



Framework PHP : LARAVEL

Installation et configuration

Mouhamed DIOP - Ingénieur de Conception en Informatique (ESP)

Objectifs spécifiques

A l'issue de cette séquence, l'étudiant devrait :

- Savoir installer un environnement de développement Laravel
- Connaitre les différentes manières d'installer Laravel
- Connaitre les configurations de base de Laravel
- Connaitre les dossiers de base de Laravel et leur utilité



Qu'est-ce que Laravel ?

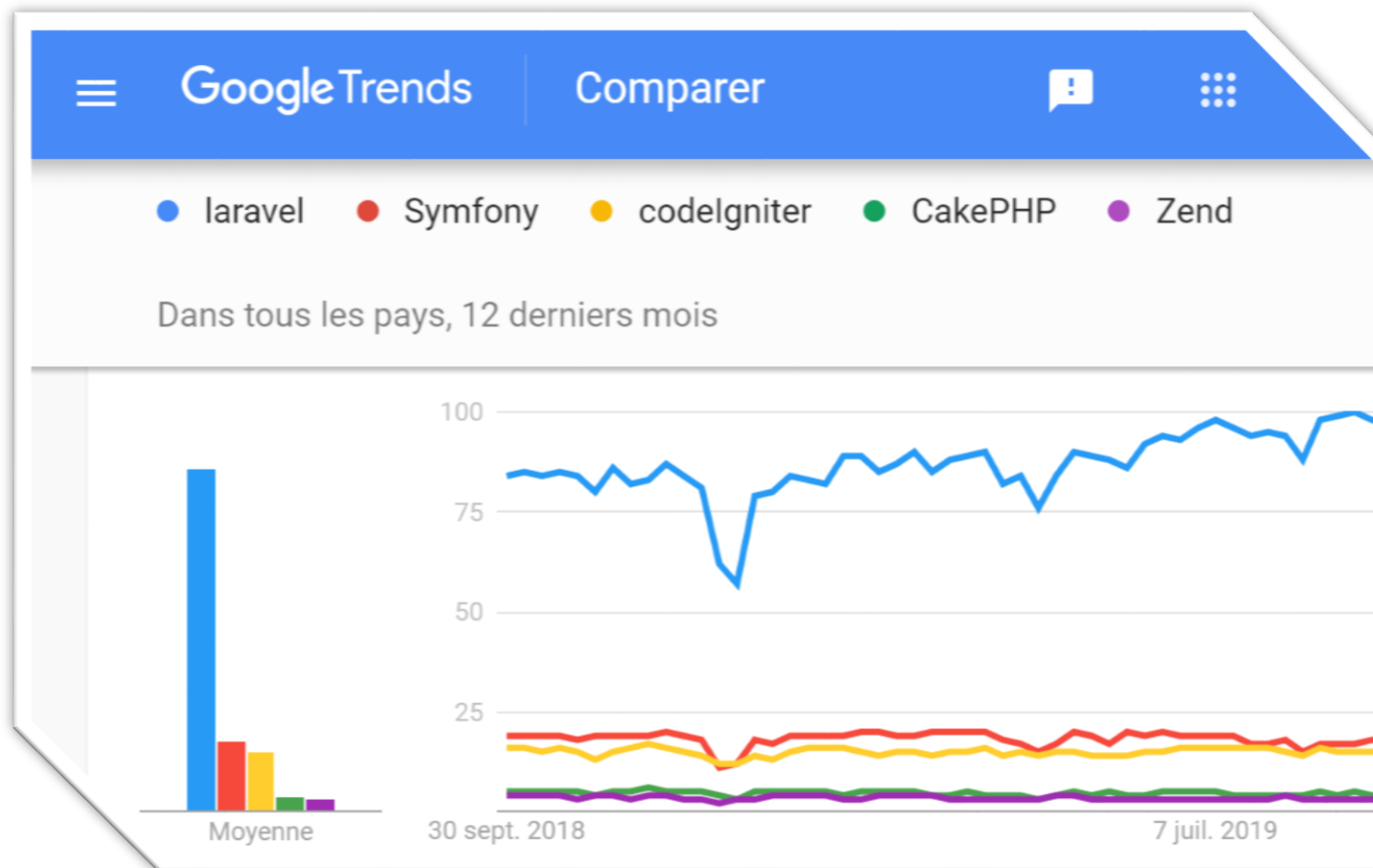
Laravel est un Framework PHP

- Il a été créé en Juin 2011 par Taylor Otwell
- Il est open source et est basé sur le Framework Symfony
- Il est écrit en PHP et est entièrement développé en Programmation Orientée Objet
- Il permet de faire des applications web respectant le modèle MVC



