

# Tema 4: Bootstrap

Semana 15: Alerts, badge, paginación, popovers, etc.



## BLOQUE 3: Bootstrap

Módulo: Desarrollo de Interfaces Web

Ciclo: Desarrollo de Aplicaciones Web

Curso: 2023 – 2024

Profesora: Rosa Medina

## Índice

1. Introducción.....	3
2. Alerts.....	3
2.1. Visualizar/ocultar alertas.....	3
3. Badge.....	4
3.1. Badges dependiendo del contenido.....	4
4. Migas de pan / breadcrumb.....	4
5. Grupo de listados.....	5
6. Modal.....	6
7. Paginación.....	6
8. Barra de progreso.....	7
9. Popovers.....	7
10. Scrollspy.....	8
10.1. Ejemplo en navbar.....	8
10.2. Ejemplo con nested nav.....	10

# 1. Introducción

En la unidad anterior vimos: botones, tarjetas, carousel, collapse, desplegables, formularios, navs y navbar. En la presente unidad, veremos:

- Alerts
- Badge
- Migas de pan
- Grupo de listados
- Modal
- Paginación
- Popovers
- Barra de progreso
- Scrollspy
- Tooltips

## 2. Alerts

Los [alerts](#) que nos ofrece Bootstrap están disponibles para textos de cualquier longitud (.alert), pudiendo tener un botón de cerrar/quitar alerta ([ver alertas JS plugin](#)). Veamos un ejemplo básicos de alerts (alerts.html):

<pre> &lt;div class="alert alert-primary" role="alert"&gt;   This is a primary alert—check it out! &lt;/div&gt; &lt;div class="alert alert-secondary" role="alert"&gt;   This is a secondary alert—check it out! &lt;/div&gt; &lt;div class="alert alert-success" role="alert"&gt;   This is a success alert—check it out! &lt;/div&gt; &lt;div class="alert alert-danger" role="alert"&gt;   This is a danger alert—check it out! &lt;/div&gt; &lt;div class="alert alert-warning" role="alert"&gt;   This is a warning alert—check it out! &lt;/div&gt; &lt;div class="alert alert-info" role="alert"&gt;   This is a info alert—check it out! &lt;/div&gt; &lt;div class="alert alert-light" role="alert"&gt;   This is a light alert—check it out! &lt;/div&gt; &lt;div class="alert alert-dark" role="alert"&gt;   This is a dark alert—check it out! &lt;/div&gt; </pre>	
--	--

### 2.1. Visualizar/ocultar alertas

Usando el plugin de alertas de JavaScript, es posible ocultar alerts, veamos cómo se usa:

- Debemos tener el plugin de alerts, o el JS compilado de Bootstrap.

- Debemos añadir un botón con la clase `.alert-dismissible`, el cual añade un padding extra a la derecha para poner el botón de cerrar.
- El botón tendrá el atributo `data-bs-dismiss="alert"`, el cual lanza la funcionalidad de JavaScript. Debemos asegurarnos que estamos utilizando el elemento `<button>`
- Para animar las alertas cuando se oculten, añadiremos las clases `.fade` y `.show`

```
<div class="alert alert-warning alert-dismissible fade show" role="alert">
  <strong>Holy guacamole!</strong> You should check in on some of those fields below.
  <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>
</div>
```

**Holy guacamole! You should check in on some of those fields below.**



Para conocer más funcionalidades sobre las alertas de Bootstrap con JS, podéis consultarlo [aquí](#).

### 3. Badge

Los [badges](#) son pequeñas etiquetas que se adaptan para añadir un texto al lado de otro. Un ejemplo de badges puede ser utilizado como para resaltar algo nuevo, al lado de una cabecera. Ejemplo (badge.html):

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

Example heading **New**

Example heading **New**

Example heading **New**

Example heading **New**

Example heading **New**

Example heading **New**

#### 3.1. Badges dependiendo del contenido

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

**Primary**

**Secondary**

**Success**

**Danger**

**Warning**

**Info**

**Light**

**Dark**

### 4. Migas de pan / breadcrumb

[Breadcrumb](#) o migas de pan, son bastante útiles para saber la navegación a seguir hasta la página en donde nos encontramos, los separadores que usamos se añaden de forma automática con CSS.

```
<nav aria-label="breadcrumb">
  <ol class="breadcrumb">
    <li class="breadcrumb-item active" aria-current="page">Home</li>
  </ol>
</nav>
```

