***Compte rendu***

***03 Avril 2020***

***NIVEAU 2 (FRAMEWORK) :***

***Module 4 : LARAVEL*** V 6.0

1. **Introduction :**

**Laravel** est un [**Framework**](https://fr.wikipedia.org/wiki/Framework)[web](https://fr.wikipedia.org/wiki/Web_application) [open-source](https://fr.wikipedia.org/wiki/Open-source) écrit en [**PHP**](https://fr.wikipedia.org/wiki/PHP), respectant le principe [**Modèle-Vue-Contrôleur**](https://fr.wikipedia.org/wiki/Mod%C3%A8le-vue-contr%C3%B4leur) et entièrement développé en [**programmation orientée objet**](https://fr.wikipedia.org/wiki/Programmation_orient%C3%A9e_objet).

1. **Installation & Structure :**

Vous devrez vous assurer que votre serveur répond aux exigences suivantes :

* PHP> = 7.2.0
* Extension PHP BCMath
* Extension PHP Ctype
* Extension PHP Fileinfo
* Extension JSON PHP
* Extension PHP Mbstring
* Extension PHP OpenSSL
* Extension PHP PDO
* Extension PHP Tokenizer
* Extension PHP XML

#### **Via Laravel Installer**

Tout d'abord, téléchargez le programme d'installation de Laravel à l'aide de Composer :

**composer global require laravel/installer**

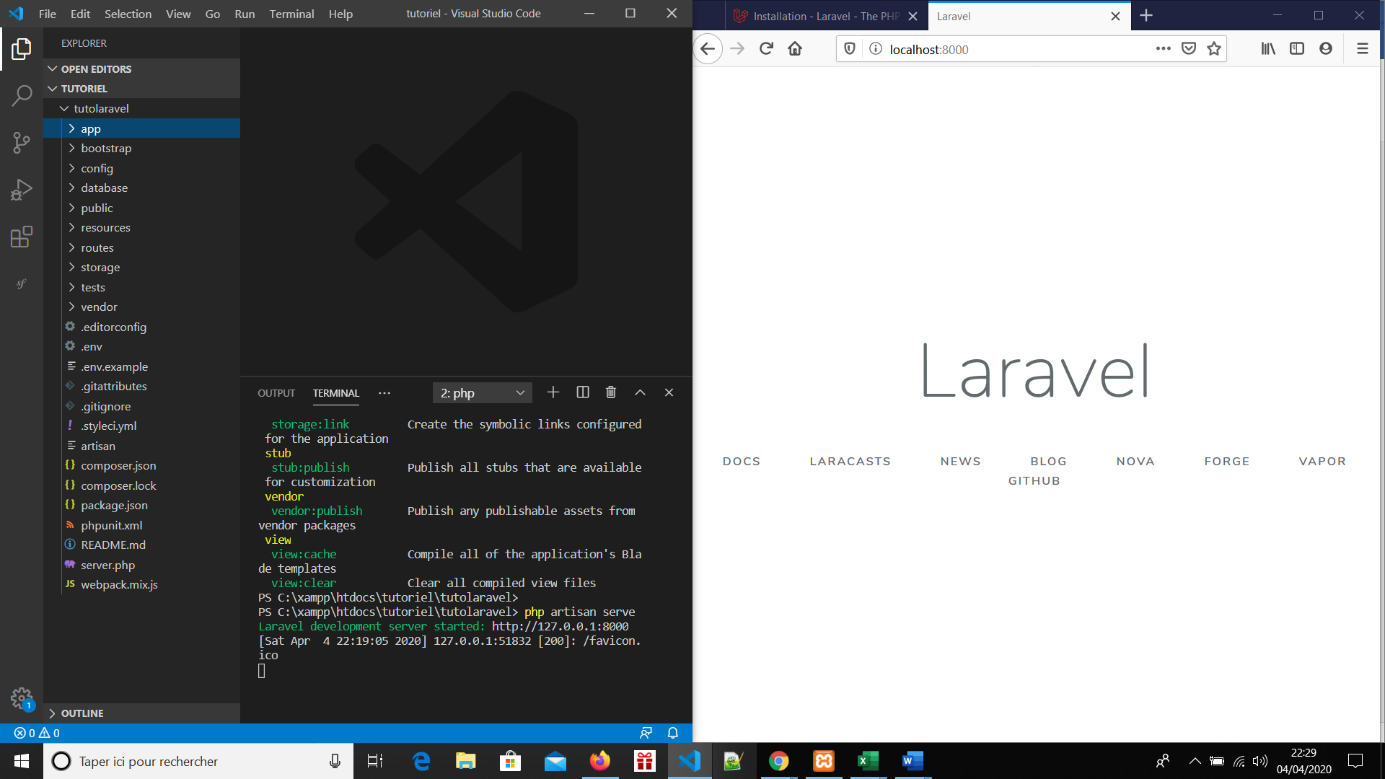
#### **Via Composer Create-Project**

Alternativement, vous pouvez également installer Laravel en émettant la commande Composer dans votre terminal : create-project

**composer create-project --prefer-dist laravel/laravel**

#### **Serveur de développement local**

Si PHP est installé localement et que vous souhaitez utiliser le serveur de développement intégré de PHP pour servir votre application, vous pouvez utiliser la serve commande Artisan. Cette commande démarrera un serveur de développement à : http://localhost:8000

**php artisan serve**

1. Les Routes / Routing :

#### **Rediriger les itinéraires :**

Si vous définissez un itinéraire qui redirige vers un autre URI, vous pouvez utiliser la méthode. Cette méthode fournit un raccourci pratique afin que vous n'ayez pas à définir un itinéraire complet ou un contrôleur pour effectuer une redirection simple : Route::redirect

**Route::redirect('/here', '/there');**

#### **Afficher les itinéraires :**

**Route::view('/welcome', 'welcome');**

#### **Paramètre requis :**

Parfois, vous devrez capturer des segments de l'URI dans votre itinéraire. Par exemple, vous devrez peut-être capturer l'ID d'un utilisateur à partir de l'URL. Vous pouvez le faire en définissant des paramètres d'itinéraire :

**Route::get('user/{id}', function ($id) {**

**return 'User '.$id;**

**});**

#### **Contraintes d’expression régulière :**

Vous pouvez contraindre le format de vos paramètres d'itinéraire à l'aide de la **where** méthode sur une instance d'itinéraire. La **where** méthode accepte le nom du paramètre et une expression régulière définissant comment le paramètre doit être contraint :

Route::get('user/{name}', function ($name) {

//

})->where('name', '[A-Za-z]+');

Route::get('user/{id}', function ($id) {

//

})->where('id', '[0-9]+');

Route::get('user/{id}/{name}', function ($id, $name) {

})->where(['id' => '[0-9]+', 'name' => '[a-z]+']);

// A revoir ce weekend

1. Les Controllers :
2. Les Middlewares :
3. Les Views Blade
4. Formulaires