

# Web Dev etc

... my software dev blog  
(Vue, Laravel, PHP, Javascript, Typescript).



## Laravel Naming Conventions

Laravel is used by many thousands of developers every day to create all kinds of websites and applications. But fortunately, there are some very commonly used naming conventions that are followed by many developers when naming their Laravel project's variables, methods and functions. Here is my overview of the best naming conventions for Laravel.

### Laravel Naming Conventions Table of contents

- **Naming Controllers**
- **Naming database tables in Laravel**
  1. **Pivot tables**
  2. **Table columns names**
  3. **Primary Key**
  4. **Foreign Keys**

- **Variables**
- **Naming Conventions for models**
  1. **Naming Models in Laravel**
  2. **Model properties**
  3. **Model Methods**
  4. **Relationships**
    1. **hasOne or belongsTo relationship (one to many)**
    2. **hasMany, belongsToMany, hasManyThrough (one to many)**
    3. **Polymorphic relationships**
- **Controllers**
- **Method naming conventions in controllers**
- **Traits**
- **Blade view files**

(Please let me know in a comment if you disagree with anything here)

## Naming Controllers

Controllers should be in singular case, no spacing between words, and end with "Controller".

Also, each word should be capitalised (i.e. BlogController, not blogcontroller).



For example: BlogController , AuthController , UserController .



Bad examples: UsersController (because it is in plural), Users (because it is missing the Controller suffix).

---

# Naming database tables in Laravel

DB tables should be in lower case, with underscores to separate words (snake\_case), and should be in plural form.



For example: `posts` , `project_tasks` , `uploaded_images` .

---



Bad examples: `all_posts` , `Posts` , `post` , `blogPosts`

---

## Pivot tables

Pivot tables should be all lower case, each model in alphabetical order, separated by an underscore (snake\_case).



For example: `post_user` , `task_user` etc.

---



Bad examples: `users_posts` , `UsersPosts` .

---

## Table columns names

Table column names should be in lower case, and snake\_case (underscores between words). You shouldn't reference the table name.



For example: `post_body` , `id` , `created_at` .

---



Bad examples: `blog_post_created_at` , `forum_thread_title` ,  
`threadTitle` .

## Primary Key

This should normally be `id` .

## Foreign Keys

Foreign keys should be the model name (singular), with `'_id'` appended to it (assuming the PK in the other table is `'id'`).



For example: `comment_id` , `user_id` .

## Variables

Normal variables should typically be in `camelCase`, with the first character lower case.



For example: `$users = ...` , `$bannedUsers = ...` .



Bad examples: `$all_banned_users = ...` , `$Users=...` .

If the variable contains an array or collection of multiple items then the variable name should be in plural. Otherwise, it should be in singular form.



For example: `$users = User::all();` (as this will be a collection of multiple User objects), but `$user = User::first();` (as this is just

one object)

---

# Naming Conventions for models

## Naming Models in Laravel

A model should be in singular, no spacing between words, and capitalised.

---



For example: `User` (`\App\User` or `\App\Models\User`, etc),  
`ForumThread` , `Comment` .

---



Bad examples: `Users` , `ForumPosts` , `blogpost` , `blog_post` ,  
`Blog_posts` .

---

Generally, your models should be able to automatically work out what database table it should use from the following method:

```
<?php
/**
 * Get the table associated with the model.
 *
 * @return string
 */
public function getTable()
{
    if (! isset($this->table)) {
        return str_replace(
            '\\', '', Str::snake(Str::plural(class_basename($this)))
        );
    }
    return $this->table;
}
```

But of course you can just set `$this->table` in your model. See the section below on table naming conventions.

I recommend that you create models and migrations at the same time by running `php artisan make:model -m ForumPost`. This will auto-generate the migration file (in this case, for a DB table name of 'forum\_posts').

## Model properties

These should be lower case, snake\_case. They should also follow the same conventions as the table column names.



For example: `$this->updated_at` , `$this->title` .

---



Bad examples: `$this->UpdatedAt` , `$this->blogTitle` .

---

## Model Methods

Methods in your models in Laravel projects, like all methods in your Laravel projects, should be camelCase but the first character lower case.



For example: `public function get()` , `public function getAll()` .

---



Bad examples: `public function GetPosts()` , `public function get_posts()` .

