

Componentes CSS de Bootstrap

Notificaciones

Las notificaciones son muy importantes en tu web. Nos permiten alertar sobre el éxito o fracaso de una operación que el usuario intenta realizar, informar sobre un cambio de la información o novedad (por ejemplo, como lo hacen las herramientas de gestión de correo electrónico que informan con un número la cantidad de mensajes sin leer) o simplemente mantienen actualizado al usuario sobre el progreso de una determinada operación que demora más de lo habitual.

Todas estas situaciones suceden en cada proyecto y el usuario debe permanecer informado en forma inmediata, clara y breve, evitando demoras o dando lugar a dobles interpretaciones. Debemos ser eficaces en la comunicación.

Para lograr este objetivo, Bootstrap nos provee 3 componentes que nos ayudarán:

- Badges
- Alertas
- Barras de progreso

Veremos cada uno de estos elementos y comentaremos situaciones de uso habitual. Te animo a que observes los sitios que cotidianamente visitas e identifiques dónde los utilizan.

Badges

Comencemos por los badges. Los badges tienen como finalidad brindar información de contexto o complementaria. Básicamente, agregan texto con un formato especial a un elemento o bien agregan un número también con formato especial. Veamos unos ejemplos que nos propone Bootstrap:

Example heading **New**

Example heading **New**

Example heading **New**

Example heading **New**

Example heading **New**

Example heading **New**

```
<h1>Example heading <span class="badge badge-secondary">New</span></h1>
<h2>Example heading <span class="badge badge-secondary">New</span></h2>
<h3>Example heading <span class="badge badge-secondary">New</span></h3>
<h4>Example heading <span class="badge badge-secondary">New</span></h4>
<h5>Example heading <span class="badge badge-secondary">New</span></h5>
<h6>Example heading <span class="badge badge-secondary">New</span></h6>
```

En este caso, observá que *NEW* aparece resaltado informando que eso es algo nuevo. Obviamente, cuando reemplacemos *NEW* por lo que nosotros necesitamos, eso es lo que queremos que el usuario rápidamente identifique. Es decir, el badge en este caso resalta lo novedad sobre el elemento que tenemos a la izquierda. Como comentario técnico, los badges siempre se crean sobre un elemento span agregando las clases badge y badge-* (* para indicar la clase contextual, primary, secondary, etc). Siempre escalan la fuente en función del elemento que lo contiene.

Tomate unos minutos y pensá en qué casos lo usarías.

Algunas situaciones comunes de uso son resaltar secciones nuevas, nuevos elementos, nuevos productos o ítems, algunos ubican fechas en el badges para resaltar la temporalidad del elemento. En las prácticas veremos más situaciones.

También podemos utilizar badges sin necesidad de que estén contenidos por algún otro elemento. Aquí tenemos 2 alternativas para hacerlo: de la misma forma que mostramos antes o bien, utilizando la clase badge-pill. Los badge-pills o pills tienen los bordes redondeados. Su uso técnico es similar a todos los badges. Son prácticos para indicar etiquetas, tags, o textos que no necesitamos que estén asociados a otras palabras sino que tienen peso por sí mismos.

Primary Secondary Success Danger Warning Info Light Dark

```
<span class="badge badge-primary">Primary</span>
<span class="badge badge-secondary">Secondary</span>
<span class="badge badge-success">Success</span>
<span class="badge badge-danger">Danger</span>
<span class="badge badge-warning">Warning</span>
<span class="badge badge-info">Info</span>
<span class="badge badge-light">Light</span>
<span class="badge badge-dark">Dark</span>
```

Primary Secondary Success Danger Warning Info Light Dark

```
<span class="badge badge-pill badge-primary">Primary</span>
<span class="badge badge-pill badge-secondary">Secondary</span>
<span class="badge badge-pill badge-success">Success</span>
<span class="badge badge-pill badge-danger">Danger</span>
<span class="badge badge-pill badge-warning">Warning</span>
<span class="badge badge-pill badge-info">Info</span>
<span class="badge badge-pill badge-light">Light</span>
<span class="badge badge-pill badge-dark">Dark</span>
```

Otro uso que podemos hacer con los badges es indicar un valor numérico que acompañe a un texto, similar a lo que hacen algunos gestores de correo electrónico con la cantidad de mensajes sin leer o algunos sitios con las notificaciones pendientes. En el ejemplo siguiente, lo hemos incluido dentro de un botón.

Notifications 4

```
<button type="button" class="btn btn-primary">
  Notifications <span class="badge badge-light">4</span>
</button>
```

Como resumen de los badges, notemos que la estructura es similar en todos los casos: utilizamos un elemento span y le indicamos las clases badge y badge-* para definir los colores o formatos.

Alertas

Las alertas (alerts) las utilizaremos para notificar inmediatamente el resultado de una acción al usuario. Por ejemplo, en caso de concretar una compra, mostraremos una alerta indicando el éxito de la operación, o en caso de fallar, indicaremos con un fondo rojo el motivo de error. Siempre que el usuario realice una acción, debemos notificar inmediatamente el resultado de la misma. En esas situaciones, las alertas son muy útiles y por lo general utilizaremos este componente que nos brinda Bootstrap.

