

# Teina Teinauri · Product Builder freelance

Outils métier, automatisation et IA appliquée · Toulouse et à distance

[teinateinauri@gmail.com](mailto:teinateinauri@gmail.com) · [linkedin.com/in/teina-automatise](https://linkedin.com/in/teina-automatise) · [github.com/Teina-max](https://github.com/Teina-max) · [portfolio-n8n.vercel.app](https://portfolio-n8n.vercel.app)

Disponible pour une mission récurrente de 2 à 3 jours par semaine. Je transforme des besoins métier parfois flous en outils internes, automatisations et prototypes IA fiables, documentés et prêts à évoluer.

## Profil

Avant de développer des produits numériques, j'ai travaillé six ans dans la logistique. Ce parcours guide ma manière de concevoir : comprendre le travail réel, réduire les frictions, livrer ce que les équipes utilisent vraiment. J'interviens en autonomie, du cadrage à la transmission, avec une communication accessible aux non-techniques.

## Compétences

| Domaine                   | Briques maîtrisées                                                 |
|---------------------------|--------------------------------------------------------------------|
| Applications métier       | TypeScript, TanStack Start, Next.js, interfaces web internes       |
| Data et backend           | Supabase, PostgreSQL, Convex, modélisation, pipelines de données   |
| Automatisation            | n8n, API REST, webhooks, synchronisations et traitements planifiés |
| Intelligence artificielle | Claude, OpenAI, Mistral, RAG, classification, OCR documentaire     |
| Qualité et livraison      | Tests automatisés, logs, monitoring, documentation, transfert      |

## Références

Missions réelles. Démonstrations et code public : répliques anonymisées des systèmes en production.

### Un assistant RAG que vous pouvez tester · démo publique open source

Posez une question sur un vrai dossier d'appel d'offres BTP : la réponse est fondée sur les pièces, chaque affirmation est citée, et si l'information n'existe pas le système le dit au lieu d'inventer.  
À tester en ligne : [rag-ao-btp-demo.vercel.app](https://rag-ao-btp-demo.vercel.app)  
Next.js · RAG · Embeddings · Garde-fous anti-hallucination · Code source : [github.com/Teina-max/rag-appels-offres-btp](https://github.com/Teina-max/rag-appels-offres-btp)

### Vision client unifiée après 16 ans d'ERP legacy · distributeur B2B

Pipeline d'import, modèle de données, dashboard de pilotage et fiche client à 360 degrés sur un historique fragmenté.

**Résultat : 16 ans d'historique centralisés, information client accessible en quelques secondes.**

TypeScript · PostgreSQL · Convex · TanStack Start · Démo interactive : [bi-distributeur-demo.vercel.app](https://bi-distributeur-demo.vercel.app)

## **Application terrain alimentée par la documentation technique · maintenance industrielle**

Recherche documentaire, RAG et classification LLM sur la documentation constructeur.

**Résultat : 92 % de précision sur le classifieur, 276 tests automatisés, procédure accessible depuis le terrain.**

TypeScript · Convex · Mistral · Python · RAG

## **Données de six magasins consolidées automatiquement · réseau retail**

Architecture de synchronisation, traitements de données et accès aux indicateurs depuis Telegram.

**Résultat : plus de 103 000 enregistrements synchronisés, 14 fonctions backend dédiées.**

Supabase · PostgreSQL · TypeScript · n8n · [Étude de cas](#)

## **Suivi offres-commandes automatisé de bout en bout · ETI industrielle**

Lecture OCR des PDF entrants, rapprochement offres-commandes, dashboard temps réel et relances automatiques.

**Résultat : plus de 20 champs extraits par document, rapprochement automatique, 40 tests de bout en bout.**

Claude Vision · Next.js · Supabase · n8n · Démo OCR : [thermopack.taiyka.net](https://thermopack.taiyka.net)

## **Modes de collaboration**

- **Renfort récurrent** : 2 à 3 jours par semaine, pour renforcer une équipe, absorber un backlog ou prendre en charge un périmètre produit.
- **Sprint de prototypage** : transformer un besoin métier ou une idée IA en démonstrateur concret, documenté et évaluable.
- **Journée d'essai payante** : tester la collaboration sur un besoin réel avant de s'engager sur une mission plus longue.

## **Méthode**

1. **Comprendre** : le processus réel, les utilisateurs concernés et leurs contraintes.
2. **Définir** : le plus petit résultat utile et ses critères de succès.
3. **Construire** : une première version testable, rapidement.
4. **Sécuriser** : les données, les erreurs et les cas limites.
5. **Transmettre** : les décisions documentées et la reprise préparée.

---

Code public et études de cas détaillées : [github.com/Teina-max](https://github.com/Teina-max) · Références clients disponibles sur demande ·  
Mis à jour en juillet 2026