

Bootstrap JS

Bootstrap JS	1
¿Qué aprenderás?	2
Introducción	2
Smooth Scroll	3
Bootstrap JS: Popover, Modal, Tooltip y Dropdown	6
Popover	7
Modal	8
Tooltip	9
Dropdown	10
Carousel de Bootstrap	11
Resumen	12



¡Comencemos!

¿Qué aprenderás?

- Implementar componentes JavaScript de Bootstrap, para agregar elementos visuales a un sitio web.

Introducción

En las siguientes páginas retomaremos el framework Bootstrap, como herramienta para construir sitios web de forma más ágil y con recursos adicionales, como vimos anteriormente. Esta vez, añadiremos componentes que utilizan jQuery y nos permitirán añadir dinamismo y efectos interesantes a nuestros proyectos.

Es importante mencionar que a partir de la versión 5 de Bootstrap, se elimina la dependencia con jQuery y se agrega VanillaJS.

¡Vamos con todo!




Smooth Scroll

Se le denomina Smooth Scroll, al efecto de transición entre diferentes secciones de una misma página sin tener que recargar, lo que mejora la experiencia del usuario y el aspecto visual de nuestro sitio.


Este efecto se puede lograr tanto con CSS, como con jQuery. En el caso de CSS, existe una propiedad llamada **scroll-behavior**.

```
html {
  scroll-behavior: smooth;
}
```

Sin embargo, aún existen navegadores que no soportan esta propiedad, por lo que se sugiere utilizar otras alternativas, como JavaScript puro o su biblioteca jQuery:

CSSOM Scroll-behavior  Usage % of all users Global 76.27%

Method of specifying the scrolling behavior for a scrolling box, when scrolling happens due to navigation or CSSOM scrolling APIs.

Current aligned Usage relative Date relative Filtered All 

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Opera Mobile *	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet
			4-40		10-27								
	12-18	2-35	41-60		28-47								4-7.4
6-10	79-86	36-83	61-86	3.1-13.1	48-71	3.2-13.7		2.1-4.4.4	12-12.1				8.2-12.0
11	87	84	87	14	72	14.2	all	81	59	87	83	12.12	13.0
		85-86	88-90	TP									

Imagen 1. Compatibilidad scroll-behavior.

Fuente: [Can I Use](https://caniuse.com/scroll-behavior).



Nota: La página [Can I Use](https://caniuse.com/) es muy utilizada para ver compatibilidad de navegadores con funcionalidades que queramos incluir en nuestros proyectos. Posee un navegador en la parte superior donde pondremos el nombre de la funcionalidad y nos mostrará una imagen como la anterior, con el detalle del navegador-versión-compatibilidad.

En el caso de jQuery, se deben tener en cuenta un par de consideraciones en el HTML. Veamos un ejemplo en [este link](#).

Si miramos el ejemplo de W3Schools, podemos identificar el código. Este se compone de dos secciones:

```
<h1>Smooth Scroll</h1>
<div class="main" id="section1">
  <h2>Section 1</h2>
  <p>Click on the link to see the "smooth" scrolling effect.</p>
  <a href="#section2">Click Me to Smooth Scroll to Section 2 Below</a>
  <p>Note: Remove the scroll-behavior property to remove smooth
scrolling.</p>
</div>

<div class="main" id="section2">
  <h2>Section 2</h2>
  <a href="#section1">Click Me to Smooth Scroll to Section 1 Above</a>
</div>
```

Cada sección, posee un id:

```
<div class="main" id="section1">
<div class="main" id="section2">
```

Y cada uno de los links, refieren al id de la sección a la que queremos ir una vez que hagamos click sobre ellos:

```
<a href="#section2">
<a href="#section1">
```

En el CSS, solo para graficar, se le está asignando un color de fondo y un alto definido:

```
<style>
#section1 {
  height: 600px;
  background-color: pink;
}

#section2 {
  height: 600px;
  background-color: yellow;
}
</style>
```

La magia se produce en el script de jQuery:

```
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"><
/script>
<script>
$(document).ready(function(){
  $("a").on('click', function(event) {
    if (this.hash !== "") {
      event.preventDefault();
      var hash = this.hash;

      $('html, body').animate({
        scrollTop: $(hash).offset().top
      }, 800, function(){

        window.location.hash = hash;
      });
    }
  });
});
</script>
```

- Al hacer click en el elemento `<a>` se valida que el valor del ancla (#) no sea vacía.

