

### Laravel là gì?

Là framework dùng để xây dựng nhanh ứng dụng – framework (bộ khung)

Mã nguồn mở (miễn phí)

### Ưu điểm của Laravel

Là framework phổ biến nhất

Cộng đồng sử dụng đông đảo

Tính bảo mật cao

Kiến trúc MVC

Tài liệu phong phú

Sử dụng tính năng mới nhất của ngôn ngữ PHP

Dễ sử dụng

Tích hợp dịch vụ gửi/nhận mail: SwiftMailer, SMTP, Mailgun, SparkPost, Mandrill, Amazon SES

Hỗ trợ cache backend để tăng tốc độ website như Memcached và Redis

Tích hợp công cụ dòng lệnh artisan

### Nhược điểm của laravel

Nâng cấp phiên bản dễ phát sinh lỗi

Trở nên cồng kềnh khi làm dự án quá nhỏ

Xem phiên bản của Laravel framework: `php artisan -version`

Artisan là giao diện dòng lệnh (command-line) dùng để tự động hóa các chức năng như sau:

Tạo template cho model, controller, event

Thao tác trên database (migrate, seed)

Sinh key mã hóa để xác thực

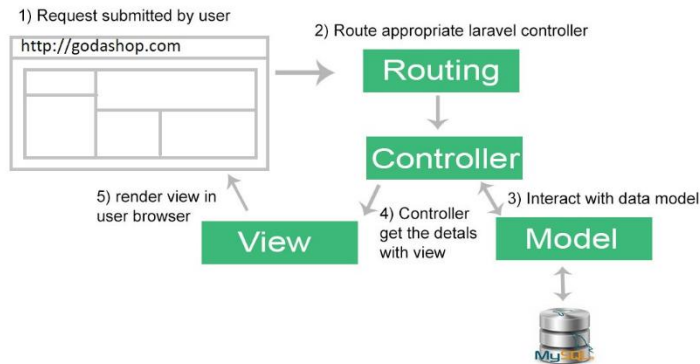
Xóa cache

Hiển thị danh sách lệnh của artisan: `php artisan list`

Xem danh sách release tại: <https://github.com/laravel/laravel/releases>

Hiển thị danh sách các routes: `php artisan route:list`

### Mô hình MVC trong laravel



<https://thayloc.com>

---

controller nằm trong đường dẫn `App\Http\Controllers`

model nằm trong đường dẫn `App\Models`

view nằm trong đường dẫn `resource\views`

### Cài laravel với version 8.5.9

```
composer create-project --prefer-dist  
laravel/laravel=8.5.9 qlsv
```

(Source code khoảng 48M)

### Cài đặt virtualhost

`laravel.qlsv.com =>` trỏ về thư mục `qlsv/public`

### Kiểm tra cài đặt

`laravel.qlsv.com`

### Tạo controller

Tạo `app\Http\Controllers/StudentController.php` bằng cách:

```
php artisan make:controller StudentController --resource --  
model=Student
```

Chọn “no” nếu không muốn tạo ra model `Student`

## Tạo route

Khai báo route resource trong **routes/web.php**

```
use App\Http\Controllers\StudentController;

Route::resource('students', StudentController::class);

//Các route sẽ được sinh ra như sau:
```

```
$ php artisan route:list
```

Domain	Method	URI	Name	Action	Middleware
	GET HEAD	/		Closure	web
	GET HEAD	api/user		Closure	api auth:api
	GET HEAD	students	students.index	App\Http\Controllers\StudentController@index	web
	POST	students	students.store	App\Http\Controllers\StudentController@store	web
	GET HEAD	students/create	students.create	App\Http\Controllers\StudentController@create	web
	GET HEAD	students/{student}	students.show	App\Http\Controllers\StudentController@show	web
	PUT PATCH	students/{student}	students.update	App\Http\Controllers\StudentController@update	web
	DELETE	students/{student}	students.destroy	App\Http\Controllers\StudentController@destroy	web
	GET HEAD	students/{student}/edit	students.edit	App\Http\Controllers\StudentController@edit	web

## Cập nhật APP\_NAME và APP\_URL trong .env

### Config database

.env:

DB\_DATABASE=laravel-demo-qlsv

config/database.php:

'charset' => 'utf8mb4',

'collation' => 'utf8mb4\_general\_ci',

## Tạo model

```
php artisan make:model Student
```

Hiển thị collection trong StudentController@index:

```
$students = Student::all(); //trả về collection
```

```
//tương tự như: $students = Student::where("name", "LIKE", "%")->get();
```

## Tạo view

Vào [Layouts Using Template Inheritance](#)

Tạo view có tên giống tên action cho dễ quản lý

Tạo thư mục resources/views/student

Tạo view index.blade.php

Dùng hàm asset: `{{ asset('...') }}` để chỉ định css, js, ...

Trong public, tạo css, js, vendor để chứa đựng css, js, và thư viện của bên thứ 3

Trong hàm index của StudentController thì return view("student.index");

Nghĩa là: render template engine index.blade.php nằm trong thư mục student (tính từ resources/views)

thayloc.com