



# LARAVEL FRAMEWORK

## BÀI 1: GIỚI THIỆU VỀ LARAVEL PHẦN 1

- ❑ Cài đặt Composer và tạo project Laravel
- ❑ Sử dụng cơ bản Laravel: Controller, View, Routing



# Laravel



## ❑ Phần 1

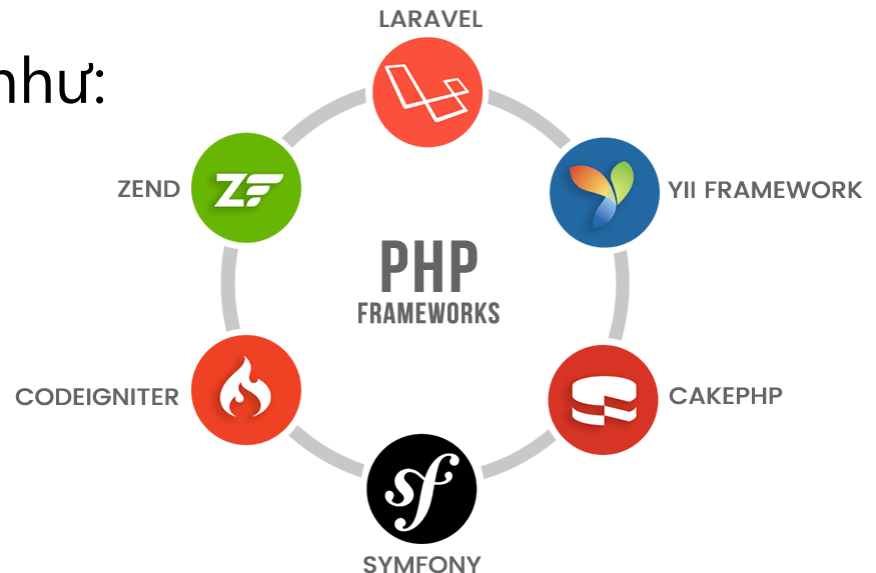
- ❖ Giới thiệu PHP framework và Laravel
- ❖ Các tính năng nổi bật của Laravel
- ❖ Giới thiệu, cài đặt và sử dụng composer
- ❖ Tạo project Laravel và thực thi project Laravel
- ❖ Cài đặt và sử dụng gói Laravel Installer
- ❖ Cấu trúc file và folder trong Laravel project

## ❑ Phần 2

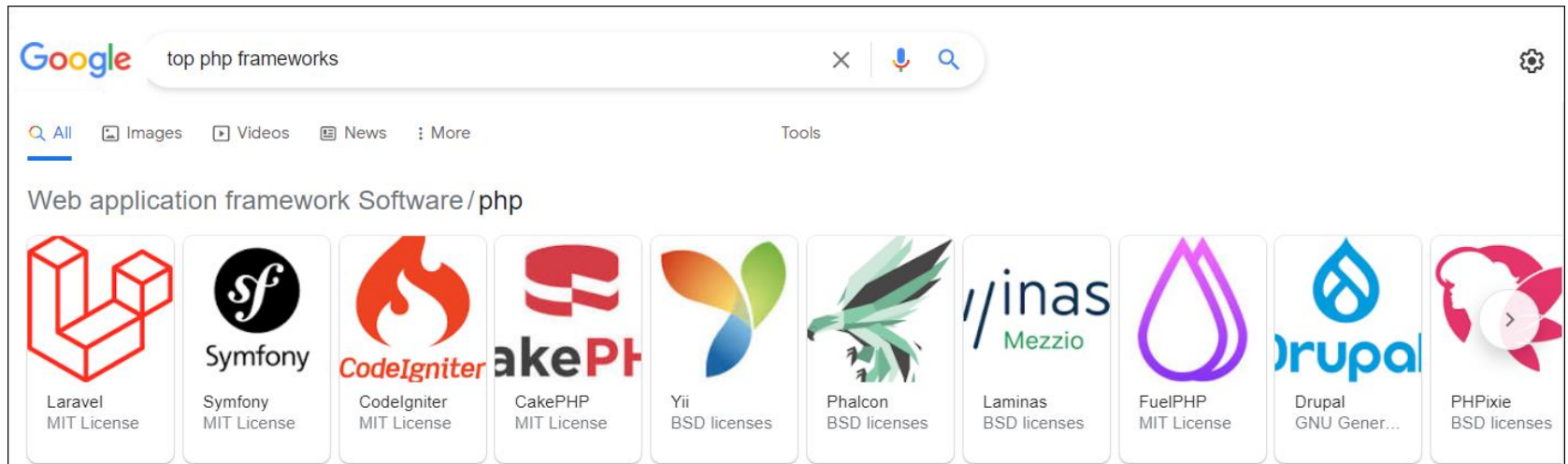
- ❖ Tạo controller trong Laravel
- ❖ Tạo action trong controller
- ❖ View trong Laravel
- ❖ Routing trong Laravel
- ❖ Routing nạp view và truyền tham số
- ❖ Routing vào action trong controller
- ❖ Tài liệu tham khảo