Home

```
<nav aria-label="breadcrumb">
  <ol class="breadcrumb">
    <li class="breadcrumb-item"><a href="#">Home</a></li>
    <li class="breadcrumb-item active" aria-
current="page">Library</li>
  </ol>
</nav>
```

[Home](#) / Library[Home](#) / [Library](#) / Data

```
<nav aria-label="breadcrumb">
  <ol class="breadcrumb">
    <li class="breadcrumb-item"><a href="#">Home</a></li>
    <li class="breadcrumb-item"><a href="#">Library</a></li>
    <li class="breadcrumb-item active" aria-current="page">Data</li>
  </ol>
</nav>
```

## 5. Grupo de listados

Hay varios modelos básicos de [listados](#) predefinidos en Bootstrap, veamos poco a poco cada uno de ellos cómo hacerlos.

```
<ul class="list-group">
  <li class="list-group-item">Cras justo odio</li>
  <li class="list-group-item">Dapibus ac facilisis in</li>
  <li class="list-group-item">Morbi leo risus</li>
  <li class="list-group-item">Porta ac consectetur ac</li>
  <li class="list-group-item">Vestibulum at eros</li>
</ul>
```

Cras justo odio

Dapibus ac facilisis in

Morbi leo risus

Porta ac consectetur ac

Vestibulum at eros

Si queremos indicar qué sección se encuentra activa, lo que podemos hacer es añadir la clase `.active` al ítem correspondiente

```
<ul class="list-group">
  <li class="list-group-item active">Cras justo odio</li>
  <li class="list-group-item">Dapibus ac facilisis in</li>
  <li class="list-group-item">Morbi leo risus</li>
  <li class="list-group-item">Porta ac consectetur ac</li>
  <li class="list-group-item">Vestibulum at eros</li>
</ul>
```

Cras justo odio

Dapibus ac facilisis in

Morbi leo risus

Porta ac consectetur ac

Vestibulum at eros

## 6. Modal

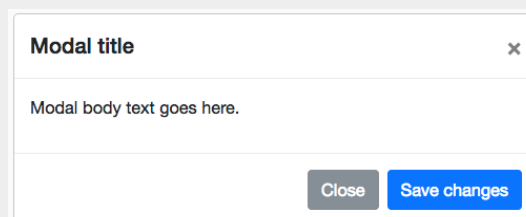
Cómo funcionan los [modal](#) de Bootstrap:

- Modal está hecho con HTML, CSS y JS. Están posicionados sobre todo lo demás, y eliminan el scroll del <body> para que el scroll sea el del contenido del modal en su lugar.
- Al hacer clic sobre el modal “backdrop” automáticamente se cerrará el modal.
- Bootstrap solo soporta una ventana de un modal a la vez. Nested modals no están soportados, de hecho, se considera que es una mala práctica utilizarlo.
- Modal usa position: fixed, el cual a veces puede ser un poco particular sobre su renderizado. Siempre que sea posible, sitúa el modal en el top del HTML para evitar interferencias con cualquier otro elemento. Probablemente, tendrás algún problema con .modal si tienes algún otro elemento fijo.
- Debido a position: fixed, hay alguna que otra advertencia a la hora de utilizar los modals en los móviles. Podemos consultar con más detalle la documentación [aquí](#).
- El atributo HTML [autofocus](#) no se aplica en los modals, este efecto lo puedes conseguir si quieres mediante JS.

### Ejemplos de uso del componente modal

Ejemplo de un modal que aparece al pulsar el botón (dinámico)

```
<div class="modal" tabindex="-1">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title">Modal title</h5>
        <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
      </div>
      <div class="modal-body">
        <p>Modal body text goes here.</p>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary"
data-bs-dismiss="modal">Close</button>
        <button type="button" class="btn btn-primary">Save changes</button>
      </div>
    </div>
  </div>
</div>
```



## 7. Paginación

La paginación se utiliza para tener un bloque de enlaces y poder acceder a ellos de forma fácil. La forma de construirlo es mediante una lista de elementos en HTML, estos

enlaces deben estar agrupados dentro del elemento `<nav>`, cada uno de los elementos tendrá la clase `.page-item` y cada enlace la clase `.page-link`, veamos un ejemplo tanto si usamos iconos como si usamos texto.