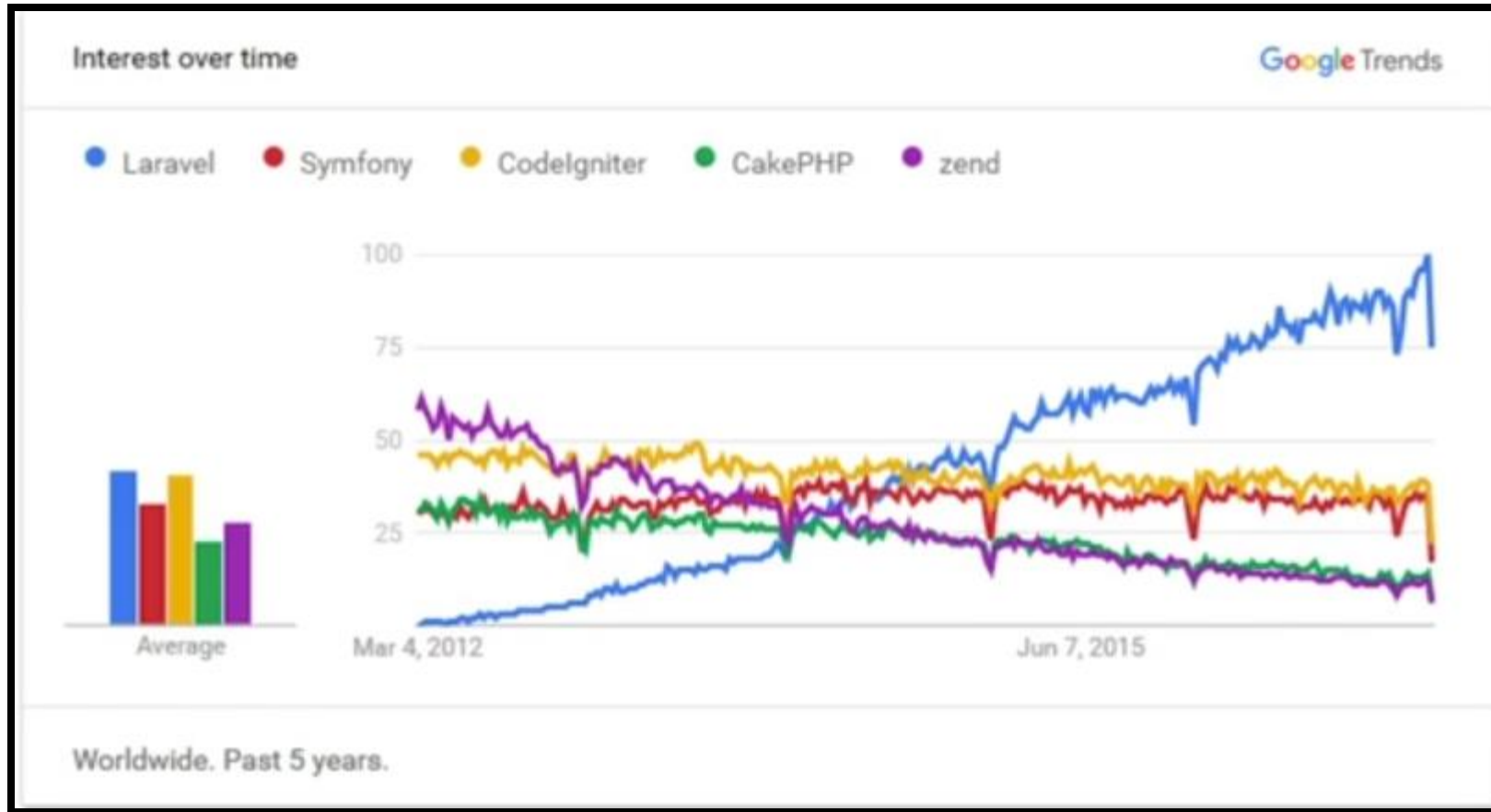
Utilisation de Laravel

Laravel est l'un des Framework PHP les plus utilisés dans le monde



Utilisation de Laravel

Il suscite de plus en plus d'intérêts auprès des développeurs



Installation de Laravel : les prérequis

Avant d'installer Laravel, il faut disposer de :

- Un interpréteur PHP (version récente)
- Un serveur de base de données
- Une extension *Mcrypt* de PHP activée
- *Composer*, un gestionnaire de dépendances



Installation de Laravel : les prérequis

Des suites logicielles « tout-en-un » existent :

- [Homestead](#) : facile à mettre en œuvre sous Linux, mais beaucoup moins conviviale avec Windows
- [Laragon](#) : de plus en plus populaire, il fonctionne sous Windows
- [WAMP](#) : largement utilisé, il fonctionne sous Windows
- Etc.



Installation de Laravel : les prérequis

Dans ce cours, nous utiliserons:

- La dernière version de WAMP (pour avoir accès d'un seul coup à MySQL et PHP)
- Composer

PS : *Ceux qui utilisent d'autres systèmes peuvent télécharger la suite logicielle qui leur sied (XAMPP ou LAMP par exemple)*



Installation de WAMP

Pour installer WAMP :

- Visitez le site <http://www.wampserver.com/>
- Téléchargez la dernière version qui correspond à votre système (32 ou 64 bits)
- Suivez les instructions d'installation



Installation de Composer

A quoi sert Composer ?

- Laravel utilise des composants d'autres sources
- Il utilise Composer pour les intégrer et pour gérer leurs dépendances
- Composer est un gestionnaire de dépendances pour PHP

En somme, quand on a besoin d'intégrer un composant logiciel dans une application, on le notifie à Composer qui va le télécharger et l'installer pour nous. Et ce, en veillant au respect de la compatibilité du composant téléchargé avec ceux existants.