Veamos un ejemplo de uso:

This is a primary alert—check it out!

This is a secondary alert—check it out!

This is a success alert—check it out!

This is a danger alert—check it out!

This is a warning alert—check it out!

This is an info alert—check it out!

This is a light alert—check it out!

This is a dark alert—check it out!

```
<div class="alert alert-primary" role="alert">
  This is a primary alert—check it out!
</div>
<div class="alert alert-secondary" role="alert">
  This is a secondary alert—check it out!
</div>
<div class="alert alert-success" role="alert">
  This is a success alert—check it out!
</div>
<div class="alert alert-danger" role="alert">
  This is a danger alert—check it out!
</div>
<div class="alert alert-warning" role="alert">
  This is a warning alert—check it out!
</div>
<div class="alert alert-info" role="alert">
  This is an info alert—check it out!
</div>
<div class="alert alert-light" role="alert">
  This is a light alert—check it out!
</div>
<div class="alert alert-dark" role="alert">
  This is a dark alert—check it out!
</div>
```

Como ves, basta con utilizar un div con las clases alert y alert-* para indicar el color de contexto.

Estas notificaciones se suelen utilizar debajo del header, o barra de navegación si el sitio posee. Deben visualizarse fácilmente.

Ahora bien, si mostramos una alerta vamos a requerir que el usuario la descarte ya que no es muy útil una notificación permanente y, por otro lado, una alerta debe aparecer sólo cuando lo necesitamos, no siempre. Veamos qué podemos hacer para atacar estas dos situaciones.

Nuevos viajes! Entra a nuestra sección del Caribe para conocer las nuevas propuestas



```
<div class="alert alert-warning alert-dismissible fade show" role="alert">
  <strong>Nuevos viajes!</strong> Entra a nuestra sección del Caribe para conocer las nuevas
  <button type="button" class="close" data-dismiss="alert" aria-label="Close">
    <span aria-hidden="true">&times;</span>
  </button>
</div>
```

Copy

En el ejemplo precedente vemos que se muestra una “X” para indicar el cierre de la notificación. Bootstrap nos facilita esta utilidad vía Javascript y debemos agregarle las clases que ves en el ejemplo para que se concrete el cierre de la notificación y que desaparezca suavemente.

Para mostrar una alerta u ocultarla, tenemos a disposición las librerías de Javascript que incluye Bootstrap. Desde la programación, podremos mostrar u ocultar las alertas como queramos con algunas animaciones que suavizan la transición. Veremos su uso en la sección de javascript.

Barras de progreso

Las barras de progreso, o progress bars, indican el avance de una operación en tiempo real. Veremos el componente que nos ofrece Bootstrap y las varias que tenemos para su uso. De todas formas, en la sección de Javascript profundizaremos sobre su uso ya que debemos utilizar un script JS para mantener actualizado el avance de la barra de acuerdo al progreso de la acción. Para que se entienda mejor, debemos monitorear la operación observando el grado de avance y con ese dato, actualizar la barra de progreso de forma acorde.

Básicamente, están formadas por dos componentes: una que indica el contexto de la barra y otro dentro de este que ocupa el rol de la barra en sí. Veamos el ejemplo de la documentación:



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

Copy

Tenemos un contenedor con el elemento div y la clase progress. Dentro de este, tenemos otro div con la clase progress-bar. El elemento interior tiene varios atributos, pero el más importante es el valor del width donde indicamos el nivel de progreso (lo que debe ocupar el azul de la barra).

Si incluimos un texto antes del cierre del div de la barra, ese texto se mostrará por defecto en color blanco y centrado en el ancho del progreso de la barra. La práctica habitual es incluir el número del porcentaje de progreso y en caso de que la barra no tenga progreso porque no realizó la acción o

tiene un valor igual a 0, se suele indicar un mensaje que explique la situación (por ejemplo, “no hay suficientes elementos” o “acción no iniciada”).

Otra configuración que podemos realizar es el valor de la altura (atributo height).



```
<div class="progress" style="height: 1px;">
  <div class="progress-bar" role="progressbar" style="width: 25%;" aria-valuenow="25" aria-v
</div>
<div class="progress" style="height: 20px;">
  <div class="progress-bar" role="progressbar" style="width: 25%;" aria-valuenow="25" aria-v
</div>
```

Copy

Como ves en el ejemplo, podemos tener una delgada línea de progreso hasta una bruta barra. Dependerá del contexto su decisión.

Las barras poseen muchas más configuraciones que dejaré que las veas desde la documentación de Bootstrap ya que son variantes de lo que ya vimos. A modo de anticipo, verás que se pueden cambiar los colores de la barra, incluir un estilo diferente de barra, ubicar varias barras de progreso dentro de una misma barra y hasta incorporar una animación vía CSS3.