Índice

[**Frameworks ¿Qué son?** 1](#_Toc99046231)

[**¿Qué es Material Design y Materialize?** 1](#_Toc99046232)

[**Material Design** 1](#_Toc99046233)

[**Materialize** 2](#_Toc99046234)

[**La lógica detrás del Grid system** 2](#_Toc99046235)

[**Documentación y características de Materialize** 3](#_Toc99046236)

[**¿Por qué es bueno utilizar en algunas ocasiones el CDN?** 3](#_Toc99046237)

[**Formas de trabajar con Materialize** 3](#_Toc99046238)

[**Componentes de Materialize** 3](#_Toc99046239)

[**Como sobrescribir una clase de materialize** 4](#_Toc99046240)

[**Otra forma de sobrescribir** 5](#_Toc99046241)

# **Frameworks ¿Qué son?**

Son **clases** y **estilos pre-definidos** para construir páginas web de forma rápida.

Materialize es un framework de CSS.

# **¿Qué es Material Design y Materialize?**

## **Material Design**

Es un concepto de diseño que sacó Google con buenas prácticas para poder empezar a generar aplicaciones y páginas web.

Define las reglas para usar:

* Colores.
* Elementos.
* Interacciones.
* Animaciones.
* Tipografías.

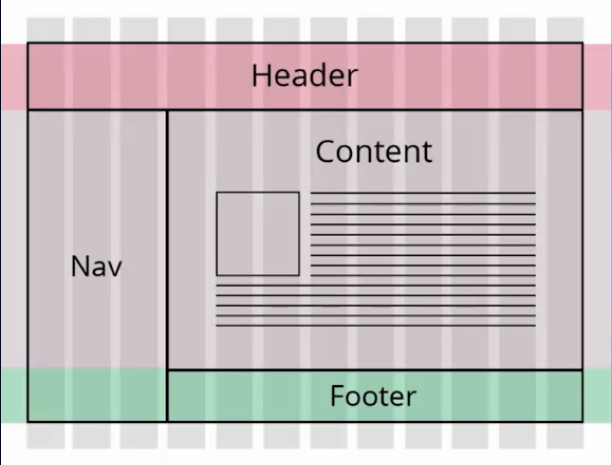
Se trata de un diseño más limpio, en el que predominan **animaciones** y **transiciones de respuesta**, el **relleno** y los **efectos de profundidad** tales como la **iluminación** y las **sombras** que proporcionan significado sobre lo que se puede tocar y cómo se va a mover.

## **Materialize**

Es un Framework que contiene las reglas y elementos de **Material Design** para implementarlos fácilmente en sitios web y Webapps usando CSS y que sean escalables.

# **La lógica detrás del Grid system**

Un **Grid System** trae 12 columnas y es como divides la posición de las etiquetas contenedoras entre cuantas columnas tenemos que utilizar para maquetar un diseño y luego poder reposicionar todos los elementos al momento de que estemos trabajando de forma responsive.



# **Documentación y características de Materialize**

## **¿Por qué es bueno utilizar en algunas ocasiones el CDN?**

Si alguna persona que ya ha utilizado alguna página que ocupe Materialize, en el CDN ya se han quedado ciertas cosas en caché.

Entonces si de repente viene a mi página, mi sitio va a cargar un poco más rápido porque ya no tendría que estar haciendo esas peticiones.

## **Formas de trabajar con Materialize**

* Trabajar con la versión Standard con CSS y JS.
* Podemos trabajar proyectos en conjunto con Sass.
* Usar **CDN**: Es recomendable trabajar con CDN ya que al ser Materialize un Framework OpenSource, hay probabilidades que los usuarios ya tengan en caché muchos elementos que ayuden a cargar más rápido nuestro sitio web.
* Usar NPM.
* Templates.

# **Componentes de Materialize**

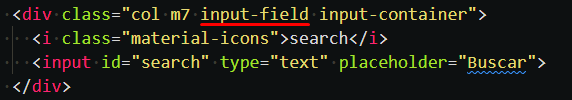
Dentro de los componentes que tiene Materialize, tenemos:

* Badges.
* Buttons.
* Cards.
* Navbar.
* Footer.
* y más!

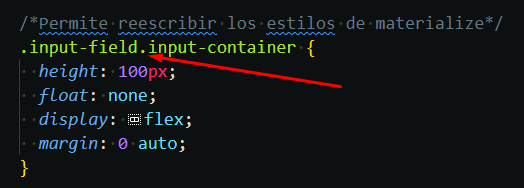
Esto nos ayuda a ahorrar mucho tiempo.

# **Como sobrescribir una clase de materialize**

Si tenemos la siguiente estructura, la cual la clase predefinida por materialize es **input-field**



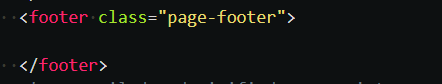
Para sobrescribir la clase con css hacemos



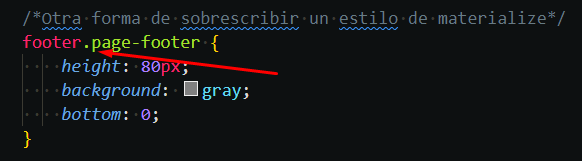
Llamamos la clase de materialize **input-field** y seguida sin espacios, llamamos a nuestra clase personalizada. De esta forma podremos sobrescribir cualquier clase ya predefinida de materialize.

## **Otra forma de sobrescribir**

Tenemos la clase predefinida **page-footer**



Para sobrescribir esta clase haremos



Llamamos a la etiqueta html y seguido, sin espacios, llamamos a la clase predefinida para luego ser sobrescrita con nuestros estilos.