

El Drupal

Daniel Julià Lundgren

PID_00168279



Els textos i imatges publicats en aquesta obra estan subjectes –llevat que s'indiqui el contrari– a una llicència de Reconeixement-Compartir igual (BY-SA) v.3.0 Espanya de Creative Commons. Podeu modificar l'obra, reproduir-la, distribuir-la o comunicar-la públicament sempre que en citeu l'autor i la font (FUOC. Fundació per a la Universitat Oberta de Catalunya), i sempre que l'obra derivada quedi subjecta a la mateixa llicència que el material original. La llicència completa es pot consultar a <http://creativecommons.org/licenses/by-sa/3.0/es/legalcode.ca>

Índex

1. Introducció	5
2. Instal·lació	6
2.1. Canviar l'idioma	9
2.2. Estructura de continguts de Drupal	9
2.3. Tipus de continguts i taxonomies	11
3. Apartats del panell d'administració	12
3.1. Contingut (<i>content</i>)	12
3.2. Estructura (<i>structure</i>)	13
3.3. Aparença (<i>appearance</i>)	15
3.4. Persones (<i>people</i>)	15
3.5. Mòduls (<i>modules</i>)	17
3.6. Configuració (<i>configuration</i>)	18
3.7. Informes (<i>reports</i>)	19
3.8. Ajuda (<i>help</i>)	20
3.9. Definir tipus de contingut	20
3.10. Definir taxonomies	22
4. Exemple pràctic. Implementació d'una publicació electrònica en Drupal	24
4.1. Brífling del projecte	24
4.2. Creació de l'estructura de continguts	24
5. Instal·lar mòduls	31
6. Taula de correspondències entre Drupal i WordPress	33
7. Integració amb xarxes socials	35
8. Creació de temes (<i>themes</i>) en Drupal	38
8.1. Modificar un tema que ja existeix	38
8.2. Utilitzar un tema <i>framework</i> com a base	39
8.3. Crear un tema des de zero	40
8.3.1. Creació d'un tema "minimalista"	40
8.3.2. Creant un tema més complet	44
8.3.3. Exercici. Creació d'un tema des de zero a partir d'una plantilla estàtica (HTML + CSS)	45

1. Introducció

El Drupal és un dels **gestors de continguts** més utilitzats a causa de la seva flexibilitat per a adaptar-se a diferents tipus de projectes (llocs corporatius, tendes en línia, publicacions electròniques, comunitats, etc.).

Curiositat

Hi ha milers de llocs web construïts sobre Drupal, alguns tan importants com el de la Casa Blanca.

El fet que el Drupal sigui un programa de codi obert i molt utilitzat ofereix un gran avantatge: **una comunitat molt gran de desenvolupadors** que ens pot solucionar molts dubtes i ens garanteix el manteniment del projecte. En la pàgina oficial de Drupal tenim accés a aquesta comunitat, a més dels arxius, mòduls i temes que podem descarregar fàcilment.

The screenshot shows the Drupal homepage with a blue header. The main content area includes a search bar, a 'Log in / Register' button, and three columns of information: 'Why Choose Drupal?' (highlighting its flexibility and community), 'Sites Made with Drupal' (featuring Symantec Connect), and 'Develop with Drupal' (listing 15,523 modules, 1,475 themes, and 16,616 developers). A 'Visit Planet Drupal' link is also present.

Observació

Per descomptat, la pàgina de Drupal també està feta en Drupal. Gairebé sempre les pàgines oficials de qualsevol CMS estan construïdes amb el propi CMS: han de donar exemple.

Lloc oficial drupal.org.

El codi font de Drupal es pot descarregar en la pàgina <http://www.drupal.org>. Els requeriments són semblats a altres CMS –bàsicament PHP i MySQL–, encara que per a saber les versions mínimes de cadascun és necessari consultar en la pàgina de Drupal les especificacions mínimes en la pàgina de Drupal.

A Drupal podem ampliar les funcionalitats de l'aplicació afegint **mòduls** (que corresponen al que serien els *plug-ins* en WordPress). Qualsevol persona de la comunitat pot desenvolupar nous mòduls per a cobrir les seves necessitats, i el normal és que després els deixi a la disposició de tota la comunitat perquè si els necessita els faci servir. La versió base de Drupal (que es denomina el *core*) ve amb alguns mòduls bàsics (molts desactivats), però per a ampliar funcionalitats, segons les necessitats, moltes vegades és necessari descarregar altres mòduls addicionals.

Vegeu també

Conceptualment, entre WordPress i Drupal hi ha molts punts en comú. En l'apartat "Taula de correspondències entre Drupal i WordPress" se sistematitzen aquestes coincidències.

2. Instal·lació

Per instal·lar el Drupal necessitem el següent:

- Els arxius del *core* descarregats de drupal.org.
- Un servidor amb PHP+MySQL (local o remot).
- Una base de dades MySQL i les dades d'accés (usuari i contrasenya). Si es tracta d'un servidor remot, accés per FTP al servidor (i l'usuari i contrasenya corresponents).
- Si el volem instal·lar en un idioma diferent de l'anglès, els arxius de traducció; per exemple, en castellà a <http://localize.drupal.org/translate/languages/es>, o en català a <http://localize.drupal.org/translate/languages/ca>.

Observació

El Drupal també permet la instal·lació amb SQLite, un sistema de base de dades que funciona com a extensió del PHP. Per tant, en aquest cas no necessitaríem servidor MySQL.

Passos de la instal·lació:

- Descomprimir els arxius descarregats del web de Drupal. Podem canviar el nom de la carpeta pel que vulguem: si inicialment es diu *drupal7.14* el podem canviar per *revista*, per exemple.
- Copiar els arxius en el servidor (local o remot) en el directori adequat ("www" o "htdocs", depenent del servidor), de manera que puguem accedir al nostre lloc, per exemple, a <http://localhost/revista> o <http://www.miservidoremoto.com/revista> (suposant que hem donat el nom *revista* al directori).
- Copiar i enganxar l'arxiu "site/default/default.settings.php" com a "settings.php".
- Crear una base de dades buida des del nostre gestor de bases de dades (per exemple "phpMyAdmin", probablement en l'adreça <http://localhost/phpmyadmin> o on sigui apropiat en el nostre servidor local o remot). Per al nostre exemple, imaginem que també hem anomenat la base de dades *revista*.
- Entrar amb el navegador a l'adreça <http://localhost/revista>.
- Seleccionar l'opció "Install Drupal in English"¹
- Apareixerà una pantalla en què se'ns demanarà el nom de la base de dades i l'usuari i contrasenya per a accedir-hi. Posem les dades i continuem.

⁽¹⁾Instruccions per a instal·lar Drupal en qualsevol idioma inicialment: <http://drupal.org/localize>.

- Si hem escrit les dades correctes, ara ens demanaran les dades de la configuració del lloc i les dades del compte de l'administrador de Drupal (si estem instal·lant en "localhost", per a provar podem posar "admin/admin", encara que això, per descomptat, no és gens segur en un lloc real). És important apuntar aquest usuari i la contrasenya, ja que els haurem d'usar per a tornar a entrar a Drupal posteriorment com a administradors.
- Continuem i si tot va bé tindrem el lloc funcionant.

Select an installation profile



► Choose profile

Choose language

Verify requirements

Set up database

Install profile

Configure site

Finished

Standard

Install with commonly used features pre-configured.

Minimal

Start with only a few modules enabled.

Save and continue

Installing Drupal



Completed 6 of 28. 21%
Installed *Text* module.

✓ Choose profile

✓ Choose language

✓ Verify requirements


✓ Set up database

► Install profile

Configure site

Finished

Configure site



- ✓ Choose profile
- ✓ Choose language
- ✓ Verify requirements
- ✓ Set up database
- ✓ Install profile
- ▶ **Configure site**

Finished

SITE INFORMATION

Site name *

Site e-mail address *

Automated e-mails, such as registration information, will be sent from this address. Use an address ending in your site's domain to help prevent these e-mails from being flagged as spam.

SITE MAINTENANCE ACCOUNT

Username *

Spaces are allowed; punctuation is not allowed except for periods, hyphens, and underscores.

E-mail address *

Password *

Password strength: **Weak**

Confirm password *

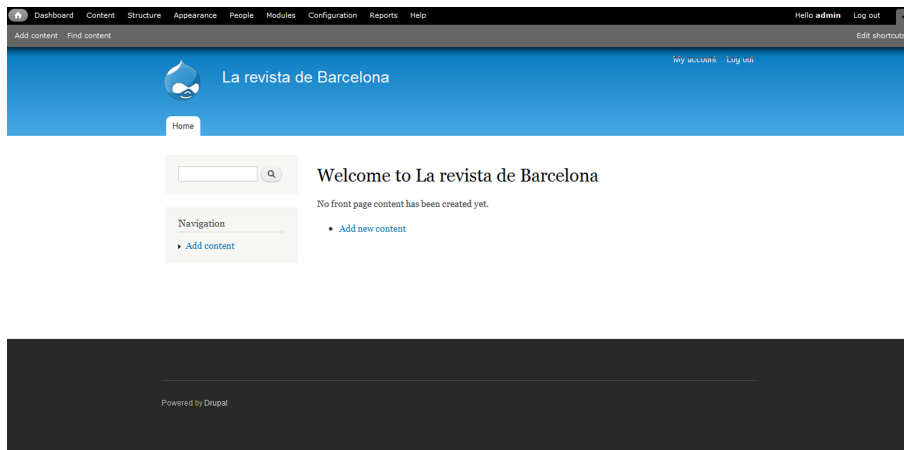
Passwords match: yes

To make your password stronger:

- Make it at least 6 characters
- Add uppercase letters
- Add numbers
- Add punctuation
- Make it different from your username

SERVER SETTINGS

Default country



The screenshot shows the Drupal front page for 'La revista de Barcelona'. The page has a blue header with the site name and a search bar. Below the header, there is a navigation menu and a main content area with a 'Welcome to La revista de Barcelona' message and a link to 'Add new content'. The footer shows 'Powered by Drupal'.

Observació

Si hem fet la instal·lació en un servidor local, és possible que ens aparegui un missatge (*warning*) en aquest punt avisant que no s'ha pogut enviar un correu electrònic. No ens hem de preocupar: això és degut simplement al fet que molts servidors locals no disposen de servidor de correu, però això no afecta el funcionament de la resta de funcionalitats.

Una vegada instal·lat tot correctament, ja hauríem de poder entrar en el lloc tant en el rol de visitant anònim com en el d'administrador. L'adreça del nostre lloc serà `http://localhost/revista/` (o el nom que li hàgim posat nosaltres).

2.1. Canviar l'idioma

Si *a posteriori* volem canviar l'idioma de la interfície de Drupal per un altre, sempre podem instal·lar el mòdul “Localization Update” i seguir els passos següents:

- Activar el mòdul *core locale* (aquest mòdul s'encarrega d'activar la funcionalitat multiidioma).
- Anar a la secció de configuració de l'idioma (a *admin/config/regional/language*) i seleccionar els idiomes que vulguem.
- Baixar i instal·lar el mòdul “localization update”.
- A partir d'aquest moment, els textos s'actualitzaran a tots els idiomes que tinguem actius.

Pàgina web

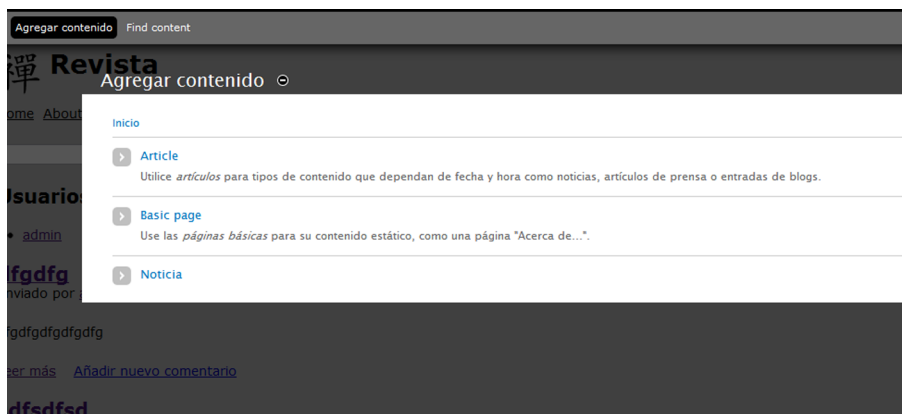
Aquests passos es troben descrits en la pàgina oficial de Drupal: <http://drupal.org/node/21145>.

2.2. Estructura de continguts de Drupal

En Drupal cada unitat de contingut és un **node**.

Aquests nodes poden ser de diferents tipus de contingut (tipus que podem definir en l'administrador). En instal·lar el Drupal en tenim dos de definits: “**Basic page**” (“Pàgina”) i “**Article**” (similars als tipus “Entry” i “Page” de WordPress).

Podem crear un contingut nou en l'opció Agregar contingut (*Create content*). Llavors haurem de triar quin dels dos tipus de contingut que hi ha volem crear. Provem a crear continguts de qualsevol.



En seleccionar “Agregar contingut”, tenim l'opció d'escollir entre els tipus que tinguem definits.

Observació

Hi ha poques diferències entre els dos tipus de continguts predefinits, “Article” i “Basic page”.

Qualsevol tipus de contingut té dos camps per defecte: “Title” (“Títol”) i “Body” (“Text”). El títol és obligatori, ja que és la manera de poder-lo identificar; el camp “Body” és opcional.

Parem esment sobretot a l'opció que apareix a baix en aquesta pantalla, “Options de publicació” (“Publishing options”). Si la despleguem veurem que hi ha tres opcions:

- Publicat (*published*). Si desmarquem aquesta opció el node es desarà però no serà visible per als visitants del nostre lloc.
- Promogut en la portada (*promoted to front page*). Si marquem aquesta opció el node apareixerà en la portada del lloc.
- Fix al començament de les llistes (*sticky at top of lists*). En cas que aquest contingut aparegui en una llista, sempre serà el primer de la llista.

Seleccionem “Promoted to front page” i publiquem un contingut de prova. Si anem a la portada del nostre web (per exemple, a localhost/revista), veurem que el node apareix en la portada. Si fem clic en el títol, veurem que ens porta a una pàgina única amb l'adreça “**node/1**”.

En Drupal tenim bàsicament **dos tipus de pàgines** (o **plantilles**):

- La portada o “Home page”, en què es mostren tots els continguts (nodes) promoguts en la portada.
- Les pàgines úniques per a cadascun dels continguts (nodes), per defecte amb la ruta “**node/xx**”, en què “xx” és l'identificador del contingut.

Si tinguéssim molts continguts promoguts en la portada, apareixeria una llista amb tots, paginada si fos necessari (amb les opcions anterior/següent), i fent clic en qualsevol dels títols arribaríem a la pàgina del node corresponent (per exemple, “node/1”, “node/2”, etc.).

Quan un node apareix en una llista (per exemple en la portada, però també en la llista d'una categoria o etiqueta), només se'n mostra un resum: és el que es denomina *mode resumit* (*teaser*) del node.

Observació

Les rutes són per defecte del tipus “node/1” si s'han activat les rutes netes (*clean urls*). En cas contrari, totes les rutes són del tipus `index.php?q=node/1`. Per a poder activar les rutes netes, s'ha de tenir activat el mòdul “**mod_rewrite**” d'Apache o del servidor web que s'estigui utilitzant.

A més del concepte de node, en la versió 7 el Drupal introdueix el concepte d'entitats (*entities*). Una entitat és una unitat de contingut més general: pot ser un node d'informació, però també usuaris, comentaris o taxonomies. El fet de tractar tots aquests elements d'una manera comuna permet una gran flexibilitat al sistema.

Pàgina web

Hi ha més informació sobre les entitats en la pàgina oficial de Drupal: <http://drupal.org/node/1261744>.

2.3. Tipus de continguts i taxonomies

Abans de començar a introduir els nostres continguts, és necessari tenir clar quina serà l'estructura del nostre projecte. En el CMS definirem aquesta estructura. Encara que en el futur la podem canviar, és molt més difícil fer-ho després que des del principi, per la qual cosa és especialment important intentar anticipar tot el que puguem.

Respecte a l'estructura de continguts, tenim **dos aspectes principals** a definir:

- Els **tipus de continguts**. Ens hem de plantejar si hi ha tipus de continguts diferenciats. Per exemple, necessitem notícies i articles d'opinió? Hem de valorar la possibilitat de definir tipus diferents en el moment en què es tracti de continguts que cal mostrar d'una manera totalment diferent, o que tinguin camps diferents. Per a diferenciar les notícies de diferents categories, en lloc d'utilitzar tipus de continguts, usarem categories, ja que els camps i la manera de visualitzar-les totes són iguals. En general, definirem pocs tipus de contingut i usarem categories i etiquetes per a organitzar el contingut de cada tipus.
- Les **taxonomies** (vocabularis de paraules clau que podem assignar als continguts). Les taxonomies són molt útils, ja que permeten crear categories fàcilment, a més d'etiquetar els continguts lliurement. Poden funcionar de dues maneres: com a categories o com a etiquetes. Una categoria és una llista "tancada" de possibilitats, mentre que una etiqueta és lliure i se'n poden afegir de noves en qualsevol moment. A efectes interns, pràcticament no hi ha cap diferència i el Drupal les tracta igual: la diferència està més aviat a nivell d'usuari. Les categories usen una llista desplegable i les etiquetes un camp de text amb "autocompletat". Podem separar les etiquetes amb comes.

Per a associar les taxonomies als tipus de contingut ho hem de fer com a camps. Cada tipus pot tenir una o diverses taxonomies associades. Més endavant veurem com es concreta això en el panell d'administració de Drupal.

3. Apartats del panell d'administració

Per entrar en el panell d'administració seleccionem “Administer” en el menú de l'esquerra o anem directament a /revista/admin (o el nom que hàgim decidit donar al nostre primer projecte en Drupal).



Menú d'administració de Drupal amb les seves opcions principals.

Veurem que hi ha nou apartats principals:

- **Panell de control** (*dashboard*). Ens dóna una visió global i el podem personalitzar.
- **Contingut** (*content*). Tot el relacionat amb els continguts.
- **Estructura** (*structure*). Tot el relacionat amb l'estructura del lloc (blocs, tipus de contingut, menús i taxonomia).
- **Aparença** (*appearance*). Aparença (bàsicament, els *themes*).
- **Persones** (*people*). Gestió dels usuaris del lloc.
- **Mòduls** (*modules*). Llista de mòduls disponibles.
- **Configuració** (*configuration*). Configuració del lloc.
- **Informes** (*reports*).
- **Ajuda** (*help*).

3.1. Contingut (*content*)

Aquí tenim tot el que està relacionat amb els continguts.

Opció “Contingut”, amb els dos apartats principals, “Contingut” i “Comentaris”.

Tenim dues pestanyes principals:

- **Contingut (*content*)**. Gestió dels continguts (nodes). Conté una llista amb tots els continguts que es poden filtrar per tipus o per estatus (publicat, no publicat, etc.). Per descomptat, també ens permet afegir nous continguts.
- **Comentaris (*comments*)**. Gestió dels comentaris que han escrit els visitants del lloc. En aquest apartat pot ser que ens trobem comentaris en estat de moderació, que l'administrador ha d'aprovar perquè siguin visibles (de manera similar a WordPress).

3.2. Estructura (*structure*)

Subopcions del menú “Estructura” (l’opció disparadors s’activa amb el mòdul corresponent).

a) Blocs (*blocks*)

Aquest apartat serveix per a assignar blocs a regions de la pantalla. Cada plantilla té unes regions definides. Podem assignar els blocs a les regions que vulguem i ordenar-les (simplement movent-les usant la creu). Els blocs són preexistents però es poden afegir nous blocs en l'opció "Add block" de la pestanya superior. És totalment equivalent a l'opció "Widget" de WordPress.

Exercici

Practiqueu en aquest apartat i associeu alguns d'aquests blocs a les zones disponibles en el tema actual.

b) Menús

Aquí podem configurar i crear menús (també són blocs, de manera que s'han d'assignar en l'apartat "Blocs"). Hi ha quatre menús predefinits i "Navigation" és el menú de l'administrador. Cada menú conté ítems que són realment les opcions del menú. Aquestes opcions es poden reordenar amb arrossegar i deixar anar (*drag-and-drop*). Cada opció del menú és bàsicament un text i un enllaç que pot ser intern (del tipus "node/1") o extern (per exemple, <http://www.uoc.edu>).

c) Taxonomies (*taxonomy*)

Les taxonomies són categories i etiquetes que es poden assignar als tipus de continguts.

d) Tipus de contingut (*content types*)

Mostra els tipus de contingut que tenim definits. Podem afegir fàcilment nous tipus si els necessitem. Tenim dues opcions principals per a cada tipus de contingut:

- Gestionar camps (*manage fields*), per a modificar els camps associats.
- Gestionar la presentació (*manage display*) per a modificar la manera en què es veuen aquests camps depenent del "mode de presentació".

