Drupal 7

I. Template and theming

1. Tạo theme mới

Tạo thư mục gốc vào thư mục sites/all/theme/

+ Tên thư mục : toanlm

+ Trong thư mục tạo 2 file

- toanlm.info: Khai báo các thành phần như name, version, core, css, javascript,...

- screenshot.png là thumbnail của theme (nhìn thấy trong backend)

Nội dung file info như sau:

name = toanlm

description = Create new template tranning drupal 7 - by toanlm

version = 0.1

core = 7.x

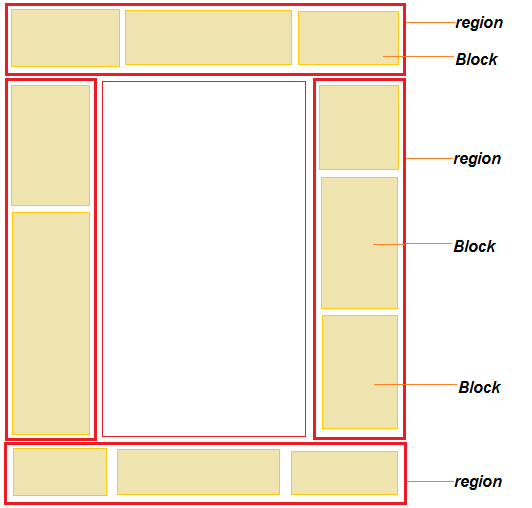
engine = phptemplate

screenshot = screenshot.png

=> Vào trong backend quản lý thêm và thấy theme toanlm vừa mới tạo.

