

CREACIÓ D'UN PLUGIN PER WORDPRESS UTILITZANT `add_action`

Part 1: Creació inicial del plugin

1- Crea un dipòsit públic de **Github** a on desar el codi del teu plugin de nom **pluginwp2_xxyy** a on **xxx** representa les 3 primeres lletres del teu nom i **yyy** les 3 primeres lletres del teu primer cognom.

2- Crea un directori de nom **peupagina** dins del directori **/var/www/html/wordpress/wp-content/plugins**. Fes que el propietari el grup del directori sigui **www-data**. Dins del directori crea un fitxer de nom **peupagina.php**. Fes que el propietari el grup del directori sigui **www-data**. Fes que **peupagina.php** sigui executable.

3- El contingut inicial de **peupagina.php** serà:

```
<?php
/*
Plugin Name: Peu de pàgina per defecte per a noves entrades
Plugin URI: Escriu l'adreça URL completa del teu plugin dins de Github
Description: Crea peus de pàgina per defecte per a noves entrades
Version: 0.1
Author: El teu nom i cognom
Author URI: L'adreça URL del teu compte de Github
License: GNU GPLv3
*/
?>
```

4- Activa el plugin a WordPress de la mateixa manera que es va veure a l'apartat 6 de la primera pràctica.

5- Comprova que s'ha activat. Puja el fitxer a **Github**. Comentari del commit: **Commit 1**

Part 2: Afegint una `add_action` per crear una entrada dins del menú lateral d'administració de WP. Utilitzant `admin_menu`.

Una **acció** obliga a **WordPress** permet executar codi propi quan es produeix un esdeveniment o s'arriba a un punt en l'execució de **WordPress**. Per exemple, podem obligar a executar un codi específic propi quan s'obre la finestra d'administració, o quan ens arriba un comentari sobre un post o quan obrim un post per veure el seu contingut, o quan esborrem un post.

La manera més senzilla de crear un acció és utilitzant la funció **`add_action()`** que podem utilitzar un cop instal·lat **WordPress**. Amb **`add_action`** podem indicar que quina funció de PHP volem executar quan es produeixi un esdeveniment en concret o arribem a un punt de l'execució.

La llista d'accions que es poden controlar amb `add_action` es troben **aquí**. Treballarem amb una que és molt senzilla d'entendre anomenada **`admin_menu`**, que permet realitzar una acció que afectarà al menú lateral cada cop que s'arriba a la pàgina d'administració de **WordPress**. Aquesta acció consistirà en crear una entrada en el menú i executar una funció pròpia creada pel desenvolupador i que afectarà a com es comporta la nova entrada.

Normalment, la funció que s'executarà, permet incloure una pàgina associada a l'entrada del menú. Per poder crear una pàgina ens caldrà treballar amb l'ordre **`add_menu_page`**. Per poder utilitzar **`add_menu_page`** ens caldrà saber el tipus de **rols i capacitats** que s'utilitzen per fer que la nova entrada aparegui o no aparegui en funció dels permisos, drets i autoritzacions de l'usuari que està validat en **WordPress**. El nostre usuari de WordPress és per defecte administrador i per tant el seu rol és **administrator** (en anglès).

Així doncs, un codi senzill per crear una entrada de menú a la barra lateral quan s'accedeix a la web d'administració seria:

```
<?php
/*
Plugin Name: Peu de pàgina per defecte per noves entrades
Plugin URI: Escriu l'adreça URL completa del teu plugin dins de Github
Description: Crea peus de pàgina per defecte per a noves entrades
Version: 0.1
Author: El teu nom i cognom
Author URI: L'adreça URL del teu compte de Github
License: GNU GPLv3
*/
function admin_menu_peu_pagina(){
    $titolPagina="Peu de pàgina per defecte";
    $titolEntrada="Peu de pàgina per defecte";
    $role="administrator";
    $id="peu_pagina_per_defecte";
    add_menu_page($titolPagina,$titolEntrada,$role,$id); ;
}
add_action('admin_menu','admin_menu_peu_pagina');
?>
```

A on **admin_menu_peu_pagina** és la funció que s'executa quan es produeix l'acció d'arribar al menú administració. Aquesta funció permet incloure una pàgina per l'entrada sempre i quan qui entra tingui assignar el role d'administrador. El valor **\$id** ha de ser únic.

