

JavaScript un lenguaje de programación interactivo

JavaScript an interactive programming language

Evelyn Tabares Valencia

Ingeniería de sistemas y computación, Universidad Tecnológica de Pereira, Pereira, Colombia

evelyn.tabares@utp.edu.co

Resumen— JavaScript es un robusto lenguaje de programación que puede ser aplicado a un documento HTML y usado para crear interactividad dinámica en los sitios web. Fue inventado por Brendan Eich, co-fundador del proyecto Mozilla, Mozilla Foundation y la Corporación Mozilla.

Puedes hacer casi cualquier cosa con JavaScript. Puedes empezar con pequeñas cosas como carruseles, galerías de imágenes, diseños fluctuantes, y respuestas a las pulsaciones de botones. Con más experiencia, serás capaz de crear juegos, animaciones 2D y gráficos 3D, aplicaciones integradas basadas en bases de datos.

Palabras clave—Lenguaje de programación, interactividad

Abstract— JavaScript is a robust programming language that can be applied to an HTML document and used to create dynamic interactivity on websites. It was invented by Brendan Eich, co-founder of the Mozilla project, Mozilla Foundation and the Mozilla Corporation.

You can do almost anything with JavaScript. You can start with small things like carousels, image galleries, fluctuating designs, and responses to button presses. With more experience, you will be able to create games, 2D animations and 3D graphics, integrated applications based on databases.

Key Word —. Programming language, interactivity

I. INTRODUCCIÓN

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. [1]

Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario. [1]

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas

escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. [1]

A pesar de su nombre, JavaScript no guarda ninguna relación directa con el lenguaje de programación Java. Legalmente, JavaScript es una marca registrada de la empresa Sun Microsystems, como se puede ver en <http://www.sun.com/suntrademarks/>. [1]

Javascript nació con la necesidad de permitir a los autores de sitio web crear páginas que permitan intercambiar con los usuarios, ya que se necesitaba crear webs de mayor complejidad. El HTML solo permitía crear páginas estáticas donde se podía mostrar textos con estilos, pero se necesitaba interactuar con los usuarios. [2]

Javascript es un lenguaje con muchas posibilidades, utilizado para crear pequeños programas que luego son insertados en una página web y en programas más grandes, orientados a objetos mucho más complejos. Con Javascript podemos crear diferentes efectos e interactuar con nuestros usuarios. [2]

Este lenguaje posee varias características, entre ellas podemos mencionar que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros. [2]

Es necesario resaltar que hay dos tipos de JavaScript: por un lado está el que se ejecuta en el cliente, este es el Javascript propiamente dicho, aunque técnicamente se denomina Navigator JavaScript. Pero también existe un Javascript que se ejecuta en el servidor, es más reciente y se denomina LiveWire Javascript. [2]

II. CONTENIDO

A principios de los años 90, la mayoría de usuarios que se conectaban a Internet lo hacían con módems a una velocidad máxima de 28.8 kbps. En esa época, empezaban a desarrollarse las primeras aplicaciones web y por tanto, las páginas web comenzaban a incluir formularios complejos. [1]

Con unas aplicaciones web cada vez más complejas y una velocidad de navegación tan lenta, surgió la necesidad de un lenguaje de programación que se ejecutara en el navegador del usuario. De esta forma, si el usuario no rellenaba correctamente un formulario, no se le hacía esperar mucho tiempo hasta que el servidor volviera a mostrar el formulario indicando los errores existentes. [1]

Brendan Eich, un programador que trabajaba en Netscape, pensó que podría solucionar este problema adaptando otras tecnologías existentes (como ScriptEase) al navegador Netscape Navigator 2.0, que iba a lanzarse en 1995. Inicialmente, Eich denominó a su lenguaje LiveScript. [1]

Posteriormente, Netscape firmó una alianza con Sun Microsystems para el desarrollo del nuevo lenguaje de programación. Además, justo antes del lanzamiento Netscape decidió cambiar el nombre por el de JavaScript. La razón del cambio de nombre fue exclusivamente por marketing, ya que Java era la palabra de moda en el mundo informático y de Internet de la época. [1]

La primera versión de JavaScript fue un completo éxito y Netscape Navigator 3.0 ya incorporaba la siguiente versión del lenguaje, la versión 1.1. Al mismo tiempo, Microsoft lanzó JScript con su navegador Internet Explorer 3. JScript era una copia de JavaScript al que le cambiaron el nombre para evitar problemas legales. [1]

Para evitar una guerra de tecnologías, Netscape decidió que lo mejor sería estandarizar el lenguaje JavaScript. De esta forma, en 1997 se envió la especificación JavaScript 1.1 al organismo ECMA European Computer Manufacturers Association). [1]

ECMA creó el comité TC39 con el objetivo de "estandarizar de un lenguaje de script multiplataforma e independiente de cualquier empresa". El primer estándar que creó el comité TC39 se denominó ECMA-262, en el que se definió por primera vez el lenguaje ECMAScript. [1]

