

Relazione per
Ingegneria del Software A
e
Reti di Calcolatori A

Valutazione pacchetto software
Drupal



Docente Prof. Agostino Poggi
Studente Claudio Pitzalis
Matricola 184949

Relazione e progetto sviluppati da Claudio Pitzalis, iscritto all'Università degli studi di Parma, facoltà di Ingegneria Informatica. Presentato per l'esame di Ingegneria del Software A e Reti di Calcolatori nell'anno accademico 2009/2010.

Indice

Drupal

Descrizione	3
Caratteristiche principali	4
Requisiti	7
Installazione	
in locale	9
in remoto	12
Uso di Drupal	16

Applicazione

Descrizione	23
Costruzione del tema	24
Costruzione di un modulo di prova	38
Il modulo fckeditor	40
Altri moduli	42
Il modulo ddblock e l'album fotografico	44
Conclusioni	45
Bibliografia	45

Drupal

Drupal è un Content Management System (CMS - sistema gestione contenuti), vale a dire un gestore di contenuti e di siti web dinamici, ed è realizzato in PHP (linguaggio di scripting lato server).

In particolare i CMS permettono a un singolo utente, a una comunità di utenti, o a un'impresa, di pubblicare facilmente, gestire e organizzare una grande varietà di contenuti su un sito web.

Alcuni esempi di CMS web:

Drupal: <http://www.drupal.org/>

Mambo: <http://www.mamboserver.com/>

PHPNuke: <http://phpnuke.org/>

AspNuke: <http://www.aspnuke.com/>

Joomla!: <http://www.joomla.org/>

per citarne alcuni dei più noti e concettualmente abbastanza simili: costituiti, cioè, da una serie di script lato server che, interagendo con un database, realizzano così, in corso di esecuzione (a run-time), le pagine web che mostrano infine i contenuti. E questi contenuti non sono semplicemente memorizzati in cartelle lato server del proprio hosting, come avviene normalmente, ad esempio, con le pagine html, ma vengono conservati nel database, e una eventuale loro modifica, rimozione o aggiunta sarà effettuata sempre attraverso interfacce web.

Una peculiarità interessante dei sistemi CMS è di facilitare ampiamente l'organizzazione e la cooperazione nella creazione di documenti e contenuti di vario genere, il cui utilizzo non è necessariamente limitato alla gestione di siti web, anche se attualmente proprio questo ne è l'uso più frequente.

Drupal, in particolare, è composto da una serie di script in **PHP** (Hypertext Pre Processor) che interagiscono con un **DMBS** (Database Management System).

Con Drupal è possibile realizzare una grande varietà di siti web (o intranet), per pubblicare articoli, circolari, news, insiemi di messaggi, commenti, form di discussione, blog, raccolta di immagini, ecc.

È open-source software, distribuito sotto licenza GPL ("GNU General Public License"): può essere liberamente scaricato, distribuito e installato ed è mantenuto, sviluppato e anche tradotto da una comunità di migliaia di utenti e sviluppatori. Questo è sicuramente uno dei grandi punti di forza dei più conosciuti CMS non proprietari e Drupal è senza dubbio, tra questi, uno dei più completi e famosi.

Un CMS, comunemente, è tanto più efficiente quanto più è specializzato e, per quanto possa essere flessibile, un sito basato su questa struttura in genere presenta un aspetto poco personalizzato se non è possibile intervenire direttamente sul codice sorgente del prodotto per modificarlo. Analogamente i contenuti saranno sempre ancorati a quanto previsto da chi ha progettato il CMS e non alle esigenze di chi pubblica il sito. Chi pubblica o gestisce il sito può usare il CMS per intervenire sui contenuti e sull'aspetto, ma generalmente (caso del software proprietario) non è in grado di intervenire direttamente (o far intervenire) sulla struttura del CMS stesso; questo è un limite strettamente connesso al vantaggio primario dei CMS: pubblicare un portale senza doverne progettare la struttura o senza possedere le conoscenze tecniche (o le risorse finanziarie) per uno sviluppo personalizzato.

Questi problemi sono risolvibili proprio utilizzando software open source: la possibilità di accedere al codice sorgente del prodotto permette di personalizzare il software sulla base delle proprie esigenze.

Tornando a Drupal, le funzionalità incorporate, insieme con migliaia di moduli aggiuntivi liberamente disponibili, permettono funzioni quali:

- Commercio elettronico
- Blog
- Ambienti di authoring collaborativo
- Forum
- Peer-to-peer
- Newsletters
- Podcasting
- Gallerie fotografiche
- Upload e download di file

e altro ancora.

Generalmente a chi si imbatte in questo software, dopo un grosso investimento specialmente in termini di tempo, si aprono nuove e interessanti possibilità prima inimmaginabili. Ed è il caso, ad esempio, dell'applicazione che ne è scaturita per questa valutazione.

Caratteristiche

Drupal è un applicazione Web, questo significa che i contenuti vengono inseriti e visualizzati attraverso un Web browser (Internet Explorer, Mozilla Firefox, Opera, Chrome, ecc.) ed è possibile realizzare con esso siti Web pubblici o locali (intranet).

E' interamente sviluppato in PHP, uno dei linguaggio più usati nel web. Utilizza come base di dati MySQL o PostgreSQL in modo nativo, ma è utilizzabile con ogni tipo di database server.

Drupal è sviluppato in modo da consentire agli utenti la registrazione e l'autenticazione così da tenere traccia di chi è autore di ogni singolo contenuto e permettere agli amministratori di definire livelli di accesso differenziati a seconda dei 'ruoli utente': come l'utente anonimo, il moderatore, l'amministratore, o livelli intermedi a cui il proprietario del sito concede solo alcune parti nella gestione dei contenuti.

È realizzato in modo modulare, consentendo, ad esempio, di aggiungere o di realizzare un testo direttamente da tastiera o con l'uso del 'copia e incolla': l'attivazione di moduli newsletter, che permettono la distribuzione di circolari e documenti, offre un'utile soluzione al problema dell'immediatezza e dell'efficienza delle operazioni di divulgazione delle informazioni. La facilità di inserimento dei contenuti infatti è tale da essere alla portata del principiante.

Al sistema di base è possibile aggiungere numerose funzionalità: attualmente esistono oltre 6000 (!) moduli scaricabili gratuitamente che consentono appunto di migliorare e ampliare il proprio lavoro.

Fondamentalmente Drupal tratta la maggior parte dei tipi di contenuti con lo stesso concetto astratto di **nodo**: le pagine, i blog e le notizie sono tutti inclusi in un unico oggetto, il nodo appunto, che contiene informazioni come titolo, contenuto, autore, ecc. La cosiddetta mappa del sito è gestito invece separatamente tramite il sistema dei menù, mentre la visualizzazione vera e propria dei contenuti viene gestito da un altro livello, fornito a sua volta da altri elementi: i temi e i moduli.

Mantenendo separati tali livelli si ottiene un'enorme flessibilità nella gestione dei contenuti: si possono mostrare, nascondere, raggrupparli in base alle utenze, alla rilevanza ('tassonomia'), alla data di creazione, e così via.

L'astrazione data dal concetto di nodo, utilizzato per tutti i contenuti, permette di gestire gli stessi, combinarli e relazionarli allo stesso modo, utilizzando un metodo comune.

Drupal permette, con un upload attraverso FTP e con una breve configurazione tramite una semplice interfaccia web, di avere un sito operativo in poco tempo. Inoltre, la gestione 'base' (inclusa nell'installazione) dei temi, degli utenti, dei forum, dei book, dei commenti, dei sondaggi, delle categorie di contenuto, permette di avere da subito tutto quello che occorre per gestire il sito web.

Come detto, è possibile in ogni momento estendere l'installazione con una grande varietà di moduli messi a disposizione dalla comunità di Drupal (<http://drupal.org/project/Modules>) o ancora modificandone il codice o sviluppandone *ex novo*, in proprio, con le necessarie capacità di sviluppo software.

Entrando ora un po' più nel dettaglio, la flessibilità di Drupal è data sostanzialmente dal meccanismo di divisione e sovrapposizione di cinque livelli distinti:

I - Dati:

sono il fulcro del sistema: tutto ciò che è visualizzato nel sito, deve prima essere inserito come dato.

II - Moduli:

agiscono a tutti gli effetti come dei plug-in che fanno parte o del 'motore' di Drupal oppure sono elementi creati dai membri della comunità di Drupal. I moduli forniscono funzionalità diverse, che permettono di espandere le capacità del sito aggiungendo funzioni come quelle elencate a pagina 4.

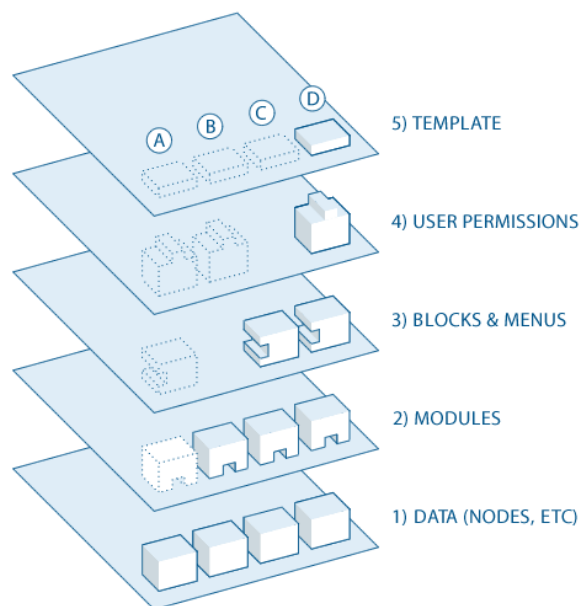


Immagine da <http://drupal.org/getting-started/before/overview>

III - Blocchi e menù:

i blocchi generalmente sono parte integrante di moduli specifici oppure possono essere creati per mostrare quello che si desidera: possono essere collocati in diversi punti (regioni) del tema.

I blocchi possono essere configurati per essere visibili solo su alcune pagine o solo a determinati utenti in base ai permessi decisi dall'amministratore del sito.

IV - Permessi utente:

come già accennato consentono di determinare 'chi' può fare 'cosa' nel sistema.

Si possono assegnare permessi per creare nuove pagine, eliminare pagine esistenti, inserire commenti o cancellarli, eseguire o meno upload, ecc. a ruoli specifici come l'utente anonimo, autenticato, moderatore, amministratore, e così via. Ad ogni utente è possibile assegnare uno o più ruoli, in modo da concedergli i relativi permessi associati.

La possibilità di realizzare aree a protezione differenziata consente di pubblicare con ampia sicurezza documenti riservati solo a particolari utenti.

V - Temi (o templates):

in questo strato, il 'più superficiale', si trova il modello del sito ed è costituito principalmente da codice XHTML e CSS, con alcune parti di PHP (i template) necessarie ad inserire nelle posizioni corrette i contenuti.

È molto importante capire come tutti i flussi di informazione passano attraverso i livelli sopra descritti, perché permette di interagire correttamente con il sistema e di 'localizzare' velocemente eventuali errori di configurazione del sistema. Se, ad esempio, un modulo non funziona come ci si aspetta, certamente ciò sarà dovuto al fatto di non aver attivato i blocchi necessari, oppure non sono stati assegnati agli utenti i giusti permessi, o altro ancora.

Questo sistema di 'flusso' dei contenuti rende Drupal adatto ad un utilizzo contemporaneo da parte di più figure professionali, mentre la separazione dei contenuti dalla presentazione lo rende adatto a qualsiasi tipo di interfaccia.

Normalmente i CMS sono pensati per adattarsi a numerose esigenze, e ognuno mantiene caratteristiche proprie che lo rendono più o meno adatto a seconda delle necessità e dell'esperienza dell'utente: alcuni sono fatti per essere facili, altri per essere funzionali. In questo contesto Drupal è considerato un CMS che guarda molto alla sostanza e meno alla forma, ma in realtà anche su quest'ultima è possibile intervenire notevolmente, dopo però aver preso molta confidenza col codice.

Per concludere, i punti di forza di questo CMS sono senza dubbio la grande flessibilità e configurabilità, la robustezza e la gestione della sicurezza.

Requisiti

I componenti richiesti per l'installazione di Drupal sono essenzialmente tre:

1. Un Web server che possa eseguire script PHP, ovvero un server in grado di pubblicare pagine Web. Il Web server predefinito è Apache: raccomandato [Apache 2.x](#) su UNIX, Linux, OS X o Windows. Ma sono supportati anche Apache 1.3, IIS5, IIS6, Lighttpd. La maggior parte degli Internet Provider mette a disposizione uno di questi Web server.

Nella scelta bisogna tenere conto del fatto che la maggior parte di sviluppo di Drupal e della distribuzione vengono eseguiti su Apache, per cui vi sono più esperienza della comunità e più collaudi eseguiti su Apache che su altri server web.