Els nodes tenen bàsicament dos modes de presentació: el mode "Resum" ("Teaser") i el mode "Predeterminat". En les llistes, per exemple la de la portada, els nodes apareixen per defecte en mode "Resum", de manera que si el text és molt llarg es retallarà automàticament per a mostrar només un fragment.

L'opció "Gestionar presentació" ens permet una gran flexibilitat, ja que podem decidir si es visualitzen (i com es visualitzen) els camps de cada tipus de continguts segons aquestes dues situacions.

Vegeu també

Veurem aquest aspecte amb més profunditat en l'apartat "Definir taxonomies".

Exercici

Publiqueu una pàgina nova de contingut ("Create content") i afegiu un text molt llarg, provinent, per exemple, de la pàgina de generació de *Lorem ipsum* (<http://www.lipsum.com/feed/html>).

Assegureu-vos que el node es publica en la portada. Una vegada publicat, aneu a la portada (clicant en el logo de Drupal o en el tema que tinguem actiu, per exemple). Vegeu com apareix el text retallat automàticament (node en format "Teaser") i a més apareix un enllaç "Read More".

3.3. Aparença (*appearance*)

Aquest és l'apartat que fa referència a l'aspecte del lloc. Com a bon gestor de continguts, el contingut i la presentació haurien d'estar sempre el més separats possible. Això s'aconsegueix gràcies als **temes** (*themes*). Així, en aquesta secció podem activar qualsevol tema i canviar opcions sobre la presentació.

The screenshot shows the 'Aparença' (Appearance) administration page in Drupal. The page title is 'Aparença' and it displays a list of active themes. The current theme is 'Bartik 7.14 (tema predeterminado)'. Below it, the 'Seven 7.14' theme is listed as an alternative. The page includes navigation tabs for 'LISTA', 'ACTUALIZAR', and 'OPCIONES'.

Opció "Aparença" amb la llista de temes disponible.

Exercici

Descarregueu i activeu el tema següent: <http://drupal.org/project/journalcrunch>.

Comprovar el resultat en la portada i la resta de seccions.

3.4. Persones (*people*)

El Drupal és bastant potent en la gestió d'usuaris. En l'apartat "Persones" podem veure la llista de tots els usuaris, crear rols i permisos per a cadascun, o definir opcions de configuració relatives als usuaris com, per exemple, la manera de registrar-se en el lloc.

<input type="checkbox"/>	NOMBRE DE USUARIO	ESTADO	ROLES	MIEMBRO DURANTE	ÚLTIMO ACCESO	OPERACIONES
<input type="checkbox"/>	admin	activo	• administrator	9 meses 3 semanas	hace 2 mins 33 segs	editar

Llista d'usuaris del lloc web en l'apartat "Persones".

Disposem de 2 pestanyes principals: Llista i Permisos.

- A **Llista** podem veure la llista de tots els usuaris registrats en el lloc, que podem filtrar per "rol", "permís" o "estat". A més, podem editar cadascun d'ells fent servir l'opció "Edit".
- Al mateix temps, a la pestanya **Permisos** disposem de 2 subpestanyes: "Permisos" i "Rols".
 - A "Permisos" podem activar o desactivar els permisos per a cadascun dels rols definits a la nostra aplicació.
 - A "Rols" veiem una llista dels rols disponibles. Per defecte n'hi ha solament 3 rols: usuari anònim, usuari registrat i administrador. Podem afegir més rols depenent de les nostres necessitats (per exemple, editor, corrector, etc.), als quals haurem d'activar els permisos corresponents.

Los permisos le dan el control sobre lo que los usuarios pueden hacer y ver en su sitio. Usted puede definir un grupo específico de permisos para cada rol. (Vea la página de [Roles](#) para crear un rol). Dos roles importantes a considerar son usuarios registrados y administradores. Cualquier permiso otorgado al rol de usuarios registrados se dará a cualquier usuario que pueda iniciar sesión en su sitio. Puede convertir a cualquier rol en el rol administrador del sitio, lo que significa que todos los nuevos permisos se le asignarán automáticamente. Puede hacer esto en la página de [Opciones de Usuario](#). Debe tener cuidado y asegurarse que sólo los usuarios de confianza obtengan este acceso y este nivel de control del sitio.

[Ocultar las descripciones](#)

PERMISO	USUARIO ANÓNIMO	USUARIO REGISTRADO	ADMINISTRATOR
Block			
Administrar bloques	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Comment			
Administrar comentarios y opciones de comentarios	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Ver comentarios	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Publicar comentarios	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Omitir aprobación de comentario	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Editar comentarios propios	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Contextual links			
Usar enlaces contextuales	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Usar enlaces contextuales para ejecutar acciones relacionadas con los elementos de una página.			

Llista de permisos per a cadascun dels rols.

3.5. Mòduls (*modules*)

En aquest apartat es mostren tots els mòduls (correspon al que serien els *plugins* en WordPress). Podem activar els mòduls que vulguem. En la llista podem veure que molts dels mòduls no estan activats per defecte.

Descargár [módulos contribuidos](#) para ampliar el funcionamiento de Drupal.

Revise e instale regularmente [las actualizaciones disponibles](#) para mantener un sitio seguro y actualizado. Ejecute el [script de actualización](#) siempre que un módulo es actualizado.

[+ Instalar nuevo módulo](#)

NÚCLEO

ACTIVADO	NOMBRE	VERSIÓN	DESCRIPCIÓN	OPERACIONES
<input type="checkbox"/>	Aggregator	7.14	Integra contenido sindicado (canales de noticias RSS, RDF y Atom).	
<input checked="" type="checkbox"/>	Block	7.14	Controla el montaje visual de los bloques con los que se construye una página. Los bloques son cajas de contenido que se representan en una zona o región de una página web. Necesitado por: Dashboard (activado)	Ayuda Permisos Configurar
<input type="checkbox"/>	Blog	7.14	Activa los blogs multi usuario.	
<input type="checkbox"/>	Book	7.14	Permite a los usuarios crear y organizar los contenidos relacionados en un índice.	
<input checked="" type="checkbox"/>	Color	7.14	Permite a los administradores cambiar el esquema de color de los temas compatibles con él..	Ayuda
<input checked="" type="checkbox"/>	Comment	7.14	Permite a los usuarios comentar y discutir contenido publicado. Necesita: Text (activado), Field (activado), Field SQL storage (activado) Necesitado por: Forum (desactivado), Tracker (desactivado)	Ayuda Permisos Configurar
<input type="checkbox"/>	Contact	7.14	Habilita el uso de formularios de contacto personal y a nivel sitio.	
<input type="checkbox"/>	Content translation	7.14	Permite que el contenido sea traducido a diferentes idiomas. Necesita: Locale (activado)	

Opció "Mòduls", amb la llista de mòduls disponibles i els actius.

3.6. Configuració (*configuration*)

Aquí apareixen tot tipus d'opcions relacionades amb la configuració del sistema. En general, són opcions que només modificarem durant el desenvolupament del projecte i que en principi no seria necessari editar una vegada ja està publicat el lloc. En qualsevol cas, aquestes són opcions a les quals només hauria de tenir accés l'administrador del lloc.



Opcions de configuració

Totes les opcions estan organitzades per categories.

Cal destacar que, en instal·lar qualsevol mòdul, és possible que en aquest apartat apareguin altres opcions que hi estiguin relacionades.

Per defecte, podem veure les categories següents:

- **Persones.** Opcions relacionades amb els usuaris, per exemple la manera de registrar-se.
- **Autoria del contingut.** Formats de text. El Drupal disposa de tres formats de text diferents:
 - *Plain text* (o text net). Només text.
 - *Filtered HTML*. L'opció per defecte, permet afegir algunes etiquetes bàsiques d'HTML, com a capçaleres (h1, h2), *strong*, etc. Podem modificar aquestes etiquetes permeses com vulguem.
 - *Full HTML*. Quan activem aquesta opció en editar el text s'admet qualsevol etiqueta d'HTML.

- **Cerca i metadades.** Opcions relatives a la cerca i a com es mostren els URL dels continguts.
- **Mitjans.** Aquí podem definir les mides predefinides per a les imatges. El Drupal construeix per defecte tres mides diferents per a cada imatge que pugem al sistema. Podem modificar la manera de crear aquestes imatges o fins i tot crear nous tipus.
- **Regional i idioma.** Les opcions relatives a l'idioma, les traduccions i la manera de mostrar les dates. Aquest apartat és molt important en qualsevol lloc multiidioma.
- **Sistema.** La informació bàsica del nostre lloc (nom, lema, etc.). Nombre de continguts publicats en la portada (per defecte són deu).
- **Interfície d'usuari.** En aquest apartat podem definir dreceres: accessos ràpids en el panell d'administració.
- **Desenvolupament.** Opcions per a desenvolupadors i per a millorar el rendiment del lloc. Una vegada publicat el projecte, hauríem d'activar la memòria cau (*cache*) de pàgines i blocs, i unificar en un sol arxiu tots els documents CSS i JavaScript del nostre tema.
- **Serveis web.** Opcions relatives a la publicació automàtica d'RSS.

The screenshot displays three configuration panels in the Drupal administration interface:

- VACIAR CACHÉ:** A button labeled "Vaciar todas las cachés".
- CACHÉ:** Contains several settings:
 - Caché de páginas para usuarios anónimos (indicated by a red arrow).
 - Caché de bloques (indicated by a red arrow).
 - Mínimo de permanencia en caché:** A dropdown menu set to "<ninguno>". Below it, text reads: "Las páginas en cache no serán recreadas hasta que no haya transcurrido al menos este periodo."
 - Caducidad de las páginas en caché:** A dropdown menu set to "<ninguno>". Below it, text reads: "El tiempo máximo que una caché externa puede utilizar una versión antigua de una página."
- OPTIMIZACIÓN DE ANCHO DE BANDA:** Contains two checked options:
 - Reunir y comprimir archivos CSS. (indicated by a red arrow).
 - Juntar los archivos JavaScript. (indicated by a red arrow).

At the bottom of the configuration area is a button labeled "Guardar configuración".

Opcions de l'apartat "Desenvolupament /rendiment".

3.7. Informes (reports)

Informes interns relacionats amb el lloc com:

- Rebre un informe de l'estat funcional del lloc i de qualsevol problema detectat.
- Actualitzacions disponibles per als mòduls i temes instal·lats.
- Missatges recents del registre.
- Esdeveniments registrats recentment.
- Frases principals en les cerques.
- Informació general dels camps en tots els tipus de contingut.
- Errors principals d'"accés denegat".
- Errors principals de "la pàgina no s'ha trobat".

