

**TECNOLÓGICO NACIONAL DE MÉXICO**

**INSTITUTO TECNOLOGICO DE MERIDA**

**INGENIERÍA EN SISTEMAS COMPUTACIONALES**



**DESARROLLO FRONTEND**

**PROFESOR:** [**Héctor Jesús Cetina Cordero**](http://tecvi.itmerida.mx/user/view.php?id=1831&course=1)

[Tarea de investigación sobre las aplicaciones responsiva con UX](http://tecvi.itmerida.mx/mod/resource/view.php?id=76287)

**Unidad 1**

**Grupo: 9SA**

**ALUMNO(A)**

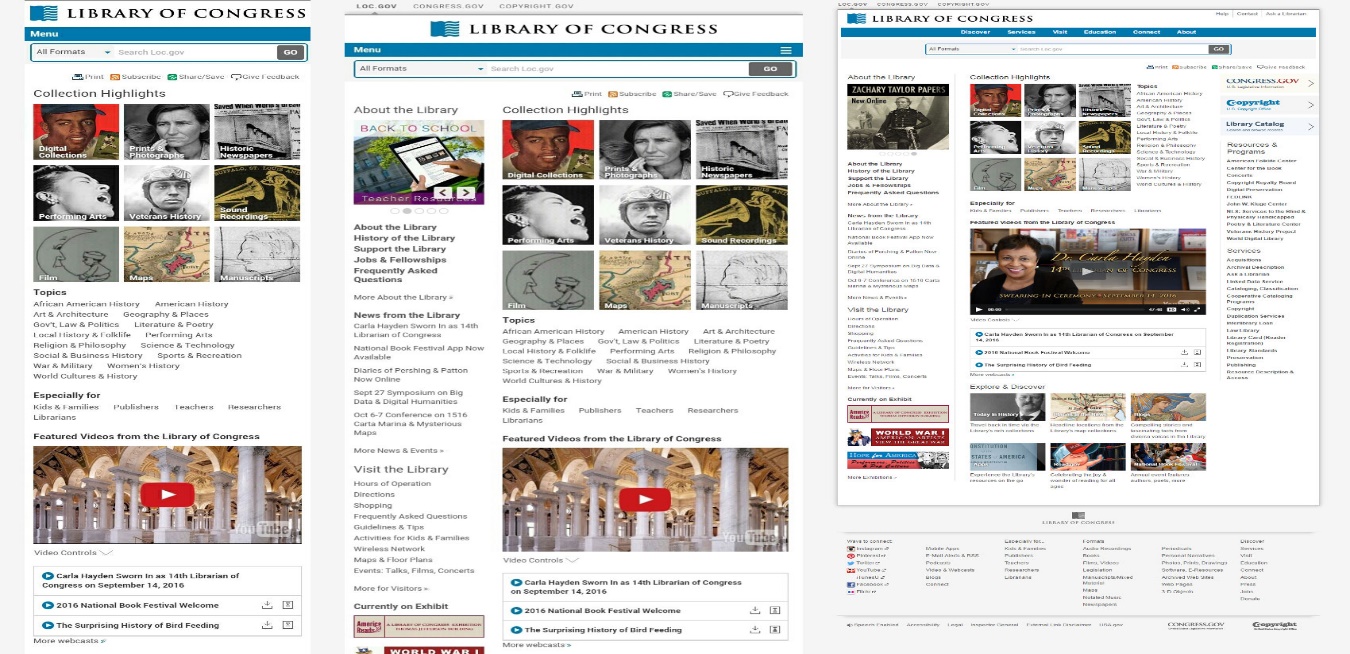
* **Alcala Morales Aldo Gaspar**

**Fecha: 6/09/2019**

El término responsive web design (RWD) fue utilizado por primera vez por el diseñador y desarrollador web Ethan Marcotte en su libro homónimo publicado en 2011 (Marcotte, 2011).

Se trata de una técnica de desarrollo de sitios y aplicaciones web que permite adaptar sus contenidos a las características y los diferentes tipos de pantalla de los diversos dispositivos desde los cuales se realiza la consulta.

La filosofía final de esta técnica procura evitar los diseños a medida y desconectados para cada tipo de dispositivo, para tratarlos como facetas de la misma experiencia. Es, por tanto, una técnica alineada con el concepto one web defendido por el W3C.



