

Astuces de l'ancien site de DanielSan - www.samare.net

- récupérées sur archive.org - sauvegarde du 12 oct 2014 -

[Astuces] Menu listant les statics classés par groupe !

Cela fait un bail que vous souhaitez réaliser un menu classant les pages statics dans leur groupe respectif ???

Rassurez-vous, c'est maintenant chose faisable avec l'astuce de la mort-qui-tue suivante ! 😊

Avant de commencer, il faut préciser que je n'ai pas encore trouvé comment gérer les pages statiques rajoutées par les plugins (avec par exemple le plugin de page contact ...), on verra cependant comment contourner ce soucis ... en attendant que je me penche sur le problème 😊

Cette astuce est normalement à rajouter dans votre menu horizontal du template header.php ou dans sidebar.php (comme il y a une légère différence de structure entre les 2, vous pourrez faire un simple copier/coller de celle que vous recherchez).

Je pense que le code est suffisamment commenté pour comprendre comme ça se passe 😊

Voici le code à insérer dans votre menu header.php

```
<?php // astuce statics classées par groupe !!!
if($plxShow->plxMotor->aStats) { // s'il y a des pages statics
    $menu_groupe = array(); // création du tableau des groupes
    $group_name='';

    // étape 1: à partir de la liste des statics, nous allons extraire leur groupe et effectuer un classement par groupe

    foreach($plxShow->plxMotor->aStats as $k=>$v) { // on boucle sur la liste des statics
        if(!empty($v['group']) AND $group_name!=$v['group']) { // si elles sont dans un groupe
            array_push($menu_groupe, $v['group']); // on remplit le tableau des groupes avec tous les noms de groupe existants
        } else {}
    }

    $menu_groupe = array_unique($menu_groupe); // on dédoublonne le tableau
    $menu_groupe = array_flip($menu_groupe); // on inverse les clefs / valeurs
    foreach($menu_groupe as $k=>$v) {
        $menu_groupe[$k] = array(); // on crée un sous tableau pour chaque groupe
    }

    foreach($plxShow->plxMotor->aStats as $k=>$v) { // on boucle sur la liste des statics
        if($v['active'] == 1 AND $v['menu'] == 'non') { // si la page est active et NON-affichée dans le menu ( les statics
            paramétrés comme affichés dans le menu seront gérés par la fonction staticList de PluXml )
            if($v['group']!="") { // si le groupe existe
                $menu_groupe[$v['group']][$k] = $v; // on rajoute une entrée pour chaque page static dans chaque groupe
            } else {}
        } else {}
    }

    // étape 2: affichage du menu
    foreach($menu_groupe as $k=>$v){ // on boucle sur les groupes
        echo "<li class=\"menu_static_groupe static_groupe_\".plxUtils::title2url($k).\">\".$k.\"</li>\"; // nouvel élément du
        menu avec le nom du groupe et une class associée
        echo "<ul>\"; // création d'une liste des statics de ce groupe
        foreach($menu_groupe[$k] as $i=>$j){ // on boucle sur chaque page static du groupe
            $url = $plxShow->plxMotor->urlRewrite('?static'.intval($i).\"/\".$menu_groupe[$k][$i]['url']); // formatage de
            l'url

            // gestion du status
            if($plxShow->staticId()){ // si nous sommes sur une page static
                $idStat = $plxShow->staticId(); // on récupère l'identifiant de la static active
                // on compare cet identifiant avec celui du menu
                if($idStat==intval($i)){ // s'il est égale à celui du menu, le status est actif
                    $status = "active";
                } else {
                    $status = "no-active";
                }
            } else {
                $status = "no-active";
            }
        } else {
            $status = "no-active";
        }
        echo "<li class=\"\".$status.\"><a href=\"\".$url.\"</a>\".$menu_groupe[$k][$i]['name'].\"</li>\"; // création du lien vers la static
    } // fin de la boucle des statics
    echo "</ul>\"; // on ferme la liste des statics
    echo "</li>\"; // on ferme l'élément du menu
    } // fin de la boucle des groupes
} else {} // magique ! ^^^
?>
```

Un point à préciser: comme je vous l'ai dit plus haut, je n'ai pas encore eu le temps de trouver comment gérer avec cette astuce les statics générées par des plugins. C'est pourquoi à ligne 21 on peut voir qu'on ne sélectionne que les statics ayant un groupe et NON-présentes dans le menu (paramètre géré dans l'administration des statics).
Nous allons donc coupler cette astuce avec la fonction staticList traditionnelle qui elle va afficher toutes les statics SAUF celles NON-affichées dans le menu. Vous me suivez ?

Maintenant l'astuce à utiliser dans votre sidebar.php

```
<?php // astuce statics classées par groupe !!!
if($plxShow->plxMotor->aStats) { // s'il y a des pages statics
    $menu_groupe = array(); // création du tableau des groupes
    $group_name='';

    // étape 1: à partir de la liste des statics, nous allons extraire leur groupe et effectuer un classement par groupe

    foreach($plxShow->plxMotor->aStats as $k=>$v) { // on boucle sur la liste des statics
        if(!empty($v['group']) AND $group_name!=$v['group']) { // si elles sont dans un groupe
            array_push($menu_groupe, $v['group']); // on remplit le tableau des groupes avec tous les noms de groupe existants
        } else {}
    }

    $menu_groupe = array_unique($menu_groupe); // on dédouble le tableau
    $menu_groupe = array_flip($menu_groupe); // on inverse les clefs / valeurs
    foreach($menu_groupe as $k=>$v) {
        $menu_groupe[$k] = array(); // on crée un sous tableau pour chaque groupe
    }

    foreach($plxShow->plxMotor->aStats as $k=>$v) { // on boucle sur la liste des statics
        if($v['active'] == 1 AND $v['menu'] == 'non') { // si la page est active et NON-affichée dans le menu ( les statics
            paramétrées comme affichées dans le menu seront gérées par la fonction staticList de PluXml )
            if($v['group']!="") { // si le groupe existe
                $menu_groupe[$v['group']][$k] = $v; // on rajoute une entrée pour chaque page static dans chaque groupe
            } else {}
        } else {}
    }

    // étape 2: affichage du menu
    foreach($menu_groupe as $k=>$v){ // on boucle sur les groupes
        echo "<h3>". $k. "</h3>\n"; // Le nom du groupe placé dans une balise de titre de niveau 3
        echo "<ul>\n"; // création d'une liste des statics de ce groupe
        foreach($menu_groupe[$k] as $i=>$j){ // on boucle sur chaque page static du groupe
            $url = $plxShow->plxMotor->urlRewrite('?static'.intval($i).'/'.$menu_groupe[$k][$i]['url']); // formatage de
l'url

            // gestion du status
            if($plxShow->staticId()){ // si nous sommes sur une page static
                $idStat = $plxShow->staticId(); // on récupère l'identifiant de la static active
                // on compare cet identifiant avec celui du menu
                if($idStat==intval($i)){ // s'il est égale à celui du menu, le status est actif
                    $status = "active";
                } else {
                    $status = "no-active";
                }
            } else {
                $status = "no-active";
            }
        }
        echo "<li class=\"\".$status.\"\"><a href=\"\".$url.\"\"
title=\"\".$menu_groupe[$k][$i]['name'].\"\">\".$menu_groupe[$k][$i]['name'].\"</a></li>\n"; // création du lien vers la static
    } // fin de la boucle des statics
    echo "</ul>\n"; // on ferme la liste des statics
} // fin de la boucle des groupes
} else {} // magique ! ^^
?>
```

