

Personalización con Sass en Bootstrap 5.3

Estructura de Archivos Recomendada

Para personalizar Bootstrap sin modificar sus archivos fuente, se sugiere la siguiente estructura:

```
vbnet
CopiarEditar
tu-proyecto/
├── scss/
│   └── custom.scss
├── node_modules/
│   └── bootstrap/
│       ├── js/
│       └── scss/
└── index.html
```

En custom.scss, importarás los archivos Sass de Bootstrap y realizarás tus personalizaciones.

Importación de Bootstrap en tu Archivo Sass

En tu archivo custom.scss, puedes importar Bootstrap de la siguiente manera:

```
scss
CopiarEditar
// Importa las funciones de Bootstrap
@import "node_modules/bootstrap/scss/functions";

// Aquí puedes sobrescribir variables predeterminadas de Bootstrap
$primary: #ff5733;

// Importa el resto de los archivos de Bootstrap
@import "node_modules/bootstrap/scss/bootstrap";
```

Este enfoque te permite sobrescribir variables antes de que se utilicen en los estilos de Bootstrap.

Compilación de Sass

Para compilar tu archivo Sass en CSS, puedes utilizar herramientas como:

Dart Sass: el compilador oficial de Sass.

Webpack, Parcel o Vite: empaquetadores de módulos que pueden manejar la compilación de Sass.

Por ejemplo, usando Dart Sass desde la terminal:

```
bash
CopiarEditar
sass scss/custom.scss css/custom.css
```

Esto generará un archivo CSS que puedes incluir en tu HTML.

Personalización Avanzada con Mapas y Mixins

Bootstrap utiliza **mapas** para definir conjuntos de variables, como colores o tamaños. Puedes modificar estos mapas para personalizar componentes:

```
scss
CopiarEditar
// Agrega un nuevo color al mapa de colores
$theme-colors: map-merge($theme-colors, (
  "nuevo-color": #123456
));
```

Además, Bootstrap proporciona **mixins** para aplicar estilos de manera consistente:

```
scss
CopiarEditar
@include media-breakpoint-up(md) {
  // Estilos para pantallas medianas en adelante
}
```

Opciones Globales

Bootstrap ofrece variables Sass para habilitar o deshabilitar características globales, como:

`$enable-rounded`: controla los bordes redondeados.

`$enable-shadows`: activa o desactiva las sombras.

`$enable-gradients`: permite o no los degradados.

Estas variables se pueden ajustar en tu archivo `custom.scss` antes de importar los archivos principales de Bootstrap.

Modos de Color

Bootstrap 5.3 introduce soporte para modos de color, como claro y oscuro, utilizando la variable `data-bs-theme`. Puedes definir tus propios modos de color personalizados y ajustar variables Sass y CSS según sea necesario.

LAYOUT:

1. Contenedores (.container)

Los contenedores son elementos fundamentales en Bootstrap que sirven para alinear y espaciar el contenido dentro de un dispositivo o ventana de visualización. Existen tres tipos principales:

`.container`: Establece un ancho máximo en cada punto de interrupción (breakpoint) y centra el contenido horizontalmente.

`.container-{breakpoint}`: Tiene un ancho del 100% hasta el punto de interrupción especificado, donde se fija un ancho máximo.

`.container-fluid`: Siempre tiene un ancho del 100%, independientemente del tamaño de la ventana.



2. Sistema de Rejilla (Grid System)

Bootstrap utiliza un sistema de rejilla basado en Flexbox que permite crear diseños responsivos y adaptables. Este sistema se basa en una estructura de 12 columnas y se compone de:

Contenedores (`.container`): Envoltorio general del contenido.

Filas (`.row`): Agrupan columnas y aseguran un alineamiento adecuado.

Columnas (`.col`): Contienen el contenido real y se ajustan según el tamaño de la pantalla.

Las clases de columna se pueden combinar con los puntos de interrupción para definir cómo se comportan en diferentes tamaños de pantalla, por ejemplo: `.col-6`, `.col-md-4`, `.col-lg-3`.



