# LARAVEL (8-2021) – FRAMEWORK PHP 1

# SOMMAIRE

https://laravel.com	<u>/</u>
https://laravel.com/docs	<u>s/10.x</u>
https://laravel.com/docs	s/11.x

Sommaire	1
Etape 1	
0 – Objectifs de l'étape 1	
Principes du cours	
1 - Laravel (v1)	
Laravel : framework PHP pour les artisans du web	4
Qu'est-ce que Laravel : c'est un framework PHP	4
Site interet : https://laravel.com/	4
Laravel 11 actuellement (en 2025).	4
Avantages / inconvénient d'un framework – Popularité des frameworks	5
Pourquoi utiliser un framework	5
Avantages – Inconvénients	5
Laravel : le framework PHP le plus populaire (2025) :	5
Différences entre framwork et micro-framework	6
Concepts de base	7
Prérequis	8
Rappel d'une architecture WEB de base.	8
2 - Installer Laravel (v3, v4)	
Mise en garde	9
Composer	
Présentation	
Tester la présence de composer sur votre machine :	
Installer composer	
Aide en ligne de commande	11
Installation sur mac	12
Packagist	13
MySQL	13
РНР	14
3 - Créer une application laravel	15
Principes	
Un dossier pour toutes nos applications	
1 : Création d'une application laravel avec composer create-project	
Principes	
Connaitre la version installée et documentation laravel	
2 : Démarrage d'une application laravel : php artisan serve	
Démarrage du serveur apache et démarrage de l'application web	
Résumé des commandes	
3 : avec Laragon (on peut sauter)	
4 : avec laravel/installer <mark>(on peut sauter)</mark>	
Installer :	
Il faut mettre à jour le \$PATH (variable d'environnement)	

Démarrer le serveur	
Résumé :	19
4 - Démarrer une application Laravel	20
php artisan	20
Aide en ligne	20
Quelques commandes artisan	20
Documentation complète	20
Démarrage du server	21
Démarrage de l'application dans un navigateur	
Résultats Laravel 11 :	22
Résultats Laravel 7 :	22
Rappel du fonctionnement client - serveur web	23
5 – Architecture, manipulation et bilan (v3, v4)	24
Architecture d'un projet Laravel :	
Architecture d'un projet Laravel	24
Quelques fichiers à regarder pour commencer	25
Les 10 dossiers installés	26
Dossiers dans « app »:	28
Dossier dans « ressources » :	28
Des explications plus complètes	29
Manipulation d'un projet laravel	
Réduction de la taille du dossier	30
Ouverture d'un projet	31
Modification de l'affichage	31
<mark>6 : TP de synthèse : installer un nouveau projet Laravel appelé « monapp »</mark>	32
Installer un nouveau projet Laravel	

# **ETAPE 1**

https://laravel.com/ https://laravel.com/docs/10.x https://laravel.com/docs/11.x

# 0 – Objectifs de l'étape 1

 $\Rightarrow$  L'étape 1 :

⇒ présente Laravel

 $\Rightarrow$  se termine par le TP du chapitre 6 : installation d'un nouveau projet.

⇒ L'objectif de l'étape 1 est de se familiariser avec les concepts, les outils, l'architecture et les commandes d'installation.

# Principes du cours

- Les parties surlignées de bleu correspondent à des installation ou exercices à faire.
- Les parties surlignées en jaune sont des concepts importants.
- Les parties marquées (on peut sauter) peuvent être sautées.
- On commence par une présentation complète de chaque étape, puis on fait le TP.

# 1 - Laravel (v1)

#### Laravel : framework PHP pour les artisans du web

## Qu'est-ce que Laravel : c'est un framework PHP

 $\Rightarrow$  Il fournit une architecture semi-finie qui sert de « patron » (« pattern ») pour notre travail.

 $\Rightarrow$  Il fournit des outils pour générer du code automatiquement avec l'outil « php artisan ».

 $\Rightarrow$  Il permet d'ajouter facilement de nouvelles briques avec l'outil « **composer** ».

