

WordPress 5

Créer un plugin WordPress

Les extensions – ou plugins en anglais – sont des fonctionnalités que l'on ajoute à un site WordPress pour réaliser une application web.



Table des matières

Les plugins WordPress

Pourquoi créer un plugin WordPress

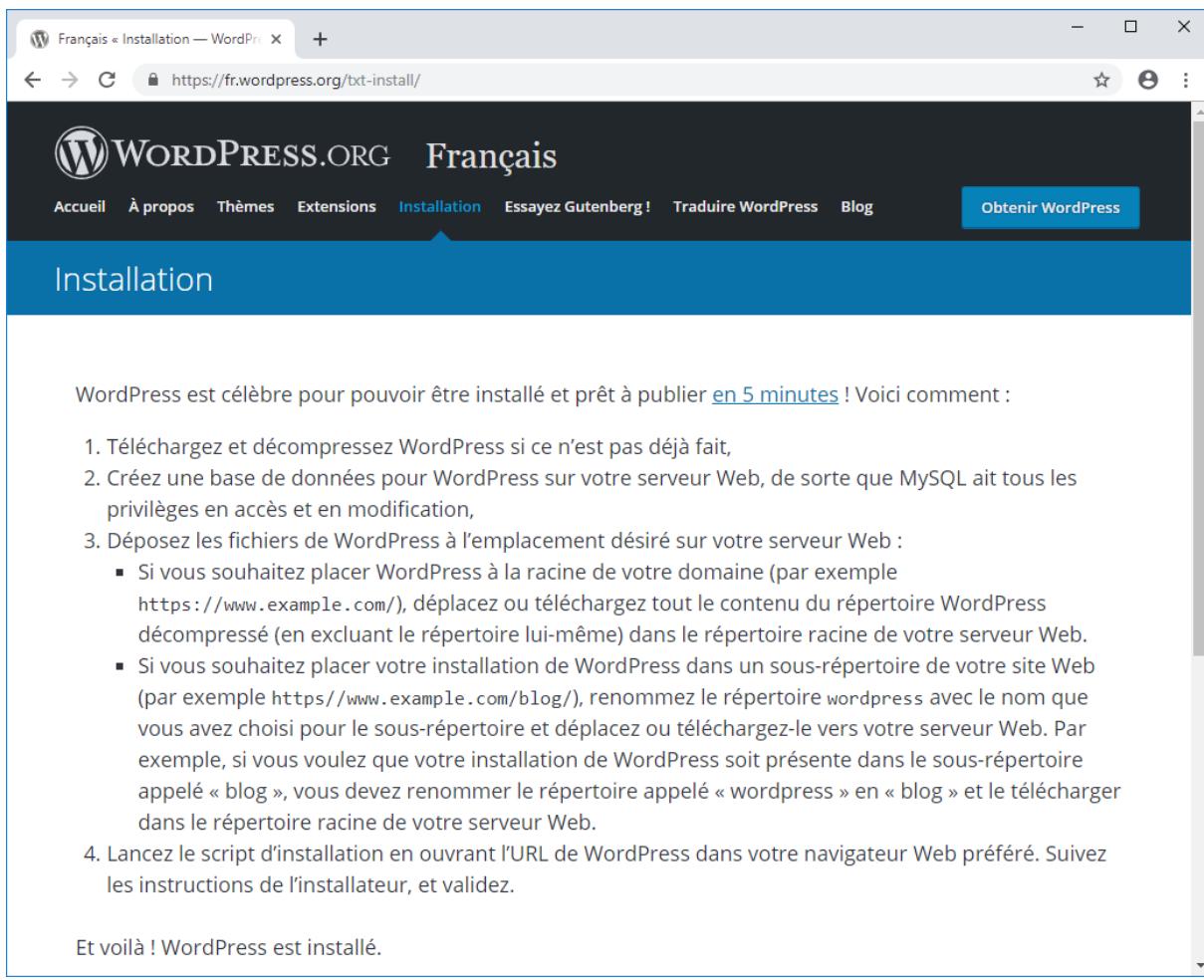
Les différentes raisons

Etude d'un plugin

Le plugin Hello Dolly

Création de notre premier plugin

Installation de WordPress



The screenshot shows a web browser window displaying the WordPress.org installation guide. The URL in the address bar is <https://fr.wordpress.org/txt-install/>. The page has a dark header with the WordPress logo and the text "WORDPRESS.ORG Français". Below the header, there's a navigation menu with links like "Accueil", "À propos", "Thèmes", "Extensions", "Installation" (which is highlighted), "Essayez Gutenberg !", "Traduire WordPress", and "Blog". A blue button labeled "Obtenir WordPress" is also visible. The main content area has a blue header bar with the word "Installation". The text on the page explains that WordPress can be installed in 5 minutes and provides step-by-step instructions for doing so.

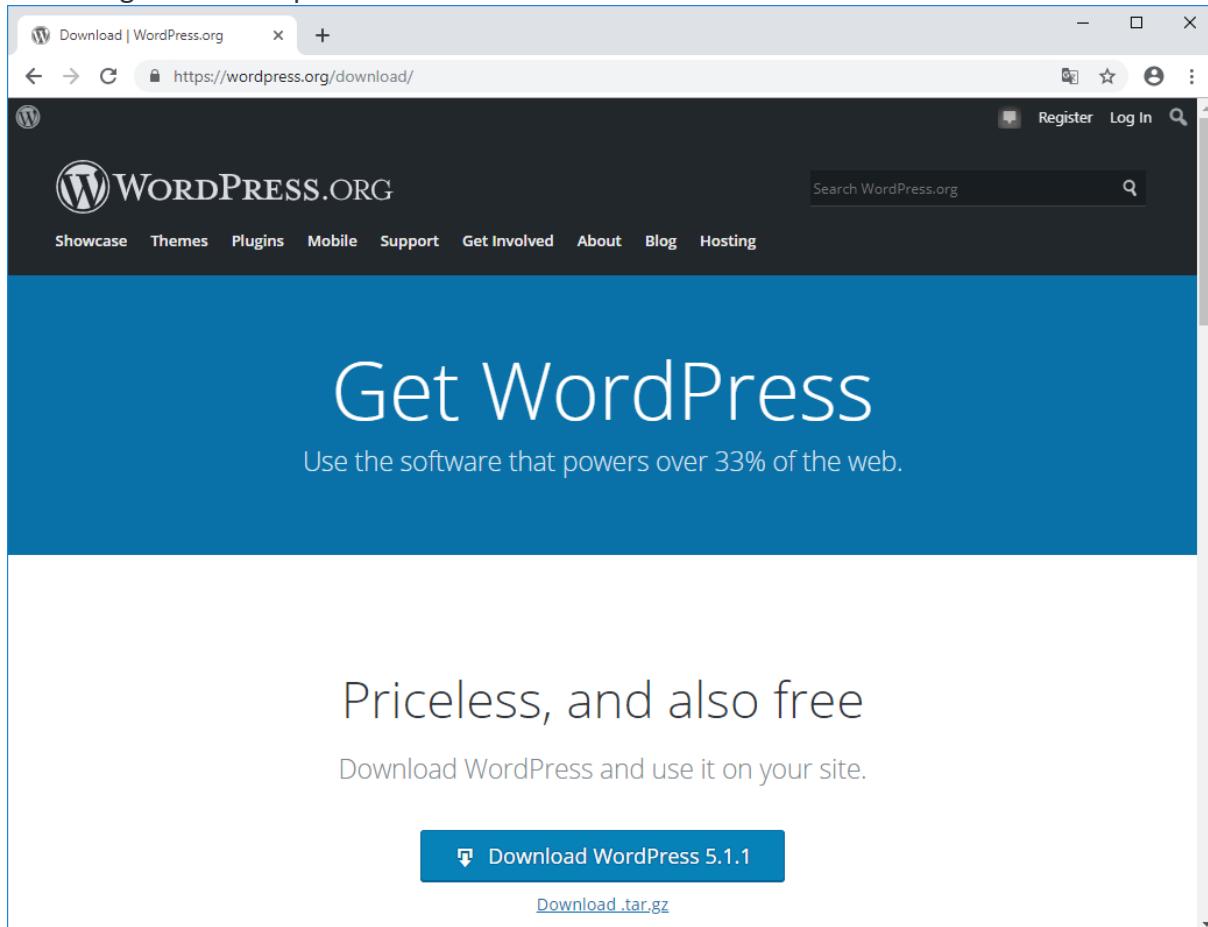
WordPress est célèbre pour pouvoir être installé et prêt à publier [en 5 minutes](#) ! Voici comment :

1. Téléchargez et décompressez WordPress si ce n'est pas déjà fait,
2. Créez une base de données pour WordPress sur votre serveur Web, de sorte que MySQL ait tous les priviléges en accès et en modification,
3. Déposez les fichiers de WordPress à l'emplacement désiré sur votre serveur Web :
 - Si vous souhaitez placer WordPress à la racine de votre domaine (par exemple <https://www.example.com/>), déplacez ou téléchargez tout le contenu du répertoire WordPress décompressé (en excluant le répertoire lui-même) dans le répertoire racine de votre serveur Web.
 - Si vous souhaitez placer votre installation de WordPress dans un sous-répertoire de votre site Web (par exemple <https://www.example.com/blog/>), renommez le répertoire `wordpress` avec le nom que vous avez choisi pour le sous-répertoire et déplacez ou téléchargez-le vers votre serveur Web. Par exemple, si vous voulez que votre installation de WordPress soit présente dans le sous-répertoire appelé « blog », vous devez renommer le répertoire appelé « `wordpress` » en « `blog` » et le télécharger dans le répertoire racine de votre serveur Web.
4. Lancez le script d'installation en ouvrant l'URL de WordPress dans votre navigateur Web préféré. Suivez les instructions de l'installateur, et validez.

Et voilà ! WordPress est installé.

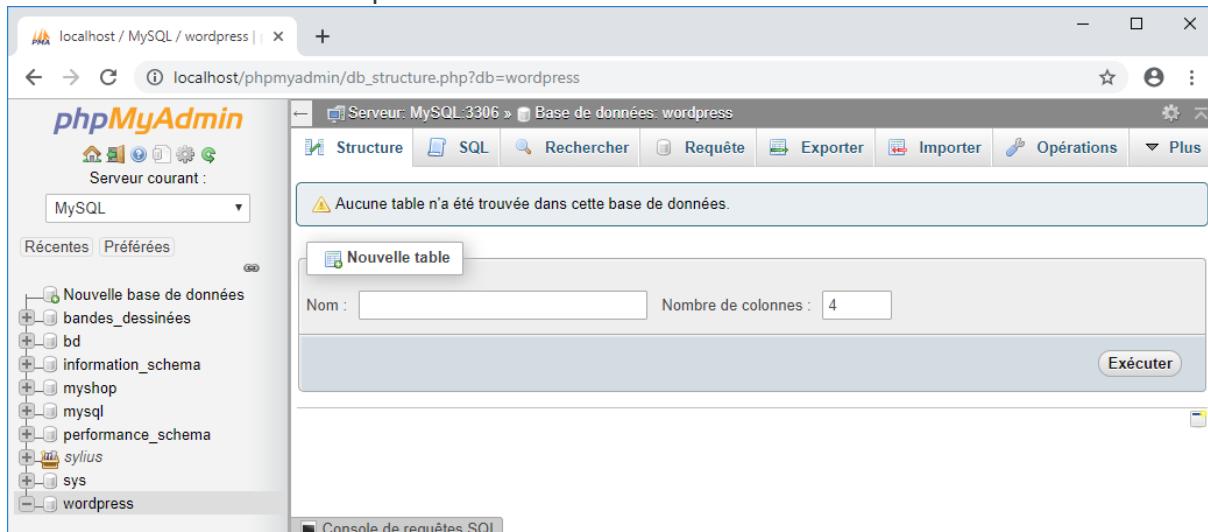
Cf. : <https://fr.wordpress.org/txt-install/>

Téléchargez et décompressez WordPress



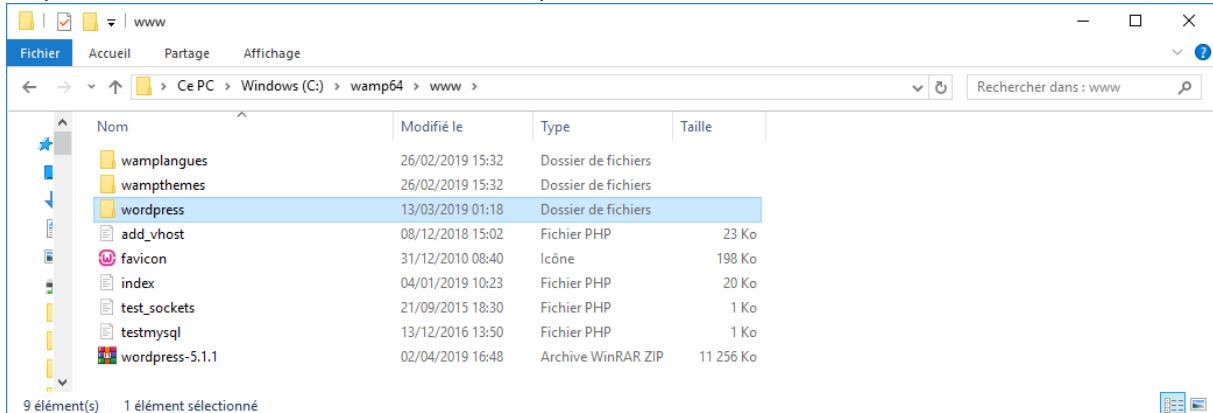
The screenshot shows the official WordPress download page at <https://wordpress.org/download/>. The page features a large blue header with the text "Get WordPress" and a subtext "Use the software that powers over 33% of the web.". Below this, a section titled "Priceless, and also free" encourages users to "Download WordPress and use it on your site." Two download buttons are visible: a large blue button labeled "Download WordPress 5.1.1" and a smaller link "Download .tar.gz". The top navigation bar includes links for Showcase, Themes, Plugins, Mobile, Support, Get Involved, About, Blog, and Hosting.

Créez une base de données pour WordPress

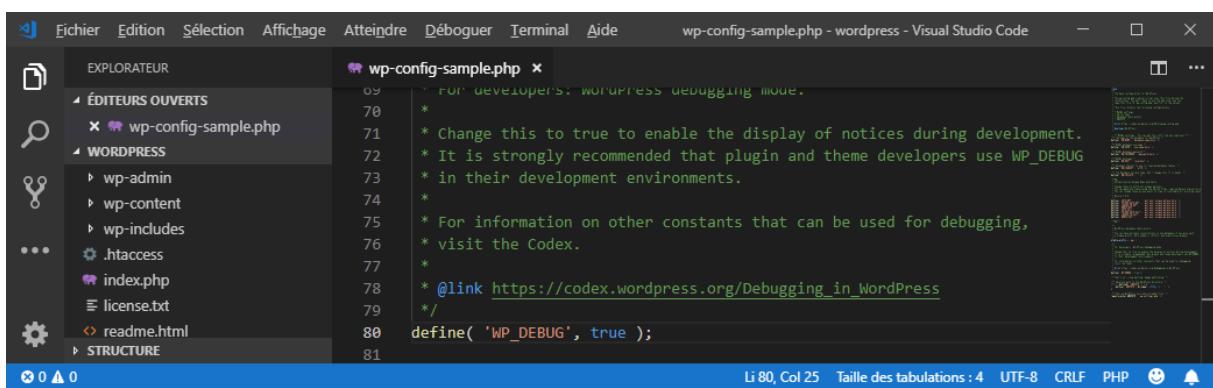
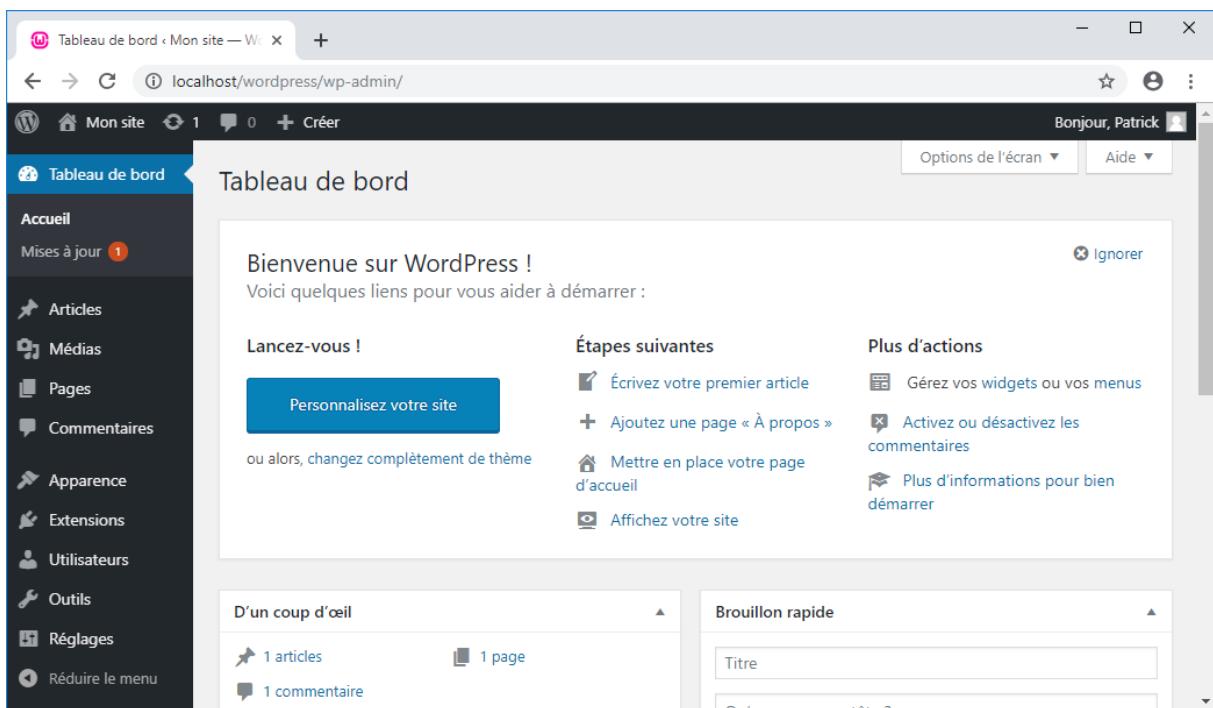


The screenshot shows the phpMyAdmin interface for managing MySQL databases. The left sidebar lists several databases: Nouvelle base de données, bandes_dessinées, bd, information_schema, myshop, mysql, performance_schema, sylis, sys, and wordpress. The main panel shows the "Structure" tab for the "wordpress" database. A message indicates "Aucune table n'a été trouvée dans cette base de données." (No table was found in this database). A form for creating a new table is displayed, with fields for "Nom" (Name) and "Nombre de colonnes" (Number of columns), both currently set to 4. An "Exécuter" (Execute) button is visible at the bottom right of the form area.

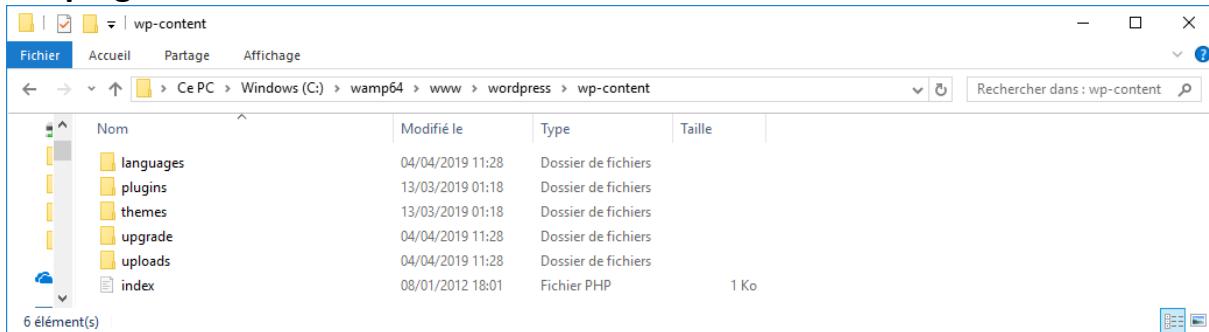
Déposez les fichiers de WordPress à l'emplacement désiré sur votre serveur Web



Lancez le script d'installation en ouvrant l'URL de WordPress dans votre navigateur Web



Les plugins WordPress



Pourquoi créer un plugin WordPress

Les différentes raisons

<https://wpformation.com/creer-plugin-wordpress/>

Etude d'un plugin

Le plugin Hello Dolly

Le fameux plugin "Hello Dolly" est embarqué à la suite de l'installation WordPress.

wp-includes/plugins/hello.php

Le plugin est composé de :

1. Un fichier PHP : nom.php, dans ce cas il s'agit de hello.php
2. Une ouverture de balise PHP (<?php). Pas besoin d'ajouter une balise de fermeture car ce n'est pas obligatoire, contrairement au HTML.
3. Un entête placé entre /* et */ qui parait être un commentaire anodin mais vous allez voir que ce n'est pas le cas.
4. Le code PHP qui permet d'ajouter une fonctionnalité à votre site.

A quoi correspondent chaque ligne de l'entête d'un plugin.

1. la ligne Plugin Name: => elle correspond au nom du plugin tel qu'il sera affiché dans la partie administrateur. On voit bien que celui-ci n'est pas obligatoirement le même que le nom du fichier, qui lui est nommé hello.php.
2. la ligne Plugin URI: => elle correspond à l'URL du plugin sur le repository (répertoire officiel), si ce n'est pas le cas, vous pouvez insérer ici l'URL de votre choix.
3. la ligne Description: => elle permet de décrire la fonctionnalité du plugin.
4. la ligne Author: => elle donne l'identité du créateur du plugin. C'est l'ancre qui apparaitra sur le lien du site de l'auteur (pastille 6).
5. la ligne Version: => elle permet d'afficher le numéro de version du plugin. À chaque nouvelle version, ce numéro évoluera.
6. la ligne Author URI: => elle fait le lien avec le site de l'extension.

Création de notre premier plugin

<https://www.youtube.com/watch?v=Z7QfH-s-15s>

<https://github.com/Alecaddd/WordPressPlugin101>

https://www.youtube.com/watch?v=0l7JTie_6jM&list=PLriKzYyLb28kR_CPMz8uierDWC2y3znI2

The screenshot shows a browser window displaying the WordPress Codex page titled 'Writing a Plugin'. The page content includes an introduction, a sidebar with navigation links like 'Home Page', 'WordPress Lessons', and 'Codex Resources', and a 'Contents' sidebar with a hierarchical list of topics related to plugin development.

Page Title: Writing a Plugin

Page URL: https://codex.wordpress.org/Writing_a_Plugin

Page Content:

- Interested in functions, hooks, classes, or methods? Check out the new [WordPress Code Reference](#)
- Writing a Plugin**
- Languages: English • [العربية](#) • [বাংলা](#) • [Español](#) • [Italiano](#) • [a Plugin 日本語](#) [한국어](#) • [Português do Brasil](#) • [Русский](#) • [עברית](#) • [中文\(简体\)](#) • [\(Add your language\)](#)
- Introduction**

WordPress Plugins allows you to easily modify, customize, and enhance a WordPress site. Instead of changing the core program code of WordPress, you can add functionality with WordPress Plugins. Here is a basic definition.

A **WordPress Plugin** is a program or a set of one or more functions written in the PHP scripting language, that adds a specific set of features or services to the WordPress site. You can seamlessly integrate a plugin with the site using access points and methods provided by the WordPress [Plugin Application Program Interface \(API\)](#).

Wish that WordPress had some new or modified functionality? The first thing to do is to search various WordPress Plugin repositories and sources to see if someone has already created a WordPress Plugin that suits your needs. If not, this article will guide you through the process of creating your own WordPress Plugin.

This article assumes you are already familiar with the basic functionality of WordPress and with PHP programming.
- Contents**
 - 1 Introduction
 - 2 Creating a Plugin
 - 2.1 Names, Files, and Locations
 - 2.1.1 Plugin Name
 - 2.1.2 Plugin Files
 - 2.1.3 Readme File
 - 2.1.4 Home Page
 - 2.2 File Headers
 - 2.2.1 Standard Plugin Information
 - 2.2.2 License
 - 2.3 Programming Your Plugin
 - 2.3.1 WordPress Plugin Hooks
 - 2.3.2 Template Tags
 - 2.3.3 Saving Plugin Data to the Database
 - 2.3.4 WordPress Options
- Codex Resources**
 - [Community portal](#)
 - [Current events](#)
 - [Recent changes](#)
 - [Random page](#)
 - [Help](#)

Header Requirements | Plugin Dev

<https://developer.wordpress.org/plugins/plugin-basics/header-requirements/>

Register Log In

Header Fields

Available header fields:

- **Plugin Name:** (required) The name of your plugin, which will be displayed in the Plugins list in the WordPress Admin.
- **Plugin URI:** The home page of the plugin, which should be a unique URL, preferably on your own website. This *must be unique* to your plugin. You cannot use a WordPress.org URL here.
- **Description:** A short description of the plugin, as displayed in the Plugins section in the WordPress Admin. Keep this description to fewer than 140 characters.
- **Version:** The current version number of the plugin, such as 1.0 or 1.0.3.
- **Author:** The name of the plugin author. Multiple authors may be listed using commas.
- **Author URI:** The author's website or profile on another website, such as WordPress.org.
- **License:** The short name (slug) of the plugin's license (e.g. GPL2). More information about licensing can be found in the [WordPress.org guidelines](#).
- **License URI:** A link to the full text of the license (e.g. <https://www.gnu.org/licenses/gpl-2.0.html>).
- **Text Domain:** The [gettext](#) text domain of the plugin. More information can be found in the [Text Domain](#) section of the [How to Internationalize your Plugin](#) page.
- **Domain Path:** The domain path let WordPress know where to find the translations. More information can be found in the [Domain Path](#) section of the [How to Internationalize your Plugin](#) page.
- **Network:** Whether the plugin can only be activated network-wide. Can only be set to *true*, and should be left out when not needed.

A valid PHP file with a header comment might look like this:

```

1 <?php
2 /**
3 * Plugin Name: WordPress.org Plugin
4 * Plugin URI: https://example.com/plugins/the-basics/
5 * Description: Basic WordPress Plugin Header Comment
6 * Version: 20160911
7 * Author: WordPress.org
8 * Author URI: https://author.example.com/
9 * License: GPL2
10 * License URI: https://www.gnu.org/licenses/gpl-2.0.html
11 * Text Domain: wporg
12 * Domain Path: /languages
13 */

```

You can play with the different header fields using the [Plugin Header Generator](#).

Outil pour générer le fichier nom-du-plugin.php

<https://app.codegenerators.io/59510e630f79a7747d6f3ed164c299d1/>

Header Requirements | Plugin Dev

<https://app.codegenerators.io/59510e630f79a7747d6f3ed164c299d1/>

Share & Embed Login Register

FIELDS

General	Plugin Name ? <input type="text" value="WordPress.org Plugin"/>
Files	Plugin URI ? <input type="text" value="https://developer.wo"/>
Fields	Description ? <input type="text" value="Basic WordPress Plug"/>
	Version ? <input type="text" value="1.0.0"/>
	Author ? <input type="text" value="WordPress.org"/>

plugin.php

```

1 <?php
2 /**
3 * Plugin Name: WordPress.org Plugin
4 * Plugin URI: https://developer.wordpress.org/plugins/the-basi
5 * Description: Basic WordPress Plugin Header Comment
6 * Version: 1.0.0
7 * Author: WordPress.org
8 * Author URI: https://developer.wordpress.org/
9 * Text Domain: wporg
10 * Domain Path: /languages
11 * License: GPL2
12 */
13

```

Displaying Generator 59510e630f79a7747d6f3ed164c299d1 All rights reserved to you! Your code, your intellectual property.

Cf. : <https://www.gnu.org/licenses/old-licenses/gpl-2.0-faq.fr.html>

Je crée un nouveau fichier que je nomme nom-du-plugin.php

J'ajoute le code suivant :

```
<?php
/**
 * Plugin Name: WordPress.org Plugin
 * Plugin URI: https://developer.wordpress.org/plugins/the-basics/
 * Description: Basic WordPress Plugin Header Comment
 * Version: 1.0.0
 * Author: WordPress.org
 * Author URI: https://developer.wordpress.org/
 * Text Domain: wporg
 * Domain Path: /languages
 * License: GPL2
 */
```

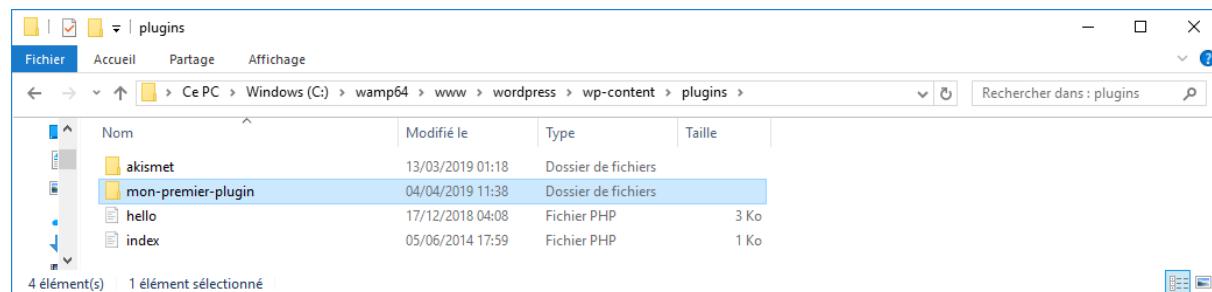
En dessous de cette entête, je vais pouvoir ajouter le code PHP qui correspond à une fonctionnalité de l'application.

J'enregistre mon fichier et je le range dans un dossier que je nomme par exemple mon-premier-plugin qui lui-même se trouve dans le dossier plugins.

Dans ce dossier je peux y enregistrer un fichier readme.txt qui donne une explication d'installation pour mon extension.

Je teste mon plugin et vérifie s'il fonctionne bien une fois activé depuis le back-office de mon application WordPress en développement.

Je peux envoyer mon plugin dans mon application WordPress en production soit via FTP, soit via le back-office mais dans ce cas il doit être compressé.



The screenshot shows the WordPress admin interface under the 'Extensions' tab. The left sidebar includes links for Tableau de bord, Articles, Médias, Pages, Commentaires, Apparence, Extensions (which is selected), Extensions installées, Ajouter, Éditeur d'extension, Utilisateurs, Outils, Réglages, and Réduire le menu. The main content area is titled 'Extensions' with an 'Ajouter' button. It displays a list of installed plugins: Akismet Anti-Spam, Hello Dolly, and Mon premier Plugin. Each plugin entry includes an 'Activer' (Activate) or 'Supprimer' (Delete) link. A search bar at the top right says 'Recherchez parmi vos extensions...'. A note at the bottom says 'Merci de faire de WordPress votre outil de création.' and 'Version 5.1.1'.

The screenshot shows the 'Modifier les extensions' (Edit Plugins) page for the 'mon-premier-plugin' plugin. The left sidebar is identical to the previous screenshot. The main content area has a title 'Modifier les extensions' and a sub-section 'Modification de mon-premier-plugin/mon-premier-plugin.php (désactivée)'. It features a dropdown 'Sélectionnez l'extension à modifier : Mon premier Plugin' and a 'Sélectionner' button. On the left, there's a code editor for the file 'Content du fichier sélectionné : mon-premier-plugin.php' containing the following PHP code:

```
1 <?php
2 /**
3 * Plugin Name: Mon premier Plugin
4 * Plugin URI: https://monsitere.com/plugin
5 * Description: Mon premier essai pour créer un plugin
6 * Version: 1.0.0
7 * Author: Patrick Croquet
8 * Author URI: https://monsitere.com/
9 * Text Domain: mon-premier-plugin
10 * Domain Path: /languages
11 * License: GPL2
12 */
13
```

On the right, there's a 'Fichiers de l'extension' (Plugin files) section with a list containing 'mon-premier-plugin.php'. At the bottom, there's a blue button 'Mettre à jour le fichier' (Update file).

```

<?php
// le silence est d'or

```

Extension	Description
Akismet Anti-Spam Activer Supprimer	Utilisé par des millions d'utilisateurs, Akismet est probablement la meilleure solution au monde pour protéger votre blog des indésirables. Il garde votre site protégé même quand vous dormez. Pour commencer : activez l'extension Akismet et allez ensuite dans la page de réglages pour configurer votre clé API. Version 4.1.1 Par Automatic Afficher les détails
Hello Dolly Activer Supprimer	Ce n'est pas qu'une extension. Elle symbolise l'espoir et l'enthousiasme de toute une génération, résumé en deux mots, qu'a notamment chanté Louis Armstrong : Hello, Dolly. Une fois activée, elle affichera une ligne aléatoirement des paroles de la chanson Hello, Dolly, en haut à droite de toutes les pages de l'administration. Version 1.7.1 Par Matt Mullenweg Afficher les détails
Mon premier Plugin Désactiver	Mon premier essai pour créer un plugin Version 1.0.0 Par Patrick Croquet Aller sur le site de l'extension

Merci de faire de [WordPress](#) votre outil de création.

