

Cinco días de HTML5

HTML



Sandra Guerrero
Zuriñe Menendez
Daniel Martinez
y otros...



desarrolloweb.com/manuales/curso-gratuito-html5.html

Introducción: Curso gratuito 5 días de HTML5

Grabaciones de las clases del Curso Gratuito de HTML5 impartidas por EscuelaIT en marzo de 2014. En este curso se ofrecieron una serie de presentaciones sobre los puntos clave que los desarrolladores deben tener en cuenta sobre HTML5.

Cada una de las 5 clases fueron impartidas en directo por un software de videoconferencia por un profesor distinto. Aquí agrupamos no solamente las grabaciones de las clases, sino también los ejemplos realizados y diapositivas de las presentaciones.

Las clases están orientadas para que los estudiantes tengan un conocimiento global de lo que te ofrece HTML5, en sus tres pilares fundamentales, como es el propio lenguaje HTML, el lenguaje para edición de los estilos CSS y el lenguaje para scripting Javascript.

También se pretendió motivar a los interesados a usar ya mismo HTML5 y aprovechar las características de los navegadores modernos, creando sitios que ofrezcan una más rica experiencia de usuario.

Encuentras este manual online en:

<http://desarrolloweb.com/manuales/curso-gratuito-html5.html>

Autores del manual

Las siguientes personas han participado como autores escribiendo artículos de este manual.

Sandra Guerrero

Licenciada en economía, consultora y formadora con amplios conocimientos de desarrollo web, marketing online, redes sociales e ingeniería de la información. Ponente en diversos foros y congresos.



Zuriñe Menendez

Desarrolladora Front-end freelance, diseñadora, maquetadora y sobre todo artista del código. Apasionada por el HTML5, microformatos y todas aquellas mejoras que caracterizan un trabajo web profesional.



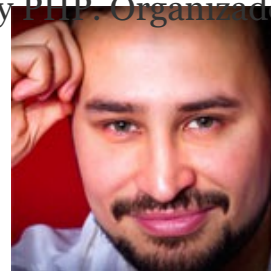
Daniel Martínez

Diseñador gráfico convertido a web



Reinaldo Aguilera

Apasionado del HTML5 y los estándares abiertos para la web, Javascript y PHP. Organizador de los Google Developer Group de Vigo, Betabeers Galicia y los #devIO de DesarrolloWeb.com.



Miguel Angel Alvarez

Miguel es fundador de DesarrolloWeb.com y la plataforma de formación online EscuelaIT. Comenzó en el mundo del desarrollo web en el año 1997, transformando su hobby en su trabajo.



Vídeo Qué es HTML5

Qué es HTML5 y qué no es, mitos y leyendas de los estándares abiertos, pasado, presente y futuro de HTML5.

Este es un vídeo de una clase dedicada a presentar HTML5, "Qué es HTML5", la primera de las sesiones del curso "5 días de HTML", impartida por Sandra Guerrero Sanmarti [@esandracom](https://twitter.com/esandracom). Es una clase introductoria sobre todo lo que nos trae el conjunto de tecnologías agrupadas bajo el término de HTML5, el propio lenguaje HTML en su versión más moderna, así como CSS3 y las API Javascript.

Si aún no te has [apuntado a este curso gratuito, puedes hacerlo desde la web de EscuelaIT](#).

Comenzamos haciendo un poco de historia del lenguaje HTML, con sus distintas versiones, pasando por XHTML, un lenguaje que trató de evolucionar el HTML pero sin los resultados ni la aceptación esperada. Después de ello, llegó HTML5, que verdaderamente cautivó a la comunidad, tanto los desarrolladores como las grandes empresas.



Podemos ver en este sitio web un poco de historia de la web, en torno a los navegadores, estándares, lenguajes, etc. <http://www.evolutionoftheweb.com/> es un sitio web que nos ayudará a ver que, a pesar que la historia de este medio es más bien corta, la rapidez con la que surgen novedades, triunfan y pasan a la historia es verdaderamente enorme.

Novedades que incluye el HTML5

Rápidamente destacamos algunas de las novedades más importantes que encontramos en el lenguaje HTML5. Nos referimos al lenguaje de marcación HTML y comentamos varias de sus etiquetas nuevas, así como campos de formulario avanzados y un poco más inteligentes.

