

Prácticas

Estilo: ▼ Tamaño letra: ▼ Interlineado: ▼

Desarrollo de Aplicaciones Web

Instrucciones generales

1. Objetivos

- Establecer unas pautas comunes para el desarrollo de las prácticas.

2. Introducción a la asignatura

La asignatura “Desarrollo de Aplicaciones Web” está dividida en dos partes, la programación del lado del cliente y la programación del lado del servidor.

Desde el nacimiento de la Web en el año 1990, se han desarrollado diferentes tecnologías para la programación de las aplicaciones web, tanto del lado del cliente como del servidor. A la hora de elegir una tecnología, un requisito imprescindible es garantizar la compatibilidad con los diferentes navegadores web que existen. Por ello, la mejor opción es emplear siempre las tecnologías que tienen la consideración de “estándar de facto”.

Las tecnologías que hoy en día son consideradas el estándar del cliente web son HTML, CSS, JavaScript y DOM. De estas cuatro tecnologías, en esta asignatura se estudiará lo mínimo para poder desarrollar una aplicación web con una funcionalidad básica. El estudio profundo de estas tecnologías está reservado para la asignatura “Programación de Clientes Web”.

En la programación del lado del servidor existe más libertad, ya que lo único que se debe cumplir es que el resultado que genere el servidor web sea HTML, CSS, JavaScript y DOM.

En esta asignatura, en la programación del lado del servidor se utiliza XAMPP (pero se puede usar otra plataforma similar, no es problema) para poder trabajar con PHP y MySQL.

3. Usabilidad y accesibilidad

La usabilidad y la accesibilidad del software es un tema muy importante que normalmente los desarrolladores no tienen en cuenta. Un desarrollador de software normalmente no tiene problemas para usar sus propios desarrollos porque, al fin y al cabo, son sus desarrollos y los conoce a la perfección. Pero ¿otras personas los podrán usar?

En el plan de estudio existe otra asignatura centrada en la usabilidad y la accesibilidad, pero ello no es excusa para que cualquier desarrollo deba ser usable y accesible. Por ejemplo, cualquier aplicación debe:

- No cambiar el comportamiento normal de los elementos de la interfaz de usuario.
- Proporcionar una respuesta (*feedback*) a cualquier acción que realice el usuario.
- Mostrar mensajes de error claros y significativos, que ayuden al usuario a entender lo que está pasando en todo momento.

En el artículo “Basic usability, basically missed” [\(1\)](#) se proporcionan algunos consejos básicos de usabilidad:

- Navigation Inconsistency.
- Structure Inconsistency.
- Returning Home.
- Signing in/Retrieving quotes.
- Letting the User Know Where They Are.
- Removal of the Hamburger Menu.
- Using Blue for Non-link Text.
- Repetitive Links.

Conviene que consultes estos consejos para no caer en los errores típicos que tienen los sitios y aplicaciones web.

4. Cómo se corrige la práctica

Las prácticas de esta asignatura son acumulativas, por lo que no es posible “saltarse” una práctica. Si por cualquier razón, una práctica no es entregada en la fecha establecida, la práctica no será recuperada posteriormente, pero sí que será necesario realizarla porque las prácticas posteriores dependerán de ella.

Además, como las prácticas son acumulativas, tienes que cumplir exactamente lo que se te pida en cada práctica, si te desvías de lo que se te pida es posible que tengas graves problemas en prácticas posteriores.

5. Documentación de las prácticas

Documentar una aplicación es aburrido, pero es algo necesario. En esta asignatura no se exige que las prácticas tengan una documentación independiente, pero sí que se exige que el código esté correctamente organizado y documentado.

6. Sobre las aplicaciones de una sola página

Según Google, una aplicación de una sola página [\(2\)](#) (*single-page application*, SPA) es:

Una aplicación de una sola página es una aplicación web o un sitio web que carga todos los recursos necesarios para navegar por el sitio en la primera carga de la página. A medida que el usuario hace clic en los enlaces e interactúa con la página, el contenido posterior se carga de forma dinámica. Con frecuencia, la aplicación actualiza la URL en la barra de dirección para emular la navegación de página tradicional, pero no se vuelve a hacer otra solicitud de página completa.

En ese pequeño párrafo de Google ya aparece uno de los problemas de las SPA, se pierden los URL y para tener ciertas funcionalidades como los marcadores (*bookmarks*), el volver atrás o el posicionamiento en los buscadores es necesario que “la aplicación actualiza la URL en la barra de dirección para emular la navegación de página tradicional”.

Antes de usar SPA te tendrías que hacer la siguiente pregunta: ¿realmente necesito SPA? En el artículo “You probably don’t need a single-page application” [\(3\)](#) dan varios consejos para responder a esa pregunta. También es muy interesante el artículo “The disadvantages of single page applications” [\(4\)](#). Y otro problema muy importante son los problemas de accesibilidad que presentan las SPA, tal como se explica en “One-page-applications are not accessible” [\(5\)](#).