- ❑ PHP framework là các nền tảng nguồn mở viết bằng ngôn ngữ PHP.
- ❑ Các PHP Framework được tạo ra để giúp phát triển ứng dụng web nhanh hơn.
- ❑ Các PHP Framework thường dùng mô hình MVC, OOP. Và cung cấp rất nhiều class, thư viện, công cụ cho lập trình viên sử dụng.
- ❑ Có nhiều PHP Framework như:
  - ❖ Laravel
  - ❖ CodeIgniter
  - ❖ CakePHP
  - ❖ Phancon
  - ❖ Yii ...

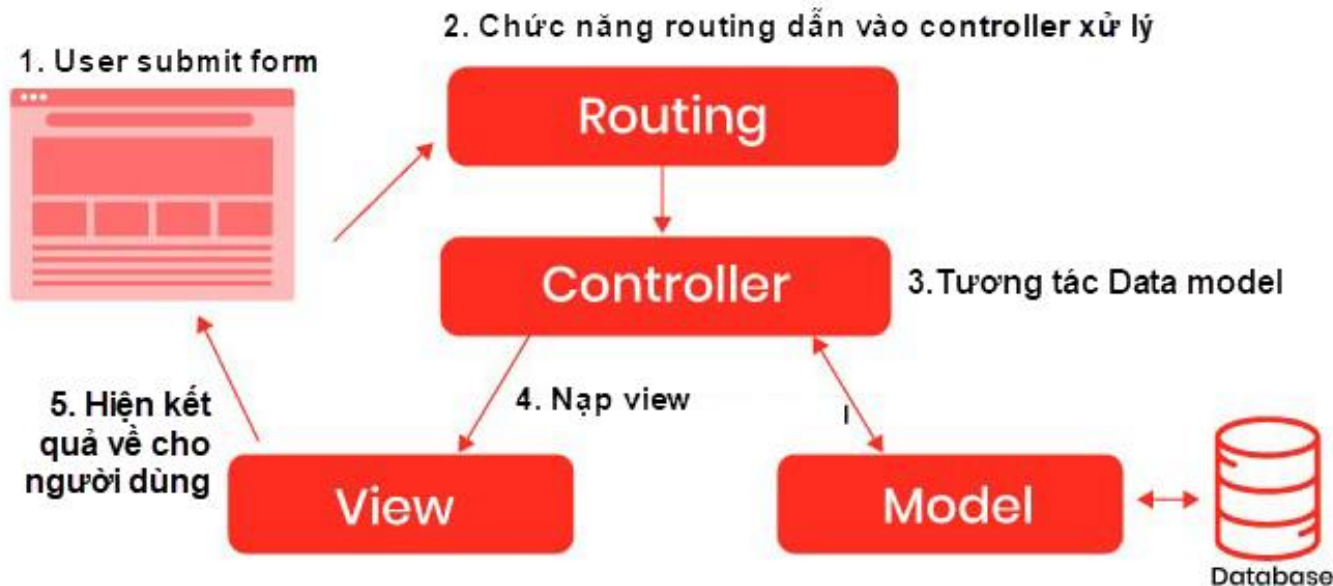


- ❑ Laravel là PHP framework rất tốt, cung cấp rất nhiều tính năng, dễ dùng.
- ❑ Rất nhiều lập trình viên ưa chuộng, cộng đồng sử dụng Laravel rộng lớn.
- ❑ Địa chỉ chính thức: <https://laravel.com/>



