

Étude stratégique pour la conception d'un chatbot

Data & IA industriel (Sales Assistant)

Sié Hans OUATTARA

1 Travail du 04/02/26

1.1 Questions clés pour la conception du chatbot

1.1.1

subsection Produits et données de référence

04/02/26

Afin de garantir la pertinence et la fiabilité des réponses fournies par le chatbot, il est essentiel d'identifier les sources de données produits officielles :

- Où sont stockées les informations produits (base de données, fichiers Excel, ERP) ?
- Quelles données sont considérées comme fiables et à jour (stock, disponibilité, description, usage) ?
- Existe-t-il une référence produit unique (ID, SKU, code interne) ?
- À quelle fréquence les données sont-elles mises à jour et par qui ?

Principe clé : les réponses du chatbot doivent toujours s'appuyer sur les sources officielles de l'entreprise.

1.1.2 Disponibilité et stock

La gestion du stock constitue un point critique afin d'éviter toute hallucination du modèle :

- Les données de stock sont-elles en temps réel ou mises à jour périodiquement ?
- Existe-t-il une API ou des requêtes SQL pour y accéder ?
- Peut-on distinguer le stock réel, réservé et les ruptures temporaires ?
- Quelle est la source de vérité officielle ?

1.1.3 Documentation produit

Pour alimenter un système de type RAG (Retrieval Augmented Generation), il est nécessaire d'identifier la documentation exploitable :

- Où sont stockées les fiches techniques et notices ?
- Sous quel format (PDF, Word, Wiki) ?
- Les documents sont-ils structurés et multilingues ?
- Existe-t-il des documents confidentiels à exclure ?

1.1.4 Utilisateurs du chatbot

L'efficacité du chatbot dépend fortement de son public cible :

- Qui utilisera le chatbot (employés, techniciens, support, clients) ?
- Quels sont les cas d'usage les plus fréquents ?
- Le chatbot doit-il uniquement informer ou également déclencher des actions ?
- Langue principale et niveau technique des utilisateurs.

1.1.5 Mise à jour et évolution

Une vision long terme est indispensable :

- Mise à jour manuelle ou automatisée ?
- Qui peut ajouter ou modifier les produits et documents ?
- Le chatbot peut-il fournir un lien vers la source de l'information ?

1.1.6 Sécurité et droits d'accès

La sécurité est un prérequis non négociable :

- Authentification requise ou non ?
- Gestion de rôles et permissions ?
- Données sensibles à masquer selon le profil ?
- Conservation d'un historique des requêtes ?

1.1.7 Contraintes techniques

Avant toute implémentation, les contraintes suivantes doivent être clarifiées :

- Cloud autorisé ou non ?
- Utilisation d'un LLM externe ?
- Données autorisées à sortir du SI ?
- Contraintes de performance et environnement de déploiement.

1.1.8 Question stratégique

Comment l'entreprise jugera-t-elle que le chatbot est une réussite à la fin du projet ou du stage ? Cette question permet de définir clairement le périmètre, les attentes réelles et les critères d'évaluation.

1.2 Étude des modèles LLM

1.2.1 Comparaison globale

Critère	Modèles Payants	Open Source Auto-hébergé
Lancement	Quasi immédiat	Lent et coûteux
Coût	Par message	Coût fixe serveur
Souveraineté	Faible	Totale
Complexité	Très simple	Élevée (infra + équipe)

1.2.2 Tableau comparatif des modèles LLM (version LaTeX)

Catégorie	Modèle	Type	Atout "Vente"	Souveraineté	Fait critique
Le plus "Humain"	Claude 3.5 Sonnet	Payant (API)	Rédaction naturelle, empathique, évite le ton robotique	Faible (Cloud US)	Parfois trop prudent, refuse certaines comparaisons
Le standard professionnel	GPT-4o	Payant (API)	Très polyvalent, excellent pour l'analyse d'images clients	Moyenne (via Azure France)	Peut devenir trop bavard si mal configuré
Le champion européen	Mistral Large 2	Souverain (Open Source)	Excellente compréhension des nuances du marché français	Maximale (auto-hébergé)	Moins performant sur les tâches liées aux images
Meilleur rapport qualité/prix	Llama 3.3 (70B)	Souverain (Open Source)	Raisonnement avancé à coût réduit	Maximale (auto-hébergé)	Nécessite une infrastructure GPU puissante
La vitesse pure	GPT-4o mini	Payant (API)	Réponses instantanées pour les questions simples (stock, prix)	Moyenne (via Azure)	Moins adapté aux conseils complexes

1.2.3 Tableau de décision – Choix du modèle pour WordPress

Priorité principale	Modèle recommandé	Méthode d'installation
Lancer rapidement et tester le concept	GPT-4o mini	Plugin AI Engine avec clé API OpenAI
Fournir un conseil produit expert et humain	Claude 3.5 Sonnet	Plugin WPBot avec API Anthropic (AWS Bedrock)
Souveraineté totale des données	Mistral Large 2	Serveur privé (VPS GPU) avec API locale connectée à WordPress
Budget maîtrisé et fort volume de requêtes	Llama 3.3 (70B)	Hébergement chez un provider spécialisé (Groq, DeepInfra)

