



Módulo de Proyecto Integrado

Pablo Murillo Ávila - 2º ASIR

Migración de la bitácora Helvia del IES Averroes a WordPress

Curso 2017/2018

Índice de contenidos

1.- Introducción	3
2.- Objetivos y requisitos del proyecto	3
3.- Estudio previo	3
3.1.- Estado actual.....	3
3.2.- Estudio de soluciones existentes	3
3.2.1.- Tipo de aplicación	3
3.2.1.1.- Aplicación tipo script.....	3
3.2.1.2.- Aplicación tipo web	4
3.2.2.- Formato de entrada.....	4
3.2.2.1.- Copia de seguridad de la base de datos de Helvia.....	4
3.2.2.2.- Web scraping	4
3.2.3.- Formato de salida.....	5
3.2.3.1.- Directamente en la base de datos	5
3.2.3.2.- Archivo XML compatible con el "Importador de WordPress"	5
3.2.4.- Solución elegida	5
4.- Plan de trabajo	6
5.- Diseño.....	6
5.1.- Diseño general.....	6
5.2.- Diseño detallado	7
5.2.1.- Copia de seguridad de Helvia	7
5.2.2.- Script PHP	8
5.2.3.- Archivo XML	9

5.2.4.- WordPress	10
6.- Implantación.....	10
6.1.- Entorno de pruebas	10
6.2.- Entorno real	16
7.- Recursos.....	21
7.1.- Herramientas hardware	21
7.2.- Herramientas software	21
7.3.- Personal	21
7.4.- Presupuesto	21
8.- Conclusiones.....	22
8.1.- Grado de consecución de objetivos	22
8.2.- Problemas encontrados	22
8.3.- Futuras mejoras	22
9.- Referencias / bibliografía.....	22
10.- Anexos.....	22

1.- Introducción

Este proyecto nace a partir de una petición del IES Averroes de Córdoba. Su intención consistirá en migrar la bitácora de la plataforma educativa Helvia de este centro a WordPress, ya que dicha plataforma se encuentra actualmente anticuada y sin soporte.

Aunque Helvia cuente con varias secciones (página web, bitácora y aulas virtuales), este proyecto se centrará únicamente en la migración de la bitácora.

Se migrarán todos los datos posibles que se puedan extraer de Helvia para ser importados en WordPress (es decir, los datos equivalentes entre ambos). Para ello, habrá que convertir todas las entradas que hay publicadas en el sitio, las categorías, los usuarios, los comentarios... al formato de WordPress.

2.- Objetivos y requisitos del proyecto

El objetivo del proyecto es la migración de la bitácora Helvia del IES Averroes a un blog desarrollado con WordPress.

Los requisitos a tener en cuenta son los siguientes:

- El acceso a los datos y los recursos originales podrá ser a través de una copia de seguridad o mediante "Web scraping".
- La migración debe realizarse de forma automatizada.
- Las entidades a migrar estarán definidas por los elementos equivalentes entre Helvia y WordPress.
- El blog de destino debe implementarse con WordPress.

3.- Estudio previo

3.1.- Estado actual

Actualmente no existe ninguna solución para llevar a cabo una migración de datos entre Helvia y WordPress de forma automática. Por lo tanto, habrá que desarrollar una completamente desde cero.

3.2.- Estudio de soluciones existentes

3.2.1.- Tipo de aplicación

Según el tipo de aplicación que se desarrolle finalmente, se proponen dos formatos diferentes: una aplicación tipo script y una aplicación web.

3.2.1.1.- Aplicación tipo script

Mediante la utilización de un script y a partir de una copia de seguridad de la base de datos MySQL de y/o mediante "Web scraping" ir exportando los datos.

Ventajas:

- No es necesario diseñar una interfaz de usuario para la aplicación.

- Necesita menos recursos para funcionar, pues la máquina en la que se ejecuta no necesita interfaz gráfica.

Inconvenientes:

- Más complejo para usuarios menos avanzados.
- La aplicación es menos llamativa.

3.2.1.2.- Aplicación tipo web

Utilizando una interfaz web para llevar a cabo la conversión de los datos.

Ventajas:

- Interfaz más sencilla y fácil de entender para un usuario normal.

Inconvenientes:

- Necesita un servidor web para funcionar.
- Hay que diseñar la interfaz de usuario además de la propia aplicación.

3.2.2.- Formato de entrada

Según el formato de entrada de los datos que se van a exportar, se proponen dos tipos: una copia de seguridad de la base de datos de Helvia o Web scraping.

3.2.2.1.- Copia de seguridad de la base de datos de Helvia

Ventajas:

- La conversión se hace de manera más rápida, ya que se dispone de toda la información en local.
- Se puede ver cómo está estructurada la base de datos.

Inconvenientes:

- Puede que no esté disponible la copia de seguridad por políticas de seguridad de la Consejería de Educación de la Junta de Andalucía.

3.2.2.2.- Web scraping

El Web scraping es una técnica que permite, mediante un programa, extraer información de un sitio web. Estos programas simulan la navegación de un humano, utilizando el protocolo HTTP.

Ventajas:

- Permite obtener información que no se podría conseguir de otra forma.

Inconvenientes:

- El proceso es mucho más lento y es más probable que haya errores durante el proceso, ya que hay que estar constantemente haciendo peticiones al servidor web.

- El lenguaje de programación que se utilice debe disponer de herramientas para poder hacerlo.

3.2.3.- Formato de salida

Según el formato de salida, se proponen los siguientes dos métodos: importar los datos directamente en la base de datos de WordPress o generar un archivo de salida tipo XML que pueda ser interpretado por el "Importador de WordPress".

3.2.3.1- Directamente en la base de datos

La aplicación transfiere directamente la información equivalente entre ambas plataformas de la base de datos original (Helvia) a la nueva (WordPress).

Ventajas:

- El proceso estaría terminado con solo ejecutar el script.

Inconvenientes:

- Hay que conocer a fondo la estructura de la base de datos de WordPress para saber cómo deben importarse los datos.
- Hay riesgo de corromper la base de datos al estar manipulándola directamente.
- Se necesita acceso total a la base de datos de WordPress, lo cual puede ser un problema si queremos hacer la importación en un WordPress en un alojamiento compartido.

3.2.3.2.- Archivo XML compatible con el "Importador de WordPress"

La aplicación generará un archivo XML, compatible con el "Importador de WordPress", con toda la información del sitio.

Ventajas:

- No es necesario manipular directamente la base de datos de WordPress, sino que es el plugin el que se encarga de insertar los datos donde corresponde.
- No es necesario tener acceso directamente a la base de datos. Si se importa en un WordPress en un hosting compartido es posible que no se tenga acceso.
- Solo es necesario trabajar con la base de datos de origen.

Inconvenientes:

- Es necesario conocer la estructura de los archivos XML que utiliza el "Importador de WordPress" para poder generarlos mediante el script.

3.2.4.- Solución elegida

Tipo de aplicación (elegida): Aplicación tipo script.

