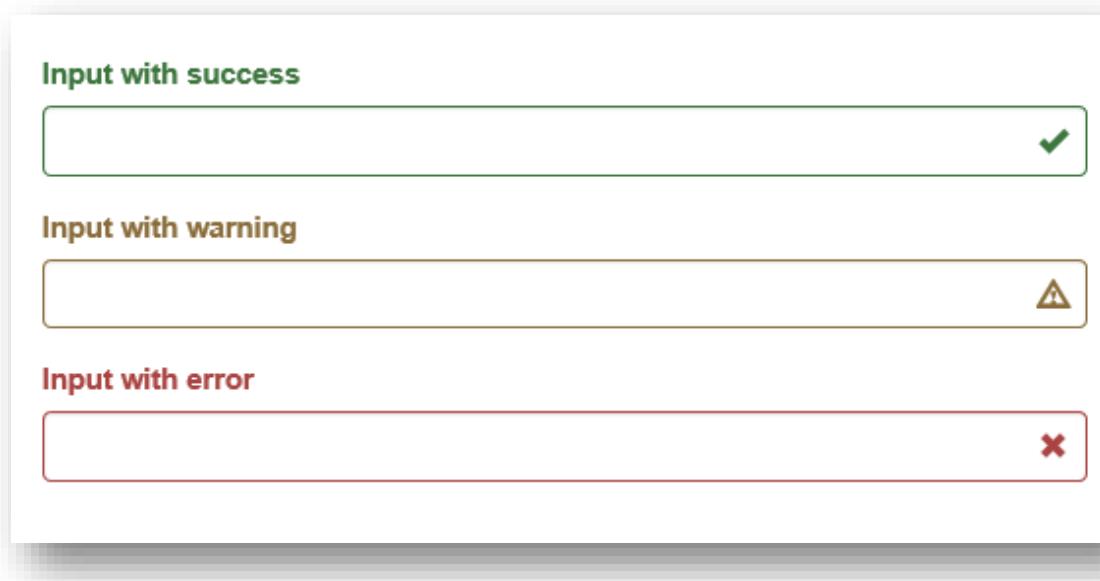
- Input with success**: The input field has a green border, indicating a successful validation state.
- Input with warning**: The input field has an orange border, indicating a warning validation state.
- Input with error**: The input field has a red border, indicating an error validation state.

Below the inputs is a group of radio buttons:

- Option one is this and that—be sure to include why it's great
- Option one is this and that—be sure to include why it's great
- Option two can be something else and selecting it will deselect option one

# Validaciones de formularios

- También pueden usarse iconos opcionales, añadiendo la clase *.has-feedback* y el ícono correspondiente.

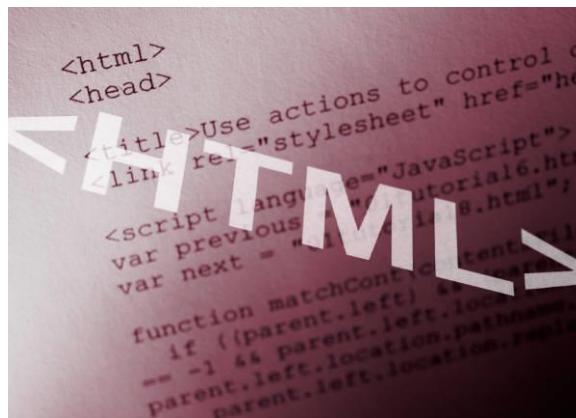


- También se pueden usar en formularios “in-line” y elementos con el atributo **.sr-only**.

# Formularios

## DEMO:

- Básicos
  - Alineados
  - Horizontales
  - Validaciones
  - Controles compatibles
  - Tamaños



# Efectos sobre otros controles

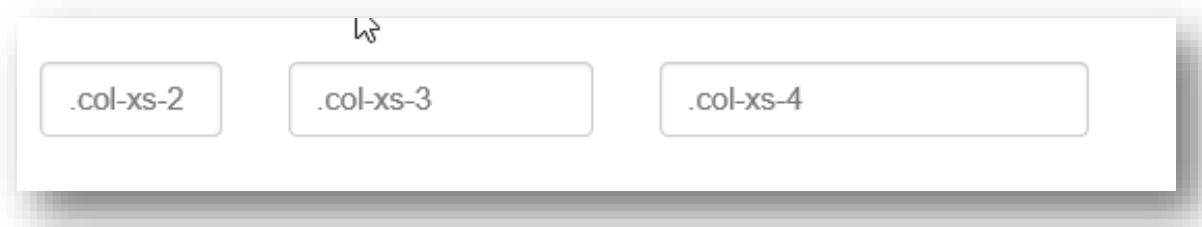
- El resto de controles de la interfaz de usuario también puede ser modificada fácilmente y muchas de las opciones más convenientes, ya las aporta Bootstrap de forma predeterminada
  - Tamaños de los controles
  - Agrupación de controles
  - Texto de ayuda
  - Botones
    - Gestión de estado y tamaño
  - Imágenes
    - Respuestas y formas (*shapes*)
    - De Ayuda
  - Utilidades
  - Clases de impresión
  - Tipografías
  - Componentes
  - Efectos visuales
  - Transiciones y animaciones
  - Columnas y gradientes.
  - Otras opciones

# Control de los tamaños de los controles

- Se pueden fijar las alturas con clases como .input-lg y los anchos con clases de columnas de cuadrícula. Por ejemplo, .col-lg-\*.

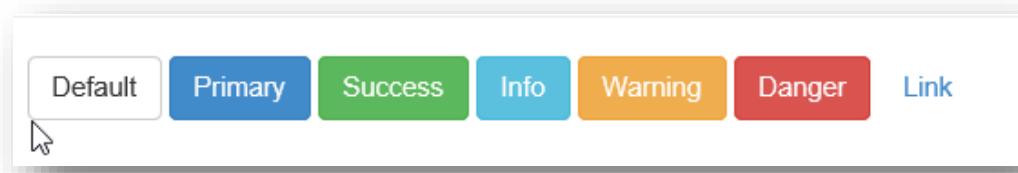


- También se pueden encapsular las entradas en columnas de cuadrícula, o cualquier elemento padre personalizado, para fácilmente imponer los anchos requeridos

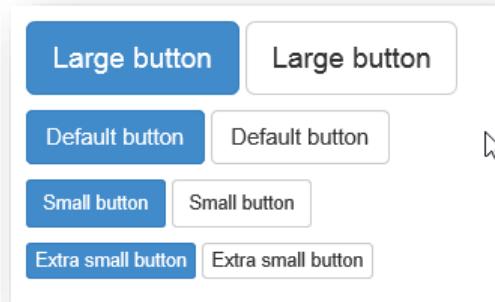


# Botones

- Se pueden usar un montón de clases disponibles para crear botones con estilo.
- Como se aplican en cascada, lo correcto es utilizar la clase base .btn y además, todas las otras clases de formato: **.btn-default**, **.btn-primary**, **.btn-success**, etc.

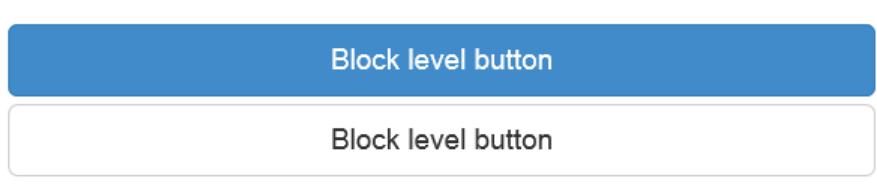


- Con ellos podemos simular enlaces y muchas otras “situaciones de IU”.
- Si se desea modificar el tamaño, pueden añadirse las **clases .btn-lg**, **.btn-sm** o **.btn-xs**



# Botones

- Se pueden crear botones a nivel de bloque, especialmente adaptados para móviles, mediante .btn-block.



