

Javascript como lenguaje de Apoyo a la Docencia vía Web

V.M. Rivas, J. R. Balsas, J.J. Aguilera, L.A. Ureña

Dpto. de Informática
Universidad de Jaén
Avda. Madrid, 35
23071 Jaén

e-mail: [vrivas, jrbalsas, jjaguile, laurena](mailto:vrivas, jrbalsas, jjaguile, laurena@ujaen.es)@ujaen.es

Resumen

Presentamos el lenguaje *Javascript* como el idóneo para realizar sitios Web con páginas interactivas, válido para evitar diferencias en aquellas páginas que, para mejorar la navegabilidad, repiten información y utilizables por el alumno como copias locales en su ordenador de modo que no necesite de conexión a Internet para manejarlas e interactuar con ellas.

1. Introducción

Es fácilmente constatable el hecho de que cada vez se usa con más frecuencia Internet, así como las tecnologías a ella asociadas, en un intento de descubrir nuevas técnicas metodológicas que permitan llevar al alumno la información más útil y en el formato más adecuado para que éste pueda asimilarla en favor de su aprendizaje.

Sería largo el proceso de citar todos aquellos trabajos que en años recientes están poniendo de manifiesto que el uso de Internet y más concretamente de la tecnología Web, debe ocupar un lugar destacado dentro del conjunto de metodologías que el profesor está en obligación de utilizar si con ello consigue mejorar la calidad de enseñanza. Trabajos como los de P. Navarro [1], C. Poyatos [2] o L.M. Marín [3], así como los de J.C. Cano [4], J.V. Oltra [5] y J.R. Balsas [6] ponen el énfasis en mostrar los distintos posibles usos que el docente puede hacer de Internet.

Las características que hacen útil Internet son, resumidamente, las siguientes:

- *Disponibilidad temporal y espacial.* Se puede acceder a la información en cualquier instante y desde cualquier sitio.

- *Mecanismo de comunicación a la vez síncrono y asíncrono.* Permite establecer comunicación entre docentes y/o alumnos tanto de manera sincronizada (IRC) como de forma asíncrona (Web, correo electrónico).
- *Herramientas gratuitas y multiplataforma.* La distribución de documentos Web cuenta con la ventaja de que los navegadores se pueden obtener gratuitamente y existen para cualquier tipo de ordenador, independientemente de su potencia de cálculo u otras prestaciones.

Por todo ello, aparecen continuamente sitios Web desde los que el alumno puede acceder a temarios, guiones de prácticas, realizar exámenes en línea o consultar sus notas.

La pretensión de este artículo es mostrar algunas de los inconvenientes con que nos podemos encontrar a la hora de distribuir información en formato de páginas Web para los alumnos y cómo el uso del sencillo lenguaje de programación *Javascript* puede ayudar a solventarlos.

2. Distribución de documentos

Una de las tareas que la utilización de páginas Web puede simplificar enormemente es la distribución de la información generada por el propio profesor entre los alumnos. Sin embargo, para un total aprovechamiento de las posibilidades de interacción que estas páginas proporcionan es necesario recurrir a la utilización de programas, comúnmente denominados *Common Gateway Interfaces (CGI)*, que generen dinámicamente desde el propio servidor de páginas Web respuestas ante la utilización que el usuario hace de la página. Esto es igual que decir que dicha interactividad no es posible si el alumno no está

conectado a Internet cuando está visualizando la página.

Las consecuencias de la anterior aseveración son claras: todo alumno necesita conexión a Internet desde su puesto de estudio, lo cual a su vez conlleva un mayor gasto económico.

La utilización de CGI lleva asociada otra ventaja adicional y es que se evita que la información que aparece repetida en varias páginas puede mostrar divergencias entre sí. Hemos de tener en cuenta que un índice de medida de la calidad de un sitio web es precisamente la navegabilidad del mismo, es decir, que entre cualesquiera dos páginas de dicho sitio la distancia medida en pulsaciones del ratón sea mínima. Sin caer en la repetición de toda la información en todas las páginas, para aumentar la navegabilidad muchas de ellas compartirán zonas comunes que, de no ser generadas automáticamente, tienen una alta probabilidad de presentar diferencias entre ellas.

Ante estos dos hechos (necesidad de conexión continua para documentos dinámicos y de utilización de generación automática de páginas Web) se hace necesaria la utilización de alguna herramienta que permita generar de forma dinámica el contenido de un sitio Web sin que el alumno tenga que estar forzosamente conectado a Internet. Tal herramienta es precisamente el lenguaje *Javascript*.

Desde hace algunos años, profesores del Dpto. de Informática de la Universidad de Jaén están usando este lenguaje precisamente para poder distribuir entre sus alumnos guiones de prácticas, temarios, etc. que puedan ser ejecutados localmente por los mismos. Además, se incluyen ejercicios y resolución de pequeños exámenes que permiten que el alumno se autoevalúe y compruebe en qué medida está aprovechando la enseñanza que recibe. De esta forma, se consigue también que la evaluación se haga de forma continua.

