

## TP CMS (séance 6 Web)

### Publication du site sur le Web

Jusqu'à maintenant vous avez travaillé sur le serveur de développement Iparla. Le site élaboré lors des séances précédentes va maintenant être mis en production c'est-à-dire publié sur le Web. Vous allez donc transférer votre site sur Adartza, la base de données restant sur Kartxila.

Pour des raisons de sécurité, Adartza ne sera probablement pas accessible directement depuis l'extérieur de l'IUT mais en passant par la VDI (virtual desktop infrastructure) MonIUT

1. Vous allez modifier l'adresse de votre site pour qu'elle soit sur adartza ( <http://adartza.iutbayonne.univ-pau.fr/~votrelogin/wordpress> ) à partir de phpMyAdmin en réalisant les opérations suivantes :
  - a. Modifiez l'adresse dans la base de données dans la table wp\_options en modifiant les valeurs des options siteurl et home.
  - b. Modifiez les liens contenus dans vos articles (ou autres types de contenu) Le plus simple est d'utiliser les requêtes *SQL* suivantes sur votre base de données (en utilisant *phpMyAdmin*, et après avoir au minimum remplacé *votrelogin* par votre login) :

-- On met à jour les options

```
UPDATE wp_options SET option_value='http://adartza.iutbayonne.univ-pau.fr/~votrelogin/wordpress' WHERE option_name='siteurl';
```

```
UPDATE wp_options SET option_value='http://adartza.iutbayonne.univ-pau.fr/~votrelogin /wordpress' WHERE option_name='home'
```

-- On met à jour le contenu des posts

```
UPDATE wp_posts set post_content = REPLACE(post_content, 'http://iparla/~votrelogin/wordpress', 'http://adartza.iutbayonne.univ-pau.fr/~votrelogin/wordpress');
```

Attention dans cette requête est indiquée l'adresse de votre site sur iparla qui n'est pas forcément <http://iparla/~votrelogin/wordpress>: il vous faut indiquer l'adresse exacte.

Attention, si vous utilisez un copier coller dans phpMyAdmin, vérifiez que l'url est correcte (en particulier le tiret après univ) et les ''.

Dans certains cas cela ne suffira pas, il faudra alors utiliser des outils vous permettant de remplacer les liens dans toutes les tables de *WordPress*.

2. Copiez les fichiers ou le dossier wordpress sur adartza en utilisant un client FTP ([ftp.exe](#) ou filezilla par exemple).

Pour vous connecter via FTP à Adartza :

**Hôte** : adartza.iutbayonne.univ-pau.fr  
**Identifiant** : votre identifiant habituel  
**Mot de passe** : votre mot de passe habituel  
**Port** : laisser le champ vide (port par défaut)  
**Dossier web** : WWW

Puis copier le dossier wordpress d'iparla dans le dossier WWW d'adartza en cohérence avec les adresses modifiées à la question précédente (dans filezilla, il suffit d'un copier coller).

Si vous ne pouvez pas vous connecter à Adartza, il faut spécifier le proxy à utiliser à l'IUT : dans filezilla, menu Edition Paramètres, cliquez sur proxy générique et choisir :

- le Type de proxy générique : http.1.1 utilisant la méthode CONNECT
- Hôte proxy : cache
- Port du Proxy 222

3. Testez votre site <http://adartza.iutbayonne.univ-pau.fr/~votrelogin/wordpress>

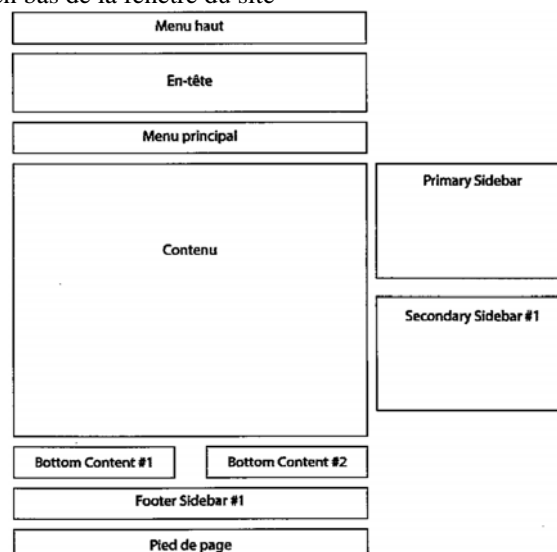
## Navigation dans un site sur WordPress

