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

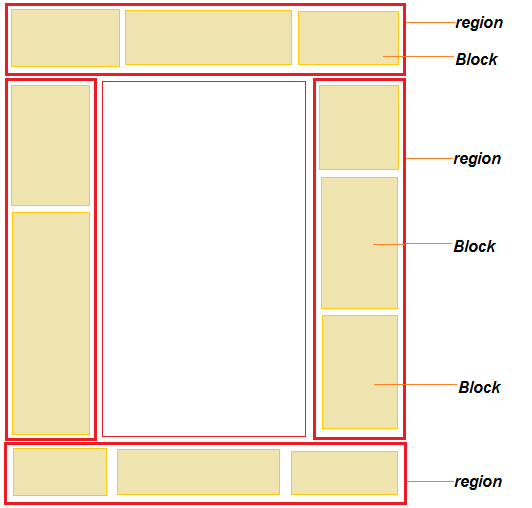
engine = phptemplate

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

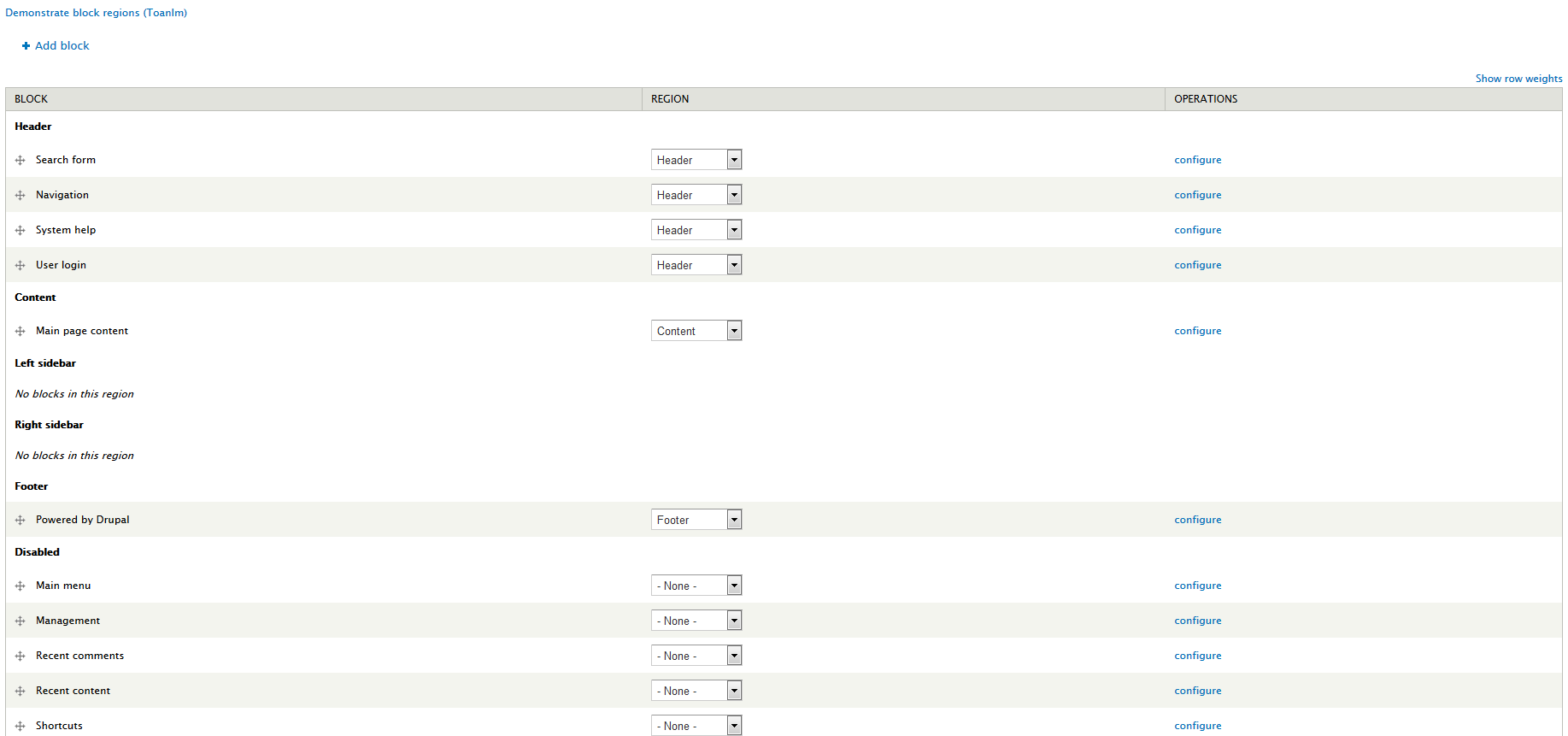
regions[footer] = Footer

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

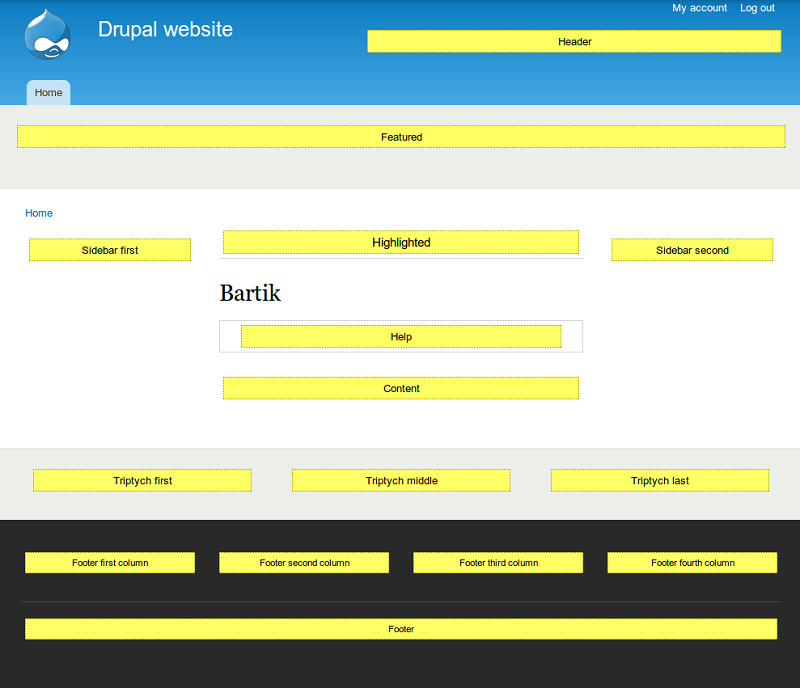
regions[footer\_firstcolumn] = Footer first column

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>