Ejemplo de paginación con texto:

```
<nav aria-label="Page navigation example">
  <ul class="pagination">
    <li class="page-item"><a class="page-link" href="#">Previous</a></li>
    <li class="page-item"><a class="page-link" href="#">1</a></li>
    <li class="page-item"><a class="page-link" href="#">2</a></li>
    <li class="page-item"><a class="page-link" href="#">3</a></li>
    <li class="page-item"><a class="page-link" href="#">Next</a></li>
  </ul>
</nav>
```



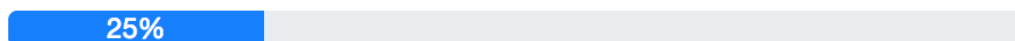
## 8. Barra de progreso

El componente de [barra de progreso](#) se construye con dos elementos HTML, algunas propiedades CSS permiten establecer el ancho y algunos atributos. Nosotros no vamos a utilizar el elemento `<progress>` de HTML sino la clase que ha diseñado Bootstrap para ello.

- Usaremos la clase `.progress` como un wrapper para indicar el máximo valor del progress bar
- Usaremos `.progress-bar` para indicar el progreso
- El `.progress-bar` requiere un estilo inline o un diseño CSS customizado para establecer el ancho
- El `.progress-bar` también requiere los atributos `role` y `aria`.

Ejemplo del % que ha realizado

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



## 9. Popovers

A continuación, vamos a estudiar [popovers](#):

- Para poder utilizar los popovers en Bootstrap, debemos incluir la librería de terceros `Popper.min.js`, y esta debe ser incluida antes de incluir `bootstrap.bundle.min.js` para que funcione.
- Necesita [popover plugin](#)

- El uso de un popover es opcional, por tanto en el caso de querer usarlo debemos ser nosotros quien lo inicialicemos.
- Un título y contenido vacío, nunca se mostrará como popover
- Especifica container:'body' para evitar problemas de renderizados en muchos componentes (como en los grupos de inputs que tiene Bootstrap, grupos de botones, etc.)
- No funciona disparar un popover en elementos ocultos.
- Los popovers en atributos o clases disabled deben ser activados en un elemento wrapper
- Cuando disparamos un popover desde un hyperlink que contiene múltiples líneas este debería ser centrado. Para ello usaremos .text-nowrap en nuestro elemento <a> para evitar este comportamiento
- Popovers deberían ser ocultos antes que sus correspondientes elementos HTML hayan sido eliminados del DOM.

Una forma de inicializar cualquier popover en una página web puede ser con el atributo data-bs-toggle:

```
const popoverTriggerList = document.querySelectorAll('[data-bs-toggle="popover"]')
const popoverList = [...popoverTriggerList].map(popoverTriggerEl => new
bootstrap.Popover(popoverTriggerEl))
```

Ejemplo completo [aquí](#).

## 10. Scrollspy

El componente [Scrollspy](#) requiere para su correcto funcionamiento:

- Debemos utilizarlo con un componente [nav](#) o [list group](#)
- Requiere la propiedad position: relative; en el element que estamos observando, normalmente es el <body>
- La etiqueta <a> es requerida y con el elemento id
- Cuando lo tengamos realizando, nuestro nav o list group se irá actualizando poco a poco, moviendo la clase .active de un ítem a otro basado en el target que tenga asignado.

### 10.1. Ejemplo en navbar

Al hacer scroll en el área de abajo del navbar, vamos viendo qué elemento está activo en el menú. En el dropdown también se irán marcando aquellos ítems que estén activos.