Nota: En esta sencilla introducción más bien enumeramos las cosas que encontramos en el estándar, pero a lo largo del curso gratuito de HTML5 ofreceremos descripciones más concretas y específicas sobre cada cosa. Por ejemplo, en la clase del martes nos

dedicaremos a comentar lo que se llama el "HTML semántico" que es una de las novedades más representativas que nos ha traído HTML5.

Se introducen también otros elementos como los microdatos o microformatos, que también ayudan a la web semántica.

Además, el propio estándar ha traído consigo otros lenguajes como CSS3, una actualización de las "Hojas de estilo en cascada", lenguajes paralelos como SVG y las API Javascript, que realmente es la parte más amplia que nos trae HTML5. Si bien las API no son exactamente actualizaciones del lenguaje Javascript, sí que suponen una ampliación extremadamente considerable de las posibilidades que tenemos los desarrolladores para controlar lo que ocurre en una web y acceder a muchas de las características o periféricos de los dispositivos.

Documento HTML5

En cuanto se refiere al lenguaje de marcación, en HTML5 se creó un documento básico nuevo. Su principal seña de identidad es el "doctype", que en HTML5 es mucho menor. Pero ojo, que para hacer un documento HTML5 no es simplemente cambiar el doctype, sino que se entiende que debemos usar además un buen conjunto de etiquetas propias, como las ARTICLE, SECTION, HEADER, FOOTER, etc.

También se encuentra el META charset, que se recomienda usar siempre con el valor UTF-8, al menos para todos los idiomas occidentales.

Mitos y leyendas HTML5

¿Cuáles son las verdades y mentiras del HTML5? hay mucho marketing alrededor de HTML5 y mucha información que no pasa de ser falso mito.

- **HTML5 hizo morir a Flash** Si bien es cierto que HTML5 le ha comido mercado a Flash, aun no es competencia seria en muchos ámbitos como son los juegos o los banner. Sin duda Flash ha perdido mucho mercado y en el mundo de la web la tendencia es desaparecer, sin embargo sigue siendo importante por ejemplo para los juegos. Lo más relevante es que ahora no necesitamos Flash para hacer muchas cosas que han quedado implementadas en el HTML5, aunque también mucha culpa de usarse cada vez menos Flash es que no es compatible con sistemas móviles.

- **HTML5 no funciona en muchos navegadores** La mayoría de las partes de HTML5 ya son perfectamente compatibles con cualquier navegador. Los modernos tienen una adopción de los estándares muy amplia, el problema son los navegadores antiguos. Sin embargo, existen maneras de conseguir compatibilidad en HTML5 en navegadores antiguos por medio de Javascript. Tendremos una clase completa del curso para explicar ésto.

- **Sólo funciona con CSS3** CSS3 amplía los lenguajes CSS1 y CSS2. Por tanto cuando programamos con CSS3 en realidad muchas de las cosas que hacemos pertenecen a los CSS más antiguos. CSS3 no sustituye al CSS tradicional, solo lo amplía con nuevas características.

- **HTML5 arruina la accesibilidad** Esto no tiene nada que ver con HTML, quizás hay

personas que piensan eso por elementos como canvas, que no tiene una alternativa para mejorar la accesibilidad. Sin embargo cuando se hizo HTML5 se hizo con mucho énfasis en la accesibilidad.

HTML5 y tecnologías asociadas

Se realiza un rapidísimo recorrido a HTML5 desde el punto de vista de las tecnologías que vienen asociadas. CSS3, MathML, SVG, Web Open Font Format, WebGL, y un montón de APIs Javascript. Como decimos, muchas de estas tecnologías las vamos a ver más adelante con mayor detalle, en siguientes sesiones de este curso.

Nuevas fronteras de HTML5

La sesión termina explicando las nuevas fronteras que nos traerá HTML5 en un corto espacio de tiempo o a medio plazo. Webapps para dispositivos, aplicaciones para ejecutarse en el core de los navegadores, desaparición de las versiones antiguas de Internet Explorer, sitios responsive, etc. No dejes de ver la charla para encontrar más información.

