

Lời nói đầu

Tài liệu được viết giống như 1 bản ghi chép, ghi lại những thứ mà mình đã đọc được từ các ebook tiếng anh, do vậy bố cục sắp xếp của nó có thể chưa chính xác, cách trình bày không theo chuẩn 1 ebook nào cả và nhiều chỗ viết chưa rõ nghĩa và không mạch lạc do hạn chế về ngoại ngữ của bản thân mình.tài liệu mang đậm tính chất cá nhân do vậy bạn sẽ bắt gặp trong tài liệu này nhiều đoạn kí tự in đậm, in màu, cỡ chữ lớn bất thường và được tô màu khác nhau, đó là các đoạn có liên quan đến nhau hay là những ghi chú quan trọng mà bạn cần phải đọc kỹ.

Phần lớn nội dung trong tài liệu này dựa trên cuốn ebook “**Beginning Drupal 7 - TODD TOMLINSON**”, nội dung trình bày các thành phần cơ bản, cách sử dụng và thiết lập chúng trong drupal 7, trong chương cuối có đề cập tới cách theming cơ bản trong drupal 7

Trong những tài liệu mình sẽ chia sẻ trong thời gian tới bao gồm :

- Cách theming chuyên sâu trong drupal 7
- Cách tạo và phát triển module trong drupal 7

Mục Lục

Lời nói đầu	1
Creating Content (tạo 1 bài viết content)	7
Các loại lựa chọn để them vào content	7
Menu settings.....	7
Revision Information.....	9
URL Path Settings.....	11
Comment Settings	11
Authoring Information	12
Publishing Options	13
Deleting Content	13
Finding Content.....	14
Users, Roles, and Permissions.....	15
User Accounts	15
Configuring User Account Settings	16
Creating Roles	19
Assigning Permissions(gán quyền)	20
Creating User Accounts(tạo tài khoản user)	21
User Generated Accounts (user tự tạo ra 1 tài khoản mới).....	24
Resetting User's Passwords (thiết lập lại pass)	27
Taxonomy/nguyên tắc phân loại).....	28
Taxonomy Overview(cái nhìn khái quát về Taxonomy)	28
Creating Vocabularies(tạo ra các nhóm Vocabulary)	29
Gán 1 taxonomy vocabulary cho 1 dạng content.....	33
Lựa chọn 1 taxonomy term khi tạo 1 bài viết content	35
Các term phân cấp.....	39
Gán nhiều hơn 1 vocabulary	41
Tạo các thanh Menu.....	41
thêm 1 phần tử vào thanh menu	42
Thêm 1 phần tử menu cho trang liên kết ngoài.....	43
Tạo ra 1 thanh menu mới	44

Cài đặt themes	46
Cách thức làm việc của drupal theme	46
Các bước cài đặt themes	46
Các cấu hình cài đặt	48
Các khối block trong drupal	48
Quản lý hiện thị các block trên website.....	48
Cấu hình các blocks.....	51
phân loại hiện thị theo loại đường dẫn URL	51
Phân loại hiện thị theo dạng content type.....	54
Phân loại theo các nhóm Roles, khi không lựa chọn thì không lọc hiện thị.....	54
Phân loại tùy chỉnh hiện thị theo từng tài khoản user	55
Các loại modules cho drupal	55
Thiết lập cấu hình và thiết lập quyền hạn user trong modules.....	56
Các modules cơ bản cần thiết.....	57
Blogging	57
Bật module Blogs trong drupal	57
Tạo 1 bài blog	59
Hiện thị danh sách cái bài blog.....	60
Tạo khối hiện thị danh sách các bài blog mới nhất	60
Forums	61
Thiết lập cài đặt Forum	61
Web Forms	65
Tạo ra 1 webform	66
Views	70
Tạo View	70
Các cài đặt trong Views.....	71
Các cài đặt cơ bản.....	71
Các cài đặt nâng cao	74
Panels	75
Các thiết kế bố cục Layout trong Panels module	77
Tạo ra 1 trang panel.....	78
Sử dụng bố cục layout tùy chỉnh Flexible Layout	81

Theming	82
Các file drupal theme tiêu chuẩn	82
Giải thích các file template PHP	83
The html.php.tpl File.....	83
The page.tpl.php File	84
The region.tpl.php File	87
Cấu hình Genesis Theme (theme gốc).....	88
Chỉnh sửa theme gốc	90
Tạo ra 1 thanh menu ngang	94
Theming the Footer	96
Theming Nodes	97
Theming Blocks.....	99
Các từ khóa quan trọng trong các files php template	100
Vai trò của file .info.....	100
Vai trò của page.tpl.php file	101
2 theme cơ bản mặc định của drupal 7	104
Sử dụng kỹ thuật chặn và ghi đè để tùy chỉnh theme	106
Mỗi quan hệ liên kết giữa các templates, stylesheets và các hàm dùng để theme	107
Thay đổi theme bằng cách ghi đè lên CSS mặc định	107
Ví dụ về việc ghi đè CSS	107
Ghi đè core stylesheets.....	109
Ghi đè các file templates	110
Ghi đè các hàm functions.....	111
Chuyển đổi các hàm themable functions vào bên trong các file templates riêng	112
Ví dụ về thao tác ghi đè trong Bartik	114
Ghi đè lên các files template mặc định.....	114
Ghi đè các hàm themable functions	116
làm việc với biến template.....	116
Chặn và ghi đè các biến	117
Tạo ra biến mới	118
Tùy chỉnh 1 theme đã có sẵn	118
Lên kế hoạch để tiến hành quá trình chỉnh sửa	119

Lựa chọn 1 theme cơ sở(theme gốc).....	119
Tạo ra 1 sub-theme mới.....	120
Tạo ra 1 bản copy của theme cơ sở(theme gốc)	120
Tạo ra sub-theme trong 1 thư mục mới	120
Xóa các files mà ta không cần.....	120
Cập nhật tên của theme thông qua sub-theme	121
Tạo ra 1 stylesheet cho sub-theme	121
Cập nhật vào trong file .info.....	121
tùy chỉnh sub-theme.....	123
cấu hình theme	124
tự động tạo ra các nội dung giả.....	124
Tùy chỉnh CSS cho phù hợp.....	125
chỉnh sửa 1 file template có sẵn mặc định	126
ghi đè 1 hàm themable function	126
Ghi đè 1 hàm core themable function.....	127
Chuyển đổi 1 hàm themable function thành 1 file template riêng.....	127
Tạo ra 1 theme mới	127
Lập kế hoạch xây dựng theme mới.....	127
tạo ra 1 theme mới thông qua cách sub-theming	128
lựa chọn 1 theme gốc cơ sở	129
Tạo ra sub-theme mới	130
Thiết lập cấu hình cho site.....	132
Styling theme mới.....	133
Mã nguồn của Fusion's theming.....	133
Tùy chỉnh styling	134
Làm việc với CSS	134
tiến hành chỉnh sửa các file template.....	134
làm việc với file template.php	135
xây dựng 1 theme mới mà không dùng cách sub-themeing.....	135
lập kế hoạch thiết kế.....	135
Tạo ra những phần tử cần thiết	136
Tạo ra file .info.....	137

Tùy chỉnh file page.tpl.php.....	140
File style.css.....	141
Thêm vào các phần tử tùy chọn	141
Theme đa chức năng Dynamic Theming.....	141
Định rõ 1 theme admin riêng biệt	142
Sử dụng nhiều page templates.....	142
tạo ra 1 template độc nhất cho trang homepage	144
Sử dụng 1 template khác cho 1 nhóm các trang pages	145
gán 1 template riêng cho 1 trang page riêng	146
Định rõ 1 template riêng biệt cho 1 user riêng biệt	147
Liên kết các phần tử với font page	147
Styling vùng region	148
styling các khối blocks.....	148
Tạo ra CSS styling động	150
Thêm 1 biến mới vào \$classes.....	152
tạo ra các selectors động cho các nodes.....	152
Thay đổi Sitename and Logo	152
Công cụ hỗ trợ theming website	153

Creating Content (tạo 1 bài viết content)

Chọn vào **add content**



Các loại nội dung cơ bản gồm 2 loại chính

A screenshot of the "Add content" page on a Drupal site. The URL "localhost" is visible in the address bar. The page has a dark header with "Add content" and "Find content" buttons. Below the header, the title "Add content" is displayed next to a gear icon. A "Home" link is in the top right. Two content types are listed: "Article" and "Basic page". Each has a brief description and a small icon. At the bottom of the page, a message says "No front page content has been created".

Các loại lựa chọn để thêm vào content

- Menu settings
- Revision information
- URL path settings
- Comment settings
- Authoring information
- Publishing options

Menu settings

Khi 1 bài viết content nào đó quan trọng cần thiết để thêm vào trong danh sách navigational menus của website.mặc định thì drupal tạo ra 1 “Main menu” và “Secondary menu”. “Main menu” thì hiện thị ở trên đầu và “Secondary menu” có thể xuất hiện dưới chân trang footer, hay ở thanh bên trái hay ở dưới thanh banner của Website – nó phụ thuộc vào theme mà ta chọn

Thực hiện gán bài viết content vào trong “Main menu” :

- ấn vào thẻ **Menu settings**
- tích chuột vào **Provide a menu link**
- ta nhìn thấy 3 trường “Menu link title,” “Parent item,” and “Weight.”

ở “**Menu link title**” điền vào các từ ngắn gọn đây chính là phần hiện thị trên thanh menu(nó khác với tiêu đề bài viết)

ở “**Parent item**” dạng sổ xuống thì ta lựa chọn vào < Main menu>

trường “**Weight.**” Là trường sắp xếp thứ tự ưu tiên, chọn Weight là 0 thì thứ tự ưu tiên sẽ sắp xếp theo thứ tự a,b,c,...thứ tự ưu tiên càng cao khi giá trị trong Weight càng nhỏ (ở thanh menu ngang thì Weight càng nhỏ thì nó sẽ ưu tiên xuất hiện ở bên trái so với Weight nặng hơn,ở thanh menu dọc thì weight càng nhỏ thì ưu tiên xuất hiện ở trên xuống)

Menu settings
First Article Content

Revision information
No revision

URL path settings
No alias

Comment settings
Open

Authoring information
By admin on 2012-01-17 14:55:23 +0700

Publishing options
Published, Promoted to front page

Provide a menu link

Menu link title
First Article Content

Description

Shown when hovering over the menu link.

Parent item <Main menu>

Weight 0

Menu links with smaller weights are displayed before links with larger weights.

Chú ý khi đã thêm mới vào <Main menu>



Revision Information

Là thông tin cho phép duyệt lại khi ta tiến hành thay đổi 1 content nào đó mà gặp phải lỗi chưa kịp lưu lại ví dụ trường hợp ta đang soạn thảo văn bản đột nhiên bị mất điện và việc lưu do đó gặp lỗi. Lúc này ta muốn khôi phục lại phần văn bản về lúc trước khi thay đổi xảy ra. Đây chính là khả năng Drupal cung cấp trong Revision Information. Nó cung cấp khả năng tự tạo ra 1 bài viết sao chép khác khi bài viết đang được tiến hành thay đổi. Thực hiện bằng cách:

- Tích chuột vào "Create new revision."
- Ở phần dưới Revision log message: ghi lại chú thích mô tả những gì bạn thay đổi trong bài viết

NÓI KHÁI QUÁT HƠN LÀ : Revision Information là thông tin giúp người admin (quản lý cấp cao hơn) kiểm duyệt lại nội dung mà người viết muốn thay đổi bên trong bài viết. Nếu thấy bài viết thay đổi không phù hợp thì có thể khôi phục lại ngay bài viết cũ trước khi có sự thay đổi đó. Có thể chuyển tiếp qua lại giữa các phiên bản của cùng 1 bài viết.

Ví dụ:

phiên bản bài viết trước khi thay đổi:

bài viết đầu tiên loại Article

[View](#) [Edit](#) [Revisions](#)

Submitted by [admin](#) on Tue, 01/17/2012 - 14:55



Creating Content (tạo 1 bài viết content)

Chọn vào add content

Các loại nội dung cơ bản gồm 2 loại chính
Article

Basic page

Tags:

[Article](#) [first](#)

Chọn Revisions lựa chọn khác phiên bản khi có sự thay đổi:

Home > First Article Content

Revisions allow you to track differences between multiple versions of your content, and revert back to older versions.

REVISION	OPERATIONS
01/17/2012 - 15:57 by admin <i>Copy of the revision from Tue, 01/17/2012 - 15:48.</i>	<i>current revision</i>
01/17/2012 - 15:55 by admin <i>đổi tiêu đề "bài viết đầu tiên loại Article" thành "Bài viết đầu tiên trong dạng Content Article". và thêm từ "nội dung đã được thay đổi" vào bên trong thân bài viết.xóa ảnh</i>	<i>revert delete</i>
01/17/2012 - 15:48 by admin <i>thay đổi tiêu đề cho phù hợp hơn với nội dung bài viết</i>	<i>revert delete</i>
01/17/2012 - 15:30 by admin	<i>revert delete</i>

ở đây có 3 phiên bản : phiên bản dưới cùng là phiên bản gốc (chú ý vào giờ tạo khi các phiên bản thay đổi này được thay đổi trong cùng 1 ngày), phiên bản thứ 2 từ dưới lên là phiên bản thay đổi tiêu đề cho phù hợp, phiên bản thứ 3 từ dưới lên là phiên bản thay đổi: **đổi tiêu đề "bài viết đầu tiên loại Article" thành "Bài viết đầu tiên trong dạng Content Article". và thêm từ "nội dung đã được thay đổi" vào bên trong thân bài viết.xóa ảnh trong bài viết.thêm vào là ảnh khác**

để lựa chọn muốn khôi phục phiên bản nào thì chọn vào **Revert**.**có thể xem trước phiên bản cần khôi phục bằng cách ấn vào link ngày tháng giờ tạo phiên bản đó**

01/17/2012 15:48 by admin
thay đổi tiêu đề cho phù hợp hơn với nội dung bài viết

Home » bài viết đầu tiên loại Article » Revisions

Revert Cancel

Mở rộng hơn: không chỉ với các dạng content mặc định có trong Drupal mà ta cũng có thể tạo ra các Revision với bất cứ dạng content tùy chỉnh nào khác

URL Path Settings

Tạo đường dẫn thân thiện giúp cho SEO.điền vào đường dẫn làm đẹp vào phần URL alias (chú ý các từ cách nhau bằng dấu “-” và không được thêm kí tự “/” vào)

URL alias

bai-viet-dau-tien

Optionally specify an alternative URL by which this content can be accessed.
relative path and don't add a trailing slash or the URL alias won't work.

Comment Settings

Menu settings First Article Content	<input checked="" type="radio"/> Open Users with the "Post comments" permission can post comments.
Revision information No revision	<input type="radio"/> Closed Users cannot post comments.
URL path settings Alias: bai-viet-dau-tien	
Comment settings Open	
Authoring information By admin on 2012-01-17 14:55:23 +0700	
Publishing options Published, Promoted to front page	

Dạng mặc định của 1 comment form

Add new comment

Your name **admin**

Subject

Comment *

Text format **Filtered HTML** ▾ More information about text formats 

• Web page addresses and e-mail addresses turn into links automatically.
• Allowed HTML tags: <a> <cite> <blockquote> <code> <dl> <dt> <dd>
• Lines and paragraphs break automatically.

Authoring Information

Menu settings First Article Content	Authored by admin Leave blank for <i>Anonymous</i> .
Revision information No revision	
URL path settings Alias: bai-viet-dau-tien	Authored on 2012-01-17 14:55:23 +0700
Comment settings Open	Format: 2012-01-17 14:55:23 +0700. The date format is YYYY-MM-DD i the time of form submission.
Authoring information By admin on 2012-01-17 14:55:23 +0700	
Publishing options Published, Promoted to front page	

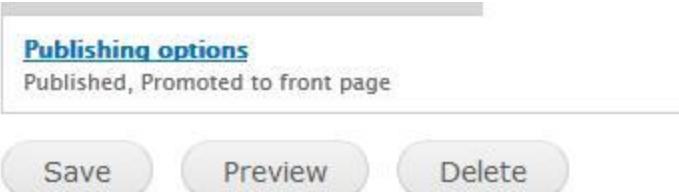
Publishing Options

Menu settings First Article Content	<input checked="" type="checkbox"/> Published
Revision information No revision	<input checked="" type="checkbox"/> Promoted to front page
URL path settings Alias: bai-viet-dau-tien	<input type="checkbox"/> Sticky at top of lists
Comment settings Open	
Authoring information By admin on 2012-01-17 14:55:23 +0700	
Publishing options Published, Promoted to front page	

Lựa chọn “Sticky at the top of list” : cung cấp 1 cơ chế luôn đảm bảo rằng bài viết luôn xuất hiện trên đầu danh sách bài viết

Deleting Content

Vào phần Edit mỗi bài viết,kéo xuống dưới cùng



Finding Content

ấn vào **Content** trên thanh menu màu đen

The screenshot shows the "Content" administration page. At the top, there's a navigation bar with links for "Dashboard", "Content", "Structure", and "Appearance", along with "Add content" and "Find content" buttons. Below this is a header with "Content" and "localhost". The main area has a breadcrumb trail "Home » Administration" and a "Add content" button. It includes sections for "SHOW ONLY ITEMS WHERE" (status: any, type: any) with a "Filter" button, "UPDATE OPTIONS" (Publish selected content dropdown with "Update" button), and a table for "CONTENT" with one row: "bài viết đầu tiên loại Article" (Article). A sidebar on the left is mostly dark.

Users, Roles, and Permissions

Users (hay là người viếng thăm website): trong drupal 7 được chia làm 2 dạng chung:

- anonymous users (người dùng vô danh)
- authenticated users (người dùng đã xác thực)

anonymous users là những khách xem thông thường, không thực hiện Login sử dụng 1 userID và pass nào cả. trong Drupal cung cấp cho bạn những khả năng để hạn chế người dùng vô danh những gì có thể làm với Website

authenticated users là những người dùng có thể đăng nhập bằng cách sử dụng 1 userID và pass

Roles là 1 cơ chế trong Drupal giúp người admin có thể định rõ loại người dùng xác thực. ta có thể định rõ các Roles(vai trò nhiệm vụ riêng của các nhóm người dùng xác thực) này dựa vào đặc thù vai trò của từng bộ phận riêng biệt ví dụ như : khách đặt mua hàng, người bán, nhân viên marketing, Hay các Roles này được phân loại dựa vào các đặc trưng theo chức năng : tác giả bài viết, người phê bình,

Roles được liên kết với 1 phần riêng biệt trong website ví dụ như : products, support, sales, homepage hay bất cứ phần nào ra chính ta định nghĩa ra

Roles là cách đơn giản nhất để phân loại các người dùng xác thực vào các nhóm đã được phân chia các quyền hạn riêng biệt

Bất cứ người dùng xác thực nào cũng có thể không được phân nhóm hay được phân vào 1 hay nhiều hơn 1 role(vai trò, nhiệm vụ). ví dụ 1 người dùng có thể được phân vào các nhóm roles(vai trò) như : bộ phận bán hàng, tác giả bài viết, hay mục quản lý sản phẩm,...

Permissions (phân quyền): là 1 cơ chế trong drupal cho phép điều khiển những người dùng đã được phân nhóm vào trong các loại role cụ thể. có hàng tá các quyền mà ta có thể lựa chọn bật hay hủy bỏ cho mỗi loại role mà ta đã định nghĩa

Ví dụ các quyền có thể được cài đặt vào trong từng role như: tạo 1 trang mới, tạo 1 bài viết mới, chỉnh sửa bài viết bất chấp ai là tác giả của nó, tính năng tìm kiếm bài viết content trên website, thêm mới tài khoản user mới

User Accounts

Tất cả các website drupal phải có nhất 1 tài khoản user, tài khoản này được tạo ra trong quá trình cài đặt, đây là tài khoản admin(toàn quyền)

Muốn tạo thêm các tài khoản user khác, Drupal cung cấp cho ta 3 sự lựa chọn:

1. users có thể tạo ra các tài khoản của họ mà không cần sự chấp nhận của admin

2. users yêu cầu 1 tài khoản mới nhưng phải đợi admin chấp thuận thì tài khoản đó mới được kích hoạt
3. chỉ duy nhất admin mới có quyền tạo ra tài khoản mới

tùy thuộc vào cách ta xem xét cách thức người dùng viếng thăm website mà đưa ra các cách lựa chọn tạo tài khoản.ví dụ nếu website của ta là website thông tin thông thường k yêu cầu người dùng phải đăng nhập để post comment.thì lựa chọn thứ 3 là cách lựa chọn tốt nhất.nếu website bao gồm những nội dung không phải ở dạng public ,và yêu cầu cần có 1 tài khoản ta nên lựa chọn cách 1,để họ tự tạo ra tài khoản của mình mà k cần admin chấp nhận

còn khi cần sự đồng ý từ admin thì lựa chọn thứ 2

Configuring User Account Settings

Để truy cập vào trang Setting, bấm vào liên kết Configuration trên thanh menu nằm ở đầu trang



ở trang Configuration, kéo chuột xuống phần có tên là “**People and Permissions.**”

A screenshot of the 'People' section in the Configuration menu. It shows two main items: 'Account settings' and 'IP address blocking'. 'Account settings' is described as 'Configure default behavior of users, including registration requirements, e-mails, fields, and user pictures.' 'IP address blocking' is described as 'Manage blocked IP addresses.' Both items have a small circular icon with a right-pointing arrow to their left.

Cái nhìn đầu tiên (đang trong tài khoản admin được tạo ra từ đầu) trong phần **Account Settings**

2 phần đầu tiên để mặc định

ANONYMOUS USERS

Name *

The name used to indicate anonymous users.

ADMINISTRATOR ROLE

Administrator role
▼
This role will be automatically assigned new permissions whenever a module is enabled. Changing this setting will not affect existing permissions.

Phân tiếp theo “**Registration and Cancellation**,” xác định rõ cách mà tài khoản user được thiết lập trong hệ thống

Ta lại quay lại 3 lựa chọn xem xét cách tạo tài khoản user đã nói ở trên:

Who can register accounts?

- Administrators only
- Visitors
- Visitors, but administrator approval is required

- Require e-mail verification when a visitor creates an account.

New users will be required to validate their e-mail address prior to logging in immediately upon registering, and may select their own e-mail address for future logins.

Lựa chọn tiếp là việc thực hiện với bài viết mà 1 user nào đó đã tạo rồi bị admin khóa user đó vào(hủy bỏ)

When cancelling a user account

- Disable the account and keep its content.
- Disable the account and unpublish its content.
- Delete the account and make its content belong to the *Anonymous user*.
- Delete the account and its content.

Users with the *Select method for cancelling account* or *Administer users* permissions

Mục tiếp theo là **Personalization** có cho phép người dùng thêm vào ảnh đại diện và chữ ký của họ trong profile hay không?

PERSONALIZATION

Enable signatures.

Enable user pictures.

Picture directory

pictures

Subdirectory in the file upload directory where pictures will be stored.

Default picture



URL of picture to display for users with no custom picture selected. Leave blank for none.

Lựa chọn cuối cùng là các thiết lập liên quan tới **Email**

E-mails

Welcome (new user created by administrator)

Edit the welcome e-mail messages sent:
[user:name], [user:mail], [site:login-url],

Welcome (awaiting approval)

Subject
An administrator created an account

Welcome (no approval required)

Account activation

Body

[user:name],

A site administrator at [site:name] has
your browser:

[user:one-time-login-url]

This link can only be used once to log in.

Account blocked

Account cancellation confirmation

Account canceled

Password recovery

Ta cũng có thể định nghĩa các trường trong phần **user's profile** bằng cách thêm vào các tên nhãn mới cho các trường mới.

Ta có thể thêm các trường mới này trong mục **MANAGE FILEDS**



Ví dụ ta muốn thêm 1 trường dữ liệu mới trong profile của user.ví dụ thêm vào trường vị trí cho user(location)

Creating Roles

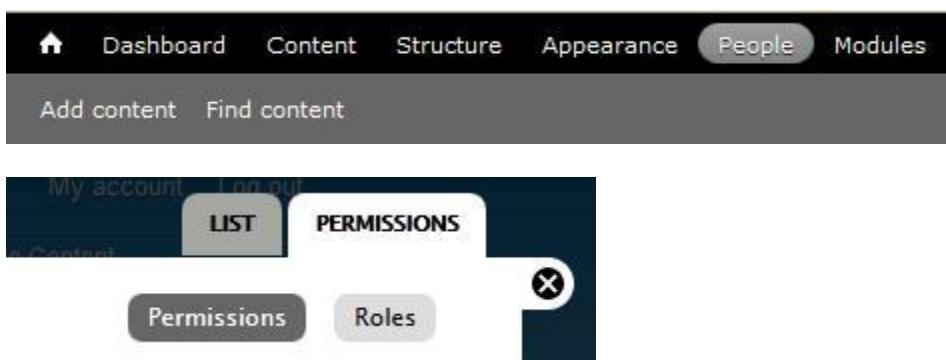
User roles là 1 cơ chế để giúp ta phân loại và nhóm các users có cùng chức năng và nhiệm vụ lại với nhau.nếu website giống như 1 ngôi trường cơ bản, thì có thể phải nhóm thành các nhóm roles như giáo viên,học sinh,cán bộ biên chế khác.nếu website là 1 website lớn mang tính tập thể cao,ta có thể phân thành các nhóm roles như : content authors, content reviewers, publishers, and forum administrators

Phần chung,để cho đơn giản ta sẽ tạo ra 1 ít nhóm roles,và phải thiết lập các quyền cho từng nhóm roles đó.tuy nhiên nó đồng nghĩa với sự ít mềm dẻo hơn trong quản lý.

Theo cách chung nhất, ta có thể phân loại ra làm 2 dạng roles:

- 1 nhóm role cho những users chuyên trách về phần sáng tác viết bài, cho việc xuất bản, quản lý bài viết content và các loại menu trên website
- 1 nhóm role được cấp quyền truy cập vào các bài viết “non-public”.điều này có nghĩa là các users này phải là authenticated users(người dùng đã xác thực) .những người dùng này có thể xem các bài viết và thêm comment vào bên trong các bài viết nhưng không thể viết bài hay chỉnh sửa và xóa các bài viết

Để tạo mới 1 nhóm role : nhấp vào **People** ở trên thanh menu >> sau đó nhấp vào thẻ **Permissions** >> ấn tiếp vào **Role** nơi mà ta có thể xem, chỉnh sửa,thêm mới xóa các nhóm Role



The screenshot shows the 'PERMISSIONS' tab selected in the top navigation bar. Below it, there are two tabs: 'Permissions' and 'Roles'. A large 'X' icon is in the top right corner. The main area is titled 'NAME' and lists three roles: 'anonymous user (locked)', 'authenticated user (locked)', and 'administrator'. Each role has a plus sign icon to its left. At the bottom right of this list is a button labeled 'Add role' with a hand cursor icon pointing at it.

Ví dụ như ta tạo ra 1 role mới tên là “**company user**”

The screenshot shows the details for the 'company user' role. It includes a 'company user' label, an 'edit role' button, and an 'edit permissions' button. Below these buttons is another 'Add role' button.

Assigning Permissions(gán quyền)

Drupal core và từng module của nó cung cấp 1 tập các quyền được định nghĩa trước mà ta phải bật hay không bật trên từng nhóm role

Việc phân quyền bằng cách : nhấn vào **People** >> chọn vào thẻ **Permission** >> hiện thị ra trang ta sẽ dùng để thiết lập các quyền cho các nhóm role

The screenshot shows the 'Permissions' tab selected in the top navigation bar. Below it, a descriptive text explains that permissions control what users can do and see on the site, mentioning the 'Authenticated Users' and 'Administrators' roles. It also notes that permissions for the 'Authenticated Users' role apply to all users who can log in.

A 'Hide descriptions' link is present. The main content is a table where permissions are mapped across six user roles: ANONYMOUS USER, AUTHENTICATED USER, ADMINISTRATOR, NEW ROLE, and COMPANY USER. The 'Block' section includes 'Administer blocks' permission, which is checked for the ADMINISTRATOR role. The 'Comment' section includes 'Administer comments' permission, which is checked for the COMPANY USER role.

PERMISSION	ANONYMOUS USER	AUTHENTICATED USER	ADMINISTRATOR	NEW ROLE	COMPANY USER
Block	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comment					
Administer comments					<input checked="" type="checkbox"/>

Trang này hiện thị danh sách tất cả các quyền trên website, muốn bật quyền nào thì ta phải tích chuột vào quyền đó

Như ví dụ ở trên, ta kéo chuột xuống phần có tên là **Node**. Ở trong phần này ta tích vào các ô dưới đây để phân quyền cho nhóm role **company user**

- Create new Article content (tạo 1 bài viết loại Article mới)
- Create new Basic page content (tạo 1 trang loại Basic mới)
- Edit own Article content
- Edit own Basic page content
- Delete own Article content
- Delete own Basic page content

Các quyền cho nhóm role **Restricted User**, kéo chuột tới phần **Comment** và tích vào các quyền sau đây:

- View comments (xem comment)
- Post comments without approval (post comments mà không cần sự chấp nhận)
- Edit own comments (chỉnh sửa comments)

Creating User Accounts(tạo tài khoản user)

ở trên ta đã định rõ các nhóm roles với cái quyền tương ứng và giờ ta sẵn sàng tạo ra các tài khoản user để tạo tài khoản mới : nhấn vào **People** >> chọn vào thẻ **List**

The screenshot shows the 'People' administration page in Drupal 7. At the top, there is a navigation bar with links: Dashboard, Content, Structure, Appearance, People (which is highlighted), Modules, Configuration, Reports, and Help. Below the navigation bar, there are links for 'Add content' and 'Find content'. On the right side of the header, it says 'Hello admin' and 'Log out'. There is also a link 'Edit shortcut'.

The main content area has a title 'People' with a 'localhost' placeholder. Below the title, there are two tabs: 'LIST' (selected) and 'PERMISSIONS'. A close button 'X' is located in the top right corner of the content area.

The 'LIST' tab displays a table of users. The columns are: a checkbox column, 'USERNAME', 'STATUS', 'ROLES', 'MEMBER FOR' (with a dropdown arrow), 'LAST ACCESS', and 'OPERATIONS'. The 'MEMBER FOR' column currently has 'any' selected in its dropdown.

Below the table, there is a section titled 'SHOW ONLY USERS WHERE' with three dropdown menus: 'role' (any), 'permission' (any), and 'status' (any). A 'Filter' button is located to the right of these dropdowns.

Underneath the filter section, there is a 'UPDATE OPTIONS' section containing a dropdown menu set to 'Unblock the selected users' and a 'Update' button.

Ta nhấn vào link **Add user**

The screenshot shows the 'Add user' form. At the top, there is a title 'People' with a 'localhost' placeholder. Below the title, there are links: Home, Administration, and a blue '+ Add user' button.

Theo ví dụ ta sẽ tạo ra 1 tài khoản user bằng cách điền vào theo các cách sau:

[Home](#) » [Administration](#) » [People](#)

This web page allows administrators to register new users. Users' e-mail addresses are used to identify them.

Username *

ngotrungh

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

E-mail address *

myloveforyounqt@gmail.com

A valid e-mail address. All e-mails from the system will be sent to this address. The e-mail address is required to receive certain news or notifications by e-mail.

Password *

Password strength:

Fair



Confirm password *

Passwords match: yes

To make your password stronger:

- Add lowercase letters
- Add uppercase letters
- Add punctuation

Provide a password for the new account in both fields.

Thiết lập phân quyền vào nhóm role : company user vừa tạo ở phía trên

Status

Blocked

Active

Roles

authenticated user

administrator

restricted user

company user

Notify user of new account

Create new account

User Generated Accounts (user tự tạo ra 1 tài khoản mới)

như ta vừa thiết lập trong [Configuring User Account Settings >> Account Settings >>](#) trong phần “**Registration and Cancellation,**” ta lựa chọn **thứ 3**, là user được phép tạo tài khoản mới nhưng phải có sự chấp nhận của admin

Ta lựa chọn tạo 1 tài khoản mới : chọn vào “**Create new account**”

The image shows a user login form titled "User login". It contains two input fields: "Username *" with "admin" entered and a red asterisk icon; and "Password *" with "*****" entered and a red asterisk icon. Below the fields is a link "Create new account" followed by "Request new password". At the bottom is a "Log in" button.

Lưu ý rằng : drupal chỉ cho phép người dùng sử dụng 1 email duy nhất, việc sử dụng lại 1 email đã có sẽ thông báo lỗi

Home » User account

User account

[Create new account](#) [Log in](#) [Request new password](#)

Username *

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

E-mail address *

A valid e-mail address. All e-mails from the system will be sent to this address. The e-mail address is required if you wish to receive certain news or notifications by e-mail.

[Create new account](#)

Khi tài khoản được tạo thì drupal sẽ gửi 1 email tới địa chỉ email mà người dùng vừa điền vào và thông báo thành công :

"Thank you for applying for an account. Your account is currently pending approval by the site administrator. In the meantime, a welcome message with further instructions has been sent to your e-mail address."

Như đã thiết lập ở trên, tài khoản mới tạo ra phải có sự cho phép của admin. do đó trong tài khoản admin ta cần kích hoạt tài khoản vừa được yêu cầu bằng tay.