Razonamiento: El tipo de aplicación más adecuado es el script. Se ha elegido el lenguaje PHP porque es sencillo y fácil de implementar, además de ser multiplataforma, ser de

código abierto y disponer de muchas funciones y documentación. La aplicación tipo web no habría aportado muchas ventajas frente al script.

Tipo de formato de entrada (elegido): Copia de seguridad de la base de datos de Helvia.

Razonamiento: Finalmente se ha tenido acceso a una copia de seguridad de la base de datos de Helvia. La alternativa era el “Web scraping”, que es un proceso mucho más lento y que además no nos permite ver la estructura de la base de datos.

Tipo de formato de salida (elegido): Archivo XML.

Razonamiento: Ya que es posible generar un archivo de importación XML conociendo cómo es su formato, es más viable hacerlo así que manipulando directamente la base de datos de WordPress.

4.- Plan de trabajo

Semana	Fase del proyecto
19/03-23/03	Introducción al proyecto.
02/04-06/04	Objetivos y requisitos.
09/04-13/04	Estudio de soluciones existentes.
16/04-20/04	Estudio de soluciones existentes y definición de solución elegida.
23/04-27/04	Preparación de un entorno de pruebas y comienzo del desarrollo de la herramienta.
30/04-04/05	Desarrollo de la herramienta.
07/05-11/05	Desarrollo de la herramienta.
14/05-18/05	Desarrollo de la herramienta.
21/05-25/05	Desarrollo de la herramienta.
28/05-01/06	Desarrollo de la herramienta.
04/06-08/06	Compra del dominio y el hosting e importación de los datos en el sitio WordPress final.
11/06-15/06	Preparación de presentación del proyecto y pruebas finales.
18/06-22/06	Presentación del proyecto.

5.- Diseño

5.1.- Diseño general



La herramienta utiliza el backup de Helvia para generar un archivo XML que posteriormente se importará al WordPress de destino mediante el plugin “Importador de WordPress”.

El Backup de Helvia, proporcionado por el IES Averroes, consta de un archivo ZIP en el que se encuentran todos los archivos del sitio y los archivos SQL de las bases de datos (de los cuales para este proyecto solo se utilizan los de la bitácora). Toda la información la contiene la base de datos a excepción del contenido de las publicaciones, que se encuentra en archivos HTML independientes para cada una de ellas.

El Script PHP es la herramienta que se encarga de extraer de la base de datos y de los archivos HTML la información necesaria para generar el archivo XML. También tiene un archivo de configuración para ajustar algunos parámetros como la conexión a la base de datos, el nombre y la descripción que tendrá el sitio, la ruta del directorio con los archivos HTML, etc.

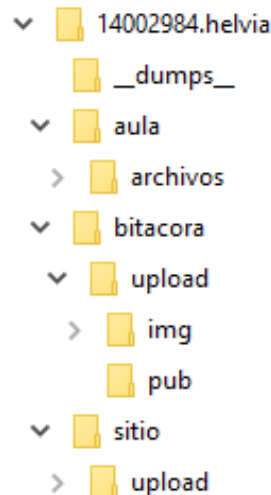
El Archivo XML generado mediante el script sigue la estructura que reconoce el "Importador de WordPress" y contiene toda la información del sitio lista para ser importada.

El WordPress final es el sitio donde importaremos el archivo XML con toda la información de la bitácora de Helvia.

5.2.- Diseño detallado

5.2.1.- Copia de seguridad de Helvia

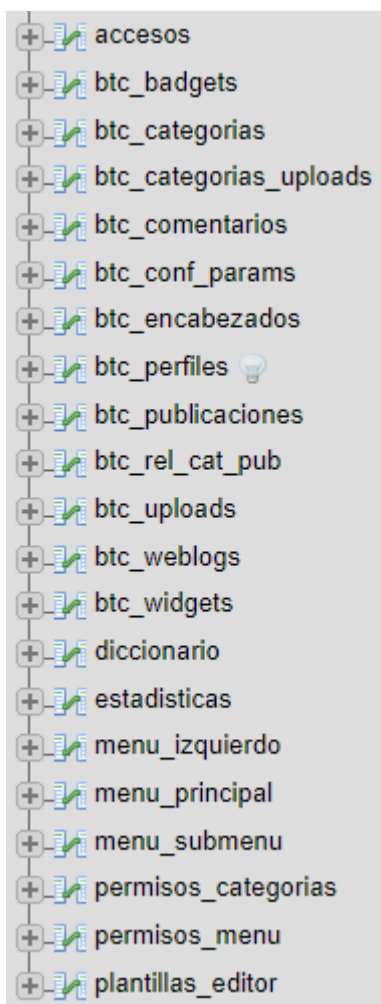
Se dispone de una copia de seguridad de Helvia completa, que tiene la siguiente estructura de directorios:



Árbol de directorios de la copia de seguridad

Nos centramos en el directorio "__dumps__", que es donde está el archivo "bitacora.sql" de la copia de seguridad y en el directorio "bitácora" y su subdirectorio "pub", donde están los archivos HTML que corresponden al contenido de cada una de las publicaciones.

Como las imágenes que se utilizan en la bitácora están alojadas en servidores externos, no es necesario hacer nada con ellas.



Base de de datos de Helvia

La base de datos contiene las siguientes tablas, de las cuales, para este proyecto se utilizan las siguientes:

- "btc_categorias": Contiene todas las categorías creadas (id, nombre, descripción, fecha de alta, etc.).
- "btc_comentarios": Contiene todos los comentarios que hay (id, fecha de alta, texto del comentario, id de la publicación y un identificador para definir si está activo o no) y los datos del usuario que lo publica (nombre y email).
- "btc_publicaciones": Contiene toda la información de las publicaciones (id, titulo, fecha de publicación, autor, email del autor, etiquetas y un identificador para definir si está activa o no), a excepción del contenido, que se encuentra en los archivos HTML.
- "btc_rel_cat_pub": Se utiliza para relacionar cada publicación con su categoría.

El resto de tablas que contiene la base de datos, tienen información propia de Helvia que no es común con WordPress y no se exportará.

5.2.2.- Script PHP

El script desarrollado es la herramienta encargada de realizar las peticiones al servidor MySQL con la base de datos de Helvia para obtener la información, y a su vez, accede a los archivos HTML de las publicaciones para obtener los contenidos.

Para ello, consta de un archivo de configuración donde se definen los parámetros de conexión a la base de datos y la ruta del directorio de las publicaciones, además de otras opciones.

El script consta de las siguientes partes:

- Definición de funciones: una para adaptar los títulos de las entradas, categorías, etiquetas y nombres de usuario al formato "slug" de WordPress (convertir los espacios en guiones, eliminar caracteres extraños, convertir todo a minúscula...) y otra para eliminar los emojis de Helvia de los comentarios, que se utiliza de forma opcional según esté definido en la configuración.
- Conexión a la base de datos.
- Creación del archivo: se crea el archivo y se añaden las primeras líneas, que incluyen el título del sitio, la descripción, la URL, etc.

- Descarga de los autores: como en la base de datos de Helvia no existe una tabla específica para los autores, sino que se definen en la misma tabla de las publicaciones junto con su email, se añaden a un array eliminando los duplicados, para posteriormente añadir una línea al archivo para cada uno de ellos.