3. Puntos de Interrupción (Breakpoints)

Los puntos de interrupción son anchos de pantalla predefinidos que permiten adaptar el diseño a diferentes dispositivos. Bootstrap define los siguientes:

Extra pequeño (`xs`): $<576\text{px}$ (sin clase infix)

Pequeño (`sm`): $\geq 576\text{px}$

Mediano (`md`): $\geq 768\text{px}$

Grande (`lg`): $\geq 992\text{px}$

Extra grande (`xl`): $\geq 1200\text{px}$

Extra extra grande (`xxl`): $\geq 1400\text{px}$

Estos puntos de interrupción se utilizan en clases como `.col-md-6` o `.d-lg-none` para aplicar estilos específicos según el tamaño de la pantalla.



4. Columnas Avanzadas

Además de las clases básicas de columnas, Bootstrap ofrece opciones avanzadas para:

Alineación vertical y horizontal: Utilizando clases como `.align-items-center` o `.justify-content-end`.

Ordenamiento: Cambiar el orden de las columnas con clases como `.order-1` o `.order-md-2`.

Desplazamiento (Offset): Mover columnas hacia la derecha con clases como `.offset-md-3`.

Estas opciones proporcionan un control más preciso sobre la disposición de las columnas en diferentes tamaños de pantalla.

5. CSS Grid (Experimental)

Bootstrap 5.1 introdujo un sistema de rejilla alternativo basado en CSS Grid, que es experimental y opcional. Para habilitarlo:

1. Desactiva las clases de la rejilla predeterminada estableciendo `$enable-grid-classes: false`.
2. Activa CSS Grid estableciendo `$enable-cssgrid: true`.
3. Recompila tu Sass.

Este sistema utiliza clases como `.grid` y `.g-col-4` y se basa en propiedades de CSS Grid como `grid-template-columns` y `gap`.

6. Utilidades de Diseño

Bootstrap incluye numerosas clases utilitarias para facilitar el diseño y la disposición de elementos:

Visualización (display): Controla cómo se muestra un elemento, por ejemplo, `.d-none` para ocultar o `.d-flex` para aplicar Flexbox.

Flexbox: Clases como `.justify-content-center` o `.align-items-start` para alinear elementos.

Espaciado (margin y padding): Clases como `.m-3` o `.p-2` para ajustar márgenes y rellenos.

Visibilidad: Clases como `.visible` o `.invisible` para mostrar u ocultar elementos sin afectar el diseño.

Estas utilidades permiten aplicar estilos de manera rápida y coherente sin necesidad de escribir CSS personalizado.

CONTENT:

1. Reboot: Normalización de Estilos

Reboot es el conjunto de estilos base de Bootstrap que proporciona una base coherente y moderna para los navegadores. Incluye:

Box-sizing: Se establece en border-box para todos los elementos, lo que facilita el cálculo de anchos y alturas.

Estilos tipográficos básicos: Define estilos predeterminados para encabezados, párrafos, listas, enlaces, entre otros.

Correcciones de inconsistencias: Ajustes para asegurar una apariencia uniforme en diferentes navegadores.



2. Tipografía

Bootstrap ofrece una tipografía limpia y accesible mediante:

Sistema de fuentes: Utiliza una pila de fuentes que prioriza las fuentes del sistema operativo para mejorar el rendimiento y la coherencia visual.

Escala tipográfica: Basada en rem, lo que facilita la adaptabilidad y la accesibilidad.

Clases utilitarias: Para ajustar rápidamente estilos como tamaño (.fs-1 a .fs-6), peso (.fw-bold, .fw-light), alineación (.text-center, .text-end), entre otros.



3. Imágenes

Bootstrap facilita el trabajo con imágenes mediante:

Clases responsivas: .img-fluid para que las imágenes se escalen adecuadamente dentro de su contenedor.

Estilos adicionales: .rounded, .rounded-circle, .img-thumbnail para aplicar bordes redondeados, formas circulares o estilos de miniatura.



4. Tablas

Las tablas en Bootstrap se pueden estilizar y hacer responsivas con:

Clases básicas: .table para aplicar estilos predeterminados.

Variantes: .table-striped, .table-bordered, .table-hover, .table-sm para diferentes estilos visuales.

Responsividad: .table-responsive para envolver la tabla y permitir el desplazamiento horizontal en dispositivos pequeños.