### 22. Création de la barre de navigation du haut

4. Dans le menu Apparence Menus, choisissez les options de l'écran suivantes : Emplacements Liens personnalisés Pages Catégories.
5. Créez un nouveau menu dont le nom sera « Barre de navigation du haut »
6. Dans la partie Réglages du menu puis Afficher l'emplacement choisir Top Menu de manière à ce que le menu soit inséré au-dessus du bandeau du site. Définissez la structure du menu en ajoutant des pages à partir du bandeau de gauche. Enregistrez le menu et visualisez le résultat.
7. Ajoutez une page de premier niveau grâce au menu Pages du tableau de bord. Visualisez le menu créé à la question précédente. Concluez.
8. Revenez au menu Apparence Menus et modifier les réglages du menu Barre de navigation du haut en cochant Ajouter automatiquement les pages de premier niveau à ce niveau. Créez une nouvelle page de premier niveau et visualisez le résultat sur le site.
9. Cliquez sur l'onglet Liens personnalisés. Ajoutez au menu un lien vers la page d'accueil de votre site et indiquez Accueil dans la Cible du lien.
10. A partir du menu Pages, rajoutez au menu les pages que vous souhaitez.
11. Modifiez l'ordre des éléments dans votre menu grâce à des cliquer-glissr.
12. Enregistrez le menu et visualisez le résultat.

### 23. La navigation latérale

13. Dans le menu Arras, vérifiez que le paramètre Arras Options – Design – Overall Design- Overall layout soit sur 2 Column Layout (Right Sidebar) qui est le choix par défaut.
14. Créez un nouveau menu à partir de Apparence Menus.
15. Dans le menu Catégories – Afficher tout, ajoutez au menu les catégories désirées.
16. Dans le menu Apparences – Widgets, observez ce que vont contenir les 5 emplacements prédéfinis dans le thème Arras :
  - Primary Sidebar : en haut de la colonne latérale
  - Secondary Sidebar #1 : en-dessous de Primary Sidebar
  - Bottom Content #1 : sous le contenu principal de la page du site
  - Bottom Content #2 : sous le contenu principal de la page du site
  - Footer Sidebar #1 : en bas de la fenêtre du site



Un widget dans WordPress est un composant d'interface graphique, un module remplissant une seule fonction, un bloc de code indépendant que l'on place dans une barre de navigation.

17. Dans la partie Widgets disponibles, cliquez sur le widget Menu de navigation et créez un menu dans l'emplacement Primary Sidebar. Choisissez un nom et sélectionnez le menu créé à la question 14. Fermez l'onglet et visualisez le résultat.
18. Créez un second menu personnalisé avec d'autres catégories que vous hiérarchiserez grâce au cliqué-glissé. Modifiez le libellé des éléments du menu en cliquant sur le triangle à droite des titres. Observez que vous retrouvez dans Original le nom initial de la catégorie.
19. Ajoutez ce menu dans la Secondary Sidebar#1 et testez-le. Observez le commentaire dans Apparence Personnaliser Widgets. Concluez.

