

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
Gestor de contenidos (Alfresco)
Aplicaciones híbridas

Tareas programadas (Quartz)
Gestor documental (Alfresco)
Inversión de control (Spring)

Control de autenticación y
acceso (Spring Security)
UDDI
Web Services
Rest Services
Social SSO
SSO (Cas)

JPA-Hibernate, MyBatis
Motor de búsqueda empresarial (Solr)
ETL (Talend)

Dirección de Proyectos Informáticos.
Metodologías ágiles
Patrones de diseño
TDD

BPM (jBPM o Bonita)
Generación de informes (JasperReport)
ESB (Open ESB)



AdictosAlTrabajo.com

"¡La primavera ha venido,
nadie sabe cómo ha sido!"



autentia
Soporte a desarrollo informático

Entra en Adictos a través de



E-mail

Contraseña

Entrar

Regístrate
Olvidé mi contraseña

[Inicio](#) [Quiénes somos](#) [Formación](#) [Comparador de salarios](#) [Nuestros libros](#) [Más](#)

» Estás en: [Inicio](#) [Tutoriales](#) [Introducción a HTML5.](#)



Jose Manuel Sánchez Suárez

Consultor tecnológico de desarrollo de proyectos informáticos.

Puedes encontrarme en [Autentia](#): Ofrecemos servicios de soporte a desarrollo, factoría y formación

Somos expertos en Java/J2EE

[Ver todos los tutoriales del autor](#)

Catálogo de servicios Autentia



Fecha de publicación del tutorial: 2010-10-09

Tutorial visitado 18.825 veces [Descargar en PDF](#)

Introducción a HTML 5.

0. Índice de contenidos.

1. Introducción.
2. A nivel de etiquetado.
3. A nivel de javascript.
4. Soporte para media.
5. Soporte para css3.
6. ¿Una página de ejemplo?.
7. ¿Cuál es el estado actual de la especificación y las implementaciones?.
8. Referencias.
9. Conclusiones.

1. Introducción

En este tutorial vamos a realizar una breve introducción a las novedades que traerá la versión 5 de html. Sí, el título está bien escrito, hablamos de html, aunque también habrá la correlativa versión en xhtml, las diferencias entre ambas solo vendrán dadas por las restricciones típicas de un documento xml bien formado.

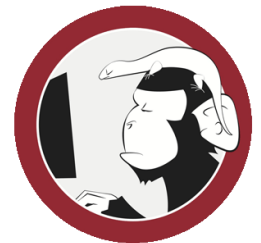
Visto lo anterior, lo primero que debemos saber es que no se trata solo de una actualización de la colección de marcas que podemos utilizar dentro del etiquetado de nuestras páginas. En versiones anteriores, frente a una nueva especificación, o al pasar a xhtml, las diferencias con las versiones anteriores venían dadas por la supresión de etiquetas en desuso y la incorporación de nuevas etiquetas que ampliaban funcionalidad. La nueva versión del estándar, además de lo anterior, viene de la mano de nuevas y muy interesantes características, como las que veremos a continuación.

Lo que debe quedar claro es que estas características, como la estandarización de CSS3, geolocalización, soporte de audio y video o la ampliación de los métodos de acceso al árbol DOM en javascript, no son estrictamente del estándar HTML5, pero se asocian con la nueva versión, le dan mucha más potencia y la hacen mucho más atractiva.

2. A nivel de etiquetado.

Tenemos las siguientes características destacables:

- Nuevas etiquetas:
 - hay una etiqueta específica para el menú de navegación, **nav**
 - hay etiquetas para distinguir secciones y artículos, pensadas para páginas de noticias o blogs, **section** y **article**
 - por fin hay un **footer**, siempre escribiendo la cabecera y hasta ahora no teníamos pie de página.
 - pero también hay un **header**, distinto del head, que serviría para insertar contenido visible en la cabecera de la página,
 - **mark**, indicará un texto resaltado,
 - **time**, indica una fecha u hora,
 - **meter**, representa un valor numérico,
 - **progress**, indica el estado de una tarea, lo que es una barra de progreso,
- Nueva colección de tags que añaden contenido semántico:
 - **aside**, anotaciones que complementan el texto pero quedan fuera del texto,
 - **figure**, se podrá asociar contenido multimedia a un título o legend,
- Nuevos tipos de componentes de formulario:
 - **input type="date | time | email | url | tel | search | ..."**, que permite introducir distintos tipos de contenidos produciendo una validación automática de los mismos, como un email válido; los input de selección de fecha y hora se pueden convertir en un date picker en función del navegador,
 - validación de campo requerido mediante el atributo **required**, así como mínimo y máximo,
 - **input spinner**, a través del atributo step y type,
 - **sliders**, a través de type="range"
 - validaciones con expresiones regulares
 - restricción de valores en un input a través de un **dataList**
 - el atributo **placeholder** permitirá asignar un texto por defecto sombreado a los inputs (el típico de Google en el



Síguenos a través de:



Últimas Noticias

» 2015: ¡Volvemos a la oficina!

» Curso JBoss de Red Hat

» Si eres el responsable o líder técnico, considérate desafortunado. No puedes culpar a nadie por ser gris

» Portales, gestores de contenidos documentales y desarrollos a medida

» Comentando el libro Start-up Nation, La historia del milagro económico de Israel, de Dan Senor & Salu Singer

[Histórico de noticias](#)

Últimos Tutoriales

» Técnicas de realización de entrevistas

» Imprimiendo documentos Office y PDF existentes con Java en entorno Windows. Batch & Print

» Enfrentate con éxito a la crisis de la hoja en blanco

» Patrones de diseño en Hadoop: Patrón Partitioner

» Movimientos de la cámara

campo de búsqueda de firefox),

Hay etiquetas que han caído en desuso y otras características también interesantes que no he incluido, podéis consultar todas las etiquetas en forma de tabla periódica en el siguiente enlace

- <http://websitesetup.org/html5-periodical-table/>

A continuación un ejemplo de documento html5:

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>HTML5 test</title>
5    <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type" />
6  </head>
7  <body>
8    <header>
9      <nav>
10        <ul>
11          <li><a href="#">Inicio</a></li>
12          <li><a href="#">Pedidos</a></li>
13          <li><a href="#">Facturas</a></li>
14          <li><a href="#">Administración</a></li>
15        </ul>
16      </nav>
17      <form>
18        <input type="search" name="q" placeholder="Búsqueda" />
19      </form>
20    </header>
21    <section>
22      <article>
23        <header>
24          <h2>Noticia #1</h2>
25        </header>
26        <section>
27          <p>Texto de la noticia #1...</p>
28        </section>
29      </article>
30      <article>
31        <header>
32          <h2>Noticia #2</h2>
33        </header>
34        <section>
35          <p>Texto de la noticia #2...</p>
36        </section>
37      </article>
38    </section>
39    <aside>
40      <form>
41        <p><label>Nombre: <input name="name" required></label></p>
42        <p><label>E-mail: <input name="email" type="email" required></label></p>
43        <p><label>Edad: <input type="number" min="0" max="99" step="1" value="33"></label></p>
44        <p><label>Edad: <input type="range" min="0" max="10" step="2" value="6"></label></p>
45        <p><label>URL: <input name="url" type="url"></label></p>
46        <p><label>Comentario: <textarea name="comment" required></textarea></label></p>
47        <p><input type="submit" value="Enviar"></p>
48      </form>
49    </aside>
50    <footer>
51      <p>Copyleft 2010</p>
52    </footer>
53  </body>
54 </html>

```

Últimos Tutoriales del Autor

» Generación de vistas HTML5 con el soporte de JSF2: pass through

» Extendiendo las reglas de SonarQube con Xpath

» Solución de problemas comunes con la integración de maven en Eclipse Luna.

» Integración de SonarQube en Eclipse.

» Monitorización de Apache Tomcat con psi-probe.

Categorías del Tutorial

 Navegadores

 HTML5

3. A nivel de javascript.

Tenemos las siguientes características destacables:

- nuevos métodos de acceso a los nodos del árbol DOM que incorporan las facilidades que, hasta ahora, solo teníamos con librerías de javascript como prototype.js o jQuery. Son métodos como:
 - **document.getElementsByClassName**: que permite recuperar todos los nodos del árbol DOM con el mismo estilo de clase,
 - **document.querySelectorAll** y **document.querySelector**: viene a sustituir lo que en jQuery es la propia función jQuery o el alias \$, heredado de prototype, que permite realizar consultas al árbol DOM para recuperar nodos con distintos selectores o combinación de los mismos,