Télécharger

The screenshot shows a GitHub repository page for 'Alecadd / WordPressPlugin101'. The repository name is 'WordPressPlugin101 / Lesson19_starter'. The 'Code' tab is selected. The 'Branch: master' dropdown shows 'WordPressPlugin101 / Lesson19_starter /'. The commit history lists several files and folders, all modified 'a year ago'. The commits are:

- Alecadd Delete package-lock.json ...
- ..
- assets Gulpfile and Package.json
- inc Gulpfile and Package.json
- src Gulpfile and Package.json
- templates Gulpfile and Package.json
- vendor Gulpfile and Package.json
- alecadd-plugin.php Gulpfile and Package.json
- composer.json Gulpfile and Package.json
- gulpfile.js Gulpfile and Package.json
- index.php Gulpfile and Package.json
- package.json Gulpfile and Package.json
- readme.md Gulpfile and Package.json
- uninstall.php Gulpfile and Package.json

At the top right, there are buttons for 'Watch 35', 'Star 109', 'Fork 123', 'Create new file', 'Upload files', 'Find file', and 'History'.

The screenshot shows a Windows File Explorer window titled 'monpremier-plugin'. The path is 'Windows (C:) > wamp64 > www > wordpress > wp-content > plugins > monpremier-plugin'. The 'Fichier' tab is selected. The contents of the folder are listed in a table:

Nom	Modifié le	Type	Taille
assets	11/04/2019 14:49	Dossier de fichiers	
inc	11/04/2019 14:49	Dossier de fichiers	
src	11/04/2019 14:49	Dossier de fichiers	
templates	11/04/2019 14:49	Dossier de fichiers	
vendor	11/04/2019 14:49	Dossier de fichiers	
alecadd-plugin	06/01/2019 20:24	Fichier PHP	2 Ko
composer.json	06/01/2019 20:24	Fichier JSON	1 Ko
gulpfile	06/01/2019 20:24	Fichier de JavaScript	3 Ko
index	06/01/2019 20:24	Fichier PHP	1 Ko
package.json	06/01/2019 20:24	Fichier JSON	2 Ko
readme.md	06/01/2019 20:24	Fichier MD	1 Ko
uninstall	06/01/2019 20:24	Fichier PHP	1 Ko

At the bottom left, it says '12 élément(s)'. At the bottom right, there are icons for 'Copier' (Copy), 'Coller' (Paste), and 'Supprimer' (Delete).

The screenshot shows the Visual Studio Code interface. The left sidebar displays a file tree for a WordPress plugin folder. The current file being edited is `gulpfile.js`, which contains Gulp task definitions. The terminal at the bottom shows a PowerShell session running commands to navigate through a WordPress installation's plugin directory.

```
// Load Gulp...of course
var gulp = require('gulp');

// CSS related plugins
var sass = require('gulp-sass');
var autoprefixer = require('gulp-autoprefixer');
var minifycss = require('gulp-uglifycss');

// JS related plugins
var concat = require('gulp-concat');
var uglify = require('gulp-uglify');
var babelify = require('babelify');
var browserify = require('browserify');
var source = require('vinyl-source-stream');
var buffer = require('vinyl-buffer');
var stripDebug = require('gulp-strip-debug');

// Utility plugins
var rename = require('gulp-rename');

Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. Tous droits réservés.

PS C:\wamp64\www\wordpress> cd wp-content
PS C:\wamp64\www\wordpress\wp-content> cd plugins
PS C:\wamp64\www\wordpress\wp-content\plugins> cd monpremier-plugin
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin>
```

<https://gulpjs.com/docs/en/getting-started/quick-start>

<https://www.npmjs.com/package/npx>

```
npm install gulp --g
gulp watch
```

Eichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide gulpfile.js - wordpress - Visual Studio Code

EXPLORATEUR

- ÉDITEURS OUVERTS
 - JS gulpfile.js wp-content\plugins\...
- WORDPRESS
 - SettingsApi.php
 - Base
 - Pages
 - Admin.php
 - Init.php
 - src
 - js
 - myscript.js
 - scss
 - modules
 - tabs.scss
 - mystyle.scss
 - templates
 - admin.php
 - cpt.php
 - taxonomy.php
 - widget.php
 - vendor
 - alecaddd-plugin.php
 - composer.json
 - index.php
 - package.json
 - readme.md

STRUCTURE

0 ▲ 0

Li 1, Col 1 Taille des tabulations : 4 UTF-8 LF JavaScript

```
// Load Gulp...of course
var gulp = require('gulp');

// CSS related plugins
var sass = require('gulp-sass');
var autoprefixer = require('gulp-autoprefixer');
var minifycss = require('gulp-uglifycss');

// JS related plugins
var concat = require('gulp-concat');
var uglify = require('gulp-uglify');
var babelify = require('babelify');
var browserify = require('browserify');
var source = require('vinyl-source-stream');
var buffer = require('vinyl-buffer');
var stripDebug = require('gulp-strip-debug');

// Utility plugins
var rename = require('gulp-rename');
var sourcemap = require('gulp-sourcemaps');
```

PROBLÈMES SORTIE CONSOLE DE DÉBOGAGE TERMINAL 1: powershell

```
PS C:\wamp64\www\wordpress> cd wp-content
PS C:\wamp64\www\wordpress\wp-content> cd plugins
PS C:\wamp64\www\wordpress\wp-content\plugins> cd monpremier-plugin
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin> npm install
```

Eichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide gulpfile.js - wordpress - Visual Studio Code

EXPLORATEUR

- ÉDITEURS OUVERTS
 - JS gulpfile.js wp-content\plugins\...
- WORDPRESS
 - SettingsApi.php
 - Base
 - Pages
 - Admin.php
 - Init.php
 - node_modules
 - src
 - js
 - myscript.js
 - scss
 - modules
 - tabs.scss
 - mystyle.scss
 - templates
 - admin.php
 - cpt.php
 - taxonomy.php
 - widget.php
 - vendor
 - alecaddd-plugin.php
 - composer.json
 - index.php
 - package-lock.json

STRUCTURE

0 ▲ 0

Li 1, Col 1 Taille des tabulations : 4 UTF-8 LF JavaScript

```
// Load Gulp...of course
var gulp = require('gulp');

// CSS related plugins
var sass = require('gulp-sass');
var autoprefixer = require('gulp-autoprefixer');
var minifycss = require('gulp-uglifycss');

// JS related plugins
var concat = require('gulp-concat');
var uglify = require('gulp-uglify');
var babelify = require('babelify');
var browserify = require('browserify');
var source = require('vinyl-source-stream');
var buffer = require('vinyl-buffer');
var stripDebug = require('gulp-strip-debug');

// Utility plugins
var rename = require('gulp-rename');
var sourcemap = require('gulp-sourcemaps');
var notify = require('gulp-notify');
var plumber = require('gulp-plumber');
```

PROBLÈMES SORTIE CONSOLE DE DÉBOGAGE TERMINAL 1: powershell

```
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin> npm install --save-dev gulp-watch
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.7 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.7: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
+ gulp-watch@5.0.1
added 14 packages from 35 contributors and audited 10099 packages in 7.154s
found 9 vulnerabilities (3 low, 1 moderate, 5 high)
  run `npm audit fix` to fix them, or `npm audit` for details
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin>
```

The screenshot shows the WordPress admin dashboard with the Alecaddd Plugin settings page open. The left sidebar has a dark theme with white text. The 'Alecadd' menu item is highlighted in blue, indicating the current plugin's settings page. The main content area has a light gray background. At the top, there is a message box with a green header bar containing the text 'Régistres enregistrés.' and a close button. Below this is a navigation bar with three tabs: 'Manage Settings' (selected), 'Updates', and 'About'. A section titled 'Settings Manager' follows, with the sub-instruction 'Manage the Sections and Features of this Plugin by activating the checkboxes from the following list.' Below this are eight toggle switches, each labeled with a feature name:

- Activate CPT Manager (switch is on)
- Activate Taxonomy Manager (switch is off)
- Activate Media Widget (switch is off)
- Activate Gallery Manager (switch is off)
- Activate Testimonial Manager (switch is off)
- Activate Templates Manager (switch is off)
- Activate Ajax Login/Signup (switch is off)
- Activate Membership Manager (switch is off)
- Activate Chat Manager (switch is off)

At the bottom of the page is a blue button labeled 'Enregistrer les modifications'.

The screenshot shows the WordPress admin dashboard with the Alecaddd Plugin settings page open. The left sidebar has a dark theme with white text. The main content area has a light background.

Alecaddd Plugin

Réglages enregistrés.

Manage Settings Updates About

Settings Manager

Manage the Sections and Features of this Plugin by activating the checkboxes from the following list.

Activate CPT Manager

Activate Taxonomy Manager

Activate Media Widget

Activate Gallery Manager

Activate Testimonial Manager

Activate Templates Manager

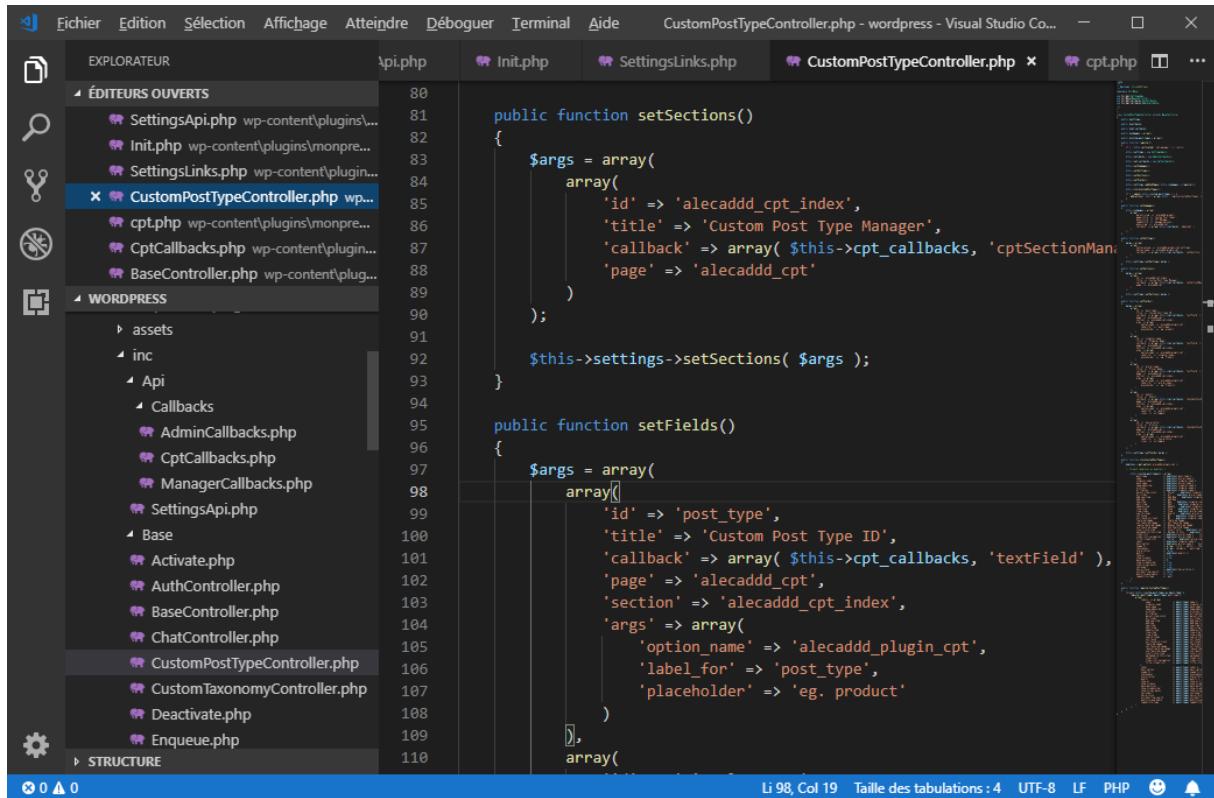
Activate Ajax Login/Signup

Activate Membership Manager

Activate Chat Manager

Enregistrer les modifications

WordPress Plugin Development - Part 26b - Modular Custom Post Types

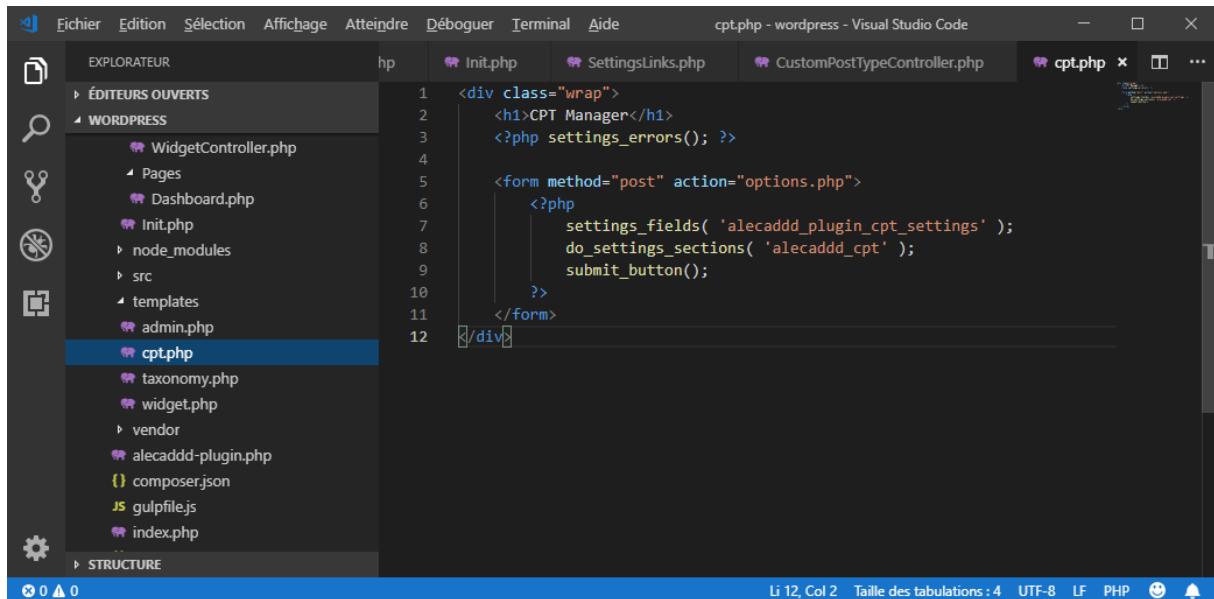


The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "WORDPRESS".
- Editor:** Displays the code for `CustomPostTypeController.php`. The code defines two methods: `setSections()` and `setFields()`. Both methods use arrays to map configuration keys to plugin settings.
- Status Bar:** Shows the current file is `CustomPostTypeController.php`, the character position is Li 98, Col 19, and the encoding is UTF-8.

```
public function setSections()
{
    $args = array(
        array(
            'id' => 'alecaddd_cpt_index',
            'title' => 'Custom Post Type Manager',
            'callback' => array( $this->cpt_callbacks, 'cptSectionManager' ),
            'page' => 'alecaddd_cpt'
        )
    );
    $this->settings->setSections( $args );
}

public function setFields()
{
    $args = array(
        array(
            'id' => 'post_type',
            'title' => 'Custom Post Type ID',
            'callback' => array( $this->cpt_callbacks, 'textField' ),
            'page' => 'alecaddd_cpt',
            'section' => 'alecaddd_cpt_index',
            'args' => array(
                'option_name' => 'alecaddd_plugin_cpt',
                'label_for' => 'post_type',
                'placeholder' => 'eg. product'
            )
        ),
        array(
```



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "WORDPRESS".
- Editor:** Displays the code for `cpt.php`. The code contains PHP and HTML. It includes a form for managing custom post type settings, specifically for the 'alecaddd_plugin_cpt' setting.
- Status Bar:** Shows the current file is `cpt.php`, the character position is Li 12, Col 2, and the encoding is UTF-8.

```
<div class="wrap">
<h1>CPT Manager</h1>
<?php settings_errors(); ?>

<form method="post" action="options.php">
    <?php
        settings_fields( 'alecaddd_plugin_cpt_settings' );
        do_settings_sections( 'alecaddd_cpt' );
        submit_button();
    ?>
</form>
</div>
```

The screenshot shows the WordPress admin interface for a custom post type named "product".

Custom Post Type ID: product

Singular Name: Product

Plural Name: Products

Public: Enabled (switch is green)

Archive: Enabled (switch is green)

Buttons:

- Enregistrer les modifications (Save Changes)

Footer:

- Merci de faire de [WordPress](#) votre outil de création.
- Version 5.1.1

The screenshot shows the WordPress admin dashboard with the title "Custom Post Types < Mon site". The left sidebar is titled "Alecadd" and includes links for Dashboard, CPT Manager, Taxonomy Manager, Chat Manager, Gallery Manager, Testimonial Manager, Templates Manager, Login Manager, Membership Manager, and Widgets Manager. The main content area is titled "CPT Manager" and "Custom Post Type Manager". It displays fields for "Custom Post Type ID" (set to "comicbook"), "Singular Name" (set to "Comic Book"), "Plural Name" (set to "Comic books"), "Public" (set to "on"), and "Archive" (set to "on"). A blue button at the bottom says "Enregistrer les modifications". At the bottom of the page, there is a footer note "Merci de faire de [WordPress](#) votre outil de création." and the text "Version 5.1.1".

CPT Manager

Custom Post Type Manager

Create as many Custom Post Types as you want.

Custom Post Type ID: comicbook

Singular Name: Comic Book

Plural Name: Comic books

Public:

Archive:

Enregistrer les modifications

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

The screenshot shows the WordPress admin interface with the title "Custom Post Types < Mon site". The left sidebar is titled "Alecadd" and includes links such as Dashboard, CPT Manager, Taxonomy Manager, Chat Manager, Gallery Manager, Testimonial Manager, Templates Manager, Login Manager, Membership Manager, Widgets Manager, and Réduire le menu. The main content area is titled "CPT Manager" and contains a message "Régistres enregistrés." A section titled "Custom Post Type Manager" allows creating multiple Custom Post Types. The current configuration is as follows:

- Custom Post Type ID: comicbook
- Singular Name: Comic Book
- Plural Name: Comic books
- Public:
- Archive:

A blue button labeled "Enregistrer les modifications" (Save changes) is at the bottom. The footer of the page includes the text "Merci de faire de [WordPress](#) votre outil de création." and "Version 5.1.1".

localhost / MySQL / wordpress / +

Serveur: MySQL:3306 » Base de données: wordpress » Table: wp_options

Colonne	Type	Fonction	Null	Valeur
option_id	bigint(20) unsigned			539
option_name	varchar(191)			<code>alecaddd_plugin_cpt</code>
				<code>a:5:{s:9:"post_type";s:9:"comicbook";s:13:"singular_name";s:10:"Comic Book";s:11:"plural_name";s:11:"Comic books";s:6:"public";s:1:"1";s:11:"has_archive";s:1:"1";}</code>
option_value	longtext			
autoload	varchar(20)			yes

Exécuter

CustomPostTypeController.php - wordpress - Visual Studio Code

```
162
163     public function storeCustomPostTypes()
164     {
165         $options = get_option('alecadd_plugin_cpt');
166
167         // foreach ($options as $option) {
168
169             $this->custom_post_types[] = array(
170                 'post_type'          => $options['post_type'],
171                 'name'               => $options['plural_name'],
172             );
173
174     }
175
176     $this->custom_post_types[] = array(
177         'post_type'          => $options['post_type'],
178         'name'               => $options['plural_name'],
179         'singular_name'      => $options['singular_name'],
180         'menu_name'          => $options['plural_name'],
181         'name_admin_bar'     => $options['singular_name'],
182         'archives'           => $options['singular_name'] . ' Archives',
183         'attributes'         => $options['singular_name'] . ' Attributes',
184         'parent_item_colon'  => 'Parent ' . $options['singular_name'],
185         'all_items'          => 'All ' . $options['plural_name'],
186         'add_new_item'       => 'Add New ' . $options['singular_name'],
187         'add_new'            => 'Add New',
188         'new_item'           => 'New ' . $options['singular_name'],
189         'edit_item'          => 'Edit ' . $options['singular_name'],
190         'update_item'        => 'Update ' . $options['singular_name'],
191         'view_item'          => 'View ' . $options['singular_name'],
192         'view_items'         => 'View ' . $options['plural_name'],
193         'search_items'       => 'Search ' . $options['plural_name'],
194         'not_found'          => 'No ' . $options['singular_name'] . ' Found',
195         'not_found_in_trash' => 'No ' . $options['singular_name'] . ' Found in Trash',
196         'featured_image'     => 'Featured Image',
197         'set_featured_image' => 'Set Featured Image',
198         'remove_featured_image' => 'Remove Featured Image',
199         'use_featured_image' => 'Use Featured Image',
200         'insert_into_item'   => 'Insert into ' . $options['singular_name'],
201         'uploaded_to_this_item' => 'Upload to this ' . $options['singular_name'],
202         'items_list'          => $options['plural_name'] . ' List',
203         'items_list_navigation' => $options['plural_name'] . ' List Navigation',
204         'filter_items_list'   => 'Filter' . $options['plural_name'] . ' List',
205         'label'               => $options['singular_name'],
206         'description'         => $options['plural_name'] . ' Custom Post Type',
207         'supports'            => array( 'title', 'editor', 'thumbnail' ),
208         'taxonomies'          => array( 'category', 'post_tag' ),
209         'hierarchical'        => false,
210         'public'              => $options['public'],
211         'show_ui'             => true,
212         'show_in_menu'        => true,
213         'menu_position'       => 5,
214         'show_in_admin_bar'   => true,
215         'show_in_nav_menus'   => true,
216         'can_export'          => true,
217         'has_archive'         => $options['has_archive'],
218         'exclude_from_search' => false,
219         'publicly_queryable'  => true,
220         'capability_type'    => 'post'
221     );
222 }
```

CustomPostTypeController.php - wordpress - Visual Studio Code

```
165
166
167     $options = get_option('alecadd_plugin_cpt');
168
169     // foreach ($options as $option) {
170
171         $this->custom_post_types[] = array(
172             'post_type'          => $options['post_type'],
173             'name'               => $options['plural_name'],
174             'singular_name'      => $options['singular_name'],
175             'menu_name'          => $options['plural_name'],
176             'name_admin_bar'     => $options['singular_name'],
177             'archives'           => $options['singular_name'] . ' Archives',
178             'attributes'         => $options['singular_name'] . ' Attributes',
179             'parent_item_colon'  => 'Parent ' . $options['singular_name'],
180             'all_items'          => 'All ' . $options['plural_name'],
181             'add_new_item'       => 'Add New ' . $options['singular_name'],
182             'add_new'            => 'Add New',
183             'new_item'           => 'New ' . $options['singular_name'],
184             'edit_item'          => 'Edit ' . $options['singular_name'],
185             'update_item'        => 'Update ' . $options['singular_name'],
186             'view_item'          => 'View ' . $options['singular_name'],
187             'view_items'         => 'View ' . $options['plural_name'],
188             'search_items'       => 'Search ' . $options['plural_name'],
189             'not_found'          => 'No ' . $options['singular_name'] . ' Found',
190             'not_found_in_trash' => 'No ' . $options['singular_name'] . ' Found in Trash',
191             'featured_image'     => 'Featured Image',
192             'set_featured_image' => 'Set Featured Image',
193             'remove_featured_image' => 'Remove Featured Image',
194             'use_featured_image' => 'Use Featured Image',
195             'insert_into_item'   => 'Insert into ' . $options['singular_name'],
196             'uploaded_to_this_item' => 'Upload to this ' . $options['singular_name'],
197             'items_list'          => $options['plural_name'] . ' List',
198             'items_list_navigation' => $options['plural_name'] . ' List Navigation',
199             'filter_items_list'   => 'Filter' . $options['plural_name'] . ' List',
200             'label'               => $options['singular_name'],
201             'description'         => $options['plural_name'] . ' Custom Post Type',
202             'supports'            => array( 'title', 'editor', 'thumbnail' ),
203             'taxonomies'          => array( 'category', 'post_tag' ),
204             'hierarchical'        => false,
205             'public'              => $options['public'],
206             'show_ui'             => true,
207             'show_in_menu'        => true,
208             'menu_position'       => 5,
209             'show_in_admin_bar'   => true,
210             'show_in_nav_menus'   => true,
211             'can_export'          => true,
212             'has_archive'         => $options['has_archive'],
213             'exclude_from_search' => false,
214             'publicly_queryable'  => true,
215             'capability_type'    => 'post'
216         );
217     );
218 }
```

The screenshot shows the WordPress admin dashboard for a site titled "Mon site". The left sidebar is dark-themed and includes links for "Tableau de bord", "Articles", "Comic books" (which is the active tab), "All Comic books", "Add New", "Catégories", "Étiquettes", "Médias", "Pages", "Commentaires", "Apparence", "Extensions", "Utilisateurs", "Outils", "Réglages", "Alecaddd", and "Réduire le menu". The top bar shows the URL "localhost/wordpress/wp-admin/edit.php?post_type=comicbook". The main area is titled "Comic books" with a "Add New" button. A search/filter section shows "Tous (0)" and "Filtrer". Below it, a table header for "No Comic Book Found" includes columns for "Titre", "Catégories", "Étiquettes", and "Date". A note at the bottom left says "Merci de faire de WordPress votre outil de création." and the version "Version 5.1.1" is at the bottom right.

W Add New Comic Book < Mon site +

localhost/wordpress/wp-admin/post-new.php?post_type=comicbook

Bonjour, Patrick

Tableau de bord Articles Comic books All Comic books Add New Catégories Étiquettes Médias Pages Commentaires Apparence Extensions Utilisateurs Outils Réglages Alecaddd Réduire le menu

Add New Comic Book

Saisissez votre titre ici

Ajouter un média Visuel Texte

Paragraphe B I Éléments

Nombre de mots : 0

Merci de faire de [WordPress](#) votre outil de création.

Publier

Enregistrer le brouillon Aperçu

État : Brouillon [Modifier](#)

Visibilité : Public [Modifier](#)

Publier tout de suite [Modifier](#)

Publier

Catégories

Toutes Plus utilisés

Non classé

+ Ajouter une nouvelle catégorie

Étiquettes

Ajouter

Séparez les étiquettes par des virgules

Choisir parmi les étiquettes les plus utilisées

Featured Image

[Set Featured Image](#)

Version 5.1.1

The screenshot shows the 'CPT Manager' page in the WordPress admin area. On the left, a sidebar lists various menu items like 'Tableau de bord', 'Articles', 'Products', etc. The 'Alecaddd' item is highlighted. The main content area is titled 'CPT Manager' and contains a message 'Régles enregistrés.' (Rules saved). Below it, the 'Custom Post Type Manager' section allows creating multiple custom post types. For the current entry, the 'Custom Post Type ID' is set to 'product', 'Singular Name' is 'Product', and 'Plural Name' is 'Products'. The 'Public' and 'Archive' options are both turned on (indicated by blue toggles). At the bottom right is a blue button labeled 'Enregistrer les modifications' (Save changes).

Products ont remplacé Comic Books

The screenshot shows the MySQL Workbench interface with a query editor window. The query being run is:

```

UPDATE `wp_options` SET `option_value` = 'a:5:{s:9:"post_type";s:7:"product";s:13:"singular_name";s:7:"Product";s:11:"plural_name";s:8:"Products";s:6:"public";s:1:"1";s:11:"has_archive";s:1:"1";}' WHERE `option_name` = 'alecaddd_plugin_cpt';
    
```

The 'option_name' field is set to 'alecaddd_plugin_cpt' and the 'option_value' field is set to the JSON string above. At the bottom right of the query editor is a red 'Exécuter' (Execute) button.

WordPress Plugin Development - Part 27 - Store Arrays in WP_Options

The screenshot shows the 'CPT Manager' settings page within the WordPress admin interface. The left sidebar has a blue-highlighted 'Alecadd' menu item. The main area is titled 'Custom Post Type Manager' with the sub-instruction 'Create as many Custom Post Types as you want.' A form is present with the following fields:

- Custom Post Type ID: comicbook
- Singular Name: Comic Book
- Plural Name: Comic books
- Public: On (blue toggle switch)
- Archive: On (blue toggle switch)

A blue 'Enregistrer les modifications' (Save changes) button is at the bottom.

WordPress Plugin Development - Part 29 - Bug Fixes

The screenshot shows the 'wp_options' table in the phpMyAdmin database interface. The table structure includes columns: option_id, option_name, option_value, and autoload. The data grid shows several rows, with the last row being the one we're interested in:

option_id	option_name	option_value	autoload
732	alecaddd_plugin cpt	a:0:{}	yes

At the bottom of the interface, there's an 'Opérations sur les résultats de la requête' (Operations on the query results) section with buttons for Imprimer, Copier dans le presse-papiers, Exporter, Afficher le graphique, and Crée une vue.

Je supprime l'enregistrement 732 alecaddd_plugin_cpt

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure. The `cpt.php` file is currently selected.
- Editor (Center):** Displays the content of the `cpt.php` file, which is part of a plugin named "CPT Manager". The code includes functions for managing custom post types, handling errors, and displaying tabs for managing post types.
- Terminal (Bottom Left):** Shows the command `npm start`.
- Status Bar (Bottom):** Shows the file path as `cpt.php - wordpress - Visual Studio Code`, the line number as `Li 23, Col 17`, the tab size as `Taille des tabulations : 4`, and the file type as `UTF-8 LF PHP`.

```
$options = ! get_option( 'alecaddd_plugin_cpt' ) ? array() : get_option( 'alecaddd_plugin_cpt' );  
var_dump($options);
```

The screenshot shows the WordPress admin dashboard with a sidebar on the left and a main content area on the right.

Left Sidebar:

- Custom Post Types < Mon site
- Créer
- Tableau de bord
- Articles
- Médias
- Pages
- Commentaires
- Apparence
- Extensions
- Utilisateurs
- Outils
- Réglages
- Alecaddd
- Dashboard
- CPT Manager
- Réduire le menu

Right Content Area:

CPT Manager

Your Custom Post Types [Add Custom Post Type](#) [Export](#)

Manage Your Custom Post Types

array(0) {}

ID	Singular Name	Plural Name	Public	Archive	Actions
----	---------------	-------------	--------	---------	---------

Merci de faire de [WordPress](#) votre outil de création.

Eichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide cpt.php - wordpress - Visual Studio Code

EXPLORATEUR # mystyle.css cpt.php CustomPostTypeController.php CptCallbacks.php

WORDPRESS

- Deactivate.php
- Enqueue.php
- GalleryControl...
- MembershipC...
- SettingsLinks.p...
- TemplateContr...
- TestimonialCo...
- WidgetControl...
- Pages
- Init.php
- node_modules
- src
- srold
- templates
- admin.php
- cpt.php**
- taxonomy.php
- widget.php
- vendor
- alecaddd-plugin....
- composer.json

STRUCTURE

```

1 <div class="wrap">
2   <h1>CPT Manager</h1>
3   <?php settings_errors(); ?>
4
5   <ul class="nav nav-tabs">
6     <li class="active"><a href="#tab-1">Your Custom Post Types</a></li>
7     <li><a href="#tab-2">Add Custom Post Type</a></li>
8     <li><a href="#tab-3">Export</a></li>
9   </ul>
10
11   <div class="tab-content">
12     <div id="tab-1" class="tab-pane active">
13       <h3>Manage Your Custom Post Types</h3>
14
15       <?php
16         //if ( ! get_option( 'alecaddd_plugin_cpt' ) ) {
17         //  $options = array();
18         //}
19         // else {
20         //  $options = get_option( 'alecaddd_plugin_cpt' );
21         //}
22
23         // $options = ! get_option( 'alecaddd_plugin_cpt' ) ? array() : get_opt
24
25         $options = get_option( 'alecaddd_plugin_cpt' ) ?: array();
26
27         echo '<table class="cpt-table"><tr><th>ID</th><th>Singular Name</th></tr>';
28

```

Li 26, Col 1 Taille des tabulations : 4 UTF-8 LF PHP

Custom Post Types < Mon site

localhost/wordpress/wp-admin/admin.php?page=alecaddd_cpt

Mon site Crée Bonjour, Patrick

Tableau de bord

- Articles
- Médias
- Pages
- Commentaires
- Apparence
- Extensions
- Utilisateurs
- Outils
- Réglages
- Alecadd
- Dashboard
- CPT Manager**
- Réduire le menu

YOur Custom Post Types Add Custom Post Type Export

Manage Your Custom Post Types

ID	Singular Name	Plural Name	Public	Archive	Actions

Merci de faire de WordPress votre outil de création. Version 5.1.1

```

151     'id' => 'has_archive',
152     'title' => 'Archive',
153     'callback' => array( $this->cpt_callbacks, 'checkboxField' ),
154     'page' => 'alecaddd_cpt',
155     'section' => 'alecaddd_cpt_index',
156     'args' => array(
157         'option_name' => 'alecaddd_plugin_cpt',
158         'label_for' => 'has_archive',
159         'class' => 'ui-toggle',
160         'array' => 'post_type'
161     )
162 );
163 );
164
165     $this->settings->setFields( $args );
166 }
167
168 public function storeCustomPostTypes()
169 {
170     $options = get_option( 'alecaddd_plugin_cpt' ) ?: array();
171
172     foreach ( $options as $option ) {
173
174         $this->custom_post_types[] = array(
175             'post_type' => $option['post_type'],
176             'name' => $option['plural_name'],
177             'singular_name' => $option['singular_name'],
178             'menu_name' => $option['plural_name'],
179         );
180     }
181 }

```

Li 170, Col 1 Taille des tabulations : 4 UTF-8 LF PHP ⚡ 🔔

```

6
7 class CptCallbacks
8 {
9
10    public function cptSectionManager()
11    {
12        echo 'Create as many Custom Post Types as you want.';
13    }
14
15    public function cptSanitize( $input )
16    {
17        $output = get_option('alecaddd_plugin_cpt');
18
19        if ( count($output) == 0 ) {
20            $output[$input['post_type']] = $input;
21
22            return $output;
23        }
24
25        foreach ( $output as $key => $value ) {
26            if ( $input['post_type'] === $key ) {
27                $output[$key] = $input;
28            } else {
29                $output[$input['post_type']] = $input;
30            }
31        }
32
33        return $output;

```

Li 19, Col 32 Taille des tabulations : 4 UTF-8 LF PHP ⚡ 🔔

Custom Post Types - Mon site — +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_cpt

Bonjour, Patrick

Tableau de bord Articles Médias Pages Commentaires Apparence Extensions Utilisateurs Outils Réglages Alecaddd Dashboard CPT Manager Réduire le menu

CPT Manager

Your Custom Post Types Add Custom Post Type Export

Custom Post Type Manager

Create as many Custom Post Types as you want.