La différence entre les 2 versions est que la première place le nom du groupe dans un élément de liste alors que le second dans un titre de niveau 3 (H3).

Voilà-voilà, j'espère que cette astuce va vous rendre service, vous pouvez commenter et surtout partager !

Ajouté le 08 avril 2012 - Classé dans : [Astuces](#) - Mots clés : [astuces](#), [contenu](#), [gestion](#), [menu](#), [PluXml](#) -

[Astuces] Connaître le nombre d'articles de la catégorie active

Il est utile de connaître le nombre d'articles présents dans la catégorie active notamment lorsqu'on lance un site.

En effet, la mise en page du site peut évoluer en fonction du nombre d'articles.

Imaginez par exemple que vous prévoyez d'[afficher la liste des articles de la catégorie active](#) mais que pour l'instant vous n'avez qu'un voir 2 articles ?

L'astuce ci-dessous vous permettra de gérer l'affichage en fonction du nombre d'articles présents dans la catégorie.

```
<?php if($plxShow->plxMotor->aCats[sprintf("%03d", $plxShow->artCatId())]["articles"]>3) : ?>
... vos actions ici
<?php endif; ?>
```

Explications:

```
$plxShow->artCatId()
```

Récupère l'identifiant de la catégorie active (un numéro)

```
sprintf("%03d", $plxShow->artCatId())
```

Formate ce nombre avec 3 chiffres (PluXml gère les catégories sous ce format)

```
$plxShow->plxMotor->aCats
```

aCats est un tableau contenant toutes les infos des catégories

```
$plxShow->plxMotor->aCats[sprintf("%03d", $plxShow->artCatId())]["articles"]
```

Va chercher le nombre d'articles de la catégorie active dans le tableau aCats ... 🍷enjoy !

J'utilise personnellement cette astuce lorsque je livre un site pour un client.

Je crée d'abord le site avec plusieurs articles, en mode "final". Puis je prévois une étape intermédiaire le temps que les articles soient publiés.

Exemple concret: un fabricant lance son premier produit ... puis 2, 3, etc ... cette anticipation m'évite de revenir sur le template du site.

Ajouté le 20 mars 2012 - Classé dans : [Astuces](#) - Mots clés : [astuces](#), [contenu](#), [gestion](#), [PluXml](#) -

[Astuces] Afficher le titre de la page des mots-clefs

Il est vrai qu'il manque aussi quelque chose dans le template par défaut: le titre de la page des mots-clefs.

Ce n'est pas vital, mais lorsqu'on est habitué à lire le titre de l'article de la page et qu'on ne tombe pas dessus, ça bouleverse un peu ...

Voici donc comment faire:

Un simple copier/coller de la ligne ci-dessous dans le template tags.php et vous aurez trouvé réponse à votre question !:

```
<h2>Liste des articles ayant pour mot-clef "<?php echo $plxShow->plxMotor->cible; ?>"</h2>
```

Note: ça ne prend pas la majuscule éventuelle du mot-clefs.

Vous pouvez [voir ce que ça donne sur ce site](#). J'en ai même profité pour rajouter le [plugin tagCloud](#) en chapô d'article ...

Ajouté le 31 janvier 2012 - Classé dans : [Astuces](#) - Mots clés : [astuces](#), [PluXml](#), [template](#), [ergonomie](#) -

[Astuces] Jouer avec la meta-balise "Description"

Nous sommes d'accord, renseigner les meta-balises c'est fastidieux ... 🤖

Trouver les bons mots-clés ça passe encore, mais rédiger la description est une opération un peu plus délicate ! Mais pourtant utile !

Découvrez avec cette astuce une autre bonne raison de renseigner ce champ et comment jouer avec cette donnée.

En théorie le chapô d'un article est son introduction, l'entrée en matière qui doit susciter l'intérêt d'aller plus loin.

La description de l'article est la synthèse de celui-ci résumée en une ou deux lignes.

Certains souhaitent générer automatiquement la description à partir du chapô.

Et si nous avions l'approche inverse ?

Renseigner et réutiliser le contenu de la meta-balise description

Découvrez comment et pourquoi !

Pour le "comment", copiez/collez simplement cette ligne de code et placez-la avant (ou à la place de) votre chapô:

```
<?php echo trim($plxShow->plxMotor->plxRecord_arts->f('meta_description')); ?>
```

Et pour le "pourquoi", regarder son utilisation dans l'affichage de la catégorie "Plugins".

Le survol de la vignette fait apparaître la description de l'article ! C'est pas génial ?! 🤖

En voilà une astuce qui vous donnera envie de renseigner tous les champs !

Ajouté le 21 janvier 2012 - Classé dans : [Astuces](#) - Mots clés : aucun -

[Astuces] Afficher un bloc avec les derniers articles d'une seule catégorie

On cherche parfois à afficher les 2-3 derniers articles d'une seule catégorie définie.