Por este motivo, algunos programadores prefieren la denominación ECMAScript para referirse al lenguaje JavaScript. De hecho, JavaScript no es más que la implementación que realizó la empresa Netscape del estándar ECMAScript. [1]

La inigualable popularidad de JavaScript como lenguaje de programación de aplicaciones web se ha extendido a otras aplicaciones y otros entornos no relacionados con la web. [1]

Herramientas como Adobe Acrobat permiten incluir código JavaScript en archivos PDF. Otras herramientas de Adobe como Flash y Flex utilizan ActionScript, un dialecto del mismo estándar de JavaScript. [1]

Photoshop permite realizar pequeños scripts mediante JavaScript y la versión 6 de Java incluye un nuevo paquete (denominado javax.script) que permite integrar ambos lenguajes. [1]

Por último, aplicaciones como Yahoo Widgets y el Dashboard de Apple utilizan JavaScript para programar sus widgets. [1]

Entre los diferentes servicios que se encuentran realizados con Javascript en Internet se encuentran:

- Correo
- Chat
- Buscadores de Información

También podemos encontrar o crear códigos para insertarlos en las páginas como:

- Reloj
- Contadores de visitas
- Fechas
- Calculadoras
- Validadores de formularios
- Detectores de navegadores e idiomas [2]

Algunas de las características de JavaScript son:

- Es Liviano. [2]
- Multiplataforma, ya que se puede utilizar en Windows, Linux o Mac. [2]
- Es Imperativo y estructurado, pues mediante un conjunto de instrucciones indica al computador qué tarea debe realizar. [2]
- Prototipado, debido a que usa prototipos en vez de clases para el uso de herencia. [2]
- Orientado a objetos. [2]
- E Interpretado, ya que no se compila para poder ejecutarse. [2]

JavaScript es soportado por la mayoría de los navegadores como Internet Explorer, Netscape, Opera, Mozilla Firefox, entre otros. [2]

Con el surgimiento de lenguajes como PHP del lado del servidor y Javascript del lado del cliente, surgió Ajax en acrónimo de (Asynchronous JavaScript And XML). El mismo es una técnica para crear aplicaciones web interactivas. Este lenguaje combina varias tecnologías: [2]

HTML y Hojas de Estilos CSS para generar estilos. Implementaciones ECMAScript, uno de ellos es el lenguaje JavaScript. [2]

XMLHttpRequest es una de las funciones más importantes que incluye, que permite intercambiar datos asincrónicamente con el servidor web, puede ser mediante PHP, ASP, entre otros. [2]

Debemos tener en cuenta que aunque Javascript sea soportado en gran cantidad de navegadores nuestros usuarios pueden elegir la opción de Activar/Desactivar el Javascript en los mismos. [1]

Desde su aparición, JavaScript siempre fue utilizado de forma masiva por la mayoría de sitios de Internet. La aparición de Flash disminuyó su popularidad, ya que Flash permitía realizar algunas acciones imposibles de llevar a cabo mediante JavaScript. [1]

Sin embargo, la aparición de las aplicaciones AJAX programadas con JavaScript le ha devuelto una popularidad sin igual dentro de los lenguajes de programación web. [1]

En cuanto a las limitaciones, JavaScript fue diseñado de forma que se ejecutara en un entorno muy limitado que permitiera a los usuarios confiar en la ejecución de los scripts. [1]

De esta forma, los scripts de JavaScript no pueden comunicarse con recursos que no pertenezcan al mismo dominio desde el que se descargó el script. Los scripts tampoco pueden cerrar ventanas que no hayan abierto esos mismos scripts. Las ventanas que se crean no pueden ser demasiado pequeñas ni demasiado grandes ni colocarse fuera de la vista del usuario (aunque los detalles concretos dependen de cada navegador). [1]

Además, los scripts no pueden acceder a los archivos del ordenador del usuario (ni en modo lectura ni en modo escritura) y tampoco pueden leer o modificar las preferencias del navegador. [1]

Por último, si la ejecución de un script dura demasiado tiempo (por ejemplo por un error de programación) el navegador informa al usuario de que un script está consumiendo demasiados recursos y le da la posibilidad de detener su ejecución. [1]

A pesar de todo, existen alternativas para poder saltarse algunas de las limitaciones anteriores. La alternativa más utilizada y conocida consiste en firmar digitalmente el script y solicitar al usuario el permiso para realizar esas acciones. [1]

La sintaxis de JavaScript es muy similar a la de otros lenguajes de programación como Java y C. Las normas básicas que definen la sintaxis de JavaScript son las siguientes: [1]

No se tienen en cuenta los espacios en blanco y las nuevas líneas: como sucede con XHTML, el intérprete de JavaScript ignora cualquier espacio en blanco sobrante, por lo que el código se puede ordenar de forma adecuada para entenderlo mejor (tabulando las líneas, añadiendo espacios, creando nuevas líneas, etc.) [1]

