

Laravel et Nest sont deux frameworks qui permettent de réaliser des API sécurisées et robustes.

Laravel est un framework en PHP fondé sur le modèle de conception MVC (Modèle-Vue-Contrôleur), Laravel facilite le développement en fournissant une structure claire et des composants préconstruits. Il intègre Eloquent, un ORM (Object-Relational Mapping) facilitant la manipulation des bases de données en la représentant sous forme d'objets, ainsi que Blade, un moteur de templating puissant. Laravel offre également des fonctionnalités avancées telles que la gestion des migrations de base de données, l'authentification utilisateur, la gestion des tâches planifiées, et il facilite le déploiement avec son gestionnaire de dépendances Composer.

Nest, quant à lui, est un framework Node.js utilisé pour construire des applications serveur efficaces et évolutives. Nest adopte une approche modulaire avec l'utilisation de concepts tels que les modules, les contrôleurs, les services, les guards... et est basé sur le langage Typescript pour une approche plus claire et sécurisée. Nest offre également des fonctionnalités telles que l'injection de dépendances, la validation des données, la gestion des exceptions, et il facilite la création d'API.

La décision de choisir Nest pour notre projet s'est révélée judicieuse, en raison de son architecture modulaire et organisée qui facilite considérablement la gestion et le développement. La structure claire et bien définie de Nest offre une approche systématique du développement backend. Cette architecture modulaire permet une meilleure organisation du code, une maintenance simplifiée et une extensibilité accrue.

Un facteur déterminant dans le choix de Nest a été la cohérence technologique avec notre front-end, développé en Vue.js. Opter pour une technologie similaire à JavaScript a été un choix stratégique pour assurer une transition fluide entre le front-end et le back-end. Cette continuité dans le langage de programmation a considérablement accéléré le processus d'apprentissage du framework, favorisant ainsi une intégration rapide et efficace dans notre équipe de développement.

En résumé, la sélection de Nest s'est faite en raison de son architecture avancée et modulaire, tandis que la cohérence avec Vue.js a renforcé la facilité d'adoption et la synergie entre le front-end et le back-end de notre application.