Comprova que el codi funciona, que es crea l'entrada, comprova el títol de la pàgina, i títol de l'entrada i que es crea una pàgina buida associada a l'entrada. Si tot funciona correctament, puja el fitxer al teu compte de **Github**. Comentari del commit: **Commit 2**

Part 3: Afegint codi a la pàgina associada a l'entrada de menú.

Anem a crear una plugin que permeti definir un **peu de pàgina** per defecte a totes les noves entrades de **WordPress**. Aprofitarem la instrucció **add_menu_page** per cridar a una funció que crei un formulari dins de la nova pàgina inclòsa per demanar el contingut del peu de pàgina. Modifica la funció **admin_menu_peu_pagina** i fes que tingui el següent codi:

```
function admin_menu_peu_pagina(){
    $titolPagina="Peu de pàgina per defecte";
    $titolEntrada="Peu de pàgina per defecte";
    $role="administrator";
    $id="peu_pagina_per_defecte";
    $nom_funcio="nouPeuPagDef";
    add_menu_page($titolPagina,$titolEntrada,$role,$id, $nom_funcio);
}
```

A continuació crea la funció **nouPeuPagDef** amb el següent codi:

```
function nouPeuPagDef(){
    include_once('formPeuPag.php');
}
```

Dins del mateix directori a on has creat **peupagina.php** crea el fitxer **formPeuPag.php** amb el següent codi:

```
</html>
  <head>
    <meta content="text/html; charset=UTF-8" http-equiv="content-type">
    <title>EXEMPLES DE FORMULARIS</title>
  </head>
  <body>
    <h1>Creació d'un peu de pàgina per defecte</h1>
    <form action="" method="POST">
      Text del peu de pàgina: <input type="text" name="peupag"><br>
<input type="submit" value="OK"/>
    </form>
  </body>
</html>
<?php
  if (isset($_POST['peupag'])) {
    update_option('footer_text', trim($_POST['peupag']));
    echo "Peu de pàgina actual: ".get_option('footer_text')."<br>";
  }
?>
```

Dins d'aquest codi, és molt important la funció [update_option](#) perquè permet enregistrar el text introduït per l'usuari en el formulari com el nou peu de pàgina actualitzant utilitzant l'opció **'footer_text'**.

Comprova que tot funciona correctament i puja la nova versió a **Github**. Comentari del commit: **Commit 3**.

Pel moment només hem creat la pàgina per recollir el nou peu de pàgina, però encara no el podem utilitzar.

Part 4: Utilitzant l'acció wp_footer

Finalment, volem que cada cop que es creï una nova entrada i accedim a l'entrada, es generi una acció que executi un codi que generi el peu de pàgina que l'usuari ha creat a l'apartat anterior. L'acció [wp_footer](#) permet executar una funció creada pel desenvolupador per modificar el peu de pàgina d'una entrada quan s'hi accedeix.

Modifica el fitxer `peupagina.php` i afegeix una nova `add_action` darrera de la que ja existia amb el següent codi:

```
add_action('wp_footer', 'afegPeuPagDef');
```

I ara crea una nova funció de nom **afegPeuPagDef** dins de **peupagina.php** amb el següent codi:

```
function afegPeuPagDef() {
  echo "<div style='color: red;
  font-size: 10px;
  margin: 10px;'>".get_option('footer_text')."</div>";
}
```

A on la funció [get_option](#) accedeix al valor de l'opció **'footer_text'**.

Encara no pujis el nou codi a **Github**.

Part 5: Comprovant que tot funciona correctament. Pujant el codi final a Github

1- Accedeix a la web d'administració de WordPress.

2- Accedeix a la nova entrada de menú lateral "Peu de pàgina per defecte".

3- Modifica el valor del peu de pàgina per defecte. Escriu " Copyright by xxxxx" a on xxxx representa el teu nom i primer cognom.

4- Crea una nova entrada de nom **Prova de peu de pàgina**. Dins del contingut escriu "**Prova de peu de pàgina**". Publica la nova entrada.

5- Accedeix a la nova entrada i comprova que es crea el peu de pàgina per defecte.

6- Si tot funciona correctament, puja el codi a **Github**. Comentari del commit: **Commit 3**.

Lliurament de la pràctica

1- El darrer dia de comprovacions és el dia **23/25-4-18**

2- Comprovació del codi pujat a **Github**.

3- Comprovació del plugin funcionant correctament

4- Preguntes de comprovació de comprensió de la pràctica.