

WordPress Temas hijo



WordPress. Temas hijo

Jesús Amieiro Becerra

Copyright © 2014 Jesús Amieiro Becerra. Algunos derechos reservados.

Este documento se publica bajo licencia [Reconocimiento-NoComercial-CompartirIgual 3.0 España](http://creativecommons.org/licenses/by-nc-sa/3.0/es/).



Versión actual: 23/05/2014

Obtén la última versión disponible en <http://fontelearn.com/es/wordpress-temas-hijo/>

Índice

1	Presentación.....	6
1.1	Contenido.....	6
1.2	Contacto.....	6
2	Introducción.....	7
2.1	¿Qué es un tema en WordPress?.....	7
2.2	Composición de un tema.....	7
2.3	¿Qué es un tema padre?.....	7
2.4	¿Qué es un tema hijo?.....	7
3	¿Por qué trabajar con temas hijo?.....	9
3.1	Ventajas.....	9
3.2	Desventajas.....	9
4	¿Cuándo debemos usar un tema hijo?.....	10
5	Elección del tema padre.....	11
5.1	Frameworks.....	11
5.2	Temas.....	11
6	Creando un tema hijo.....	12
6.1	Requisitos del tema hijo.....	12
6.2	Elección del tema padre.....	12
6.3	La hoja de estilos.....	12
6.4	Archivos del tema.....	14
6.5	Ubicaciones en el tema.....	14
6.6	functions.php.....	15
6.7	Hooks.....	15
6.8	Internacionalización.....	16
7	Control de versiones.....	17
8	Ejercicio práctico.....	18
8.1	Requisitos del tema hijo.....	18
9	Entorno de desarrollo.....	19
9.1	Navegadores.....	19
9.2	Plugins Chrome.....	19
9.3	Plugins Firefox.....	19
9.4	Compatibilidad con navegadores.....	19
9.5	Editor/IDE.....	19
9.6	Servidor.....	20
9.6.1	LAMP Debian.....	20
9.6.2	LAMP Fedora.....	20
9.6.3	OSX.....	20
9.6.4	Windows.....	20
9.7	Descarga e instalación de WordPress.....	20
9.8	Instalación de los datos de prueba.....	20
10	Desarrollando un tema hijo.....	21
10.1	Clonado del tema.....	21
10.2	Análisis del proyecto.....	22
10.2.1	Commit 1.....	22
10.2.2	Paso 2.....	23
10.2.3	Paso 3.....	24

10.2.4 Paso 4.....	25
10.2.5 Paso 5.....	25
10.2.6 Paso 6.....	26
10.2.7 Paso 7.....	26
10.2.8 Paso 8.....	29
10.2.9 Paso 9.....	31
10.2.10 Paso 10.....	36
10.2.11 Paso 11.....	37
10.2.12 Paso 12.....	37
10.2.13 Paso 13.....	38
10.2.14 Paso 14.....	38
10.2.15 Paso 15.....	39
10.2.16 Paso 16.....	39
10.2.17 Paso 17.....	39
10.2.18 Paso 18.....	40
10.2.19 Paso 19.....	41
10.2.20 Paso 20.....	41
10.2.21 Paso 21.....	41
10.2.22 Paso 22.....	43
10.2.23 Paso 23.....	43
10.2.24 Paso 24.....	44
10.2.25 Paso 25.....	45
10.2.26 Paso 26.....	45
10.2.27 Paso 27.....	46
10.2.28 Paso 28.....	46
10.2.29 Paso 29.....	47
10.2.30 Paso 30.....	48
10.2.31 Paso 31.....	48

1 Presentación

Este es uno de los libros de una serie sobre WordPress que podrás encontrar en fontelearn.com. WordPress es un sistema de gestión de contenidos (CMS) que goza de una gran popularidad, acaparando casi el 60% de la cuota de mercado de los CMS y un 22,1% de todos los sitios web a nivel mundial.

1.1 Contenido

Este libro está en versión **alfa**. Esto quiere decir que le faltan contenidos y, aunque hemos tratado de evitar cualquier tipo de error, puede que encuentres alguno. En ese caso te agradeceríamos que lo notificaras a través de cualquiera de los métodos de contacto indicados en el capítulo *1.2 Contacto*.

¿Qué vas a encontrar en este libro?

En el capítulo *2 Introducción* vamos a ver qué es un tema en WordPress, cómo se compone un tema, qué es un tema padre y qué es un tema hijo.

En el capítulo *3 ¿Por qué trabajar con temas hijo?* vamos a ver las ventajas y desventajas de trabajar con un tema hijo respecto a un tema padre.

En el capítulo *4 ¿Cuándo debemos usar un tema hijo?* explicaremos en qué casos es adecuado usar un tema hijo y en cuáles debemos decantarnos por un tema normal.

En el capítulo *5 Elección del tema padre* veremos las distintas alternativas que tenemos para elegir un tema padre y los motivos de elección de uno u otro.

En el capítulo *6 Creando un tema hijo* vamos a entrar en materia y ver las características técnicas particulares de la creación de un tema hijo para WordPress.

En el capítulo *7 Control de versiones* veremos unas nociones básicas de por qué es interesante desarrollar un tema usando una herramienta de control de versiones como es Git.

A partir del capítulo *8 Ejercicio práctico* entremos de lleno en materia práctica. En este capítulo detallamos el trabajo que nos tendrá ocupados en los siguientes capítulos, como es el desarrollo de un tema hijo real.

En el capítulo *9 Entorno de desarrollo* veremos el entorno software necesario para la realización de un tema.

En el capítulo *10 Desarrollando un tema hijo* vamos a ver un caso real de un desarrollo de un tema hijo que se encuentra en producción en varios blogs, entre ellos el conocido blog de tecnología [Error 500](#).

1.2 Contacto

Puedes ponerte en contacto con nosotros a través de cualquiera de los métodos de contacto indicados en la dirección web <http://fontelearn.com/es/contacto/>

2 Introducción

Antes de entrar en materia y ver los elementos que caracterizan a un tema hijo en WordPress vamos a ver qué es un tema en WordPress y cómo funciona el sistema de temas de WordPress.

2.1 ¿Qué es un tema en WordPress?

WordPress nos permite cambiar la apariencia de nuestro sitio web mediante temas; es decir, cambiando el tema activo cambiamos la apariencia de nuestro sitio web. Pero es algo más que un cambio de apariencia, ya que si solo estuviésemos cambiando la apariencia esto implicaría que solo cambiaríamos el diseño visual.

Los temas de WordPress van más allá y nos permiten proveer mucho más control sobre el aspecto y la presentación del material en el sitio web.

WordPress está diseñado de tal forma que tenemos separado:

- El **núcleo** de WordPress, donde está la funcionalidad básica, común a todas las instalaciones.
- Los **plugins**, que nos permiten extender la funcionalidad de WordPress.
- El **contenido**, como son las páginas y entradas, que se almacena en la base de datos y en el sistema de ficheros del servidor.
- La forma en que el sitio es presentado, mediante el propio **tema**.

2.2 Composición de un tema

Un tema no es más que un conjunto de archivos que trabajan conjuntamente produciendo una **interfaz gráfica con un diseño unificado**. Consta de:

- Archivos PHP.
- Archivos CSS.
- Archivos gráficos: jpg, png,...
- Archivos JavaScript.
- Fuentes.
- Texto: licencias, archivos readme,...
- ...

2.3 ¿Qué es un tema padre?

Un tema padre es un tema completo, que incluye todos los elementos que una plantilla de WordPress necesita para poder funcionar. Todos los temas, excepto los temas hijo, son temas padre.

2.4 ¿Qué es un tema hijo?

Los temas hijos son la forma en la que WordPress nos permite extender un tema, realizar modificaciones y añadir funcionalidades que el tema padre no tiene.

Un tema hijo hereda toda la funcionalidad y apariencia del tema padre, permitiéndonos realizar modificaciones sin tener que tocar el tema principal; es decir, mantenemos toda la parametrización separada de los ficheros del tema padre original, de tal forma que cuando se producen actualizaciones del tema padre estas no sobrescriben al tema hijo.

Un tema hijo es casi igual a un tema normal en WordPress:

- Se ubica en el directorio de temas con los otros temas.
- Puede tener los mismos archivos que un tema normal: PHP, CSS, gráficos, JS,...

La principal diferencia y característica de los temas hijo viene con los archivos de la plantilla, ya que el tema hijo usará todos los archivos del tema padre excepto que disponga de un archivo más específico en el tema hijo. Veremos cómo funciona y las excepciones en determinados archivos.

No es posible crear temas nieto: temas hijo de temas hijo.

3 ¿Por qué trabajar con temas hijo?

A continuación vamos a ver cuáles son las ventajas y desventajas de usar un tema hijo.

3.1 Ventajas

Las principales ventajas de trabajar con un tema hijo son:

- Poder actualizar el tema padre sin perder los cambios, al tener separados en dos directorios el tema padre y el tema hijo, por lo que podemos actualizar el tema padre sin perder nada en el tema hijo.
- Incrementar la velocidad de desarrollo del tema, ya que todo el trabajo "sucio" está hecho: archivo de error 404, plantilla de búsqueda, pruebas, ...

3.2 Desventajas

Pero también tenemos que pagar un pequeño peaje al trabajar con temas hijos, fundamentalmente:

- Una pequeña disminución del rendimiento, ya que:
 - WordPress tiene que buscar las plantillas en 2 sitios, la del tema hijo y la del padre.
 - Tenemos que incluir una directiva `@import` en la CSS, con lo que tenemos que hacer una llamada más al servidor web desde el navegador.
- Para obtener el mayor partido posible al tema padre tenemos que conocer todas sus funcionalidades, lo que implica una curva de aprendizaje, variable en función del tema elegido.
- Dependencia de un proveedor externo, que en cualquier momento puede cesar en el soporte y/o en las actualizaciones del tema padre. Para tratar de minimizar estos problemas lo más adecuado es utilizar temas populares o que tengan detrás una empresa con una trayectoria contrastada en la comunidad. Y, en último caso, como la mayoría de temas son software libre, siempre podemos actualizarlos nosotros o apoyarnos en la gran comunidad de desarrolladores de software libre.

4 ¿Cuándo debemos usar un tema hijo?

El uso de temas hijos en WordPress no es una panacea para la creación de un tema WordPress. Hay casos en los que es más adecuado un tema hijo y otros en los que será oportuno utilizar un tema normal.

Lo primero que tenemos que tener claro es cuándo debemos usar un tema hijo, ya que no en todos los casos un tema hijo es la mejor opción. Es una opción adecuada usar un tema hijo cuando:

- Vamos a realizar cambios menores sobre el tema de partida.
- Cuando somos principiantes y no estamos capacitados técnicamente como para realizar el desarrollo del tema desde cero.

No es una buena elección cuando los cambios son tan grandes que tiene más sentido desarrollar un tema desde cero o elegir otro tema.

5 Elección del tema padre

Una vez que tenemos claro que el desarrollo de un tema hijo es la opción adecuada, el siguiente punto es la elección del tema base en función de la funcionalidad buscada en el tema que queremos desarrollar, ya que, por ejemplo, podemos tener un tema que sea adecuado como tema padre para un blog pero no para una web en la que en la portada tengamos que tener un slider responsive a ancho completo o 3 widgets en el pie de página.

Y dentro de la búsqueda del tema padre nos vamos a encontrar con dos casuísticas diferenciadas: frameworks o temas. La línea que separa a los frameworks de los temas es muy fina. En los siguientes puntos trataremos de ver las diferencias.

5.1 Frameworks

Los frameworks aportan mucha funcionalidad, de tal forma que dispondremos de muchas opciones pero de poco estilo. Como ejemplos de frameworks tenemos

- [Pagelines](#)
- [Hybrid](#)
- [Genesis](#)
- [Thematic](#)

5.2 Temas

Por el contrario, los temas tienen una funcionalidad más limitada pero bastante diseño. Temas como

- [TwentyFourteen](#)
- [TwentyThirteen](#)
- [Quark](#)

son temas interesantes como punto de partida para realizar temas hijos.

6 Creando un tema hijo

Qué necesitamos para crear un tema. Pues en un tema, como mínimo, 2 archivos:

- `style.css`
- `index.php`

Sin embargo, para un tema hijo, lo único que se requiere es un archivo, el `style.css`. El resto de archivos los obtiene del tema padre, excepto aquellas partes que queremos modificar, que tendremos que copiar el archivo desde el tema padre al hijo e introducir las modificaciones oportunas.

6.1 Requisitos del tema hijo

Lo primero que tenemos que tener claro es cuáles son los requisitos del tema que queremos desarrollar, para ver si tiene sentido desarrollarlo como tema hijo de un tema padre conocido por nosotros o si tiene más sentido desarrollarlo como un tema normal.

Entre los requisitos del tema se pueden encontrar:

- Tema adaptativo.
- Cabecera con logotipo: centrado, alineado a la izquierda.
- Menú principal alineado a la derecha o justo debajo del encabezado en la versión de escritorio.
- Número de columnas del blog: 1, 2, 3,... alineación de la columna principal.
- Elementos a incluir en la columna secundaria: buscador, últimos tweets, nube de etiquetas, ...
- Tipografía que vamos a utilizar, tanto si se elige una combinación de fuentes habituales en el sistema (Arial, Helvetica, sans-serif) o fuentes web.
- Sistema de paginación.
- Presencia de sliders.
- ...

6.2 Elección del tema padre

Tras tener claros los requisitos del tema hijo que queremos desarrollar lo siguiente que vamos a hacer es ver si existe algún tema padre (framework o tema) que conozcamos y que se pueda adaptar a los requisitos del tema hijo que acabamos de describir.

Tras analizar las diferencias entre el tema padre y los requisitos del nuevo tema hijo comenzamos el desarrollo del tema hijo.

6.3 La hoja de estilos

Lo primero que hacemos es crear un directorio con el nombre del tema y dentro el archivo `style.css`.

Veamos como es el archivo de estilo en un tema hijo, que está ubicado en un directorio al mismo nivel que el tema padre.

Empieza siendo idéntico a un tema normal, con el nombre y la URI del tema, el nombre y la URI del autor, pero al final aparecen dos líneas específicas del tema hijo:

- `Template`, que indica cuál es el directorio donde se encuentra el tema padre.
- `Text Domain`, utilizado principalmente para la traducción del tema.

```

/*
Theme Name: Error 500
Theme URI: http://fontethemes.com/es/error-500/
Author: Fonte Themes
Author URI: http://fontethemes.com/
Description:
License:
License URI:
Tags:
Template: quark
Text Domain: error500
*/

```

Tras crear este archivo de estilo *style.css* dentro de un directorio en el directorio de temas, ya está disponible en la zona privada para ser activado.