- Cuando están activos, los botones aparecen como presionados (cambio de color de fondo, etc.)
- Si bien se pueden usar las clases de botón en elementos `<a>` y `<button>`, solamente los elementos `<button>` son compatibles con nuestros componentes de navegación y de barras de navegación.
- Si añadimos `.active (.btn-lg active)` cambia el comportamiento-
- También podemos añadirlo a los elementos `<a>`
- **Nota:** *si ponemos la opacidad de un botón en el 50% no se puede hacer click*

# Botones

## DEMO:

- Variantes, formatos y estados



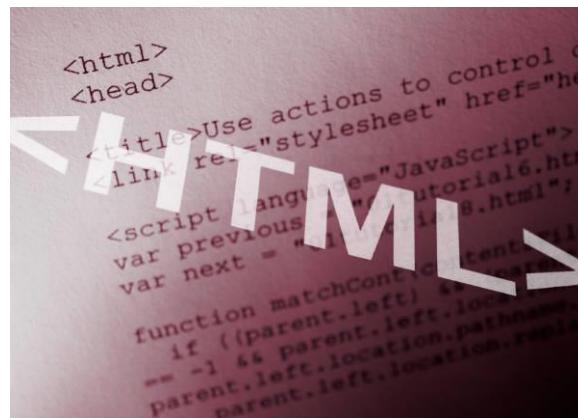
# Imágenes

- Podemos formatear las imágenes e incluir comportamiento, al tiempo que se permiten efectos especiales, cambios de estado, etc.
- Las imágenes de Bootstrap 3 pueden modificarse para que tenga reacciones, al agregar la clase **.img-responsive**.
- Esto aplica ***max-width: 100%;*** y ***height: auto;*** a la imagen para que se expanda o contraiga bien según el elemento padre.
- Si utilizamos imágenes SVG, hay que tener presente que en Internet Explorer 8-10, los SVG with .img-responsive quedan desproporcionados. Para arreglarlo, se añade ***width: 100% \9;*** donde haga falta.
- También podemos recortar las imágenes en ciertos formatos predeterminados con las clases: **.img-rounded**, **.img-circle** e **img.thumbnail**
- Si usamos **.thumbnail**, se pone un borde alrededor de la imagen

# Imágenes

## DEMO:

- Imágenes



# Utilidades

- Bootstrap incluye un buen conjunto de utilidades adicionales para diversas situaciones de desarrollo. Es difícil encuadrarlas en un apartado, por lo que las denominamos genéricamente utilidades.
- Colores y fondos contextuales.

```
<p class="bg-primary">...</p>
<p class="bg-success">...</p>
<p class="bg-info">...</p>
<p class="bg-warning">...</p>
<p class="bg-danger">...</p>
```
- Definen unos tonos de color para primer plano y fondo de los elementos con significados concretos por coherencia global de un sitio:

```
<p class="text-muted">...</p>
<p class="text-primary">...</p>
<p class="text-success">...</p>
<p class="text-info">...</p>
<p class="text-warning">...</p>
<p class="text-danger">...</p>
```

# Utilidades

- Elementos flotantes
  - Flotar un elemento a la derecha o a la izquierda es muy habitual en la mayoría de diseños web. Por eso Bootstrap 3 define dos clases CSS genéricas llamadas **.pull-left** y **.pull-right** que puedes aplicar sobre cualquier elemento
- Elementos centrados
  - Aplica la clase especial **.center-block** para centrar horizontalmente cualquier elemento (el elemento centrado se convierte en un elemento de bloque)
- Limpieza de ***floats***
  - Cuando un diseño utiliza muchos elementos flotantes, es común tener que limpiar un elemento para que no le afecten otros elementos flotantes. Bootstrap 3 utiliza para ello el famoso hack **.clearfix** creado originalmente por Nicolas Gallagher
- Ocultar y mostrar elementos
  - Otras de las utilidades incluidas por Bootstrap 3 son las clases **.show** y **.hide**, que muestran y ocultan cualquier elemento.

# Utilidades Responsive

- Se basan en el estándar CSS3 de "***Media Queries***" que se concluyó ya hace tiempo.
- Las utilidades ***responsive*** permiten diseñar más rápidamente sitios web móviles, ya que muestran u ocultan elementos en función del tipo de dispositivo que utiliza el usuario para navegar.
- También se incluyen clases para mostrar/ocultar elementos al imprimir la página.
- Deben utilizarse con moderación y siempre para mejorar el aspecto de los contenidos en cada tipo de dispositivo.

# Utilidades Responsive

- Además, sólo funcionan para los elementos de bloque y las tablas, por lo que no podrás aplicarlos a los elementos en línea.
- Utiliza alguna de estas clases o combina varias entre sí para definir cómo se ven tus contenidos en cada tipo de dispositivo:
  - teléfono = menos de 768px;
  - tabletas = más de 768px;
  - ordenador = más de 992px;
  - ordenador grande = más de 1200px;
- Hay otras combinaciones y variantes que dependen de la resolución a nivel de píxel, etc.

# Utilidades Responsive

- Las Tabla siguiente muestra las posibilidades

Clase	Teléfonos	Tablets	Ordenador	Ordenador grande
.visible-xs	Visible	Oculto	Oculto	Oculto
.visible-sm	Oculto	Visible	Oculto	Oculto
.visible-md	Oculto	Oculto	Visible	Oculto
.visible-lg	Oculto	Oculto	Oculto	Visible
.hidden-xs	Oculto	Visible	Visible	Visible
.hidden-sm	Visible	Oculto	Visible	Visible
.hidden-md	Visible	Visible	Oculto	Visible
.hidden-lg	Visible	Visible	Visible	Oculto

- Igualmente, puedes utilizar estas clases para definir qué contenidos se muestran al imprimir la página:

Clase	Navegador	Impresora
.visible-print	Oculto	Visible
.hidden-print	Visible	Oculto

# Iconos

- Bootstrap incluye 180 iconos creados mediante una fuente especial llamada **Glyphicon Halflings**.
  - Aunque esta fuente normalmente no es gratuita, su creador permite utilizar estos iconos gratuitamente dentro de Bootstrap 3.
- El listado completo de los iconos disponibles, puede verse analizando las fuentes incluidas

 .glyphicon .glyphicon-adjust	 .glyphicon .glyphicon-align-center	 .glyphicon .glyphicon-align-justify	 .glyphicon .glyphicon-align-left
 .glyphicon .glyphicon-align-right	 .glyphicon .glyphicon-arrow-down	 .glyphicon .glyphicon-arrow-left	 .glyphicon .glyphicon-arrow-right
 .glyphicon .glyphicon-arrow-up	 .glyphicon .glyphicon-asterisk	 .glyphicon .glyphicon-backward	 .glyphicon .glyphicon-ban-circle

