

# Starting Laravel project via Apache web server

## Installing apache

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

## Installing composer

```
$ sudo apt-get update
$ sudo apt-get install curl php5-cli git
$ curl -sS https://getcomposer.org/installer | sudo php --
--install-dir=/usr/local/bin --filename=composer
```

## Adding php module to apache

Note: By default, php module is not added to apache. The list of installed modules is available in `/etc/apache2/mods-available`. In previous versions of apache, the modules were added via main apache configuration file, but now, each module has its own conf and/or load file in `/etc/apache2/mods-available` and the files in this directory are included in apache main conf as you can see in `/etc/apache2/apache2.conf`:

```
...
IncludeOptional mods-enabled/*.load
IncludeOptional mods-enabled/*.conf
...
```

In order to add php module to apache:

```
$ sudo apt-get update
```

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

## Git repo configurations (optional)

**If you are going to use a laravel project from a git repo and not making the project by hand, there is some stuff to do:**

### Adding missed vendor directory

By default, composer adds `vendor` directory to `.gitignore` file so that it is not committed/pushed to the repository. This is because `vendor` directory contains dependencies of project and every one can install the dependencies by hand. In order to add the dependencies and make `vendor` directory reappear:

```
$ cd /path/to/laravel/project  
$ composer update
```

Wait some moment so that composer finishes its job.

### Setting permissions

The `storage` directory of the project needs global write access. So run the following command to achieve this:

```
$ cd /path/to/laravel/project  
$ chmod -R o+w storage
```

### Generating key via artisan (Why??)

```
$ cd /path/to/laravel/project  
$ cp .env.example .env  
$ php artisan key:generate
```

Make sure the command finishes successfully.

# Adding an apache virtual host

We need to add an apache virtual host and make it point to our laravel project. To do so:

```
$ vim /etc/apache2/sites-available/mysite.conf
```

Add the following lines to the file:

```
<VirtualHost *:80>
    DocumentRoot /path/to/laravel/project/public
    ServerName mysite.local # Choose your own name
    ErrorLog /path/to/apache/logfiles/error.log # Choose your own
path
    CustomLog /path/to/apache/logfiles/access.log combined #
Choose your own path
    <Directory /path/to/laravel/project/public>
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>
</VirtualHost>
```

Then restart apache service:

```
$ sudo service apache2 restart
```

Enable site:

```
$ sudo a2ensite mysite.conf
```

# Adding server to hosts file

```
$ vim /etc/hosts
```

Add the following line:

```
127.0.0.1 mysite.local # This should be the same as ServerName in
previous section
```

Now, open your browser and type in address bar your site name (e.g. <http://mysite.local>). You should see your project homepage.