**DESARROLLO DE APLICACIONES WEB**

**EV.3**

**GRUPO EQDAW03**

**ÍNDICE**

1-FOND-END AND BACK-END

2-DIRECTORIOS Y USOS

3-ANÁLISIS DE USUARIOS

4-FICHA DE SERVIDOR

5-INSTALACIÓN DE WORDPRESS

6-CONFIGURACIÓN DE WORDPRESS

7-COPIAS DE SEGURIDAD

8-LICENCIAS

9-ETIQUETAS

**Blog y su documentación**

**-Conceptos front-end y back-end :**

**Front-end :** En diseño web hace referencia a lo que el usuario ve al navegar

y en el diseño del software es la parte de un sistema de software

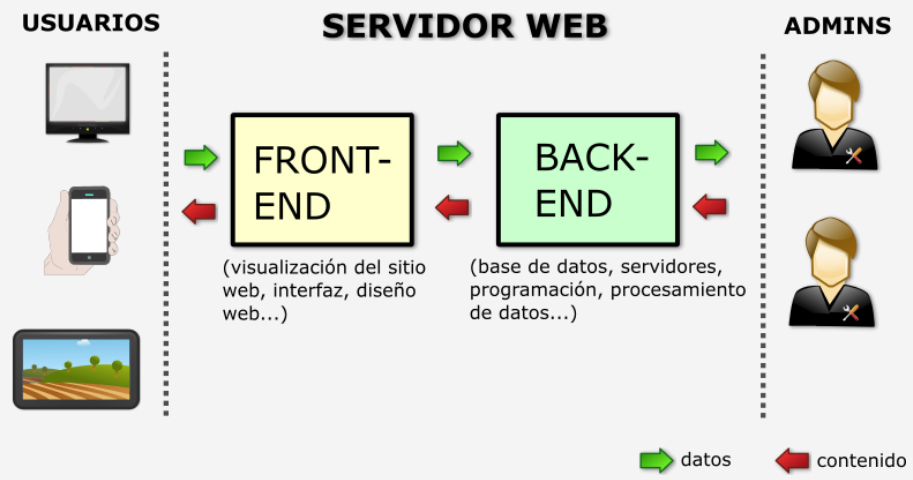
con el cual un usuario interactúa directamente(interfaz de usuario),

por ejemplo: Las entradas que ve un usuario en nuestro wordpress.

**Back-end :** En diseño web es todo el procesamiento y programas que no se ve

y en diseño de software comprende los componentes que nos manda el front-end mediante su salida,

por ejemplo: Las aplicaciones necesarias para mostrar las entradas.



**-Directorios utilizados y su uso :**

-Carpeta raíz : En esta directorio se encuentra todo el contenido de nuestro blog wordpress

aquí es donde llega nuestros visitantes al entrar al blog con la dirección

web o la ip de nuestro blog (front-end), se incluye los posts, comentarios

menús vídeos, todo el contenido que es visible para los visitantes.

En nuestro caso: /var/www/html

-wp-admin : Para poder meterse dentro de este directorio es necesario un usuario administrador. En este directorio se encuentra los directorios y archivos que

maneja los administradores del blog (back-end).

-wp-includes : En este directorio se encuentra todas las funciones de nuestro blog como

los certificados, el css, las fuentes, imágenes etc.

-wp-content : En este directorio se guardan los Plugin, temas, contenido multimedia

que tengamos en nuestro blog, el idioma de nuestro blog y un fichero

index.php que impide el acceso a los directorios de esta carpeta.

**-Análisis de usuarios necesarios :**

-Usuario SSH = admin ; contraseña = admin

-Usuario S.O. = root ; contraseña = 12345Abcde

-Usuario base de datos = root ; contraseña = 12345Abcde

-Usuario blog = root ; contraseña = 12345Abcde

**Crear Usuarios SSH:**

1-Creamos el directorio del usuario nuevo

-mkdir /home/”nombre del usuario” en este caso admin.

2-Damos permisos al directorio del usuario

-chmod 775 /home/admin

3-Creamos un grupo para los usuarios ssh

-groupadd gssh

4-Creamos el usuario, asignado su gropo el home de usuario y los comandos que puede utilizar.

-useradd -g gssh -d /home/admin -s /bin/bash admin

5-Creamos la contraseña para el usuario

-passwd admin

6-Modificar /etc/ssh/sshd\_config para que podamos conectarnos mediante ssh

-nano /etc/ssh/sshd\_config

Añadimos al final del fichero (AllowUsers admin), esto nos indica que usuarios pueden conectarse mediante ssh.

7-Reiniciamos ssh

-/etc/init.d/ssh restart

En este caso como nuestros clientes no nos han proporcionado que usuarios debemos crear

hemos decidido crear el mismo usuario para todo ya que es mas fácil recordar dicho usuario y su contraseña.