#### Donc, il faut se familiariser avec :

 $\Rightarrow$  l'architecture,

 $\Rightarrow$  php artisan

 $\Rightarrow$  composer.

## Site interet : https://laravel.com/

 $\Rightarrow$  Le site donne accès à la documentation : <u>https://laravel.com/docs/11.x</u>

 $\Rightarrow$  On peut la traduire en français dans le navigateur.

## Laravel 11 actuellement (en 2025).

 $\Rightarrow$  Laravel a été lancé en juin 2011

## Avantages / inconvénient d'un framework – Popularité des frameworks

### Pourquoi utiliser un framework

https://kinsta.com/fr/blog/frameworks-php/

- $\Rightarrow$  1. Un développement plus rapide
- $\Rightarrow$  2. Moins de code à écrire
- $\Rightarrow$  3. Bibliothèques pour les tâches communes
- $\Rightarrow$  4. Suivre les bonnes pratiques de codage
- $\Rightarrow$  5. Plus sûr que d'écrire vos propres applications
- $\Rightarrow$  6. Un meilleur travail d'équipe
- $\Rightarrow$  7. Plus facile à entretenir

#### Avantages – Inconvénients

- $\Rightarrow$  Avantages :
  - $\Rightarrow$  on code plus vite, plus propre, plus sécurisé !
  - $\Rightarrow$  Tous les frameworks se ressemblent un peut.

## $\Rightarrow$ Inconvénients :

- $\Rightarrow$  il faut apprendre le framework ! C'est long !
- ⇒ C'est spécifique (Laravel ce n'est pas comme Zend ou Symfony).
- $\Rightarrow$  C'est plus ou moins pérenne.
- $\Rightarrow$  On n'apprend pas vraiment le langage de base.

#### Laravel : le framework PHP le plus populaire (2025) :

 $\Rightarrow$  <u>https://kinsta.com/fr/blog/frameworks-php/</u>

#### Différences entre framwork et micro-framework

- ⇒ Un **framework** fournit une architecture semi-finie qui sert de « patron » (« pattern ») pour notre travail.
  - ⇒ En général, un framework propose l'installation d'une application minimaliste avec toute l'architecture des dossiers et des fichiers pour la suite.
  - ⇒ En général, un framework impose un modèle structuré et sécurisé.
  - $\Rightarrow$  C'est une grosse architecture, complexe et associé à beaucoup d'outils
  - $\Rightarrow$  La prise en main est complexe.
  - ⇒ Il existe de très nombreux frameworks : Laravel (PHP), Django (Python), Ruby On Rail (Ruby), Spring (Java), Angular (JavaScript/TypeScript), etc.
- ⇒ Un **micro-framework** est un cadre de développement minimaliste qui fournit uniquement les fonctionnalités de base nécessaires pour créer une application.
  - ⇒ En général, un micro framework ne propose pas l'installation d'une application minimaliste.
  - $\Rightarrow$  Il laisse une très grande liberté au programmeur.
  - $\Rightarrow$  Il permet de coder facilement une application minimaliste.
  - ⇒ Express (Node.js) et Flask (Python) sont les 2 micro-frameworks les plus populaires. Slim est un micro-framework PHP.

# **Concepts de base**

https://kinsta.com/fr/blog/frameworks-php/

- Architecture d'un projet Laravel. MVC (on revient sur le MVC à l'étape 3).
- **Routage** (pour faciliter les chemins d'accès aux pages du site)
- **ORM** <u>Eloquent</u> (pour faciliter l'usage de la BD). La notion de « migration ».
- Moteur de templates <u>Blade</u> (pour faciliter l'écriture des Vues du MVC).
- La commande « php artisan »
- Installation Composer Packagist

# Prérequis

- Architecture WEB de base.
- PHP
- P.O.O.
- SGBD-R : MySQL ou équivalent.
- HTML-CSS (Bootstrap, un peu)

# Rappel d'une architecture WEB de base.



# 2 - Installer Laravel (v3, v4)

# Mise en garde

- L'installation peut s'avérer compliquée (ou se passer sans problème) ! Il ne faut pas se décourager.
- Quand on passe des commandes en ligne de commandes, il faut lire les messages qui s'affiche (au moins la fin) : on y trouve parfois des pistes de solutions.
- Il faut chercher sur Google des solutions.
- Accrochez-vous. On finit par y arriver !