3. El lenguaje *Javascript*

Javascript [7][8] es un lenguaje de programación de alto nivel creado por la empresa *Netscape* que puede ser usado para visualizar páginas Web en cualquier navegador: *Netscape Navigator*, *Internet Explorer* u *Opera*, por citar los más usados.

Javascript es un lenguaje interpretado por el propio navegador Web, el cual lee las instrucciones, las traduce y ejecuta. El código de un programa en *Javascript* puede insertarse directamente dentro del código HTML de una página Web o bien en ficheros externos que son invocados desde una página Web. En este último caso, para poder ejecutar dichos ficheros externos, es necesario incluir en la página Web el URL de acceso a ellos. Dicho URL puede ser bien absoluto bien relativo, en cuyo caso permite que se puedan usar ficheros *Javascript* en copias locales de sitios Web sin necesidad de estar conectado a Internet.

La figura 1 muestra un ejemplo de código en *Javascript* en el que se ejecuta tanto código externo a la página web como código interno.

```
<HTML>
<HEAD>
  <TITLE>Prueba página
  Javascript</TITLE>
  <SCRIPT SRC="fichero.js">
  </SCRIPT>
</HEAD>
<BODY>
  <P>Bienvenido</P>
  <SCRIPT>
    document.writeln( "Mi
página" );
  </SCRIPT>
</BODY>
</HTML>
```

Figura 1. Ejemplos de inserción de código *Javascript* en una Página Web.

Una de las características más destacables de *Javascript* es que permite utilizar matrices asociativas así como programación basada en objetos. Estas técnicas de programación simplifican notablemente el trabajo del programador al permitirle alcanzar niveles mayores de abstracción a la hora de desarrollar sus aplicaciones.

No podemos obviar dos desventajas del uso de este lenguaje. Por un lado la reducida velocidad de traducción y ejecución de las sentencias, dado que los lenguajes interpretados son más lentos que los compilados. Por otro los problemas mínimos, pero existentes, de compatibilidad entre la versión de *Javascript* que admite el navegador *Netscape* y la

que admite *Internet Explorer*. Para resolver este problema, *Javascript* incluye funciones para detectar en qué navegador se está ejecutando la página Web por lo que el programador puede distinguir qué código *Javascript* se ejecutará si el navegador es uno o si es el otro.

4. Uso para la docencia

Como ejemplo real de lo comentado se muestra el sitio Web de la asignatura *Programación de Aplicaciones* [9], de la titulación de *Ingeniería Técnica en Topografía* de la Universidad de Jaén. Asignatura optativa de 2º curso e impartida por el Departamento de Informática, y más concretamente el área de Lenguajes y Sistemas Informáticos.

En un esfuerzo por hacer del sitio Web de dicha asignatura uno de los recursos didácticos fundamentales de apoyo para profesor y alumno, se han desarrollado un conjunto de bibliotecas de funciones en *Javascript* encaminadas a aumentar la navegabilidad del sitio Web y reducir inconsistencias. Destaca la organización en forma de ventanas usada para agrupar porciones de información que suelen aparecer juntas. Así, se han declarado objetos de tipo *Ventana* que permiten tanto configurar la apariencia externa de la ventana (colores, gráficos, tipografía utilizada y otros), como añadir subobjetos de tipo *Contenido*. Estos objetos *Contenido* pueden incorporar texto plano y/o código HTML y, opcionalmente, una dirección URL a la que apuntar. La ventaja del uso de *Javascript* radica en que cada uno de estos objetos puede estar definido en ficheros distintos, y por tanto utilizables desde diferentes páginas Web. Tenemos así que la información está en un solo fichero, pero puede mostrarse en tantos sitios como sea necesario y además hacerlo de forma particularizada para cada página.

Las figuras 2 y 3 muestran, respectivamente, la definición de un objeto *ventana* y de un objeto *contenido* y el mecanismo por el cual dicho objeto se añade a la ventana para ser mostrado.

```

/*
- - - - -
Fichero en Javascript:
ventanas.js
- - - - -
*/

/**
 * Método para mostrar una ventana
 */
function muestra_ventana(
    modo_visualización ) {
    // Muestra el contenido de la
    // ventana
    . . .
} // fin muestra_ventana

/**
 * Objeto VENTANA
 */
function Ventana(
    imagen, imagen_ancho,
    imagen_alto
    ventana_titulo,
    ventana_href_titulo,
    ventana_ancho_por_defecto,
    ventana_alineación_vertical,
    ventana_color_titulo,
    ventana_fondo_titulo,
    ventana_color_contenido,
    ventana_fondo_contenido,
    ventana_cell_padding,
    ventana_cell_spacing ) {

    // asignación de parámetros a
    // variables de instancia
    this.titulo=ventana_titulo;
    . . .

    // asignación de métodos
    this.mostrar=muestra_ventana;
    . . .
} // fin de ventana