REGION

Khai báo các region cho website: Region là vùng để hiện các block của website



Khai báo các region và khai báo các file css, js cho template

regions[header] = Header

regions[content]= Content

regions[sidebar\_left] = Left sidebar

regions[sidebar\_right] = Right sidebar

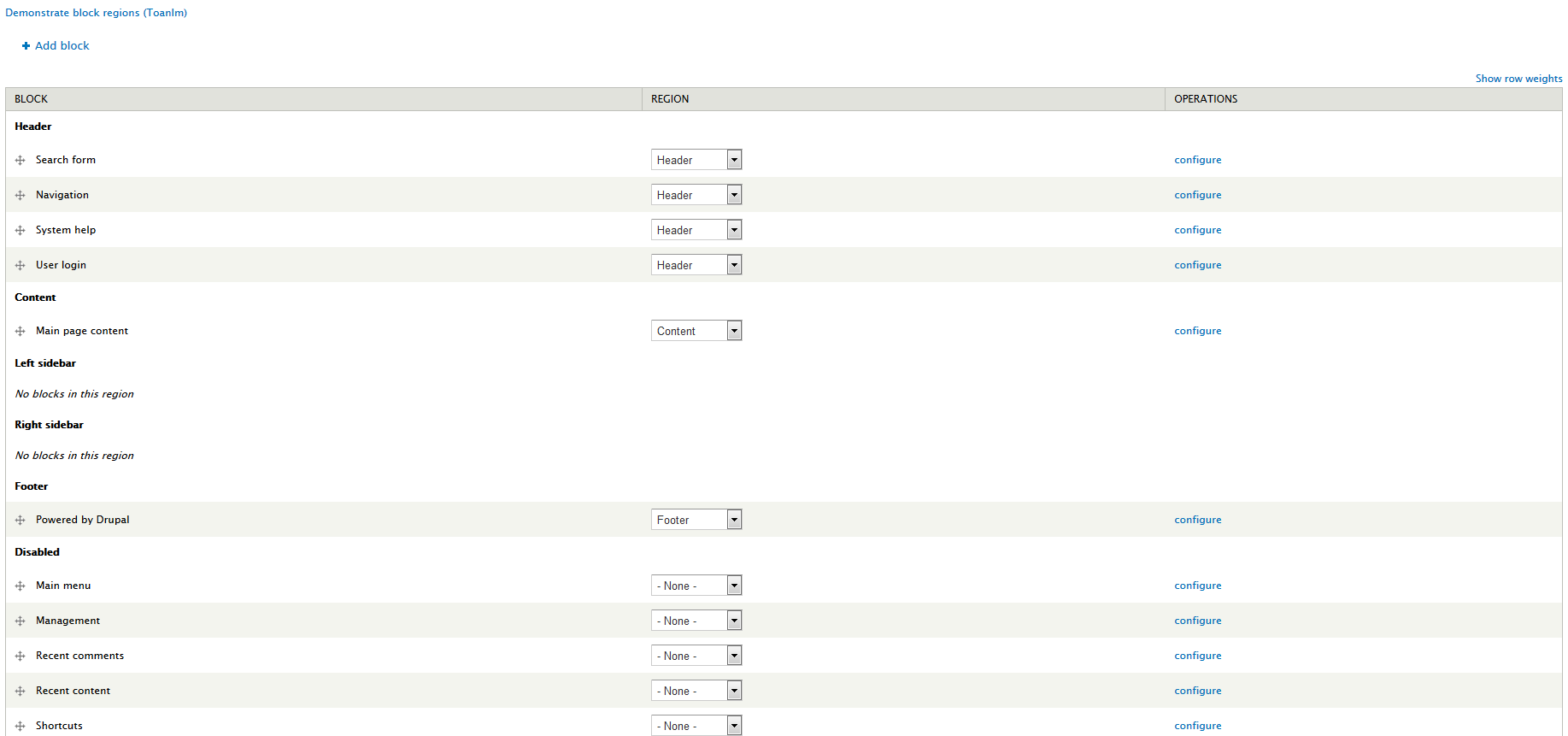
regions[footer] = Footer

stylesheets[all][] = style.css

stylesheets[print][] = print.css

scripts[] = main.js

Lúc đó vào backend/structure sẽ có các phần quản lý, có thể add các block vào các region mình đã khai báo.



Tạo region, sau đó vào backend chọn các block đặt vào các region tương ứng. Còn việc thêm block thì thực hiện bình thường trong backend, hệ thống drupal đã có sẵn.

Thêm 1 ví dụ khác về regions cho wesbite có cấu trúc phức tạp hơn.

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

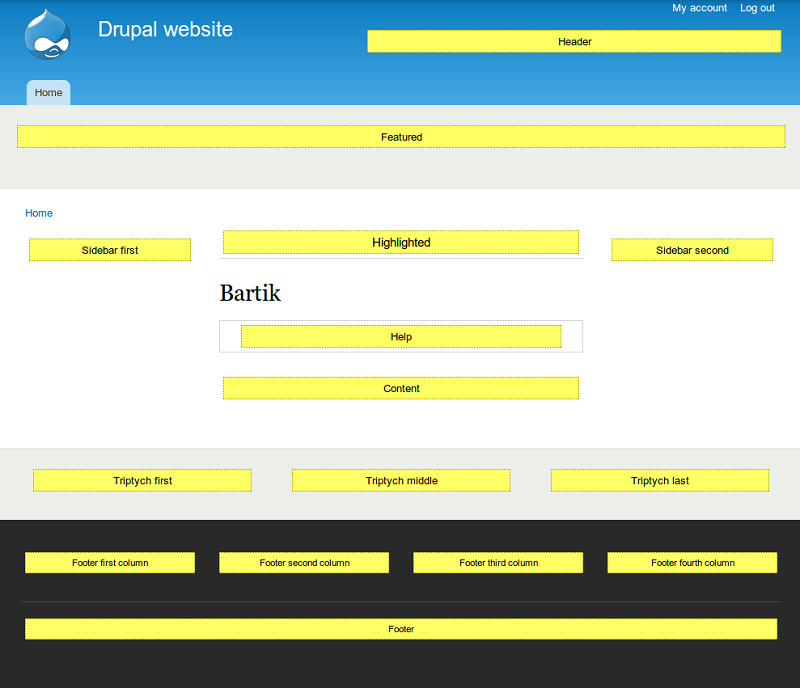
regions[footer\_firstcolumn] = Footer first column

regions[footer\_secondcolumn] = Footer second column

regions[footer\_thirdcolumn] = Footer third column

regions[footer\_fourthcolumn] = Footer fourth column

regions[footer] = Footer



BLOCK

Ví dụ như các khối: đăng nhập, tìm kiếm, online, ...

Có các cách tạo block như sau:

+ Tự tạo: Tạo trong backend và tùy chọn nội dung

+ Tạo từ view: Lúc tạo view hệ thống sẽ tạo thêm 1 block mặc định cùng tên với view đó, và cần kích hoạt để sử dụng block đó.

+ Cài đặt module từ bên ngoài vào, tùy module có viết thêm block không, nếu có vào trong kích hoạt lên và sử dụng.

=> Clear cache trong drupal: Configuration/development/performance/clear cache.

Lưu ý: sau khi clear cache, nếu có thay đổi trong cấu trúc của region (ví dụ như đổi tên) thì cần vào quản lý block để thay đổi và thiết lập lại vị trí hiển thị regions của block đó.

Cài đặt với block: Tùy vào mỗi block sẽ có các setting khác nhau, về cơ bản mặc định drupal sẽ có các cài đặt sau.