**-Ficha del servidor**

|  |  |
| --- | --- |
| Nombre | srv03 |
| Procesador | 1 núcleo |
| Ram | 512 MB |
| Disco Duro | 20 GB |
| Sistema Operativo | Ubuntu-16.04-standard\_16.04-1\_amd64 |
| Usuarios S.O. | root 12345Abcde |
| Datos protocolos TCP/IP | ip:10.1.3.1/16 get:10.1.0.1 puerto:22 puerto:80 |
| Datos sistemas instalados | Apache2 2.4.5, php7.0.27-0+deb9ul , mariadb-server 10.1.26, mariadb-client 10.1.26 |

**-Instalación de WordPress :**

1-apt-get update : actualización para los índices de los repositorios que utilicemos.

2-apt-get install apache2 : instalar el servidor apache HTTP.

Comprobar: systemctl status apache2.

3-apt-get install php7.0 libapache-mod-php7.0 : instalar la ultima versión del interprete PHP

y módulos para Apache.

Comprobar: Crear .php en /var/www/html con código php (<?php phpinfo(); ?)

Acceder por host/archivo.php en el navegador.

4-apt-get install php7.0-mysql mariadb-server mariadb-client : instalación del la base de

datos mariadb cliente y servidor y el módulo PHP requerido para acceder a la base de

de datos desde el servidor Apache HTTP.

5-systemctl start mariadb : arrancar el servicio mariadb.

6-mysql\_secure\_installation : ejecutar script para modificar algunos aspectos.

-Set root password? Yes : establece una contraseña para root y garantiza que nadie

pueda iniciar sesión sin autorización adecuada.

-Remove anonymous users? Yes : remover el usuario anónimo, para que no se puede.

meter cualquiera a la base de datos.

-Disallow root login remotely? Yes : esto asegura que no se pueda adivinar la contraseña de root de la red.

-Remove test database and access to it? Yyes : eliminar la base de datos ya creada

llamada 'prueba' para que nadie tenga acceso a ella.

-Reload privilege tables now? Yes : actualiza las tablas de privilegios asegura que

todos los cambios realizados hasta el momento tomará efecto.

7-Todavía se puede iniciar sesión en mariadb desde localhost con la cuenta raíz sin que se

solicite contraseña. Para evitar posibles problemas de seguridad iniciamos sesión en la

base de datos con el usuario raíz y ejecutamos lo siguiente:

-use mysql; : para usar una base de datos mysql.

-update user set plugin='' where User='root'; actualizar el usuario root quitando los

datos de la fila Plugin.

-flush privileges; : para aplicar la modificación de los usuarios modificados.

-quit : salir de la base de datos.

8-Reiniciar para aplicar los cambios realizados hasta ahora.

-systemctl restart apache2.service mariadb.service : reiniciar

-systemctl enable apache2.service mariadb.service : activar

9-Comprobar que Apache y mariadb han abierto los puertos y que están en estado de escucha.

-netstat -tlp | egrep ‘http|mysql’

10-Instalar wordpress.

-Primero nos conectamos a la base de datos (mysql -u root -p)

-Creamos la base de datos para wordpress (CREATE DATABASE wordpress)

en este caso la hemos llamado wordpress.

-Dar al usuario root todos los privilegios de la base de datos

(GRANT ALL PRIVILEGES ON wordpress.\* TO 'wordpress-user'@'localhost' IDENTIFIED BY 'wordpress-pass';).

-Actualizamos los privilegios (FLUSH PRIVILEGES;)

-Descargamos la última versión de wordpress (wget <http://wordpress.org/latest.tar.gz>)

-Extraemos los datos dentro de /*var/*www/html/

(tar xfz latest.tar.gz /var/www/html)

-Si queremos borramos el index.html de /var/www/html

(rm /var/www/html/index.html).

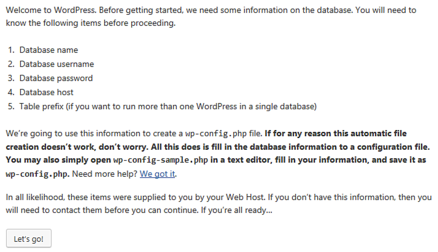
Modificamos los permisos de las carpetas de Apache para el grupo www-data,

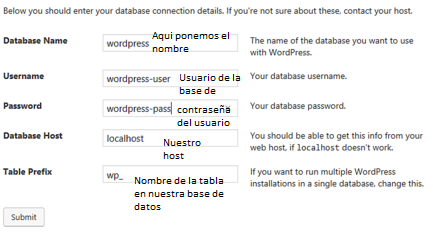
esto permitirá al servidor web crear o modificar archivos y directorios bajo esta ruta.

