# **Diseñando con Rejilla**

Bootstrap incluye una rejilla fluida pensada para móviles (Mobile first) y que cumple con el diseño web responsivo. Esta rejilla crece hasta 12 columnas a medida que crece el tamaño de la pantalla del dispositivo.

Mobile-first significa que se debe comenzar realizando el diseño para móviles (puesto que es la pantalla mas limitada o restrictiva) y luego ir pensando en el diseño para pantallas más grandes.

El sistema de rejilla de Bootstrap se basa en la creación o disposición del contenido de nuestra web de rejillas flexibles, las cuales se escalarán al tamaño y posición adecuada de forma automática dependiendo del tamaño de la pantalla.

**Elemento Contenedor**

El sistema de rejilla tiene que ser utilizado dentro de uno de los dos elementos contenedores que provee Bootstrap: Container o container-fluid. Es importante tener en cuenta que estos elementos se utilizan como raíz de la rejilla y no se podrán anidar uno dentro de otros.

**Funcionamiento del sistema de rejilla**

El sistema de rejilla está pensando para ayudarnos en la disposición de los contenidos de nuestra web y su adaptación a los diferentes tamaños de pantalla de forma automática. Para ellos tenemos que poner el contenido dentro de celdas o columnas que irán dentro de filas. Cada fila se puede dividir hasta en 12 columnas, pero seremos nosotros los que definiremos el número de columnas deseando para cada tamaño de pantalla.

**Detalles del funcionamiento de este sistema:**

* Las columnas irán agrupadas dentro de filas (.row). Las filas (.row) se deben colocar dentro de una etiqueta contenedora: .container (para ancho fijo) o .container-fluid (para poder ocupar todo el ancho), esto permitirá alinear las celdas y asignarles el espaciado correcto.
* El contenido se debe disponer dentro de columnas o celdas, las cuales deben de ser el único hijo posible de las filas (.row), las cuales, a su vez, serán el único hijo posible del contenedor (.container o .container-fluid).
* Al seguir este orden el sistema de rejilla funcionará correctamente, creando el espaciado interior y los márgenes apropiados dependiendo de las dimensiones de la pantalla.
* Cada fila se puede dividir hasta un máximo de 12 columnas, pero somos nosotros los que tendremos que definir el número de columnas en el que queremos dividir cada fila y su ancho para cada tamaño de pantalla. Por ejemplo: 3 columnas de igual ancho.
* Si el tamaño total de las columnas de una fila excede de 12 el tamaño sobrante se colocará en la siguiente fila.
* El tamaño de las columnas se especificará con clases css que Bootstrap define para cada tamaño de pantalla, por ejemplo. col-md-XX, donde XX es el tamaño de la columna, que podrá tomar valores entre 1 y 12.

**Tipos de rejillas**

Bootstrap utiliza un sistema de rejilla basado en 12 columnas que se adaptan a diferentes tamaños de pantalla

1. **Rejilla Básica:**

Utiliza. col y. row.

col para dividir el espacio en partes iguales entre las columnas de una fila .row

<div class="container">

<div class="row">

<div class="col">

Column

</div>

<div class="col">

Column

</div>

<div class="col">

Column

</div>

</div>

</div>

1. **Rejilla estableciendo el ancho de columna**

El diseño automático de las columnas de cuadrícula flexbox también significa que puede establecer el ancho de una columna y hacer que las columnas hermanas cambien de tamaño automáticamente a su alrededor. Puede usar clases de cuadrícula predefinidas (como se muestra a continuación), mixins de cuadrícula o anchos en línea.

<div class="container">

<div class="row">

<div class="col">columna 1 </div>

<div class="col-6">columna 2 </div>

<div class="col">columna 3 </div>

</div>

<div class="row">

<div class="col">columna 4</div>

<div class="col-5">columna5 </div>

<div class="col"> columna 6 </div>

</div>

</div>

1. **Rejillas con opciones de cuadricula o rejillas responsivas por tamaño de pantalla**

El sistema de cuadricula de Bootstrap pueden adaptarse a los seis puntos de interrupción que personalice. Los seis niveles de cuadricula predeterminados son los siguientes:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Extra pequeño (xs) | Pequeño  (sm) | Medio  (md) | Grande  (Ig) | Extra grande (xl) | Extra extra grande  (xxl) |
| Contenedor | Ninguno (auto) | 540px | 720px | 960px | 1140px | 1320px |
| Prefijo de clase | . col- | . col-sm- | . col-md- | . col-lg | . col-xl- | . col-xxl- |

1. **Rejillas de contenido de ancho variable**

Usar col-{breakpoint}-auto clases a columnas de tamaño basadas en el ancho natural de su contenido. Si se desea transformar una rejilla de anchura fija en una rejilla de anchura variable que ocupa toda la anchura del navegador, se debe reemplazar la clase CSS .container por .container-fluid en el elemento que encierra a todos los demás elementos de la rejilla:

<div class="container-fluid">

<div class="row">

...