Soit en page d'accueil si l'on souhaite les mettre en avant ou dans la sidebar pour favoriser un peu plus leur lecture ...

Le cas concret est une entreprise présente sur des salons et souhaite mettre en avant leurs prochaines présences.

Elle indiquera donc sur toutes les pages "Nous serons présents au salon du nautisme à Toulon du 01 au 06 août au stand 12".

Cela simplifie la gestion des dernières news ...

Pour se faire, rien de plus simple, il vous suffit d'utiliser la fonction [lastArtList](#)

Vous pouvez voir un exemple en bas de la sidebar: j'ai affiché le dernier article de la catégorie "Vie du site".

Comment faire ? Comme d'hab', copiez/collez le code ci-dessous:

```
<h3>Dernières News</h3>
<ul>
<?php $plxShow->lastArtList('<li><h4><a href="#art_url" title="Lire la derni&egrave;re news
#art_title">#art_title</a></h4><p>#art_chapo</p><p class="more"><a href="#art_url" title="Lire la derni&egrave;re news
#art_title">En savoir plus</a></p></li>',1,001); ?>
</ul>
```

Dans cet exemple, 2 représente les 2 derniers articles et 001 le numéro de la catégorie.

Ajouté le 21 janvier 2012 - Classé dans : [Astuces](#) - Mots clés : [ergonomie](#), [gestion](#), [contenu](#) -

[Astuces] Afficher le titre de la catégorie

Il manque quelque chose dans le thème par défaut ... comment afficher le titre de la catégorie !

Un simple copier/coller du code ci-dessous dans le template categorie.php résoudra ce manque.

Tant que j'y suis, j'y ai rajouté le code pour afficher la description de la catégorie ... 😊

```
<h2><?php echo plxUtils::strCheck($plxShow->plxMotor->aCats[$plxShow->plxMotor->cible]['name']) ?></h2>
```

```
<div class="cat_description"><?php echo plxUtils::strCheck($plxShow->catDescription()) ?></div>
```

Note: si vous respectez les règles d'utilisation des balises HTML, le titre de la catégorie étant ici de niveau 2 (<h2></h2>), profitez en pour passer celui des titres des articles en <h3> ...

Ajouté le 19 janvier 2012 - Classé dans : [Astuces](#) - Mots clés : [contenu](#) -

[Astuces] Afficher la liste des articles de la catégorie active

Dans la lignée du [Menu listant les articles classés par catégories](#), voici une autre astuce qui consiste à afficher simplement la liste des articles de la catégorie active.

On la place souvent à gauche en tant que sous-menu ou en pied de page pour proposer à l'internaute d'aller visiter d'autres pages intéressantes.

Vous pouvez en voir un exemple sur cette page, juste dessous la zone article suivant/précédent.

Pour se faire, copiez (tout simplement) ce code et placez-le à l'endroit stratégique 📍 dans votre template article.php.

Le chiffre 20 indique le nombre de lien que vous souhaitez afficher.

```
<div id="autres_articles">
    <h4>Articles concernant la catégorie <?php $plxShow->artCat(); ?></h4>
    <ul>
<?php
    $idCat = sprintf("%03d", $plxShow->artCatId());
    $plxShow->lastArtList('<li class="#art_status"><a href="#art_url" title="#art_title">#art_title</a></li>', '20', $idCat);
?>
    </ul>
</div>
```

Explications:

- On relève d'abord l'identifiant de la catégorie active avec la fonction `$plxShow->artCatId()`.
- On la reformate selon 3 chiffres grâce à la fonction `sprintf()`
- Puis on utilise la fonction d'affichage des derniers articles (`$plxShow->lastArtList`) en filtrant ceux concernant la catégorie active (notre `$idCat`).

Tadaaam ! 😊

Ajouté le 18 janvier 2012 - Classé dans : [Astuces](#) - Mots clés : [ergonomie](#), [menu](#) -

[Astuces] Un formulaire optimal

Comment réaliser un bon formulaire web ?

En jouant sur l'esthétique ? En utilisant des principes ergonomiques ? En usant de convivialité ?

Découvrez d'abord ces 2 lois très profitables: **la loi de Fitts et la loi de Hick**.

1/ Loi de Fitts:

Le temps nécessaire pour atteindre une cible dépend de sa taille et de son éloignement

> au lieu de passer à l'étape suivante sur une nouvelle page web, débrouillez-vous pour afficher cette étape sur la même page et au le plus près de la position du pointeur. (en faisant défiler la page automatiquement par exemple)

2/ Loi de Hick

Le temps nécessaire pour prendre une décision est proportionnel au nombre d'options possibles.

> si le nombre d'options suivantes peut-être réduit en fonction du choix précédent, faites-le.

En utilisant "simplement" ces 2 lois, les internautes rempliront plus rapidement et facilement vos formulaires !

Ajouté le 22 décembre 2011 - Classé dans : [Astuces](#) - Mots clés : [ergonomie](#) -

[Astuces] Menu listant les articles classés par catégories

Il est intéressant d'afficher dans notre menu les liens pointant vers nos catégories. Mais c'est encore mieux de pouvoir afficher en plus les articles associés !

Vous pouvez voir un exemple sur ce site avec son menu horizontal.

Découvrez comment avec l'astuce ci-dessous à placer dans le template header.php :

```
<?php
$menu = array('001', '002'); // lister ici les catégories à afficher.
foreach($menu as $id_cat) {
    $plxShow->catList('', '<li id="menu_#cat_id" class="#cat_status">', $id_cat);
    echo "\n";
    $plxShow->catList('', '<a href="#cat_url" title="#cat_name">#cat_name</a>', $id_cat); //affiche le lien vers la catégorie
    echo "\n";
    echo "<ul>\n";
    $nb_item = 7; // nombre d'articles maxi dans le menu
    $plxShow->lastArtList('<li class="#art_status"><a href="#art_url"
title="#art_title">#art_title</a></li>', $nb_item, $id_cat); //liste les articles de cette catégorie
    $nbArtCat = $plxShow->nbAllArt('', $id_cat);
    if($nb_item > $nbArtCat) { // si le nombre d'articles de la catégorie est supérieur à $nb_item, on affiche un lien
"Voir tous les articles"
        $plxShow->catList('', '<li id="menu_#cat_id"><a href="#cat_url" title="Voir tous les articles de la
cat&eacute;gorie #cat_name">Voir tous les articles</a></li>', $id_cat);
    } else {
        echo "\n";
        echo "</ul>\n";
    }
    echo "</li>\n";
}
?>
```