Acabamos este artículo con el [link para descarga del documento usado como diapositivas en la clase](#).

A continuación encuentras el vídeo de la clase emitida.

Para ver este vídeo es necesario visitar el artículo original en:
<http://desarrolloweb.com/articulos/video-que-es-html5.html>

Este artículo es obra de *Sandra Guerrero*

Fue publicado por primera vez en 15/04/2014

Disponible online en <http://desarrolloweb.com/articulos/video-que-es-html5.html>

Video Web Semántica

Este vídeo es una exposición dedicada a las características y ventajas de la Web Semántica.

Esta es la segunda clase del Curso Gratuito 5 días de HTML5, impartida por Zuriñe Menendez @zmencay, en la que nos habló de la Web Semántica. Fue una exposición en la que pudimos apreciar la importancia de la Web Semántica y cómo HTML5 ha posibilitado su implantación.

Durante la clase se remarcaron diversos conceptos, comenzando por la propia semántica, y cómo aplicarla en un diseño, desde el punto de vista estético, pero también en términos de código HTML, por medio de la selección correcta de etiquetas según su significado. También repasamos otros asuntos importantes como los microformatos, que nos ayudan a aplicar semántica en piezas concretas de contenido. Vimos al final unos ejemplos en código para poder concretar todo el conocimiento en un documento HTML.



Por qué Web semántica

Semántico se refiere a cargado de significado. La Web semántica hace referencia a aquella que nos consigue comunicar un significado cuando la leemos las personas, pero además también cuando la interpretan las máquinas.

En este primer punto Zuriñe nos indicó los motivos por los que existe la Web semántica y los objetivos que persigue. Entender para qué sirve el sitio, saber cuál es nuestro objetivo, comprender lo que el diseñador desea transmitir y codificar correctamente.

Desde el punto de vista de las personas, cargar de significado a una web comienza por tener un diseño apropiado, aplicando colores, formas y estética que nos ayude a entender para qué están las cosas. Si algo es un enlace no debe verse como un botón. Si hay un disparador de una acción "cancelar" no debe verse igual que uno que sirve para "aceptar".

Pero no solo la web la leen las personas, también hay máquinas que consumen el contenido de

las páginas. Por ejemplo, Google o cualquier robot que analice el contenido de la Web. Incluso hay personas que, por problemas de visión, no son capaces de asimilar el contenido de una web de manera visual. Para todos ellos la semántica se tiene que aportar en función a otros elementos y ahí es donde cobra valor cargar de significado al código HTML que escribimos.

Así pues, hay etiquetas en el HTML que aportan un significado a aquello que se está incluyendo en un documento, como HEADER, FOOTER, ARTICLE, etc. Esto es algo que se ha tomado con mucha importancia dentro de HTML5 y de hecho, la mayoría de las nuevas etiquetas disponibles en el lenguaje son etiquetas que aportan un significado a aquello que incluyen en un documento HTML.

Semántica y SEO

SEO significa Search Engine Optimization y son un conjunto de técnicas que sirven para hacer que una página aparezca mejor posicionada en buscadores. Pues bien, según nos explicó Zuriñe, la semántica nos ayuda a realizar un mejor SEO. Para ello tenemos que procurar escribir un código limpio y usar ciertas etiquetas o selectores CSS que nos ayuden a identificar el contenido por su función.

Los buscadores son como usuarios ciegos, porque no saben cómo se ve una web, solo saben leer su código. Existen etiquetas que directamente le hacen entender qué es lo que están incluyendo, como un encabezamiento, un navegador, un contenido accesorio en el lateral, etc. Cualquier ayuda que le ofrezcamos a un buscador para interpretar el contenido de nuestra página y saber qué función tiene cada parte y qué importancia dentro del conjunto del documento, es positiva.

Etiquetas semánticas

Sirven para identificar lo que va asociado a ellas. Es de lo que estamos hablando todo el rato. Son etiquetas como HEADER, SECTION, ARTICLE, FOOTER, NAV, etc. Estas etiquetas sirven para dar entender determinadas cosas.