3.8. Ajuda (*help*)

Aquest apartat no és més que una llista de documents d'ajuda destinat als usuaris de l'eina d'administració del Drupal.



Apartat "Ajuda"

3.9. Definir tipus de contingut

Una vegada vistes les opcions principals del panell d'administració, veurem com es configura l'estructura de continguts d'un projecte bàsic.

El Drupal disposa per defecte de dos tipus de contingut, "Pàgina" i "Article". No obstant això, en molts projectes tindrem la necessitat de definir nous tipus.

Per exemple, si volguéssim crear una tenda, probablement creariem el tipus de contingut “Producte”, que tindria una sèrie de camps especials (preu, referència, etc.), o, en cas de voler crear un *portfolio*, podríem crear un tipus de contingut “Projecte” que tingués com a camps especials diverses imatges i una categoria per a cada projecte, etc.

Definir tipus de contingut en Drupal és molt fàcil. Només cal anar a l'apartat “Estructura/tipus de contingut” (“Structure/types”) i activar l'opció “Afegir tipus de contingut” (“Add content type”).

Així, per a afegir un tipus nou que anomeni “Notícia”, en el camp “Type” haurem d'escriure un codi intern per a aquest tipus (que podria ser també “Notícia”) i afegir una descripció d'ajuda que apareixerà quan algú creï un contingut d'aquest tipus.

Cada tipus de contingut té unes opcions predeterminades. Aquestes opcions les podem veure en la part inferior d'aquesta pantalla. Per exemple, podem decidir si es publica per defecte o no en la portada (“Promoted to front page”) i podem activar o desactivar els comentaris sobre aquest tipus de contingut (“Comment settings”). Els dos tipus de contingut predefinitos en Drupal tenen un comportament per defecte diferent. Els articles es publiquen en la portada i admeten comentaris, mentre que les “Pàgines” fan el contrari.

Una vegada creat un nou tipus de contingut aquest apareixerà en la llista de tipus de continguts i podrem crear continguts d'aquest tipus en l'opció “Create content”.

Los tipos de contenido individuales pueden tener diferentes campos, comportamientos y permisos asignados.

Nombre *

Producto Nombre de sistema: producto [Editar]

El nombre legible para humanos de este tipo de contenido. Este texto será mostrado como parte del listado en la página *Agregar nuevo contenido*. Es recomendable que este nombre comience con una letra mayúscula y contenga solamente letras, números y espacios. Este nombre debe ser único.

Descripción

Un producto de mi tienda

Describir este tipo de contenido. Este texto es lo que se muestra en la página *Añadir nuevo contenido*.

Opciones del formulario de envío Nombre	Opciones predefinidas <input type="checkbox"/> Publicado <input checked="" type="checkbox"/> Colocado en la página principal <input type="checkbox"/> Fijo al comienzo de las listas <input type="checkbox"/> Crear nueva revisión Usuarios con el permiso de <i>Administrar contenido</i> podrán sobrescribir estas opciones.
Opciones de publicación No publicado, Colocado en la página principal	Funcionamiento multi idioma <input checked="" type="radio"/> Desactivado <input type="radio"/> Activado Active el funcionamiento multi idioma para este tipo de contenido. Si está habilitado, se agregará al formulario de edición un campo selector de idioma, lo que le permitirá elegir entre uno de los idiomas habilitados . Si está inhabilitado, los mensajes nuevos se salvan en el idioma predeterminado. El contenido existente no se verá afectado por cambios en esta opción.
Opciones de presentación No mostrar información del envío	
Opciones de comentarios Abrir, Hilos de comentarios, 50 comentarios por página	
Opciones del menú	

Creació d'un nou tipus de contingut “producte” a “Estructura/tipus de contingut”.

Exercici

Creeu un tipus de contingut nou en l'administrador i publiqueu un node d'exemple d'aquest tipus de contingut.

3.10. Definir taxonomies

Les **taxonomies** són una eina bàsica per a poder estructurar els continguts ja que, entre altres coses, permeten:

- Definir categories en el contingut i, per tant, definir l'estructura relacionada.
- Etiquetar els continguts com vulguem.
- Fer una llista (automàticament) dels continguts a partir d'aquestes categories o etiquetes.

A continuació imaginem un cas concret: per al tipus de contingut “Notícia” anterior volem la possibilitat de diferenciar dos tipus de notícia: “Esports” i “Política”. Per a això haurem d'anar a l'apartat “Estructura/Taxonomia”, activar l'opció “Agregar vocabulari” i donar un nom al nostre vocabulari (per exemple, *Tipus de notícia*) i una descripció si la creiem necessària. En aquest moment ja tindrem un vocabulari, però buit. En qualsevol moment podem agregar termes a aquest vocabulari (en l'opció “Agregar termes”).

Afegirem els termes *política* i *esport* al nostre vocabulari.

Nombre *
 Nombre de sistema: tipo_de_noticia [\[Editar\]](#)

Descripción

Creació d'un nou vocabulari “Tipus de notícia” a “Estructura/Taxonomia”

Nombre *

Descripción

Formato de texto [Más información sobre los formatos de texto](#)

- Las direcciones de las páginas web y las de correo se convierten en enlaces automáticamente.
- Etiquetas HTML permitidas: <a> <cite> <blockquote> <code> <dl> <dt> <dd>

- Saltos automáticos de líneas y de párrafos.

Alias de URL

 Especifique de forma opcional una URL alternativa a mediante la que acceder a este término. Use una ruta relativa y no añada la barra final o el alias de URL no funcionará.

RELACIONES

Términos padres

Peso *

 Los términos se muestran en orden ascendente por su peso.

Una vegada creat el vocabulari podem agregar termes.

Finalment, perquè aquest vocabulari estigui vinculat amb els continguts l'hem d'afegir com a camp al tipus de contingut. Així, si el volem vincular al tipus de contingut “Notícies”. haurem de seguir els passos següents:

- Anar a “Estructura/Tipus de contingut/Gestionar camps”.
- Agregar un nou camp amb l'etiqueta “Tipus de notícia” i nom intern “field_tipus_notícia”.
- Com a tipus de camp hem de seleccionar “Referència de terme” per a vincular el camp amb un vocabulari de la taxonomia. A “Control” podem triar com funcionarà: en forma de llista desplegable, autocompletat (etiquetatge lliure) o com a caselles de selecció.
- Desar.
- En el pas següent, seleccionarem el vocabulari que volem vincular. En el nostre exemple, “Tipus de notícia”.
- Finalment podem canviar les opcions per defecte del camp, com l'etiqueta que es mostrarà a l'usuari, si és un camp obligatori o no, o el nombre de valors (es podrien vincular diversos valors simultàniament si fos necessari).

The screenshot shows the Drupal administration interface for managing content types. The 'Gestionar campos' modal is open for the 'Noticia' content type. The modal contains a table of existing fields and options to add a new field.

ETIQUETA	NOMBRE	CAMPO	CONTROL	OPERACIONES
+	Title	title	Elemento del módulo Nodo	
+	Body	body	Texto largo y resumen	Área de texto con un resumen editar eliminar
+	foto	field_foto	Imagen	Imagen editar eliminar
+	Lugar	field_lugar	Texto	Campo de texto editar eliminar

Below the table, there are two sections for adding new fields:

- Agregar nuevo campo:** Includes a text input for the label (currently 'Tipo de noticia'), a text input for the field name (currently 'field_tipus_noticia'), a dropdown menu for the field type (currently 'Referencia de término'), and a dropdown menu for the control type (currently 'Lista de selección').
- Agregar campo existente:** Includes a dropdown menu to select an existing field (currently '- Selecciona un campo existente -') and a dropdown menu to select a control type (currently '- Selecciona un control -').

A 'Guardar' button is located at the bottom of the modal.

Vinculació del vocabulari “Tipus de notícia” al tipus de contingut “Notícia”

A partir d'aquest moment, en afegir nous continguts de tipus “Notícia”, tindrem el “Tipus de notícia” com a camp en forma de llista desplegable. De la mateixa manera podem afegir altres camps que puguem necessitar a aquest o a qualsevol altre tipus de contingut.

Exercici

Creeu el tipus de contingut “Notícia” i el vocabulari “Tipus de notícia”. Afegiu un contingut de prova almenys per a cadascuna de les categories.

4. Exemple pràctic. Implementació d'una publicació electrònica en Drupal

Per posar en pràctica els coneixements bàsics de Drupal imaginarem un exercici concret. Es tracta de la implementació d'una publicació electrònica amb determinats requeriments. Partirem d'un brífing (*briefing*) o descripció bàsica de l'objectiu de la publicació i a continuació veurem els passos necessaris per a construir-la.

4.1. Brífing del projecte

La revista electrònica serà una publicació d'actualitat molt simple amb els requisits que detallem a continuació:

- La secció principal mostrarà **notícies** que podran ser, per exemple, dels tipus següents: esports, política, economia, gent, temps i internacional.
- Volem crear un apartat d'opinió en què els editors puguin escriure articles com si es tractés d'un blog.
- També tindrem una secció que només mostri **fotos** d'actualitat.
- També necessitem crear una pàgina de **contacte** amb una presentació de la revista.
- Tindrem sempre visible un **menú** amb les seccions (categories) a més dels apartats de fotos, opinió i contacte.
- Totes les pàgines tindran l'opció de **compartir** en xarxes socials.

L'objectiu serà aconseguir tenir una versió bàsica d'aquesta publicació funcionant i, a més, crear un nou rol anomenat *Editor* amb dret a publicar continguts, però no a canviar opcions dins del panell d'administració.

4.2. Creació de l'estructura de continguts

A partir dels requisits, deduïm que necessitarem crear dos tipus de contingut, a part dels que ja hi ha per defecte (entrada i pàgina):

- Notícia
- Opinió

Observació

Assumim que ja tenim el Drupal instal·lat tal com s'ha explicat i que coneixem la dinàmica de funcionament de l'administrador de Drupal.

Quant al contingut, les notícies tindran un camp de foto i, a més, les categories que hem decidit. El contingut de l'apartat "Opinió" també tindrà un camp de foto, però de moment no necessita categories.

1) Creació d'un vocabulari "Categoria" per a les notícies

Començarem creant les categories. Per a això, anirem a l'apartat "Estructura/Taxonomia" i crearem un vocabulari nou anomenat "Categoria". Una vegada creat aquest vocabulari, afegirem la llista de termes amb esports, política, etc.

En Drupal podem crear tants vocabularis com vulguem i poden funcionar com a *tags* (etiquetes lliures) o categories (llista limitada de possibilitats).