#### Composer

**Présentation** 

- Composer est un équivalent pour PHP du « **npm** » pour Node.js.
- Composer est un équivalent pour PHP du « **pip** » pour Python.

Langage	РНР	Python	Node.js
Package Manager	composer	pip	npm

- Composer permet d'installer une application Laravel avec toutes ses dépendances.
- Composer installe les versions des packages qui doivent être installés en les téléchargeant dans votre projet.
- Composer permet de **mettre à jour toutes les dépendances** en une seule commande, en déclarant seulement les bibliothèques dont vous dépendez (c'est la commande « **composer updat**e »). Il gère le fait que nos bibliothèques dépendent d'autres bibliothèques,
- Pour utiliser « composer », tout se passe dans le fichier **composer.json** à la racine de votre projet. Ce fichier va définir toutes les informations concernant votre projet ainsi que toutes ses dépendances : la clé require précise les modules et bibliothèques dont on a besoin ainsi que leurs versions.

#### Tester la présence de composer sur votre machine :

```
composer --version
composer -v
```

# • Si composer n'est pas présent, installer le sur votre machine.

## Installer composer

- Pour pouvoir utiliser Laravel, on commence donc par installer « Composer ».
- Le site de « Composer » permet son installation : https://getcomposer.org/

#### <u>Aide en ligne de commande</u>

• Une fois installé, les commandes suivantes permettent de récupérer de l'information :

>composer -h >composer list

- Usage basique : <u>https://getcomposer.org/doc/01-basic-usage.md</u>
- Liste des commandes : <u>https://getcomposer.org/doc/03-cli.md</u>

#### **Installation sur mac**

```
php -r "copy('https://getcomposer.org/installer', 'composer-
setup.php');"
php -r "if (hash_file('sha384', 'composer-setup.php') ===
'dac665fdc30fdd8ec78b38b9800061b4150413ff2e3b6f88543c636f7cd84f6db9
189d43a81e5503cda447da73c7e5b6') { echo 'Installer
verified'.PHP_EOL; } else { echo 'Installer corrupt'.PHP_EOL;
unlink('composer-setup.php'); exit(1); }"
php composer-setup.php
php -r "unlink('composer-setup.php');"
sudo mv composer.phar /usr/local/bin/composer
composer --version
```

#### **Packagist**

- Packagist.org est le site de dépots des packages PHP (packages ou paquets, bibliothèques, librairies)
- <u>https://packagist.org/</u> = The PHP Package Repository
- Ce site permet de chercher tous les packages disponibles mais aussi d'enregistrer vos propres packages.
- Il y en a un très grand nombre : dans la pratique, mieux vaut savoir ce qu'on cherche !

#### **MySQL**

- Laravel utilise le SGBD MySQL. Il faut donc qu'il soit installé sur nos machines.
- > Pour installer MySQL sur PC et sur MAC) :
- On peut passer par son WAMP ou équivalent : l'installation a été faite automatiquement.

#### PHP

- Laravel utilise du PHP, il faut donc que le PHP soit installé sur nos machines.
- > Pour savoir si PHP est installé :

php --version php -v

- Pour laravel 11, il faut une version PHP 8.2 minimum
- Pour laravel 10, il faut une version PHP 8.1 minimum
- > Pour savoir où est le PHP :

where php

- > Pour installer PHP sur MAC :
- https://www.php.net/manual/fr/install.macosx.packages.php

brew install php

Pour installer php 8.2 (le minimum pour Laravel 11) : brew install php@8.2

Passer en php8.2 sur mac BigSur, c'est compliqué !!!

## > Pour mettre à jour brew avec git :

```
cd $(brew --repo)
git fetch -all
git reset --hard origin/master
```

- > Pour installer PHP sur PC :
- Il y a plusieurs techniques.
- On peut passer par son WAMP ou équivalent : l'installation a été faite automatiquement.

