

Interfaces de Usuario

Grado en Ingeniería Informática

Curso 2021 / 2022

TECNOLOGÍAS WEB - JAVASCRIPT

Tabla de contenido

INTRODUCCIÓN	2
DEDICACIÓN	2
ENTREGA.....	2
EVALUACIÓN.....	2
LENGUAJES DE SCRIPT DEL CLIENTE	3
JAVASCRIPT	3
LIBRERÍAS JAVASCRIPT	4
JQUERY	4
EJERCICIO	6
ENUNCIADO	6
MATERIAL REQUERIDO	9
NORMATIVA.....	9
NORMAS DE REALIZACIÓN	9
NORMAS DE ENTREGA.....	9
REFERENCIAS.....	10

Introducción

El segundo bloque tiene por objetivo conocer un lenguaje de script que permita la ejecución de código en el navegador como medio para mejorar la interacción con el sistema. Para ello, se realizará una breve introducción a JavaScript, se mostrará la sintaxis básica de este lenguaje de programación, se revisarán una serie de ejemplos, se verá el uso de librerías de este lenguaje y se resolverá un ejercicio de programación. La presente sección realiza una breve introducción a las condiciones de realización del bloque de ejercicios: dedicación estimada, fecha de entrega y condiciones de evaluación. A las condiciones aquí expuestas deberán añadirse las normas de realización y entrega recogidas al final del presente documento.

Dedicación

La realización del bloque de ejercicios de programación requerirá una dedicación estimada de DIEZ HORAS por alumno, repartidas a lo largo de cuatro semanas de trabajo y cinco sesiones en aula pequeña. Durante las sesiones en aula pequeña, además de resolver dudas de carácter general, se explicarán las tecnologías implicadas y se analizarán ejemplos sobre las mismas.

Entrega

La entrega del ejercicio de programación se realizará en la semana ocho del cuatrimestre (ver la planificación de prácticas publicada en Aula Global como “Calendario de prácticas”) a lo largo del día de la sesión en aula pequeña.

Evaluación

Los criterios de evaluación se centrarán, entre otros, en: la **funcionalidad**, el **código** y el **diseño de la interfaz de usuario**.

Este segundo bloque de ejercicios tendrá un valor de un 10% sobre la nota final de la evaluación continua. La copia de los ejercicios tanto de compañeros como de internet implicará el **suspense directo** de la evaluación continua, tanto para el grupo que copie como para el grupo autor de la versión original (en su caso).

Lenguajes de script del cliente

Los lenguajes de script del cliente permiten programar fragmentos de código ejecutados en el propio navegador como medio para mejorar la interacción de las aplicaciones web. Los lenguajes de script del cliente son esenciales para poder disponer de aplicaciones web dinámicas, cuyo comportamiento varíe dependiendo de las necesidades del usuario, las condiciones de ejecución, o el contexto de ejecución.

JavaScript

JavaScript es un lenguaje interpretado, orientado a objetos (aunque usa tipado débil) y ejecutable en un entorno anfitrión (el navegador web) que le provee de un contexto de ejecución, en forma de objetos predefinidos que representan diferentes elementos del navegador y la página, así como otras facilidades. Las características más destacables de JavaScript, en lo referente a las interfaces de usuario, son:

Modificación del texto en un documento HTML: es posible insertar en un documento un determinado texto (v.g. el valor de una variable).

Reacción ante eventos: puede asociarse la ejecución de una función a la ocurrencia de un determinado suceso (v.g. la página se ha cargado, el usuario ha pulsado un botón). Esta característica es clave para la programación del comportamiento de la interfaz de usuario.

Lectura/modificación de etiquetas HTML: JavaScript permite añadir, modificar o borrar cualquier elemento del documento HTML, lo cual abre la posibilidad de modificar la estructura, el contenido, la presentación o el enlazado de la información. Esto se consigue proporcionando al programador una interfaz DOM (Document Object Model), que le permite manipular mediante una sencilla notación el árbol del documento HTML.

Validación de datos de entrada: es sencillo programar las comprobaciones necesarias para asegurarse de que los datos introducidos por el usuario son válidos, antes de ser enviados a un agente de procesamiento.

Las diferentes versiones de JavaScript que implementan los navegadores en cuanto a las interfaces DOM hacen que la programación en este lenguaje sea especialmente laboriosa para garantizar la compatibilidad con diferentes navegadores. Esta particularidad ha de tenerse en cuenta al diseñar interfaces para la web.

