

# **jDate.js**

## **Manual Técnico**

**Jesús Ángel Luna Guerrero**  
Abril 2014.

# Índice

1. Introducción .....	Página 1
2. Breve descripción del trabajo .....	Página 1
3. Introducción a las tecnologías utilizadas .....	Página 1
4. Presentación de la herramienta .....	Página 2
5. Descripción de los distintos elementos de configuración .....	Página 3
6. Distintos ejemplos de uso .....	Página 5
7. Conclusiones .....	Página 7

# 1. Introducción

Hoy en día, el manejo de las fechas y las horas en una página web (foro, red social, etc.) es una tarea que se lleva a cabo con bastante frecuencia, esta, normalmente, llevada a cabo por el webmaster o el administrador de la página.

Lo que se espera con esta herramienta, es poder automatizar esta tarea, utilizando una librería en JavaScript que pueda ser configurable en términos de formato de fecha, elementos HTML que contendrá los datos, formato de la entrada de los datos, etc.

## 2. Breve descripción del trabajo

jDate.js es una librería JavaScript que puede ser utilizada, por ejemplo, en redes sociales o foros para el manejo y la personalización de las fechas y las horas de los posts, entradas o cualquier otro caso en el que unos datos temporales estén involucrados dentro de una página web.

La configuración y uso de la herramienta es bastante sencilla, añadiendo en la cabecera, la biblioteca JavaScript y seguidamente la configuración que deseemos. Así automáticamente, una vez se cargue los datos de la web, el software hará su trabajo y las fechas serán traducidas en el mismo momento, con las características que deseemos.

## 3. Introducción a las tecnologías utilizadas

- **JavaScript:** (abreviado comúnmente "JS") es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas<sup>4</sup> aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. JavaScript se interpreta en el agente de usuario, al mismo tiempo que las sentencias van descargándose junto con el código HTML.

- **jQuery:** es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. Fue presentada el 14 de enero de 2006 en el BarCamp NYC. jQuery es la biblioteca de JavaScript más utilizada.

jQuery es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia

MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privativos.<sup>2</sup> jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, como son:

- Selección de elementos DOM.
- Interactividad y modificaciones del árbol DOM, incluyendo soporte para CSS 1-3 y un plugin básico de XPath.
- Eventos.
- Manipulación de la hoja de estilos CSS.
- Efectos y animaciones.
- Animaciones personalizada .
- AJAX.
- Soporta extensiones.
- Utilidades varias como obtener información del navegador, operar con objetos y vectores, funciones para rutinas comunes, etc.
- Compatible con los navegadores Mozilla Firefox 2.0+, Internet Explorer 6+, Safari 3+, Opera 10.6+ y Google Chrome 8+.

## 4. Presentación de la herramienta

La idea de esta biblioteca, surge cuando se han realizado en varias ocasiones para páginas o aplicaciones web, funciones que deben de trabajar con horas o fechas en distintos formatos, un trabajo que en la mayoría de las ocasiones es repetitivo y conlleva bastante tiempo.

Por lo tanto, se decide crear una biblioteca que aglomere de forma sencilla varias funciones para manejar dichos datos temporales, que pueden ser tomados de un SSOO, un SGBD, etc...

El uso de esta extensión es muy sencillo y fácil de instalar, normalmente cuando usamos jQuery se necesitaría agregar 2 ficheros dentro de la etiqueta `<head>` de los archivos HTML.

```
<head>
  <script src="jquery-2.1.0.min.js"></script>
  <script src="config.js"></script>
</head>
```

Para nuestro caso (jDate), lo único que deberíamos hacer es añadir una línea mas y poner la biblioteca en la ruta que especifiquemos.














```
<head>
  <script src="jquery-2.1.0.min.js"></script>
  <script src="jDate.js"></script>
  <script src="config.js"></script>
</head>
```

A partir de ahí, a todos los elementos que indiquemos en el config.js se le aplicará la configuración indicada (como la que tenemos a continuación).

```
jDate_options = {
  type: 'socialNetwork',
  sourceInput: 'unixFormat',
};
```

Como hemos dicho, la configuración de la herramienta es mínima y muy fácil, y alguien con unos conocimientos mínimos de JavaScript podría utilizarla sin problema ninguno y además con muy buenos resultados.

Al depender el programa de varias tecnologías es necesario saber con que navegadores este funcionaría o no, por eso, se presenta la siguiente tabla en la que se muestran todas la compatibilidades o incompatibilidades:

								
	2.0+	8.0+	10.0+	6.0	7.0	8.0	9.0	3.0+
jDate								

## 5. Descripción de los distintos elementos de configuración

jDate tiene cuatro modos de trabajo: formatDate, socialNetwork, difference y countdown, cada uno de ellos necesita una configuración mínima y si esta no se da, cogerá los parámetros por defecto que vienen configurados en la misma herramienta, por lo tanto, hay que saber que se está haciendo en cada momento porque puede que la salida no sea la deseada.

- **formatDate:** es el modo en el que podemos poner la fecha y la hora de la forma que mejor le convenga al usuario.

Como ejemplo, se podría decir que los formatos de unix o de iso (que son los más utilizados) no son los más representativos.