Installation de Composer

La procédure d'installation varie en fonction du système

- Sous Windows, un fichier d'installation est téléchargeable [ici](#)
- Téléchargez-le et suivez la procédure d'installation
 - Quand le chemin vers PHP vous est demandé, renseignez-le
 - Pour trouver PHP, allez dans le dossier de WAMP et cherchez **php.exe** dans le dossier **bin/php/phpx.y.z** (x, y et z correspondant à la version de PHP installée)
- Vérifier que l'installation s'est bien passée en tapant, dans la console, la commande **composer -V**
 - Elle devrait afficher la version installée de Composer

PS : Pour l'installation de Composer dans d'autres systèmes, vous pouvez visiter [ce site](#) pour plus de détails



Installation de Laravel

Une fois Composer installé, vous pouvez créer un projet Laravel de plusieurs manières :

- Via l'installateur de Laravel (Laravel Installer)
- *Via la commande « create-project » de Composer*
- *Via le clonage direct du dépôt **git** de Laravel*

NB : *La première méthode est celle généralement utilisée. Si toutefois on souhaite utiliser une version bien définie de Laravel, la seconde peut être la mieux adaptée*



Installation de Laravel

Via Laravel Installer

- **Etape 1** : Téléchargez l'installateur Laravel avec Composer en tapant dans la console **composer global require laravel/installer**
- **Etape 2** : Ajoutez **%USERPROFILE%\AppData\Roaming\Composer\vendor\bin** dans la variable d'environnement **PATH**
- **Etape 3** : Créez un projet Laravel en tapant la commande **laravel new nom_application**
 - *nom_application* fait référence ici au nom du projet et correspond au dossier censé contenir le code de l'application (créé automatiquement)