2) Creació del tipus de contingut "Notícia"

A continuació crearem el tipus de contingut "Notícia", amb dos camps especials: un serà la categoria que hem creat anteriorment i l'altre un camp d'imatge que anomenarem *Foto*. Per a això, anem a l'apartat "Estructura/Tipus de contingut" i seleccionem "Afegir tipus de contingut". En aquest moment veurem tot un seguit de propietats que es poden ajustar a aquest tipus de contingut per defecte. Podem deixar les opcions tal com estan per defecte. Si és necessari, ja les canviarem després.

A continuació afegirem els dos camps que necessitem. En l'apartat "Gestionar camps" afegirem primer la categoria, donant el nom que vulguem a aquest camp –per exemple, *Categoria* – i seleccionant sobretot que es tracta d'un camp "Referència de terme".

En el pas següent indicarem que volem utilitzar el vocabulari "Categoria". Tot seguit, indicarem altres dades, com el text d'ajuda, si es tracta o no d'un camp obligatori, el nombre de valors que pot tenir, etc. En el nostre cas, també deixarem totes les opcions en els seus valors per defecte (en cas necessari ho podem canviar en un altre moment).

En aquest moment el nostre tipus de contingut "Notícia" té tres camps:

- El títol
- El cos
- La categoria

Finalment, afegirem un altre camp d'imatge que anomenarem *Foto*, també amb totes les opcions per defecte.

En aquest moment ja tenim creat el nostre tipus de contingut "Notícia".

Exercici

Creeu almenys dos continguts d'aquest tipus, afegint en cada cas una foto i una categoria. Vegeu el resultat en la portada del nostre lloc web.

Ens falta un pas important: decidir com es mostren els camps en les dues situacions principals en què es poden mostrar els continguts. Per a això cal anar a l'apartat “Gestionar presentació” de cada tipus de contingut.

Tenim dos modes de presentació per defecte:

- Predeterminat: correspon a la pàgina única del contingut.
- Resum (*teaser*): correspon al mode resumit que apareix en les llistes (per exemple, en la portada, o en la llista d'una categoria o *tag*).

Modificant aquestes opcions podem decidir, entre altres coses,

- L'ordre dels camps.
- Si apareix o no l'etiqueta de cada camp.
- En quina mida apareix la imatge.
- Etc.

3) Creació del tipus de contingut “Opinió”

Seguirem els mateixos passos que en el cas anterior per crear un tipus de contingut “Opinió”, però en aquest cas sense afegir-hi una categoria, només el camp Foto.

4) Creació del tipus de contingut “Foto”

També crearem un tipus de contingut “Foto”. En aquest tipus de contingut eliminarem el camp de text (cos) i afegirem un camp de foto d'imatge.

Arribats a aquest punt tindrem cinc tipus de contingut definits:

- “Article”
- “Basic Page”
- “Notícia”
- “Opinió”
- “Foto”

5) Creació de continguts de prova amb el mòdul Devel

Quan creem un lloc web moltes vegades ens interessa tenir continguts de prova. El mòdul Devel ens permet crear continguts ficticis per a provar com apareixen segons l'estructura que hem generat.

Instal·larem el mòdul <http://drupal.org/project/devel> i activarem “Devel generate”. Una vegada activat aquest mòdul, en l'apartat “Configuració/Desenvolupament de l'opció” ens apareixerà “Crear contingut”.

Exercici

Creeu cinquanta continguts de tipus notícia i comproveu el resultat.

6) Apartat “Contacte”

Crearem l'apartat “Contacte” com una pàgina normal de tipus de contingut “Basic page”, per la qual cosa no és necessari crear cap tipus de contingut nou per a aquest apartat. En crear la pàgina, ens assegurarem d'afegir-la al menú principal.

7) Instal·lació d'un editor de textos més potent per a l'administrador

En un projecte real, en aquest punt ens adonarem que probablement necessitem un editor de textos més potent (com el que té el WordPress per defecte). Per a solucionar aquest problema podem instal·lar mòduls com FCKEditor o TinyMCE, o encara millor el metamòdul <http://drupal.org/project/wysiwyg>. Es tracta d'un metamòdul en el sentit que és una base per a instal·lar diversos editors de textos i activar-los, però per a això haurem de descarregar addicionalment els arxius de l'editor que vulguem utilitzar (seguint les instruccions).

8) Creació d'entrades en el menú per a cadascuna de les categories

Per aconseguir les entrades per a cadascuna de les seccions (categories) que hem definit, anirem a l'apartat “Estructura/Menús” i editarem els enllaços de “Main menu” (menú principal). Els menús consisteixen en dos camps, el text que es mostra a l'usuari i l'enllaç que activa.

De fet, qualsevol terme de qualsevol vocabulari té un ID, i la ruta `taxonomy/id` ens portarà sempre a la llista de tots els continguts que inclouen aquest terme.

Per a saber l'enllaç que activa cada categoria, podem fer clic en aquesta categoria en alguna pàgina i copiar la ruta de l'enllaç del navegador, però hem de tenir en compte que les rutes sempre han de ser relatives. Per tant, en el menú sempre introduïrem la ruta interna.

Suposant que la nostra ruta absoluta sigui (inclou el codi de l'idioma actual) `http://localhost/revista/es/taxonomy/term/5`, la ruta relativa que hem d'introduir en el menú és: `taxonomy/term/5`.

En cas contrari no podríem traslladar el contingut del lloc a un altre servidor, ja que les rutes serien diferents.

És important tenir en compte que, en introduir els termes d'un vocabulari, podem assignar en el camp "Àlies d'URL" l'adreça que volem que aparegui en el navegador, per exemple "Esports" en lloc de "taxonomy/term/27". Això és molt millor tant per a l'usuari com per als cercadors d'Internet.

9) Definir rols específics per a l'editor

En un projecte real també necessitarem crear rols especials per al personal que s'encarregarà d'administrar el lloc web.

Per crear un rol que pugui, per exemple, crear i editar contingut sense accés a la resta d'opcions de Drupal, seguirem els passos següents:

- En l'apartat "Persones/Permisos/Rols" crearem un rol nou anomenat *Editor*.
- Posteriorment, anirem a l'apartat "Permisos" i activarem per a aquest rol només els permisos relatius a l'apartat "Node access" per crear, editar i esborrar continguts.
- Finalment, crearem un usuari en l'apartat "Persones/Llista" i li assignarem el rol "Editor".

Exercici

Proveu que efectivament funciona, creant el rol "Editor" amb permisos específics, un usuari amb aquest rol i, posteriorment, entrant en el lloc en un navegador diferent (perquè no mantingui la sessió d'administrador), utilitzant un usuari amb rol "Editor" i comprovant que podem crear i editar contingut però sense tenir accés a la resta de les opcions del panell d'administració.

10) Llista de fotos

En la nostra proposta havíem dit que hi hauria un apartat "Foto" que mostraria fotos de temes d'actualitat. En Drupal **no hi ha una manera immediata de mostrar una llista de continguts d'un tipus determinat**. Per a solucionar aquest problema podem utilitzar el mòdul "**Vistes**" ("Views"). Atenció: aquest mòdul depèn d'un altre, <http://drupal.org/project/ctools>, que caldrà que instal·lem prèviament (si no ho fem, veurem un avís sobre aquest tema en instal·lar "Views").

"Vistes" és un dels mòduls més útils de Drupal, ja que ens permet crear fàcilment pàgines o blocs personalitzats filtrant i ordenant continguts com vulguem.

Exercici

Instal·leu el mòdul “Vistes” i activeu-lo.

Per crear una pàgina especial amb la llista de fotos seguirem els passos següents una vegada instal·lat el mòdul “Vistes”:

- Anar a “Estructura/vistes”.
- Agregar una nova vista.
- Donar-hi el nom *Fotos* i filtrar per tipus de contingut “Foto”.
- Donar el nom que vulguem a l'adreça de la pàgina (per exemple *Fotos*).
- Aquesta serà l'adreça que haurem d'utilitzar si la volem afegir al menú principal.

Utilitzant el panell de la vista també podem crear fàcilment un bloc amb fotos, personalitzar els camps que volem que apareguin, etc.

11) Llista d'articles d'opinió

De la mateixa manera, també crearem l'apartat on es podran veure els articles d'opinió. Crearem una vista “Opinió” que faci la llista de totes les entrades d'aquest tipus i una entrada en el menú per a aquesta secció. També crearem un bloc amb els títols dels últims cinc articles d'opinió i els assignarem a la barra lateral.

Ja hem creat l'estructura de continguts que necessitem (tipus de contingut, taxonomies i vistes). El pas següent serà crear el tema que mostrarà aquests continguts a l'usuari. Exactament igual que amb WordPress, una manera àgil d'aconseguir un tema personalitzat és crear un tema derivat d'un altre i personalitzar el CSS.

12) Personalitzar un tema base

Si utilitzem un tema base, el millor (encara que no imprescindible) és utilitzar algun tema *framework*, com per exemple “Zen”.

Un tema *framework* és el que defineix el *layout* i els estils corresponents, però que no descriu la part de presentació.

Pàgina web

Podem trobar el tema “Zen” en el web oficial: <http://drupal.org/project/zen>.

Els passos per a crear un tema derivat de “Zen” es poden trobar en el “readme.txt” del tema. Bàsicament, consisteixen a copiar la subcarpeta “Starterkit” com a carpeta del nostre tema i canviar totes les referències a “starterkit”.

13) Alternativa: instal·lar un tema *framework responsive*

Una altra possibilitat seria utilitzar com a base un tema *responsive* que s'adapti a pantalles de diferents mides.

Alguns dels millors que podem trobar per a Drupal són Omega (<http://drupal.org/project/omega>) i Adaptive Theme (<http://drupal.org/project/adaptivetheme>).

Aquests temes, a part d'adaptar-se a les diferents resolucions i dispositius, també optimitzen les imatges per a diferents resolucions i tenen moltes opcions per als desenvolupadors.

Vídeo

Screencast sobre com es pot personalitzar “Zen”:

<http://drupalonlinetraining.com/youtube/linwclark/drupal-7-basic-theming-zen-and-css>

Pàgina web

Més informació sobre com es poden crear subtemes en la pàgina de Drupal a “Creating a sub-theme”: <http://drupal.org/node/225125>.

5. Instal·lar mòduls

En instal·lar el Drupal, de partida disposem d'una sèrie de mòduls que podem activar en cas de necessitat. Aquests mòduls que formen part de la instal·lació inicial són els denominats *del nucli* o *core*.

A més dels mòduls del nucli, podem instal·lar centenars de mòduls addicionals creats per la comunitat Drupal. A la pàgina oficial en podem veure més de 10.000, organitzats per temes.