Voici le menu tel quel du site de samare (avec le lien accueil, les statics et tout et tout):

```
<!-- DEBUT Menu_site -->
<div id="menu">
<ul>
<li <?php if ($plxShow->mode()=="home"):?>class="active"<?php endif; // active l'élément si on est sur la page
d'accueil ?>>
<a href="<?php $plxShow->urlRewrite();?>" title="Retour &agrave; l'accueil de <?php $plxShow->mainTitle();
?>">Accueil</a>
</li>
<?php
$menu = array('002', '003'); // lister ici les catégories à afficher.
foreach($menu as $id_cat) {
    $plxShow->catList('', '<li id="menu_#cat_id" class="#cat_status">', $id_cat);
    echo "\n";
    $plxShow->catList('', '<a href="#cat_url" title="#cat_name">#cat_name</a>', $id_cat); //affiche le lien vers
la catégorie
    echo "\n";
    echo "<ul>\n";
    $nb_item = 6;
    $plxShow->lastArtList('<li class="#art_status"><a href="#art_url"
title="#art_title">#art_title</a></li>', $nb_item, $id_cat); //liste les articles de cette catégorie
    $nbArtCat = $plxShow->nbAllArt('', $id_cat);
    if($nb_item > $nbArtCat) {
        $plxShow->catList('', '<li id="menu_#cat_id"><a href="#cat_url" title="Voir tous les articles de
la cat&eacute;gorie #cat_name">Voir tous les articles</a></li>', $id_cat);
    }

    echo "\n";
    echo "</ul>\n";
    echo "</li>\n";
}
?>
<?php $plxShow->staticList('', '<li id="#static_id" class="#static_status"><a href="#static_url"
title="#static_name">#static_name</a></li>'); ?>
<?php $plxShow->pageBlog('<li id="#page_id" class="#page_status"><a href="#page_url"
title="#page_name">#page_name</a></li>'); ?>
<li><a href="<?php $plxShow->urlRewrite('index.php?article8/menu-listant-les-articles-classes-par-categories'); ?>"
class="aide right"><span><b>Il y a une ASTUCE ici <i>!</i></b> Cliquez pour d&eacute;couvrir comment cr&eacute;er un menu
listant les articles par cat&eacute;gorie.</span></a></li>
</ul>
</div>
<!-- FIN Menu_site -->
```

Ensuite, vous pouvez utiliser ces règles CSS pour mettre en forme ce menu:

```
/* ----- #menu GABARIT ----- */
#menu { clear: both;
position: relative; z-index: 10;
width: 100%; margin: 0;
font-size: 15px; text-align: left; }
#menu ul {display: block; width: 1000px; height: 30px; margin: 0 auto;}
#menu ul li {list-style: none; position: relative; float: left; display: block; margin: 0 5px;}
#menu ul li a {display: block; line-height: 30px; padding: 0 10px; text-align: center}
#menu ul li ul {display: block; width: 280px; height: 0; overflow: hidden; padding: 0; position: absolute;}
#menu ul li ul li {clear: both; width: 280px; text-align: left;}
#menu ul li ul li a {line-height: 1.2em; width: 260px; padding: 10px; margin: 5px 0; text-align: left}
#menu ul li:hover ul {height: auto; padding: 20px 0;}

/* ----- #menu STYLE ----- */
#menu ul li ul {background-color: #FFF}
Enjoy!
```

Ajouté le 14 décembre 2011 - Classé dans : [Astuces](#) - Mots clés : [PluXml](#), [gestion](#), [creation](#), [menu](#) -

[Astuces] clearfix: annulez un float quasi automatiquement

Aïe la galère la gestion des blocs flottants ...

On souhaite aligner certains blocs à l'horizontal,

MAIS POURQUOI celui d'après n'est-il pas à sa place ??? 🤔

Dessous, j'ai dit que je le voulais de-ssous !!! ^^ 🤔

On se calme, une solution simple et élégante existe, le CLEARFIX CSS !

Dans un premier temps, créez cette règle CSS:

```
.clearfix:after {
  content: ".";
  display: block;
  height: 0;
  clear: both;
  visibility: hidden;
}
```

Puis lorsque vous souhaitez annuler le flottement des blocs,

il vous suffit tout simplement d'**attribuer la class clearfix** au dernier bloc flottant.

Que signifie cette règle ?

Après le bloc définit par la class clearfix, on ajoute un block de hauteur nulle, invisible, annulant le flottement des 2 côtés et contenant un point.

Finis les chutes de ch'veux ! (et surtout allègement de votre feuille de style ...) 🤔🤔

Ajouté le 11 décembre 2011 - Classé dans : [Astuces](#) - Mots clés : [PluXml](#), [astuces](#), [css](#) -

[Astuces] Attribuez automatiquement un identifiant et une class au body

Il vous est possible de différencier vos styles selon la catégorie, l'article et le mode.

Une méthode est d'associer une feuille de style au mode, au template, à la catégorie, à l'article qui ne s'affichera qu'en fonction du type de la page.

Une autre méthode est possible: en attribuant automatiquement un identifiant et une class au body.

L'identifiant doit être unique sur une page, on lui donnera celui de l'article, la page static ou la categorie.

La class est plutôt "généraliste", on lui attribuera celui du mode et du template.

Voici le code à insérer dans votre template header.php en lieu et place de <body>:

```
<?php
$body_class = $plxShow->mode()." ".str_replace('.php','',$plxShow->plxMotor->template);
$body_id = $plxShow->mode();

if($plxShow->mode()=='static'){
    $body_id .= $plxShow->staticId();
} elseif($plxShow->mode()=='article'){
    $body_id .= $plxShow->artId();
} elseif($plxShow->mode()=='categorie'){
    $body_id .= $plxShow->catId();
}

?>
<body class="<?php echo $body_class; ?>" id="<?php echo $body_id; ?>">
```

Ajouté le 08 décembre 2011 - Classé dans : [Astuces](#) - Mots clés : aucun -

[Boîte à outils] Des maquettes web

Ajouté le 07 juin 2012

Avant de se lancer dans la création d'un template esthétique, il est de bon usage de passer par l'étape maquette. Cette étape permet de positionner les différents blocs de contenus ou des fonctions, sans forcément développer un template en CSS !