Per realizzare un'installazione locale, è invece necessario installare localmente un Web server: per questo generalmente viene suggerito XAMPP, un web server completo di tutto il necessario per far girare Drupal. XAMPP è un progetto open source basato proprio su Apache ed incorpora PHP e MySQL: come si vedrà in seguito, in questo contesto, per una prima valutazione del pacchetto Drupal, è stata fatta un'installazione di Drupal in locale utilizzando proprio XAMPP

2. L'interprete PHP. Drupal è un insieme di moduli di codice PHP: le pagine Web sono generate dinamicamente in base alle richieste ricevute, a partire dal codice PHP. Il sistema che converte il codice PHP in pagine Web fruibili dall'utente, è chiamato, appunto, interprete PHP. Per un'installazione Web, bisogna pertanto verificare che il proprio provider supporti PHP, per un'installazione locale, vale quello detto poco sopra su XAMPP, altrimenti è possibile scaricare liberamente l'interprete dal sito www.php.net. È suggerita la versione 5.2.x

3. Un database SQL. I dati di configurazione e i contenuti inseriti dagli utenti sono archiviati in un database SQL. Il database maggiormente consigliato per la versione più attuale 6.x di Drupal (in particolare la versione di questa valutazione sarà la 6.19) è MySQL 4.1 o maggiore scaricabile liberamente dal sito mysql.com, ma è supportato anche PostgreSQL 7.4 o maggiore. Per un'installazione Web, sarà necessario verificare che il proprio provider supporti i succitati database, per un'installazione locale, vale ancora il discorso fatto su XAMPP.

Un'installazione di base richiede un minimo di 3MB di spazio su disco ma, se si installano anche moduli e temi addizionali, si possono superare i 40MB, a cui va aggiunto lo spazio necessario per l'occupazione del database, dei file caricati ed altro.

Per quanto riguarda invece i requisiti di competenza richiesti a chi gestisce il sistema, nel caso di un'applicazione essenziale, ma completa, nonostante Drupal sia scritto in PHP, non è necessaria una conoscenza di questo linguaggio per utilizzarlo. Bisognerà usare solo alcune precauzioni per l'installazione accertandosi, ad esempio, che il proprio provider sia compatibile con la versione di Drupal o lo stesso PHP. Inoltre bisognerà verificare i permessi di alcune cartelle. Questi problemi ovviamente non sono riscontrabili in un'installazione in locale.

L'unico strumento indispensabile per fare alcune modifiche, che si vedranno in seguito, è un potente editor di testo. In questo caso, notepad++, un software free-ware liberamente scaricabile, ha svolto egregiamente queste funzioni.

Una volta che l'installazione è andata a buon fine, invece, sia per la gestione di Drupal da parte degli amministratori come per gli utenti di una sua applicazione, sarà sufficiente un Web browser come Internet Explorer, Mozilla Firefox, Opera, Chrome, ecc., sia per Web pubblici, sia per quelli locali nel caso di intranet.

Nel caso invece vi sia volontà o necessità di personalizzare in maniera più o meno sensibile la propria applicazione, requisiti indispensabili sono la buona conoscenza del linguaggio html e del CSS per modificare significativamente l'aspetto delle pagine web, mentre per modifiche più profonde al comportamento delle pagine o per la creazione di nuovi temi o moduli, bisogna avere una certa praticità nei codici javascript e nel linguaggio PHP.

Installazione

Per una installazione **locale**, il metodo che qui viene proposto è l'utilizzo del già citato XAMPP.

Andare su <http://www.apachefriends.org/en/xampp.html> e prelevare XAMPP per la propria piattaforma. Per chi non ha dimestichezza con questo pacchetto esistono guide adeguate per l'installazione su ogni sistema operativo:

Per Windows: <http://www.apachefriends.org/it/xampp-windows.html>

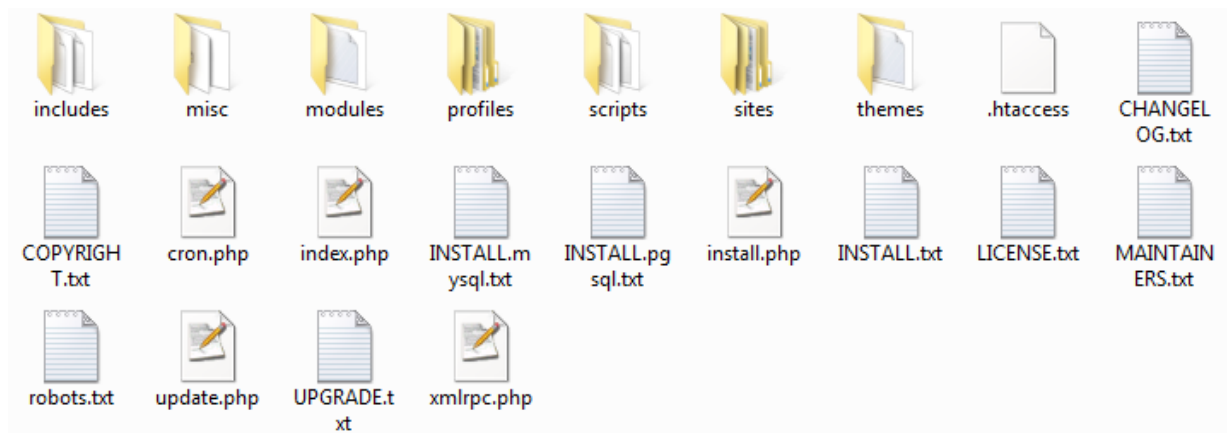
Per Linux: <http://www.apachefriends.org/it/xampp-linux.html>

Per Mac OS X: <http://www.apachefriends.org/it/xampp-macosx.html>

Una volta che l'installazione è andata a buon fine, bisognerà avviare XAMPP con i relativi servizi, in particolare apache e mysql.

A questo punto, dal sito <http://drupal.org>, è possibile scaricare l'ultima versione stabile di Drupal: al momento della stesura di questa valutazione, è disponibile la versione 6.19, mentre la versione 7.x è ancora in versione beta.

Scompattando il file compresso, ci si trova davanti ai seguenti file e cartelle:



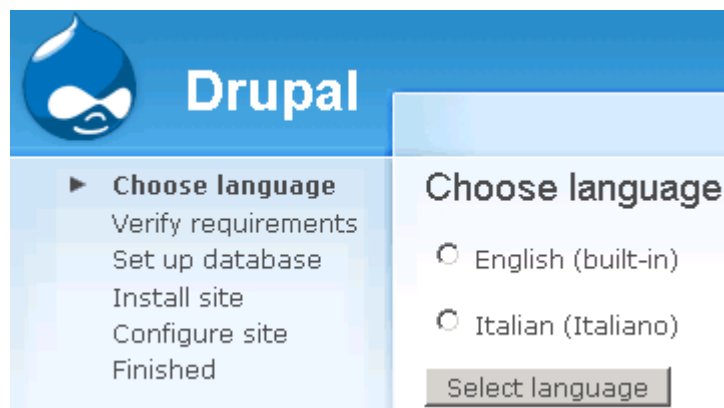
Copiare il tutto in X:\xampplite\htdocs\Drupal, dove X è la radice del percorso dove è stato installato XAMPP (ovviamente la cartella di destinazione 'Drupal' può avere un altro nome che sia significativo per l'applicazione desiderata). Se si desidera installare Drupal in italiano, si deve importare un file che contenga la traduzione di tutto (o quasi) il programma: questa traduzione è attualmente disponibile qui:

<http://www.drupalitalia.org/sites/drupalitalia.org/files/translations/drupal-6.17-it-translations.tgz>

di cui, estraendone il contenuto, si trovano tre cartelle (modules, profiles e themes) che si possono copiare, sovrascrivendole tranquillamente, nella cartella principale, in quanto viene semplicemente aggiunto, per ogni modulo, profilo o tema, un file di traduzione che altro non è che un file di testo con suffisso '.po'

Ecco ora i passi principali da eseguire per una corretta installazione:

1. andare su <http://127.0.0.1/phpmyadmin/> (o <http://localhost/phpmyadmin/> che è equivalente) dove, con estrema semplicità, si dovrà creare un db vuoto (con un nome sensato, ad esempio db-drupal)
2. A questo punto sarà possibile procedere con la configurazione avviando il browser e andando su <http://127.0.0.1/Drupal/>:



3. Una volta scelta la lingua, prima di proseguire, è necessario fare alcune modifiche importanti e dare autorizzazioni al file settings.php che si dovrà precedentemente creare: esiste una versione di default di questo file che trova in sites/default/default.settings.php all'interno della directory principale di Drupal. Questo file va copiato senza spostarlo o modificarlo, la copia va rinominata settings.php e lasciato nella stessa directory; in seguito bisognerà rendere il file editabile: in Linux 'chmod 666 sites/default/settings.php', in Windows invece si devono cambiare i permessi del file (tasto destro su file, finestra 'Protezione') e controllare che non sia marcato come "Sola lettura".

```

1  <?php
2  // $Id: default.settings.php,v 1.8.2.4 2009/09/14 12:59:18 goba Exp $
3
4  /**
5   * @file
6   * Drupal site-specific configuration file.
7   *
8   * IMPORTANT NOTE:
9   * This file may have been set to read-only by the Drupal installation
10  * program. If you make changes to this file, be sure to protect it again
11  * after making your modifications. Failure to remove write permissions
12  * to this file is a security risk.
13  *
14  * The configuration file to be loaded is based upon the rules below.
15  *
16  * The configuration directory will be discovered by stripping the
17  * website's hostname from left to right and pathname from right to
18  * left. The first configuration file found will be used and any
19  * others will be ignored. If no other configuration file is found
20  * then the default configuration file at 'sites/default' will be used.
21  *
22  *
23  * To provide prefixes for specific tables, set $db_prefix as an array.
24  * The array's keys are the table names and the values are the prefixes.
25  * The 'default' element holds the prefix for any tables not specified
26  * elsewhere in the array. Example:
27  *
28  *
29  *   $db_prefix = array(
30  *     'default'   => 'main_',
31  *     'users'     => 'shared_',
32  *     'sessions' => 'shared_',
33  *     'role'      => 'shared_',
34  *     'authmap'  => 'shared_',
35  *   );
36  *
37  * Database URL format:
38  *   $db_url = 'mysql://username:password@localhost/databasename';
39  *   $db_url = 'mysqli://username:password@localhost/databasename';
40  *   $db_url = 'pgsql://username:password@localhost/databasename';
41  */
42  $db_url = 'mysqli://Sqlxxxxx:xxxxx@xx.xxx.xxx.xxx/Sqlxxxxx_x';
43  $db_prefix = '';
44
45  /**
46   * Access control for update.php script

```

Alcuni frammenti del file settings.php

Il file settings.php potrà essere in qualsiasi momento modificato, con cautela e con i relativi permessi, per indicare, per esempio, un diverso database o inserire altri tipi di dati.

4. Tornando ora alla pagina di configurazione di Drupal, una volta scelta la lingua desiderata, si potrà proseguire con la configurazione del database: verranno

chiesti alcuni dati che devono corrispondere ai parametri inseriti durante la creazione del database in phpadmin, come visto nel punto 1. Si fa solo notare che generalmente la selezione è su mysql o msqli, che è anche la soluzione più fornita dai servizi di hosting. Esiste anche la possibilità di effettuare alcune impostazioni avanzate che qui non vengono espone.

5. Ormai al termine dell'installazione, se tutto è stato impostato correttamente, dovrebbe presentarsi questa finestra:

Per il nome del sito, che sarà utilizzato per identificare il sito Drupal, è consigliabile inserire un nome efficace.

Per quanto riguarda l'indirizzo e-mail, pur parlando di un'installazione in locale, per una valutazione completa del pacchetto, è possibile attivare il modulo 'Mercury' compreso nel pacchetto XAMPP che offre un servizio di trasporto di posta elettronica valido, permettendo di verificare, all'interno di un'applicazione

Drupal, il funzionamento della messaggistica. Quindi l'indirizzo e-mail del sito, che verrà utilizzato ad esempio nella procedura di registrazione di nuovi utenti, dovrà essere del tipo *info@local-host*.

Il pannello successivo consentirà di configurare l'account di amministratore del sito, cioè colui che nel sito ha ogni tipo

di permesso: in Drupal un solo utente può avere permessi di amministratore. Inseriti quindi il nome dell'utente amministratore, l'e-mail e la password che a esso si vuole associare, non rimane, nell'ultimo pannello, che inserire il fuso orario, decidere se utilizzare URL SEF (Search Engine Friendly, cioè una URL che sia esplicativa per l'utente e attinente al contenuto per i motori di ricerca) e abilitare la notifica automatica degli aggiornamenti di Drupal disponibili.

Terminata l'installazione si ha immediatamente a disposizione un efficace, ma contemporaneamente semplice sistema per la creazione di contenuti.

Per un'installazione in **remoto**, la questione diventa per certi aspetti un po' più complicata, in quanto, rispetto all'installazione locale, dove tutto è 'sotto mano', non vi è immediato accesso a vari file per l'amministrazione del PHP e spesso non vi è possibilità di rendere alcune cartelle o file accessibili o scrivibili.

Vi sono sostanzialmente due metodi, entrambi testati per questa valutazione: nel primo metodo si procede come nell'esposizione fatta per l'installazione in locale, utilizzando però una cartella contenente il pacchetto Drupal di cui è stato fatto l'upload, tramite FTP, in una directory del proprio hosting, e tenendo conto dei limiti imposti dalla maggior parte dei provider. In particolare, bisognerà verificare, nel proprio server, se vi è la possibilità di accedere ai permessi di alcuni file per consentirne la modifica o il semplice accesso. Molti hosting di tipo Windows praticamente non consentono in alcun modo l'installazione di Drupal: oltre a poter modificare i permessi di alcune cartelle e file, è anche richiesto il file `.htaccess` per rinominare gli URL e sono necessarie le librerie GD le quali permettono al programmatore di creare vere e proprie immagini nei vari formati. Conseguentemente, l'ambiente 'naturale' per le ultime versioni di Drupal risulta essere un server con sistema operativo Linux e, come già specificato, un server Apache: la maggior parte dei provider offre tale ambiente nel loro server di produzione.