- Descarga de categorías: se hacen consultas a la base de datos y se añade al archivo el bloque de líneas correspondiente para cada una de ellas.

- Descarga de etiquetas: en Helvia tampoco existe una tabla para las etiquetas. También se encuentran en una columna de la tabla de las publicaciones separadas por comas. Se hace una consulta de esta columna, se separan por las comas y se van añadiendo a un array, del cual luego se eliminan las duplicadas y, por último, se añade al archivo el bloque de líneas que corresponde a cada etiqueta.

El archivo XML comparte las mismas IDs para las categorías y las etiquetas, así que se utiliza un contador común para ambas con el que se van generando.

- Descarga de publicaciones y comentarios: se hacen consultas a la base de datos para obtener las publicaciones. También, para cada publicación, se hacen consultas individuales para obtener sus comentarios y categorías. Para obtener el contenido, se genera la ruta del archivo HTML al que hay que acceder a partir del ID de la publicación y la ruta del directorio donde se encuentran todos estos archivos que se haya especificado en la configuración del script.

Una vez termina de ejecutarse el script, generará un archivo XML que podremos importar en el WordPress de destino.

5.2.3.- Archivo XML

El XML es un meta-lenguaje que permite definir lenguajes de marcas, utilizado para almacenar datos en forma legible.

El archivo XML es generado por el script siguiendo la estructura que reconoce el "Importador de WordPress" y combina toda la información extraída de la base de datos y de los archivos HTML de las publicaciones.

Su estructura es la siguiente:

- Definición de espacios de nombre ("excerpt", "content", "wfw", "dc" y "wp").

- Etiqueta "<channel>", que engloba el resto del contenido y etiquetas del archivo.

- Etiquetas de configuración del sitio "<title>", "<link>", "<description>", etc.

- Etiquetas "<wp:author>", cada una de ellas contiene otras etiquetas que proporcionan información de un autor.

- Etiquetas "<wp:category>", cada una de ellas contiene otras etiquetas que contienen la información de una categoría.

- Etiquetas "<wp:tag>", cada una de ellas contiene otras etiquetas que contienen la información de una etiqueta.

- Etiquetas "<wp:term>". En WordPress, tanto las categorías como las etiquetas (y algunos otros) se consideran "objetos", ya que comparten características similares. Existe una etiqueta de este tipo para cada categoría y para cada etiqueta como un único conjunto.

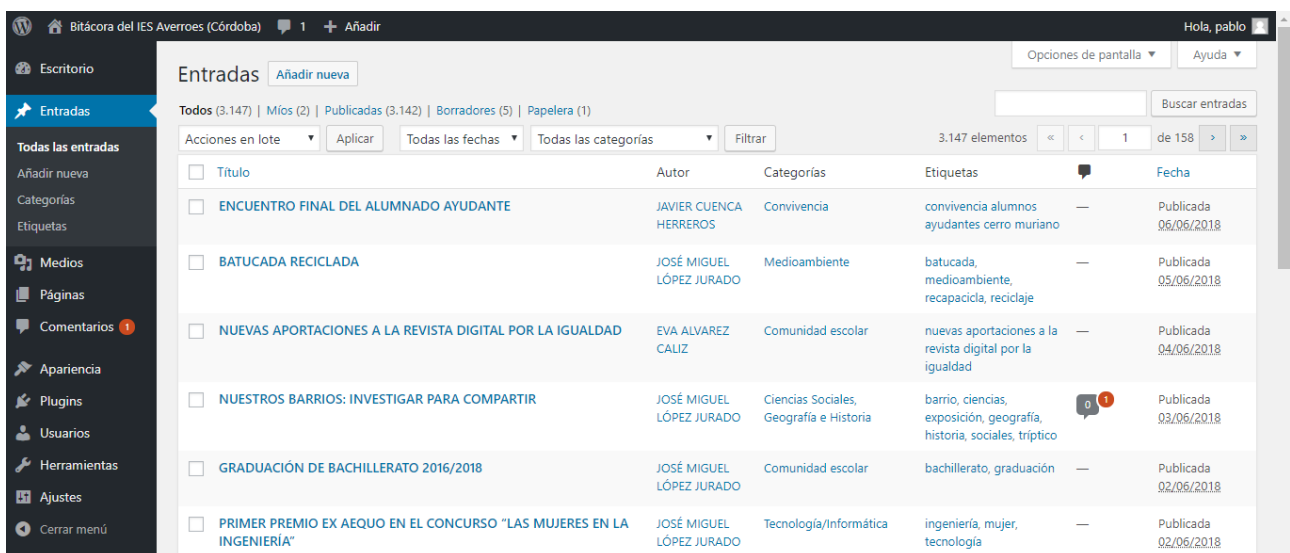
- Etiquetas “<item>”, contiene otras etiquetas que contienen la información relacionada con cada publicación (título, link, fecha y hora de publicación, contenido, opciones, etc).

5.2.4.- WordPress

WordPress es un sistema de gestión de contenidos enfocado a la creación de cualquier tipo de página web. Está desarrollado en el lenguaje PHP para entornos que ejecuten MySQL y Apache, bajo licencia GPL y es software libre.

La estructura que utiliza WordPress es parecida a la de Helvia, por lo que la mayoría de información se puede importar sin problemas.

En WordPress hay entradas, las cuales tienen etiquetas, están dentro de categorías y pertenecen a un único autor.



Escritorio del panel de administración de WordPress

6.- Implantación

6.1.- Entorno de pruebas

La implantación de la solución elegida se ha realizado primero utilizando un entorno de pruebas en una máquina virtual Debian 9.

En esta máquina virtual se han instalado PHP 7, MySQL y el servidor web Apache. Posteriormente se ha instalado WordPress en el servidor web de forma manual, descargándolo desde su página oficial (<https://es.wordpress.org/txt-download/>), descomprimiéndolo en el directorio del sitio y creando el sitio correspondiente en Apache.

```

root@Pablo: ~
root@wordpress:/var/www/wordpress# ls -l
total 220
-rw-r--r-- 1 root root 595 may 17 10:25 backup.xml
-rwxr-x--- 1 www-data www-data 418 sep 25 2013 index.php
-rwxr-xr-- 1 www-data www-data 17935 may 31 18:26 licencia.txt
-rwxr-xr-- 1 www-data www-data 19935 may 31 18:26 license.txt
-rwxr-xr-- 1 www-data www-data 7638 may 31 18:26 readme.html
-rwxr-xr-- 1 www-data www-data 5458 may 31 18:26 wp-activate.php
drwxr-x--- 9 www-data www-data 4096 abr 23 20:00 wp-admin
-rwxr-x--- 1 www-data www-data 364 dic 19 2015 wp-blog-header.php
-rwxr-xr-- 1 www-data www-data 1889 may 31 18:26 wp-comments-post.php
-rwxr-x--- 1 www-data www-data 3197 abr 26 22:49 wp-config.php
-rwxr-x--- 1 www-data www-data 3237 abr 23 20:00 wp-config-sample.php
drwxr-x--- 7 www-data www-data 4096 jun 14 21:00 wp-content
-rwxr-x--- 1 www-data www-data 3669 ago 20 2017 wp-cron.php
drwxr-x--- 18 www-data www-data 12288 abr 23 20:00 wp-includes
-rwxr-x--- 1 www-data www-data 2422 nov 21 2016 wp-links-opml.php
-rwxr-x--- 1 www-data www-data 3306 ago 22 2017 wp-load.php
-rwxr-xr-- 1 www-data www-data 37760 may 31 18:26 wp-login.php
-rwxr-x--- 1 www-data www-data 8048 ene 11 2017 wp-mail.php
-rwxr-x--- 1 www-data www-data 16246 oct 4 2017 wp-settings.php
-rwxr-xr-- 1 www-data www-data 30091 may 31 18:26 wp-signup.php
-rwxr-x--- 1 www-data www-data 4620 oct 24 2017 wp-trackback.php
-rwxr-x--- 1 www-data www-data 3065 ago 31 2016 xmlrpc.php
root@wordpress:/var/www/wordpress#