Pero como los archivos del tema hijo sobrescriben a los correspondientes archivos del tema padre, ahora mismo el tema hijo no tendrá ningún estilo porque la hoja de estilo del tema hijo está vacía: solo tenemos la cabecera de comentarios.

Esto estaría bien si quisiéramos escribir la CSS del tema hijo desde cero, pero no es lo que buscamos con un tema hijo, tanto si usamos un tema normal, del que vamos a aprovechar muchas de sus características, como si usamos un framework, del que, por lo menos, vamos a aprovechar sus resets y demás CSS de inicio.

Habitualmente, cuando creamos un tema hijo, lo que buscamos es realizar pequeñas modificaciones del archivo padre, o, por lo menos, tener unas CSS reseteadas y listas para usar. Por lo tanto, lo que tenemos que hacer es incluir el archivo de estilo del tema padre.

Esto lo hacemos mediante la directiva *@import*, que se encarga de llamar a la hoja de estilos del tema padre:

```

/* -----
   Importo las CSS del tema padre
   ----- */

@import url("../quark/style.css");

```

Por ahora tenemos un tema con un único archivo, la CSS, que tiene una cabecera y se encarga de importar la CSS del tema padre, por lo que ahora tenemos la misma apariencia que en el tema padre. Si queremos hacer cambios es aquí donde tenemos que hacerlos. Por ejemplo, si quiero usar una fuente web:

```
@import url(http://fonts.googleapis.com/css?
```

```
family=Source+Sans+Pro);
```

O si quiero cambiar el color de los enlaces:

```
a, a:visited {  
    color: #2790ae;  
    text-decoration: underline;  
}
```

Son CSS normales, que **sobrescriben a las CSS del tema padre**.

6.4 Archivos del tema

Cualquier archivo que añadamos al tema hijo sobrescribirá a su equivalente en el tema padre; es decir, WordPress ejecutará la versión existente en el tema hijo y no en el tema padre. Además podemos añadir otros archivos en el tema hijo que no aparecen en el tema padre: nuevas plantillas, fuentes, CSS, imágenes, ...

Para modificar un archivo desde el tema padre lo que haremos será copiar el archivo original del tema padre en el directorio del tema hijo y haremos las modificaciones que necesitemos; por ejemplo, si queremos introducir un copyright en el pie de la página, copiaremos el fichero *footer.php* desde el directorio del tema padre al directorio del tema hijo y ahí realizaremos los cambios.

Este procedimiento nos puede traer algún problema cuando se realizan cambios en el tema padre, ya que el tema hijo no recibe los cambios. Es aquí donde más aprovecharemos un conocimiento profundo del tema padre para poder llevar estos cambios al tema hijo si fuese necesario. Y usar una herramienta de control de versiones nos ayudará mucho en esta labor.

6.5 Ubicaciones en el tema

Cuando trabajamos con temas, para obtener la ruta absoluta al directorio de la plantilla usamos

```
get\_template\_directory\(\)
```

y la URI mediante la función

```
get\_template\_directory\_uri\(\)
```

Si usamos esta misma función en un tema hijo nos devuelve el directorio del tema padre, por lo que para conocer la ubicación del tema hijo usaremos la función

```
get\_stylesheet\_directory\(\)
```

que nos devuelve la ruta de la hoja de estilo del tema hijo. Si queremos obtener la URI base del tema usaremos la función

```
get\_stylesheet\_directory\_uri\(\)
```

6.6 functions.php

Cualquier archivo que tengamos en el tema hijo va a sobrescribir el mismo archivo del tema padre, excepto el archivo *functions.php* del tema hijo, que no sobrescribe al fichero *functions.php* del tema padre, ya que en este caso lo que pasaría sería que se eliminaría toda la funcionalidad creada en el tema padre dentro de este archivo. Y esta funcionalidad posiblemente es uno de los motivos de usar ese tema como padre para el tema hijo.

En vez de sobrescribir el archivo, los dos archivos son incluidos en el resultado final, permitiendo al tema hijo tener su propio archivo de funciones, de tal forma que en cualquier momento podemos añadirle funcionalidad al tema hijo.

Hay que tener en cuenta que **el archivo *functions.php* del tema hijo se incluye antes que el del tema padre**. Para evitar problemas, en muchos temas en los temas normales se comprueba que no exista previamente la función, ya que si no machacaría la anterior, por si queremos sobrescribirla en el archivo *functions.php* del tema hijo. Es lo que se denominan “*pluggable functions*”.

La siguiente comprobación se introduce en el tema padre para que se pueda declarar en el tema hijo y sustituya a esta versión.

```
if ( ! function_exists( 'my_theme_setup' ) ) :  
    function my_theme_setup() {  
        [...]  
    }  
}
```

6.7 Hooks

Mediante los *hooks* podemos añadir funcionalidad en los temas y en los temas hijos, comportándose de una forma similar.

Al estar incluidos en el archivo *functions.php* podemos tener un problema si tenemos los mismos *hooks* en el *functions.php* del tema padre y del hijo. Dado que los archivos del tema hijo se incluyen primero, cualquier *hook* que establezcamos en el tema hijo puede ser sobrescrito por los *hooks* del tema padre, causándonos un problema en el desarrollo del tema hijo.

Para evitar este problema podemos usar un *hook* de acción en el tema hijo para desenganchar cualquier *hook* problemático del archivo de funciones del tema padre.

Para ello usaremos la acción [after_setup_theme](#), que se ejecuta después de que ambos archivos de funciones hayan sido cargados para eliminar cualquier acción o filtrar los *hooks* hechos en el tema padre.

Por ejemplo, si queremos cambiar la función que compone el título en el tema padre, añadimos la función “cambio_title” mediante una acción “after_setup_theme” en el archivo *functions.php* del tema hijo, que se encarga de borrar el filtro del tema padre y añade el filtro del tema hijo.

```
function error500_wp_title( $title, $sep ) {  
    $title .= "Designed by Fonte Themes";  
    return $title;  
}
```

```

}
function cambio_title( ) {
    remove_filter( 'wp_title', 'quark_wp_title', 10, 2 );
    add_filter('wp_title', 'error500_wp_title', 10, 2 );
}
add_action( 'after_setup_theme', 'cambio_title' );

```

De la misma forma podemos eliminar cualquier otra acción que añada el tema padre mediante la función [remove_action\(\)](#).

6.8 Internacionalización

El tema hijo usa los archivos .mo de internacionalización que tiene el tema padre.

Si en algún momento necesitamos más cadenas, lo primero que tendremos que hacer es crear las cadenas en los archivos PHP, usando:

```

__( 'Text to translate', 'error500' )
_e( 'Text to translate', 'error500' )

```

El segundo parámetro es el dominio de texto (Text domain) que insertamos en la hoja de estilo.

A continuación, con estas cadenas, creo los ficheros .mo para el idioma o idiomas que necesitamos. Recomendamos partir del inglés y usar para castellano el archivo de traducción.

