Laravel

Projet Authentification Rapide

Auteur : ***Berthe Seydou***

04/07/2016

Table des matières

[Authentification rapide avec Laravel : 2](#_Toc458073206)

[1. Création de projet Laravel : 2](#_Toc458073207)

[2. Génération des composants d’authentification : 2](#_Toc458073208)

[3. Configuration du serveur mail : 3](#_Toc458073209)

[4. Personnalisation : 3](#_Toc458073210)

Prérequis : Projet Inscription Laravel. (Lien Projet « AuthRapide » [ICI](https://github.com/ArianeExtern/EtudeLaravel.git)).

# Authentification rapide avec Laravel :

Dans ce projet, il sera question de mettre en place un système d’authentification en utilisant le module d’authentification de Laravel prédéfini. Pour ce faire, on passera par les étapes suivantes :

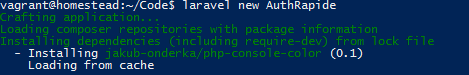
🡪Création d’un projet Laravel.

🡪Générer les composants du module d’authentification.

🡪Configurer le serveur mail.

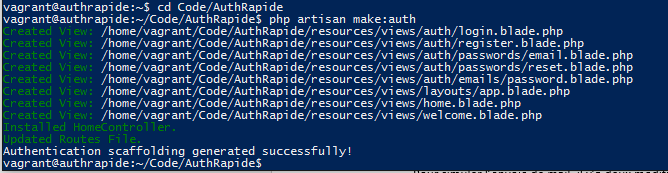
🡪Personnaliser.

# Création de projet Laravel :

Pour entamer ce projet, nous allons commencer par créer un nouveau projet Laravel nommé « **AuthRapide** », ce qui se fait comme ci-dessous :

# Génération des composants d’authentification :

Pour mettre en place ces composants, on utilise l’utilitaire en ligne de commande de Laravel « **php artisan** » qui propose une option pour faire cette tâche. On se déplace tout d’abord dans le répertoire du projet et on exécute la commande « **php artisan make:auth**». Ce qui donne l’illustration suivante :



A la suite de l’exécution de cette commande, on se retrouve avec les répertoires et fichiers générer à savoir : le répertoire qui contient les **Controller** gérant l’authentification «**app/http/Controller/Auth** » s’il n’est pas déjà présente vu qu’il est auto-générer, le répertoire qui contient les vues nécessaire pour l’authentification, l’enregistrement et mise à jour du mot de passe d’un utilisateur et il y’a aussi des fichiers jouant le rôle d’intermédiaire entre une requête et l’action associé « **MiddleWare** ». Ci-dessus on peut voir les répertoires et fichiers générer.

A ce niveau tous devra fonctionner à merveille sauf l’envoie de mail pour la réinitialisation du mot de passe. En guise de teste il suffit lance l’application sur le navigateur, ce qui nous donne une sortie :

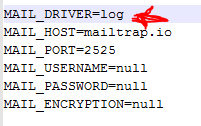


Les parties entourées en rouge permettent respectivement d’accéder au formulaire d’enregistrement et d’authentification.

# Configuration du serveur mail :

Pour simuler l’envoie de mail, il y’a deux modification à faire :

🡪Modifier le fichier « **app/.env** » pour spécifier le pilote d’envoi de mail, par défaut le pilote utilisé est « **SMTP »**, comme nous sommes en local, nous allons utiliser le fichier **log** de notre application pour l’envoie de mail. On modifie ainsi la ligne « **MAIL DRIVER** » comme suit :



NB : Cette modification est fait en guise de teste sinon en production on utilise un vrai pilote comme « **SMTP** ».

🡪On modifie aussi le fichier « **app/config/mail** » pour spécifier l’envoyeur du mail, pour ce faire on modifier la ligne « **FROM** » comme suit :



Dans les deux parties entourées en rouge, on spécifie l’adresse mail et le nom d’utilisateur, c’est nécessaire sinon, il y’aura une erreur disant que l’on ne peut pas envoyer un mail sans l’adresse d’envoyeur.

**NB : Après l’envoie de mail, il faudra voir le fichier « storage/log/laravel.log » pour voir le mail, ainsi on copie le lien du mail sur le navigateur ça va nous renvoyer vers une page de réinitialisation de mot de passe.**

# Personnalisation :

A ce niveau, nous serons tentés par personnaliser deux aspects de l’application à savoir les vues et les routes.

🡪Pour les vues il faut se rendre dans le répertoire « **ressources/views/** » qui contient toutes vues de l’authentification à savoir :

[

‘welcome.blade.php ‘ => ‘Page d’accueil de l’application ‘,

‘home.blade.php’ => ‘La page de redirection après l’authentification’,

‘auth/login.blade.php’ => ‘La page de login’,

‘auth/register.blade.php’ => ‘La page d’enregistrement’,

‘auth/emails/password.blade.php’ => ‘La vue afficher lors de la reception du mail’,

‘auth/password/email.blade.php’ => ‘Le formulaire d’envoi de mail’,

‘auth/password/reset.blade.php’ => ‘Le formulaire de réinitialisation de mot de passe’,

]

🡪Pour personnaliser les routes, il faudra essentiellement voir le fichier « **routes.php** » et aussi les controller d’authentification.

**NB : En cas d’erreur concernant la base de données lors de l’enregistrement d’un utilisateur, il faudra utiliser la commande « php artisan migrate » pour mettre à jour la base de données la base de données.**