

CH

Architecture

1. Backend (FastAPI) :

- Gère les requêtes utilisateur via des points de terminaison REST API.
- Charge un fichier JSON contenant des FAQ pour répondre aux questions courantes.
- Utilise `fuzzywuzzy` pour trouver des correspondances entre la question de l'utilisateur et les données existantes.
- Si aucune correspondance n'est trouvée, il utilise un modèle de génération de texte (`openai-gpt`) pour générer une réponse.

2. Frontend (HTML/CSS/JavaScript) :

- Une interface utilisateur minimaliste.
- Une fenêtre de chat peut être activée/désactivée via une icône.
- Les questions/réponses sont affichées dans un format de discussion.
- Les données sont envoyées au backend via des appels AJAX.

3. Dataset JSON :

- Contient les questions/réponses et les catégorise pour faciliter l'intégration future.

Fonctionnement Étape par Étape

1. Frontend :

- L'utilisateur pose une question via l'interface.
- La question est envoyée à l'API via une requête POST.
- Les suggestions préremplies permettent de poser rapidement des questions courantes.

2. Backend :

- La requête est reçue et analysée.
- La fonction `find_best_match` recherche une correspondance dans le dataset local (via `fuzzywuzzy`).

- Si une correspondance est trouvée, la réponse correspondante est retournée.
- Si aucune correspondance n'est trouvée, le modèle GPT génère une réponse.
- La réponse est envoyée au frontend.

3. **Affichage :**

- La réponse du backend est affichée dans la fenêtre de discussion.
-