```

1  <script language="javascript">
2    document.getElementsByClassName("style1");
3
4    document.querySelectorAll("ul li:nth-child(odd)");
5
6    document.querySelector("table.test > tr > td");
7  </script>

```

- soporte **drag & drop** nativo.

```

1  <script language="javascript">
2    var dragItems = document.querySelectorAll('[draggable=true]');
3
4    for (var i = 0; i < dragItems.length; i++) {
5      addEvent(dragItems[i], 'dragstart', function (event) {
6        event.dataTransfer.setData('Text', this.id);
7      });
8    }
9
10   var bin = document.querySelector('#dropable');
11
12   addEvent(bin, 'drop', function (e) {
13     if (e.stopPropagation) e.stopPropagation();
14
15     var el = document.getElementById(e.dataTransfer.getData('Text'));
16     el.parentNode.removeChild(el);
17
18     return false;
19   });

```

```
20
21 </script>
```

- **web storage** o almacenamiento local de datos, que viene a solventar las limitaciones de almacenamiento de las cookies, permitiendo almacenar mucha más información en el cliente dando pie a trabajar con aplicaciones web como si fuesen aplicaciones de escritorio en modo offline, para ello también disponemos de un API de caché que permite indicar a través de un fichero de manifest, qué recursos deben ser cacheados,

```
1 <script language="javascript"> ?
2
3 window.localStorage['value'] = document.getElementById("comments");
4
5 document.getElementById("comments") = window.localStorage['value'];
6
7 </script>
```

- existirá la posibilidad de disponer de **bases de datos SQL en cliente**, ¿habíais pensado alguna vez en la posibilidad de tener que escribir sentencias SQL en javascript?, pues sí; ¿que será lo siguiente? ¿cuánto tardaremos en tener una versión de hibernate para javascript? ;-),

```
1 <script language="javascript"> ?
2 var db = window.openDatabase("Items", "1.0");
3 db.transaction(function(tx) {
4   tx.executeSql("SELECT * FROM Item", [], successCallback, errorCallback);
5 });
6 </script>
```

- **geolocalización**: mediante una simple invocación a un método javascript nativo, podremos obtener la localización del equipo cliente, previa aceptación del usuario a facilitar dicha información,

```
1 <script language="javascript"> ?
2 if (navigator.geolocation) {
3   navigator.geolocation.getCurrentPosition(function(position) {
4     var lat = position.coords.latitude;
5     var lng = position.coords.longitude;
6   });
7 }
8 </script>
```

- **web workers**: permite la ejecución de scripts en segundo plano, son procesos de larga duración y pueden consumir mucha memoria que hasta ahora se podían implementar con la función window.setTimeout(). Los workers tienen las siguientes ventajas: se ejecutan en threads separados, de forma concurrente, no bloquean la interfaz de usuario, pueden ser dedicados (al tab) o compartidos por varios tabs o por la ventana e, incluso, podrían persistir al cierre de la misma.

```
1 <script language="javascript"> ?
2 var worker = new Worker('myjavascript.js');
3 worker.onmessage = function(event) { alert(event.data); };
4 </script>
```

- **web sockets**: es la implementación de la comunicación bidireccional entre el cliente y el servidor, lo que podemos denominar push o comet, y que antes hacíamos con Ajax a través del objeto XMLHttpRequest. Con HTML5 ya no será necesario su uso puesto que provee de un mecanismo estándar para realizar dicha comunicación invocando directamente a un método send() que remitirá la información al servidor.

