

# La era de la multiplicidad de dispositivos

*"Piense más allá del escritorio y cree diseños que respondan a las necesidades de sus usuarios, sin importar cuán grande o pequeña sea la pantalla."*

Ethan Marcotte — Diseñador gráfico.

## Responsive web design



Hace años que la web se maquetaba de manera tal que un solo código fuente nos permite adaptar nuestro sitio a todos los dispositivos. Tal como ocurrió a en los inicios de internet, múltiples personas y equipos se encargaron de generar cambios para mejorar las formas de trabajo y adaptarlas a las necesidades de cada época. Según Statista, en 2020 hay [3,5 mil millones de teléfonos inteligentes en el mundo](#), por lo que el diseño de los sitios web debe poder abrirse no solo en computadoras de escritorio o portátiles...

Al hecho de diseñar en función del dispositivo se lo conoce como **Responsive Web Design**. Abhishek Shastri lo define de la siguiente manera:

***"El Responsive Web Design es la forma práctica de crear un sitio web adecuado para trabajar en cada dispositivo y cada tamaño de pantalla, sin importar cuán grande o pequeño, móvil o de escritorio. El sitio web se construye receptivo teniendo en cuenta los usuarios de todos los dispositivos multimedia."*** [\(Making a responsive web design\)](#).

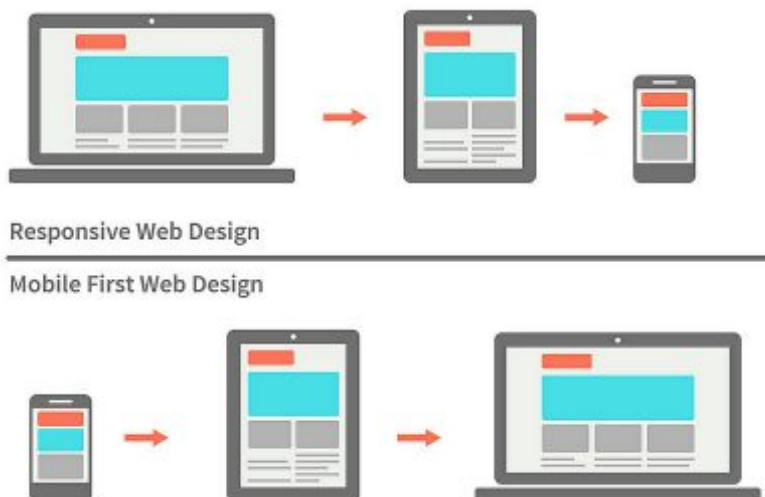


Javier Torres hace un recorrido histórico y nos cuenta cómo fue el origen de esta práctica tan necesaria:

*“Todo comenzó con [Ethan Marcotte](#) cuando introdujo el término Responsive Web Design y publicó un libro con el mismo nombre. Y es que no hace tanto que se pensaba y diseñaba la web pensando en tamaños fijos de pantalla y lo que ahora se llama web 2.0 antes simplemente era internet y lo que ahora llamamos Diseño Web Responsive antes ni siquiera existía, o al menos no del modo actual.*

*El período anterior a la era Responsive (cuando no existían ni smartphones, ni tablets, y lo más de los más era un ordenador portátil) los diseñadores web correteábamos detrás de los fabricantes de hardware y monitores para adaptar nuestros diseños a los nuevos modelos de pantalla. La diversidad de dispositivos se resumía a pantallas más o menos grandes.” ([Un poco de historia de Diseño Web Responsive](#))*

Otro hito importante fue cuando [Luke Wroblewski](#) lanzó en 2011 su libro “[Mobile first](#)”. Generó un cambio de perspectiva que modificó las formas en que **diseñamos sitios y aplicaciones web**. Se convirtió en un modelo a seguir, que tiene en cuenta primero el teléfono móvil y a los/as usuarios/as.



Es fundamental pensar la forma de concentrar toda la información en una única web que pueda **adaptarse** a los distintos dispositivos. En esta bitácora vamos a acercarnos a esta forma de pensamiento que tenemos que desarrollar.

