



## **Facultad de ingeniería**

Técnico en Ingeniería en Ciencias de la Computación

### **“Investigación II (Bootstrap) ”**

#### **Estudiantes:**

- Katherine Valeria Renderos Romeo RR250127
- Gabriela Stephani Figueroa Calderón FC250109

**Docente:** Ing. Edwin Bonilla

**Asignatura:** Lenguaje de Marcado y Estilo Web

**Fecha:** 09/05/2025

# Índice

<b>Introducción</b> .....	4
<b>2. Marco Teórico</b> .....	5
2.1 ¿Qué es Bootstrap? .....	5
2.2 Diseño Responsivo .....	5
2.2.1 Importancia del Uso de Bootstrap .....	5
2.3 Sistema de Rejilla (Grid System) .....	5
2.3.1 Tipos de Rejillas .....	6
2.4 Contenedores.....	6
2.5 Imágenes Responsivas.....	6
2.6 Tablas Responsivas.....	6
2.7 Formularios Responsivos.....	7
<b>3. Definición y conceptos básicos de Bootstrap</b> .....	7
3.1 ¿Qué es Bootstrap? .....	7
3.2 Conceptos Básicos.....	7
3.3 Características Fundamentales de Bootstrap.....	7
Uso de Elementos de Bootstrap .....	8
3.4 Ventajas y Desventajas de Bootstrap.....	8
3.4.1 Ventajas .....	8
3.4.2 Desventajas .....	8
<b>4. Diseñando con rejilla</b> .....	9
4.1 Elemento Contenedor.....	9
4.2 Funcionamiento del sistema de rejilla .....	9
4.3 Detalles del funcionamiento de este sistema: .....	10
<b>5. Tipos de rejillas</b> .....	11
5.1 Rejilla Básica: .....	11
5.2 Rejilla estableciendo el ancho de columna.....	12
5.3 Rejillas con opciones de cuadrícula o rejillas responsivas por tamaño de pantalla.....	12
5.4 Rejillas de contenido de ancho variable.....	13
5.5 Rejillas de desplazamiento de columnas .....	14
5.6 Rejillas Anidadas o Anidando columnas .....	14

5.7	Rejilla de columnas con diseño automático .....	15
5.8	Rejilla de ancho igual .....	15
<b>6.</b>	<b>Manejo de filas y columnas .....</b>	<b>17</b>
6.1	Breakpoints (Responsive) .....	17
6.2	Tipos de Columnas.....	18
<b>7.</b>	<b>Tipos de contenedores.....</b>	<b>20</b>
7.1	Tipos de contenedores .....	20
7.1.1	Tabla comparativa de anchos máximos (max-width).....	20
7.2	Contenedor predeterminado.....	21
7.3	Contenedores responsivos.....	21
7.4	Contenedores fluidos.....	21
7.5	Personalización con Sass.....	22
<b>8.</b>	<b>Imágenes responsivas .....</b>	<b>22</b>
<b>9.</b>	<b>Tablas responsivas .....</b>	<b>23</b>
8.1	Accesibilidad de Bootstrap Responsive.....	24
8.2	Como usar el Responsive .....	24
8.3	Responsivas por Breakpoint.....	25
<b>9.</b>	<b>Formulario responsivo .....</b>	<b>26</b>
9.1	Controles de formulario .....	27
9.2	Formulario inline.....	27
9.3	Formulario horizontal .....	27
9.4	Estados de validación de un formulario .....	28
<b>10.</b>	<b>Repositorio de Git Hub.....</b>	<b>28</b>
<b>11.</b>	<b>Ejemplos de Bootstrap (Capturas) .....</b>	<b>28</b>
<b>12.</b>	<b>Conclusión .....</b>	<b>29</b>
<b>13.</b>	<b>Bibliografía.....</b>	<b>30</b>

## Introducción

En la actualidad, el desarrollo web y la creación de sitios web responsivos son fundamentales. Bootstrap, es un framework de código abierto desarrollado por Twitter, y se ha convertido en uno de los más populares y ampliamente utilizados, debido al conjunto de funcionalidades y herramientas que posee basado en HTML, CSS y JavaScript. Este framework Permite la creación de interfaces limpias y funcionales; además una de las características fundamentales de este framework es el diseño responsivo es decir se adaptan automáticamente a diferentes tamaños de pantalla.