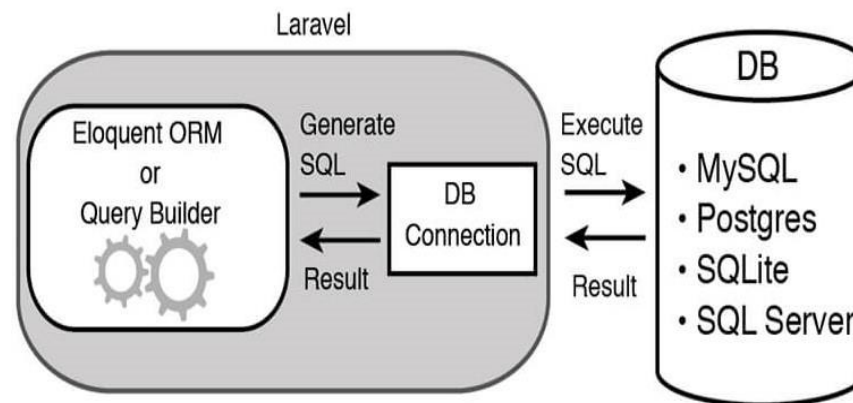
## 1. Hỗ trợ lập trình MVC

- ❖ Laravel hỗ trợ phát triển website theo mô hình MVC.
- ❖ Sử dụng MVC trong Laravel rất dễ dàng, linh động, uyển chuyển.



## 2. Query Builder và Eloquent ORM

- ❖ Đây là hai công cụ có sẵn trong Laravel giúp tương tác với database.
- ❖ Sử dụng hai công cụ này rất dễ dàng khi làm việc với database



## 3. Blade Template

- ❖ Laravel tích hợp sẵn Blade Template Engine. Đây là công cụ giúp tạo layout và các view rất uyển chuyển và mạnh mẽ.

## 4. Artisan

- ❖ Artisan là công cụ dòng lệnh có sẵn trong Laravel.
- ❖ Artisan là 1 trợ thủ đắc lực cho developer. Như giúp chạy project, tạo controller, model, migration, seeder...



A word cloud centered around the text "PHP ARTISAN". The words are arranged in a circular pattern around the central text. The words include: APP, AUTH, SCHEDULE, ROUTE, QUEUE, MIGRATE, STORAGE, KEY, VIEW, DB, EVENT, CONFIG, NOTIFICATIONS, MAKE, CACHE, SESSION, and VENDOR. The words "PHP ARTISAN" are the largest and most prominent in the center.

## 5. Authentication

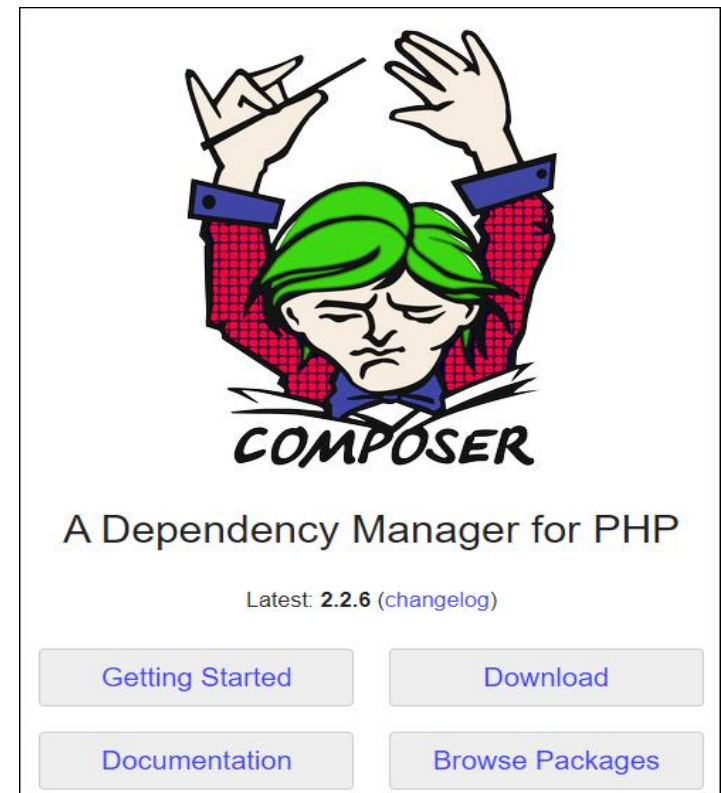
- ❖ Authentication trong Laravel giúp tạo các chức năng đăng ký, đăng nhập, đổi mật khẩu, quên mật khẩu, kiểm tra đăng nhập...rất nhanh chóng và đơn giản.