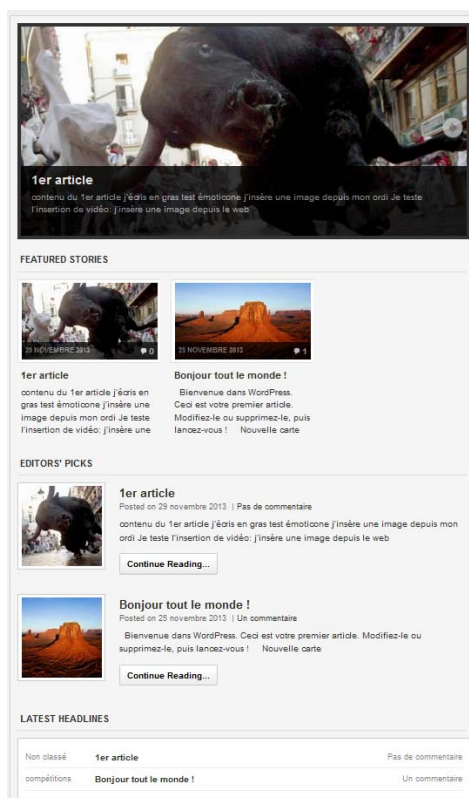
## 24. Le texte personnalisé en HTML

20. Dans le menu Apparences – Widgets, placez le widget Texte dans l'emplacement Bottom Content#1.
21. Dans la partie texte, tapez un code html décrivant les informations habituelles d'un site (Site officiel de l'élevage des lamas, Directeur de la publication...).
22. Créez une page contenant les mentions légales et insérez un lien vers celle-ci dans votre code html.
23. Enregistrez, fermez et testez.

## 25. Les articles de la page d'accueil

Dans le thème Arras, la page d'accueil est structurée en différentes zones d'affichage :

- Diaporama - SlideShow: photo d'illustration (Image à la une), titre et extrait de chaque article
- Contenus à la une #1 – Featured stories : affichage en colonne avec une photo, la date de publication, le titre et l'extrait de chaque article
- Editors'pick : affichage en ligne des mêmes éléments que Contenus à la une
- Latest headlines - Options des actualités : liste chronologique des titres des articles



Diaporama – Slideshow

Contenus à la une #1 – Featured stories

Editors'pick

Latest headlines - Options des actualités

23. Créez 4 catégories d'articles qui correspondront aux 4 modes d'affichages : En ce moment (diaporama), Actualités (contenus à la une), Remarqués (Editors' picks), Brèves (Options des actualités).

24. Faites en sorte que quelques articles se rattachent à chacune de ces catégories (créez les articles ou modifiez leur catégorie).
25. Dans le menu Arras – Post Types &Tax. – Post types, dans la zone Home (page d'accueil), vérifiez que chaque élément soit associé à Articles.
26. Dans l'onglet Taxonomies, vérifiez que chaque élément soit associé à Catégories.
27. Dans le menu Arras – Theme Options – Home, paramétrez les catégories et le nombre d'articles voulus pour chaque zone graphique de la page d'accueil. Renommez ces zones en français.
28. Modifiez le mode d'affichage (Display Type) d'un élément et testez les possibilités.

### 29. Le fil d'Ariane

Il s'agit pour l'internaute de savoir où il se trouve dans le site. Pour cela vous pouvez utiliser un plugin ou suivre la démarche suivante

30. Dans le fichier functions.php du répertoire wordpress/wp-content/themes/arras, rajouter le code suivant qui permet d'insérer un fil d'Ariane dans vos pages. (Vous pouvez également accéder à ce fichier depuis le menu Apparence – Editeur en choisissant functions.php (Fonctions du thème))

```
// Fonction qui permet d'ajouter un fil d'Ariane
function GkAriane()
{
    if (!is_home()) {
        echo '<a href="';
        echo get_option('home');
        echo '>';
        bloginfo('name');
        echo "</a>>>";
        if (is_category() || is_single()) {
            the_category('-');
            if (is_single()) {
                echo ">>>";
                the_title();
            }
        }
    }
}
```

31. Rajoutez le code suivant à l'endroit où vous voulez insérer le fil d'Ariane par exemple dans le fichier header.php en ligne 97 :

```
<?php GkAriane(); ?>
```

### 32. La gestion des articles avec les mots clés

33. Dans le menu, Apparence – Widget, utilisez le widget Nuage de mots clés pour le faire apparaître dans votre site.

Utilisez des plugin pour améliorer les outils de recherche de manière à ce que les mots recherchés dans un article soient mis en évidence, pour afficher la liste des articles les plus lus, pour afficher une page d'avertissement en cas de maintenance et d'arrêt du site...

