







Titre Professionnel « Développeur Web et Web Mobile »

Documentation ECF Wordpress

SUJET

Élaboration d'un blog personnel.

Les apprenants doivent élaborer un blog personnel :

Contenu attendu:

- Une page statique en accueil contenant : une présentation, une image, les liens vers le pote-folio et le cv, les liens vers les réseaux sociaux
- Une page de blog, commentaires activés pour les membres, soumis à validation de l'administrateur, avec un menu par date et un menu des mots clefs.
- Le footer contient, en plus des mentions légales, un formulaire de contact programmé dans un plugin avec des paramètres administrables.
- Mettre en place un système de newsletter pour les membres (plugins autorisé).
- Mettre en place un rôle de « correcteur » qui pourra corriges les articles de blogs, parus ou en brouillon, et les commentaires sans avoir les droits pour modifier les statuts des articles/commentaires (plugins autorisé).
- Le front-end doit utiliser un thème personnalisé, squelette autorisé mais pas de thème enfant.

Le processus de création doit être documenté le code commenté et respecter les règles de sécurité.

TABLE DES MATIERES

Suj	jet		1
1.	Conc	eption	3
		Vireframe	
2.	Sécur	ité	4
		Captcha	
,		Hacces	
3.	. Plugins		
	3.1.	User Role Editor : Gestion simple des rôles utilisateurs	
	3.2.	WPForms Lite : Formulaire de contact et Newsletter	
4.	Paaes	S	
		Activations	
		Chargement des scripts	
		Déclaration des menus	
		Widgets	
		Filtres pour templates	
		Pages	

1. CONCEPTION

1.1. WIREFRAME

Wireframe Wordpress Eval

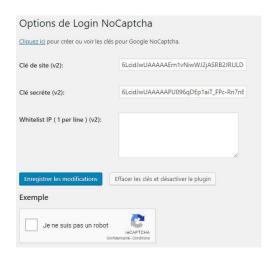


2. SECURITE

2.1. CAPTCHA

Ajout d'un plugin captcha pour sécuriser l'accès





2.2. HTACCES

Ajout d'un fichier captcha à la racine du site pour renforcer la sécurité

```
# Désactiver l'affichage du contenu des répertoires

Options All -Indexes

# Masquer les informations du serveur

ServerSignature Off

# Activation du suivi des liens symboliques

Options +FollowSymLinks

# Choix du fuseau horaire

SetEnv TZ Europe/Paris

# Encodage par défaut des fichiers textes et HTML

AddDefaultCharset UTF-8

# Protéger le fichier wp-config.php

cfiles wp-config.php>
order allow,deny

deny from all

</files>

# Protéger les fichiers .htaccess et .htpasswds

<Files ~ "^.*\.([Hh][Tt][AaPp])">
order allow,deny

deny from all

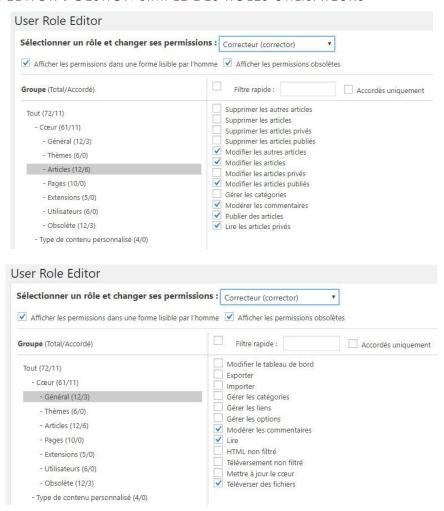
satisfy all

</files>
```

3. PLUGINS

- Cookie notice pour le RGPD
- Font Awesome pour logo Réseaux Sociaux

3.1. USER ROLE EDITOR: GESTION SIMPLE DES ROLES UTILISATEURS



3.2. WPFORMS LITE: FORMULAIRE DE CONTACT ET NEWSLETTER



4. PAGES

- Mentions légales
- Politique de confidentialité (RGPD)

5. CODE

```
5.1. FUNCTIONS.PHP
```

5.1.1. ACTIVATIONS

```
// Activation des images à la une
add_theme_support( 'post-thumbnails' );
```

```
// Activation tag des titres balise <title>
add_theme_support( 'title-tag' );
```