```

```

/**
 * Método para mostrar contenido
 */
function muestra_contenido(
    modo_visualización ) {
    // Muestra un CONTENIDO
    . . .
} // fin muestra_contenido

/**
 * Objeto CONTENIDO
 */
function Contenido(
    texto, url ) {
    this.texto=texto;
    this.url=url;
    this.mostrar=muestra_contenido;
} // fin de CONTENIDO

/*
- - - - -
Fichero en Javascript:
teoria.js
- - - - -
*/

// Declaración de variables de
// tipo ventana y contenido

var teoria=new Ventana(
    lista de parámetros );
var tema_1=new Contenido(
    "Tema 1. Introducción",
    "teoria/tema_1.html" );
teoria.aniade( tema_1 );

```

Figura 2. Ficheros con la declaración de los objetos *Ventana* y *Contenido*, así como variables de sendos tipos de objetos.

```

/*
- - - - -
Fichero en HTML:
teoria.html
- - - - -
*/

<HTML>
<HEAD>
<TITLE>Teoría de PDA</TITLE>
<SCRIPT SRC="ventanas.js">
</SCRIPT>
<SCRIPT SRC="teoria.js">
</SCRIPT>
</HEAD>
<BODY>
    <P>Bienvenido...</P>
    <SCRIPT>
        teoria.mostrar( MODO_LISTA );
    </SCRIPT>
</BODY>
</HTML>

```

Figura 3. Uso de la variable *teoria* en la página Web.

Las figuras 4 y 5 muestran dos páginas distintas de este sitio Web en las cuales se hace uso de la ventana de *teoría* la cual está siendo utilizada en dos páginas distintas: la primera (Fig. 4) en la página principal de la sección de teoría del sitio Web; la segunda (Fig. 5) en el documento guía de la asignatura entregado a los alumnos a principio de curso.

Aunque solo se han incluido dos páginas como ejemplo, el uso de *Javascript* se hace también recomendable en la tarea de plantear ejercicios al alumno de forma que este los pueda ejecutar en local y se le puedan corregir automáticamente. Las posibilidades de esta técnica aumentan si tenemos en cuenta que los ficheros en *Javascript* están escritos en texto plano por lo que pueden ser generados automáticamente por cualquier aplicación realizada a tal efecto y ser distribuidos posteriormente a los alumnos.

5. Conclusión

En este trabajo se ha intentado poner de manifiesto la utilidad del lenguaje de programación *Javascript* a la hora de realizar

sitios web para docencia en los cuales se maximice la navegabilidad al tiempo que se minimizan las inconsistencias debidas a repetición de información en distintas páginas web del mismo sitio.

Además, se propone este lenguaje como base para la realización de ejercicios de autoevaluación para el alumno, el cual puede realizarlos desde una copia local del sitio web en su ordenador, sin requerir de una conexión constante al servidor Web de la asignatura.



Fig. 4. Uso de la ventana *teoría* en la página principal de la sección de teoría del sitio Web



Fig. 5. Uso de la ventana *teoría* en la página con la guía de la asignatura en el sitio Web

- [1] P. Navarro. *El impacto de Internet en la educación universitaria*. II Congreso Nacional de Usuarios de Internet e Infovía, Mundo Internet'97. Vol. I. ISBN:84-922440-1-. Madrid, 1997
- [2] C. Poyatos. *El impacto de la Internet en la Enseñanza y Aprendizaje del Español como Lengua Extranjera*. II Congreso Nacional de Usuarios de Internet e Infovía, Mundo Internet'97. Vol. I. ISBN:84-922440-1-. Madrid, 1997
- [3] L.M. Marín, A. Gámez. *Internet como Herramienta de apoyo a la docencia*. VII CUIE, Vol I(II). Huelva, 1999.
- [4] J.C. Cano, J. Sahuquillo, M. Alonso, J.A. Gil. *Uso de Powerpoint e Internet en la Enseñanza Universitaria*. JENUIT'99.Zaragoza, 1999.
- [5] J.V. Oltra, J.M. Torralba, G.M. Portilla. *Teledirección de proyectos fin de carrera. Clasificación de las Herramientas de Internet para la Docencia en P.F.C. en Informática*. JENUIT'99.Zaragoza, 1999.
- [6] J.R. Balsas, J.J. Aguilera, M. García, V. M. Rivas, L.A. Ureña. *Sistema de Autoevaluación Guiado a través de Internet*. Jornadas UNED-2000, Palencia, 2000.
- [7] D. Flanagan. *JavaScript: the definitive guide*. Ed. O'Reilly, 1998.
- [8] Central de documentación sobre Javascript en Netscape:
<http://developer.netscape.com/tech/javascript/index.html>
- [9] Página de la asignatura *Programación de Aplicaciones*:
http://wwwdi.ujaen.es/~vrivas/docencia/curso_2000_2001/pda_topografia