Recientemente hemos realizado un desarrollo a medida para Wordpress que requería de la creación de un plugin para poder hacer un pago Online, pero sin la necesidad de tener una catálogo de productos ni de un carrito de la compra.

Una vez terminada la creación del módulo, tuvimos la necesidad de recibir las comunicaciones del TPV online y cambiar, así, el estado del pedido realizado.

Lo podríamos haber hecho con un fichero totalmente independiente de Wordpress, pero entonces no podemos aprovechar todas las ventajas que nos ofrece su framework. Pues os mostramos como crear un fichero totalmente independiente 'standalone' de Wordpress y poder seguir utilizando su framework, con todas sus funciones y variables.

Código PHP independiente de Wordpress

Lo que tenéis que hacer es crear un fichero PHP en cualquier parte de la estructura de ficheros de Wordpress. En nuestro caso, como el fichero independiente formaba parte de la funcionalidad de un plugin, lo hemos creado dentro de la carpeta del mismo plugin.

Por tanto, su ubicación tiene la siguiente estructura:

```
/RAIZ_DEL_PROYECTO/wp-content/plugins/NOMBRE_PLUGIN/STANDALONE.php
```

Y para llamarlo sin pasar por Wordpress utilizaremos la siguiente ruta:

```
http://VUESTRO_DOMINIO//wp-content/plugins/NOMBRE_PLUGIN/STANDALONE.php
```

Perfecto, ya estamos llamando un fichero sin pasar por Wordpress, pero tampoco podemos hacer uso de todas las funciones y variables de Wordpress, así que ahora tenemos que iniciar el framework de Wordpress: *wp-load.php*.

Para hacerlo ponemos la siguiente línea al principio del todo del fichero:

```
<?php       require_once('wp-load.php');  ?>
```

Tened en cuenta la ubicación de vuestro fichero para que el *require_once* no falle. Así pues, añadid tantos *../* como sean necesarios para llegar a la raíz del Wordpress. En nuestro caso, tenemos que descender desde la carpeta de plugins:

```
<?php require_once('../../../wp-load.php'); ?>
```

A partir de este punto ya podemos empezar a usar todas las funcionalidades que nos aporta el framework de Wordpress, como el acceso a la base de datos y las cookies.

Por ejemplo, si queremos saber si un usuario tiene permisos de administrador para poder ver nuestro fichero, podemos utilizar las funciones normales de Wordpress para comprobarlo:

```
<?php require_once('wp-load.php'); if ( !is_user_logged_in() && !current_user_can( 'manage_options' ) ) die( 'Si no eres administrador no puedes ver esta página.' ); ?>
```

A partir de este punto, ya podéis implementar vuestro desarrollo a medida para que haga las funcionalidades necesarias, como modificar un registro en base de datos o cualquier otra cosa.

Si te ha gustado y te ha parecido útil no dudes en compartirlo: