

## API-KANBAN

ELFETEH  
NOUHAYLA



## **NestJS :**

NestJS est un framework Node.js qui utilise TypeScript pour développer des applications côté serveur. Il est basé sur l'architecture Angular, ce qui facilite la transition pour ceux qui ont déjà travaillé avec Angular. NestJS offre une modularité élevée grâce à l'utilisation de modules, et il intègre également des concepts tels que les intercepteurs, les filtres et les guards, fournissant ainsi un contrôle accru sur le cycle de vie des requêtes.

## **Avantages de NestJS :**

- Utilisation de TypeScript, ce qui améliore la maintenabilité du code et la détection précoce des erreurs.
- Architecture modulaire qui facilite l'organisation du code.
- Inspiré par Angular, offrant une transition fluide pour les développeurs familiers avec ce framework.

## **Inconvénients de NestJS :**

- Peut sembler trop lourd pour de petits projets.
- La courbe d'apprentissage peut être raide pour les nouveaux développeurs.

# **Laravel :**

Laravel est un framework PHP populaire qui suit le paradigme MVC (Model-View-Controller). Il est réputé pour sa simplicité et son élégance, offrant des fonctionnalités telles que l'ORM Eloquent pour interagir avec la base de données, Blade pour les vues, et Artisan pour la ligne de commande.

## **Avantages de Laravel :**

- Syntaxe expressive et élégante.
- Eloquent, un ORM puissant, facilite l'interaction avec la base de données.
- Large écosystème avec des packages prêts à l'emploi.

## **Inconvénients de Laravel :**

- Langage PHP, qui peut être perçu comme moins moderne que TypeScript.
- L'abstraction peut rendre difficile la compréhension de certains mécanismes sous-jacents.

# **Justification du choix de NestJS :**

Dans le contexte du développement d'une API pour une application Kanban Board, plusieurs critères ont conduit à la sélection de NestJS.

- **TypeScript :** NestJS utilise TypeScript comme langage de programmation, offrant des avantages significatifs en termes de sécurité et de maintenabilité du code. La typage statique de TypeScript permet de détecter les erreurs potentielles lors de la phase de développement, garantissant ainsi une meilleure qualité du code.
- **Architecture modulaire :** L'architecture modulaire de NestJS facilite l'organisation du code source. Cela est particulièrement avantageux pour un projet Kanban Board qui peut comporter plusieurs fonctionnalités distinctes telles que la gestion des listes, des cartes, des utilisateurs, etc. Chaque fonctionnalité peut être encapsulée dans un module, simplifiant ainsi la gestion du code.
- **Inspiration d'Angular :** Pour une équipe qui a déjà une expérience avec Angular, l'inspiration d'Angular dans NestJS facilite la transition et permet de capitaliser sur les compétences déjà acquises. Les développeurs Angular peuvent réutiliser certaines connaissances et structures de code, accélérant ainsi le processus de développement.
- **Support de WebSockets :** Les tableaux Kanban peuvent bénéficier de fonctionnalités en temps réel, comme la mise à jour instantanée des cartes et des listes. NestJS offre un support intégré pour les WebSockets, ce qui facilite l'implémentation de fonctionnalités en temps réel sans avoir à recourir à des bibliothèques tierces.
- **Écosystème complet :** NestJS est livré avec un écosystème complet qui comprend des modules, des intercepteurs, des filtres et d'autres fonctionnalités intégrées. Cela évite la nécessité d'ajouter de nombreuses dépendances tierces, simplifiant ainsi la gestion des dépendances du projet.
- **En conclusion,** NestJS a été choisi pour sa combinaison d'avantages, notamment la sécurité offerte par TypeScript, son architecture modulaire, son inspiration d'Angular et son support intégré des WebSockets. Ces caractéristiques répondent aux besoins spécifiques d'un projet Kanban Board, offrant une base solide pour le développement d'une API robuste et évolutive.

**Merci pour votre  
attention**

---