

# Rapport sur l'utilisation de l'IA dans l'exercice technique – Uduma

Ibrahim Ouedraogo

2025-08-17

## Table des matières

1	Contexte	1
2	Tâches ayant bénéficié d'une assistance IA	2
3	Prompts/questions utilisés	2
4	Validation et modifications manuelles	3
5	Travail réalisé intégralement par moi-même	3
6	Conclusion	3

## 1 Contexte

Dans le cadre de l'exercice technique fourni par **Uduma**, j'ai conçu et modélisé une base de données relationnelle sous **PostgreSQL** avec extension spatiale (**PostGIS**), puis traduit cette structure en un **schéma Pydantic** afin de l'intégrer dans une application moderne FastAPI.

Mon rôle s'est inscrit dans une posture **technique sénior et managériale**, avec pour objectifs :

- Garantir la qualité du modèle conceptuel et relationnel.
- Superviser l'usage d'outils modernes (ORM Pydantic) pour aligner l'infrastructure SQL et le code applicatif.

- Optimiser le temps de réalisation en combinant **expertise personnelle** et **assistance sélective via IA**.
- 

## 2 Tâches ayant bénéficié d'une assistance IA

L'IA a été utilisée de manière **ciblée**, non pour déléguer le travail de conception, mais comme **accélérateur** dans les étapes de traduction technique et de validation.

- **Génération du fichier sql `init.sql`**, conformément à la structure de la base de données
  - **\*\*Réécriture le procédure de déploiement avec Makefile**
    - Simplification des commandes de déploiement et de gestion des environnements.
  - **Tâches de débogage** concernant certaines erreurs.
    - Identification et correction des problèmes liés aux migrations Pydantic.
    - Résolution des conflits de schéma entre PostgreSQL et Pydantic.
  - **Documentation des contraintes et relations**
    - Assistance IA pour formater les clés étrangères et contraintes d'unicité.
  - Gain de temps sur la formalisation des liens entre tables dimensionnelles et la table de faits.
- 

## 3 Prompts/questions utilisés

Quelques exemples de requêtes formulées à l'IA :

- « *Crée le schéma Pydantic équivalent à ce script PostgreSQL avec PostGIS* »
  - « *Crée le schéma Pydantic équivalent à ce script PostgreSQL avec PostGIS* »
  - « *Garde les contraintes d'unicité et les relations explicites entre tables dimensionnelles et fact table* »
  - « *Comment représenter le type PostGIS `geometry(Point,4326)` dans Pydantic* »
-

## 4 Validation et modifications manuelles

Après génération par l'IA, j'ai appliqué mon expertise pour :

- **Contrôler la conformité** : vérification des mappings Pydantic (`BigInt`, `Float`, `DateTime`) et des contraintes uniques (`@@unique`).
  - **Adapter au contexte Uduma** :
    - Préciser que les index SQL ne sont pas natifs dans Pydantic mais peuvent être intégrés dans les migrations.
    - Renforcer la lisibilité des relations dans le schéma.
  - **Vérification métier** : alignement du modèle généré avec la logique attendue (points de mesure, relevés, statuts, enregistreurs).
- 

## 5 Travail réalisé intégralement par moi-même

- La **conception de l'architecture du système** et le choix des outils à utiliser pour l'exercice.
  - La **conception globale de la base** et son alignement avec les besoins du cas d'usage Uduma.
  - La **vérification technique** de la cohérence SQL/Pydantic (notamment sur les types et relations).
  - La **documentation** de l'usage IA pour garantir transparence et méthodologie.
- 

## 6 Conclusion

L'utilisation de l'IA dans ce travail a été **stratégique et limitée** :

un levier de productivité sur la traduction syntaxique et la génération de code ORM, tout en laissant la **conception, la validation et la contextualisation** entièrement sous ma responsabilité.

Cette approche illustre mon rôle de **senior confirmé** :

- Déléguer intelligemment certaines tâches automatisables à l'IA.
- Garder la maîtrise sur l'architecture, les choix techniques et la qualité finale.
- Produire un livrable robuste, aligné au besoin, et documenté de manière transparente.