Nel caso specifico, per lo sviluppo dell'applicazione utile a questa valutazione, è stato necessario, a pagamento, convertire il primitivo hosting Aruba da Windows a Linux. È sufficiente però fare alcune ricerche per trovare, purtroppo, interi forum che trattano argomenti legati alle numerose problematiche riscontrate nell'installazione del pacchetto Drupal proprio su provider Aruba: la soluzione, istintiva, sarebbe quella di cercare direttamente provider immediatamente compatibili col pacchetto, ma con alcuni accorgimenti è possibile avere un'applicazione perfettamente funzionante anche con Aruba a patto, come detto, che si lavori su piattaforma Linux.

Approfondendo la questione, quando Drupal crea la cartella `files`, di default nella directory `sites/default` di Drupal, come sopra detto, provvede anche ad inserirvi un file `.htaccess` per scopo cautelativo e per impedire eventuali infrazioni da parte di utenti esperti. Ma tale file, in base alla configurazione del server su cui non è possibile intervenire, può creare problemi di visualizzazione dei file e immagini inserite, appunto, nella cartella `files`. Normalmente viene suggerito, per risolvere il problema, di aprire il

file `.htaccess` e commentarne tutte le righe, tramite il carattere cancelletto, oppure svuotarlo. Non va cancellato del tutto perché, alla prima occasione, Drupal lo ricrea.

```
1 SetHandler Drupal_Security_Do_Not_Remove_See_SA_2006_006
2 Options None
3 Options +FollowSymLinks
```

Il file `.htaccess` originale creato da Drupal

È molto importante accertarsi che la cartella `files` funzioni correttamente, perché essa è utilizzata anche da alcuni moduli e dal tema `garland`, il tema usato di default, e ospita i `files` `css` compattati.

In realtà, per un miglior funzionamento del sistema e per non incorrere in spiacevoli sorprese durante, ad esempio l'upload di file o l'inserimento di URL errati, la soluzione migliore travata consiste nel modificare nel seguente modo il file `.htaccess`

```
1
2 RewriteCond %{REQUEST_FILENAME} !-f
3 RewriteCond %{REQUEST_FILENAME} !-d
4 RewriteCond %{REQUEST_URI} !=/favicon.ico
5 RewriteRule ^(.*)$ index.php?q=$1 [L,QSA]
6
7
8 # $Id: .htaccess,v 1.90.2.5 2010/02/02 07:25:22 dries Exp $
9
```

Il file `.htaccess` modificato

Se durante la costruzione del sito si decidesse, per vari motivi, di scegliere una cartella diversa da quella di default, bisognerà effettuare la stessa modifica, appena sopra esposta, nel file `.htaccess` che si andrà creando all'interno di questa nuova cartella.

Per quanto riguarda invece le comuni problematiche dovute all'hosting Aruba, in aggiunta alle modifiche appena esposte, si risolve ogni inconveniente eseguendo modifiche su un altro file `.htaccess`, questa volta però posizionato nella cartella principale: vanno cercate le righe

`Options -Indexes` (riga 11)

`Options +FollowSymLinks` (riga 14)

`DirectoryIndex index.php` (riga 26)

a cui va aggiunto in testa il carattere cancelletto, `#`, anche se sarebbe sufficiente semplicemente cancellarle, ma normalmente questo metodo non fa parte della politica di una buona programmazione.

Un'ultima modifica importante, che è stata particolarmente difficile da trovare e che serve per evitare il temibile 'Internal Server Error', è quella di estrarre un blocco di righe dall'ultima struttura di controllo, racchiusa tra i tag `<IfModule mod_rewrite.c>` e `</IfModule>`, per inserirlo in coda ad essa, come illustrato qui di seguito:

```

108
109 # Rewrite URLs of the form 'x' to the form 'index.php?q=x'.
110 RewriteCond %{REQUEST_FILENAME} !-f
111 RewriteCond %{REQUEST_FILENAME} !-d
112 RewriteCond %{REQUEST_URI} !=/favicon.ico
113 RewriteRule ^(.*)$ index.php?q=$1 [L,QSA]
114 </IfModule>
115
116 # $Id: .htaccess,v 1.90.2.5 2010/02/02 07:25:22 dries Exp $
117

```

.htaccess prima della modifica

```

109
110 </IfModule>
111 # Rewrite URLs of the form 'x' to the form 'index.php?q=x'.
112 RewriteCond %{REQUEST_FILENAME} !-f
113 RewriteCond %{REQUEST_FILENAME} !-d
114 RewriteCond %{REQUEST_URI} !=/favicon.ico
115 RewriteRule ^(.*)$ index.php?q=$1 [L,QSA]
116 # $Id: .htaccess,v 1.90.2.5 2010/02/02 07:25:22 dries Exp $
117

```

Modifica al file .htaccess della cartella principale di Drupal

A questo punto, il procedimento è del tutto simile a quello previsto per una installazione locale tranne il fatto che 'localhost' sarà sostituito da un indirizzo internet, URL, esterno.

Un altro metodo di installazione testato è quello, una volta costruita e perfezionata l'applicazione (a tutti gli effetti un sito) in locale, di utilizzare il modulo aggiuntivo 'Backup and Migrate' messo a disposizione nel sito ufficiale di Drupal: questa modalità implica una certa competenza nell'uso di MySQL e l'acquisizione di un livello medio di praticità nella gestione dei moduli di Drupal.

Brevemente, una volta installato il modulo, eseguito l'accesso tramite browser al sito Drupal locale e attivato il modulo, nella sezione "Backup and Migrate" si presentano varie tabs: Backup/Export DB, Restore/Import DB, Saved Backups e Backup Schedule.

In Backup/Export DB sono elencate le tabelle del database che si intende salvare e quelle che si intende escludere dal Backup. Si può decidere se salvare il file di backup sul proprio pc o in una cartella protetta dello spazio web.

Successivamente, una volta entrati nella pagina di amministrazione database remota del proprio server, sarà possibile importare il file appena salvato. A questo punto, tramite FTP, si dovrà esportare per intero la propria cartella locale con l'installazione di Drupal sul server remoto e dopo aver modificato il file settings.php, dove verranno inserite le coordinate corrette del nuovo database appena creato, il sito sarà perfettamente funzionante.

In verità esiste un'altra procedura d'installazione di Drupal offerta però solo da alcuni provider: questi permettono una diretta installazione del pacchetto attraverso un certo numero di semplici click col mouse e senza necessità di alcun accesso al database e del trasferimento di dati tramite ftp. Questa procedura è ottima per utenti inesperti, i quali saranno in ogni caso in possesso di un valido strumento per gestire il proprio sito.

Uso di Drupal

L'utilizzo primario che generalmente ci si propone una volta installato il pacchetto, è creare dei contenuti, in particolare pagine web, da diffondere in rete. Al primo accesso, nella homepage della directory di Drupal, si trova una pagina di benvenuto dove vengono brevemente indicate le possibilità di base date da questo CMS:



The screenshot shows the Drupal administration interface. On the left is a sidebar with the user name 'admin' and a menu with options: 'Profilo utente', 'Crea contenuto', 'Amministra', and 'Esci'. The main content area is titled 'Benvenuto su Drupal! Questa traduzione è stata realizzata con l'apporto della comunità italiana di Drupal: [Drupal Italia](#)'. Below the title, it says 'Segui questi passi per impostare il tuo sito ed incominciare ad usarlo:' followed by a numbered list of four steps: 1. Configura il tuo sito web, 2. Abilita le funzioni aggiuntive, 3. Personalizza il design del tuo sito, and 4. Incomincia ad inserire i contenuti. Each step includes a brief description and links to relevant sections. At the bottom of the main content area, there is a small Drupal logo.

Saltando al momento la parte riguardante la configurazione e la personalizzazione del sito, si può passare direttamente alla pagina '*Crea contenuto*':

Come si può notare, di default esistono due tipi di contenuto già attivati, ma è possibile aggiungerne altri creati ad hoc. L'installazione standard in ogni modo è fornita con una serie di moduli disattivati che consentono di inserire altri tipi di contenuto:

- Argomento di discussione del forum: attivato dal modulo Forum, rappresenta l'inizio di un thread di discussione all'interno del forum.
- Pagina Book: contenuto che viene organizzato e gestito all'interno di una visione più ampia che prende il nome di Book. Creando del contenuto che andrà a far parte di un book, verrà automaticamente visualizzata una navigazione che permette di accedere alle pagine correlate.
- Messaggio del blog: questo particolare contenuto consente di creare un blog all'interno di Drupal.
- Sondaggio: questo contenuto rappresenta una domanda che viene posta ai visitatori, i quali voteranno una risposta predefinita. È possibile impostare la durata del sondaggio ed il suo stato.

Naturalmente non esistono limiti per la creazione di nuovi tipi di contenuto, ed installando moduli di terze parti reperibili nel sito ufficiale o in altri repository è possibile espandere le possibilità che vengono fornite.

Date quindi le enormi possibilità offerte da Drupal, e che sarebbe impossibile in poche pagine esporle tutte, ci si soffermerà solo sulla creazione di una pagina, il cui modello è comunque la base fondamentale per lo sviluppo di un semplice sito.

La Pagina è adatta per creare contenuto statico, che è cioè sempre valido e che non cambia quasi mai, a differenza della Storia che è adatta per contenuti che sono legati, ad esempio, alla data di pubblicazione, come le news ed i comunicati stampa.

Come è possibile notare dalla figura qui sotto, oltre al titolo da dare alla pagina, salta subito all'occhio il blocco centrale chiamato Corpo:

Home > Crea contenuto

Crea Pagina

Titolo: *

—▷ **Impostazioni del menù**

Corpo: Dividi il sommario nella posizione del cursore

—▷ **Formato input**

—▷ **Informazione sulla revisione**

—▷ **Impostazioni commento**

—▷ **Informazioni sull'autore**

—▷ **Opzioni di pubblicazione**

Dentro il corpo è possibile inserire direttamente il testo desiderato, che sarà visualizzato con lo stile di default del tema, ma un aspetto interessante di Drupal è la possibilità di inserire i contenuti, oltre che scrivendo del semplice testo, utilizzando sia codice html che codice php. È possibile ad esempio scrivere il codice con un editor esterno e poi importarlo tramite un copia e incolla.

All'amministratore del sistema, ma anche agli utenti registrati con i dovuti permessi, viene data possibilità di inserire i testi secondo i seguenti 3 formati predefiniti:

- ✓ **Filtered HTML:** una versione ridotta dell'html, dove solamente alcuni tag dell'html saranno consentiti, mentre tutti gli altri tag verranno filtrati, eliminati cioè dall'output finale. I tag html permessi di default sono: <a> <cite> <code> <dl> <dt> <dd>. In più i paragrafi verranno automaticamente convertiti in tag html di tipo
.

- ✓ **PHP code:** permette di inserire all'interno della pagina del codice php. Il codice php deve essere incluso all'interno dei tag '<?php', per iniziare il blocco di codice php e '?>' per chiudere il blocco di codice php.
- ✓ **Full HTML:** una versione completa del codice html, dove tutti i tag sono consentiti (quindi anche ad esempio le tabelle). Anche in questo caso i paragrafi verranno automaticamente convertiti in tag html di tipo '
'. Si possono anche utilizzare tag html abbinati al foglio di stile principale per adottare una formattazione differente da quella classica.

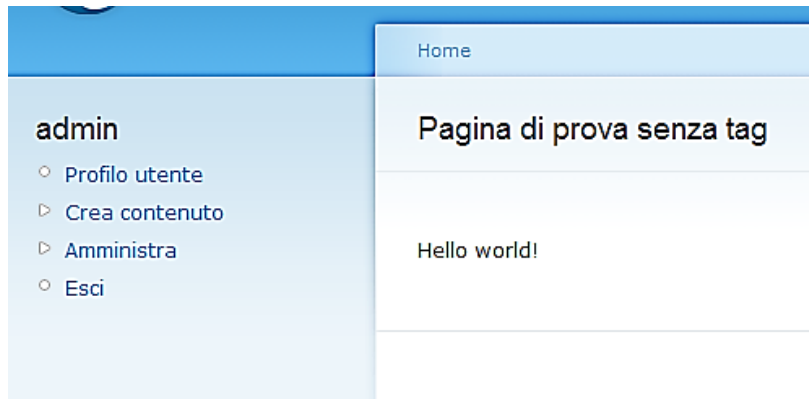


Come si può intuire da questa schermata, l'amministratore del sistema, in un'area apposita, può configurare quali tag rendere disponibili agli utenti o creare nuovi tipi di filtri. È consigliato abilitare gli utenti registrati all'utilizzo di un filtro che consenta un inserimento di tutti i tag HTML, ma non è prudente consentire l'esecuzione di codice php, poiché utenti esperti potrebbero tentare di violare la sicurezza del sistema.

