

Comment personnaliser des Templates dans le Framework Gantry

Par Luke Douglas

<http://webbering.com>

Traduction Française Par Serge Billon

<http://web54.fr>

Les gens merveilleux de [RocketTheme](#) ont développé un [Framework : Gantry](#) , qui rend la personnalisation de sites web nettement plus facilesi vous apprenez quelques petites choses. Voilà l'objet de ce simple guide.

Pour ce guide j'utiliserai le template Gantry par défaut pour Joomla 2.5 qui fait partie du [Gantry Template Bundle pour Joomla 2.5](#).

Je vous conseille d'installer le Pack puisque les plugins associés sont inclus.

Normalement, afin de personnaliser n'importe quel Template gantry, vous pouvez aussi bien modifier les CSS qu'ajouter vos propres images.

Il est conseillé de ne pas modifier les fichiers de base de Gantry parce qu'ils pourraient être remplacés lors des mises à jour du template et que vous perdriez toutes les modifications des CSS effectuées.

L'ajout d'images personnalisées au template tel que des images, logos, images de fond de modules images de fond des menus, etc... est simple : vous avez juste à télécharger vos images dans le dossier du template ou vous souhaitez qu'elles soient situées.

Dans Gantry, la structure de base des dossiers est la suivante :

- **admin**
- **css**
- **custom**
- **features**
- **html**
- **images**
- **js**
- **lib**

Les dossiers **admin**, **html**, **images** et **lib** comportent des sous-dossiers mais pour ce guide nous allons juste jeter un oeil sur le dossier images. Le dossier images est décomposé en sous-dossier :

- **images**
 - **body**
 - **header**
 - **icons**
 - **logo**
 - **menus**

Une décision que vous devez prendre est celle du logo. Dans le dossier **images/logo**, vous pouvez voir **logo.png**. Si vous utilisez le nom de cette image de fond, vous pouvez juste remplacer **logo.png** avec votre propre **logo.png**. Par conséquent, aucun changement ne sera nécessaire pour le nom de fichier image de votre logo dans les CSS. Toutefois, si vous téléchargez votre propre image avec un nom différent, comme par exemple **logo-perso.jpg**, vous devrez changer le nom de fichier de votre background-image dans votre CSS.

Ceci nous emmène à la prochaine étape, qui est les changements des feuilles de styles CSS.



J'ai inclus le tutoriel issu d'un forum posté par [Arifin FinLy](#) de Medan, Indonésie, sur 'Comment personnaliser vos fichiers CSS de votre installation Gantry' lequel je pense devrait faire partie de tous les templates Gantry. Commençons donc par l'article d'Arifin dont j'ai modifié certains paragraphes.

Comment ajouter des fichiers CSS *+

Ce tutoriel vous guidera sur la manière d'ajouter un style css personnalisé à un template construit avec Gantry sans vous soucier de perdre les changements quand le template ou gantry sera mis à jour.

Nous avons juste besoin d'ajouter une nouvelle fonctionnalité à Gantry. Nous n'avons pas besoin de hacker les fichiers d'origine.

1. Téléchargez la pièce jointe ci dessous et décompressez la sur votre bureau.
[@mycustomstyle.zip](#) (687 Bytes)
2. Placez mycustomstyle.php dans ce dossier sur le serveur:
<Racine de Joomla >/templates/<Votre Template >/features/
3. Placez mycustomstyle.php dans ce dossier sur le serveur:
<Racine de Joomla>/templates/<Votre Template>/css/
4. Ajoutez vos codes css personnalisés dans la mycustomstyle.css comme vous le souhaitez.

D'entrée, Gantry jettera un oeil dans le dossier 'features' pour voir s'il y a une quelconque modification PHP. En voyant le fichier **mycustomstyle.php**, il le chargera et exécutera son code. Dans ce cas, le fichier **mycustomstyle.php** file contient du code qui ajoute **mycustomstyle.css** à la section 'head' de la page web au moment de sa construction. Ce fichier peut être modifié pour ajouter d'autres feuilles de style CSS aussi bien que des fichiers javascript dans la partie 'head'. Je suis sûr que d'autres modifications personnalisées peuvent être ajoutées à ce fichier **mycustomstyle.php** mais nous nous en tiendrons strictement aux changements CSS pour ce tutoriel.