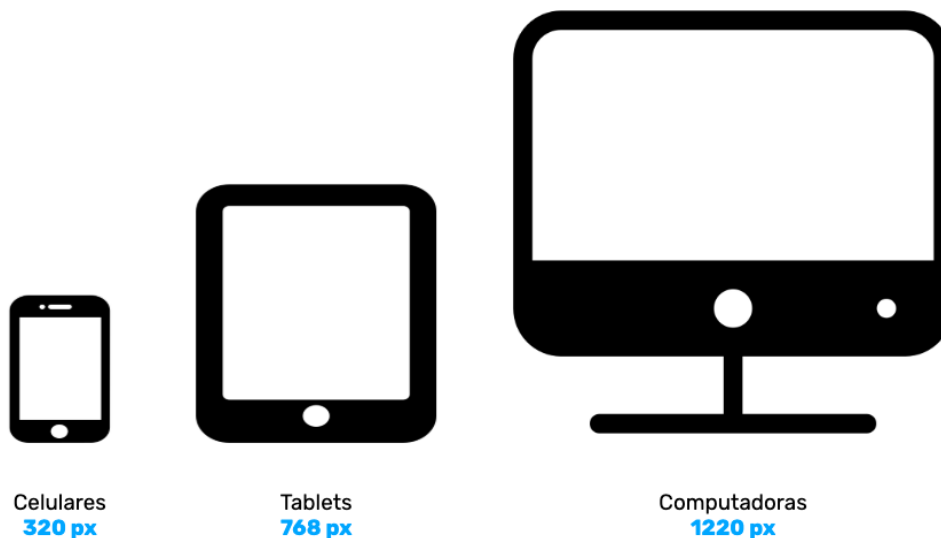


## Criterios para diseñar

Cuando nos referimos a [Responsive Web Design \(RWD\)](#) no solo incluimos computadoras o celulares, sino también netbooks, tablets e impresoras. Tratándose de dispositivos de distintos de tamaños, nos servimos de **convenciones** para poder desarrollar primero en forma más general y luego en forma más específica.

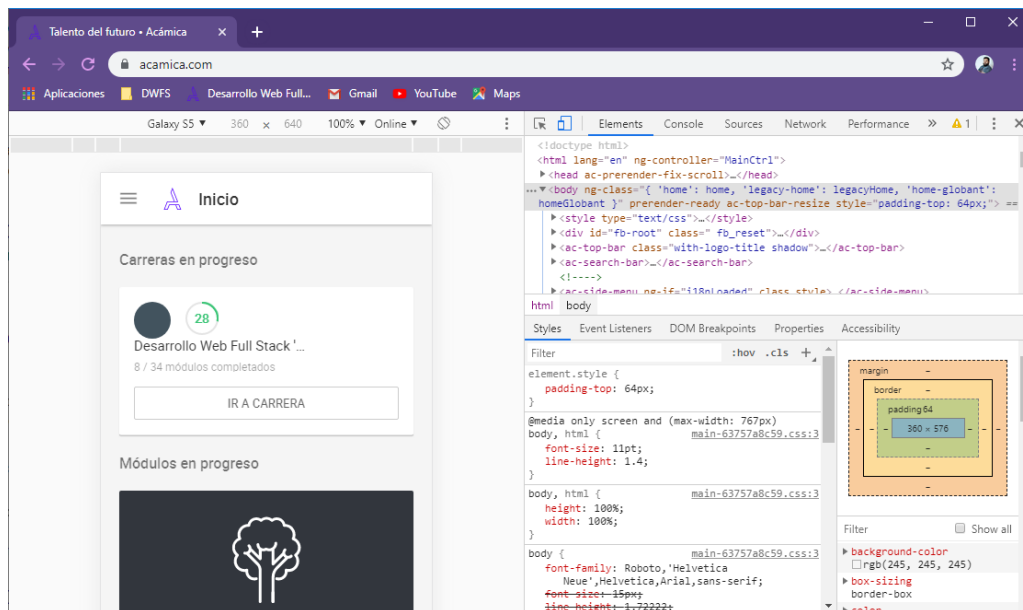
Estas convenciones se denominan **breakpoint** y constituyen los anchos de pantalla aceptados por convención para aplicar estilos en función de los distintos dispositivos, así como también su orientación vertical u horizontal.