Para más información sobre JavaScript se recomienda consultar las siguientes referencias: En primer lugar, el tutorial JavaScript de w3c schools [1], que recoge mediante ejemplos interactivos las características del lenguaje, y ofrece una referencia extensiva de los objetos predefinidos y los objetos DOM. De igual manera, en [2] se ofrecen tutoriales tanto introductorios como para usuarios avanzados (en ocasiones, dependientes del navegador) que pueden servir como consulta rápida. Por último, se recomienda utilizar los libros [3], [4] y [5] como manuales de referencia. Las publicaciones [3] y [4] están disponibles en Safari Books Online, mientras la [5] está disponible online con acceso abierto.

Librerías JavaScript

El aumento progresivo de las funcionalidades proporcionadas por las aplicaciones web y, especialmente, la necesidad de disponer de aplicaciones web con una interacción más rica, ha incrementado de manera considerable la complejidad de su desarrollo. A fin de resolver este problema, en los últimos tiempos han surgido una serie de librerías de JavaScript que proporcionan componentes pre-implementados que pueden ser ensamblados durante el desarrollo de una aplicación web. Ejemplos de librerías de JavaScript serían Dojo Toolkit, Google Web Toolkit, Yahoo UI Library o jQuery. En este bloque de ejercicios utilizaremos esta última librería, quizá la más popular y usada por las más grandes compañías web: Google, Microsoft, IBM, Netflix.

jQuery

jQuery es una librería de código abierto diseñada para facilitar el desarrollo de las interfaces web y el uso de JavaScript. El objetivo de jQuery se puede resumir con el lema “write less, do more” (traducción, “escribe menos, haz más”): proporcionar componentes pre-implementados que permitan implementar funciones complejas con una sola línea de código, en lugar de las múltiples necesarias si se utilizara solo JavaScript. Además, la librería es compatible con la mayoría de los navegadores hoy en uso.

Siendo una librería de JavaScript, las instrucciones jQuery se guardan en ficheros JavaScript.

La característica principal de esta librería es que permite cambiar el contenido de una página web sin necesidad de recargarla, mediante la manipulación del objeto DOM, eventos, efectos y peticiones AJAX. Para ello utiliza funciones `$()` o `jQuery()`. Así, la sintaxis se compone de un selector para seleccionar un elemento HTML seguido de una acción: **`$(selector).action()`**. Se describen ahora las características más conocidas de la librería jQuery.

Interacción con los documentos HTML: selección y manipulación de los componentes del árbol DOM y de las propiedades definidas en las hojas de estilo CSS.

Gestión de los eventos HTML: los controladores de eventos son métodos que se ejecutan cuando hay cierta interacción con el documento HTML. Generalmente, se dice que tal acción o conjunto de acciones ha sido desencadenada (“triggered” o “fired”) por un evento. Por ejemplo, la instrucción **`$(document).ready(function)`** enlaza una función **function** con el evento **ready** del objeto **document** y esta se ejecutará en el momento en que termine la carga del documento. Otros ejemplos de eventos son el **click**, el **double click** y el **mouseover** del ratón en un elemento seleccionado.

Efectos y animaciones: acciones especiales que se pueden asociar a los elementos HTML. Ejemplos de efectos son los siguientes:

- ✓ **hide()** / **show()** para ocultar / mostrar un elemento;
- ✓ **slideDown()** / **slideUp()** / **slideToggle()** para deslizar un elemento en diferentes direcciones;

- ✓ **animate()** para animar un elemento, especificando las propiedades CSS que se quieren animar como parámetros de input.

Para más información sobre jQuery se recomienda consultar las siguientes referencias: En primer lugar, el tutorial jQuery de w3schools [6], que recoge mediante ejemplos interactivos las características de la librería, y ofrece una referencia extensiva de las acciones, efectos y animaciones posibles. De igual manera, en [7] se ofrece la página oficial de la librería, con varia documentación y tutoriales tanto introductorios como para usuarios avanzados que pueden servir como consulta rápida. Por último, se recomienda utilizar el libro [8] como manual de referencia, disponible en Safari Books Online.