```

Archivos de WordPress

```

root@Pablo: ~
root@wordpress:/etc/apache2/sites-available# cat wordpress.conf
<VirtualHost *:80>
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/wordpress

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
root@wordpress:/etc/apache2/sites-available#

```

Configuración del sitio en Apache

```

root@Pablo: ~
root@wordpress:/var/www/wordpress# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 67
Server version: 10.1.26-MariaDB-0+deb9u1 Debian 9.1

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> create database wordpress;
Query OK, 1 row affected (0.00 sec)

MariaDB [(none)]> create user wordpress identified by 'wordpress';
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> grant all on wordpress.* to 'wordpress'@'localhost';
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]>

```

Creación del usuario y la base de datos en MySQL

Se copia el archivo “wp-config-sample.php” renombrándolo como “wp-config.php” y configurando los parámetros de la base de datos.

```
root@Pablo: ~
GNU nano 2.7.4                                Fichero: wp-config.php

* @package WordPress
*/

// ** Ajustes de MySQL. Solicita estos datos a tu proveedor de alojamiento web. ** //
/** El nombre de tu base de datos de WordPress */
define('DB_NAME', 'wordpress');

/** Tu nombre de usuario de MySQL */
define('DB_USER', 'wordpress');

/** Tu contraseña de MySQL */
define('DB_PASSWORD', 'wordpress');

/** Host de MySQL (es muy probable que no necesites cambiarlo) */
define('DB_HOST', 'localhost');

/** Codificación de caracteres para la base de datos. */
define('DB_CHARSET', 'utf8mb4');

/** Cotejamiento de la base de datos. No lo modifiques si tienes dudas. */
define('DB_COLLATE', '');
```

Configuración de WordPress

Se accede desde el navegador y se siguen los pasos del asistente de instalación.

Información necesaria

Por favor, debes facilitarnos los siguientes datos. No te preocupes, siempre podrás cambiar estos ajustes más tarde.

Título del sitio	<input type="text" value="Prueba"/>
Nombre de usuario	<input type="text" value="pablo"/> <small>Los nombres de usuario pueden tener únicamente caracteres alfanuméricos, espacios, guiones bajos, guiones medios, puntos y el símbolo @.</small>
Contraseña	<input type="password" value="pablo"/> <input type="button" value="Ocultar"/> <div>Muy débil</div> <p>Importante: Necesitas esta contraseña para acceder. Por favor, guárdala en un lugar seguro.</p>
Confirma la contraseña	<input checked="" type="checkbox"/> Confirma el uso de una contraseña débil.
Tu correo electrónico	<input type="text" value="pablo@localhost"/> <small>Comprueba bien tu dirección de correo electrónico antes de continuar.</small>
Visibilidad en los motores de búsqueda	<input type="checkbox"/> Disuade a los motores de búsqueda de indexar este sitio <small>Depende de los motores de búsqueda atender esta petición o no.</small>
<input type="button" value="Instalar WordPress"/>	

Asistente de instalación de WordPress

Además de montar este WordPress, se creará una base de datos y un usuario más en el servidor MySQL para los datos de la copia de seguridad de Helvia. Luego se importa la copia de seguridad con el comando "source".

```
root@Pablo: ~
root@wordpress:/var/www/wordpress# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 71
Server version: 10.1.26-MariaDB-0+deb9u1 Debian 9.1

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> create database helvia;
Query OK, 1 row affected (0.00 sec)

MariaDB [(none)]> create user helvia identified by 'helvia';
Query OK, 0 rows affected (0.01 sec)

MariaDB [(none)]> grant all on helvia.* to 'helvia'@'%';
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> use helvia;
Database changed
MariaDB [helvia]> source bitacora.sql
```

Creación de la base de datos y el usuario para la copia de seguridad de Helvia e importación de la copia de seguridad

Una vez hecho, lo siguiente es editar el archivo de configuración (config.php) del plugin para definir los parámetros de conexión a la base de datos y la configuración del sitio.

También se debe conocer la ruta del directorio donde se encuentran los archivos HTML del contenido de las publicaciones para definirla.

```
1 <?php
2 // Parámetros de configuración del Conversor Helvia/WordPress 1.0
3 // Pablo Murillo Ávila
4
5
6 // Conexión a la base de datos
7 $servidor = "localhost";
8 $usuario = "helvia";
9 $contrasena = "helvia";
10 $basededatos = "helvia";
11
12 // Ruta de los archivos de las publicaciones
13 $ruta_pub = "../pub/";
14
15 // Configuración del sitio
16 $titulo = "Bitácora del IES Averroes (Córdoba)";
17 $descripcion = "Ejemplo de descripción del sitio.";
18 $link = "http://192.168.56.2/";
19
20 // Eliminar emoticonos de Helvia de los comentarios (en WordPress se mostrarán como imágenes rotas)
21 $eliminar_emoticonos = "no";
22
23 // Limitar publicaciones para descargar (contando a partir de la publicacion_inicio, se descargarán el máximo de publicaciones
24 // definido en limite_de_publicaciones)
25 $limitar_publicaciones = "no";
26 $publicacion_inicio = "0";
27 $limite_de_publicaciones = "500";
28
29 // Archivo de salida
30 $nombre_archivo = "backup.xml";
31
```

Archivo de configuración del script (config.php)

Se lanza el script para que genere el archivo XML con toda la información.

```
root@Pablo: ~
root@wordpress:~/compartida/script# php helvia.php
[21:54:37 INFO]: Conversor Helvia/WordPress.
[21:54:37 INFO]: Conectando a la base de datos...
[21:54:37 INFO]: Conexión establecida a la base de datos.
[21:54:37 INFO]: Creando archivo de importación de WordPress...
[21:54:37 INFO]: Descargando autores...
[21:54:37 INFO]: Se han descargado 11 autores.
[21:54:37 INFO]: Descargando categorías...
[21:54:37 INFO]: Se han descargado 32 categorías.
[21:54:37 INFO]: Descargando etiquetas...
[21:54:37 INFO]: Se han descargado 1575 etiquetas.
[21:54:37 INFO]: Descargando publicaciones y comentarios...
[21:55:42 INFO]: Se han descargado 3147 publicaciones.
[21:55:42 INFO]: Archivo de de importación de WordPress 'backup.xml' creado.
[21:55:42 INFO]: El proceso ha terminado correctamente.
root@wordpress:~/compartida/script#
```

Ejecución del script

```
root@Pablo: ~
root@wordpress:~/compartida/script# ls -l
total 18108
-rwxrwxrwx 1 root root 18519811 jun 14 21:55 backup.xml
-rwxrwxrwx 1 root root      965 jun 14 21:48 config.php
-rwxrwxrwx 1 root root    12675 jun 14 10:22 helvia.php
root@wordpress:~/compartida/script#
```

Archivo backup.xml creado

Lo siguiente es instalar en el WordPress de destino el plugin “Importador de WordPress”.