Custom Post Type ID	comicbook
Singular Name	Comic Book
Plural Name	Comic books
Public	<input checked="" type="checkbox"/>
Archive	<input checked="" type="checkbox"/>

Enregistrer les modifications

Merci de faire de [WordPress](#) votre outil de création. Version 5.1.1

The screenshot shows the 'CPT Manager' settings page. A new custom post type 'comicbook' has been created with the singular name 'Comic Book' and plural name 'Comic books'. Both 'Public' and 'Archive' options are enabled. The 'Enregistrer les modifications' button is visible at the bottom.

Custom Post Types - Mon site — +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_cpt

Bonjour, Patrick

Tableau de bord Articles Comic books Médias Pages Commentaires Apparence Extensions Utilisateurs Outils Réglages Alecaddd Dashboard CPT Manager Réduire le menu

CPT Manager

Réglages enregistrés.

Your Custom Post Types Add Custom Post Type Export

Manage Your Custom Post Types

ID	Singular Name	Plural Name	Public	Archive	Actions
comicbook	Comic Book	Comic books	TRUE	TRUE	EDIT - DELETE

Merci de faire de [WordPress](#) votre outil de création. Version 5.1.1

The screenshot shows the 'Manage Your Custom Post Types' list. It displays one item: 'comicbook' with a singular name of 'Comic Book' and a plural name of 'Comic books'. Both 'Public' and 'Archive' are set to TRUE. An 'EDIT - DELETE' link is shown for the item.

```

    . $name . ']" value="" placeholder="' . $args['placeholder'] . '" required>';

on_name . ']' . $name . '1" value="1" class=""><label for="' . $name . '">' . $name . '</div></div>

```

Li 50, Col 48 Taille des tabulations : 4 UTF-8 LF PHP ⚡ 📰

```

        'remove_featured_image' => 'Remove Featured Image',
        'use_featured_image' => 'Use Featured Image',
        'insert_into_item' => 'Insert into ' . $option['singular_name'],
        'uploaded_to_this_item' => 'Upload to this ' . $option['singular_name'],
        'items_list' => $option['plural_name'] . ' List',
        'items_list_navigation' => $option['plural_name'] . ' List Navigation',
        'filter_items_list' => 'Filter' . $option['plural_name'] . ' List',
        'label' => $option['singular_name'],
        'description' => $option['plural_name'] . ' Custom Post Type',
        'supports' => array( 'title', 'editor', 'thumbnail' ),
        'taxonomies' => array( 'category', 'post_tag' ),
        'hierarchical' => false,
        'public' => isset($option['public']) ?: false,
        'show_ui' => true,
        'show_in_menu' => true,
        'menu_position' => 5,
        'show_in_admin_bar' => true,
        'show_in_nav_menus' => true,
        'can_export' => true,
        'has_archive' => isset($option['has_archive']) ?: false,
        'exclude_from_search' => false,
        'publicly_queryable' => true,
        'capability_type' => 'post'
    );
}

public function registerCustomPostTypes()
{
}

```

Li 208, Col 19 Taille des tabulations : 4 UTF-8 LF PHP ⚡ 📰

Modification lignes 208 et 215

Eichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide cpt.php - wordpress - Visual Studio Code

EXPLORATEUR # mystyle.css cpt.php CustomPostTypeController.php CptCallbacks.php

ÉDITEURS OUVERTS WORDPRESS

- admin.php
- cpt.php
- taxonomy.php
- widget.php
- vendor
- alecaddd-plugin....
- composer.json
- gulpfile.js
- index.php
- package-lock.json
- package.json
- readme.md
- uninstall.php
- hello.php
- index.php
- themes
- upgrade
- uploads
- index.php
- wp-includes
- .htaccess

STRUCTURE

```

14 <h3>Manage Your Custom Post Types</h3>
15
16 <?php
17     //if ( ! get_option( 'alecaddd_plugin_cpt' ) ) {
18     // $options = array();
19     //} else {
20     // $options = get_option( 'alecaddd_plugin_cpt' );
21     //}
22
23     // $options = ! get_option( 'alecaddd_plugin_cpt' ) ? array() : get_option( 'alecaddd_plugin_cpt' );
24
25     $options = get_option( 'alecaddd_plugin_cpt' ) ?: array();
26
27     echo '<table class="cpt-table"><tr><th>ID</th><th> Singular Name</th><th>Plural Name</th><th>Public</th><th>Archive</th><th>Meta Box</th><th>Actions</th></tr>';
28
29     foreach ($options as $option) {
30         $public = isset($option['public']) ? "TRUE" : "FALSE";
31         $archive = isset($option['has_archive']) ? "TRUE" : "FALSE";
32
33         echo "<tr><td>{$option['post_type']}


Li 30, Col 13 Taille des tabulations : 4 UTF-8 LF PHP


```

Custom Post Types < Mon site — +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_cpt

Mon site 0 + Crée Bonjour, Patrick

Tableau de bord

- Articles
- Comic books
- Médias
- Pages
- Commentaires
- Apparence
- Extensions
- Utilisateurs
- Outils
- Réglages
- Alecddd
- CPT Manager
- Réduire le menu

CPT Manager

Your Custom Post Types Add Custom Post Type Export

Custom Post Type Manager

Create as many Custom Post Types as you want.

Custom Post Type ID	product
Singular Name	Product
Plural Name	Products
Public	<input checked="" type="checkbox"/>
Archive	<input checked="" type="checkbox"/>

Enregistrer les modifications

Merci de faire de [WordPress](#) votre outil de création. Version 5.1.1

Custom Post Types - Mon site — +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_cpt

Mon site 0 Crée Bonjour, Patrick

CPT Manager

Réglages enregistrés.

Your Custom Post Types Add Custom Post Type Export

Manage Your Custom Post Types

ID	Singular Name	Plural Name	Public	Archive	Actions
comicbook	Comic Book	Comic books	TRUE	TRUE	EDIT - DELETE
product	Product	Products	FALSE	FALSE	EDIT - DELETE

Merci de faire de [WordPress](#) votre outil de création. Version 5.1.1

localhost / MySQL / wordpress — +

localhost/phpmyadmin/sql.php?server=1&db=wordpress&table=wp_options&pos=0

Serveur: MySQL 3306 » Base de données: wordpress » Table: wp_options

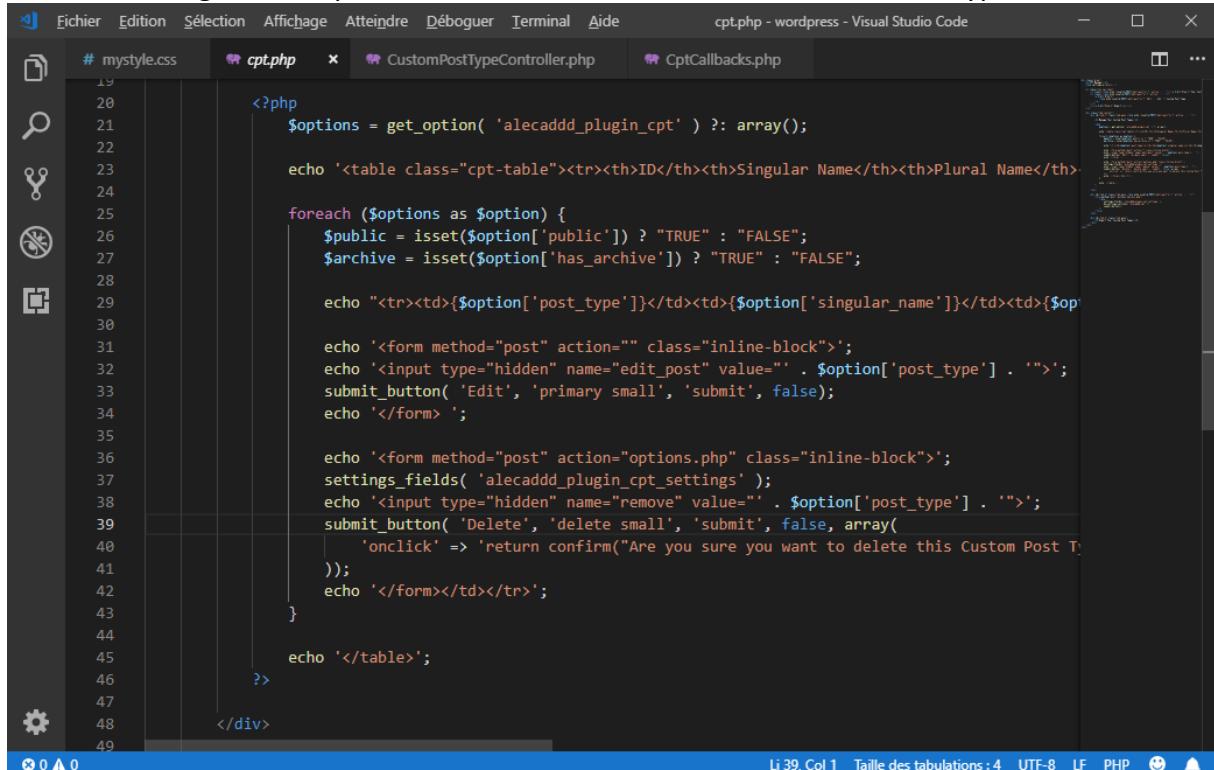
Parcourir Structure SQL Rechercher Insérer Exporter Importer Privileges Opérations Déclencheurs

Colonne	Type	Fonction	Null	Valeur
option_id	bigint(20) unsigned			732
option_name	varchar(191)			alecaddd_plugin_cpt
option_value	longtext			<pre>a:2:{s:9:"comicbook";a:5:{s:9:"post_type";s:9:"comicbook";s:13:"singular_name";s:10:"Comic Book";s:11:"plural_name";s:11:"Comic books";s:6:"public";s:1:"1";s:11:"has_archive";s:1:"1";s:7:"product";a:3:{s:9:"post_type";s:7:"product";s:13:"singular_name";s:7:"Product";s:11:"plural_name";s:8:"Products";}}</pre>
autoload	varchar(20)			yes

Exécuter

Console de requêtes SQL

WordPress Plugin Development - Part 30 - How to Delete a Custom Post Type

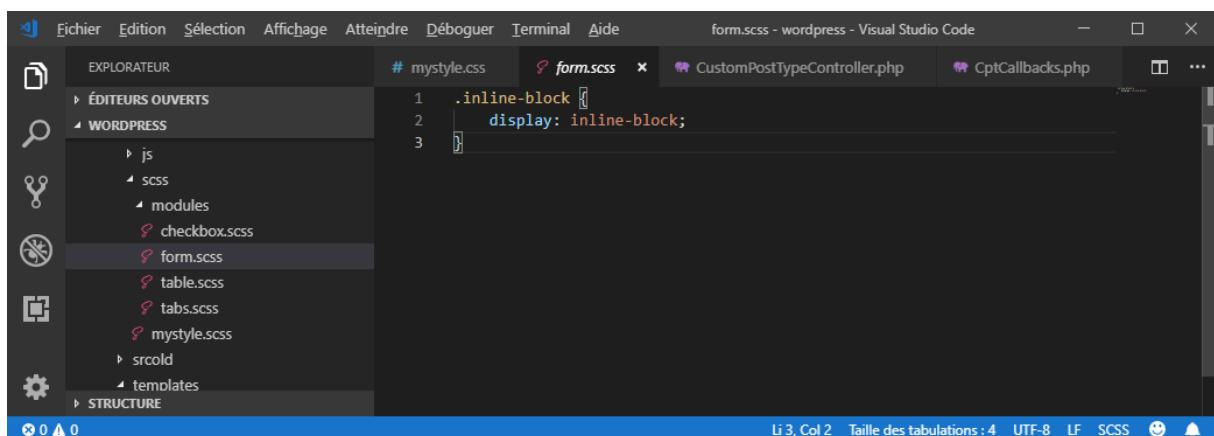


The screenshot shows a Visual Studio Code window with several tabs open. The active tab is 'cpt.php'. The code in the editor is as follows:

```
19 <?php
20     $options = get_option( 'alecaddd_plugin_cpt' ) ?: array();
21
22     echo '<table class="cpt-table"><tr><th>ID</th><th>Singular Name</th><th>Plural Name</th>';
23
24     foreach ($options as $option) {
25         $public = isset($option['public']) ? "TRUE" : "FALSE";
26         $archive = isset($option['has_archive']) ? "TRUE" : "FALSE";
27
28         echo "<tr><td>{$option['post_type']}

The status bar at the bottom indicates: Li 39, Col 1 Taille des tabulations : 4 UTF-8 LF PHP


```



The screenshot shows a Visual Studio Code window with several tabs open. The active tab is 'form.scss'. The code in the editor is as follows:

```
1 .inline-block {
2     display: inline-block;
3 }
```

The sidebar on the left shows the file structure under 'WORDPRESS': js, SCSS (modules, checkbox.scss, form.scss, table.scss, tabs.scss), mystyle.scss, scroll, templates.

The status bar at the bottom indicates: Li 3, Col 2 Taille des tabulations : 4 UTF-8 LF SCSS

gulp watch

The screenshot shows the WordPress CPT Manager interface. On the left, there's a sidebar with various menu items like Tableau de bord, Articles, Comic books, Products, Médias, Pages, Commentaires, Apparence, Extensions, Utilisateurs, Outils, Réglages, and Alecaddd (which is selected). Below that is a dashboard with links for Dashboard and CPT Manager, and a "Réduire le menu" button.

The main area is titled "CPT Manager" and "Manage Your Custom Post Types". It displays a table with two rows:

ID	Singular Name	Plural Name	Public	Archive	Actions
comicbook	Comic Book	Comic books	TRUE	TRUE	<button>Edit</button> <button>Delete</button>
product	Product	Products	FALSE	FALSE	<button>Edit</button> <button>Delete</button>

At the bottom, there's a message: "Merci de faire de [WordPress](#) votre outil de création." and "Version 5.1.1".

The screenshot shows Visual Studio Code with several tabs open: CptCallbacks.php, CustomPostTypeController.php, CptCallbacks.php, and cpt.php. The CptCallbacks.php tab contains the following PHP code:

```

public function cptSectionManager()
{
    echo 'Create as many Custom Post Types as you want.';
}

public function cptSanitize( $input )
{
    var_dump($_POST);
    die();

    $output = get_option('alecaddd_plugin_cpt');

    if ( count($output) == 0 ) {
        $output[$input['post_type']] = $input;
    }

    return $output;
}

foreach ($output as $key => $value) {
    if ($input['post_type'] === $key) {
        $output[$key] = $input;
    } else {
        $output[$input['post_type']] = $input;
    }
}

```

The status bar at the bottom indicates: Li 17, Col 5 Taille des tabulations : 4 UTF-8 LF PHP

Below the code editor, a browser window is open at <localhost/wordpress/wp-admin/>, showing the URL <localhost/wordpress/wp-admin/options.php>. The page content includes the following PHP code:

```

array(6) { ["option_page"]=> string(28) "alecaddd_plugin_cpt_settings" ["action"]=> string(6) "update" ["_wpnonce"]=> string(10) "b5dca4ac67" ["_wp_http_referer"]=> string(47) "/wordpress/wp-admin/admin.php?page=alecaddd_cpt" ["remove"]=> string(9) "comicbook" ["submit"]=> string(6) "Delete" }

```

```

Fichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide CptCallbacks.php - wordpress - Visual Studio Code
EXPLORATEUR # mystyle.css CustomPostTypeController.php CptCallbacks.php ...
ÉDITEURS OUVERTS WORDPRESS
    assets
    JS mystyle.js
    JS mystyle.js.map
    # mystyle.css
    # mystyle.css.map
    assetsold
    inc
        Api
        Callbacks
            AdminCallbacks.php
            CptCallbacks.php
            ManagerCallbacks.php
            SettingsApi.php
        Base
        Pages
        Init.php
    node_modules
    src
        JS
        SCSS
            modules
STRUCTURE
14
15     public function cptSanitize( $input )
16     {
17         $output = get_option('alecaddd_plugin_cpt');
18
19         if ( isset($_POST["remove"]) ) {
20             unset($output[$_POST["remove"]]);
21
22             return $output;
23         }
24
25         if ( count($output) == 0 ) {
26             $output[$input['post_type']] = $input;
27
28             return $output;
29         }
30
31         foreach ( $output as $key => $value ) {
32             if ( $input['post_type'] === $key ) {
33                 $output[$key] = $input;
34             } else {
35                 $output[$input['post_type']] = $input;
36             }
37         }
38
39     }
40
41
Li 20, Col 21 (5 sélectionné) Taille des tabulations : 4 UTF-8 LF PHP ⚡

```

Custom Post Types < Mon site — +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_cpt

Bonjour, Patrick

Tableau de bord

- Articles
- Products
- Médias
- Pages
- Commentaires
- Apparence
- Extensions
- Utilisateurs
- Outils
- Réglages
- Alecaddd

CPT Manager

Réglages enregistrés.

Your Custom Post Types Add Custom Post Type Export

Manage Your Custom Post Types

ID	Singular Name	Plural Name	Public	Archive	Actions
product	Product	Products	FALSE	FALSE	<button>Edit</button> <button>Delete</button>

localhost / MySQL / wordpress / +

localhost/phpmyadmin/index.php?db=wordpress&table=wp_options&target=sql.php

Serveur: MySQL_3306 » Base de données: wordpress » Table: wp_options

Colonne	Type	Fonction	Null	Valeur
option_id	bigint(20) unsigned			732
option_name	varchar(191)			alecaddd_plugin_cpt
				a:1:{s:7:"product";a:3:{s:9:"post_type";s:7:"product";s:13:"singular_name";s:7:"Product";s:11:"plural_name";s:8:"Products";}}
option_value	longtext			
autoload	varchar(20)			yes

Exécuter

Console de requêtes SQL

Custom Post Types < Mon site — +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_cpt

Mon site

Bonjour, Patrick

CPT Manager

Réglages enregistrés.

Your Custom Post Types Add Custom Post Type Export

Custom Post Type Manager

Create as many Custom Post Types as you want.

Custom Post Type ID	comicbook
Singular Name	Comic Book
Plural Name	Comic books
Public	<input checked="" type="checkbox"/>
Archive	<input checked="" type="checkbox"/>

Enregistrer les modifications

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

CPT Manager

Réglages enregistrés.

Your Custom Post Types Add Custom Post Type Export

Manage Your Custom Post Types

ID	Singular Name	Plural Name	Public	Archive	Actions
product	Product	Products	FALSE	FALSE	Edit Delete
comicbook	Comic Book	Comic books	FALSE	TRUE	Edit Delete

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

```

public function textField( $args )
{
    $name = $args['label_for'];
    $option_name = $args['option_name'];
    $value = '';

    if ( isset($_POST["edit_post"]) ) {
        $input = get_option( $option_name );
        $value = $input[$_POST["edit_post"]][$name];
    }

    echo '<input type="text" class="regular-text" id="' . $name . '" value="' . $value . '"';
}

public function checkboxField( $args )
{
    $name = $args['label_for'];
    $classes = $args['class'];
    $option_name = $args['option_name'];
    $checked = false;

    if ( isset($_POST["edit_post"]) ) {
        $checkbox = get_option( $option_name );
        $checked = isset($checkbox[$_POST["edit_post"]][$name]) ? true : false;
    }

    echo '<div class="' . $classes . '"><input type="checkbox" id="' . $name . '" checked="checked"' . ($checked ? ' checked="checked"' : '') . '>';
}

```

Custom Post Types < Mon site — +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_cpt

Bonjour, Patrick

Tableau de bord

Articles

Products

Comic books

Médias

Pages

Commentaires

Apparence

Extensions

Utilisateurs

Outils

Réglages

Alecaddd

Dashboard

CPT Manager

Réduire le menu

CPT Manager

Your Custom Post Types Edit Custom Post Type Export

Custom Post Type Manager

Create as many Custom Post Types as you want.

Custom Post Type ID: product

Singular Name: Product

Plural Name: Products

Public:

Archive:

Enregistrer les modifications

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

This screenshot shows the 'Edit Custom Post Type' screen for a custom post type named 'product'. The form includes fields for 'Custom Post Type ID' (set to 'product'), 'Singular Name' (set to 'Product'), and 'Plural Name' (set to 'Products'). There are also toggle switches for 'Public' (which is turned on) and 'Archive' (which is turned off). A blue button at the bottom left says 'Enregistrer les modifications' (Save changes).

Custom Post Types < Mon site — +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_cpt

Bonjour, Patrick

Tableau de bord

Articles

Products

Comic books

Médias

Pages

Commentaires

Apparence

Extensions

Utilisateurs

Outils

Réglages

Alecaddd

Dashboard

CPT Manager

Réduire le menu

CPT Manager

Your Custom Post Types Add Custom Post Type Export

Manage Your Custom Post Types

ID	Singular Name	Plural Name	Public	Archive	Actions
product	Product	Products	TRUE	FALSE	Edit Delete
comicbook	Comic Book	Comic books	FALSE	TRUE	Edit Delete

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

This screenshot shows the 'Manage Your Custom Post Types' screen. It lists two custom post types: 'product' (singular name 'Product', plural name 'Products', public status 'TRUE', archive status 'FALSE') and 'comicbook' (singular name 'Comic Book', plural name 'Comic books', public status 'FALSE', archive status 'TRUE'). Each row has 'Edit' and 'Delete' buttons in the 'Actions' column.

WordPress Plugin Development - Part 32 - Export PHP with Code Prettify

The screenshot shows the WordPress Admin interface for the 'Exporter' (Export) page. The left sidebar is dark-themed and includes links for 'Tableau de bord', 'Articles', 'Products', 'Comic books', 'Médias', 'Pages', 'Commentaires', 'Apparence', 'Extensions', 'Utilisateurs', 'Outils' (selected), 'Outils disponibles', 'Importer', 'Exporter' (selected), 'Exporter les données', 'Effacer les données', 'Règlages', and 'Alecaddd'. The main content area has a light background and displays the 'Exporter' section. It contains instructions about creating an XML file, a description of what the XML format contains (articles, pages, comments, custom fields, terms, menus, etc.), and a 'Choisir les données à exporter' (Select data to export) section. Under this section, 'Tout le contenu' (All content) is selected, and other options like 'Articles', 'Pages', 'Products', 'Comic books', and 'Médias' are available. A blue button labeled 'Télécharger le fichier d'export' (Download the export file) is at the bottom. At the bottom of the page, a note says 'Merci de faire de WordPress votre outil de création.' and 'Version 5.1.1'.

The screenshot shows the WordPress Admin interface for the 'CPT Manager' (Custom Post Types Manager) page. The left sidebar is dark-themed and includes links for 'Tableau de bord', 'Articles', 'Products', 'Comic books', 'Médias', 'Pages', 'Commentaires', 'Apparence', 'Extensions', 'Utilisateurs', 'Outils', and 'Règlages'. The 'Alecaddd' link is also present. The main content area has a light background and displays the 'CPT Manager' section. It includes tabs for 'Your Custom Post Types' (selected), 'Add Custom Post Type', and 'Export'. Below the tabs, there is a section titled 'Export Your Custom Post Types' with a large input field for pasting code. A note at the bottom of the page says 'Merci de faire de WordPress votre outil de création.'

The screenshot shows a web browser window titled "Generate WordPress Post Type". The URL is https://generatewp.com/post-type/. The page features a dark header with the "GenerateWP" logo and navigation links for "Generators", "Snippets", "Join", and "Login". Below the header, there's a large image of a computer screen displaying code. The main content area has a title "Post Type Generator". A horizontal menu bar includes "General", "Post Type", "Labels", "Options", "Visibility", "Query", "Permalinks", "Capabilities", and "Rest API". On the left, there's a sidebar with a bell icon. The central content is divided into three sections: "Overview", "Usage", and "Examples".

Overview

Use this tool to create custom code for Post Types with `register_post_type()` function.

Usage

- Fill in the user-friendly form.
- Click the “Update Code” button.
- Copy the code to your project.
- Or save it as a snippet and share with the community.

Examples

If you are still learning how to use this tool, check out the following examples:

- Products
- Articles
- Testimonials
- Property
- Car

The screenshot shows a web browser window titled "google/code-prettify: An embeddable script that makes source-code snippets in HTML prettier.". The URL is https://github.com/google/code-prettify. The page is a GitHub repository page for the "code-prettify" project. It features a dark header with the GitHub logo and navigation links for "Pull requests", "Issues", "Marketplace", and "Explore". The main content area shows the repository details: "Code", "Issues 164", "Pull requests 40", "Projects 0", "Insights", "Watch 194", "Star 4,606", "Fork 775". Below this, there's a description: "An embeddable script that makes source-code snippets in HTML prettier." A summary bar shows "471 commits", "3 branches", "17 releases", "25 contributors", and "Apache-2.0". A list of recent commits is displayed, each with a small icon, author, commit message, and timestamp.

Author	Commit Message	Time Ago
Stargateur and mikesamuel	Rust: Add missing dyn keyword, and add new reserved key word async, a...	13 days ago
distrib	Remove RawGit dependency. (#573)	a month ago
docs	Remove RawGit dependency. (#573)	a month ago
examples	Remove RawGit dependency. (#573)	a month ago
js-modules	Remove RawGit dependency. (#573)	a month ago
loader	Remove RawGit dependency. (#573)	a month ago
src	Rust: Add missing dyn keyword, and add new reserved key word async, a...	13 days ago
styles	add a redirect in demo page if loaded directly	2 years ago
tasks	Patch 2: grunt (#469)	2 years ago
tests	Kotlin support (#496)	2 years ago
tools/closure-compiler	Patch 2: grunt (#469)	2 years ago

code-prettify

0.1.0 • Public • Published 2 years ago

Readme 0 Dependencies 7 Dependents 1 Versions

JavaScript code prettifier

An embeddable script that makes source-code snippets in HTML prettier.

- Works on HTML pages.
- Works even if code contains embedded links, line numbers, etc.
- Simple API: include some JS & CSS and add an onload handler.
- Lightweights: small download and does not block page from loading while running.
- Customizable styles via CSS. See the [themes gallery](#).
- Supports all C-like, Bash-like, and XML-like languages. No need to specify the language.
- Extensible language handlers for other languages. You can specify the language.
- Widely used with good cross-browser support. Powers <https://code.google.com/> and <http://stackoverflow.com/>

[See an example.](#)

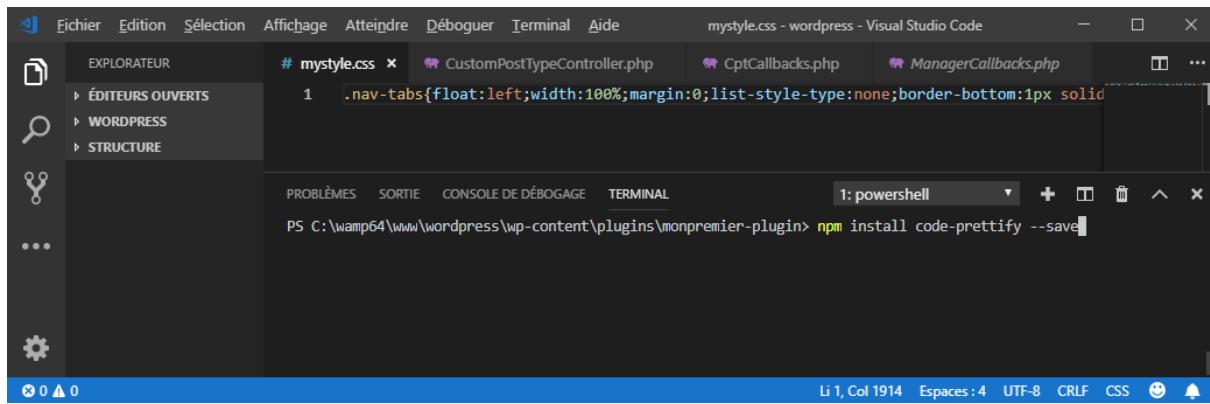
install

```
> npm i code-prettify
```

weekly downloads
1398

version 0.1.0	license Apache-2.0
open issues 164	pull requests 40
homepage github.com	repository github

npm install code-prettify



```

43     "presets": [
44         "env"
45     ],
46     "browserify": {
47         "transform": [
48             "browserify-shim"
49         ]
50     },
51     "browser": {
52         "jquery": "./node_modules/jquery/dist/jquery.js"
53     },
54     "browserify-shim": {
55         "jquery": "$"
56     },
57     "license": "GPL-3.0",
58     "dependencies": {
59         "code-prettify": "^0.1.0"
60     }
61 }

```

Li 18, Col 7 (22 sélectionné) Taille des tabulations : 4 UTF-8 LF JSON ⚡ 📁

Usage

Put code snippets in `<pre class="prettyprint">...</pre>` or `<code class="prettyprint">...</code>` and it will automatically be pretty-printed.

```

<pre class="prettyprint">class Voila {
public:
    // Voila
    static const string VOILA = "Voila";

    // will not interfere with embedded <a href="#voila2">tags</a>.
}</pre>

```

Update Code Save Snippet

```

// Register Custom Post Type
function custom_post_type() {

    $labels = array(
        'name'          => __( 'Post Types', 'Post Type General Name', 'text_domain' ),
        'singular_name' => __( 'Post Type', 'Post Type Singular Name', 'text_domain' ),
        'menu_name'     => __( 'Post Types', 'text_domain' ),
        'name_admin_bar'=> __( 'Post Type', 'text_domain' ),
        'archives'      => __( 'Item Archives', 'text_domain' ),
        'attributes'   => __( 'Item Attributes', 'text_domain' ),
        'parent_item_colon'=> __( 'Parent Item:', 'text_domain' ),
        'all_items'     => __( 'All Items', 'text_domain' ),
    )
}

```

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a tree view of files under "WORDPRESS". The "cpt.php" file is selected.
- Editor:** Displays the content of the "cpt.php" file. The code is as follows:

```

62     <?php foreach ($options as $option) { ?>
63         <h3><?php echo $option['singular_name']; ?></h3>
64         <pre class="prettyprint">
65             // Register Custom Post Type
66             function custom_post_type() {
67                 $labels = array(
68                     'name'          => _x( 'Post Types', 'Post Type General Name' ),
69                     'singular_name' => _x( '<?php echo $option["singular_name"]; ?>', 'Post Type Singular Name' ),
70                     'menu_name'      => __( '<?php echo $option["plural_name"]; ?>' ),
71                     'plural_name'   => __( '<?php echo $option["plural_name"]; ?>' ),
72                     'name_admin_bar'=> __( 'Post Type', 'text_domain' ),
73                 );
74             }
75         );
76     }

```

- Status Bar:** Shows "Li 127, Col 1 Taille des tabulations : 4 UTF-8 LF PHP".

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a tree view of files under "WORDPRESS". The "cpt.php" file is selected.
- Editor:** Displays the content of the "cpt.php" file. The code is as follows:

```

117     'publicly_queryable'    => true,
118     'capability_type'       => 'page',
119   );
120   register_post_type( '<?php echo $option["post_type"]; ?>', $args );
121 }
122 }
123 add_action( 'init', 'custom_post_type', 0 );
124
125 </pre>
126
127 <?php } ?>
128
129 </div>
130 </div>

```

- Status Bar:** Shows "Li 127, Col 1 Taille des tabulations : 4 UTF-8 LF PHP".