Estos archivos los colocamos el directorio *languages* dentro de nuestro tema hijo. Por ejemplo

```
languages/es_ES.po
```

A continuación, en el archivo functions.php del tema hijo tenemos que incluir el código para que cargue las cadenas de traducción del tema hijo, mediante el hook `after_setup_theme`, que se llama durante la carga de cada página, después de la inicialización del tema.

```

function error500_setup() {
    load_child_theme_textdomain( 'error500',
    get_stylesheet_directory() . '/languages' );
}
add_action( 'after_setup_theme', 'error500_setup' );

```

7 Control de versiones

Para controlar todos los pasos que vamos realizando os recomendamos el uso de un sistema de gestión de versiones, en particular de Git, sobre todo para poder ver las diferencias entre la plantilla de partida y alguna posible actualización en el tema padre o para poder detectar en qué momento hemos introducido algún error, quién lo ha introducido,...

En fontelearn.com puedes descargar un curso de [Fundamentos de Git](#) para introducirte a esta magnífica herramienta de control de versiones. Te recomendamos empezar a usarla cuanto antes.

8 Ejercicio práctico

A continuación vamos a ver cómo crear un tema hijo a partir del tema [Quark](#), que puedes descargar desde [WordPress.org](#) o desde la cuenta de [GitHub](#) del autor.

El tema que vamos a desarrollar se llama “*error500*”. Podemos verlo en funcionamiento y descargarlo en la dirección web <http://error500.fontethemes.com/>.

8.1 Requisitos del tema hijo

Lo primero que tenemos que tener claro es cuáles son los requisitos del tema que queremos desarrollar, para ver si tiene sentido desarrollarlo como tema hijo de un tema padre conocido o si tiene más sentido desarrollarlo como un tema normal.

En este caso, los requisitos del tema son:

- Tema adaptativo, con versiones distintas en PC, tableta y móvil.
- Cabecera con:
 - Título y subtítulo en la zona superior izquierda.
 - Menú principal alineado a la derecha en la versión de escritorio.
- Blog a 2 columnas en el escritorio.
 - En la columna izquierda el contenido principal.
 - En la columna de la derecha elementos accesorios (widgets), tales como el buscador, los últimos tweets, una nube de etiquetas,...
- Fuentes web. En este caso la elegida ha sido la “Source Sans Pro”, que es una tipografía libre de tipo sans serif creada por Paul D. Hunt para Adobe Systems. Se encuentra disponible en [Google Fonts](#) para su enlace en las CSS o en [GitHub](#) para su descarga.
- Paginación.

9 Entorno de desarrollo

9.1 Navegadores

- [Google Chrome](#)
- [Apple Safari](#)
- [Mozilla Firefox](#)
- [Opera](#)
- [Internet Explorer](#)

9.2 Plugins Chrome

- [MeasureIt](#)
- [Web Developer](#)
- [ColorZilla](#)
- [Window Resizer](#)
- Botón derecho → Inspeccionar elemento.

9.3 Plugins Firefox

- [MeasureIt](#)
- [Web Developer](#)
- [ColorZilla](#)
- [Window Resizer](#)

9.4 Compatibilidad con navegadores

- [BrowserShot](#)
- IE: [modern.IE](#)

9.5 Editor/IDE

- [Vim](#)
- [Emacs](#)
- [Sublime Text](#)
- [Coda](#) (OS X)
- [Textmate](#) (OS X)
- [PhpStorm](#)
- [NetBeans](#) (HTML5 & PHP bundle)

9.6 Servidor

9.6.1 LAMP Debian

En el siguiente enlace se indica [cómo instalar LAMP en Debian](#).

9.6.2 LAMP Fedora

En el siguiente enlace se indica [cómo instalar LAMP en Fedora](#).

9.6.3 OSX

[MAMP](#) instala un servidor local en un Mac OS X (MacBook o iMac). Incluye Apache, MySQL, PHP, Python, Perl, PhpMyAdmin, SQLite Manager y phpLiteAdmin.

9.6.4 Windows

[WampServer](#) instala un entorno de desarrollo web en Windows. Incluye Apache2, PHP, MySQL, PhpMyAdmin, SqlBuddy y Xdebug.

9.7 Descarga e instalación de WordPress

```
wget http://wordpress.org/latest.zip
```

Realizamos la famosa instalación de WordPress en 5 minutos.

9.8 Instalación de los datos de prueba

Descargar el archivo [Theme unit test data](#).

```
wget https://wpcom-themes.svn.automattic.com/demo/theme-unit-test-data.xml
```

En el panel de administración de WordPress vamos a:

Herramientas -> Importar -> WordPress

Tenemos que instalar en plugin de importación [WordPress Importer](#).

Una vez descargado e instado el plugin mediante el asistente de instalación de plugins, hacemos click en el botón “Activar plugin e iniciar la importación”.

Seleccionamos el archivo que acabamos de descargar “theme-unit-test-data.xml” y hacemos clic en “Subir el archivo e importar”.

En la opción “Assign Authors” dejamos todo tal y como está, de tal forma que importa los 4 autores que trae el archivo.

En “Import Attachments” seleccionamos la opción “Download and import file attachments” y hacemos clic en el botón “Submit”.

Tras un tiempo de ejecución, en función de la potencia del equipo y de la conexión a Internet, tenemos realizada la importación de datos de prueba.

WordPress nos recuerda que debemos de actualizar las contraseñas y los roles de los usuarios importados. Esto es muy importante si lo llevamos a cabo en un equipo expuesto en Internet.

10 Desarrollando un tema hijo

10.1 Clonado del tema

Para obtener la última versión del plugin nos situamos en la carpeta

```
cd /wp-content/themes/
```

de WordPress y traemos la última versión del tema, usando Git.

```
sudo git clone https://github.com/amieiro/error500.git
```

Luego cambiamos los permisos para no tener ningún problema de ejecución

```
sudo chmod 775 error500 -R
```

Lo siguiente es ver el historial de commits del proyecto, mediante la orden

```
cd error500  
git log --oneline --reverse
```

```
0228f45 Initial commit. Error500 theme as child theme from Quark 1.2.9.  
3494b99 Change the CSS in the main elements.  
218f864 Copy the header.php file from the Quark theme 1.2.9  
7f94ab1 Remove the logo and put a text title  
457e30d Change the style in the header  
3fd1424 Change the CSS in the main menu  
4933e7d Change the CSS in the sidebar  
4929aef Changes in the CSS in the main page, paginating elements and jQuery  
Archive List Widget  
6f2a621 Add the Media Queries  
7fe179a Add the images folder and 3 logos  
9283072 Copy the footer.php file from the Quark theme 1.2.9  
33a4112 Add the logo in the footer and a link to fontethemes.com  
2860baf Add the responsive menu for mobiles  
59cb37b Change the left padding in the responsive menu  
4e5f923 Add the social images in the media directory  
ab5aeaf Change the right padding in the sidebar images  
0bc5a20 Copy the content.php file from the Quark theme 1.2.9  
fe913f8 Change the position of the number of comments  
8fbd9e1 Include in the functions.php file the script to invoke a button toggle
```

```
1957d92 Arrange some problems in the end of the line
e2e3c22 Add the banner container in the home page
dbbaeb0 Add the GNU GPL 2.0 license
fc61859 Small changes in the footer
59c4a4d Remove the banner from the frontpage if it shows the latest posts
07e7cd0 Increase the line height in the titles
e9f463f Add the README.md
5b74a63 Update README.md
247caac Resolve a bugfix with the homepage banner
fc03400 Add the .gitignore file to the project
cdd9471 Small changes in the .entry-content li
529fc72 Removed the img border: .entry-content img
```

Ejecuto el siguiente comando para ocultar unas marcas de final de línea que aparecen en los diff por un error en la codificación de los archivos al inicio del proyecto.

```
sudo git config --global core.whitespace cr-at-eol
```

El proyecto tiene 31 commits

```
git rev-list HEAD --count
```

```
31
```

10.2 Análisis del proyecto

Lo siguiente que vamos a hacer es revisar los pasos en orden cronológico inverso.