5. Figuras

Para asociar imágenes con leyendas, Bootstrap ofrece:

Contenedor .figure: Agrupa la imagen y su leyenda.

Imagen `.figure-img`: Aplica estilos específicos a la imagen dentro de una figura.

Leyenda `.figure-caption`: Estiliza el texto descriptivo asociado a la imagen.



6. Utilidades de Contenido

Bootstrap proporciona diversas clases utilitarias para mejorar la presentación del contenido:

Listas: `.list-unstyled`, `.list-inline` para listas sin estilos predeterminados o en línea.

Bloques de código: `.pre-scrollable` para permitir el desplazamiento en bloques de código largos.

Textos: `.text-muted`, `.text-primary`, `.text-success`, etc., para aplicar colores contextuales al texto.

FORMS & CONTENT:



Formularios (Forms)

Bootstrap 5.3 ofrece una amplia gama de herramientas para crear formularios accesibles y personalizables. A continuación, se destacan los principales aspectos:

1. Controles de Formulario

Bootstrap proporciona estilos personalizados para elementos como `<input>`, `<select>`, `<textarea>`, y más. Al aplicar clases como `.form-control` o `.form-select`, se mejora la apariencia y funcionalidad de estos elementos, asegurando una experiencia de usuario coherente y moderna. [Bootstrap](#)

2. Grupos de Entrada (Input Groups)

Los grupos de entrada permiten combinar campos de formulario con elementos adicionales, como botones o textos, utilizando la clase `.input-group`. Esto es útil para crear componentes como campos de búsqueda con botones integrados.

3. Etiquetas Flotantes (Floating Labels)

Con las etiquetas flotantes, puedes colocar las etiquetas dentro de los campos de entrada, las cuales se desplazan hacia arriba cuando el usuario comienza a escribir. Esto se logra utilizando la clase `.form-floating`.

4. Diseño de Formularios

Bootstrap facilita la creación de diferentes diseños de formularios: [Bootstrap](#)

Formularios en línea: Utilizando la clase `.row` y clases de columna para alinear elementos horizontalmente.

Formularios horizontales: Aplicando clases como `.row` y `.col-form-label` para alinear etiquetas y campos en una misma línea.

Formularios con cuadrícula personalizada: Combinando el sistema de cuadrícula de Bootstrap con clases de formulario para diseños más complejos.

5. Validación de Formularios

Bootstrap integra estilos para la validación de formularios, permitiendo mostrar mensajes de error o éxito. Puedes utilizar la validación nativa de HTML5 o implementar validaciones personalizadas con JavaScript. [Bootstrap](#)

Componentes (Components)

Bootstrap 5.3 incluye una variedad de componentes reutilizables que facilitan la construcción de interfaces de usuario consistentes y responsivas:

1. Botones (Buttons)

Los botones se estilizan con la clase `.btn` y se pueden personalizar con variantes como `.btn-primary`, `.btn-secondary`, entre otras. También es posible ajustar su tamaño y estado (activo, deshabilitado, etc.).

2. Tarjetas (Cards)

Las tarjetas son contenedores flexibles que pueden incluir títulos, texto, imágenes, enlaces y más. Se construyen utilizando la clase `.card` y sus elementos internos, como `.card-body`, `.card-title`, y `.card-text`. [Bootstrap](#)

3. Listas de Grupo (List Groups)

Las listas de grupo permiten mostrar una serie de contenidos relacionados en una lista con estilo. Se crean con la clase `.list-group` y sus elementos `.list-group-item`. [Bootstrap](#)

4. Alertas (Alerts)

Las alertas proporcionan mensajes de retroalimentación para acciones del usuario. Se implementan con la clase `.alert` y variantes como `.alert-success`, `.alert-danger`, etc.

5. Modales (Modals)

Los modales son ventanas emergentes que se superponen al contenido principal. Se construyen con la clase `.modal` y se controlan mediante JavaScript para mostrar u ocultar según sea necesario.

6. Menús Desplegables (Dropdowns)

Los menús desplegables permiten mostrar una lista de opciones al hacer clic en un elemento. Se crean con la clase `.dropdown` y se personalizan con variantes y opciones de alineación.