## Gestion des utilisateurs

24. Dans le menu Utilisateurs, visualisez Tous les utilisateurs. Vérifiez que vous êtes présent en tant qu'administrateur.

Les rôles disponibles dans WordPress sont les suivants :

- a. Administrateur : tous les droits
- b. Editeur :
  - i. création, modification, enregistrement et publication des articles et des pages (même s'il ne les a pas créés)
  - ii. pas d'accès à la gestion des extensions ni aux réglages du site
- c. Auteur :
  - i. écriture, enregistrement et publication de ses propres articles
  - ii. Pas d'accès aux pages ni aux liens
- d. Contributeur :
  - i. écriture et enregistrement de ses propres articles
  - ii. pas d'accès à la publication des articles
  - iii. pas d'insertion de média et donc d'images
- e. Abonné : utilisateur référencé sur le site mais sans aucun droit particulier

25. Ajoutez un utilisateur en tant qu'abonné.

26. Dans le menu Utilisateurs, dans Tous les utilisateurs, vérifiez que cet utilisateur a bien été créé. Survolez son nom à la souris et choisissez modifier. Modifiez les couleurs de l'interface, le rôle et le mot de passe puis mettez à jour.

27. Connectez-vous au site en tant que cet utilisateur. Vérifiez votre paramétrage et modifiez ce profil.

28. Dans le menu Réglages – Général, autorisez tout le monde à s'enregistrer en tant qu'abonné.

29. Déconnectez-vous et testez l'inscription à partir de l'écran de connexion.

30. Installez l'extension Members.

31. Créez un nouveau rôle et créez des utilisateurs qui auront ce rôle.

## Thèmes dans WordPress

### 34. Composition des thèmes

Un thème est stocké dans un répertoire nominatif du type : wp-content/themes/nom-du-thème. Il est constitué :

- a. des fichiers de modèles (templates files) permettant l'affichage des différentes parties du site, certains sont utilisés pour toutes les pages d'autres sont plus spécifiques. Ce sont des fichiers php appelé par WordPress pour générer du html.
- b. des fichiers css pour la mise en forme : style.css contient les informations du thème
- c. d'un fichier screenshot.png page du thème qui apparaît dans l'admin
- d. d'un fichier functions.php permettant de proposer différentes options d'administration et de rajouter des fonctionnalités.

| Page appelée   | Template utilisé             |
|--|------------------------------|
| Accueil  | home.php                     |
| Liste des articles selon un critère (annuelle, mensuelle ou journalière) | archive.php<br>ex : date.php |
| Les articles d'une catégorie   | category.php                 |
| Les articles d'un mot-clef   | tag.php                      |
| Résultat de recherche  | search.php                   |
| Article  | single.php                   |
| Page   | page.php                     |
| Page recherchée n'existant pas.  | 404.php                      |