Antes de continuar vamos a la zona de administración de WordPress y activamos el tema error500.

10.2.1 Commit 1

En este paso regresamos al primer commit, en el que creamos la CSS del tema hijo vacía: solo tiene cabecera y el import a la hoja del tema padre.

```
sudo git reset --hard 0228f45 # reset_01
ls
```

```
style.css
```

```
cat style.css
```

```

/*
Theme Name:      Error 500
Theme URI:      http://fontethemes.com/es/error-500/
Description:    Error 500 theme
Author:         Jesus Amieiro
Author URI:     http://fontethemes.com
License:       GNU General Public License v2 or later
License URI:   http://www.gnu.org/licenses/gpl-2.0.html
Tags:          white, one-column, two-columns, right-sidebar, custom-menu,
translation-ready
Template:       quark
Version:        2.0.0
*/

/* -----
   Import the parent CSS theme
   ----- */
@import url("../quark/style.css");

```

10.2.2 Paso 2

Vamos al segundo commit. En este paso añadimos una fuente Google externa, las CSS en los elementos principales con resets, colores de fuentes y enlaces y eliminamos la imagen de fondo en el body.

```

sudo git reset --hard 3494b99 # reset_02
git diff 0228f45..3494b99

```

Los cambios introducidos se pueden ver a continuación

```

/* -----
   Import the external font
   ----- */

@import url(http://fonts.googleapis.com/css?family=Source+Sans+Pro);

/* -----
   Apply the external font
   Change the font color: #333332
   Apply the resets in the theme
   Change the a and a:visited color: #2790ae with underline
   Change the a:hover color: #0f6780

```

```

Remove the background image
----- */

html, body, div, span, applet, object, iframe,
h1, h2, h3, h4, h5, h6, p, blockquote, pre,
a, abbr, acronym, address, big, cite, code,
del, dfn, em, font, ins, kbd, q, s, samp,
small, strike, strong, sub, sup, tt, var,
dl, dt, dd, ol, ul, li,
fieldset, form, label, legend,
table, caption, tbody, tfoot, thead, tr, th, td {
    border: 0;
    color:#333332;
    font-family: 'Source Sans Pro', sans-serif;
    margin: 0;
    outline: 0;
    padding: 0;
    vertical-align: baseline;
}

a, a:visited {
    color: #2790ae;
    text-decoration: underline;
}

a:hover{
    color:#0f6780;
}

body {
    background-image:none !important;
}

```

10.2.3 Paso 3

Copiamos el fichero header.php del tema padre

```

sudo git reset --hard 218f864 # reset_03
git diff 3494b99..218f864

```

No mostramos la salida completa del diff porque es el contenido completo del archivo que acabamos de

copiar, que tiene 108 líneas.

```
diff --git a/header.php b/header.php
new file mode 100755
index 0000000..de6f715
--- /dev/null
+++ b/header.php
@@ -0,0 +1,108 @@
```

10.2.4 Paso 4

Borramos el logotipo de la cabecera e insertamos un título de texto

```
sudo git reset --hard 7f94ab1 # reset_04
git diff 218f864..7f94ab1
```

10.2.5 Paso 5

Cambiamos el estilo en del título y del subtítulo del encabezado y oculto el banner horizontal.

```
sudo git reset --hard 457e30d # reset_05
git diff 7f94ab1..457e30d
```

```
/* -----
    Blog title
----- */
.site-title h1 a {
    font-size:3.75rem;
    font-weight: bold;
    color:#333332;
}

/* -----
    Blog subtitle
----- */
h2.site-description {
    font-size:0.9rem;
    color:#333332 !important;
}
/* -----
```



```

        Remove the horizontal banner
    ----- */
    #bannercontainer {
        display:none;
    }

```

10.2.6 Paso 6

Cambiamos las CSS en el menú principal

```

sudo git reset --hard 3fd1424 # reset_06
git diff 457e30d..3fd1424

```

```

/* -----
    Main menu
    ----- */
.main-small-navigation a, .main-navigation a {
    display: block;
    text-decoration: none;
    color: #333332;
    font-size: 1.3rem;
    font-style: normal;
    font-weight: normal;
}

.main-small-navigation a:hover, .main-navigation a:hover {
    color:#0f6780 !important;
}

```

10.2.7 Paso 7

Cambiamos las CSS en la barra lateral

```

sudo git reset --hard 4933e7d # reset_07
git diff 3fd1424..4933e7d

```

```

/* -----
    Sidebar. Widget
    ----- */
.widget {

```

```

    margin: 0 0 1.5em !important;
}

.widget-title {
    color: #333332;
    font-size: 1rem;
    border-bottom: 1px #333332 dotted;
    margin-bottom: 0.625rem;
    margin-right: 1rem;
    padding-left: 0.3rem;
    text-transform: uppercase;
}

.textwidget img {
    float:right;
    padding:1rem 2rem 0 0;
}

/* -----
   Sidebar. Search form
   ----- */
#searchform {
    text-align:right;
    margin: 10px 10px 0px 0px;
}

.widget_search label {
    display:none;
}

#searchsubmit {
    display:none;
}

#searchform input[type="text"] {
    padding: 0.1rem 0.5rem !important;
    margin:0;
    height: 2rem;
}

```

```

/* -----
   Sidebar. Twitter widget. http://wordpress.org/plugins/really-simple-
   twitter-feed-widget/
   ----- */
.widget_reallysimpletwitterwidget ul {
    margin: 0 0 0 1.250rem;
    padding: 0;
}

.widget_reallysimpletwitterwidget ul li {
    list-style-type: none;
    margin: 0.188rem 1.250rem 0.625rem 0.625rem;
    padding: 0 0 0.5rem 0;
    border-bottom: 1px #000 dotted;
    font-size: 0.875rem;
}

/* -----
   Sidebar. Subscription form
   ----- */
.subscription-mail {
    text-align: center;
}

.subscription-text {
    font-family: 'Source Sans Pro', sans-serif;
    padding: 0.1rem 0.5rem !important;
    margin: 0;
    height: 2rem;
}

.subscription-button {
    font-family: 'Source Sans Pro', sans-serif;
    padding: 0.1rem 0.5rem !important;
    height: 2rem;
    font-size: 0.875rem;
}