Bootstrap posee un sistema de cuadrícula flexible y una amplia variedad de herramientas y componentes como menús, formularios, botones, tablas entre otros. Es decir, no solo facilita la creación de diseños responsivos que se adapta a diferentes tamaños de pantalla si no que también permite personalizar de manera rápida y fácil nuestro sitio web o interface, debido a que Bootstrap incluye una alta gama de componentes listos para usar Esta investigación aborda los conceptos fundamentales necesarios para poder iniciarse, conocer y aprender sobre Bootstrap. Abarca conceptos como el sistema de rejillas que se basa en 12 columnas además encontramos los diferentes tipos de rejillas, así como el manejo adecuado de columnas, filas, formularios responsivos, tablas e imágenes. Es importante conocer y comprender estos elementos para aprovechar al máximo las funcionalidades de Bootstrap.

## **2. Marco Teórico**

### **2.1 ¿Qué es Bootstrap?**

Bootstrap es un framework de código abierto para el desarrollo de interfaces web que facilita la creación de sitios y aplicaciones responsivas. Su diseño se basa en HTML, CSS y JavaScript, y proporciona una serie de componentes prediseñados que permiten a los desarrolladores crear páginas web modernas y funcionales de manera rápida y eficiente.

### **2.2 Diseño Responsivo**

El diseño responsivo es una característica fundamental de Bootstrap. Esto significa que las páginas creadas con este framework se adaptan automáticamente a diferentes tamaños de pantalla, desde dispositivos móviles hasta pantallas de escritorio. Esta flexibilidad se logra gracias a su sistema de rejilla fluida (Grid System).

#### **2.2.1 Importancia del Uso de Bootstrap**

Bootstrap permite acelerar el desarrollo web, garantizar un diseño coherente y adaptativo, y simplificar el mantenimiento del código gracias a su estructura modular y reutilizable.

### **2.3 Sistema de Rejilla (Grid System)**

El sistema de rejilla de Bootstrap se basa en un esquema de 12 columnas que permiten organizar el contenido de manera flexible. Los elementos se distribuyen en filas (.row) y columnas (.col), las cuales se adaptan al tamaño de la pantalla.

### 2.3.1 Tipos de Rejillas

- **Rejilla Básica:** Utiliza filas y columnas que se distribuyen equitativamente en el espacio disponible.
- **Rejilla con Ancho Definido:** Las columnas pueden tener un ancho específico, definido mediante clases preestablecidas como `.col-6` (50% del ancho).
- **Rejilla Responsive:** Las columnas se ajustan automáticamente al tamaño del dispositivo, utilizando prefijos como `.col-sm-`, `.col-md-`, `.col-lg-` y `.col-xl-`.

## 2.4 Contenedores

Los contenedores son la base del sistema de rejilla en Bootstrap y permiten alinear y organizar el contenido de manera controlada. Los principales tipos son:

- `.container`: Contenedor de ancho fijo, que se adapta a diferentes puntos de interrupción (breakpoints).
- `.container-fluid`: Contenedor de ancho completo, que abarca el 100% del ancho disponible.

## 2.5 Imágenes Responsivas

Bootstrap permite que las imágenes se adapten automáticamente al tamaño del contenedor padre gracias a la clase `.img-fluid`, que aplica las propiedades `max-width: 100%` y `height: auto`, garantizando que las imágenes no se desborden ni pierdan proporción.

## 2.6 Tablas Responsivas

Las tablas en Bootstrap pueden volverse responsivas envolviéndolas en un contenedor `.table-responsive`, permitiendo que se desplacen horizontalmente en pantallas pequeñas.

## 2.7 Formularios Responsivos

Bootstrap facilita la creación de formularios adaptativos utilizando clases como `.form-control` para entradas de texto, `.form-group` para agrupar elementos, y opciones avanzadas como formularios en línea (`.form-inline`) o formularios horizontales (usando el sistema de rejilla).

