



ELEARNING TOTAL

Desarrollo Web HTML5 y CSS3 – Unidad 8

Diseño Web Responsive HTML5 y CSS3

Unidad 8: Responsive Design



Indice

Unidad 8: Responsive Design

Responsive Design





Objetivos

Que el alumno logre:

- Conocer e implementar los nuevos elementos incorporados en CSS3 para el diseño multiplataforma.





Responsive Design

Hoy en día ha aumentado de forma exponencial el uso dispositivos móviles, además de la demanda de contenido para estos, por lo que cada vez es más necesario una web adaptable a los distintos dispositivos.

¿Qué es el Responsive Design?

Esta técnica de diseño web consiste en crear una estructura de una página web que según el tamaño de la pantalla (o ventana) en la que se visualice variará su contenido para que siempre sea visible y cómodo de usar desde computadoras de escritorio, tablets y smartphones, y se puede poner en práctica esta forma de adaptar el contenido a todo tipo de resoluciones con hojas de estilo CSS o con JavaScript (tenemos que tener en cuenta también en que puede haber personas o bots que tengan desactivado JavaScript).

Beneficios

La web se visualizará en todos los dispositivos que usemos correctamente, sin necesidad de hacer zoom y se adaptará a los giros en dispositivos móviles.

Google tiene en cuenta que páginas usan diseños que se adaptan a dispositivos móviles.

Ayudamos a las personas que tienen discapacidades visuales a que puedan usar más fácilmente la web.

¿Cómo implementarlo?

Podemos ver dos opciones, tener una web existente o crearla de cero, si existe una web, dependiendo de su complejidad puede ser relativamente sencillo adaptarla o puede ser una tarea tediosa (sobre todo si es un CMS con una plantilla desastrosa, a nivel de estructura).



Tipos de Responsive Designs

Podemos determinar los siguientes tipos de responsive design:

Adaptándose al ancho

Este tipo de diseño es uno de los más comunes, que se distingue por no mover en exceso los elementos de la web e intentar que se adapten al ancho de pantalla reduciendo el menú (a veces cambiando la disposición), redimensionando las imágenes y poco más, como ejemplos pongo los siguientes:

- [Mejorando.la](http://mejorando.la/) (<http://mejorando.la/>)

Cascada de columnas

En este caso tenemos una web con varias columnas que al visualizarse en pantallas estrechas se pondría una debajo de la otra seguidamente, por ejemplo:

- [Modernizr](http://modernizr.com/) (<http://modernizr.com/>)
- [Wee Nudge](http://weenudge.com/) (<http://weenudge.com/>).

Reestructuración

Como el propio título indica en esta variante cambiamos la estructura de los elementos disponiéndolos de una forma distinta según el tamaño de ventana y se puede hacer de muchas formas (hay solo depende del diseño que queramos) por ejemplo:

- [CSS-Tricks](http://css-tricks.com/) (<http://css-tricks.com/>)
- [Food Sense](http://foodsense.is/) (<http://foodsense.is/>)

Cambio del diseño visual

Esta se podría decir que no es una forma de estructurar el contenido para hacerlo más fácil de manejar fuera del navegador de escritorio, sino que según el tamaño de la ventana adapta el diseño de una forma u otra por



motivos estéticos, y suele usarse en las típicas web que dan la información sobre algo de forma rápida y concisa, como en las siguientes:

- Neovada (<http://www.neovada.com/>)
- Design made in Germany (<http://www.designmadeingermany.de/magazin/5/>)

Viewport

Esta meta-etiqueta fue creada en principio por **Apple** para sus dispositivos móviles, pero se ha convertido en todo un estándar que es soportado por la mayoría de los dispositivos móviles (smartphones, tablets y gran parte de móviles de gama media y baja).

Su uso es totalmente necesario, ya que sino el navegador establece el ancho con el que prefiere visualizar una página en lugar de usar el ancho del que dispone (es decir, si la pantalla de nuestro móvil tiene 400px y el navegador detecta que lo óptimo sería visualizarla con 700px así lo hará si no usamos esta meta-etiqueta).

```
<meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no">
```

Se pueden usar los siguientes parámetros (separados por comas):

- **Width:** ancho de la página (se puede establecer en píxeles o como device-width y usará el ancho del que dispone).
- **Height:** alto de la página, actúa igual que el width.
- **Initial-scale:** escala o zoom inicial de la página (este y los demás tipos de escala se establecen con valores como 1.0 para no tener zoom o 2.5 para tener un zoom del 2,5 de aumento, por ejemplo).
- **Minimum-scale:** zoom mínimo que podemos hacer en la página.
- **Maximum-scale:** zoom máximo que podemos hacer en la página.
- **User-scalable:** establece si está permitido o no hacer zoom (yes/no).



Resumen

En esta Unidad...

Trabajamos con Responsive Design

