

# Diseño para Gestores de contenidos

CMS. Content Management System

## Presentación

» Daniel Julià

djulia@eina.edu

twitter: @daniel\_julia

## Programa

Introducción a los CMS

Wordpress

Drupal

Enfocado al diseño, pero también el contenido

Ejercicios con cada CMS

## CMS

“Un sistema de gestión de contenidos (en inglés Content Management System, abreviado CMS) es un programa que permite crear una estructura de soporte (o framework) para la **creación y administración de contenidos**, principalmente en páginas web, por parte de los participantes.”

(Wikipedia)

# CMS

Un gestor de contenidos (CMS) es una aplicación web usada para **crear, editar, gestionar y publicar** contenido digital. La gestión de los contenidos se hace normalmente usando un **navegador web** convencional y a través de un **panel de administración**, al cual tiene acceso el administrador o los editores de la publicación.

Una característica común a todos los CMS es que no son necesarios conocimientos informáticos avanzados para poder gestionar el contenido de un sitio web. Un **editor** de contenidos necesitará saber como manejar la herramienta pero en principio no necesita tener conocimientos de html, css o lenguajes de servidor.

# Evolución de los contenidos en internet

- Estáticos
- Dinámicos (base de datos)
- Gestores de contenidos

# CMS

Los CMS tienen un **panel de administración** donde podemos editar todo el contenido

## Ejemplos

# Ejemplo

afuster.masterdg.com

afuster.masterdg.com/wp-admin

## Ejemplo

- Editando contenidos
- Creando contenidos
- Cambiando estructura del sitio (bloques, tipos de contenido)
- Cambiando el "tema"

## Ventajas de usar un CMS

- No es necesario saber programar para usarlos
- Código abierto. Seguridad
- Comunidad. Solución de dudas. Documentación
- Libertad (casi) total en el diseño
- Se pueden ampliar las funcionalidades fácilmente (plugins)
- Navegación coherente (usabilidad), SEO, etc...

## Tipos de CMS

- <http://www.cmsmatrix.org/>
- [http://en.wikipedia.org/wiki/List\\_of\\_content\\_management\\_systems](http://en.wikipedia.org/wiki/List_of_content_management_systems)
- Cada CMS tiene una estructura de contenidos diferente
- Siempre se pueden ampliar mediante plugins/extensiones/módulos

## Clasificación

Según el lenguaje del servidor

- Java
- **PHP**
- Ruby
- Python
- etc....

## Clasificación

Según la propiedad del código

- De **código abierto** (open source). La licencia GNU (General Public License) es la más extendida
- Código propietario

## Ventajas del código abierto

- Coste nulo o bajo.
- Coste de alojamiento más bajo que soluciones propietarias, ya que las aplicaciones del servidor también se basan en plataformas de código abierto, como "LAMP" (Linux, Apache, Mysql, PHP)
- Acceso al código fuente para hacer cualquier modificación que creamos conveniente
- Posiblemente, la más importante, en muchos casos existe una comunidad de desarrolladores que se encargan de mantenerlo y de su actualización permanente
- Solución de dudas a través de esta comunidad.

## Clasificación

Según el **tipo de aplicación** para el cual está pensado

- Creación y administración de blogs
- Portales de empresas
- Entornos educativos
- Redes sociales
- Tiendas
- etc..

## Componentes que necesitamos en el servidor

### LAMP

- **L**inux . Sistema operativo
- **A**pache . Servidor web
- **M**ysql . Base de datos
- **P**hp . Lenguaje del servidor

XAMP (X= Mac Windows o Linux)

## ¿Que necesitaremos saber?

- Casi nada de Apache
- Muy poquito de mysql
- Un poco de PHP
- **Bastante html**
- **Bastante de como funcionan los CMS**
- **Mucho CSS**

## Servidor

- ftp (cuenta específica)
- mysql (cuenta específica, gestor de bases de datos)
- panel de administración del servidor (ejemplo)

## Alojamiento web

- Servidor compartido
- Servidor virtual privado
- Servidor dedicado

Ejemplo: facebook tiene más de 50.000 servidores para su funcionamiento

## Mantenimiento CMS

- Backups
- Actualizaciones (seguridad)

## Servidor de pruebas (local)

- No es necesario FTP!
- Xampp
- Probarlo

## Que CMS vamos a ver

- Wordpress
- Drupal

## Un CMS para Blogs, Wordpress

- **Wordpress** <http://www.wordpress.org>
- Ejemplo [www.rac1.org](http://www.rac1.org)
- Ejemplos: <http://wpinspiration.com/tag/red/#img-669>

## Drupal

Más "genérico"  
[www.drupal.org](http://www.drupal.org)

- Gestión de usuarios y roles
- Búsqueda avanzada
- Gestión de bloques
- Caché y "throttling"
- URLs limpias
- Publicación en RSS
- Integración con OpenID
- Posibilidad de agregar feeds
- Roles
- Multi-idioma
- etc,...

## + Drupal

- Posibilidad de añadir módulos,...
  - e-commerce
  - e-learning
  - redes sociales
  - etc...
- [www.whitehouse.gov](http://www.whitehouse.gov)
- [www.ubuntu.com](http://www.ubuntu.com)
- Más ejemplos  
<http://www.java360.com/showcases/showcase-of-amazing-drupal-powered-websites/>
- Un 1% de todas las web están hechas en Drupal!

## Wikis

- Wikipedia ? ([MediaWiki](#))
- Muchos otros CMS para Wikis

## E-learning

- Moodle

<http://moodle.org/>

## Redes sociales

- <http://www.pligg.com>
- <http://www.meneame.com>
- <http://www.elgg.com>

## Otros CMS

- Google sites
- Joomla
- Foros, Phpbb <http://www.phpbb.com/>
- etc...

## CMS vs Frameworks

- Frameworks
  - Para programadores
  - Flexibilidad total
  - En el caso de webs basadas en contenidos, no merece la pena invertir en un desarrollo en framework en lugar de CMS
  - Ejemplos, code igniter, Zend, Yii, etc...

## Objetivos de la asignatura

- Poder adaptar "cualquier" diseño a una plantilla (theme) de un CMS determinado
- Conocer las posibilidades de los CMS a nivel de administrador, para personalizar la estructura de contenidos
- Poder crear un sitio web complejo utilizando un CMS como Wordpress o Drupal

## Creación de "themes"

3 posibilidades (de menos a más complejo)

- Personalizando el css de un tema ya existente  
Creando un tema "hijo" de otro que ya existe
- Utilizando un tema "framework"
- Creando un tema desde cero

## Resumen

Un CMS es un sistema que **simplifica** la creación y mantenimiento de una página web. Existen CMS adaptados a cada **tipo de aplicación**. Uno de los objetivos principales de cualquier CMS es **separar el contenido de la presentación**