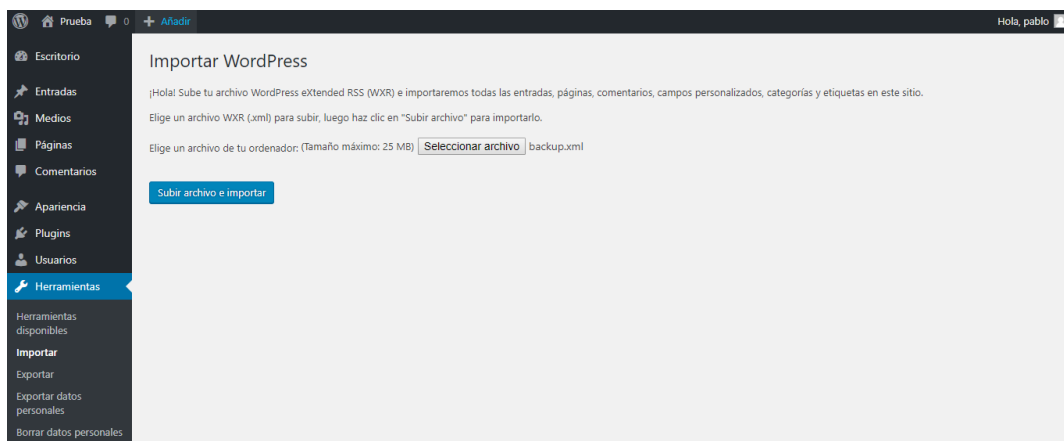


Importador de WordPress

Y, por último, una vez instalado, en el menú lateral hacemos, en “Importar”, se selecciona la opción “WordPress” y se carga el archivo que ha generado el script. Como el archivo ocupa unas 18 MB, hay que ampliar el límite de subida de archivos por defecto de PHP editando la siguiente opción en el archivo “php.ini”:

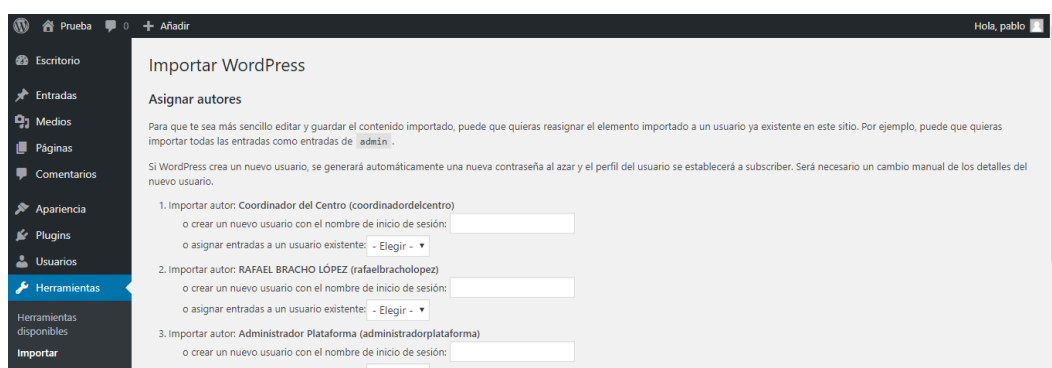
```
; Maximum allowed size for uploaded files.  
; http://php.net/upload-max-filesize  
upload_max_filesize = 25M
```

Archivo php.ini



Importador de WordPress

El asistente pedirá importar los autores, se dejan tal cual están ya que el nuevo sitio estará vacío y no habrá ninguno creado.



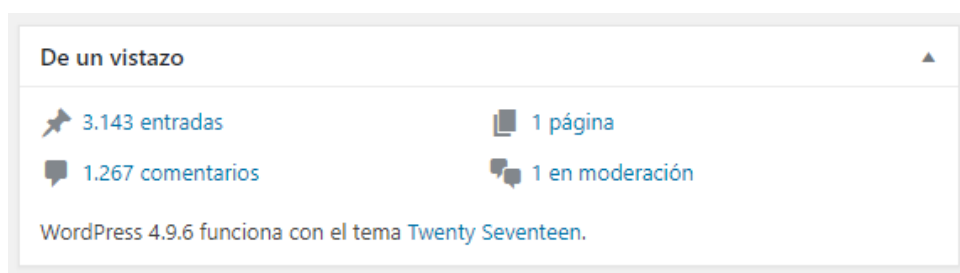
Importación de autores

Se marca la casilla de “Descarga e importa archivos adjuntos” y se hace clic en enviar para comenzar la importación.

Una vez haya terminado, ya estará toda la información importada en el nuevo sitio.



Proceso terminado correctamente



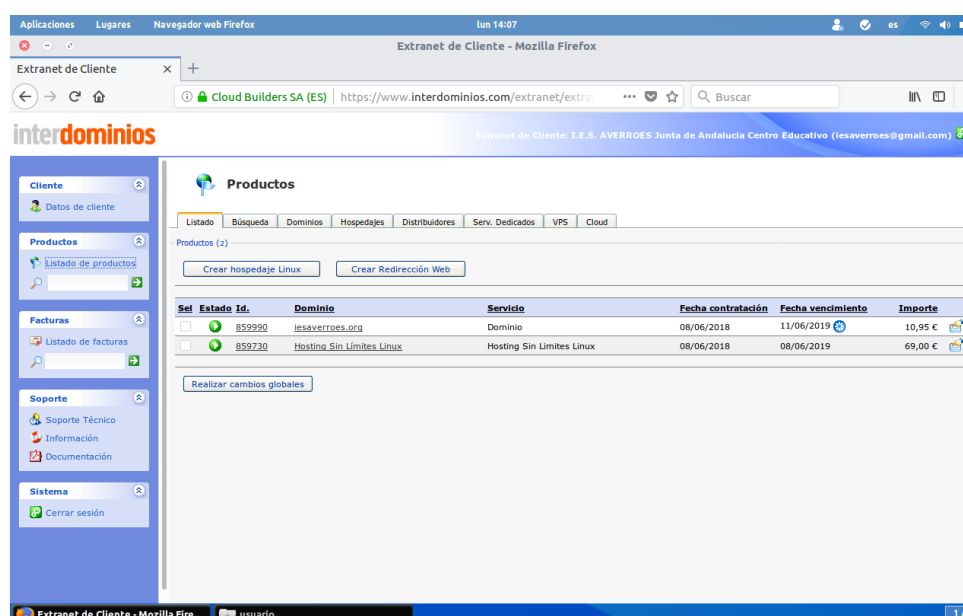
Información importada

6.2.- Entorno real

A la hora de llevar el proyecto al entorno real, la decisión que ha tomado el IES Averroes ha sido contratar un Hosting Linux ilimitado en Interdominios (alojamiento compartido).