## 6. Security

- ❖ Laravel hỗ trợ thuật toán Bcrypt giúp mã hóa password
- ❖ Laravel hỗ trợ phát sinh các CSRF token dùng trong các form để chống tấn công giả mạo CSRF
- ❖ Laravel hỗ trợ tính năng Routing protected – bảo vệ các route
- ❖ Laravel cũng hỗ trợ Encryption - mã hóa và giải mã dữ liệu

- ❖ Composer là chương trình miễn phí giúp cài đặt, quản lý các gói thư viện cho các nền tảng PHP mã nguồn mở.
- ❖ Mỗi PHP framework dùng rất nhiều thư viện nguồn mở, mỗi thư viện nguồn mở có nhiều version khác nhau. Cho nên composer rất hữu ích.
- ❖ Khi cần thêm/cập nhật/gỡ bỏ thư viện nào trong project, bạn chỉ cần khai báo, composer sẽ giúp tải về qua một server cộng đồng.



- ❑ Download composer: <http://getcomposer.org>
- ❑ Cài đặt composer: Chạy file download, không có gì đặt biệt khi cài đặt, cứ để các cấu hình mặc định khi cài đặt.

The image shows a composite of two screenshots from the Composer website and its installation process. The left screenshot shows the 'getcomposer.org' website with the 'Download Composer' section. It highlights the 'Windows Installer' link with a red arrow labeled '2' and the 'Download' button with a red arrow labeled '1'. The right screenshot shows the 'Settings Check' and 'Select Destination Location' steps of the Composer-Setup.exe installer. In the 'Settings Check' window, the 'Next' button is highlighted with a red arrow. In the 'Select Destination Location' window, the 'Next' button is also highlighted with a red arrow, and the default installation path 'C:\composer' is shown in the text box.

getcomposer.org

Download Composer Latest: v2.1

Windows Installer

The installer - which requires that you have PHP already installed - will install the command-line PHP binary so you can simply call `composer` from any command prompt.

Download and run [Composer-Setup.exe](#) - it will install Composer.

Latest: **2.1.14** ([changelog](#))

A preview release for our next minor version is available! Try out **2.2.0-RC1** ([changelog](#)) now using `composer self-update --preview`

Getting Started Download

Settings Check

We need to check your PHP and other settings.

Choose the command-line PHP you want to use:

E:\xampp\php\php.exe Browse...

Back Next Cancel

Select Destination Location

Where should Composer be installed?

Setup will install Composer into the following folder.

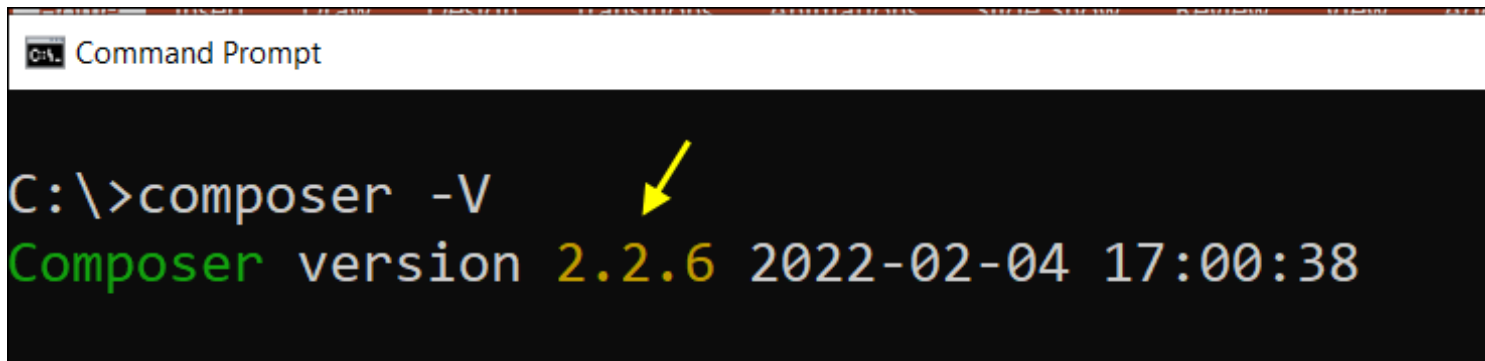
To continue, click Next. If you would like to select a different folder, click Browse.

