

Instalación de Proyecto Laravel

Felipe Sebastián Canales Saavedra

Javier Andrés Reyes Gonzalez

Felipe Aracena Carvacho

Pof. Sebastián Salazar

Ingeniería de Software

5 de agosto de 2015

Índice

A. Introducción	2
B. Requerimientos	3
B.1. Requerimientos de software	3
B.2. Requerimientos de hardware	3
C. Instalación y configuración	4
C.1. Instalación de Apache Servidor en Linux	4
C.2. Instalación de PostgreSQL	5
C.3. Instalación de php	5
C.3.1. Dar preferencia al archivo index.php	5
C.4. Instalación de Laravel	6
C.4.1. Instalación de composer	6
C.5. Instalación de un proyecto desde un repositorio	6
C.5.1. Clonar el repositorio	6
C.5.2. Instalar dependencias necesarias	6
C.6. Creación de Virtual Hosts	7

Resumen

Pequeño manual de instalación que plantea resolver la duda de como instalar un proyecto de Laravel, en Linux de manera genérica y sencilla

A. Introducción

El presente manual, es parte de los conocimientos teóricos del ramo de Ingeniería de Software, con el fin de guiarnos durante el proceso de instalación y correcta ejecución de un proyecto de Laravel 5.0.*, agregar a esto la instalación de PostgreSQL y la creación de la base de datos.

Este manual, intentará (en lo posible) de entregar instrucciones claras y concisas para llevar a cabo su fin. Facilitando así el proceso para usuarios noveles.

Es pertinente mencionar, que esta aplicación web, se está perfeccionando constantemente, por lo que este manual del mismo modo se irá actualizando.

B. Requerimientos

B.1. Requerimientos de software

Los requerimientos óptimos de software, consisten en lo siguiente:

- Motor DB: PostgreSQL 9.x
- Framework PHP: Laravel 5.0.x
- IDE: Atom 1.0.2
- Entorno de pruebas: Vagrant + Homestead
- Sistema Operativo: Ubuntu 14.04 x64 Server LTS

B.2. Requerimientos de hardware

Los requerimientos de hardware, consisten en lo siguiente:

- Procesador Intel a 1Ghz
- Mínimo 2048 Mb de memoria RAM
- 20GB de disco duro, incluida el área de intercambio
- Tarjeta gráfica integrada

C. Instalación y configuración

C.1. Instalación de Apache Servidor en Linux

A continuación se detallan los pasos de instalación del ambiente LAMP (Linux, Apache, MySQL, PHP) en Linux.

Todos los pasos realizados posteriormente, serán con privilegios de administrador. Si usted no los posee, es conveniente que se ponga en contacto con el administrador de sus sistema para otorgarle dichos privilegios.

En primer lugar, se instalará apache, facilmente es posible realizar esto desde la terminal o linea de comandos que proporciona linux:

En la linea de comandos, colocamos lo siguiente

```
sudo apt-get update
sudo apt-get install apache2
```

Al terminar la instalación, nuestro servidor estará instalado, para verificar que todo esté en regla, y no haya fallado nada, basta con probar alguna de estas dos direcciones en nuestro navegador:

```
http://127.0.0.1
http://localhost
```

Si no se cuenta con entorno gráfico, bastará con realizar un ping para ver si existe conexión:

```
ping 127.0.0.1
ping localhost
```

Para poder configurar el servidor, nos serviremos de esta paleta de comandos:

```
sudo service apache2 [start | restart | stop | status]
```

C.2. Instalación de PostgreSQL

En este caso, se obviará la instalación de MySQL y se dará prioridad a PostgreSQL.

La instalación, de PostgreSQL ocurre de manera sencilla, en la terminal basta con ingresar el siguiente comando y tendremos PostgreSQL instalado:

```
sudo apt-get install postgresql postgresql-client
```

Posterior a la instalación, se hace necesario configurar la contraseña para el usuario postgres:

```
sudo su postgres
```

Una vez dentro:

```
psql
```

Dentro del cliente de PostgreSQL debes ejecutar la siguiente sentencia SQL y presiona la tecla Enter:

```
ALTER USER Postgres WITH PASSWORD '<password>';
```

Para iniciar sesión, para poder crear otros usuarios y/o bases de datos:

```
psql -U postgres -W -h localhost
```