# Menús desplegables

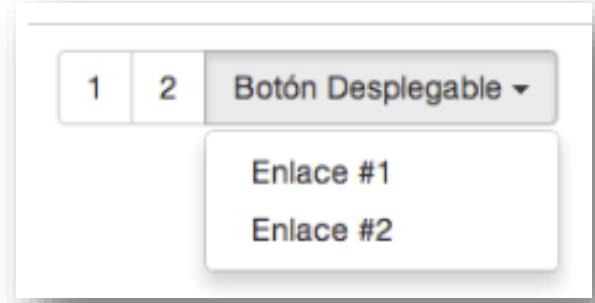
- Este componente permite mostrar una lista de enlaces como si fuera un menú desplegable con la lista de acciones que el usuario puede realizar
- Para que funcionen sus características interactivas, es necesario utilizar también el **plugin *dropdown*** de JavaScript
- El siguiente ejemplo de código muestra cómo implementarlo

# Menús desplegables

```
<ul class="dropdown-menu" role="menu" aria-labelledby="dropdownMenu1">
  <li role="presentation">
    <a role="menuitem" tabindex="-1" href="#">Acción</a>
  </li>
  <li role="presentation">
    <a role="menuitem" tabindex="-1" href="#">Otra acción</a>
  </li>
  <li role="presentation">
    <a role="menuitem" tabindex="-1" href="#">Otra acción más</a>
  </li>
  <li role="presentation" class="divider"></li>
  <li role="presentation">
    <a role="menuitem" tabindex="-1" href="#">Acción separada</a>
  </li>
</ul>
```

# Grupos de botones

- Para crear un grupo de botones, encierra varios botones con la clase **.btn** dentro de un elemento con la clase **.btn-group**.
- Combinando varios elementos **.btn-group** puedes crear una **toolbar** o barra de herramientas basada en botones
- Para cambiar el tamaño de los botones de un grupo, en vez de establecer el tamaño individualmente a cada botón, puedes aplicar las clases **.btn-group-\*** directamente al elemento que agrupa a todos ellos.
- Bootstrap 3 también permite mostrar menús desplegables dentro de los grupos de botones. Para ello, incluye un elemento con la clase **.btn-group** dentro de otro elemento con la clase **.btn-group**.



# Botones Desplegables

- Los botones desplegables se crean añadiendo un botón dentro de cualquier elemento con la clase **.btn-group**
- Botones desplegables compuestos
  - Los botones compuestos se crean realmente con dos botones diferentes: el primero contiene la acción principal y el segundo simplemente muestra una flecha descendente y al pulsarla, se despliega para mostrar el menú de opciones
- Los botones desplegables funcionan bien con cualquier tamaño de botón
- Los botones desplegables en Bootstrap 3 se despliegan hacia abajo ("dropdown" en inglés). Si quieres desplegar el menú hacia arriba ("dropup" en inglés), añade la clase **.dropup** al elemento contenedor del botón

# Elementos de navegación

- Aplica la clase **.nav** para crear un elemento de navegación y después aplica la clase **.nav-tabs** para mostrar sus enlaces en forma de pestaña
- Aplicando la clase **.nav-pills** al mismo código HTML del ejemplo anterior, los enlaces se muestran como botones ("pills" en inglés) en vez de como pestañas
- La navegación con botones también se puede mostrar verticalmente añadiendo la clase **.nav-stacked**.
- Bootstrap 3 también permite crear elementos de navegación que ocupan toda la anchura del elemento en el que se encuentran y en los que todos sus enlaces tienen la misma anchura.

# Elementos de navegación

- Para ello, añade la clase **.nav-justified** al elemento de navegación. Ten en cuenta que este comportamiento sólo es posible para los dispositivos con una anchura superior a **768px**. En el resto de dispositivos, el elemento de navegación se muestra verticalmente
- Usando el plugin dropdown de JavaScript y un poco de código HTML, es posible añadir menús desplegables tanto a las pestañas como a los botones

# Paginadores

- Bootstrap 3 incluye dos componentes para paginar los contenidos de tu sitio o aplicación web. Se basan en la clase **.pagination**
- El primero es un completo paginador que muestra enlaces para todas las páginas de resultados y el segundo es un paginador simple con los enlaces Anterior y Siguiente. Se pueden combinar con las clases **.pagination-lg** o **.pagination-sm** para controlar el tamaño
- El paginador por defecto de Bootstrap 3 está basado en el del sitio Rdio y queda bien tanto para las aplicaciones como para los sitios web. Como el paginador es bastante grande, el usuario nunca lo pierde de vista.
- Además su diseño escala bien a un gran número de páginas y sus enlaces son fáciles de usar.



# Paginadores

- Puedes usar un paginador simple, mediante la clase .pager y si lo prefieres, también puedes alinear los enlaces a la izquierda y a la derecha con las clases CSS especiales .previous y .next.



# Etiquetas

- Marcan un bloque de texto (habitualmente un elemento span) de manera visual muy resaltada para indicar situaciones o informaciones especiales. Basadas en la clase **.label** y sus variantes: **.label-primary**, **.label-success**, etc.
- Pueden cambiarse de tamaño a voluntad



- Añade cualquiera de las siguientes clases CSS específicas para modificar el tipo de etiqueta que se muestra, derivadas de las clases de resalte

Por defecto

Principal

Éxito

Información

Aviso

Peligro

# Mensajes de alerta

- Marcan un bloque de texto dentro de un marco con fondo para mostrar un mensaje y el color de fondo da una indicación del tipo de mensaje.
- Para mostrar un mensaje de alerta, se encierra dentro de un elemento con la clase **.alert** tanto el texto del mensaje como el botón opcional para ocultar la alerta.
- Además de esta clase base, también debes aplicar cualquiera de las otras cuatro clases **.alert-\*** para indicar explícitamente el tipo de mensaje (advertencia, error, éxito, información)
- Los mensajes de alerta no definen un estilo por defecto. Por eso es obligatorio indicar siempre tanto la clase base **.alert** como uno de sus modificadores (**.alert-success**, etc.)

# Mensajes de alerta

```
<div class="alert alert-success">...</div>
<div class="alert alert-info">...</div>
<div class="alert alert-warning">...</div>
<div class="alert alert-danger">...</div>
```

## Mensajes de alerta

Tincidunt integer eu augue augue nunc elit dolor, luctus placerat scelerisque euismod, iaculis eu lacinia nunc mi elit, vehicula ut laoreet ac, aliquam sit amet justo nunc tempor, metus vel.

Tincidunt integer eu augue augue nunc elit dolor, luctus placerat scelerisque euismod, iaculis eu lacinia nunc mi elit, vehicula ut laoreet ac, aliquam sit amet justo nunc tempor, metus vel.

Tincidunt integer eu augue augue nunc elit dolor, luctus placerat scelerisque euismod, iaculis eu lacinia nunc mi elit, vehicula ut laoreet ac, aliquam sit amet justo nunc tempor, metus vel.

Tincidunt integer eu augue augue nunc elit dolor, luctus placerat scelerisque euismod, iaculis eu lacinia nunc mi elit, vehicula ut laoreet ac, aliquam sit amet justo nunc tempor, metus vel.