## 3. Definición y conceptos básicos de Bootstrap

### 3.1 ¿Qué es Bootstrap?

Bootstrap es un framework de código abierto que permite desarrollar sitios web y aplicaciones móviles de forma rápida y eficiente. Está basado en HTML, CSS y JavaScript, y proporciona una colección de componentes prediseñados y herramientas listas para usar.

### 3.2 Conceptos Básicos

- **Framework:** Conjunto de herramientas y recursos que facilitan el desarrollo de aplicaciones.
- **Responsive Design:** Diseño adaptable a diferentes tamaños de pantalla (móviles, tablets, PCs).
- **Componentes:** Elementos ya diseñados como botones, menús, formularios, etc.
- **Grid System:** Sistema de cuadrícula de 12 columnas para estructurar el diseño de forma flexible.

### 3.3 Características Fundamentales de Bootstrap

- **Flexibilidad:** Permite personalizar diseños con poco código.
- **Responsive Design:** Se adapta automáticamente a cualquier dispositivo.
- **Grid System:** Organización visual mediante filas y columnas.
- **Componentes Prediseñados:** Botones, menús, navegación, formularios, tablas, etc.

- **Plugins JavaScript:** Carruseles, ventanas modales, tooltips, entre otros.

### Uso de Elementos de Bootstrap

- Fáciles de usar y personalizar.
- Gratuitos y disponibles para modificar según necesidades.
- Compatibles con navegadores modernos.
- Interactividad asegurada mediante integración con JavaScript.
- Versátiles para proyectos grandes o pequeños.

## 3.4 Ventajas y Desventajas de Bootstrap

### 3.4.1 Ventajas

- Basado en tecnologías comunes: HTML, CSS, JS.
- Diseño adaptable a todo tipo de dispositivos.
- Rico en funciones: Menos necesidad de escribir código desde cero.
- Apoyo y actualizaciones constantes de la comunidad.
- Gratis y de código abierto.

### 3.4.2 Desventajas

- Limitada personalización en comparación con diseño desde cero.
- Problemas de compatibilidad con navegadores antiguos.
- Riesgo de dependencia excesiva, lo que puede afectar la escalabilidad y mantenimiento del sitio.



## 4. 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 más 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.

### 4.1 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.

### 4.2 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.

#### 4.3 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.

## 5. Tipos de rejillas

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

### 5.1 Rejilla Básica:

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

#### Ejemplo

```
<div class="container">
```

```
<div class="row">
```

```
<div class="col">
```

```
Column
```

```
</div>
```

```
<div class="col">
```

```
Column
```

```
</div>
```

```
<div class="col">
```

```
Column
```

```
</div>
```

```
</div>
```

```
</div>
```

## 5.2 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.

### Ejemplo

```
<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>
```

## 5.3 Rejillas con opciones de cuadrícula o rejillas responsivas por tamaño de pantalla

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

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

## 5.4 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:

### Ejemplo

```
<div class="container-fluid">
```

```
<div class="row">
```

```
...
```

```
</div>
```

```
</div>
```

## 5.5 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

## 5.6 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.

### Ejemplo

<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>

## 5.7 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.

## 5.8 Rejilla de ancho igual

Se agrega cualquier número 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>



## 6. Manejo de filas y columnas

**Contenedor (.container):** Centraliza y proporciona un ancho máximo al contenido.

.container: ancho fijo dependiendo del breakpoint.

.container-fluid: ocupa el 100% del ancho.

- **Fila (.row):** Contiene columnas. Añade márgenes negativos y alinea contenido horizontalmente.
- **Columna (.col):** Elemento flexible dentro de .row. Bootstrap divide el espacio horizontal en **12 columnas** por fila.