The screenshot shows a web browser displaying a GitHub page from the "code-prettify" repository. The page content includes:

- A heading "For example:" followed by a code snippet:

```

<script src="https://cdn.jsdelivr.net/gh/google/code-prettify@master/loader/run_prettify.js?lang=css&skin=sunburst"></script>

```

- A note explaining the code: "The above specifies the `lang` parameter to also load the CSS language extension and the `skin` parameter to load the `sunburst` skin."
- A section titled "Serving your own JS & CSS" with instructions: "You can [download](#) the scripts and styles and serve them yourself. Make sure to include both the script and a stylesheet:"

```

<link rel="stylesheet" type="text/css" href="prettify.css">
<script type="text/javascript" src="prettify.js"></script>

```

- A note: "Then run the `PR.prettyPrint()` function once your page has finished loading. One way to do this is via the `onload` handler thus:"

```

<body onload="PR.prettyPrint()">

```

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure under "WORDPRESS". The "src/js" folder is expanded, showing files like "myscriptjs", "myscriptoldjs", "scss", "srcold", "templates", "vendor", "alecaddd-plugin.php", "composer.json", "gulpfile.js", "index.php", "package-lock.json", and "package.json".
- Editor (Center):** Displays the content of the "myscriptjs" file.
- Status Bar (Bottom):** Shows "Li 5, Col 7 (15 sélectionné)" and "Taille des tabulations : 4" followed by file encoding information: "UTF-8 LF JavaScript".

```

1 import 'code-prettify';
2
3 window.addEventListener("load", function() {
4
5     PR.prettyPrint();
6
7     // store tabs variables
8     var tabs = document.querySelectorAll("ul.nav-tabs > li");
9
10    for (var i = 0; i < tabs.length; i++) {
11        tabs[i].addEventListener("click", switchTab);
12    }
13
14    function switchTab(event) {
15        event.preventDefault();
16
17        document.querySelector("ul.nav-tabs li.active").classList.remove("active");
18        document.querySelector(".tab-pane.active").classList.remove("active");
19
20        var clickedTab = event.currentTarget;
21        var anchor = event.target;
22        var activePaneID = anchor.getAttribute("href");
23
24        clickedTab.classList.add("active");
25        document.querySelector(activePaneID).classList.add("active");
26    }
27
28 });
29 });

```

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure under "WORDPRESS". The "src/styles" folder is expanded, showing files like "desert.css", "doxy.css", "sons-of-obsidian.css", "sunburst.css", "CHANGES.md", "COPYING", "package.json", and "README.md".
- Editor (Center):** Displays the content of the "mystyle.css" file.
- Status Bar (Bottom):** Shows "Li 1, Col 1" and "Espaces : 4" followed by file encoding information: "UTF-8 LF SCSS".

```

1 @import '../node_modules/code-prettify/styles/desert.css';
2
3 @import 'modules/tabs';
4 @import 'modules/checkbox';
5 @import 'modules/table';
6 @import 'modules/form';

```

The screenshot shows the WordPress CPT Manager interface. On the left, there's a sidebar with various menu items like Articles, Products, Comic books, Médias, Pages, Commentaires, Apparence, Extensions, Utilisateurs, Outils, Réglages, and a selected item 'Alecaddd'. The main area has tabs for 'Your Custom Post Types', 'Add Custom Post Type', and 'Export' (which is highlighted). Below that, it says 'Export Your Custom Post Types' and 'Product'. A code block shows the PHP code for registering a custom post type:

```
// Register Custom Post Type
function custom_post_type() {
    $labels = array(
        'name'                  => __x( 'Post Types', 'Post Type General Name', 'text_domain' ),
        'singular_name'          => __x( 'Product', 'Post Type Singular Name', 'text_domain' ),
        'menu_name'              => ___( 'Products', 'text_domain' ),
        'plural_name'             => ___( 'Products', 'text_domain' ),
        'name_admin_bar'          => ___( 'Post Type', 'text_domain' ),
        'archives'                => ___( 'Item Archives', 'text_domain' ),
        'attributes'              => ___( 'Item Attributes', 'text_domain' ),
        'parent_item_colon'       => ___( 'Parent Item:', 'text_domain' ),
        'all_items'                => ___( 'All Items', 'text_domain' ),
        'add_new_item'              => ___( 'Add New Item', 'text_domain' ),
        'add_new'                    => ___( 'Add New', 'text_domain' ),
        'new_item'                     => ___( 'New Item', 'text_domain' ),
        'edit_item'                      => ___( 'Edit Item', 'text_domain' ),
        'update_item'                   => ___( 'Update Item', 'text_domain' ),
        'view_item'                      => ___( 'View Item', 'text_domain' ),
        'view_items'                     => ___( 'View Items', 'text_domain' ),
        'search_items'                   => ___( 'Search Item', 'text_domain' ),
        'not_found'                      => ___( 'Not found', 'text_domain' ),
        'not_found_in_trash'           => ___( 'Not found in Trash', 'text_domain' )
    );
}
```

WordPress Plugin Development - Part 33 - Create a Custom Taxonomy Manager

The screenshot shows a Visual Studio Code editor window. The title bar says 'TaxonomyCallbacks.php - wordpress - Visual Studio Code'. The left sidebar shows a file tree with a folder named 'WORDPRESS' containing 'plugins', 'akismet', 'monpremier-plugin', 'assetsold', 'inc', 'Api', 'Callbacks', and several PHP files like 'AdminCallbacks.php', 'CptCallbacks.php', 'ManagerCallbacks.php', 'TaxonomyCallbacks.php', 'SettingsApi.php', 'Base.php', 'Activate.php', 'AuthController.php', 'BaseController.php', 'ChatController.php', 'CustomPostTypeController.php', 'CustomTaxonomyController.php', 'Deactivate.php', 'Enqueue.php', 'GalleryController.php', 'MembershipController.php', 'SettingsLinks.php', 'TemplateController.php', and 'TestimonialController.php'. The main editor area displays the 'TaxonomyCallbacks.php' file:

```
<?php
/**
 * @package AlecadddPlugin
 */
namespace Inc\Api\Callbacks;

class TaxonomyCallbacks
{
    public function taxSectionManager() {
        echo 'Create as many Custom Taxonomies as you want.';
    }

    public function taxSanitize( $input ) {
        $output = get_option('alecaddd_plugin_tax');

        if ( isset($_POST["remove"]) ) {
            unset($output[$_POST["remove"]]);
        }

        return $output;
    }

    if ( count($output) == 0 ) {
        $output[$input['taxonomy']] = $input;
    }

    return $output;
}

foreach ( $output as $key => $value ) {
    if ( $input['taxonomy'] === $key ) {
        $output[$key] = $input;
    } else {
        $output[$input['taxonomy']] = $input;
    }
}

return $output;
}
```

At the bottom, there are status icons for file changes, a terminal tab, and other development tools.

```

# mystyle.css
# TaxonomyCallbacks.php x
$output[$input['taxonomy']] = $input;
}

return $output;
}

public function textField( $args )
{
    $name = $args['label_for'];
    $option_name = $args['option_name'];
    $value = '';

    if ( isset($_POST["edit_taxonomy"]) ) {
        $input = get_option( $option_name );
        $value = $input[$_POST["edit_taxonomy"]][$name];
    }

    echo '<input type="text" class="regular-text" id="' . $name . '" name='
}

public function checkboxField( $args )
{
    $name = $args['label_for'];
    $classes = $args['class'];
    $option_name = $args['option_name'];
    $checked = false;

    if ( isset($_POST["edit_taxonomy"]) ) {
        $checkbox = get_option( $option_name );
        $checked = isset($checkbox[$_POST["edit_taxonomy"]][$name]) ?: false;
    }

    echo '<div class="' . $classes . '"><input type="checkbox" id="' . $name . '" name='
}

```

```

# mystyle.css
# TaxonomyCallbacks.php x
# CustomTaxonomyController.php x
# taxonomy.php x
<?php
/**
 * @package AlecadddPlugin
 */
namespace Inc\Base;

use Inc\Api\SettingsApi;
use Inc\Base\BaseController;
use Inc\Api\Callbacks\AdminCallbacks;
use Inc\Api\Callbacks\TaxonomyCallbacks;

/**
 *
 */
class CustomTaxonomyController extends BaseController
{
    public $settings;

    public $callbacks;

    public $tax_callbacks;

    public $subpages = array();

    public $taxonomies = array();
}

```

Eichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide CustomTaxonomyController.php - wordpress - Visual Studio C... □ X

EXPLORATEUR # mystyle.css TaxonomyCallbacks.php CustomTaxonomyController.php x taxonomy.php □ ...

ÉDITEURS OUVERTS

WORDPRESS

- assetsold
- inc
 - Api
 - Callbacks
 - AdminCallbacks.php
 - CptCallbacks.php
 - ManagerCallbacks.php
 - TaxonomyCallbacks.php
 - SettingsApi.php
- Base
- Activate.php
- AuthController.php
- BaseController.php
- ChatController.php
- CustomPostTypeController.php
- CustomTaxonomyController.php
- Deactivate.php
- Enqueue.php
- GalleryController.php
- MembershipController.php
- SettingsLinks.php
- TemplateController.php
- TestimonialController.php
- WidgetController.php
- Pages
- Init.php
- incold
- node_modules
- src
 - js
 - scss
 - modules

STRUCTURE

```

public function register()
{
    if ( ! $this->activated( 'taxonomy_manager' ) ) return;

    $this->settings = new SettingsApi();

    $this->callbacks = new AdminCallbacks();

    $this->tax_callbacks = new TaxonomyCallbacks();

    $this->setSubpages();

    $this->setSettings();

    $this->setSections();

    $this->setFields();

    $this->settings->addSubPages( $this->subpages )->register();

    $this->storeCustomTaxonomies();

    if ( ! empty( $this->taxonomies ) ) {
        add_action( 'init', array( $this, 'registerCustomTaxonomy' ) );
    }
}

public function setSubpages()
{
    $this->subpages = array(
        array(
            'parent_slug' => 'alecaddd_plugin',
            'page_title' => 'Custom Taxonomies',
            'menu_title' => 'Taxonomy Manager',
            'capability' => 'manage_options',
            'menu_slug' => 'alecaddd_taxonomy',
            'callback' => array( $this->callbacks, 'adminTaxonomy' )
        )
    );
}

```

Li 12, Col 4 Taille des tabulations : 4 UTF-8 LF PHP 🎉 1

Eichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide CustomTaxonomyController.php - wordpress - Visual Studio C... □ X

EXPLORATEUR # mystyle.css TaxonomyCallbacks.php CustomTaxonomyController.php x taxonomy.php □ ...

ÉDITEURS OUVERTS

WORDPRESS

- assetsold
- inc
 - Api
 - Callbacks
 - AdminCallbacks.php
 - CptCallbacks.php
 - ManagerCallbacks.php
 - TaxonomyCallbacks.php
 - SettingsApi.php
- Base
- Activate.php
- AuthController.php
- BaseController.php
- ChatController.php
- CustomPostTypeController.php
- CustomTaxonomyController.php
- Deactivate.php
- Enqueue.php
- GalleryController.php
- MembershipController.php
- SettingsLinks.php

STRUCTURE

```

public function setSettings()
{
    $args = array(
        array(
            'option_group' => 'alecaddd_plugin_tax_settings',
            'option_name' => 'alecaddd_plugin_tax',
            'callback' => array($this->tax_callbacks, 'taxSanitize')
        )
    );
    $this->settings->setSettings( $args );
}

public function setSections()
{
    $args = array(
        array(
            'id' => 'alecaddd_tax_index',
            'title' => 'Custom Taxonomy Manager',
            'callback' => array($this->tax_callbacks, 'taxSectionManager'),
            'page' => 'alecaddd_taxonomy'
        )
    );
    $this->settings->setSections( $args );
}

```

Li 60, Col 53 Taille des tabulations : 4 UTF-8 LF PHP 🎉 1

```
public function setFields()
{
    $args = array(
        array(
            'id' => 'taxonomy',
            'title' => 'Custom Taxonomy ID',
            'callback' => array($this->tax_callbacks, 'textField'),
            'page' => 'alecaddd_taxonomy',
            'section' => 'alecaddd_tax_index',
            'args' => array(
                'option_name' => 'alecaddd_plugin_tax',
                'label_for' => 'taxonomy',
                'placeholder' => 'eg. genre',
                'array' => 'taxonomy'
            )
        ),
        array(
            'id' => 'singular_name',
            'title' => 'Singular Name',
            'callback' => array( $this->tax_callbacks, 'textField' ),
            'page' => 'alecaddd_taxonomy',
            'section' => 'alecaddd_tax_index',
            'args' => array(
                'option_name' => 'alecaddd_plugin_tax',
                'label_for' => 'singular_name',
                'placeholder' => 'eg. Genre',
                'array' => 'taxonomy'
            )
        ),
        array(
            'id' => 'hierarchical',
            'title' => 'Hierarchical',
            'callback' => array( $this->tax_callbacks, 'checkboxField' ),
            'page' => 'alecaddd_taxonomy',
            'section' => 'alecaddd_tax_index',
            'args' => array(
                'option_name' => 'alecaddd_plugin_tax',
                'label_for' => 'hierarchical',
                'class' => 'ui-toggle',
                'array' => 'taxonomy'
            )
        )
    );
    $this->settings->setFields( $args );
}
```

The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure under "WORDPRESS". Key files listed include: myscript.js, myscript.js.map, mystyle.css, mystyle.css.map, assetsold, inc (containing Api, Callbacks, AdminCallbacks.php, CptCallbacks.php, ManagerCallbacks.php, TaxonomyCallbacks.php, SettingsApi.php), Base (containing Activate.php, AuthController.php, BaseController.php, ChatController.php, CustomPostTypeControll..., CustomTaxonomyContro..., Deactivate.php, Enqueue.php, GalleryController.php, MembershipController.p..., SettingsLinks.php, TemplateController.php, TestimonialController.php, WidgetController.php), Pages, Init.php, incold, node_modules, src, and is.
- Code Editor (Center):** Displays the content of `CustomTaxonomyController.php`. The code defines two methods: `storeCustomTaxonomies()` and `registerCustomTaxonomy()`. The `storeCustomTaxonomies()` method reads options from `'alecaddd_plugin_tax'`, stores them into an array of labels, and then creates a taxonomy object with various properties like hierarchical, labels, show_ui, etc. The `registerCustomTaxonomy()` method iterates over the taxonomy objects and calls `register_taxonomy()` for each.
- Bottom Status Bar:** Shows the current position as "L1 124, Col 19", the tabulation size as "Taille des tabulations : 4", the encoding as "UTF-8", the line separator as "LF", the file type as "PHP", and a notification icon with the number "1".

The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** On the left, showing icons for mystyle.css, TaxonomyCallbacks.php, CustomTaxonomyController.php, taxonomy.php (the active tab), and Init.php.
- Code Editor:** The main area displays the content of the taxonomy.php file. The code is a PHP script that generates an HTML page for managing custom taxonomies. It includes sections for tabs (tab-1, tab-2, tab-3) and a table for listing taxonomies with columns for ID, Singular Name, and a form for editing or deleting them.
- Status Bar:** At the bottom, it shows "Li 64, Col 7 Espaces : 4 UTF-8 LF PHP" and a notification icon with "1".

```
1 <div class="wrap">
2   <h1>Taxonomy Manager</h1>
3   <?php settings_errors(); ?>
4
5   <ul class="nav nav-tabs">
6     <li class="<?php echo !isset($_POST["edit_taxonomy"]) ? 'active' : '' ?>"><a href="#tab-1">Your 1</a>
7     <li class="<?php echo isset($_POST["edit_taxonomy"]) ? 'active' : '' ?>">
8       <a href="#tab-2">
9         <?php echo isset($_POST["edit_taxonomy"]) ? 'Edit' : 'Add' ?> Taxonomy
10      </a>
11    </li>
12    <li><a href="#tab-3">Export</a></li>
13  </ul>
14
15  <div class="tab-content">
16    <div id="tab-1" class="tab-pane <?php echo !isset($_POST["edit_taxonomy"]) ? 'active' : '' ?>">
17
18      <h3>Manage Your Custom Taxonomies</h3>
19
20      <?php
21        $options = get_option( 'alecaddd_plugin_tax' ) ?: array();
22
23        echo '<table class="cpt-table"><tr><th>ID</th><th>Singular Name</th><th class="text-center">Actions</th></tr>';
24
25        foreach ($options as $option) {
26          $hierarchical = isset($option['hierarchical']) ? "TRUE" : "FALSE";
27
28          echo "<tr><td>{$option['taxonomy']}</td><td>{$option['singular_name']}</td><td class='text-center'>";
29
30          echo '<form method="post" action="" class="inline-block">';
31          echo '<input type="hidden" name="edit_taxonomy" value="' . $option['taxonomy'] . '">';
32          submit_button( 'Edit', 'primary small', 'submit', false );
33          echo '</form> ';
34
35          echo '<form method="post" action="options.php" class="inline-block">';
36          settings_fields( 'alecaddd_plugin_tax_settings' );
37          echo '<input type="hidden" name="remove" value="' . $option['taxonomy'] . '">';
38          submit_button( 'Delete', 'delete small', 'submit', false, array(
39            'onclick' => 'return confirm("Are you sure you want to delete this Custom Taxonomy?")'
40          ));
41          echo '</form></td></tr>';
42
43        }
44
45      echo '</table>';
46
47    </div>
48  </div>
```

The screenshot shows the WordPress admin dashboard with the Alecaddd Plugin settings page open. The left sidebar has a dark theme with various menu items like Tableau de bord, Articles, Products, Comic books, Médias, Pages, Commentaires, Apparence, Extensions, Utilisateurs, Outils, Réglages, and Alecaddd. The Alecaddd item is highlighted in blue. The main content area has a light gray background. At the top, there are tabs for Manage Settings, Updates, and About. Below that, a section titled "Settings Manager" contains a list of features with toggle switches. Most features have their toggle switched to the "on" position, indicated by a blue circle. The features listed are: Activate CPT Manager, Activate Taxonomy Manager, Activate Media Widget, Activate Gallery Manager, Activate Testimonial Manager, Activate Templates Manager, Activate Ajax Login/Signup, Activate Membership Manager, and Activate Chat Manager. At the bottom of this section is a blue button labeled "Enregistrer les modifications". At the very bottom of the page, there is a footer note "Merci de faire de WordPress votre outil de création." and the text "Version 5.1.1".

Alecaddd Plugin < Mon site — WordPress

Bonjour, Patrick

Tableau de bord

Articles

Products

Comic books

Médias

Pages

Commentaires

Apparence

Extensions

Utilisateurs

Outils

Réglages

Alecaddd

Dashboard

CPT Manager

Réduire le menu

Alecaddd Plugin

Manage Settings Updates About

Settings Manager

Manage the Sections and Features of this Plugin by activating the checkboxes from the following list.

Activate CPT Manager

Activate Taxonomy Manager

Activate Media Widget

Activate Gallery Manager

Activate Testimonial Manager

Activate Templates Manager

Activate Ajax Login/Signup

Activate Membership Manager

Activate Chat Manager

Enregistrer les modifications

Merci de faire de WordPress votre outil de création.

Version 5.1.1

The screenshot shows the WordPress Taxonomy Manager page. The left sidebar is dark with white icons and text, showing links like Tableau de bord, Articles, Products, Comic books, Médias, Pages, Commentaires, Apparence, Extensions, Utilisateurs, Outils, Réglages, and Alecaddd (which is highlighted). Below these are Dashboard, CPT Manager, and Taxonomy Manager. At the bottom of the sidebar are Réduire le menu and a link to the WordPress creation tool. The main area has a light gray header with tabs: Your Taxonomies (selected), Add Taxonomy, and Export. Below the header is a section titled "Manage Your Custom Taxonomies" containing a table with columns: ID, Singular Name, Hierarchical, and Actions. A message at the bottom left says "Merci de faire de WordPress votre outil de création." and the version is listed as "Version 5.1.1".

This screenshot shows the "Add Taxonomy" form in the Taxonomy Manager. The left sidebar is identical to the previous screenshot. The main area has a light gray header with tabs: Your Taxonomies (disabled), Add Taxonomy (selected), and Export. Below the header is a section titled "Custom Taxonomy Manager" with the sub-instruction "Create as many Custom Taxonomies as you want." It contains three input fields: "Custom Taxonomy ID" with the value "genre", "Singular Name" with the value "Genre", and a "Hierarchical" toggle switch which is turned on. At the bottom is a blue button labeled "Enregistrer les modifications" (Save changes). A message at the bottom left says "Merci de faire de WordPress votre outil de création." and the version is listed as "Version 5.1.1".

localhost / MySQL / wordpress / +

localhost/phpmyadmin/sql.php?server=1&db=wordpress&table=wp_options&pos=0

Server: MySQL 3.306 » Base de données: wordpress » Table: wp_options

	Parcourir	Structure	SQL	Rechercher	Insérer	Exporter	Importer	Priviléges	Opérations	Déclencheurs
	option_id	option_name		option_value	autoload					
<input type="checkbox"/>	110	widget_nav_menu	Éditer Copier Supprimer	a:1:{s:12:"_multiwidget";i:1;}	yes					
<input type="checkbox"/>	111	widget_custom_html	Éditer Copier Supprimer	a:1:{s:12:"_multiwidget";i:1;}	yes					
<input type="checkbox"/>	112	cron	Éditer Copier Supprimer	a:5:{i:1556022487;a:1:{s:34:"wp_privacy_delete_old...";i:-1;}}	yes					
<input type="checkbox"/>	113	theme_mods_twentynineteen	Éditer Copier Supprimer	a:1:{s:18:"custom_css_post_id";i:-1;}	yes					
<input type="checkbox"/>	125	_site_transient_update_core	Éditer Copier Supprimer	O:8:"stdClass":4:{s:7:"updates";a:1:{i:0;O:8:"stdClass":4:{s:11:"last_checked";i:1556018893;}};	no					
<input type="checkbox"/>	126	can_compress_scripts	Éditer Copier Supprimer	1	no					
<input type="checkbox"/>	127	_site_transient_update_themes	Éditer Copier Supprimer	O:8:"stdClass":4:{s:12:"last_checked";i:1556018893;}	no					
<input type="checkbox"/>	130	_site_transient_update_plugins	Éditer Copier Supprimer	O:8:"stdClass":5:{s:12:"last_checked";i:1556018893;}	no					
<input type="checkbox"/>	144	recently_activated	Éditer Copier Supprimer	a:1:{s:9:"hello.php";i:1556020431;}	yes					
<input type="checkbox"/>	231	text_example	Éditer Copier Supprimer	Coucou	yes					
<input type="checkbox"/>	232	first_name	Éditer Copier Supprimer	Patrick	yes					
<input type="checkbox"/>	462	alecadd plugin	Éditer Copier Supprimer	a:9:{s:11:"cpt_manager";b:1;s:16:"taxonomy_manager...";i:1;}	yes					
<input type="checkbox"/>	663	_site_transient_timeout_browser_127868b9556d0b7328...	Éditer Copier Supprimer	1556028134	no					
<input type="checkbox"/>	664	_site_transient_browser_127868b9556d0b7328ae4585e...	Éditer Copier Supprimer	a:10:{s:4:"name";s:6:"Chrome";s:7:"version";s:13:"7.0.3615.65";s:11:"platform";s:10:"Windows 10";s:10:"userAgent";s:32:"Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/7.0.3615.65 Safari/537.36";s:11:"acceptLanguage";s:18:"fr-FR,fr;q=0.9,en-US;q=0.8,en;q=0.7";s:10:"accept";s:11:"text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8";s:12:"acceptEncoding";s:12:"gzip, deflate";s:13:"acceptCharset";s:12:"utf-8,*";s:14:"referrerPolicy";s:11:"no-referrer";s:15:"crossOrigin";s:11:"use-credentials";s:16:"isSecure";b:1;};	no					
<input type="checkbox"/>	732	alecadd plugin cpt	Éditer Copier Supprimer	a:2:{s:7:"product";a:4:{s:9:"post_type";s:7:"produ...";s:11:"category";s:11:"category-type";s:10:"taxonomies";s:11:"taxonomies-type";};s:11:"cpt_manager";b:1;};	yes					
<input type="checkbox"/>	733	alecadd plugin tax	Éditer Copier Supprimer	a:0:{}	yes					

Tout cocher Avec la sélection : [Éditer](#) [Copier](#) [Supprimer](#) [Exporter](#)

<< < 5 > >> Tout afficher Nombre de lignes : 25 Filtrer les lignes : Chercher dans cette table Trier sur l'index: PRIMARY (ASC)

Opérations sur les résultats de la requête

Imprimer | Copier dans le presse-papiers | Consule de requêtes SQL | Exporter | Afficher le graphique | Créer une vue

Je crée l'enregistrement 733 si nécessaire

The screenshot shows the WordPress Taxonomy Manager page. The left sidebar has a blue header for 'Alecadd' and 'Taxonomy Manager'. The main content area has a title 'Taxonomy Manager' and a message 'Réglages enregistrés.' Below it are tabs for 'Your Taxonomies' (selected), 'Add Taxonomy', and 'Export'. A table titled 'Manage Your Custom Taxonomies' lists one item: 'genre' with 'Genre' as the singular name and 'TRUE' as the hierarchical status. Action buttons 'Edit' and 'Delete' are to the right of the table. The bottom of the page includes a footer with 'Merci de faire de WordPress votre outil de création.' and 'Version 5.1.1'.

Custom Taxonomies < Mon site – +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_taxonomy

Mon site 0 + Crée Bonjour, Patrick

Taxonomy Manager

Réglages enregistrés.

Your Taxonomies Add Taxonomy Export

Manage Your Custom Taxonomies

ID	Singular Name	Hierarchical	Actions
genre	Genre	TRUE	Edit Delete

Alecadd

Dashboard CPT Manager Taxonomy Manager Réduire le menu

Merci de faire de WordPress votre outil de création.

Version 5.1.1

Custom Taxonomies < Mon site - +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_taxonomy

Mon site 0 + Crée Bonjour, Patrick

Tableau de bord Articles Products Comic books Médias Pages Commentaires Apparence Extensions Utilisateurs Outils Réglages Alecaddd Dashboard CPT Manager Taxonomy Manager Réduire le menu

Taxonomy Manager

Your Taxonomies Edit Taxonomy Export

Custom Taxonomy Manager

Create as many Custom Taxonomies as you want.

Custom Taxonomy ID: genre

Singular Name: Genre

Hierarchical:

Post Types:

- post
- page
- attachment
- wp_block
- product
- comicbook

Enregistrer les modifications

Merci de faire de [WordPress](#) votre outil de création. Version 5.1.1

The screenshot shows the 'Taxonomy Manager' page in the WordPress admin area. A new taxonomy named 'genre' is being created. The 'Hierarchical' option is selected. The 'Post Types' section lists 'post' as the only selected type. A blue button at the bottom says 'Enregistrer les modifications' (Save changes).

Custom Taxonomies < Mon site - +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_taxonomy

Mon site 0 + Crée Bonjour, Patrick

Tableau de bord Articles Products Comic books Médias Pages Commentaires Apparence Extensions Alecaddd Tous les articles Ajouter Catégories Étiquettes Genre

Taxonomy Manager

Add Taxonomy Export

Custom Taxonomies

Singular Name	Hierarchical	Actions
genre	Genre	Edit Delete

localhost/wordpress/wp-admin/edit-tags.php?taxonomy=genre

The screenshot shows the 'Taxonomy Manager' page again, but now the 'genre' taxonomy is listed in the 'Custom Taxonomies' table. It has a singular name of 'genre' and is set to be hierarchical. There are 'Edit' and 'Delete' buttons for this entry.

The screenshot shows the WordPress admin dashboard with the title "Genre < Mon site — WordPress". The left sidebar is visible with various menu items like "Articles", "Genre", "Products", "Comic books", etc. The main content area is titled "Genre" and shows the "Add New Genre" form. The "Nom" field is empty. A note below it says "Ce nom est utilisé un peu partout sur votre site.". The "Slug" field is also empty. A note below it says "Le slug est la version normalisée du nom. Il ne contient généralement que des lettres minuscules non accentuées, des chiffres et des traits d'union.". The "Parent Genre" dropdown is set to "Aucun". A note below it says "Assignez un terme parent pour créer une hiérarchie. Par exemple, le terme Jazz serait le parent de Bebop et Big Band.". The "Description" text area is empty. A note below it says "La description n'est pas très utilisée par défaut, cependant de plus en plus de thèmes l'affichent.". At the bottom of the form is a blue "Add New Genre" button. To the right of the form, there are two tables for existing genres. The top table has columns: Nom, Description, Slug, and Total. It shows one entry: "Aucune catégorie trouvée.". The bottom table has the same columns and shows one entry: "Aucun". There are "Actions groupées" and "Appliquer" buttons above each table. At the bottom of the page, a footer note says "Merci de faire de [WordPress](#) votre outil de création." and "Version 5.1.1".

Custom Taxonomies < Mon site - +

localhost/wordpress/wp-admin/admin.php?page=alecaddd_taxonomy

Mon site Crée Bonjour, Patrick

Tableau de bord Articles Products Comic books Médias Pages Commentaires Apparence Extensions Utilisateurs Outils Réglages Alecaddd Dashboard CPT Manager Taxonomy Manager Réduire le menu

Taxonomy Manager

Your Taxonomies Edit Taxonomy Export

Custom Taxonomy Manager

Create as many Custom Taxonomies as you want.

Custom Taxonomy ID: genre

Singular Name: Genre

Hierarchical:

Post Types:

- post
- page
- attachment
- wp_block
- product
- comicbook

Enregistrer les modifications

Merci de faire de [WordPress](#) votre outil de création. Version 5.1.1

The screenshot shows the 'Taxonomy Manager' page in the WordPress admin area. A sidebar on the left lists various site management options. The main panel is titled 'Custom Taxonomy Manager' and contains fields for 'Custom Taxonomy ID' (set to 'genre'), 'Singular Name' (set to 'Genre'), and a 'Hierarchical' toggle switch which is turned on. Below these are several 'Post Types' checkboxes: 'post' (on), 'page' (off), 'attachment' (off), 'wp_block' (off), 'product' (on), and 'comicbook' (on). At the bottom is a blue 'Enregistrer les modifications' button. A note at the bottom of the page says 'Merci de faire de [WordPress](#) votre outil de création.' and indicates the version is 'Version 5.1.1'.

localhost / MySQL / wordpress / +

localhost/phpmyadmin/sql.php?server=1&db=wordpress&table=wp_options&pos=0

Serveur: MySQL 3306 » Base de données: wordpress » Table: wp_options

Parcourir Structure SQL Rechercher Insérer Exporter Importer Priviléges Opérations Plus

Colonne	Type	Fonction	Null	Valeur
option_id	bigint(20) unsigned			733
option_name	varchar(191)			alecaddd_plugin_tax
option_value	longtext			<pre>a:1:{s:5:"genre";a:4:{s:8:"Taxonomy";s:5:"genre";s:13:"singular_name";s:5:"Genre";s:12:"hierarchical";s:1:"1";s:7:"objects";a:3:{s:4:"post";s:1:"1";s:7:"product";s:1:"1";s:9:"comicbook";s:1:"1";}}}</pre>

Console de requêtes SQL

The screenshot shows the 'wp_options' table in the 'wordpress' database via phpMyAdmin. The table has four columns: 'option_id', 'option_name', 'option_value', and 'Fonction'. There are two rows. The first row has 'option_id' value 733 and 'option_name' empty. The second row has 'option_name' value 'alecaddd_plugin_tax' and 'option_value' containing a serialized array: 'a:1:{s:5:"genre";a:4:{s:8:"Taxonomy";s:5:"genre";s:13:"singular_name";s:5:"Genre";s:12:"hierarchical";s:1:"1";s:7:"objects";a:3:{s:4:"post";s:1:"1";s:7:"product";s:1:"1";s:9:"comicbook";s:1:"1";}}}'.

```

    182     );
    183
    184     }
    185     }
    186
    187     public function registerCustomTaxonomy()
    188     {
    189         foreach ($this->taxonomies as $taxonomy)
    190         {
    191             $objects = isset($taxonomy['objects']) ? array_keys($taxonomy['objects']) : null;
    192             register_taxonomy( $taxonomy['rewrite']['slug'], $objects, $taxonomy );
    193         }
    194     }

```

Custom Taxonomies < Mon site - +

localhost/wordpress/wp-admin/admin.php?page=alecadddd_taxonomy

Bonjour, Patrick

Taxonomy Manager

Your Taxonomies Edit Taxonomy Export

Custom Taxonomy Manager

Create as many Custom Taxonomies as you want.

Custom Taxonomy ID	<input type="text" value="chapter"/>
Singular Name	<input type="text" value="Chapter"/>
Hierarchical	<input checked="" type="checkbox"/>
Post Types	<input checked="" type="checkbox"/> post <input type="checkbox"/> page <input type="checkbox"/> attachment <input type="checkbox"/> wp_block <input type="checkbox"/> product <input type="checkbox"/> comicbook

Enregistrer les modifications

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

The screenshot shows the WordPress Taxonomy Manager interface. On the left, a sidebar lists various site components: Articles, Products, Comic books, Médias, Pages, Commentaires, Apparence, and Extensions. The 'Articles' section is currently selected, indicated by a blue arrow icon. The main content area is titled 'Taxonomy Manager' and displays a table of custom taxonomies. The table has columns for 'Singular Name' and 'Hierarchical'. It contains two rows:

Singular Name	Hierarchical	Actions	
Genre	TRUE	Edit	Delete
chapter	Chapter	Edit	Delete

At the top of the main content area, there are buttons for 'Tous les articles', 'Ajouter', 'Catégories', 'Étiquettes', 'Genre', and 'Chapter'. There are also links for 'Add Taxonomy' and 'Export'. The status bar at the bottom of the browser window shows the URL: localhost/wordpress/wp-admin/admin.php?page=alecaddd_taxonomy.