Ví dụ ta vừa yêu cầu thêm mới tài khoản **thanhthuy**. ta truy cập vào **People >> thẻ List >>** sẽ thấy danh sách các user yêu cầu



The screenshot shows the Drupal 7 User Management interface. At the top, there are links for 'Add content' and 'Find content'. On the right, there is an 'Edit shortcut' link. Below these, a blue button labeled '+ Add user' is visible. A search bar with placeholder text 'Search users' is present. Under the search bar, there is a section titled 'SHOW ONLY USERS WHERE' with dropdown menus for 'role' (set to 'any'), 'permission' (set to 'any'), and 'status' (set to 'any'). To the right of these dropdowns is a 'Filter' button. Below this, there is a section titled 'UPDATE OPTIONS' with a dropdown menu set to 'Unblock the selected users' and a 'Update' button. The main area displays a table of users:

	USERNAME	STATUS	ROLES	MEMBER FOR	LAST ACCESS	OPERATIONS
<input type="checkbox"/>	thanhthuy	blocked		27 sec	never	edit
<input type="checkbox"/>	ngotrung	active	• company user	24 min 20 sec	never	edit
<input type="checkbox"/>	admin	active	• administrator	2 days 6 hours	5 sec ago	edit

Ta chọn **OPERATIONS >> edit** tại user **thanhthuy** có **STATUS : blocked**

ta cần thiết lập lại tại 2 mục **Status** và **Roles**.chuyển Status từ blocked sang Active và phân user này vào 1 nhóm Role nào đó

Status

Blocked

Active

Roles

authenticated user

administrator

restricted user

company user

Resetting User's Passwords (thiết lập lại pass)

1 trong những đặc tính của drupal giúp người quản trị tiết kiệm hàng giờ làm việc mỗi năm là cung cấp khả năng cho những users reset lại password của họ mà phải gửi email tới admin và yêu cầu họ reset lại password cho họ

Bằng cách ấn vào **Request new password**

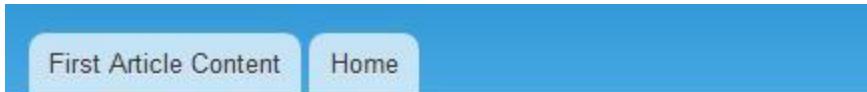
User login

Username *
 *

Password *
 *

• Create new account
• Request new password 

Log in



Home » User account

User account

[Create new account](#) [Log in](#) [Request new password](#)

Username or e-mail address *

E-mail new password

Taxonomy(nguyên tắc phân loại)

1 trong những tính năng mới của drupal dễ gây nhầm lẫn là Taxonomy(nguyên tắc phân loại).ta sẽ sử dụng Taxonomy để phân nhóm các bài viết content vì vậy những khách viếng thăm dễ dàng tìm thấy các thông tin có liên quan đến 1 chủ đề riêng nào đó

Taxonomy Overview(cái nhìn khái quát về Taxonomy)

Nhìn theo quan điểm đơn giản nhất thì Taxonomy cung cấp 1 cơ chế nhằm để phân nhóm các bài content trên website.ví dụ như : lợi ích đến từ 1 thư viện chính là do những người thủ thư mất khá nhiều thời gian để kiểm tra từng quyển sách , movie, tạp chí, music rồi xắp xếp chúng vào các giá sách trong thư viện.các thủ thư,nhà xuất bản phải cẩn thận suy nghĩ về cách để phân nhóm từng loại 1 vì vậy thật dễ dàng để xác định vị trí của các loại này trong thư viện.không chỉ dễ để tìm kiếm đúng thứ cần tìm trong thư viện mà còn khá dễ dàng xác định những loại tương tự như vậy trong thư viện.hãy tưởng tượng như thế này, bạn đang bước vào thư viện và xác định nơi chứa các thứ liên quan đến web design.ở nơi chứa đó, bạn có thể tìm thấy các cuốn sách về các chủ đề trong phạm vi từ việc sử dụng photoshop để tạo ra các bản thiết kế tới cách lập trình bằng PHP

Tóm lại việc phân loại những thứ có liên quan vào 1 chỗ là rất quan trọng cho việc tìm kiếm

Taxonomy khá giống với việc phân nhóm các loại có trong thư viện nhưng là phân nhóm các bài viết content trong website

Trong drupal taxonomy được phân chia làm 2 đối tượng : **vocabulary** và **term**

1 vocabulary đơn giản là 1 nhóm các terms có liên quan tới nhau, với term là các 1 từ hay 1 nhóm từ miêu tả 1 khía cạnh riêng biệt của 1 **vocabulary**.hiểu theo cách khác thì vocabulary giống như 1 nhóm category và 1 term giống như 1 nhãn tag

Để có 1 cái nhìn thực tế hơn ta xét ví dụ như : ta đang tạo ra 1 website tập trung vào tin tức về thể thao.các khách viếng thăm mục tiêu của trang web là những người quan tâm tới những thông tin liên quan tới những đội yêu thích của họ.chúng ta phải tổ chức sắp xếp nội dung bài viết theo các loại thể thao ví dụ như:

- Football
- Baseball
- Basketball
- Hockey
- Soccer

Các đội tương ứng :

- Ravens
- Trailblazers
- Lakers
- Raiders
- Yankees

Ta cần phải định nghĩa 1 vocabulary cho từng loại thể thao và phân các terms vào vocabulary đó với từng loại thể thao đã được liệt kê.tu cũng có thể tạo ra vocabulary khác cho các đội tuyển và phân từng đội tuyển đã liệt kê ở trên thành 1 term tương ứng .ta có thể sử dụng taxonomy để liệt kê tất cả các bài viết liên quan tới 1 vocabulary riêng biệt trên 1 trang

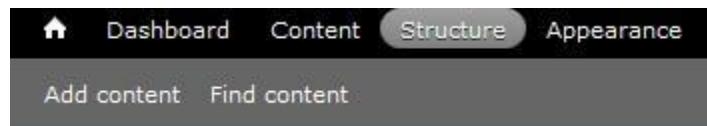
Creating Vocabularies(tạo ra các nhóm Vocabulary)

Bước đầu tiên là ta sử dụng Taxonomy để xác định và tạo ra các **vocabularies** –cái mà ta sẽ dựa vào đó để phân nhóm các bài viết content trên website, điều này phụ thuộc vào nội dung chủ yếu và độ mở rộng của những thứ ta muốn đề cập đến trên trang web.cũng có thể ta chỉ cần 1 vocabulary hay 1 vài vocabularies.đây không hẳn là câu trả lời chính xác và cũng chẳng có 1 quy tắc nào giúp ta xác định rõ có bao nhiêu vocabulary mà website của ta cần.

Cách tốt nhất là ta phải nghĩ xem phải gộp các content như thế nào?các content có liên quan với nhau (ví dụ như các loại thể thao) thì 1 vocabulary có lẽ là đủ cho những gì mà ta cần.nếu các content không có liên quan với nhau (ví dụ như website tổng hợp tin tức nói về các vấn đề khác nhau : công nghệ, sức khỏe, nghề nghiệp, lĩnh vực giải trí) thì có lẽ ta cần 1 vài **vocabularies** ,mỗi vocabulary tương ứng với mỗi loại tin

Ta luôn biết chắc rằng luôn phải có ít nhất 1 vocabulary.khởi tạo bằng cách :

Ấn vào **Structure** trên thanh admin menu >> **Taxonomy** >> **Add vocabulary**



The screenshot shows the 'Structure' section of the Drupal 7 administration menu. It includes links for 'Blocks', 'Content types', 'Menus', and 'Taxonomy'. A sub-menu for 'Creating Content (tạo bài viết content)' is visible at the bottom.

The screenshot shows the 'Taxonomy' section of the Drupal 7 administration menu. It displays a vocabulary named 'Tags' with options to 'edit vocabulary', 'list terms', and 'add terms'. A 'VOCABULARY NAME' table header is also visible.

Ví dụ: ta thêm vào vocabulary các loại thể thao

The screenshot shows the Drupal 7 administration interface under the 'Structure' section. At the top, there is a success message: 'Created new vocabulary *các loại thể thao*'. Below this, a descriptive text explains that Taxonomy is for categorizing content. A table lists two vocabularies: 'các loại thể thao' and 'Tags'. Each row has 'edit vocabulary', 'list terms', and 'add terms' links. A 'Save' button is at the bottom left, and a 'Show row weights' link is at the top right.

VOCABULARY NAME	OPERATIONS
⊕ các loại thể thao	edit vocabulary list terms add terms
⊕ Tags	edit vocabulary list terms add terms

Bước tiếp theo ta thêm vào danh sách các terms vào vocabulary **các loại thể thao**

Ta ấn vào “Add terms”. ở form “Add terms” chú ý tới “URL alias” để làm Seo

Name *

Description

thông tin liên quan tới bóng bầu dục

Text format Filtered HTML ▾ More information about text formats ⓘ

- Web page addresses and e-mail addresses turn into links automatically.
- Allowed HTML tags: <a> <cite> <blockquote> <code> <dl> <dt> <dd>
- Lines and paragraphs break automatically.

URL alias

 ↵

Optionally specify an alternative URL by which this term can be accessed. Use a relative path and don't add a trailing slash or the URL alias won't work.

► RELATIONS

Chú ý tới việc xắp xếp thứ tự của các term trong 1 vocabulary, cũng dựa vào sự ưu tiên **Weight**

The screenshot shows the Drupal 7 Taxonomy interface for the vocabulary 'Các loại thể thao'. At the top, there are tabs: LIST (selected), EDIT, MANAGE FIELDS, and MANAGE DISPLAY. Below the tabs, the breadcrumb navigation shows: Home > Administration > Structure > Taxonomy. A note says: 'You can reorganize the terms in Cạ́c lōi th̄e thao using their drag-and-drop handles, and group terms under a parent term by sliding them under a to the right of the parent.' A 'Add term' button is visible. A 'Hide row weights' link is at the top right of the table. The table lists four terms:

NAME	WEIGHT	OPERATIONS
bóng bầu dục	0	edit
bóng chày	0	edit
bóng rổ	0	edit
bóng đá	0	edit

At the bottom are 'Save' and 'Reset to alphabetical' buttons.

Gán 1 taxonomy vocabulary cho 1 dạng content

Để tiến hành gán (phân nhã̂n) thì yêu cầu admin phải cập nhật tất cả các dạng content phải được tham chiếu tới các vocabulary. do vậy ta tiến hành như sau: (thêm 1 trường mới vào 1 content type, trường này sẽ liên kết và lấy dữ liệu cụ thể là các terms có trong vocabulary)

Ấn vào **Structure >> Content types >> Manage fields**

The screenshot shows the Drupal 7 Content types interface. The top navigation bar has tabs: Dashboard, Content, Structure (selected), Appearance. Below the navigation is a search bar with 'Add content' and 'Find content' buttons.

Structure 

localhost

Home » Administration

Blocks

Configure what block content appears in your site's sidebars and other regions.

Content types 

Manage Content types, including default status, front page promotion, comment settings, etc.

Menus

Add new menus to your site, edit existing menus, and rename and reorganize menu links.

Taxonomy

Manage tagging, categorization, and classification of your content.

Content types 

localhost

Home » Administration » Structure

+ Add content type

NAME	OPERATIONS
Article (Machine name: article) Use <i>articles</i> for time-sensitive content like news, press releases or blog posts.	edit manage fields  manage display delete
Basic page (Machine name: page) Use <i>basic pages</i> for your static content, such as an 'About us' page.	edit manage fields  manage display delete

LABEL	WEIGHT	PARENT	NAME	FIELD	WIDGET
Title	-5	- None -	title	Node module element	
Tags	-4	- None -	field_tags	Term reference	Autocomplete te widget (tagging)
Body	-4	- None -	body	Long text and summary	Text area with a summary
Image	-1	- None -	field_image	Image	Image

Add new field

Label	Loại Thể Thao	Weight	-5	Parent	- None -	Field name	field_loai_the_thao	Type of data to store	Term reference	Select list

Ta đã thực hiện thêm 1 trường mới tên là “**Loại Thể Thao**” – là 1 trường tham chiếu tới các **terms** chứa trong **vocabulary** “các loại thể thao” vào dạng content **Artice**

Lựa chọn 1 taxonomy term khi tạo 1 bài viết content

Create Article 

[Home](#) » Add content 

Loại Thể Thao

bóng chày 

điền vào thông tin về loại thể thao dựa vào danh sách sẵn xuống

Title *

bóng đá là gì?

Tags

football,bóng đá 

Enter a comma-separated list of words to describe your content.

Body (Edit summary)

bóng đá là môn bóng mỗi bên 11 người tranh giành nhau 1 quả bóng có găng sút bóng vào lưới đối phương

Khi truy nhập vào các bài liên quan tới term “**bóng đá**” nó sẽ liệt kê 1 dãy các bài viết liên quan

<http://localhost/drupal-7.7/bong-da>

bóng đá

[View](#) [Edit](#)

thông tin liên quan đến bóng đá

làm cách nào để giành chiến thắng

Submitted by [admin](#) on Wed, 01/18/2012 - 19:02

bóng đá là môn thể thao mang tính đồng đội rất cao và tuân thủ rất chặt chẽ nguyên tắc. tuy nhiên kỹ năng chơi bóng ở vài cá nhân xuất sắc cũng mang lại nhiều đột biến

luật chơi trong bóng đá

Submitted by [admin](#) on Wed, 01/18/2012 - 18:57

trong bóng đá không được dùng tay để chơi,.....

Tags:

[football](#) [rules](#)

Chú ý : trong mỗi term này đều được drupal tự động tạo ra 1 **RSS FEED** tương ứng

luật chơi trong bóng đá

Submitted by admin on Wed, 01/18/2012 - 18:57

trong bóng đá không được dùng tay để chơi,.....

Tags:

football rules

bóng đá là gì?

Submitted by admin on Wed, 01/18/2012 - 18:56

bóng đá là môn bóng mỗi bên 11 người tranh giành nhau 1 quả bóng.có gắng

Tags:

football bóng đá



<http://localhost/drupal-7.7/taxonomy/term/3/feed>

The screenshot shows a Drupal 7 node page. At the top, there is a yellow header bar with an orange RSS icon, the text "Subscribe to this feed using", and a dropdown menu "Live Bookmarks". Below this is a checkbox "Always use Live Bookmarks to subscribe to feeds." and a "Subscribe Now" button. The main content area has a title "localhost - bóng đá" and a subtitle "thông tin liên quan đến bóng đá". A purple link "làm cách nào để giành chiến thắng" is shown with a timestamp "Wednesday, January 18, 2012 7:02 PM". Below it is a text block about football. Another purple link "luật chơi trong bóng đá" is shown with a timestamp "Wednesday, January 18, 2012 6:57 PM". Below it is a text block about football rules. A section titled "Tags:" lists a single tag: "• [bóng đá](#)".

Các term phân cấp

Ví dụ ta muốn là 1 danh sách term phân cấp như sau :

- Bóng rổ
 - Giải phổ thông trung học
 - Giải đại học
 - Mùa giải 1
 - Mùa giải 2
 - Mùa giải 3
 - Giải nhà nghề chuyên nghiệp

Ta chú ý vào trong form **ADD TERM**, phần **Relations** để lựa chọn **Parent terms**

URL alias

giai-bong-ro-dai-hoc-mua-giai-1

Optionally specify an alternative URL by which this term can be accessed. Use a trailing slash or the URL alias won't work.

▼ RELATIONS

Parent terms

bóng rổ
-Giải phô thông trung học
-Giải đại học
bóng đá

Weight *

0

Terms are displayed in ascending order by weight.

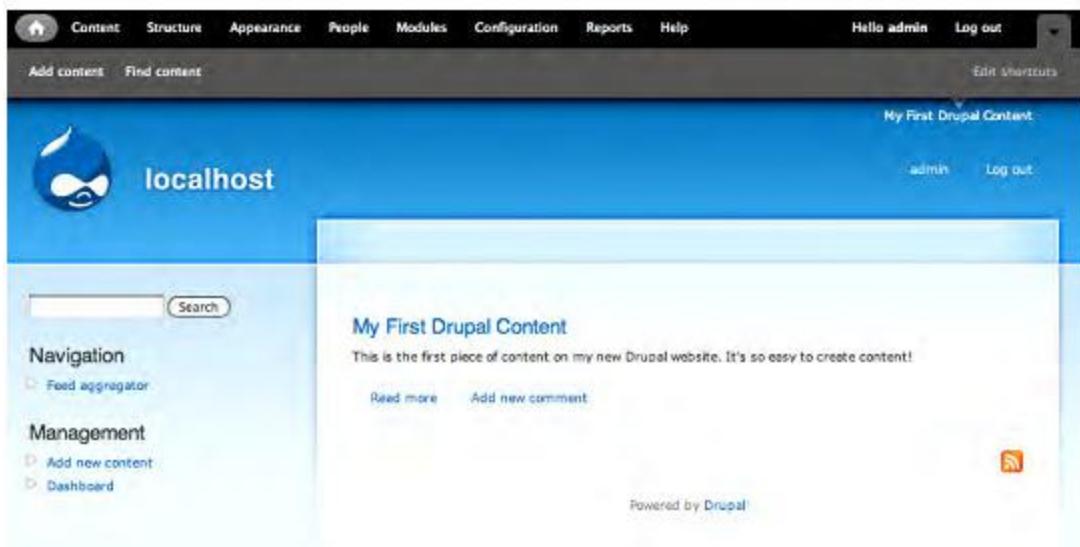
danh sách các terms phân cấp vừa thêm vào

NAME	WEIGHT	OPERATIONS
bóng bầu dục	0 <input type="button" value="▼"/>	edit
bóng chày	0 <input type="button" value="▼"/>	edit
bóng rổ	0 <input type="button" value="▼"/>	edit
Giải nhàng nghề chuyên nghiệp	0 <input type="button" value="▼"/>	edit
Giải phổ thông trung học	0 <input type="button" value="▼"/>	edit
Giải đại học	0 <input type="button" value="▼"/>	edit
Mùa giải 1	0 <input type="button" value="▼"/>	edit
Mùa giải 2	0 <input type="button" value="▼"/>	edit
Mùa giải 3	0 <input type="button" value="▼"/>	edit
bóng đá	0 <input type="button" value="▼"/>	edit

Gán nhiều hơn 1 vocabulary

Đơn giản là ta thêm 1 trường mới vào 1 dạng content type và trường này tham chiếu đến các terms thuộc 1 vocabulary khác

Tạo các thanh Menu



Nhìn vào hình minh họa ta sẽ thấy có 6 menu:

- thanh menu màu đen trên đầu bắt đầu với **Content**
- thanh menu màu nâu ở ngay dưới bắt đầu với **Add Content**
- thanh menu màu xanh nước biển với **Admin and Log out**
- the secondary menu nằm ở cột phía bên phải, bắt đầu với **Home**
- thanh menu ở phía tay trái có tên là **Management**
- the menu that is embedded in the article on the page, starting with “Create content.”

Có 3 cách cơ bản cho việc tạo ra các thành phần trên menu trong drupal:

1. thêm 1 phần tử nào đó vào 1 thanh menu đã tồn tại. drupal 7 bản thân đã có sẵn vài thanh menu. tất cả những gì ta cần là thêm các phần tử này vào thanh menu
2. tạo ra 1 thanh menu mới
3. gắn thanh menu vào 1 vùng nào đó trên trang web

thêm 1 phần tử vào thanh menu

thường có 2 loại phần tử ta có thêm chúng vào thanh menu của trang web:

- 1 link dẫn tới thành phần đã tồn tại trên trang web (ví dụ : bài viết content, 1 danh sách các bài viết content liên kết với 1 taxonomy term, ...)
- 1 link dẫn tới trang nằm ngoài website của chúng ta

Có 2 cách để thêm các phần tử vào thanh menu:

- Ta có thể sử dụng thanh menu admin để tạo ra 1 phần tử menu mới

- Ta có thể tạo 1 link liên kết menu từ chính thành phần content , nó nằm trong phần chỉnh sửa content

Cách thực hành tốt nhất là sử dụng ngay trong phần tạo ra bài viết content hay ngay trong phần tạo ra các thành phần khác.lí do để ta sử dụng cách này là vì khi ta xóa 1 bài viết content thì drupal cũng tự động xóa nó thì menu.nếu ta sử dụng cách làm thủ công là tạo ra phần tử menu bằng cách sử dụng thanh menu admin thì ta sẽ trở thành như người chủ của website và phải ghi nhớ để xóa bỏ các phần tử không tồn tại đó bằng tay từ thanh menu

Tạo menu theo cách 1 là chọn vào thẻ “Provide a menu”

The screenshot shows the 'Edit' screen for a content item titled 'My First Drupal Content'. On the left, there's a sidebar with several settings sections: 'Menu settings' (selected), 'Revision information' (No revision), 'URL path settings' (Alias: my-first-content-item), 'Comment settings' (Open), 'Authoring information' (By admin on 2010-03-13 09:20:28 - 0800), and 'Publishing options' (Published, Promoted to front page). On the right, under the 'Menu settings' section, there's a form with the following fields:

- Provide a menu link
- Menu link title: My First Drupal Content
- Parent item: <Main menu>
- Weight: 0

A note at the bottom states: "Menu links with smaller weights are displayed before links with larger weights."

Thêm 1 phần tử menu cho trang liên kết ngoài

Cách tạo:

Ấn vào **Structure >> Menus >> Add link** ở thanh menu cần thêm vào (ví dụ **Main menu**)



The screenshot shows the 'Structure' menu on a Drupal 7 site. The 'Menus' option is highlighted with a cursor icon pointing to it. Other options visible include 'Blocks', 'Content types', 'Taxonomy', and 'Administration'. The URL in the browser bar is 'localhost'.

Home » Administration » Structure » Menus

Menu link title *

Liên kết bán tranh

The text to be used for this link in the menu.

Path *

http://traditionalgallery.net

The path for this menu link. This can be an internal Drupal path such as `node/add` or a as `http://drupal.org`. Enter `<front>` to link to the front page.

Description

trang web bán tranh online

Shown when hovering over the menu link.

Enabled

Menu links that are not enabled will not be listed in any menu.

Show as expanded

If selected and this menu link has children, the menu will always appear expanded.

Tạo ra 1 thanh menu mới

Structure >> Menus >> Add menu

The screenshot shows the 'Menus' administration page with three menu items listed:

- Main menu**: Described as used for major site sections, often in a top navigation bar. Operations: list links, edit menu, add link.
- Management**: Contains links for administrative tasks. Operations: list links, edit menu, add link.
- Mạng Xã Hội**: Described as sharing links to various social networks. Operations: list links, edit menu, add link.

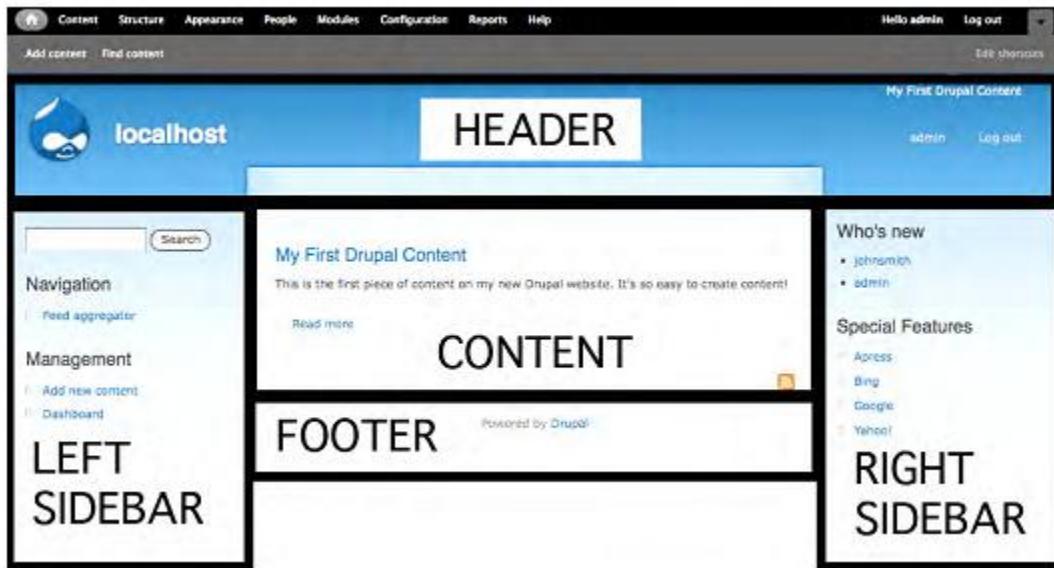
Muốn hiện thị thanh menu vừa tạo ta phải vào **Structure >> Blocks >>** tìm tới phần **Disabled** tới tên **mạng xã hội**

The screenshot shows the 'Blocks' administration page under the 'Footer' tab. In the 'Disabled' section, the 'Mạng Xã Hội' block is selected, indicated by a cursor icon over its name.

Block	Status	Action
Powered by Drupal	Footer	configure
Management	- None -	configure
Mạng Xã Hội	- None -	configure
Primary links	- None -	configure
Recent comments	- None -	configure
Recent content	- None -	configure
Secondary links	- None -	configure
Shortcuts	- None -	configure

Cài đặt themes

Các thành phần cơ bản của theme drupal 7 :



Cách thức làm việc của drupal theme

Ta có thể download theme ở : <http://drupal.org/project/themes> và <http://themegarden.org/>

Các bước cài đặt themes

ấn vào **Appearance >> Install new theme**



localhost

Appearance +

Home » Administration

! There was a problem checking available updates for Drupal. See the [available updates](#) page for more information and to install your missing updates.

Set and configure the default theme for your website. Alternative themes are available.

+ Install new theme ↗

ENABLED THEMES



Bartik 7.7 (default theme)
A flexible, recolorable theme with many regions.
[Settings](#)

localhost

Appearance +

Home » Administration » Appearance

You can find [modules](#) and [themes](#) on [drupal.org](#). The following file extensions are supported: *zip*, *tar*, *tgz*, *gz*, *bz2*.

Install from a URL

For example: <http://ftp.drupal.org/files/projects/name.tar.gz>

Or

Upload a module or theme archive to install
 [Browse...](#)
For example: *name.tar.gz* from your local computer

Install

chú ý: tới phần **Administration theme**

ADMINISTRATION THEME

Administration theme

Seven

Choose "Default theme" to always use the same theme as the rest of the site.

Use the administration theme when editing or creating content

Các cấu hình cài đặt

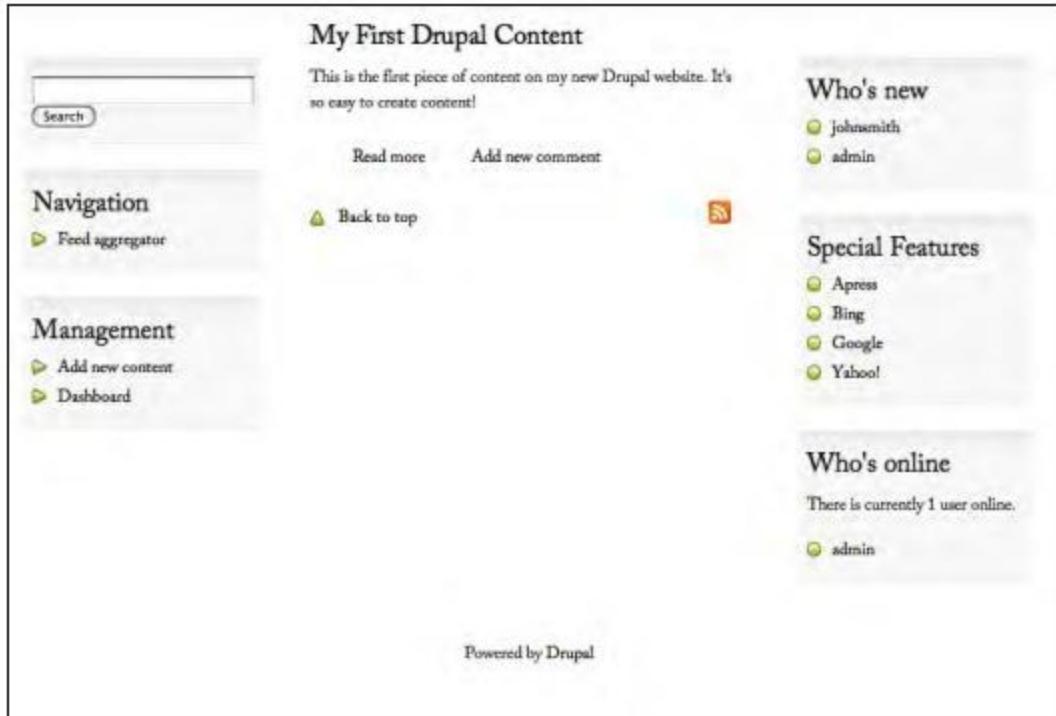


Các khối block trong drupal

Ta sử dụng các khối block để làm nơi chứa nội dung bài viết content trong đó. hay nó còn được gọi là **"widgets"**.(nó thường bao gồm : form dùng để đăng nhập, các bài viết blogs mới nhất, danh sách những thành viên hiện tại đang đăng nhập vào website, thời tiết,...) các widget này được xác định rõ 1 vị trí trên trang web

Quản lý hiện thị các block trên website

Ví dụ về các khối block sẽ xuất hiện trên trang web như sau:



Trên hình minh họa xuất hiện 7 khối block

Các block thuộc dạng menu bao gồm : **Navigation, Management, Special Features**

Khối block hỗ trợ : **Search**

Khối block thông tin : **"Who's new," "Who's online," and "Powered by Drupal"**

Để truy cập vào trang thể hiện 1 danh sách các block ta tiến hành theo như sau:

Structure >> Blocks

BLOCK	REGION	OPERATIONS
Header		
⊕ Main menu	Header	configure
Help		
⊕ System help	Help	configure
Highlighted		
<i>No blocks in this region</i>		
Featured		
<i>No blocks in this region</i>		
Content		
⊕ Main page content	Content	configure
Sidebar first		

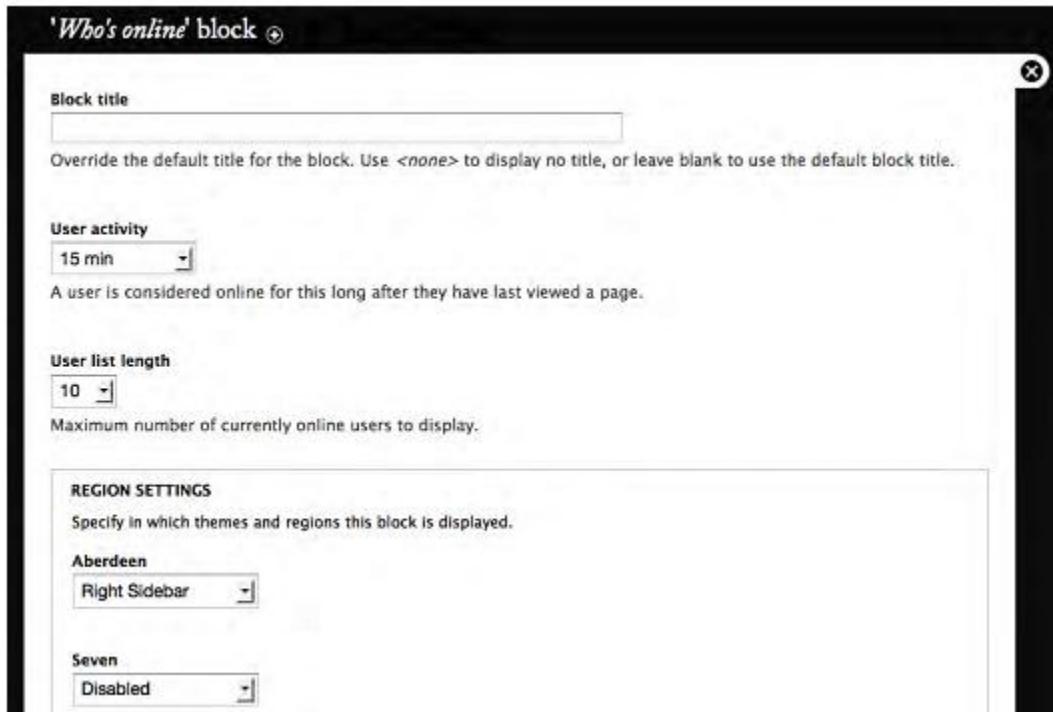
Việc sắp xếp hiện thị các khôi block khá là đơn giản ta chỉ cần dùng chuột kéo thả di chuột vào phần hiện thị tên block 

BLOCK	REGION	OPERATIONS
Header		
⊕ Main menu	Header	configure
Help		
⊕ System help*	Help	configure

Cấu hình các blocks

Ví dụ ta muốn cấu hình khối block “Who’s online” chỉ xuất hiện ở trang chủ ta thiết lập cấu hình theo các bước:

Structure >> Blocks >> xác định khối block có tên “Who’s online” >> configure



Chú ý vào phần **User activity** là thời gian users đăng nhập vào được ít nhất 15 phút thì mới đủ điều kiện nhìn thấy khối block này

Phần tiếp theo **User list length** là phần trở nên quan trọng khi trong 1 ứng dụng mạng xã hội, đây chính là số lượng thành viên hiện thị tối đa là 10 (ví dụ như tên và ảnh hiện thị thì được thể hiện tối đa là 10)

ở phần **Region settings** là cấu hình cài đặt vị trí hiện thị của block ở các theme khác nhau (theme user và theme admin)

ở phần **Visibility settings** là cấu hình liên quan đến việc hiện thị (khác với đặt vị trí hiện thị)

phân loại hiện thị theo loại đường dẫn URL

- muốn khối block hiện thị ở trong những trang nào đó ta lựa chọn vào thẻ **Pages** >> lựa chọn vào ô “**Only the listed pages**” (ví dụ ở đây ta muốn chỉ hiện thị ở trang chủ thì ta đánh ô text box vào : **<front>**)

- tuy nhiên khi muốn hiện thị ở những trang riêng biệt khác nên sử dụng URL của hệ thống (không phải URL đã được làm đẹp).các URL hệ thống :
 - **node/*** là trang hiện thị nội dung chi tiết 1 bài content nào đó (ví dụ node/1 thì block chỉ hiện thị ở trang có node id là 1) – URL đầy đủ **<http://localhost/drupal-7.7/node>**
 - **taxonomy/*** hay **taxonomy/term/** là trang hiện thị toàn bộ các bài viết khi chúng đã được phân nhóm vào trong term nào đó của vocabulary – URL đầy đủ **<http://localhost/drupal-7.7/taxonomy/term/>**
 - vấn đề với **URL alias** làm đẹp, sẽ trở nên khá tiện lợi tuy nhiên có sự thay đổi xảy ra nó sẽ dẫn tới việc link không chính xác.ví dụ: ta muốn hiện thị danh sách bài viết được gán nhãn là term “bóng đá” trong vocabulary “các loại thể thao”.URL đầy đủ **<http://localhost/drupal-7.7/taxonomy/term/3>** ta muốn hiện thì “block cơ bản” tại trang này



Ta truy cập vào block configure truyền vào tham số ***/term/3**

Visibility settings

Pages Restricted to certain pages	Show block on specific pages <input type="radio"/> All pages except those listed <input checked="" type="radio"/> Only the listed pages
Content types Not restricted	
Roles Not restricted	
Users Not customizable	

node/*
<front>
taxonomy/term/9
*/term/3

Specify pages by using their paths. Enter one path per line. The '*' character is a wildcard. Example paths are *blog* for the blog page and *blog/** for every personal blog. <front> is the front page.

khi ta tiến hành làm đẹp URL alias cho đường dẫn này thành <http://localhost/drupal-7.7/bong-da>

tất nhiên tham số **/term/3* ở trên vẫn còn giá trị hiện thị tại trang này.tuy nhiên đôi khi ta không biết được hoặc vì ta đã làm đẹp khá nhiều URL khác liên quan.lúc này ta thay đổi tham số là **bong-da**

Visibility settings

Pages Restricted to certain pages	Show block on specific pages <input type="radio"/> All pages except those listed <input checked="" type="radio"/> Only the listed pages
Content types Not restricted	
Roles Not restricted	
Users Not customizable	

node/*
<front>
taxonomy/term/9
bong-da 

Specify pages by using their paths. Enter one path per line. The '*' character is a wildcard. Example paths are *blog* for the blog page and *blog/** for every personal blog. <front> is the front page.

Phân loại hiện thị theo dạng content type

Có thể phân loại hiện thị dựa vào các dạng content type, khi không chọn gì cả thì nó không phân loại theo dạng content type nào (không lọc hiện thị)

Visibility settings

Pages Restricted to certain pages	Show block for specific content types <input type="checkbox"/> Article <input type="checkbox"/> Basic page
Content types Not restricted	Show this block only on pages that display content of the given type(s). If you select no types, there will be no type-specific limitation.
Roles Not restricted	
Users Not customizable	

Phân loại theo các nhóm Roles, khi không lựa chọn thì không lọc hiện thị

Visibility settings

Pages Restricted to certain pages	Show block for specific roles <input type="checkbox"/> anonymous user <input type="checkbox"/> authenticated user <input type="checkbox"/> administrator <input type="checkbox"/> restricted user <input type="checkbox"/> company user Show this block only for the selected role(s). If you select no roles, the block will be visible to all users.
Content types Not restricted	
Roles Not restricted	
Users Not customizable	

Phân loại tùy chỉnh hiện thị theo từng tài khoản user

Mỗi loại tùy chỉnh users có thể lựa chọn tùy chỉnh các khối blocks

Visibility settings

Pages Restricted to certain pages	Customizable per user <input checked="" type="radio"/> Not customizable <input type="radio"/> Customizable, visible by default <input type="radio"/> Customizable, hidden by default Allow individual users to customize the visibility of this block in their account settings.
Content types Not restricted	
Roles Not restricted	
Users Not customizable	

Các loại modules cho drupal

Đơn giản ta truy cập vào **Modules**



ấn vào “Install new module”

The screenshot shows the 'Modules' page in Drupal 7. At the top, there are three buttons: 'LIST', 'UPDATE', and 'UNINSTALL'. Below them is a search bar with the placeholder 'Search modules...'. Under the search bar, there's a link to 'Install new module'. The main content area has two sections: 'Install from a URL' with a text input field and 'Upload a module or theme archive to install' with a file input field and a 'Browse...' button. Both sections include examples of supported file types: 'zip', 'tar', 'tgz', 'gz', and 'bz2'. A large 'Install' button is located at the bottom left of the form.