Nota: Otro artículo donde se habla de semántica en el documento HTML es [Etiquetas semánticas del HTML5](#).

Hay un montón de etiquetas que han surgido para dar cabida a la web semántica. También nos señalan en la charla las etiquetas para formularios para ayudar a la entrada de datos.

Microformatos

Este es otro de los asuntos que vimos en la clase y nos ayudó a entender cómo se pueden entregar informaciones adicionales a los navegadores, esta vez a través de clases CSS.

Existen microformatos para muchos asuntos interesantes, por ejemplo crear tarjetas de visita en una web. Por medio de una serie de clases CSS (class) podemos indicar qué es cada parte de la tarjeta de visita, como el nombre el email, la dirección web, teléfonos, etc. En la charla lo

vimos con un ejemplo.

Ejemplo de página HTML5 semántica

En el último tercio de clase nos dedicamos a analizar un código HTML e identificar las partes semánticas. Veremos por medio de código diversos asuntos clave en esta charla, desde el uso de etiquetas nuevas del HTML5 que aportan un significado hasta el uso de los microformatos.

Además pudimos ver componentes típicos de una página en HTML5, desde el documento básico hasta la inclusión de scripts para compatibilidad.

Vídeo y materiales para descarga

Los materiales usados durante la presentación, diapositivas y el ejemplo se pueden [descargar en un archivo comprimido](#).

Para encontrar todo el contenido de esta clase, lo mejor es que puedas ver el vídeo de la grabación.

Para ver este vídeo es necesario visitar el artículo original en:
<http://desarrolloweb.com/articulos/video-web-semantica.html>

Este artículo es obra de *Zuriñe Menendez*
Fue publicado por primera vez en 16/04/2014
Disponible online en <http://desarrolloweb.com/articulos/video-web-semantica.html>

Vídeo Conociendo CSS3

Clase de introducción a la maquetación web y las posibilidades del lenguaje CSS, en concreto CSS3 con HTML5.

Esta es una clase emitida en directo, la sesión 03 del curso gratuito de HTML5 que estamos realizando en EscuelaIT, impartida por Daniel Martínez @Wakkos. Todos conocemos a Daniel y sabemos lo mucho que nos puede enseñar, y de manera tan amena como él lo hace, así que no hay realmente mucha más introducción que hacer, solo recomendar ver el vídeo que encontrarás al final de este artículo.

CSS es el lenguaje con el que aplicamos estilo a una web, con HTML definimos el contenido y con CSS le decimos cómo debe de presentarse ese contenido. Puede parecer que aplicar CSS es simplemente saberse unos cuantos atributos y selectores y escribirlos sin mucho orden, pero como nos dice Daniel, **la parte más importante para el éxito de un CSS es la ESTRUCTURACIÓN**.

En ese sentido esta clase no solo nos da un recorrido a las características de las CSS3, sino que el mayor valor lo encontraremos gracias a una enorme cantidad de consejos y de pautas para escribir mejor código CSS, mantenible, escalable, etc.



Fundamentos de las CSS

Daniel comienza ofreciendo una serie de fundamentos de las hojas de estilo en cascada, enseñándonos como integrar un archivo CSS en una página web. Vemos cómo podemos aplicar estilo a elementos que se seleccionan por medio de diversos selectores.

El trabajo de un desarrollador front-end consiste en aplicar CSS de una manera que permita un correcto mantenimiento del proyecto, pero tal como es el CSS actual ¿qué es lo que un desarrollador echa en falta?

Preprocesadores CSS

Lo que nos falta para poder estructurar un código CSS es lo que nos ofrecen los preprocesadores, nos referimos a variables, includes, etc. Para demostrarnos esta tarea Daniel nos pone varios ejemplos y nos demuestra cómo él **organiza el CSS de un proyecto ayudado con el preprocesador Less**.

El objetivo de la clase no es aprender preprocesadores, pero sí saber por dónde tenemos que continuar nuestro aprendizaje cuando ya dominemos las hojas de estilo en cascada.

Diseño adaptable

De nuevo, Daniel nos ayuda a entender los fundamentos de una de sus grandes pasiones, que es el diseño adaptable, lo que a veces conocemos como "Responsive". No se trata de maquetar un sitio web pensando en nuestro ordenador y nuestra pantalla, sino pensar que **el sitio debe poder ser navegable por cualquier tipo de dispositivo**, tablet, móvil, etc.