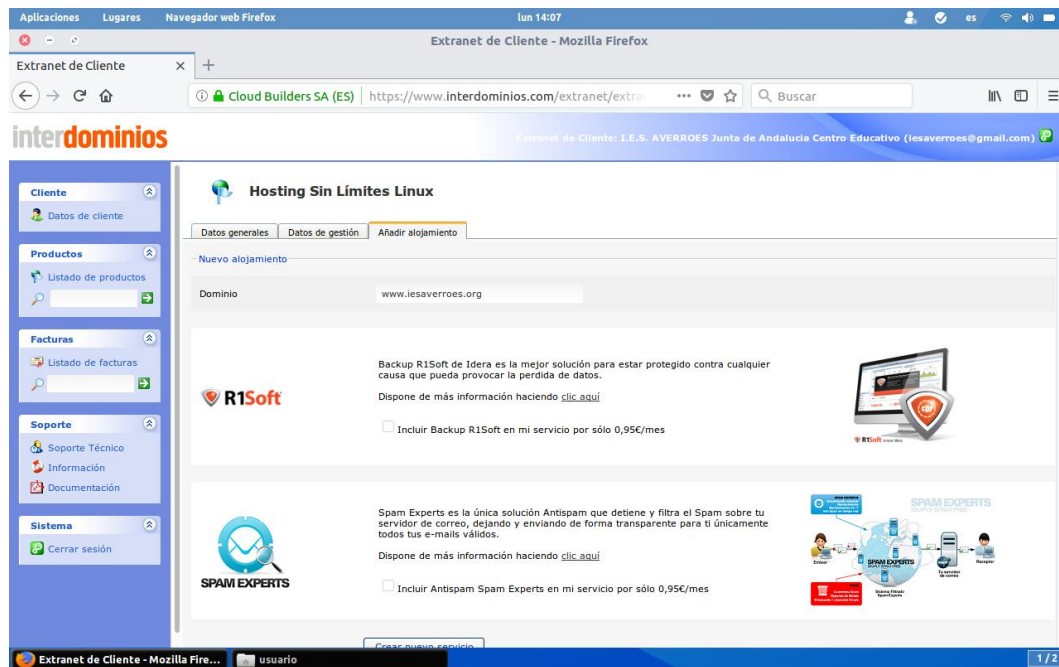
Lo primero a lo que se accede es a la extranet de cliente, un panel en el que se pueden gestionar todos los productos que se tienen contratados.

En este caso, se ha contratado un “Hosting sin límites Linux” y un “Dominio”.



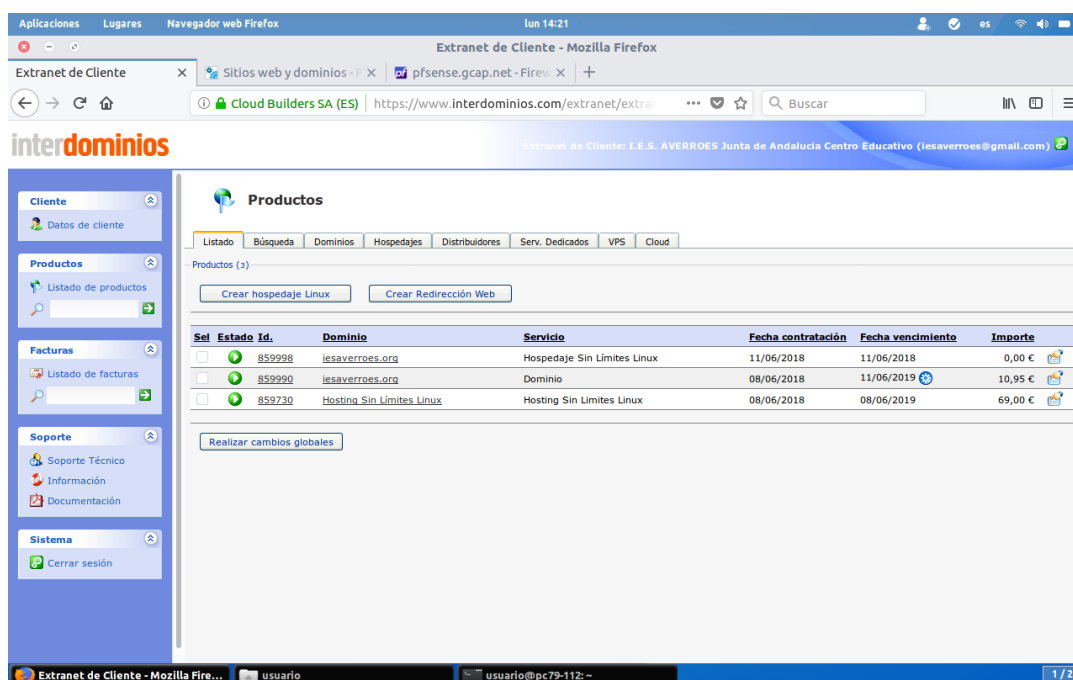
Extranet de cliente de Interdominios

Para que estos dos productos trabajen conjuntamente hay que enlazarlos, para ello, se selecciona “Hosting sin límites Linux” y “Añadir alojamiento”. Se escribe el nombre de dominio con el que queremos enlazarlos, se desmarcan las dos casillas de abajo (opcional) y se hace clic en “Crear nuevo servicio”.