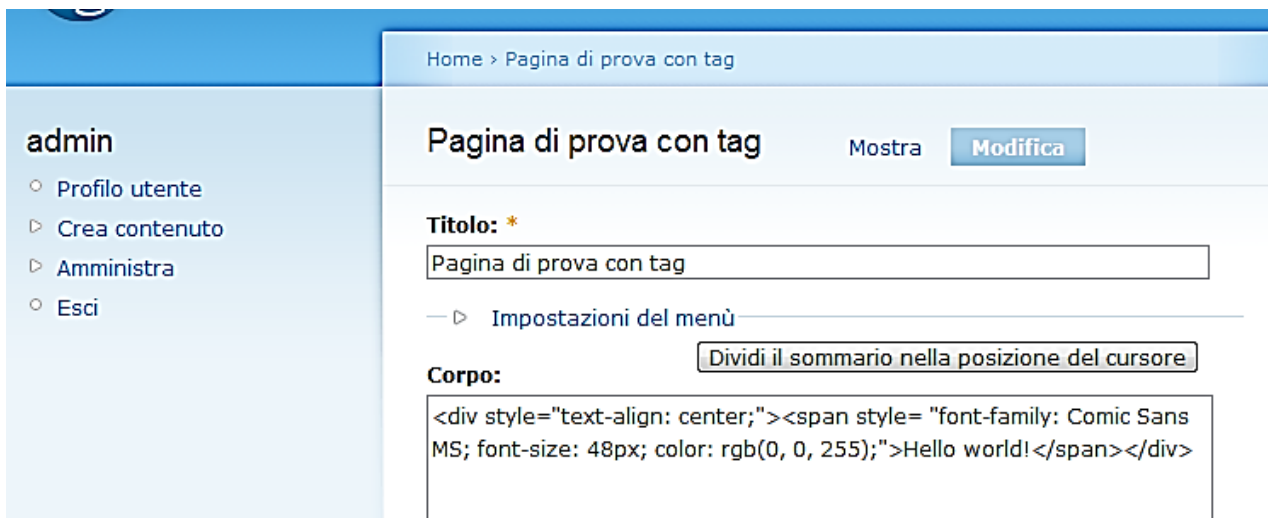
Nella sequenza sottostante si può verificare quello finora scritto sulla composizione di una pagina e l'inserimento del contenuto con o senza tag html:



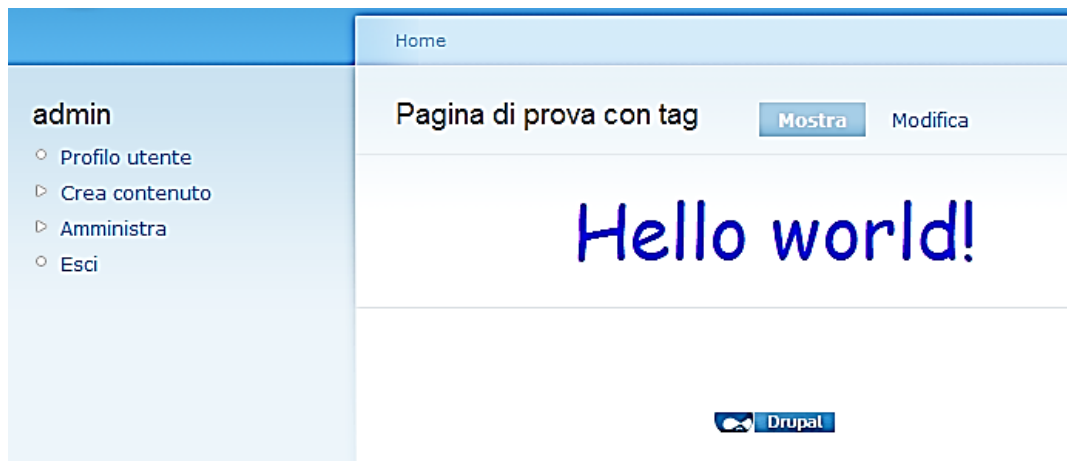
Creazione pagina senza l'uso di tag html...



... e risultato.



Creazione pagina con l'uso di tag html...



...e risultato.

L'immagine qui sotto invece esemplifica questa grande capacità e flessibilità di Drupal di creare pagine personalizzate di una certa complessità. Ovviamente gli amministratori o gli utenti più esperti avranno uno strumento in mano particolarmente potente. Si può notare, infatti, come sia possibile inserire direttamente nella pagina anche script più o meno complessi, oltre ai normali tag per lo stile del contenuto.

Corpo:

```

<h1 style="margin-top: 0px; margin-right: 0px; margin-bottom: 0px; margin-left: 0px; padding-top: 0px; padding-right: 0px; padding-bottom: 0px; padding-left: 0px; font-weight: normal; font-family: 'Comic Sans MS'; font-size: 24px; text-align: center; "><span style="font-family: 'Comic Sans MS'; ">Pagina in allestimento</span></h1>
<div id="dockContainer2">
<div>
<div>
<div id="dock2">&nbsp;</div>
</div>
</div>
</div>
<script type="text/javascript">

var dock = new MacStyleDock(
document.getElementById('dock2'),
[
{
name : '/drupal/images/hteledidattico',
extension : '.png',
sizes : ['', ''],
onclick : function(){
window.location = '/drupal/node/1';
}
}
],
60,
120,
1);

</script>

```

Una volta terminata la compilazione del corpo della pagina è possibile vederne subito un'anteprima, altrimenti si prosegue con le impostazioni successive che consistono nel rendere o meno visibile il contenuto all'utente finale, di promuoverlo o meno in prima pagina, di creare una nuova versione, creare cioè un duplicato del contenuto che poi potrà essere corretto e modificato, mantenendo uno storico delle modifiche apportate, prima di essere visualizzato, e infine di abilitare o disabilitare la lettura o la creazione di commenti del contenuto da parte degli utenti.

Nel campo 'Impostazione menù' si potrà associare il contenuto ad un menù, gestito nell'area 'Amministra > Costruzione del sito > Menù', scegliendo il testo del link corrispondente, la posizione all'interno di questo menù e, opzionalmente, il 'peso' che consente di posizionare le voci più pesanti in basso e le voci più leggere in alto.

Tutti i contenuti creati saranno visualizzabili all'interno dell'area 'Amministra > Gestione dei Contenuti > Contenuto': di ogni contenuto viene visualizzato il titolo, il tipo, l'autore e lo stato.

Contenuto [maggiori informazioni...]

Mostra solo le voci del tipo

tipo è Pagina

e dove stato è

lingua

Opzioni aggiornamento

<input type="checkbox"/>	Titolo	Tipo	Autore	Stato	Lingua	Operazioni
<input type="checkbox"/>	Attività a scelta	Pagina	admin	pubblicato	Italiano	modifica
<input type="checkbox"/>	Telematica A	Pagina	admin	pubblicato	Italiano	modifica
<input type="checkbox"/>	Strumenti per Applicazioni Web	Pagina	admin	pubblicato	Italiano	modifica
<input type="checkbox"/>	Sistemi Elettronici Industriali A	Pagina	admin	pubblicato	Italiano	modifica
<input type="checkbox"/>	Dinamica e Controllo delle Macchine	Pagina	admin	pubblicato	Italiano	modifica
<input type="checkbox"/>	Amministrazione di Reti di Calcolatori	Pagina	admin	pubblicato	Italiano	modifica
<input type="checkbox"/>	Reti di telecomunicazione A	Pagina	admin	pubblicato	Italiano	modifica
<input type="checkbox"/>	Grafica computazionale tecnica	Pagina	admin	pubblicato	Italiano	modifica

Finora sono state indicate solo alcune delle caratteristiche principali del pacchetto Drupal e anche le indicazioni date sull'utilizzo dello stesso sono superficiali rispetto al grande spazio di manovra che in realtà Drupal può offrire. Ci si avvalerà dell'applicazione che si va ad esporre per analizzare più approfonditamente altri aspetti sul funzionamento dei moduli e dei temi di Drupal.

Applicazione

L'applicazione più naturale per l'uso di Drupal, si è compreso pienamente essere la costruzione di un sito internet.

In particolare il progetto è quello di modificare e migliorare vistosamente un sito ambizioso già esistente, nato per dare ausilio agli studenti del corso "teledidattico" di Ingegneria Informatica e Ingegneria Meccanica di Parma: il sito era nato precedentemente in maniera un po' frettolosa e con nozioni appena discreti, da parte dell'amministratore, di codice html. Dopo gli studi conseguiti con la materia 'Strumenti per Applicazioni Web' tenuta dal prof. Calabrese, il sito è stato ampliato e migliorato aggiungendo un po' di codice PHP e javascript per dare la possibilità agli altri studenti di effettuare il login ed avere un minimo di interattività col sistema.

Il sito è ancora provvisoriamente attivo e si trova all'indirizzo

<http://www.pitzalisnet.it/utenti/login.php>.

e la relativa relazione si può ancora trovare qui:

<http://www.pitzalisnet.it/drupal/teledidattico/spaw/Progetto%20teledidattico.pdf>

Senza entrare troppo nei dettagli, il sito, pur essendo molto frequentato dagli studenti del corso, aveva dei livelli di sicurezza piuttosto limitati e anche l'interattività, come l'invio di commenti o l'upload di file, era un po' contorto e poco invitante. Inoltre costringeva l'amministratore a continui controlli diretti e a dover rispondere a continue richieste di accesso al sito.

Il sito è recentemente ricostruito grazie a Drupal ed è attualmente visibile su

<http://www.pitzalisnet.it/drupal>.

Caratteristiche

La trama del progetto iniziale doveva essere sviluppata in modo tale da poter essere facilmente ampliata e implementata anche per estenderne i servizi, col supporto dei docenti e dei tutori, a tutti gli studenti, con particolare attenzione a quegli studenti lavoratori che intendono iscriversi all'Università di Parma malgrado la sospensione della modalità "teledidattica".

Di seguito i punti fondamentali prefissati per la realizzazione del sito:

- deve essere 'essenziale' e funzionale per gli scopi che si prefigge
- deve essere costruito in modo da permettere a qualunque utente una facile ricerca di ciò di cui ha bisogno, pertanto il sito deve contenere
 - ✓ un blocco con un indice sempre a portata di mano
 - ✓ un blocco di ricerca dei contenuti all'interno del sito stesso

- deve consentire facilmente il download di file inseriti dall'amministratore o da utenti registrati
- deve poter registrare i nuovi utenti in modo univoco, semplice e sicuro
- deve dare la possibilità, solo agli utenti registrati, di intervenire attivamente con un proprio contributo
 - ✓ inserendo dei commenti direttamente nella pagina relativa all'argomento selezionato
 - ✓ inserendo, all'interno del commento stesso, uno o più file, come appunti in formato pdf o altro tipo di documentazione inerente, attraverso un upload efficace che verifichi le dimensioni e il formato corretto del file.
- deve abilitare gli utenti registrati a inviarsi messaggi privati, senza necessariamente passare attraverso il proprio client di posta, per uno scambio diretto di informazioni e di dati
- deve dare all'amministratore la possibilità di avere traccia, con estrema semplicità, di ciò che avviene nel sito:
 - ✓ nuove registrazioni
 - ✓ accessi anonimi
 - ✓ accessi autenticati
 - ✓ pagine visitate
 - ✓ commenti aggiunti
 - ✓ upload eseguiti
 - ✓ ecc.

Oltre a ciò, vista la tematica del sito, bisogna dare un'impostazione architettonica tale da dare l'opportunità ad altri utenti, a cui verranno dati particolari privilegi, di aggiornare alcune pagine o costruirne di nuove, per estendere così il servizio anche a studenti del nuovo ordinamento o, perché no, anche di altri corsi.

Per ogni punto dovranno essere seguiti alcuni accorgimenti come un certo ordine e una corretta distribuzione dei dati. Ad esempio, il sistema dovrà creare automaticamente una cartella per ogni utente, entro cui saranno collocati i rispettivi upload e solo essi.

Si è pensato di evitare l'inserimento di blog o forum, non solo perché già abbondantemente presenti in internet, ma perché la modalità di accesso ai commenti per ogni argomento è più che sufficiente.







Infine, i requisiti richiesti all'amministratore e agli utenti sono gli stessi descritti nella parte più generale sul pacchetto Drupal: in pratica sarà comunque sufficiente avere un collegamento internet e saper navigare attraverso un browser...

Costruzione del tema

La prima cosa da decidere è il tipo di layout che si desidera implementare per l'aspetto del sito. Esistono per questo i cosiddetti temi, che identificano tutto ciò che consente di visualizzare i contenuti del sito secondo una rappresentazione grafica de-

finita dal tema stesso. I contenuti del sito sono sempre gli stessi, ma vengono visualizzati e posizionati in modo diverso a seconda del tema scelto.

Come si può vedere dalla schermata sottostante, dopo l'installazione di Drupal, vi sono a disposizione alcuni temi già presenti di cui uno solo, il Garland, è attivato ed è il primo tema con cui generalmente si fanno i conti. È possibile quindi fare una scelta, aiutandosi anche dall'anteprima disponibile.