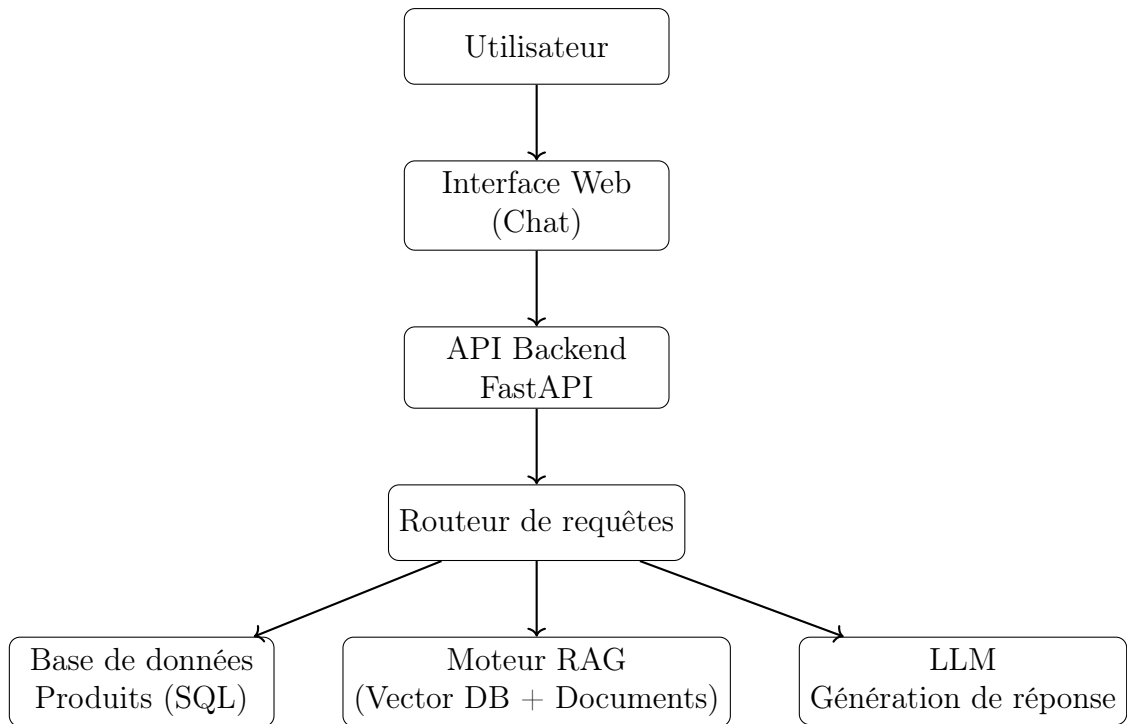


FIGURE 1 – Architecture fonctionnelle du chatbot Data & IA industriel

1.2.4 archi

1.3 Intégration WordPress

Pour un site WordPress orienté vente, l'enjeu principal n'est pas uniquement le modèle, mais l'accès aux données produits. Des plugins professionnels tels que *AI Engine* ou *WPBot* permettent de connecter le LLM au catalogue WooCommerce et aux pages produits, tout en laissant le choix entre API externe ou serveur souverain.

2 Travail du 05/04/26

Dossier / Fichier	Description
CHATBOT-SALES-PRO/	Racine du projet du chatbot « Sales Assistant ».
data/	Contient les données sources utilisées par le chatbot.
products.json	Catalogue produit structuré (références, descriptions, disponibilité).
company_info.txt	Informations générales de l'entreprise (SAV, livraison, politique commerciale).
core/	Cœur logique et intelligence artificielle du chatbot.
__init__.py	Initialisation du module Python.
loader.py	Chargement, nettoyage et découpage des données.
rag_engine.py	Moteur RAG (recherche vectorielle et contextualisation).
database.py	Gestion de la base vectorielle (ex. ChromaDB).
api/	Couche d'exposition du chatbot vers l'extérieur.
main.py	Serveur FastAPI recevant les requêtes du site WordPress.
config/	Configuration centralisée de l'application.
settings.py	Paramètres globaux (LLM, chemins, seuils, modes).
scripts/	Scripts utilitaires et tâches automatisées.
ingest_data.py	Indexation et mise à jour de la mémoire du chatbot.
tests/	Tests fonctionnels et validation des réponses.
test_responses.py	Vérification de la cohérence et de la qualité des réponses.
logs/	Journaux d'exécution pour le suivi et le débogage.
.env	Variables d'environnement et clés API sensibles.
requirements.txt	Liste des dépendances Python du projet.
README.md	Documentation générale et instructions d'utilisation.

TABLE 3 – Structure de dossiers du projet chatbot « Sales Assistant »