# > Pour mettre à jour PHP sur PC:

- Il y a plusieurs techniques.
- On peut passer par son WAMP ou équivalent.
- On peut aussi charger une version « à la main » : <u>https://windows.php.net/download/</u> : development package, safe, x64.
- Il faut ensuite la mettre dans C:/WAMP/bin/php et créer un dossier.
- Il faut aussi mettre à jour le PATH dans les variables d'environnement.

# 3 - Créer une application laravel

## **Principes**

https://laravel.com/docs/9.x/configuration

- 3 techniques :
  - avec composer create-project : le plus simple
  - avec laravel/installer : plus compliqué (on peut passer)
  - avec Laragon (un genre de WAMP) : que sous Windows (on peut passer)
- Pour tout installer, on utilisera en priorité la première technique : composer create-project

#### Un dossier pour toutes nos applications

- Dans votre dossier de cours Laravel, créez un dossier Applications
  - Ouvrez un terminal dans un dossier Lavarel/Applications

### 1: Création d'une application laravel avec composer create-project

#### **Principes**

• Composer propose des outils qui vont installer un squelette d'application PHP avec un serveur WEB.

## Créer un projet Laravel

• Pour créer un projet Laravel nommé « monApp », on passera la commande :

composer create-project laravel/laravel monApp

- Pour installer la version 11 :
   composer create-project laravel/laravel monApp "11.\*"
- Pour installer la version 10 :

composer create-project laravel/laravel monApp "10.\*"

- > Documentation composer create project :
- <u>https://getcomposer.org/doc/03-cli.md#create-project</u>

#### Connaitre la version installée et documentation laravel

- Une fois l'installation finie, aller dans le dossier monApp puis utiliser php artisan
- Dans le dossier laravel, tapez : php artisan --version

cd monApp php artisan --version

- > Documentation Laravel :
- <u>https://laravel.com/docs/11.x/installation</u>
- On peut choisir la version

## 2 : Démarrage d'une application laravel : php artisan serve

#### Démarrage du serveur apache et démarrage de l'application web

```
>cd monApp
>php artisan serve
```

• La commande php artisan serve démarre le serveur de développement et précise l'adresse pour accéder à l'application dans un message :

Starting Laravel development server: http://127.0.0.1:8000

- On peut démarrer l'application laravel dans un navigateur : localhost :8000 ou 127.0.0.1 :8000 URL>localhost : 8000
- Php artisan serve démarre un serveur. A ce stade on n'a pas besoin de WAMP.

<pre>&gt;composer create-project laravel,</pre>	/laravel monApp
<pre>&gt;cd monApp &gt;php artisan serve</pre>	
aravel × +	- L
C () localhost:8000	🛱 🤍 🛠 🚺 🥘 Confirmer votre ident
	LA LA
Larovel         •           •         •           •         •           •         •	► Laracasts Laracasts offers thousands of video tutorials on Laravel, PHP, and JavaScript development. Check them out, see for yourself, and massively level up your development skills in the process.
	Laravel News Laravel News is a community driven portal and newsletter aggregating all of the latest and most important news in the Laravel ecosystem, including new package releases and tutorials.
Documentation Laravel has wonderful documentation covering every aspect of the framework. Whether you are a newcomer or have prior experience with Laravel, we recommend reading our documentation from beginning to end.	Vibrant Ecosystem Laravel's robust library of first-party tools and libraries, such as <u>Forge</u> . <u>Vapor. Nova. Envoyer</u> , and <u>Herd</u> help you take your projects to the next level. Pair them with powerful open source libraries like <u>Cashier</u> . <u>Dusk</u> , <u>Echo</u> <u>Horizon</u> , <u>Sanctum</u> , <u>Telescope</u> , and more.

#### **3 : avec Laragon** (on peut sauter)

- Laragon est un très bon outil mais disponible uniquement sur Windows !
- <u>http://laragon.org</u>

#### 4 : avec laravel/installer (on peut sauter)

- Documentation Laravel :
  - https://laravel.com/docs/8.x/installation#the-laravel-installer

#### **Installer :**

• On commencer par installer le laravel/installer comme dépendance globale de Composer.

