

GRADO EN INGENIERÍA ELECTRÓNICA  
INDUSTRIAL Y AUTOMÁTICA  
**TRABAJO FIN DE GRADO**

***RED INALÁMBRICA DE SENSORES  
POR RADIOFRECUENCIA***

**ANEXO I - SERVIDOR LAMP**

**Alumno/Alumna:** Dieguez Martín Alexander

**Director/Directora (1):** Oleagordia Aguirre Iñigo Javier

**Curso:**2018-2019

**Fecha:**<XXXX, día, mes, año>

## **ÍNDICE**

1. INTRODUCCIÓN.....	4
2. RESUMEN DE FUNCIONAMIENTO.....	5
3. INSTALACIÓN DE LAMP.....	5
3.1 SISTEMA OPERATIVO.....	5
3.2 SERVIDOR WEB DE LA INFRAESTRUCTURA LAMP.....	6
3.3 PHP.....	7
3.4 SERVIDOR DE BASES DE DATOS .....	8
4. CONFIGURACIÓN DEL SERVIDOR.....	10



## **ÍNDICE DE FIGURAS**

Figura 1.1 Esquema de los procesos LAMP.....	4
Figura 3.1 Comprobación instalación Apache2.....	6
Figura 3.2 Comprobación instalación PHP.....	7
Figura 3.3 Página de acceso al servidor.....	9
Figura 3.4 Cambio del nombre del host en página no-ip.....	10
Figura 3.5 Configuración de los parámetros del sensor DHT11.....	10
Figura 3.6 Valores de temperatura y humedad en el servidor.....	11



## 1. INTRODUCCIÓN

Existen dos tipos de páginas web: estáticas y dinámicas. Las webs estáticas están alojadas en un servidor y creadas en lenguaje HTML. Están pensadas principalmente para mostrar una información permanente a la que es posible acceder en cualquier momento. El contenido de las páginas web dinámicas, por su parte, se genera de forma activa cada vez que se realiza una solicitud. Estas últimas resultan muy útiles cuando la temática de la web requiere actualizaciones regulares.

Las páginas web dinámicas requieren una infraestructura especial, que consta de un sistema operativo, un servidor web, un gestor de bases de datos y uno o más lenguajes de programación; facilidades que ofrece por ejemplo, el servidor LAMP.

LAMP es una plataforma de código abierto para el desarrollo WEB que usa Linux como sistema operativo, Apache como servidor Web, MySQL como base de datos relacional y PHP como lenguaje de scripts orientado a objetos. Los componentes individuales se acumulan unos sobre otros, por lo que esta plataforma también recibe el nombre de LAMP (del inglés 'apilar').

Es una de las opciones preferidas gracias a su bajo coste y su alta disponibilidad. Además, sus componentes individuales pueden ser reemplazados fácilmente por aquellos con las mismas funciones.