Si vous préférez ne pas télécharger les fichiers, voici le code que vous pouvez utiliser pour créer les fichiers directement:

mycustomstyle.php

```
<?php

/**
 * @package Custom Feature (Add custom stylesheet) – RocketTheme
 * @version August 1, 2010
 * @author RocketTheme http://www.rockettheme.com
 * @copyright Copyright (C) 2007 - 2010 RocketTheme, LLC
 * @license http://www.gnu.org/licenses/gpl-2.0.html GNU/GPLv2 only
 */

defined('JPATH_BASE') or die();

gantry_import('core.gantryfeature');

class GantryFeatureMyCustomStyle extends GantryFeature {

    var $_feature_name = 'mycustomstyle';

    function isEnabled() {

        return true;

    }

    function isInPosition($position)
    {

        return false;

    }

    function init() {

        global $gantry;

        //demo styles

        $gantry->addStyle("mycustomstyle.css");

    }

}
```

mycustomstyle.css

```
/* Add Your Custom CSS Codes Here */
```

Comment ajouter du code CSS personnalisé correctement*

La personnalisation de code que je trouve le plus demandé sur ce forum est : 'comment réduire l'espace (margin ou padding) d'un élément'.

Pour l'instant, c'est lié la plupart du temps à l'élément par défaut .rt-block. Je vais donc vous expliquer comment utiliser vos propres code CSS pour surcharger le code par défaut.

1. Par exemple, je veux réduire le padding et le margin de .rt-block dans le showcase.

Dans le template Gantry, vous trouverez :

1. `<div id="rt-showcase">`
2. `<div class="rt-grid-12 rt-alpha rt-omega">`
3. `<div class="rt-block">`
4. `Du contenu`
5. `</div>`
6. `</div>`
7. `</div>`

2. Avec firebug, vous verrez que les margin et padding ont été définis par défaut ainsi :

1. `/* Grid Block */`
2. `.rt-block {padding: 15px;margin-bottom: 10px;position: relative;}`

3. Si vous voulez surcharger (anglicisme : override) le padding et le margin et le mettre dans le style CSS personnalisé ci dessus, soyez attentifs à ce qui suit :

MAUVAIS CODE

1. `.rt-block {padding: 5px; margin-bottom:5px;}`

Raison pour laquelle cela ne fonctionnera pas correctement :

- Il y a d'autres éléments utilisant .rt-block et qui seront affectés par ce changement.

CODE CORRECT

1. `#rt-showcase .rt-block {padding: 5px; margin-bottom:5px;}`

Veuillez noter que vous devez placer #rt-showcase devant .rt-block, ainsi nous pouvons nous assurer que le .rt-block qui sera affecté par le code personnalisé est seulement celui qui se trouve à l'intérieur de la div #rt-showcase .

Note: Je tiens à montrer à Arifin ma grande reconnaissance pour ce si merveilleux guide aussi bien que pour les fichiers d'installation. Cela m'a simplifié la vie de pouvoir personnaliser mes templates sans m'inquiéter de ce que les mises à jour puissent écraser mes changements.

Continuons maintenant avec d'autres astuces de ma part!

Comment Ajouter Différents Changements de Feuilles de Styles**

Au cas où vous ne le sauriez pas, dans Joomla 2.5, vous pouvez sélectionner un template dans le Gestionnaire de Templates et cliquer sur dupliquer afin de créer une copie de ce template. Vous pouvez effectuer des changements pour ce template dupliqué, l'assigner à différents éléments de menu et ainsi avoir une apparence différente selon les pages.

Toutefois, que se passerait-il si vous vouliez avoir différentes images de fond pour certains modules ou pour certaines DIV avec des styles différents ? Avec les options du template, ce serait virtuellement impossible mais avec la personnalisation des CSS, ce sera facilement exécuté.

Disons que vous voulez avoir une image de fond (background-image) pour votre corps de page (body) sur la page d'accueil mais différentes autres pour toutes les autres pages.

Donc, vous dupliquez le template Gantry pour avoir deux Templates Gantry dans votre gestionnaire de templates. Vous ouvrez l'un ou l'autre, appliquez le style 1, assignez le template à toutes les autres pages que 'accueil' et vous en faites le template par défaut. Ensuite, vous ouvrez l'autre template gantry, appliquez le style 2 et l'assignez à la page 'accueil' mais pas aux autres pages.

Téléchargez maintenant vos images de fond personnalisées dans le dossier '**templates/<Nom de votre template>/images/body**'. Pour ce tutoriel, nous appellerons ces deux images '**bg-body-style1.jpg**' et '**bg-body-style2.jpg**' mais vous pouvez utiliser n'importe quel nom de fichier.

Ensuite, déterminez où les changements devront être faits en utilisant un utilitaire de debuggage dans un navigateur tel que Firebug pour Firefox. Pour ce tutoriel, j'utiliserai le code du template Gantry par défaut.

Allez dans le dossier '**templates/<Nom de votre template>/css**' et modifiez la feuille de style '**mycustomstyle.css**'.

Voici les codes exemple:

```
/* Add Your Custom CSS Codes Here */

.cssstyle-style1 body {background: #999 url(../images/body/bg-body-style1.jpg) !important;}

.cssstyle-style2 body {background: #fff url(../images/body/bg-body-style2.jpg) !important;}
```

Sauvegardez vos changements.

De base, nous disons à Gantry que si le template est assigné au style 1, alors il utilise la première déclaration, configure la couleur de fond à #999 et utilise l'image de fond 'bg-body-style1.jpg'. Si le template est assigné au style 2, alors il utilise la seconde déclaration, paramètre la couleur de fond à #fff et utilise 'bg-body-style2.jpg' comme image de fond.

Ainsi vous pouvez avoir différents fonds pour différentes pages mais avec tout de même une restriction ! Vous ne pouvez avoir autant de différents fonds qu'il y a de différents styles pour chaque template. Il existe une façon d'augmenter le nombre de styles, mais cela va bien au delà de l'objectif de ce tutoriel car cela implique de modifier directement les fichiers de base du template Gantry.

Exemples de modifications personnalisées**

Je vous présente quelques exemples issus du monde réel de code CSS que j'ai utilisé et qui fonctionne :

```
/* Add Your Custom CSS Codes Here */

.cssstyle-style1 #rt-menu .rt-container
{
height:36px !important;line-height:36px !important;
width:960px !important;
background: transparent url(..images/menus/bg-menu-light-transparent.png) top left no-repeat !
important;
}

.cssstyle-style2 #rt-menu .rt-container
{
height:36px !important;
line-height:36px !important;
width:960px !important;
background: #ffffff url(..images/menus/bg-menu.png) top left no-repeat !important;
}

.cssstyle-style1 #rt-main .rt-container
{
background: transparent url(..images/header/bg-header-transparent.png) !important;
}

.cssstyle-style2 #rt-main .rt-container
{
background: #ffffff !important;
}
```

```

.cssstyle-style1 #rt-maintop .rt-container
{
background: transparent url(..images/body/bg-header-transparent.png) top left no-repeat !important;
}

.cssstyle-style2 #rt-maintop .rt-container
{
background: #ffffff !important;
}

.cssstyle-style1 #rt-copyright .rt-container
{
background: transparent url(..images/body/bg-copyright-light-transparent.png) top left no-repeat !
important;
}

.cssstyle-style2 #rt-copyright .rt-container
{
background: #ffffff url(..images/body/bg-copyright-light.png) top left no-repeat !important;
}

.cssstyle-style1 #rt-feature .rt-container
{
background: transparent url(..images/header/bg-header-transparent.png) !important;
}

.cssstyle-style2 #rt-feature .rt-container
{
background: #ffffff !important;
}

```

Notez bien que j'utilise le code '**!important**' dans chacune de ces déclarations. Ce n'est pas toujours nécessaire mais si un changement CSS que j'ai fait ne marche pas (alors qu'il le devrait), j'ajoute ce code.

À l'exception d'une paire de modules, qui pour une raison quelconque ne prennent pas en charge les paramètres suivants, j'ai été capable de faire des personnalisations en utilisant la désignation '**.cssstyle-style#**'.

Comment Ajouter du Code Javascript Personnalisé

Si vous avez un fichier javascript à ajouter pour l'utiliser sur votre site, cela peut être fait grâce au fichier **mycustomstyle.php**.

D'abord, envoyez votre fichier javascript dans le dossier '**templates/<Le nom de votre template>/js**' .

Ouvrez le fichier **mycustomstyle.php** dans le dossier '**templates/<Le nom de votre template>/features**' et cherchez cette ligne dans la section '<head>' mais qui pourrait avoir un autre nom que '**mycustomstyle.js**' :

```
$gantry->addScript("mycustomstyle.js");
```

Si vous la trouvez, ajoutez votre script comme cela en modifiant **mycustomstyle2.js** ' pour le nom de votre fichier javascript:

```
$gantry->addScript(array(mycustomstyle.js ', mycustomstyle2.js '));
```

Enregistrez votre fichier.

Toutefois, si vous ne trouvez pas cette ligne, alors recherchez une ligne comme celle ci :

```
$gantry->addStyles(array('mycustomstyle.css'));
```

Ajoutez la ligne ci dessous en changeant 'mycustomscript.js' pour le nom de votre script.

```
$gantry->addScript("mycustomstyle.js ");
```

Enregistrez le fichier.

Rappelez vous que l'ordre de chargements des fichiers javascript sera celui dans lequel vous les avez entrés. La même chose vaut pour les CSS multiples qui pourraient être chargés à partir du fichier **mycustomstyle.php**.

Donc, vous pourriez tout aussi bien utiliser un fichier javascript externe en utilisant le lien URL javascript direct. Par exemple: Utilisez **https://ajax.googleapis.com/ajax/libs/jquery/1.5.1/jquery.min.js** à la place de **mycustomstyle2.js** dans le code ajouté. Cela devrait inclure dynamiquement votre fichier Javascript à chaque fois qu'une page sera chargée sur votre site.

J'espère que ce tutoriel vous sera utile dans votre découverte de Gantry.

Luke Douglas
Webbering.com

- * - Tutorial content by Arifin FinLy
- ** - Tutorial content by Luke Douglas
- + - Edited by Luke Douglas