Recomanacions sobre l'ús de mòduls:

- Assegurar-se que el mòdul és compatible amb la versió de Drupal que usem.
- Assegurar-se que és estable. Això es pot comprovar en la llista de mòduls de drupal.org perquè apareix de color verd; si està en desenvolupament (vermell) probablement funcioni però no tindrem fiabilitat total, per la qual cosa no és recomanable en un projecte comercial.
- Llegir a fons la documentació del mòdul abans d'instal·lar-lo per a assegurar-nos que compleix amb les especificacions que volem.
- En qualsevol cas, sempre l'hauríem d'instal·lar en un servidor de proves abans de fer-ho en el servidor de producció. D'aquesta manera, podrem detectar problemes abans de deixar-lo per definitiu. Una vegada comprovat, ja el podrem instal·lar en el servidor final

Downloads

Recommended releases

Version	Downloads	Date	Links
7.x-3.3	tar.gz (1.51 MB) zip (1.72 MB)	2012-Feb-22	Notes
6.x-2.16	tar.gz (1.21 MB) zip (1.35 MB)	2011-Nov-14	Notes

Other releases

Version	Downloads	Date	Links
6.x-3.0	tar.gz (1.13 MB) zip (1.31 MB)	2012-Jan-04	Notes

Development releases

Version	Downloads	Date	Links
7.x-3.x-dev	tar.gz (1.54 MB) zip (1.76 MB)	2012-Jun-09	Notes
6.x-3.x-dev	tar.gz (1.1 MB) zip (1.26 MB)	2012-May-19	Notes

Diferents versions i codi de colors en la pàgina del mòdul Views.

Com hem dit, hi ha infinitat de mòduls, amb funcionalitats totalment diverses, alguns tan complets que requeririen diversos capítols.

Exercici

Instal·leu i proveu algun d'aquests mòduls en un servidor de proves:

- XML Sitemap (<http://drupal.org/project/xmlsitemap>). Genera automàticament un arxiu sitemap.xml compatible amb Google en l'arrel del servidor.
- Pathauto (<http://drupal.org/project/pathauto>). Genera noms d'adreces per a les pàgines segons els títols del contingut (per exemple, /unexemple en lloc de /node/24).
- Ubercart (<http://drupal.org/project/ubercart>). Mòdul de comerç electrònic de gran complexitat i amb moltíssimes opcions. En realitat, es tracta d'un grup extens de mòduls.
- Simplenews (<http://drupal.org/project/simplenews>). Sistema complet de gestió de *newsletters*.
- FiveStar (<http://drupal.org/project/fivestar>). Sistema de votació/valoració dels usuaris.
- Organic Groups (<http://drupal.org/project/og>). Gestió de grups d'usuaris.

6. Taula de correspondències entre Drupal i WordPress

Arribats a aquest punt, és molt probable que hàgim trobat similituds entre les plataformes WordPress i Drupal. De fet, molts conceptes en totes dues plataformes coincideixen però tenen noms diferents.

Vegem a continuació una taula d'equivalències d'alguns d'aquests conceptes:

WordPress	Drupal	Comentaris
<i>Plug-ins</i>	Mòduls	En altres plataformes es diuen <i>extensions</i> . El Drupal parteix inicialment amb molts més mòduls que el WordPress.
Ginys (<i>widgets</i>)	Blocs	En WordPress existeix, a més, arrossegar i deixar caure.
Tema	Tema	En tots dos casos els temes són dins de la carpeta "Themes" del servidor.
Escriptori	Escriptori	Visió general del panell d'administració.
Entrades i pàgines	Articles i pàgines	Els dos tipus de contingut que hi ha per defecte en totes dues plataformes.
.php	.tpl.php	Extensió de les plantilles del tema.
functions.php	template.php	Arxiu del tema per a afegir instruccions PHP de cada sistema.
style.css	nom.info	Arxiu en què es declaren les opcions del tema.

A primera vista podríem considerar el Drupal com una ampliació de WordPress, en el sentit que té més funcionalitats per defecte, sense necessitat d'instal·lar mòduls.

Algunes d'aquestes funcionalitats que el WordPress (tret que instal·lem *plug-ins*) no té per defecte són les següents:

- Possibilitat de definir nous tipus de contingut.
- Possibilitat de definir camps en els tipus de contingut i de decidir l'ordre o com es visualitzen.
- Gestió de continguts en múltiples idiomes.
- Suport per a RDF.

- El Drupal no disposa d'un editor de textos WYSIWYG preinstal·lat (el WordPress sí, TinyMCE), però en permet instal·lar una diversitat.
- Sistema de gestió d'usuaris, rols i permisos molt més complet.

Com a contrapartida, la corba d'aprenentatge de Drupal és més llarga que la de WordPress a causa de la seva complexitat.

El Drupal és probablement una opció per a llocs web complexos amb una estructura de continguts que s'hagi de personalitzar a fons segons els requisits.

7. Integració amb xarxes socials

Actualment, una aplicació web no es pot considerar completa si no implementa funcionalitats que la vinculin amb xarxes socials. Si els continguts que es publiquen en el nostre lloc es publiquen a més en altres xarxes, n'augmentarem molt la visibilitat i aquesta visibilitat és una porta d'entrada a noves visites.

A més d'usar les xarxes com a mitjà de publicació, també podem **integrar components de xarxa social** en el nostre lloc seguint algunes normes elementals:

- Que els usuaris del web puguin generar continguts (creant tipus de contingut amb els permisos apropiats per a usuaris registrats o anònims).
- Que els usuaris quedin identificats amb el seu nom i el seu avatar (imatge de perfil).
- Que puguin crear grups i contactes, i es puguin enviar missatges entre ells. Aquestes funcionalitats no existeixen per defecte però es poden proporcionar mitjançant mòduls addicionals.
- A més, podem facilitar l'accés utilitzant de comptes externs:
 - OpenId (mòdul del *core*)
 - Facebook Connect (mòdul addicional)
 - Twitter (mòdul addicional)

PERSONALIZACIÓN

Permitir firmas.

Activar imágenes de usuario.

Directorio de imágenes

Subdirectorio de la carpetas de archivos subidos en el que se guardarán las imágenes.

Imagen predeterminada

URL de la imagen que se debe mostrar para los usuarios que no hayan personalizado su foto. Déjelo vacío para no usar

Estilo de presentación de imagen

El estilo seleccionado se usará en la presentación y la imagen original se mantendrá. Los estilos se pueden configurar en

A "Configuració/Opcions del compte" podem activar les imatges de perfil dels usuaris i fins i tot associar una imatge predeterminada.

Hi ha diversos mòduls addicionals (no inclosos en el *core*) que faciliten l'ús del CMS com a xarxa social:

- **Organic Groups** (<http://drupal.org/project/og>). Permet als usuaris crear i administrar els seus propis grups. Cada grup pot tenir subscriptors i mantenir una pàgina principal del grup en què els subscriptors es comuniquen entre ells.
- **User Relationships** (http://drupal.org/project/user_relationships). Permet als usuaris crear relacions entre ells. És un component bàsic per a una xarxa social o qualsevol altre lloc en què els usuaris s'hagin de comunicar entre ells.
- **PrivateMsg** (<http://drupal.org/project/privatemsg>). Permet als visitants del lloc web enviar-se missatges privats. Perfecte per a llocs orientats a la comunitat, com els fòrums i llocs de xarxes socials.
- **Invite** (<http://drupal.org/project/invite>). Les invitacions són importants per crear a efectes de xarxa i el creixement exponencial d'una comunitat d'interessos. En aquest mòdul s'afegeix una funció "Convidar un amic".
- **Contact Importer** (http://drupal.org/project/contact_importer). Permet importar usuaris d'altres xarxes socials.
- **Fivestar** (<http://drupal.org/project/fivestar>) (depèn de la Voting API). Sistema de votació per als continguts.
- **Radioactivity** (<http://drupal.org/project/radioactivity>). Sistema per a prioritzar informació segons la popularitat: el més votat, comentat, vist, etc.
- **Activity** (<http://drupal.org/project/activity>). Mostra informació de l'activitat de cada usuari (tal com fa Facebook, per exemple).
- **Feeds** (<http://drupal.org/project/feeds>). Permet integrar feeds (fonts d'informació en RSS) que generen automàticament contingut (nodes).
- **Twitter** (<http://drupal.org/project/twitter>). Integra el compte de Drupal amb el de Twitter, de manera que els continguts que publiquem en el nostre lloc es publiquen automàticament a Twitter
- **Gravatar** (<http://drupal.org/project/gravatar>). Integra Gravatar com a imatge de perfil dels usuaris. Gravatar és un sistema global d'imatges de perfil associades a comptes de correu. En realitat, Gravatar és un servei que permet obtenir la imatge de perfil de qualsevol usuari a partir de la seva adreça de correu. En activar aquest mòdul, la imatge de perfil dels usuaris vindrà del servei Gravatar en lloc del sistema intern de Drupal.

Pàgina web

Per a més informació es pot consultar la pàgina web de Gravatar: <http://es.gravatar.com/>.

Altres mòduls que poden ser interessants són:

- Subscriptions (<http://drupal.org/project/subscriptions>). Permet als usuaris subscriure's als continguts.
- CAPTCHA (<http://drupal.org/project/captcha>). Seguretat *antibots*.

Lectura recomanada

Per saber-ne més:
"Community building
and social networking
modules" (<http://drupal.org/node/206724>).

8. Creació de temes (*themes*) en Drupal

L'aspecte del nostre lloc desenvolupat en Drupal per als usuaris sempre es **personalitza** per mitjà d'un *theme* (tema).

En aquest sentit, és totalment equivalent a WordPress.

Un **tema** correspon a una sèrie d'**arxius** (**plantilles**) dins d'una carpeta comuna, que normalment se situa en la ruta Sites/All/Themes (de manera anàloga, els mòduls se situen a Sites/All/Modules).

En la pràctica (igual que en WordPress), també hi ha tres possibilitats per a personalitzar un tema:

- Utilitzar un tema derivat d'un que ja existeix (*child theme*).
- Utilitzar un tema derivat d'un tema *framework*.
- Crear un tema des de zero.

8.1. Modificar un tema que ja existeix

Sens dubte, l'opció més senzilla és descarregar un tema que ja existeix, per exemple de la pàgina de temes oficial de drupal.org, copiar-lo en el directori corresponent (Sites/All/Themes) i activar-lo des del panell d'administració.

Una vegada hem activat el tema, sempre podrem modificar el CSS per a personalitzar-lo segons les nostres necessitats, encara que no és la manera correcta de fer-ho, ja que no podríem actualitzar el tema a noves versions sense el risc de perdre els nostres canvis.