# Mensajes de alerta

- Opcionalmente los mensajes de alerta pueden mostrar un botón de cierre para ocultar la alerta. Para que funcione este comportamiento, es necesario incluir el plugin alerts de JavaScript.
- Al hacerlo, se incluye un aspa que permite cerrar la alerta

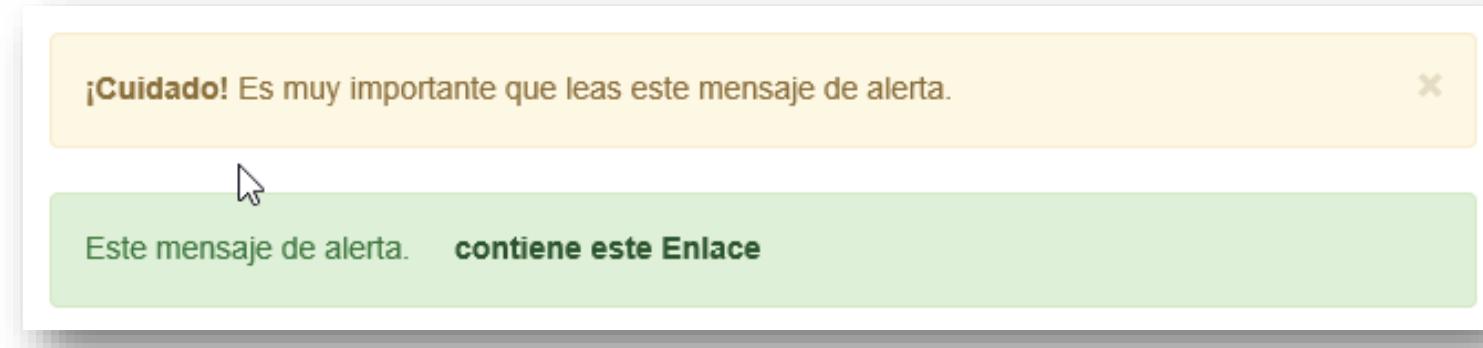
```
<div class="alert alert-warning alert-dismissible">
  <button type="button" class="close" data-dismiss="alert"> &times; </button>
  <strong>¡Cuidado!</strong> Es muy importante que leas este mensaje de alerta.
</div>
```

- El atributo **data-dismiss="alert"** garantiza el comportamiento correcto en todos los dispositivos.

# Mensajes de alerta

- Si el mensaje de la alerta contiene enlaces, resulta conveniente aplicarles la clase .alert-link para que su aspecto se adapte al del resto del mensaje.

```
<div class="alert alert-success">  
  <a href="#" class="alert-link">...</a>  
</div>
```



# 6

## Jumbotron, objetos multimedia y plugins JavaScript

# Jumbotron

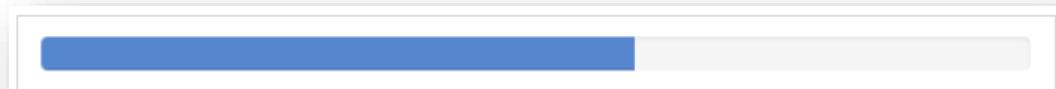
- Se trata de un componente para mostrar contenidos de forma muy destacada. Se basa en la clase .jumbotron
- Si no lo encierras dentro de un elemento .container, ocupa toda la anchura del dispositivo. Si lo encierras dentro de un .container, solamente ocupará la anchura del contenedor y mostrará esquinas redondeadas



# Barras de progreso

- Las barras de progreso permiten mostrar de forma continua el estado de ejecución de una tarea
- Utilizan las transiciones y animaciones de CSS3 para algunos de sus efectos. Estas características no están disponibles en Internet Explorer 9 o anterior ni tampoco en las versiones antiguas de Firefox

```
<div class="progress">
  <div class="progress-bar" role="progressbar" aria-valuenow="60"
      aria-valuemin="0" aria-valuemax="100" style="width: 60%;">
    <span class="sr-only">60% completado</span>
  </div>
</div>
```



# Objetos multimedia

- Bootstrap 3 define varios estilos genéricos para crear diferentes tipos de componentes formados por imágenes y texto, como por ejemplo los comentarios de un blog, los tweets, etc.
- El estilo por defecto para estos objetos alinea el elemento multimedia (imagen, vídeo , audio) a la derecha o a la izquierda del texto mediante la propiedad float

```
<div class="media">
  <a class="pull-left" href="#">
    
  </a>
  <div class="media-body">
    <h4 class="media-heading">Título del contenido</h4>
    ...
  </div>
</div>
```

# Objetos multimedia

64x64

## Título del contenido

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus sce  
purus odio, vestibulum in vulputate at, tempus viverra turpis. F  
vulputate fringilla. Donec lacinia congue felis in faucibus.

64x64

## Título del contenido

Cras sit amet nibh libero, in gravida nulla. Nulla vel metus sce  
purus odio, vestibulum in vulputate at, tempus viverra turpis. F  
vulputate fringilla. Donec lacinia congue felis in faucibus.

64x64

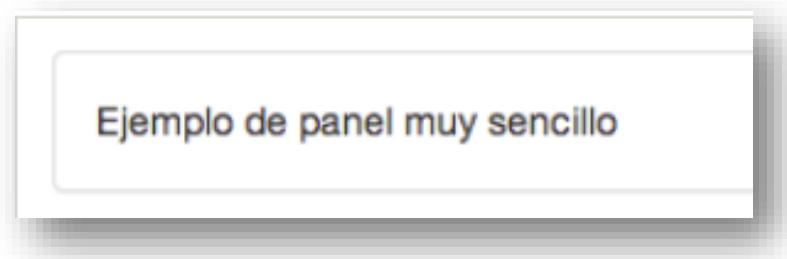
## Título del contenido anidado

Cras sit amet nibh libero, in gravida nulla. Nulla ve  
commodo. Cras purus odio, vestibulum in vulputat  
condimentum nunc ac nisi vulputate fringilla. Done

# Paneles

- En ocasiones es necesario encerrar cierto contenido dentro de una caja. En estos casos deberías considerar el uso del componente para paneles.
- Por defecto, al aplicar la clase .panel a un elemento solamente se añade un borde muy sencillo y algo de relleno con la propiedad padding

```
<div class="panel panel-default">  
  <div class="panel-body">  
    Ejemplo de panel muy sencillo  
  </div>  
</div>
```



Ejemplo de panel muy sencillo

# Paneles con título o pie

- Cuando lo necesites, añade un título al panel aplicando al texto del título la clase `.panel-heading`. Opcionalmente puedes utilizar cualquiera de las etiquetas de título (`<h1>`, ..., `<h6>`) para aplicar también esos estilos. Para los pies, utiliza la clase `.panel-footer`

```
<div class="panel panel-default">
  <div class="panel-heading">
    <h3 class="panel-title">Título del panel con estilo h3</h3>
  </div>
  <div class="panel-body">
    Contenido del panel
  </div>
</div>
```

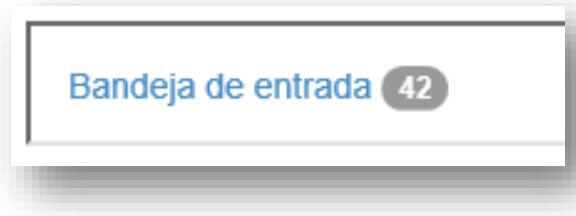
Título del panel con estilo h3

Contenido del panel

# Badges

- Los badges se emplean para destacar elementos nuevos o que no han sido leídos.
- Añade badges a los enlaces, los elementos de navegación de Bootstrap y muchos otros tipos de elementos mediante la etiqueta **<span class="badge">**.

```
<a href="#">Bandeja de entrada<span class="badge"> 42</span></a>
```



- Se ocultan automáticamente cuando no hay elementos nuevos o sin leer, gracias al selector :empty de CSS (#IE8)

# Encabezado de página

- Resulta muy común que las páginas de sitios web como blogs, periódicos y revistas muestren de forma muy destacada el titular de la página con alguna otra información relacionada.
- Por ello Bootstrap 3 define un componente llamado "encabezado de página" que puedes utilizar para encerrar este tipo de elementos.
- Normalmente, el contenido de este elemento, al que se aplica la clase .page-header, está formado por un elemento `<h1>` y un elemento `<small>` asociado para la información secundaria, aunque puede contener cualquier otro tipo de elemento

```
<div class="page-header">
  <h1>Encabezado de página <small>con un texto secundario</small></h1>
</div>
<p>...</p>
<p>...</p>
```

# Encabezado de página

## Encabezado de página con un texto secundario

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

# Pozos (Wells)

- El estilo por defecto de este componente simplemente aplica un efecto de tipo hundido (en inglés, "inset") al elemento
- Se aplica mediante la clase **.well**
- Permite las típicas variantes de tamaño: **.well-sm-\* , .well-lg-\***
- Los pozos definen dos clases modificadoras para añadir esquinas redondeadas y para controlar el espacio de relleno de forma predeterminada.