```

10.2.8 Paso 8

Cambiamos las CSS en la página principal, en los elementos de paginación y en el plugin [jQuery Archive List Widget](#)

```
sudo git reset --hard 4929aef # reset_08
git diff 4933e7d..4929aef
```

```
/* -----
   Main page
   ----- */

/* Styles in the entry titles */
.entry-header h1, .entry-header h1 a, .entry-header h1 a:visited {
    color: #333332;
    font-size: 2.5rem;
    line-height: 2rem;
    font-weight: bold;
    text-decoration: none;
    padding: 1.6rem 0 1rem 0;
}

.entry-header h1 a:hover{
    color:#0f6780;
}

/* Remove the header-meta borders */
.header-meta {
    border:0;
}

/* Remove elements in the header-meta */
.header-meta .fa-calendar, .header-meta .entry-date, .header-meta .fa-pencil,
a.url.fn.n, .header-meta .post-categories {
    display:none !important;
}

/* Align the comments to the left */
.header-meta .comments-link {
    float:left !important;
}
```

```

}

/* Insert a border to the images */
.entry-content img {
  border: 1px #DDD solid;
}

/* Change the p style in the entry content */
.entry-content p {
  -webkit-font-smoothing: antialiased;
  color: #333332;
  letter-spacing: .01rem;
  font-style: normal;
  font-weight: 300;
  font-size: 21px;
  line-height: 28px;
  margin: 1rem 0 1rem 0;
}

/* Change the blockquote style in the entry content */
.entry-content blockquote p {
  color: #333332;
  letter-spacing: .01rem;
  font-style: normal;
  font-weight: 500;
  font-size: 21px;
  line-height: 28px;
  margin: 0 1.5rem 0 1.5rem;
  padding: 0 0 0 1rem;
  border-left: 3px #DDD solid;
  -webkit-font-smoothing: antialiased;
}

/* -----
   Paginating elements
   ----- */

li a:hover.page-numbers {
  background-color: #2790ae;

```

```

        color: #fff;
    }
    .nav-previous a, .nav-next a {
        display:none !important;
    }

    /* -----
       jQuery Archive List Widget. http://wordpress.org/plugins/jquery-archive-
       list-widget/
       ----- */
    ul.jaw_widget, .jaw_years ul
    {
        list-style-type: none;
    }

    .jaw_symbol {
        display:none;
    }

```

10.2.9 Paso 9

Añadimos las Media Queries

- <= 800px
- <= 520px

```

sudo git reset --hard 6f2a621 # reset_09
git diff 4929aef..6f2a621

```

```

/* =====
   Media Queries
   ===== */
/* -----
   <= 800px
   ----- */
@media only screen and (max-width: 800px) {
    header .main-navigation {
        margin-top: 1rem;
    }
}

```

```

/* Center the main menu */
#headercontainer {
    text-align:center;
}

.main-navigation {
    display: inline-block;
}

.main-navigation a {
    display: block;
}

/* Go full Width at less than 800px */
.grid_1_of_12,
.grid_2_of_12,
.grid_3_of_12,
.grid_4_of_12,
.grid_5_of_12,
.grid_6_of_12,
.grid_7_of_12,
.grid_8_of_12,
.grid_9_of_12,
.grid_10_of_12,
.grid_11_of_12,
.grid_12_of_12 {
    width: 100%;
}

.col {
    margin: 0;
}

.main-small-navigation .nav-menu, .nav-menu ul {
    padding: 0;
}

.main-small-navigation .nav-menu ul {
    padding-left: 1rem;
}

.entry-content blockquote p {
margin: 0 0 0 0;
}

```

```

padding: 0 0 0 1rem;
}
}
/* -----
<= 520px
----- */
@media only screen and (max-width: 520px) {
  .site-title h1 a {
    font-size:2.5rem;
    font-weight: bold;
    //height: 4rem;
    line-height: 0rem;
    padding: 0;
    margin: 0;
    color:#000;
  }
  .site-title h1 {
    text-align:right;
    padding: 0;
    margin: 0;
  }
  h2.site-description {
    display:none;
  }
  .menu-toggle {
    font-style: normal;
    font-weight: 400;
    font-size: 1rem;
    line-height: 1.25;
    margin: 1rem 0 0 0;
    font-size: 100%;
    vertical-align: middle;
  }
  .menu-toggle,
  .btn,
  input[type="submit"] {
    display: inline-block;
    *display: inline;
    padding: 0.5rem 0.875rem;

```



```

margin-bottom: 0;
*margin-left: .3em;
font-size: 16px;
font-size: 1rem;
line-height: 1.25;
color: #fff;
text-align: center;
text-shadow: 0 1px 1px rgba(255, 255, 255, 0.75);
vertical-align: middle;
cursor: pointer;
background-color: #000 !important;
*background-color: #000;
background-image: -ms-linear-gradient(top, #ffffff, #000);
background-image: -webkit-gradient(linear, 0 0, 0 100%, from(#000),
to(#000));
background-image: -webkit-linear-gradient(top, #000, #000);
background-image: -o-linear-gradient(top, #000, #000);
background-image: linear-gradient(top, #000, #000);
background-image: -moz-linear-gradient(top, #000, #000);
background-repeat: repeat-x;
border: 1px solid #cccccc;
*border: 0;
border-color: rgba(0, 0, 0, 0.1) rgba(0, 0, 0, 0.1) rgba(0, 0, 0,
0.25);
border-color: #000 #000 #000;
border-bottom-color: #000;
-webkit-border-radius: 0px;
-moz-border-radius: 0px;
border-radius: 0px;
filter:
progid:dximagetransform.microsoft.gradient(startColorstr='#ffffff',
endColorstr='#e6e6e6');
*zoom: 1;
-webkit-box-shadow: none;//inset 0 1px 0 rgba(255, 255, 255, 0.2), 0
1px 2px rgba(0, 0, 0, 0.05);
-moz-box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px
rgba(0, 0, 0, 0.05);
box-shadow: inset 0 1px 0 rgba(255, 255, 255, 0.2), 0 1px 2px rgba(0,
0, 0, 0.05);
}
.menu-toggle, a.btn {
color: #FFF;

```

```

        text-decoration: none;
        background-color: #000;
        font-size: 1.5rem;
        font-style: normal;
        font-weight: normal;
    }

    .menu-toggle:hover,
    .btn:hover,
    input[type="submit"]:hover {
        color: #FFF;
        text-decoration: none;
        background-color: #000;
        *background-color: #000; /* Buttons in IE7 don't get borders, so
darken on hover */
        background-position: 0 -0px;
        -webkit-transition: background-position 0.1s linear;
        -moz-transition: background-position 0.1s linear;
        -ms-transition: background-position 0.1s linear;
        -o-transition: background-position 0.1s linear;
        transition: background-position 0.1s linear;
    }

    .main-small-navigation {
        text-align:left;
        min-width:15rem;
        max-width:100%;
        width:100% !important;
        float:left;
        clear:left;
        position:relative;
        z-index:20;
    }

    .main-small-navigation ul li,
    .main-navigation ul li {
        background-color: #2790ae;
        margin: 0.1rem 0;
        padding: 0.2rem 0 0 0.3rem;
        text-align: left;
    }

```

```

}

.main-small-navigation ul li a, .main-small-navigation ul li a:hover,
.main-navigation ul li a, .main-navigation ul li a:hover {
    color:#FFF !important;
    background-color:#2790ae;
}

.main-small-navigation a, .main-small-navigation a:visited {
    font-size: 1.5rem;
    font-style: normal;
    font-weight: normal;
}

.grid_5_of_12 {
    width:80% !important;
    float:right;
    border: 0px #000 solid;
    padding: 0;
    margin: 0.5rem 0 0 0;
}

.grid_7_of_12 {
    width:20% !important;
    float:left;
    border: 0px #000 solid;
    padding: 0;
    margin: 0;
}
}
}