The screenshot shows the 'Alecaddd Plugin' settings page in the WordPress admin interface. The left sidebar is dark-themed with white icons and text, showing the 'Alecaddd' plugin is currently selected. The main content area has a light gray background. At the top, there are three tabs: 'Manage Settings' (selected), 'Updates', and 'About'. Below this, a section titled 'Settings Manager' contains the following options, each with a blue toggle switch:

- Activate CPT Manager
- Activate Taxonomy Manager
- Activate Media Widget
- Activate Gallery Manager
- Activate Testimonial Manager
- Activate Templates Manager
- Activate Ajax Login/Signup
- Activate Membership Manager

The screenshot shows the 'Widgets' settings page in the WordPress admin interface. The left sidebar is dark-themed with white icons and text, showing the 'Apparence' (Appearance) section is selected. The main content area displays various widget options in a grid format. One specific widget, 'Alecadd Media Widget: Custom Text', is highlighted and expanded. This widget has the following configuration fields:

- Title: Custom Text
- Image: A placeholder box with a 'Select Image' button.
- Buttons: 'Supprimer | Terminé' (Delete | Done) and 'Enregistré' (Save).

```

document.querySelector(activePaneID).classList.add("active");

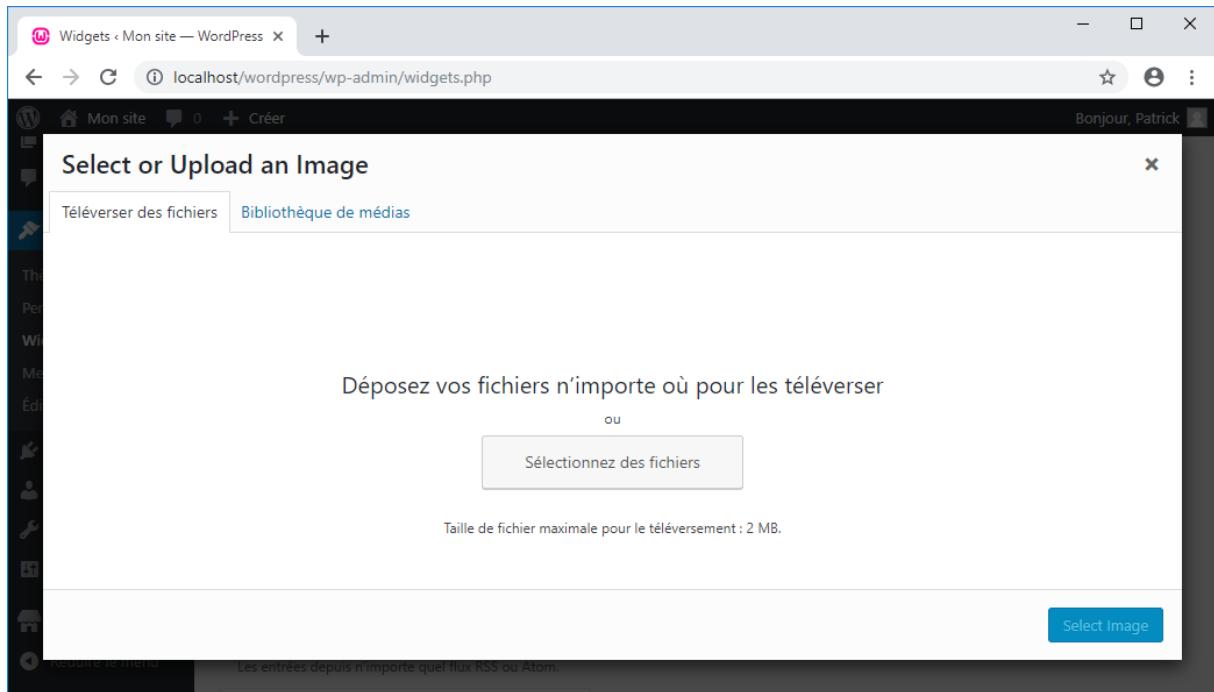
jQuery(document).ready(function ($) {
    $(document).on('click', '.js-image-upload', function (e) {
        e.preventDefault();
        var $button = $(this);

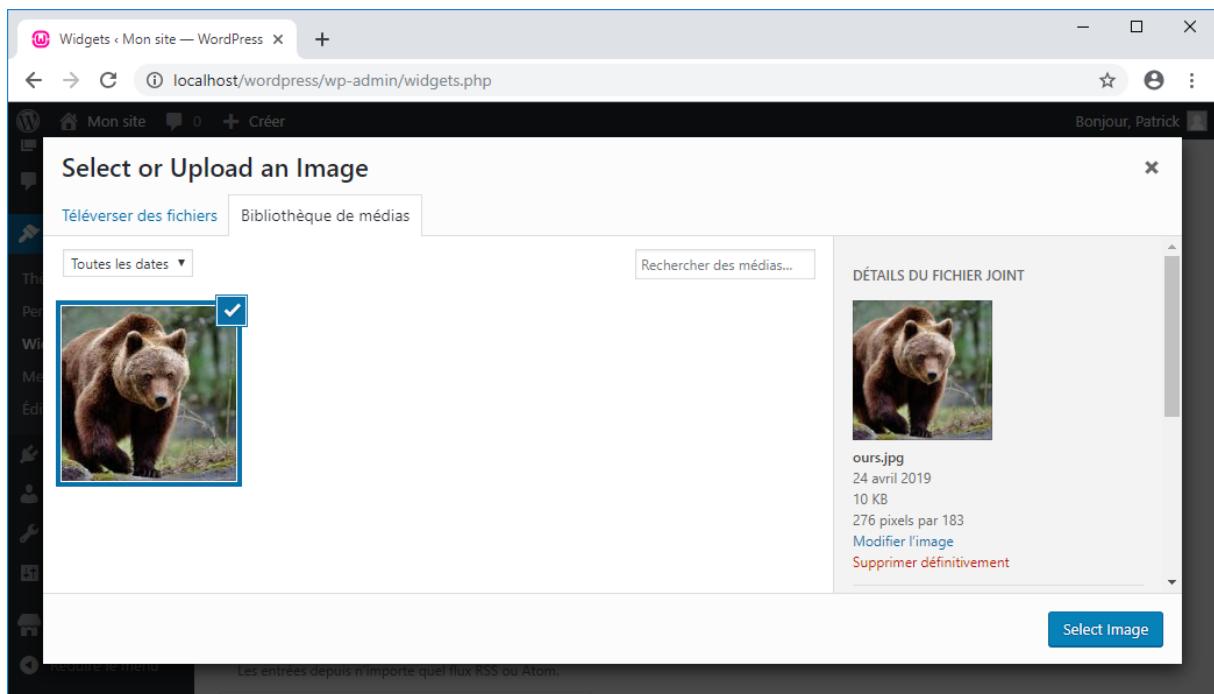
        var file_frame = wp.media.frames.file_frame = wp.media({
            title: 'Select or Upload an Image',
            library: {
                type: 'image' // mime type
            },
            button: {
                text: 'Select Image'
            },
            multiple: false
        });

        file_frame.on('select', function () {
            var attachment = file_frame.state().get('selection').first().toJSON();
            $button.siblings('.image-upload').val(attachment.url);
        });

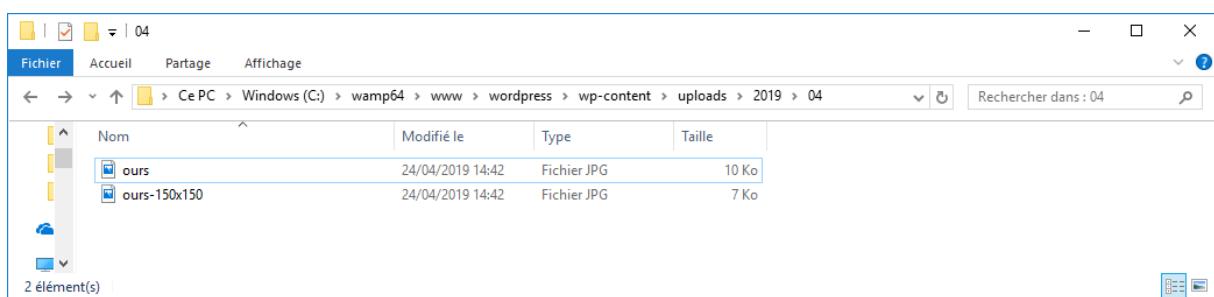
        file_frame.open();
    });
});

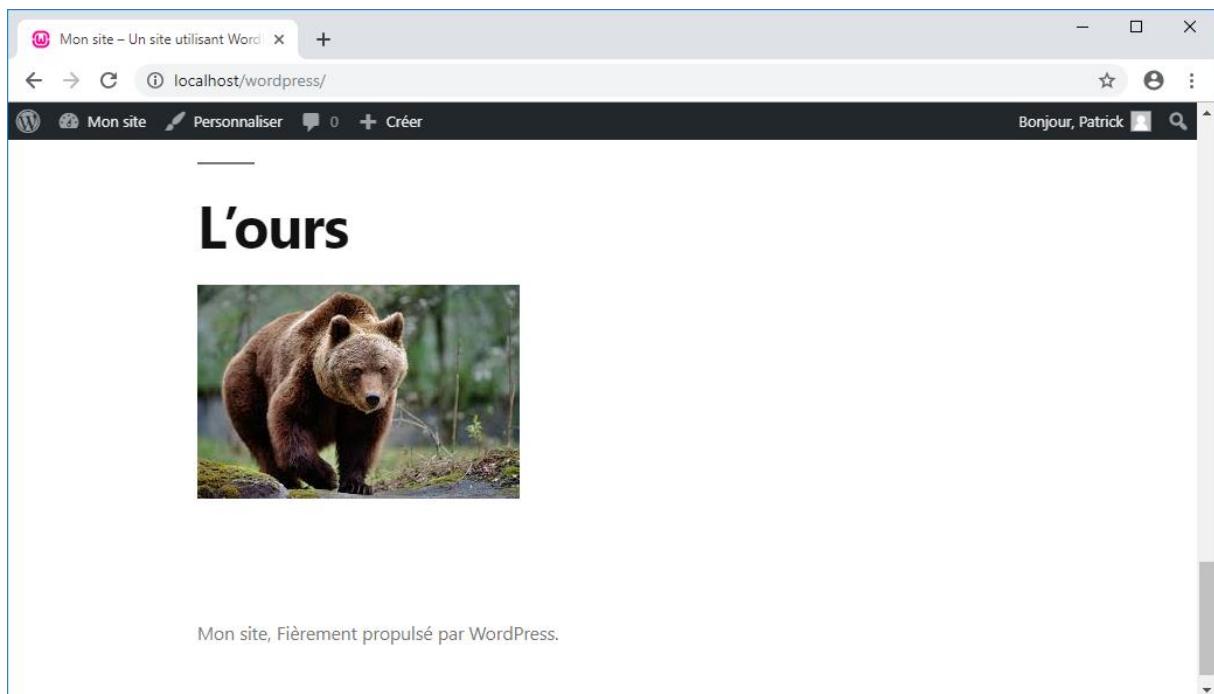
```



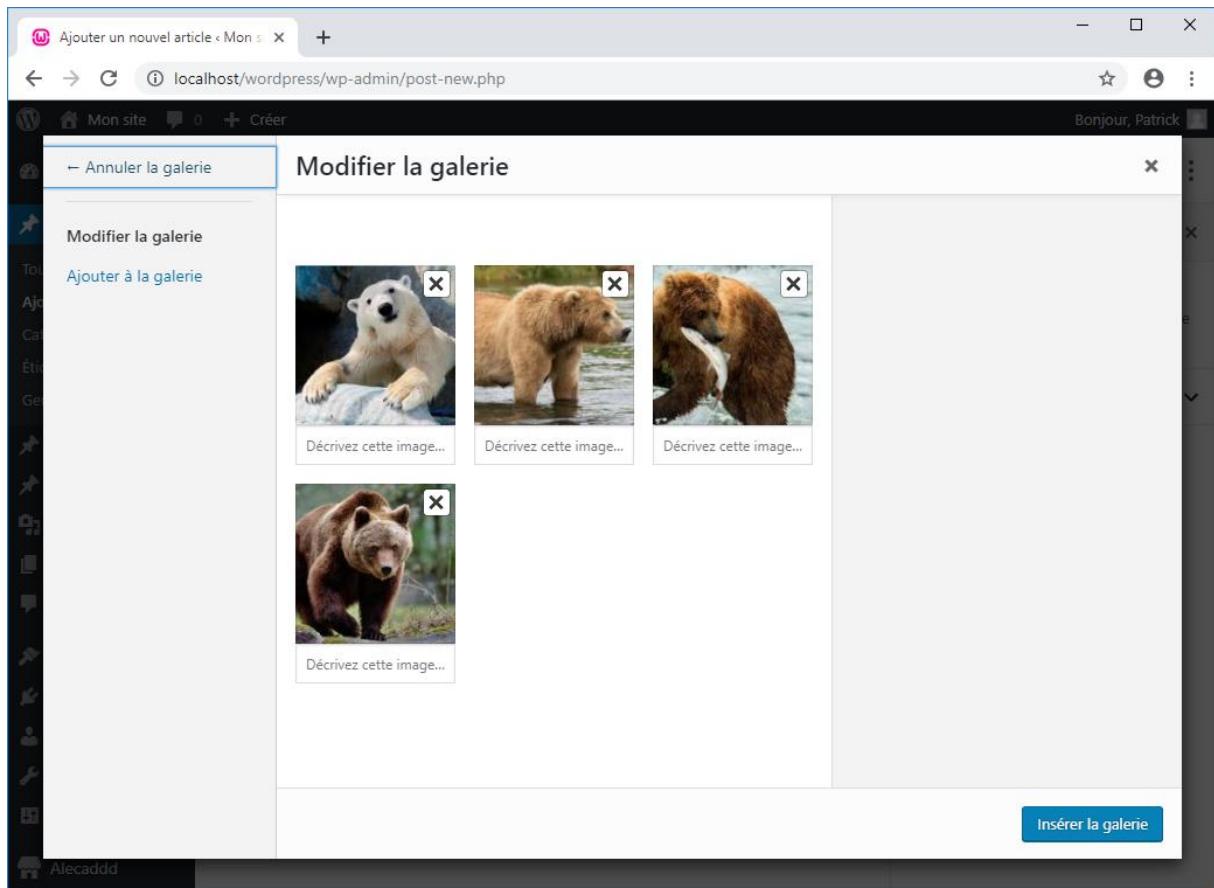


The screenshot shows the WordPress Widgets panel. On the left, the **Widgets** menu item is selected. In the main area, a **Alecaddd Media Widget: Custom Text** is configured with a title of **L'ours**. The **Image:** field contains the URL <http://localhost/wordpress/wp-content/uploads/2019/04/ours.jpg>. Below it are buttons for **Select Image**, **Supprimer** (Delete), and **Enregistrer** (Save).





A screenshot of the WordPress media library interface. The title 'Créer une galerie' is at the top. On the left, there is a sidebar with various icons. In the center, there are four images of bears: a polar bear, two brown bears, and one brown bear highlighted with a blue border. To the right, there is a 'DÉTAILS DU FICHIER JOINT' section for the selected image, showing 'ours.jpg', '24 avril 2019', '10 KB', '276 pixels par 183', 'Modifier l'image', and 'Supprimer définitivement'. Below this, there are fields for 'Adresse web' (set to 'http://localhost/wordpress/wi'), 'Titre' (set to 'ours'), 'Légende', 'Texte alternatif', and 'Description'. At the bottom, there are buttons for 'Créer une nouvelle galerie' and 'Effacer'.



The screenshot shows the WordPress post editor with a featured image of a bear and the title 'Les ours'. A gallery block is inserted, displaying four images of bears. The left sidebar shows the 'Articles' section selected. The right sidebar includes buttons for 'Enregistrer le brouillon' (Save draft), 'Prévisualiser' (Preview), and 'Publier...' (Publish). The 'Bloc' tab is selected in the top right of the sidebar. The 'Galerie' settings panel is open, showing options for 'Colonnes' (Columns) set to 2, and a toggle for 'Recadrer les images' (Crop images) which is turned off. The 'Lier à' (Link to) dropdown is set to 'Aucun' (None).

Mon site – Un site utilisant WordPress

localhost/wordpress/

Mon site Personnaliser 0 + Créer Bonjour, Patrick

Les ours



Patrick 24 avril 2019 Non classé

Laisser un commentaire Modifier

The screenshot shows the WordPress admin dashboard with the Alecaddd plugin active. The left sidebar has a dark theme with white icons. The main menu item 'Alecadd' is highlighted in blue. The right panel displays a list of management tools, each with a toggle switch:

- Activate CPT Manager (on)
- Activate Taxonomy Manager (on)
- Activate Media Widget (on)
- Activate Gallery Manager (off)
- Activate Testimonial Manager (on)
- Activate Templates Manager (off)
- Activate Ajax Login/Signup (off)
- Activate Membership Manager (off)
- Activate Chat Manager (off)

A blue button at the bottom says "Enregistrer les modifications".

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

The screenshot shows the 'Testimonial Manager' page. The left sidebar has a dark theme with white icons. The main title 'Testimonial Manager' is displayed prominently. The sidebar shows the following menu items:

- Dashboard
- CPT Manager
- Taxonomy Manager
- Testimonial Manager** (highlighted in blue)
- Widgets Manager

The screenshot shows the 'Alecadd' settings page. The left sidebar has a dark theme with white icons. The main menu item 'Alecadd' is highlighted in blue. The right panel shows the same list of management tools as the first screenshot, with the 'Testimonial Manager' option also highlighted in blue. A message at the bottom says "Merci de faire de [WordPress](#) votre outil de création." and "Version 5.1.1".

Commentaires < Mon site — WordPress

localhost/wordpress/wp-admin/edit-comments.php

Bonjour, Patrick

Tableau de bord

Articles Products Comic books Médias Pages Commentaires Apparence Extensions Utilisateurs Outils Réglages Alecaddd Réduire le menu

Commentaires

Tous (1) | Le mien (0) | En attente (0) | Approuvé (1) | Indésirable (0) | Mis à la corbeille (0)

Actions groupées Appliquer Tous les types de commentaires Filtrer 1 élément

Auteur	Commentaire	En réponse à	Envoyé le
 Un commentaireur WordPress wordpress.org wapuu@wordpress.example	Bonjour, ceci est un commentaire. Pour débuter avec la modération, la modification et la suppression de commentaires, veuillez visiter l'écran des Commentaires dans le Tableau de bord. Les avatars des personnes qui commentent arrivent depuis Gravatar.	Bonjour tout le monde ! Voir l'article	04/04/2019 à 11 h 28 min

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

Dashicons | WordPress Developer

https://developer.wordpress.org/resource/dashicons/#testimonial

Register Log In

WORDPRESS.ORG

Showcase Themes Plugins Mobile Support Get Involved About Blog Hosting Get WordPress

Developer Resources: Dashicons



dashicons-testimonial

[Copy CSS](#)

[Copy HTML](#)

[Copy Glyph](#)

Dashicons is the official icon font of the WordPress admin as of 3.8.

For new icon requests, please [create a new issue on the Dashicons github](#).

For any issues that appear within WordPress core, please [create a new ticket on trac](#). Use the "administration" component and the "UI" focus when creating the new ticket, and be sure to include "Dashicons" somewhere in the text of the ticket.

Dashicons is licensed under [GPLv2](#), or any later version with font exception.

```

28
29     public function testimonial_cpt ()
30     {
31         $labels = array(
32             'name' => 'Testimonials',
33             'singular_name' => 'Testimonial'
34         );
35
36         $args = array(
37             'labels' => $labels,
38             'public' => true,
39             'has_archive' => false,
40             'menu_icon' => 'dashicons-testimonial',
41             'exclude_from_search' => true,
42             'publicly_queryable' => false,
43             'supports' => array( 'title', 'editor' )
44         );
45
46         register_post_type ( 'testimonial', $args );
47     }

```

Li 40, Col 50 (21 sélectionné) Taille des tabulations : 4 UTF-8 LF PHP

Testimonials < Mon site — WordPress

localhost/wordpress/wp-admin/edit.php?post_type=testimonial

Mon site | Crée

Bonjour, Patrick

Tableau de bord

- Articles
- Products
- Comic books
- Médias
- Pages
- Commentaires
- Testimonials**
- Ajouter
- Apparence
- Extensions
- Utilisateurs
- Outils
- Réglages
- Alecaddd
- Réduire le menu

Testimonials Ajouter

Tous (0)

Aucune publication trouvée.

Merci de faire de [WordPress](#) votre outil de création. Version 5.1.1

The screenshot shows the Visual Studio Code interface with the file 'TestimonialController.php' open. The code is a PHP class definition for a testimonial controller, specifically for rendering a features box. It includes logic for handling nonce fields, getting post meta data, and rendering HTML input fields for author name, email, and approval status. The code uses Bootstrap-like classes for styling.

```
public function render_features_box($post)
{
    wp_nonce_field( 'alecaddd_testimonial', '_alecaddd_testimonial_nonce' );

    $data = get_post_meta( $post->ID, '_alecaddd_testimonial_key', true );
    $name = isset($data['name']) ? $data['name'] : '';
    $email = isset($data['email']) ? $data['email'] : '';
    $approved = isset($data['approved']) ? $data['approved'] : false;
    $featured = isset($data['featured']) ? $data['featured'] : false;
}



<label class="meta-label" for="alecaddd_testimonial_author">Author Name</label>
    <input type="text" id="alecaddd_testimonial_author" name="alecaddd_testimonial_author" class="form-control" value="{$name}">



<label class="meta-label" for="alecaddd_testimonial_email">Author Email</label>
    <input type="email" id="alecaddd_testimonial_email" name="alecaddd_testimonial_email" class="form-control" value="{$email}">



<label class="meta-label w-50 text-left" for="alecaddd_testimonial_approved">Approved</label>
    <div class="text-right w-50 inline">
        <div class="ui-toggle inline"><input type="checkbox" id="alecaddd_testimonial_approved" checked="">
            <label for="alecaddd_testimonial_approved"><div></div></label>
        </div>
    </div>


```

The screenshot shows the WordPress admin dashboard with the title 'Ajouter un nouvel article'. The left sidebar is the 'Tableau de bord' with various menu items like 'Articles', 'Products', 'Comic books', 'Médias', 'Pages', 'Commentaires', 'Testimonials', and 'Ajouter'. The 'Testimonials' item is currently selected. The main area has a text input field for the title, a rich text editor toolbar, and a text area for the content. On the right, there are two panels: 'Publier' (Publish) which includes options for saving as a draft, publishing, and publishing all at once, and 'Testimonial Options' which allow setting the author name, email, and approval status. At the bottom, there's a note about WordPress and the version information.

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

Modifier l'article < Mon site — WordPress

localhost/wordpress/wp-admin/post.php?post=26&action=edit

Bonjour, Patrick

Tableau de bord

- Articles
- Products
- Comic books
- Médias
- Pages
- Commentaires
- Testimonials**

Ajouter

Modifier l'article Ajouter

Le grognement de l'ours

Ajouter un média Visuel Texte

Paragraphe B I Émphasis Émphasis Émphasis Émphasis Émphasis Émphasis Émphasis

L'ours grogne.

Ours qui grogne À regarder plus tard Partager

Nombre de mots : 2 Dernière modification par Patrick, le 24 avril 2019 à 15 h 56 min

Publier

- État : Publié [Modifier](#)
- Visibilité : Public [Modifier](#)
- Publié le : 24 avril 2019 à 15 h 56 min [Modifier](#)

Déplacer dans la corbeille Mettre à jour

Testimonial Options

Author Name: Patrick

Author Email: patrick.croquet@afpa.fr

Approved:

Featured:

Merci de faire de [WordPress](#) votre outil de création. Version 5.1.1

Fichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide TestimonialController.php - wordpress - Visual Studio Code

```
TestimonialController.php x
10
11 /**
12 *
13 */
14 class TestimonialController extends BaseController
15 {
16     public $callbacks;
17
18     public $subpages = array();
19
20     public function register()
21     {
22         if ( ! $this->activated( 'testimonial_manager' ) ) return;
23
24         add_action( 'init', array( $this, 'testimonial_cpt' ) );
25         add_action( 'add_meta_boxes', array( $this, 'add_meta_boxes' ) );
26         add_action( 'save_post', array( $this, 'save_meta_box' ) );
27         add_action( 'manage_testimonial_posts_columns', array( $this, 'set_custom_columns' ) );
28         add_action( 'manage_testimonial_posts_custom_column', array( $this, 'set_custom_columns_data' ) );
29         add_filter( 'manage_edit-testimonial_sortable_columns', array( $this, 'set_custom_columns_sortab
30     }
31 }
```

Li 32, Col 39 Taille des tabulations : 4 UTF-8 LF PHP 🎉 🔔

The screenshot shows the Visual Studio Code interface with the file 'TestimonialController.php' open. The code defines a public function 'set_custom_columns' that takes an array '\$columns' as input. It then performs several operations: it removes the 'title' and 'date' keys from the array, creates new keys 'name', 'title', 'approved', 'featured', and 'date' with specific values ('Author Name', the original title, 'Approved', 'Featured', and the original date respectively), and finally returns the modified array. The code editor includes line numbers from 128 to 143.

```
public function set_custom_columns($columns)
{
    $title = $columns['title'];
    $date = $columns['date'];
    unset( $columns['title'], $columns['date'] );

    $columns['name'] = 'Author Name';
    $columns['title'] = $title;
    $columns['approved'] = 'Approved';
    $columns['featured'] = 'Featured';
    $columns['date'] = $date;

    return $columns;
}
```

The screenshot shows the WordPress admin dashboard with the 'Testimonials' menu item selected. The main area displays a list of testimonial posts. There are two entries:

<input type="checkbox"/>	Titre	Date
<input type="checkbox"/>	Le grognement de l'ours	Publié il.y.a 47 minutes
<input type="checkbox"/>	Titre	Date

Below the list are filtering and grouping options. The left sidebar shows other menu items like 'Articles', 'Products', and 'Comments'. The bottom right corner indicates the version is 5.1.1.

Fichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide TestimonialController.php - wordpress - Visual Studio Code

```
19
20     public function register()
21     {
22         if ( ! $this->activated( 'testimonial_manager' ) ) return;
23
24         add_action( 'init', array( $this, 'testimonial_cpt' ) );
25         add_action( 'add_meta_boxes', array( $this, 'add_meta_boxes' ) );
26         add_action( 'save_post', array( $this, 'save_meta_box' ) );
27         add_action( 'manage_testimonial_posts_columns', array( $this, 'set_custom_columns' ) );
28         add_action( 'manage_testimonial_posts_custom_column', array( $this, 'set_custom_columns_data' ), 10 );
29         add_filter( 'manage_edit-testimonial_sortable_columns', array( $this, 'set_custom_columns_sortab
```

Li 32, Col 1 Taille des tabulations : 4 UTF-8 LF PHP ⚡ 📰

Fichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide TestimonialController.php - wordpress - Visual Studio Code

```
128
129     public function set_custom_columns($columns)
130     {
131         $title = $columns['title'];
132         $date = $columns['date'];
133         unset( $columns['title'], $columns['date'] );
134
135         $columns['name'] = 'Author Name';
136         $columns['title'] = $title;
137         $columns['approved'] = 'Approved';
138         $columns['featured'] = 'Featured';
139         $columns['date'] = $date;
140
141         return $columns;
142     }
143
```

Li 154, Col 75 Taille des tabulations : 4 UTF-8 LF PHP ⚡ 📰

Fichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide TestimonialController.php - wordpress - Visual Studio Code

```
143
144     public function set_custom_columns_data($column, $post_id)
145     {
146         $data = get_post_meta( $post_id, '_alecaddd_testimonial_key', true );
147         $name = isset($data['name']) ? $data['name'] : '';
148         $email = isset($data['email']) ? $data['email'] : '';
149         $approved = isset($data['approved']) && $data['approved'] === 1 ? '<strong>YES</strong>' : 'NO';
150         $featured = isset($data['featured']) && $data['featured'] === 1 ? '<strong>YES</strong>' : 'NO';
151
152         switch($column) {
153             case 'name':
154                 echo '<strong>' . $name . '</strong><br/><a href="mailto:' . $email . '">' . $email . '</a>';
155                 break;
156
157             case 'approved':
158                 echo $approved;
159                 break;
160
161             case 'featured':
162                 echo $featured;
163                 break;
164         }
165     }
166
```

Li 168, Col 1 Taille des tabulations : 4 UTF-8 LF PHP ⚡ 📰

The screenshot shows the Visual Studio Code interface with the file 'TestimonialController.php' open. The code defines a public function 'set_custom_columns_sortable' that maps column names to database fields: 'name', 'approved', and 'featured'. The code editor includes standard VS Code icons for file operations, search, and settings.

```
167     public function set_custom_columns_sortable($columns)
168     {
169         $columns['name'] = 'name';
170         $columns['approved'] = 'approved';
171         $columns['featured'] = 'featured';
172
173     }
174
175 }
```

Li 163, Col 1 Taille des tabulations : 4 UTF-8 LF PHP

The screenshot shows the WordPress admin dashboard under the 'Testimonials' menu. It displays a list of one testimonial entry. The columns are 'Author Name', 'Titre', 'Approved', 'Featured', and 'Date'. The testimonial details are: Author Name - Patrick, Titre - Le grognement de l'ours, Approved - YES, Featured - NO, Date - Publié il y a 17 heures. The sidebar on the left shows other admin menus like 'Articles', 'Products', and 'Comments'.

Author Name	Titre	Approved	Featured	Date
Patrick patrick.croquet@afpa.fr	Le grognement de l'ours	YES	NO	Publié il y a 17 heures

Merci de faire de [WordPress](#) votre outil de création. Version 5.1.1

The screenshot shows the Visual Studio Code interface with the file 'TestimonialController.php' open. This version of the code includes a 'register' function that sets up various hooks and filters related to the testimonial post type. The code editor includes standard VS Code icons for file operations, search, and settings.

```
19
20     public function register()
21     {
22         if ( ! $this->activated( 'testimonial_manager' ) ) return;
23
24         $this->settings = new SettingsApi();
25
26         $this->callbacks = new TestimonialCallbacks();
27
28         add_action( 'init', array( $this, 'testimonial_cpt' ) );
29         add_action( 'add_meta_boxes', array( $this, 'add_meta_boxes' ) );
30         add_action( 'save_post', array( $this, 'save_meta_box' ) );
31         add_action( 'manage_testimonial_posts_columns', array( $this, 'set_custom_columns' ) );
32         add_action( 'manage_testimonial_posts_custom_column', array( $this, 'set_custom_columns_data' ) );
33         add_filter( 'manage_edit-testimonial_sortable_columns', array( $this, 'set_custom_columns_sortab
34
35         $this->setShortcodePage();
36
37 }
```

Li 42, Col 1 Taille des tabulations : 4 UTF-8 LF PHP

```
37
38     public function setshortcodePage()
39     {
40         $subpage = array(
41             array(
42                 'parent_slug' => 'edit.php?post_type=testimonial',
43                 'page_title' => 'Shortcodes',
44                 'menu_title' => 'Shortcodes',
45                 'capability' => 'manage_options',
46                 'menu_slug' => 'alecaddd_testimonial_shortcode',
47                 'callback' => array( $this->callbacks, 'shortcodePage' )
48             )
49         );
50
51         $this->settings->addSubPages( $subpage )->register();
52     }
53 }
```

Li 56, Col 1 Taille des tabulations : 4 UTF-8 LF PHP ⚡

```
1 <?php
2 /**
3  * @package AlecadddPlugin
4  */
5 namespace Inc\Base;
6
7 use Inc\Api\SettingsApi;
8 use Inc\Base\BaseController;
9 use Inc\Api\Callbacks\TestimonialCallbacks;
```

Li 49, Col 11 Taille des tabulations : 4 UTF-8 LF PHP ⚡

EXPLORATEUR

- EDITEURS OUVERTS
- WORDPRESS
 - assetsold
 - assetsold2
 - assetsold3
 - inc
 - Api
 - Callbacks
 - TestimonialCallbacks.php
 - AdminCallbacks.php
 - CptCallbacks.php
 - ManagerCallbacks.php
 - TaxonomyCallbacks.php

TestimonialController.php *

```
37
38     public function setshortcodePage()
39     {
40         $subpage = array(
41             array(
42                 'parent_slug' => 'edit.php?post_type=testimonial',
43                 'page_title' => 'Shortcodes',
44                 'menu_title' => 'Shortcodes',
45                 'capability' => 'manage_options',
46                 'menu_slug' => 'alecaddd_testimonial_shortcode',
47                 'callback' => array( $this->callbacks, 'shortcodePage' )
48             )
49         );
50
51         $this->settings->addSubPages( $subpage )->register();
52     }
53 }
```

Li 56, Col 1 Taille des tabulations : 4 UTF-8 LF PHP ⚡

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure under "WORDPRESS". The "inc\Api\Callbacks" folder is expanded, and "TestimonialCallbacks.php" is selected.
- Code Editor (Right):** Displays the content of "TestimonialCallbacks.php". The code defines a class "TestimonialCallbacks" extending "BaseController". It contains a public function "shortcodePage" which returns a require_once statement for a template file.
- Status Bar (Bottom):** Shows "Li 16, Col 1 Espaces : 4 UTF-8 LF PHP".

```
<?php
/*
 * @package AlecadddPlugin
 */
namespace Inc\Api\Callbacks;

use Inc\Base\BaseController;

class TestimonialCallbacks extends BaseController
{
    public function shortcodePage()
    {
        return require_once( "$this->plugin_path/templates/testimonial.php" );
    }
}
```

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure under "templates". "testimonial.php" is selected.
- Code Editor (Right):** Displays the content of "testimonial.php". The code includes a namespace declaration and a use statement for "Inc\Base\BaseController".
- Status Bar (Bottom):** Shows "Li 4, Col 4 Espaces : 4 UTF-8 LF PHP".

```
/*
 */
namespace Inc\Api\Callbacks;

use Inc\Base\BaseController;
```

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure under "templates". "testimonial.php" is selected.
- Code Editor (Right):** Displays the content of "testimonial.php". The code defines a class "wrap" with several methods: "Testimonial Shortcode", "Testimonial Form Shortcode", and "Testimonial SlideShow Shortcode". Each method contains a shortcode definition.
- Status Bar (Bottom):** Shows "Li 8, Col 1 Espaces : 4 UTF-8 LF PHP".

```
<div class="wrap">
    <h1>Testimonial Shortcode</h1>
    <p>Testimonial Form Shortcode</p>
    <p><code>[testimonial-form]</code></p>
    <p>Testimonial SlideShow Shortcode</p>
    <p><code>[testimonial-slideshow]</code></p>
</div>
```

The screenshot shows the WordPress admin dashboard. The left sidebar is titled "Tableau de bord" and includes links for Articles, Products, Comic books, Médias, Pages, Commentaires, and Testimonials (which is currently selected). Below these are links for Apparence, Extensions, Utilisateurs, Outils, Réglages, and a user profile for "Alecaddd". At the bottom of the sidebar are "Ajouter" and "Shortcodes" buttons. The main content area is titled "Testimonial Shortcode" and contains two sections: "Testimonial Form Shortcode" with the shortcode [testimonial-form] and "Testimonial SlideShow Shortcode" with the shortcode [testimonial-slideshow]. A footer message says "Merci de faire de [WordPress](#) votre outil de création." and indicates "Version 5.1.1".