The image displays two instances of the 'well' component. The top instance is a large box with rounded corners and a light gray background. It contains a cursor icon and a block of Latin placeholder text. The bottom instance is a smaller box with rounded corners and a white background, also containing a block of Latin placeholder text.

Tincidunt integer eu augue augue nunc elit dolor, luctus placerat scelerisque euismod, iaculis eu lacus nunc mi elit, vehicula ut laoreet ac, aliquam sit amet justo nunc tempor, metus vel. Placerat suscipit, orci nisl iaculis eros, a tincidunt nisi odio eget lorem nulla condimentum tempor mattis ut vitae feugiat augue cras ut metus a risus iaculis scelerisque eu ac ante.

Tincidunt integer eu augue augue nunc elit dolor, luctus placerat scelerisque euismod, iaculis eu lacus nunc mi elit, vehicula ut laoreet ac, aliquam sit amet justo nunc tempor, metus vel. Placerat suscipit, orci nisl iaculis eros, a tincidunt nisi odio eget lorem nulla condimentum tempor mattis ut vitae feugiat augue cras ut metus a risus iaculis scelerisque eu ac ante.

# Los “plugin” de JavaScript

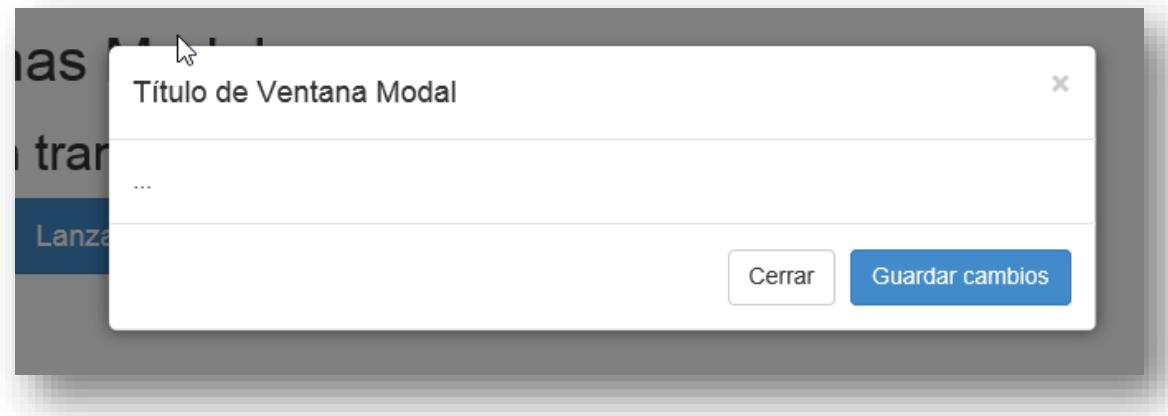
- Son componentes de Bootstrap, que según la documentación oficial están “más llenos de vida con una docena de plug-in personalizados de jQuery. Incluya todos fácilmente, o uno por uno”
- Se pueden incluir los plug-in individualmente (usando archivos \*.js individuales de Bootstrap) o todos a la vez (usando bootstrap.js o el bootstrap.min.js minimizado).
- Se pueden usar todos los plug-in de Bootstrap simplemente por medio de API marcado sin escribir una sola línea de código de JavaScript.
  - Este es el API de primera clase de Bootstrap y debería ser tu primera consideración cuando utilices un plug-in.
  - Están en constante crecimiento, y –de hecho- pueden estudiarse aparte, ya que dependen de aportaciones de terceros
  - A pesar de que los básicos se usan desde HTML, algunos de los avanzados requieren de alguna programación (los relacionados con eventos, etc.)

# Transiciones y animaciones

- Se basan en las nuevas capacidades de CSS3, por lo que se requiere un soporte base por el navegador.
- Para efectos sencillos de transición, incluya transition.js junto a otros archivos de JS. Si está usando el bootstrap.js compilado (o minimizado), no hace falta agregarlo. Ya se ha incorporado
- No obstante, es posible utilizar las transiciones y animaciones directamente desde CSS3 sin temor a que “colisionen” con el marco de trabajo de Bootstrap

# Ventanas modales

- Podemos hacer ventanas modales muy fácilmente, añadiendo las clases extra **.modal**, **.modal-dialog**, **.modal-content**, así como otras derivadas que permiten formatos concretos de ventana y contenido.
- También podemos controlar el tamaño de las ventanas con las clases derivadas **.modal-sm** y **.modal-lg**



# Tabuladores

- Podemos hacer ventanas modales muy fácilmente, añadiendo las clases extra **.modal**, **.modal-dialog**, **.modal-content**, así como otras derivadas que permiten formatos concretos de ventana y contenido.
- También podemos controlar el tamaño de las ventanas con las clases derivadas **.modal-sm** y **.modal-lg**

[Inicio](#)   [Perfil](#)

[Mensajes](#)

[Otros](#)

Tincidunt integer eu augue augue nunc elit dolor, luctus placerat scelerisque euismod, iaculis eu lacus nunc mi elit, vehicula ut laoreet ac, aliquam sit amet justo nunc tempor, metus vel. Placerat suscipit, orci nisl iaculis eros, a tincidunt nisi odio eget lorem nulla condimentum tempor mattis ut vitae feugiat augue cras ut metus a risus iaculis scelerisque eu ac ante. Fusce non varius purus aenean nec magna felis fusce vestibulum velit mollis odio sollicitudin lacinia aliquam posuere, sapien elementum lobortis tincidunt, turpis dui ornare nisl, sollicitudin interdum turpis nunc eget.

# Etiquetas flotantes (Tooltips)

- Inspirado por el excelente plugin de jQuery .tipsy escrito por Marco Jason;
- Estos Tooltips son una versión actualizada, que no se basan en imágenes, usan CSS3 para animaciones y atributos data- para el almacenamiento local.
- Admiten la indicación de la dirección en la que deseamos que se presente la etiqueta: left, right, top y bottom si lo asociamos al atributo personalizado data-placement.
- Esto último puede dar algún problema (dependiendo del contexto)

```
<button type="button" class="btn btn-default" data-toggle="tooltip" data-placement="left"  
title="Tooltip por la izquierda">Tooltip por la izquierda</button>
```

# Carruseles

- El complemento "presentación de diapositivas" usa un plug-in genérico para mostrar componentes de forma cíclica definiendo elementos que se verán como un carrusel.
- Existe un conjunto completo de variantes para las distintas modalidades que ya son populares
- Esto incluye los controles asociados al carrusel, de forma que el usuario pueda controlar el desplazamiento