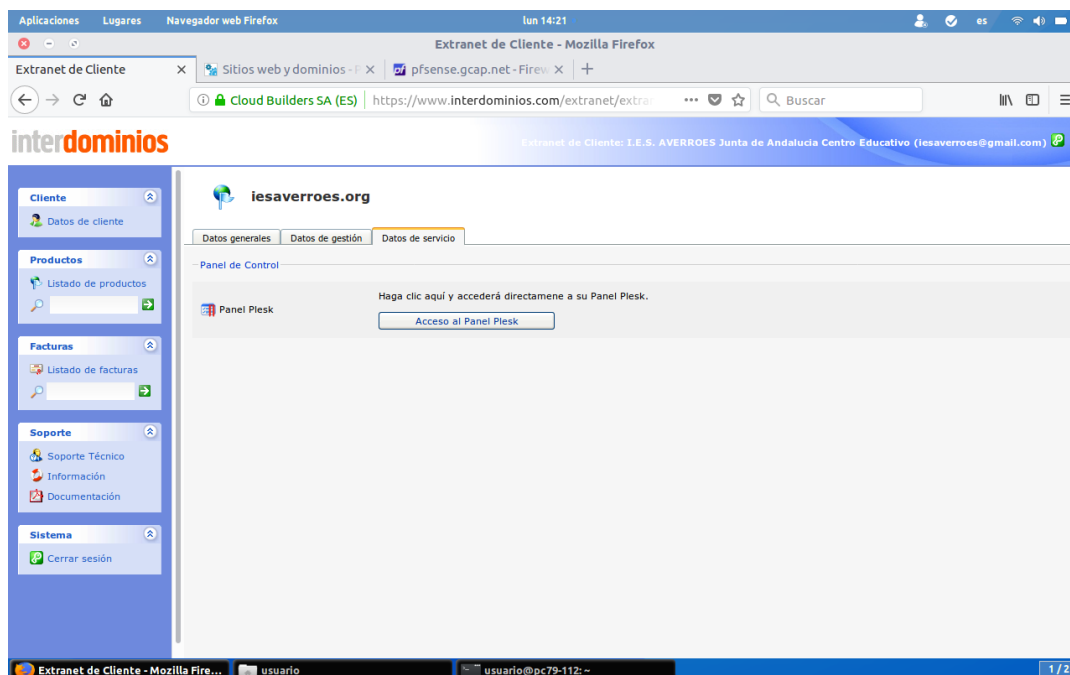
Creación del servicio de hospedaje

Una vez se haya creado, aparecerá un nuevo producto en la lista llamado “Hospedaje sin límites Linux”.

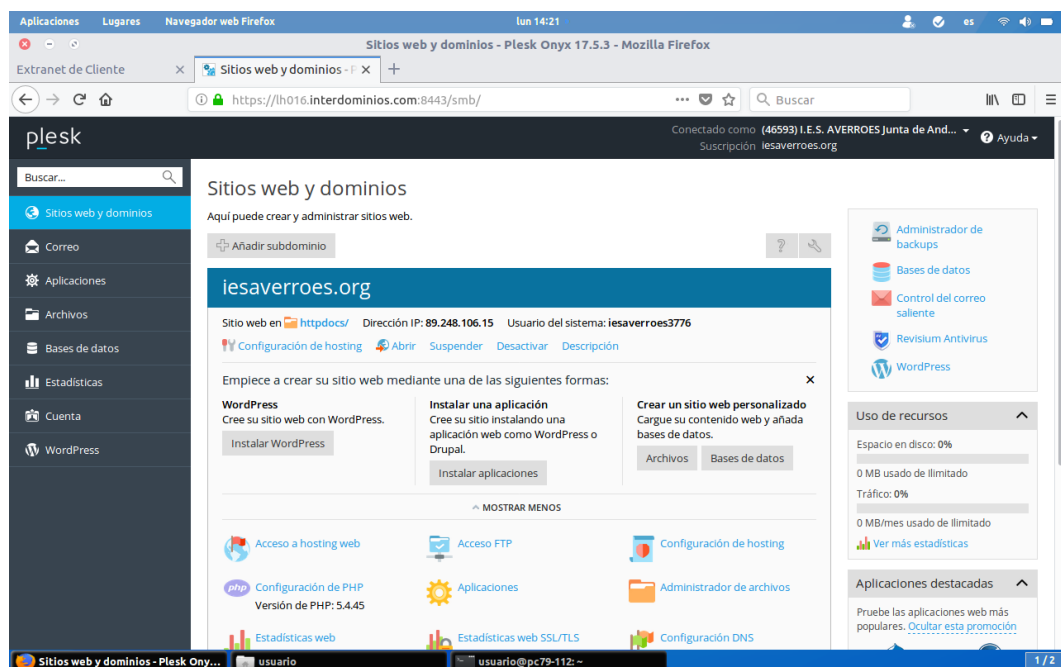


Servicio de hospedaje creado

Haciendo clic sobre este nuevo producto, y después en “Datos de servicio”, se accede al Panel Plesk.