The screenshot shows the Visual Studio Code editor with the file "TestimonialController.php" open. The code defines a class with a public function "register" that adds various hooks and filters related to testimonial posts. The code is as follows:

```
19 public function register()
20 {
21     if ( ! $this->activated( 'testimonial_manager' ) ) return;
22
23     $this->settings = new SettingsApi();
24
25     $this->callbacks = new TestimonialCallbacks();
26
27     add_action( 'init', array( $this, 'testimonial_cpt' ) );
28     add_action( 'add_meta_boxes', array( $this, 'add_meta_boxes' ) );
29     add_action( 'save_post', array( $this, 'save_meta_box' ) );
30     add_action( 'manage_testimonial_posts_columns', array( $this, 'set_custom_columns' ) );
31     add_action( 'manage_testimonial_posts_custom_column', array( $this, 'set_custom_columns_data' ), 10 );
32     add_filter( 'manage_edit-testimonial_sortable_columns', array( $this, 'set_custom_columns_sortable' ) );
33
34     $this->setshortcodePage();
35
36     add_shortcode( 'testimonial-form', array( $this, 'testimonial_form' ) );
37
38 }
39
```

The status bar at the bottom shows "Li 42, Col 1 Taille des tabulations : 4 UTF-8 LF PHP 🎉 🚨".

The screenshot shows the Visual Studio Code interface with the following details:

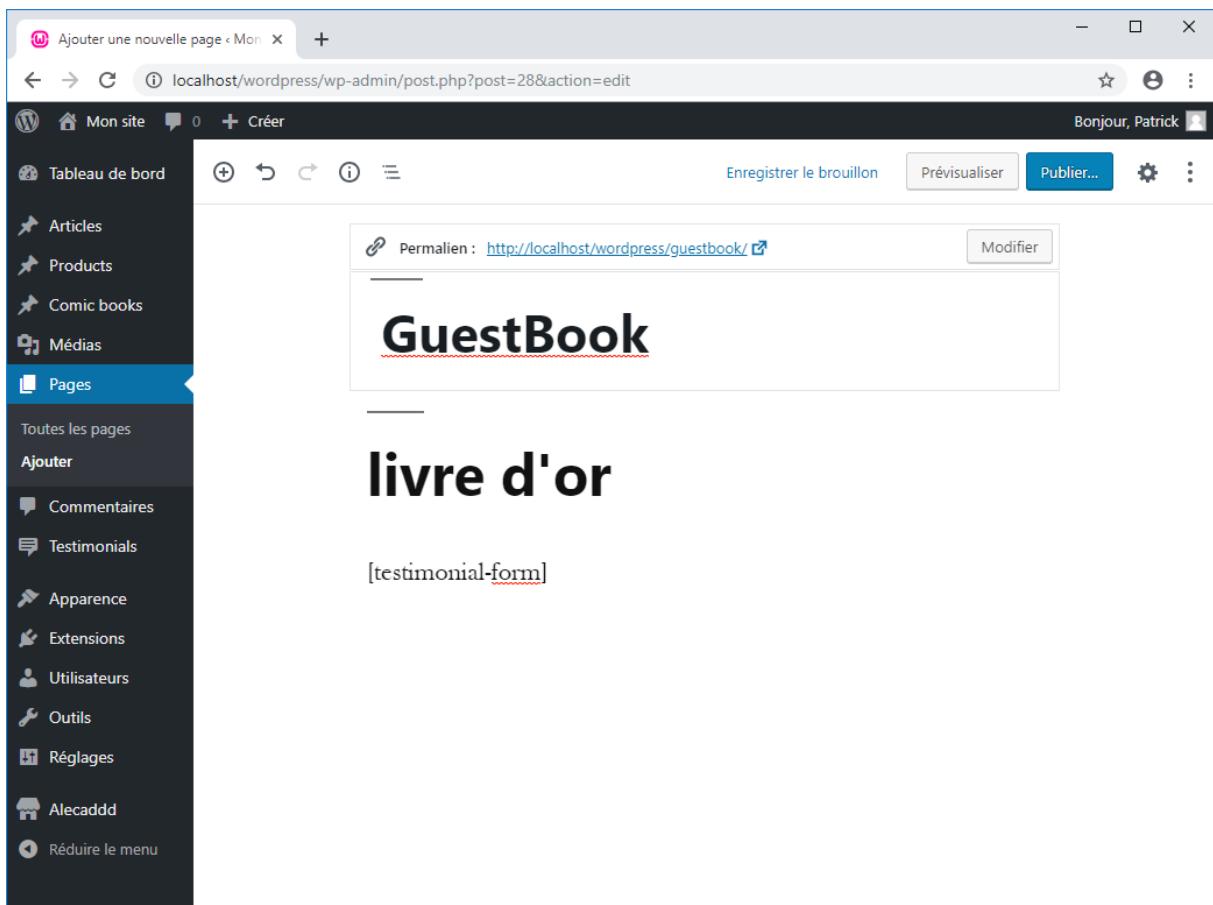
- File Explorer:** Shows files like `TestimonialController.php`, `TestimonialCallbacks.php`, and `testimonial.php`.
- Editor:** Displays the code for the `testimonial_form()` method in `TestimonialController.php`. The code includes `ob_start()`, `require_once` for `contact-form.php`, and `return ob_get_clean();`.
- Status Bar:** Shows "Li 50, Col 1 Taille des tabulations : 4 UTF-8 LF PHP" and a bell icon.

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the `templates` folder containing `contact-form.php`, `admin.php`, and `cpt.php`. `contact-form.php` is currently selected.
- Editor:** Displays the code for the `testimonial_form()` method in `TestimonialController.php`, which includes the path to `contact-form.php`.
- Status Bar:** Shows "Li 43, Col 53 (16 sélectionné) Taille des tabulations : 4 UTF-8 LF PHP" and a bell icon.

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows files like `TestimonialController.php`, `TestimonialCallbacks.php`, `testimonial.php`, and `contact-form.php`.
- Editor:** Displays the HTML form code for `contact-form.php`, including fields for Name, Email, and Message, and a submission button.
- Status Bar:** Shows "Li 26, Col 1 Taille des tabulations : 4 UTF-8 LF PHP" and a bell icon.



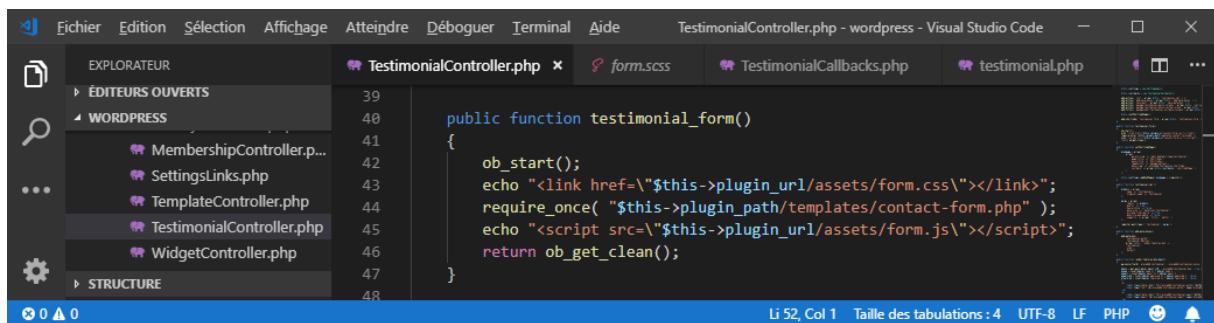
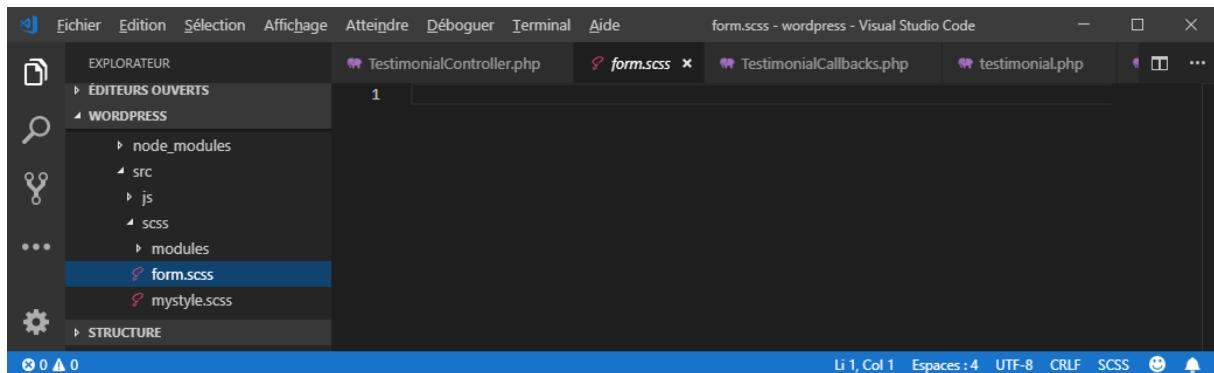
```
public function testimonial_form()
{
    ob_start();
    require_once( "$this->plugin_path/templates/contact-form.php" );
    echo "<script src=\"$this->plugin_url/src/js/form.js\"></script>";
    return ob_get_clean();
}
```

This screenshot shows a Visual Studio Code window with the file 'TestimonialController.php' open. The code defines a public function 'testimonial_form()' that outputs a script tag pointing to 'form.js'. The code editor shows lines 39 through 46. The status bar at the bottom indicates the file is in PHP mode, with encoding set to UTF-8 and line 44, column 47.

```
document.addEventListener('DOMContentLoaded', function(e) {
    console.log('This is Ready!');
});
```

This screenshot shows a Visual Studio Code window with the file 'form.js' open. The code defines a document event listener for 'DOMContentLoaded' that logs 'This is Ready!' to the console. The code editor shows lines 1 through 3. The status bar at the bottom indicates the file is in JavaScript mode, with encoding set to UTF-8 and line 1, column 1.

The screenshot shows a WordPress guestbook page titled "livre d'or". The page includes fields for "Your Name", "Your Email", and "Your Message", each with a required message below it. A "Submit" button is present. To the right, the browser's developer tools are open, specifically the "Elements" tab, which displays the HTML structure of the page. The "site-content-contain" class is highlighted in blue. The "Console" tab shows a log entry for "jquery-migrate.min.js" version 1.4.1.



A screenshot of Visual Studio Code showing a Gulpfile.js file. The code defines variables for project URLs, styles, and scripts, and sets up a browserSync task with specific proxy and https configurations.

```
28 var reload      = browserSync.reload;
29
30 // Project related variables
31 var projectURL  = 'https://test.dev';
32
33 var styleSRC     = 'src/scss/mystyle.scss';
34 var styleFront   = 'src/scss/form.scss';
35 var styleURL     = './assets/';
36 var mapURL       = './';
37
38 var jsSRC        = 'src/js/';
39 var jsAdmin      = 'myscript.js';
40 var jsFront      = 'form.js';
41 var jsFiles      = [jsAdmin, jsFront];
42 var jsURL        = './assets/';
43
44 var styleWatch   = 'src/scss/**/*.scss';
45 var jsWatch      = 'src/js/**/*.js';
46 var phpWatch     = '**/*.php';
47
48 // Tasks
49 gulp.task('browser-sync', function() {
50   browserSync.init({
51     proxy: projectURL,
52     https: {
53       key: '/Users/alecaddd/.valet/Certificates/test.dev.key',
54       cert: '/Users/alecaddd/.valet/Certificates/test.dev.crt'
55     }
56   });
57 })
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
```

A screenshot of Visual Studio Code showing a Gulpfile.js file. The code defines a task named 'js' that maps entries, transforms them with babelify, bundles them, pipes them through source, buffer, and sourcemaps, strips debug information, initializes sourcemaps, pipes through uglify, sourcemaps.write, and finally dest, and then streams the result to browserSync.

```
75 gulp.task('js', function() {
76   jsFiles.map(function(entry) {
77     return browserify({
78       entries: [jsSRC + entry]
79     })
80     .transform( babelify, { presets: [ 'env' ] } )
81     .bundle()
82     .pipe( source( entry ) )
83     .pipe( buffer() )
84     .pipe( gulpif( options.has( 'production' ), stripDebug() ) )
85     .pipe( sourcemaps.init({ loadMaps: true }) )
86     .pipe( uglify() )
87     .pipe( sourcemaps.write( '.' ) )
88     .pipe( gulp.dest( jsURL ) )
89     .pipe( browserSync.stream() );
90   });
91 });
92
93
```

<https://gulpjs.com/docs/en/getting-started/quick-start>

<https://www.npmjs.com/package/npx>

The screenshot shows a Visual Studio Code interface with several tabs open at the top: TestimonialController.php, TestimonialCallbacks.php, testimonial.php, contact-form.php, and gulpfile.js. The terminal tab is active, displaying the following PowerShell session:

```

PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin> npm rm --global gulp
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin> node --version
v11.12.0
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin> npm --version
6.9.0
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin> npx --version
6.9.0
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin> npm install -g npx
C:Users\1702726\AppData\Roaming\npm\npx -> C:Users\1702726\AppData\Roaming\npm\node_modules\npx\index.js
+ npx@10.2.0
added 484 packages from 651 contributors in 38.131s
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin> npx --version
6.9.0
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin> npm install --global gulp-cli
C:Users\1702726\AppData\Roaming\npm\gulp -> C:Users\1702726\AppData\Roaming\npm\node_modules\gulp\cli\bin\gulp.js
+ gulp-cli@2.2.0
added 236 packages from 157 contributors in 16.573s
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin>

```

The status bar at the bottom indicates: Li 1, Col 1 Taille des tabulations : 4 UTF-8 LF JavaScript

The screenshot shows a Visual Studio Code interface with several tabs open at the top: TestimonialController.php, JS form.js, TestimonialCallbacks.php, testimonial.php, and contact-form.php. The JS form.js tab is active, displaying the following HTML code:

```

1 <form id="alecadddd-testimonial-form" action="#" method="post" data-url=<?php echo admin_url('admin-ajax.php'); ?>>
2
3     <div class="field-container">
4         <input type="text" class="field-input" placeholder="Your Name" id="name" name="name" required>
5         <small class="field-msg error">Your Name is Required</small>
6     </div>
7
8     <div class="field-container">
9         <input type="email" class="field-input" placeholder="Your Email" id="email" name="email" required>
10        <small class="field-msg error">Your Email is Required</small>
11        <small class="field-msg error" data-error="invalidEmail">The Email address is not valid</small>
12    </div>
13
14    <div class="field-container">
15        <textarea name="message" id="message" class="field-input" placeholder="Your Message" required></textarea>
16        <small class="field-msg error">A Message is Required</small>
17    </div>
18
19    <div class="field-container">
20        <div>
21            |   <button type="submit" class="btn btn-default btn-lg btn-sunset-form">Submit</button>
22        </div>
23        <small class="field-msg js-form-submission">Submission in process, please wait&hellip;</small>
24        <small class="field-msg success js-form-success">Message Successfully submitted, thank you!</small>
25        <small class="field-msg error js-form-error">There was a problem with the Contact Form, please try again!</small>
26    </div>
27
28 </form>

```

The status bar at the bottom indicates: Li 1, Col 11 (25 sélectionné) Taille des tabulations : 4 UTF-8 LF PHP

Ajouter un "Virtual Host" +

localhost/add_vhost.php?lang=french

Ajouter un VirtualHost - Retour à l'accueil

Version 3.1.7 - 64bit french

Les fichiers ont été modifiés, le virtual host monsite.test a été créé

Messages de la console pour actualisation des DNS :

Vous pouvez ajouter un autre VirtualHost en validant "Ajouter un VirtualHost!"
 Cependant, pour que ces nouveaux VirtualHost soient pris en compte par Apache, vous devez lancer l'item
 Redémarrage DNS
 du menu Outils par Clic-Droit sur l'icône Wampmanager. (Ceci ne peut, hélas, pas être fait automatiquement)

Eichier Edition Sélection Affichage Atteindre Déboguer Terminal Aide gulpfile.js - wordpress - Visual Studio Code

testimonialController.php TestimonialCallbacks.php testimonial.php contact-form.php JS gulpfile.js JS myscript.js

```

29
30 // Project related variables
31 var projectURL = 'http://monsite.test';
32
33 var styleSRC = 'src/scss/mystyle.scss';
34 var styleFront = 'src/scss/form.scss';
35 var styleURL = './assets/';
36 var mapURL = './';
37
38 var jsSRC = 'src/js/';
39 var jsAdmin = 'myscript.js';
40 var jsFront = 'form.js';
41 var jsFiles = [jsAdmin, jsFront];
42 var jsURL = './assets/';
43

```

PROBLÈMES SORTIE CONSOLE DE DÉBOGAGE TERMINAL

1: node

```

PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin> gulp watch
[14:19:05] Using gulpfile C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin\gulpfile.js
[14:19:05] Starting 'styles'...
[14:19:06] Finished 'styles' after 8.08 ms
[14:19:06] Starting 'js'...
[14:19:06] Finished 'js' after 8.36 ms
[14:19:06] Starting 'default'...
[14:19:06] Finished 'default' after 576 µs
[14:19:06] Starting 'browser-sync'...
[14:19:06] Finished 'browser-sync' after 8.5 ms
[14:19:06] Starting 'watch'...
[14:19:10] Finished 'watch' after 4.76 s
[Browsersync] Proxying: http://monsite.test
[Browsersync] Access URLs:
-----
Local: http://localhost:3000
External: http://172.17.82.73:3000
-----
UI: http://localhost:3001
UI External: http://localhost:3001
-----
[Browsersync] 4 files changed (mystyle.css.map, mystyle.css, form.css.map, form.css)
[Browsersync] 2 files changed (form.js.map, form.js)
[Browsersync] 2 files changed (myscript.js.map, myscript.js)
[Browsersync] Reloading Browsers... (buffered 6 events)

```

0 ▲ 0 Li 37, Col 1 Taille des tabulations : 4 UTF-8 LF JavaScript

```
#alecadddd-testimonial-form {
    .field-container {
        position: relative;
        margin-bottom: 20px;
    }

    .field-msg {
        display: none;
        position: absolute;
        left: 2px;
        bottom: -14px;
        font-size: 9px;
        text-transform: uppercase;
        font-weight: 600;
        letter-spacing: 0.05em;

        &.show {
            display: block;
        }
    }

    .error {
        color: #rgb(144, 19, 19);
    }

    .success {
        color: #rgb(47, 129, 47);
    }
}
```

PROBLÈMES SORTIE CONSOLE DE DÉBOGAGE TERMINAL

```
External: http://172.17.82.73:3000
-----
UI: http://localhost:3001
UI External: http://localhost:3001

[Browsersync] 4 files changed (mystyle.css.map, mystyle.css, form.css.map, form.css)
[Browsersync] 2 files changed (form.js.map, form.js)
[Browsersync] 2 files changed (myscript.js.map, myscript.js)
[Browsersync] Reloading Browsers... (buffered 6 events)
[14:33:31] Starting 'styles'...
[14:33:31] Finished 'styles' after 2.72 ms
[Browsersync] 4 files changed (mystyle.css.map, mystyle.css, form.css.map, form.css)
```

Li 1, Col 1 Espaces : 4 UTF-8 LF SCSS ⚡ 📰

```
public function testimonial_form()
{
    ob_start();
    echo "<link rel=\"stylesheet\" href=\"$this->plugin_url/assets/form.css\" type=\"text/css\" media=\"all\" />";
    require_once( "$this->plugin_path/templates/contact-form.php" );
    echo "<script src=\"$this->plugin_url/assets/form.js\"></script>";
    return ob_get_clean();
}
```

Li 51, Col 5 Taille des tabulations : 4 UTF-8 LF PHP ⚡ 📰

The screenshot shows the 'Réglages généraux' (General Settings) page in the WordPress admin interface. The left sidebar has 'Réglages' selected. The main area contains the following fields:

- Titre du site**: Mon site
- Slogan**: Un site utilisant WordPress
En quelques mots, décrivez la raison d'être de ce site.
- Adresse web de WordPress (URL)**: <http://monsite.test>
- Adresse web du site (URL)**: <http://monsite.test>
Saisir ici une adresse si vous souhaitez que la page d'accueil de votre site soit différente du répertoire de votre installation WordPress.
- Adresse de messagerie**: patrick.croquet@afpa.fr
Cette adresse est utilisée à des fins d'administration. Si vous la modifiez, nous enverrons un message à la nouvelle adresse afin de la confirmer. La nouvelle adresse ne sera pas active tant que vous ne l'aurez pas confirmée.

The screenshot shows the WordPress dashboard after the initial setup. The left sidebar has 'Tableau de bord' selected. The main area displays the welcome message and steps to start:

Bienvenue sur WordPress !
Voici quelques liens pour vous aider à démarrer :

Lancez-vous !

Personnalisez votre site

ou alors, changez complètement de thème

Étapes suivantes

- Écrivez votre premier article
- Ajoutez une page « À propos »
- Mettez en place votre page d'accueil
- Affichez votre site

Plus d'actions

- Gérez vos widgets ou vos menus
- Activez ou désactivez les commentaires
- Plus d'informations pour bien démarrer

GuestBook – Mon site

Non sécurisé | monsite.test/guestbook/

MON SITE

Un site utilisant WordPress

GUESTBOOK

Modifier

livre d'or

Patrick Patrick

patrick.croquet@afpa.fr

Coucou

Submit

Fièrement propulsé par WordPress

Elements

```
> <header class="entry-header">...</header>
<!-- .entry-header -->
<div class="entry-content"> == $0
  ::before
    <h2>livre d'or</h2>
    <link rel="stylesheet" href="http://monsite.test/wp-content/plugins/monpremier-plugin/assets/form.css" type="text/css" media="all">
  <form id="alecaddd-testimonial-form" action="#" method="post" data-url="http://monsite.test/wp-admin/admin-ajax.php">
    <div class="field-container">...</div>
    <div class="field-container">...</div>
    <div class="field-container">...</div>
    <div class="field-container">...</div>
  </form>
  <script src="http://monsite.test/wp-content/plugins/monpremier-plugin/assets/form.js"></script>
  ::after
</div>
<!-- .entry-content -->
</article>
<!-- #post-## -->
</main>
<!-- #main -->
```

Styles Event Listeners DOM Breakpoints Properties Accessibility

Console What's New

JQMIGRATE: Migrate is installed, version jquery-migrate.min.js?ver=1.4.1:2 1.4.1

Prevent form to submit form.js:8

http://monsite.test/wp-admin/admin-ajax.php form.js:26

GuestBook – Mon site

Non sécurisé | monsite.test/guestbook/

MON SITE

Un site utilisant WordPress

GUESTBOOK

Modifier

livre d'or

Patrick Patrick

patrick.croquet@afpa.fr

Coucou

Submit

THERE WAS A PROBLEM WITH THE CONTACT FORM, PLEASE TRY AGAIN!

Fièrement propulsé par WordPress

Network

Name	Stat...	Type	Initiator	Size	Time	Waterfall
admin-ajax.php	400	fetch	form.js:41	300 B	68 ...	

1 / 58 requests | 300 B / 63.1 KB transferred | 1 B / 1.2 MB resources | Finish: 7.7 min

Console What's New

JQMIGRATE: Migrate is installed, version jquery-migrate.min.js?ver=1.4.1:2 1.4.1

POST http://monsite.test/wp-admin/admin-ajax.php 400 (Bad Request) form.js:41

The screenshot shows a browser window with a guestbook form. The form fields contain "Patrick Patrick", "patrick.croquet@afpa.fr", and "Coucou". A "Submit" button is present. Below the form, an error message says "THERE WAS A PROBLEM WITH THE CONTACT FORM, PLEASE TRY AGAIN!". The Network tab of the developer tools is open, showing a request to "admin-ajax.php" with the following details:

Name	Headers	Preview	Response	Cookies	Timing
admin-ajax.php	556202856; wp-settings-1=mfold%3D0%26editor%3Dti nymce%26libraryContent%3Dbrowse				
	Host: monsite.test				
	Origin: http://monsite.test				
	Referer: http://monsite.test/guestbook/				
	User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/73.0.3683.103 Safari/537.36				
	▼ Form Data	view source	view URL encoded		
	name: Patrick Patrick				
	email: patrick.croquet@afpa.fr				
	message: Coucou				

The response status is 400 Bad Request.

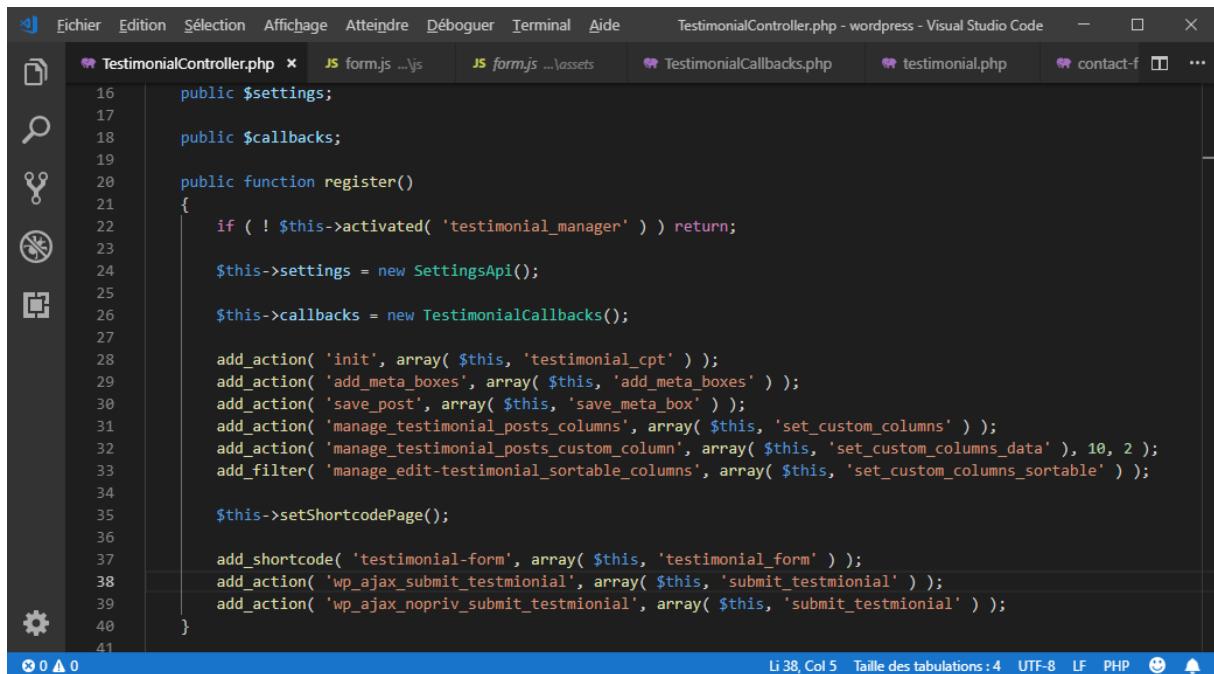
The screenshot shows the Visual Studio Code interface with the "contact-form.php" file open. The code is as follows:

```

Eichier Edition Sélection Affichage Atteindre Déboguer Terminal Aide contact-form.php - wordpress - Visual Studio Code
nialController.php JS form.js ...js JS formjs ...assets TestimonialCallbacks.php testimonial.php contact-form.php ...
1 <form id="alecaddd-testimonial-form" action="#" method="post" data-url="php echo admin_url('admin-ajax.php'); ?&gt;
2
3   &lt;div class="field-container"&gt;
4     &lt;input type="text" class="field-input" placeholder="Your Name" id="name" name="name" required&gt;
5     &lt;small class="field-msg error" data-error="invalidName"&gt;Your Name is Required&lt;/small&gt;
6   &lt;/div&gt;
7
8   &lt;div class="field-container"&gt;
9     &lt;input type="email" class="field-input" placeholder="Your Email" id="email" name="email" required&gt;
10    &lt;small class="field-msg error" data-error="invalidEmail"&gt;The Email address is not valid&lt;/small&gt;
11  &lt;/div&gt;
12
13  &lt;div class="field-container"&gt;
14    &lt;textarea name="message" id="message" class="field-input" placeholder="Your Message" required&gt;&lt;/textarea&gt;
15    &lt;small class="field-msg error" data-error="invalidMessage"&gt;A Message is Required&lt;/small&gt;
16  &lt;/div&gt;
17
18  &lt;div class="field-container"&gt;
19    &lt;div&gt;
20      &lt;button type="submit" class="btn btn-default btn-lg btn-sunset-form"&gt;Submit&lt;/button&gt;
21    &lt;/div&gt;
22    &lt;small class="field-msg js-form-submission"&gt;Submission in process, please wait&amp;hellip;&lt;/small&gt;
23    &lt;small class="field-msg success js-form-success"&gt;Message Successfully submitted, thank you!&lt;/small&gt;
24    &lt;small class="field-msg error js-form-error"&gt;There was a problem with the Contact Form, please try again!&lt;/small&gt;
25  &lt;/div&gt;
26
27  &lt;input type="hidden" name="action" value="submit_testimonial"&gt;
28
29 &lt;/form&gt;
</pre

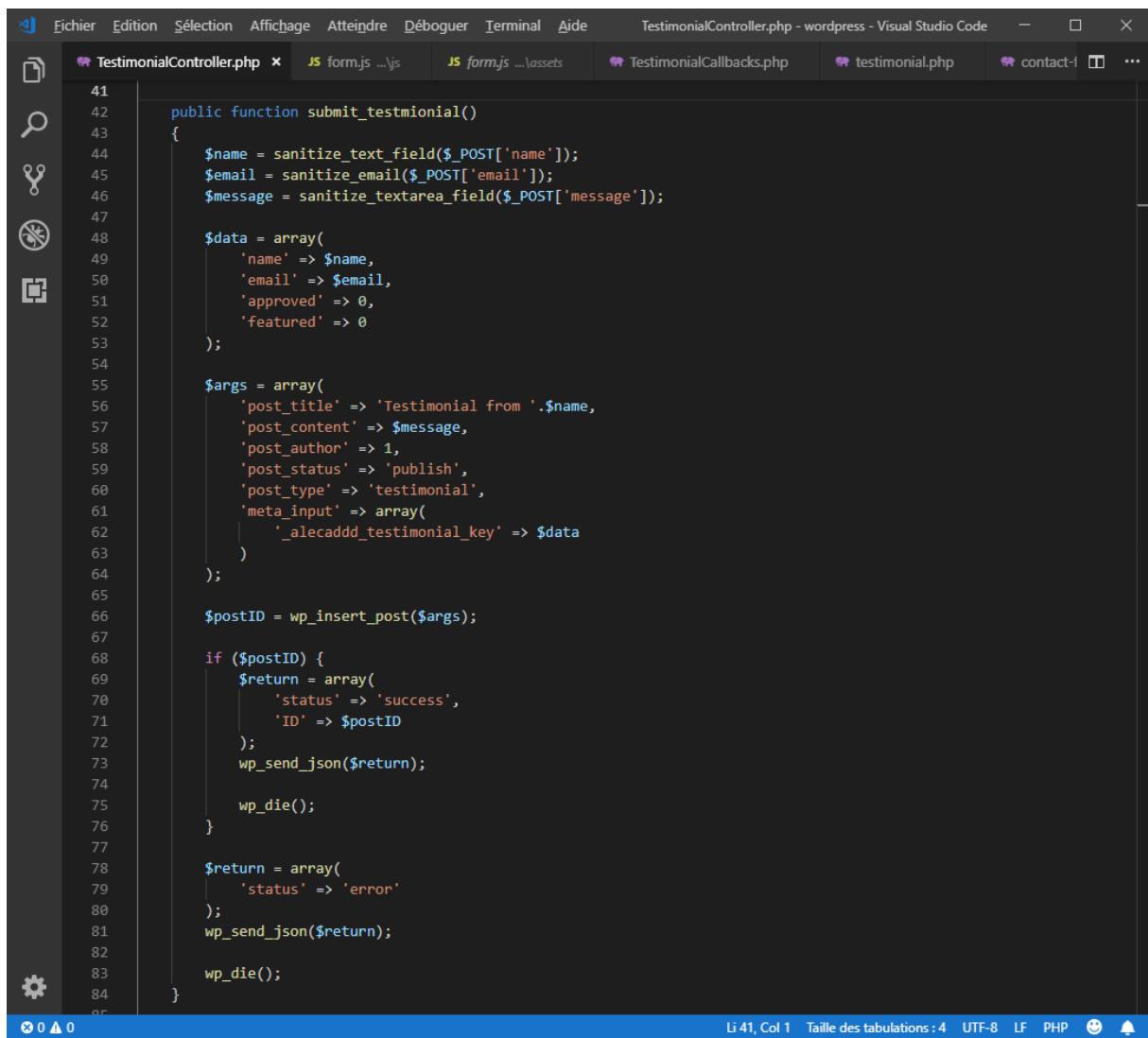
The status bar at the bottom indicates: Li 27, Col 1 Taille des tabulations : 4 UTF-8 LF PHP ⓘ 🔔