Se distinguen las mayúsculas y minúsculas: al igual que sucede con la sintaxis de las etiquetas y elementos XHTML. Sin embargo, si en una página XHTML se utilizan indistintamente mayúsculas y minúsculas, la página se visualiza correctamente, siendo el único problema la no validación de la página. En cambio, si en JavaScript se intercambian mayúsculas y minúsculas el script no funciona. [1]

No se define el tipo de las variables: al crear una variable, no es necesario indicar el tipo de dato que almacenará. De esta forma, una misma variable puede almacenar diferentes tipos de datos durante la ejecución del script. [1]

No es necesario terminar cada sentencia con el carácter de punto y coma (;): en la mayoría de lenguajes de programación, es obligatorio terminar cada sentencia con el carácter ;. Aunque JavaScript no obliga a hacerlo, es conveniente seguir la tradición de terminar cada sentencia con el carácter del punto y coma (;). [1]

Se pueden incluir comentarios: los comentarios se utilizan para añadir información en el código fuente del programa. Aunque el contenido de los comentarios no se visualiza por pantalla, si que se envía al navegador del usuario junto con el resto del script, por lo que es necesario extremar las precauciones sobre la información incluida en los comentarios. [1]

JavaScript define dos tipos de comentarios: los de una sola línea y los que ocupan varias líneas. [1]

Un guión Script es un programa que está contenido internamente dentro de una página HTML (que es el método original de escribir guiones) o reside en un archivo externo (el método preferido actualmente). [3]

En las páginas HTML, como está incluido en la etiqueta script <script>, el texto del guión no aparece en la pantalla del

usuario, y el navegador web es el que ejecuta el programa JavaScript. [3]

La etiqueta <script> se encuentra muy a menudo dentro de la sección <head> de la página HTML, pero también puede incluir guiones, si lo desea, en la sección <body>. [3]

Los guiones internos que escriben texto en la pantalla, o que escriben HTML, es preferible insertarlos en la sección <body> como en el guión que se muestra a continuación: [3]

```
<!DOCTYPE html>
<html>
<head>
  <title>Apenas un guión</title>
</head>
<body>
<h1>
  <script language="javascript" type="text/javascript">
    document.write ("¡Hola Mundo!")
  </script>
</h1>
</body>
</html>
```

Tomada de : <https://disenowebakus.net/javascript.php>

III. CONCLUSIONES

JavaScript es un lenguaje de programación interpretado, lo que significa que no se pueden detectar la mayoría de errores en el código hasta que se ejecutan los scripts. De esta forma, antes de considerar un script como correcto, es necesario probarlo en todos los navegadores sobre los que se vaya a utilizar. [1]

Cuando se producen errores durante la ejecución de un script, los navegadores proporcionan cierta información útil para descubrir el punto exacto en el que se ha producido el error y su posible solución. Solucionar los errores de un script se denomina "depurar el script" o "debugear el script" (término que viene de la palabra inglesa "debug", que significa "eliminar los errores de una aplicación"). [1]

Desafortunadamente, no todos los navegadores proporcionan la misma información útil, lo que complica la solución de los errores para cada tipo de navegador. A continuación se muestran las herramientas que proporciona cada navegador para detectar y corregir los errores en JavaScript.

JavaScript es un lenguaje del cliente; es decir, está diseñado para desempeñar su trabajo en su computadora (el de Usted), no en el servidor. Por ello, JavaScript sufre algunas limitaciones inherentes, la mayoría de ellas por razones de seguridad. [3]

JavaScript no permite leer o escribir archivos en las máquinas cliente. En gran medida eso es bueno, porque seguramente no

le gustaría que una página Web pudiera leer los archivos de su disco duro, o que fuera capaz de escribir virus en su disco o capaz de manipular los archivos de su computadora. La única excepción es que Javascript puede escribir en el archivo cookie del navegador, pero incluso esta característica tiene sus limitaciones. [3]

Javascript no permite la escritura de archivos en las máquinas servidor. Hay varias formas de hacer esto (como almacenando los contadores de impactos de la página o rellenando formularios de datos), pero JavaScript no tiene permitido hacer estas cosas. En cambio, necesitará tener un programa en su servidor para manipular y almacenar estos datos. Por ejemplo, puede tratarse de un CGI escrito en un lenguaje como Perl o PHP. [3]

JavaScript no puede cerrar una ventana que no ha abierto. Esto es así para evitar que un sitio tome el control de su navegador y empiece a cerrar ventanas de otros sitios.

JavaScript no puede leer información de una página Web abierta procedente de otro servidor. [3]

WEBGRAFIA

- [1] «uniwebsidad,» [En línea]. Available: <https://uniwebsidad.com/libros/javascript/capitulo-1/breve-historia>. [Último acceso: 28 Mayo 2019].
- [2] D. P. Valdés, «maestros del web,» 03 julio 2007. [En línea]. Available: <http://www.maestrosdelweb.com/que-es-javascript/>. [Último acceso: 28 Mayo 2019].
- [3] I. H. Delgado, «AKUS.NET,» [En línea]. Available: <https://disenowebakus.net/javascript.php>. [Último acceso: 28 5 2019].