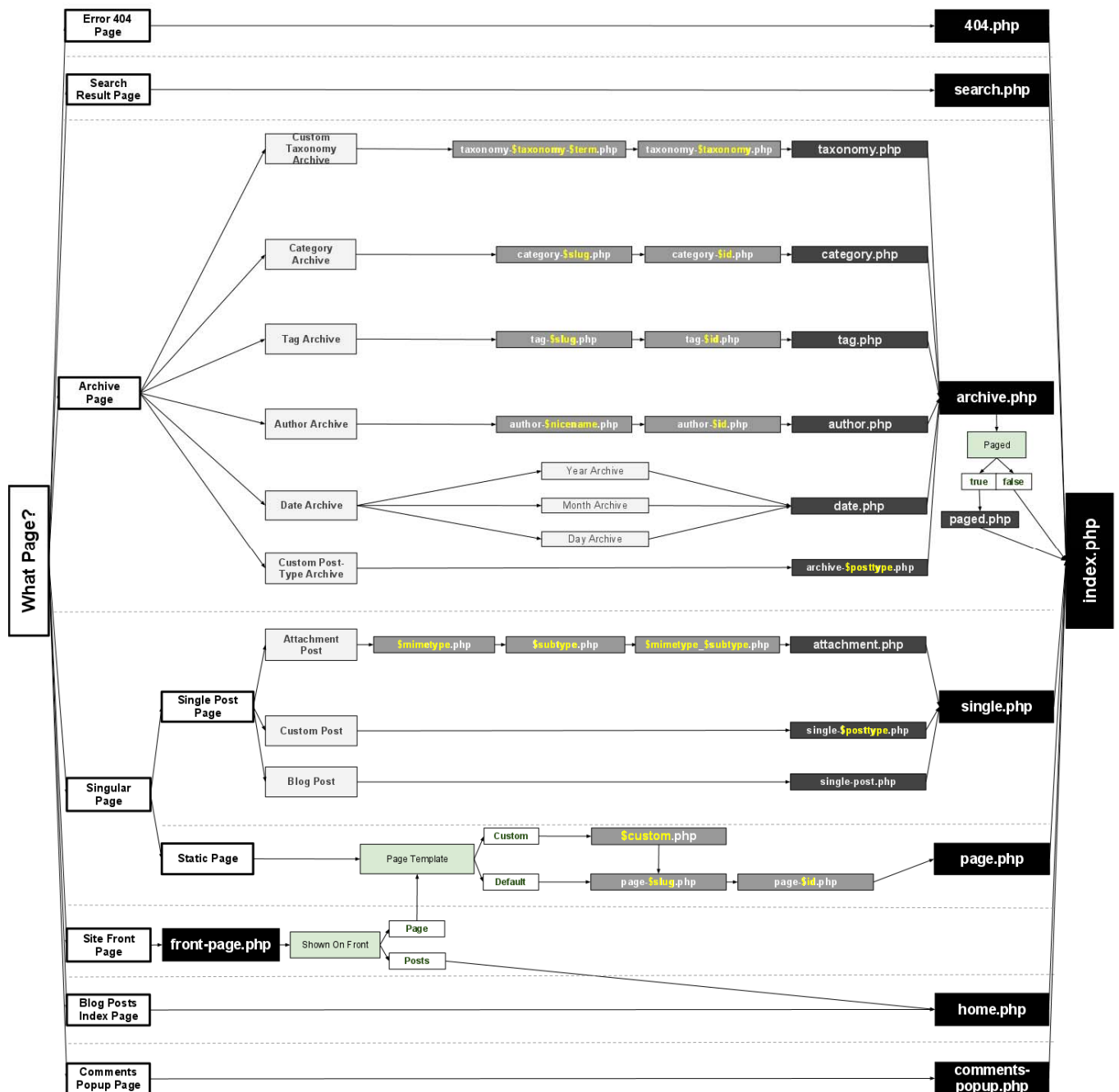
### 35. Définition du template à utiliser pour une page

En fonction de la page demandée, WordPress teste l'existence de template de moins en moins spécifique jusqu'à ce que l'un d'eux existe. C'est celui-là qui est alors utilisé. Le schéma page suivante définit le chemin que WordPress emprunte pour choisir le template à utiliser ([http://codex.wordpress.org/Template\\_Hierarchy](http://codex.wordpress.org/Template_Hierarchy)).

Par exemple, l'internaute veut afficher les archives d'Avril 2010

1. WordPress rentre dans la boucle au niveau de Archive Page car il s'agit d'une page **d'archive**.
2. Ensuite il bifurque vers Date Archive car il s'agit d'une archive **datée**.
3. Ici on a écrit /2010/04. Il s'agit donc d'une archive **mensuelle**.
4. WordPress tombe alors sur date.php et l'utilisera s'il existe.
5. Si date.php n'existe pas, WordPress utilisera le template archive.php.
6. Si ce dernier n'existe pas non plus, il utilisera index.php.