```



```
16     public $settings;
17
18     public $callbacks;
19
20     public function register()
21     {
22         if ( ! $this->activated( 'testimonial_manager' ) ) return;
23
24         $this->settings = new SettingsApi();
25
26         $this->callbacks = new TestimonialCallbacks();
27
28         add_action( 'init', array( $this, 'testimonial_cpt' ) );
29         add_action( 'add_meta_boxes', array( $this, 'add_meta_boxes' ) );
30         add_action( 'save_post', array( $this, 'save_meta_box' ) );
31         add_action( 'manage_testimonial_posts_columns', array( $this, 'set_custom_columns' ) );
32         add_action( 'manage_testimonial_posts_custom_column', array( $this, 'set_custom_columns_data' ), 10, 2 );
33         add_filter( 'manage_edit-testimonial_sortable_columns', array( $this, 'set_custom_columns_sortable' ) );
34
35         $this->setshortcodePage();
36
37         add_shortcode( 'testimonial-form', array( $this, 'testimonial_form' ) );
38         add_action( 'wp_ajax_submit_testimonial', array( $this, 'submit_testimonial' ) );
39         add_action( 'wp_ajax_nopriv_submit_testimonial', array( $this, 'submit_testimonial' ) );
40     }

```



```
41     public function submit_testimonial()
42     {
43         $name = sanitize_text_field($_POST['name']);
44         $email = sanitize_email($_POST['email']);
45         $message = sanitize_textarea_field($_POST['message']);
46
47         $data = array(
48             'name' => $name,
49             'email' => $email,
50             'approved' => 0,
51             'featured' => 0
52         );
53
54         $args = array(
55             'post_title' => 'Testimonial from '.$name,
56             'post_content' => $message,
57             'post_author' => 1,
58             'post_status' => 'publish',
59             'post_type' => 'testimonial',
60             'meta_input' => array(
61                 '_alecadd_testimonial_key' => $data
62             )
63         );
64
65         $postID = wp_insert_post($args);
66
67         if ($postID) {
68             $return = array(
69                 'status' => 'success',
70                 'ID' => $postID
71             );
72             wp_send_json($return);
73
74             wp_die();
75         }
76
77         $return = array(
78             'status' => 'error'
79         );
80         wp_send_json($return);
81
82         wp_die();
83     }

```

The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** On the left, showing icons for files like 'TestimonialController.php', 'JS form.js ...js', 'JS form.js ...assets', 'TestimonialCallbacks.php', 'testimonial.php', and 'contact-l'.
- Code Editor:** The main area displays a block of JavaScript code. The code handles form validation and an AJAX POST request to submit a testimonial. It uses promises and the fetch API.
- Status Bar:** At the bottom, it shows 'Li 58, Col 1 Taille des tabulations : 4 UTF-8 LF JavaScript' and some status icons.

```
11 let data = {
12   name: testimonialForm.querySelector('[name="name"]').value,
13   email: testimonialForm.querySelector('[name="email"]').value,
14   message: testimonialForm.querySelector('[name="message"]').value,
15 }
16
17 // validate the email
18 if (!validateEmail(data.email)) {
19   testimonialForm.querySelector('[data-error="invalidEmail"]').classList.add('show');
20   return;
21 }
22
23 if (!data.name) {
24   testimonialForm.querySelector('[data-error="invalidName"]').classList.add('show');
25   return;
26 }
27
28 if (!data.message) {
29   testimonialForm.querySelector('[data-error="invalidMessage"]').classList.add('show');
30   return;
31 }
32
33 // ajax http post request
34 let url = testimonialForm.dataset.url;
35 let params = new URLSearchParams(new FormData(testimonialForm));
36
37 testimonialForm.querySelector('.js-form-submission').classList.add('show');
38
39 fetch(url, {
40   method: "POST",
41   body: params,
42 }).then(res => res.json())
43   .catch(error => {
44     resetMessages();
45     testimonialForm.querySelector('.js-form-error').classList.add('show');
46   })
47   .then(response => {
48     resetMessages();
49
50     if (response === 0 || response.status === 'error') {
51       testimonialForm.querySelector('.js-form-error').classList.add('show');
52       return;
53     }
54
55     testimonialForm.querySelector('.js-form-success').classList.add('show');
56     resetForm(testimonialForm);
57   });
58
59 })
60});
```

https://www.youtube.com/watch?v=J4KIUaUnA5c&list=PLriKzYyLb28kR_CPMz8uierDW_C2y3znI2&index=54

GuestBook – Mon site

localhost:3000/guestbook/

MON SITE

Un site utilisant WordPress

GUESTBOOK

livre d'or

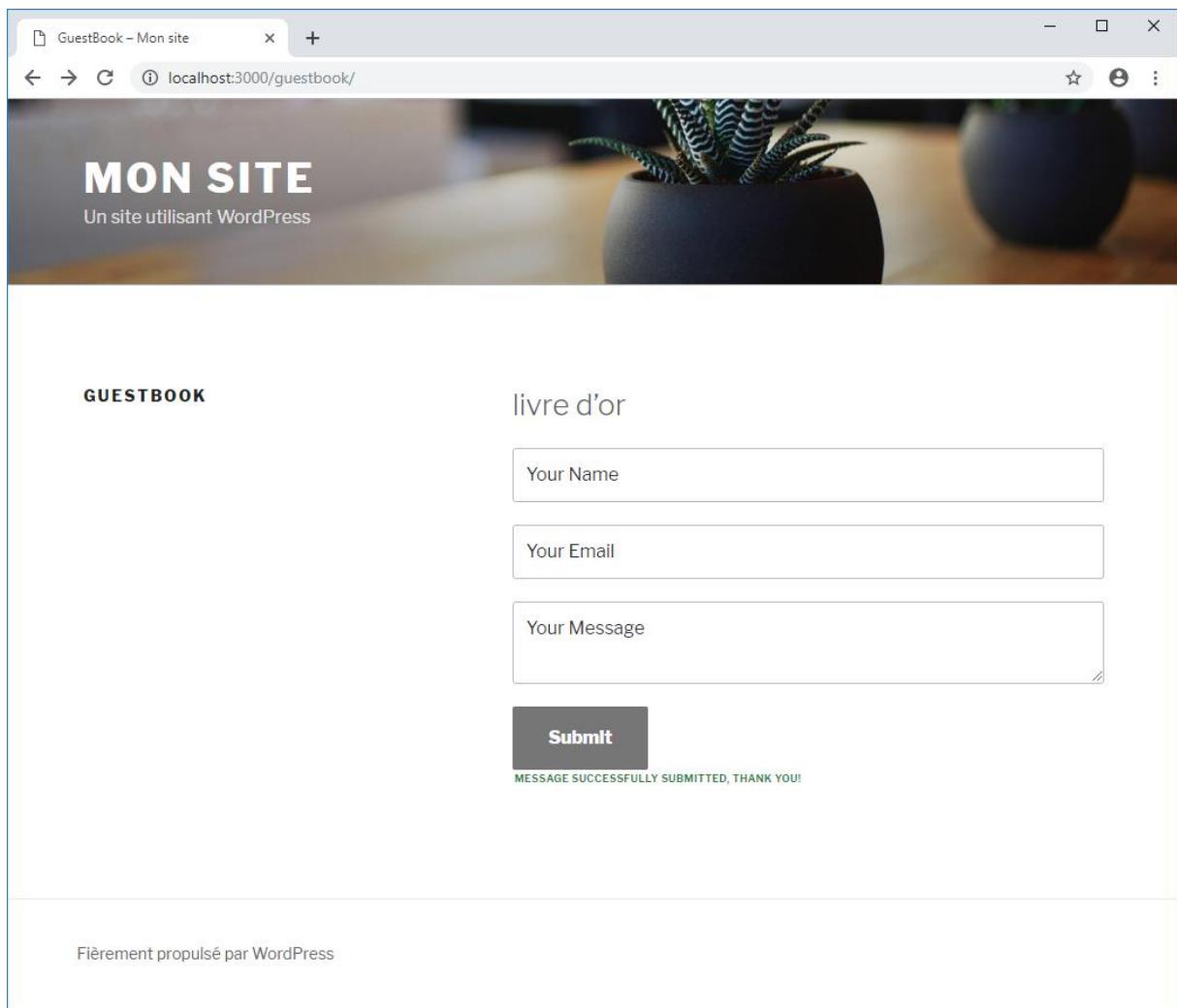
Patrick Croquet

patrick.croquet@afpa.fr

C'est mon message de test.

Submit

Fièrement propulsé par WordPress



Testimonials < Mon site — WordPress

localhost:3000/wp-admin/edit.php?post_type=testimonial&ids=39

Bonjour, Patrick

Tableau de bord

Articles Products Comic books Médias Pages Commentaires Testimonials

Testimonials Ajouter Shortcodes

Apparence Extensions Utilisateurs Outils Réglages Alecaddd Réduire le menu

Testimonials Ajouter

1 publication déplacée dans la corbeille. Annuler

Tous (2) | Publié (2) | Corbeille (5)

Actions groupées Appliquer Toutes les dates Filtrer 2 éléments

<input type="checkbox"/> Author Name	Titre	Approved	Featured	Date
<input type="checkbox"/> Patrick Croquet patrick.croquet@afpa.fr	Testimonial from Patrick Croquet	NO	NO	Publié il y a 1 minute

<input type="checkbox"/> Author Name	Titre	Approved	Featured	Date
<input type="checkbox"/> Patrick patrick.croquet@afpa.fr	Le grognement de l'ours	YES	YES	Publié 24/04/2019

Actions groupées Appliquer 2 éléments

Merci de faire de WordPress votre outil de création.

Version 5.1.1

The screenshot shows the WordPress admin dashboard for the 'Testimonials' section. The left sidebar has a dark theme with white text. The 'Testimonials' menu item is highlighted in blue. The main content area shows a table of testimonials with columns for Author Name, Titre, Approved, Featured, and Date. One testimonial is marked as moved to the trash ('Corbeille'). The bottom right corner indicates the version is 5.1.1.

Modifier l'article < Mon site — W +

localhost:3000/wp-admin/post.php?post=40&action=edit

Bonjour, Patrick

Tableau de bord

Articles Products Comic books Médias Pages Commentaires Testimonials Ajouter Shortcodes Apparence Extensions Utilisateurs Outils Réglages Alecaddd Réduire le menu

Modifier l'article Ajouter

Testimonial from Patrick Croquet

Ajouter un média Visuel Texte

Paragraphe B I Émphasis Égaliser Étiquette

C'est mon message de test.

Nombre de mots : 5 Dernière modification le 29 avril 2019 à 14 h 00 min

Publier

État : Publié [Modifier](#)

Visibilité : Public [Modifier](#)

Publié le : 29 avril 2019 à 14 h 00 min [Modifier](#)

Déplacer dans la corbeille

Mettre à jour

Testimonial Options

Author Name: Patrick Croquet

Author Email: patrick.croquet@afpa.fr

Approved

Featured

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a folder structure with files like "form.js", "TestimonialController.php", and "form.js ...\\assets".
- Search Bar:** Displays the file name "TestimonialController.php".
- Code Editor:** The main area contains PHP code for a testimonial submission controller. The code includes sanitization of input fields (\$name, \$email, \$message), creation of post arguments (\$args) including meta inputs, and insertion of the post using wp_insert_post(). It also handles returning JSON responses based on success or error.
- Bottom Status Bar:** Shows the current file is "TestimonialController.php - wordpress - Visual Studio Code", the line number is "Li 41, Col 1", the tab size is "Taille des tabulations : 4", and the file type is "UTF-8 LF PHP".

```
JS form.js ...js TestimonialController.php x JS form.js ...\\assets

41     public function submit_testimonial()
42     {
43         if (! DOING_AJAX || ! check_ajax_referer('testimonial-nonce', 'nonce') ) {
44             return $this->return_json('error');
45         }
46
47
48         $name = sanitize_text_field($_POST['name']);
49         $email = sanitize_email($_POST['email']);
50         $message = sanitize_textarea_field($_POST['message']);
51
52         $data = array(
53             'name' => $name,
54             'email' => $email,
55             'approved' => 0,
56             'featured' => 0,
57         );
58
59         $args = array(
60             'post_title' => 'Testimonial from ' . $name,
61             'post_content' => $message,
62             'post_author' => 1,
63             'post_status' => 'publish',
64             'post_type' => 'testimonial',
65             'meta_input' => array(
66                 '_alecaddd_testimonial_key' => $data
67             )
68         );
69
70         $postID = wp_insert_post($args);
71
72         if ($postID) {
73             return $this->return_json('success');
74         }
75
76         return $this->return_json('error');
77     }
```

The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** Shows a file named "form.js" under the "assets" folder.
- Code Editor:** Displays the following JavaScript code:

```
document.addEventListener('DOMContentLoaded', function(e) {
  let testimonialForm = document.getElementById('alecaddd-testimonial-form');

  testimonialForm.addEventListener('submit', (e) => {
    e.preventDefault();

    // reset the form messages
    resetMessages();

    // collect all the data
    let data = {
      name: testimonialForm.querySelector('[name="name"]').value,
      email: testimonialForm.querySelector('[name="email"]').value,
      message: testimonialForm.querySelector('[name="message"]').value,
      nonce: testimonialForm.querySelector('[name="nonce"]').value
    }
  });
});
```

The code handles the DOMContentLoaded event to get the testimonial form by ID. It then adds a submit event listener to prevent the default submission. Inside the submit event listener, it calls a function to reset messages and collects data from input fields: name, email, message, and nonce. The data is stored in a variable named "data".

```

1 <?php
2 /**
3  * @package AlecadddPlugin
4 */
5 namespace Inc\Base;
6
7 use Inc\Api\SettingsApi;
8 use Inc\Base\BaseController;
9 use Inc\Api\Callbacks\TestimonialCallbacks;
10
11 /**
12 *
13 */
14 class TestimonialController extends BaseController
15 {
16     public $settings;
17
18     public $callbacks;
19
20     public function register()
21     {
22         if ( ! $this->activated( 'testimonial_manager' ) ) return;
23
24         $this->settings = new SettingsApi();
25
26         $this->callbacks = new TestimonialCallbacks();
27
28         add_action( 'init', array( $this, 'testimonial_cpt' ) );
29         add_action( 'add_meta_boxes', array( $this, 'add_meta_boxes' ) );
30         add_action( 'save_post', array( $this, 'save_meta_box' ) );
31         add_action( 'manage_testimonial_posts_columns', array( $this, 'set_custom_columns' ) );
32         add_action( 'manage_testimonial_posts_custom_column', array( $this, 'set_custom_columns_data' ), 10, 2 );
33         add_filter( 'manage_edit-testimonial_sortable_columns', array( $this, 'set_custom_columns_sortable' ) );
34
35         $this->setshortcodePage();
36
37         add_shortcode( 'testimonial-form', array( $this, 'testimonial_form' ) );
38         add_shortcode( 'testimonial-slideshow', array( $this, 'testimonial_slideshow' ) );
39         add_action( 'wp_ajax_submit_testimonial', array( $this, 'submit_testimonial' ) );
40         add_action( 'wp_ajax_no_priv_submit_testimonial', array( $this, 'submit_testimonial' ) );
41     }

```

Li 38, Col 1 Taille des tabulations : 4 UTF-8 LF PHP ⚡ 📁

```

98
99     public function testimonial_slideshow()
100    {
101        ob_start();
102        echo "<link rel='stylesheet' href='".$this->plugin_url/assets/slider.css'" type='text/css' media='all' />";
103        require_once( "$this->plugin_path/templates/slider.php" );
104        echo "<script src='".$this->plugin_url/assets/slider.js'></script>";
105        return ob_get_clean();
106    }

```

Li 98, Col 1 Taille des tabulations : 4 UTF-8 LF PHP ⚡ 📁

EXPLORATEUR

- ÉDITEURS OUVERTS
- WORDPRESS
 - node_modules
 - src
 - js
 - slider.js**
 - form.js
- STRUCTURE

JS form.js TestimonialController.php

```

98
99     public function testimonial_slideshow()
100    {
101        ob_start();
102        echo "<link rel='stylesheet' href='".$this->plugin_url/assets/slider.css'" type='text/css' media='all' />";
103        require_once( "$this->plugin_path/templates/slider.php" );
104        echo "<script src='".$this->plugin_url/assets/slider.js'></script>";
105        return ob_get_clean();
106    }

```

Li 98, Col 1 Taille des tabulations : 4 UTF-8 LF PHP ⚡ 📁

The screenshot shows the Visual Studio Code interface. The title bar reads "slider.js - wordpress - Visual Studio Code". The left sidebar displays a file tree under "WORDPRESS" with "myscript.js" and "slider.js" selected. The main editor area contains the following JavaScript code:

```
1  document.addEventListener('DOMContentLoaded', function(e) {
2      //
3  });
```

The status bar at the bottom shows "Li 3, Col 4" and "JavaScript".

The screenshot shows the Visual Studio Code interface. The title bar reads "gulpfile.js - wordpress - Visual Studio Code". The left sidebar displays a file tree with "form.js" selected. The main editor area contains the following JavaScript code:

```
29
30 // Project related variables
31 var projectURL = 'http://monsite.test';
32
33 var styleSRC = 'src/scss/mystyle.scss';
34 var styleForm = 'src/scss/form.scss';
35 var styleSlider = 'src/scss/slider.scss';
36 var styleURL = './assets/';
37 var mapURL = './';
38
39 var jsSRC = 'src/js/';
40 var jsAdmin = 'myscript.js';
41 var jsForm = 'form.js';
42 var jsSlider = 'slider.js';
43 var jsFiles = [jsAdmin, jsForm, jsSlider];
44 var jsURL = './assets/';
45
46 var styleWatch = 'src/scss/**/*.scss';
47 var jsWatch = 'src/js/**/*.js';
48 var phpWatch = '**/*.php';
49
```

The status bar at the bottom shows "Li 29, Col 1" and "JavaScript".

The screenshot shows the Visual Studio Code interface. The title bar reads "gulpfile.js - wordpress - Visual Studio Code". The left sidebar displays a file tree with "form.js" selected. The main editor area contains the following JavaScript code:

```
62
63 gulp.task( 'styles', function() {
64     gulp.src( [styleSRC, styleForm, styleSlider] )
65         .pipe( sourcemaps.init() )
66         .pipe( sass({
67             errLogToConsole: true,
68             outputStyle: 'compressed'
69         }) )
70         .on( 'error', console.error.bind( console ) )
71         .pipe( autoprefixer({ browsers: [ 'last 2 versions', '> 5%', 'Firefox ESR' ] }) )
72         .pipe( sourcemaps.write( mapURL ) )
73         .pipe( gulp.dest( styleURL ) )
74         .pipe( browserSync.stream() );
75 });
76
```

The status bar at the bottom shows "Li 62, Col 1" and "JavaScript".

```
PS C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin> gulp watch
[15:10:41] Using gulpfile C:\wamp64\www\wordpress\wp-content\plugins\monpremier-plugin\gulpfile.js
[15:10:41] Starting 'styles'...
[15:10:41] Finished 'styles' after 9.06 ms
[15:10:41] Starting 'js'...
[15:10:41] Finished 'js' after 9.87 ms
[15:10:41] Starting 'default'...
[15:10:41] Finished 'default' after 571 µs
[15:10:41] Starting 'browser-sync'...
[15:10:41] Finished 'browser-sync' after 8.2 ms
[15:10:41] Starting 'watch'...
[15:10:46] Finished 'watch' after 4.94 s
[Browsersync] Proxying: http://monsite.test
[Browsersync] Access URLs:
Local: http://localhost:3000
External: http://172.17.82.73:3000
UI:
External: http://localhost:3001
[Browsersync] 4 files changed (mystyle.css.map, mystyle.css, form.css.map, form.css)
[Browsersync] 2 files changed (slider.js.map, slider.js)
[Browsersync] 2 files changed (form.js.map, form.js)
[Browsersync] 2 files changed (myscript.js.map, myscript.js)
[Browsersync] Reloading Browsers... (buffered 8 events)
[15:12:04] Starting 'styles'...
[15:12:04] Finished 'styles' after 2.05 ms
[Browsersync] 6 files changed (mystyle.css.map, mystyle.css, form.css.map, form.css, slider.css.map, slider.css)
```

```
EXPLORATEUR
ÉDITEURS OUVERTS
WORDPRESS
templates
admin.php
contact-form.php
contact-formold.php
cpt.php
cptold.php
cptold2.php
cptold2.php
taxonomy.php
```

```
1 <h1>Slider</h1>
2 
```

The screenshot shows the WordPress admin interface. The left sidebar is dark-themed and includes links for Articles, Products, Comic books, Médias, Pages, Commentaires, and Testimonials. The 'Testimonials' link is highlighted in blue, indicating the current active page. Below the sidebar, there's a sub-menu for Shortcodes, which is also highlighted in blue. The main content area has a title 'Testimonial Shortcode'. It contains two sections: 'Testimonial Form Shortcode' with the code '[testimonial-form]' and 'Testimonial SlideShow Shortcode' with the code '[testimonial-slideshow]'. At the bottom of the content area, there's a footer note 'Merci de faire de [WordPress](#) votre outil de création.' and the text 'Version 5.1.1'.

The screenshot shows the WordPress post editor interface. The left sidebar has a dark theme with white icons and labels. The main content area displays a page titled "Testimonial". The editor toolbar contains standard rich-text editing tools like bold, italic, and alignment options. A visual editor panel below the toolbar shows the shortcode "[testimonial-slideshow]" highlighted. The right sidebar contains a metabox for "Document" settings, showing "Visibilité" as "Public" and "Publier" as "29 avril 2019 3:23". There are also sections for "Déplacer vers la corbeille", "Permalien", "Image mise en avant", "Discussion", and "Attributs de page". The top navigation bar shows the URL "monsite.test/wp-admin/post.php?post=42&action=edit".

A screenshot of a WordPress dashboard. The top navigation bar shows the title "Testimonial – Mon site" and the URL "Non sécurisé | monsite.test/testimonial/". Below the navigation are buttons for "Mon site", "Personnaliser", "Créer", and "Modifier la page". On the right, a user "Patrick" is logged in with a "Bonjour, Patrick" greeting. The main content area features a large banner image of a succulent in a dark pot. Overlaid on the banner is the text "MON SITE" in large bold letters and "Un site utilisant WordPress" below it. Below the banner, there's a section titled "TESTIMONIAL" with a "Modifier" link. A large "Slider" placeholder is visible. At the bottom, a footer bar says "Fièrement propulsé par WordPress".

Modifier l'article < Mon site — W x +

Non sécurisé | monsite.test/wp-admin/post.php?post=40&action=edit

Bonjour, Patrick

Tableau de bord

- Articles
- Products
- Comic books
- Médias
- Pages
- Commentaires
- Testimonials**

Ajouter

Shortcodes

Apparence

Extensions

Utilisateurs

Outils

Réglages

Alecaddd

Réduire le menu

Modifier l'article [Ajouter](#)

Testimonial from Patrick Croquet

Ajouter un média Visuel Texte

Paragraphe B I Éléments

C'est mon message de test.

Nombre de mots : 5 Dernière modification le 29 avril 2019 à 14 h 00 min

Publier

État : Publié [Modifier](#)

Visibilité : Public [Modifier](#)

Publié le : 29 avril 2019 à 14 h 00 min [Modifier](#)

Déplacer dans la corbeille

[Mettre à jour](#)

Testimonial Options

Author Name : Patrick Croquet

Author Email : patrick.croquet@afpa.fr

Approved :

Featured :

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

This screenshot shows the WordPress admin dashboard for editing a testimonial. The left sidebar has 'Testimonials' selected. The main area shows a testimonial from 'Patrick Croquet' with the text 'C'est mon message de test.' The right sidebar contains publishing options like visibility and approval status, and testimonial-specific options for author name and email. At the bottom, there's a note about WordPress and the current version.

The screenshot shows the WordPress Testimonials dashboard. The sidebar on the left includes links for Tableau de bord, Articles, Products, Comic books, Médias, Pages, Commentaires, Testimonials (selected), Ajouter, Shortcodes, Apparence, Extensions, Utilisateurs, Outils, Réglages, Alecaddd, and Réduire le menu. The main area displays a list of three testimonial entries:

Author Name	Titre	Approved	Featured	Date
Patrick patrick.croquet@afpa.fr	Testimonial from Patrick	NO	NO	Publié il y a 2 heures
Patrick Croquet patrick.croquet@afpa.fr	Testimonial from Patrick Croquet	YES	YES	Publié il y a 2 heures
Patrick patrick.croquet@afpa.fr	Le grognement de l'ours	YES	YES	Publié 24/04/2019

Below the table, there are buttons for Actions groupées, Appliquer, Toutes les dates, Filtrer, and another Actions groupées button.

The screenshot shows a Visual Studio Code editor window with several tabs open. The active tab is 'form.js'. The code in 'form.js' is as follows:

```

1 <?php
2
3 $args = array(
4     'post_type' => 'testimonial',
5     'post_status' => 'publish',
6     'posts_per_page' => 5,
7     'meta_query' => array(
8         array(
9             'key' => '_alecaddd_testimonial_key',
10            'value' => 's:8:"approved";i:1;s:8:"featured";i:1;',
11            'compare' => 'LIKE'
12        )
13    );
14 );
15
16 $query = new WP_Query( $args );
17
18 if ($query->have_posts()) :
19     echo '<ul>';
20     while ($query->have_posts()) : $query->the_post();
21         echo '<li>' . get_the_title() . '<p>' . get_the_content() . '</p></li>';
22     endwhile;
23     echo '</ul>';
24 endif;
25
26 wp_reset_postdata();

```

localhost / MySQL / wordpress / +

localhost/phpmyadmin/sql.php?db=wordpress&table=wp_postmeta&pos=0

Serveur: MySQL:3306 » Base de données: wordpress » Table: wp_postmeta

Parcourir Structure SQL Rechercher Insérer Exporter Importer Priviléges Opérations Plus

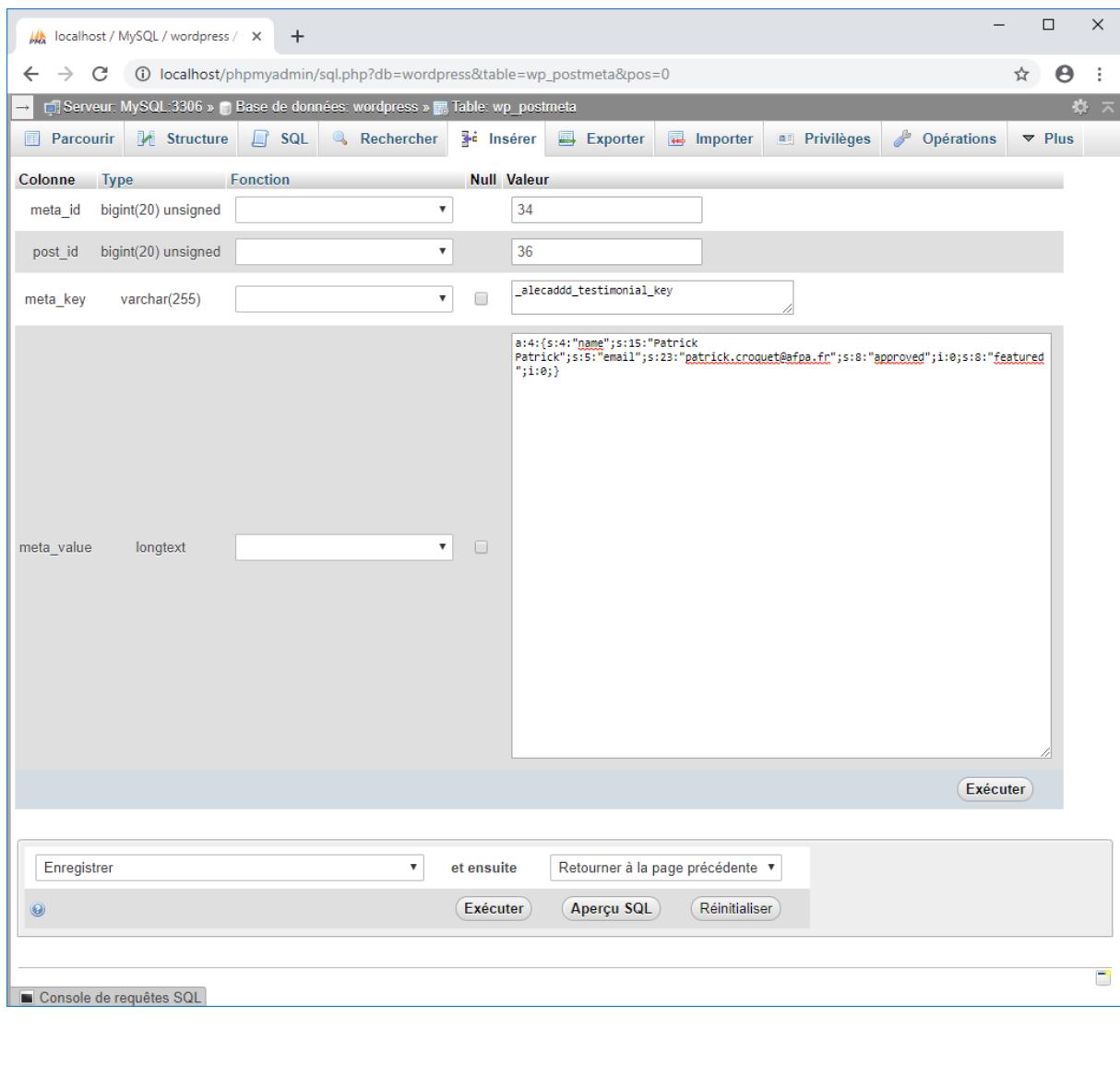
Colonne	Type	Fonction	Null	Valeur
meta_id	bigint(20) unsigned			34
post_id	bigint(20) unsigned			36
meta_key	varchar(255)			_alecaddd_testimonial_key
meta_value	longtext			<pre>b:4:{s:4:"name";s:15:"Patrick Patrick";s:5:"email";s:23:"patrick.croquet@afpa.fr";s:8:"approved";i:0;s:8:"featured";i:0;}</pre>

Exécuter

Enregistrer et ensuite Retourner à la page précédente

Exécuter Aperçu SQL Réinitialiser

Console de requêtes SQL



Testimonial – Mon site

Non sécurisé | monsite.test/testimonial/?preview_id=42&preview_nonce=b9a25b370b&preview=true

Mon site Personnaliser 0 + Créez Modifier la page Bonjour, Patrick

MON SITE

Un site utilisant WordPress

TESTIMONIAL

Modifier

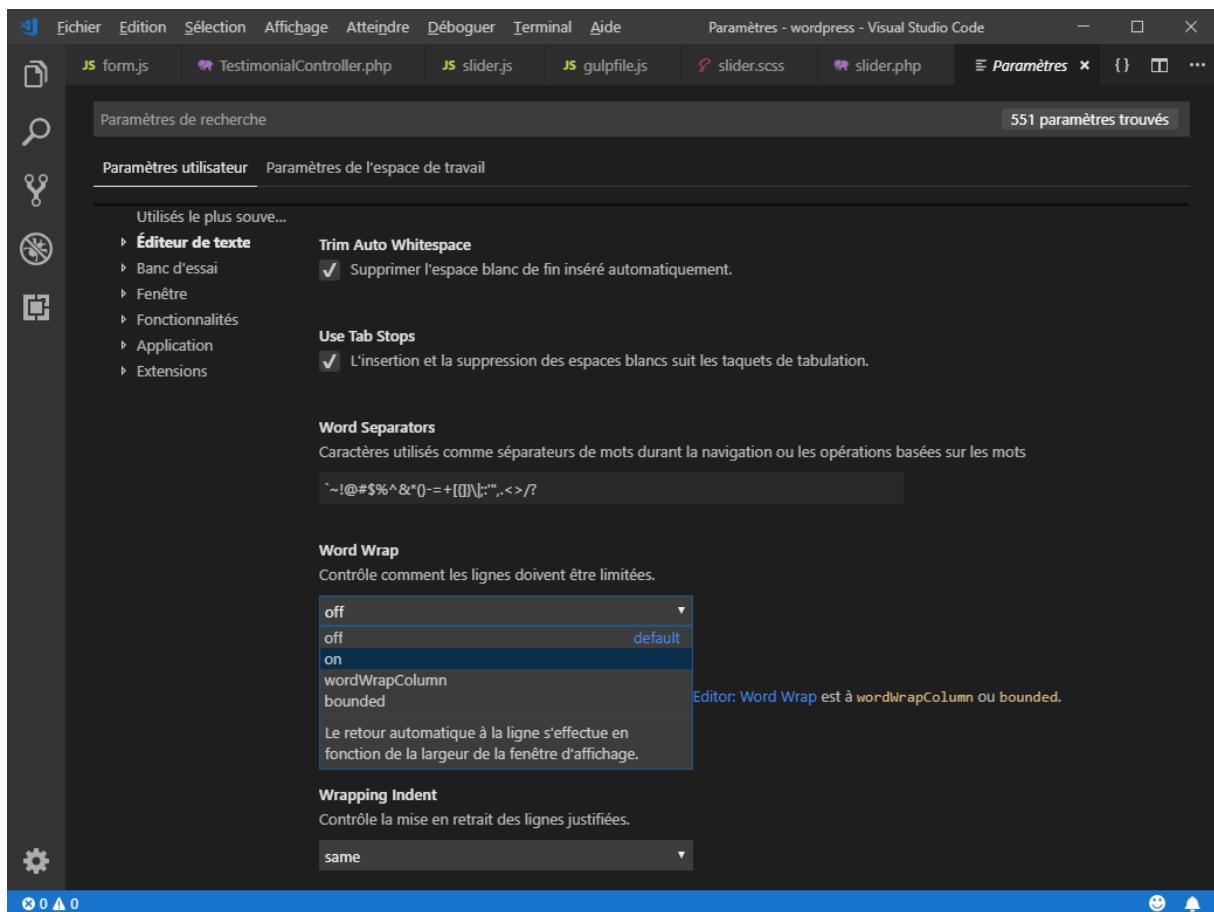
- Testimonial from Patrick Croquet
C'est mon message de test.
- Le grognement de l'ours
L'ours grogne.