### 6.1 Breakpoints (Responsive)

Usa prefijos para ajustar el diseño en diferentes tamaños de pantalla:

Breakpoint	Prefijo de Clase	Min-width
<b>xs</b>	.col-	0px
<b>sm</b>	.col-sm-	576px
<b>md</b>	.col-md-	768px
<b>lg</b>	.col-lg-	992px
<b>xl</b>	.col-xl-	1200px
<b>xxl</b>	.col-xxl-	1400px

## 6.2 Tipos de Columnas

- `.col`: Auto-layout, divide el espacio restante en partes iguales.
- `.col-{n}`: Ocupa **n de 12 columnas**.
- `.col-{breakpoint}-{n}`: Responsive con tamaño específico.
- `.col-auto`: Se ajusta al contenido.

### Ejemplos de uso

*html*

*<!-- Tres columnas iguales -->*

*<div class="container">*

*<div class="row">*

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

*<div class="col">2</div>*

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

*</div>*

*</div>*

-----

-----

*<!-- Distribución responsive -->*

*<div class="row">*

*<div class="col-sm-6 col-md-4">A</div>*

*<div class="col-sm-6 col-md-8">B</div>*

*</div>*

## Clases útiles

- `.row-cols-{n}`: Fuerza cantidad de columnas por fila.
- `.gx-*` y `.gy-*`: Controlan el **gutter** horizontal y vertical.
- `.g-0`: Sin gutters.
- `.justify-content-*`, `.align-items-*`: Controlan alineación en flexbox.

**Anidamiento:** Puedes tener `.row` dentro de una `.col` para crear subgrillas.

Sass (opcional para personalización avanzada)

## Variables:

`$grid-columns: 12;`

`$grid-gutter-width: 1.5rem;`

Mixins:

`// Crea un contenedor para una serie de columnas`

`@include make-row();`

## 7. Tipos de contenedores

Los **contenedores** son elementos esenciales en Bootstrap. Su función principal es **encerrar, rellenar y alinear el contenido** dentro de un área visible del dispositivo (viewport). Estos elementos son imprescindibles cuando se utiliza el sistema de cuadrícula de Bootstrap.

### 7.1 Tipos de contenedores

Bootstrap ofrece tres tipos principales de contenedores:

- `.container`: define un **ancho máximo (max-width)** que varía según el tamaño del viewport.
- `.container-fluid`: ocupa siempre el **100% del ancho disponible**, en todos los tamaños de pantalla.
- `.container-{breakpoint}`: tiene **ancho completo** hasta alcanzar un tamaño de pantalla específico (breakpoint), y luego adopta un **max-width** como `.container`.

#### 7.1.1 Tabla comparativa de anchos máximos (max-width)

	Extra small <576px	Small ≥576px	Medium ≥768px	Large ≥992px	X-Large ≥1200px	XX-Large ≥1400px
<code>.container</code>	100%	540px	720px	960px	1140px	1320px
<code>.container-sm</code>	100%	540px	720px	960px	1140px	1320px
<code>.container-md</code>	100%	100%	720px	960px	1140px	1320px
<code>.container-lg</code>	100%	100%	100%	960px	1140px	1320px
<code>.container-xl</code>	100%	100%	100%	100%	1140px	1320px
<code>.container-xxl</code>	100%	100%	100%	100%	100%	1320px
<code>.container-fluid</code>	100%	100%	100%	100%	100%	100%

## 7.2 Contenedor predeterminado

El contenedor básico (.container) tiene un **ancho fijo y sensible** que se adapta a cada breakpoint.

html

<div class="container">

<!-- Tu contenido aquí -->

</div>

## 7.3 Contenedores responsivos

Los **contenedores responsivos** permiten usar clases que tienen **100% de ancho hasta cierto breakpoint**, momento en el cual se aplican valores de max-width específicos para pantallas más grandes.

html

<div class="container-sm"> 100% de ancho hasta el breakpoint "sm"</div>

<div class="container-md"> 100% de ancho hasta "md"</div>

<div class="container-lg"> 100% de ancho hasta "lg"</div>

<div class="container-xl"> 100% de ancho hasta "xl"</div>

<div class="container-xxl"> 100% de ancho hasta "xxl"</div>

## 7.4 Contenedores fluidos

Utiliza .container-fluid cuando necesites que el contenido **abarque el ancho completo del viewport**, sin importar el tamaño de la pantalla.

html

```
<div class="container-fluid">  
<!-- Contenido de ancho completo -->  
</div>
```