Le fichier index.php est donc le dernier recours : il est indispensable. Il est parfois utilisé comme page d'accueil de manière à ce que cette page soit affichée si le template recherché n'existe pas.



### 36. Includes

Ces fichiers décrivent des éléments répétitifs qui seront inclus dans différentes pages complètes. Les includes principalement utilisés sont :

- header.php : le début du code, avec la doctype, le <head>, le début du <body>
- footer.php : la fin du code, avec le pied de page et la fermeture des balises </body> et </html>
- sidebar.php : une colonne qui contient souvent le formulaire de recherche, un bouton RSS, la liste des catégories, un nuage de tags...
- searchform.php : le formulaire de recherche, tout seul
- comments.php : le module de commentaires, présent dans single.php
- loop.php : la boucle utilisée pour afficher les Posts



### 37. Modèles spécifiques

Les thèmes utilisent d'autres fichiers de modèles spécifiques.

32. Identifiez les modèles spécifiques utilisés par le thème Arras : dans la partie administration, ouvrez une page et faites apparaître les modèles disponibles dans la partie Attributs de la page.
33. Dans le répertoire wordpress/wp-content/themes/arras, identifiez les fichiers définissant ces modèles en fonction de leur nom et regardez le contenu de l'un d'entre eux.

### 38. Marqueurs de modèles

Les marqueurs de modèles (template tags) sont des instructions PHP placées dans des éléments HTML formatés avec des styles css. Ils appellent un fichier PHP et l'insère là où se fait l'appel.

34. Ouvrez le fichier index.php du thème Arras. Qu'en concluez-vous ?
35. Ouvrez le fichier home.php du thème Arras et constatez que la première ligne est la suivante :

```
<?php get_header(); ?>
```

Ce marqueur appelle le fichier header.php et affiche tout son contenu en début de fichier et donc en début d'affichage dans la fenêtre du navigateur.

36. Ouvrez le fichier header.php et observez sa constitution.
37. Ouvrez le fichier index.php du thème TwentyTwelve. Observez la présence de la même ligne de code que dans home.php. Observez ensuite la présence de la ligne :

```
<div id="primary" class="site-content">
```

Les deux boîtes imbriquées <div>, id="primary" et id="content", affichent le contenu rédactionnel du site.

38. Dans content, trouvez l'appel à la boucle WordPress :

```
get_template_part( 'content', get_post_format() )
```

39. Ouvrez le fichier content.php pour comprendre ce que fait cette ligne de code.
40. Revenez au fichier index.php et observez les deux dernières lignes de code qui permettent d'afficher la colonne latérale droite et le pied de page. Identifiez les fichiers appelés pour réaliser cela et vérifiez leur présence dans le répertoire du thème.
41. Ouvrez le fichier header.php et observez qu'il commence par la structure de base d'un fichier html (doctype head body). Parcourez le fichier.
42. Ouvrez le fichier footer.php et observez qu'il contient les balises de fermeture du body et du html.
43. Parcourez la page [http://codex.wordpress.org/Template\\_Tags](http://codex.wordpress.org/Template_Tags) pour identifier les différents marqueurs de modèles disponibles.
44. Visualisez les informations sur bloginfo().

### 39. Création d'un thème basique

#### Etape 1: Définir la structure

Réalisez la maquette du site. Ici on se contentera de définir les fichiers header.php, index.php, sidebar.php et footer.php et la feuille de style style.css.

#### Etape 2: Le dossier du thème

Créez un dossier reprenant le nom du thème sans espace, ni caractères accentués, ni spéciaux, ni majuscules et placez-le dans /wp-content/themes/.

Par la suite, vous créerez un fichier screenshot.png qui contiendra une vignette de la page d'accueil aux dimensions imposées de 300x240 pixels (en utilisant par exemple GIMP). Vous le placerez à la racine du dossier de votre thème. Ainsi vous aurez un aperçu du thème dans l'administration du site, Apparence – Thèmes.