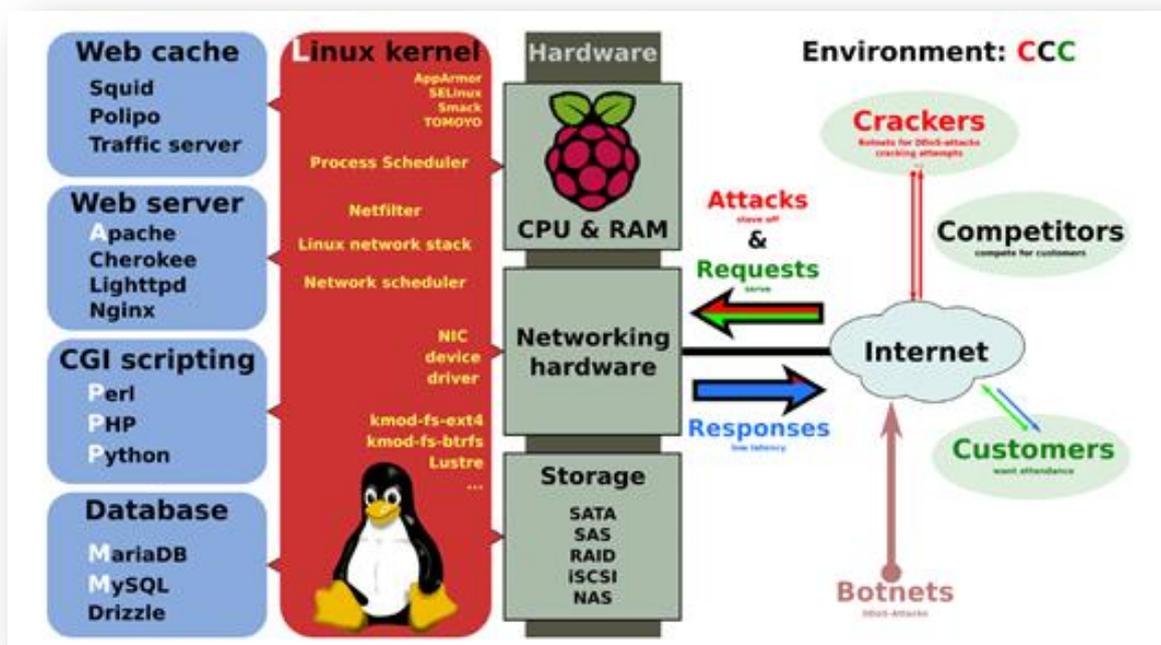


Figura 1.1 Esquema de los procesos LAMP

## 2. RESUMEN DE FUNCIONAMIENTO

Lamp está compuesto por cuatro componentes: Linux, Apache, MySQL y PHP. Su funcionamiento es muy simple. Linux sirve como sistema operativo base para ejecutar el servidor web Apache. Este último no puede interpretar contenidos dinámicos, pero es aquí donde PHP entra a ejercer sus funciones de programación del lado del servidor. El proceso funciona entonces de la siguiente manera: Apache le envía un código fuente al intérprete PHP, incluyendo la información correspondiente sobre las acciones del visitante de la web, y permite el acceso a la base de datos MySQL. El resultado es devuelto a Apache y este se muestra finalmente en el navegador web del visitante.

## 3. INSTALACIÓN

### 3.1. SISTEMA OPERATIVO

El sistema operativo utilizado para la instalación del servidor web es el Raspbian. Se trata de una distribución del sistema operativo GNU/Linux y por lo tanto libre, basado en Debian para la placa Raspberry pi, orientado a la enseñanza de informática.

Destaca también por su menú 'raspi-config' que permite configurar el sistema operativo sin tener que modificar archivos de configuración manualmente. Entre sus funciones, permite expandir la partición root para que ocupe toda la tarjeta de memoria, configurar el teclado, aplicar overclock, etc...

Para poder realizar la instalación del sistema operativo en la tarjeta Raspberry, se necesita volcar la imagen en una micro SD. Para ello primero se tiene que realizar un formateo en FAT32 mediante el programa '*SD Card Formatter*', y una vez vaciado la tarjeta en el formato correspondiente, mediante el programa '*Win32DiskImager*' se vuelva la imagen en la micro SD.

Una vez finalizado el proceso anterior, la micro SD ya está lista para poder instalar el sistema operativo. Para ello simplemente se introduce la micro en la Raspberry y automáticamente efectúa todo el proceso de instalación. Mediante un cable hdmi conectado a un monitor es posible seguir todo este proceso.

Tras el proceso de instalación se configura el escritorio en el idioma correspondiente, la red wi-fi, el teclado, etc... y ya es posible empezar a instalar los paquetes correspondientes del servidor mediante la pantalla de comandos.

Para poder dar por terminado este proceso es necesario realizar una serie de actualizaciones para poder verificar que todo el sistema se encuentra actualizado. Para ello se utilizan los siguientes códigos:

#### Actualizar Raspberry Pi

- `sudo rpi-update`



## Actualizar el sistema

- `sudo apt update`
- `sudo apt upgrade`

### 3.2. SERVIDOR WEB DE LA INFRAESTRUCTURA LAMP

Lo primero que se va a instalar es el servidor web. Este servirá para infinidad de procesos, y no solo para la infraestructura LAMP. Simplemente con colocar una página web estática en el sitio adecuado se podrá ver desde nuestra red local. O incluso verlo desde cualquier sitio utilizando OpenVPN. Para instalar el servidor web, se tienen que ejecutar las siguientes ordenes.

- `sudo apt install apache2`

Una vez completada la instalación, se puede comprobar si todo el proceso a salido correctamente, abriendo el navegador de internet y escribiendo la dirección IP de la Raspberry PI, que en este caso sería: `http://192.168.0.24`.

Si todo funciona correctamente aparecerá en el navegador una página como la que se muestra a continuación.

