

10 Herramientas JavaScript Necesarias Para HTML5

HTML5 llego para mejorar la vida de los desarrolladores, pero, ¿Que seria de [HTML5](#) si no dependiera de otras tecnologías web?. No es para nada desconocido que **HTML5 sigue siendo un estándar relativamente nuevo**, aunque existe un montón de proyectos de código abierto, y de pago, que pueden ayudar a mejorar el trabajo con HTML5 y [Javascript](#) ;)



La herramientas de JavaScript son necesarias para algunos **nuevos elementos de HTML5**, tal es el caso de `<canvas>`, `<audio>` y WebGL. Es por esto, que en este artículo te compartiremos algunas herramientas, códigos, Frameworks y librerías de JavaScript para mejorar de gran manera el desarrollo que realices en HTML5.

1.

HTML5

Boilerplate



Boilerplate le permite construir grandes sitios mediante una plantilla descargable, que incluye **Normalize.css**, **jQuery** y **Modernizr**. También puedes solicitar una estructura hecha en un **FrameWork** para realizar el trabajo que no has podido realizar.

HTML5 Boilerplate también incluye la configuración de Apache, con la cual se te facilitara, para que puedas tener un gran rendimiento con menos esfuerzo.

2. Fabric.js

Esta es una **librería JS** para para que puedas usarla sin problema con los nuevos elementos de HTML5 como Canvas. Puedes usarla para analizar los elementos SVG y el uso de los componentes de objetos, tales como la agrupación y sombreado. **Fabric.js** le ayuda a crear animaciones con facilidad, a partir

de figuras geométricas simples a través de imágenes reales.

Por supuesto, no es necesario Fabric.js para crear dibujos sobre Canvas, este último no te permite crear formas más complejas para manipular/animar con facilidad. Cuenta con un sistema de eventos extenso para permitir llevar a cabo eventos, como el hover con el ratón. También le permite crear por arrastrar y soltar pixel.

3. Modernizr

Es una de las herramientas JavaScript más populares de la Web y se incluye dentro de algunos Framework. Con Modernizr, puede escribir tanto condicionales en JS y CSS para ayudar a crear las páginas, incluso cuando el navegador no es compatible aún los nuevos elementos HTML5.

Funciona mediante la ejecución de la página de carga con el fin de recoger las características y luego **crear un objeto JS**, que a su vez agrega clases a los elementos HTML para que puedas usar CSS. Es de código abierto y permite a los diseñadores utilizar HTML5 y CSS3, con retrocesos para que los usuarios que no disponen de los navegadores con sus nuevas versiones, puedan tener una gran experiencia de usuario.

4. Backbone.js

Te permite **estructurar las aplicaciones web** añadiendo modelos con los eventos de clave y valor. Se puede conectar todo a la API JSON preconfigurado y es en sí mismo a numerosas funciones de gestión de eventos. **Backbone** te permite representar datos como modelos y cuando una acción del usuario hace que los

cambios se produzcan, el modelo automáticamente dispara un evento de cambio. Esto significa que en lugar de tener que «escribir código de pegamento», si el modelo cambia actualiza el código HTML en consecuencia.

5. Dirty Markup

Prefieres usar las herramientas en línea en lugar del trabajo a nivel local? Entonces **Dirty Markup** te gustará, especialmente si eres relativamente nuevo en el **desarrollo web**. Es una aplicación gratuita que puedes usar para la limpieza de código de forma rápida y sencilla, y no sólo sirve para usar con JS. También puede limpiar HTML5 y CSS con él, ya que combina HTML Tidy, CSS y JS Tidy. Sin duda es una gran herramienta para que puedas dar algo de estilo y no perderte en las líneas de código que hayas escrito ;)

Junto a todo esto se utiliza el Editor de Ace, que es un editor de código en línea que tiene características similares a los editores como Sublime texto. Para Dirty marcado usted tiene la opción de elegir el código que desea limpiar; sólo pegarlo en el cuadro editor junto al cuadro de opciones y luego sólo tiene que pulsar el botón 'limpia'. ¡Simple!

6. JFormer

Esta es una herramienta sencilla y eficaz para la adición de figuras y está escrito bajo jQuery. Se le permite validar formularios, cliente y servidor, y procesarlos, usando Ajax, sin necesidad de cambiar las páginas. Los formularios que se genera utilizando **JFormer** cumplen con los estándares necesarios.

Es de uso gratuito, y se distribuye con la licencia MIT. Existen plantillas ya preparadas para ayudar a hacer las cosas mucho más rápido y fácil para tu desarrollo. Puedes editar el CSS con el fin de dar la personalización que desees dar a tu proyecto web.

7. Live.js

Esta es una herramienta de monitoreo que vigila el trabajo de [Javascript](#) y [CSS](#) mediante el envío de peticiones al HEAD del servidor. Entonces cualquier cambio necesario se aplican de forma dinámica para el CSS, o para HTML y Javascript volverá a cargar automáticamente la página.

Es un **Framework** y un lenguaje independiente, lo que significa que se puede utilizar con varios idiomas y es compatible con Firefox, Chrome, Safari, Opera y IE6 +.

8. Chart.js



Esta es una herramienta muy útil para [crear gráficos estadísticos](#) de todo tipo (gráficos circulares, encuestas, dona gráfico etc) utilizando los **elementos canvas y SVG en HTML5**. Sin embargo, los gráficos que se pueden crear no brindan una interacción para el usuario. Para esto, puedes usar **Fabric.JS y con SVG para dar la interacción** que deseas.

Te permite crear gráficos complejos y cuando se utiliza con SVG permitirá una mejor desplazamiento y rendimiento de la página.

9. Grunt

Es un recurso bastante popular, [Grunt](#) es un corredor de tarea para Javascript que permite automatizar las tareas comunes que tienden a ser repetitivas, incluyendo minificación; compilación; prueba de unidad; y más ... Grunt también tiene un ecosistema grande y útil que contiene un montón de extensiones para ayudar a automatizar varias tareas.

10. Cocos2D-javascript

También pensamos en quienes les gusta **desarrollar juegos**, Cocos2D-javascript es un motor de juegos y gráficos 2D para **ejecutar juegos en el navegador**. Esto significa que puedes realizar el renderizado 2D en el navegador sin necesidad de extensiones adicionales, como Flash o [Java](#).

Mientras que la documentación indica que no todos los navegadores son compatibles con el renderizado en el navegador, debido a la falta de **soporte para HTML5**, definitivamente esto no llevara mucho tiempo ;) La mayoría de los navegadores que se han probado con Canvas y SVG realizan el trabajo sin problemas.

Eso es todo amigos!

Sin duda alguna, la web está permitiendo usar una serie de tecnología web que ayudan al diseño y así mejorar la experiencia del usuario. JavaScript es lo suficientemente potente como para brindar una mejora considerable. Existen algunos grandes ejemplos de cómo se puede usar JavaScript, ¿por qué no darle una oportunidad y empezar a experimentar con los nuevos elementos HTML5?