

INFO834 - MiniProjet Redis & MongoDB

Charlotte - Louna - Mathieu

[Dépôt GitHub](#)

Ce projet vise à créer une application de chat à l'aide de base de donnée NoSQL, ici Redis et MongoDB.

Structure

Le proje est structuré en 2 parties :

- le frontend réalisé en html/css et java script contenant:
 - Les pages html (accueil, connexion, inscription et conversation)
 - Leurs css respectifs + un fichier de style global
 - Les scripts java script propre à chaque page (portant le nom de celle-ci) + un script `fonction.js` qui regroupe toutes le fonctions génériques ou utilisées plusieurs fois
 - Un répertoire pour stocker les images
- le backend codé avec python les fichiers suivants :
 - **init.py** : permet de faire les packages
 - **mongodb_api.py** : défini les fonctions nécessaires liées aux informations stockées dans mongoDB
 - **redis_api.py** : défini les fonctions nécessaires liées aux informations stockées dans redis
 - **api.py** : défini les routes qui vont être appelées par le frontend
 - **.env** : défini l'environnement

Comme dit précédemment, les informations sont stockées dans 2 bases de données. MongoDB stocke tous les messages envoyés sous la forme :

- `{"text": "Bonjour Monde", "sender": "Charlotte", "timestamp": "2025-04-16", "reciever": "Louna"}`
-

Tandis que Redis stocke toutes les informations liées aux utilisateurs et à leur connexion.

Fonctionnalité

L'application *ChatOQP* permet de se connecter ou de créer un compte si on n'en possède pas encore un.

Une fois connecté, vous arrivez sur la page accueil où vous pouvez voir le chat global et envoyez des messages. Sur le côté se trouve quelques statistiques tels que les utilisateurs en ligne, l'utilisateur qui envoie le plus de message et celui qui en reçoit le plus. Un autre onglet gère le chat privé qui permet de converser avec un utilisateur spécifique qu'on peut choisir via une liste sur le côté. Dans cet onglet, on peut également voir les anciennes conversations (comme une vraie messagerie).

Une fois vos tâches finies, vous pouvez vous déconnecter.

Amélioration

Le sujet étant large de nombreuses améliorations sont possibles. Nous pensons que ces fonctionnalités supplémentaires pourraient être appréciable:

- Le hashage du mot de passe pour plus de sécurité
- L'ajout d'une barre de recherche pour trouver un utilisateur avec qui converser plutôt qu'une simple liste qui peut vite devenir peu pratique si le nombre d'utilisateur devient important
- Trouver une façon de déconnecter proprement l'utilisateur si l'onglet est fermé ou inactif depuis trop longtemps. Nous avions une fonctionnalité qui le faisait au début mais lorsque l'on changeait de page, cela réinitialisait la SessionStorage donc nous l'avons enlevé et n'avons pas encore trouvé de solution satisfaisante
- Eventuellement rajouter d'autres statistiques