**Apache2 Debian Default Page**

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

**Configuration Overview**

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```

/etc/apache2/
-- apache2.conf
... parts.conf
-- mods-enabled
|   |-- *.load
|   |-- *.conf
-- conf-enabled
|   |-- *.conf
-- sites-enabled
|   |-- *.conf

```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or virtual host configurations, respectively.
- They are activated by symlinking available configuration files from their respective `*-available/` counterparts. These should be managed by using our helpers `a2enmod`, `a2dismod`, `a2ensite`, `a2dissite`, and `a2enconf`, `a2disconf`. See their respective man pages for detailed information.
- The binary is called `apache2`. Due to the use of environment variables, in the default configuration, `apache2` needs to be started/stopped with `/etc/init.d/apache2` or `apache2ctl`. **Calling `/usr/bin/apache2` directly will not work** with the default configuration.

**Document Roots**

By default, Debian does not allow access through the web browser to *any* file apart of those located in `/var/www`, **public\_html** directories (when enabled) and `/usr/share` (for web applications). If your site is using a web document root located elsewhere (such as in `/srv`) you may need to whitelist your document root directory in `/etc/apache2/apache2.conf`.

The default Debian document root is `/var/www/html`. You can make your own virtual hosts under `/var/www`. This is different to previous releases which provides better security out of the box.

**Reporting Problems**

Please use the `reportbug` tool to report bugs in the Apache2 package with Debian. However, check **existing bug reports** before reporting a new bug.

Please report bugs specific to modules (such as PHP and others) to respective packages, not to the web server itself.

Figura 3.1 Comprobación instalación Apache2

### 3.3 PHP

El lenguaje de programación PHP es uno de los más extendidos para el desarrollo de páginas web. La ventaja de utilizar PHP para el desarrollo de páginas web es que permite crear páginas web dinámicas, es decir, que se generan cuando u usuario visita la página.

Por defecto Raspbian dispone de PHP versión 5, sin embargo, muchos servicios de hosting ya ofrecen la versión 7, que es más rápida y eficiente. Es por ello que se realizó la instalación de la versión 7, con las siguientes ordenes.

- `sudo apt-get install php7.0`

Una vez completado el proceso, es necesario editar un fichero.

- `sudo nano/var/www/html/info.php`

Dentro de este fichero se coloca lo siguiente:

```
<?php
phpinfo()
?>
```

Para finalizar se realiza un reboot del sistema. La comprobación de que todo lo realizado anteriormente funciona correctamente se realiza de forma similar como con la instalación de apache2, ya que se debe introducir `http://192.168.0.24/index.php` en el navegador web. Si todo funciona bien, el navegador mostrará lo siguiente.

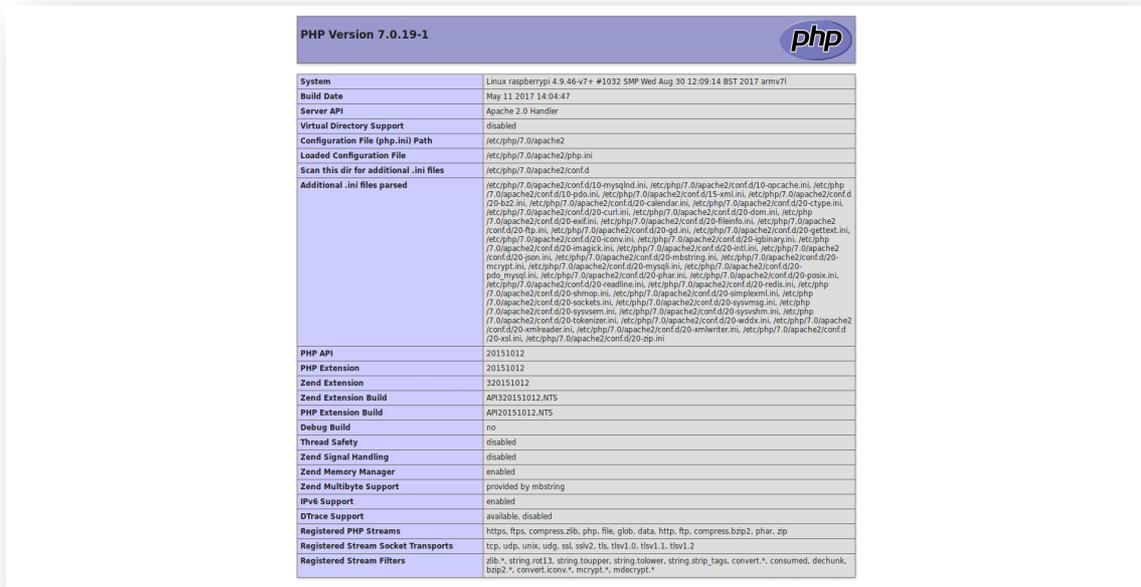


Figura 3.2 Comprobación instalación PHP



### 3.4 SERVIDOR DE BASES DE DATOS

Respecto al servidor de bases de datos, se escogerá MariaDB en lugar de MySQL, que era el estándar hasta hace unos años. En 2008 MySQL fue adquirida por un tercero y cambió su licencia. MariaDB tiene licencia GPL v2.

Sin embargo, no es solo la licencia lo único que cambia uno de otro, sino que existen algunas diferencias notables entre ambos servidores. Entre ellas MariaDB ha añadido tablas a nivel de sistema para mejorar la optimización de los datos. Por otro lado, MariaDB tiene un rendimiento superior a MySQL.

El único inconveniente que se puede tener utilizando MariaDB es de compatibilidad, a pesar de que ambos servidores presentan un elevado grado de compatibilidad. Como el sistema creado se sabe que no va a tener problemas de compatibilidad, se procede a la instalación mediante las siguientes ordenes.

- *sudo apt install mariadb-server*

Durante el proceso de instalación se preguntará por la contraseña a utilizar. Se trata de una contraseña importante, ya que posteriormente se necesitará para instalar otras aplicaciones web. A continuación se necesitan realizar los siguientes cambios.

¿Change the root password?

*No*

¿Remove anonymous users?

*No*

¿Disallow root login remotely?

*No*

¿Remove test database and access to it?

*No*

¿Reload privilege tables now?

*Yes*

Una vez realizado estos pequeños cambios se termina la instalación aportando los privilegios necesarios al usuario para poder acceder al servidor web. Para ello se realizan las siguientes ordenes.

- *mysql -u root -p*
- *GRANTALL PRIVILEGES on \*.\* to 'root'@'localhost' IDENTIFIED BY '\*\*\*\*\*'*
- *FLUSH PRIVILEGES*
- *\q*
- *service mysql stop*
- *service mysql start*



Para finalizar se comprueba que todo funciona correctamente, y que se tiene acceso al servidor mediante el usuario y pw. configurados anteriormente. Para ello se introduce lo siguiente en el navegador web: 192.168.0.24/phpmyadmin/ y mostrará la siguiente página.