7. Sistema de Clases Base y Modificadores

Bootstrap utiliza una nomenclatura de clases base y modificadoras para sus componentes. Por ejemplo, `.btn` es la clase base para botones, mientras que `.btn-primary`

es una variante modificadora. Esto permite una personalización y extensión más sencilla de los estilos.

Principales Componentes

A continuación, se describen algunos de los componentes más utilizados en Bootstrap 5.3:

Botones (.btn): Permiten crear botones con diferentes estilos y tamaños. Se pueden combinar con clases como .btn-primary, .btn-lg, .btn-block, entre otras.

Tarjetas (.card): Contenedores flexibles que pueden incluir encabezados, pies de página, imágenes y contenido variado. Son ideales para mostrar información agrupada de manera atractiva.

Listas de grupo (.list-group): Permiten mostrar una serie de elementos en una lista con estilos consistentes. Se pueden utilizar para menús, listas de tareas, entre otros.

Alertas (.alert): Muestran mensajes de retroalimentación al usuario, como confirmaciones o advertencias. Se pueden personalizar con clases como .alert-success, .alert-danger, etc.

Modales (.modal): Ventanas emergentes que se superponen al contenido principal, útiles para mostrar información adicional o formularios sin abandonar la página actual.

Menús desplegables (.dropdown): Permiten mostrar una lista de opciones al hacer clic en un elemento, como un botón o enlace.

Acordeones (.accordion): Componentes que permiten expandir y contraer secciones de contenido, útiles para mostrar información de manera compacta.

Carruseles (.carousel): Permiten mostrar una serie de imágenes o contenido en un formato de presentación deslizante.

Barras de navegación (.navbar): Componentes de navegación que se adaptan a diferentes tamaños de pantalla y pueden incluir enlaces, formularios y otros elementos interactivos.

Paginación (.pagination): Proporcionan controles para navegar entre páginas de contenido, como listas de artículos o resultados de búsqueda.

Personalización con Sass

Bootstrap 5.3 está construido con Sass, lo que permite una personalización avanzada de los componentes: [bootstrap.p2hp.com+2GitHub+2Bootstrap+2](https://getbootstrap.com/docs/5.3/customize/sass/)

Variables Sass: Se pueden modificar variables predefinidas para cambiar aspectos como colores, tamaños y espacios.

Mixins y funciones: Facilitan la creación de estilos personalizados reutilizables.

Mapas de Sass: Permiten generar variantes de componentes de manera eficiente mediante bucles `@each`.

HELPERS:

Helpers en Bootstrap 5.3

Los *helpers* en Bootstrap 5.3 son clases utilitarias diseñadas para facilitar tareas comunes de diseño y comportamiento en tus proyectos web. Estas clases permiten aplicar estilos específicos de manera rápida y eficiente, sin necesidad de escribir CSS personalizado.

1. Stacks (.vstack y .hstack)

Introducidos en Bootstrap 5.1, los *stacks* son atajos que simplifican la creación de diseños utilizando Flexbox: [Bootstrap](#)

`.vstack`: Apila elementos verticalmente. [Bootstrap+6Bootstrap+6Bootstrap+6](#)

`.hstack`: Coloca elementos horizontalmente. [Bootstrap](#)
[Blog+5Bootstrap+5Bootstrap+5](#)

Ambas clases pueden combinarse con las utilidades de espaciado, como `.gap-*`, para controlar el espacio entre los elementos. [Bootstrap](#)

2. Color y Fondo

Estas clases combinan las utilidades de texto (`.text-*`) y fondo (`.bg-*`) para establecer combinaciones de colores con contraste adecuado. Utilizan la función `color-contrast()` de Sass para garantizar la legibilidad del texto sobre diferentes fondos.

[Bootstrap+1Bootstrap+1](#)

3. Posicionamiento

Bootstrap ofrece clases para posicionar elementos de manera fija o adhesiva:

`.fixed-top` y `.fixed-bottom`: Fijan un elemento en la parte superior o inferior de la ventana.

`.sticky-top` y `.sticky-bottom`: Hacen que un elemento se adhiera al borde correspondiente al hacer scroll.

