

# rapport veille back

## Laravel

Laravel, framework PHP, est reconnu pour sa syntaxe élégante et ses fonctionnalités conviviales. Avec un écosystème robuste, il facilite le développement rapide d'applications web grâce à des outils puissants comme Eloquent pour la gestion de la base de données et Blade pour les vues. Sa syntaxe expressive et ses fonctionnalités intégrées en font un choix populaire. Laravel suit le modèle MVC, mais sa courbe d'apprentissage peut être raide.

## Nest JS

Nest JS, framework Node.js, se distingue par sa scalabilité et sa flexibilité. Inspiré par Angular, il offre une structure modulaire adaptée à la création d'applications maintenables. Grâce à TypeScript, les développeurs bénéficient d'un typage statique. Intégrant les décorateurs et l'injection de dépendances, Nest JS favorise une progression naturelle pour les développeurs habitués à l'orienté objet. Son architecture modulaire permet une intégration transparente avec diverses bibliothèques. Nest JS s'impose comme un choix robuste dans l'écosystème Node.js.

## Mon choix

La connaissance préalable de Symfony par l'équipe facilite la transition vers Nest JS, notamment par la similarité des concepts et des principes entre Symfony et Nest JS. Contrairement à Laravel, qui, bien que reposant sur PHP, introduit une utilisation importante de classes abstraites, Nest JS maintient une cohérence avec les principes SOLID, déjà présents dans Symfony. Cette cohérence favorise une lecture plus claire du code, préservant ainsi les standards de travail d'équipe établis avec Symfony. En choisissant Nest JS, nous nous assurons de conserver une approche plus uniforme dans le développement, alignée avec les pratiques déjà établies et appréciées au sein de l'équipe.