</div>

</div>

1. **Rejillas de desplazamiento de columnas**

La clase **.col-md-offset-\*** permite desplazar cualquier columna hacia su derecha. Estas clases aumentan el tamaño del margen izquierdo de la columna en una cantidad equivalente a esas columnas

Ejemplo:

La clase .col-md-offset-4 desplaza la columna a una anchura equivalente a 4 columnas

1. **Rejillas Anidadas o Anidando columnas**

Para anidar su contenido con la cuadrícula predeterminada, agregue una nueva .row y conjunto de .col-sm-\* columnas dentro de un existente .col-sm-\* columna. Las filas anidadas deben incluir un conjunto de columnas que sumen hasta 12 o menos.

<div class="container">

<div class="row">

<div class="col-sm-3">

Level 1: .col-sm-3

</div>

<div class="col-sm-9">

<div class="row">

<div class="col-8 col-sm-6">

Level 2: .col-8 .col-sm-6

</div>

<div class="col-4 col-sm-6">

Level 2: .col-4 .col-sm-6

</div>

</div>

</div>

</div>

</div>

1. **Rejilla de columnas con diseño automático**

Utilice clases de columna específicas de punto de interrupción para un fácil tamaño de columna sin una clase numerada explícita como .col-sm-6.

1. **Rejilla de ancho igual**

Se agrega cualquier numero de clases sin unidades para cada punto de interrupción que necesite y cada columna tendrá el mismo ancho.

Ejemplo

<div class="container">

<div class="row">

<div class="col">

1 of 2

</div>

<div class="col">

2 of 2

</div>

</div>

<div class="row">

<div class="col">

1 of 3

</div>

<div class="col">

2 of 3

</div>

<div class="col">

3 of 3

</div>

</div>

</div>

**Imágenes Responsivas**

Para asegurar que las imágenes dentro de una web sean adaptables y se ajusten correctamente a diversas resoluciones de pantalla, Bootstrap proporciona la clase .img-fluid. Aplicar esta clase a una imagen le ofrece max-width:100% y height\_auto, lo que permite que la imagen escale proporcionalmente al ancho del contenedor padre, evitando que sobrepase sus límites. Para aplicar esta clase, simplemente incluye .img-fluid en el elemento <img>. Es decir que las imágenes en Bootstrap se hacen responsiva con .img-fluid y se aplica max-width:100% y height: auto; a la imagen para que escale con el ancho de su elemento padre.

Este ajuste sencillo garantiza que las imágenes dentro del diseño se mantengan siempre dentro de los limites del contenedor padre, mejorando la experiencia del usuario en dispositivos con diferentes tamaños de pantalla y resoluciones. Es especialmente útil en diseños responsivos, asegurando que la visualización de las imágenes sea eficiente y optima en cualquier dispositivo, desde móviles hasta desktops.

La .img-fluid clase hace que una imagen sea responsiva al aplicarle automáticamente " max-width: 100%;y height: auto;". Como resultado:

* La imagen se escala con el ancho del elemento principal.
* El navegador no hace que la imagen sea más grande que su contenedor.
* La imagen no crece más allá de su tamaño original, lo que puede reducir la calidad.
* La imagen se reduce si la ventana gráfica se vuelve más angosta que el ancho especificado.

Para configurar la .img-fluid clase en tus imágenes, simplemente define la <img>etiqueta de esta manera: **<img src="..." class="img-fluid" alt=”…”>**

**Formularios Responsivos en Bootstrap**

Los controles de formulario de Bootstrap amplia nuestros estilos de formulario reiniciados con clases. Úsalos para personalizar sus visualizaciones y lograr una representación más consistente en todos los navegadores y dispositivos.

Asegúrese de utilizar un type atributo apropiado en todas las entradas, por ejemplo, email para la dirección de correo electrónico o number para la información numérica para aprovechar los controles de entrada mas nuevos como la verificación de correo electrónico, la selección de números y más

Bootstrap aplica estilos a los elementos de tipo formulario para convertirlos en elementos responsivos, mejorar su apariencia y permitirnos crear diferentes alineaciones. La estructura básica de un formulario es la siguiente:

<form>

<div class="form-group">

<label for="exampleInputEmail1">Email address</label>

<input type="email" class="form-control" id="exampleInputEmail1"

aria-describedby="emailHelp" placeholder="Enter email">

</div>

</form>

Controles de formulario

Los controles de formulario textual como <input>s, <select>s y <textarea>s, se estilizan con la. form-controlclase. Incluye estilos para la apariencia general, el estado del foco, el tamaño y más.

Formulario inline

Mediante la utilización de la clase .form-inline sobre la etiqueta <form> podemos crear formularios que se dispondrán en una sola línea.

Formulario horizontal

Mediante el uso del sistema de rejilla de Bootstrap podemos crear formularios con disposición en horizontal. Para esto tendremos que crear una fila (.row) por cada grupo, y situar la etiqueta y el input cada uno en una columna.

Estados de validación de un formulario

Bootstrap también incluye clases para aplicar diferentes estados de validación a un formulario. Para utilizarlo simplemente tenemos que añadir las clases: .is-valid o .is-invalid sobre el propio input. De esta forma, el color de los elementos del formulario cambiará.