A pesar de la precisa traducción al catalán propuesta por el Termcat, tanto en nuestro territorio como en la mayoría de países hispanohablantes se utilizan indistintamente los términos diseño web responsivo, diseño web adaptativo e, incluso, diseño web adaptable para referirse a la técnica propuesta por Marcotte. Es necesario precisar que, en el ámbito de especialidad que nos ocupa, se distingue entre estas dos técnicas. La gran diferencia entre el diseño web responsivo y el diseño web adaptativo es que mientras que, a través de diferentes soluciones técnicas que parten de una única versión del código fuente, el primero permite adaptar los contenidos del sitio web o de la aplicación a cualquier tamaño de pantalla, en el segundo los contenidos se adaptan a esos mismos dispositivos o pantallas gracias a la detección de las características específicas del dispositivo y a la selección de la versión del código fuente adecuada según esas características.

Técnicamente, se implementa con el uso de las reglas media queries, un módulo del lenguaje CSS que permite definir diferentes estilos según el tipo de medio y la resolución de pantalla del dispositivo. En el ejemplo siguiente, la regla CSS aplicada sobre el selector “.fondoverde” sólo se activará si la consulta se realiza desde una pantalla (screen) y la ventana gráfica (viewport) presenta un ancho igual o inferior a 480px.

@media screen and (max-width: 480px) { .fondoverde { background-color: lightgreen; }

Comparativa entre el desarrollo de una versión móvil y el de un sitio web responsivo.

|  |  |  |
| --- | --- | --- |
| Concepto | Versión móvil | Web responsivo |
| Desarrollo | El tiempo y el coste de desarrollo acostumbra a ser superior al de un sitio web responsivo, al tratarse de un producto diferenciado del web para dispositivos de sobremesa. | El tiempo y el coste de desarrollo acostumbra a ser inferior al de una versión móvil porque se trata de un único producto para todas las plataformas. |
| Mantenimiento | Más complicado y caro porque se han de mantener dos versiones del código fuente (sitio web principal y versión móvil) | Más fácil porque se ha de mantener una sóla versión del código fuente. Por contra, cualquier cambio en la interfaz deberá ser testeado en multiples dispositivos. |
| Experiencia de usuario | Ofrecen una experiencia de usuario específica, optimizada para dispositivos con pantallas reducidas. Las tareas complejas se simplifican o se omiten, para pasar a estar sólo disponibles en las versiones principales del sitio web. | Procuran ofrecer una experiencia de usuario similar con independencia del dispositivo. Esto no necesariamente es siempre positivo. En ocasiones, no basta con adaptar los mismos elementos y patrones de interacción a una pantalla más pequeña, sino que se hace necesario rediseñarlos para garantizar una buena UX. |

El diseño UX o “experiencia de usuario” es, lo creamos o no, el elemento más importante de una página web. Podemos tener una web preciosa y perfectamente diseñada que, si no garantiza una buena experiencia de usuario, no nos sirve de nada.

El diseño UX (User Experience) consiste en planear y diseñar cómo va a ser la experiencia de un usuario cuando entra en una página web y navega por ella. Es, en definitiva, el diseño enfocado a los usuarios.

Que el diseño UX es importante no sólo lo digo yo. Google ha dejado claro que lo más importante para él es la experiencia de los usuarios en las páginas web. Una página puede tener todo el SEO del mundo que si Google no considera que es user friendly, no nos la va a posicionar.

Para hacer un buen diseño UX simplemente tenemos que hacerlo pensando en los usuarios y pondiéndoselo lo más fácil posible. Si para llegar a un sitio tiene que clicar un botón, ese botón tiene que ser bien visible y estar claro que es un botón. Parece sencillo y de sentido común, pero a la hora de crear una página web, son tantos los elementos que tenemos que tener en cuenta que al final nos acabamos olvidando de que lo más importante es que sea útil para el usuario.

También tienes que tener muy en cuenta una cosa y es que los usuarios no están en tu mente. Es muy probable que tú tengas una idea muy clara y la entiendas a la perfección. Esto no quiere decir que el resto del mundo la vaya a entender igual. No es que los usuarios sean tontos, como se creía en la publicidad de los años 60. Al contrario, los usuarios son muy listos y saben perfectamente lo que quieren. Sobre todo a día de hoy que la mayoría de las personas tienen acceso a Internet de una forma u otra.