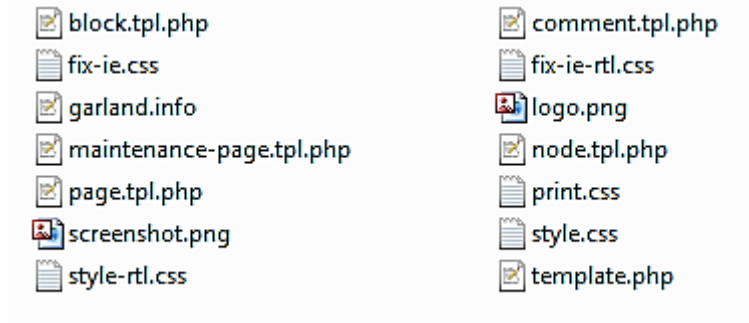
Anteprima	Nome	Versione	Attivato	Predefinito	Operazioni
	Bluemarine Tema realizzato con tabelle con più colonne e uno schema di colori grigio e azzurro.	6.19	<input type="checkbox"/>	<input type="radio"/>	
	Chameleon Tema minimalista con colori tenui realizzato con tabelle.	6.19	<input type="checkbox"/>	<input type="radio"/>	
	Garland Tema senza tabelle, con colonne multiple, a larghezza variabile, con colori modificabili (predefinito).	6.19	<input checked="" type="checkbox"/>	<input checked="" type="radio"/>	configura
	Marvin Tema in scala di grigi con riquadri realizzati con tabelle.	6.19	<input type="checkbox"/>	<input type="radio"/>	
	Minnelli Tema senza tabelle, con colori modificabili, con colonne multiple, a larghezza fissa.	6.19	<input type="checkbox"/>	<input type="radio"/>	
	Pushbutton Tema realizzato con tabelle, azzurro con più colonne e toni arancioni.	6.19	<input type="checkbox"/>	<input type="radio"/>	

Naturalmente non è solo possibile installare temi diversi da questi scaricandoli dalla rete (ve ne sono sia free come a pagamento), ma è anche possibile crearsene uno in proprio, o partendo da zero o da un tema già esistente. Per fare ciò bisogna capire come è strutturato e da cosa è costituito un tema.

Intanto i temi si trovano nella cartella principale, nella directory Themes, e sono costituiti principalmente da :

- uno o più file di stile, il principale dei quali è nominato **style.css**
- un file che definisce lo scheletro del tema: **page.tpl.php**
- alcuni file che definiscono invece la struttura dei contenuti della pagina:
 - per la stampa dell'oggetto nodo: **node.tpl.php**, che definisce il contenuto di ogni singolo nodo
 - per la stampa dell'oggetto blocco: **block.tpl.php**
 - per la stampa dell'oggetto box: **box.tpl.php**, che aggiunge un box attorno ad alcuni elementi

- per la stampa dell'oggetto commento: **comment.tpl.php**
- un file, necessario a seconda del tema scelto, **template.php**, che permette di intervenire più a fondo nel tema con particolari funzioni, aggiungendo, ad esempio, altre regioni.
- uno o più file immagine, come il logo, favicon.ico, e icone varie



Parte del contenuto della cartella /themes/garland

In pratica, ciascun file di tipo .tpl.php contiene un insieme di codice HTML, PHP e CSS necessario a definire rispettivamente la struttura, il contenuto e l'aspetto di ciascuna porzione della pagina. Il ruolo e la gerarchia dei templates più utilizzati sono visibili nella figura qui sotto:

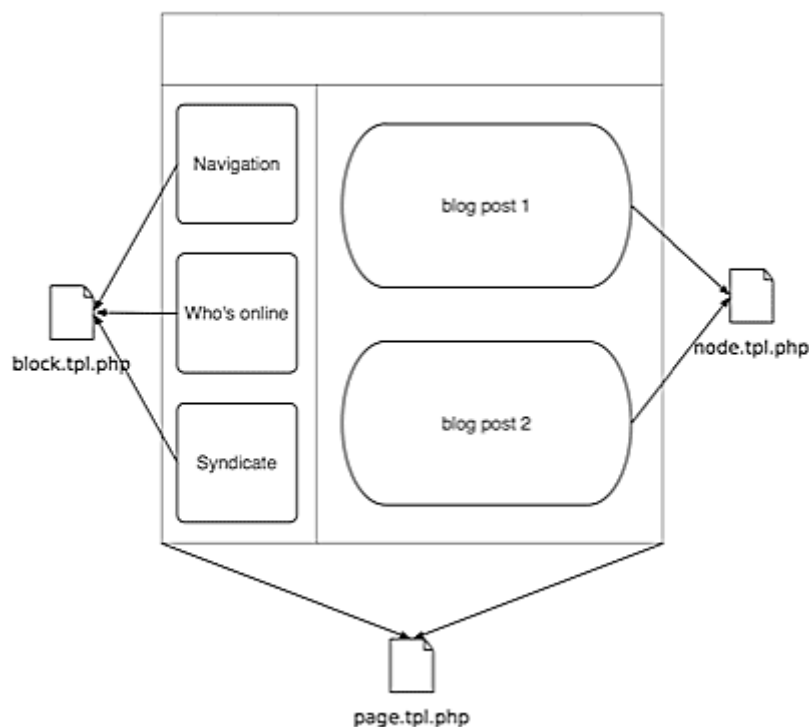
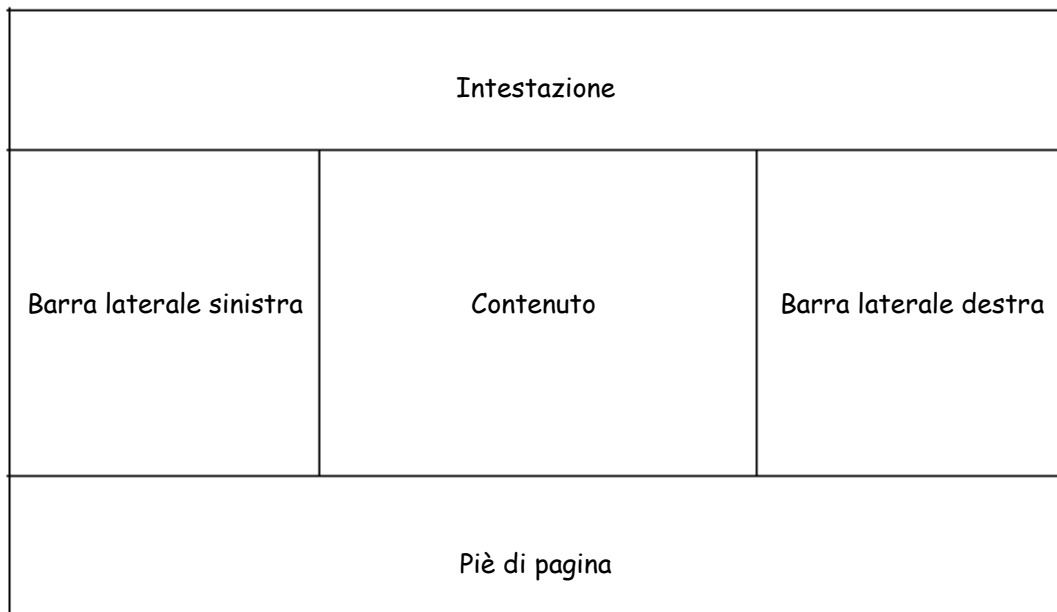


Immagine tratta dal testo "Pro Drupal Development"
di John K. VanDyk e Matt Westgate

Ognuno di questi file .tpl.php, detti appunto templates, chiama determinate variabili per mostrarne i relativi contenuti e informazioni, come il titolo della pagina, la data, il contenuto principale, l'utente che l'ha creato, ecc.

Analizzare un modello di base per ciascun template renderebbe questa valutazione troppo estesa, ma per capire almeno in parte il funzionamento del sistema, e quindi le successive modifiche eseguite per l'applicazione, si sceglie allora di dare qualche informazione più approfondita solo sul template principale.

Più in dettaglio, come si può notare qui sotto, il file **page.tpl.php** è suddiviso in 5 parti, ogni parte equivale ad una zona della pagina: barra laterale sinistra, barra laterale destra, contenuto, intestazione, piè di pagina.



Di seguito vengono riportati il significato e la funzione di alcune fra le principali variabili del template `page.tpl.php`, nell'ordine in cui compaiono nel codice. Si tenga conto che si parla sempre di un template facente parte del tema 'garland' e che in altri temi lo stesso file può essere molto diverso, pur rimanendone simile la struttura

\$language - La lingua con cui si sta mostrando il sito.

\$head - codice html generato dalla funzione `drupal_get_html_head()`.

\$head_title - Il testo mostrato nell'header (intestazione) della pagina.

\$styles - Necessario per far funzionare i fogli di stile: restituisce i file .css richiesti dalla pagina.

\$scripts - html per caricare i files JavaScript e i settings JS. I files javascript devono essere inseriti manualmente nel template della pagina (`page.tpl.php`).

\$left - L'html per la barra laterale sinistra...

\$right - e per la barra laterale destra.

\$site_name - Mostra il nome del sito, a meno che non sia disabilitato nei settings del tema (admin).

\$site_slogan - Lo slogan del sito, a meno che non sia disabilitato nei settaggio del tema da parte dell'amministratore.

\$logo - Il path del logo, come definito nella configurazione del tema da parte dell'amministrazione.

\$primary_links - Contiene i links di navigazione primari.

\$secondary_links - Contiene i links di navigazione secondari.

\$search_box - Mostra il campo di ricerca se il box di ricerca è abilitato nella configurazione del tema da parte dell'amministratore.

\$mission - Il testo della 'mission' (la finalità) del sito, che viene mostrato se la mission è abilitata nel settaggio del tema.

\$tabs e **\$tabs2** - Codice html per mostrare i tabs primari, generalmente in cima alla pagina, e i tabs secondari

\$title - Il titolo della pagina (diverso dal titolo dell'intestazione, `head_title`), generalmente il titolo del nodo o della sezione che viene mostrato sopra al contenuto.

\$show_messages e **\$messages** - Codice html per i messaggi di stato e di errore che vengono mostrati in cima al contenuto della pagina.

\$help - Testo di help dinamico, utilizzato nelle pagine di amministrazione.

\$content - Il contenuto principale di una pagina, in codice html, generato da Drupal.

\$feed_icons - Una stringa con tutte le icone degli RSS feeds disponibili per la pagina corrente.

\$footer_message - Il messaggio del 'footer' (piè di pagina) definito durante la fase di settaggio nelle pagine di amministrazione.

\$closure - Deve essere inserita in fondo pagina per chiamare gli eventuali javascript dinamici, al termine del caricamento della pagina stessa.

Più sotto si trova il codice completo del file di default `page.tpl.php` del tema garland: una volta interpretato il significato di ogni variabile, la disposizione del loro contenuto all'interno della pagina e individuati gli id e le classi di cui fanno parte, è possibile, con estrema facilità, creare il layout desiderato modificando o ricreando il file di stile `style.css` che è quello più generale della pagina, ma anche quelli relativi ai vari blocchi e moduli che andranno a costituire la pagina stessa, come `node.css` all'interno del modulo nodo, `system-menus.css` all'interno del modulo di sistema per i menù, ecc.

H:\pitzalisnet.it\drupal\themes\garland\page.tpl.php

```

<?php
// $Id: page.tpl.php,v 1.18.2.1 2009/04/30 00:13:31 goba Exp $
?><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="<?php print $language->language ?>"
" lang="<?php print $language->language ?>" dir="<?php print $language->dir ?>">
  <head>
    <?php print $head ?>
    <title><?php print $head_title ?></title>
    <?php print $styles ?>
    <?php print $scripts ?>
    <!--[if lt IE 7]>
      <?php print phptemplate_get_ie_styles(); ?>
    <![endif]-->
  </head>
  <body<?php print phptemplate_body_class($left, $right); ?>>

<!-- Layout -->
  <div id="header-region" class="clear-block"><?php print $header; ?></div>

  <div id="wrapper">
    <div id="container" class="clear-block">

      <div id="header">
        <div id="logo-floater">
          <?php
            // Prepare header
            $site_fields = array();
            if ($site_name) {
              $site_fields[] = check_plain($site_name);
            }
            if ($site_slogan) {
              $site_fields[] = check_plain($site_slogan);
            }
            $site_title = implode(' ', $site_fields);
            if ($site_fields) {
              $site_fields[0] = '<span>'. $site_fields[0] . '</span>';
            }
            $site_html = implode(' ', $site_fields);

            if ($logo || $site_title) {
              print '<h1><a href="'. check_url($front_page) .' " title="'. $site_title .
'">';
              if ($logo) {
                print '';
              }
              print $site_html . '</a></h1>';
            }
          ?>
        </div>

        <?php if (isset($primary_links)) : ?>
          <?php print theme('links', $primary_links, array('class' => 'links
primary-links')) ?>