L'opció correcta és **crear un tema derivat**. Per a això hem de seguir els passos següents:

- Crear una carpeta nova a Sites/All/Themes amb el nom del nostre tema, per exemple /exemple.
- En aquesta carpeta crear un arxiu de text, amb el mateix nom que la carpeta, exemple.info en què hi haurà les dades principals del tema (correspon a style.css de WordPress).

Observació

El panell de Drupal ofereix també la possibilitat de pujar l'arxiu directament o copiar l'URL de l'arxiu gzipped del tema, però a efectes pràctics el resultat és el mateix, copiar els arxius en un directori de Sites/All/Themes.

- En l'arxiu info definirem la informació que defineix el tema.

```
;el nom del tema
name = Exemple

description = Tema derivat d'un altre

; obligatori, la versió del "core"
core = 7.x

; full d'estils, almenys en definim un de nou, deriva els del pare
stylesheets[all][] = exemple.css

;necessitem definir totes les regions, encara que siguin en el pare
;obligatòriament hi ha d'haver la regió 'content'
regions[content] = Contingut

;el tema del qual deriva, ha d'existir a sites/all/themes
base theme=unaltretema
```

Exercici

Visiteu la pàgina de temes de Drupal, escolliu-ne un, descarregueu-lo i activeu-lo en la nostra instal·lació local de Drupal. Creeu un tema derivat i activeu-lo. Personalitzeu algun detall del CSS i observeu el resultat.

8.2. Utilitzar un tema *framework* com a base

Utilitzar un tema base qualsevol no és, sovint, la millor opció, ja que probablement tingui una estructura fixa difícil de modificar.

Hi ha un tipus de temes pensats especialment per a crear temes derivats, els denominats **temes *framework***.

Entre aquests, tenim “Zen” (<http://drupal.org/project/zen>). “Zen” és un tema especial de Drupal en el qual estan definides totes les etiquetes en el CSS, però estan “buides”. És un tema *framework* que té com a objectiu facilitar-nos al màxim la creació d'un tema totalment personalitzat, però fent al mateix temps part del treball per nosaltres.

Les etiquetes del CSS corresponen a les etiquetes de la plantilla de manera que, bàsicament, l'única cosa que hem de fer en molts casos és modificar l'arxiu CSS per a crear la nostra personalització.

“Zen” ens facilita la feina de crear un tema derivat, ja que ens proporciona una plantilla bàsica per a això; es tracta de la carpeta Starterkit. Simplement l’hem de copiar a “Sites/All/Themes” i fer una sèrie de canvis documentats a <http://drupal.org/node/1549668>.

El tema “Zen” no sols està pensat per a modificar el CSS, sinó que també permet modificar les plantilles “tpl.php”, la qual cosa dóna una gran flexibilitat per a crear temes a mida amb una estructura diferent.

Hi ha molts més temes *framework* per a Drupal, entre els quals podem destacar:

- Omega.
- AdaptativeTheme.
- Fusion.

8.3. Crear un tema des de zero

Com a tercera opció veurem com podem crear el nostre propi tema partint de zero. Aquest és el sistema que triaríem en cas de voler adaptar un disseny creat en HTML i CSS a una plantilla dinàmica de Drupal. És el **mètode** que ofereix **més flexibilitat en el disseny i l’estructura de navegació**, però també, i en conseqüència, **el que inclou més complexitat i cost en temps**.

Les plantilles del tema defineixen tant l’estructura de navegació del lloc, com la presentació (bàsicament definida per l’arxiu o arxius CSS).

Hem de tenir identificades les zones de la pàgina en què volem que el contingut sigui dinàmic, que es transformaran en les regions. Convencionalment, aquestes regions són la capçalera, una barra lateral i el *footer*, però pot ser que en necessitem d’altres.

Abans de veure com s’implementa un tema complet veurem un cas extrem: el tema més simple que podem desenvolupar.

8.3.1. Creació d’un tema “minimalista”

Intentarem crear el tema més simple possible que permeti la navegació pels continguts de Drupal. Primer hem de pensar un nom per al tema, i posarem aquest nom a la carpeta en què desarem els arxius. En el nostre cas imaginarem que el nostre primer tema es diu *minimalista*. Per tant, la ruta de la carpeta serà: Sites/All/Themes/minimalista.

Per a crear la nostra plantilla (*theme*) a mida necessitem almenys **dos arxius**, els més importants:

- **minimalista.info**: la configuració de la nostra plantilla.

Observació

Sempre és important fixar-se que la versió dels temes o mòduls que baixem de drupal.org o un altre lloc sigui compatible amb la versió del *core* que fem servir.

Pàgina web

Llista completa de temes *framework* o *starter*: <http://drupal.org/node/323993>.

- **page.tpl.php**: la plantilla que utilitzaran totes les pàgines.

En l'arxiu **minimalista.info**, que no és més que un arxiu de text, definim el nom, els fulls d'estil, els arxius JavaScript i les regions del nostre tema.

En el nostre cas **minimalista** podríem tenir el següent:

```
name = Minimalista ;el nom del tema
description = Tema molt molt simple d'exemple
engine = phptemplate ; obligatori
stylesheets[all][] = style.css ; full d'estils, podem afegir més
regions[content] = Contingut ;aquesta regió és obligatòria!
regions[la_meva_regio] = Una sola regió extra ;
core = "7.x" ;obligatori, quina versió de Drupal es requereix
```

Una vegada hem creat aquest arxiu de text i l'hem desat com a **minimalista.info**, ens queda crear la plantilla principal del tema, anomenada "**page.tpl.php**". La plantilla "**page.tpl.php**" es correspon amb el contingut del `<body>`, de manera que aquesta etiqueta HTML **no** ha d'estar en la plantilla.

Com que estem creant un tema **minimalista**, el contingut podria ser el següent:

```
<h1>La meva plantilla</h1>

<?php if ($page['content']) : ?>
<div class="content">
<?php print render ($page['content']); ?>
<!-- mostrar el contingut de la regió content -->
</div>
<?php endif; ?>

<?php if ($page[la_meva_regio']) : ?>
<div class="la_meva_regio">
<?php print render ($page['la_meva_regio']); ?>
<!-- mostrar contingut de la regió la_meva_regio -->
</div>
<?php endif; ?>
```

Veiem que la plantilla té algunes instruccions de PHP incrustades. Aquestes instruccions s'encarreguen d'escriure dins del document el contingut assignat a les regions que tenim definides.

La primera línia significa 'si hi ha contingut definit en la regió *content*'. A continuació, la instrucció

```
print render ($page['content']);"
```

imprimeix el contingut d'aquesta regió en la pàgina, i el mateix per a la regió "la_meva_regio".

Però, com es tradueix això en l'administrador de Drupal?

Per a veure'l en funcionament el que hem de fer en primer lloc és activar aquest tema minimalista en el panell d'administració, en l'opció Aparença (*Appearance*).

El *content* d'una pàgina de Drupal és el contingut que es mostra en la pàgina. Si som en la portada, serà la llista de nodes publicats en la portada en format "Teaser"; si som en la pàgina única d'un node ("node/xx"), serà el contingut complet d'un node, o seran els resultats d'una cerca, etc. Per tant hi haurà molt poques circumstàncies en què "\$content" no tingui contingut.

Fent un paral·lelisme amb WordPress,

```
print render ($page['content']);
```

equivol al *loop* de WordPress.

En aquest sentit, la regió "content" actua d'una manera especial. La resta de regions només mostrarà els blocs que li hàgim assignat en el panell d'administració en l'apartat "Estructura/Blocs". En aquest cas correspon als *sidebars* de WordPress.

En el nostre exemple, "la_meva_regio" serà una regió a la qual haurem d'assignar blocs mitjançant el panell d'administració. En entrar a "Estructura/Blocs" veurem que podem assignar els blocs a la llista de regions definides a info, en el nostre cas només "content" i "la_meva_regio":

```
<d7_regions_minimalista.jpg>
```

The screenshot shows the 'Blocks' configuration page in Drupal. At the top, there's a breadcrumb 'Home » Administration » Structure' and a 'MINIMALISTA' theme indicator. Below the header, there's an introductory paragraph and a link to 'Demonstrate block regions (Minimalista)'. A '+ Add block' button is visible. The main part of the page is a table with three columns: 'BLOCK', 'REGION', and 'OPERATIONS'. The table is divided into three sections: 'Contenido', 'Una sola región extra', and 'Disabled'. In the 'Contenido' section, several blocks are listed with their assigned region set to 'Contenido'. A dropdown menu is open for the 'Main page content' block, showing options: 'Contenido', '- None -', 'Contenido', and 'Una sola región extra'. The 'Main menu' block is assigned to the 'Disabled' region.

Assignació de blocs a les regions definides en el nostre arxIU .info.

Si hem vinculat els blocs del formulari de cerca i el *user login* a la regió “la_meva_regio”, la màgia del Drupal farà que en la nostra plantilla, la instrucció

```
<?php print render ($page['la_meva_regio']); ?>
```

escriuri el contingut de tots aquests blocs automàticament en el seu lloc.

Una vegada activat el nostre tema “minimalista” podem navegar per tots els continguts del nostre lloc. Per descomptat falten detalls importants, com per exemple el full d'estils. A continuació crearem un **full d'estils** també minimalista. Per a això editarem un arxIU anomenat *style.css* i afegirem els estils assignats als selectors que hem introduït en la plantilla “page.tpl.php”:

```
.mycontent{
background-color: #ffcccc;
width:80%;
float:left;
}
.la_meva_regio{
background-color: #ccccff;
width:20%;
float:right;
}
```

D'aquesta manera podrem diferenciar visualment les dues regions que tenim definides en la nostra plantilla.



Resultat visual en aplicar l'anterior CSS al tema

Observeu que si naveguem pel lloc, el que canvia és el *content*, mentre que el contingut de la regió “la_meva_regio” és sempre el mateix. En Drupal, per defecte, els blocs assignats a les regions (excepte “content”) es veuen en totes les pàgines del lloc. Exactament igual que en WordPress amb els ginys, que per defecte es veuen en totes les pàgines

Observació

Si analitzem el codi font de la nostra pàgina, veurem que el Drupal ha afegit moltes coses a l'HTML que tenim en el nostre tema: entre altres coses, el codi de la capçalera de la pàgina, que inclou el nom, fulls d'estil i JavaScripts, etc. També afegeix el *tag* “body” amb classes específiques per a cada node o secció. Més endavant veurem que hi ha una altra plantilla més general que també podem editar, que és la que engloba “page.tpl.php”, anomenada “html.tpl.php”. En aquesta plantilla tenim la part del document externa al cos (*head*, etc.).