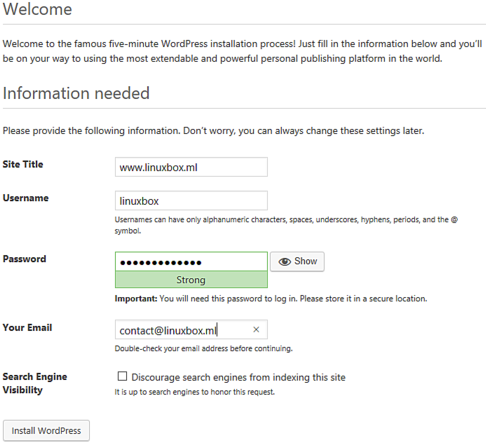
(chmod –R 775 /var/www/html/)

(chgrp –R www-data /var/www/html/)

11-Con todo esto ya podemos instalar nuestro wordpress para ello tenemos que ir a un

navegador y poner nuestra ip así accederemos al panel de instalación de wordpress.





**-Configuración de WordPress : 10.1.3.1/wp-admin/**

-Tema utilizado : Twenty Seventeen theme

-Nombre del blog : Eqdaw03

-Widgets añadidos : Licencia y calendario

-Logo e icono del blog.

-Instalación de plugin para hacer backup de nuestro blog.

**-Copias de seguridad del blog :**

**-**Para poder hacer las copias de seguridad de nuestro blog usaremos el Plugin (Updraft Plus)

con este Plugin podemos hacer copias de seguridad en la nube como

(Dropbox, Google Drive).

Este Plugin puede hacer copia de seguridad de los archivos y de la base de datos de wordpress.

Puedes programar en qué fecha quieres hacer la copia de seguridad.

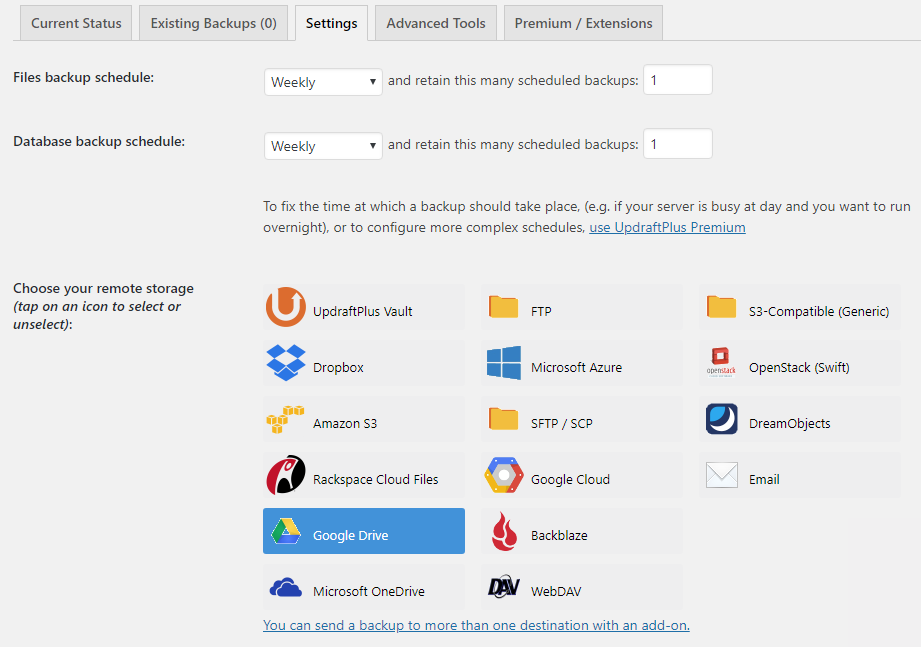
En este caso tendremos una copia de los datos cada semana ya que cada semana

modificaremos la entrada del blog añadiendo contenido al blog y las guardaremos en

google drive.

Porque elegir Updraftplus :

* + Soporta el programado manual y automático cada 4,8,12 horas ; días semanas, meses e incluso por una fecha específica.
  + Es multiplataforma, funciona en cualquier S.O que sea capaz de instalar wordpress en su sistema.
  + Tiene más opciones de almacenamiento más remotas que cualquier otro Plugin de copia de seguridad de WordPress
  + Toda la base de datos está encriptada
  + Puedes importar o restaurar cualquier backup hecho por otro Plugin de backups.
  + Tiene un foro para cualquier duda o soporte que necesites.



**-Licencia utilizada para el blog :**

**-**Reconocimiento-NoComercial-CompartirIgual (CC BY-NC-SA)

Esta licencia permite a otros entremezclar, ajustar y construir a partir de su obra con fines no comerciales, siempre y cuando le reconozcan la autoría y sus nuevas creaciones estén bajo una licencia con los mismos términos.

**-Etiquetas utilizadas:**

[EGIBIDE](http://10.1.3.1/tag/egibide/) : Encontramos todos los trabajos relacionados con el centro.

EQDAW03 : Encontramos todos nuestros posts relacionados con el reto.

ESPORTS : Encontramos diferentes tipos de artículos relacionados con los E-sports.

RETO : Encontramos todos nuestras entradas con contenido relacionado con el reto.