C:\composer Browse...

Back Next Cancel

Trong command line, có thể dùng 1 số lệnh sau:

- ❑ Xem version của composer: **composer -V**
- ❑ Xem hướng dẫn sử dụng: **composer -h**
- ❑ Cập nhật các gói thư viện trong project: **composer update**
- ❑ Cài thêm thư viện vào project : **composer require symfony/mailer**
- ❑ Xóa thư viện khỏi project : **composer remove monolog/monolog**



```
Command Prompt
C:\>composer -V
Composer version 2.2.6 2022-02-04 17:00:38
```

- ❑ Mở command line, chuyển vào folder chứa project
- ❑ Gõ lệnh:

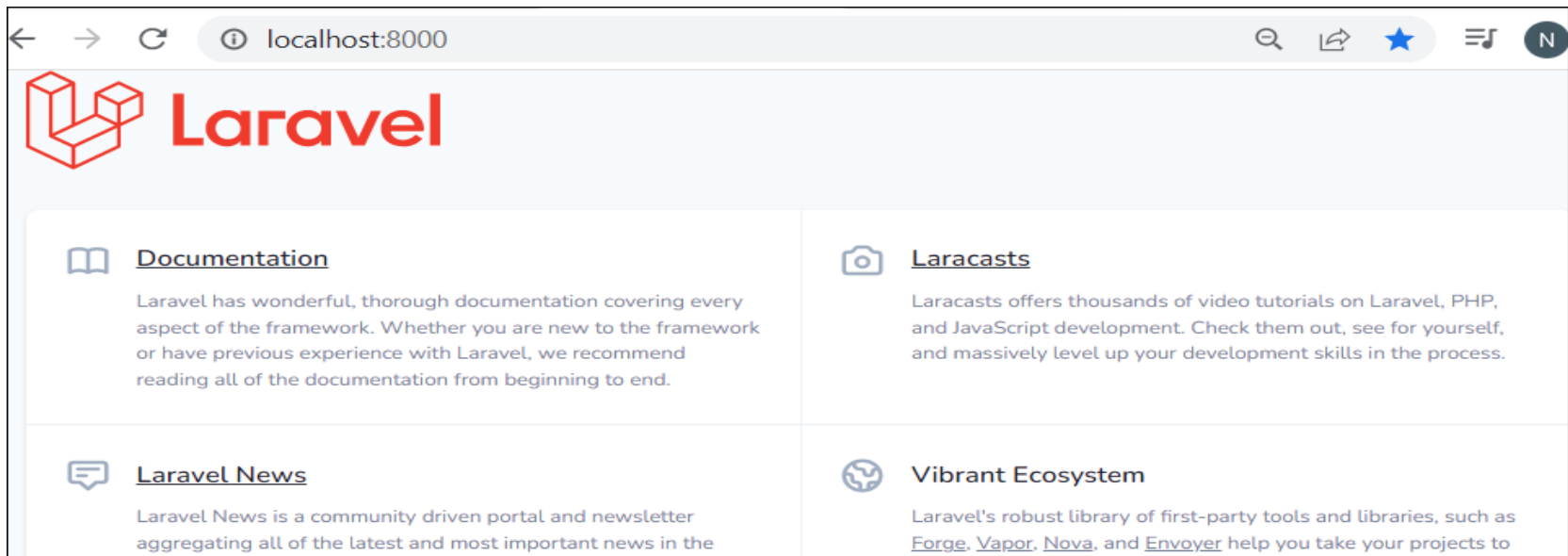
**composer create-project laravel/laravel myProject**