```

```
H:\pitzalisnet.it\drupal\themes\garland\page.tpl.php
    <?php endif; ?>
    <?php if (isset($secondary_links)) : ?>
        <?php print theme('links', $secondary_links, array('class' => 'links
secondary-links')) ?>
    <?php endif; ?>

</div> <!-- /header -->

<?php if ($left): ?>
    <div id="sidebar-left" class="sidebar">
        <?php if ($search_box): ?><div class="block block-theme"><?php print
$search_box ?></div><?php endif; ?>
        <?php print $left ?>
    </div>
<?php endif; ?>

<div id="center"><div id="squeeze"><div class="right-corner"><div class=
"left-corner">
    <?php print $breadcrumb; ?>
    <?php if ($mission): print '<div id="mission">'. $mission .'</div>'; endif; ?>
    <?php if ($tabs): print '<div id="tabs-wrapper" class="clear-block">'; endif;
?>
    <?php if ($title): print '<h2'. ($tabs ? ' class="with-tabs" : '') .'>'.
$title .'</h2>'; endif; ?>
    <?php if ($tabs): print '<ul class="tabs primary">'. $tabs .'</ul></div>';
endif; ?>
    <?php if ($tabs2): print '<ul class="tabs secondary">'. $tabs2 .'</ul>';
endif; ?>
    <?php if ($show_messages && $messages): print $messages; endif; ?>
    <?php print $help; ?>
    <div class="clear-block">
        <?php print $content ?>
    </div>
    <?php print $feed_icons ?>
    <div id="footer"><?php print $footer_message . $footer ?></div>
</div></div></div></div> <!-- /.left-corner, /.right-corner, /#squeeze, /#center
-->

<?php if ($right): ?>
    <div id="sidebar-right" class="sidebar">
        <?php if (!$left && $search_box): ?><div class="block block-theme"><?php
print $search_box ?></div><?php endif; ?>
        <?php print $right ?>
    </div>
<?php endif; ?>

</div> <!-- /container -->
</div>
<!-- /layout -->

<?php print $closure ?>
</body>
</html>
```

Modello base del file page.tpl.php nel tema 'garland'

Alcune delle funzioni che fanno parte di questo codice, sono implementate nel seguente file template.php, che, insieme al page.tpl.php, dà una visione abbastanza completa di come agisce il sistema per la pubblicazione di una pagina.

```

<?php
// $Id: template.php,v 1.16.2.3 2010/05/11 09:41:22 goba Exp $

/**
 * Sets the body-tag class attribute.
 *
 * Adds 'sidebar-left', 'sidebar-right' or 'sidebars' classes as needed.
 */
function phptemplate_body_class($left, $right) {
  if ($left != '' && $right != '') {
    $class = 'sidebars';
  }
  else {
    if ($left != '') {
      $class = 'sidebar-left';
    }
    if ($right != '') {
      $class = 'sidebar-right';
    }
  }

  if (isset($class)) {
    print ' class="'. $class .'';
  }
}

/**
 * Return a themed breadcrumb trail.
 *
 * @param $breadcrumb
 *   An array containing the breadcrumb links.
 * @return a string containing the breadcrumb output.
 */
function phptemplate_breadcrumb($breadcrumb) {
  if (!empty($breadcrumb)) {
    return '<div class="breadcrumb">'. implode(' > ', $breadcrumb) . '</div>';
  }
}

/**
 * Override or insert PHPTemplate variables into the templates.
 */
function phptemplate_preprocess_page(&$vars) {
  $vars['tabs2'] = menu_secondary_local_tasks();

  // Hook into color.module
  if (module_exists('color')) {
    _color_page_alter($vars);
  }
}

/**
 * Add a "Comments" heading above comments except on forum pages.
 */
function garland_preprocess_comment_wrapper(&$vars) {
  if ($vars['content'] && $vars['node']->type != 'forum') {
    $vars['content'] = '<h2 class="comments">'. t('Comments') . '</h2>'. $vars['content'];
  }
}

```

```

}
}

/**
 * Returns the rendered local tasks. The default implementation renders
 * them as tabs. Overridden to split the secondary tasks.
 *
 * @ingroup themeable
 */
function phptemplate_menu_local_tasks() {
  return menu_primary_local_tasks();
}

/**
 * Returns the themed submitted-by string for the comment.
 */
function phptemplate_comment_submitted($comment) {
  return t('!datetime - !username',
    array(
      'username' => theme('username', $comment),
      '!datetime' => format_date($comment->timestamp)
    ));
}

/**
 * Returns the themed submitted-by string for the node.
 */
function phptemplate_node_submitted($node) {
  return t('!datetime - !username',
    array(
      'username' => theme('username', $node),
      '!datetime' => format_date($node->created),
    ));
}

/**
 * Generates IE CSS links for LTR and RTL languages.
 */
function phptemplate_get_ie_styles() {
  global $language;

  $iecss = '<link type="text/css" rel="stylesheet" media="all" href="'. base_path() .
  path_to_theme() .'/fix-ie.css" />';
  if ($language->direction == LANGUAGE_RTL) {
    $iecss .= '<style type="text/css" media="all">@import "'. base_path() . path_to_theme() .
    '/fix-ie-rtl.css";</style>';
  }

  return $iecss;
}

```

Il file template.php nel tema 'garland'

Riassumendo, una volta capiti i modi di agire del codice di ogni singolo file, sarà possibile intervenire su di esso per modificare la struttura di base o l'ordine degli elementi, per inserire o eliminare singole variabili php viste sopra e, ancora, per mostrare o nascondere i contenuti che più interessano. Infine si potrà modificare il foglio di stile style.css in un editor di testo intervenendo sugli identificatori, le classi, le sottoclassi, ecc.

Dopo questo lungo preambolo, necessario per comprendere meglio il lavoro svolto nella realizzazione dell'applicazione, si tenterà di spiegare in pochi punti in che modo è stato costruito e personalizzato il sito.

Intanto, anziché realizzare un tema da zero, è stato scelto, per praticità, di utilizzare un tema già esistente, il 'garland' stesso. Si è proceduto quindi a farne una copia, rinominare quest'ultima 'garlan-custom' e posizionarla, tramite ftp, nella cartella di installazione di Drupal, in sites/all/themes.

Ritornando alla pagina di amministrazione di Drupal è stato quindi possibile selezionare questo nuovo tema che è venuto a essere la base per immediate modifiche. L'intervento successivo, perciò, è stato quello di ricostruire il file di stile css: l'idea era di dare un aspetto non troppo differente al precedente sito, e di rendere i blocchi 'trasparenti', togliendo, cioè, gli sfondi che contraddistinguono tipicamente i blocchi dei vari temi di Drupal, e inserendo un unico sfondo di background, identico alla precedente versione del sito. Stessa cosa si può dire per il font utilizzato. La sostituzione, poi, del logo (che è un gif animato) e del titolo del sito sono risultati altrettanto elementari.

```
1  /* $Id: style.css,v 1.38.2.4 2009/09/14 13:10:47 goba Exp $ */
2
3  /**
4   * Garland, for Drupal 6.x
5   * Stefan Nagtegaal, iStyledThis [dot] nl
6   * Steven Wittens, acko [dot] net`
7   *
8   * If you use a customized color scheme, you must regenerate it after
9   * modifying this file.
10  */
11
12  /**
13   * Generic elements
14  */
15  body {
16    margin: 0;
17    padding: 0;
18    background: #edf5fa;
19    font: 12px/170% Verdana, sans-serif;
20    color: #494949;
21  }
22
23  input {
24    font: 12px/100% Verdana, sans-serif;
25    color: #494949;
26  }
27
28  textarea, select {
29    font: 12px/160% Verdana, sans-serif;
30    color: #494949;
31  }
32
33  h1, h2, h3, h4, h5, h6 {
34    margin: 0;
35    padding: 0;
36    font-weight: normal;
37    font-family: Helvetica, Arial, sans-serif;
38  }
39
40  h1 {
41    font-size: 170%;
42  }
```

Prime righe del file style.css originale

```
11  /**
12   * Modified by Claudio Pitzalis
13   * Garland_custom, for www.pitzalisnet.it/drupal
14   * August, 2010
15   */
16
17  /**
18   * Generic elements
19   */
20  body {
21     margin: 0;
22     padding: 0;
23
24     font: 14px/170% Comic Sans MS, Verdana, sans-serif;
25     color: #191919;
26     background-image: url(/drupal/images/sfondi-mac.jpg);
27     background-attachment: fixed;
28  }
29
30  input {
31     font: 14px/100% Comic Sans MS, Verdana, sans-serif;
32     color: #494949;
33  }
34
35  textarea, select {
36     font: 14px/160% Comic Sans MS, Verdana, sans-serif;
37     color: #494949;
38  }
39
40  h1, h2, h3, h4, h5, h6 {
41     margin: 0;
42     padding: 0;
43     font-weight: normal;
44     font-family: Comic Sans MS, Helvetica, Arial, sans-serif;
45  }
46
47  h1 {
48     font-size: 170%;
49  }
50
51  h2 {
52     font-size: 160%;
53     line-height: 130%;
54  }
55
56  h3 {
57     font-size: 140%;
58  }
```

Parte delle modifiche effettuate

Come già scritto nella prima parte, Drupal offre strumenti potenti, ma contemporaneamente semplici, per la definizione dei vari contenuti che si vogliono di volta in volta inserire, e dal momento che la ricostruzione da zero del precedente sito avrebbe comportato un lavoro non indifferente, sono stati utilizzati questi strumenti per trasferire le pagine da un sito all'altro, sia pure armati di pazienza, ma con relativa semplicità. Si è trattato in sostanza, durante la creazione delle pagine, come visto a pa-

gina 18 e successive, di inserire di volta in volta, all'interno della sezione 'corpo', il codice html, compreso tra i tag <body> e </body>, così com'era, delle pagine del sito pre-esistente: a parte qualche piccola variazione al codice per il ripristino, ad esempio, dei link, ogni pagina prendeva subito la forma e l'aspetto desiderati.

Il contenuto di ogni pagina veniva automaticamente inserito in un 'nodo': infatti nella parte degli indirizzi del browser era possibile notare indirizzi come drupal/node/1, .../node/2, ecc. che non avevano di fatto nessuna corrispondenza concreta con file e cartelle presenti nel server, visualizzabili tramite ftp. Di fatto neppure la cartella drupal/node è presente nel server: questa è una conferma di come il server crei via via le pagine inserendo i dati nel database, e da lì recuperarli per visualizzarli grazie al codice php.

Con l'interfaccia di amministrazione è stato particolarmente semplice definire un menù a cascata che nascondesse o meno link o sottomenù: così l'indice delle materie è stato suddiviso in anni di corso, mostrando, di ogni anno, le relative materie di studio.



Scelta del menù di navigazione

Tutto ciò che nel layout non è possibile modificare attraverso l'interfaccia offerta da Drupal, lo sarà invece intervenendo direttamente sul codice dei fogli di stile CSS.

Oltre al corpo delle pagine è possibile inserire blocchi precostituiti o confezionati in proprio, sia ai lati come nell'intestazione e a piè pagina. Ma un blocco può esso stesso costituire il contenuto di una pagina: ciò è molto utile per inserire contenuti solo in pagine specificate dall'amministratore o solo a utenti registrati.

Si riporta a titolo esemplificativo il procedimento della costruzione di un blocco e del relativo posizionamento in una pagina scelta.

Come sotto-applicazione, si è pensato di inserire, solo a lato della pagina introduttiva, un piccolo blocco di testo scorrevole con alcune informazioni. Il blocco è stato pensato per consentire una facile modifica del testo da parte dell'amministratore.

La procedura eseguita è stata quella di andare nella pagina *Amministra > Blocchi > Aggiungi blocco*:



Pagina per la creazione di un blocco

Come si può notare l'impostazione è del tutto simile a quella per la costruzione di una pagina. Il nome dato al blocco è semplicemente 'Testo scorrevole'. Impostato quindi il formato di input come 'full html', si è inserito il seguente codice:

```

1 <script type="text/javascript" src="/drupal/sites/all/themes/garland_custom/js/scroll.js"
  ></script>
2 <div id="marqueecontainer" onMouseOver="copyspeed=pausespeed" onMouseOut=
  "copyspeed=marqueespeed">
3 <div id="vmarquee" style="position:absolute"><!--
4 *****
5 * Di seguito va inserito il testo *
6 *****
7 -->
8 <p style="text-align: center; ">In questa nuova versione del sito, ricostruito completamente
  col CMS drupal, l'utente registrato potrà intervenire in maniera divertente grazie a un
  text editor ricco di funzioni dato dal modulo FCKeditor.<br />
9 Inoltre potrà inviare messaggi privati direttamente ad altri utenti senza
  necessariamente passare dal proprio servizio e-mail.<br />
10 Prossimamente verranno implementate altre funzioni interessanti.</p>
11 </div>
12 </div>

```

È possibile naturalmente inserire anche delle immagini, tenendo conto però dei limiti imposti dai margini del blocco.

Nel semplice codice viene chiamato uno script preventivamente inserito in una cartella apposita e, a seguire, è possibile immettere il testo desiderato. Lo script permette lo stop dello scorrimento del testo quando il mouse è posizionato su di esso. Tutto questo per evitare il tag non standardizzato `<marquee>`.

È stata quindi impostata la visibilità del blocco nella pagina (nodo) desiderata.

▼ Impostazioni specifiche di visibilità per pagina

Mostra il blocco su pagine specifiche:

- Mostra su ogni pagina eccetto quelle elencate.
- Mostra solo sulle pagine elencate.
- Mostra se il seguente codice PHP restituisce TRUE (modalità PHP, solo esperti).

Pagine:

```
<front>
node/19
```

Inserisci una pagina per linea come percorsi di Drupal. Il carattere '*' è un jolly. Percorsi di esempio sono *blog* per la pagina blog e *blog-jolly* per ogni blog personale. *<front>* è la prima pagina (front page). Se viene scelta la modalità PHP, inserisci codice PHP tra `<?php ?>`. Tieni presente che eseguire codice PHP non corretto può danneggiare il tuo sito Drupal.