```

10.2.10 Paso 10

Añadimos el directorio “images” y tres logotipos.

```

sudo git reset --hard 7fe179a # reAset_10
git diff 6f2a621..7fe179a

```

```

diff --git a/images/fonte_themes_developed_by-150x78.png
b/images/fonte_themes_developed_by-150x78.png
new file mode 100755
index 0000000..b82c0d1
Binary files /dev/null and b/images/fonte_themes_developed_by-150x78.png
differ
diff --git a/images/fonte_themes_developed_by.png
b/images/fonte_themes_developed_by.png
new file mode 100755
index 0000000..a323927
Binary files /dev/null and b/images/fonte_themes_developed_by.png differ
diff --git a/images/fonte_themes_developed_by_white.png
b/images/fonte_themes_developed_by_white.png
new file mode 100755
index 0000000..bc549b3
Binary files /dev/null and b/images/fonte_themes_developed_by_white.png differ

```

10.2.11 Paso 11

Copiamos el fichero footer.php del tema padre

```

sudo git reset --hard 9283072 # reset_11
git diff 7fe179a..9283072

```

No mostramos la salida completa del diff porque es el contenido completo del archivo que acabamos de copiar, que tiene 63 líneas.

```

diff --git a/footer.php b/footer.php
new file mode 100755
index 0000000..041b610
--- /dev/null
+++ b/footer.php
@@ -0,0 +1,63 @@

```

10.2.12 Paso 12

Añadimos el logotipo en el pie de la página y un enlace a fontethemes.com

```

sudo git reset --hard 33a4112 # reset_12
git diff 9283072..33a4112

```

```

echo '<p><a href="http://fontethemes.com" alt="Fonte Themes"
target="_blank"></a></p>';
```

10.2.13 Paso 13

Añadimos el menú adaptativo para móviles y cambiamos las etiquetas en la definición del tema en el fichero style.css.

```
sudo git reset --hard 2860baf # reset_13
git diff 33a4112..2860baf
```

```
diff --git a/header.php b/header.php
index fb9ec75..bdd1026 100755
--- a/header.php
+++ b/header.php
@@ -66,7 +66,7 @@

<h3 class="menu-toggle assistive-text"><i class="fa fa-bars"></i><!--?php
esc_html_e( 'Menu', 'quark' ); ?>--></h3>
```

```
diff --git a/style.css b/style.css
index cd9d3ff..41104d9 100755
--- a/style.css
+++ b/style.css
@@ -6,7 +6,7 @@
```

```
Tags: black, gray, dark, light, one-column, two-columns, right-sidebar, fluid-
layout, responsive-layout, custom-background, custom-header, custom-menu,
editor-style, featured-image-header, featured-images, full-width-template,
microformats, post-formats, sticky-post, theme-options, threaded-comments,
translation-ready
```

10.2.14 Paso 14

Cambio el padding-left en el menú adaptativo

```
sudo git reset --hard 59cb37b # reset_14
git diff 2860baf..59cb37b
```

```
.main-small-navigation .nav-menu ul {
-         padding-left: 1rem;
+         padding-left: 0rem;
```

```
}
```

10.2.15 Paso 15

Añado las imágenes sociales en el directorio “media”

```
sudo git reset --hard 4e5f923 # reset_15
git diff 59cb37b..4e5f923
```

```
diff --git a/images/google_square_black.png b/images/google_square_black.png
new file mode 100755
index 0000000..ccd5cab
Binary files /dev/null and b/images/google_square_black.png differ
diff --git a/images/rss_square_black.png b/images/rss_square_black.png
new file mode 100755
index 0000000..479b05c
Binary files /dev/null and b/images/rss_square_black.png differ
diff --git a/images/twitter_square_black.png b/images/twitter_square_black.png
new file mode 100755
index 0000000..855174e
Binary files /dev/null and b/images/twitter_square_black.png differ
```

10.2.16 Paso 16

Cambiamos el padding derecho en las imágenes de los widgets. La mayoría van a estar en la parte derecha.

```
sudo git reset --hard ab5aeaf # reset_16
git diff 4e5f923..ab5aeaf
```

```
.textwidget img {
    float:right;
-   padding:1rem 2rem 0 0;
+   padding:1rem 1rem 0 0;
}
```

10.2.17 Paso 17

Copiamos el fichero content.php del tema padre

```
sudo git reset --hard 0bc5a20 # reset_17
git diff ab5aeaf..0bc5a20
```

No mostramos la salida completa del diff porque es el contenido completo del archivo que acabamos de copiar, que tiene 63 líneas.

```
diff --git a/content.php b/content.php
new file mode 100755
index 0000000..5608e14
--- /dev/null
+++ b/content.php
@@ -0,0 +1,63 @@
```

10.2.18 Paso 18

Cambiamos la posición del número de comentarios

```
sudo git reset --hard fe913f8 # reset_18
git diff 0bc5a20..fe913f8
```

```
@@ -22,7 +22,7 @@
                                <a href="<?php the_permalink(); ?>"
title="<?php echo esc_attr( sprintf( esc_html__( '
                                </h1>
                                <?php } // is_single() ?>
-                                <?php quark_posted_on(); ?>
+                                <?php // quark_posted_on(); ?>
                                <?php if ( has_post_thumbnail() && !is_search() ) { ?>
                                    <a href="<?php the_permalink(); ?>" title="<?
php echo esc_attr( sprintf( esc_html__( 'Permalin
                                <?php
the_post_thumbnail( 'post_feature_full_width' ); ?>
@@ -54,6 +54,7 @@
                                // Only show the tags on the Single Post page
                                quark_entry_meta();
                                } ?>
+                                <?php quark_posted_on(); ?>
                                <?php edit_post_link( esc_html__( 'Edit', 'quark' ) .
' <i class="fa fa-angle-right"></i>', ' <div clas
                                <?php if ( is_singular() &&
get_the_author_meta( 'description' ) && is_multi_author() ) {
```

10.2.19 Paso 19

Incluimos el archivo functions.php y dentro de él el script para hacer visible o no visible el botón del menú en dispositivos móviles.

```
sudo git reset --hard 8fbd9e1 # reset_19
git diff fe913f8..8fbd9e1
```

```
diff --git a/functions.php b/functions.php
new file mode 100755
index 0000000..3bbcce4
--- /dev/null
+++ b/functions.php
@@ -0,0 +1,5 @@
+<?php
+
+// Include this script to invoke a button toggle for the main navigation menu
on small screens
+wp_register_script( 'small-menu',
trailingslashit( get_template_directory_uri() ) . 'js/small-menu.js',
array( 'jquery' ), '2
+wp_enqueue_script( 'small-menu' );
```

10.2.20 Paso 20

Modificamos algunos problemas en los finales de línea

```
sudo git reset --hard 1957d92 # reset_20
git diff 8fbd9e1..1957d92
```

Afecta a demasiado contenido como para mostrar la salida; además no es un commit significativo desde el punto de vista de la construcción del tema.