# Referencias

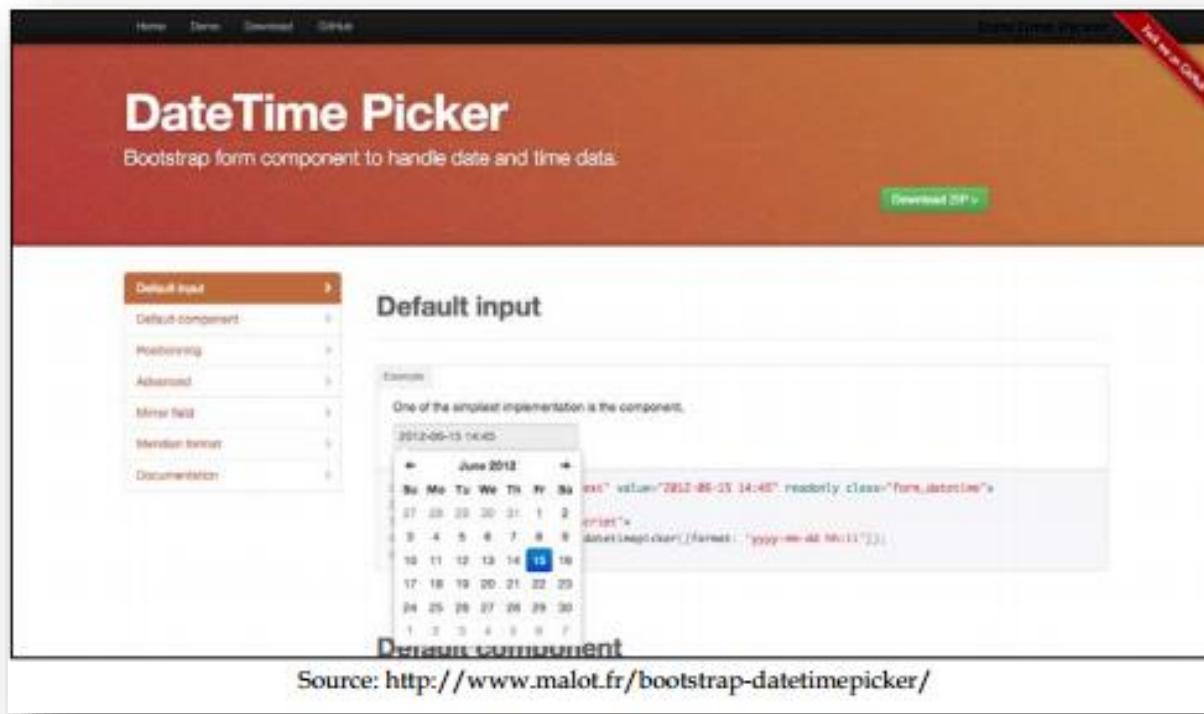
- Para evitar colisiones de nomenclatura de Bootstrap con otras librerías: <http://www.yelp.com>
  - Antecede con un prefijo –y- todas las clases de Bootstrap, evitando de esa forma una posible colisión.
  - El programador solo tiene que incluir la librería y programar como siempre.
- Librería para detección de modos de pantalla:
  - <https://github.com/maciej-gurban/responsive-bootstrap-toolkit>

# Complementos (Plug-ins)

- Existen un buen conjunto de complementos disponibles para Bootstrap, pero algunos de ellos destacan por su popularidad, y el nivel de implantación.
- Todos ellos, requieren de la presencia de jQuery (preferiblemente, en las versiones > 1.9
- También es posible reconfigurar el comportamiento de Bootstrap mediante la modificación de los ficheros LESS disponibles en el paquete estándar o en los temas vinculados con él.

# Complementos (Plug-ins)

- Bootbox.js (<http://bootboxjs.com/>)
  - Permite personalizar las cajas de diálogo modales
- DateTime Picker



# Complementos (Plug-ins)

- Bootstrap Modal: Permite manejar múltiples cajas de diálogo modales, mediante un sistema de escucha de sus eventos respectivos.

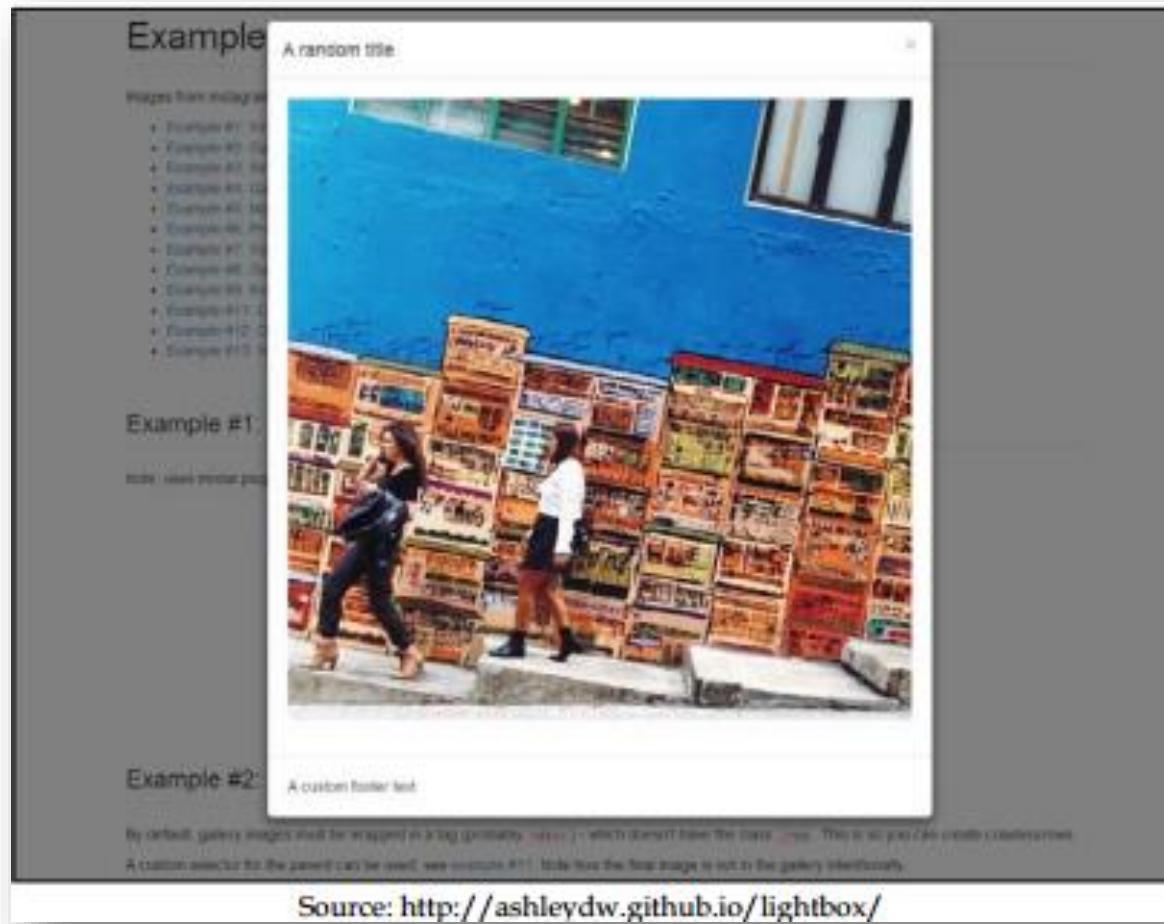
The screenshot shows a GitHub repository page for "Bootstrap-Modal". The main title is "Bootstrap-Modal" with the subtitle "Responsive, Stackable, AJAX and more.". Below the title, there are "Star" and "Fork" counts of 2,494 and 640 respectively. A modal dialog titled "Responsive" is displayed, containing two input fields labeled "Some Input" and "Some More Input". At the bottom of the modal are "Close" and "Save changes" buttons. On the left side of the page, there is a code snippet for creating a responsive modal:

```
1. <div id="responsive">
2.   <div class="modal">
3.     <button type="button" class="close" data-dismiss="modal">&amptimes
4.     <h3>Responsive</h3>
5.   </div>
6.   <div class="modal">
7.     <div class="row">
8.       <div class="col-sm-6">
9.         <h4>Some Input</h4>
10.        <p><input type="text" /></p>
11.        <p><input type="text" /></p>
12.        <p><input type="text" /></p>
13.        <p><input type="text" /></p>
14.        <p><input type="text" /></p>
15.        <p><input type="text" /></p>
16.      </div>
17.      <div class="col-sm-6">
18.        <h4>Some More Input</h4>
19.        <p><input type="text" /></p>
20.        <p><input type="text" /></p>
21.        <p><input type="text" /></p>
22.        <p><input type="text" /></p>
23.      </div>
24.    </div>
25.  </div>
26. </div>
```