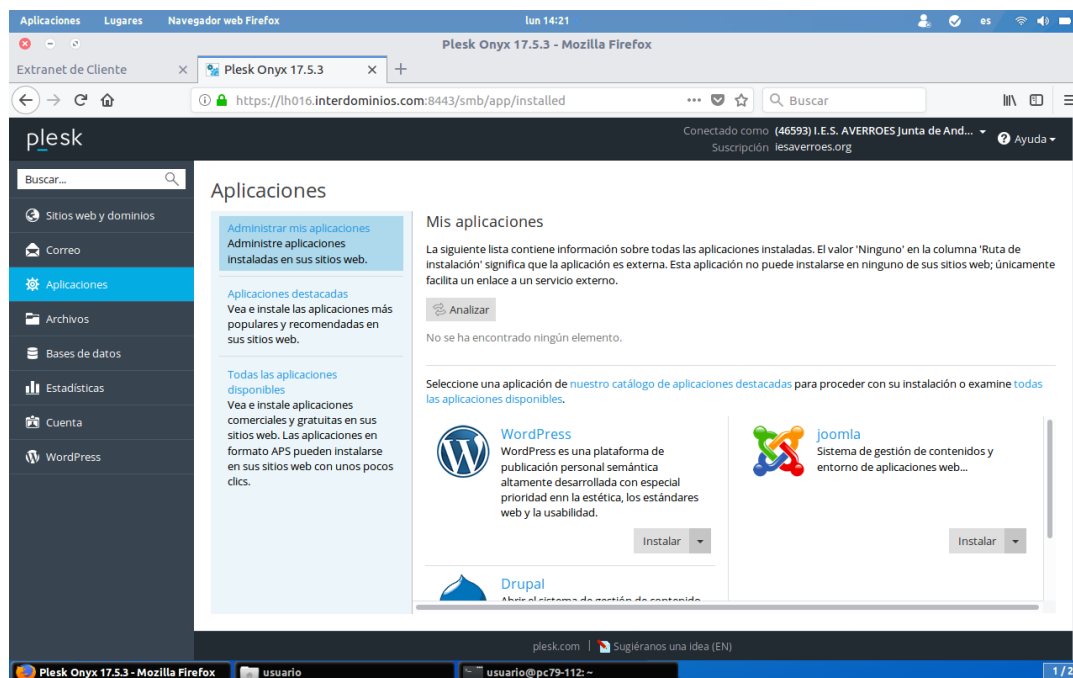


Acceso al Panel Plesk

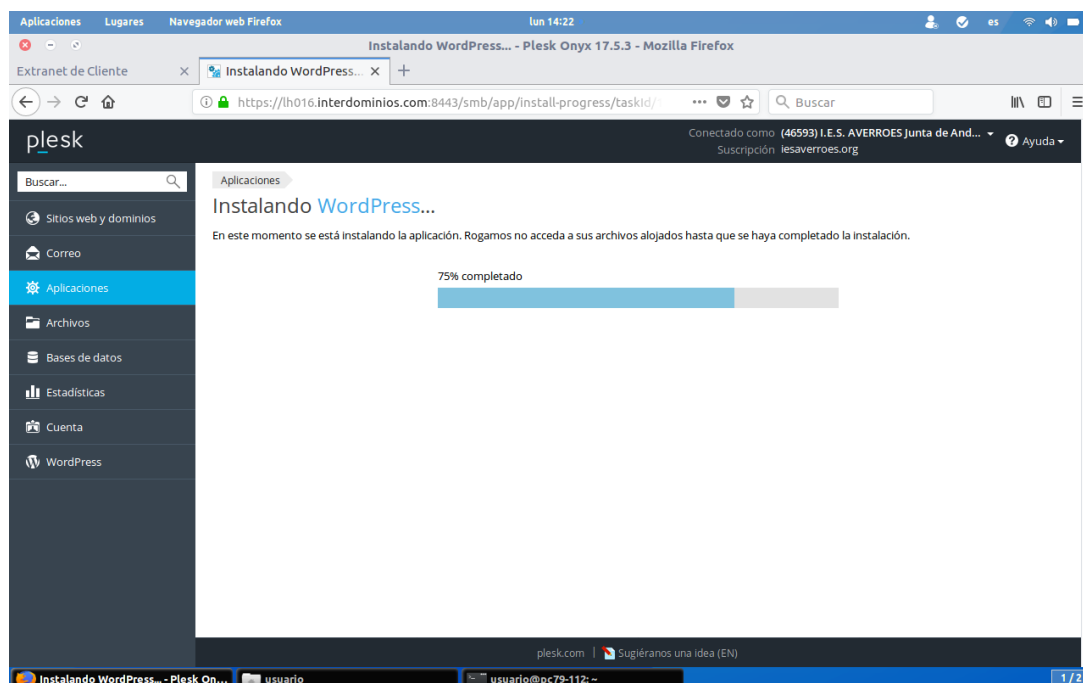


Panel Plesk

Desde este panel se puede instalar WordPress de forma automática. Para hacerlo, se selecciona en el menú de la izquierda “Aplicaciones”.



Menú de aplicaciones



Instalación automática de WordPress

Una vez se completa la instalación, desde la opción “WordPress” del menú de la izquierda, accedemos al escritorio del panel de administración de WordPress.



Menú de WordPress en Plesk

A partir de aquí, el proceso para importar el archivo generado por el script al nuevo sitio es similar que en el entorno de pruebas. Con la diferencia de que, al ser un alojamiento compartido, no es posible aumentar el límite de subida de archivos, por lo que hay que dividir el archivo en varias partes que no superen el límite que viene establecido por defecto.

Para solucionar esto, activamos el límite de publicaciones en la configuración del script. Hay que definir la entrada por la que el script va a empezar y las que va a contener el archivo que se generará a partir de esta.

```
// Limitar publicaciones para descargar (contando a partir de la publicacion_inicio, se descargarán el máximo de publicaciones  
// definido en limite_de_publicaciones)  
$limitar_publicaciones = "si";  
$publicacion_inicio = "0";  
$limite_de_publicaciones = "500";
```

Opción para limitar el número de entradas del archivo

Hay que tener en cuenta que esta opción solo limita las publicaciones, por lo que todos los archivos que se generen tendrán las mismas etiquetas, categorías y autores. A la hora de importar los archivos en WordPress solo habrá que importar los autores la primera vez, las siguientes solo habrá que asignarles el contenido en lugar de crearlos de nuevo.

También se han instalado los siguientes plugins:

- Akismet, para controlar el spam. Se necesita tener una cuenta para registrarlo o una clave API.
- JetPacks, para estadísticas de uso, SEO, personalización, etc. Se necesita una cuenta para poder usarlo.
- ShareThis, para integración con redes sociales (botones para compartir). Se necesita una cuenta para poder usarlo.
- WordPress Database Backup, para copias de seguridad.
- WP Super Cache, para caché (reducir el tiempo de carga de las páginas).

Y se le ha aplicado a la página el tema Radiate.

7.- Recursos

7.1.- Herramientas hardware

Se ha utilizado un portátil ASUS con procesador Intel Core i5-6200U y 8 GB de memoria RAM.

También se ha utilizado el hardware del servidor de Interdominios donde se ha alojado finalmente la página.

7.2.- Herramientas software

El proyecto ha sido realizado bajo Windows 10 Home, para el desarrollo del script se ha utilizado el programa Notepad++.

Se ha utilizado también VirtualBox para correr una máquina virtual Debian, en la que se ha instalado un servidor MySQL para volcar la copia de seguridad, PHP y un servidor Apache para un WordPress de pruebas. En este mismo servidor es donde también se ejecuta el script.

En el hosting compartido, se utiliza el Panel Plesk y se ha instalado WordPress desde este. En WordPress se han instalado los siguientes plugins: JetPack, Akismet, ShareThis, WordPress Database Backup y WP Super Cache. También se ha instalado el tema Radiate.

7.3.- Personal

Este proyecto ha sido desarrollado por Pablo Murillo Ávila, alumno del CFGS de Administración de Sistemas Informáticos en Red.

7.4.- Presupuesto

Para la elaboración del proyecto se han contratado los servicios de hosting y el dominio en Interdominios con una duración de 1 año. Por otro lado, se incluye la mano de obra.

Servicio	Precio
Hosting sin límites Linux	69,00€
Dominio	10,95€
Mano de obra	20€/hora × 26 horas = 520€
SUMA = 599,95€	
IVA 21% = 125,99€	
TOTAL = 725,94€	

8.- Conclusiones

8.1.- Grado de consecución de objetivos

El objetivo básico de este proyecto era la migración de la bitácora Helvia del IES Averroes a un blog desarrollado con WordPress y se ha llevado a cabo finalmente.

- El acceso a los datos y los recursos originales podrá ser a través de una copia de seguridad o mediante "Web scraping": Totalmente conseguido.
- La migración debe realizarse de forma automatizada: Totalmente conseguido.
- Las entidades a migrar estarán definidas por los elementos equivalentes entre Helvia y WordPress: Totalmente conseguido.
- El blog de destino debe implementarse con WordPress: Totalmente conseguido.

8.2.- Problemas encontrados

El principal problema que se ha presentado ha sido el del límite de subida de archivos en el alojamiento compartido. Al no poder cambiarlo, se le ha añadido al script una opción más para limitar el número de publicaciones que contendrá el archivo XML mediante 2 variables y añadiendo a la consulta las opciones "limit" y "offset".

8.3.- Futuras mejoras

Una de las posibles mejoras para esta herramienta es conseguir que la opción para limitar el número de publicaciones vaya generando distintos archivos automáticamente.

Otra sería la de exportar las imágenes de Helvia en el caso de que el importador no fuese capaz de importar las imágenes originales. En este caso no ha sido necesario por estar alojadas en servidores externos.

9.- Referencias / bibliografía

Para la realización de este proyecto se ha hecho uso de los siguientes materiales:

- <http://www.php.net/>
- <https://stackoverflow.com/questions/2955251/php-function-to-make-slug-url-string>
- [https://es.wikipedia.org/wiki/Extensible Markup Language](https://es.wikipedia.org/wiki/Extensible_Markup_Language)
- <https://es.wikipedia.org/wiki/WordPress>

10.- Anexos

- conversor_helviawordpress.zip