+ Pages: Một website có nhiều page, tùy chọn hiển thị page nào và không hiển thị page nào thì setting

+ Content type: Tích vào content type mà muốn block hiển thị, còn lại nó ẩn đi

+ Roles: Tích vào nhóm có quyền xem block này

SETTING Theme

Ví dụ: có icon các social network, chúng ta muốn cho người dùng nhập thông tin url cho các social đó.

Tạo file : theme-settings.php

Hoạt động của file setting này là hook vào các function core của sysem drupal

<?php

function toanlm\_form\_system\_theme\_settings\_alter(&$form, $form\_state) {

$form['google\_url'] = array(

'#type' => 'textfield',

'#title' => t('Google Url'),

'#description' => t("Place this text of google url."),

);

}

?>

để lưu được field kia, thì thêm khai báo biến trong file info.

settings[google\_url] = 'http://google.com'

Sau đó quay lại file setting để lấy giá trị của value cho input: '#default\_value' => theme\_get\_setting('google\_url', 'toanlm'),

google\_url: là tên biến trong settings file

toanlm: là tên module

Trong trường hợp nhiều option cùng một tính chất, để cho dễ quản lý và dễ nhìn ta nên group chúng lại và quản lý:

$form['social'] = array(

'#type' => 'fieldset',

'#title' => 'Social setting',

'#collapsible' => true,

'#collapsed' => false,

);

sau đó custom lại field trên từ mảng 1 chiều thành 2 chiều.

$form['google\_url'] thành $form['social']['google\_url']

để sử dụng các biến setting này, ta sử dụng gọi như sau trong file view.

$social\_display = theme\_get\_setting('social\_display', 'toanlm'); // Custom này tương tự với dự án joomla hôm trước, mình tùy chỉnh menu, background, color, background:, ...

Render block to view: ví dụ với region regions[main\_menu] = Main menu custom

<div id="navigation-main">

<?php if($page['main\_menu']) : ?>

<?php print render($page['main\_menu']); ?>

<?php endif; ?>

</div>

VIEW SLIDESHOW

Cài đặt module view slideshow vào

Tạo content type sử dụng module slidershow trên

Sau đó tạo field thuộc content type đó với tên là image và kiểu image.

Sau đó add slide và: menu content và chọn kiểu slider vừa tạo ở bước trên, tạo ví dụ 3 slider item

==> Cài module view để add view trong drupal + ctools

Tạo view: chọn kiểu là slider item, format slidershow

Không chọn tạo page, nhưng chọn tạo block và chọn kiểu slidershow (save and continue)

Tại FIELDS, add và chọn file để show (cần cài liblary + jquery plugin)

-===> và slidershow đã chạy + nếu cần custom lại style

Có thể custom thêm kích thước vùng ảnh hiển thị : https://www.youtube.com/watch?v=bX7ErAHJElE

Module debug block, pages, nodes trong drupal: Devel - sử dụng hàm krumo($bien);

==> page.tpl.php: trang khai báo chung những regions của layout template website, html markup, các biến nội dung, navigation,...

==> Sử dụng 1 số common biến: $site\_name, $logo, $title, $main\_menu, $directory...

Muốn không hiển thị title của block thì tại title: <none>

THÊM BÀI VIẾT VÀ NỘI DUNG CHO WEBSITE

Thêm bài viết đơn hoặc các bài viết dạng danh sách cho website, vào content/add content và chọn 1 trong 2 kiểu là Article va basic page.

Trong này có các tùy chọn thêm:

Revision informartion:  
Tạo một bản sao trước khi thay đổi, đây là cách bạn tạo một bản nháp khác nhau khi không vừa ý với bản đang chỉnh sửa có thể quay về bản nháp trước đó.  
URL path settings:  
Thay vì sử dụng địa chỉ mặc định của Drupal như localhost/node/1 bạn có thể thay thế bằng địa chỉ thay thế như localhost/thong-tin-ve-adidas để thân thiện hơn với người dùng và bộ máy tìm kiếm. Đây là cách bạn có thể chỉnh riêng lẻ từng bài hay sử dụng module để hỗ trợ tính năng này.  
Authoring information:  
Bạn có thể thay đổi về tác giả bài viết là ai và thời gian đăng bài.  
Publishing options:  
Các tùy chọn gồm:

Published: Bài viết được đăng trên website.

Promoted to front page: Đưa lên trang chủ

Sticky at top of lists: Đưa bài viết vào top các bài quan trọng.

TAXONOMY trong drupal: tương ứng với category, gồm 2 thành phần là vocabulary và term.