```
1 <script language="javascript"> ?
2 var socket = new WebSocket(location);
3 socket.onopen = function(event) {
4   socket.postMessage("Hola mundo, WebSocket");
5 }
6 socket.onmessage = function(event) { alert(event.data); }
7 socket.onclose = function(event) { alert("closed"); }
8
9 </script>
```

- **notificaciones**: huyendo de los alerts, permiten notificar al usuario eventos que se producen en una ventana fuera del navegador, con lo que, aunque esté minimizada la misma se mostrarán al usuario,

```
1 <script language="javascript"> ?
2 function RequestPermission (callback) {
3   window.webkitNotifications.requestPermission(callback);
4 }
5
6 function showNotification(){
7   if (window.webkitNotifications.checkPermission() > 0) {
8     RequestPermission(showNotification);
9   }
10  else {
11    window.webkitNotifications.createNotification("image.png",
12      "Titulo", "Cuerpo").show();
13  }
14 }
15
16 </script>
```

4. Soporte para media.

- Soporte para audio y video: que nos permitirá insertar contenido media sin necesidad de hacer uso de flash,

```
1 <audio src="mp3/sonido.mp3" type="audio/mp3" controls> ?
2   Your browser does not support the <code>audio</code> element.
3 </audio>
4 <br />
5 <video src="video/presentacion.ogg" autoplay />
```

Es bastante interesante la compatibilidad con navegadores con versiones anteriores, de modo que si el navegador no soporta html5 se renderizará el contenido de las etiquetas, en el que se podría introducir, por ejemplo, un plugin en flash que proporcione el mismo soporte de audio o video.

- Posibilidad de acceso a los **dispositivos de micro y webcam**, que nos permitirán:
 - mostrar el video al propio usuario,

- grabar el video y audio,
- enviar y recibir imágenes de la webcam,
- **svg**: renderización de gráficos en función de un documento SVG (Scalable Vector Graphics),

```

1 <svg>
2   <circle id="myCircle"
3     cx="100" cy="75" r="50"
4     fill="blue"
5     stroke="firebrick"
6     stroke-width="3" />
7   <text x="60" y="155">Hello World</text>
8 </svg>

```

- **canvas**: nos permitirá renderizar imágenes sin necesidad de plugins, generando de forma dinámica el contenido a través de javascript , y es el elemento de la polémica, en el que muchos han visto la muerte del flash, como hasta ahora lo conocíamos,

```

1 <canvas id="mycanvas" width="200" height="100">
2   Your browser does not support the <code>canvas</code> element.
3 </canvas>
4
5 <script language="javascript">
6
7   var canvas = document.getElementById('mycanvas');
8
9   var context = canvas.getContext('2d');
10
11   contexto.fillRect(50, 0, 10, 150);
12
13 </script>

```

5. Soporte para css3.

Tenemos las siguientes características destacables:

- un estilo nativo para esquinas redondeadas, gradient y sombreado.

```

1 <style type="text/css">
2   #style1 {
3     border-radius: 15px;
4   }
5 </style>

```

- será posible utilizar cualquier tipo de fuente, no solo las que tenga instaladas el cliente. Para ello, bastará con añadir al estilo la url de la tipografía, del ttf.

```

1 <style type="text/css">
2   @font-face {
3     font-family: 'Comics';
4     src: url(Comics.ttf);
5   }
6   header {
7     font-family: 'Comics';
8   }
9 </style>

```

- text wrapping con mayor parametrización que permite por ejemplo incluir puntos suspensivos...

```

1 <style type="text/css">
2   div {
3     text-overflow: ellipsis;
4   }
5 </style>

```

- opacidad estándar en todos los navegadores,

```

1 <div style="background: rgba(255, 0, 0, 1) ;"></div>

```

- soporte para transiciones entre estilos basadas en el concepto análogo de jQuery, con un periodo de cadencia,

```

1 <style type="text/css">
2
3   #box {
4     -webkit-transition: margin-left 1s ease-in-out;
5   }
6
7 </style>
8
9 <script language="javascript">
10
11   document.getElementById('box').className = 'left';
12   document.getElementById('box').className = 'right';
13
14 </script>

```

- animaciones, al estilo de las que disponemos con script.aculo.us para prototype.js o con jqueryUI para jQuery.