NB : Le téléchargement de l'installateur se fait une seule fois. Les projets Laravel se créent uniquement avec la commande **laravel new**



Installation de Laravel

Via la commande « create-project » de Composer

- Dans la console, tapez l'une des commandes suivante :
composer create-project laravel/laravel nom_projet
composer create-project laravel/laravel nom_projet "5.8.*"
 - *nom_projet* fait référence ici au nom du projet et correspond au dossier censé contenir le code de l'application
 - Le *5.8.** de la seconde commande fait référence à la version de Laravel souhaitée. A défaut, la dernière version est téléchargée.

NB : Pour optimiser, il est préférable de préciser l'option « *--prefer-dist* » comme suit :

composer create-project --prefer-dist laravel/laravel nom_projet
composer create-project --prefer-dist laravel/laravel nom_projet "5.8.*"



Installation de Laravel

*Via le clonage du dépôt **git** de Laravel*

- **Etape 1** : Clonage du dépôt **git**
 - Tapez l'une des commandes suivantes dans la console :
git clone https://github.com/laravel/laravel.git
git clone https://github.com/laravel/laravel.git -b 5.8
 - *L'option **-b** de la seconde commande permet de choisir la version de Laravel souhaitée. A défaut, la dernière version est téléchargée.*
- **Etape 2** : Installation des dépendances
 - Se positionner dans le répertoire racine du projet et de taper la commande suivante :
composer install
- **Etape 3** : Génération de la clé de cryptage de l'application
 - Renommer le fichier **.env.example** en **.env**
 - Depuis la racine du projet, saisir la commande **php artisan key:generate**



Installation de Laravel

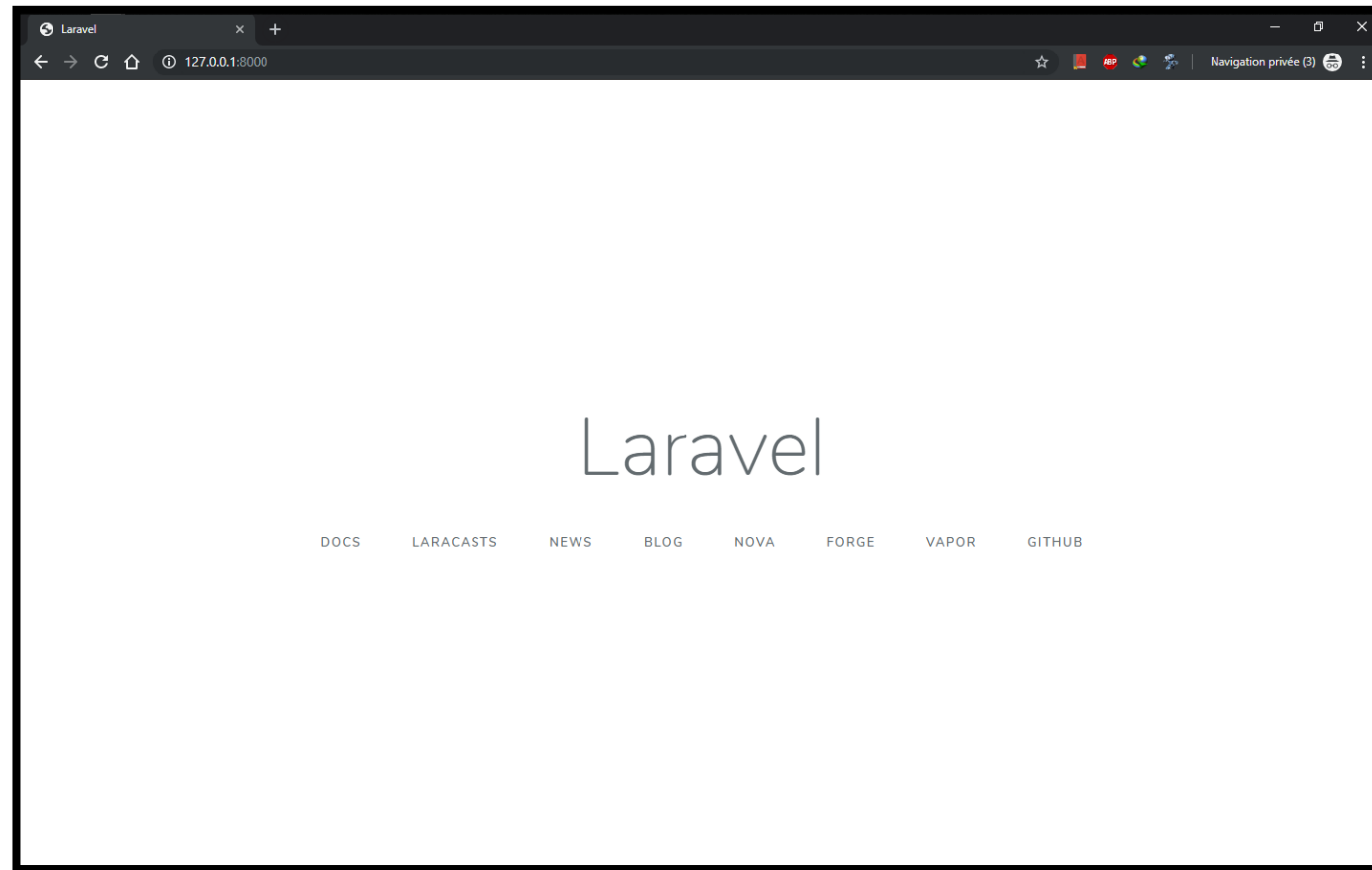
A la fin de l'installation, vous pouvez tester le projet ainsi créé