Một vocabulary có nhiều term liên quan tới nhau,

Tạo mới: vào structure/ taxonomy/ add vocabulary

vd: product, sau đó edit term và add các term vào: clothes, electronic, food and drink

sử dụng taxonomy, ...(later)

USER trong drupal 7 (peoples)

+ Tạo user mới

+ Gán quyền cho user

+ Phân quyền user

+ Thay đổi mật khẩu

VIEW trong drupal 7 (như đã trình bày ở trên thì view trong drupal bản thân core của nó không có sẵn, cho nên chúng ta cần cài thêm để sử dụng)

Nắm được và sử dụng các khối chức năng trong drupal của view

MENU trong drupal

Mặc định Drupal 7 có sẵn 4 menu gồm:

Main menu: Đây là menu chính của website.  
Usermenu: Đây là 1 menu dành cho user.  
Navigation menu: Đây là menu điều hướng của website.  
Management menu: Đây là menu dành cho admin hay nhà quản trị website.

Thêm mới menu

Để thêm mới 1 menu, ta vào admin/structure/menu. Hình bên dưới là giao diện tổng quát để quản trị menu.

Làm quen các thao tác với menu trong drupal

CONFIGURATION trong drupal

Content authoring

Mục này có thể nói quan trọng nhất là "**Text format**". Text format cho phép người dùng nhập liệu như thế nào vào website. Mặc định của Drupal 7 có 3 dạng text format gồm:

**Plain tex**t: Dạng văn bản thuần, không chấp nhận bất kỳ html nào. thường dùng cho dạng Text field.

**Filtered HTML**: Chỉ cho phép người dùng dùng 1 số thẻ HTML xác định, admin có thể tùy chỉnh theo ý mình. ví dụ như không cho thành viên thông thường được phép sử dụng thẻ <a>, <ol> .....

**Full HTML**: Cho phép người dùng sử dụng toàn bộ các thẻ HTML mà trình duyệt website có thể hiểu.

Khi được kết hợp với CKeditor hay WYSIWYG (What You See Is What You Get), full html sẽ cực kỳ quan trọng cho phép chúng ta tùy ý trình bày và trang trí văn bản như trong MS Word.

Ngoài ra chúng ta có thể Add thêm 1 dạng text format rồi config theo yêu cầu của chúng ta. Nhưng thường thì chúng ta không cần.

Development

Mục này cho phép chúng ta tùy chỉnh cấu hình về Performance của website. Chúng bao gồm:

**Performance**: Clear cach website và tùy chỉnh các cach cho website để tối ưu hóa tốc độ website dành cho người lướt web.

**Logging and errors**: Hệ thống lưu lại các thông tin, các vấn đề gặp phải như thế nào? Để xem các vấn đề hoạt động của website, hãy vào mục **Recent log messages** của mục **Reports**, hoặc vào link **admin/reports/dblog**.

**Maintenance mode**: Chuyển website sang chế độ bảo trì, khi trong chế độ bảo trì, những **roles** nào của thành viên được phép tiếp tục sử dụng website?

Media

Các tùy chỉnh về file, hình ảnh, nơi lưu chúng trên host... Mục này khá quan trọng và chúng ta cũng thường xuyên sử dụng.

**File system**: Mặc định các file upload lên website sẽ được lưu tại đâu? đường dẫn mặc định là **sites/default/files**. Thông thường thì chúng ta cũng không cần phải chỉnh sửa gì trong này. Vì files chúng ta upload lên website chứa trong thư mục default/files cũng khá hợp lý và gọn gàng rồi. Trừ trường hợp chúng ta muốn đặt files upload lên nơi khác thì mới sửa thôi. Lời khuyên là không nên thay làm gì.

**Image Styles**: Phần này khá quan trọng, hầu như dự án website nào dù nhỏ hay lớn cũng dùng đến nó.

Image styles để làm gì?

Khi giao website cho người khác sử dụng, cập nhật bài viết, hình ảnh lên website thì người sử dụng thường upload các files có kích thước không đồng đều. lúc thì 800x600, lúc thì 500x500 .... tỷ lệ hình khác nhau, khi đó ta phải tạo ra 1 số kiểu dáng ảnh (images style) theo kích thước đúng theo ý muốn của chúng ta về giao diện, ví dụ như 200x200. Khi đó, tất cả các files khách hàng upload sẽ được lưu trong mục default files với kích thước thật của files, đồng thời tạo ra 1 files khác với kích thước mà ta định trong image styles. nếu có nhiều styles thì hệ thống tạo ra nhiều hình ảnh luôn. mỗi style 1 hình.