## 7.5 Personalización con Sass

Bootstrap personalizamos los tamaños máximos de los contenedores a través del mapa \$container-max-widths, definido en \_variables.scss:

```
$container-max-widths: (  
  sm: 540px,  
  md: 720px,  
  lg: 960px,  
  xl: 1140px,  
  xxl: 1320px );  
  
// Definición del mixin  
@mixin make-container($padding-x: $container-padding-x) {  
  width: 100%;  
  padding-right: $padding-x;  
  padding-left: $padding-x;  
  margin-right: auto;  
  margin-left: auto; }
```

## 8. 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 límites 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: **``**

## 9. Tablas responsivas

Las tablas responsivas permiten que las tablas se desplacen horizontalmente, mejorando la visualización en dispositivos con pantallas pequeñas. Bootstrap ofrece varias opciones para hacer que las tablas sean responsivas.

## 8.1 Accesibilidad de Bootstrap Responsive

Las tablas responsivas utilizan overflow-x para desplazarse. Asegúrate de que los lectores de pantalla puedan interpretar correctamente el contenido y que la tabla sea comprensible sin desplazarse.

## 8.2 Como usar el Responsive

Usa `.table-responsive` para permitir que las tablas se desplacen horizontalmente en cualquier tamaño de pantalla.

```
<div class="table-responsive">
  <table class="table">
    <thead>
      <tr>
        <th>#</th>
        <th>Nombre</th>
        <th>Apellido</th>
        <th>Handle</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <th scope="row">1</th>
        <td>Mark</td>
        <td>Otto</td>
        <td>@mdo</td>
      </tr>
      <tr>
        <th scope="row">2</th>
```



```

<td>Jacob</td>
<td>Thornton</td>
<td>@fat</td>
</tr>
</tbody>
</table>
</div>

```

### 8.3 Responsivas por Breakpoint

Bootstrap permite crear tablas responsivas solo para ciertos tamaños de pantalla usando clases específicas:

- *.table-responsive-sm*: Hasta pantallas pequeñas (576px).
- *.table-responsive-md*: Hasta pantallas medianas (768px).
- *.table-responsive-lg*: Hasta pantallas grandes (992px).
- *.table-responsive-xl*: Hasta pantallas extra grandes (1200px).
- *.table-responsive-xxl*: Hasta pantallas extra extra grandes (1400px).

#### **Ejemplo:**

```

<div class="table-responsive-sm">
<table class="table">
<thead>
<tr>
<th>#</th>
<th>Nombre</th>
<th>Apellido</th>
<th>Handle</th>
</tr>

```

```

    </thead>
    <tbody>
      <tr>
        <th scope="row">1</th>
        <td>Mark</td>
        <td>Otto</td>
        <td>@mdo</td>
      </tr>
      <tr>
        <th scope="row">2</th>
        <td>Jacob</td>
        <td>Thornton</td>
        <td>@fat</td>
      </tr>
    </tbody>
  </table>
</div>

