

Neurona - Apprends, Partage, Progresse

Développeur Web & Web Mobile

Thomas PENA-BERMOND

09/01/2025

# Introduction

Neurona est une application web (SaaS) conçue pour centraliser l'apprentissage et la veille technologique des développeurs. Elle regroupe la documentation, les discussions communautaires et l'actualité tech en une seule interface.

## Pourquoi ce projet ?

En tant que développeur, je constate souvent que l'information est dispersée : documentations trop techniques, forums obsolètes ou articles éparpillés. La recherche de solutions devient chronophage.

J'ai développé Neurona pour répondre à ce problème de fragmentation. L'objectif est de fournir un "hub" de connaissances où l'IA assiste la création de cours structurés pour les juniors, tout en offrant aux seniors un espace pour partager leur expertise.

## Stack Technique

J'ai choisi la stack MERN pour sa cohérence (JavaScript/TypeScript sur l'ensemble de l'architecture) et sa modernité.

- Front-end : React avec Vite pour la rapidité. J'utilise Tailwind CSS pour l'intégration des composants (shadcn/ui) couplé à du SCSS pour le styling personnalisé modulaire.
- Back-end : Node.js et Express pour la création d'une API REST sécurisée.
- Base de données : MongoDB. Ce choix est stratégique pour stocker des documents JSON flexibles, ce qui est idéal pour gérer les contenus générés par l'IA.

## Fonctionnalités clés

Ce projet de validation de titre DWWM (Développeur Web & Web Mobile) intègre les fonctionnalités suivantes :

- Authentification sécurisée : Système complet (Inscription/Connexion) avec hachage des mots de passe (bcrypt) et gestion de session par Token JWT.
- Génération de documentation par IA : Intégration de l'API Google Gemini pour générer des cours complets sur demande. Le système récupère également des illustrations via l'API Pexels basées sur des mots-clés fournis par l'IA.
- Espace Communauté : Système de CRUD complet permettant aux membres de poster des sujets d'entraide et d'échanger.
- Veille Technologique : Agrégation et affichage d'articles récents via une API externe.

# Défis techniques rencontrés

Le développement de cette application m'a confronté à plusieurs problématiques concrètes que j'ai dû résoudre :

- Formatage des données IA : Les réponses de l'IA n'étaient pas toujours des JSON valides, ce qui faisait planter le front-end. J'ai résolu cela en utilisant la librairie Zod pour forcer un schéma de sortie strict avant de renvoyer les données au client.
- Problèmes de CORS : La communication entre le front et le back, notamment pour la gestion des cookies et des credentials, a nécessité une configuration fine des en-têtes HTTP et des origines autorisées.
- Gestion du temps (MVP) : Travaillant en autonomie complète, j'ai dû prioriser les fonctionnalités essentielles (MVP) et reporter certaines options complexes pour respecter les délais de présentation.

*Note : Ce projet n'est pas entièrement terminé. J'ai dû le réaliser en totale autonomie et en distanciel en raison de contraintes personnelles. Ce contexte particulier m'a permis de démontrer ma capacité à gérer seul un projet complet et à surmonter les difficultés techniques sans supervision et sans aide.*

Lien utile : [Neurona](#)