Thiết lập cấu hình và thiết lập quyền hạn user trong modules

Không phải tất cả các module cung cấp những thiết lập như thế này

Ví dụ như thiết lập cấu hình và phân quyền cho module **views**

The screenshot shows the 'Views' configuration page. It lists two modules: 'Views' and 'Views UI'. The 'Views' module is enabled, while 'Views UI' is disabled. For the 'Views' module, there is a 'Permissions' link with a magnifying glass icon and a 'Configure' link with a gear icon. The 'Views UI' module also has a 'Configure' link with a gear icon. The table columns are: ENABLED, NAME, VERSION, DESCRIPTION, and OPERATIONS.

ENABLED	NAME	VERSION	DESCRIPTION	OPERATIONS
<input checked="" type="checkbox"/>	Views	7.x-3.0	Create customized lists and queries from your database. Requires: Chaos tools (enabled) Required by: Views content panes (enabled), Views UI (enabled)	Permissions 
<input type="checkbox"/>	Views UI	7.x-3.0	Administrative interface to views. Without this module, you cannot create or edit your views. Requires: Views (enabled), Chaos tools (enabled)	Configure 

Chọn ấn vào Permissions

PERMISSION	ANONYMOUS USER	AUTHENTICATED USER	ADMINISTRATOR	RESTRICTED USER	COMPANY USER
cancelling own account <i>Warning: Give to trusted roles only; this permission has security implications.</i>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Views Administer views Access the views administration pages.	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Bypass views access control Bypass access control when accessing views.	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Các modules cơ bản cần thiết

Blogging

Blogging là 1 chức năng được sử dụng khá phổ biến trên internet. blogging hiểu theo đơn giản là thao tác của việc ghi chép lại và xuất bản các suy nghĩ, ý tưởng của bạn về 1 chủ đề hay 1 nhóm các đối tượng liên quan đến nhau nào đó. blogging có thể xem như 1 dạng ghi chép cá nhân hay 1 dạng nhật ký

Bật module Blogs trong drupal

Ấn vào **Module >>** trong phần **Core module** tìm tới **Blog module**



<input type="checkbox"/>	Blog	7.7	Enables multi-user blogs.
<input type="checkbox"/>	Book	7.7	Allows users to create and organize related content in an outline.

Ta có thể thiết lập các nhóm roles và các quyền hạn cho module blog này. ví dụ ta tạo 1 nhóm roles mới là “**blogger**” và phân quyền cho nhóm roles này là :

- Create new Blog entry content.
- Edit own Blog entry content.
- Delete own Blog entry content.

Tạo 1 tài khoản **blog user** và phân tài khoản này vào nhóm roles **blogger**

People >> thẻ PERMISSIONS >> ROLES

The screenshot shows the Drupal 7 'Permissions' page under 'Administration > People > Permissions'. The 'Roles' tab is active. A table lists existing roles: 'anonymous user (locked)', 'authenticated user (locked)', 'administrator', 'restricted user', and 'company user'. At the bottom of the table, there is a text input field containing 'blogger' and a button labeled 'Add role' with a hand cursor icon pointing at it. The URL in the browser bar is 'localhost'.

NAME	OPERATIONS
anonymous user (locked)	edit permissions
authenticated user (locked)	edit permissions
administrator	edit role edit permissions
restricted user	edit role edit permissions
company user	edit role edit permissions
blogger	Add role

>> edit permissions (tại role **blogger**) >> chú ý ở phần **Node**

<i>Blog entry: Create new content</i>	<input checked="" type="checkbox"/>
<i>Blog entry: Edit own content</i>	<input checked="" type="checkbox"/>
<i>Blog entry: Edit any content</i>	<input type="checkbox"/>
<i>Blog entry: Delete own content</i>	<input checked="" type="checkbox"/>
<i>Blog entry: Delete any content</i>	<input type="checkbox"/>

Thêm vào user **bloguser**

People >> thẻ LIST >> Add user

Tạo 1 bài blog

Trước tiên ta đăng xuất ra khỏi tài khoản admin hiện tại và đăng nhập vào bằng tài khoản bloguser vừa tạo

Sau khi đăng nhập lài ấn **Add content (lưu ý là bloguser này đã được giới hạn ở quyền hạn)**



Create Blog entry

Title *

Body ([Edit summary](#))

More information about text formats [?](#)

Text format [Filtered HTML](#) [▼](#)

- Web page addresses and e-mail addresses turn into links automatically.
- Allowed HTML tags: `<a> <cite> <blockquote> <code> <dl> <dt> <dd>`
- Lines and paragraphs break automatically.

Hiện thị danh sách bài blog

Drupal sử dụng 1 định dạng URL riêng để hiện thị các bài viết blogs. ví dụ : <http://localhost/drupal-7.7/blog/7>

Từ “**blog**” sẽ thông báo cho drupal rằng nó sẽ hiện thị 1 danh sách các bài viết blog và “7” chính là userID (UID) là id của tài khoản viết các bài blog này

Danh sách các bài viết blog này được drupal sắp xếp theo thời gian tạo ra bài viết, các bài mới nhất được xếp nằm trên

Tạo khối hiện thị danh sách các bài blog mới nhất

[Structure](#) >> [Blocks](#) >> tìm tới “Recent blog posts”

The screenshot shows the 'Recent blog posts' configuration under the 'Sidebar second' section. It includes a dropdown menu set to 'Sidebar second' and a 'configure' link.

Recent blog posts

- Nguyen The Son
- Nguyễn Thị Tiên
- bloging là gì ?

[More](#)

Forums

Xét 1 ví dụ về Forums module trong drupal 7 core

The screenshot shows the 'Support' forum page with a list of topics:

Topic	Posts	Last post
Post installation	98777 2362 new	356784 4 min 40 sec ago by WolfSoul
Before you start	4868 89 new	20853 7 hours 26 min ago by calefilm
Installing Drupal	11748 135 new	50329 1 hour 59 min ago by Jer
Upgrading Drupal	4104 88 new	16654 1 hour 20 min ago by veganiatriathlete
Converting to Drupal	1991 44 new	7750 3 hours 47 min ago by .Sam.
Module development	24181 394 new	80153 43 min 39 sec ago by baal32
Theme development	14680 348 new	55387 1 hour 46 min ago by jtsevern
Translations	1227 34 new	3990 1 hour 32 min ago by MarkARgent

Thiết lập cài đặt Forum

Để bật Forums module : ấn vào **Modules >> tìm tới Forum module** (tất nhiên là tìm trong phần **Core**)

 Forum	7.7	Provides discussion forums. Requires: Taxonomy (enabled), Options (enabled), Field (enabled), SQL storage (enabled), Comment (enabled), Text (enabled)
--	-----	---

Các quyền có thể thiết lập cho module này là :

- Administer forums
- Create new forum topic content
- Edit own forum topic content
- Edit any forum topic content
- Delete own forum topic content
- Delete any forum topic content

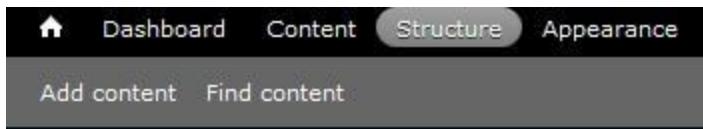
Chỉ tài khoản admin mới được gán quyền “**Administer forums,**” “**Edit any forum topic content,**” and “**Delete any forum topic content.**”

Nếu ta muốn những users bao gồm cả user vô danh có quyền đăng bài trong forums , ta cần phải bật các quyền “**Create new forum topic content,**” “**Edit own forum topic content,**” and “**Delete own forum topic content**” dành cho user vô danh hay user đã đăng ký, hay 1 nhóm các role mà ta đã định nghĩa

Ví dụ : ta phân quyền cho user đã xác thực : **Create, Edit, and Delete own forum topics**

Để tạo ra forums ta thực hiện theo 2 bước cơ bản:

- Tạo ra 1 nơi chứa forum và sau đó tạo ra các chủ đề để khách viếng thăm có thể tham gia vào.để tạo nơi chứa : ấn vào **Structure >> Forums** .tại đây ta có thể tạo ra nơi chứa mới và các topics mới.nơi chứa là nơi chứa đựng các topics và website của nó có thể gồm nhiều nơi chứa nếu cần thiết.theo ví dụ ở đây ta sẽ tạo ra 1 nơi chứa container có tiêu đề là “**nơi chứa container**” >> **Add container**



Forums localhost

LIST SETTINGS X

Home » Administration » Structure

Forums contain forum topics. Use containers to group related forums.

More help

Add container Add forum

Home » Administration » Structure » Forums

Container name *
nơi chứa container

Short but meaningful name for this collection of related forums.

Description
là 1 kho lưu trữ các forums và topics có liên quan đến nhau

Description and guidelines for forums within this container.

Parent *

Containers are usually placed at the top (root) level, but may also be placed inside another container or forum.

- Với container vừa tạo ra, giờ ta thêm forums để khách viếng thăm có thể đặt câu hỏi và phản hồi lại >> **Add forum**

Home » Administration » Structure



The forum container *nơi chứa container* has been updated.

Forums contain forum topics. Use containers to group related forums.



+ Add container + Add forum

Tạo ra 1 forum “**Tại sao lại học Drupal ?**” chứa trong 1 container “**nơi chứa container**”

ở đây chú ý rằng: 1 container là gói chứa nhiều forums bên trong nó, và bên trong mỗi forums sẽ chứa nhiều topics, ở trong topics là cấp con nhất mới là nơi trao đổi thông tin

Home » Administration » Structure » Forums

Forum name *

Tại sao lại học Drupal ?

Short but meaningful name for this collection of threaded discussions.

Description

nơi trao đổi các chia sẻ kinh nghiệm về câu hỏi muôn thủa giữa các CMS tại sao lại chọn Drupal

Description and guidelines for discussions within this forum.

Parent *

nơi chứa container ▾

Forums may be placed at the top (root) level, or inside another container or forum.

nơi chứa container

+ Add new Forum topic

Forum	Topics	Posts	Last post
Tại sao lại học Drupal ? nơi trao đổi các chia sẻ kinh nghiệm về câu hỏi muôn thửa giữa các CMS.tại sao lại chọn Drupal	1	1	By admin 1 min 17 sec ago
tết đi đâu chơi cho đỡ chán ? bàn luận các dự định đi chơi trong những ngày nghỉ tết	0	0	n/a

Lưu ý có 1 module phát triển gọi là “advanced forum” (http://drupal.org/project/advanced_forum).

Cung cấp các 1 số tính năng nâng cao như là “đóng mác mọi topics trong 1 forum hay tất cả các forum đã được đọc”, các thống kê như về số lượng topics, bài viết, users, user mới thoát, các user online hiện tại, hiện thị số bài viết mới,....

Web Forms

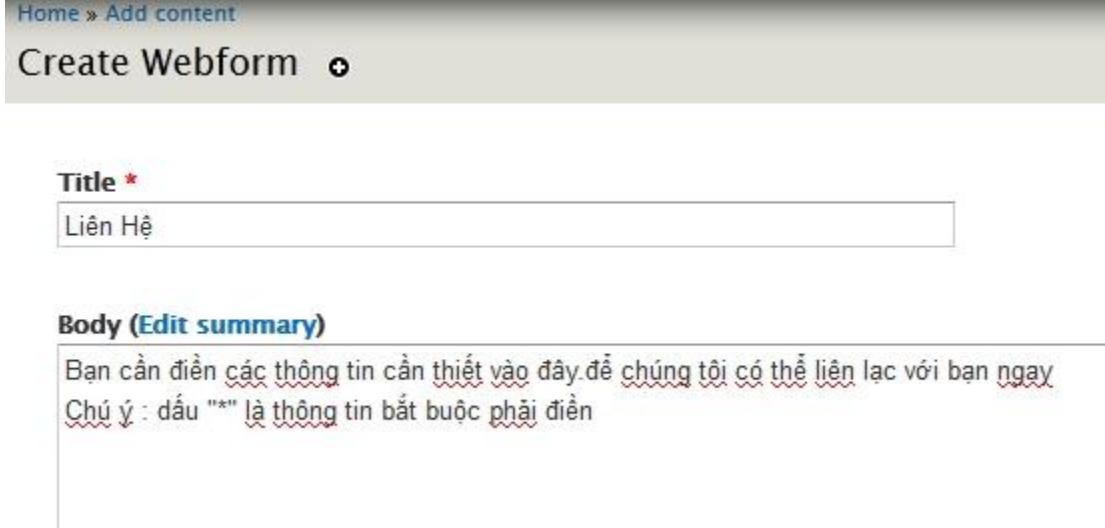
Tải module tại: <http://drupal.org/project/webform>

Ta có thể phân quyền “Edit own Webform submissions” cho user vô danh hay user đã xác thực nếu ta muốn những người dùng users này có thể cập nhật hay sửa lại thông tin mà họ đã gửi đi trước đó

PERMISSION	ANONYMOUS USER	AUTHENTICATED USER	ADMINISTRATOR	RESTRICTED USER	COMPANY USER	BLOGGER
Generally an administrative permission.						
Access own webform submissions	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Edit own webform submissions	<input checked="" type="checkbox"/> 	<input checked="" type="checkbox"/> 	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Delete own webform submissions	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Tạo ra 1 webform

Ấn vào **Content > Add new content** > chọn vào dạng **Webform** > ta điền tiêu đề của Webform ở phần **Title** và viết vài dòng chú thích vào trong **Body**



Home » Add content

Create Webform •

Title *

Liên Hệ

Body (Edit summary)

Bạn cần điền các thông tin cần thiết vào đây để chúng tôi có thể liên lạc với bạn ngay.
Chú ý: dấu "*" là thông tin bắt buộc phải điền

Sau khi **Save**, ở trang tiếp theo là nơi ta tạo ra các thành phần của webform

Chú ý các thiết lập lựa chọn cho các trường thêm vào:

Label : nhãn tên trường

Field key : là tên trường bên trong database

Default value : ta có thể cài đặt sẵn giá trị mặc định cho trường dữ liệu, có thể sử dụng các tham số hệ thống cho các giá trị mặc định này (dựa vào các tham số được chú thích sẵn dưới **TOKEN VALUES**)

Default value

`%username`

The default value of the field.

TOKEN VALUES

You may use special tokens in this field that will be replaced with dynamic values.

Basic tokens

- `%username` – The name of the user if logged in. Blank for anonymous users.
- `%useremail` – The e-mail address of the user if logged in. Blank for anonymous users.
- `%ip_address` – The IP address of the user.
- `%site` – The name of the site (i.e. `localhost`)
- `%date` – The current date, formatted according to the site settings.

Node tokens

- `%nid` – The node ID.
- `%title` – The node title.

Special tokens

- `%profile[key]` – Any user profile field or value, such as `%profile[name]` or `%profile[profile_first_name]`
- `%get[key]` – Tokens may be populated from the URL by creating URLs of the form `http://example.com/my-form?foo=bar`. Using the token `%get[foo]` would print "bar".
- `%post[key]` – Tokens may also be populated from POST values that are submitted by forms.

In addition to `%get` and `%post`, the following super tokens may be used, though only with logged-in users: `%server`, `%cookie`, and `%request`. For example `%server[HTTP_USER_AGENT]` or `%session[id]`.

Description : phần chú thích mô tả rõ thêm cho trường dữ liệu

Validation : là phần thêm điều kiện hợp lệ vào trường dữ liệu.nếu muốn trường bắt buộc phải điền ta phải đánh dấu vào ô **mandatory** hay muốn trường này là duy nhất trong các dữ liệu thu thập (ví dụ mỗi user chỉ có duy nhất 1 email xác định user đó) thì ta đánh dấu vào ô **Unique** hay đưa ra chuỗi kí tự dài nhất có thể nhập vào ô **Maxlength**

▼ **VALIDATION**

Mandatory

Check this option if the user must enter a value.

Unique

Check that all entered values for this field are unique. The same value is not allowed to be used twice.

Maxlength

Maximum length of the textfield value.

Display width and height: phụ thuộc vào loại trường dữ liệu mà ta lựa chọn (**textbox or textarea**) giúp ta cài đặt tùy chỉnh chiều rộng và chiều cao của 1 trường dữ liệu khi hiện thị

Resizable: cung cấp khả năng có thể thay đổi kích thước của trường dữ liệu bằng việc kéo chuột

Disabled: thiết lập ngăn cản user thay đổi giá trị mặc định của trường dữ liệu

Private: giới hạn hiện thị, trường dữ liệu này chỉ hiện thị ra đối với tài khoản user với quyền truy cập vào “**results access**” – “**access all Webform results**”

Label display

Determines the placement of the component's label.

Disabled

Make this field non-editable. Useful for setting an unchangeable default value.

Private

Private fields are shown only to users with results access.

Phân quyền để hiện ra trường đánh dấu **private** trong Webform “**access all Webform results**”

PERMISSION	ANONYMOUS USER	AUTHENTICATED USER	ADMINISTRATOR	RESTRICTED USER	COMPANY USER	BLOGGER
Webform						
Access all webform results	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Grants access to the "Results" tab on all webform content. Generally an administrative permission.						
Access own webform results	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Grants access to the "Results" tab to the author of webform content they have created.						

Đối với kiểu trường **Select** cho phép lựa chọn nhiều giá trị nếu chọn vào ô **Multiple**. Kiểu hiện thị mặc định sẽ là các ô **radio** khi tích vào ô **listbox** thì kiểu hiện thị là kiểu **xổ xuống**

Multiple
Check this option if the user should be allowed to choose multiple values.

Options 
 Sales|Sales
 Service|Service
 Support|Support
 Other|Other

Key-value pairs **MUST** be specified as "safe_key|Some readable option". Only alphanumeric characters and underscores are allowed as a key. One option per line. Option groups may be specified with <Group Name>, <> can be used to insert items at the root of the menu after specifying a group.

 **Token values**

Load a pre-built option list
 Use a pre-built list of options rather than entering options manually. Options will not be editable if using pre-built list.

Validation
 Mandatory
Check this option if the user must enter a value.

 **Display**

Listbox
Check this option if you want the select component to be of listbox type instead of radio buttons or checkboxes.

Randomize options
Randomizes the order of the options when they are displayed in the form.

Cuối cùng là thiết đặt **Form settings**



Tại đây ta thiết lập thông báo thành công khi khách gửi thành công thông tin

Views

Các chức năng của module View:

- Hiện thị ra các bài viết mới nhất và sắp xếp chúng theo giảm dần theo thời gian đăng bài
- Hiện thị 1 danh sách các công ty
- Hiện thị 1 gallery ảnh
- Hiện thị 1 danh sách các bài viết được lọc theo 1 đối tượng nào đó (điều kiện)
- Tạo ra 1 RSS FEED

Tạo View

Ta cần tạo ra 1 hàng loạt các bài viết content nhằm làm thí dụ cho việc tạo ra View

Ấn vào **Structure >> Views >> Add new view**

chú ý vào các thiết lập sau:

View name: tên này phải là duy nhất (giống như kiểu ID), giống như kiểu tên biến chứa các kí tự chữ cái và dấu gạch dưới ,không có dấu cách.ví dụ ta đặt tên là **upcoming_events**

View description: thông tin mô tả thêm về view đang tạo

View tag: phân nhóm view này vào 1 chủ đề hay 1 nhóm chủ đề nào đó

View type: 1 danh sách các dạng của các thông tin được lưu giữ trong database dựa vào đó mà ta có thể tạo ra các View.các lựa chọn thông thường lừa là dạng **Node**

Modify the display(s) of your view below or add new displays

Displays

The screenshot shows the 'Displays' section of the 'demo view' configuration. The 'Page*' display is currently selected. A context menu is open at the top right of the 'Page' display area, with the 'Add' option highlighted. Other options in the menu are Attachment, Block, Content pane, Context, Feed, and Page. The title of the view is 'Title: demo view'.

Ta cần hiểu rõ các từ khóa:

- **Attachment** : được sử dụng để tạo View hay được sử dụng như là dạng hiện thị thứ 2 được thêm vào trong view thứ 1.hơn thế đây là cách đơn giản để tạo ra nhiều View trong phạm vi cùng 1 View.đây là 1 đặc tính cao cấp ta có thể tìm hiểu thêm ở www.drupal.org/project/views
- **Block** : là khối chứa nội dung content ở bên trong được gán trong 1 vùng trên trang Web
- **Feed** : là cơ chế tự động tạo ra RSS feeds
- **Page** : là 1 View được diễn tả trong chế độ toàn trang full-page và được truy cập thông qua 1 URL

Các cài đặt trong Views

Các cài đặt cơ bản

TITLE	PAGE SETTINGS
Title: demo view	Path: /demo-view
FORMAT	Menu: No menu
Format: Unformatted list Settings	Access: Permission View published content
Show: Fields Settings	
FIELDS	HEADER add
Content: Title	FOOTER add
	PAGER
FILTER CRITERIA	Use pager: Full Paged, 10 items
Content: Published (Yes)	More link: No
SORT CRITERIA	
Content: Post date (desc)	

Name : ta có thể thiết lập tên name cho View hơn là tên mặc định sẵn của nó.để thay đổi name, ta nhấn vào tên mặc định và thay đổi giá trị của nó thành tên mong muốn.giá trị điền vào ở đây chỉ hiện thị cho thành viên user và không được hiện thị ra cho khác viếng thăm

▼ Page details

Display name: Page

Title : được hiện thị ở trên đầu các danh sách các Views.ta nên thay đổi thành từ có nghĩa để khách viếng thăm có thể hiểu được View này nói về cái gì

Use Pager : được sử dụng cho View có chứa hàng tá đối tượng.ta có thể giới hạn số hiện thị các đối tượng này xuất hiện trên 1 trang (mặc định giới hạn là 10).có 4 lựa chọn hiện thị phân trang pager

PAGER
Use pager: Mini <input checked="" type="radio"/> Mini pager, 3 items
More link: No

For

- Display a specified number of items
- Display all items
- Paged output, full pager
- Paged output, mini pager

You may also adjust the [settings](#) for the currently selected pager.

1. “Display a specified number of items” hiện thị đúng số đối tượng giới hạn mà không phân trang
2. “Display all items” hiện thị toàn bộ đối tượng mà không phân trang
3. “Paged output, full pager” thực hiện phân trang với các chức năng đầy đủ bao gồm “Jump to the start of the list,” “Jump to the end of the list,” “Previous,” and “Next.”
4. “Paged output, mini pager” thực hiện phân trang với 2 chức năng rút gọn “Previous,” and “Next.”

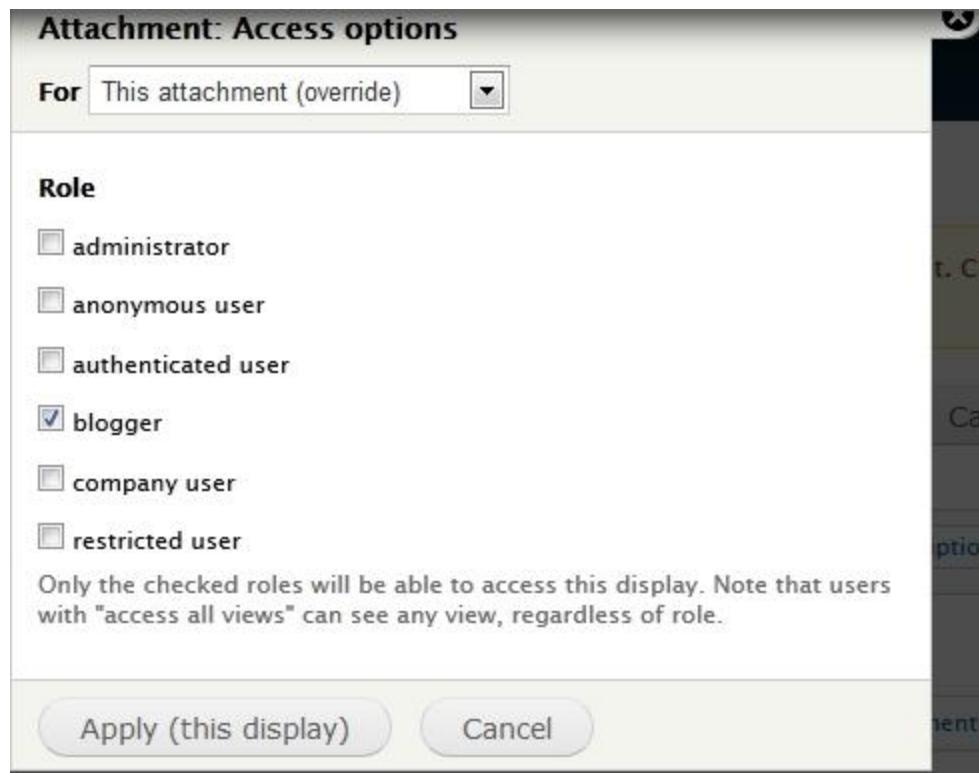
Items Per Page : giới hạn số lượng đối tượng hiện thị trên mỗi trang. **Offset :** (độ dịch) ví dụ như ta không muốn hiện thị 10 đối tượng đầu tiên mà muốn hiện thị bắt đầu từ đối tượng thứ 11 thì ta điền vào Offset giá trị là 10.

The screenshot shows the 'Defaults: Pager options' configuration page. It includes fields for 'Items per page' (set to 10), 'Offset' (set to 0), 'Pager ID' (set to 0), and 'Number of pages'. Below these, there's an 'EXPOSED OPTIONS' section with two checkboxes: 'Expose items per page' (unchecked) and 'Expose Offset' (unchecked). At the bottom are 'Update' and 'Cancel' buttons.

More link : nếu view bị giới hạn bởi trường **items-per-page** là 1 số nhỏ hơn số mà ta đã định nghĩa cho view, và ta không quyết định sử dụng phân trang pager, ta có thể tạo ra liên kết “**Read more**”, liên kết mà khách viếng thăm sẽ ấn vào để xem được toàn bộ danh sách các đối tượng

Page settings : chú ý tới phần **Access** cho phép ta định nghĩa dạng khách viếng thăm có quyền truy cập vào View bằng cách lựa chọn từ danh sách các nhóm **Roles** hay phân quyền **permissions**

The screenshot shows the 'PAGE SETTINGS' configuration page. It displays the 'Path' as /demo-view, 'Menu' as 'No menu', and 'Access' as 'Permissions'. A cursor is hovering over the 'View published content' link under the 'Access' section.



Các cài đặt nâng cao

Use Ajax : sử dụng Ajax trong phân trang , trong sắp xếp table, hay trong chế độ lọc **exposed** (để khách viếng thăm cũng có thể tùy chỉnh)

OTHER

Machine Name: [page](#)

Comment: [No comment](#)

Use AJAX: [No](#)

Hide attachments in summary: [No](#)

Use aggregation: [No](#)

Query settings: [Settings](#)

Field Language: [Current user's language](#)

Caching: [None](#)

CSS class: [None](#)

Theme: [Information](#)

Distinct : trong nhiều trường hợp View trả về nhiều đối tượng trùng lặp (hiếm gặp).để loại bỏ các đối tượng trùng lặp ta chuyển **Use aggregation: No** thành **Yes** và **Query settings**

Caching : có thể có trường hợp khi View trả về 1 số lượng lớn các đối tượng nên cần mất nhiều thời gian hơn để giảm thiểu thời gian này ta nên bật caching

CSS Class : là 1 đặc tính cho phép ta gán thuộc tính class cho view

Theme : hiện thị các file trong template, chú ý các file in đậm

Header và Footer : cho phép ta đặt 1 đoạn text hay 1 trường dữ liệu nào đó vào header hay footer của View

Relationships : cung cấp khả năng liên kết tới với các thông tin liên quan. ví dụ: ta có 1 loại content type cho 1 nhân viên, và trong đó có 1 trường dữ liệu tham chiếu tới vị trí của nhân viên. thông tin về vị trí này được lưu trữ trong các bài content về vị trí. và yêu cầu của View là gọi ra danh sách các nhân viên và địa chỉ vị trí của họ. bởi vì vị trí của nhân viên không được lưu trữ trong bài content về nhân viên, nên ta cần 1 phương thức liên kết cả 2 dạng content type để cùng hiện thị thông tin về nhân viên và vị trí nhân viên trong 1 đối tượng trong View. để tìm hiểu thêm về cơ chế này ta vào trang

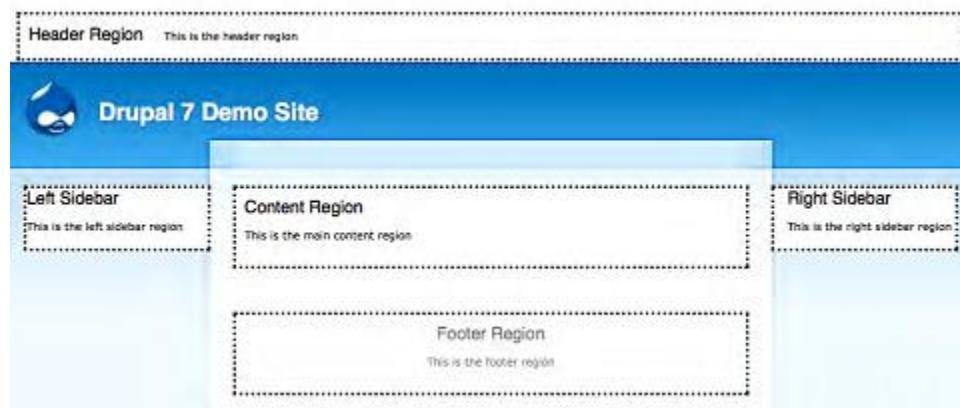
www.drupal.org/project/views

Cụ thể hơn : nó cho phép ta kết nối 1 node với 1 node thứ 2 khi nó sử dụng module CCK **node_reference module or user_reference module**

Arguments : cung cấp cơ chế chuyển tiếp được tham số hiện thị vào View thông qua URL. Views sẽ được lọc theo tham số truyền vào qua URL này

Panels

Các bố cục trang bị ràng buộc bởi các vùng xác định bởi theme mà ta đang sử dụng. ví dụ : nếu ta sử dụng theme Garland thì nhìn vào ảnh phía dưới : nhận thấy có 3 vùng cơ bản **the left sidebar, the right sidebar, and the center content area** (chú ý : ta loại bỏ 2 vùng **the footer and header** vì những vùng này không thường dùng để đặt nội dung content thường chứa các đoạn text)



Vùng chính được sử dụng ở theme trên là vùng **the center content region**, trong nhiều trường hợp ta phải phân chia **the content region** thành nhiều vùng nhỏ hơn “**mini-regions**.”

Chú ý tới ví dụ sau, ta có 7 ô phân chia “**panel panes**” (hay còn gọi là vùng “**mini-regions**.”) được chia ra vào trong 6 dòng:

- Dòng trên cùng có 1 ô panel pane là nơi chứa các thẻ điều hướng cho các loại sách, phim, âm nhạc books, movies, and music.
- Dòng thứ 2 có 1 ô panel pane dành cho slideshow hay được gọi là “Featured Picks.”
- Dòng thứ 3 có 2 ô panel pane, 1 ô ở bên trái có tiêu đề là “Featured” và nó hiện thị các đối tượng đặc sắc ở thư viện, 1 ô ở bên phải hiện thị các truyện story đặc sắc
- Dòng thứ 4 bao gồm 1 thanh banne chạy thông báo tin tức
- Dòng thứ 5 là 1 thanh slideshow khác
- Dòng thứ 6 hiện thị lựa chọn ngôn ngữ cho trang web

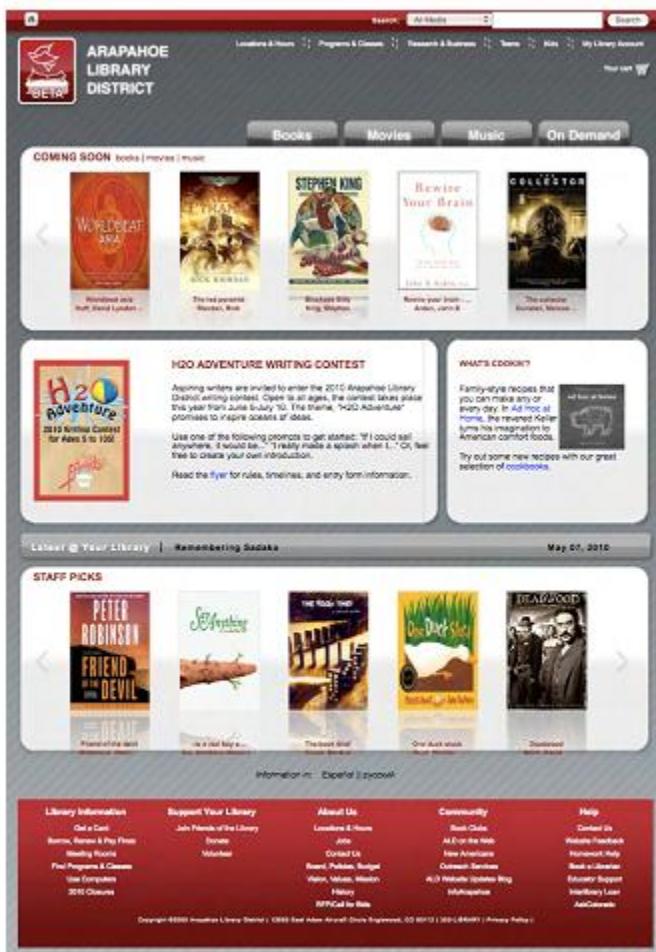


Illustration 12-1. A site using a complex panels layout

Ví dụ thứ 2 đơn giản hơn 1 chút



Illustration 12-2. A simple two row, three column panels layout

ở trên vùng the content region được chia thành 2 dòng và mỗi dòng chia thành 3 cột

khi tạo ra 1 website, ta phải sử dụng Views cho mỗi ô panel pane để định nghĩa nội dung content sẽ được hiện thị ở trong đó.sử dụng Taxonomy , lựa chọn các taxonomy term tương ứng cho mỗi ô panel pane.

Các thiết kế bố cục Layout trong Panels module

Panels module cung cấp 8 thiết kế bố cục Layout tiêu chuẩn và 1 “flexible” layout

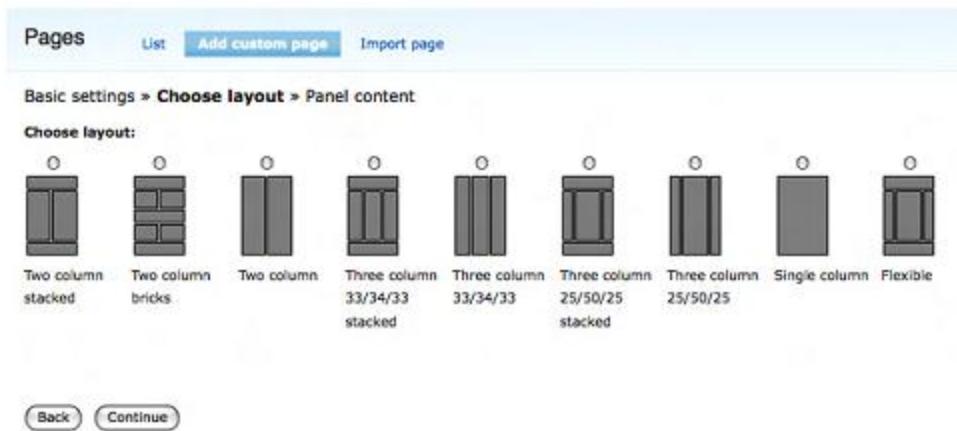


Figure 12-2. Panel layouts

Tạo ra 1 trang panel

Nếu 1 trong 8 bố cục Layout cơ bản không phù hợp, ta có thể phải tạo ra 1 layout tùy chỉnh dựa vào flexible layout. ví dụ : giả sử ta muốn chia the content region thành 3 dòng với dòng đầu tiên có 1 cột, dòng thứ 2 có 2 cột và dòng thứ 3 có 1 cột

Ấn vào **Structure >> Panels >> creating a new panel page**

- **Administrative title:** tiêu đề xuất hiện trong trang admin – trang hiện thị tất cả danh sách các trang panels mà ta đã tạo ra.(không được hiện thị cho khách viếng thăm)
- **Machine name:** tên drupal lưu trong cơ sở dữ liệu bên trong dùng để tham chiếu tới trang panel
- **Administrative description:** thông tin mô tả thêm
- **Path:** đường dẫn URL để truy cập tới trang panel
- **Variant type:** (dạng biến thể) ta chọn dạng panel
- **Optional features:** cho phép ta bật các tính năng : quản lý truy cập access control (giới hạn những ai có thể thấy trang panel này), Visible menu item (đặt 1 liên kết link tới trang panel vào menu), Selection rules (lựa chọn quy tắc) và contexts (phạm vi) là 2 đặc tính nâng cao ta có thể tìm hiểu thêm tại <http://drupal.org/project/panels>.