Así pues, daniel nos ofreció una clase maestra sobre cómo debemos comenzar a trabajar un diseño adaptable, cómo aplicar las mediaqueries y cómo usar nuevas características de las CSS3 para la maquetación y solventar diversos problemas típicos que nos encontramos en el día a día. Cómo usar unidades relativas y cómo calcular las dimensiones de los elementos, también de manera relativa.

Ejemplos y vídeo de la clase

A lo largo de la clase se realizó un ejemplo de maquetación responsive completo, en el que pudimos apreciar algunas de las claves para poder adaptar el diseño a dispositivos y ordenadores con pantallas grandes. Puedes [descargar el código de este enlace](#).

El vídeo lo encuentras a continuación.

Para ver este vídeo es necesario visitar el artículo original en:
<http://desarrolloweb.com/articulos/video-conociendo-css3.html>

Este artículo es obra de *Daniel Martínez*
Fue publicado por primera vez en 17/04/2014
Disponible online en <http://desarrolloweb.com/articulos/video-conociendo-css3.html>

Vídeo APIs Javascript de HTML5

Qué son las API de HTML5, qué posibilidades nos ofrecen y un par de ejemplos sobre cómo se pueden invocar esas API Javascript para implementar funcionalidades de navegadores y sitios modernos.

Esta es la 4ª clase del [Curso Gratuito de HTML5 que estamos impartiendo en EscuelaIT](#), dedicada a las API de HTML5 y cómo Javascript nos puede ayudar facilitando la incorporación de funcionalidades avanzadas en un sitio web. Fue impartida por Reinaldo Aguilera @reialguilera.

A lo largo de las clases anteriores de este curso hemos podido conocer dos de los tres pilares fundamentales de HTML5, que es el propio lenguaje HTML y CSS. Pero no podemos hablar de HTML5 sin darle la importancia que se merece a Javascript, ya que este lenguaje se ha introducido de lleno en el "core" (núcleo) de los navegadores y hoy en día nos ofrece gracias a los motores Javascript rendimiento y posibilidades similares a los lenguajes de propósito general.

Javascript es la estrella de HTML5, ya que la mayoría de las novedades que han aparecido en esta nueva oleada de estándares abiertos está basada en Javascript y lo que se llaman las API de HTML5. En esta clase haremos un rápido recorrido a las distintas API disponibles, así como realizaremos un par de ejemplos prácticos, con el API de geolocalización y con el API de audio/vídeo.



Al final de este artículo podrás ver el vídeo de la clase impartida en directo por videoconferencia. Antes te ofrecemos un rápido resumen.

Lenguajes predeterminados en HTML5

Antes de comenzar a entrar de lleno en materia se explicaron a manera de repaso algunas de las características de HTML5, para hacer luego hincapié en los lenguajes predeterminados en esta versión del lenguaje.

Antes para incluir estilos CSS usábamos una etiqueta como esta:

```
<link type="text/css" rel="stylesheet" href="estilos.css">
```

Ahora ya se entiende que el lenguaje para aplicar estilos a la página va a ser siempre CSS, por lo que la nueva forma de la etiqueta sería esta:

```
<link rel="stylesheet" href="estilos.css">
```

Javascript como lenguaje de scripting ha tenido un similar tratamiento. Antes en HTML4 especificábamos el lenguaje que se iba a usar.

```
<script type="text/javascript" src="mi_javascript.js"></script>
```

Pero ahora con HTML5 se define de manera predeterminada Javascript como el lenguaje para usar en el scripting de una página web, por lo que no necesitamos especificarlo y nos quedaría la etiqueta de esta manera:

```
<script src="mi_javascript.js"></script>
```

Aclaremos que ningún navegador te va a mostrar un error si indicas los lenguajes predeterminados, simplemente te los puedes ahorrar. Quizás sea una diferencia sutil a la que no se le deba dar más importancia, pero es interesante para entender cómo estos lenguajes forman parte íntima del HTML5.