```

1 <style type="text/css">
2   div {
3     -webkit-animation-name: pulse;
4     -webkit-animation-duration: 2s;
5     -webkit-animation-iteration-count: infinite;
6     -webkit-animation-timing-function: ease-in-out;
7     -webkit-animation-direction: alternate;
8   }
9 </style>

```

6. ¿Una página de ejemplo?.

Lo primero que debemos conocer es que el DOCTYPE de HTML5 es el siguiente:

```

1 <!DOCTYPE html>

```

El DOCTYPE es lo que diferencia las versiones de los documentos html, y define el contrato que deben seguir los documentos XML en general. En HTML5 es sencillo, no hay referencia a DTD, de modo que los navegadores que no lo soporten harán uso de su versión de html por defecto.

Me voy a remitir en este punto a la página de ejemplo que ha generado la gente de Google para explorar las capacidades de HTML5 y que se usó en el reciente [Google D3vF3st de Madrid](#) para demostrar la compatibilidad de Google Chrome con el estándar; por esto último la recomendación, hoy día, es verla con Google Chrome.

<http://slides.html5rocks.com>

También es buena recomendación:

<http://html5demos.com/>

donde se puede ver la compatibilidad hoy día con los navegadores.

7. ¿Cuál es el estado actual de la especificación y las implementaciones?

La especificación está aún en borrador, aunque sufrirá ya pocas modificaciones y, si bien en el arranque se barajaban como fechas para tener las primeras implementaciones 2012 e incluso 2022, los navegadores soportan ya bastante del estándar, algunos más que otros. La idea es que la especificación se cerrará cuando haya, al menos, dos implementaciones completas del estándar, esto es, dos navegadores que la soporten al 100%.

Para saber qué de compatible es tu navegador favorito con HTML5, puedes visitar la siguiente página:
<http://www.html5test.com>.

¿Comenzaríamos ahora a desarrollar basándonos en HTML5?, si no fuese una aplicación para internet y tuviésemos poder de decisión sobre qué navegador tendrá instalado el cliente, sí. Podríamos decir que estamos en un primer estadio de la tecnología y que si lo pillamos a tiempo nos podemos subir al carro y estar preparados para cuando esté maduro.

8. Referencias.

- <http://dev.w3.org/html5/html-author/>
- <http://slides.html5rocks.com>
- <http://www.html5test.com>
- <http://joshduck.com/periodic-table.html>
- <http://www.chromeexperiments.com/>
- <http://html5demos.com/>
- <http://code.google.com/p/html5-shims/wiki/LinksandResources>

9. Conclusiones.

Como os podéis imaginar, esta breve introducción da pie a muchos más tutoriales sobre cada una de las nuevas características de HTML5, iremos poco a poco sumergiéndonos en ellos.

Siempre intentaremos no tener que bajar a nivel del lenguaje de cliente para desarrollar, porque es más difícil depurar, siempre estaremos a expensas de cómo se comporte el código en los distintos navegadores,... que os voy a contar, que no sepáis, a los que habéis estado trabajando los últimos años con interfaces|documentos web.

La esperanza es que tendremos librerías de componentes visuales que nos den soporte a las nuevas características, las librerías de componentes JSF se terminarán adaptando y, para los que trabajen con Spring MVC ya hay una primera implementación basada en HTML5 de un proyecto llamado [Summer](#), como la continuación del Spring ;-)

Un saludo.

Jose

jmsanchez@autentia.com

A continuación puedes evaluarlo:

[Regístrate para evaluarlo](#)

Por favor, vota +1 o compártelo si te pareció interesante

[Share](#) |

[+1](#) 3

Anímate y coméntanos lo que pienses sobre este **TUTORIAL**:

» [Regístrate](#) y accede a esta y otras ventajas «



Esta obra está licenciada bajo licencia [Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

PUSH THIS

Page Pushers

Community

Help?

no clicks

0 people brought clicks to this page

+ + + + + + + +

powered by [karmacracy](#)

Copyright 2003-2015 © All Rights Reserved | [Texto legal y condiciones de uso](#) | [Banners](#) | [Powered by Autentia](#) | [Contacto](#)

W3C XHTML 1.0

W3C CSS

XML RSS

XML ATOM