---

## Relationships

## **hasOne or belongsTo relationship (one to many)**

These should be singular form and follow the same naming conventions of normal model methods (camelCase, but with the first letter lower case)



For example: `public function postAuthor()` , `public function phone()` .

---

## **hasMany, belongsToMany, hasManyThrough (one to many)**

These should be the same as the one to many naming conventions, however, it should be in plural.



For example: `public function comments()` , `public function roles()` .

---

## **Polymorphic relationships**

These can be a bit awkward to get the naming correct.

Ideally, you want to be able to have a method such as this:

```
<?php
public function category()
{
    return $this->morphMany('App\Category', 'categoryable');
}
```

And Laravel will by default assume that there is a `categoryable_id` and `categoryable_type`.

But you can use the other optional parameters for `morphMany` ( `public function morphMany($related, $name, $type = null, $id = null, $localKey = null)` ) to change the defaults.

## Controllers

### Method naming conventions in controllers

These should follow the same rules as model methods. I.e. camelCase (first character lowercase).

In addition, for normal CRUD operations, they should use one of the following method names.

VERB	URI	TYPICAL METHOD NAME	ROUTE NAME
GET	/photos	index()	photos.index
GET	/photos/create	create()	photos.create
POST	/photos	store()	photos.store
GET	/photos/{photo}	show()	photos.show
GET	/photos/{photo}/edit	edit()	photos.edit
PUT/PATCH	/photos/{photo}	update()	photos.update
DELETE	/photos/{photo}	destroy()	photos.destroy

## Traits

Traits should be be adjective words.



For example: `Notifiable` , `Dispatchable` , etc.

---



# Blade view files

Blade files should be in lower case, snake\_case (underscore between words).



For example: `all.blade.php` , `all_posts.blade.php` , etc

---

## RELATED POSTS

[Laravel naming conventions](#)

[Laravel features you might not have used!](#)

[Vue Test Utils advanced features](#)

[How to set up Vue 3 \(guide\)](#)

## ABOUT ME

---

### WEB DEV ETC

I'm a uk based software developer, working mostly with Laravel (PHP) and Vue (JS/Typescript).

This is my small blog about software development related topics.

I mostly use this blog to have somewhere to refer to when I forget how to set something up.

---

## Comments



Laravel Naming Conventions

## What do you think?

101 Responses



Upvote



Funny



Love



Surprised



Angry



Sad

[Comments](#)

[Community](#)



[Privacy Policy](#)



[Login](#) ▾



[Recommend](#) 2



[Tweet](#)



[Share](#)

[Sort by Best](#) ▾

Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS [?](#)

Name

[@njoguamo](#) • 2 years ago

Another great article.

*(To comment without signing up to Disqus, fill out your name (under "or sign up with Disqus") and press "post as a guest" in the button that appears!)*

## About me

This is my site where I post some software development posts/content.

Welcome to my software development blog and website. There are a selection of posts about various topics here - mostly focusing on Laravel, Vue, PHP, JS or Typescript.

I have a few public Laravel Packages here too.

## Follow me elsewhere



## Popular Posts

- ☆ [How to use Vue 3](#)
- ☆ [Vue-Test-Utils Guide: Things You Might Not Know About Vue-Test-Utils](#)
- ☆ [Laravel features you may not know about](#)
- ☆ [Suspending in Bash: Quickly switch between running processes in bash](#)
- ☆ [The Repository Pattern in PHP \(and Laravel\)](#)
- ☆ [Laravel Naming Conventions](#)
- ☆ [Vue 3 composition API examples](#)

## Blog Post Categories

- 📁 [General](#)
- 📁 [WebDev Programming](#)
- 📁 [PHP](#)
- 📁 [PHP Design Patterns](#)
- 📁 [MySQL / MariaDB](#)
- 📁 [Laravel](#)
- 📁 [HTML](#)
- 📁 [CSS](#)
- 📁 [Javascript](#)
- 📁 [Javascript Testing](#)
- 📁 [Vue](#)
- 📁 [Vim Tricks and Tips](#)
- 📁 [Server Stuff](#)
- 📁 [Security and netsec](#)
- 📁 [Bash and Linux CLI commands](#)
- 📁 [Testing](#)

---

[Home](#)[Sitemap](#)