- Se ejecuta el método [preventDefault\(\)](#), que evita el comportamiento por defecto del elemento al hacer click sobre él.
- Asigna el valor del ancla a la variable hash.
- Se activa el método [animate\(\)](#), con la propiedad scrollTop, que obtiene el valor del borde superior del elemento a través del offset del mismo y una duración de la animación de 800 milisegundos.
- Finalmente, realiza la ejecución del scroll hacia la zona definida.

Bootstrap JS: Popover, Modal, Tooltip y Dropdown

El Framework Bootstrap, contiene muchos elementos visuales e interacciones de usuario que funcionan con JavaScript, no obstante no todos necesitan de este script. Cuando elegimos algún elemento de Bootstrap lo primero que tenemos que hacer es ir a su [documentación](#) y revisar en los primeros apartados si existe alguna sección que especifique que se tenga que ejecutar método para habilitar la funcionalidad del componente.

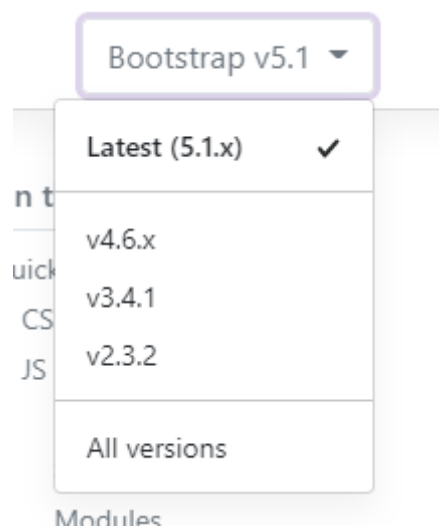


Imagen 2. Trabajando con bootstrap.

Fuente: [Bootstrap 5.1](#).

Siempre es recomendable copiar el código que aparece en la documentación ya que este se actualiza de forma constante, en base a la versión seleccionada.

Ahora conoceremos algunos elementos de Bootstrap.

Popover

Un popover es un elemento visual que se utiliza para mostrar un mensaje emergente, con una referencia específica a un tema.

Podremos verlo en acción en el siguiente enlace [documentación](#).

Si queremos añadirlo en nuestros proyectos solo debemos incluir las siguientes instrucciones:

```
var popoverTriggerList =  
[].slice.call(document.querySelectorAll('[data-bs-toggle="popover"]'))  
var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {  
  return new bootstrap.Popover(popoverTriggerEl)  
})
```

Esto añadirá un tooltip a cada elemento que tenga el atributo `data-bs-toggle="popover"`. También contamos con los atributos `title` y `data-bs-content` para definir el título y el contenido del popover, por ejemplo tenemos el siguiente botón HTML.

```
<button  
type="button"  
class="btn btn-lg btn-danger"  
data-bs-toggle="popover"  
title="Popover title"  
data-bs-content="And here's some amazing content. It's very engaging.  
Right?">Click to toggle popover</button>
```

Como puedes observar, tenemos los 3 atributos mencionados anteriormente con los que declaramos que el elemento tendrá un popover y cuál será su título y contenido, y visualmente se vería como la siguiente imagen.

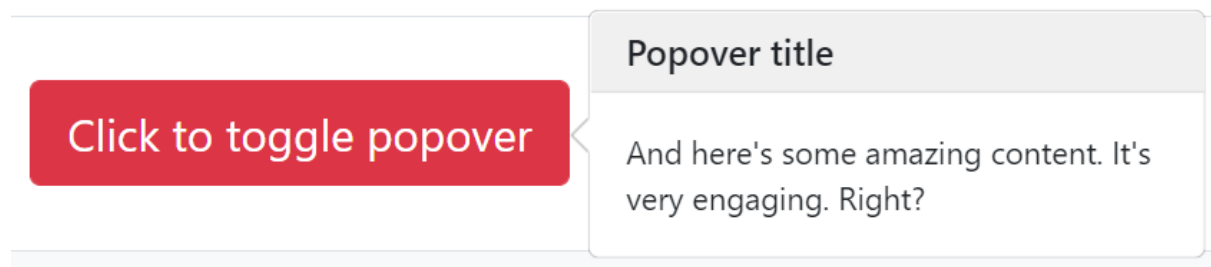


Imagen 3. Ejemplo Popover.

Fuente: [Bootstrap](#).