Pages List Add custom page Import page

Administrative title:
about us panel page
The name of this page. This will appear in the administrative interface to easily identify it.

Machine name:
about_us_panel_page
The machine-readable name of this page. It must be unique, and it must contain only alphanumeric characters and underscores. Once created, you will not be able to change this value!

Administrative description:
This is the panel page that is used for the about us page.

A description of what this page is, does or is for for administrative use.

Path:
http://localhost/drupal/about-us
The URL path to get to this page. You may create named placeholders for variable parts of the path by using %name for required elements and %name for optional elements. For example: "/node/%node/%", "%category/forum" or "dashboard/%page". These named placeholders can be turned into contexts on the arguments form.

Make this your site home page.
To set this panel as your home page you must create a unique path name with no % placeholders in the path. The site home page is currently set to /node on the Site Information configuration form.

Variant type:
Panel

Optional features:

Access control
 Visible menu item
 Selection rules
 Contexts
Check any optional features you need to be presented with forms for configuring them. If you do not check them here you will still be able to utilize these features once the new page is created. If you are not sure, leave these unchecked.

[Continue](#)

Figure 12-3. Creating a new panel page

Ấn vào **Continue** ở trang tiếp theo là nơi ta lựa chọn dạng bố cục layout .theo ví dụ trên ta chọn layout **Two column stacked**

The screenshot shows the 'Choose layout' step of a page configuration wizard. At the top, it says 'Basic settings » Choose layout » Panel settings » Panel content'. Below that, there's a 'Category' section with a dropdown menu set to 'Columns: 2'. Underneath, there are three layout options labeled 'Two column', 'Two column bricks', and 'Two column stacked'. The 'Two column stacked' option has a blue circular icon with a white dot and a small hand cursor icon pointing at it, indicating it is selected. At the bottom are 'Back' and 'Continue' buttons.

ấn tiếp **Continue**

ấn tiếp **Continue** sẽ tới trang mà ta gán các thứ vào các ô panel panes trên trang panel

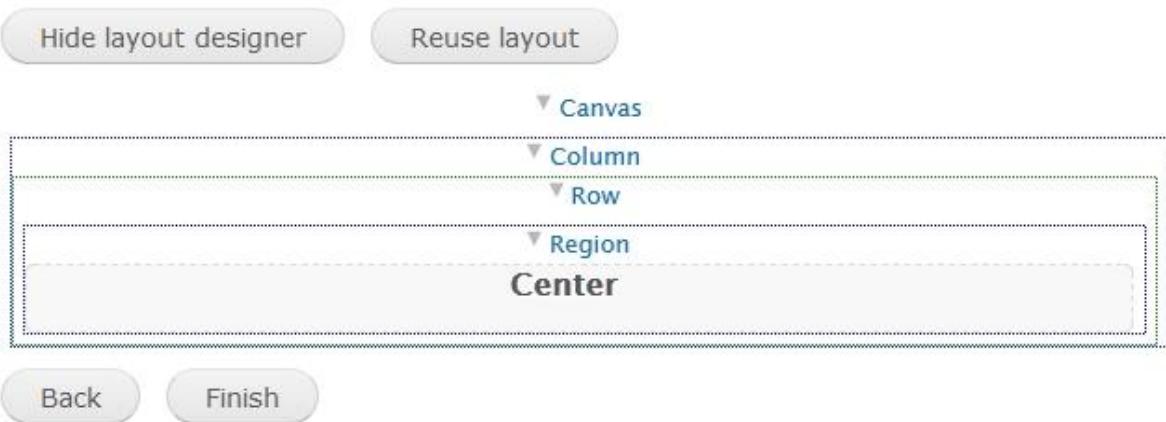
The screenshot shows the 'Panel content' step of the page configuration wizard. It displays a 'Title type' dropdown set to 'Manually set'. Below it is a 'Title' input field containing 'About Us'. A note below the title says: 'The title of this panel. If left blank, a default title may be used. Set to No Title if you want the title to actually be blank.' There are four dashed rectangular boxes representing panel panes: 'Top', 'Left side', 'Right side', and 'Bottom'. Each box has a small circular icon with a white dot. At the bottom are 'Back' and 'Finish' buttons.

Sử dụng bố cục layout tùy chỉnh Flexible Layout

The screenshot shows the 'Basic settings' step of the 'Choose layout' wizard. The path in the breadcrumb is: Home » Administration » Structure » Pages. The title of the page is 'Basic settings » Choose layout » Panel settings » Panel content'. A dropdown menu labeled 'Category' is set to 'Builders'. Below it, a section titled 'Builders' shows a single item named 'Flexible' with a thumbnail icon. At the bottom are 'Back' and 'Continue' buttons.

The screenshot shows the 'Panel content' step of the 'Choose layout' wizard. The path in the breadcrumb is: Home » Administration » Structure » Pages. The title of the page is 'Basic settings » Choose layout » Panel settings » Panel content'. A dropdown menu labeled 'Title type' is set to 'Manually set'. A text input field for 'Title' contains the text 'thiết kế layout tùy chỉnh'. A note below says: 'The title of this panel. If left blank, a default title may be used. Set to No Title if you want the title to actually be blank.' At the bottom are 'Show layout designer', 'Reuse layout', and 'Center' buttons, along with 'Back' and 'Finish' buttons.

Chọn "Show layout designer"



Theming

Các file drupal theme tiêu chuẩn

Drupal themes bao gồm 1 vài file, tất cả chúng được chứa trong thư mục có cùng tên với tên của theme /sites/all/themes

file	Thông tin mô tả
<themename>.info	Mỗi drupal theme phải có 1 file .info, và tên của nó phải trùng với tên của theme.ví dụ trong trường hợp là Genesis theme thì file này có tên là genesis.info.trong file này ta phải định nghĩa các thuộc tính sau : <ul style="list-style-type: none"> • Tên của theme (e.g., Genesis) • Phiên bản drupal hỗ trợ (e.g., 7.x) • Tên và vị trí tất cả các file css cần thiết của theme • Tên và vị trí các file JS được sử dụng trong theme • Tên các vùng regions được sử dụng trong theme
page.tpl.php	Mỗi drupal theme cần phải có 1 file page.tpl.php.file này chứa hỗn hợp mã HTML và ma PHP và nó định nghĩa bố cục layout toàn cục và cấu trúc của trang page trong website.file này cũng mang tới các thành phần key Drupal core khác thông qua 1 số các file được nhúng vào.mã trong file có dạng như : <?php print \$somevariable; ?>.ví dụ ta tìm cấu trúc mã <?php print \$styles; ?>.đoạn mã này sẽ lấy toàn bộ file CSS đã được định nghĩa trong file .info vào trong trang page.ta tập trung vào cách thức định nghĩa các vùng regions trong file .info và cách thức các thuộc tính CSS được áp dụng vào các vùng regions này.ví dụ như đoạn mã <?php print render(\$page['footer']); ?> Để tìm hiểu thêm về page.tpl.php ta xem thêm ở trang drupal.org/theme-guide
node.tpl.php	Định nghĩa cách các nodes được bố trí khi hiện thị trên website.ví dụ như : sự bố trí hiện thị của the title, thông tin về the author, the

	taxonomy terms, the body, and pictures được nhúng vào trong node
html.tpl.php	Bao gồm tất cả các phần tử nằm trong phần <head>...</head> của 1 trang HTML
region.tpl.php	Định nghĩa cách các vùng regions được bố trí và biểu diễn trên website
block.tpl.php	Định nghĩa cách các khối block được bố trí và biểu diễn trên website
field.tpl.php	Định nghĩa cách các trường dữ liệu fields được bố trí và biểu diễn trên website
style.css	1 Drupal theme phải có ít nhất 1 file style.css

Ngoài ra cũng còn những file tùy chọn khác trong thư mục theme. Những theme files thường là file có dạng .tpl cho từng phần riêng biệt (ví dụ như phần block)

Giải thích các file template PHP

The html.php.tpl File

Nó hiện thị cấu trúc HTML căn bản của 1 trang page trên Drupal site. file này tập trung vào các phần tử nằm trong thẻ mở <HTML> và bắt đầu của thẻ <body>. trong đoạn mã ở dưới, ta có thể nhận thấy rằng html.tpl.php file cung cấp các phần tử giống như định nghĩa DOCTYPE, các định nghĩa RDF, HTML, 1 vài thẻ DIV, 1 vài đoạn nhỏ mã PHP dùng để in ra nội dung content chứa trong các biến variables (được giải thích rõ trong bảng phía dưới)

Xét đoạn mã sau :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
 "http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang=<?php print $language->language; ?>" version="XHTML+RDFa 1.0" dir=<?php print $language->dir; ?>"<?php print $rdf_namespaces; ?>>

<head profile=<?php print $grddl_profile; ?>>
<?php print $head; ?>
<title><?php print $head_title; ?></title>
<?php print $styles; ?>
<?php print $scripts; ?>
</head>
<body class=<?php print $classes; ?>> <?php print $attributes;?>>
<div id="skip-link">
<a href="#main-content"><?php print t('Skip to main content'); ?></a>
</div>
<?php print $page_top; ?>
<?php print $page; ?>
<?php print $page_bottom; ?>
```

```
</body>
</html>
```

Các biến được sử dụng trong html.tpl.php File

Biến	Thông tin mô tả
\$css	1 mảng các files CSS dành cho trang hiện tại
\$language	(đối tượng) loại ngôn ngữ website sử dụng
\$language->language	Chứa trình bày bản dịch của nó
\$language->dir	Bao gồm chỉ dẫn ngôn ngữ
\$rdf_namespaces	Tất cả các tiền tố RDF namespace được sử dụng trong HTML document
\$grddl_profile	1 GRDDL profile cho phép các tác nhân trích xuất ra các dữ liệu RDF
\$head_title	1 phiên bản đã được biến đổi của page title, được sử dụng trong thẻ TITLE
\$head	Đánh dấu trong phần HEAD (bao gồm các thẻ meta, thẻ keyword, ...)
\$styles	Các thẻ style cần thiết để nhập vào tất cả các CSS files
\$scripts	Các thẻ Script cần thiết để nạp các JavaScript files
\$page_top	Khởi tạo đánh dấu từ bất cứ modules nào biến đổi trang page, biến này luôn luôn xuất ra đầu tiên
\$page	Là nội dung content của trang page biểu diễn; drupal sẽ thay thế \$page bằng nội dung trong page.tpl.php
\$page_bottom	Đóng sự đánh dấu từ bất cứ modules nào, biến này luôn luôn xuất ra sau cùng
\$classes	Chuỗi các class được sử dụng để tạo CSS.mặc định biến \$classes bao gồm các class : html front logged-in one-sidebar sidebar-first page-node toolbar toolbar-drawer

The page.tpl.php File

File này tập trung vào các phần tử nằm trong <body> and </body> và bao gồm cấu trúc HTML của trang page, bao gồm các thẻ DIV và những đoạn nhỏ mã PHP.nhìn lại file html.tpl.php , ta lưu ý tới đoạn mã

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

Giá trị trong biến \$page chính là nội dung content trong file page.tpl.php

Ví dụ như biến \$logo giữa các mã HTML cần thiết để hiển thị logo của website nếu nó được thiết lập.chú ý tới điều kiện if sẽ kiểm tra xem logo có được thiết lập hay k?

Cấu trúc page.tpl.php file được chứa trong thư mục /modules/system như sau:

```
<div id="page-wrapper"><div id="page">

    <div id="header"><div class="section clearfix">

        <?php if ($logo): ?>
            <a href=<?php print $front_page; ?>" title=<?php print t('Home'); ?>" rel="home"
                id="logo">
                <img src=<?php print $logo; ?>" alt=<?php print t('Home'); ?>" />
            </a>
        <?php endif; ?>

        <?php if ($site_name || $site_slogan): ?>
            <div id="name-and-slogan">
                <?php if ($site_name): ?>
                    <?php if ($title): ?>
                        <div id="site-name"><strong>
                            <a href=<?php print $front_page; ?>" title=<?php print t('Home'); ?>" rel="home"><span><?php print $site_name; ?></span></a>
                        </strong></div>
                    <?php else: /* Use h1 when the content title is empty */ ?>
                        <h1 id="site-name">
                            <a href=<?php print $front_page; ?>" title=<?php print t('Home'); ?>" rel="home"><span><?php print $site_name; ?></span></a>
                        </h1>
                    <?php endif; ?>
                <?php endif; ?>

                <?php if ($site_slogan): ?>
                    <div id="site-slogan"><?php print $site_slogan; ?></div>
                <?php endif; ?>
            </div> <!-- /#name-and-slogan -->
        <?php endif; ?>
        <?php print render($page['header']); ?>

    </div></div> <!-- /.section,/#header -->

    <?php if ($main_menu || $secondary_menu): ?>
        <div id="navigation"><div class="section">
            <?php print theme('links__system_main_menu', array('links' => $main_menu,
                'attributes' => array('id' => 'main-menu', 'class' => array('links', 'clearfix')), 'heading'
                => t('Main menu'))); ?>
            <?php print theme('links__system_secondary_menu', array('links' => $secondary_menu,
                'attributes' => array('id' => 'secondary-menu', 'class' => array('links', 'clearfix')),
                'heading' => t('Secondary menu'))); ?>
        </div></div> <!-- /.section,/#navigation -->
```

```
<?php endif; ?>

<?php if ($breadcrumb): ?>
<div id="breadcrumb"><?php print $breadcrumb; ?></div>
<?php endif; ?>

<?php print $messages; ?>

<div id="main-wrapper"><div id="main" class="clearfix">

  <div id="content" class="column"><div class="section">
    <?php if ($page['highlighted']): ?><div id="highlighted"><?php print
    render($page['highlighted']); ?></div><?php endif; ?>
    <a id="main-content"></a>
    <?php print render($title_prefix); ?>
    <?php if ($title): ?><h1 class="title" id="page-title"><?php print $title;
    ?></h1><?php endif; ?>
    <?php print render($title_suffix); ?>
    <?php if ($tabs): ?><div class="tabs"><?php print render($tabs); ?></div><?php
    endif; ?>
    <?php print render($page['help']); ?>
    <?php if ($action_links): ?><ul class="action-links"><?php print
    render($action_links); ?></ul><?php endif; ?>
    <?php print render($page['content']); ?>
    <?php print $feed_icons; ?>
  </div></div> <!-- /.section, #content -->

  <?php if ($page['sidebar_first']): ?>
    <div id="sidebar-first" class="column sidebar"><div class="section">
      <?php print render($page['sidebar_first']); ?>
    </div></div> <!-- /.section, #sidebar-first -->
  <?php endif; ?>

  <?php if ($page['sidebar_second']): ?>
    <div id="sidebar-second" class="column sidebar"><div class="section">
      <?php print render($page['sidebar_second']); ?>
    </div></div> <!-- /.section, #sidebar-second -->
  <?php endif; ?>
</div></div> <!-- /main, #main-wrapper -->
<div id="footer"><div class="section">
  <?php print render($page['footer']); ?>
</div></div> <!-- /.section, #footer -->

</div></div> <!-- /#page, #page-wrapper -->
```

Các biến được sử dụng trong page.tpl.php file

Biến	Thông tin mô tả
\$base_path	Đường dẫn URL cơ sở trong hệ thống Drupal. ở mức tối thiểu nhất mặc định là /
\$directory	Định vị thư mục chứa template e.g., modules/system or themes/bartik
\$is_front	TRUE nếu trang hiện tại là trang chủ
\$logged_in	TRUE nếu người dùng đã đăng ký và đăng nhập vào
\$is_admin	TRUE nếu người dùng có quyền truy cập tới các trang admin
\$front_page	Là URL của trang chủ, sử dụng URL này để thay thế \$base_path khi ấn liên kết tới trang chủ. nó bao gồm cả vùng ngôn ngữ hay tiền tố ngôn ngữ
\$logo	Đường dẫn tới ảnh logo, được định rõ trong theme's configuration
\$site_name	Là tên của website, được thiết lập trong theme settings
\$site_slogan	Là câu khẩu hiệu slogan của website, được thiết lập trong theme settings
\$main_menu (array)	1 mảng chứa các liên kết main menu links
\$secondary_menu (array)	1 mảng chứa các liên kết se condary menu links
\$breadcrumb	Định vị cấp trang hiện tại
\$title_prefix	Là 1 mảng dẫn suất bổ sung tạo ra bởi các modules nhằm hiện thị phía trước thẻ title chính
\$title	Là tiêu đề trang page title, sử dụng trong nội dung HTML content hiện thời
\$title_suffix (array)	Là 1 mảng dẫn suất bổ sung tạo ra bởi các modules nhằm hiện thị phía sau thẻ title chính
\$message	Trạng thái và các thông báo lỗi
\$tabs (array)	Các tabs liên kết đến các trang pages con có cấp thấp hơn trang hiện tại
\$action_links (array)	Các thao tác địa phương với trang page ví dụ như "Add menu" trang thanh menu admin
\$feed_icons	1 chuỗi kí tự của tất cả feed icons cho trang hiện tại
\$node	Đối tượng node

Các biến \$page['help'], \$page['highlighted'], \$page['content'], \$page['sidebar_first'], \$page['sidebar_second'], \$page['header'], and \$page['footer'] biểu diễn các vùng regions trên page.

The region.tpl.php File

File này tập trung vào cách các vùng regions hiện trị trên website. region.tpl.php file mặc định cực kỳ đơn giản. về bản chất nó chỉ hiện thị nội dung content được gán cho 1 vùng region nào đó

```
<?php if ($content): ?>
<div class="<?php print $classes; ?>">
  <?php print $content; ?>
</div>
<?php endif; ?>
```

Biến	Thông tin mô tả
\$content	Nội dung content cho vùng region, nó đặc trưng cho khối block
\$classes	1 chuỗi các class dùng để tạo CSS.
\$region	Tên của biến region được định nghĩa trong theme's .info file
\$is_admin	TRUE nếu người dùng hiện tại là admin
\$is_front	TRUE khi vùng này hiện thị ở trang chủ
\$logged_in	TRUE khi người dùng hiện tại đăng nhập vào hệ thống

Cấu hình Genesis Theme (theme gốc)

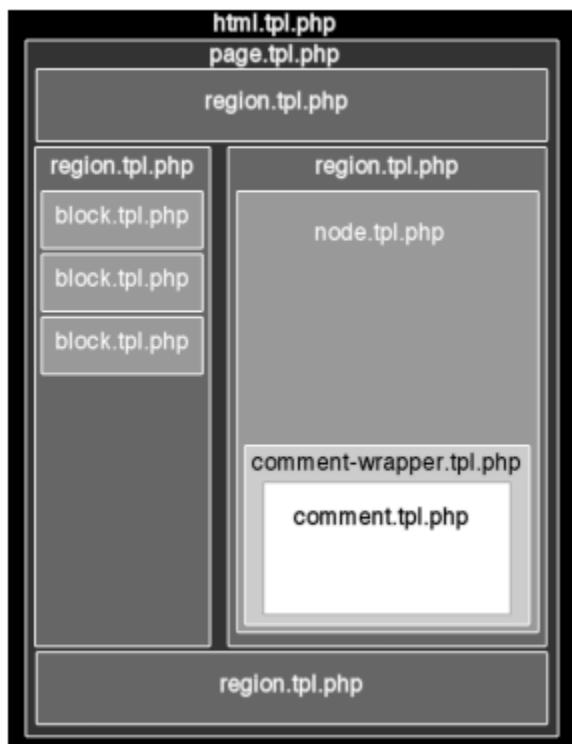
Ta cài đặt theme gốc tại www.drupal.org/project/genesis. Chú ý rằng việc cài đặt này sẽ tạo ra 2 phiên bản Genesis base and Genesis subtheme.đa số người tạo theme drupal sử dụng subthemes như là cơ chế tạo ra 1 theme mới dựa vào theme nền tảng

Để tạo ra theme mới thực hiện theo các bước sau:

1. Sao chép genesis_SUBTHEME (dựa vào theme cơ bản làm sẵn để tạo ra theme mới) trong thư mục /sites/all/themes/genesis vào thư mục /sites/all/themes ta tạo mới ra
2. Đổi tên thư mục vừa copy thành genesis_mytheme (sử mytheme thành tên thư mục ta vừa tạo mới ra)
3. Trong thư mục /sites/all/themes/mytheme ta sửa tên genesis_SUBTHEME.info thành genesis_mytheme.info . file có đuôi .info phải được đặt tên giống với tên thư mục chứa nó
4. Sửa thông tin trong genesis_mytheme.info , ta tiến hành thay đổi các giá trị sau:
 - a. Thay đổi tên từ Genesis SUBTHEME thành Genesis mytheme
 - b. Thay đổi thông tin mô tả của theme cũ thành thông tin mô tả cho theme mới sẽ tạo ra
5. Chỉnh sửa template.php . sử dụng tính năng search and replace thay đổi mọi thành phần trong genesis_SUBTHEME thành genesis_mytheme

6. Sao chép các file từ thư mục /sites/all/themes/genesis/templates sang thư mục /sites/all/themes/genesis_mytheme :

- a. page.tpl.php
- b. node.tpl.php
- c. block.tpl.php
- d. field.tpl.php
- e. region.tpl.php
- f. comment.tpl.php
- g. html.tpl.php



bước tiếp theo là bật theme bằng cách ấn vào Appearance >> tìm tới phần “disabled themes” >> ta sẽ nhìn thấy 3 phiên bản của Genesis theme: base, the off-the-shelf subtheme, và theme vừa được tạo mới từ theme subtheme có tên là mytheme >> ấn vào “Enable and set default” ở theme mới vừa tạo ra và sau đó trở lại trang chủ.ta sẽ nhìn thấy giao diện như hình dưới



Figure 13-1. The standard Genesis theme

Ta sẽ thực hiện sử dụng theme gốc Genesis starter theme để tạo ra website trông giống như

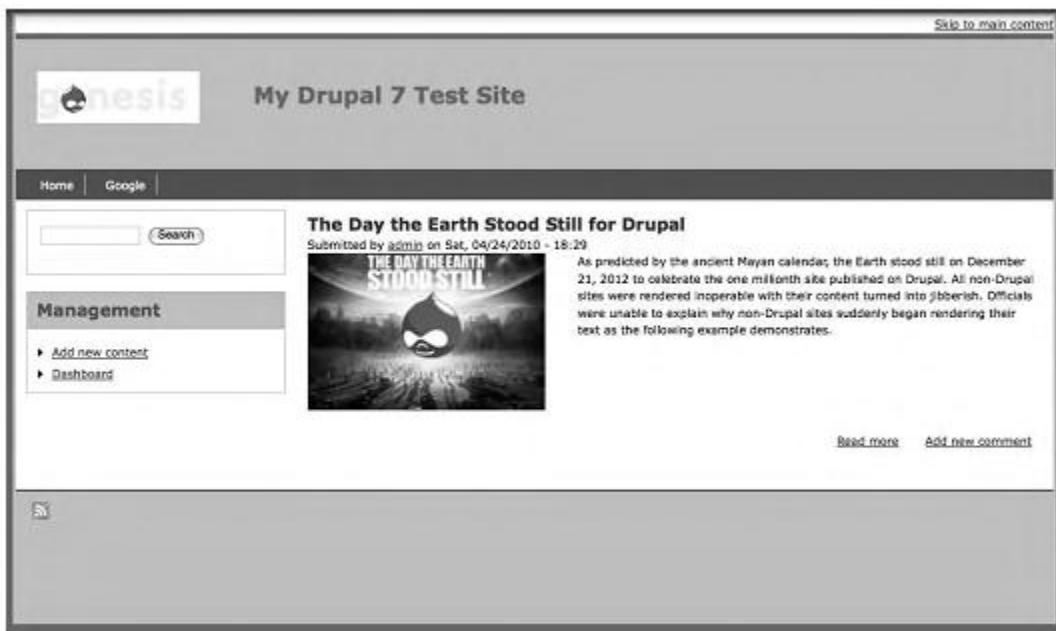


Figure 13-2. A sample site using the updated theme

Chỉnh sửa theme gốc

Trước khi bắt đầu vào chỉnh sửa, ta phải hiểu cấu trúc căn bản của theme gốc. Đầu tiên ta phải chú ý tới là các vùng regions được định nghĩa trong theme. Trong trường hợp với theme gốc ta xem xét trong file `genesis_mytheme.info` (tìm thấy trong thư mục `/sites/all/themes/genesis_mytheme`). theme gốc có 12 vùng regions :

1. Page top
2. Leaderboard
3. Header
4. Help
5. Secondary content
6. Highlighted content
7. Content
8. First sidebar
9. Second sidebar
10. Tertiary content
11. Footer
12. Page bottom

Vị trí các vùng regions trong theme :

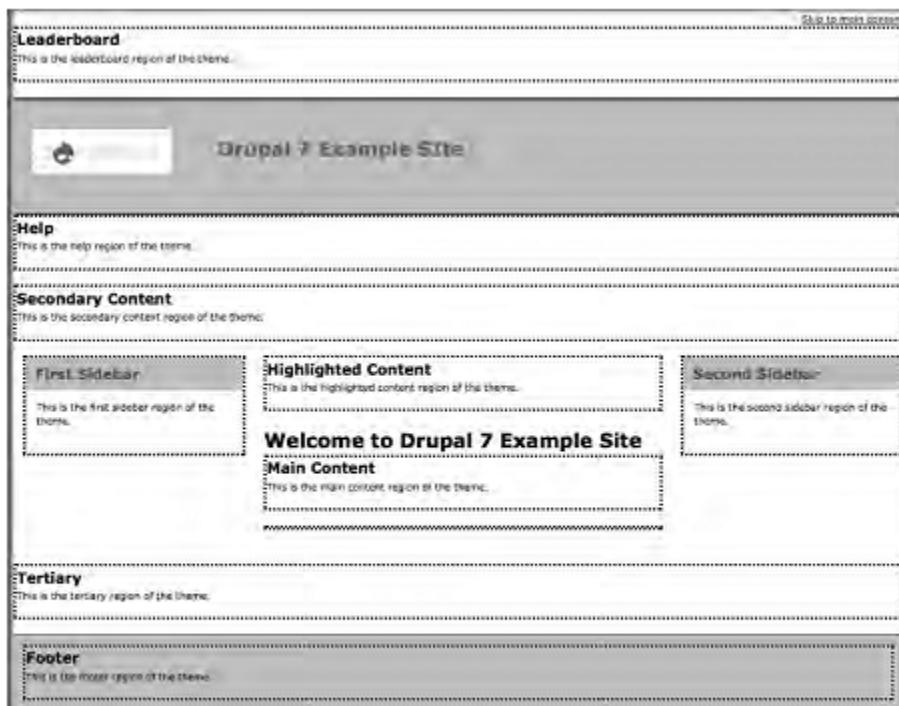
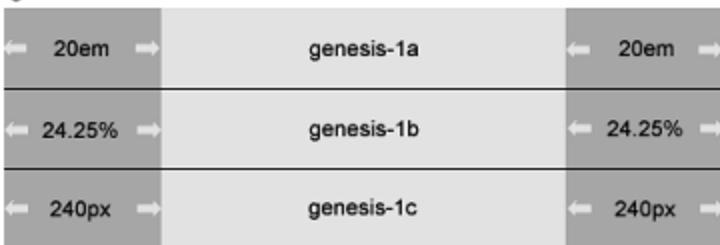


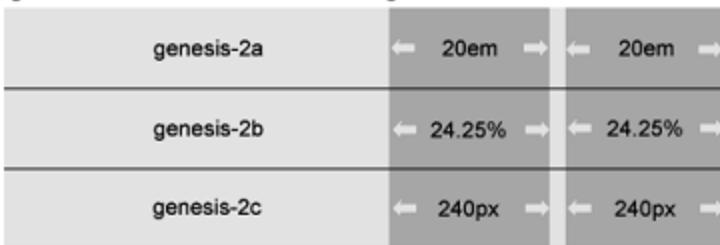
Figure 13-3. Position of the standard Genesis regions on a sample page

Theme gốc genesis cung cấp 1 vài đặc tính cực kỳ hữu dụng đơn giản để sử dụng , nó còn bao gồm cả khả năng lựa chọn thiết kế layout cho trang website mới. Genesis cung cấp 9 lựa chọn layout cơ bản được nhóm vào trong 3 nhóm

genesis-1 Standard 3 column



genesis-2 Both sidebars on the right



genesis-3 Both sidebars on the left

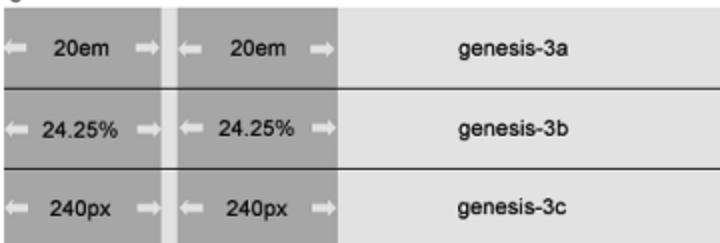


Figure 13-4. Standard Genesis layout options

Ta sẽ lựa chọn genesis-1c layout. theo ví dụ ta đặt ra ta sẽ phải sửa lại html.tpl.php file nằm trong thư mục /sites/all/themes/genesis_mytheme. trong file này ta sẽ thay đổi CSS ID được gán cho thẻ <body> từ 1 thành 1c có dạng như sau:

```
<body id="genesis-1c" <?php print $attributes;?>>
```

Bước tiếp theo ta thiết lập lại độ rộng của theme. mặc định Genesis có width : 100%. ta tiến hành sửa lại width thành 960px bằng cách sửa lại file CSS (được tìm thấy trong thư mục /sites/all/themes/genesis_mytheme/css) có 2 CSS ID cần thay đổi : 1 là cho trình duyệt IE6 và cái còn lại dành cho tất cả trình duyệt khác. tìm đến #container CSS ID trong page.css file và thay đổi như sau:

```
#container {
  width: 960px; /* Width for IE6 */
}
```

```
body > #container {
```

```
width: 960px; /* Width for all other browsers */  
min-width: 760px; /* set a minimum width */  
/*max-width: 960px; set a maximum width */  
}
```

H tiếp tục thay đổi màu của trang web theo tông xanh như sau:

Primary background color: #528544

Secondary color: #287314

Tertiary color: #9CD88C

Ta thay đổi tiếp trong file page.css

```
body {  
padding: 0 10px;  
background-color: #528544;  
}
```

```
#container {  
width: 960px; /* Width for IE6 */  
background-color: #ffffff;  
}
```

Bước thay đổi tiếp theo thay đổi phần header

```
#header {  
background-color: #9CD88C;  
border-top: 6px solid #528544;  
border-bottom: 3px solid #287314;  
height: 120px;  
}
```

Tiếp theo ta sửa lại phần logo và phần site title những phần được biểu diễn trong vùng header

```
#logo {  
position: relative;  
float: left;  
margin-left: 20px;  
margin-right: 50px;  
margin-top: 30px;  
}  
#site-name {  
font-weight: bold; /* Bold or else the line heights vary. */  
font-size: 2em;  
line-height: 1.5;  
position: relative;
```

```
float: left;  
margin-top: 35px;  
}
```

Ta xóa bỏ dấu gạch dưới và thay đổi màu sắc text trong page.css file, ta tìm #site-name a:link and #site-name a:visited CSS IDs và sửa thành:

```
#site-name a:link {  
    text-decoration: none;  
    color: #528544;  
}  
  
#site-name a:visited {  
    text-decoration: none;  
    color: #528544;  
}
```

Bố cục layout phần header khi đã sửa lại

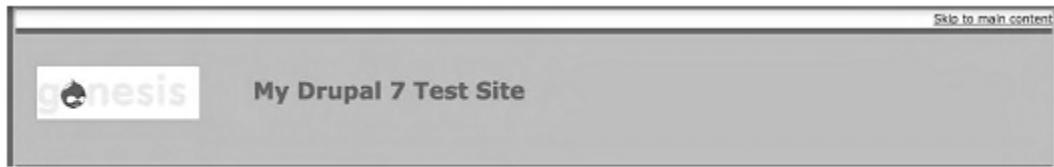


Figure 13-5. The revised header layout after CSS changes

Tạo ra 1 thanh menu ngang

Yêu cầu là ta phải tạo ra thanh menu ngang nằm dưới header. bước đầu tiên ta phải thêm vùng mới bằng cách sửa trong genesis_mytheme.info và thêm vào dòng sau vào cuối cùng

```
regions[main_menu_links] = Main Menu Links
```

đoạn mã trên chỉ dẫn cho drupal thêm vùng mới có tên trong cơ sở dữ liệu là main_menu_links vào các vùng regions trong theme. giá trị bên phải là tên sẽ được hiện thị trong trang block configuration

bước tiếp theo là tiến hành cập nhật lại page.tpl.php nơi ta sẽ thêm đoạn mã chỉ thị cho drupal nơi sẽ hiện thị vùng main_menu_links này. ta cần kiểm tra điều kiện nếu vùng mới này không chứa thứ gì thì ta muốn drupal không hiện thị nó

ta sẽ thêm đoạn mã sau vào phía sau dòng mã "</div><!--header -->" trong page.tpl.php file :

```
<?php if ($page['main_menu_links']): ?>  
    <div id="main-menu-wrapper" class="clearfix">  
        <div class="main-menu-inner"><?php print render($page['main_menu_links']); ?></div>
```

```
</div>
<?php endif; ?>
```

Điều kiện if sẽ kiểm tra xem có gì được gán vào vùng 'main_menu_links' hay không.nếu có thì các dòng tiếp theo định nghĩa 1 cấu trúc DIV dùng để biểu diễn vùng đó (thẻ DIV với id="main-menu-wrapper").quan trọng nhất là dòng dùng hàm Print trong php để in ra nội dung content của vùng 'main_menu_links'

Ta có thể dùng cách làm như trên khi tạo 1 vùng mới nào đó

Bước tiếp theo ta cập nhật vào trong page.css file những thuộc tính CSS dùng để biểu diễn thanh menu ngang

```
#main-menu-wrapper {
background-color:#287314;
border-bottom:2px solid #9CD88C;
height:25px;
}
```

Tiếp ta thêm mới định nghĩa CSS cho main-menu-inner class.thêm các dòng sau:

```
.main-menu-inner {
position: relative;
float: left;
height: 25px;
}
```

Tiếp ta định nghĩa cách các đối tượng menu sẽ xuất hiện trên thanh menu.bởi vì drupal menus được biểu diễn dưới dạng dọc giống như danh sách không có thứ tự trong HTML.ta sẽ ép buộc các đối tượng menu biểu diễn theo dạng ngang bằng cách thiết lập thuộc tính CSS display: inline; thực hiện như sau:

```
.main-menu-inner ul.menu li {
display: inline;
list-style: none;
border-right: 1px solid #9CD88C;
padding-right: 10px;
padding-left: 10px;
padding-bottom: 3px;
}
```

```
.main-menu-inner ul.menu li a {  
color: #ffffff;  
text-decoration: none;  
}
```

Tiếp ta thêm định nghĩa CSS cho sự kiện khi di chuột lên 1 đối tượng menu thì màu chữ sẽ thay đổi sang màu đen

```
.main-menu-inner ul.menu li a:hover {  
color: #000;  
}
```

Giờ thanh menu ngang đã sẵn sàng cho việc thiết lập và hiện thị

Drupal caches tất cả các vùng regions được định nghĩa trong template, có nghĩa là khi ta thay đổi trong file .info thì drupal không hiện thị ra sự thay đổi đó cho tới khi ta làm mới sites cache

Ta reset lại cache bằng cách : ấn vào Configuration >> Performance >> “Clear all caches”

Bước tiếp theo là gán lại Secondary menu từ vị trí hiện tại sang vùng mới ta vừa tạo ra lúc trước.bằng cách : ấn vào Structure >> Blocks >> tìm tới Secondary menu >> thay đổi vị trí hiện thị

Để thử nghiệm thanh menu ngang , ta thêm vào 1 đối tượng menu mới vào Secondary menu

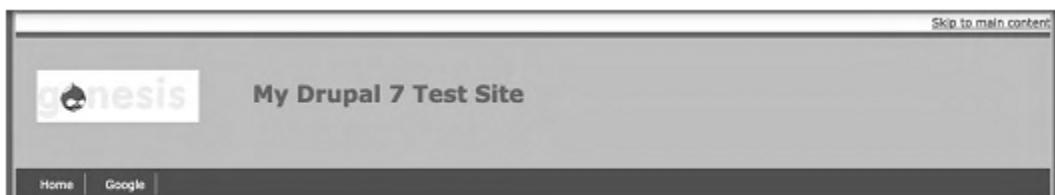


Figure 13-6. Horizontal menu themed

Xem xét lại ta nhận thấy text trong phần body container chạm vào viền bên trái, do vậy sửa lại 1 chút. DIV ID mà theme gốc sử dụng cho vùng body là #columns ID.xác định CSS ID này trong page.css file và thêm vào đoạn CSS sau:

```
adding-left: 10px;
```

tiếp ta thêm thuộc tính border vào #container.xác định #container ID trong page.css file thêm vào đoạn sau:

```
border: 3px solid #9CD88C;
```

Theming the Footer

Tìm tới #footer DIV ID sửa lại nó như sau:

```
#footer {  
background-color: #9CD88C;  
height: 100px;  
padding: 10px;  
border-top: 2px solid #287314;  
}
```

Ta muốn xóa từ “Powered by Drupal” và phần menu được hiện thị ra

Để xóa từ ta vào trang Blocks gán block chứa “Powered by Drupal” thành none

Để xóa thanh menu,, ta chỉnh sửa page.tpl.php file và xóa những dòng sau

```
<?php if ($secondary_menu_links): ?>  
    <div id="secondary-menu-wrapper" class="clearfix">  
        <div class="secondary-menu-inner"><?php print $secondary_menu_links; ?></div>  
    </div>  
<?php endif; ?>
```

Sau khi xóa bỏ những dòng sau và làm mới trang lại, ta sẽ nhìn vào phần footer và thấy sự thay đổi

Theming Nodes

Drupal cung cấp cơ chế điều khiển cách các node hiện thị trên 1 trang page.mặc định , 1 node được hiện thị với title ở trên đầu, the author, date, and time nằm dưới the title.sau đó tới phần ảnh và tới phần thân body của node.

Ta sẽ chỉnh sửa nodes.css file để ảnh sẽ nằm ở bên trái đoạn text

Đầu tiên ta tạo ra 1 bài viết Article content (bài viết có đính kèm ảnh)

The Day the Earth Stood Still for Drupal

Submitted by admin on Sat, 04/24/2010 - 18:29



As predicted by the ancient Mayan calendar, the Earth stood still on December 21, 2012 to celebrate the one millionth site published on Drupal. All non-Drupal sites were rendered inoperable with their content turned into gibberish. Officials were unable to explain why non-Drupal sites suddenly began rendering their text as the following example demonstrates.

[Read more](#) [Add new comment](#)

Figure 13-7. Unthemed node

Tiếp theo, ta sửa lại nodes.css file ở trong thư mục /sites/all/themes/genesis_mytheme/css.trong fiile này ta tìm kiếm class field-type-image.và chỉnh sửa như sau:

```
.field-type-image {  
position: relative;  
float: left;  
margin-right: 15px;  
margin-bottom: 10px;  
}
```

Ta thêm đoạn CSS sau

```
.node-inner {  
padding-right: 15px;  
}
```

Sau khi làm mới lại ta thấy sự thay đổi như sau:

The Day the Earth Stood Still for Drupal

Submitted by admin on Sat, 04/24/2010 - 18:29



As predicted by the ancient Mayan calendar, the Earth stood still on December 21, 2012 to celebrate the one millionth site published on Drupal. All non-Drupal sites were rendered inoperable with their content turned into gibberish. Officials were unable to explain why non-Drupal sites suddenly began rendering their text as the following example demonstrates.

[Read more](#) [Add new comment](#)

Figure 13-8. Themed node

Để thay đổi những thứ khác nữa ta chỉnh sửa tại node.tpl.php and nodes.css files.

Theming Blocks

Ví dụ như ta muốn là đẹp phần block này



Figure 13-9. Unthemed block

Để thực hiện điều này ta tiến hành chỉnh sửa trong blocks.css file.ta sẽ thay đổi .sidebar .b lock .block-content class

```
.sidebar .block .block-content {  
    border: 1px solid #9CD88C;  
    padding: 10px;  
}
```

Tiếp đến ta thay đổi cách định nghĩa block title, chỉnh sửa .sidebar .block h2 thành như sau :

```
.sidebar .block h2 {  
    background-color: #9CD88C  
    padding-left: 10px;  
    padding-top: 5px;  
    padding-bottom: 5px;  
    color: #287314;  
}
```

Làm mới lại trang kết quả sec như sau :



Figure 13-10. Themed block

Các từ khóa quan trọng trong các files php template

Vai trò của file .info

Xét ví dụ ở đây là file bartik.info(themes/bartik/bartik.info):

```
; $Id: bartik.info,v 1.5 2010/11/07 00:27:20 dries Exp $
name = Bartik
description = A flexible, recolorable theme with many regions.
package = Core
version = VERSION
core = 7.x
stylesheets[all][] = css/layout.css
stylesheets[all][] = css/style.css
stylesheets[all][] = css/colors.css
stylesheets[print][] = css/print.css
regions[header] = Header
regions[help] = Help
regions[page_top] = Page top
regions[page_bottom] = Page bottom
regions[highlighted] = Highlighted
regions[featured] = Featured
regions[content] = Content
regions[sidebar_first] = Sidebar first
regions[sidebar_second] = Sidebar second
regions[triptych_first] = Triptych first
regions[triptych_middle] = Triptych middle
regions[triptych_last] = Triptych last
regions[footer_firstcolumn] = Footer first column
regions[footer_secondcolumn] = Footer second column
regions[footer_thirdcolumn] = Footer third column
regions[footer_fourthcolumn] = Footer fourth column
regions[footer] = Footer
settings[shortcut_module_link] = 0
; Information added by drupal.org packaging script on 2011-01-05
version = "7.0"
project = "drupal"
datestamp = "1294208756"
```

chú ý tới các cấu hình địa chỉ file căn bản nó sẽ cho ta biết các thông tin quan trọng về theme và các file thêm đang sử dụng, các vùng trong theme :

- The theme's name
- Description

- Version and compatibility info
- Theme engine required
- Stylesheets needed
- Regions included

Vai trò của page.tpl.php file

page.tpl.php file ta có thể xác định ví trí của chúng trong từng theme riêng biệt. Trong ví dụ ta xét tới page.tpl.php trong thư mục nhân của hệ thống : **modules/system**

```
<div id="page-wrapper"><div id="page">
  <div id="header"><div class="section clearfix">
    <?php if ($logo): ?>
      <a href=<?php print $front_page; ?>" title=<?php print
t('Home'); ?>" rel="home" id="logo">
        <img src=<?php print $logo; ?>" alt=<?php print t('Home');
?>" />
      </a>
    <?php endif; ?>
    <?php if ($site_name || $site_slogan): ?>
      <div id="name-and-slogan">
        <?php if ($site_name): ?>
          <?php if ($title): ?>
            <div id="site-name"><strong>
              <a href=<?php print $front_page; ?>" title=<?php
print t('Home'); ?>" rel="home"><span><?php print $site_name; ?></
span></a>
            </strong></div>
        <?php else: /* Use h1 when the content title is empty */ ?>
      <h1 id="site-name">
        <a href=<?php print $front_page; ?>" title=<?php
print t('Home'); ?>" rel="home"><span><?php print $site_name; ?></
span></a>
      </h1>
    <?php endif; ?>
    <?php endif; ?>
    <?php if ($site_slogan): ?>
      <div id="site-slogan"><?php print $site_slogan; ?></div>
    <?php endif; ?>
  </div> <!-- /#name-and-slogan -->
  <?php endif; ?>
  <?php print render($page['header']); ?>
</div></div> <!-- /.section, /#header -->
```

```
<?php if ($main_menu || $secondary_menu): ?>
<div id="navigation"><div class="section">
  <?php print theme('links__system_main_menu', array('links'
=> $main_menu, 'attributes' => array('id' => 'main-menu', 'class' =>
array('links', 'inline', 'clearfix')), 'heading' => t('Main menu')));
?>
  <?php print theme('links__system_secondary_menu',
array('links' => $secondary_menu, 'attributes' => array('id' =>
'secondary-menu', 'class' => array('links', 'inline', 'clearfix')),
'heading' => t('Secondary menu'))); ?>
</div></div> <!-- /.section,/#navigation -->
<?php endif; ?>
<?php if ($breadcrumb): ?>
<div id="breadcrumb"><?php print $breadcrumb; ?></div>
<?php endif; ?>
<?php print $messages; ?>
<div id="main-wrapper"><div id="main" class="clearfix">
  <div id="content" class="column"><div class="section">
    <?php if ($page['highlighted']): ?><div id="highlighted"><?php
print render($page['highlighted']); ?></div><?php endif; ?>
    <a id="main-content"></a>
    <?php print render($title_prefix); ?>
    <?php if ($title): ?><h1 class="title" id="page-title"><?php
print $title; ?></h1><?php endif; ?>
    <?php print render($title_suffix); ?>
    <?php if ($tabs): ?><div class="tabs"><?php print
render($tabs); ?></div><?php endif; ?>
    <?php print render($page['help']); ?>
    <?php if ($action_links): ?><ul class="action-links"><?php
print render($action_links); ?></ul><?php endif; ?>
Understanding PHPTemplate Themes
[ 76 ]
  <?php print render($page['content']); ?>
  <?php print $feed_icons; ?>
</div></div> <!-- /.section,/#content -->
<?php if ($page['sidebar_first']): ?>
  <div id="sidebar-first" class="column sidebar"><div
class="section">
    <?php print render($page['sidebar_first']); ?>
  </div></div> <!-- /.section,/#sidebar-first -->
<?php endif; ?>
<?php if ($page['sidebar_second']): ?>
  <div id="sidebar-second" class="column sidebar"><div
class="section">
    <?php print render($page['sidebar_second']); ?>
  </div></div> <!-- /.section,/#sidebar-second -->
<?php endif; ?>
</div></div> <!-- /#main,/#main-wrapper -->
```

```
<div id="footer"><div class="section">
<?php print render($page['footer']); ?>
</div></div> <!-- ./section, #footer -->
</div></div> <!-- #page, #page-wrapper -->
```

Nhìn vào đoạn mã trên ta thấy rằng template dựa vào rất nhiều loại cấu trúc điều kiện để xử lý các vùng hiện thị đối với user ta trích ra đoạn mã bao gồm các thành phần : **the logo, the site name, and the site slogan.**

```
<?php if ($logo): ?>
  <a href=<?php print $front_page; ?>" title=<?php print
t('Home'); ?>" rel="home" id="logo">
    <img src=<?php print $logo; ?>" alt=<?php print t('Home');
?>" />
  </a>
<?php endif; ?>
<?php if ($site_name || $site_slogan): ?>
  <div id="name-and-slogan">
    <?php if ($site_name): ?>
      <?php if ($title): ?>
        <div id="site-name"><strong>
          <a href=<?php print $front_page; ?>" title=<?php
print t('Home'); ?>" rel="home"><span><?php print $site_name; ?></
span></a>
        </strong></div>
      <?php else: /* Use h1 when the content title is empty */ ?>
    <h1 id="site-name">
      <a href=<?php print $front_page; ?>" title=<?php
print t('Home'); ?>" rel="home"><span><?php print $site_name; ?></
span></a>
    </h1>
    <?php endif; ?>
  <?php endif; ?>
  <?php if ($site_slogan): ?>
    <div id="site-slogan"><?php print $site_slogan; ?></div>
  <?php endif; ?>
</div> <!-- #name-and-slogan -->
<?php endif; ?>
```

Đoạn mã ngay phía sau **<?php print render(\$page['header']); ?>**
Dùng để in ra toàn bộ vùng header

Đoạn mã tiếp theo tương tự giống đoạn mã phía trên dùng để hiện thị ra **primary and secondary links:**

```
<?php if ($main_menu || $secondary_menu): ?>
```

```
<div id="navigation"><div class="section">
  <?php print theme('links__system_main_menu', array('links'
=> $main_menu, 'attributes' => array('id' => 'main-menu', 'class' =>
array('links', 'inline', 'clearfix')), 'heading' => t('Main menu')));
?>
  <?php print theme('links__system_secondary_menu',
array('links' => $secondary_menu, 'attributes' => array('id' =>
'secondary-menu', 'class' => array('links', 'inline', 'clearfix')),
'heading' => t('Secondary menu'))); ?>
</div></div> <!-- /.section, /#navigation -->
<?php endif; ?>
```

Cấu trúc tiếp theo dùng để hiện thị **breadcrumbs and the messages text**:

```
<?php if ($breadcrumb): ?>
  <div id="breadcrumb"><?php print $breadcrumb; ?></div>
<?php endif; ?>
<?php print $messages; ?>
```

1 vài cấu trúc khác :

Tên của vùng region	Đoạn mã code chứa vùng region đó
Content region	<?php print render(\$page['content']); ?>
Sidebar first region	<?php print render(\$page['sidebar_first']); ?>
Sidebar second region	<?php print render(\$page['sidebar_second']); ?>
Footer region	<?php print render(\$page['footer']); ?>

2 theme cơ bản mặc định của drupal 7

theme-Seven

theme này được sử dụng cho hệ thống admin của drupal 7. theme này nằm ở trong thư mục : **themes/seven**. theme này bao gồm các files như sau :

►	bartik	Folder
►	engines	Folder
►	garland	Folder
	README.txt	Plain Text
►	seven	Folder
	ie.css	CSS style sheet
	ie6.css	CSS style sheet
►	images	Folder
	jquery.ui.theme.css	CSS style sheet
	logo.png	Portable Network Graphics Image
	maintenance-page.tpl.php	PHP: Hypertext Preprocessor (PHP) document
	page.tpl.php	PHP: Hypertext Preprocessor (PHP) document
	reset.css	CSS style sheet
	screenshot.png	Portable Network Graphics image
	seven.info	TextEdit Document
	style-rtl.css	CSS style sheet
	style.css	CSS style sheet
	template.php	PHP: Hypertext Preprocessor (PHP) document
	vertical-tabs.css	CSS style sheet
►	stark	Folder
►	tests	Folder

Theme này đơn giản hóa chỉ sử dụng 2 file template : page.tpl.php and maintenance-page.tpl.php

2 file này dùng để ghi đè lên các file template mặc định trong nhân.những file không có để ghi đè sẽ
được sử dụng giống như các file template trong nhân core

theme-Bartik

theme này thì có cách tiếp cận phức tạp hơn và nó nằm trong thư mục : **themes/bartik** .chú ý rằng
trong theme không chỉ bao gồm các file ghi đè page.tpl.php mà nó còn bao gồm các file dạng khác của
template như : **template.php** và **các file css**

└── bartik	Folder
└── bartik.info	TextEdit Document
└── color	Folder
└── css	Folder
└── colors.css	CSS style sheet
└── ie-rtl.css	CSS style sheet
└── ie.css	CSS style sheet
└── ie6.css	CSS style sheet
└── layout-rtl.css	CSS style sheet
└── layout.css	CSS style sheet
└── maintenance-page.css	CSS style sheet
└── print.css	CSS style sheet
└── style-rtl.css	CSS style sheet
└── style.css	CSS style sheet
└── images	Folder
└── logo.png	Portable Network Graphics image
└── screenshot.png	Portable Network Graphics image
└── template.php	PHP: Hypertext Preprocessor (PHP) document
└── templates	Folder
└── comment-wrapper.tpl.php	PHP: Hypertext Preprocessor (PHP) document
└── comment.tpl.php	PHP: Hypertext Preprocessor (PHP) document
└── maintenance-page.tpl.php	PHP: Hypertext Preprocessor (PHP) document
└── node.tpl.php	PHP: Hypertext Preprocessor (PHP) document
└── page.tpl.php	PHP: Hypertext Preprocessor (PHP) document
└── engines	Folder
└── garland	Folder
└── README.txt	Plain Text
└── seven	Folder
└── stark	Folder
└── tests	Folder

Ngoài page.tpl.php cản bản theme này còn cung cấp cho ta các file tùy chọn khác :

- comment.tpl.php
- comment-wrapper.tpl.php
- maintenance-page.tpl.php
- node.tpl.php

Sử dụng kỹ thuật chặn và ghi đè để tùy chỉnh theme

Ta sẽ đi tìm hiểu các phần sau:

- Mỗi quan hệ liên kết giữa các templates, stylesheets và các hàm dùng để theme
- Cách nào để có thể ghi đè styles and stylesheets
- Cách nào để có thể ghi đè templates and themable functions(các hàm có khả năng theme)

- Các sử dụng các biến template

Trong các ví dụ ta sẽ làm việc với **Bartik theme**.

Mỗi quan hệ liên kết giữa các templates, stylesheets và các hàm dùng để theme

Default templates cung cấp cho ta nhanh nhất và đơn giản nhất để bắt đầu tùy chỉnh theme theo ý muốn.các file quan trọng trong template là block and page templates.

Cách đơn giản nhất là ta copy template mặc định này vào trong thư mục mà ta định tạo ra template mới.tại đây ta chỉ cần chỉnh sửa những gì theo ý ta muốn : ví dụ như đơn giản là thay đổi vùng lựa chọn hay phức tạp hơn thêm các biến mới vào template

Default stylesheets

The themable functions được cho là ít phức tạp hơn so với các file templates và nó được xử lý theo cách xử lý của hàm.1 số trong chúng bị phân tán trong hệ thống, nhiều khi muốn xác định được chúng ta cần xác định các từ khóa của các module riêng lẻ.(ở drupal 7 thì hệ thống đã ít phục thuộc hơn vào các hàm này)

Công cụ dùng để giúp việc theming theo cách này : http://drupal.org/project/devel_themer

Thay đổi theme bằng cách ghi đè lên CSS mặc định

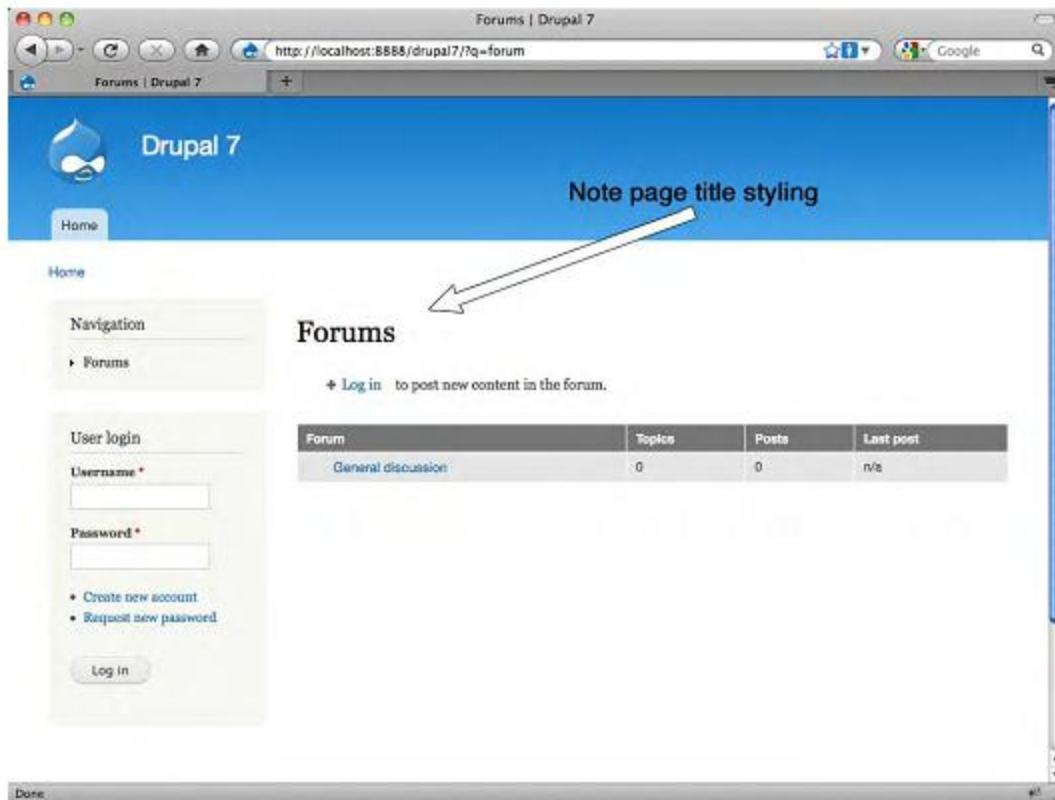
Drupal bao gồm 1 lượng lớn các stylesheets.để ghi đè lên các stylesheets này, ta chỉ cần đặt nó vào trong thư mục theme mới, mà không cần phải sửa trực tiếp vô số các stylesheets mặc định đó

CSS bên trong theme đang được kích hoạt sẽ có mức ưu tiên cao nhất.nếu có sự đối lập xảy ra với định nghĩa CSS khác, thì định nghĩa CSS trong theme được kích hoạt luôn thắng.lưu ý rằng các định nghĩa CSS chưa được định nghĩa trong theme mới này sẽ sử dụng định nghĩa CSS mặc định có sẵn của hệ thống

Để tạo ra 1 file CSS mới, ta tạo nó đặt trong thư mục của theme mới tạo, và sau đó tham chiếu tới file đó trong file .info

Ví dụ về việc ghi đè CSS

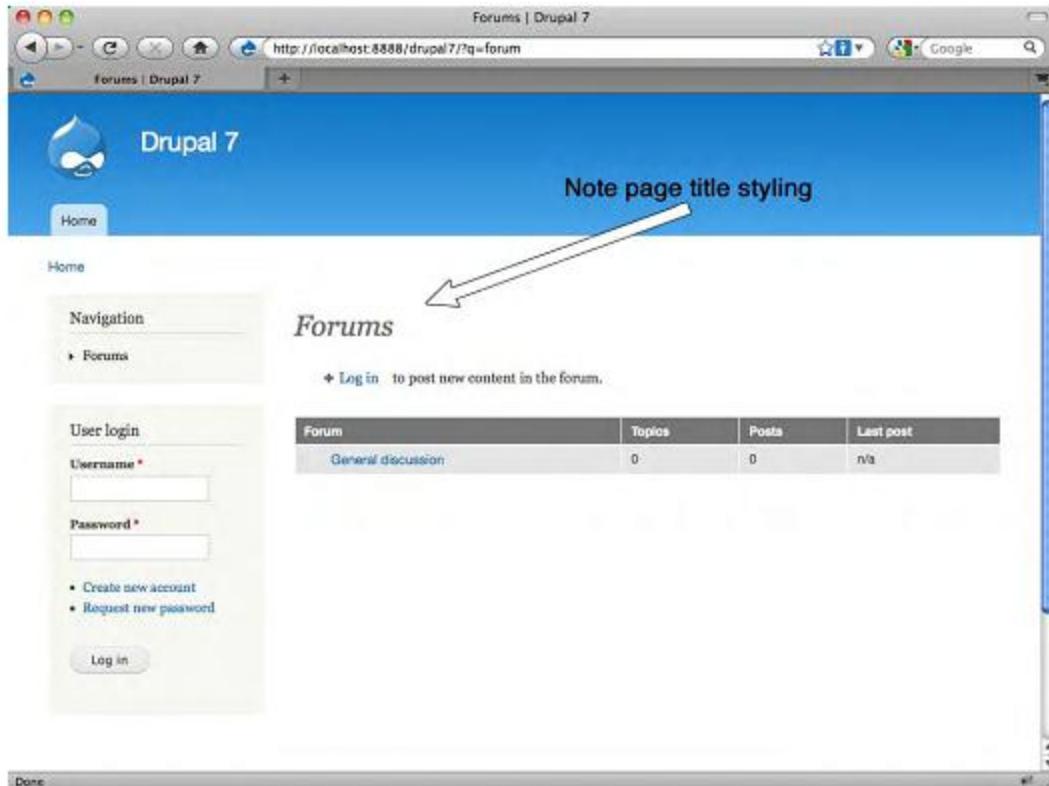
Page title trong drupal được định nghĩa trong CSS là **.title**.với theme mặc định Bartik, **.title** không bao gồm định nghĩa CSS nào vì vậy nó sẽ sử dụng CSS mặc định sẵn có của hệ thống.và nó sẽ trông như thế này :



Ta tiến hành ghi đè CSS bằng cách đơn giản là thêm định nghĩa CSS **.title** vào bên trong file **Bartik theme's style.css** (`/themes/bartik/style.css`) .ta thêm vào đoạn mã như sau :

```
.title {  
color: #666;  
font-style: italic;  
}
```

Giờ ta load lại trang và được kết quả sau khi ghi đè CSS như thế này :



Ghi đè core stylesheets

Ta không chỉ muốn ghi đè CSS ở mức selector mà còn muốn ghi đè cả file CSS đã tồn tại sẵn, để thực hiện được như vậy đơn giản ta chỉ cần tạo ra 1 file mới có cùng tên với file CSS cần ghi đè và đặt nó vào trong thư mục theme được kích hoạt. cách thực hiện ghi đè kiểu này không khôn ngoan cho lắm, và để tránh tình trạng nạp vào những file không cần thiết, ta phải chú ý tiến hành thay đổi trong **template.php** file

Sử hàm **hook_css_alter()** để thay thế, thêm mới hay loại trừ các CSS files từ các trang trước tới khi các files này thực hiện xuất ra.

Đây là 1 ví dụ lấy từ **Seven theme's template.php file**. hàm **hook_css_alter()** được sử dụng để thông báo tới hệ thống rằng **Seven theme** đang được ghi đè 2 **core stylesheets** (**ghi đè 2 file vertical-tabs.css và jquery.ui.theme.css**)

```
/**  
 * Implements hook_css_alter().  
 */  
function seven_css_alter(&$css) {  
 // Use Seven's vertical tabs style instead of the default one.  
 if (isset($css['misc/vertical-tabs.css'])) {  
 $css['misc/vertical-tabs.css']['data'] = drupal_get_path('theme', 'seven') . '/vertical-tabs.css';
```

```
}

// Use Seven's jQuery UI theme style instead of the default one.
if (isset($css['misc/ui/jquery.ui.theme.css'])) {
  $css['misc/ui/jquery.ui.theme.css']['data'] = drupal_get_path('theme', 'seven') .
  '/jquery.ui.theme.css';
}

}
```

Để tìm hiểu thêm về hàm `hook_css_alter()` → truy cập vào : http://api.drupal.org/api/drupal/modules-system--system.api.php/function/hook_css_alter/7

Chú ý : khi ghi đè mà thấy vẫn chưa thay đổi thì ta phải tiến hành xóa cache bằng cách :
Configuration Manager → Performance page → Clear cached data

Ghi đè các file templates

Đây là 1 kỹ thuật khá dễ dàng và mạnh mẽ trong việc tùy chỉnh. bản chất của việc này là tạo ra 1 bản sao chép mới từ những template files mặc định có sẵn và đặt chúng vào trong thư mục theme đang kích hoạt. Vì Drupal luôn ưu tiên các những gì có trong thư mục được kích hoạt, nên những file mới này sẽ ghi đè lên các file mặc định của hệ thống.

Tóm lại là kỹ thuật này đơn giản là tạo ra bản sao chép từ các file templates mặc định và sau đó chỉnh sửa mã code bên trong cáo file mới tạo ra này :

1. Xác định các templates mà ta thấy cần
2. Sao chép các file templates mà ta sẽ tiến hành tùy chỉnh sau đó
3. Dán các file này vào thư mục theme mới
4. Thay đổi mã code trong các file này
5. Lưu lại file
6. Xóa cache và xem thay đổi

Lợi ích của việc này là ta có thể sử dụng lại các file đã sửa đổi lại vào 1 theme khác. Chính vậy mà kỹ thuật này thường sử dụng trong trang web đa theme

Ví dụ cho kỹ thuật này :

page.tpl.php file là 1 trong các file PHP Template quan trọng. file này có tác động rất lớn đến sự xuất hiện hiển thị trong trình duyệt – nó định nghĩa tất cả bố cục của 1 page trong website.

Mục đích : bạn muốn tạo ra 1 trang tùy chỉnh riêng (hay 1 tập các trang riêng) để thêm tính đa dạng vào trong website, để hoàn thành mục đích này, ta không cần phải cài đặt thêm 1 theme nào khác nữa, đơn giản hơn là ta chỉ việc tạo ra 1 phiên bản khác từ page.tpl.php file có sẵn và tùy chỉnh nó để nó hiện thị

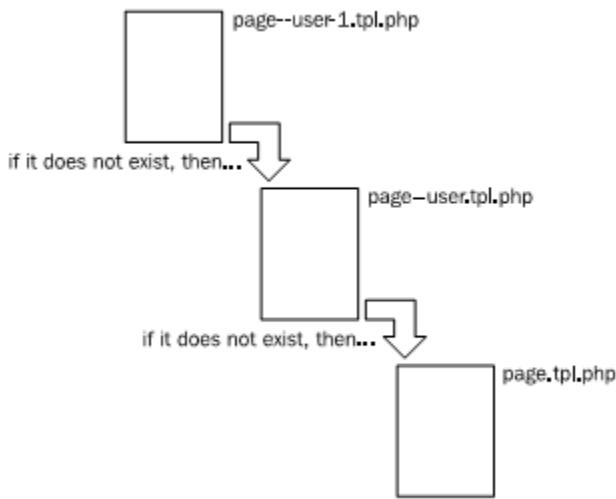
khi bắt gặp 1 điều kiện nhất định nào đó.để điều khiển sự hiển thị của template mới, ta phải xem lại quy cách đặt tên theo phân cấp trong drupal

Ví dụ : ta muốn tùy chỉnh lại trang page dành cho user pages, và trang page này sẽ được hiện thị khi 1 user chọn vào My account link.nếu không có sự định nghĩa đặc biệt nào thì drupal sẽ sử dụng định nghĩa trong page.tpl.php file mặc định.nếu ta muốn 1 trang page tùy chỉnh sẽ được hiện thị ra, thì ta sẽ cần phải chặn trang page mặc định và hiện thị ra trang page mà ta lựa chọn.để làm được việc này , ta sẽ cần phải tạo ra 1 file template mới có tên theo dạng : page--user.tpl.php và phải đặt nó vào trong thư mục theme được kích hoạt

Do vậy khi user chọn vào My account link thì hệ thống sẽ ưu tiên hiện thị file page--user.tpl.php hơn là page.tpl.php file mặc định(lưu ý tới cấu trúc đặt tên trong drupal 7 sử dụng 2 dấu gạch ngang --)

Để định nghĩa 1 cách cụ thể hơn nữa cho những user riêng biệt, ta đặt tên file như sau : page--user-1.tpl.php hiện thị ra trang dành cho user có ID=1

Drupal là 1 hệ thống nhất quán nó tuân theo quy tắc là ưu tiên sự cụ thể sau đó mới tới cái chung mặc định, càng cụ thể sẽ càng được ưu tiên



Ghi đè các hàm functions

Các hàm có khả năng dùng để theming hay còn gọi là Themable functions có vị trí rải rác khắp nơi trong hệ thống.và đa số chúng nằm trong các modules riêng biệt, các Themable functions có thể bị ghi đè bằng cách sao chép các hàm functions và đặt chúng vào bên trong template.php file của theme đang kích hoạt nơi ta chúng có thể bị sửa đổi cho phù hợp với những gì ta cần

template.php file là 1 file không bắt buộc (tùy chọn) trong 1 PHPTemplate theme.khi file này có xuất hiện, thì drupal sẽ nhìn vào những chỉ thị hướng dẫn bên trong file, vì vậy nó trở thành nơi thích hợp để ta tiến hành ghi đè các functions (preprocessing functions)

cái nhìn tổng quan về nguyên tắc xử lý được sử dụng trong kỹ thuật này:

1. Nếu file không tồn tại, tạo ra 1 file mới có tên là template.php nằm trong thư mục theme
2. Tìm tới các functions mà ta muốn tùy chỉnh
3. Sao chép function gốc và dán nó vào bên trong template.php file(file mới vừa tạo ra)
4. Đổi tên hàm function
5. Tạo thay đổi với hàm function vừa đổi lại tên và lưu lại file
6. Xóa cache và xem kết quả

Chú ý rằng : bằng cách tiến hành bằng cách biến đổi bên trong 1 file nằm trong thư mục theme, ta có thể thêm các tùy chỉnh vào 1 site mà không phải động vào các file nhân hệ thống(core files)

Nếu site sử dụng nhiều hơn 1 theme thì cách tiếp cận này không tối ưu

Sự phân cấp của các hàm themable function sẽ được gọi thông qua cách sử dụng đặt tên theo định chuẩn.các hàm themable functions mặc định có thể được xác định thông qua tên của chúng, tất cả chúng sử dụng theo cách đặt tên sau: **theme_functionname()** hay **tên theme_tên hàm()**

Ví dụ: hàm themable function mặc định dùng để điều khiển hiện thị vết **breadcrumb** của drupal được đặt tên là **themeBreadcrumb()**

(hàm **breadcrumb** mặc định nằm trong **includes/theme.inc** file)

Trong lúc thực hiện xử lý, drupal đã được thiết kế để tìm xem có sự ghi nào vào các themable functions nào hay không trước khi nó xử lý theo các hàm functions mặc định của hệ thống.hệ thống tiến hành tìm kiếm này theo thứ tự ưu tiên sau:

- **themename_functionname** (for example, **bartikBreadcrumb** – là hàm ghi đè của hàm themeBreadcrumb trong theme Bartik)
- **theme_functionname** (for example, **themeBreadcrumb**)

Chuyển đổi các hàm themable functions vào bên trong các file templates riêng

Kỹ thuật then chốt để trở nên mạnh mẽ hơn là tạo ra các files template riêng lẻ, và chúng dùng để dành riêng cho việc ghi đè cho các hàm themable functions riêng biệt.biến đổi 1 hàm function vào 1 template mới sẽ mang đến cho ta nhiều sự mềm dẻo hơn là sự chỉnh sửa đơn giản các hàm function nằm bên trong **template.php** file.bằng cách tạo ra các file templates ngoài các hàm **themable functions**, ta có thể tách các function vào trong từng phần tử và khiến việc theming trở nên dễ sửa đổi và quản lý truy cập dễ dàng hơn (**tách các hàm themable functions ra thành các file template riêng rẽ để dễ tìm và quản lý**)

Thực hiện theo cách bước sau:

1. Tạo ra 1 file **.tpl.php** mới bên trong thư mục theme
2. Đặt tên file mới này bằng cách lấy theo tên hàm gọi chúng và tiến hành xóa bỏ đi tiền tố **(theme_)** và thay đổi dấu gạch dưới thành dấu gạch ngang.ví dụ cho cách đặt tên này như sau:

từ hàm **theme_comment_view** ta biến đổi nó thành **comment-view.tpl.php**

3. Dán mã code trong hàm function liên quan đến **sự hiển thị output** vào trong file mới này
4. Tạo thay đổi các mã trong file mới
5. Lưu file lại
6. Xóa cache

Giả sử rằng ta muốn tạo ra 1 file dành riêng cho hàm **breadcrumb function** hàm này có tên là **themeBreadcrumb** (hàm fuction được chứa trong thư mục **includes/theme.inc**)

Ta nhìn vào mã gốc hàm function này :