Trong đó **myProject** là tên folder project của bạn. Khi hoàn tất, sẽ thấy chữ **Application key set successfully**.

```
E:\PHP3>composer create-project laravel/laravel myProject
Creating a "laravel/laravel" project at "./myProject"
Installing laravel/laravel (v9.0.1)
- Installing laravel/laravel (v9.0.1): Extracting archive
Created project in E:\PHP3\myProject
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 109 installs, 0 updates, 0 removals
- Locking brick/math (0.9.3)
- Locking dflydev/dot-access-data (v3.0.1)
- Locking doctrine/infer (2.0.4)
No publishable resources for tag [laravel-assets].
Publishing complete.
> @php artisan key:generate --ansi
Application key set successfully.
E:\PHP3>
```

- ❑ Vào folder project và chạy lệnh **php artisan serve**
- ❑ Mở trình duyệt và gõ <http://localhost:8000>

```
E:\PHP3\myProject>php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Sat Feb 19 19:13:04 2022] PHP 8.1.2 Development Server (http://127.0.0.1:8000)
```



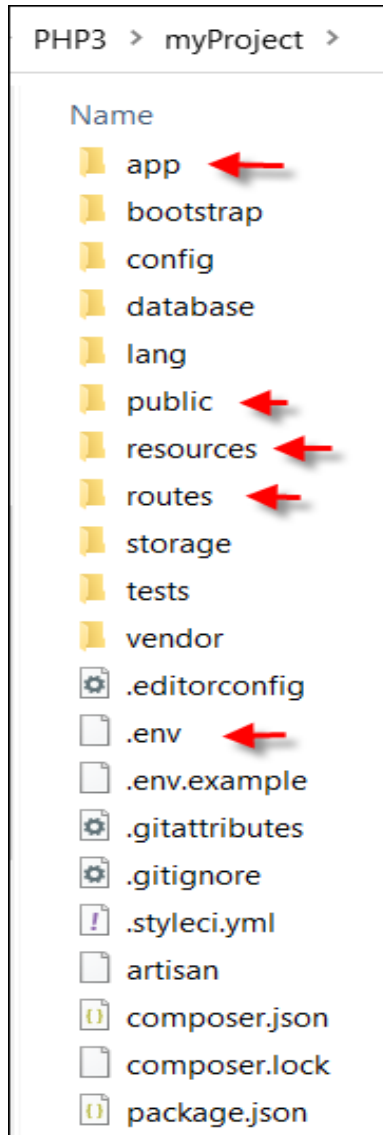
- ❑ Ngoài cách dùng lệnh **create-project** , có thể tạo project Laravel bằng cách dùng gói Laravel Installer.
- ❑ Cài đặt gói Lavavel Installer: Trong command line:  
**composer global require laravel/installer**
- ❑ Tạo project với Laravel Installer:  
**laravel new myPro1**

```
E:\PHP3>laravel new myPro1
```

The image shows the Laravel logo, which consists of the word "Laravel" in a stylized, outlined font. The letters are formed by red dashed lines on a dark background.

```
Creating a "laravel/laravel" project at "./myPro1"  
Installing laravel/laravel (v9.0.1)  
- Installing laravel/laravel (v9.0.1): Extracting archive  
Created project in E:\PHP3\myPro1
```

# CẤU TRÚC FILE TRONG LARAVEL PROJECT



- ❑ **app**: chứa controller, model, middleware...
- ❑ **bootstrap**: chứa file khởi động framework...
- ❑ **config**: chứa các file cấu hình db, mail, session...
- ❑ **database**: các file quản lý db như migration, seeder...
- ❑ **public**: chứa file index.php như cổng vào cho tất cả các request đến project.
- ❑ **resources**: chứa các file views, css, js...
- ❑ **routes**: chứa các file route trong project như: web.php, api.php, console.php
- ❑ **storage**: chứa các file cache, session, log... sinh ra trong quá trình thực thi project.
- ❑ **tests**: chứa các file tests
- ❑ **vendor**: các thư viện dùng trong laravel
- ❑ **.env** : file cấu hình chính của Laravel
- ❑ **artisan**: công cụ artisan của Laravel





