

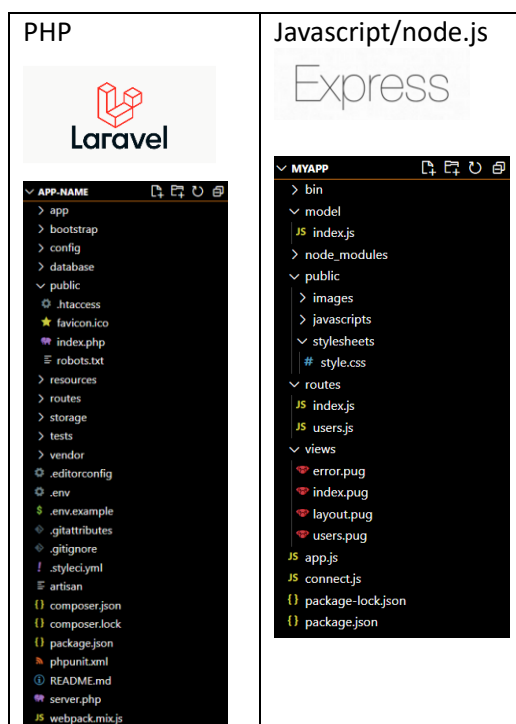
# Veille technologique

## 1. Choix du theme

### Les Frameworks

Un Framework, ou un « cadre de travail » est un ensemble cohérent de composant logiciels structurels qui sert à créer ls fondations ainsi que les grandes lignes de tout ou partie d'un logiciel, c'est-à-dire une architecture.

Exemples :



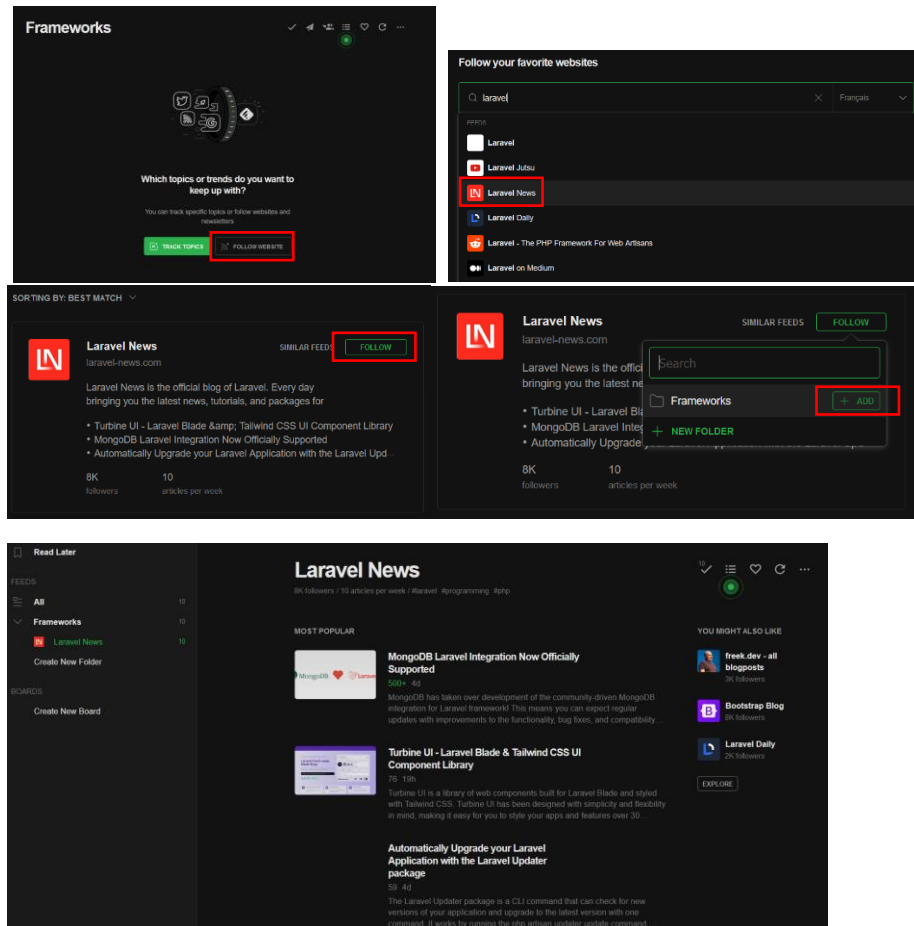
L'étude des Framework est pertinente par le fait que ceci est en accord avec le cadre de mon projet professionnel : la manière qu'on structure et développe le code évolue et s'améliore au fils du temps. Donc il est important de suivre les nouvelles technologies et mises à jour fait.

## 2. Choix et mise en place des outils

### 1. Agrégateur

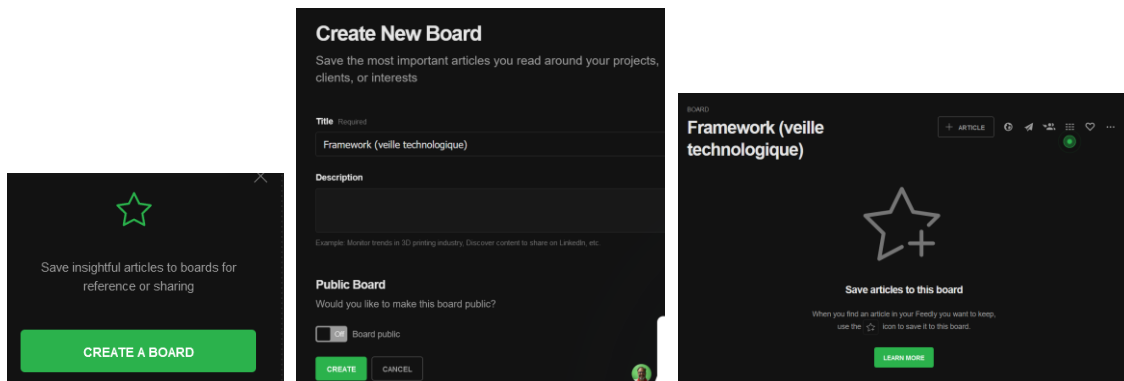
L'agrégateur que j'utilise est <https://feedly.com/>

Il faut d'abord créer un dossier, je l'ai nommé « Frameworks »