The ID for **excluding or including** this element is `admin/build/block/configure/block/2.edit-pages`.

Impostazioni di visibilità

Il blocco sarà visibile solo nella pagina introduttiva del sito.

Ed infine, è stata impostata la posizione del blocco appena creato:

Blocco	Settore	Operazioni
Barra laterale sinistra		
+ Navigazione	Barra laterale sinistra ▼	configura
+ Testo scorrevole.	Barra laterale sinistra ▼	configura elimina
+ Accesso utente	Barra laterale sinistra ▼	configura
+ Slide show	Barra laterale sinistra ▼	configura
Barra laterale destra		
+ Chi è on-line	Barra laterale destra ▼	configura

Per ciò che concerne altri aspetti fondamentali del sito, Drupal offre all'amministratore un'ottima interfaccia per la registrazione degli utenti, permettendo vari livelli di automazione. È quindi possibile effettuare una registrazione in modo da ricevere, tramite e-mail, le istruzioni inserite dall'amministratore per il completamento della stessa.

L'utente registrato avrà automaticamente alcuni diritti aggiuntivi rispetto ad un utente anonimo: potrà intervenire attivamente ai commenti, potrà effettuare l'upload, potrà accedere alla lista utenti per inviare messaggi e, inoltre, potrà accedere ad alcuni moduli aggiuntivi inseriti nel portale.

Molte di queste attività sono possibili grazie a moduli standard che offre Drupal, ma come già descritto in precedenza, questi moduli possono essere ulteriormente implementati o se ne possono aggiungere di nuovi in base alle necessità del sito.

Costruzione di un modulo di prova

Prima di entrare nello specifico di particolari moduli installati, è bene capire come siano in generale sviluppati e quale sia la loro tipologia.

Drupal trova i moduli da installare nella directory **/modules**, che contiene i moduli di sistema, e nella sottodirectory **/sites/all/modules** che contiene i moduli aggiuntivi.

I file contenuti nella cartella di ogni modulo sono principalmente 4:

- **nomemodulo.info** che contiene informazioni sul modulo
- **nomemodulo.install** che contiene codice per la installazione del modulo
- **nomemodulo.module** che contiene codice php
- **nomemodulo.css** che contiene gli stili css del modulo (non sempre presente)

I moduli si dividono in due tipologie: **Block** e **Node**.

I moduli **block** sono moduli che generalmente vengono posizionati sulle barre laterali. Drupal chiama la funzione `nomemodulo_block($op=='view')` con opzione uguale a `view` per ottenere il contenuto del blocco. In particolare Drupal si aspetta un array `$block` di ritorno con i seguenti valori impostati:

```
$block['subject'] = t('nomemodulo');
```

```
$block['content'] = contenuto del blocco;
```

Il moduli **node** generano pagine di contenuto come blog, forum, book pages.

Le funzioni minime che definiscono un modulo sono:

- **nomemodulo_block()** -> inserisce il nome del blocco nella lista blocchi
- **nomemodulo_help()** -> descrizione del blocco all'interno della configurazione blocco
- **nomemodulo_perm()** -> vengono definiti i permessi di accesso al modulo

Si riporta di seguito il codice di un modulo di prova costruito appositamente per testarne il funzionamento: è un esempio minimo di un modulo per Drupal con l'implementazione del blocco, della sezione di help e dei permessi. L'unica operazione compiuta dai blocchi è stampare delle stringhe senza alcuna pretesa di utilità pratica.

Il modulo è stato installato nell'applicazione di questa valutazione e, attivandolo, è possibile testarlo.

```

1 ; $Id$
2 name = Modulo di prova
3 description = "Modulo di prova per la funzionalità blocco"
4 core = 6.x
5 dependencies[] = block
6 dependencies[] = node
7 package = modules
8 version = "6.x-1.0"

```

Il file .info:

contiene le informazioni essenziali del modulo e permette a Drupal di trovarlo

```

1 <?php
2 // $Id$
3
4 /**
5  * Implementazione di hook_install().
6  */
7 function modulo_di_prova_install() {
8     drupal_set_message( t("Il tuo blocco è disponibile sotto !link",
9         array('!link' => l('Amministra > Costruzione del sito > Blocchi ', 'admin/build/block'))
10    ));
11     watchdog("modulo_di_prova","Modulo modulo_di_prova installato");
12 }
13
14 /**
15  * Implementazione di hook_uninstall().
16  */
17 function modulo_di_prova_uninstall() {
18     drupal_set_message( t("Il modulo è stato disinstallato") );
19     watchdog("modulo_di_prova","Modulo test block disinstallato");
20 }
21 ?>

```

Il file .install:

contiene le procedure (hook) per installare/disinstallare/aggiornare una tabella o inizializzare le proprie variabili.

```

<?php
/**
 * Implementazione hook_help()
 */
function modulo_di_prova_help($section, $arg) {
    switch ($section) {
        case 'admin/help#modulo_di_prova':
            return t('Modulo di prova per la creazione di blocchi');
    }
}
/**
 * Implementazione hook_block()
 */
function modulo_di_prova_block($op = 'list', $delta = 0, $edit = array()) {
    switch ($op) {
        case 'list':
            $blocks[0]['info'] = t('Test esempio blocco modulo di prova numero 1');
            $blocks[1]['info'] = t('Test esempio blocco modulo di prova numero 2');
            return $blocks;
        case 'view':
            switch ($delta) {
                case 0:
                    $block['subject'] = t('Titolo del blocco #1');
                    $block['content'] = modulo_di_prova_content(1);
                    break;
                case 1:
                    $block['subject'] = t('Titolo del blocco #2');
                    $block['content'] = modulo_di_prova_content(2);
                    break;
            }
            return $block;
    }
}
}

```

Il file .module:

contiene il codice operativo del modulo, cioè il progetto vero e proprio, con le funzioni sopra descritte.

In questo caso non è stato scomodato il database, ma dentro la funzione _block() è possibile inserire query anche piuttosto complesse.

```
/**
 * Implementazione di hook_perm().
 * Amministra -> Gestione utente -> Permessi
 */
function modulo_di_prova_perm()
{
    return array('administra modulo_di_prova', 'accedi a modulo_di_prova');
}
/**
 * Ritorna il contenuto da visualizzare a seconda del blocco richiesto
 */
function modulo_di_prova_content($which_block) {
    if ($which_block == 1) {
        return t("Primo blocco");
    }
    if ($which_block == 2) {
        return t("Secondo blocco");
    }
}
?>
```

Dopo questa necessaria esposizione sull'implementazione di un modulo, si può tornare a descrivere la struttura del sito e, in particolare, sulla scelta dei moduli aggiuntivi per dare strumenti idonei agli utenti. Ad esempio si è cercata una modalità per rendere l'accesso ai commenti, da parte degli utenti, il più completo possibile, fornendo, con un'unica interfaccia, la possibilità, oltre che lasciare il proprio contributo testuale in maniera divertente e accattivante, di inserire documenti o altro materiale eseguendo l'upload attraverso la medesima interfaccia con pochi semplici passi e in maniera guidata, inoltre anche l'inserimento di link ad altre pagine dovrà essere immediato.

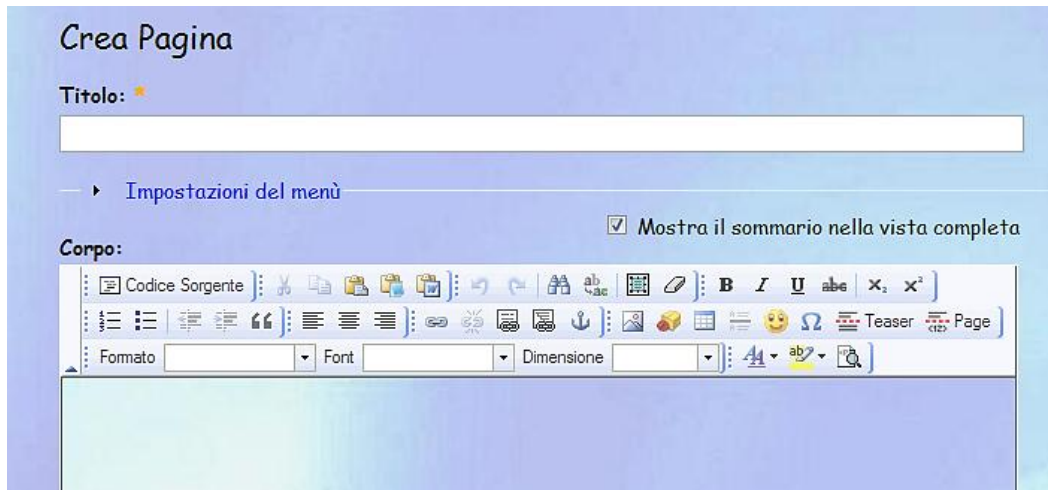
Il modulo fckeditor

Il modulo che ha reso possibili queste funzioni aggiuntive è fckeditor: un modulo che permette di sostituire i campi di testo di Drupal con un editor HTML visuale, a volte chiamato editor WYSIWYG (**What You See Is What You Get**: "quello che vedi è quello che ottieni"). Questo editor di testo HTML porta con sé molte delle potenti funzioni di editing WYSIWYG di editor desktop, noti come Word per il web. È un modulo relativamente leggero e non richiede, da parte del client, alcun tipo di installazione sul proprio computer.

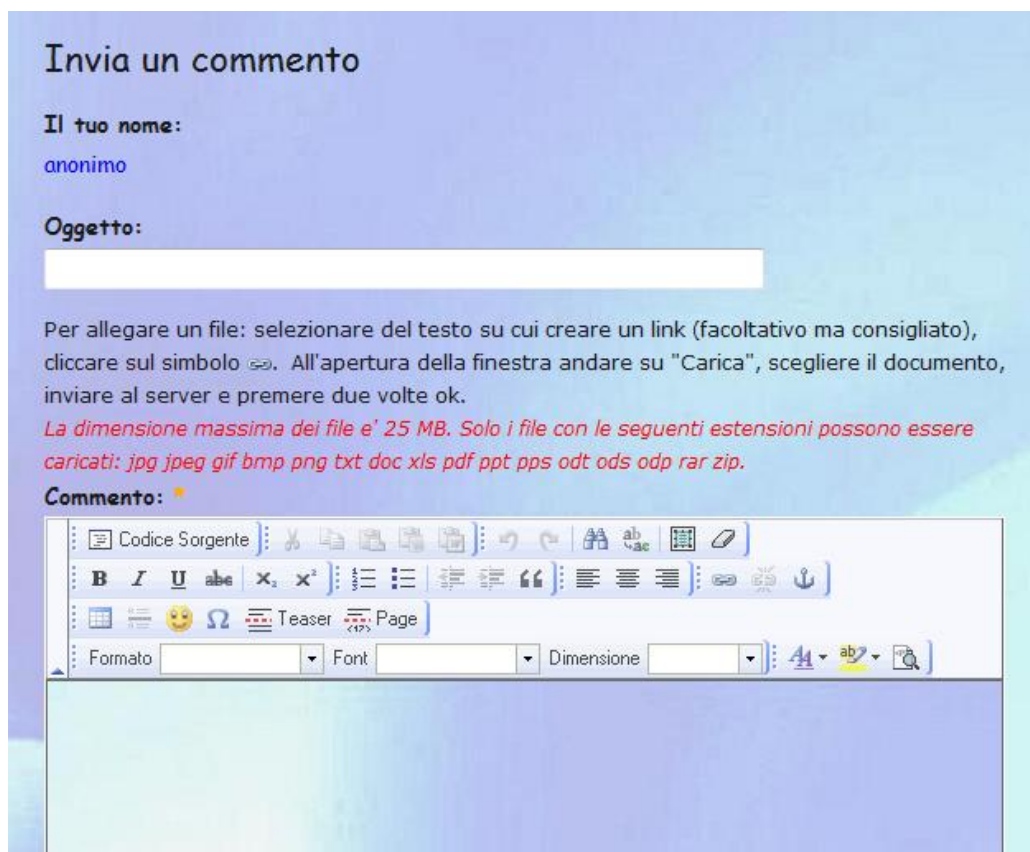
Una volta scaricati liberamente, prima il file fckeditor-6.x-2.1.tar.gz dal sito ufficiale di drupal, e successivamente il file Fckeditor_2.6.6, quest'ultimo, invece, dal sito ckeditor.org, si dovrà scompattarli e inserire la cartella, risultante dal secondo file, nella cartella risultante dal primo: quest'ultima, tramite ftp, andrà quindi inviata alla directory sites/all/modules di drupals. Si tratta di fatto, in breve, di un modulo adottato come 'involucro' per utilizzare un WISIWIG di terze parti, nato per altre applicazioni.

Una volta attivato, con alcune modifiche indispensabili a rendere utilizzabile il modulo, non solo sarà possibile per l'amministratore creare contenuti con maggiore semplicità attraverso un discreto numero di strumenti, senza necessariamente avere conoscenza di codice html, ma anche per l'utente, coi diritti appositi, sarà consentito inserire commenti in maniera piacevole.