En effet, si vous foncez directement dans le développement, à chaque modification il va falloir vous replonger dans le code ... perte de temps.

Voyons comment être plus efficace.

Nous avons déjà vu qu'il existait des [gabarits papier](#). Ceux-ci servent à "gribouiller" en direct avec votre client ou "machinalement" sur le coin de votre bureau, dans le métro, ...

Vous pouvez cependant arriver à un résultat plus propre pour une meilleure visualisation et communication (certains clients ont du mal à imaginer un rendu).

En effet une maquette est caractéristique du site web afin de disposer d'un support représentatif qui permet de débiter le développement.

Il n'est donc pas demandé que la maquette produite réalise la totalité des fonctionnalités demandées par le cahier des charges ni qu'elle soit représentative à 100% de la conception définitive.

Pour y arriver, vous pouvez utiliser des logiciels graphiques (illustrator, photoshop, ...) et y tracer des blocs types. Vous pouvez même créer votre propre bibliothèque d'éléments que vous insérerez au fur et à mesure de votre conception.

Un bloc menu par-ci, un bloc recherche par-là, etc...

Cependant, il existe déjà des outils pour ce type d'utilisation: les [wireframes](#)

Parmi ceux que j'ai pu trouver et tester, il y en a un qui a retenu mon attention: [MockFlow](#)

En effet celui-ci s'utilise en directement en ligne: vous créez une page et un layout avec le nombre de ligne et de colonne pour y déposer vos éléments par un simple déplacement de ceux présents dans la bibliothèque.

Vous pouvez ensuite exporter votre création sous différents formats: PDF, image, etc ...



Voici un rendu d'une page d'accueil réalisée en une 10aine de minutes:

[Boîte à outils] Du faux texte

Ajouté le 09 mai 2012

Lorsque vous créez un template de zéro, vous n'avez généralement pas de texte pour vérifier votre composition et mises en page.

Il existe des générateurs de texte de substitution comme dont le plus connu est le Lorem Ipsum:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque ornare, sem eu molestie sodales, est justo eleifend nunc, sit amet pharetra libero risus id mi. Proin dapibus purus augue, ut suscipit odio. Etiam venenatis dapibus lorem blandit egestas. Mauris ut nisi non purus pretium volutpat ac a erat. Phasellus vitae mi ipsum, ut hendrerit metus. Vestibulum molestie suscipit enim a sagittis. Pellentesque cursus metus a urna dignissim adipiscing cursus dolor adipiscing.

Quisque iaculis tempor fringilla. Sed vulputate pulvinar faucibus. Donec lectus risus, dignissim nec euismod vitae, sodales eget massa. In hac habitasse platea dictumst. Phasellus vitae imperdiet diam. Mauris in quam orci, et ultrices nisl. Aenean ut suscipit mi. Maecenas tempor nibh vitae nunc condimentum nec commodo arcu lacinia.

Il a l'avantage de donner l'impression que c'est un vrai texte avec une vraie signification puisque sa syntaxe est réelle.

Le site <http://lipsum.com/> vous permet de créer ce faux texte en choisissant le nombre de paragraphe ou de mot souhaité.

Un outil bien pratique et mieux que du bla-bla répété !

Classé dans : [Boîte à outils](#) - Mots clés : [template](#), [contenu](#), [creation](#), [astuces](#)

[Boîte à outils] Le nombre d'or

Ajouté le 09 mai 2012

Lorsqu'on cherche à donner une dimension à une forme, on ne sait pas trop laquelle lui donner.

Si vous hésitez, utilisez le nombre d'or !

Demandez à 100 personnes de dessiner un rectangle qu'elles trouvent bien proportionné.
Maintenant faites la moyenne du ratio longueur/largeur et vous tomberez sur ... 1,61803399 !

Ce chiffre est le nombre d'or. On le retrouve partout autour de nous et pas seulement dans ce que nous construisons mais aussi dans la nature.

Allez [faire un petit tour sur wikipédia](#) si vous voulez creuser le sujet.

Il est souvent utilisé, alors pourquoi pas vous ?!

Si vous souhaitez scinder une largeur de 100 en 2 parties esthétiquement bien répartie selon la moyenne, alors une partie fera 62 et l'autre 38.

Si vous modifiez ce ratio vous modifierez aussi la perception de l'une sur l'autre ...

Et en web-design, ça donne quoi ?

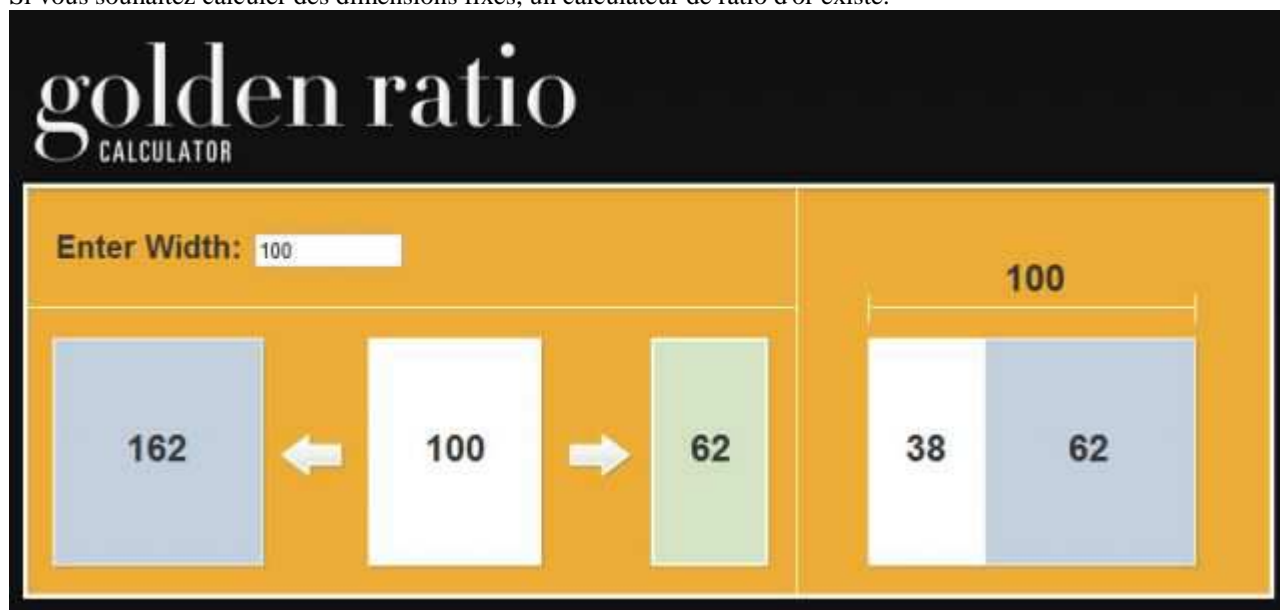
Le gabarit ci-dessous est composé selon le ratio d'or 62% / 38%.