C.3. Instalación de php

Como ya es usual se usará la terminal o línea de comandos para realizar la instalación:

```
sudo apt-get install php5 libapache2-mod-php5 php5-mcrypt
```

C.3.1. Dar preferencia al archivo index.php

Normalmente, si un usuario solicita un directorio al servidor, lo primero que hará Apache es buscar un archivo de nombre index.html. Tenemos que decirle a Apache que estamos interesados en que primero busque los archivos index.php.

```
sudo gedit /etc/apache2/mods-enabled/dir.conf
```

Lo que resultaría en algo como esto:

```
DirectoryIndex index.html index.cgi index.pl index.php  
index.xhtml index.htm
```

El truco, mover a la primera posición el nombre *index.php*, resultando en lo que sigue

```
DirectoryIndex index.php index.html index.cgi index.  
plindex.xhtml index.htm
```

Y si nuestro servidor estaba ejecutandose, basta reiniciarlo y listo:

```
sudo service apache2 restart
```

En el caso de estar en un entorno Windows, es altamente recomendado instalar algún paquete que encapsule cada uno los componentes mencionados anteriormente, como por ejemplo XAMPP

C.4. Instalación de Laravel

Antes de empezar es necesario tener instalado curl en nuestro sistema:

```
sudo apt-get install curl
```

C.4.1. Instalación de composer

Procedemos a instalar composer de la siguiente manera:

```
curl -sS https://getcomposer.org/installer | php
```

Luego para poder ejecutarlo de manera global (dentro de cualquier carpeta de nuestro sistema), movemos composer.phar a la ruta *usr/local/bin*, esto se hace con el siguiente comando:

```
sudo mv ~/composer.phar /usr/local/bin/composer
```

En este momento ya podemos digitar el comando: `composer create-project laravel/laravel --prefer-dist 5.0.` para poder crear un proyecto vacío*

C.5. Instalación de un proyecto desde un repositorio

Antes de comenzar, se asumirá que ya se cuenta con la instalación y configuración de git, en caso contrario, dirigirse a : Instalación de git en Linux y para configurar: Documentación

C.5.1. Clonar el repositorio

Iniciamos clonando con git el repositorio que contiene el proyecto Laravel que se desea instalar:

```
git clone https://github.com/fcanalesS/ProyectoWeb.git
```

C.5.2. Instalar dependencias necesarias

Se procede a instalar las dependencias necesarias para asegurar la instalación de los paquetes restantes:

```
composer install
```

De esta forma nuestro proyecto parcial, ahora está completo y listo para trabajar en él.

C.6. Creación de Virtual Hosts

Realizar este proceso se hace necesario para poder probar nuestros proyectos, con URLs amigables y así evitarnos diversos problemas.

- Lo primero, es dirigirnos a esta carpeta:

```
cd /etc/apache2/sites-available/
```

- Realizar un respaldo de ésta:

```
sudo cp 000-default.conf proyecto.local.conf
```

- Ahora procedemos a editar nuestro archivo para que quede de la siguiente forma:

```
<VirtualHost *:80>
    ServerAdmin admin@localhost
    ServerName universidad.dev #url del proyecto
    DocumentRoot /var/www/html/ProyectoWeb/public #
        Ruta al proyecto
    <Directory /var/www/html/ProyectoWeb/public> #
        Ruta al proyecto
        Options Indexes FollowSymLinks MultiViews
        AllowOverride All
        Order allow, deny
        allow from all
    </Directory>

    ErrorLog ${APACHELOG_DIR}/error.log
    CustomLog ${APACHELOG_DIR}/access.log combined
</VirtualHost>
```

Procedemos a indicarle a Apache que utilice nuestro nuevo documento: *sudo a2ensite proyecto.local.conf*

- Ahora debemos activar nuestro archivo de configuración en el archivo hosts:

```
sudo nano /etc/hosts
```

Y agregamos la siguiente línea:

```
127.0.0.1 localhost
127.0.1.1 homestead homestead
127.0.0.1 universidad.dev #<Agregamos esta línea

# The following lines are desirable for IPv6 capable
hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Obligatoriamente se debe activar el módulo rewrite con el siguiente comando: *sudo a2enmod rewrite*

- Para finalizar reiniciamos nuestro servidor:

```
sudo service apache2 restart
```


Una vez finalizado todo el proceso, colocamos universidad.dev en la barra de direcciones y debería mostrar lo siguiente:

