



[←Revenir à la théorie.](#)

Laravel

1. Introduction à Laravel

Laravel est un Framework web Open-Source écrit en PHP. Il est basé sur le modèle-vue-contrôleur ([MVC](#)) et est développé en [POO](#).

C'est Taylor Otwell qui l'a écrit en juin 2011: Laravel a fêté ses 10 ans.

30% de son code est basé sur [Symfony](#): Symfony component.

Il connaît une grande popularité. Une grande communauté d'utilisateurs du Framework s'est formée. En 2016, c'était le projet PHP le mieux noté sur GitHub.

Les principales fonctionnalités de Laravel sont:

- Le routage de requête.
- Le mapping objet-relationnel (ORM):un système baptisé Eloquent.
- L'authentification.
- Les vues (avec Blade).
- La migration de base de données.
- La gestion des exceptions.
- Les tests unitaires.
- un moteur de template.
- un système d'authentification pour les connexions.
- un système de validation.
- un système de pagination.
- un système de migration pour les bases de données.
- un système d'envoi d'emails.
- un système de cache.
- un système de gestion des événements.
- un système d'autorisations.
- une gestion des sessions.

- un système de localisation.
- un système de notifications.

Laravel s'efforce de fournir une expérience de développement incroyable tout en fournissant des fonctionnalités puissantes telles qu'une injection de dépendance approfondie, une couche d'abstraction de base de données expressive, des files d'attente et des tâches planifiées, des tests unitaires et d'intégration, etc.

Que vous soyez novice en PHP ou en framework web ou que vous ayez des années d'expérience, Laravel est un framework qui peut évoluer avec vous.

2. Qu'est-ce qu'un Framework ?

D'après Wikipedia un framework informatique est un "ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un logiciel". Autrement dit une base cohérente avec des briques toutes prêtes à disposition. Il existe des frameworks pour tous les langages de programmation et en particulier pour PHP.

3. Parcours d'une requête http

Lorsque vous effectuez une requête du type <http://www.crazycupcakes.com/>

Le flux s'effectue de la sorte dans Laravel:

1. L'utilisateur demande au navigateur d'afficher la page
<http://www.crazycupcakes.com/>
2. Le browser contacte le serveur web et lui demande d'afficher la page /
3. Laravel analyse la requête (url) et va prendre la bonne route (fichier Routes/web.php). Une Route va soit contacter:
 - i. Un contrôleur: `Route::get('/', [CupcakeController::class, 'index'])`; qui va recevoir une réponse par exemple de votre base de données (Modèle) via Eloquent et chargera une vue.
 - ii. Une Vue qui sera au final une page web que l'utilisateur final verra. Elle se nomme par exemple `cupcakes.blade.php`. Elle peut recevoir la réponse d'un contrôleur.
 - iii. Une simple string: Ca affichera votre chaîne de caractère sans HTML.

ok ça ne sert pas à grand chose... 😊

iv. Un tableau JSON.

4. Analysez le framework

Pour comprendre Laravel, il faut aussi analyser ce qu'on appelle la signature des méthodes que l'on utilise. Il n'est pas possible de retenir tout de Laravel. Parfois, lire la définition d'une méthode, d'une classe nous permet de mieux comprendre ce qu'elle fait.

Apprendre Laravel, c'est tester, lire de la Doc et analyser aussi les fonctions que l'on utilise. Mais surtout, c'est de la pratique. Beaucoup de pratique.

5. Documentation

- <https://laravel.com/docs/11.x> : Site officiel.
- <https://laravel-france.com/> : Laravel France.
- <http://laravel.io> : le forum officiel.
- <https://laracasts.com> : pas mal de vidéos.

[←Revenir à la théorie.](#)