

Interfaces de Usuario

Grado en Ingeniería Informática

Curso 2017 / 2018

Ejercicios de programación

(Enunciado)

TECNOLOGÍAS PARA EL DESARROLLO DE INTERFACES DE USUARIO WEB

Lenguajes de script del cliente (JavaScript&jQuery)

Tabla de contenidos

1. Introducción	2
Dedicación	2
Entrega	2
Evaluación	2
Estructura del documento.....	2
2. Lenguajes de script del cliente	3
Javascript.....	3
Librerías Javascript	4
jQuery.....	4
3. Ejercicios	6
Enunciados	6
Material requerido	10
4. Normativa	11
Normas de realización.....	11
Normas de entrega.....	11
5. Referencias.....	12

1. Introducción

El segundo bloque de ejercicios 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 resolverán tres ejercicios 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 tres semanas de trabajo y tres 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 siete del cuatrimestre (ver la planificación de prácticas publicada en Aula Global como “Calendario de prácticas”). La entrega se limitará a uno de los tres ejercicios de programación. El ejercicio a entregar será diferente para cada grupo de la asignatura y se indicará al inicio de la sesión de entrega.

Evaluación

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

El bloque de ejercicios tendrá un valor de un 33% sobre la nota final de los ejercicios de programación; es decir, un 10% sobre la nota final de la evaluación continua. La copia de los ejercicios tanto de compañeros como de internet implicará el **suspenso directo** de la evaluación continua, tanto para el grupo que copie como para el grupo autor de la versión original (en su caso).

Estructura del documento

El documento se divide en tres secciones en las cuales se presenta la tecnología utilizada en el bloque de ejercicios, los enunciados de los ejercicios a resolver, y la normativa de entrega y realización de dichos ejercicios de programación, y los criterios de evaluación.

2. 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. Ejemplos de lenguajes de script del servidor serían VBScript o JavaScript. En este bloque de ejercicios utilizaremos este último lenguaje.

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] y [4] como manuales de referencia. Ambas publicaciones están disponibles en Safari Books Online.

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 [5], 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 [6] 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 [7] como manual de referencia, disponible en Safari Books Online.

3. Ejercicios

El segundo bloque de ejercicios agrupa un total de tres ejercicios de programación, todos ellos considerados como obligatorios o entregables. A continuación, se enunciarán los ejercicios, proponiendo ejemplos a utilizar como referencia y el material proporcionado para su realización.

Enunciados

Ejercicio 1.

El objetivo de este ejercicio es conocer y practicar con los principales elementos de JavaScript para elaborar un formulario para la realización del registro en una página web.

La página deberá ser la misma del ejercicio 1 de la práctica anterior (Bloque I – HTML5&CSS3), donde se aportarán los siguientes cambios:

- Se añadirá una sección similar a los dos contenedores ya en la página (información personal y facturación) con información propia de la cuenta: nombre de usuario y contraseña.
- Todos los campos en el formulario, incluyendo información de la cuenta, información personal y de facturación, serán editables exceptuando el correo electrónico.
- Al final del formulario, se añadirá un check para “He leído y acepto las Condiciones de uso” y dos botones, “guardar” y “borrar”.
- El formulario quedará constituido por los siguientes campos, algunos de ellos obligatorios y con un cierto formato según lo que se especifica
 - 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)
 - Idioma (Opcional) – Se elige uno entre un listado de idiomas
 - Imagen de perfil (Opcional)
 - Información de facturación:
 - Dirección (Obligatorio)

- Método de pago (Obligatorio) – Dependiendo del método de pago, se cambiarán los campos de la siguiente forma
 - Si se selecciona tarjeta de crédito, se pedirá el número de la tarjeta, la fecha de caducidad y el código de validez
 - Si se selecciona transferencia bancaria, se enseñarán los datos del beneficiario, número de cuenta y banco al que realizar la transferencia
 - Si se selecciona PayPal, se enseñará un enlace a la página de PayPal
 - He leído y acepto las Condiciones de uso (Obligatorio)
 - Botones “guardar” y “borrar”
- Al cargar la página, se abrirá una caja de diálogo modal pidiendo correo electrónico y contraseña. Si ya existe una cookie con el mismo correo electrónico, se cargará el formulario poniendo en cada campo la información guardada en la cookie. Si no existe, se abrirá el formulario con todos los campos vacíos, exceptuando el correo electrónico y la contraseña.
- Al dar al botón “guardar” se almacenará una cookie con el correo electrónico y toda la información contenida en el formulario. Si ya existe una cookie con el mismo correo electrónico, se comprobará si alguna información contenida en los campos ha sido modificada. En caso afirmativo se enseñará una caja de diálogo modal informando del cambio, mientras que en caso negativo se comunicará que no ha habido cambios.
- Al dar al botón “borrar”, se restablecerá la información inicial del formulario.
- La validación del formulario se podrá realizar mediante: HTML5, JavaScript y jQuery.

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:

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

HTML Forms and Input – Explicación sobre los campos que se pueden añadir a un formulario

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

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

Ejercicio 2.

El objetivo de este ejercicio es conocer y practicar con los principales elementos de JavaScript para elaborar una página web con una galería dinámica de videos.

La página deberá ser la misma del ejercicio 2 de la práctica anterior (Bloque I – HTML5&CSS3), donde se aportarán los siguientes cambios:

- Al cargar la página, se abrirá una caja de diálogo modal pidiendo correo electrónico y contraseña. Si existe una cookie con el correo electrónico

especificado, se cargará la página y se pondrá en la parte derecha de la cabecera el nombre de usuario y una imagen de perfil. Si no existe la cookie, se abrirá el ejercicio anterior.

- Al darle al icono de like y de compartir, se incrementará el contador.
- Los otros videos puesto en la galería pueden arrastrarse y soltarse en la parte central de la página. Al soltarse, se intercambiarán con el video en la parte central de la pagina.
- Al darle al icono + debajo de cada video de la galería, se abrirá un texto con una descripción del video.

Para la resolución del ejercicio, se recomienda consultar las siguientes entradas de JavaScript de la página de w3schools, entre otras:

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)

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

Ejercicio 3.

El objetivo de este ejercicio es conocer y practicar con los principales beneficios de la librería jQuery. Para ello se realizará una página web que contenga un juego de memoria donde se tendrán que emparejar las imágenes iguales.

El juego se estructurará en los siguientes pasos.

1. En el primer paso, se le pedirá al usuario el numero N de imágenes con las que quiere jugar, limitado entre 3 y 10, y el tiempo de juego en segundos, limitado entre 10 y 120.
2. Una vez que el usuario haya elegido el numero de imágenes, se generará un tablero con $N \times 2$ casillas con un contador de parejas encontradas, un timer y un botón “reiniciar”. Cada casilla puede contener un color de fondo igual para todas o una imagen. En total, habrá N parejas de casillas con la misma imagen. Inicialmente, todas las casillas enseñarán la cara con el color de fondo escondiendo la imagen.
3. Al hacer click encima de una casilla, la casilla “se dará la vuelta” enseñando la imagen que tiene asociada.
4. Si se hace click encima de dos casillas, se comprobará si las imágenes asociadas son iguales. Si son iguales, se incrementará el contador y se dejarán las casillas dadas la vuelta. Si no son iguales, se volverán a esconder las imágenes.
5. Si se emparejan todas las casillas, el contador enseñará el numero N (numero total de parejas), se parará el timer, y se enseñará al usuario un mensaje informándole que ha ganado y en cuanto tiempo.
6. El timer irá descontando los segundos a partir del numero indicado en el paso 1 por el usuario. Si llega a 0 antes de que el usuario pueda emparejar todas las casillas, se enseñará un mensaje informando de que se ha perdido la partida. **La implementación del mecanismo del timer es opcional.**

7. Al darle al botón “reiniciar”, se reiniciará el juego a partir del paso 1. En cada iteración del juego, se tiene que cambiar la distribución de las casillas en el tablero.

Para la resolución del ejercicio, se recomienda consultar la página oficial de jQuery (<http://jqueryui.com/>) y entradas de JavaScript, HTML5 y CSS3 animations.

Material requerido

Para la realización de los ejercicios no se requerirá de ningún editor o herramienta específica. No obstante, se recomienda utilizar algún editor gratuito estilo Notepad++ o HTML-Kit. Por otro lado, para la resolución de errores, **se recomienda utilizar algún depurador de código Javascript**. Un ejemplo de depurador para Internet Explorer sería The DebugBar, mientras que para Firefox se recomienda utilizar la extensión Firebug. Los profesores de la asignatura no darán soporte de estas herramientas.

4. Normativa

La realización y entrega de los ejercicios 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 de los ejercicios de programación deberá respetar las siguientes normas:

- Los ejercicios deberán realizarse en grupos de dos personas. Los integrantes del grupo deberán ser del mismo grupo de prácticas, no pudiendo cambiar durante los ejercicios de programación.
- Todos los ejercicios deberán probarse en las últimas versiones de los navegadores: Firefox, Chrome e Internet Explorer.
- Los ejercicios se resolverán utilizando JavaScript, HTML5 y CSS3.
- La resolución de referencia para la resolución de los ejercicios será de 1024x768.

IMPORTANTE. La resolución de dudas de carácter particular se realizará en tutorías. En ningún caso se resolverán dudas por correo electrónico.

Normas de entrega

La entrega del ejercicio de programación correspondiente se realizará al comienzo 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 la actividad habilitada para el caso.

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:

- **ej0N.** Directorio raíz del ejercicio. Ficheros HTML.
- **ej0N/style.** Hojas de estilo.
- **ej0N/images.** Imágenes y recursos gráficos.
- **ej0N/script.** Ficheros Javascript.

dónde **N** se corresponde con cada uno de los números de los ejercicios de programación (de 1 al 5) que deberán entregarse. Por ejemplo, aquel grupo que deba entregar el ejercicio 3 deberá tener un directorio raíz de nombre ej03.

IMPORTANTE. En ningún caso, se aceptarán prácticas entregadas por correo electrónico, en tutorías, o fuera del horario de la sesión correspondiente.

5. 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] “*jQuery Tutorial*”, Tutorial jQuery de W3 Schools, disponible en <http://www.w3schools.com/jquery/default.asp>
- [6] “*jQuery Official WebSite*”, disponible en <http://jquery.com/>
- [7] “*Learning jQuery*”, Jonathan Chaffer, Ed. Pckt Publishing, 2011