- La zone centrale en gris clair+foncé fait 62% de la largeur de la fenêtre.
- Le titre du site (header) fait 38% de la largeur centrale et donc la zone #header_extra fait 62% ...
- la sidebar fait 38% de la zone centrale
- la liste des commentaires fait 62% de la zone de contenu et donc la zone de formulaire 38%
- etc, etc ...

Ici le ratio ne concerne que les largeurs. Si vous voulez pousser plus loin, il faudrait connaître la hauteur de la fenêtre pour obtenir des divines proportions ...

Si vous souhaitez calculer des dimensions fixes, un calculateur de ratio d'or existe:



<http://goldenratiocalculator.com>

Classé dans : [Boîte à outils](#) - Mots clés : [ergonomie](#), [contenu](#), [template](#), [creation](#)

[Boîte à outils] Des gabarits/templates vierges

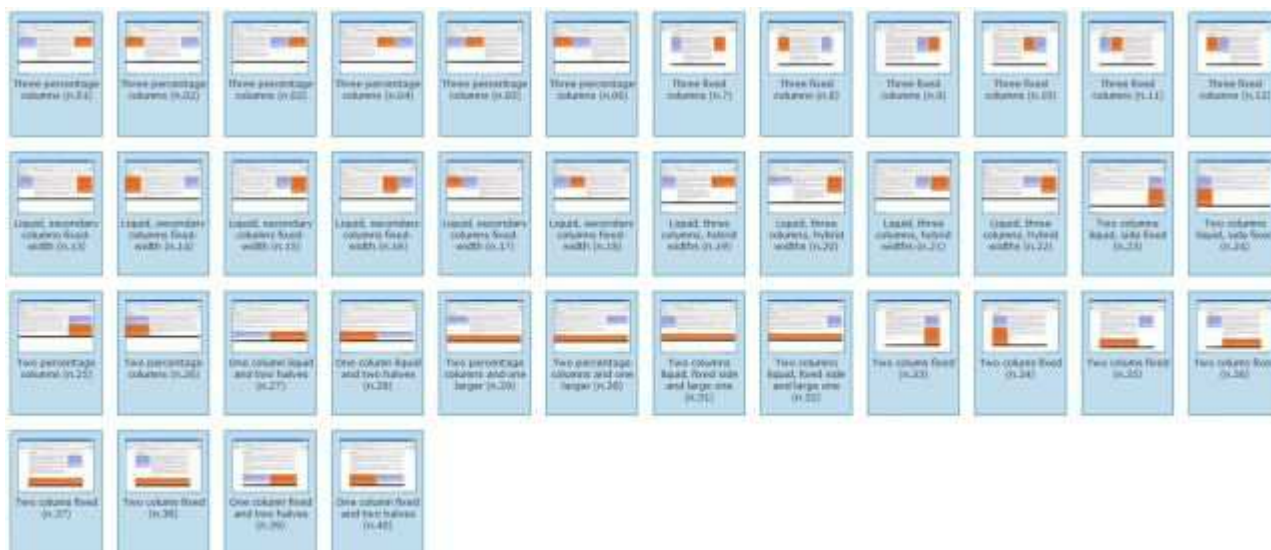
Ajouté le 19 février 2012

Pour vous aider à créer votre template, prenez un bon départ avec un gabarit vierge bien formaté.

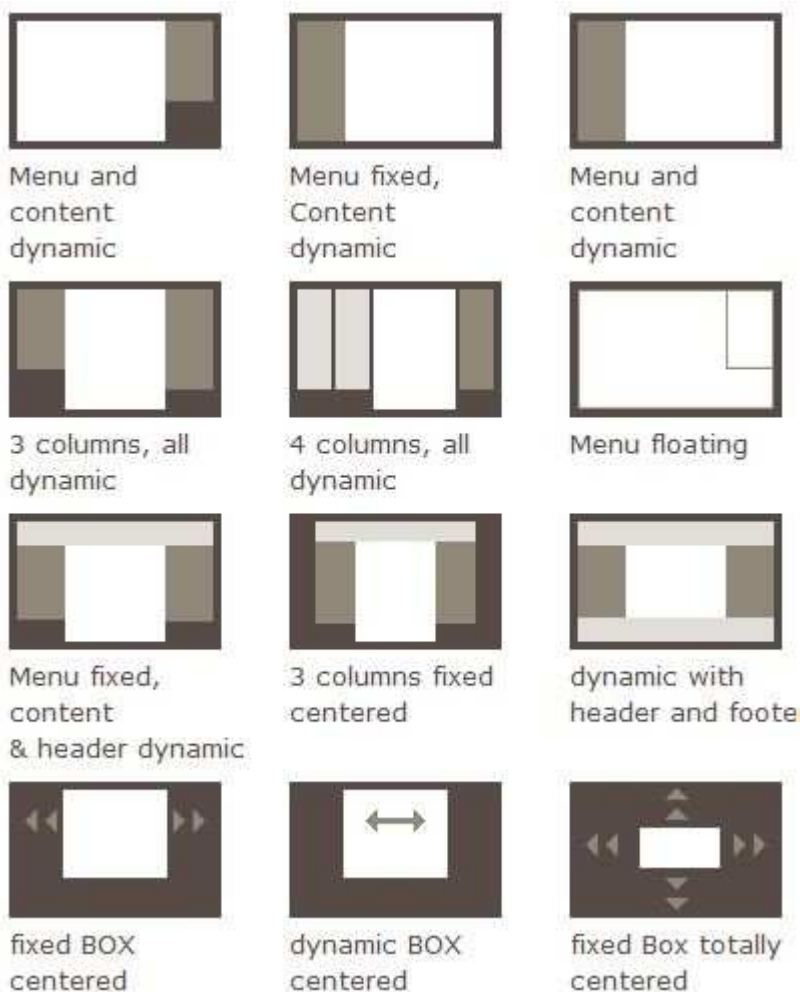
Vous trouverez en effet sur ces sites une série de template à n colonne fixes ou fluides.