>composer global req laravel/installer

• On crée une application avec laravel :

>laravel new monApp

• Problème : laravel n'est pas reconnu !!!

#### Il faut mettre à jour le \$PATH (variable d'environnement)

• Sur MAC :

```
>echo $PATH
>export PATH=$PATH:$HOME/.composer/vendor/bin
>echo $PATH
```

- La modification ne sera valable que sur le terminal où l'on fait l'export.
- On peut alors exécuter laravel dans ce terminal
- On peut aussi ne pas changer le path et passer la commande :

```
>$HOME/.composer/vendor/bin/laravel new monApp
```

- Sur PC :
  - On ajoute une nouvelle variable d'environnement :

```
%USERPROFILE%\AppData\Roaming\Composer\vendor\bin
```

- Pour accéder aux variables d'environnement : cherchez variables, allez dans modifier les variables d'environnement, cliquez sur variables d'environnement, double-cliquez sur le path des variables utilisateur, ajoutez la variable.
- On peut alors exécuter laravel dans un terminal
- Sur Linux :
  - Cf doc laravel et google !

#### Démarrer le serveur

```
>cd monApp
>php artisan serve
```

• La commande php artisan serve démarre le serveur de développement et précise l'adresse pour accéder à l'application dans un message :

Starting Laravel development server: <a href="http://l27.0.0.1:8000">http://l27.0.0.1:8000</a>

#### Aide en ligne de commande

>\$HOME/.composer/vendor/bin/laravel	-V
>\$HOME/.composer/vendor/bin/laravel	-h
>\$HOME/.composer/vendor/bin/laravel	list

• Il n'y a qu'une commande : la commande « new »

>\$HOME/.composer/vendor/bin/laravel new monApp

## Résumé :

>composer global req laravel/installer
># mise à jour du path
>laravel new monApp
>cd monApp
>php artisan serve

# 4 - Démarrer une application Laravel

#### php artisan

- « php artisan » est l'interface de ligne de commande incluse avec Laravel.
- La commande « php artisan » s'utilise dans un projet laravel.

#### Aide en ligne

• On commence par se mettre dans son projet :

>cd monApp

• Commande pour connaitre la version :

>php artisan -version

• Commande pour lister les commandes existantes

>php artisan list

#### **Quelques commandes artisan**

```
>php artisan make
>php artisan migrate
>php artisan serve
Etc.
```

- La commande make sert à créer du code (créer des controleurs par exemple).
- La commande migrate sert à créer la BD.
- La commande serve démarre le serveur.
- Etc.

#### **Documentation complète**

- A partir de l'accueil de la documentation on accède à ce qu'on veut :
- <u>https://laravel.com/docs/11.x/</u>
- Artisan : https://laravel.com/docs/11.x/artisan#main-content
- Migration : <u>https://laravel.com/docs/11.x/migrations#main-content</u>
- Etc.

#### Démarrage du server

- La commande Laravel « php artisan serve » permet d'exécuter des applications sur le serveur de développement PHP.
- Elle propose un port disponible pour démarrer un client.
- Pour démarrer le serveur :

>cd monApp >php artisan serve

• On obtient un message :

Starting Laravel development server: http://127.0.0.1:8000

• L'adresse proposée permet d'exécuter l'application dans un navigateur

#### Démarrage de l'application dans un navigateur

• Une fois le serveur démarré et le message obtenu :

Starting Laravel development server: http://127.0.0.1:8000

- Dans un navigateur, on passe l'adresse renvoyée par php artisan serve :
  - <u>http://127.0.0.1:8000</u>
    - ou encore :
  - http://localhost:8000