Ejercicio

El objetivo de este ejercicio es conocer y practicar con los principales elementos de JavaScript para elaborar una página dinámica que enseñe actividades y eventos organizados por categorías.

Enunciado

La página deberá ser la misma del ejercicio de la práctica anterior (Bloque I – HTML5 y CSS3), donde se aportarán los siguientes cambios.

- En la cabecera:
 - Al pinchar el botón para iniciar sesión, se abrirá una ventana pop up. Una ventana pop up o emergente es un elemento que al abrirse se superpone a la página actual y su tamaño depende del contenido. En la esquina superior derecha de la ventana pop up, habrá que añadir un botón para cerrar la misma. La ventana emergente contendrá un formulario pidiendo el nombre de usuario y la contraseña, y un botón.
 - Al pinchar el botón, se buscará una cookie con los datos introducidos. Si no existe la cookie, se le notificará al usuario que el correo electrónico especificado no está dado de alta. Si ya existe la cookie, se cerrará la ventana pop up y se cambiarán los botones para iniciar sesión y darse de alta por una imagen (la imagen de perfil del usuario o una por defecto si no está disponible), un texto (el nombre del usuario) y un menú de opciones. Entre las opciones, se incluirán: mi perfil, mis experiencias, y cerrar sesión.
 - La opción mi perfil abrirá una ventana pop up con la información de la cuenta del usuario ya introducida en el formulario para darse de alta (cómo se explica a continuación). El usuario podrá modificar su nombre de usuario, la foto de perfil y sus intereses. Al guardar los cambios realizados, se tendrán que reflejar también en la cookie correspondiente.
 - La opción mis experiencias abrirá una ventana pop up con un listado de experiencias introducidas por el usuario y un botón para añadir una nueva. Para añadir una nueva, en la misma ventana pop up se pedirá un título, una descripción, un lugar y una foto. La nueva experiencia se añadirá a la cabeza del listado. Todas las experiencias podrán borrarse.
 - La opción cerrar sesión abrirá una ventana pop-up preguntando confirmación antes de cerrar la sesión actual. En caso afirmativo, se volverá al diseño de la cabecera con los botones para iniciar sesión y darse de alta.

- Al pinchar el botón para darse de alta, se abrirá una ventana pop up con un formulario constituido por los siguientes campos, algunos de ellos obligatorios y con un cierto formato según lo que se especifica en los siguientes puntos.
 - Información de la cuenta
 - Nombre de usuario (Obligatorio)
 - Contraseña (máximo 8 caracteres, donde los caracteres permitidos son letras [a-z] y dígitos [0-9]) (Obligatorio)
 - Información personal
 - Nombre y Apellidos (Obligatorio)
 - Email (seguirá el formato nombre@dominio.extensión) (Obligatorio)
 - Fecha de nacimiento (dd/mm/aaaa) (Obligatorio)
 - Foto de perfil (Opcional)
 - Me interesa ... (listado de opciones, como por ejemplo turismo cultural, turismo ambiental, turismo histórico, turismo de negocios, ...)
 - He leído y acepto las Condiciones de uso (Obligatorio)
 - Botones para guardar y borrar
 - Al pinchar el botón para guardar, se validará el contenido de cada uno de los campos del formulario. Si no hay errores, se cerrará la ventana pop up y se almacenará una cookie con toda la información contenida en el formulario. Si existe una cookie con el mismo correo electrónico, se le notificará al usuario que ya existe una cuenta asociada al correo electrónico especificado.
 - Al pinchar el botón para borrar, se restablecerá la información inicial del formulario.
 - La validación del formulario se podrá realizar mediante: HTML5, JavaScript y jQuery.
- En la primera sección:
 - El mapa tendrá que ser *responsive*. Atención, las coordenadas de las áreas definidas para los enlaces tendrán que mantenerse al cambiar el tamaño de la página.

