

Interfaces de Usuario

Grado en Ingeniería Informática

Curso 2025 / 2026

TECNOLOGÍAS WEB - JAVASCRIPT

Tabla de contenido

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	10
NORMATIVA.....	10
NORMAS DE REALIZACIÓN	10
NORMAS DE ENTREGA	10
REFERENCIAS.....	11

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 cuatro 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”) antes de la sesión en aula pequeña (grupo reducido).

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 **suspenseo 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 w3cschools [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. A continuación, se presentará el enunciado, así como el material proporcionado y requerido para su realización.

Enunciado

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

- Galería de imágenes (*Carrusel*) de las páginas *Home* y *Version b*:
 - Contendrá un mínimo de 3 packs de viaje para mostrar en modo carrusel.
 - Al hacer clic en los botones con flechas se avanzará o retrocederá en los diferentes packs. Si al estar en el primero se presiona el botón ">", se pasará al segundo. En cambio, si se presiona estando en el tercero, se regresará al primero. Lo mismo pero en sentido inverso con el botón "<".
 - Además de permitirse iterar sobre los packs manualmente, estos también cambiarán automáticamente sin necesidad de que se pulsen los botones cada 2 segundos.
 - Al hacer clic en el botón *Comprar* se accederá a la página *Versión c* correspondiente al pack de viaje que se pretende comprar.
- Página *Home*:
 - Se implementará la funcionalidad asociada al botón *Iniciar sesión*. Al hacer clic en él, deberá comprobarse la existencia de los datos introducidos en los campos *Usuario* y *Contraseña* en una *cookie* o en *local storage*. Si los datos han sido registrados previamente, se iniciará sesión, dirigiéndose a la página *Version b*. Si los datos no se encuentran entre los registrados anteriormente, el inicio de sesión no se habrá podido realizar y se mostrará una ventana emergente con un mensaje de error al usuario.
 - Al hacer clic en el botón *Registro*, se accederá a la página de registro *Version a*, la cual contendrá un formulario que permitirá ingresar los datos del usuario.

- **Página *Versión a*:**

- Campo de texto *Nombre** (admitirá mínimo 3 caracteres de longitud).
- Campo de texto *Apellidos** (compuesto por mínimo dos cadenas de caracteres de 3 caracteres de longitud cada una).
- Campo de texto *Correo electrónico** (admitirá valores tipo [nombre@dominio.extensión](#)).
- Campo de texto *Confirmar correo electrónico** (igual al campo anterior).
- Campo de inserción de fecha para *Fecha de nacimiento* (abrirá un calendario que permitirá elegir la fecha de nacimiento). Deberá controlar los posibles valores introducidos para evitar datos irreales.
- Campo de texto *Login** (representará el nombre de inicio de sesión y estará formado por mínimo 5 caracteres de longitud).
- Campo de contraseña *Password** (8 caracteres de longitud, con mínimo 2 números, 1 carácter especial, 1 letra mayúscula y 1 letra minúscula).
- Campo de inserción de archivo para *Subir imagen de perfil** (abrirá carpeta de imágenes para seleccionar una imagen). Los únicos formatos de archivo admitidos serán *webp, png y jpg*.
- Campo de marcado *Acepto la Política de privacidad* (checkbox). Conforme sea marcado/desmarcado habilitará/deshabilitará el botón *Guardar datos y acceder*.
- Botón *Guardar datos y acceder*. Al hacer clic sobre él, se controlará que se haya aceptado la política de privacidad del campo previo. Además, deberán validarse los campos obligatorios (los que vienen marcados con *) y el formato de cada uno conforme a lo indicado anteriormente. Al terminar la comprobación, los datos del usuario se guardarán en una cookie o como alternativa en el *local storage*. Al guardar los datos, se enseñará un mensaje informando al usuario que se ha registrado correctamente. En caso de errores o falta de información, se informará al usuario a través de un mensaje también. Si el registro ha sido satisfactorio, se iniciará sesión, llevándose a la página *Versión b*

- **Página *Versión b*:**

- Se controlará el acceso a esta página, no permitiéndose el mismo sin haberse realizado previamente un inicio de sesión o registro válidos.
- El nombre y la imagen de perfil del usuario ahora sí deben ser los introducidos en el formulario de "Registro" (a diferencia del ejercicio anterior en el que eran simulados).

- Se implementará la funcionalidad asociada al botón *Cerrar sesión*. Al hacer clic en él, se abrirá una ventana emergente que pedirá confirmación con el texto “¿Desea cerrar sesión?” y dos botones que mostrarán los textos: “Confirmar” y “Cancelar”. Si el usuario hace clic en “Cancelar” se cerrará la ventana no llevándose a cabo ninguna otra acción. Si el usuario hacer clic en “Confirmar”, se abrirá la página de inicio *Home* y se deberá cerrar la sesión.
- La sección “Últimos consejos” mostrará los tres últimos consejos que hayan sido añadidos (independientemente del usuario que lo haya hecho). Se mostrarán mediante un texto hipervinculado a la página en la cual se muestre el consejo completo (título y descripción). La página en la que se muestre no tiene que ser real en este caso.
- Al pulsar el botón *Enviar* del formulario que permite introducir consejos, se comprobará si el valor introducido en el título tiene una longitud mínima de 15 caracteres. Se hará la misma comprobación con la descripción (en este caso con una longitud mínima de 30). En caso de cumplir ambos campos con lo indicado, el consejo se añadirá al comienzo de la sección “Últimos consejos” y se almacenará en la lista de consejos en *local storage*, permitiéndose así que sea mostrado también en sucesivos inicios de sesión incluso de otros usuarios.
- **Página Versión c:**
 - Campo *Nombre completo** (admitirá mínimo 3 caracteres de longitud).
 - Campo de texto *Correo electrónico** (admitirá valores tipo [nombre@dominio.extensión](#)).
 - Campo *Tipo de tarjeta** (inicialmente seleccionado un valor vacío para forzar a que el usuario seleccione entre *Visa*, *Mastercard* o *American Express*).
 - Campo *Número de tarjeta** (por simplicidad se admitirá como válido un número de tarjeta formado por 13, 15, 16 o 19 dígitos).
 - Campo *Nombre del titular de la tarjeta** (admitirá mínimo 3 caracteres de longitud).
 - Campo de inserción de fecha para *Fecha de caducidad** (abrirá un calendario que permitirá elegir la fecha de caducidad). Deberá controlarse que la fecha no haya expirado en el momento de la compra.
 - Campo *CVV** (admitirá 3 dígitos).
 - Botón *Comprar*. Al hacer clic sobre él, deberán validarse los campos obligatorios (los que vienen marcados con *) conforme a lo indicado anteriormente. Si son válidos

se simulará la compra mostrando una ventana emergente con el mensaje “Compra realizada”. En caso contrario se informará del error.

- Botón *Borrar*. Al hacer clic sobre él se limpiará el formulario, devolviéndolo a su estado original.
- Las versiones tablet y móvil tendrán que implementar la misma funcionalidad que la versión desktop.

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)

Asynchronous JavaScript (https://www.w3schools.com/js/js_asynchronous.asp)

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

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

HTML Web Storage – Explicación sobre el almacenamiento local de 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 similares. 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 y Chrome, asegurándose que se vea y funcione correctamente.

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