APIs HTML5 en Javascript

Luego Reinaldo realiza un rápido pero interesante recorrido a las tecnologías HTML5 y donde vemos que la mayoría de ellas son nuevas características en las que necesitamos Javascript como lenguaje de programación.

Más del 70% de las nuevas características de HTML5 son lo que llamamos las API. En plan general, para que nos entendamos, un API es un conjunto de funciones ya listas que nos ofrece un sistema. Es algo relacionado con cualquier lenguaje de programación, no solo con HTML5 y Javascript. Esas funciones nos sirven para operar con diversos procesos útiles que vamos a necesitar para interactuar con cualquier tipo de sistema.

En el contexto de un navegador existen muchas utilidades, las cuales que podemos invocar por medio de APIs HTML5, no solo para acceder a los elementos de una página, sino también a sus periféricos. Para todo ello se han desarrollado APIs en Javascript que nos permiten acceder a sistemas de almacenamiento, geolocalización, comunicación con servicios web, periféricos como la cámara, elementos multimedia y un largo etc.

Ejemplos de desarrollo de APIs HTML5

En la segunda mitad de la clase Reinaldo nos ofreció unos cuantos ejemplos sobre trabajo con Javascript aprovechando las características de HTML5, centradas en las API de Geolocalización y Multimedia (Audio/Vídeo).

En geolocalización vimos cómo podemos preguntar al navegador la posición geográfica donde se encuentra el usuario. Nos ofrece una lista de datos, como latitud, longitud y altitud, que luego podemos usar para cualquier cosa, como ofrecerle servicios cercanos a donde se encuentra ese usuario. En el ejemplo que se mostró en la clase vimos cómo esa información puede ayudarnos a mostrar un mapa de Google Maps donde podamos localizar al usuario.

Luego se vio un ejemplo sobre cómo se insertan elementos de audio y vídeo en una web con HTML5 y luego cómo podemos controlar la reproducción de los mismos por medio de controles personalizados, a los que aplicamos funcionalidad usando el API de HTML5 para multimedia. Así podemos colocar cualquier botón (o enlace) creado por nosotros y que nos sirva, por ejemplo, para reproducir o pausar un audio. O cómo podemos extraer información adicional de un vídeo para mostrar asuntos diversos, como el segundo en el que estamos reproduciendo en cada instante.

Ejemplos realizados diapositivas y vídeo de la clase

Estamos esperando los ejemplos y diapositivas usadas por Reinaldo, en cuanto las tengamos las publicaremos aquí mismo.

El vídeo de la clase lo puedes ver ya mismo en Youtube, a continuación.

Para ver este vídeo es necesario visitar el artículo original en:

<http://desarrolloweb.com/articulos/video-api-javascript-html5.html>

Este artículo es obra de *Reinaldo Aguilera*

Fue publicado por primera vez en 22/04/2014

Disponible online en <http://desarrolloweb.com/articulos/video-api-javascript-html5.html>

Vídeo compatibilidad HTML5

Cómo podemos aplicar compatibilidad con HTML5 en navegadores antiguos con la librería Javascript Modernizr. 5 prácticas esenciales aplicando graceful degradation.

Esta es la clase 05 de los "[5 días de HTML5](#)", que termina la serie de sesiones dedicadas a los estándares de la web abierta en el curso gratuito de EscuelaIT. Estuvo dedicada a la compatibilidad de las características de HTML5 todo tipo de navegadores con y fue impartida por Miguel Angel Alvarez @midesweb.

Cuando realizamos un sitio web queremos que sea lo más impactante posible, que utilice la mayor cantidad de recursos avanzados y presentes en las webs modernas, pero también queremos que se pueda usar desde navegadores antiguos, quizás no con la misma experiencia de usuario, pero sí al menos una suficientemente digna.

Para ello aplicaremos HTML5 y alguna de las técnicas que vamos a conocer en esta clase y que ahora os resumimos. Al final del texto encontrarás el vídeo de la clase, así como los ejemplos y las diapositivas utilizadas.