DEMO

- Demo cài đặt Laravel





# LARAVEL FRAMEWORK

## BÀI 1: SỬ DỤNG CƠ BẢN LARAVEL PHẦN 2

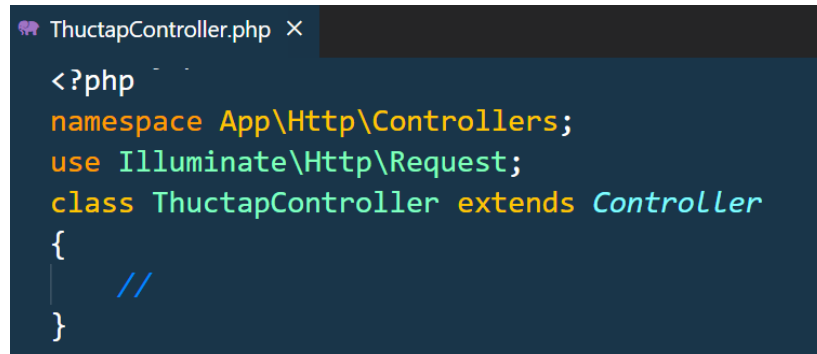
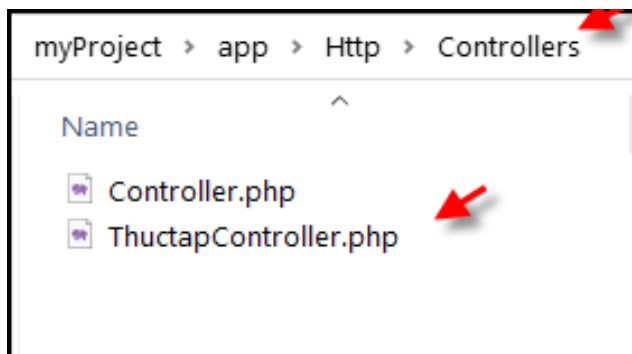
- ❑ Mỗi controller là 1 file đặt trong app/Http/Controllers
- ❑ Mỗi controller chứa nhiều hàm (action) và phải kế thừa từ controller gốc của Laravel
- ❑ Để tạo controller, dùng lệnh sau:

**php artisan make:controller ThuctapController**

Trong đó **ThuctapController** là tên controller muốn tạo.

```
E:\PHP3\myProject>php artisan make:controller ThuctapController  
Controller created successfully.
```

```
E:\PHP3\myProject>
```



- ❑ Action là các hàm định nghĩa trong controller để thực hiện một chức năng nào đó cho người dùng.
- ❑ Ví dụ: tạo action trả về tích của 2 số

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
class ThuctapController extends Controller {
    public function tinhhtich($a,$b){
        return $a*$b;
    }
}
```

- ❑ View là file chứa code html, css, js, php và dữ liệu cho user xem.
- ❑ Mỗi view là 1 file được lưu trong folder **resources\views**
- ❑ Nạp view từ trong action của controller hoặc trong route
- ❑ Khi nạp view, có thể truyền biến cho view.

```
<p>  
Hôm nay là ngày <?=date('d/m/Y')?><br/>  
<b> Kính chào quý khách</b>  
</p>
```

- ❑ Routing là chức năng khai báo các route (đường dẫn) , đưa user đến các chức năng trong project
- ❑ Người xem gọi đến các chức năng trong project thông qua các route đã khai báo trong project.
- ❑ Các route được khai báo trong file **routes/web.php** của project
- ❑ Mỗi route có thể trực tiếp nạp view hoặc dẫn đến các action trong controller
- ❑ Mỗi route có method http và có thể có tham số.

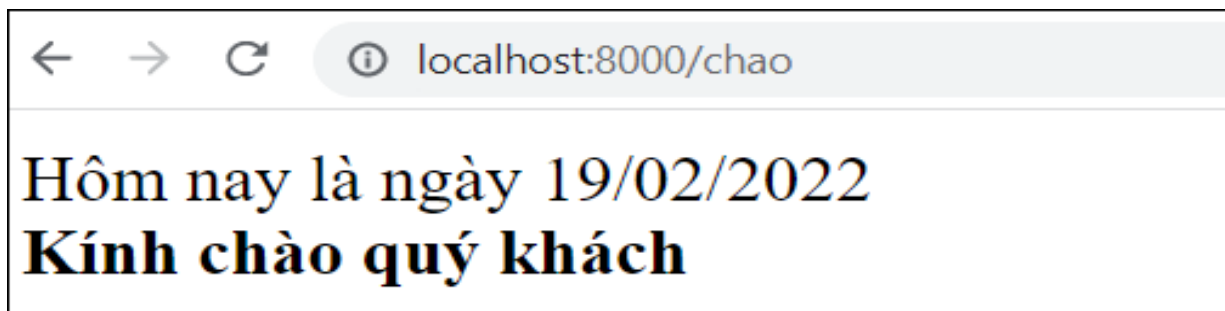
- ❑ Routing nạp view: dùng **Route::get** hoặc **Route::view**