Il ne vous reste plus qu'à les remplir et styliser !

Un bon exercice pour celles et ceux qui souhaitent partir de zéro mais pas trop ... 😊



[Des templates vierges sur http://blog.html.it/layoutgala/](http://blog.html.it/layoutgala/)



[D'autres templates vierges sur http://www.intensivstation.ch/en/templates/](http://www.intensivstation.ch/en/templates/)

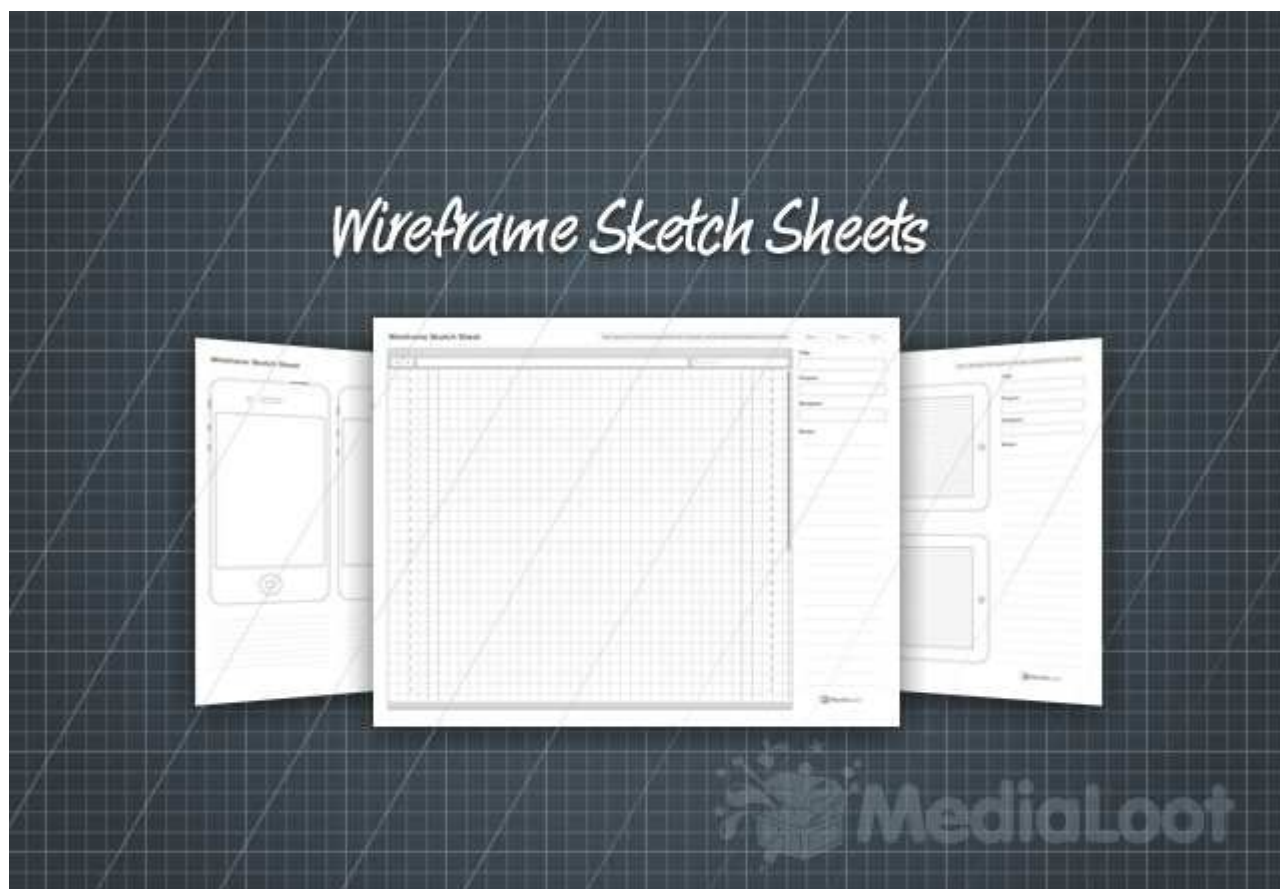
Classé dans : [Boîte à outils](#) - Mots clés : [creation](#), [css](#), [template](#)

[Boîte à outils] Des gabarits papier pour croquer votre site web

Ajouté le 19 février 2012

Lorsqu'on commence la création d'un site web, on gribouille généralement quelques blocs sur du papier. Il y a des choses que l'on fait à chaque fois: tracer la fenêtre, positionner des guides, etc ...

Vous trouverez sur ce site des gabarits vierges au format PDF que vous pourrez imprimer !



[Téléchargez, imprimez et gribouillez les gabarits papiers sur http://medialoot.com/item/free-wireframe-sketch-sheets/](http://medialoot.com/item/free-wireframe-sketch-sheets/)

Classé dans : [Boîte à outils](#) - Mots clés : [creation](#), [css](#), [template](#)

[Plugins] tagCloud: vos mots-clefs dans les nuages !

(NB : une fonctionnalité similaire a été intégrée par défaut à la version 5.1.7 de PluXml)



1. Attribue à votre liste de mots-clefs une class en fonction du nombre de fois que ce mot-clef est utilisé. Il vous suffit ensuite de définir à quoi correspond cette class avec un style CSS (taille, couleur, etc ...)
2. Affiche la liste dans un ordre aléatoire.

