Architecture du système NeoSanté RAG

1. Vue d'ensemble du système

La plateforme NeoSanté RAG est une application web de télémédecine fondée sur une architecture microservices distribuée. Elle intègre des services d'authentification, d'analyse multimodale (texte et image), de visioconférence et d'intelligence artificielle. L'objectif est de fournir une infrastructure sécurisée, scalable et orientée IA pour assister les médecins et les patients dans les processus de diagnostic et de suivi clinique.

2. Composants fonctionnels principaux

- auth-svc (Authentification & Sécurité) : Gère l'authentification des utilisateurs (patients, médecins, administrateurs) via JWT et OAuth2. Implémente un modèle RBAC (Role-Based Access Control) et assure la conformité RGPD/HIPAA.
- API Gateway : Point d'entrée unique pour toutes les requêtes HTTP(S). Réalise le routage, la validation des tokens et la gestion des quotas de requêtes.
- patient-svc (Données Patient) : Service responsable du stockage, de la gestion et de la consultation des profils et historiques patients. Utilise PostgreSQL et l'extension pgvector pour les données vectorielles.
- document-ingest-svc : Assure l'ingestion et le traitement des documents médicaux. Effectue la reconnaissance optique de caractères (OCR) à partir de fichiers PDF et images via Tesseract/PyMuPDF4LLM.

- RAG + Analyse multimodale : Cœur de l'intelligence artificielle. Met en œuvre le paradigme Retrieval-Augmented Generation (RAG) pour générer des résumés et recommandations à partir des données médicales indexées.
- chat-svc : Service de gestion des conversations IA ↔ utilisateur. Utilise WebSocket pour une communication bidirectionnelle en temps réel.
- teleconsult-svc : Fournit la téléconsultation vidéo en temps réel entre patients et médecins via WebRTC.
- notif-svc : Gère les notifications automatiques (rendez-vous, rappels, prescriptions) via WebPush et e-mail SMTP.

3. Infrastructure et déploiement

Les microservices sont conteneurisés via Docker et orchestrés sous Kubernetes. Les environnements de développement, de staging et de production sont déployés sur des plateformes cloud (AWS, Azure ou GCP). Les pipelines CI/CD sont automatisés avec GitHub Actions. Le stockage des fichiers est géré par MinIO/S3.

4. Observabilité et supervision

L'observabilité du système repose sur un ensemble d'outils intégrés : Grafana pour la visualisation des métriques, Loki pour la centralisation des logs et OpenTelemetry pour le traçage distribué. Ces outils garantissent la surveillance continue, la résilience et la maintenance proactive de la plateforme.

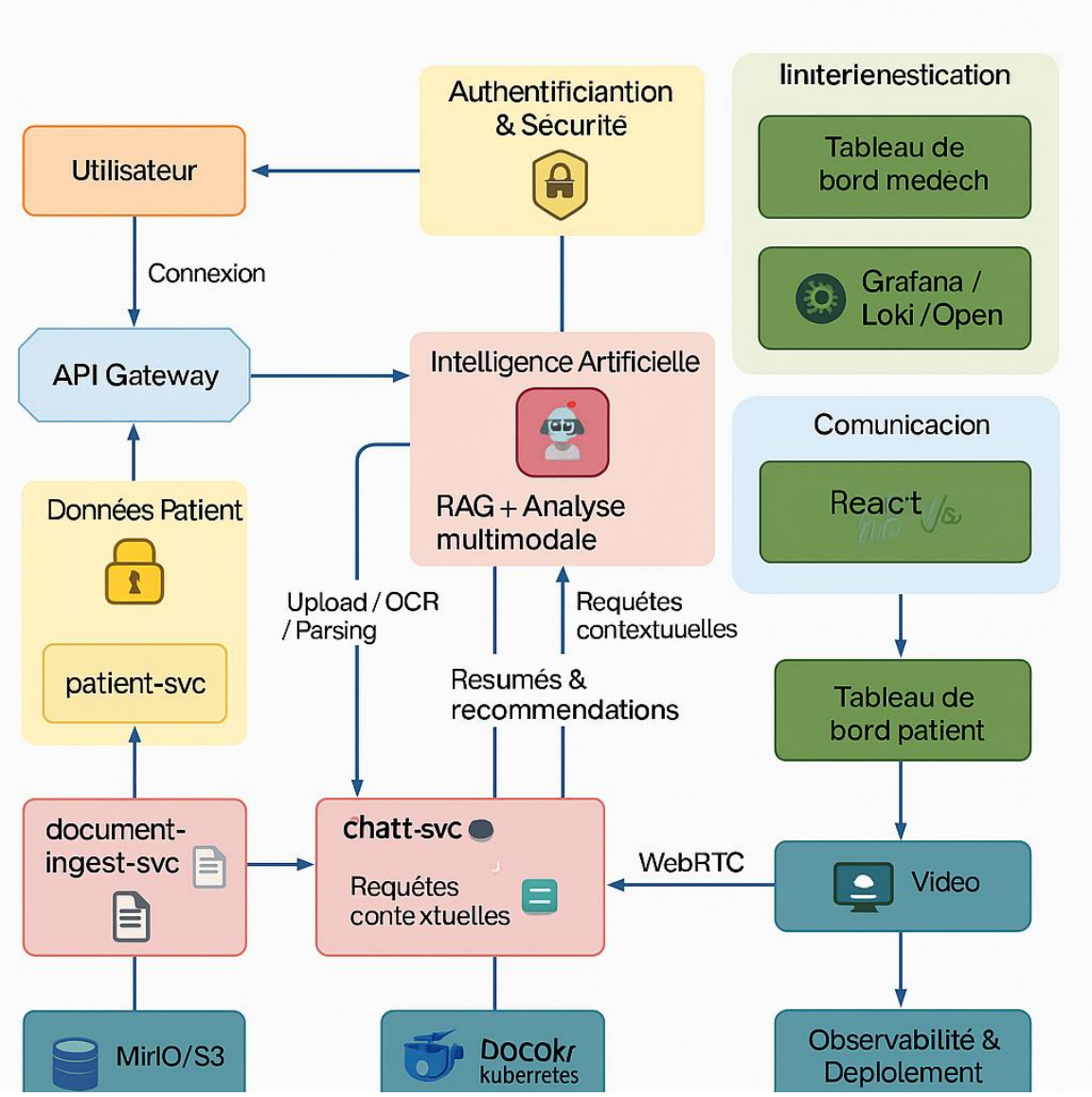
5. Interfaces utilisateurs

L'interface front-end est développée avec React (Vite). Elle comprend deux tableaux de bord : le Tableau de bord patient (consultations, prescriptions, notifications) et le Tableau de bord médecin (suivi des patients, synthèses IA,

indicateurs cliniques). La communication avec le backend s'effectue exclusivement via l'API Gateway, garantissant la sécurité et l'isolation des services.

6. Schéma d'architecture globale

Le schéma ci-dessous illustre l'architecture microservices et les flux de communication du système NeoSanté RAG :



Architecture globale du système NeoSanté RAG

7. Flux d’exécution global

1. L’utilisateur s’authentifie via le service auth-svc et reçoit un token JWT.
2. Toutes les requêtes passent par l’API Gateway pour validation et routage.
3. Les documents médicaux sont traités et indexés via document-ingest-svc et vector-svc.
4. Le moteur RAG produit des réponses contextualisées transmises au chat-svc.
5. Les consultations vidéo sont gérées par teleconsult-svc (WebRTC).
6. Les notifications sont émises par notif-svc.
7. Les performances et logs sont suivis via Grafana et Loki.