Ví dụ: khách upload 1 tấm ảnh 1000x1000, hệ thống sẽ lưu nó. và đồng thời tạo thêm 1 file 200x200 do ta set thêm 1 images styles 200x200.

Để tạo thêm 1 image styles, ta vào mục **media\images style\add style**

Chọn phương thức xuất hình mà chúng ta muốn, ví dụ như **Resize, Scale, Rotate, Scale and Crop...** và bấm vào nút **Add**.

Bây giờ đơn giản, hãy nhập kích thước width và height chúng ta muốn. Và click vào **Add effect**

Mục Preview cho ta thấy, khi upload 1 tấm ảnh 800px x 600px, hệ thống sẽ tạo ra 1 hình ảnh style images khác với 200px x 200px.

**People**

Cho phép chúng ta tùy chỉnh các vấn đề về thành viên. Trong đó có **IP address block** để khóa 1 số IP mà chúng ta muốn họ không vào được website. Rất hiếm khi sử dụng. Account setting thì cho phép chúng ta tùy chỉnh những thứ liên quan đến tài khoản của thành viên. ví dụ như enable chức năng upload avatar của thành viên, signature của thành viên, nội dung của emails gửi đến thành viên khi đăng ký như thế nào, add thêm fields "Background" chẳng hạn.

Region and language

Cho phép chúng ta tùy chỉnh ngày giờ của hệ thống. Format ngày giờ date-time theo ý chúng ta. Ngoài ra, mục này quan trọng nhất là hệ thống đa ngôn ngữ. hiện tại chúng ta chưa cài modules liên quan đến đa ngôn ngữ nên cũng chẳng có gì làm nhiều ở mục này. sau này đến bài về hệ thống đa ngôn ngữ trong drupal, mình sẽ hướng dẫn sử dụng sâu hơn.

Để thêm mới 1 định dạng ngày giờ, ta vào **admin/config/regional/date-time**, sau đó **Add date Type**

System

Quan trọng nhất là site information, cho phép chúng ta thay đổi 1 số thông tin chính của website như **Site name, Site slogan, Site email address, trang chủ là trang nào, trang 403, 404 là trang nào**? Ta có thể sử dụng 1 bài viết của drupal, 1 view của drupal làm trang chủ, trang 403, 404.

THEME BASIC

Tập trung hướng dẫn chúng ta những can thiệp cơ bản vào 2 file themes\_name.info và page.tpl.php

Trường hợp muốn tạo nó là sub theme của 1 theme nào đó, trong file .info thêm dòng

base theme = garland // với garland là tên parent theme

Cách cách dùng css.

+ Khai báo toàn cục: trong file info - stylesheets[all][] = css/mytheme.css

+ Khai báo trong module:

$path = drupal\_get\_path('module', 'mysite');

// Include mysite.css.

drupal\_add\_css($path . '/css/mysite.css');

+ Khai báo dùng inline:

// Include inline CSS.

drupal\_add\_css('<style type="text/css">body

{ color: #000; }</style>', 'inline'); (Color module custom - greate)

Trường hợp với Internet explore

;stylesheets-conditional[IE][all][] = css/ie.css

stylesheets-conditional[lte IE 7][all][] = css/ie7.css

stylesheets-conditional[lte IE 6][all][] = css/ie6.css

stylesheets-conditional[IE 8][all][] = css/ie8.css

Điều kiện trong module với ie:

drupal\_add\_css(path\_to\_theme() . '/fix-ie.css', array('group' =>

CSS\_THEME, 'browsers' => array('IE' => 'lt IE 7', '!IE' => FALSE),

'preprocess' => FALSE));

======================

JS trong theme drupal, chèn vào file info - scripts[] = js/hello.js

Thêm vào 1 page cụ thể:

(function ($) {

Drupal.behaviors.mysiteHello = {

attach: function() {

alert("Hello World!!");

}

};

}(jQuery));

====================> STOP THEME <=======================

TOPIC: Module in drupal

Trong drupal có 3 kiểu module

+ module trong thư mục modules thư mục gốc là Core modules

+ module trong thư mục sites/all/modules là Add-on modules

+ module trong thư mục sites/default/modules là Custom modules

Ví dụ: Tạo module kiểu custom có tên là first, ta cần tạo thêm 2 file bên trong floder first là

- first.info: chức thông tin về module

- first.module:

Sau khi tạo xong vào quản lý module sẽ thấy module xuất hiện, và kích hoạt module đó lên nếu muốn.

===> có thể tạo case test: Tạm thời bỏ qua phần này, ...