Modal

Un modal es otro elemento importante que podemos utilizar gracias a las características de Bootstrap. Sirve para añadir diálogos o mostrar notificaciones. Podemos ver su efecto en la siguiente [documentación](#).

Este componente no necesita de una inicialización en JavaScript como el popover, basta con especificar en un elemento el atributo `data-bs-toggle="modal"` y el atributo `data-bs-target` cuyo valor debe ser el id del modal que tengamos integrado (incluyendo el símbolo #), así como se muestra en el siguiente código HTML.

```
<!-- Button trigger modal -->
<button class="btn btn-primary" data-bs-toggle="modal"
data-bs-target="#exampleModal">
  Launch demo modal
</button>

<!-- Modal -->
<div class="modal fade" id="exampleModal" >
  <div class="modal-dialog">
    <div class="modal-content p-5">
      Probando modal...
    </div>
  </div>
</div>
```


Podremos observar su funcionamiento en la documentación de Bootstrap, presionando el botón Launch demo modal.

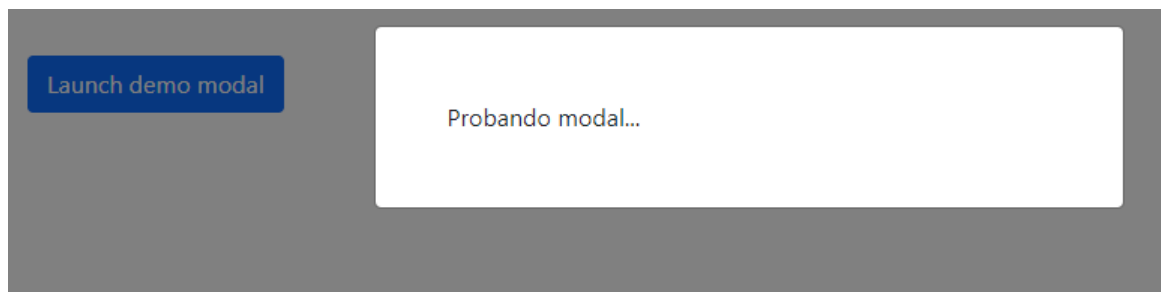


Imagen 4. Funcionamiento de Modal.

Fuente: [Bootstrap](#).

Como podrás observar, este código de ejemplo se ha simplificado bastante en comparación con el de la documentación oficial en donde se tiene una header, body y un footer, sin embargo esto no impide el funcionamiento de este componente ni su personalización gráfica.

Tooltip

Un tooltip es una herramienta aparentemente similar al popover pero que tiene un concepto semántico diferente. El tooltip solo mostrará una ayuda cuando pasemos el puntero sobre la palabra o botón donde se haya añadido. Como su nombre lo dice, es una pista respecto al funcionamiento del elemento en particular.

Si accedes a su [documentación](#) podrás ver la forma de implementarlo en tu proyecto, la cual consiste en integrar las siguientes instrucciones al script de tu sitio web.

```
var tooltipTriggerList =
[].slice.call(document.querySelectorAll('[data-bs-toggle="tooltip"]'))
var tooltipList = tooltipTriggerList.map(function (tooltipTriggerEl) {
  return new bootstrap.Tooltip(tooltipTriggerEl)
})
```

Este código añade un tooltip a cada elemento que tenga el atributo `data-bs-toggle="tooltip"` que en conjunto con los atributos `data-bs-placement` y `title`, se estará definiendo la posición y el título del tooltip, por ejemplo tenemos el siguiente botón HTML.

```
<button type="button" class="btn btn-secondary" data-bs-toggle="tooltip" data-bs-placement="top" title="Tooltip on top">
  Tooltip on top
</button>
```

El cual nos mostraría un tooltip por la parte superior del elemento con el título indicado, tal y como se muestra en la siguiente imagen.

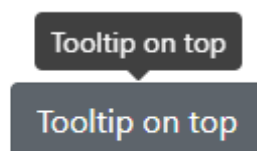


Imagen 5. Ejemplo de Tooltip.

Fuente: [Bootstrap](#).

Dropdown

De acuerdo a la [documentación](#), un dropdown servirá para mostrar un listado desplegable con opciones o links. Existen varias formas de utilizarlo. La más simple es la siguiente:

```
<div class="dropdown">
  <button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
    Dropdown button
  </button>
  <div class="dropdown-menu" aria-labelledby="dropdownMenuButton">
    <a class="dropdown-item" href="#">Action</a>
    <a class="dropdown-item" href="#">Another action</a>
    <a class="dropdown-item" href="#">Something else here</a>
  </div>
</div>
```

Lo que incluirá un dropdown button que mostrará los ítems (Action, Another action, Something else here).