Nelle immagini seguenti si possono notare le nuove modalità di inserimento dei contenuti nel corpo della pagina, da parte dell'amministratore, rispetto a come visto precedentemente, e come gli utenti registrati siano in possesso di svariati strumenti per dare il loro contributo.



Nuovo aspetto, per l'amministratore, dell'interfaccia per la creazione di una pagina, grazie al modulo fckeditor



Gli strumenti messi a disposizione agli utenti registrati

Nell'impostare il modulo, è stata fatta attenzione a dove i file dovevano essere caricati nel server: nel momento in cui carica un file nel server, a ogni utente è automaticamente destinata una cartella, dove verranno, d'ora in poi, depositati i relativi upload, col nome dell'utente stesso, all'interno della directory site/files. Questo, tra l'altro, consente all'amministratore di tenere un certo ordine e di sapere 'chi' fa 'cosa' per intervenire celermente in caso di abusi.

All'interno del modulo è presente uno javascript dove sono stati definiti gli strumenti per l'amministratore e quelli per l'utente registrato:

```

27  This toolbar is dedicated to users with "Full HTML" access
28  some of commands used here (like 'FontName') use inline styles,
29  which unfortunately are stripped by "Filtered HTML" filter
30  */
31  FCKConfig.ToolbarSets['DrupalFull'] = [
32    ['Source'],
33    ['Cut','Copy','Paste','PasteText','PasteWord'],
34    ['Undo','Redo','-','Find','Replace','-','SelectAll','RemoveFormat'],
35    ['Bold','Italic','Underline','StrikeThrough','-','Subscript','Superscript'],
36    ['OrderedList','UnorderedList','-','Outdent','Indent','Blockquote'],
37    ['JustifyLeft','JustifyCenter','JustifyRight'],
38    ['Link','Unlink','LinkToNode','LinkToMenu','Anchor'], /* */
39    ['Image','Flash','Table','Rule','Smiley','SpecialChar','DrupalBreak','DrupalPageBreak'],/**/
40    '/',
41    ['FontFormat','FontName','FontSize'],
42    ['TextColor','BGColor','ShowBlocks']
43  ] ;
44  FCKConfig.ToolbarSets['Autenticated_User'] = [
45    ['Source'],
46    ['Cut','Copy','Paste','PasteText','PasteWord'],
47    ['Undo','Redo','-','Find','Replace','-','SelectAll','RemoveFormat'],
48    ['Bold','Italic','Underline','StrikeThrough','-','Subscript','Superscript'],
49    ['OrderedList','UnorderedList','-','Outdent','Indent','Blockquote'],
50    ['JustifyLeft','JustifyCenter','JustifyRight'],
51    ['Link','Unlink','Anchor'],
52    ['Table','Rule','Smiley','SpecialChar','DrupalBreak','DrupalPageBreak'],/**/
53    '/',
54    ['FontFormat','FontName','FontSize'],
55    ['TextColor','BGColor','ShowBlocks']
56  ] ;

```

Sezione dello javascript fckeditor.congig.js

Altri piccoli interventi sono stati fatti al file comment.module all'interno del modulo comment per potersi meglio interfacciare con fckeditor.

Naturalmente sarà sempre possibile scegliere di utilizzare un codice sorgente come in precedenza.

Un'ultima interessante funzione trasversale svolta, è quella, una volta inseriti i contenuti, di poterne vedere immediatamente il codice sorgente: un modo per imparare le basi del codice html.

Altri moduli

Si fa ora solo una breve menzione di alcuni altri moduli attivati:

- Il modulo **path**, che ha consentito di rinominare gli URL con nomi espressivi. Si tratta di un SEO (Search Engine Optimization) già presente nell'installazione stan-

dard di drupal, molto efficace e di facile utilizzo. Per rendere funzionante il modulo, si sono attivati gli URL Semplificati dal percorso amministra > configurazione del sito > URL semplificati. L'utilizzo di SEO, più in generale, permette di rinominare il percorso per aprire la pagina desiderata e permette di assegnare alla medesima pagina un nome facilmente interpretabile, sia dagli utenti sia dai motori di ricerca, che lo confrontano con il titolo della pagina. Ad esempio, per la pagina con l'indice delle materie, si è scelto l'url **Indice**, per cui il percorso che apparirà sarà [.../drupal/Indice](#) anziché il più anonimo [.../drupal/node/1](#) o, peggio ancora, [.../drupal/?q=node/1](#).

Quindi, l'utilità offerta dal sistema degli URL semplificati, è la possibilità di assegnare a un nodo qualsiasi un nome qualsiasi, cioè abbinare ai nodi generati da Drupal gli URL semplificati scelti.

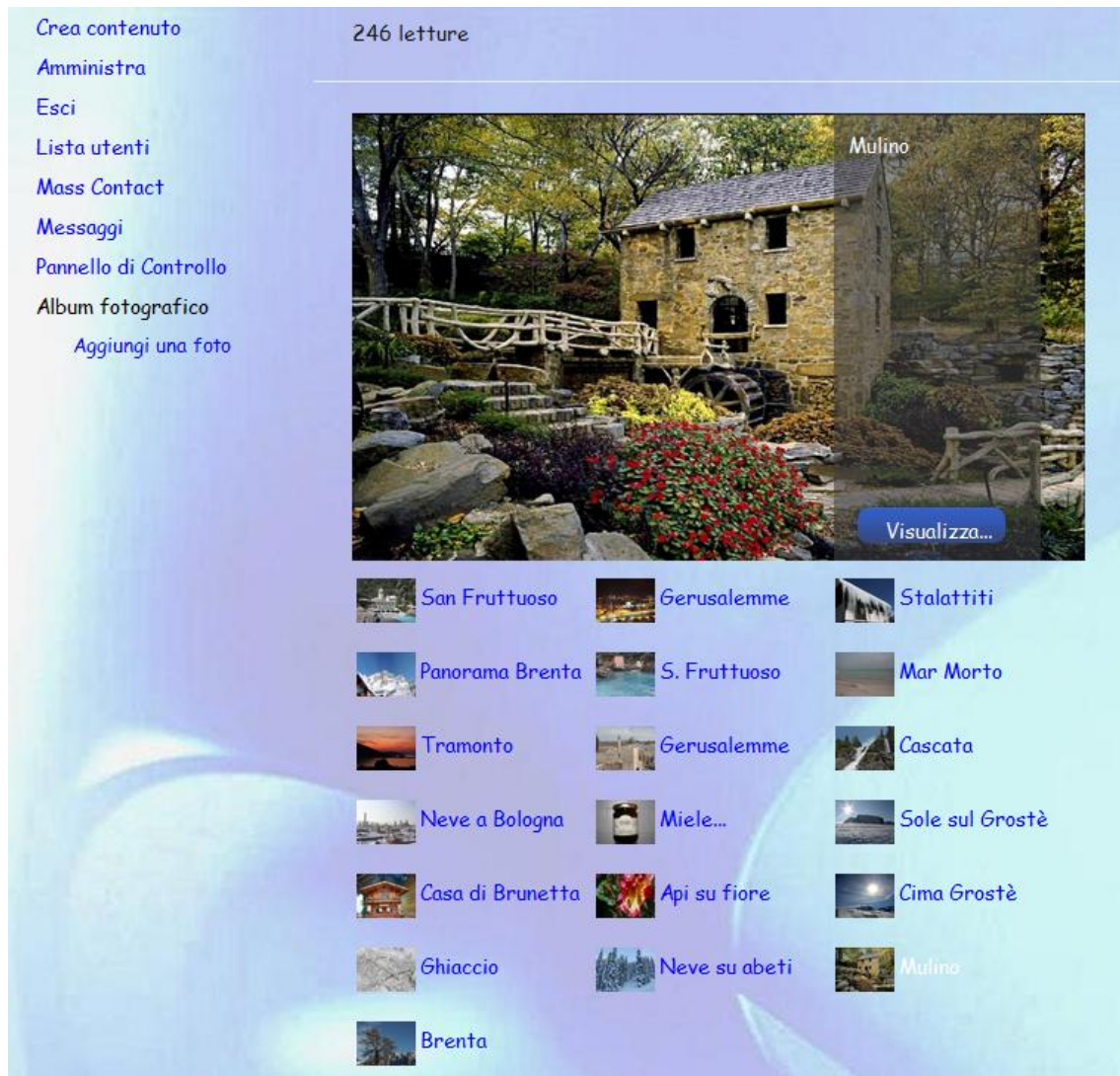
- Il modulo **pathauto**, scaricato e installato, utile ad automatizzare le funzioni del modulo precedente. In pratica viene automaticamente creato un URL semplificato in base al titolo dato alla pagina. È naturalmente sempre possibile intervenire manualmente.
- I moduli **user delete** e **user list**, che permettono, il primo all'utente di potersi cancellare dalla registrazione, il secondo di consultare la lista degli utenti registrati.
- Il modulo **search**, che consente una ricerca per parole chiave in tutto il sito.
- **Statistics** registra le statistiche di accesso al sito.
- **Tracker** permette di tracciare i contenuti recenti degli utenti.
- **Private messages** consente agli utenti di scambiarsi messaggi privati.
- **Privatemsg Email Notification** invece informa gli utenti sui nuovi messaggi privati via e-mail.
- **Block user messages** permette agli utenti di bloccare altri utenti dal mandare loro messaggi.

Inoltre vi sono ulteriori moduli, ad uso esclusivo dell'amministratore, che gli consentono un agile controllo ed eventuale intervento sul sito, come **Mass Contact** che attiva l'invio in massa di e-mail agli utenti registrati, **Backup and Migrate**, di cui si è già discusso, che crea il backup o permette la migrazione del database di Drupal in modo rapido e senza dati non necessari, e così via.

Oltre ai moduli, un altro degli strumenti offerti all'amministratore, è la possibilità di tradurre le interfacce: ampiamente usato in questo sito, favorisce le traduzioni dei vari moduli dalla lingua originale inglese a quella italiana. È importante notare come, al termine di una traduzione di un modulo, sia possibile esportare la stessa e renderla disponibile per condividerla con altri utenti, attraverso, ad esempio, i portali di [drupalitalia](#).

Il modulo ddblock e l'album fotografico

Come appendice a questa applicazione, e a titolo prettamente 'ludico', è stato inserito un modulo, **dynamic display block**, che ha consentito, dopo complesse procedure e ritocchi al codice, di creare un album fotografico. Gli utenti registrati possono creare proprie pagine con una foto che viene salvata in una cartella utente e visualizzata, in modo casuale, in una sorta di album fotografico pubblico:



L'album fotografico come si presenta agli utenti

Per rendere l'aspetto desiderato è stato necessario inserire ulteriori moduli d'appoggio, come **View**, che crea elenchi personalizzati di query dal database, **Views exporter**, che permette l'esportazione di più viste in una sola volta, **Imagefield**, che definisce i campi di tipo immagine, ecc.

Il modulo Dynamic Display Block, da solo, ha permesso di creare, molto velocemente, una piccola slide show, inserita permanentemente nella barra laterale sinistra. È stato sufficiente creare una cartella all'interno della directory sites/default/files e inserirvi all'interno il numero di foto desiderate: attraverso l'interfaccia di configurazione

del modulo si è indicata la cartella, inseriti alcuni parametri come tipo di dissolvenza e velocità della stessa, abilitare i permessi necessari, e la slide show si è resa subito disponibile all'interno dell'elenco dei blocchi.

Conclusioni

La scelta di Drupal come piattaforma per la gestione del proprio sito web è certamente una scelta che, dopo un primo impatto, porta un certo entusiasmo, specialmente nel momento in cui si prende confidenza con il pacchetto.

Con Drupal è possibile arrivare alla costruzione di siti web del tutto professionali, grazie anche, e soprattutto, alla community di Drupal.

Con l'uso, però, ci si rende presto conto quanto sia importante, nella primissima fase di progettazione, la ricerca dell'host o del servizio di hosting che favorisce l'utilizzo di questo software. Drupal, infatti, richiede caratteristiche che non tutti gli host sono in grado di garantire e, specialmente in Italia, piena di piccoli host a basso costo che non garantiscono, rispetto ad altri host stranieri, le stesse prestazioni, è facile incorrere in host dove è sì possibile, con numerose difficoltà, installare Drupal, ma infine si è costretti a rinunciare a una serie di moduli aggiuntivi che possono rallentare di molto il sistema. Ciò è dovuto evidentemente al richiamo continuo del server sul database per estrapolare da esso i contenuti richiesti, creare quindi la pagina web tramite php, sempre dal lato server, per inviare infine il tutto al client che ne ha fatto domanda. È evidente quindi che un numero eccessivo di moduli appesantisce il sistema.

In conclusione, se si è propensi a utilizzare Drupal per la costruzione del proprio portale è bene investire in host più costosi (o magari rivolgersi all'estero...), ma che garantiscano spazio, supporto, caratteristiche e soprattutto prestazioni, in fatto di velocità di risposta del server alle richieste del client. Il costo viene in qualche modo controbilanciato dalla gratuità di questo potente CMS, che, come si è detto, è open source, e dalla soddisfazione di poter realizzare appieno il proprio progetto grazie all'implementazione di più funzioni.

Bibliografia

- <http://drupal.org>
- <http://drupalitalia.org>
- <http://www.drupal.it>
- "Pro Drupal Development" by John VanDyk & Matt Westgate - Apress
- <http://ckeditor.com>
- <http://ddblock.myalbums.biz>