10.2.21 Paso 21

Añadimos el contenedor del banner en la portada para combinar con el plugin MetaSlider

```
sudo git reset --hard e2e3c22 # reset_21
git diff 1957d92..e2e3c22
```

```
diff --git a/style.css b/style.css
index fe9b650..1853e05 100755
--- a/style.css
```



```

+++ b/style.css
@@ -92,6 +92,12 @@ h2.site-description {
    display:none;
}

+.home #bannercontainer {
+    display:block;
+    overflow:auto;
+    //height:27rem;
+}
+
+/* -----
+    Main menu
+----- */
@@ -190,6 +196,14 @@ h2.site-description {
}

/* -----
+    Footer
+----- */
+
+.site-footer p {
+    color: #ccc;
+}
+
+/* -----
+    Main page
+----- */

@@ -470,3 +484,20 @@ ul.jaw_widget, .jaw_years ul
    margin: 0;
}
}
+
+/* -----
+    Metaslider plugin
+----- */
+.metaslider, .flexslider {
+    width:100% !important;

```

```
+}
+
+.banner {
+    width:100% !important;
+    padding: 0 !important;
+    max-width: 3500px;
+}
+
+.banner .widget {
+    margin: 0 !important;
+}
```

10.2.22 Paso 22

Añadimos la licencia GNU/GPL v2

```
sudo git reset --hard dbbaeb0 # reset_22
git diff e2e3c22..dbbaeb0
```

```
diff --git a/gpl-2.0.txt b/gpl-2.0.txt
new file mode 100644
index 0000000..d159169
--- /dev/null
+++ b/gpl-2.0.txt
@@ -0,0 +1,339 @@
```

10.2.23 Paso 23

Pequeños cambios en el footer

```
sudo git reset --hard fc61859 # reset_23
git diff dbbaeb0..fc61859
```

```
diff --git a/footer.php b/footer.php
index bd0b42a..66ad258 100755
--- a/footer.php
+++ b/footer.php
@@ -49,7 +49,7 @@
```

```

        <?php if ( of_get_option( 'footer_content',
quark_get_credits() ) ) {
            echo '<div class="row smallprint">';
-            echo '<p><a href="http://fontethemes.com" alt="Fonte
Themes" target="_blank"><img src="' . get_styles
            echo apply_filters( 'meta_content',
wpkses_post( of_get_option( 'footer_content', quark_get_credits()
            echo '</div> <!-- /.smallprint -->';
        } ?>

diff --git a/images/fonte_themes_developed_by_white_200_55.png
b/images/fonte_themes_developed_by_white_200_55.png
new file mode 100644
index 0000000..626de66
Binary files /dev/null and b/images/fonte_themes_developed_by_white_200_55.png
differ

diff --git a/style.css b/style.css
index 1853e05..03cd580 100755
--- a/style.css
+++ b/style.css
@@ -203,6 +203,14 @@ h2.site-description {
    color: #ccc;
}

+.smallprint p {
+    color: #ccc;
+}
+
+.site-footer .widget_reallysimpletwitterwidget ul li {
+    color: #ccc !important;
+}
+

```

10.2.24 Paso 24

Borramos el contenedor del banner de la portada si se muestran los últimos posts.

```

sudo git reset --hard 59c4a4d # reset_24
git diff fc61859..59c4a4d

```

```

diff --git a/style.css b/style.css
index 03cd580..4a32361 100755
--- a/style.css
+++ b/style.css
@@ -92,7 +92,7 @@ h2.site-description {
        display:none;
    }

-.home #bannercontainer {
+.page #bannercontainer {
        display:block;
        overflow:auto;
        //height:27rem;

```

10.2.25 Paso 25

Cambiamos el alto de línea en los títulos, ya que se pisaban los textos.

```

sudo git reset --hard 07e7cd0 # reset_25
git diff 59c4a4d..07e7cd0

```

```

diff --git a/style.css b/style.css
index 4a32361..270ab5a 100755
--- a/style.css
+++ b/style.css
@@ -219,7 +219,7 @@ h2.site-description {
    .entry-header h1, .entry-header h1 a, .entry-header h1 a:visited {
        color: #333332;
        font-size: 2.5rem;
-       line-height: 2rem;
+       line-height: 2.5rem;
        font-weight: bold;
        text-decoration: none;
        padding: 1.6rem 0 1rem 0;

```

10.2.26 Paso 26

Añadimos el archivo README.md para publicar en GitHub.

```

sudo git reset --hard e9f463f # reset_26

```

```
git diff 07e7cd0..e9f463f
```

```
diff --git a/README.md b/README.md
new file mode 100644
index 0000000..c672d4a
--- /dev/null
+++ b/README.md
@@ -0,0 +1,8 @@
+Error 500 WordPress theme =====
+
+**Error 500** is a clean and responsive WordPress theme, developed as a
[Quark child theme][1].
+
+You can see a demo in the [demo page][2] (in spanish) and its main
characteristics in the [characteristics page][3] (in spani
+
+ [1]: http://quarktheme.com/
+ [2]: http://error500.fontethemes.com/
+ [3]: http://error500.fontethemes.com/caracteristicas/
```

10.2.27 Paso 27

Actualizamos el archivo README.md

```
sudo git reset --hard 5b74a63 # reset_27
git diff e9f463f..5b74a63
```

```
diff --git a/README.md b/README.md
index c672d4a..6d3562d 100644
--- a/README.md
+++ b/README.md
@@ -1,4 +1,5 @@
-Error 500 WordPress theme =====
+Error 500 WordPress theme
+=====
```

10.2.28 Paso 28

Resolvemos un error con el banner de la portada

```
sudo git reset --hard 247caac # reset_28
```

```
git diff 5b74a63..247caac
```

```
diff --git a/style.css b/style.css
index 270ab5a..52b365d 100755
--- a/style.css
+++ b/style.css
@@ -92,7 +92,7 @@ h2.site-description {
     display:none;
 }

-.page #bannercontainer {
+.home.page #bannercontainer {
     display:block;
     overflow:auto;
     //height:27rem;
@@ -256,8 +256,8 @@ h2.site-description {
     letter-spacing: .01rem;
     font-style: normal;
     font-weight: 300;
-font-size: 21px;
-line-height: 28px;
+font-size: 1.3rem;
+line-height: 1.75rem;
     margin: 1rem 0 1rem 0;
 }
```

10.2.29 Paso 29

Añado el archivo .gitignore al proyecto

```
sudo git reset --hard fc03400 # reset_29
git diff 247caac..fc03400
```

```
diff --git a/.gitignore b/.gitignore
new file mode 100644
index 0000000..8dc7a10
--- /dev/null
+++ b/.gitignore
```

```
@@ -0,0 +1 @@
+todo.txt
```

10.2.30 Paso 30

Realizamos cambios pequeños en “.entry-content li”

```
sudo git reset --hard cdd9471 # reset_30
git diff fc03400..cdd9471
```

```
diff --git a/style.css b/style.css
index 52b365d..4830c7e 100755
--- a/style.css
+++ b/style.css
@@ -250,7 +250,7 @@ h2.site-description {
 }

/* Change the p style in the entry content */
-.entry-content p {
+.entry-content p, .entry-content li {
    -webkit-font-smoothing: antialiased;
    color: #333332;
    letter-spacing: .01rem;
@@ -261,6 +261,10 @@ h2.site-description {
    margin: 1rem 0 1rem 0;
 }

+.entry-content li {
+    margin: 1rem 0 -1rem 2rem;
+}
+
/* Change the blockquote style in the entry content */
.entry-content blockquote p {
    color: #333332;
```

10.2.31 Paso 31

Borramos el contorno en las imágenes en las entradas “.entry-content img”

```
sudo git reset --hard 529fc72 # reset_31
```

```
git diff cdd9471..529fc72
```