Por todo ello, **en esta asignatura no se requiere que la práctica se desarrolle como una SPA, es más, se desaconseja totalmente.**

7. Sobre el scroll infinito

El scroll infinito es otra técnica de desarrollo web moderna que “mola mucho”, pero que en la mayoría de las situaciones causa problemas graves. Los artículos “Infinite Scroll: Let’s Get To The Bottom Of This” [\(6\)](#) y “Infinite Scrolling Is Not for Every Website” [\(7\)](#) explican los problemas que presenta el scroll infinito.

Es más, se cree que el scroll infinito causa adicción entre los usuarios y hay incluso un movimiento para pedir su prohibición: “Social media apps are ‘deliberately’ addictive to users” [\(8\)](#) y “US could ban ‘addictive’ autoplay videos and infinite scrolling online” [\(9\)](#).

Por ello, **en esta asignatura tampoco se debe hacer uso del scroll infinito.**

8. Sobre los frameworks

Según la Wikipedia [\(10\)](#), un framework es “un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar”.

En el desarrollo web existen frameworks para casi todas las tecnologías que se emplean. Hasta HTML tiene sus propios frameworks, o más bien, plantillas para comenzar a desarrollar una página web. Dos de los más famosos son HTML5 Boilerplate [\(11\)](#) e Initializr [\(12\)](#). Para las otras tecnologías que se emplean en esta asignatura (CSS, JS y PHP) existen cientos de frameworks.

Un framework es una gran ayuda para el desarrollo de grandes aplicaciones. Cuando un framework se usa correctamente, guía el desarrollo, reduce las posibilidades de error y mejora la productividad. Sin embargo, cuando se está aprendiendo una nueva tecnología o paradigma de desarrollo, un framework dificulta el aprendizaje porque oculta muchos detalles que es importante conocer. Por eso, **en esta asignatura no se puede usar un framework.**

9. Consejos generales

- Todas las páginas deben tener una presentación consistente. Para ello, deben tener unos elementos comunes que se repitan en todas las páginas (cabecera y pie de página, barra de navegación). En el pie de página incluye los nombres de los autores de la práctica, un aviso de copyright con el año y alguna información más.
- Almacena cada práctica en un directorio independiente; al final de la asignatura debes tener un directorio por cada práctica semanal. Cuando comiences una nueva práctica, copia y renombra el directorio de la práctica anterior.
- El fichero de la página principal de un sitio web suele tener el nombre index.html. Sólo como curiosidad, los servidores web suelen tener configurada la siguiente lista de páginas por defecto para la página principal [\(13\)](#): index.htm, index.html, index.php, index.php3, index.php5, index.php4, index.shtml, default.htm, default.html, index.py, default.shtml, index.pl, index.cgi y home.html.
- Se recomienda usar el juego de caracteres UTF-8.
- Ten cuidado con los nombres de los ficheros, utiliza únicamente letras del alfabeto inglés y números, no uses espacios en blanco y emplea únicamente minúsculas.
- Ten cuidado con la caché del navegador, consulta una explicación sobre los problemas y soluciones en el artículo “Ayuda: Cómo limpiar la caché” [\(14\)](#). Recuerda esto durante toda la asignatura.

[\(1\) https://uxdesign.cc/basic-usability-basically-missed-f05fd69d2c13](#)

[\(2\) https://developers.google.com/analytics/devguides/collection/analyticsjs/single-page-applications?hl=es-419](#)

[\(3\) https://journal.plausible.io/you-probably-dont-need-a-single-page-app](#)

[\(4\) https://adamsilver.io/articles/the-disadvantages-of-single-page-applications/](#)

[\(5\) http://www.craigabbott.co.uk/one-page-applications-are-not-accessible](#)

- (6) <https://www.smashingmagazine.com/2013/05/infinite-scrolling-lets-get-to-the-bottom-of-this/>
- (7) <https://www.nngroup.com/articles/infinite-scrolling/>
- (8) <https://www.bbc.com/news/technology-44640959>
- (9) <https://www.theguardian.com/media/2019/jul/31/us-could-ban-addictive-autoplay-videos-and-infinite-scrolling-online>
- (10) <https://es.wikipedia.org/wiki/Framework>
- (11) <https://html5boilerplate.com/>
- (12) <http://www.initializr.com/>
- (13) *El orden de configuración es importante porque el servidor web va a intentar encontrar los archivos en el orden indicado.*
- (14) https://es.wikipedia.org/wiki/Ayuda:C%C3%B3mo_limpiar_la_cach%C3%A9