- Al pinchar los enlaces definidos en el mapa, el contenido de las páginas externas implementadas en la práctica anterior deberá incluirse en una ventana pop up en la página principal. Una ventana pop up es un elemento que al abrirse se superpone a la página actual y su tamaño depende del contenido. En la esquina superior derecha de la ventana pop up, habrá que añadir un botón para cerrar la misma.
- En la segunda sección:
 - Al escribir un texto y darle al botón, se comprobará que el texto no está vacío y se filtrarán las experiencias de la tercera sección según la(s) palabra(s) buscada(s). Para ello, se esconderán las experiencias que no contienen los términos de búsqueda ni en el título ni en la descripción. Si ninguna experiencia cumple con los criterios, se enseñará un mensaje de error al usuario.
- En la tercera sección:
 - Al pinchar la foto o en el título de cada experiencia, se abrirá una ventana pop up con el mismo diseño de la que se ha descrito para los enlaces del mapa.
 - Las experiencias podrán reordenarse mediante *drag and drop*.
- Las versiones tablet y móvil tendrán que ser completamente funcionales.
- Al diseñar los mensajes de error, habrá que tener en cuenta los principios vistos en las clases magistrales.

Para la resolución del ejercicio, se recomienda consultar las siguientes entradas de JavaScript de la página de w3schools: <http://www.w3schools.com/js/default.asp>, entre otras:

DOM CSS (http://www.w3schools.com/js/js_htmlDOM.asp)

JS Functions (http://www.w3schools.com/js/js_functions.asp)

JS Events (http://www.w3schools.com/js/js_events.asp)

JS HTML DOM (http://www.w3schools.com/js/js_htmlDOM.asp)

JS Cookie (http://www.w3schools.com/js/js_cookies.asp)

Forms Validation (http://www.w3schools.com/js/js_validation.asp)

HTML Forms and Input – Explicación sobre los campos de un formulario

HTML5 Input Types – Explicación sobre los nuevos campos para introducir datos.

Material requerido

Para la realización de los ejercicios no se podrá utilizar ningún editor de páginas HTML, marcos de trabajo o herramientas específicas, como por ejemplo Dreamweaver. No obstante, se recomienda utilizar algún editor gratuito estilo Notepad++, HTML-Kit, Visual Studio Code, Sublime Text, Brackets o Atom. Los profesores de la asignatura no darán soporte de estas herramientas.

Normativa

La realización y entrega del ejercicio de programación vendrá regida por la siguiente normativa. El incumplimiento de alguna de estas normas implicará la imposibilidad de obtener una **nota superior al tres** en el bloque de ejercicios de programación.

Normas de realización

La resolución del ejercicio de programación deberá respetar las siguientes normas:

- El ejercicio deberá realizarse en grupos de dos personas. Los integrantes del grupo deberán ser del mismo grupo de prácticas, no pudiendo cambiar durante los bloques de práctica.
- El ejercicio se resolverá utilizando código válido HTML 5, CSS 3, JavaScript y jQuery.
- El ejercicio deberá probarse en Firefox 79, Chrome 84, o superiores.

IMPORTANTE. La resolución de dudas de carácter particular se realizará en tutorías.

Normas de entrega

La entrega del ejercicio de programación se realizará durante el día de la sesión indicada en la introducción de este enunciado. La entrega deberá respetar las siguientes normas:

- La entrega se realizará a través de Aula Global. Se debe entregar un único fichero comprimido **.zip** o **.rar** de nombre:

ep02_grXX.rar

dónde **XX** corresponde con el identificador del grupo de prácticas. Por ejemplo, el grupo de prácticas 5 del grupo 81 deberá genera un fichero con nombre:

ep02_gr05.rar

La estructura en carpetas de los ficheros entregados deberá ser la siguiente:

- **ej.** Directorio raíz del ejercicio. Ficheros HTML.
- **ej/style.** Hojas de estilo.
- **ej/images.** Imágenes y recursos gráficos.

- **ej/script.** Scripts JavaScript.

Referencias

- [1] “JavaScript Tutorial”, Tutorial JavaScript de W3 Schools, disponible en <http://www.w3schools.com/js>
- [2] “JavaScript tutorials”, disponible en <http://www.javascriptkit.com/javatutors>
- [3] “The JavaScript PocketGuide”, Lenny Burdette, Ed. PeachPit Press, 2010
- [4] “JavaScript Step by Step”, Steve Suehring, Ed. Microsoft Press, 2008
- [5] “Eloquent JavaScript”, disponible en <http://eloquentjavascript.net>
- [6] “jQuery Tutorial”, Tutorial jQuery de W3 Schools, disponible en <http://www.w3schools.com/jquery/default.asp>
- [7] “jQuery Official WebSite”, disponible en <http://jquery.com>
- [8] “Learning jQuery”, Jonathan Chaffer, Ed. Pckt Publishing, 2011