

Application desktop de Chat basée sur WebSocket

L'objectif est de comprendre le fonctionnement du protocole de communication **webSocket**, dans le cadre d'une application de Chat reposant sur le framework FastAPI.

Backend

A l'origine, une application minimaliste qui permet la connexion/déconnexion des utilisateurs et la diffusion instantanée de leurs échanges.

Phases de développement:

- App basique et tests de connexion via [Simple Websocket Client](#)
- Connexion par user_id et structuration des datas {message: str, to_id: int, from_id: int|None = None}
- Identification par login/pwd et cookie
- Sécurisation wss et cryptage des données
- Mise à disposition sur serveur pour diffusion globale de l'api
- Gestion de rooms

[Dépôt Backend](#)

Frontend

Application desktop destinée à être utilisée sous Ubuntu et Windows afin de tester la diffusion d'app multi-plateforme. Elle repose sur CustomTkinter basé sur Tkinter (python).

Phases de développement:

- App basique et tests de communication avec des onglets [Simple Websocket Client](#)
- Ajout d'une fenêtre initiale de connexion (credentials)
- Ajout d'un AppManager pour centraliser la gestion des transitions entre fenêtres et éviter les importations circulaires.
- Gestion des cookies
- Utilisation d'un protocole sécurisé de communication (wss)
- Optimisation UI/UX
 - Mode light/dark
 - Optimisation UI, dispositions des outils (connect/disconnect, search, etc...)
- Prod > exécutables Linux et Windows

[Dépôt Frontend](#)