# Install Laravel

# General

## Tạo Model

## Tạo View

## Tạo Controler

# Template Laravel

## Create HTML CSS

## Custom Layer

# Command Creation

## Create Migrate

## Create Model

## Create Controler

## Start Server

## Start Migrate

## Update Composer

# Migration

## Kết nối bảng trong Laravel

+ Migrate: Tạo mối liên kết trong Migrate

Trong Post migrate

|  |
| --- |
| $table -> integer('idUser') -> unsigned() -> default(0);  $table->foreign('idUser')  ->references('id')->on('users')  ->onDelete('cascade'); |

+ Tạo kết nối trong cở Model

|  |
| --- |
| // returns the instance of the user who is author of that post  **public function** getAuthor()  {  **return** $this->belongsTo('App\User','idUser');  } |

+ Lấy dữ liệu trong bảng

|  |
| --- |
| <p class=*"lead"*>  by <a href=*"index.php"*><?php **echo** $post -> getAuthor -> name ?></a>  </p> |

# Advand

## Create Utltil – helper in Laravel

C1.

### Custom Classes in Laravel 5, the Easy Way

This answer is applicable to general custom classes within Laravel. For a more Blade-specific answer, see [**Custom Blade Directives in Laravel 5**](http://stackoverflow.com/questions/28290332/best-practices-for-custom-helpers-on-laravel-5#36888067).

**Step 1:** Create your Helpers (or other custom class) file and give it a matching namespace. Write your class and method:

<?php // Code within app\Helpers\Helper.php

namespace App\Helpers;

class Helper

{

public static function shout(string $string)

{

return strtoupper($string);

}

}

**Step 2:** Create an alias:

<?php // Code within config/app.php

'aliases' => [

...

'Helper' => App\Helpers\Helper::class,

...

**Step 3:** Use it in your Blade template:

<!-- Code within resources/views/template.blade.php -->

{!! Helper::shout('this is how to use autoloading correctly!!') !!}

**Extra Credit:** Use this class anywhere in your Laravel app:

<?php // Code within app/Http/Controllers/SomeController.php

namespace App\Http\Controllers;

use Helper;

class SomeController extends Controller

{

public function \_\_construct()

{

Helper::shout('now i\'m using my helper class in a controller!!');

}

...

### C2: Create file in upload

Create a helpers.php file in your app folder and load it up with composer:

"autoload": {

"classmap": [

...

],

"psr-4": {

"App\\": "app/"

},

"files": [

"app/helpers.php" // <---- ADD THIS

]

},

After adding that to your composer.json file, run the following command:

composer dump-autoload

Call file: {{ \App\Helper::function\_name($value) }}

## Đường dẫn thân thiện

<https://github.com/cviebrock/eloquent-sluggable>

<http://kungfuphp.com/laravel-framework-5/bai-18-toi-uu-cau-truc-duong-dan-urls-trong-laravel-5.html>

### Add dependency

First, you'll need to install the package via Composer:

$ composer require cviebrock/eloquent-sluggable

Then, update config/app.php by adding an entry for the service provider.

'providers' => [

// ...

Cviebrock\EloquentSluggable\ServiceProvider::class,

];

Finally, from the command line again, publish the default configuration file:

php artisan vendor:publish --provider="Cviebrock\EloquentSluggable\ServiceProvider"

### Updating your Eloquent Models