- 1398185636 (unix)
- 2013-05-21T15:00:00+0200 (iso)

### Parámetros configurables:

- type: 'formatDate'
- sourceInput: 'unixFormat' o 'isoFormat'
- \* utc: 'false' o 'true'
- \* content: [{name: 'date',  
separator: '/',},  
{name: 'time',  
separator: ':',},

```

    {name: 'seconds',
      separator: ':'},
  ]
  • * error: 'false' o 'true'

```

*Nota: los parámetros marcados con '\*' son opcionales.*

- **socialNetwork:** es el modo que se utilizaría para indicar la fecha y la hora de una entrada en un blog o en una red social.

Automáticamente, calcula si el tiempo transcurrido es menor que minutos, horas, días o semanas y solo si es mayor pondría una fecha formateada. Solo funciona con fechas anteriores a la actual, sino daría error.

### Parámetros configurables:

- type: 'socialNetwork'
- sourceInput: 'unixFormat' o 'isoFormat'
- \* error: 'false' o 'true'

*Nota: los parámetros marcados con '\*' son opcionales.*

Para este modo se recomienda el uso de `isoFormat` puesto que tiene mejor manejo de las horas en UTC y en otro caso el calculo de las mismas podría ser erróneo.

- **difference**, al igual que `socialNetwork` es un modo que se utilizaría para indicar la fecha y la hora de una entrada en un blog o en una red social. Soluciona el problema de cuando las fechas son superiores a la actual.

Se recomienda el uso de `socialNetwork` si se sabe de antemano que no existirá tal problema, puesto que `difference` conlleva mayor computo.

### Parámetros configurables:

- type: 'difference'
- sourceInput: 'unixFormat' o 'isoFormat'
- \* error: 'false' o 'true'

*Nota: los parámetros marcados con '\*' son opcionales.*

Al igual que en el modo anterior (socialNetwork) recomienda el uso de isoFormat puesto que tiene mejor manejo de las horas en UTC y en otro caso el calculo de las mismas podría ser erróneo.

- **countDown**, es el modo que se utilizaría para llevar una cuenta atrás, indicando, para este caso, la fecha de finalización.

Posee dos submodos:

- `clock`: que calcula al detalle el tiempo que falta (días, minutos y segundos).
- `onlyDays`: solo dice los días que faltan para llegar a la fecha deseada.

### Parámetros configurables:

- type: 'countDown'
- sourceInput: 'unixFormat' o 'isoFormat'
- selector: 'clock' o 'onlyDays'
- description: 'texto que deseemos'
- \* error: 'false' o 'true'

*Nota: los parámetros marcados con '\*' son opcionales.*

Hay que tener especialmente **cuidado** con el parámetro description, puesto que sino lo ponemos o lo dejamos en blanco se puede producir un error.

A continuación se harán varias recomendaciones que serán de gran utilidad para prevenir distintas situaciones de error:

- No olvidar nunca los parámetros type y sourceInput puesto que son los únicos que son necesarios en todos los modos.
- Prestar atención a los parámetros opcionales porque pueden ocasionarnos que las salidas no sean las deseadas.
- Cuidado cuando el parámetro error esté a true porque el principal fallo que suele ocurrir es que indiquemos un sourceInput como unixFormat y estemos introduciendo un isoFormat.

## 6. Distintos ejemplos de uso

A continuación se presentan varios ejemplos de uso y sus salidas o resultados correspondientes para que el usuario tenga una idea general de cada modo presente en la biblioteca:

- Ejemplo para **formatDate**:

```
<p class="date1">2014-04-22T19:54:39+02:00</p>
```

```
<p class="date2">1398621068</p>
```

```
jDate_options1 = {
    type: 'formatDate',
    sourceInput: 'isoFormat',
    utc: 'false',
    content:[{name: 'date',
              separator: '/'},
            {name: 'time',
              separator: ':'},
            {name: 'seconds',
              separator: ':'}],
};

$("p.date1").drawDate(jDate_options1);
```

- Ejemplo para **socialNetwork**:

```
<p class="date3">2014-04-22T19:54:39+02:00</p>
<p class="date4">1398632400</p>
<p class="date5">2014-06-09T09:00:00+02:00</p>
```

```
jDate_options3 = {
    type: 'socialNetwork',
    sourceInput: 'isoFormat',
};

$("p.date3").drawDate(jDate_options3);
```

- Ejemplo para **difference**:

```
<p class="date6">2014-06-09T09:00:00+02:00</p>
<p class="date7">1398632400</p>
```



```
jDate_options6 = {
    type: 'difference',
    sourceInput: 'isoFormat',
};

$("#p.date6").drawDate(jDate_options6);
```

- Ejemplo para **countDown**:

```
<p class="date8">2014-06-09T09:00:00+02:00</p>
<p class="date9">1402012800</p>
```

```
jDate_options8 = {
    type: 'countDown',
    sourceInput: 'isoFormat',
    selector: 'clock',
    description: 'el examen de SLCS',
};

$("#p.date8").drawDate(jDate_options8);
```

## 7. Conclusiones

- Se ha creado una herramienta fácil de usar para quienes posean unos conocimientos mínimos de JavaScript y jQuery.
- Es fácil modificarla y añadirle nuevas mejoras al tratarse de Software Libre.
- La herramienta desarrollada es bastante útil y eficaz en el ámbito de las páginas web y ayuda de gran modo al administrador ahorrándole tiempo y añadiendo simplicidad a su código.