phpMyAdmin

**Bienvenido a phpMyAdmin**

Idioma - Language

Español - Spanish

Iniciar sesión

Usuario:

Contraseña:

Continuar

Figura 3.3 Página de acceso al servidor



## 4. CONFIGURACIÓN DEL SERVIDOR

Se puede realizar el cambio en el nombre del dominio de la página para que en vez de conectarnos a la página mediante el número de ip de la raspberry pi, se pueda poner cualquier otro nombre. Mediante la plataforma 'www.no-ip.com' es posible cambiar el dominio. En nuestro caso se ha llamado 'tfgredinalambrica.ddns.net'. Es por ello que ahora únicamente poniendo este nombre en el directorio de URL se podrá acceder al servidor.

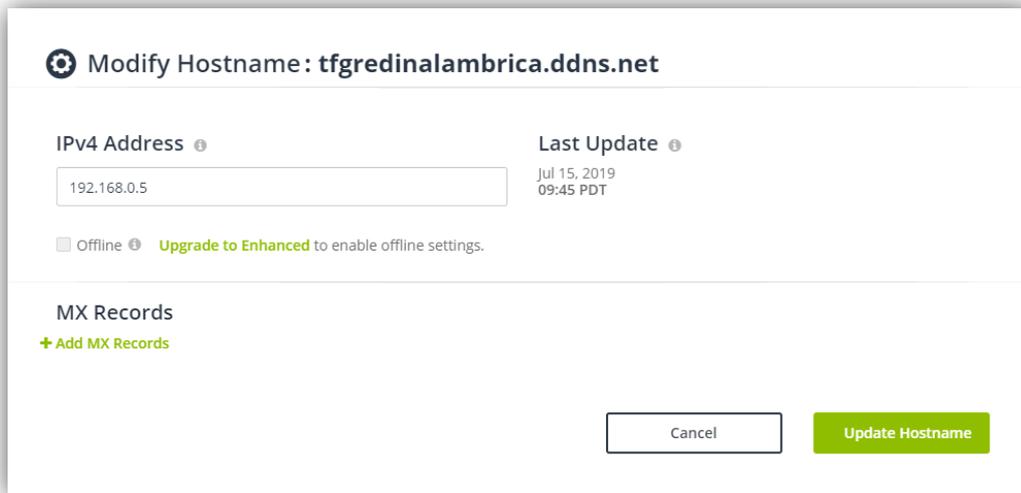


Figura 3.4 Cambio del nombre del host en página no-ip.

Teniendo el servidor funcionando correctamente se acaba la instalación del servidor LAMP configurando el *phpmyadmin* con las entradas a utilizar del DHT11, concretamente de temperatura y humedad.

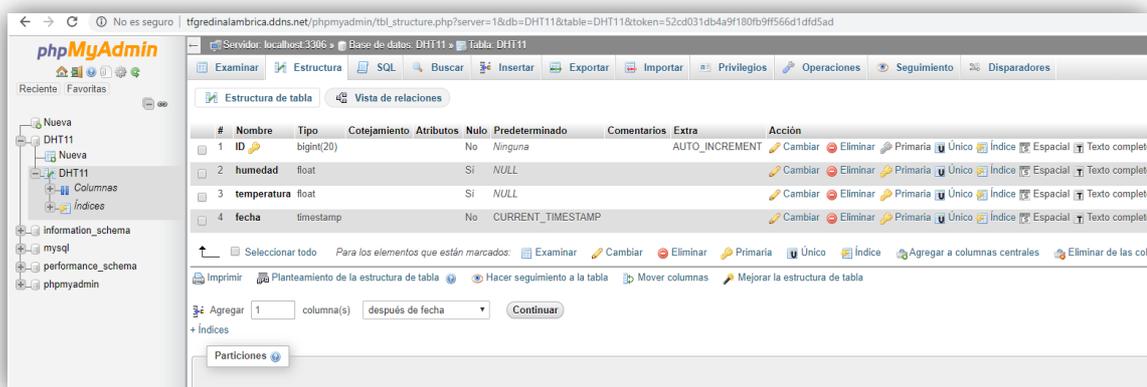
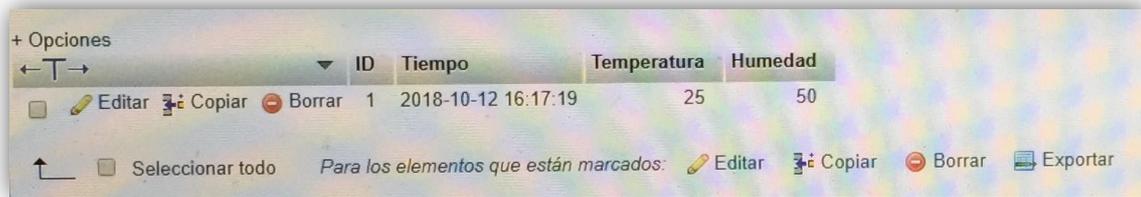


Figura3.5 Configuración de los parámetros del sensor DHT11



The screenshot shows a web interface with a table containing one row of data. The table has four columns: ID, Tiempo, Temperatura, and Humedad. The row contains the values 1, 2018-10-12 16:17:19, 25, and 50. Above the table is a header with '+ Opciones' and a search icon. Below the table is a toolbar with 'Seleccionar todo' and a context menu for the selected row with options: 'Editar', 'Copiar', 'Borrar', and 'Exportar'.

ID	Tiempo	Temperatura	Humedad
1	2018-10-12 16:17:19	25	50

*Figura 3.6 Valore de temperatura y humedad en el servidor.*

Por último solo faltaría configurar el programa de Arduino del módulo ESP8266 con la dirección y contraseña de la red wi-fi en la que estén conectados ambos dispositivos y colocar la dirección ip de la raspberry pi para poder enviar los datos al servidor de forma correcta.

Hay que tener especial precaución a la hora de poner en marcha el servidor, ya que según la red wi-fi en la que estén conectados los dispositivos. La dirección ip de la raspberry pi va cambiando, es decir, que se tienen que modificar toda la programación donde intervenga la dirección ip para poder vincular ambos dispositivos. Con ello compilar el script con el nombre y contraseña de la red wi-fi en el módulo ESP8266.