**Lenguaje JavaScript**

Se utiliza principalmente del lado del cliente aunque se puede utilizar del lado del servidor. Actualmente y gracias a tecnologías como AJAX es utilizado para enviar y recibir información del servidor.

Como principales ventajas, tenemos que destacar que es un lenguaje de scripting seguro y fiable, cuyos scripts tienen capacidades limitadas, debido a la seguridad.

Como desventajas, podríamos mencionar que el código debe descargarse por completo y es visible por cualquier usuario. JavaScript

**Frameworks** Angular 2+

**Lenguaje PHP**

Es un lenguaje enfocado en la creación de webs dinámicas. Sus scripts son interpretados por el servidor y genera código HTML. Requiere Apache o IIS con librerías de PHP. Hereda su sintaxis de C, Java y Perl.

Como principales ventajas, hemos de decir que es un lenguaje fácil de aprender y muy rápido. Soporta la orientación a objetos y utiliza un lenguaje multiplataforma. Además, puede conectarse con una gran cantidad de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server… No necesita que se definan los tipos de variables. Uno de sus aspectos más llamativos es que está diseñado con el fin de ser un lenguaje muy seguro para escribir CGI, más que Perl o C.

Es el lenguaje base que utilizan la mayoría de CMS o gestores de contenidos más extendidos como WordPress, PrestaShop, Drupal o Joomla!

**Frameworks de php**

* Laravel
* Symfony
* Phalcon
* Zend Framework

**Lenguaje Python**

Considerado por muchos el lenguaje más limpio a la hora de programar. El código, al igual que JavaScript, es interpretado y no compilado.

Algo curioso en este lenguaje es que permite a los programadores elegir un estilo de programación concreto (objetos, estructurado, funcional…), debido a que es un lenguaje de programación multiplataforma.

Como ventajas de Python, destacamos que es libre y de fuente abierta, de propósito general. Cuenta con muchas funciones y librerías y es multiplataforma y fácil de programar. Por otro lado, su principal desventaja es que, al ser un lenguaje interpretado, es bastante lento.

**Frameworks**

* Django. Django is a free and open-source Python framework that enables developers to develop complex code and applications effectively and quickly. ...
* CherryPy. CherryPy is an open-source Python web development framework that implants its very own multi-strung server. ...
* Pyramid. ...
* TurboGears. ...
* Web2Py. ...
* Bottle. ...
* Tornado. ...
* Flask.

**Lenguaje Ruby**

Como el anterior, es un lenguaje interpretado y está orientado a objetos. Hereda su sintaxis de Phyton y Perl. El lenguaje puede cargar librerías de extensiones dinámicamente si el sistema operativo lo permite. Además, es un lenguaje portátil.

Otra ventaja que ofrece es que cualquiera puede encontrar en su página web gran cantidad de información y tutoriales.

El desarrollo web es una tarea que requiere mucho tiempo, por lo que elegir un lenguaje para tu hosting con el que sentirse cómodo es crucial.

**Frameworks**

* Ruby on Rails. Ruby on Rails is the most popular web framework for Ruby programming language. ...
* Sinatra. Unlike other web frameworks for Ruby, Sinatra is designed as a domain specific language (DSL). ...
* Padrino
* Cuba
* Hanami
* Roda.
* Goliath.
* Scorched

Página web creada con Ruby

1. **Twitter.** Quizá sea una “trampa” en la lista, pero había que mencionar a esta enorme red social. Millones de usuarios de todo el mundo la utilizan a diario. En sus inicios, se creó sobre Ruby, aunque a día de hoy cambió a Scala para gestionar mejor la gran cantidad de mensajes en tiempo real.
2. **Airbnb.** Se trata de una empresa y una plataforma de software dedicada a la oferta de alojamientos a particulares y turísticos. El nombre es un acrónimo de airbed and breakfast (colchón inflable y desayuno). Airbnb tiene una oferta de más de 2.000.000 propiedades en 192 países y más de 33.000 ciudades.
3. **Github.** No podíamos olvidarnos de la mayor red social de desarrolladores a nivel mundial. El repositorio de código de Github está construido sobre Ruby, tiene más de 10 millones de usuarios y almacena más de 25 millones de repositorios.

Página web creada con Python

1. Instagram
2. Google
3. Spotify
4. Netflix