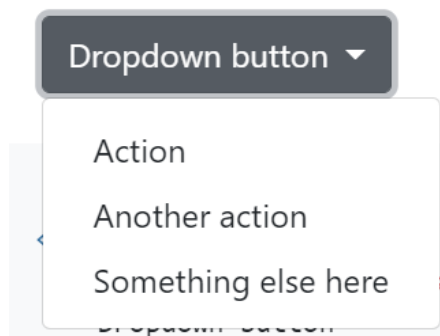


Imagen 6. Funcionamiento de dropdown.
Fuente: [Bootstrap](#).

Carousel de Bootstrap

Un carousel, es un componente de presentación de diapositivas para recorrer los elementos (imágenes o diapositivas de texto). Para hacerlo iremos a la [documentación](#). En el siguiente ejemplo, se muestra un carousel de 3 diapositivas:

```
<div id="carouselExampleSlidesOnly" class="carousel slide"
data-ride="carousel">
  <div class="carousel-inner">
    <div class="carousel-item active">
      
    </div>
    <div class="carousel-item">
      
    </div>
    <div class="carousel-item">
      
    </div>
  </div>
</div>
```

Lo que podemos [observar](#), es una transición de cada una de ellas.

Además, es posible agregarle controles (previo, siguiente) como en este ejemplo:

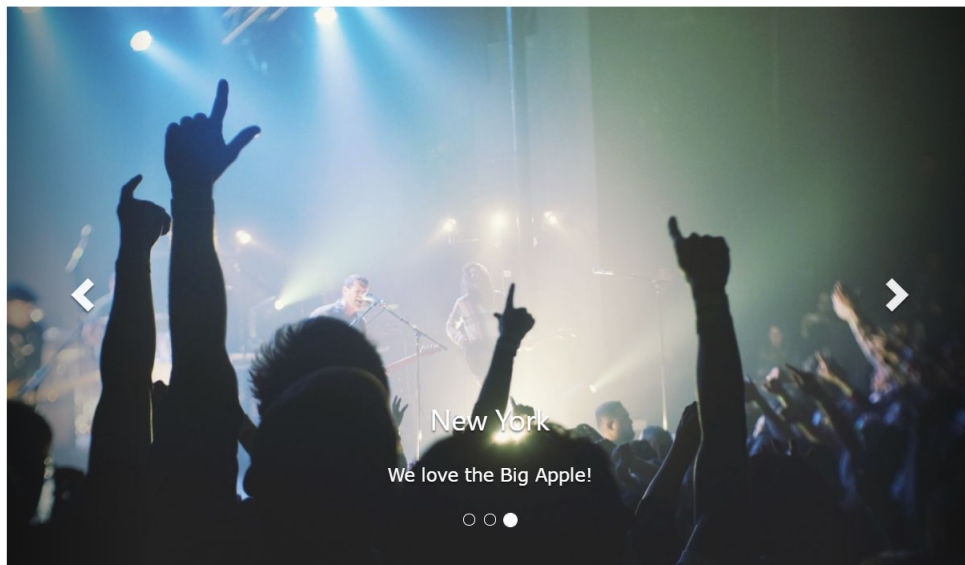


Imagen 7. Carousel Bootstrap.

Fuente: [W3Schools](https://www.w3schools.com/bootstrap/bootstrap_carousel.asp).

Resumen

- Se le denomina Smooth Scroll, al efecto de transición entre diferentes secciones de una misma página sin tener que recargar, lo que mejora la experiencia del usuario y el aspecto visual de nuestro sitio.
- Un popover es un elemento visual que se utiliza para mostrar un mensaje emergente, con una referencia específica a un tema.
- Un modal es otro elemento importante que podemos utilizar gracias a las características de Bootstrap. Sirve para añadir diálogos o mostrar notificaciones.
- Un tooltip es una herramienta aparentemente similar al popover pero que tiene un concepto semántico diferente y solo mostrará una ayuda cuando pasemos el puntero sobre la palabra o botón donde se haya añadido.
- Un dropdown servirá para mostrar un listado desplegable con opciones o links.
- Un carousel es un componente de presentación de diapositivas para recorrer los elementos (imágenes o diapositivas de texto).