At the bottom of the page, there is a "View Demo" button and a source link: "Source: <http://jschr.github.io/bootstrap-modal/>".

# Complementos (Plug-ins)

- Bootstrap Lightbox: Lightbox basado en el "plug-in" nativo de Bootstrap y totalmente personalizable.



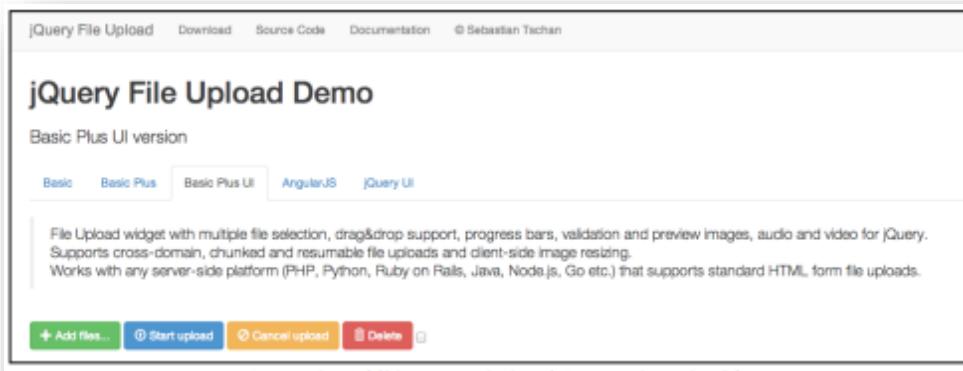
# Complementos (Plug-ins)

- Font Awesome. Incluye una serie de glifos añadidos a los de la versión gratuita que viene con Bootstrap



Source: <http://fortawesome.github.io/Font-Awesome/>

- jQuery File Upload (utiliza Bootstrap por defecto)



Source: <http://blueimp.github.io/jQuery-File-Upload/>

# Complementos (Plug-ins)

- Notify.js: permite crear sistemas de notificaciones vinculados a eventos de la aplicación, al estilo de la moderna "*Notification API*"
- Se distribuye con una configuración predeterminada para Bootstrap

The screenshot shows the GitHub page for the Notify.js project. At the top right, there's a green badge with a checkmark and the text "Hello World". Below it, a green ribbon banner says "1.0 on GitHub". The main title is "Notify.js". On the left, there's a "Summary" section with a brief description and a code snippet: `$().notify("Hello World");`. To the right, there's a "Download" section with two buttons: "Download notify.min.js" and "Download notify.js". Below that, there's a link to "Build a custom notify.js". Under the "Basic Usage" heading, there are two sections: "Element Notifications" and "Global Notifications". The "Element Notifications" section shows a code snippet: `$(".elem-demo").notify("Hello Box");` followed by the text "Like this

**Summary**  
Notify.js is a jQuery plugin to provide simple yet fully customizable notifications. The javascript code snippets in this documentation with the green edge are runnable by clicking them.

```
$().notify("Hello World");
```

**Download**  
Download pre-packaged with the bootstrap notification style:  
[Download notify.min.js](#) [Download notify.js](#)

Or, create a custom build with a different style  
[Build a custom notify.js](#)

**Basic Usage**

**Element Notifications**  
You can place notifications on any element:  
`$(".elem-demo").notify("Hello Box");`  
Like this

**Global Notifications**  
If you don't specify an element, notification will appear in the top left (unless you specify a different position - See [Positioning](#))  
`$().notify("I'm over here!");`

Source: <http://notifyjs.com/>

# Complementos (Plug-ins)

- typeahead.js.: Se trata de un complemento que permite la implantación y diseño de interfaces mejoradas para el API de Autocompletación de JS5.
- Está construido completamente por ingenieros de Twitter y funciona perfectamente con Bootstrap

The screenshot shows the GitHub repository page for typeahead.js. At the top, there's a logo with a blue bird icon and the text "typeahead.js" in bold. Below it, a subtitle reads "a fast and fully-featured autocomplete library". A search bar contains the letter "b". Below the search bar is a list of projects:

Project	Language
scala-bootstrapper	Ruby
bootstrap	JavaScript
bootstrap-server	JavaScript
bower-server	Ruby
bower	JavaScript

At the bottom of the screenshot, a source link is provided: "Source: <http://twitter.github.io/typeahead.js/>".

# Complementos (Plug-ins)

- X-editable.: Plug-in que permite editar directamente el contenidos de páginas Web

The screenshot shows the homepage of the X-editable project. At the top, there's a header with the title "X-editable" and a subtitle "In-place editing with Twitter Bootstrap, jQuery UI or plain jQuery". Below the header, a brief description states: "This library allows you to create editable elements on your page. It can be used with any engine (bootstrap, jquery-ui, jquery only) and includes both popup and inline modes. Please try out demo to see how it works." There are two buttons: "Demo in new window" (green) and "Documentation" (blue). Below the buttons, it says "Current version 1.5.1 changelog" and "Follow @vitalets".

At the bottom, there are three sections for different builds:

- Bootstrap (2 or 3)**: "Twitter Bootstrap has awesome Form inputs and Popover plugin allowing pretty in-place editing. For date input it uses bootstrap-datepicker." It shows a screenshot of a Bootstrap form with an input field containing "awesome" and a red "Edit" button. Buttons at the bottom are "Download 1.5.1 build" and "BS3 build".
- jQuery UI (1.8+)**: "Starting from jQuery UI 1.8 it has Tooltip plugin. In combination with Button widget it is suitable enough for editable form and popover. You can apply any theme of jQuery UI Themersons." It shows a screenshot of a jQuery UI form with an input field containing "awesome" and a red "Edit" button. Button at the bottom is "Download jquery-ui build".
- jQuery (+ poshytip plugin)**: "The simplest build of X-editable. For inline mode you need only jquery. For popover mode it uses Poshytip plugin. For date-time fields you can use combodate type based on momentjs." It shows a screenshot of a jQuery form with an input field containing "awesome" and a red "Edit" button. Button at the bottom is "Download jquery build".

