

# Breve historia del diseño web para diseñadores

---

Por Ricardo Dargence

Cómo evolucionó el diseño web y cómo se fue cerrando la brecha entre el código y el diseño.

```
<a name="www"></a>
<table width="500% border=
  <tr>
    <td height="68" width=
    <td> <form name=login
    <input type=hidden na
```

Algo que siempre me he preguntado es: ¿por qué las cosas simples se vuelven más difíciles, y las fáciles se vuelven más difíciles, al momento de crear un sitio web? Me di cuenta de que para conocer la respuesta hay que saber el origen, evolución e historia del diseño web. Aquí te

contaré cómo evolucionó el diseño web y cómo se fue cerrando la brecha entre el código y el diseño.

## Las edades oscuras del diseño web (1989)



El comienzo del diseño web fue bastante oscuro, ya que las pantallas eran literalmente negras y solo unos pocos píxeles monocromáticos podían verse en ellas. El diseño se realizaba mediante símbolos y tabulación (tecla Tab). Ahora avancemos al tiempo en que apareció la Interfaz Gráfica de Usuario: la era de las tablas.

## Tablas - El comienzo (1995)



El nacimiento de los navegadores que podían mostrar imágenes fue el primer paso hacia el diseño web tal como lo conocemos hoy. La opción más cercana para estructurar información fue el concepto de tablas en HTML. Así que poner tablas dentro de las tablas y descubrir formas inteligentes de mezclar células estáticas con células fluidas, fue lo que comenzó la era del diseño web, según el libro de David Siegel, *Creando Sitios Asesinos*.

El problema fue la dificultad para mantener estructuras tan frágiles. Este también fue el momento en que el término *slicing designs* se hizo popular. Los diseñadores hacían un diseño elegante, pero los desarrolladores podían dividirlo en pequeños pedazos y descubrir la manera de hacer las características del diseño web funcionarían mejor.

Por otro lado, las tablas tenían algunas características asombrosas como la capacidad de alinear las cosas verticalmente y definir sus medidas en píxeles o en porcentajes. En ese entonces las tablas nos dieron lo más cercano a la creación de una cuadrícula, y este fue un

gran beneficio. También fue el momento en que muchos desarrolladores decidieron que no les gustaba la codificación de *front-end*.

### **JavaScript viene al rescate (1995)**



JavaScript fue la respuesta a las limitaciones del HTML. Por ejemplo, ¿necesitabas una ventana emergente, deseabas modificar dinámicamente el orden de algo? La respuesta fue JavaScript. Sin embargo, el problema principal fue que JavaScript se tenía que colocar sobre el área que hace que la web funcione, y además se debía cargar por separado.

Muy a menudo se usó como un parche fácil para un desarrollador perezoso, pero es muy poderoso si se usa sabiamente. Hoy en día preferimos evitar JavaScript si obtenemos el mismo resultado utilizando CSS. Sin embargo, JavaScript se mantiene vigente a través de librerías front-end (jQuery) y back-end (Node.js).

### **La era dorada de la libertad - Flash (1996)**



Para romper las limitaciones del diseño de páginas web nació una tecnología que prometía una libertad nunca antes vista. El diseñador podría crear formas, diseños, animaciones, interacciones, usar cualquier fuente y todo esto en una sola herramienta: Flash. El resultado final se empaquetaba como un archivo y luego se enviaba al navegador para que este lo mostrara.

Es decir, mientras un usuario tuviera el último complemento de flash y algo de tiempo libre para esperar hasta que se cargara el sitio, podría ver magia en el diseño web. Esta fue la época dorada para las páginas de inicio, las animaciones de introducción, todo tipo de efectos interactivos.

Desafortunadamente, no era demasiado abierto ni fácil de buscar y, ciertamente, consumía mucha potencia de procesamiento. Cuando Apple decidió abandonarlo en su primer iPhone (2007), Flash comenzó a decaer. Al menos para diseño web.

### **CSS toma la batuta y se crean las hojas de estilo en cascada (1998)**



Casi al mismo tiempo que Flash, nació un mejor enfoque para estructurar el diseño desde un

punto de vista técnico: las hojas de estilo en cascada (CSS). El concepto básico consistía en separar el contenido de la presentación. Así que la apariencia y el formato se creaban en CSS, pero el contenido en HTML. Las primeras versiones de CSS distaban mucho de ser flexibles, pero el mayor problema era la compatibilidad con los navegadores.

Pasaron algunos años antes de que los navegadores comenzaran a admitirlo por completo y, a menudo, tenía muchos errores. Esta fue la era en que algunos navegadores tenían características nuevas y otros no las tenían, todo ello se convirtió en una auténtica pesadilla para el desarrollador web. Para ser claros, CSS no es un lenguaje de codificación, más bien, es un lenguaje declarativo. ¿Deben los diseñadores web aprender a codificar? Tal vez. ¿Deben entender cómo funciona CSS? ¡Absolutamente!