Plantillas front-end

Como primer item de esta clase nos referimos rápidamente a la [plantilla front-end "Boilerplate"](#) que es un conjunto de ficheros que conforman un punto de partida común para cualquier proyecto front-end. En realidad tienes un archivo index.html y una serie de elementos adicionales que deberían estar presentes en cualquier sitio web, como hojas de estilo css, archivos de Javascript, librerías Javascript, etc.

Podemos usarla como base para crear proyectos sólidos, donde estamos usando aquellos módulos o piezas más deseables. Pero ojo, que no es para usarla tal cual. Lo bueno de plantillas como éstas es que te permite saber qué cosas podrían ser deseables en un proyecto, de modo que de entrada ya sabes qué "ingredientes" ponerle a tu sitio web. Sin embargo, todo lo que

está en Boilerplate tiene un motivo para estar ahí, por lo que lo bonito de usar este paquete es que tengas la iniciativa y curiosidad para explorarlo bien y saber qué es cada cosa, tener libertad para cambiarlas, configurarlas quitar unas y poner otras y en definitiva, tomar un control consciente sobre lo que estás haciendo.

Entre algunas de las cosas que Boilerplate se preocupa es de la compatibilidad con navegadores antiguos y las herramientas que utiliza son justamente las que os explicamos en esta clase.

Por qué nos preocupan los navegadores antiguos

Cuando trabajamos con HTML5 estamos incorporando tecnologías muy diversas, para items como semántica, CSS3, APIs javascript para el control de dispositivos, almacenamiento de datos en el navegador, etc. Cada una de esas tecnologías tiene un estado de especificación determinado y a lo largo del tiempo va cambiando, hasta que se convierten en candidatas a recomendación y es cuando los navegadores las empiezan a implementar en su "core".

Existen en HTML5 muchas tecnologías con grados distintos de especificación y también existen muchos navegadores en el mercado y todo ello nos provoca que haya un abanico muy grande de situaciones, que además van variando a lo largo del tiempo.

Para resolver nuestras dudas y saber hasta que punto una de las cosas que incorpora HTML5 está o no adoptada por la mayoría de los navegadores, tenemos el sitio caniuse.com ("can i use", puedo usar), que nos ofrece un listado de características y el grado de implementación en navegadores.

Técnicas para aplicar compatibilidad

¿Debemos adaptarnos a navegadores antiguos? ¿O mejor adaptarnos a los navegadores modernos? realmente lo que buscamos es la compatibilidad de un sitio en los dos entornos, de modo que al menos se pueda usar correctamente. Para ello tenemos principalmente dos técnicas que podemos seguir para que el sitio o la aplicación web se adapte a todo tipo de navegadores.

Progressive enhancement: Un producto sencillo con intención de ser altamente compatible al que le aplicas mejoras para aquellos navegadores que las soportan. Es decir, partes de un sitio en el que te has preocupado por ofrecer una funcionalidad y experiencia de usuario básica, de modo que funcione en todo el panorama de navegadores. Luego a ese sitio le vas aplicando mejoras que harás solamente disponibles para los navegadores que las pueden interpretar.

Graceful degradation: Un producto complejo pensando en navegadores modernos en el que aplicas alternativas para aquellos navegadores que no son compatibles. De entrada haces tu sitio pensando en las características más modernas y deseables. Luego, para los navegadores que no las soportan creas alternativas o respuestas a fallos que posibiliten el uso de tu aplicación o sitio web.

Nosotros animamos más a implementar Graceful degradation porque el producto que vas a conseguir está más adaptado a los tiempos que corren. No te interesa diseñar o desarrollar

para navegadores antiguos, porque así es difícil que tu producto llegue a tener un resultado impactante, acorde con los tiempos que corren. Aplicar graceful degradation te permite aprovechar las características del HTML5 y definir las alternativas necesarias para ofrecer a los usuarios de navegadores antiguos una experiencia de usuario digna.

Ten en cuenta también que no todo en la web, y en el uso de HTML5, debe requerir compatibilidad necesariamente. Aceptamos que los navegadores tienen características distintas y que una web no debe necesariamente verse igual en todos los navegadores que la puedan llegar a reproducir. La propia corriente del diseño "responsive", que vimos en la clase 3 de este curso, por la cual se procura que los sitios web se adapten a todas las dimensiones y resoluciones de las pantallas, te implica que una página no se va a ver nunca igual en dispositivos o pantallas diferentes. Lo que hacemos es adaptarnos. De eso también se trata la compatibilidad del HTML5 que queremos explicaros.