```
/**  
 * Returns HTML for a breadcrumb trail.  
 *  
 * @param $variables  
 * An associative array containing:  
 * - breadcrumb: An array containing the breadcrumb links.  
 */  
function themeBreadcrumb($variables) {  
  $breadcrumb = $variables['breadcrumb'];  
  if (!empty($breadcrumb)) {  
    // Provide a navigational heading to give context for breadcrumb  
    // links to  
    // screen-reader users. Make the heading invisible with .element-  
    // invisible.  
    $output = '<h2 class="element-invisible">' . t('You are here') .  
    '</h2>';  
    $output .= '<div class="breadcrumb">' . implode(' » ',  
    $breadcrumb) . '</div>';  
    return $output;  
  }  
}
```

Ví dụ : ta muốn ở phía trên đầu hiện ra dòng text “You are here” với tất cả mọi người truy cập, ta muốn tất cả các dòng hiện thị ra nằm trên 1 dòng và muốn đóng gói nó vào trong thẻ div để dễ dàng CSS

Và đây là những gì ta phải tiến hành làm:

1. Tạo ra 1 file mới, đặt nó vào bên trong thư mục **Bartik theme** và đặt tên nó thành **breadcrumb.tpl.php** (lưu ý tên hàm function của nó là **themeBreadcrumb**)

2. Thêm các dòng sau trong hàm function liên quan tới **output** vào bên trong file vừa mới tạo ra

```
<div><span class="breadcrumb-title"><?php print t('You are here :'); ?></span>
<span class="breadcrumb"><?php print implode(' » ', $breadcrumb); ?></span></div>
```

3. Lưu file lại

Sau khi xóa cache và làm mới trình duyệt , ta sẽ nhìn thấy giao diện bố cục mới của **breadcrumb**

Chú ý rằng : mã code là dạng HTML cơ bản được làm styling được đóng gói xung quanh cấu trúc **print** của PHP .để tạo làm đẹp styling đơn giản là ta định nghĩa CSS cho 2 selectors '**breadcrumb-title**' and '**breadcrumb**'

Ví dụ về thao tác ghi đè trong Bartik

Giờ ta xét tới sự ghi đè có trong theme mặc định **Bartik**.tác giả của theme này sử dụng 1 số cách ghi đè và nó sẽ cho chúng ta thấy 1 vài mẫu ví dụ về sự ghi đè

Ta nhìn vào trong thư mục **themes / Bartik** nhận thấy rằng trong nó có 1 thư mục danh riêng cho việc lưu các **CSS files**(**thư mục themes / Bartik / css**) và 1 thư mục riêng khác lưu các file templates riêng biệt của theme(**thư mục themes / Bartik / templates**) và ta cũng chú ý tới sự có mặt của **template.php file**

Ghi đè lên các files template mặc định

Bartik bao gồm 1 vài phiên bản được chỉnh sửa của các files template mặc định.nội dung bên trong những file này đã biến đổi so với các file gốc mặc định có cùng tên với nó nằm bên trong hệ thống

Bartik sử dụng cấu trúc PHPTemplate, khi ta mở thư mục **/templates** ta để ý tới các file có đuôi **.tpl.php**.ta sẽ thấy được danh sách các file dùng để ghi đè lên các file gốc có sẵn của hệ thống :

File template	Thư mục gốc
comment.tpl.php	/modules/comment/
Comment-wrapper.tpl.php	/modules/comment/
node.tpl.php	/modules/node/
maintenance-page.tpl.php	/modules/system/

Giờ ta tiến hành so sánh phiên bản mặc định có sẵn của **node.tpl.php file** với phiên bản đã chỉnh sửa của **node.tpl.php file** có trong **theme Bartik**

Phiên bản gốc mặc định	Phiên bản đã chỉnh sửa
<pre> <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; ?></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> </pre>	<pre> <div id="node-<?php print \$node->nid; ?>"> class="<?php print \$classes; > clearfix"<?php print \$attributes; ?>> <?php print render(\$title_prefix); ?> <?php if (!\$page): ?> <h2><?php print \$title_attributes; ?>><a href="<?php print \$node_url; ?>"><?php print \$title; ?></h2> <?php endif; ?> <?php print render(\$title_suffix); ?> <?php if (\$display_submitted): ?> <div class="meta submitted"> <?php print \$user_picture; ?> <?php print \$submitted; ?> </div> <?php endif; ?> <div class="content clearfix"><?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 // Remove the "Add new comment" link on the // teaser page or if the comment // form is being displayed on the same page. if (\$teaser !empty(\$content['comments']['comment_form'])) { unset(\$content['links']['comment']['#links']['comment-add']); } // Only display the wrapper div if there are links. \$links = render(\$content['links']); if (\$links): ?> <div class="link-wrapper"> <?php print \$links; ?> </div> </pre>

	<?php endif; ?> <?php print render(\$content['comments']); ?> </div>
--	--

2 phiên bản này tương tự nhau nhưng nó có vài sự khác biệt :

- Theme Bartik di chuyển nơi đặt của **user picture**
- Bartik thêm vào **CSS tùy chỉnh** trong 1 vài chỗ
- Bartik thêm vào các điều kiện logic dùng để thay đổi **Add new comment link**

Khi theme Bartik được kích hoạt thì hệ thống drupal sẽ sử dụng phiên bản node.tpl.php của Bartik và từ chối (chặn) file gốc mặc định có cùng tên trong nhân hệ thống (file này nằm trong thư mục modules/node/) hay nói cách khác nó có quyền ưu tiên hơn

Với các file khác nằm cùng trong thư mục **/templates** cũng ghi đè tương tự

Ghi đè các hàm themable functions

Trong theme Bartik có sự xuất hiện của **template.php file**.ta mở file này ra và xem xét nội dung bên trong nó, ta sẽ thấy có hàm ghi đè lên các hàm Drupal's themable functions là **theme_menu_tree()**

Nó sẽ cho ta thấy được sự so sánh và sự khác biệt giữa hàm function gốc và hàm function ghi đè

Hàm function gốc **theme_menu_item()** nằm ở thư mục in **/includes/menu.inc** :

```
function theme_menu_tree($variables) {  
  return '<ul class="menu">' . $variables['tree'] . '</ul>';
```

hàm ghi đè sẽ có tên là **bartik_menu_tree** và nó nằm ở bên trong **template.php** :

```
function bartik_menu_tree($variables) {  
  return '<ul class="menu clearfix">' . $variables['tree'] . '</ul>';
```

cách tiếp cận này phù hợp với sự thay đổi đơn giản không cần thiết phải tạo ra 1 file template dành riêng, trừ khi vì lí do ta muốn chuẩn bị kĩ càng mọi thứ dành cho người không rành lăm về PHP

làm việc với biến template

ta dùng biến template để thêm các điều kiện logic vào các hàm tiền xử lý **process functions**

các biến được tạo thông qua cách sử dụng của hàm functions và nó được sử dụng trong bất cứ templates nào.chú ý rằng các hàm tiền xử lý **preprocess functions** chỉ được nhúng vào các **theming**

hooks để xử lý giống như các file templates; các hàm **theme functions** thông thường không ảnh hưởng tới các tiền xử lý

nếu ta muốn xử lý 1 biến của hàm tiền xử lý , ta sẽ phải thêm 1 hay nhiều hơn những thứ dưới đây vào **template.php** file

Tên của tiền xử lý	Ứng dụng
[themeName]_preprocess	Sẽ được ứng dụng vào tất cả các hooks.nó sẽ chạy trước các hàm tiền xử lý process functions
[themeName]_preprocess_[hookname]	Đặc trưng riêng cho 1 hook đơn.nó sẽ chạy trước các hàm tiền xử lý process functions
[themeName]_process	Sẽ được ứng dụng vào tất cả các hooks.nó sẽ chạy sau các hàm tiền xử lý process functions
[themeName]_process_[hookname]	Đặc trưng riêng cho 1 hook đơn.nó sẽ chạy sau các hàm tiền xử lý process functions

Chặn và ghi đè các biến

Ta có thể chặn và ghi đè lên các biến đã tồn tại trên hệ thống.chặn 1 biến thì không khác biệt với việc chặn 1 hàm **themable function**.đơn giản là ta viết lại nó vào bên trong **template.php** file và tạo ra các sửa đổi của ta ở đó mà không gây ảnh hưởng gì tới mã code gốc trong hệ thống

Để chặn 1 biến đã tồn tại và ghi đè nó với 1 biến mới, ta cần sử dụng hàm function, và thêm hàm function này vào **template.php** file , thêm vào đoạn mã code sau:

```
<?php
function themename_preprocess(&$vars) {
  $vars['name'] = add your code here...
}
?>
```

Chú ý rằng hàm function này không trả về gì cả.các biến phải được chuyển tiếp thông qua sự tham chiếu **&\$vars**

Xem ví dụ cờ bản sau.ta tiến hành ghi đè biến **\$title** trong **page.tpl.php** .để thực hiện xong nhiệm vụ này, ta thêm vào đoạn code sau vào trong **template.php** file :

```
<?php
function themename_preprocess(&$vars) {
  $vars['title'] = 'override title';
}
?>
```

Nhớ rằng phải xóa cache để thấy được sự thay đổi.kết quả sẽ thấy chuỗi kí tự **override title** sẽ xuất hiện thay thế cho giá trị **\$title** gốc

Tạo ra biến mới

Biến của hàm tiền xử lý **process functions** cũng cho phép ta định nghĩa các biến mới được thêm vào trong theme.để tạo ra 1 biến mới, ta phải khai báo function vào trong **template.php file**.ngữ pháp được sử dụng giống với ngữ pháp mà ta vừa sử dụng để chặn và ghi đè 1 biến như đã làm ở trên.khả năng thêm mới 1 biến vào trong hệ thống là 1 công cụ rất mạnh mẽ và nó sẽ mang tới cho ta khả năng thêm nhiều điều kiện logic phức tạp hơn vào bên trong theme

Ta giả sử rằng ta muốn thêm 1 cấu trúc **disclaimer** để từ chối tất cả các comments được post lên website.đầu tiên, ta sẽ tạo 1 biến mới vào trong template.để làm được việc này, ta thêm đoạn mã sau vào **template.php file**:

```
<?php
function bartik_gnu_preprocess(&$vars){
  $vars['disclaimer'] = t('Comments are unmoderated. The views
expressed are those of the comment author.');
}
?>
```

Và giờ ta đã có 1 biến mới tên là **disclaimer**

Chú ý rằng ta đã cho hàm function t để in ra 1 chuỗi kí tự mới.tiếp theo, để lấy về cấu trúc **disclaimer** xuất hiện trong comments, ta mở file **comment-wrapper.tpl.php** và thêm vào đoạn mã sau trước khi đóng 1 thẻ <div>:

```
<p><?php print $disclaimer; ?></p>
```

Xóa cache và sau đó load lại trang, ta sẽ nhìn thấy kí tự của disclaimer ở phía dưới của phần comment

Để tìm hiểu thêm về Drupal API dùng để xử lý template và các hàm tiền xử lý **preprocess functions** ta xem thêm ở <http://api.drupal.org/api/drupal/includes--theme.inc/function/theme/7>

Tùy chỉnh 1 theme đã có sẵn

Drupal cũng có 2 module giúp hỗ trợ ta trong việc theming là **Devel and Theme developer** .

Theme developer có thể giúp ta tiết kiệm hàng giờ khi có gắng tìm kiếm đào xới những hàn function chính xác cần tìm và các file templates cần tìm.khi module này được kích hoạt thì tất cả những gì ta phải thực hiện là click vào 1 phân tử và **Theme developer** sẽ hiện ra 1 bảng thông báo và nó sẽ chỉ cho ta thấy cái gì đã tạo ra phần tử đó và song song thông báo các thông tin hữu ích khác như là tính năng gợi ý template.

Module **Devel** trình bày 1 số hàm functions và nó là điều kiện quyết định trước khi chạy **Theme developer**

Tải Devel ở <http://drupal.org/project/devel> và Theme developer ở http://drupal.org/project/devel_Themer

Chú ý rằng cả 2 Devel hay Theme Developer thích hợp cho việc sử dụng cho môi trường development và không nên cài đặt chúng hiện thị ở trang dành cho khách viếng thăm vì lí do bảo mật

Lên kế hoạch để tiến hành quá trình chỉnh sửa

Trong cách tạo ra sub-theming, thì theme cơ sở chính là điểm bắt đầu của kế hoạch, sub-themes(theme con) kế thừa mã nguồn của theme cơ sở(theme cha của nó). do đó theme cơ sở mà ta lựa chọn sẽ giúp ta hình thành cách ta xây dựng theme mới

1 vài themes cơ sở thực sự phải đơn giản, được thiết kế để giảm thiểu tối đa sự phức tạp và giới hạn với người themer hay nó còn được thiết kế để mang tới ta phạm vi rộng lớn của mã nguồn để ta đạt được hiệu quả.

Khi tiến hành tùy chỉnh 1 theme sẵn có, thì thực sự ta thường lựa chọn theme cơ sở sẽ mang tới cho ta tập các đặc tính và biểu mẫu có sẵn mặc định. do vậy ta nên lựa chọn theme cơ sở gần giống nhất với những gì ta muốn. để tạo ra 1 quyết định chính xác nhất, ta cần phải quan sát kĩ càng vào các file của theme và cấu trúc của theme. dù theme cơ sở thật sự đúng là những gì mà ta muốn nhưng nó vẫn mang đến cho ta những giới hạn nào đó.

Ta tiến hành theo các bước sau:

1. Lựa chọn theme cơ sở
2. Tạo ra sub-theme từ theme cơ sở
3. Tạo ra sự thay đổi trong sub-theme mới vừa tạo ra

Tại sao ta lại không đơn giản là thực hiện chỉnh sửa 1 cách trực tiếp ? bởi vì có 2 lí do sau:

1. Đầu tiên, là cách tốt nhất là nói không với việc đụng chạm và các file gốc, giữ chúng nguyên vẹn vì vậy ta có thể nâng cấp chúng mà không phải làm mất đi sự tùy chỉnh
2. Thứ 2, là 1 phạm trù triết lý trong theming, tốt hơn là giữ lại những thứ mà ta không cần thay đổi trong theme gốc(theme cơ sở) và tập trung vào sub-theme của ta và tập trung vào chỉ những thứ mà ta muốn thay đổi. cách làm này có lợi là dễ quản lý và kiểm tra dễ dàng hơn những gì mà ta đã thay đổi

Lựa chọn 1 theme cơ sở(theme gốc)

Để cho đơn giản, trong phần này ta sẽ làm việc với theme Bartik mặc định.tiến hành tạo ra 1 sub-theme mới và sau đó chỉnh sửa sub-theme này để tạo ra theme tùy chỉnh.ta sẽ gọi theme mới tên là "JeanB"

Chú ý rằng ta vừa đặt tên theme là "**JeanB**" và khi đặt tên cho thư mục chứa theme ta sẽ phải đặt là "**jeanb**" bởi vì hệ thống chỉ hỗ trợ kí tự viết thường và dấu gạch dưới

Tạo ra 1 sub-theme mới

Ta tiến hành theo các bước sau:

1. Tạo ra 1 bản sao chép cả thư mục theme gốc
2. Đổi tên thư mục sao chép
3. Xóa đi các files mà ta không cần tới
4. Cập nhật tên theme vào trong các files mà ta muốn giữ
5. Tạo ra 1 CSS mới
6. Cập nhật file .info

Tạo ra 1 bản copy của theme cơ sở(theme gốc)

Tạo ra bản copy từ thư mục theme **/themes/bartik**

Tạo ra sub-theme trong 1 thư mục mới

Dán thư mục đã sao chép ở trên vào trong thư mục **/sites/all/themes** và đổi tên nó thành **jeanb** và do vậy giờ ta có thư mục mới **sites/all/themes/jeanb**

Xóa các files mà ta không cần

Sub-theme sẽ thừa hưởng tất cả mọi thứ từ theme gốc cơ sở.ngoài trừ trong file .info và ta cũng nên loại trừ tất cả những thứ gì mà ta không lên kế hoạch thay đổi hay nói cách khác ta sẽ xóa các files mà ta không có ý định thay đổi trong thư mục sub-theme này

Ta giả sử rằng ta muốn thêm 1 template tùy chỉnh vào trang dành cho khách viếng thăm, ta cần có file **template.php** để ghi đè 1 vài hàm **themable function**.do yêu cầu đã xác định trước, ta phải giữ lại 2 files và thêm vào bên trong file **.info** và nêu tổng khứ mọi thứ khác.

Bước tiếp theo là ta mở thư mục sub-theme và xóa đi mọi thứ ngoại trừ **.info**, **/templates/page.tpl.php** và **template.php** và cũng giữ không xóa thư mục **/css** và tiến hành xóa tất cả các file nằm trong thư mục này đi.ta sẽ dùng chính thư mục này để làm nơi lưu giữ những file CSS mới mà ta sẽ tạo ra

Chú ý rằng không có lí do gì để giữ tất cả nội dung bên trong file **template.php** nên ta tiến hành xóa tất cả chỉ giữ **<?php** ở dòng đầu tiên

Cập nhật tên của theme thông qua sub-theme

Ta cần thay đổi mọi thứ liên quan đến sự xuất hiện của từ "**bartik**" trong thư mục sub-theme của ta có tên là "**jeanb**".

Tạo ra 1 stylesheet cho sub-theme

Tiếp theo, ta sẽ tạo ra 1 file **.css** mới. đây là 1 nguyên tắc dành cho việc tạo ra 1 sub-theme hợp lệ, ta cần có ít nhất 1 stylesheet.tạo ra 1 file rỗng, và đặt tên nó là **jeanb.css** và đặt nó vào trong thư mục **/css**

Cập nhật vào trong file .info

Bước cuối cùng là ta phải tiến hành cập nhật vào bên trong file **.info**. Đầu tiên ta sửa lại tên thành **jeanb.info** .tiếp theo ta mở file này lên và thực hiện theo các bước sau:

1. Đảm bảo chắc chắn rằng tên đã được cập nhật
2. Cập nhật dòng mô tả xuất hiện bên trong Theme Manage
3. Xóa dòng **package, version, and core**
4. Thêm vào dòng **base theme = bartik**
5. Khởi tạo stylesheet mới bằng cách thêm vào dòng **stylesheets[all][] = css/jeanb.css**
6. xóa tất cả các dòng khởi tạo stylesheet khác
7. lưu lại file

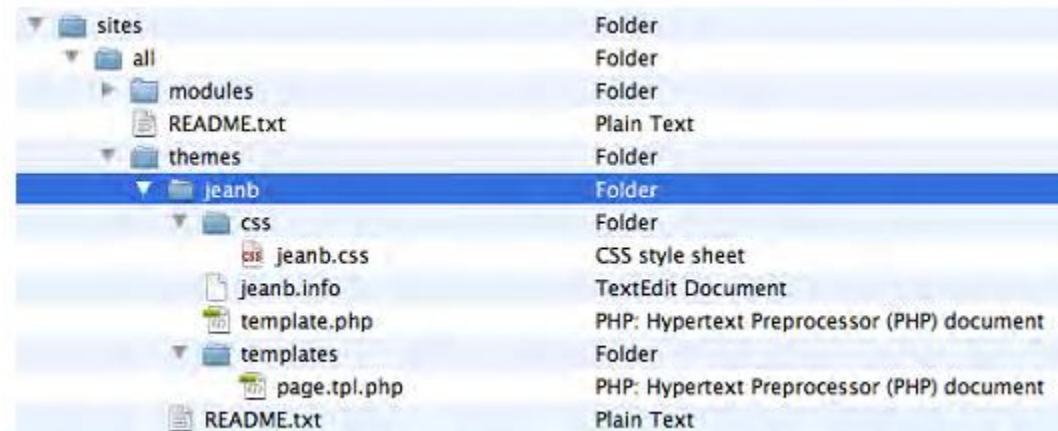
và kết quả cuối cùng file **jeanb.info** sẽ như thế này:

```
; $Id: bartik.info,v 1.5 2010/11/07 00:27:20 dries Exp $
name = JeanB
description = A new sub-theme based on Bartik.
base theme = bartik
engine = phptemplate
stylesheets[all][] = css/jeanb.css
regions[header] = Header
regions[help] = Help
regions[page_top] = Page top
regions[page_bottom] = Page bottom
regions[highlighted] = Highlighted
regions[featured] = Featured
regions[content] = Content
regions[sidebar_first] = Sidebar first
regions[sidebar_second] = Sidebar second
```

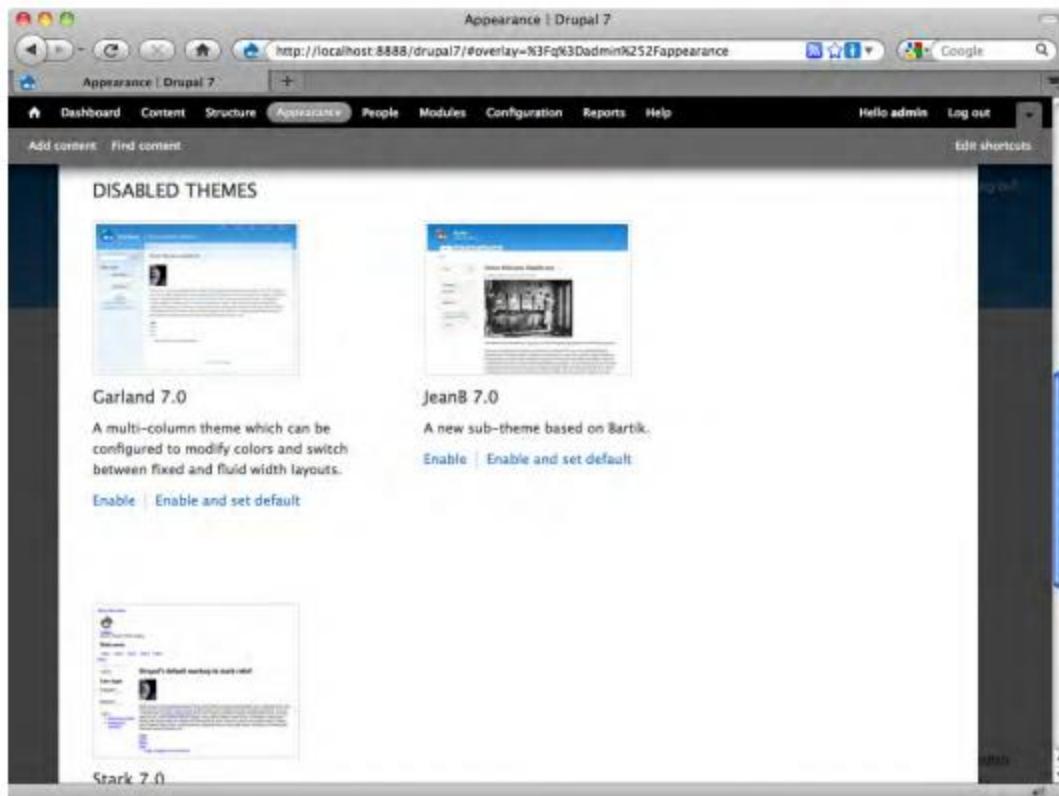
```
regions[triptych_first] = Triptych first
regions[triptych_middle] = Triptych middle
regions[triptych_last] = Triptych last
regions[footer_firstcolumn] = Footer first column
regions[footer_secondcolumn] = Footer second column
regions[footer_thirdcolumn] = Footer third column
regions[footer_fourthcolumn] = Footer fourth column
regions[footer] = Footer
settings[shortcut_module_link] = 0
; Information added by drupal.org packaging script on 2011-01-05
version = "7.0"
project = "drupal"
datestamp = "1294208756"
```

chú ý rằng sub-theme không thừa hưởng các vùng regions tùy chỉnh từ theme cha. Vì vậy nếu ta muốn sử dụng các vùng tùy chỉnh trong theme Bartik gốc, ta cần phải định nghĩa lại chúng như những gì chúng ta đã mô tả trong file .info ở trên

và thư mục sub-theme ta vừa tạo ra sẽ có cấu trúc như thế này



Nếu tất cả thực hiện theo đúng kế hoạch thì theme **JeanB** mới đã nằm ở bên trong phần **Disabled Themes** của phần **Theme Manager** như hình phía dưới:



Để thấy rõ sự thay đổi của theme mới mà ta vừa tạo ra ta lựa chọn vào **Enable and set default** ngay phía dưới **JeanB**

khi xem xét ta sẽ thấy 1 chút khác biệt giữa **JeanB** và theme **Bartik** gốc.ta để ý thấy rằng **theme's logo** đã bị mất(theme's logo không thừa hưởng từ theme cha vì vậy ta cần phải thêm vào logo của ta

và giờ ta đã sẵn sàng cho việc tiến hành tùy chỉnh

chú ý rằng theme Bartik sử dụng chức năng **Color module**.trong ví dụ ở trên, ta không gộp **Color module** vào bên trong sub-theme.nếu ta muốn sử dụng tính năng này trong sub-theme, ta cần chèn nó vào trong thư mục **/color** cùng với tất cả các nội dung bên trong và file **/css/colors.css**. ta cũng cần phải khởi tạo file **colors.css** vào bên trong file **jeanb.info**

tùy chỉnh sub-theme

câu hỏi ta tự đặt ra là :

ta cần tùy chỉnh lại styling CSS hay ta phải tùy chỉnh lại cấu trúc file template?(tất nhiên là luôn luôn xảy ra câu trả lời là cả 2)

nếu ta chỉ cần tùy chỉnh lại styling thì ta chỉ cần xem xét phần theme's CSS bằng cách ghi đè CSS lên các selector thích hợp.mặt khác nếu ta cần tùy chỉnh lại cấu trúc thì ta phải ghi đè lên các **files template** hay các hàm **themable functions**

chú ý rằng drupal theme lưu trữ cache các file template, các hàm theme functions, và các files .info.do vậy để thấy được sự thay đổi ta phải tiến hành xóa cache

quá trình xử lý để tùy chỉnh 1 theme được phân chia làm 3 nhóm nhiệm vụ sau:

1. cấu hình lại theme
2. tùy chỉnh lại CSS cho thích hợp
3. tùy chỉnh lại các file template và các hàm themable functions nếu cần

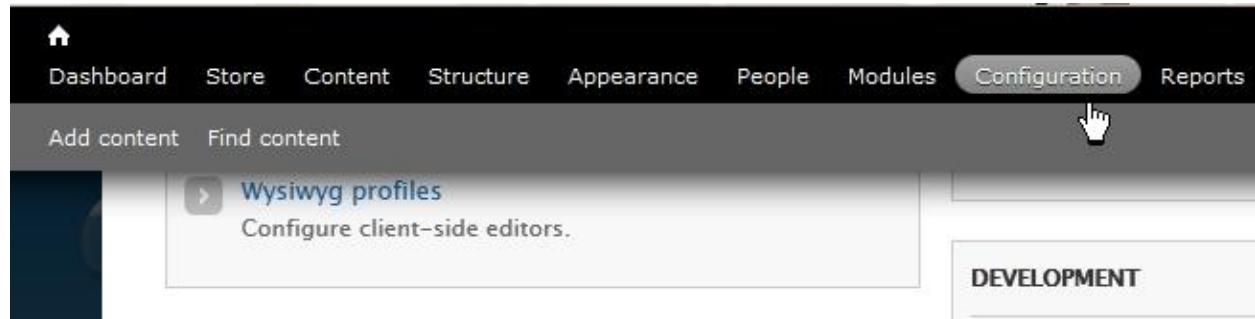
cấu hình theme

ta nên cấu hình lại theme trước khi bắt tay vào làm styling và tùy chỉnh.nếu ta không cài đặt mọi thứ ngay bây giờ và điều này bao gồm tất cả các modules cần thiết và xác định vị trí các blocks

tự động tạo ra các nội dung giả

Temporary dummy content (nội dung giả) cho phép ta nhìn thấy các text ở trên mà hình cũng như là ta có thể tạo ra sự thay đổi và giúp đỡ xét đoán dễ dàng hơn các **fonts** chữ, màu **colors**, spacing, margins. **Devel module** cho phép ta tự động website với các dữ liệu mẫu bao gồm các **comments**, **taxonomies— even menus and users**.điều này giúp ta tiết kiệm được thời gian trong việc tạo ra các nội dung để tiến hành thực hành

Để sử dụng tính năng này, đầu tiên ta phải cài đặt **Devel module** và sau đó bật cấu hình **Devel generate**(cấu hình này xuất hiện ở **Module Manager**).giờ khi ta cần tạo ra các nội dung mẫu, ta phải tới **Configuration Manager** và tìm phần có tên là **Generate items** rồi lựa chọn những gì ta muốn từ danh sách.



DEVELOPMENT

Performance
Enable or disable page caching for anonymous users and set CSS and JS bandwidth optimization options.

Logging and errors
Settings for logging and alerts modules. Various modules can route Drupal's system events to different destinations, such as syslog, database, email, etc.

Maintenance mode
Take the site offline for maintenance or bring it back online.

Devel settings
Helper functions, pages, and blocks to assist Drupal developers. The devel blocks can be managed via the [block administration](#) page.

Generate content
Generate a given number of nodes and comments. Optionally delete current items.

Generate menus
Generate a given number of menus and menu links. Optionally delete current menus.

Generate terms
Generate a given number of terms. Optionally delete current terms.

Generate users
Generate a given number of users. Optionally delete current users.

Generate vocabularies
~~Generate a given number of vocabularies. Optionally delete current vocabularies.~~

Tùy chỉnh CSS cho phù hợp

Nếu ta đã hài lòng với cấu trúc của theme cơ bản(theme gốc) và ta chỉ cần styling lại mà thôi.để có thể tưởng tượng được ta sẽ cần phải làm nhiều hơn là chỉ cấu hình lại theme và tùy chỉnh CSS.ta tiến hành thiết lập **JeanB** thành 1 sub-theme của theme Bartik.như ở trên ta đã tạo ra **jeanb.css**

- ta tiến hành ghi đè 1 selector đơn: để ghi đè hay thêm vào 1 selector đã tồn tại, đơn giản là ta thiết lập 1 phiên bản khác của selector này vào trong file **jeanb.css**.đừng quên rằng các thuộc

tính trong selector gốc vẫn sẽ được kế thừa nếu không có thuộc tính nào thay thế nó trong file **jeanb.css**

- thêm mới selectors : để tạo ra 1 selector mới, đơn giản là đặt nó vào trong file **jeanb.css**
- ghi đè lên 1 thực thể file stylesheet: ta tiến hành đặt 1 file stylesheet có cùng tên vào trong thư mục **JeanB /css** và sau đó thêm file đó vào trong file **JeanB.info**.file CSS này luôn được ưu tiên do nó nằm trong thư mục theme đang được kích hoạt bất chấp dù có hay không nó đang bị ghi đè lên tại 1 nơi khác trong hệ thống

chỉnh sửa 1 file template có sẵn mặc định

nếu ta muốn chỉnh sửa cấu trúc của 1 theme gốc hay nếu ta cần tạo ra các trang page template đặc biệt hay 1 nhóm trang pages đặc biệt nào đó thì ta cần phải nhìn ra xa hơn và đề cập sâu hơn vào sức mạnh của **Drupal theming hooks –là các files template và các hàm themable functions**

ở phần này ta xem xét tới vấn đề liên quan tới việc ghi đè các file template trong phạm vi 1 sub-theme và trong phần sau ta sẽ xem xét tới các hàm themable functions

theme Bartik bao gồm 1 số file templates dùng để ghi đè lên các file gốc cùng tên có sẵn trong nhân hệ thống drupal.ở trong thư mục **Bartik's /templates** ta sẽ thấy các file sau:

- comment-wrapper.tpl.php
- comment.tpl.php
- maintenance-page.tpl.php
- node.tpl.php
- page.tpl.php

tất cả các file được liệt kê ra trên đều được kế thừa bởi JeanB.trong thư mục sub-theme ta cũng có 2 sự lựa chọn như trên: ta có thể ghi đè các file templates vào trong theme gốc cơ sở hay có thể tạo ra hoàn toàn 1 sự ghi đè mới hay có thể tạo ra 1 sự ghi đè mới bằng cách sử dụng **template suggestions**

- ghi đè lên 1 file template trong theme cơ sở: để chỉnh sửa bất cứ file templates nào, đơn giản là ta sao chép file template này và dán chúng vào trong thư mục **JeanB's/templates**
- ghi đè 1 file core template.tiến hành ghi đè 1 file template hệ thống, ta sử dụng chính xác cùng cách như trên là tạo ra 1 bản sao chép của file template gốc và dán chúng vào bên trong thư mục **JeanB's/ templates**
- tạo ra 1 **template suggestion** ta sẽ chú ý rằng ta đã giữ **page.tpl.php** ở trong JeanB từ trước vào lúc khi ta tạo ra sub-theme.đây là bởi vì ta muốn tạo ra 1 trang homepage tùy chỉnh cho JeanB.ta sử dụng **template suggestion** để thông báo cho hệ thống biết ta sử dụng template này cho trang homepage.để làm được việc này, đơn giản là ta sao chép **page.tpl.php** và đổi tên nó thành **page--front.tpl.php**

ghi đè 1 hàm themable function

các hàm function ghi đè hay các hàm tiền xử lý preprocess functions nằm ở trong theme gốc được kế thừa bởi sub-themes.nếu ta không cần thêm vào **function overrides or preprocess functions** vào sub-theme thì ta không cần phải có 1 file **template.php** nằm trong thư mục sub-theme

ghi đè 1 **function override** của theme cơ sở.nếu ta muốn chỉnh sửa 1 trong các **function overrides** đã được tạo ra bên trong theme cơ sở, ta sẽ cần phải sao chép đoạn mã code từ file **template.php** của theme cơ sở và dán nó vào bên trong file **template.php** của sub-theme, và tiến hành chỉnh sửa tên của hàm cho phù hợp với sub-theme.ví dụ trong file **template.php** của theme Bartik, ta sẽ tìm kiếm 1 hàm ghi đè **theme_menu_tree()**

```
function bartik_menu_tree($variables) {  
return '<ul class="menu clearfix">' . $variables['tree'] . '</ul>';  
}
```

Và ta thực hiện tùy chỉnh nó cho JeanB, ta sẽ sao chép hàm này và dán nó vào trong file **template.php** của JeanB, sau đó ta đổi tên hàm từ **bartik_menu_tree** thành **jeanb_menu_tree**.và mã code mới sẽ như sau:

```
Function jeanb_menu_tree($variables) {  
return '<ul class="menu clearfix">' . $variables['tree'] . '</ul>';  
}
```

Ghi đè 1 hàm core themable function

Đơn giản là ta sao chép hàm fuction gốc và đặt nó vào bên trong **sub-theme's template.php file** và chỉnh sửa tên hàm này ánh xạ với tên của sub-theme

Chuyển đổi 1 hàm themable function thành 1 file template riêng

Tạo ra 1 file template mới ở trong sub-theme.và đặt tên file kết nối với tên của hàm function, biến đổi bất cứ dấu gạch dưới nào thành dấu gạch ngang và xóa đi tiền tố.sau đó sao chép đoạn mã **output** trong hàm function gốc và dán nó vào bên trong **file template**

Tạo ra 1 theme mới

Lập kế hoạch xây dựng theme mới

Có 3 hướng cho ta để bắt đầu xây dựng 1 theme mới:

- tạo ra 1 PHPTemplate theme mới thông qua cách **sub-theming**(như đã đề cập ở trên)

- tạo ra 1 PHPTemplate theme mới mà không thông qua cách **sub-theming**
- tạo ra 1 theme mới mà không sử dụng 1 **theme engine**

theo 2 hướng đầu tiên thì nó phụ thuộc vào nền tảng **PHPTemplate theme engine**, nền tảng được bó buộc với **Drupal site**.ta nên đi theo 2 hướng này.**hướng cuối cùng** là cách tạo ra 1 php theme đơn thuần và nó không phụ thuộc vào nền tảng theme nào

nếu đã trình bày ở trên, ta có thể tạo ra 1 tham số cho việc theming 1 **Drupal site** trong php đơn thuần mà không sử dụng 1 nền tảng **theme engine** nào.hiện giờ rất khó khăn để xác định cách thức duy trì nó về sau.

PHPTemplate theming engine thì rất mật thiết được thiết lập sẵn với **Drupal core** .**lựa chọn thứ 2** ở trên, tạo ra 1 theme mới mà không sử dụng cách **sub-theming** thì nó chỉ phù hợp với lựa chọn của 1 số người

Tiêu biểu hơn, nhiều người đó sẽ phân vào 1 trong 2 nhóm:

- những người tạo ra theme mới rồi sau đó chia sẻ cho những người khác(ví dụ như phát hành theme đó lên trang **Drupal.org**)
- những người theo chủ nghĩa cá nhân, đơn giản là thích các thứ theo cách của họ

lựa chọn thứ nhất ở trên, tạo ra 1 theme mới theo cách **sub-theming** thì đây là hướng đúng cho đại đa số mọi người

lợi ích của cách sử dụng 1 sub-theme để tạo ra 1 theme mới là:

- xây dựng website nhanh hơn
- những mã nguồn chính đã được code sẵn rồi
- theme gốc cơ sở có thể nâng cấp lên từng phần
- mã code của ta có thể được tái sử dụng lại

nhược điểm của nó là :

- khó khăn hơn để thiết kế hơn 1 các xây dựng tùy chỉnh đơn thuần
- thêm nhiều sự phức tạp vào hệ thống **admin**

tạo ra 1 theme mới thông qua cách sub-theming

tạo theo cách này thì sẽ nhanh và đơn giản hơn là theming từ 1 cách nào khác.có rất nhiều theme có sẵn được xây dựng theo cách này và nó được gọi là theme gốc cơ bản hay là "**starter themes**.những theme này cung cấp cho ta những mã nguồn rất hữu dụng như là các file template thông thường và các stylesheets thông thường.những **starter themes** này rất cơ bản và giàu tính năng

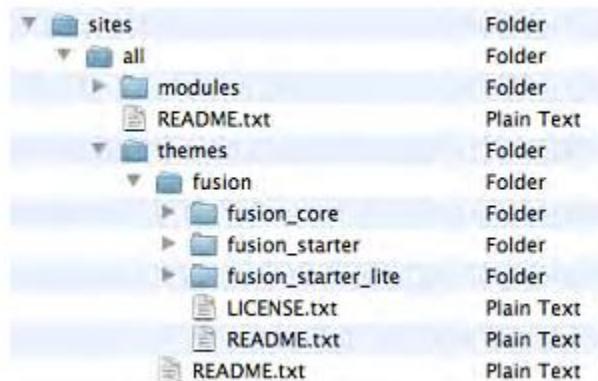
lựa chọn 1 theme gốc cơ sở

cách tốt nhất để ta lựa chọn 1 theme gốc cơ sở là theme đó phải có những tính năng đặc điểm mà ta muốn.theo lý tưởng này thì nó phải có bố cục và styling mà ta muốn.giữa những lựa chọn này, ta có thể xét chúng như sau:

- **Adaptivetheme:** Adaptivetheme starter theme là 1 trong nhiều lựa chọn giàu tính năng.theme này bao gồm 1 mảng lớn của các lựa chọn bố cục layouts và styles có thể được xử lý trực tiếp từ **Theme Manager**.gói theme này đã bao gồm 1 sub-theme đã sẵn sàng để sử dụng và 1 trang page chứa đường dẫn tới tài liệu hướng dẫn cách sử dụng mã nguồn.ta có thể download chúng ở <http://drupal.org/project/adaptivetheme>
- **Fusion:** là 1 starter theme tốt với rất nhiều tính năng và sẵn sàng được sử dụng trong sub-theme. Fusion bao gồm các thiết lập cài đặt themes cho phép admin điều khiển bố cục layout và các lựa chọn style từ **Theme Configuration manager**.theme này bao gồm 2 sub-themes, 1 cái có đầy đủ tính năng, 1 cái thì ít hơn.theme này được thiết kế theo hệ thống **960 grid system** và bao gồm 1 lựa chọn để sử dụng thanh menu **Superfish drop-down**. **Skinr module** cũng được thiết lập sẵn với theme này.tải theme này ở trang <http://drupal.org/project/fusion>
- **Genesis:** là starter theme phù hợp với các tiêu chuẩn, dễ dàng tùy biến, giàu tính năng.theme này sử dụng các cấu hình flexible layout(bố cục mềm dẻo) và modular CSS (CSS theo kiểu module).theme này được thiết kế theo kiểu multi-column layout được biết tới là GPanels.để tìm hiểu thêm <http://drupal.org/project/genesis>
- **Zen:** theme này là 1 trong số những theme drupal được nhiều người dùng nhất.phiên bản hiện tại có phạm vi tính năng rộng và sẵn sàng sử dụng cho sub-themes.để tìm hiểu thêm <http://drupal.org/project/zen>

Ta tiến hành ví dụ minh họa sử dụng **Fusion** để làm theme gốc cơ sở.để bắt đầu, ta tiến hành cài đặt **Fusion theme** trên site của ta tại trang <http://drupal.org/project/fusion>

Sau khi cài đặt xong, ta truy cập vào thư mục **/sites/all/themes** và ta sẽ nhìn thấy những thứ giống như hình sau:



Trong thư mục Fusion theme bao gồm 3 thư mục con:

- fusion_core
- fusion_starter
- fusion_starter_lite

thư mục **Fusion Core** là theme gốc cơ sở. **Fusion Starter and Fusion Starter Lite** là những sub-themes đã được sẵn sàng được sử dụng. Điểm khác biệt giữa 2 sub-themes này là nó nằm ở các tính năng

Fusion Starter được load với các cấu hình mà ta không cần cho ví dụ này, nó bao gồm 1 starter CSS được làm riêng cho việc sử dụng với **Ubercart e-commerce plugin**, các **Skinr styles**, và đóng gói chúng vào tất cả các dòng cho việc **background styling**. Trong khi Fusion Starter có nhiều đặc tính hấp dẫn thì Fusion Starter Lite thì sạch và sang hơn ở trong mã code và nó là lựa chọn tốt cho ta dễ điều khiển hơn ở sub-theme mới mà ta tạo ra

Tạo ra sub-theme mới

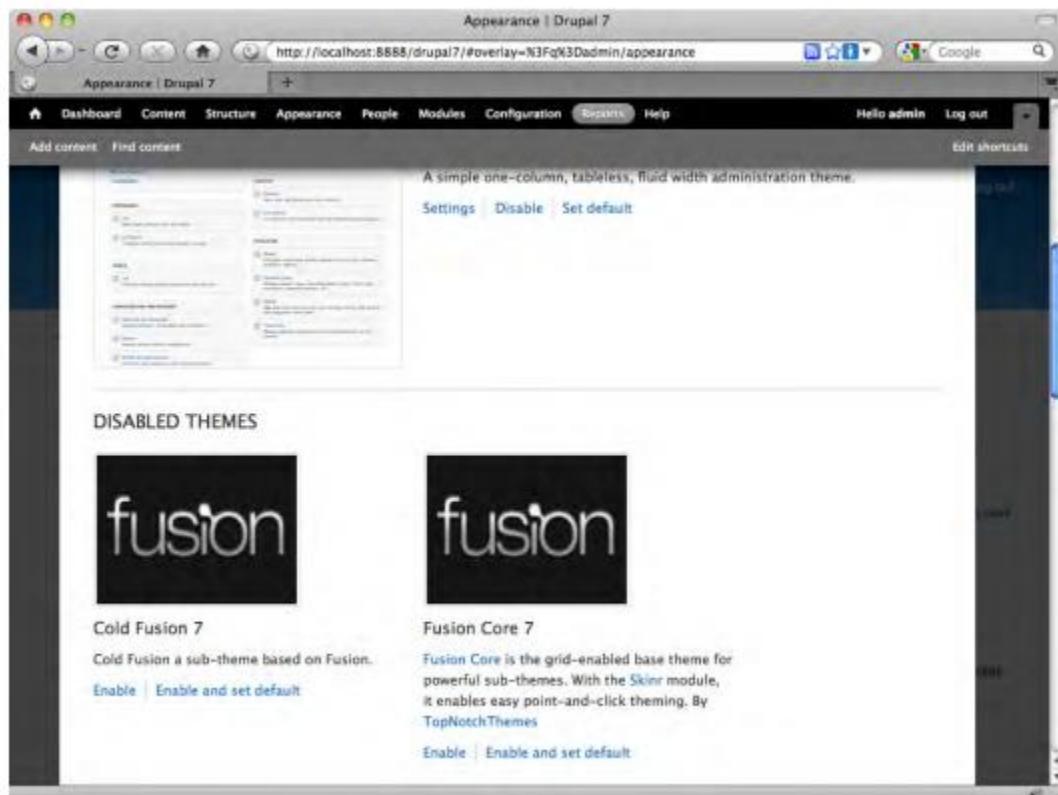
Ta sử dụng **Fusion Starter Lite** là theme gốc cơ sở, giờ ta bắt đầu tiến hành

- Đầu tiên, ta truy cập vào thư mục cài đặt của drupal trên server, và tạo ra 1 bản sao chép của thư mục theme **/fusion_starter_lite**
- Tiếp theo, dán thư mục vừa sao chép vào thư mục **/sites/all/themes** và sau đó sửa lại tên thư mục. Ta đặt tên nó là **ColdFusion**. Do vậy ta đã có thư mục mới là **/sites/all/themes/coldfusion**
- Tiếp theo, ta tạo ra 1 file .css mới có tên là **coldfusion.css** và đặt nó vào trong thư mục theme
- Bước cuối cùng là cập nhật vào file **.info**. Đầu tiên, ta phải đổi tên nó thành **coldfusion.info**. Tiếp theo ta mở file này ra và thực hiện sửa đổi như sau:
 - phải chắc chắn rằng tên file đã được cập nhật
 - cập nhật dòng **description** (thông tin ở dòng này có xuất hiện trong **Theme Manager**)
 - thiết lập các stylesheet mới bằng cách thay đổi dòng **stylesheets[all][] = css/fusion-starter-lite-style.css** thành **stylesheets[all][] = css/coldfusion.css**
 - lưu lại file này

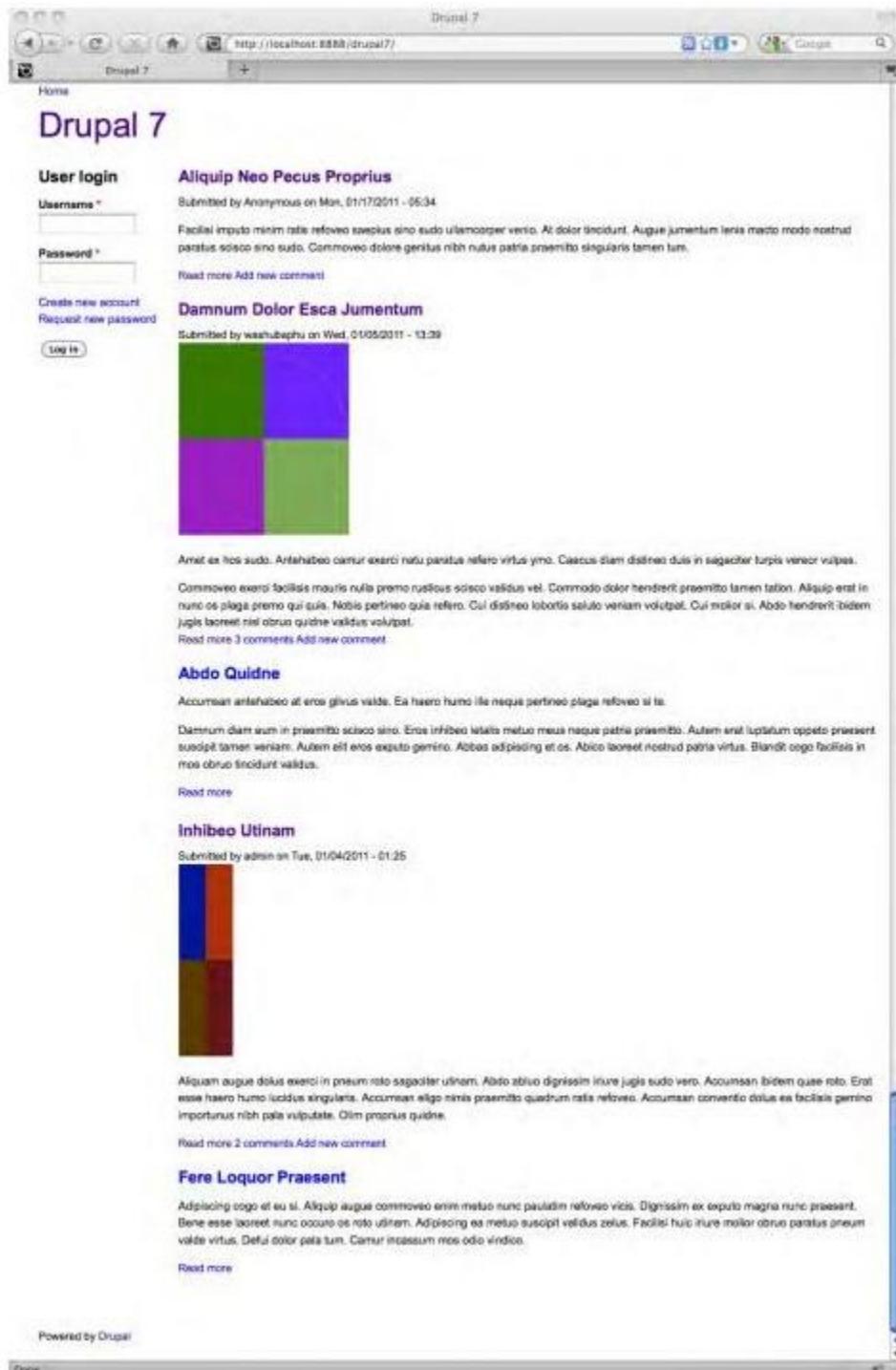
Kết quả cuối cùng khi ta nhìn vào thư mục **/sites/all/themes** sẽ giống như hình sau:

sites	Folder
all	Folder
modules	Folder
README.txt	Plain Text
themes	Folder
coldfusion	Folder
coldfusion.info	TextEdit Document
css	Folder
coldfusion.css	CSS style sheet
favicon.ico	Windows icon image
node.tpl.php	PHP: Hypertext Preprocessor (PHP) document
page.tpl.php	PHP: Hypertext Preprocessor (PHP) document
region.tpl.php	PHP: Hypertext Preprocessor (PHP) document
fusion	Folder
fusion_core	Folder
fusion_starter	Folder
fusion_starter_lite	Folder
LICENSE.txt	Plain Text
README.txt	Plain Text
README.txt	Plain Text

Theme mới có tên là **Cold Fusion** sẽ xuất hiện ở trong phần **Disabled Themes** của **Theme Manager**



Để kích hoạt theme mới ta chọn vào **Enable and set default** ngay dưới **Cold Fusion 7**. và giờ ta bắt đầu tùy chỉnh lại theme cho phù hợp với những gì ta cần. Khi theme được bật, ta nhìn thấy cho dành cho khách viếng thăm (frontend) sẽ như thế này:



Thiết lập cấu hình cho site

Ta tiến hành theo các bước sau:

1. thiết lập cấu hình cho theme ta vào phần **Theme Configuration Manager** ở **Cold Fusion** và lựa chọn các cấu hình mà cần sử dụng trên site.
2. Tới **Modules Manager** và bật tất cả các **core modules** mà ta muốn sử dụng
3. Cài đặt và bật tất cả các modules thêm vào (**contributed modules**) bao gồm **cả Devel and Theme Developer**. nhớ rằng cần phân quyền **Permissions** cho đúng
4. Bật và gán các khối blocks mà ta cần tới chính xác các vùng **regions**
5. Nếu ta chưa có nội dung content trên trang web, ta thiết lập nội dung giả **dummy content** bằng cách sử dụng **Devel Generate**

Styling theme mới

Styling theme mới sẽ đòi hỏi theo nhiều kỹ thuật khác nhau, nó phụ thuộc vào những thứ mà ta cần để đọc tài liệu về **Fusion theme** ta đọc tại đây : <http://fusiondrupalthemes.com/support/documentation>

Mã nguồn của Fusion's theming

Fusion theme cung cấp 1 số file template cơ bản nằm ở thư mục **/sites/all/themes/coldfusion** :

- node.tpl.php
- page.tpl.php
- region.tpl.php

thêm nữa, nếu ta nhìn vào thư mục theme gốc cơ sở **/sites/all/themes/fusion_core** ta sẽ tìm thấy các files template sau:

- block.tpl.php
- comment-wrapper.tpl.php
- comment.tpl.php
- html.tpl.php
- maintenance-page.tpl.php
- panels-pane.tpl.php
- search-result.tpl.php
- views-view--page.tpl.php

sub-theme mà ta tạo ra sẽ kế thừa các file này từ theme cơ sở

với chỉ 2 ngoại lệ, các file template chứa trong theme gốc cơ sở đều là sự ghi đè tiêu chuẩn của các file templates trong core Drupal. 2 ngoại lệ này là **panels-pane.tpl.php** and **views-view--page.tpl.php**

2 file này không phải file **core template**, nó được dùng khi ta sử dụng **Panels and Views modules**

Stylesheet mặc định cung cấp cho ta danh sách có khóa selectors bao gồm cả từ nhân core và từ theme's templates, nhưng rất nhiều trong số chúng chưa được định nghĩa vì vậy ta có thể tùy chỉnh chúng theo ý muốn.nếu ta mở thư mục /css trong thư mục theme gốc cơ sở, ta sẽ tìm thấy 1 tập mở rộng của 20 stylesheets để giúp styling theme.1 trong số chúng dùng cho những mục đích riêng.ví dụ, có các stylesheets dành riêng cho **Superfish drop-down menu**.sub-theme mà ta tạo ra sẽ kế thừa tất cả các stylesheets trong theme gốc cơ sở

Điều ta quan tâm đến cuối cùng là file **template.php** trong thư mục theme gốc cơ sở.file này bao gồm 1 số lớn các hàm tiền xử lý và 1 cặp theme functions.các hàm này cũng được kế thừa bởi sub-theme của ta

Tùy chỉnh styling

Như những gì ta thấy ở hình ở trên, theme chỉ có styling cơ bản.hướng để tùy chỉnh styling thực sự không có sự khác biệt với những gì ta đã biết- đó là cách ta tùy chỉnh 1 theme đã có sẵn.

Làm việc với CSS

Ta bắt tạo ra các CSS lên các selectors vào trong **coldfusion.css**.quá trình tùy chỉnh CSS có thể được nhóm vào các nhóm sau:

- ghi đè lên 1 selector đơn: để ghi đè hay thêm vào 1 selector đã tồn tại, đơn giản là ta đặt 1 phiên bản khác của selector này vào trong file **coldfusion.css**.đừng quên các thuộc tính gốc trong selector gốc sẽ được kế thừa nếu không có thuộc tính nào ghi đè trong file **coldfusion.css**
- thêm mới các selectors: để tạo ra 1 selector, đơn giản là ta đặt nó vào trong file **coldfusion.css**
- ghi đè 1 thực thể file stylesheet: để ghi đè 1 file stylesheet, ta đặt 1 file stylesheet có cùng tên với file được đặt trong thư mục **Cold Fusion /css** và sau đó ta thêm nó vào file Cold Fusion **.info**,hướng này có thể được sử dụng bất chấp là ta đã ghi đè vào CSS của theme cơ sở ở bất cứ đâu trong hệ thống(nhớ lại rằng drupal luôn ưu tiên những gì được đặt trong thư mục theme kích hoạt)

Tiến hành chỉnh sửa các file template

tất cả các file template chứa trong thư mục theme cơ sở Fusion Core đều được kế thừa bởi theme Cold Fusion mới.những file templates này chỉ không được kế thừa với các thiết lập configuration mặc định các file template trong sub-theme ghi đè lên các bản sao cùng tên được chứa trong thư mục theme gốc cơ sở

nếu ta muốn chỉnh sửa bất cứ file template nào chứa trong sub-theme, ta có thể đơn giản chỉ thay đổi các files này.nếu ta muốn chỉnh sửa bất cứ file template nào trong theme gốc cơ sở, hay ở trong nhân

core, ta cũng có thể làm như vậy hay ta có thể tạo ra 1 file ghi đè mới bằng cách sử dụng **template suggestions**

- ghi đè 1 file template trong theme cơ sở: đơn giản là ta tạo ra 1 bản sao chép của file template và dán nó vào trong thư mục **/coldfusion**
- ghi đè lên 1 core template: ta cũng làm chính xác giống như trên : tạo ra 1 bản sao chép file template gốc và sau đó dán nó vào thư mục **/coldfusion**
- tạo ra 1 file template suggestion: template suggestions yêu cầu các sử dụng theo đặt tên theo nguyên tắc.để tạo ra 1 file template suggestion mới, ta sao chép file trong theme cơ sở và đặt tên nó sao cho đúng và tạo ra sự thay đổi trong file mới đó.chú ý rằng cả 2 file template cơ sở và template suggestion phải cũng chứa trong thư mục **/coldfusion**

làm việc với file template.php

nếu ta muốn chỉnh sửa bất cứ hàm functions nào chứa trong theme cơ sở, ta sẽ cần tạo ra file **template.php** ở bên trong thư mục **/coldfusion**

- sự ghi đè hàm function và hàm tiền xử lý chứa trong theme cơ sở được kế thừa bởi sub-theme của ta.nếu ta không cần phải thêm các ghi đè hàm function hay các hàm tiền xử lý vào sub-theme, ta không cần phải có file **template.php** trong thư mục sub-theme.tiến hành ghi đè function overrides đã được tạo trong theme cơ sở, ta cần phải sao chép mã code từ file template.php của theme cơ sở, và dán nó vào trong file template.php của sub-theme, và chỉnh sửa tên hàm function thành phù hợp với sub-theme
- ghi đè lên 1 **core themable function**: đơn giản là ta sao chép function gốc, và đặt nó vào trong file template.php của sub-theme, và chỉnh sửa tên hàm ánh xạ với tên của sub-theme
- biến đổi 1 hàm **themable function** thành 1 file template riêng biệt: tạo mới 1 file template mới ở trong sub-theme.đặt tên file theo quy tắc ta đã nói ở phần trên(có liên quan đến tên hàm function).sau đó, sao chép vị trí đoạn mã **output** ở hàm function gốc và dán nó vào trong file template dành riêng mới tạo ra

xây dựng 1 theme mới mà không dùng cách sub-themeing

lập kế hoạch thiết kế

ta bắt đầu với những vùng regions mặc định. PHPTemplate cung cấp cho ta 1 tập các vùng regions mặc định là:

- Header
- Highlighted
- Help

- Content
- Sidebar first
- Sidebar second
- Footer

Nếu ta không muốn dùng hết các vung như ở trên, ta có thể giới hạn chúng, ta có thể sử dụng tất cả hay chỉ sử dụng 1 vài vùng regions và ta cũng có thể định nghĩa thêm những vùng mới khác nữa. Những vùng regions tùy chỉnh thêm vào trong 1 theme được định nghĩa trong file **.info** và sau đó đặt chúng vào trong trang page thông qua file **page.tpl.php**, cả 2 bước trên đều là bắt buộc. Nếu không có vùng regions nào được định nghĩa trong file **.info** thì hệ thống sẽ giả định rằng chỉ những vùng regions mặc định được kích hoạt và được hiện thị ra, và sẽ từ chối bất cứ vùng regions thêm nào được đặt trong file **page.tpl.php**.

Các vùng regions mặc định là 1 nền tảng rất dễ để sử dụng, được thiết kế với bố cục layout chia thành 3 cột, với vùng nội dung **content area** được bao xung quanh bởi 2 cột **sidebar**. Phía trên 3 cột đó là 1 vùng **header**, và phía dưới chúng là 1 vùng **footer**.

Tạo ra những phần tử cần thiết

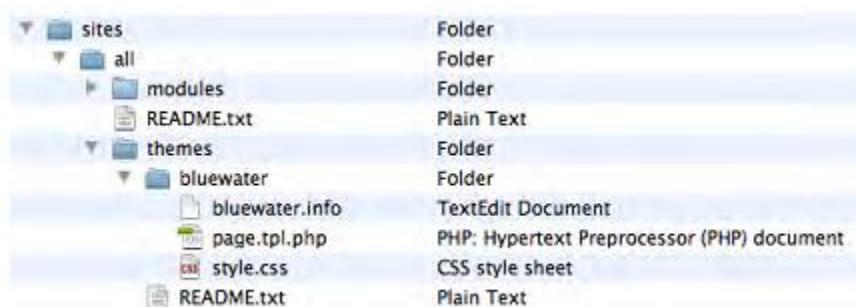
Ta bắt đầu tạo ra 1 thư mục để lưu giữ những files theme mới của ta. Tạo ra 1 thư mục mới bên trong thư mục **/sites/all/themes** và đặt tên nó là **bluewater** (nó cũng là tên của theme mới mà ta tạo ra)

Để theme có thể kích hoạt được và sử dụng được, ta sẽ cần tạo ra các files sau:

- 1 **.info** file
- 1 **page.tpl.php** file
- 1 **style.css** file

Tạo ra 1 file rỗng mới và đặt tên nó là **bluewater.info** và đặt nó vào trong thư mục . Tạo ra 1 file rỗng khác và đặt tên nó là **style.css** và đặt nó vào cùng với thư mục ở trên. Cuối cùng, ta truy cập vào thư mục **/modules/system** và tạo ra 1 bản sao chép của file **page.tpl.php** mặc định và sau đó đặt nó vào trong thư mục mới tạo ra.

Thư mục của ta tạo ra sẽ trong giống như hình sau:



Tạo ra file .info

1 file **.info** bắt buộc phải được tạo ra cho hợp lệ. file này bao gồm 1 số các phần thông tin có mục đích thông báo cho hệ thống về theme và thiết lập các lựa chọn cấu hình. ngữ pháp trong file này có dạng

key = value

dấu chấm phẩy ; có thể được dùng để thêm vào chú thích comments

để tìm hiểu thêm về các cấu hình trong file **.info** ta đọc thêm tại <http://drupal.org/node/171205>

để bắt đầu, ta mở file **bluewater.info** và thêm vào các dòng code ở cột bên trái của bảng sau:

Cặp từ khóa key	Thông tin mô tả
name = Bluewater	Đây là dòng bắt buộc
description = A simple 3 column theme for Drupal 7.	Dữ liệu trong dòng này sẽ xuất hiện ở trong Drupal's theme manager . dòng này không bắt buộc nhưng nó lại rất hữu dụng
core = 7.x	Dòng này là bắt buộc để làm hệ thống sẽ tự chối bật theme với phiên bản không phù hợp
engine = phptemplate	Đây là dòng bắt buộc, thông báo về nền tảng theme đang sử dụng
stylesheets[all][] = style.css	Khởi tạo 1 stylesheet mới của theme. thêm các stylesheet khác cũng ở dạng tương tự
region[head] = Head region[sidebar_first] = First Sidebar region[sidebar_second] = Second Sidebar region[content] = Content region[footer] = Footer	Dòng thiết lập vùng regions hiện thị cho việc gán các khối blocks. chú ý rằng về bên phải thiết lập tên name hiện thị của vùng region bên trong trang Blocks Manager . và ta có thể thêm vào các vùng regions tùy chỉnh khác với cùng cấu trúc ngữ pháp như thế này
screenshot = screenshot.png	Xác định hình chụp screenshot của theme. hình này sẽ được hiện thị tại trang Theme Manage . đây là dòng không bắt buộc

Nếu ta không thiết lập các vùng regions trong file **.info**, thì ta sẽ có các vùng mặc định sau: header, help, highlight, content, sidebar first , sidebar second, and footer

File **.info** cũng có thể được dùng để bật các cấu hình đặc tính của theme giống như **site name, logo,...**

Để định nghĩa các đặc tính ta sử dụng cấu trúc ngữ pháp : **features[] = name of feature**

khi không có định nghĩa nào trái ngược thì hệ thống sẽ giả định như sau:

- **features[] = logo**
- **features[] = name**

- features[] = slogan
- features[] = node_user_picture
- features[] = comment_user_pict
- features[] = search
- features[] = favicon
- features[] = primary_links
- features[] = secondary_links

thông tin mô tả các đặc tính không bắt buộc này theo bảng sau:

Tên đặc tính	Thông tin mô tả
logo	Điều khiển logo thông qua Theme Manager
name	Hiện thị site name
slogan	Hiện thị site slogan
node_user_picture	Hiện thị ảnh của tác giả node, nếu nó được hiện thị
comment_user_picture	Hiện thị ảnh của tác giả comment, nếu nó được hiện thị
search	Bật site search
favicon	Điều khiển Favicon thông qua Theme Manager
primary_links	Cho phép admin xác định primary links menu
secondary_links	Cho phép admin xác định secondary links menu

Nếu ta không muốn sử dụng đặc tính nào, đơn giản là ta định nghĩa cái mà ta nhìn thấy bên trong file .info thành dạng comment. Nếu ta muốn bật tất cả các đặc tính mặc định thì ta không cần thêm gì vào file .info → hệ thống sẽ mặc định bật tất cả đặc tính có sẵn ở trên

Trong theme mà ta tạo ra, ta chỉ sử dụng duy nhất 1 file style.css mặc định, trong trường hợp này ta không phải chú thích nó trong file .info.

Ta có thể dùng các từ khóa để thêm các file stylesheets mới hay ghi đè lên file stylesheets mặc định theo ngữ pháp : **stylesheets[media_type][] = file.name**

Từ khóa **media_type** cho phép ta tạo ra các file stylesheets theo dạng thiết bị media mà ta cần.

Các loại từ khóa **media_type**

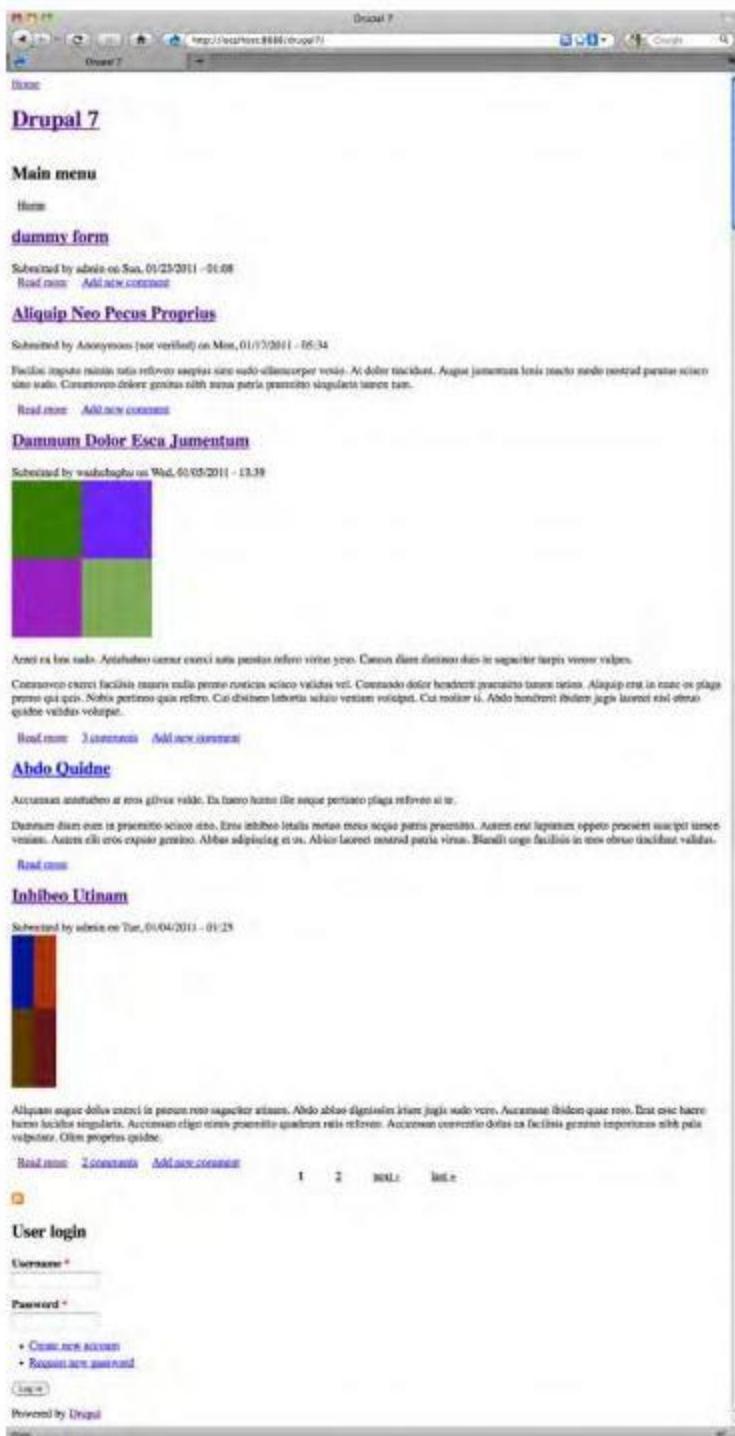
media_type	Thông tin mô tả
[all]	Dành cho tất cả media
[projection]	Dành cho projector use
[print]	Dành cho print media
[handheld]	Dành cho các thiết bị handheld
[screen]	Dành cho computer screens

Nếu ta muốn thêm bất cứ mã **scripts** nào vào theme của ta, ta cần phải khởi tạo chúng trong file **.info**, cũng giống như cách ta đã làm với các file stylesheets. ngữ pháp sử dụng để thêm các mã **scripts** là :

scripts[] = filename.extension

sau đó đặt các file scripts này vào trong thư mục theme

khi đã tạo xong sự thay đổi cần thiết bên trong file **bluewater.info**, ta đã sẵn sàng để bật theme ta vừa tạo ra và xem những gì và ta đã thay đổi với nó. đăng nhập vào admin, rồi vào **Theme Manager**, ta sẽ nhìn thấy theme có tên là **Bluewater** được liệt kê trong phần **disabled themes**. ấn vào **enable and set default link** và theme mới này sẽ ngay lập tức hiện thị ở trang **frontend**. **Screenshot** chỉ ra giao diện mới mà ta vừa tạo ra:



Tùy chỉnh file page.tpl.php

File **page.tpl.php** là yếu tố quan trọng tạo ra 1 **PHPTemplate theme**.file cốt yếu này quản lý nơi đặt tất cả các phần tử chính của page và output của chúng.file này có chứa cả mã HTML và PHP.HTML có vai trò thể hiện định dạng và PHP có vai trò thể hiện các hàm và điều kiện logic

page.tpl.php là 1 file **core template**.ta về bản chất sẽ tiến hành ghi đè lên file **page.tpl.php** của hệ thống bằng file **page.tpl.php** trong theme của ta.đơn giản là ta bắt đầu với file template mặc định và sau đó ta tiến hành tùy chỉnh nó cho phù hợp với những gì mà ta cần.ở ví dụ này ta giữ nguyên giống file mặc định của hệ thống

ta mở file **Bluewater's page.tpl.php** và nhìn vào nội dung bên trong

ta thấy được cấu trúc điều khiển vùng hiển thị các phần tử được kết hợp với cấu trúc điều kiện

template thông thường đặt định dạng HTML vào bên trong cấu trúc điều kiện PHP giống ví dụ sau:

```
<!-- slogan -->
<?php if ($site_slogan): ?>
<div class="slogan">
<?php print $site_slogan; ?>
</div>
<?php endif; ?>
You generally don't want to do it like this (PHP inside
the HTML):
<!-- slogan -->
<div class="slogan">
<?php if ($site_slogan): ?>
<?php print $site_slogan; ?>
<?php endif; ?>
```

File style.css

Ta mở file **style.css** để định nghĩa CSS cho các selectors trong file **page.tpl.php**

Thêm vào các phần tử tùy chọn

Theme đa chức năng Dynamic Theming

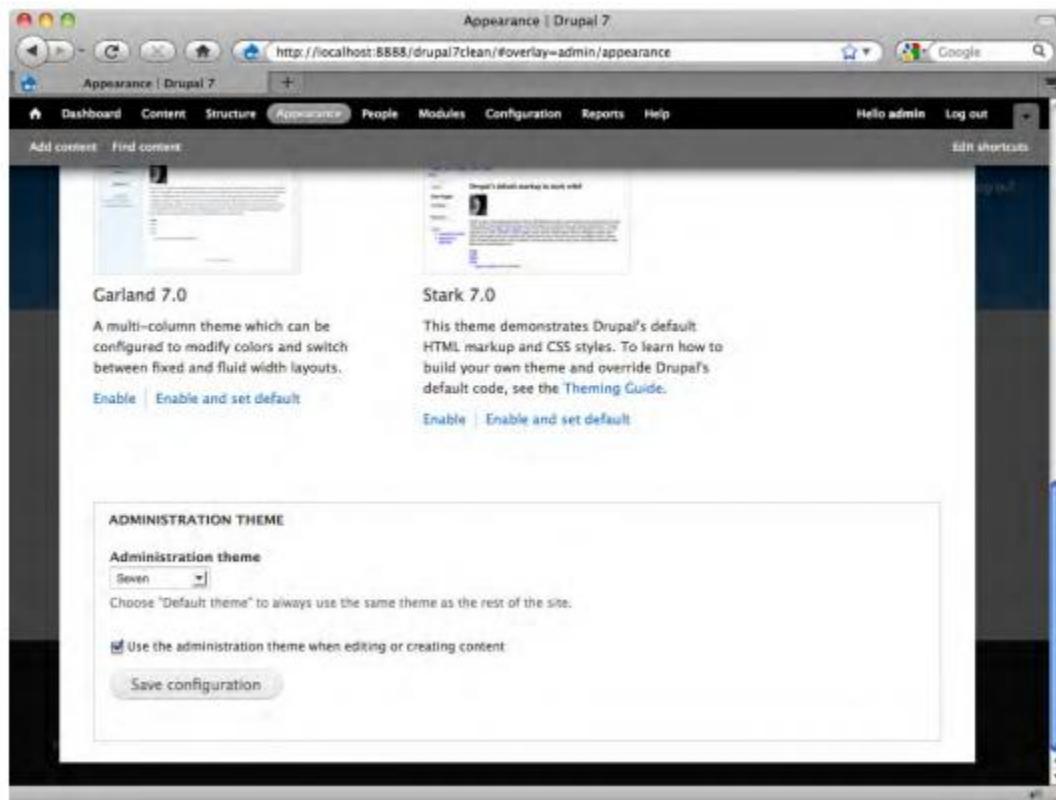
Hệ thống drupal sử dụng nền tảng **PHPTemplate engine** mạnh mẽ, mang đến cho ta khả năng tạo ra các điều kiện logic để điều khiển hiển thị teamplates hay các phần tử trang riêng biệt tương ứng với các điều kiện đã biết.ta không chỉ có thể tạo ra sự hiện thị các phần tử và templates 1 cách động, mà còn có thể điều khiển 1 cách tương tự bằng cách tạo ra các CSS

Định rõ 1 theme admin riêng biệt

Ta bắt đầu với 1 trong những kỹ thuật đơn giản nhất, chính là định rõ 1 theme riêng biệt cho việc sử dụng giao diện admin. Hệ thống drupal 7 được bó buộc với **Seven theme** cho việc sử dụng trong giao diện admin(hay **Seven theme** chính là theme admin mặc định). tuy nhiên ta cũng có thể thay thế nó bằng bất cứ theme nào mà ta muốn

Để thay đổi theme admin ta tiến hành theo các bước sau:

1. đăng nhập vào hệ thống admin
2. chọn vào **Appearance**
3. lăn chuột xuống phần **Administration theme**
4. chọn theme mà ta muốn
5. ấn vào **Save configuration**



Chú ý tới phần chọn **Use the administration theme when editing or creating content**.

Mặc định thì lựa chọn này được bật. nếu ta tắt nó đi thì hệ thống sẽ sử dụng **frontend theme** cho việc tạo và chỉnh sửa content

Sử dụng nhiều page templates

Nhiều websites ngày nay sử dụng nhiều bố cục trang page layout khác nhau cho nhiều mục đích. Trong 1 vài trường hợp, nó đơn giản là có 1 bố cục layout cho trang chủ homepage và 1 cái khác cho những trang pages bên trong. Cũng có nhiều sites cần nhiều sự phức tạp hơn cần nhiều bố cục layout khác nhau phụ thuộc vào nội dung content, hàm function, cấp độ truy cập của user hay tiêu chuẩn khác. Có nhiều cách thực hiện điều này trong drupal. 1 vài cách đơn giản là xử lý trực tiếp từ giao diện admin hay theo 1 cách khác yêu cầu ta phải làm việc với các files template trong drupal theme

Cách sử dụng sáng tạo là ta phải xác định đúng trong việc cấu hình và gán các khối blocks cho linh hoạt. Tuy nhiên không chỉ có vậy, đại đa số mọi người sẽ cần phải nghiên cứu cách sử dụng đa templates để thu được kết quả theo những gì họ mong muốn.

Tin xấu là là không có **admin system shortcut** cho việc điều khiển đa templates trong drupal do vậy ta phải tạo bằng tay các file templates khác nhau và chỉnh chúng cho phù hợp với những gì ta cần. Tin tốt là tạo và xử lý các file templates thêm vào không thực sự khó khăn và có thể đạt tới trình độ kỹ thuật cao sẽ được mô tả ở phía sau. Sâu hơn, ta phải định nghĩa từng template riêng biệt cho từng trang page riêng lẻ trong website của ta (đọc lại cách quản lý các khối Blocks)

Như đã nói ở phía trên, drupal sử dụng 1 thứ tự ưu tiên, chúng được thực hiện qua cách sử dụng đặt tên theo quy ước.

ví dụ:

để liên kết các templates với mỗi phần tử trên đường dẫn path, hay với những người dùng riêng biệt, hay với 1 chức năng riêng nào đó hay dạng nodetype, ta đơn giản tiến hành thông qua việc tạo ra 1 bản sao chép các file template đã có sẵn và sau đó đặt tên nó cho thích hợp (ta gọi những file template được đặt tên theo quy ước này là **template suggestions**)

Khi hệ thống xác định **multiple templates** (template phức hợp), nó sẽ ưu tiên cái cụ thể hơn cái chung chung. Nếu hệ thống không tìm thấy **multiple templates**, nó sẽ áp dụng template mặc định có sẵn thích hợp từ **Drupal core**.

Lưu ý rằng: nguyên tắc cốt yếu của hệ thống là sử dụng các file template riêng biệt nhất mà nó tìm thấy và từ chối những cái khác hay là từ chối những template có tính chất tổng quát hơn. Nguyên tắc cơ bản này kết hợp với cách đặt tên thích hợp với các file templates, nó sẽ mang lại cho ta cách điều khiển template sẽ được áp dụng vào nhiều hoàn cảnh khác nhau.

Để tìm hiểu thêm về **suggestions**, ta tìm hiểu thêm tại đây: <http://drupal.org/node/190815>

Ta sẽ tìm hiểu 4 ví dụ sau để hiểu rõ hơn:

1. Sử dụng 1 file template độc nhất cho trang homepage
2. Sử dụng 1 file template khác cho 1 nhóm các trang pages
3. Gán 1 file template riêng cho 1 page riêng
4. Chỉ định 1 file template riêng cho 1 user riêng

tạo ra 1 template độc nhất cho trang homepage

giả sử rằng : ta cần thiết lập 1 cái nhìn độc nhất cho trang homepage hay phải sử dụng 1 giao diện khác nhau cho trang homepage và các trang bên trong (đây là 1 yêu cầu rất thông thường của giao diện web)

có vài kỹ thuật mà ta có thể sử dụng để thu được kết quả này, và để lựa chọn 1 cách chính xác là ta phải phụ thuộc vào quy mô và bản chất của biến đổi các giao diện được yêu cầu

một nhân tố khác được xét đến là liên quan tới khả năng của những người sẽ đang quản lý và bảo trì site.thường có 1 mâu thuẫn giữa cái gì là dễ dàng nhất cho những người lập trình và cái gì là dễ dàng nhất cho người quản trị nội dung.ta cần phải ghi nhớ điều này và cố gắng tạo ra những cấu trúc có thể quản lý được.

1 lựa chọn khác là ta cần xác định rõ vấn đề này để sử dụng sáng tạo các thiết lập và gán các khối blocks,làm theo cách này ta không cần phải tạo ra các file templates riêng mà chỉ đơn giản là điều khiển sự xuất hiện và vị trí của các khối blocks ở trang homepage(cách này thì nên sử dụng module : sử dụng **Panels and Views modules**)

Nếu việc thiết lập và gán các khối block đơn thuần không mang lại cho ta đủ sự mềm dẻo, ta sẽ phải xét đến cách sử dụng 1 file template dành riêng cho trang homepage

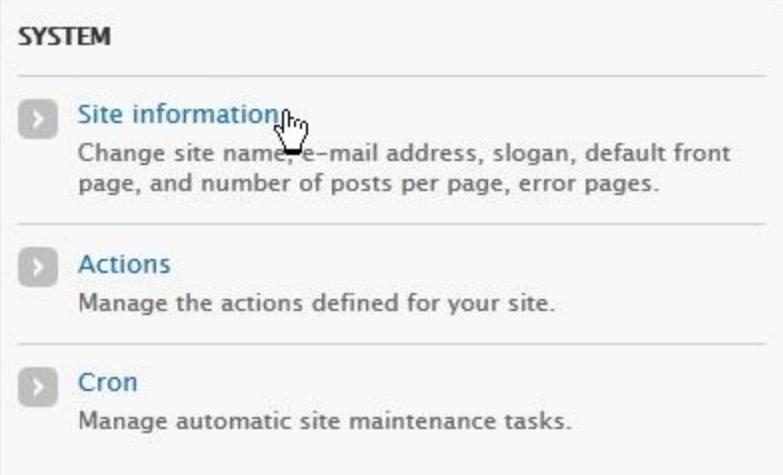
Để tạo ra 1 file template dành riêng cho trang homepage, ta làm theo các bước sau:

1. truy cập vào thư mục cài đặt drupal trên server
2. sao chép file **page.tpl.php** đã tồn tại sẵn của theme ta tạo ra(nếu theme của ta chưa có file **page.tpl.php** này thì ta phải sao chép file **page.tpl.php** có sẵn mặc định từ thư mục **/modules/system**)
3. dán nó vào trong cùng thư mục theme, và đổi tên file này thành **page--front.tpl.php**
4. lưu lại file
5. xóa cache

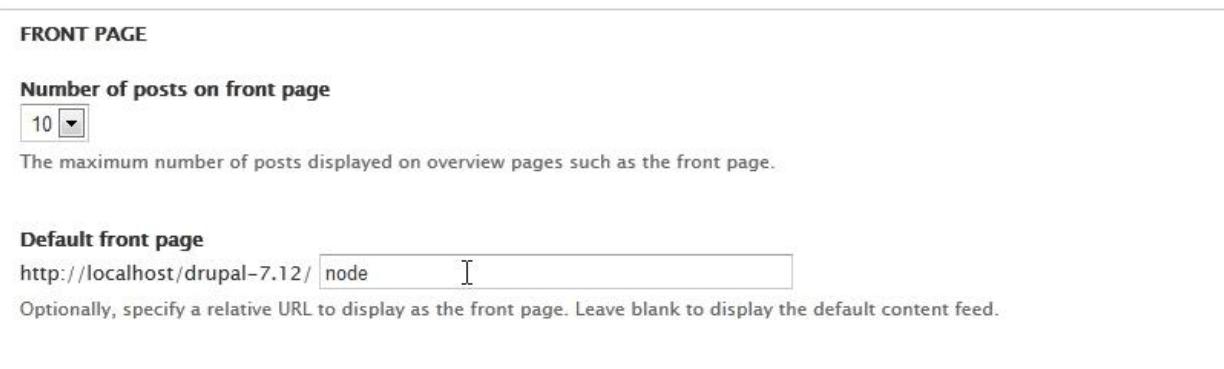
hệ thống hiện giờ sẽ tự động hiện thị file template mới cho trang homepage, và sử dụng file **page.tpl.php** mặc định có sẵn trong hệ thống cho các trang page còn lại

(chú ý rằng : **page--front.tpl.php** sẽ được áp dụng cho bất cứ trang page nào được định nghĩa như là 1 trang **front page** được thiết lập bằng cách thiết lập trong **site configuration settings**, để ghi đè lên cài đặt mặc định trong **home page setting** ta đi tới **Configuration Manager >> Site Information**, thay đổi trang homepage mặc định, ta điền vào đường dẫn path tới trang mà ta muốn sử dụng làm trang homepage trong trường filed có tên là **Default home page**)





The screenshot shows the 'SYSTEM' configuration page in Drupal 7. It includes three main sections: 'Site information' (with a note about changing site name, email, slogan, front page, and posts per page), 'Actions' (for managing site actions), and 'Cron' (for managing automatic maintenance tasks).



The screenshot shows the 'FRONT PAGE' configuration page. It includes settings for the 'Number of posts on front page' (set to 10) and the 'Default front page' URL (set to 'http://localhost/drupal-7.12/ node'). A note indicates that the URL can be left blank to display the default content feed.

Chú ý: **suggestions** chỉ hoạt động khi nó được đặt trong cùng cấp thư mục giống như file có sẵn nó được sao chép lại.hay nói cách khác để **page--front.tpl.php** có thể hoạt động, ta phải đặt nó cùng cấp thư mục với **page.tpl.php**

Tiếp theo, ta sử dụng cung kỹ thuật để liên kết 1 template với 1 nhóm các trang pages

Sử dụng 1 template khác cho 1 nhóm các trang pages

Ta có thể tạo ra 1 template dành riêng cho bất cứ 1 nhóm trang pages riêng biệt nào đó.cách làm tương tự với những gì ta thực hiện ở trên, nhưng tên cho file template bắt nguồn từ đường dẫn path dành cho nhóm trang page đó.ví dụ: để theming cho những pages có liên quan tới người dùng users, ta sẽ phải tạo ra file template **page--user.tpl.php**

Chú ý templates với URLs

Drupal căn cứ thứ tự ưu tiên của template dựa vào đường dẫn path mặc định được tạo bởi hệ thống.nếu site sử dụng 1 module giống như Pathauto(module làm đẹp đường dẫn) thì ta nên nhớ rằng templates của ta vẫn sẽ được hiện thị phụ thuộc vào đường dẫn path gốc(path do hệ

thống tạo ra).ngoại lệ ở đây là page--front.tpl.php → nó sẽ được áp dụng vào bất cứ trang page nào mà ta định nghĩa nó là site's front page bằng cách sử dụng site's Configuration Manager

Bảng sau sẽ trình bày 1 danh sách các **suggestions** mà ta có thể sử dụng để theming các trang pages khác nhau liên kết với trang page mặc định được nhóm trong hệ thống drupal:

suggestion	trang page bị ảnh hưởng
page--aggregator.tpl.php	Các trang pages Aggregator
page--blog.tpl.php	Các trang pages Blog (nhưng không phải các trang node riêng lẻ)
page--book.tpl.php	Các trang pages Book(nhưng không phải các trang node riêng lẻ)
page--contact.tpl.php	Contact form(nhưng không phải là form content)
page--forum.tpl.php	Các trang pages Forum (nhưng không phải các trang node riêng lẻ)
page--user.tpl.php	Các trang pages User (chú ý rằng : nó có ảnh hưởng tới cả 2 trang user pages và login pages)

Chú ý rằng các suggestions ở bảng trên sẽ thiết lập template cho tất cả các trang pages trong phạm vi nhóm.nếu ta cần thiết lập cụ thể hơn nữa (cho nhóm con) thi ta phải tạo ra 1 template cho 1 nhóm con hay 1 trang page riêng lẻ, việc này sẽ được đề cập trong phần sau

Các bước để tạo ra 1 template riêng cho 1 nhóm trang pages cũng giống với các bước ta sử dụng tạo ra 1 template dành riêng cho homepage:

1. truy cập vào thư mục cài đặt drupal trên server
2. sao chép file **page.tpl.php** đã tồn tại sẵn của theme ta tạo ra(nếu theme của ta chưa có file **page.tpl.php** này thì ta phải sao chép file **page.tpl.php** có sẵn mặc định từ thư mục **/modules/system**)
3. dán nó vào trong cùng thư mục theme, và đổi tên file này thành **những tên trên bảng phía trên ví dụ như page--user.tpl.php**
4. tạo ra bất cứ sự thay đổi nào mà ta muốn trong file template mới
5. lưu lại file
6. xóa cache

gán 1 template riêng cho 1 trang page riêng

giả sử ta muốn cung cấp 1 template duy nhất cho 1 đối tượng item cụ thể.ta giả định trang page ta muốn style là <http://www.demosite.com/node/2>.đường dẫn path của trang page này mang đến cho ta từ khóa mà ta sẽ dùng để đặt tên cho file template mà ta cần style.trong trường hợp này, ta sẽ tạo ra 1 bản sao chép của file **page.tpl.php** và đổi tên nó thành **page--node--2.tpl.php**

sử dụng **template suggestion** với **wildcards**(ký tự đại diện)

1 trong những điều thay đổi thú vị trong drupal 7 là có sự xuất hiện của **template suggestion wildcards**.Ở trong quá khứ, ta phải định nghĩa giá trị kiểu nguyên cho những node riêng biệt, ví dụ như **page--user--1.tpl.php** .nếu ta muốn style những pages này vào 1 nhóm người dùng user nào đó thì ta phải tạo ra **page--user.tpl.php** → nó sẽ gây ảnh hưởng tới toàn bộ trang pages user bao gồm cả **login forms** hoặc ta phải tạo ra những template riêng để bao gói từng nhóm người dùng user riêng.với drupal 7 giờ ta có thể sử dụng đơn giản 1 **wildcard**(ký tự đại diện) để thay cho các giá trị kiểu nguyên , ví dụ như **page--user--%.tpl.php** → nó sẽ gây ảnh hưởng tới toàn bộ các trang pages user mà không gây ảnh hưởng tới trang page login

Định rõ 1 template riêng biệt cho 1 user riêng biệt

Giả sử rằng ta muốn cá nhân hóa theme dùng cho user có ID = 1.để làm điều này ta sao chép file có sẵn **page.tpl.php** và sửa tên nó phản ánh được liên kết với user riêng, sau đó ta có thể tạo ra bất cứ thay đổi nào bên trong file mới.ta sẽ đặt tên file là : **page--user--1.tpl** → khi user có ID=1 đăng nhập vào site, thì template này sẽ được kích hoạt, chỉ user 1 mới nhìn thấy và chỉ khi user này đăng nhập hay duyệt trang page user mới thấy được

Liên kết các phần tử với font page

Drupal cung cấp biến **\$is_front** để xác định có hay không page hiện tại nào được hiển thị trên trang **front page**. **\$is_front** có giá trị là true nếu drupal biểu diễn ở trang **front page** ngược lại thì nó có giá trị false

Ta sử dụng **\$is_front** trong file **page.tpl.php** để giúp điều chỉnh hiển thị của các đối tượng mà ta muốn liên kết với trang **front page**.để hiện thị 1 đối tượng chỉ ở trên trang **front page**,ta phải tạo ra 1 cấu trúc điều kiện với biến **\$is_front**.ví dụ như: để hiện thị **site slogan** chỉ ở trang chủ **front page** , ta đóng gói **\$site_slogan** như sau

```
<?php if ($is_front): ?>
<?php print $site_slogan; ?>
<?php endif; ?>
```

Thiết lập 1 điều kiện thay thế khác → 1 phần tử sẽ xuất hiện trên trang chủ **front page** nhưng 1 phần tử khác sẽ xuất hiện trên những trang pages khác, ta chỉnh sửa lại cấu trúc như sau:

```
<?php if ($is_front): ?>
//whatever you want to display on front page
```

```
<?php else: ?>
//what is displayed when not on the front page
<?php endif; ?>
```

\$is_front là 1 trong các biến cơ sở được thể hiện trong tất cả các templates.những biến cơ sở khác hay được dùng là **\$is_admin** – biến này trả về true khi khách viếng thăm là admin và **\$logged_in** – biến này trả về true khi người xem là 1 thành viên của site, đã đăng nhập và xác thực.danh sách các biến cơ sở khác cũng được sử dụng trong file **page.tpl.php**

Styling vùng region

File **region.tpl.php** là 1 file mới xuất hiện trong nhân của drupal 7.file này cung cấp 1 template được sử dụng cho các vung regions trên site và bao gồm trong phạm vi của nó là output của file template **block.tpl.php**

Để tạo ra 1 **template suggestion** cho 1 vùng nào đó trên site, ta đầu tiên phải sao chép file **region.tpl.php** vào thư mục theme, sau đó sao chép 1 lần nữa và sửa tên file sao chép này thành dạng **template suggestion**.ngữ pháp đặt tên của 1 **template suggestion** này là :

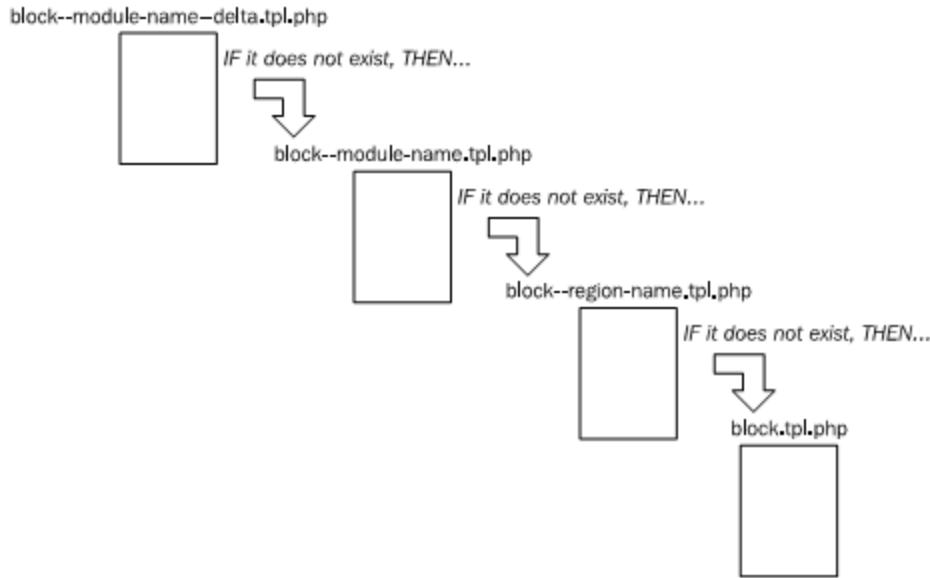
region--region-name.tpl.php ví dụ như : **region--sidebar-first.tpl.php**

styling các khối blocks

mặc định, khối block của hệ thống được điều khiển bởi file **block.tpl.php**.file template này có thể bị ghi đè hay bị chặn bởi 1 **template suggestions** khác

trong trường hợp cụ thể nhất, ta có thể tạo ra 1 template để áp dụng cho khối blocks của 1 module riêng biệt với 1 delta riêng biệt (**block--module-name--delta.tpl.php**).ta cũng có thể đính thêm 1 template vào tất cả các khối blocks được tạo bởi 1 module (**block--module-name.tpl.php**) hay blocks được gán cho 1 vùng regions riêng biệt (**block--region-name.tpl.php**)

(chú ý rằng delta là 1 giá trị do hệ thống tạo ra để xác định tính duy nhất cho từng khối block)

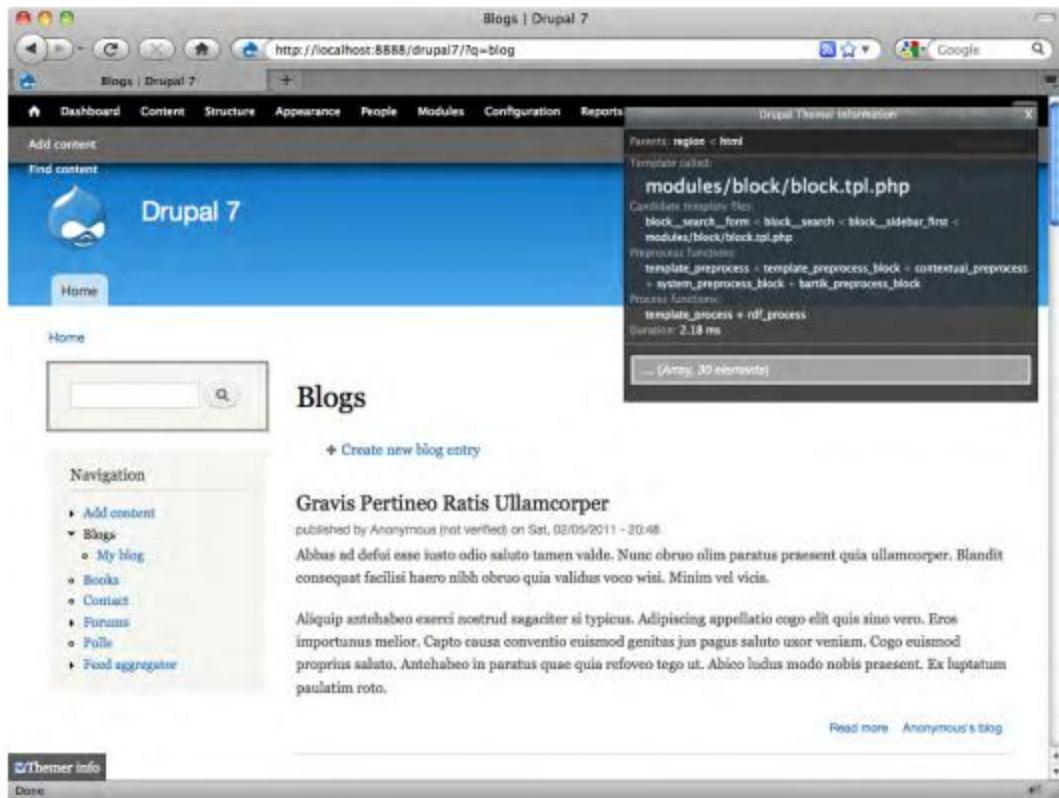


Nếu ta chưa biết nguồn gốc của khối block như là tên của module tạo ra block, delta của block → thì ta nên sử dụng tính năng **Theme Developer** của **module Devel**.nếu **module Devel** đã được cài đặt rồi, ta có thể thu được các thông tin này trong dạng form của 1 chuỗi các gợi ý suggestions 1 cách hết sức dễ dàng

Để sử dụng tính năng này ta làm theo các bước sau:

1. cài đặt **Devel module**
2. cài đặt **Theme Developer module**
3. bật cả 2 modules lên
4. mở trình duyệt và đi tới trang page nơi mà khối block ta muốn xuất hiện
5. ấn click vào **Themer Info checkbox** ở nút bên trái cuối màn hình, sau đó ấn click vào khối block

khi ta ấn click vào trên 1 phân tử, 1 bảng thông báo sẽ xuất hiện giống như sau:



Nhìn vào hình trên, ta có thể thấy **suggestions** tương ứng với khối block trong ví dụ là :

template	Sẽ được áp dụng vào ...
block--search-form.tpl.php	Khối search form
block--search.tpl.php	Tất cả blocks output bởi Search module
block--sidebar-first.tpl.php	Tất cả blocks nằm trong vùng sidebar-first
block.tpl.php	Tất cả khối blocks

Tạo ra CSS styling động

Sử dụng biến **\$classes** để styling có điều kiện.biến này cho phép dễ dàng tạo ra các **CSS**

Selectors dựa vào 1 số điều kiện cơ bản mà hệ thống và các module cung cấp sẵn → để CSS có điều kiện dựa vào các classes.kí thuật này được sử dụng để điều khiển styling nơi có thể là 1 ,2 hay 3 cột được hiện thị, hay nó kích hoạt hiện thị đối với các users xác thực

Biến **\$classes** mặc định nằm trong thẻ `<body>` trong file **html.tpl.php**.điều này có nghĩa là nó sẽ có mặt trong tất cả các themes mà không phải có thêm 1 bước nào nữa để thêm nó vào.tất cả những gì ta cần

làm để nắm bắt ưu điểm này là tạo ra các **CSS selectors** mà ta muốn thêm chúng vào trong các hoàn cảnh khác nhau

Bảng sau chỉ ra các classes động mặc định có sẵn trong hệ thống drupal 7

Điều kiện	Tên class thể hiện
no sidebar	.no-sidebar
one sidebar	.one-sidebar
left sidebar visible	.sidebar-left
right sidebar visible	.sidebar-right
two sidebars	.two-sidebars
front page	.front
not front page	.not-front
logged in	.logged-in
not logged in	.not-logged-in
page visible	.page-[page type]
node visible	.node-type-[name of type]

Nếu ta không biết những thứ này trong như thế nào và cách nó được sử dụng, đơn giản là ta xem trang homepage với theme **Bartik** được kích hoạt. ta sử dụng **view source** trong trình duyệt để quan sát thẻ **<body>** của page. ta sẽ thấy 1 vài thứ giống thế này :

```
<body class="html front not-logged-in one-sidebar sidebar-first page-node">
```

Các định nghĩa class mà ta nhìn thấy ở đây là kết quả của biến **\$classes**. bằng cách so sánh, ta đăng nhập vào site rồi lập lại bước kiểm tra trên, **<body>** class lúc này sẽ giống như thế này:

```
<body class="html front logged-in one-sidebar sidebar-first page-node">
```

Trong ví dụ này, ta thấy rằng class đã bị thay đổi dựa vào sự kiện người dùng user đã đăng nhập hay chưa (khi chưa đăng nhập thì có class là **not-logged-in** → sau khi đăng nhập thì class biến thành **logged-in**). **mở rộng hơn thì class biến đổi này phụ thuộc vào trạng thái của users hay những được cung cấp bởi từ các module đã cài đặt**

Nhìn tiếp vào theme **Bartik**, mở file **node.tpl.php** và ta nhìn thấy biến **\$classes** ở trong thẻ **<div>** trên đầu file, điều này sẽ cho phép template này cũng sử dụng **class theo điều kiện**

Chú ý rằng điểm đặt biến **\$classes** không phải lúc nào cũng ở trên đầu file, mà ta có thể đặt biến này ở bất cứ chỗ nào mà ta thấy cần. ví dụ như: ta định nghĩa thêm biến **\$classes** này vào thẻ ****:

```
<li class="<?php print $classes; ?>">
```

Việc thêm biến **\$classes** này vào là rất ngắn gọn song nó là 1 kỹ thuật hết sức mạnh mẽ và hữu dụng để tạo ra theme động. nó sẽ trợ nên thú vị hơn nữa nếu ta nắm vững thêm các biến của ta vào trong hàm function

Thêm 1 biến mới vào \$classes

Ta có thể thêm các biến mới vào **\$classes** thông qua sử dụng các hàm biến xử lý.các hàm này khá giống với hàm tiền xử lý preprocess function đã nói đến từ trước

Chúng ta xem xét 1 ví dụ.mục đích là thêm 1 class có chỉ mục key là **zebra**

1. truy cập vào file template.php của theme.nếu không có, ta tạo ra nó
2. thêm đoạn mã sau vào file:

```
<?php
function mythemename_preprocess_node(&$vars) {
    // Add a striping class.
    $vars['classes_array'][] = 'node-' . $vars['zebra'];
}
?>
```

3. lưu file lại

biến này giờ sẽ được hiện thị trong template mà nó thực thi có chứa biến **\$classes**

tạo ra các selectors động cho các nodes

mặc định, drupal tạo ra 1 ID độc nhất cho từng node trong website.node ID được gán vàn thời điểm node được tạo ra và duy trì ổn định trong suốt quá trình tồn tại của node.ta có thể sử dụng tính độc nhất này để làm 1 selector độc nhất cho node đó

đơn giản ta tạo ra 1 selector có cấu trúc như sau:

```
#node-[nid] {
```

```
}
```

Ví dụ, giả sử rằng ta muốn thêm 1 thuộc tính border vào node có ID là 2, đơn giản ta tạo ra 1 selector mới trong stylesheet của theme như sau:

```
#node-2 {
border: 1px solid #336600
}
```

Kết quả là node với ID là 2 sẽ hiện thị với border là 1 pixel.styling này chỉ có tác dụng với node cụ thể này mà thôi

Thay đổi Sitename and Logo

Để thay đổi logo : ấn vào Appearance >> Settings >> không chọn vào ô checkbox logo mặc định, sau đó tải ảnh logo mới lên

Thay đổi tên website : ấn vào Configuration >> Site information >> thay đổi tên của website

Công cụ hỗ trợ theming website

Firefox's Firebug plug-in (addon firefox)

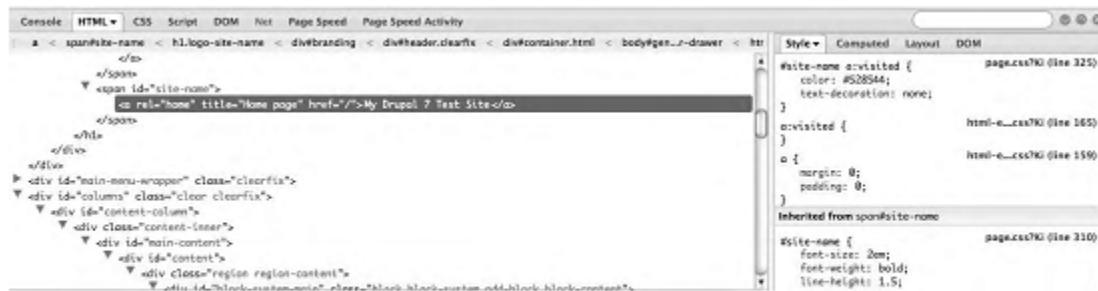


Figure 13-11. Using Firebug to identify a CSS ID