#### **Résultats Laravel 11 :** C Laravel × + \_ → C ③ localhost:8000 🔄 🔍 🛧 🖸 🚺 🔞 Confirmer votre identité Laracasts 🕼 Laravel $\rightarrow$ Laracasts offers thousands of video tutorials on Laravel, PHP, and JavaScript development. Check them out, see for yourself, and massively level up your development skills in the process. Laravel News + Laravel News is a community driven portal and newsletter aggregating all of the latest and most important news in the Laravel ecosystem, including new package releases and tutorials. (1) Vibrant Ecosystem Documentation Laravel's robust library of first-party tools and libraries, such as Forge. Vapor, Nova, Envoyer, and Herd help you take your projects to the next level. Pair them with powerful open source libraries like <u>Cashier</u>, <u>Dusk</u>, <u>Echo</u>, Laravel has wonderful documentation covering every aspect of the framework. Whether you are a newcomer or have prior experience with Laravel, we recommend reading our documentation from beginning to end. Horizon, Sanctum, Telescope, and more. $\rightarrow$

Laravel v11.42.1 (PHP v8.2.0)

#### **Résultats Laravel 7 :**

⊳ C	<b>Q</b> 127.0.0.1:8000		4					Ο Δ	۱ ا
				ara					
				.arc		<i>,</i>			
	DOCS	LARACASTS	NEWS	BLOG	NOVA	FORGE	VAPOR	GITHUB	

*Laravel* – *P1* – *page* 22/32

Rappel du fonctionnement client - serveur web



# 5 – Architecture, manipulation et bilan (v3, v4)

# Architecture d'un projet Laravel :

# Architecture d'un projet Laravel

- L'architecture d'un projet Laravel, c'est l'organisation des dossiers et des fichiers.
- Dans monApp, on trouve :
  - $\Rightarrow$  10 dossiers
  - $\Rightarrow$  14 fichiers
- C'est complexe et compliqué :



#### Quelques fichiers à regarder pour commencer

#### .env

- Il contient entre autres des infos sur la BD :
  - le type de BD : mysql,
  - l'utilisateur, son mot de passe : à adapter à votre système (root sans mot de passe sur PC, root mot de passe root sur Mac)
  - le nom de la BD (pour la BD, la collation utf8mb4\_general\_ci permet d'intégrer les smiley) : il faut créer une BD correspondant à ce nom pour développer l'application.

# composer.json

• La clé « require » précise les dépendances de notre application laravel.

#### Les 10 dossiers installés

## ▶ 1 : app

- Contient tout le code généré pour l'application.
- Il contient de nombreux autres dossiers qui sont détaillés après.

## > 2 : bootstrap

– Permet l'initialisation du framework bootstrap.

## ➤ 3 : config

– Contient l'ensemble des fichiers de configurations de l'application.

## ➤ 4 : database

– Contient les informations pour le SGBD.

## ▷ 5 : public

- **Point d'entrée** unique de l'application par l'intermédiaire du fichier « index.php ».
- Seul dossier visible de l'extérieur. Tous les autres sont cachés et protégés des attaques de l'extérieur.
- Contient également vos différents assets tels que les images, les fichiers CSS et JavaScripts compilés.

## ▶ 6 : resources

- Contient, dans le dossier « views », l'ensemble des fichiers de Vues gérés par défaut par le moteur de template de Laravel : Blade.
- Contient les fichiers de langues dans le cas d'une application multilingues.

## > 7 : routes

– Contient des fichiers listant l'ensemble des routes utilisées par votre application.

## > 8 : storage

- Contient des fichiers compilés automatiquement par le système.
- > 9 : tests
  - Contient les tests unitaires. Laravel utilse PHPUnit.

## > 10 : Vendor

 Contient les dépendances Composer associées à l'application (c'est l'équivalent de node\_modules pour Nodes).

#### Dossiers dans « app »:

- L'essentiel de l'application se trouve dans le dossier app.
- Les 2 dossiers les plus importants sont :

## > Http

- Contient les contrôleurs de l'architecture MVC.
- Documentation Laravel associée : <u>https://laravel.com/docs/11.x/controllers</u>

## > Models

- Contient les modèles de l'architecture MVC (les classes correspondant à la BD).
- Documentation Laravel associée : <u>https://laravel.com/docs/11.x/eloquent</u>

#### **Dossier dans « ressources » :**

- Les vues de l'application se trouvent dans le dossier app.
- Le dossier les plus importants est :
- views
  - Contient les vues de l'architecture MVC : les fichiers pour le moteur de templating Blade
  - Documentation Laravel associée : <u>https://laravel.com/docs/11.x/blade</u>

#### Des explications plus complètes

• On trouve des explications plus ou moins complètes dans la doc Laravel et sur des sites web. Mais on ne va pas s'y intéresser tout de suite.