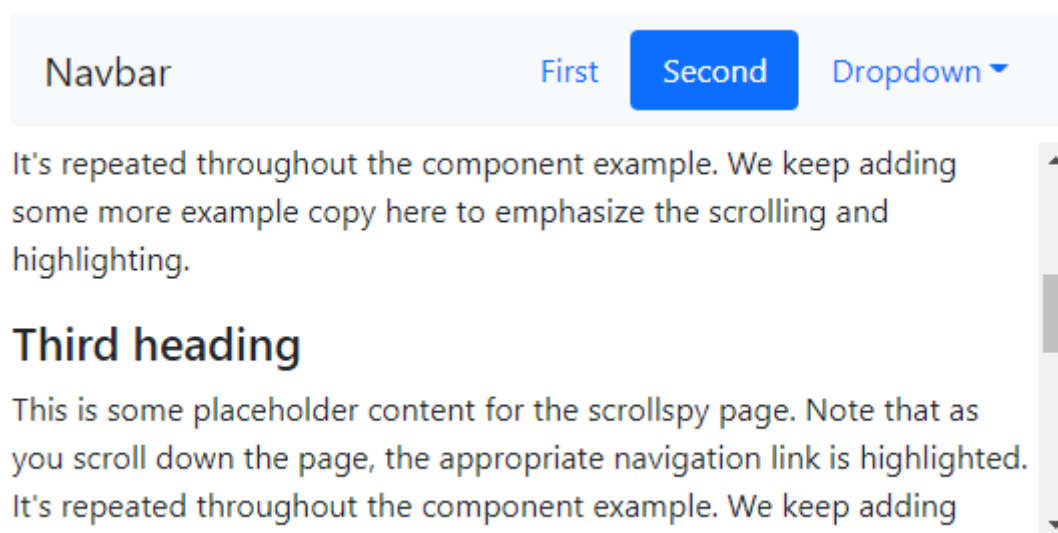
```
<nav id="navbar-example2" class="navbar navbar-light bg-light px-3">
  <a class="navbar-brand" href="#">Navbar</a>
  <ul class="nav nav-pills">
    <li class="nav-item">
      <a class="nav-link" href="#scrollspyHeading1">First</a>
```



```

</li>
<li class="nav-item">
  <a class="nav-link" href="#scrollspyHeading2">Second</a>
</li>
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle" data-bs-toggle="dropdown" href="#" role="button" aria-
expanded="false">Dropdown</a>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="#scrollspyHeading3">Third</a></li>
    <li><a class="dropdown-item" href="#scrollspyHeading4">Fourth</a></li>
    <li><hr class="dropdown-divider"></li>
    <li><a class="dropdown-item" href="#scrollspyHeading5">Fifth</a></li>
  </ul>
</li>
</ul>
</nav>
<div data-bs-spy="scroll" data-bs-target="#navbar-example2" data-bs-offset="0" class="scrollspy-
example" tabindex="0">
  <h4 id="scrollspyHeading1">First heading</h4>
  <p>...</p>
  <h4 id="scrollspyHeading2">Second heading</h4>
  <p>...</p>
  <h4 id="scrollspyHeading3">Third heading</h4>
  <p>...</p>
  <h4 id="scrollspyHeading4">Fourth heading</h4>
  <p>...</p>
  <h4 id="scrollspyHeading5">Fifth heading</h4>
  <p>...</p>
</div>

```



## 10.2. Ejemplo con nested nav

El scrollspy también funciona con nav nested. Si un nested .nav está active, su padre también lo estará. Al hacer scroll en el área que hay al lado del navbar podemos ver cómo las clases del navbar van cambiando en función del punto en el que nos encontramos.

```
<nav id="navbar-example3" class="navbar navbar-light bg-light flex-column align-items-stretch p-3">
  <a class="navbar-brand" href="#">Navbar</a>
  <nav class="nav nav-pills flex-column">
    <a class="nav-link" href="#item-1">Item 1</a>
    <nav class="nav nav-pills flex-column">
      <a class="nav-link ms-3 my-1" href="#item-1-1">Item 1-1</a>
      <a class="nav-link ms-3 my-1" href="#item-1-2">Item 1-2</a>
    </nav>
    <a class="nav-link" href="#item-2">Item 2</a>
    <a class="nav-link" href="#item-3">Item 3</a>
    <nav class="nav nav-pills flex-column">
      <a class="nav-link ms-3 my-1" href="#item-3-1">Item 3-1</a>
      <a class="nav-link ms-3 my-1" href="#item-3-2">Item 3-2</a>
    </nav>
  </nav>
</nav>

<div data-bs-spy="scroll" data-bs-target="#navbar-example3" data-bs-offset="0" tabindex="0">
  <h4 id="item-1">Item 1</h4>
  <p>...</p>
  <h5 id="item-1-1">Item 1-1</h5>
  <p>...</p>
  <h5 id="item-1-2">Item 1-2</h5>
  <p>...</p>
  <h4 id="item-2">Item 2</h4>
  <p>...</p>
  <h4 id="item-3">Item 3</h4>
  <p>...</p>
  <h5 id="item-3-1">Item 3-1</h5>
  <p>...</p>
  <h5 id="item-3-2">Item 3-2</h5>
  <p>...</p>
</div>
```

## Navbar

Item 1

Item 1-1

Item 1-2

Item 2

Item 3

Item 3-1

Item 3-2

incididunt sint et culpa culpa incididunt  
adipiscing magna magna occaecat.  
Nulla ipsum cillum eiusmod sint elit  
excepteur ea labore enim consectetur in  
labore anim. Proident ullamco ipsum  
esse elit ut Lorem eiusmod dolor et  
eiusmod. Anim occaecat nulla in non  
consequat eiusmod velit incididunt.

## Item 2

Quis magna Lorem anim amet ipsum do  
mollit sit cillum voluptate ex nulla  
tempor. Laborum consequat non elit  
enim exercitation cillum aliqua  
consequat id aliqua. Esse ex consectetur