- Ouvrez la console et positionnez vous dans le répertoire du projet
- Tapez la commande ***php artisan serve***
 - Un serveur local de développement est ainsi mis à votre disposition
- Depuis votre navigateur, accédez à l'adresse <http://127.0.0.1:8000/>
 - *La page d'accueil de Laravel devrait s'afficher*



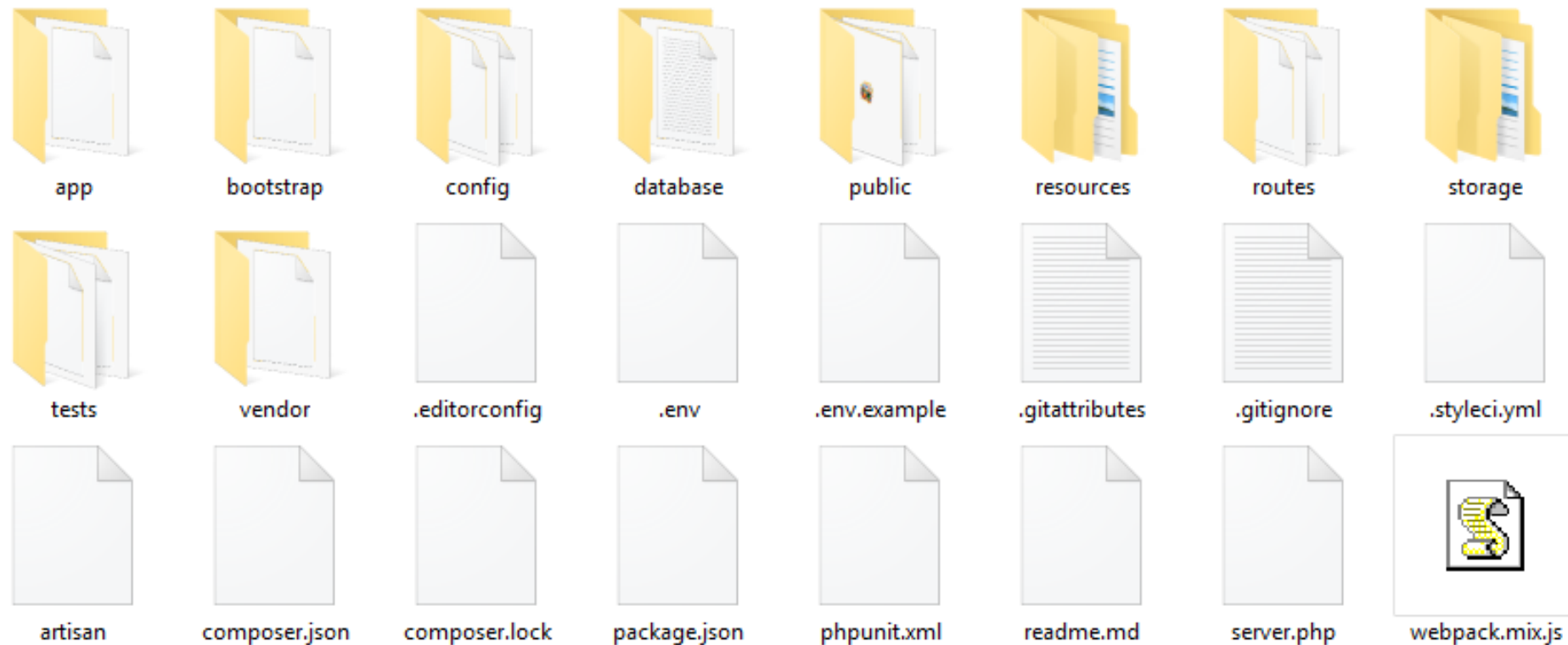
Installation de Laravel

La page d'accueil de Laravel ressemble à la figure suivante



Organisation de Laravel

Laravel contient un certain nombre de dossiers et de fichiers



Organisation de Laravel

Les dossiers dans le répertoire racine

- **app** : contient les principaux éléments de l'application
- **bootstrap** : contient les scripts d'initialisation de Laravel pour le chargement automatique des classes, la fixation de l'environnement et des chemins, et pour le démarrage de l'application
- **config** : contient tous les fichiers de configuration de l'application (authentification, cache, base de données, espaces de noms, emails, systèmes de fichiers, session, ...)
- **database** : contient les fichiers de migrations et de remplissage de la base de données
- **public** : contient tout ce qui doit être accessible publiquement (images, CSS, scripts, ...). Il est donc le seul dossier censé être accessible directement par URL
- **resources** : contient les vues ainsi que d'autres fichiers (fichiers LESS, SASS, ...)
- **routes** : contient la définition des routes de l'application
- **storage** : données temporaires de l'application (vues compilées, caches, ...)