Si vous souhaitez avoir le même rendu que sur ce blog, vous pouvez rajouter ces règles de CSS dans votre feuille de style:

```
/* ----- TAGCLOUD ----- */
ul#tagCloud li{float: left;}
ul#tagCloud li a{margin: 0 10px 10px 0;}
ul#tagCloud li.tag-smallest a {font-size: 0.6em;}
ul#tagCloud li.tag-small a {font-size: 0.8em;}
ul#tagCloud li.tag-medium a {font-size: 1em;}
ul#tagCloud li.tag-large a {font-size: 1.5em;}
ul#tagCloud li.tag-largest a {font-size: 2em;}
```

[Téléchargez tagCloud V.1.1 du 16/01/2012](#)

- Fiche produit
 - Titre:tagCloud
 - Dernière version:1.1 du 16/01/2012
 - Version PluXml supportée:5.1.5
 - Fichier config.:non
 - Fichier aide:oui
 - Panneau admin.:non
 - Gestion des langues:non
 - Nécessite une feuille de style:recommandé
 - Nécessite jQuery:non
 - Licence:CC-BY-NC-SA



- Captures d'écran



- Historique

- V.1.0 du 01/12/2011: lancement !
- V1.1 du 16/01/2012: correction d'une coquille dans le code présent dans l'aide pour insérer le plugin dans le template ...

Ajouté le 11 décembre 2011 - Classé dans : [Plugins](#) - Mots clés : [PluXml](#), [plugins](#), [creation](#), [tagcloud](#) -

[Plugins] filAriane: informez l'internaute de l'endroit où il se trouve

⇒ voir le plugin officiel équivalent plxMyBreadcrumb de Stéphane



Ce plugin propose une aide à la navigation sous forme de signalisation de la localisation du lecteur dans votre site internet.

A placer dans le template header.php

[Téléchargez filAriane V.1.1 du 21/01/2012](#)

- Fiche produit
 - Titre:filAriane
 - Dernière version:1.1 du 21/01/2012
 - Version PluXml supportée:5.1.5
 - Fichier config.:non
 - Fichier aide:oui
 - Panneau admin.:non
 - Gestion des langues:non
 - Nécessite une feuille de style:non
 - Nécessite jQuery:non
 - Licence:CC-BY-NC-SA

- Captures d'écran

Ajouté le 11 décembre 2011 - Classé dans : [Plugins](#) - Mots clés : [PluXml](#), [plugins](#), [creation](#), [tagcloud](#) -

- Historique

V.1.1 du 01/01/2012: lancement !

V.1.0 du 21/01/2012: petit oubli (non-dramatique): ajout du fil pour les archives



-

Ajouté le 11 décembre 2011 - Classé dans : [Plugins](#) - Mots clés : [PluXml](#), [plugins](#), [creation](#), [breadcrumb](#), [fil d'ariane](#) -

[Plugins] artPS: affichez les liens suivants/précédents



Certains connaissent cette fonction proposée par Stéphane qu'on rajoutait dans le coeur de PluXml, le voici maintenant en plugin !

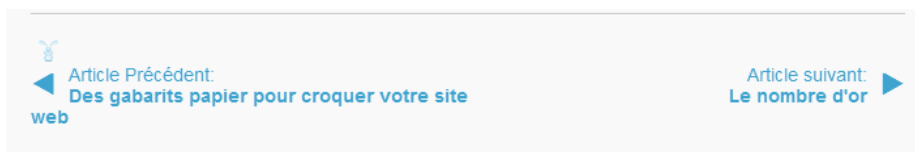
Un paramètre permet d'afficher les articles suivants/précédents dans sa catégorie ou dans l'ensemble du site.

Cadeau bonus! un exemple de feuille de style est proposé dans l'aide !

[Téléchargez artPS V.1.1 du 16/01/2012](#)

- Fiche produit
 - Titre:artPS
 - Dernière version:1.1 du 16/01/2012
 - Version PluXml supportée:5.1.5
 - Fichier config.:non
 - Fichier aide:oui
 - Panneau admin.:non
 - Gestion des langues:oui
 - Nécessite une feuille de style:un exemple en cadeau !
 - Nécessite jQuery:non
 - Licence:CC BY-NC-SA 3.0

- Captures d'écran



- Historique

V.1.0 du 01/12/2011: lancement !

V1.1 du 16/01/2012: correction d'une coquille dans le code présent dans l'aide pour utiliser la feuille de style



Ajouté le 06 décembre 2011 - Classé dans : [Plugins](#) - Mots clés : [PluXml](#), [plugins](#), [gestion](#), [creation](#) -

[Plugins] champArt: ajoutez, gérez et regroupez des champs à vos articles !



Isoler certaines données de votre contenu permet d'optimiser son affichage et la gestion de votre site. L'exemple le plus connu est d'associer une image à un article. Ce plugin vous permet de créer autant de champs que vous souhaitez et de les regrouper.

CeriseS sur le gâteau:

1. dans l'admin, le champ s'affiche que s'il est renseigné (comme pour le chapô de l'article)
2. en jouant sur les paramètres vous pouvez facilement afficher ou utiliser les valeurs souhaitées
3. et d'autres petites astuces dans le fichier d'aide !

Un plugin à utiliser sans modération !

[Téléchargez champArt V.1.0 du 01/12/2011](#)

Un problème ? Une question ?

[Rendez-vous sur la page consacrée à champArt sur le forum de PluXml](#)

- Fiche produit
 - Titre:champArt
 - Dernière version:1.0 du 01/12/2011
 - Version PluXml supportée:5.1.5
 - Fichier config.:oui
 - Fichier aide:oui
 - Panneau admin.:non
 - Gestion des langues:oui
 - Nécessite une feuille de style:non
 - Nécessite jQuery:non
 - Licence:CC-BY-NC-SA

- Captures d'écran



- Historique

V.1.0 du 01/01/2012: lancement !



Ajouté le 06 décembre 2011 - Classé dans : [Plugins](#) - Mots clés : [PluXml](#), [plugins](#), [gestion](#), [creation](#) -