Source: <http://vitalets.github.io/x-editable/>

# Personalizando Bootstrap

- Se pueden usar temas ya creados <https://startbootstrap.com/>
- Crear un tema propio <https://pikock.github.io/bootstrap-magic/app/> o <http://www.lavishbootstrap.com/> ; y descargar el css
- Crear un archivo css que sobreescriba los estilos de bootstrap
- Los css personalizados se deben incluir después del css base de bootstrap

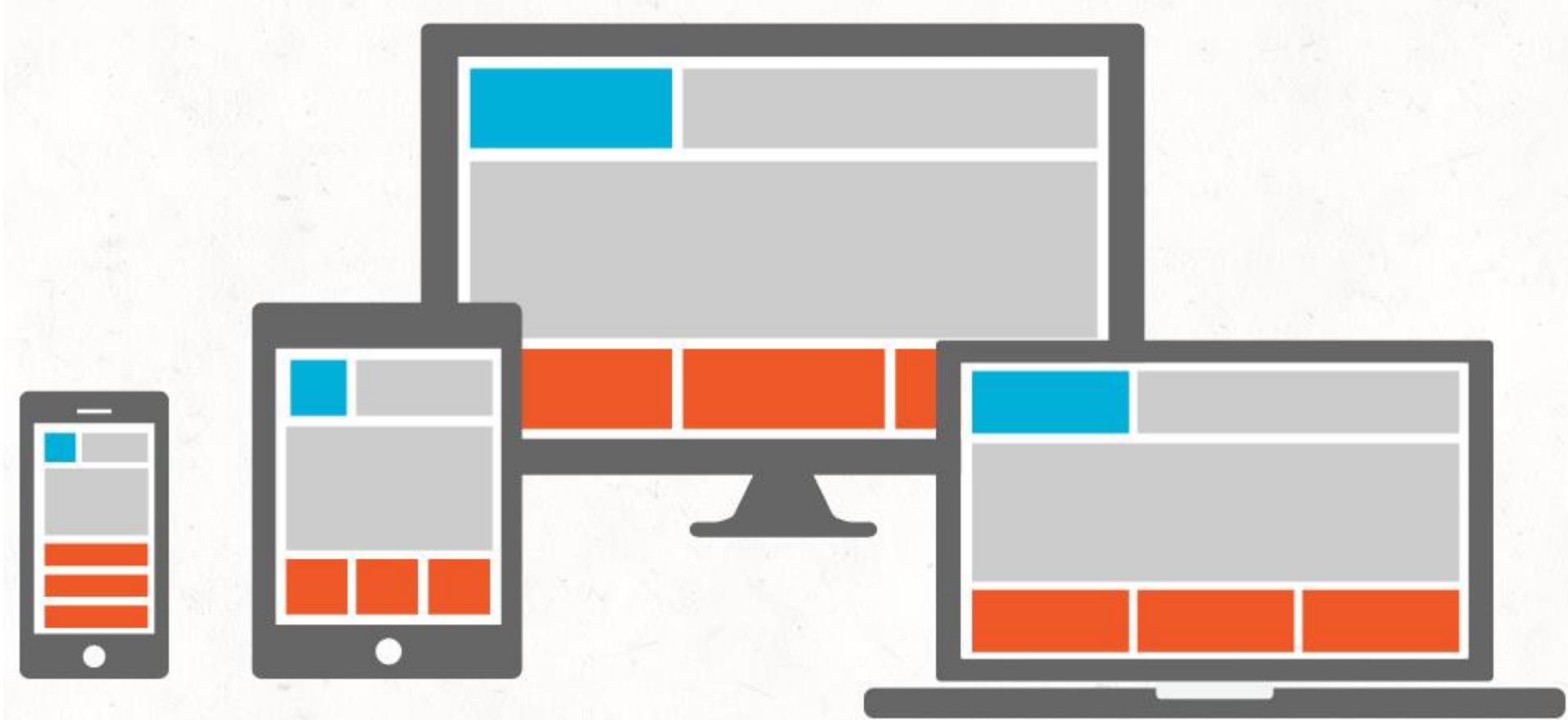
# Alternativas a Bootstrap: Material Design

- Proyecto de Google que propone un sistema unificado que combina teoría, recursos y herramientas para crear experiencias digitales.
- <https://material.io/>
- A efectos de la implementación web:
  - <http://materializecss.com/>
  - <https://www.muicss.com/>
  - <https://getmdl.io/>



## Pongámoslo en práctica

- Genera las web que cumplan los siguientes layouts





A screenshot of a website contact form on a desktop browser. The header features a mountain landscape background with a green circular logo containing a small mountain icon. The logo has the word "CALDERA" above "FORMS". Below the header, there are three navigation links: "Home", "Purchase History", and "Checkout".

**CONTACT US**  
Edit

**Your Details**  
Let us know how to get back to you.

**First name \***  **Last name \***  **Email Address \***

**How can we help?**  
Feel free to ask a question or simply leave a comment.

**Comments / Questions \***

A screenshot of the same contact form displayed on a mobile device. The top navigation bar includes a menu icon (three horizontal lines) and the word "Menu".

**CONTACT US**  
Edit

**Your Details**  
Let us know how to get back to you.

**First name \***

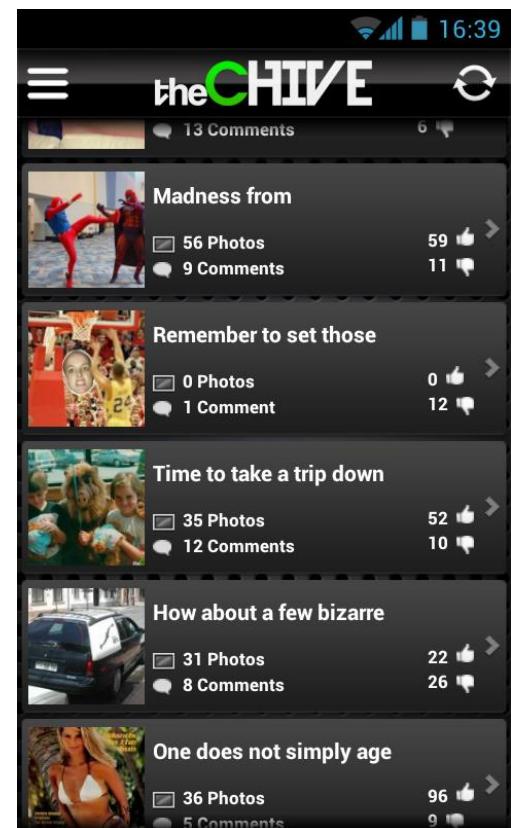
**Last name \***

**Email Address \***



# Pongámoslo en práctica

- Haz que theChive sea responsive
- <http://cdn-www.bluestacks.com/AndroidTapp/2012/03/The-Chive-Picture-list.png>





## Haciendo más responsive a BananaTube

- En equipo discute la conveniencia de usar Bootstrap en la implementación responsiva de Banana.
- Implementa la versión responsive de BananaTube usando Bootstrap
- BananaTube debe ser usable en dispositivos pequeños (smartphones), medianos (tablets), grandes (ordenador/portatil) y extragrandes (TV)



[...]**netmind**

WeKnowIT

Barcelona

C. Almogàvers, 123  
08018 Barcelona  
Tel. 93 304.17.20  
Fax. 93 304.17.22

Madrid

Plaza Carlos Trías Bertrán, 7  
28020 Madrid  
Tel. 91 442.77.03  
Fax. 91 442.77.07

[www.netmind.es](http://www.netmind.es)



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE ENERGÍA, TURISMO  
Y AGENDA DIGITAL

**red.es**



ESTRATEGIA DE  
EMPRENDIMIENTO Y  
EMPLEO JUVENIL  
*garantía juvenil*



Agenda Digital para España



**UNIÓN EUROPEA**

Fondo Social Europeo  
*“El FSE invierte en tu futuro”*