- **tests** : contient les fichiers de tests unitaires.
- **vendor** : contient tous les composants de Laravel et de ses dépendances

Organisation de Laravel

Les principaux fichiers dans le répertoire racine

- ***artisan*** : outil accessible en ligne de commande pour des tâches de gestion
- ***composer.json*** : fichier utilisé par Composer pour assurer la gestion des dépendances
- ***phpunit.xml*** : fichier de configuration de PHPUnit (pour les tests unitaires)
- ***.env*** : fichier utilisé pour spécifier certains paramètres de Laravel



Configuration de Laravel

Il peut être nécessaire d'effectuer quelques configurations :

- Le dossier **storage** doit être accessible en écriture
- Le fichier **public/.htaccess** peut éventuellement être modifié pour définir des alias
- Le fichier **config/database.php** (ou **.env**) peut éventuellement être mise à jour pour renseigner les paramètres de connexion à la base de données
- Dans le fichier **.env**, spécifier la valeur **true** pour **APP_DEBUG**, afin d'avoir des messages plus verbeux en cas d'erreurs.



Liens utiles

Pour aller plus loin...

- <https://laravel.com/docs/5.8/installation>
- <https://getcomposer.org/doc/00-intro.md>
- <https://getcomposer.org/doc/>
- <https://getcomposer.org/doc/03-cli.md>
- <https://laracasts.com/series/laravel-from-scratch-2018/episodes/2>
- <https://openclassrooms.com/fr/courses/3613341-decouvrez-le-framework-php-laravel/3616434-installation-et-organisation>