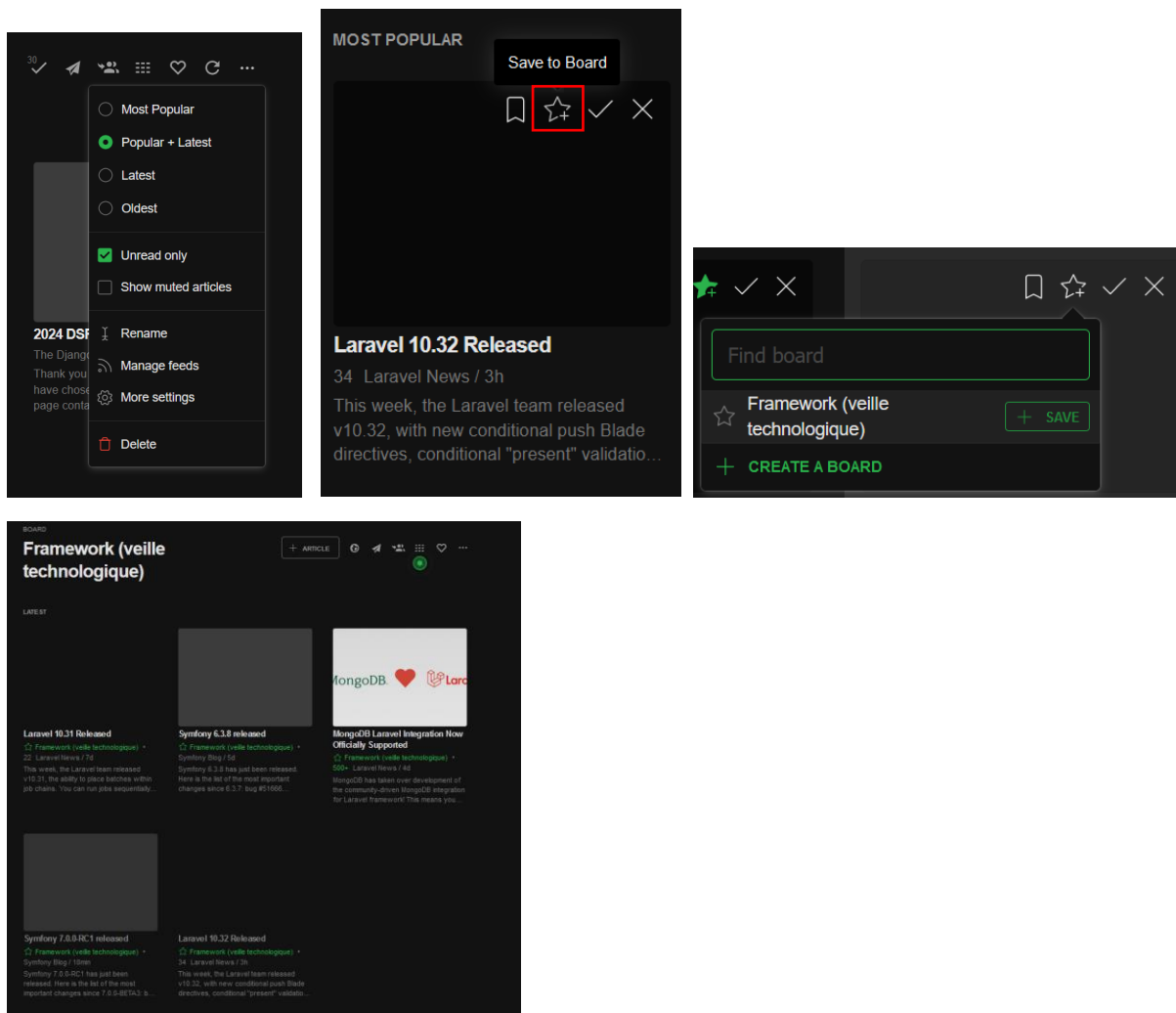


## 2. Curateur

Sur feedly, on peut créer un « board » qui permet de stocker les articles pertinents.



On peut filtrer les articles par pertinence sur Feedly.



### 3. Diffusion

La diffusion de ma veille technologique est par ce site : <https://digipad>

Je copie la date de publication et nom de l'auteur.

Puis j'effectue un résumé de l'article en quelques lignes.

Les sources que je suis sont :

Laravel news	Newsletter officielle de Laravel	<a href="https://laravel-news.com/newsletter">https://laravel-news.com/newsletter</a>
Symfony blog	Blog officiel de Symfony	<a href="http://feeds.feedburner.com/symfony/blog">http://feeds.feedburner.com/symfony/blog</a>
The django weblog	Blog officiel de django	<a href="http://www.djangoproject.com/rss/weblog/">http://www.djangoproject.com/rss/weblog/</a>
React.js blog	Blog officiel de react.js	<a href="https://react.dev/blog">https://react.dev/blog</a>
Vue.js	Blog officiel de vue.js	<a href="https://blog.vuejs.org/">https://blog.vuejs.org/</a>

J'ai choisi des blogs et newsletter car ceci va produire des articles ayant des sources fiables.

Puis je sélectionne les articles qui décrit une évolution pertinente de leurs Framework.

The screenshot shows a web application interface with a grid of article cards. The cards are for various frameworks and technologies:

- Vue.js**: Article about the new Vue.js 3.0 release, highlighting its performance improvements and the introduction of the Composition API.
- React.js**: Article about the new React.js 18 release, focusing on the new concurrent rendering mode and the new hooks.
- Next.js**: Article about the new Next.js 12 release, mentioning the new experimental features and the new default export for the API routes.
- Django**: Article about the new Django 4.0 release, discussing the new security features and the new default settings.
- Laravel**: Article about the new Laravel 9 release, highlighting the new security features and the new default settings.
- Symfony**: Article about the new Symfony 5.4 release, mentioning the new security features and the new default settings.

Each card includes a title, a date, an author, and a brief summary of the article content. The interface is clean and modern, with a dark theme.

## 4. Conclusion

Entre 2022 et 2024, les frameworks de développement web ont suivi un parcours évolutif cohérent :

Tout d'abord, une focalisation sur l'amélioration des performances a été observée. Tous les frameworks ont œuvré pour accélérer les applications. Par exemple, React 18 a introduit des fonctionnalités telles que le rendu côté serveur amélioré, visant à offrir une expérience utilisateur plus fluide.

Ensuite, une simplification des flux de travail a été mise en avant. Les frameworks ont intégré des fonctionnalités facilitant le développement. Laravel 11 a simplifié la structure des répertoires, rendant ainsi l'organisation des fichiers plus intuitive.

La sécurité des applications a également été renforcée. Les développeurs ont accordé une attention particulière à ce domaine. Symfony 6.4 a ainsi renforcé la sécurité en fournissant des informations plus claires sur les problèmes de sécurité, assurant ainsi une meilleure protection des applications.

Par la suite, une intégration de nouvelles fonctionnalités a été observée pour répondre aux besoins en constante évolution. Django 5.0 a par exemple introduit les "groupes de champs" pour simplifier le processus de rendu des formulaires, offrant ainsi une expérience de développement plus fluide et intuitive.

Enfin, les frameworks ont continué à maintenir leur compatibilité avec les dernières technologies. Ils ont ainsi assuré le support des dernières versions des langages et des normes de l'industrie. Vue 3.4 a pris en charge les dernières versions de JavaScript, garantissant une meilleure performance et une compatibilité accrue avec les autres outils et bibliothèques du secteur.