#### Etape 3: Le fichier header.php

Il contient l'en-tête html du site, l'en-tête <head>, le titre du site, le titre et le slogan du blog, le lien vers la feuille de style et l'insertion de la barre de navigation.

```
<!DOCTYPE html>      <!--en-tête html-->
<html>

<head>
    <meta charset="<?php bloginfo('charset') ?>" />
    <title>                                <!--titre du site-->
    <?php bloginfo('name') ?> -            <!--titre du blog-->
    <?php bloginfo('description') ?>      <!--slogan du blog-->
</title>
    <link rel="stylesheet" type="text/css" media="screen" href="<?php bloginfo('stylesheet_url') ?>" />
    <!--lien vers la feuille de style-->
</head>

<body>
    <div id="header">
        <h1><?php bloginfo('name') ?></h1>      <!--titre du blog-->
        <h3><?php bloginfo('description') ?></h3> <!--slogan du blog-->
    </div>
    <div id="navBar ">
        <?php wp_nav_menu(); ?>                <!--insertion de la barre de navigation-->
    </div>
```

#### Etape 4: Le fichier index.php

```
<?php get_header() ; ?>                                <!--appel du header.php -->
<div id="main">
    <div id="content">                                    <!--affichage des articles-->
        <?php if (have_posts()) : ?>                    <!--boucle WordPress simplifiée-->
            <?php while (have_posts()) : the_post(); ?>
                <h3><?php the_title(); ?></h3>          <!--slogan du blog-->
                <p>Publié le <?php the_date(); ?>
                    et rédigé par <?php the_author(); ?>
                </p>
                <p> <?php the_content(); ?> </p>
                <p> <?php the_category(); ?> <?php the_tags(); ?> </p>
            <?php endwhile; ?>
        <?php endif; ?>
```



```

    </div>
    <?php get_sidebar() ; ?>                                <!--affichage de la barre latérale droite-->
    <?php get_footer() ; ?>                                <!--affichage du pied de page-->

```

#### Etape 5: Le fichier sidebar.php

```

<div id="primary">                                <!--emplacement prédéfini pour les widgets-->
    <?php get_search_form(); ?>                    <!--insertion du formulaire de recherche-->
    <?php wp_get_archives('type=monthly'); ?>      <!--articles triés par mois-->
    <?php wp_register(); ?>                        <!--lien pour se connecter à l'interface administration-->
    <?php wp_loginout(); ?>                        <!--lien pour se déconnecter-->
</div>

```

#### Etape 6: Le fichier footer.php

```

    </div>                                <!--fin de la boite main ouverte dans index.php-->
    <div id="footer">                        <!--pied de page -->
        <p><?php print(date(Y)); ?>          <!--année en cours -->
        - &copy ; <?php bloginfo('name') ?> - <!--copyright avec nom du blog -->
        Conception et réalisation : Martin Dupont <!--texte sur le concepteur -->
    </p>
    </div>
</body>
</html>

```

#### Etape 7: Le fichier style.css

```

/*
Theme Name : Theme1
Theme URI : http://monsite.net
Description : Premier essai de conception d'un thème
Version : 1.0
Author : Martin Dupont
Author URI : http://www.monsite.net
*/

body {
    margin:0;
    padding:0;
}

#header {
    width: 790px;
    margin:0 auto;
    padding: 5px;
    border-bottom: solid 1px #000;
}

#navBar {
    width: 790px;
    margin:0 auto;
    padding: 5px;
    border-bottom: solid 1px #000;
}

```

```
#main {  
    width: 800px;  
    margin:0 auto;  
    overflow:auto;  
}  
  
#content {  
    width: 590px;  
    padding: 5px;  
    float: left;  
}  
  
#primary {  
    width: 190px;  
    padding: 5px;  
    float: left;  
}  
  
#footer {  
    width: 790px;  
    margin:0 auto;  
    clear: both;  
    padding: 5px;  
    border-top: solid 1px #000;  
}
```

Il ne vous reste plus qu'à personnaliser et tester ces fichiers.