```
Route::get('/chao',function(){  
    return view('xinchao');  
});
```

hoặc

```
Route::view('/chao', 'xinchao');
```

Request **chao** <http://localhost:8000/chao>  
sẽ thấy được view:



❑ Có thể truyền tham số cho view qua 1 mảng. Ví dụ:

```
Route::get('/chuc',function(){  
    $str = "Kính chúc vạn sự bình an";  
    $t = "Chào quý khách!";  
    return view('chuc',['title'=>$t,'chuc'=>$str]);  
});
```

hoặc

```
Route::view('/chuc','chuc', [  
    'title' => 'Kính chúc vạn sự bình an',  
    'chuc' => 'Chào quý khách!'   
]);
```



## ❑ Views/chuc.php

```
<p class='today'>  
Hôm nay <?=date('d/m/Y')?>  
</p>  
<h4> <?=$title?>  
<em><?=$chuc?></em>  
</h4>
```

← → ↻ ⓘ localhost:8000/chuc

Hôm nay 19/02/2022

**Kính chúc vạn sự bình an *Chào quý khách!***

## ❑ Routing trong routes/web.php

```
Use App\Http\Controllers\ThuctapController;  
Route::get('cong/{a}/{b}', [ThuctapController::class, 'cong2so']);
```

## ❑ Http/Controllers/ThuctapController.php

```
<?php //ThuctapController  
Namespace App\Http\Controllers;  
use Illuminate\Http\Request;  
class ThuctapController extends Controller {  
    public function cong2so($a,$b){  
        return View('cong', ['kq' => $a + $b] );  
    }  
}
```

## ❑ Views/.cong.php

```
<p> Kết quả phép cộng là :  
<b><?=$kq?></b>  
</p>
```

← → ↻ ⓘ localhost:8000/cong/2/3

Kết quả phép cộng là : 5



# DEMO

- Demo tạo controller, view, routing



## ❑ **Laravel documentation**

- ❖ Tài liệu chính thức của Laravel: <https://laravel.com/docs/>
- ❖ Sử dụng routing: <https://laravel.com/docs/routing>
- ❖ Tạo và sử dụng view: <https://laravel.com/docs/views>
- ❖ Sử dụng artisan: <https://laravel.com/docs/artisan>
- ❖ TutorPoint: <https://www.tutorialspoint.com/laravel/>

## ❖ **Composer documentation**

- ❖ <https://getcomposer.org/doc/03-cli.md>

## ❑ Phần 1

- ❖ Giới thiệu PHP framework và Laravel
- ❖ Các tính năng nổi bật của Laravel
- ❖ Giới thiệu và cài đặt và sử dụng composer
- ❖ Tạo project Laravel và thực thi project Laravel
- ❖ Cài đặt và sử dụng gói Laravel Installer
- ❖ Cấu trúc file và folder trong Laravel project

## ❑ Phần 2

- ❖ Tạo controller trong Laravel
- ❖ Tạo action trong controller
- ❖ View trong Laravel
- ❖ Routing trong Laravel
- ❖ Routing nạp view và truyền tham số
- ❖ Routing vào action trong controller

## ❑ Tài liệu tham khảo





**Cảm ơn**