Por qué Modernizr

[Modernizr es una librería de Javascript](#) que nos sirve para realizar compatibilidad con características de HTML5. Nos permite implementar "graceful degradation" que es la técnica de compatibilidad que encontramos más acertada.

También permite algo que es importante y tendencia desde hace muchos años, que es la detección de capacidades. Antiguamente, para saber si podemos hacer una cosa o no, detectábamos si el navegador del usuario era Internet Explorer, o Firefox (o Opera, Chrome, o más antiguamente, Netscape). Sin embargo hace tiempo los desarrolladores se dieron cuenta que no era una buena práctica. Observa que, lo que un día puede no soportar un navegador, al cabo de un tiempo y en una versión más moderna, puede que esté perfectamente incorporado y soportado.

Por ello, lo que se hace con Modernizr y en el mundo del desarrollo está más comúnmente aceptado, es preguntar por las características. Osea, al navegador le pregunto si tiene el elemento tal o cual y si están disponibles. Si era así quiere decir que puedo usar esas características sin preocupaciones. Si no era así quiere decir que no se pueden usar esas partes del HTML5 en el navegador del usuario y por tanto debo implementar vías alternativas que funcionen para él.

Prácticas de compatibilidad

Dedicamos 45 minutos, osea 3 cuartas partes de la clase de la hora de duración, a hacer prácticas relacionadas con la compatibilidad de navegadores y con el uso de Modernizr. Entre todas las cosas que puedes hacer con la librería nos centramos en varias que resultan bastante útiles, pero que no revisten demasiada complejidad. De ese modo creemos que es más sencillo entender la potencia e idoneidad de Modernizr como aliado a HTML5 en cuanto a compatibilidad se refiere.

1.- Compatibilidad con etiquetas semánticas: vimos que los navegadores antiguos no aceptan las etiquetas nuevas como header, nav, article, etc. Solamente con incluir la librería Modernizr (si la descargamos con el componente html5shiv) permite que esas etiquetas sean comprendidas por los navegadores como Internet Explorer 8 o inferiores.

2.- Estilos alternativos: vimos cómo podemos aplicar CSS diferente si un navegador no soporta una característica determinada. Esto lo conseguimos gracias a las class de CSS que nos incorpora Modernizr al documento cuando encuentra o no ciertas características.

3.- Mostrar un mensaje de error cuando no encuentra algo: esta práctica nos permite mostrar unos contenidos u otros cuando se encuentra o no con ciertas características de HTML5. Esto nos puede ayudar para mostrar mensajes de error que solo verán usuarios de navegadores no compatibles o quizás ocultar algunos componentes que no tenga sentido que los vean los usuarios de ciertos navegadores que no los soportan.

4.- Cargar un polyfill: cómo incorporar una librería para los navegadores que no tienen cierta funcionalidad. En un primer caso mostramos cómo implementar un fallback para el elemento canvas para Internet Explorer.

5. Cargar un polyfill con Modernizr.load(): Acabamos la clase práctica viendo cómo nos facilita la labor Modernizr.load y la carga condicionar de scripts Javascript con un ejemplo sencillo para aplicar la utilidad conocida como "placeholder" en navegadores que la soportan, por medio de jQuery.

Ejemplos, diapositivas y vídeo de la clase

Los [ejemplos de la clase los puedes ver y descargar en github](#). Date cuenta que es todo el Boilerplate, en el que hemos modificado tan solo algunos archivos.

Las [diapositivas también las puedes descargar](#), por si te apetece verlas con calma o tenerlas para tu documentación.

Por último puedes aquí reproducir el contenido completo de la clase.

Para ver este vídeo es necesario visitar el artículo original en:
<http://desarrolloweb.com/articulos/video-compatibilidad-html5.html>

Este artículo es obra de *Miguel Angel Alvarez*
Fue publicado por primera vez en 23/04/2014
Disponible online en <http://desarrolloweb.com/articulos/video-compatibilidad-html5.html>