 $\Rightarrow$  Doc Laravel : <u>https://laravel.com/docs/11.x/structure</u>

 $\Rightarrow$  Site web : <u>doc ici</u>

• C'est théorique. On va se familiariser au fur et à mesure !

# > node\_modules

- Un autre dossier apparaitra rapidement :le dossier node\_modules
  - Contient les packages pour Nodes si on l'utilise.

## **Manipulation** d'un projet laravel

#### Réduction de la taille du dossier

- > Taille du dossier
- Dossier monApp : 52 MO
- Dossier vendor : 51,6 MO
- Suppression du dossier vendor et composer update
- On peut supprimer le dossier vendor : notre application ne pèse plus que 0,55 MO !

 $\Rightarrow$  « php artisan serve » ne marche plus

• On peut refabriquer le fichier vendor :

>composer update >php artisan serve Starting Laravel development server: http://127.0.0.1:8000

#### **Ouverture d'un projet**

Avec VS Code, on ouvre le dossier monApp

#### Modification de l'affichage

• Le fichier de description de la page d'accueil est dans le dossier « ressources / views »

 $\Rightarrow$  Fichier « ressources / views / welcome.blade.php »

- Modifiez le fichier welcome.blade.php pour obtenir le résultat suivant :
  - ⇒ On a rajouté « Cours Laravel Golden Collar». Le lien amène sur la page du site <u>http://bliaudet.free.fr/article.php3?id\_article=348</u>

 $\Rightarrow$  L'exercice sert à regarder la page blade et à la modifier.

🕑 Laravi	el × +	- 0
÷ → C	O localhost:8000	🛐 🔍 🖈 🖸 🛛 🔞 Confirmer votre identité
	C.	P
¥	Loravel	► Laracasts Laracasts offers thousands of video tutorials on Laravel, PHP, and JavaScript development. Check them out, see for yourself, and massively level up your development skills in the process.
		► Laravel News Laravel News is a community driven portal and newsletter aggregating all of the latest and most important news in the Laravel ecosystem, including new package releases and tutorials.
Doct Larave Larave	In the worderful documentation covering every aspect of the work. Whether you are a newcomer or have prior experience with at, we recommend reading our documentation from beginning to end.	Vibrant Ecosystem Laravel's robust library of first-party tools and libraries, such as <u>Forge</u> . <u>Vapor</u> , <u>Nova</u> , <u>Envoyer</u> , and <u>Herd</u> help you take your projects to the next level. Pair them with powerful open source libraries like <u>Cashier</u> , <u>Dusk</u> , <u>Echo</u> , <u>Horizon</u> , <u>Sanctum</u> , <u>Telescope</u> , and more.
Đ	Cours Laravel Golden Collar Cours Laravel en pdf.	
	Laravel v11.42.1 (PH	(PHP v8.2.0)

# 6 : TP de synthèse : installer un nouveau projet Laravel appelé « monapp »

### Installer un nouveau projet Laravel

Suivre les étapes présentées dans le cours et qui sont résumées ici :

- 1. Affichez la version de PHP de votre ordinateur. Vérifiez qu'elle est compatible avec Laravel 11.
- 2. Installez Composer. Affichez la version de composer.
- 3. Créez un projet Laravel.
- 4. Vérifiez la version de Laravel.
- 5. Démarrez ce projet dans un navigateur.
- 6. Relisez le chapitre sur l'architecture d'un projet Laravel.
- 7. Supprimez le dossier vendor. Vérifiez que le projet ne peut plus être démarré. Recréez le dossier vendor (avec composer uniquement à ce stade). Redémarrez l'application.
- 8. Ajoutez " Cours Laravel Golden Collar " avec un lien vers : http://bliaudet.free.fr/article.php3?id\_article=348

Le graphisme à ajouter est le suivant :

Cours Laravel Golden Collar
Cours Laravel en pdf.