Los **breakpoint** más comunes para trabajar como mínimo son:



Puedes usar la herramienta [developer's tool](#) –se encuentra en todos los browsers– para ver los breakpoints. Para hacerlo en Google Chrome, debes hacer clic derecho sobre la pantalla y seleccionar “inspeccionar” del menú contextual. Se mostrará la pantalla de responsive design a la derecha:





¿Qué sucede si queremos modificar estilos en función de las distintas **resoluciones** del **viewport** (el área del navegador en la cual se renderizará el body del documento HTML)?

En este caso los **breakpoint** no sirven por sí solos, sino que debemos incorporarlos a CSS y para ello están las **media query**. Éstas son reglas que se introducen con el objetivo de crear o modificar propiedades específicas para distintas resoluciones.

### [abre card ejemplo]

Observa [este ejemplo](#) y presta atención a cómo cambia el tamaño de las cajas y también su color de fondo cuando el **viewport** es menor a 600px (pista: tienes que achicar el navegador lo suficiente).

### [cierra card ejemplo]



## Diseño de tipografía responsiva

Al diseñar sitios web, también es fundamental pensar en uno de los elementos más importantes para comunicar el contenido: los **tamaños de fuentes** (además de la tipografía por supuesto).

Al momento de trabajar con un diseño adaptable entre dispositivos, existen **unidades relativas** que son indispensables, ya que nos van a permitir basarnos en el tamaño de la pantalla, por ejemplo, o tomar como referencia una única medida. Si bien hay algunas más complejas, estas son las más usadas:

- Unidad `em`: hace referencia al tamaño de la tipografía del elemento que nos encontremos estilando. Su nombre se remonta a que inicialmente se tomaba el tamaño de la letra M, o em en inglés.
- Unidad `rem`: relativa al tamaño de fuente del root element `<html>`. El root element es ni más ni menos que el elemento HTML más importante de una página web y que contiene todo el documento: `<html>`.

Existen otras unidades de medida para trabajar en función del tamaño del **viewport** según el tipo de resolución para una correcta visualización. Estas son importantes y nos permiten modificar:

- Unidad `vw`: relativa al ancho del viewport.
- Unidad `vh`: relativa al alto del viewport.
- Unidad `vmin`: relativa al lado más chico del viewport, independientemente de su disposición.
- Unidad `vmax`: relativa al lado más grande del viewport, independientemente de su disposición.


En el próximo encuentro comenzaremos a utilizar los criterios del RWD en nuestros proyectos para que sean acordes a las demandas de los/as usuarios/as. El diseño responsive es fundamental para que usuarios/as puedan acceder a tu sitio desde múltiples dispositivos, y por eso ¡Google favorece a tu sitio si usas [diseños responsive](#)!



## Live Coding

¿Te quedaste con ganas de más? ¿Prefieres escuchar más que leer para aprender? ¿Quieres ver al director de la carrera - Daniel Segovia - codear lo que te contamos en la bitácora? Aquí te dejamos un video:

[CSS - Responsive Web Design](#)

 **Recuerda cargar todas las preguntas** que te hayan surgido sobre los temas cubiertos en la presente bitácora en el **Trello** de tu grupo de estudio. Tu equipo docente armará el repaso en clase a partir de esto.

## ¡Prepárate para el próximo encuentro!

### Profundiza

En el encuentro conoceremos cómo se diseña un sitio o aplicación web de forma responsiva y pondremos en práctica los criterios de Mobile First. Te recomendamos que leas los siguientes artículos:



[Concepto Mobile First](#)



[8 Excellent Mobile First Design Examples for Designer Inspiration](#)



[MDN - Usando la etiqueta meta viewport para controlar la composición en los navegadores móviles](#)

### Herramientas



[Google Mobile-friendly Test](#) es una herramienta de Google para testear si un sitio web es amigable para dispositivos móviles. Google rankea el



posicionamiento de los sitios web, entre otras cosas, por su capacidad de mostrarse correctamente.

### Comunidad



Existen muchos artículos que tocan el responsive web design desde diferentes enfoques. Lee al menos 3 artículos y selecciona datos interesantes para compartir con los docentes y compañeros en el próximo encuentro.

### Challenge



Dibuja 2 cajas de 400px que se muestren una al lado de la otra. Cuando el viewport es menor a 800px haz que las cajas se vean una debajo de la otra.