## **El gran desafío del desarrollo para móviles - Rejillas y marcos (2007)**



Navegar por la web en teléfonos móviles era todo un desafío. Además de los diferentes diseños para dispositivos, también había problemas de paridad de contenido: ¿Debería ser el mismo diseño en pantalla pequeña o debería ser reducido ¿Dónde colocar todos los anuncios bonitos y parpadeantes en esa pequeña pantalla? Estas fueron algunas de las grandes cuestionamientos.

La velocidad también fue un problema, ya que los sitios tenían mucho contenido que cargar y poco tiempo para hacerlo. Después de algunas iteraciones, el sistema de cuadrícula 960 ganó, y la división de 12 columnas se convirtió en algo que los diseñadores empezaron a utilizar a diario.

El siguiente paso fue estandarizar los elementos comúnmente utilizados como: formularios, navegación, botones y empaquetarlos de una manera fácil y reutilizable. Básicamente, se hacía una biblioteca de elementos visuales que contenía todo el código.

Los ganadores aquí fueron Bootstrap y Foundation. La desventaja es que los diseños a menudo tenían el mismo aspecto y los diseñadores aún no podían acceder a ellos si no sabían cómo funcionaba el código, pero aún así se volvió una de las herramientas de diseño web más importantes.

## **Un genio desarrolla el Diseño web responsivo y abre un nuevo sendero (2010)**



Un tipo brillante llamado Ethan Marcotte decidió desafiar el estilo existente y propuso aplicar el mismo contenido a diferentes layouts de un mismo diseño. Así se acuñó el término

diseño web *responsive*, que marcó un antes y un después en la historia del diseño web. Técnicamente aún se utilizaba HTML y CSS, por lo que fue un avance conceptual.

Sin embargo, hay muchos conceptos erróneos aquí. Para un diseñador, *responsive* significa adaptar y probar múltiples diseños. Para el cliente significa que funciona en el teléfono. Para un desarrollador es la forma en que se colocan las imágenes, las velocidades de descarga, la semántica, etc. El principal beneficio aquí es la paridad de contenido, lo que significa que el mismo sitio web funciona en todas partes.

## **La Simplificación de elementos da prioridad al mensaje (2012)**



Esto sonará extraño pero es real: el diseño de más layouts requiere más tiempo, por lo que decidimos agilizar el proceso eliminando los sofisticados efectos de sombra y volviendo a las raíces del diseño, priorizando el contenido. Por eso ahora utilizamos la fotografía fina, la tipografía, las ilustraciones nítidas y los diseños cuidadosos.

La simplificación de elementos visuales o el llamado diseño plano también es parte del proceso. El principal beneficio es que ahora es prioritario dar el mensaje antes que el diseño bonito. Los botones brillantes se reemplazan por iconos y eso nos permite usar imágenes vectoriales y fuentes de iconos. Las fuentes web ofrecen hermosa tipografía. Lo curioso es que la web estuvo cerca de esto desde el principio. Pero, en fin... a esto me refería con que lo difícil se vuelve sencillo.

## **El futuro luce brillante y lleno de oportunidades**



El santo grial del diseño web ha sido hacerlo realmente visual y llevarlo al navegador. ¡Imagina que los diseñadores simplemente mueven las cosas alrededor de la pantalla y sale un código limpio! Y no me refiero a cambiar el orden de las cosas, isino a tener total flexibilidad y control! ¡Imagina que los desarrolladores no tienen que preocuparse por la compatibilidad del navegador y pueden centrarse en la resolución de problemas reales! Técnicamente, hay algunos conceptos que apoyan el movimiento en esa dirección.

Las nuevas unidades en CSS como `vh` , `vw` (altura y anchura de la ventana gráfica) permiten una flexibilidad mucho mayor para posicionar los elementos.

También solucionará el problema que ha desconcertado a tantos diseñadores: por qué centrar algo verticalmente en CSS es tan difícil. Flexbox es otro concepto genial que forma parte de CSS. Permite crear diseños y modificarlos con una sola propiedad en lugar de

escribir gran cantidad de código.

Y finalmente, los componentes web son elementos aún más valiosos. Es un conjunto de componentes agrupados, es decir, una galería, un formulario de registro, etc. Esto introduce un flujo de trabajo más sencillo, donde los elementos se convierten en componentes básicos que se pueden reutilizar y actualizar por separado.

¿Qué piensas del futuro de la web?

Publicado el 22/07/2019

---

## Bibliografía

- [Wikipedia. Historia del diseño web](#)
- [Elementos de una página web](#)
- [Slideshare.net. Historia del diseño web](#)



ISSN 1851-5606

<https://foroalfa.org/articulos/breve-historia-del-diseno-web-para-disenadores>