8.3.2. Creant un tema més complet

Per descomptat, podem definir un tema tan en profunditat com vulguem. Totes les parts d'una pàgina en Drupal provenen d'alguna plantilla que es pot redefinir, per exemple els blocs, els comentaris, els nodes, etc.

Per a crear una plantilla sempre cal partir de l'original. La plantilla original és la que s'utilitza per defecte si no n'hem especificat cap. En aquest enllaç, per exemple, tenim l'original de la plantilla “page.tpl.php” i les variables predefinides per Drupal que podem utilitzar. La resta de plantilles les podem trobar en la pàgina de documentació de Drupal.

Sempre que no definim alguna de les plantilles, el Drupal usarà la que hi ha per defecte. La idea sempre és redefinir el que ja hi ha.

Alguns exemples de plantilles:

Observació

A diferència de WordPress, en Drupal les variables predefinides són diferents segons la plantilla que utilitzem. A més, aquí no existeix el concepte de *loop*.

- **html.tpl.php.** Com hem comentat anteriorment, és la que inclou l'estructura del document HTML (excepte el contingut de cos que correspon a "page.tpl.php").
- **node.tpl.php.** La plantilla d'un node. S'utilitza per a mostrar qualsevol contingut de qualsevol tipus de contingut. Podem definir plantilles específiques per a cada tipus de contingut utilitzant el seu nom, per exemple "node--noticia.tpl.php", suposant que tenim un tipus de contingut "Notícia". Correspon al contingut del *loop* en WordPress.
- **field.tpl.php.** Atès que els tipus de contingut poden tenir camps específics, cada camp pot tenir una plantilla especial. "field.tpl.php" defineix la plantilla per a tots els camps per defecte, però podríem definir la plantilla "field-name-field_foto" en el cas d'un camp anomenat "field_foto".

Imaginant que ja tenim un disseny maquetat en HTML + CSS, amb una sola estructura de pàgina, els passos que hauríem de seguir per a crear un tema funcional serien:

- Copiar l'HTML i el CSS de la nostra plantilla de la maquetació en la carpeta del tema i canviar el nom de l'arxiu HTML per "page.tpl.php" (aquest serà la plantilla principal del nostre tema).
- Crear l'arxiu ".info".
- Copiar-lo a "Sites/All/Themes" i activar-lo.

8.3.3. Exercici. Creació d'un tema des de zero a partir d'una plantilla estàtica (HTML + CSS)

Partirem d'una maquetació estàtica en HTML i CSS.

En aquest exercici partirem dels arxius següents:

- index.html. La plantilla principal.
- nasa_estils_responsive.css. El full d'estils.
- Arxius d'imatge en /img.

L'objectiu és aconseguir que aquesta plantilla estàtica es converteixi en dinàmica i s'integri amb Drupal. Per a això hem de seguir els passos següents. Atenció, és important esborrar la memòria cau (*cache*) de Drupal cada vegada que fem canvis en el nostre tema, ja que si no, no podrem veure els canvis.

1) Copiar la plantilla estàtica com a carpeta dins de "Sites/All/Themes" (per exemple, "Sites/All/Themes/nasa").

Material

Els arxius es poden obtenir a http://www.kiwoo.org/recursos/ejemplos/drupal/maquetacion_nasa_responsive.zip

2) Afegir un arxiu de text anomenat *nasa.info*. En aquest arxiu hem d'indicar la informació bàsica i els fulls d'estil que utilitzem.

```
name = Nasa
description = Un tema basat en una plantilla estàtica prèvia
core = 7.x
project = "nasa"
regions[content] = Content
regions[sidebar] = Sidebar
regions[footer] = Footer
stylesheets[all][] = nasa_estils_responsive.css
```

3) Fer una còpia de “index.html” com a “page.tpl.php”. Eliminar el contingut del principi fins a <body> inclòs, i al final eliminar també </body></html>. La plantilla “page.tpl.php” solament és del contingut de la pàgina (el que hi ha dins de <body>).

4) A partir d'aquest moment ja podríem activar el tema (per a provar-lo), però apareixerien alguns problemes:

- El contingut és estàtic (naturalment).
- Si hi ha imatges en l'HTML no apareixen (atès que el Drupal no troba la ruta).

Per a solucionar el problema de la ruta de les imatges les hem de substituir per rutes dinàmiques. Per exemple, “img/badge_001.png” s'ha de convertir en:

```
<?=$base_path.$directory?>/img/badge_001.png.
```

Consell

Podem fer un *replace all* en el nostre editor.

5) On apareix el contingut? Aquesta instrucció mostra el contingut de la regió “content”:

```
<?= render($page['content']);?>
```

La regió “content” és la més important, ja que mostra el contingut que ha d'anar en cada pàgina. Sense aquesta regió, totes les pàgines serien iguals.

Per a la resta de regions hem de canviar el nom *content* pel nom de la regió corresponent. Les regions que s'han d'utilitzar a “page.tpl.php” són les definides en el “.info”, però també n’hi ha d’altres d’internes de Drupal (com “help”, “action_links”, etc.).

6) Podem consultar totes les variables que es poden utilitzar a “page.tpl.php”, incloses les regions per defecte a <http://api.drupal.org/api/drupal/modules%21system%21page.tpl.php/7>.

7) En aquest exercici només creem una plantilla, la principal (“page.tpl.php”), però en podríem definir moltes altres. Podem trobar una llista de totes les plantilles possibles a <http://drupal.org/node/190815>.

8) Menú. A “page.tpl.php” hi ha una variable que conté les entrades del menú. Ho podríem comprovar escrivint en la plantilla:

```
<?php
print_r($main_menu);
?>
```

No obstant això, necessitem el menú en format de llista desordenada. Afortunadament, el Drupal proporciona una funció que ens ajuda a fer-ho:

```
<?php print theme_links( array('links' => $main_menu, 'attributes' => array('class' =>
array('lamevaclase'))); ?>
```

(“\$main_menu” és el nom del menú principal. Hi ha molts paràmetres que es poden utilitzar. També podem optar per copiar aquest codi d'un altre tema).

9) *Footer* dinàmic. No és més que una altra regió (definida en l'arxiu .info) i, per tant, per a fer-la aparèixer:

```
<?php
print render($page['footer']);
?>
```

10) Barra lateral. De la mateixa manera, on volem que aparegui el codi, farem:

```
<?php
print render($page['sidebar']);
?>
```

Lectura recomanada

En relació amb aquests paràmetres es pot consultar: “Theme Links” (http://api.drupal.org/api/drupal/includes%21theme.inc/function/theme_links/7).

11) És necessari mostrar informació interna del sistema com *tabs* o missatges. “\$messages” són els missatges interns de Drupal que apareixen, per exemple, en enviar correctament els camps d'un formulari, etc. Perquè puguem veure aquests missatges en el nostre tema hem d'afegir sempre:

```
<?php print $messages; ?>
```

Els *tabs* són pestanyes que afegeix Drupal en algunes pàgines, normalment destinades a l'administrador del web.

Un cas típic és l'opció d'editar un contingut quan estem connectats com a administradors (*tabs*; veure, *edit*):

```
<?php if ($tabs): ?><?php print render($tabs); ?><?php endif; ?>
```

12) Podem definir altres plantilles. Per exemple, si necessitem una estructura diferent especial per al *home* modificarem:

```
page--front.tpl.php
```

13) De fet, fins i tot podríem crear plantilles per a un node en concret:

```
page--node--xx.tpl.php
```

en què “xx” és l'ID del node.

Això ofereix bastant flexibilitat en la implementació del nostre disseny.

14) Una de les plantilles més importants és “node.tpl.php”. Aquesta és **la plantilla de cadascun dels continguts**.

La plantilla base de “node.tpl.php” es troba a “Modules/Node”. La podem utilitzar com a base i modificar-la.

```
<div id="node-<?php print $node->nid; ?>" class="<?php print $classes; ?> clearfix"
<?php print $attributes; ?>>
<?php print $user_picture; ?>
<?php print render($title_prefix); ?>
<?php if (!$page): ?>
<h2<?php print $title_attributes; ?><a href="<?php print $node_url; ?>">
<?php print $title; ?></a></h2>
<?php endif; ?>
```



```
<?php print render($title_suffix); ?>
<?php if ($display_submitted): ?>
<div class="submitted"><?php print $submitted; ?> </div>
<?php endif; ?>
<div class="content"><?php print $content_attributes; ?>>
<?php // We hide the comments and links now so that we can render them later.
hide($content['comments']);
hide($content['links']);
print render($content); ?>
</div>
<?php print render($content['links']); ?>
<?php print render($content['comments']); ?>
</div>
```

Si ho comparem amb WordPress, “node.tpl.php” seria l'interior del *loop*. Ja hem comentat que en Drupal el concepte de *loop* no existeix.

Quan en la plantilla principal escrivim:

```
<?php print render($page['content'])?>
```

El Drupal inclou tantes vegades com faci falta les plantilles “node.tpl.php” segons els continguts que hi hagi en la pàgina.

15) Podem definir plantilles per a cada tipus de contingut. Per exemple, per al tipus de contingut “Notícia” crearem “node--noticia.tpl.php” (compte: són dos guions!).

16) A més, podem tenir plantilles per als camps especials, com per exemple “field--field_audio.tpl.php”. Atenció, **en fer canvis en el nostre tema és important esborrar la memòria cau (cache) de Drupal abans de provar el resultat** .

En la nostra plantilla especial per al tipus de contingut podem mostrar els camps on vulguem, primer amagant-los i després mostrant-los on vulguem. Per exemple:

```
<?php
hide($content['field_foto']);
print render($content);?>
<div id="foto">
<?php print render($content['field_foto']);?>
</div>
```

Atenció, en el panell d'administració, en l'opció *Tipus de contingut*, podem canviar si es mostren els camps o no (en mode “Teaser” i en mode “Pàgina”) i com es mostren d'una manera molt més fàcil. Només modificarem la plantilla si no hi ha una altra opció.

17) En les nostres plantilles de node podem utilitzar variables per a mostrar la informació de manera diferent segons on ens trobem. Alguns exemples:

“\$teaser”: Ens indica si el node està en mode “Teaser” o no.

“\$is_front”: Ens indica si el node està en la portada.

\$logged_in: Indica que l'usuari està *loginado*, etc.

Hi ha moltes altres variables documentades a Drupal.org.

Resultat de l'exercici

Maquetació HTML+CSS convertida en tema de Drupal (a nivell molt bàsic):
http://www.kiwoo.org/recursos/ejemplos/drupal/maquetacion_nasa_responsive.zip