```

## 9. Formulario responsivo

Los controles de formulario de Bootstrap amplía 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 más 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>
```

## 9.1 Controles de formulario

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

## 9.2 Formulario inline

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

## 9.3 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.

## 9.4 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.

## 10. Repositorio de Git Hub

[https://github.com/kvrenderos/InvestigacionII\\_LME.git](https://github.com/kvrenderos/InvestigacionII_LME.git)

## 11. Ejemplos de Bootstrap (Capturas)

[https://udbedu-my.sharepoint.com/:w:/r/personal/rr250127\\_alumno\\_udb\\_edu\\_sv/\\_layouts/15/Doc.aspx?sourcedoc=%7B3D5D9751-8EC2-46AB-8713-349F1B55DAF5%7D&file=cap\\_ejemplos.docx&action=default&mobileredirect=true](https://udbedu-my.sharepoint.com/:w:/r/personal/rr250127_alumno_udb_edu_sv/_layouts/15/Doc.aspx?sourcedoc=%7B3D5D9751-8EC2-46AB-8713-349F1B55DAF5%7D&file=cap_ejemplos.docx&action=default&mobileredirect=true)

## 12. Conclusión

En conclusión, Bootstrap se ha consolidado como uno de los frameworks más utilizados para el desarrollo de interfaces web responsivas debido a su flexibilidad, simplicidad y capacidad de adaptación a múltiples dispositivos. A través de su sistema de rejillas, permite a los desarrolladores estructurar páginas de manera coherente y eficiente, ajustándose automáticamente a diferentes resoluciones. Además, el manejo de filas, columnas y contenedores facilita la organización del contenido, garantizando una distribución adecuada en cualquier pantalla. El uso de imágenes, tablas y formularios responsivos asegura que los elementos visuales y funcionales se presenten correctamente sin importar el tamaño del dispositivo. En resumen, Bootstrap es una herramienta fundamental para la creación de sitios web modernos, accesibles y adaptativos.

## 13. Bibliografía

Marujita. (2023, mayo 14). *Bootstrap*. Muy Tecnológicos.

<https://muytecnologicos.com/diccionario-tecnologico/bootstrap>

Otto, M., & Thornton, J. (s/f). *Sistema de cuadrícula (Grid)*. Esdocu.com. Recuperado el 2 de mayo de 2025, de <https://getbootstrap.esdocu.com/docs/5.1/layout/grid/>

(Otto & Thornton, s/f-a)

Otto, M., & Thornton, J. (s/f-a). *Contenedores*. Esdocu.com. Recuperado el 6 de mayo de 2025, de <https://getbootstrap.esdocu.com/docs/5.1/layout/containers/>

(Otto & Thornton, s/f-c)

Otto, M., & Thornton, J. (s/f-c). *Tablas*. Esdocu.com. Recuperado el 7 de mayo de 2025, de <https://getbootstrap.esdocu.com/docs/5.1/content/tables/>

(Bootstrap Table responsive - free examples & tutorial, s/f)

*Bootstrap Table responsive - free examples & tutorial*. (s/f). MDB - Material Design for Bootstrap. Recuperado el 7 de mayo de 2025, de <https://mdbootstrap.com/docs/standard/extended/table-responsive/>

*Sistema de rejilla | Bootstrap 4*. (s.f.). <https://ajgallego.gitbook.io/bootstrap-4/sistema-de-rejilla>

Universidad Don Bosco. (2020). *Guía de laboratorio #4: Introducción al diseño responsivo con el framework Bootstrap*. Facultad de Ingeniería, Escuela de Computación. [https://www.udb.edu.sv/udb\\_files/recursos\\_guias/informatica-ingenieria/lenguajes-interpretados-en-el-cliente-\(ingenieria\)/2020/ii/guia-4.pdf](https://www.udb.edu.sv/udb_files/recursos_guias/informatica-ingenieria/lenguajes-interpretados-en-el-cliente-(ingenieria)/2020/ii/guia-4.pdf)

W3Schools.com. (s.f.).

[https://www.w3schools.com/bootstrap5/bootstrap\\_grid\\_system.php](https://www.w3schools.com/bootstrap5/bootstrap_grid_system.php)

CertiDevs. (2024, 6 septiembre). *Imágenes responsive en Bootstrap*. CertiDevs.

<https://certidevs.com/tutorial-bootstrap-imagenes-responsive>

*Contributors, M. o. J. T. A. B. (s. f.). Imágenes.*

<https://getbootstrap.esdocu.com/docs/5.1/content/images/>

*Contributors, M. o. J. T. A. B. (s. f.-b). Forms.*

<https://getbootstrap.com/docs/4.0/components/forms/>

*Formularios | Bootstrap 4. (s. f.).* <https://ajgallego.gitbook.io/bootstrap-4/componentes-responsive/formularios>

*Contributors, M. o. J. T. A. B. (s. f.-c). Grid system.*

<https://getbootstrap.com/docs/5.0/layout/grid/>

*Responsive Images With Bootstrap: Techniques and Examples. (2024, 1 octubre).*  
*Cloudinary.* <https://cloudinary.com/guides/responsive-images/4-techniques-for-creating-responsive-images-with-bootstrap>