También existen variantes responsivas como `.sticky-sm-top` para aplicar estos comportamientos en diferentes tamaños de pantalla. [Bootstrap](#)

4. Ratios

Las clases `.ratio` permiten mantener una relación de aspecto específica para elementos como videos o imágenes. Las relaciones predefinidas incluyen:

`.ratio-1x1`

`.ratio-4x3`

`.ratio-16x9`

`.ratio-21x9`

También es posible definir relaciones personalizadas utilizando variables CSS.

[Bootstrap+3Bootstrap+3Bootstrap+3](#)

5. API de Utilidades Personalizadas

Bootstrap 5.3 introduce una API que permite crear utilidades personalizadas mediante Sass. Algunas características incluyen:

Variables CSS: Genera variables locales en lugar de reglas tradicionales de propiedad-valor. [Bootstrap+1Bootstrap+1](#)

Clases Personalizadas: Permite cambiar el prefijo de las clases generadas.

Estados: Genera variantes para pseudo-clases como :hover o :focus. [Bootstrap](#)

Esta flexibilidad facilita la creación de estilos reutilizables y adaptados a las necesidades específicas de tu proyecto.

UTILITIES:



Utilidades en Bootstrap 5.3

Bootstrap 5.3 ofrece una amplia gama de clases utilitarias que permiten aplicar estilos específicos de manera rápida y eficiente, sin necesidad de escribir CSS personalizado. Estas utilidades están diseñadas para ser responsivas y facilitar el desarrollo de interfaces modernas y adaptables.



Espaciado: Márgenes y Relleno

Las clases de espaciado permiten controlar los márgenes (m-) y el relleno (p-) de los elementos. Se utilizan combinaciones como:

.m-3: Aplica un margen de 1rem en todos los lados. [elchininet.github.io+1Bootstrap+1](#)

.pt-2: Aplica un relleno superior de 0.5rem.

Estas clases también admiten variantes responsivas, como .mb-md-4, que aplica un margen inferior de 1.5rem a partir del punto de interrupción md. [elchininet.github.io](#)



Posicionamiento y Visualización

Bootstrap proporciona clases para controlar la propiedad display y la visibilidad de los elementos: [Bootstrap+3getbootstrap.cn+3elchininet.github.io+3](#)

.d-none: Oculta un elemento.

.d-flex: Convierte un elemento en un contenedor flex. [elchininet.github.io+2getbootstrap.cn+2Bootstrap+2](#)

.visible / .invisible: Controlan la visibilidad sin afectar el flujo del documento. [Bootstrap](#)

Estas clases también tienen variantes responsivas, como .d-sm-block, que muestra el elemento como bloque a partir del punto de interrupción sm.

Colores y Fondos

Las utilidades de color permiten aplicar estilos de texto y fondo:

.text-primary: Aplica el color primario al texto. [AlmaBetter+2Bootstrap+2Bootstrap+2](#)

.bg-success: Aplica un fondo de color verde indicando éxito.

Estas clases ayudan a mantener una paleta de colores coherente en toda la aplicación.

Tipografía

Bootstrap incluye clases para modificar la apariencia del texto: [getbootstrap.cn](#)

.text-center: Centra el texto. [Bootstrap+6elchininet.github.io+6Bastaki Software Solutions L.L.C-FZ+6](#)

.fw-bold: Aplica negrita al texto. [Bootstrap+9Bootstrap+9getbootstrap.cn+9](#)

.fst-italic: Aplica cursiva al texto.

.text-uppercase: Convierte el texto a mayúsculas.

Estas utilidades facilitan la personalización de la tipografía sin necesidad de escribir CSS adicional.

Flexbox

Bootstrap aprovecha Flexbox para crear diseños flexibles y responsivos. Algunas clases útiles incluyen:

.d-flex: Establece un contenedor flex.

.justify-content-center: Centra los elementos horizontalmente.

.align-items-start: Alinea los elementos al inicio verticalmente.

.flex-column: Organiza los elementos en una columna.

Estas clases permiten construir diseños complejos de manera sencilla.

API de Utilidades Personalizadas

Bootstrap 5.3 introduce una API basada en Sass que permite crear utilidades personalizadas:

Variables CSS: Se pueden generar variables locales en lugar de reglas tradicionales de propiedad-valor.

Clases Personalizadas: Es posible cambiar el prefijo de las clases generadas.

Estados: Se pueden generar variantes para pseudo-clases como :hover o :focus.