Ours qui grogne

À regarder plus tard Partager

▶



The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** Shows files like form.js, TestimonialController.php, slider.js, gulpfile.js, slider.scss, and slider.php.
- Editor:** Displays the content of the slider.php file. The code is a PHP script that queries the database to retrieve testimonial posts and displays them in a slider. It uses the WP_Query class and loops through the results to echo HTML for each testimonial.
- Bottom Status Bar:** Shows file statistics (0 lines, 0 changes), encoding (UTF-8), and other settings.

```
wp-content > plugins > monpremier-plugin > templates > slider.php
1  <?php
2
3  $args = array(
4      'post_type' => 'testimonial',
5      'post_status' => 'publish',
6      'posts_per_page' => 5,
7      'meta_query' => array(
8          array(
9              'key' => '_alecaddd_testimonial_key',
10             'value' => 's:8:"approved";i:1;s:8:"featured";i:1;',
11             'compare' => 'LIKE'
12         )
13     );
14 );
15
16 $query = new WP_Query( $args );
17
18 if ($query->have_posts()) :
19     $i = 1;
20
21     echo '<div class="ac-slider--wrapper"><div class="ac-slider--container"><div class="ac-slider--view"><ul>';
22
23     while ($query->have_posts()) : $query->the_post();
24         $name = get_post_meta( get_the_ID(), '_alecaddd_testimonial_key', true )[ 'name' ] ?? '';
25
26         echo '<li class="ac-slider--view__slides' . ($i === 1 ? ' is-active' : '') . '"><p
27             class="testimonial-quote">' . get_the_content() . "</p><p class='testimonial-author'>~ '$name.' ~</p></li>";
28
29         $i++;
30     endwhile;
31
32     echo '</ul></div><div class="ac-slider--arrows"><span class="arrow ac-slider--arrows__left">&#x3c;</span><span
33             class="arrow ac-slider--arrows__right">&#x3e;</span></div></div>';
34
35 endif;
36
37 wp_reset_postdata();
```

Testimonial – Mon site

Non sécurisé | monsite.test/testimonial/

Mon site Personnaliser Créez Modifier la page Bonjour, Patrick

TESTIMONIAL

Modifier

- "C'est mon message de test."

~ Patrick Croquet ~

- "L'ours grogne.

~ Patrick ~

<>

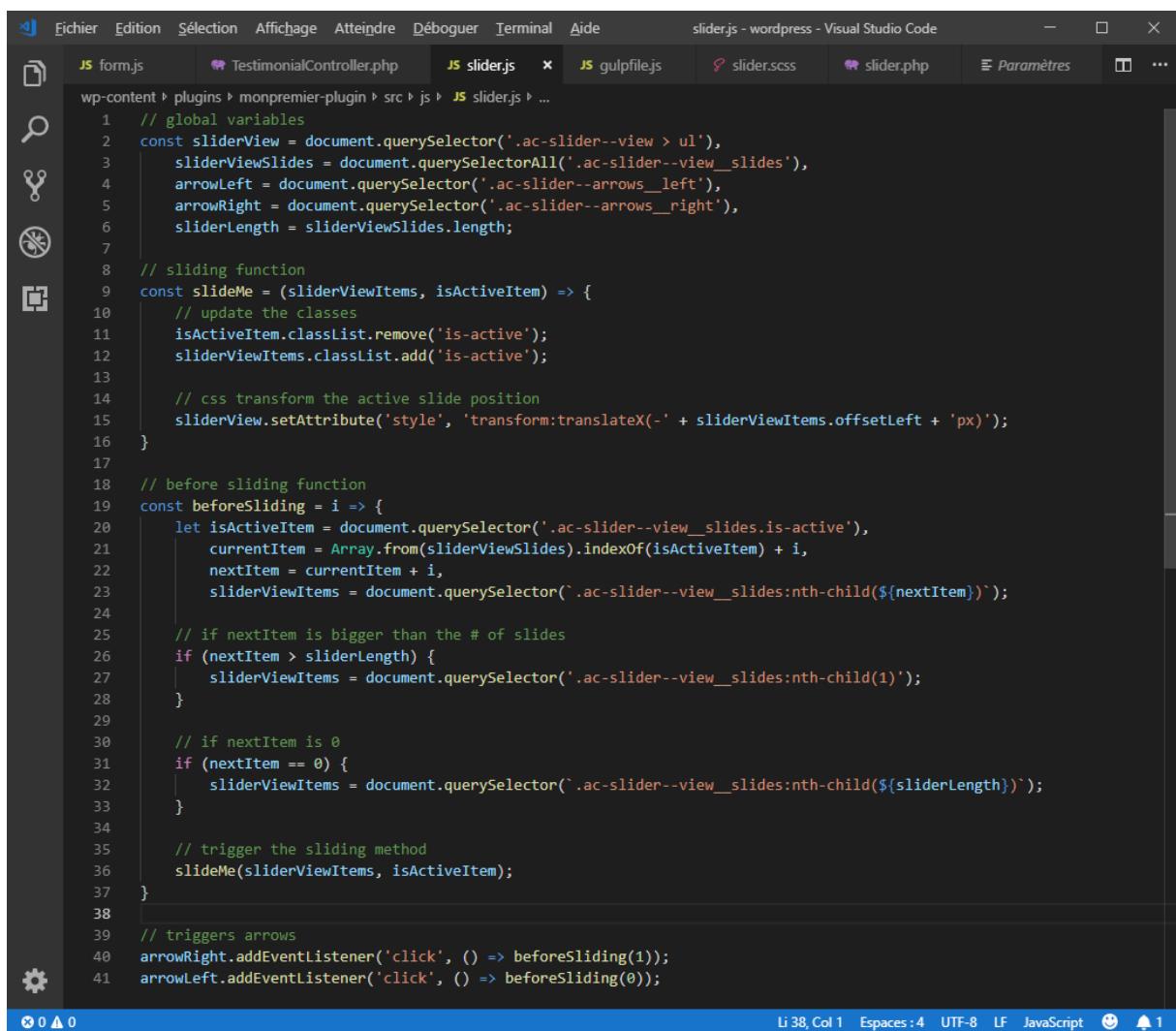
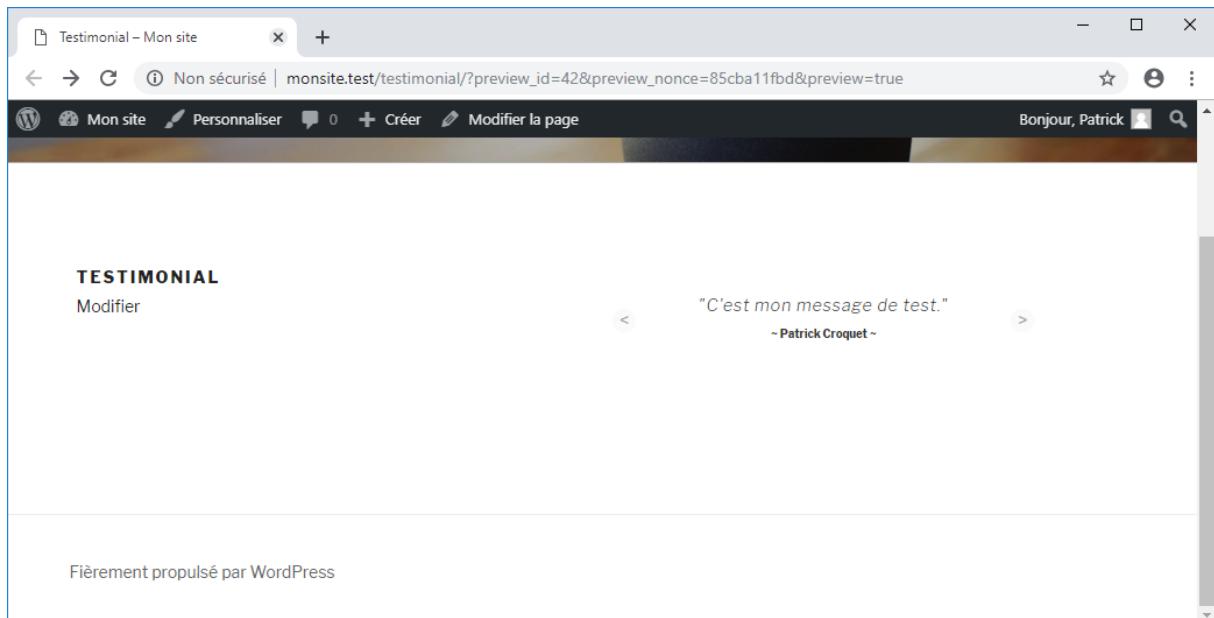
https://www.youtube.com/watch?v=DzKg9TMR7i4&list=PLriKzYyLb28kR_CPMz8uierDW_C2y3znI2&index=56

Eichier Edition Sélection Affichage Atteindre Déboguer Terminal Aide slider.scss - wordpress - Visual Studio Code

form.js TestimonialController.php slider.js gulpfile.js slider.scss slider.php Paramètres

```
wp-content › plugins › monpremier-plugin › src › sass › slider.scss › ...
1 .ac-slider--wrapper {
2   margin: 0 auto;
3   max-width: calc(90% - 150px);
4 }
5
6 .ac-slider--container {
7   position: relative;
8   width: 100%;
9   .ac-slider--view {
10     display: block;
11     width: 100%;
12     margin: 0 auto;
13     overflow: hidden;
14     ul {
15       white-space: nowrap;
16       text-align: center;
17       list-style: none;
18       margin: 0;
19       padding: 0;
20       transition: transform 0.25s ease-out;
21       .ac-slider--view__slides {
22         white-space: normal;
23         display: inline-block;
24         width: 100%;
25         position: relative;
26       }
27     }
28 }
```

Li 5, Col 1 Espaces : 4 UTF-8 LF SCSS 1 0 0 0

A screenshot of Visual Studio Code showing the "slider.js" file. The code is a JavaScript file for a slider component. It defines global variables for the slider view, slides, and arrows, and implements a "slideMe" function to handle slide transitions. It also defines a "beforeSliding" function to handle arrow clicks and trigger the sliding method. The code uses document.querySelectorAll and Array.from to select elements and find the active item.

```
wp-content > plugins > monpremier-plugin > inc > Base > TestimonialController.php

98     public function testimonial_slideshow()
99     {
100         ob_start();
101         echo "<link rel=\"stylesheet\" href=\"$this->plugin_url/assets/slider.css\" type=\"text/css\" media=\"all\" />";
102         require_once( $this->plugin_path/templates/slider.php );
103         echo "<script src=\"$this->plugin_url/assets/slider.js\"></script>";
104         return ob_get_clean();
105     }
106
107 }
```

Li 98, Col 1 Taille des tabulations : 4 UTF-8 LF PHP

```
wp-content > plugins > monpremier-plugin > inc > Base > TestimonialController.php

143     public function add_meta_boxes()
144     {
145         add_meta_box(
146             'testimonial_author',
147             'Testimonial Options',
148             array( $this, 'render_features_box' ),
149             'testimonial',
150             'side',
151             'default'
152         );
153     }
154
155 }
```

Li 143, Col 1 Taille des tabulations : 4 UTF-8 LF PHP

```
wp-content > plugins > monpremier-plugin > inc > Base > TestimonialController.php

155     public function render_features_box($post)
156     {
157         wp_nonce_field( 'alecaddd_testimonial', 'alecaddd_testimonial_nonce' );
158
159         $data = get_post_meta( $post->ID, '_alecaddd_testimonial_key', true );
160         $name = isset($data['name']) ? $data['name'] : '';
161         $email = isset($data['email']) ? $data['email'] : '';
162         $approved = isset($data['approved']) ? $data['approved'] : false;
163         $featured = isset($data['featured']) ? $data['featured'] : false;
164         ?>
165         <p>
166             <label class="meta-label" for="alecaddd_testimonial_author">Author Name</label>
167             <input type="text" id="alecaddd_testimonial_author" name="alecaddd_testimonial_author" class="widefat"
168             value=<?php echo esc_attr( $name ); ?>">
169         </p>
170         <p>
171             <label class="meta-label" for="alecaddd_testimonial_email">Author Email</label>
172             <input type="email" id="alecaddd_testimonial_email" name="alecaddd_testimonial_email" class="widefat"
173             value=<?php echo esc_attr( $email ); ?>">
174         </p>
175         <div class="meta-container">
176             <label class="meta-label w-50 text-left" for="alecaddd_testimonial_approved">Approved</label>
177             <div class="text-right w-50 inline">
178                 <div class="ui-toggle inline"><input type="checkbox" id="alecaddd_testimonial_approved"
179                 name="alecaddd_testimonial_approved" value="1" <?php echo $approved ? 'checked' : '' ; ?>>
180                 <label for="alecaddd_testimonial_approved"><div></div></label>
181             </div>
182             <div class="meta-container">
183                 <label class="meta-label w-50 text-left" for="alecaddd_testimonial_featured">Featured</label>
184                 <div class="text-right w-50 inline">
185                     <div class="ui-toggle inline"><input type="checkbox" id="alecaddd_testimonial_featured"
186                     name="alecaddd_testimonial_featured" value="1" <?php echo $featured ? 'checked' : '' ; ?>>
187                     <label for="alecaddd_testimonial_featured"><div></div></label>
188                 </div>
189             </div>
190         </div>
191     }
192 }
```

Li 155, Col 1 Taille des tabulations : 4 UTF-8 LF PHP

The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** Shows files like `m.js`, `TestimonialController.php`, `slider.js`, `TemplateController.php`, and `slider.js`.
- Editor:** Displays the `TestimonialController.php` file content.
- Code Content:**

```
192 public function save_meta_box($post_id)
193 {
194     if (! isset($_POST['alecaddd_testimonial_nonce'])) {
195         return $post_id;
196     }
197
198     $nonce = $_POST['alecaddd_testimonial_nonce'];
199     if (! wp_verify_nonce( $nonce, 'alecaddd_testimonial' )) {
200         return $post_id;
201     }
202
203     if ( defined( 'DOING_AUTOSAVE' ) && DOING_AUTOSAVE ) {
204         return $post_id;
205     }
206
207     if (! current_user_can( 'edit_post', $post_id ) ) {
208         return $post_id;
209     }
210
211     $data = array(
212         'name' => sanitize_text_field( $_POST['alecaddd_testimonial_author'] ),
213         'email' => sanitize_email( $_POST['alecaddd_testimonial_email'] ),
214         'approved' => isset($_POST['alecaddd_testimonial_approved']) ? 1 : 0,
215         'featured' => isset($_POST['alecaddd_testimonial_featured']) ? 1 : 0,
216     );
217     update_post_meta( $post_id, '_alecaddd_testimonial_key', $data );
218 }
219
220 }
```
- Status Bar:** Shows "Li 192, Col 1 Taille des tabulations : 4 UTF-8 LF PHP 🎉 🚙"

```

<?php
/*
Template Name: Two Columns Layout
*/
get_header(); ?>
<h1>Custom 2 Columns Template</h1>
<p>This is loading properly!</p>
<?php
get_footer();

```

Alecaddd Plugin < Mon site — Wix

Non sécurisé | monsite.test/wp-admin/admin.php?page=alecaddd_plugin

Bonjour, Patrick

Alecaddd Plugin

Manage Settings Updates About

Settings Manager

Manage the Sections and Features of this Plugin by activating the checkboxes from the following list.

Activate CPT Manager	<input checked="" type="checkbox"/>
Activate Taxonomy Manager	<input checked="" type="checkbox"/>
Activate Media Widget	<input checked="" type="checkbox"/>
Activate Gallery Manager	<input type="checkbox"/>
Activate Testimonial Manager	<input checked="" type="checkbox"/>
Activate Templates Manager	<input checked="" type="checkbox"/>
Activate Ajax Login/Signup	<input type="checkbox"/>
Activate Membership Manager	<input type="checkbox"/>
Activate Chat Manager	<input type="checkbox"/>

Enregistrer les modifications

Merci de faire de [WordPress](#) votre outil de création.

Version 5.1.1

Ajouter une nouvelle page < Mon site +

Non sécurisé | monsite.test/wp-admin/post-new.php?post_type=page

Bonjour, Patrick

Tableau de bord

Articles
Products
Comic books
Médias
Pages

Toutes les pages
Ajouter
Commentaires
Testimonials
Apparence
Extensions
Utilisateurs
Outils
Réglages
Alecaddd
Réduire le menu

ma nouvelle page

Commencez à écrire ou saisissez « / » pour choisir un bloc

Enregistrer le brouillon Prévisualiser Publier... ⚙️ ⋮

Document Bloc

État et visibilité

Visibilité Public
Publier Immédiatement
 En attente de relecture

Image mise en avant

Discussion

Attributs de page

Modèle :
Two Columns Layout
Modèle par défaut
Two Columns Layout
(Aucun parent)

Ordre 0

ma nouvelle page – Mon site +

Non sécurisé | monsite.test/ma-nouvelle-page/

Bonjour, Patrick

Mon site Personnaliser Créez Modifier la page

MON SITE
Un site utilisant WordPress

Les ours

24 AVRIL 2019 PAR PATRICK

Recherche...

ARTICLES RÉCENTS

Les ours

Bonjour tout le monde !

COMMENTAIRES RÉCENTS

Un commentateur WordPress dans Bonjour tout le monde !

ARCHIVES

```
front-page.php - wordpress - Visual Studio Code
```

EXPLORATEUR

ÉDITEURS OUVERTS

WORDPRESS

- incold3
- node_modules
- page-templates
 - front-page.php
 - two-columns-tpl.php
- src
- srcold

STRUCTURE

wp-content > plugins > monpremier-plugin > page-templates > front-page.php

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

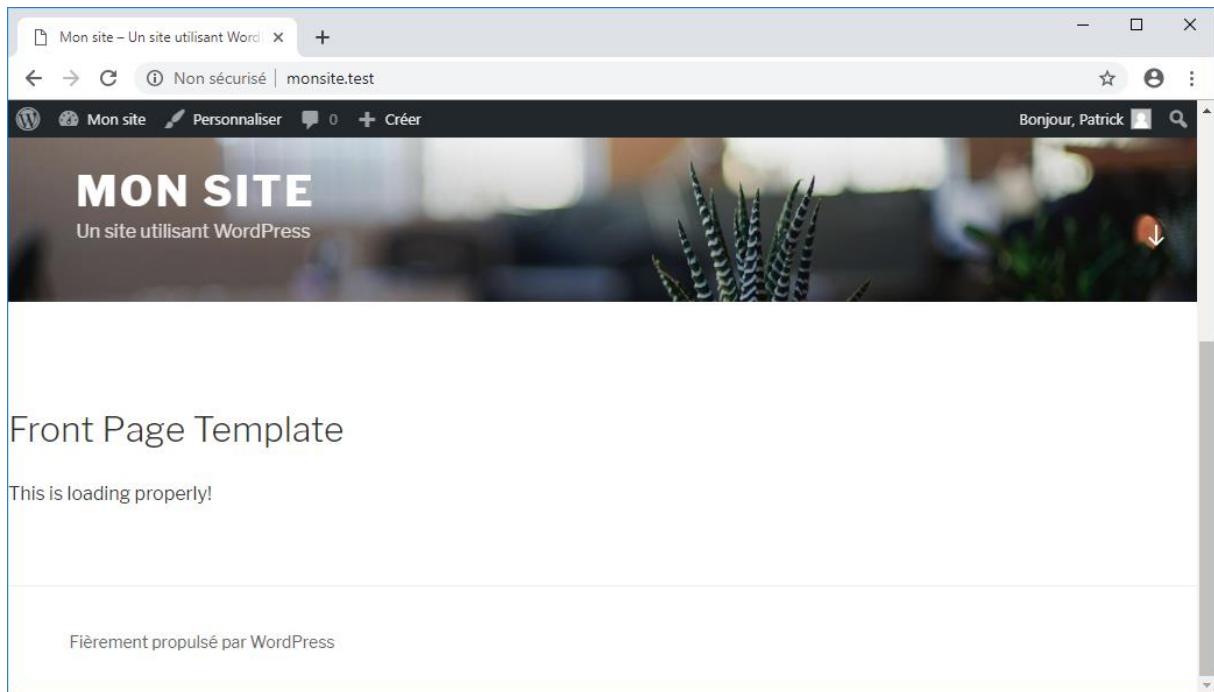

# Front Page Template



This is loading properly!

<?php  
get_footer();
```

Li 10, Col 1 Espaces : 4 UTF-8 LF PHP



The screenshot shows the WordPress Alecaddd Plugin settings page. The left sidebar has a dark theme with white icons and text. The main content area has a light gray background. At the top, there's a navigation bar with tabs: "Manage Settings" (selected), "Updates", and "About". Below that is a section titled "Settings Manager" with the sub-instruction: "Manage the Sections and Features of this Plugin by activating the checkboxes from the following list." A list of features follows, each with a blue toggle switch:

- Activate CPT Manager (on)
- Activate Taxonomy Manager (on)
- Activate Media Widget (on)
- Activate Gallery Manager (off)
- Activate Testimonial Manager (on)
- Activate Templates Manager (on)
- Activate Ajax Login/Signup (on)
- Activate Membership Manager (off)
- Activate Chat Manager (off)

At the bottom of the main content area is a blue button labeled "Enregistrer les modifications".

At the very bottom of the page, there's a footer note: "Merci de faire de [WordPress](#) votre outil de création." and "Version 5.1.1".

Eichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide AuthController.php - wordpress - Visual Studio Code

EXPLORATEUR wp-content \ plugins \ monpremier-plugin \ inc \ Base \ AuthController.php

ÉDITEURS OUVERTS WORDPRESS

- WORDPRESS
 - assetsold4
 - assetsold4b
 - assetsold5
 - inc
 - Api
 - Base
 - Activate.php
 - AuthController.php**
 - AuthControllerold.php
 - BaseController.php
 - ChatController.php
 - CustomPostTypeController.php
 - CustomTaxonomyController.php
 - Deactivate.php
 - Enqueue.php
 - GalleryController.php
 - MembershipController.php
 - SettingsLinks.php
 - TemplateController.php
 - TemplateControllerold.php
 - TestimonialController.php
 - TestimonialControllerold.php
 - WidgetController.php
 - Pages
 - Init.php
 - incnew
 - incold
 - incold2
 - incold3
 - node_modules
 - page-templates

STRUCTURE

```

1 <?php
2 /**
3 * @package AlecadddPlugin
4 */
5 namespace Inc\Base;
6
7 use Inc\Base\BaseController;
8
9 /**
10 *
11 */
12 class AuthController extends BaseController
13 {
14     public function register()
15     {
16         if ( ! $this->activated( 'login_manager' ) ) return;
17
18         add_action( 'wp_enqueue_scripts', array( $this, 'enqueue' ) );
19         add_action( 'wp_head', array( $this, 'add_auth_template' ) );
20     }
21
22     public function enqueue()
23     {
24         wp_enqueue_style( 'authstyle', $this->plugin_url . 'assets/auth.css' );
25         wp_enqueue_script( 'authscript', $this->plugin_url . 'assets/auth.js' );
26     }
27
28     public function add_auth_template()
29     {
30         if ( is_user_logged_in() ) return;
31
32         $file = $this->plugin_path . 'templates/auth.php';
33
34         if ( file_exists( $file ) ) {
35             load_template( $file, true );
36         }
37     }
38 }
```

Li 6, Col 1 Taille des tabulations : 4 UTF-8 LF PHP

Eichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide AuthController.php - wordpress - Visual Studio Code

EXPLORATEUR wp-content \ plugins \ monpremier-plugin \ inc \ Base \ AuthController.php

ÉDITEURS OUVERTS WORDPRESS

- WORDPRESS
 - front-page.php
 - two-columns-tpl.php
 - src
 - js
 - auth.js**
 - Form.js
 - myscript.js

STRUCTURE

```

1 <?php
2 /**
3 * @package AlecadddPlugin
4 */
5 namespace Inc\Base;
6
7 use Inc\Base\BaseController;
8
9 /**
10 */
11
```

Li 6, Col 1 Taille des tabulations : 4 UTF-8 LF PHP

Eichier Édition Sélection Affichage Atteindre Déboguer Terminal Aide auth.js - wordpress - Visual Studio Code

EXPLORATEUR premier-plugin\...

ÉDITEURS OUVERTS WORDPRESS

- WORDPRESS
 - myscript.js
 - slider.js
 - src
 - scss
 - modules
 - auth.scss**

STRUCTURE

```

1
```

Li 1, Col 1 Espaces : 4 UTF-8 CRLF JavaScript

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure with files like TestimonialController.php, AuthController.php (the active tab), slider.js, auth.js, and TemplateController.php.
- Code Editor:** Displays the PHP code for the AuthController.php class. The code handles user authentication by registering scripts and enqueueing styles.
- Status Bar:** Shows the current position (Li 21, Col 1), tabulation settings (Taille des tabulations : 4), encoding (UTF-8), and file type (PHP).

```
<?php
/**
 * @package AlecadddPlugin
 */
namespace Inc\Base;

use Inc\Base\BaseController;

/**
 *
 */
class AuthController extends BaseController
{
    public function register()
    {
        if ( ! $this->activated( 'login_manager' ) ) return;

        add_action( 'wp_enqueue_scripts', array( $this, 'enqueue' ) );
        add_action( 'wp_head', array( $this, 'add_auth_template' ) );
    }

    public function enqueue()
    {
        wp_enqueue_style( 'authstyle', $this->plugin_url . 'assets/auth.css' );
        wp_enqueue_script( 'authscript', $this->plugin_url . 'assets/auth.js' );
    }

    public function add_auth_template()
    {
        if ( is_user_logged_in() ) return;

        $file = $this->plugin_path . 'templates/auth.php';

        if ( file_exists( $file ) ) {
            load_template( $file, true );
        }
    }
}
```

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under the 'WORDPRESS' folder, including files like front-page.php, two-columns-tpl.php, auth.js (the active tab), form.js, myscriptr.js, slider.js, auth.scss, and form.scss.
- Code Editor:** Displays the JavaScript code for auth.js, which handles the visibility of an authentication form using DOM events.
- Status Bar:** Shows the current position (Li 15, Col 4), tabulation settings (Espaces : 4), encoding (UTF-8), and file type (JavaScript).

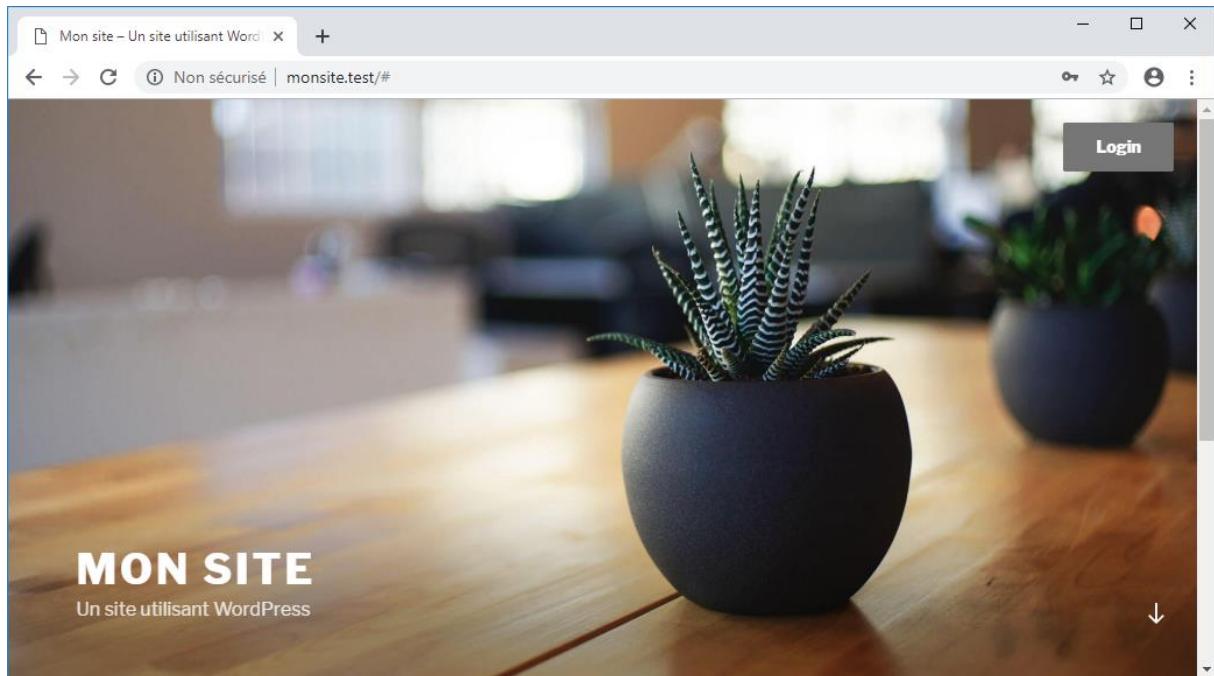
```
document.addEventListener('DOMContentLoaded', function(e) {
    const showAuthBtn = document.getElementById('alecaddd-show-auth-form'),
          authContainer = document.getElementById('alecaddd-auth-container'),
          close = document.getElementById('alecaddd-auth-close');

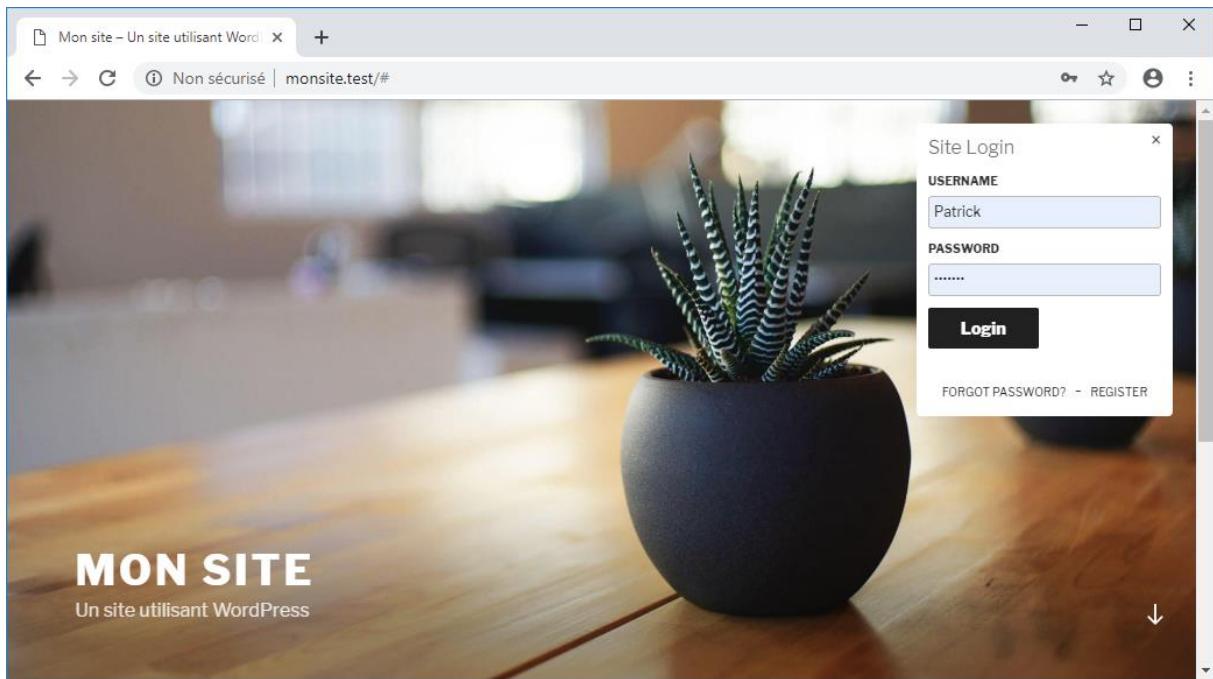
    showAuthBtn.addEventListener('click', () => {
        authContainer.classList.add('show');
        showAuthBtn.parentElement.classList.add('hide');
    });

    close.addEventListener('click', () => {
        authContainer.classList.remove('show');
        showAuthBtn.parentElement.classList.remove('hide');
    });
});
```

The screenshot shows the Visual Studio Code interface with the authController.php file open. The code is a PHP script for a login form. It includes HTML for a form with fields for username and password, and buttons for login, forgot password, and register. It also includes hidden fields for action and nonce.

```
wp-content > plugins > monpremier-plugin > templates > auth.php
1  <form id="alecaddd-auth-form" action="#" method="post" data-url=<?php echo admin_url('admin-ajax.php'); ?>
2  |   <div class="auth-btn">
3  |   |   <input class="submit_button" type="button" value="Login" id="alecaddd-show-auth-form">
4  |   |</div>
5  |   <div id="alecaddd-auth-container" class="auth-container">
6  |   |   <a id="alecaddd-auth-close" class="close" href="#">&times;</a>
7  |   |   <h2>Site Login</h2>
8  |   |   <label for="username">Username</label>
9  |   |   <input id="username" type="text" name="username">
10 |   |   <label for="password">Password</label>
11 |   |   <input id="password" type="password" name="password">
12 |   |   <input class="submit_button" type="submit" value="Login" name="submit">
13 |   |   <p class="status" data-message="status"></p>
14 |
15 |   |   <p class="actions">
16 |   |   |   <a href=<?php echo wp_lostpassword_url(); ?>">Forgot Password?</a> - <a href=<?php echo
17 |   |   |   wp_registration_url(); ?>">Register</a>
18 |   |   </p>
19 |
20 |   |   <input type="hidden" name="action" value="alecaddd_login">
21 |   |   <?php wp_nonce_field( 'ajax-login-nonce', 'alecaddd_auth' ); ?>
22 |   |</div>
23 |</form>
```





MON SITE

Un site utilisant WordPress