5.1.2. CHARGEMENT DES SCRIPTS

```
// Chargement des scripts
function sicca_theme_scripts() {
    wp_deregister_script( 'jquery' );
    wp_enqueue_script( 'jquery', 'https://ajax.googleapis.com/ajax/libs/jquery/3.4.0/jquery.min.js');
    // Pour Bootstrap qui dépend de JQuery
    wp_enqueue_script( 'bootstrapjs', 'https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js', array(
'jquery' ), false, false );
    wp_enqueue_style( 'bootstrapcss', 'https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css');
    wp_enqueue_style( 'css', get_theme_file_uri( '/css/style.css' ));

    // Possibilité de répondre à un commentaire
    if ( is_singular() && comments_open() && get_option( 'thread_comments' ) ) {
        wp_enqueue_script( 'comment-reply' );
    }
}
add_action( 'wp_enqueue_scripts', 'sicca_theme_scripts' );
```

5.1.3. DECLARATION DES MENUS

```
// Déclarer ses menus
register_nav_menus(
    array(
```

```
'primaire' => __('menu primaire', 'test')
);
```

5.1.4. WIDGETS

```
function widget_sicca() {
    * Création de Sidebar
    * @param string|array Construit une barre latérale basée sur les valeurs'name' et'id'.
    $args = array(
                       => __( 'Sidebar de droite', 'theme_text_domain'),
       'name'
                       => 'sidebar_droite',
       'description' => '',
       'before_widget' => '<span id="%1" class="widget %2">',
       'after_widget' => '</span>',
       'before_title' => '<h4 class="widgettitle">',
    register_sidebar( $args );
    $args = array(
                       => __( 'Sidebar pour le footerfooter', 'theme_text_domain'),
       'before_widget' => '<span id="%1" class="widget %2">',
       'after_widget' => '</span>',
       'before_title' => '<h4 class="widgettitle">',
       'after_title' => '</h4>',
    register_sidebar( $args );
add_action( 'widgets_init', 'widget_sicca' );
```

5.1.5. FILTRES POUR TEMPLATES

```
/* Filtre pour template perso en fonction de l'id d'article, de l'auteur, de la catégorie ou du tag
qui scan les fichiers tel single-75.php, single-author-sicca.php, single-cat-portfolio.php...
*/
function my_single_template()
{
```

```
global $wp_query, $post;
    if(file_exists(TEMPLATEPATH.'/single-' . $post->ID . '.php'))
        return TEMPLATEPATH.'/single-' . $post->ID . '.php';
    $curauth = get_userdata($wp_query->post->post_author);
    if(file_exists(TEMPLATEPATH . '/single-author-' . $curauth->user_nicename . '.php'))
        return TEMPLATEPATH . '/single-author-' . $curauth->user_nicename . '.php';
    elseif(file_exists(TEMPLATEPATH . '/single-author-' . $curauth->ID . '.php'))
        return TEMPLATEPATH . '/single-author-' . $curauth->ID . '.php';
    foreach((array)get_the_category() as $cat){
        if(file_exists(TEMPLATEPATH . '/single-cat-' . $cat->slug . '.php'))
            return TEMPLATEPATH . '/single-cat-' . $cat->slug . '.php';
       elseif(file_exists(TEMPLATEPATH . '/single-cat-' . $cat->term_id . '.php'))
            return TEMPLATEPATH . '/single-cat-' . $cat->term_id . '.php';
    $wp_query->in_the_loop = true;
    foreach((array)get_the_tags() as $tag){
        if(file_exists(SINGLE_PATH . '/single-tag-' . $tag->slug . '.php'))
            return SINGLE_PATH . '/single-tag-' . $tag->slug . '.php';
       elseif(file_exists(SINGLE_PATH . '/single-tag-' . $tag->term_id . '.php'))
            return SINGLE_PATH . '/single-tag-' . $tag->term_id . '.php';
    $wp_query->in_the_loop = false;
    if(file_exists(TEMPLATEPATH . '/single.php'))
       return TEMPLATEPATH . '/single.php';
       return $single;
add_filter( "single_template", "my_single_template" );
```

Ensuite il suffit de créer une page single.php ou avec son id comme single-75.php pour créer un template spécial pour ce type d'article.

5.1.6. PAGES

Template des pages simple :

Et le contenu de l'article « contenu-article.php

```
// Récupération de l'ID de l'image à la une puis affection de la variable
$idImageAlaUne=get_post_thumbnail_id();
$imgSrc=wp_get_attachment_url( $idImageAlaUne );
?>
<!-- class responsive et width à 100 % -->
<img class="img-responsive" src="<?php echo $imgSrc; ?>" alt="" style="width:100%">
<?php
// autre méthode pour afficher thumbnail
// the_post_thumbnail('large')
?>
<h1><?php the_title();?></h1>
<h4><?php the_date();?></h4>
```