

Page d'accueil / Liste des conversations

Fonctionnalités principales :

- **Liste des conversations**
 - Affiche toutes les conversations existantes, triées par date de dernière modification (optionnel).
- **Bouton "Nouvelle conversation"**
 - Crée une nouvelle entrée dans la table `conversations`
 - Redirige immédiatement vers la page de cette conversation
 - La nouvelle conversation est ajoutée dynamiquement à la liste
- **Bouton "Supprimer"**
 - Supprime une conversation de la base (`DELETE FROM conversations`)
 - Supprime en cascade tous les messages associés (`ON DELETE CASCADE`)
 - Met à jour automatiquement la liste sans recharge
- **Clic sur une conversation**
 - Redirige vers la page de conversation correspondante
 - Charge l'historique des messages et les métadonnées (LLM, RAG actifs, etc.)

Page de gestion des RAG (Retrieval-Augmented Generation)

Fonctionnalités principales :

- **Liste des RAG**
 - Affiche pour chaque corpus :
 - Nom du RAG
 - Nombre de documents
 - Date de création
- **Bouton "Créer un RAG"**
 1. Ouvre un sélecteur de dossier
 2. Lit tous les fichiers PDF
 3. Extrait le texte de chaque document
 4. Découpe le texte en *chunks* cohérents
 5. Génère les *embeddings*
 6. Stocke dans la base vectorielle
 7. Crée une entrée dans la table `rag_corpus`
 8. Affiche une notification de succès
- **Bouton "Supprimer"**
 - Supprime l'entrée dans `rag_corpus`
 - Supprime en cascade :
 - Les documents (`documents`)
 - Les chunks et embeddings associés
 - Rafraîchit automatiquement la liste
- **Bouton "Mettre à jour"**
 - Ouvre un dialogue avec deux options :
 - **Ajouter des PDF** : permet de sélectionner de nouveaux fichiers à inclure
 - **Retirer des PDF** : propose une liste avec cases à cocher pour retirer certains documents

Page de gestion des modèles LLM

Création / configuration d'un modèle :

1. L'utilisateur choisit un **provider** : OpenAI, Anthropic, Cohere, ou Local
 2. Il saisit les paramètres :
 - o Nom personnalisé (ex : "*Claude Production*")
 - o Modèle exact (ex : "*claude-3-sonnet*")
 - o Clé API (si nécessaire)
 - o URL API personnalisée (optionnelle pour LM Studio ou proxy local)
 - o Paramètres avancés : `temperature`, `max_tokens`, `top_p`, etc.
-

Page de conversation

Interface principale :

- **Dropdown "Modèle LLM"**
 - o Liste toutes les configurations présentes dans `llm_configs`
 - o La sélection met à jour `conversation.llm_config_id`
 - o Les messages suivants utiliseront ce LLM
 - **Cases à cocher "RAG actifs"**
 - o Liste tous les corpus de `rag_corpus`
 - o Les choix sont enregistrés dans `conversation_context.active_rags`
 - o Seuls les RAG cochés sont utilisés lors des recherches sémantiques
 - **Cases à cocher "Notes actives"** (cf. section suivante)
 - o Active ou désactive des notes personnelles pour le contexte IA
 - **Zone de chat**
 - o Chaque message utilisateur :
 1. Est sauvegardé dans `messages`
 2. Déclenche une recherche vectorielle sur les RAG et Notes cochés
 3. Construit le prompt avec le contexte pertinent
 4. Envoie le tout au LLM sélectionné
 5. Affiche la réponse
 6. Sauvegarde la réponse dans `messages`
-

Page de gestion des Notes personnelles (nouvelle fonctionnalité)

Objectif :

Créer un espace où l'utilisateur peut écrire, modifier, organiser et réutiliser ses propres contenus (résumés, rappels, scripts, etc.) comme sources contextuelles à activer/désactiver par conversation.

Fonctionnalités proposées :

- **Liste des notes**
 - Affiche :
 - Titre de la note
 - Extrait (début du contenu)
 - Date de création / modification
- **Bouton "Nouvelle note"**
 - Ouvre un éditeur (texte libre, markdown optionnel)
 - Permet de saisir un titre et un contenu
 - À la sauvegarde :
 1. Crée une entrée dans `notes`
 2. Découpe le texte en *chunks*
 3. Crée les embeddings
 4. Stocke dans la base vectorielle
 5. Affiche un message de confirmation
- **Bouton "Modifier"**
 - Ouvre la note en mode édition
 - Si le texte est modifié, les embeddings sont régénérés
- **Bouton "Supprimer"**
 - Supprime l'entrée `notes`
 - Supprime tous les chunks et embeddings associés

Intégration dans les conversations :

- Dans la page de conversation, une section "**Notes actives**"
 - Liste toutes les notes personnelles existantes
 - Cases à cocher par note
 - Sauvegarde dans `conversation_context.active_notes`
 - Les notes cochées sont utilisées dans la recherche contextuelle avec les RAG

Améliorations UX/UI suggérées

- **Tri et recherche** dans les listes (conversations, notes, RAG)
- **Tags** ou catégories pour organiser les RAG et notes
- **Affichage contextuel** : survol d'un RAG ou note → aperçu rapide du contenu
- **Éditeur markdown** pour les notes
- **Historique de configuration LLM par conversation** (optionnel, utile pour debug)