

# Pente : Lobby

D.Moreaux

9 octobre 2024

## 1 But de l'étape

La gestion de la connexion et du lobby sera implémentée. Comme il n'y a pas encore de DB, la connexion sera acceptée si le mot de passe est "ok" et refusée si le mot de passe est différent. On peut également accepter le mot de passe "new" si on veut préparer le support à l'auto-crédation de comptes.

De la même manière, les statistiques du joueur seront initialisées avec des valeurs identifiables (1 victoire, 2 défaites, 3 parties jouées, score de 1000).

Côté client, on utilisera la fonction `input()` pour effectuer les entrées et on affichera à l'aide de la fonction `print()`. Cela sera remplacé par une interface GUI dans le projet suivant.

## 2 Serveur

Le serveur implémentera les packets liés à l'authentification et au lobby (lister les parties, créer une partie, rejoindre une partie et autres).

Pour le moment, le test de mot de passe sera remplacé par un simple test par rapport à des chaînes statiques. Cela permet de limiter au maximum le code qui devra être changé plus tard (lors de l'ajout de la DB) tout en permettant de tester la connexion.

On peut gérer la liste des parties de deux manières différentes : soit une liste liée unique, soit des listes séparées pour les parties dans le lobby (en attente du second joueur) et les parties en cours.

Chaque partie devra disposer d'un identifiant numérique unique qui sera transmis par le client lorsqu'il demandera à rejoindre la dite partie. Ces identifiants devront être transmis en même temps que les autres informations (nom de l'adversaire, son rang, ...)

Quand une partie est rejointe, le serveur commencera par préparer les structures pour la partie mais une fois cela fait, enverra aux deux joueurs le

message de fin de partie sans attendre qu'un coup soit joué. La victoire sera déterminée aléatoirement et les scores des joueurs mis à jour.

### 3 Client

Côté Python, on gèrera l'authentification à l'aide de deux `input()` successifs (login et mot de passe) et les deux informations seront envoyées en même temps au serveur (dans la version finale, on disposera d'une interface graphique avec deux champs d'entrée).

La liste des parties ouvertes sera affichée avec les numéros des parties et un `input()` permettra de sélectionner la partie que l'on désire rejoindre. Une valeur incorrecte (-1, 0 ou une chaîne de caractère au choix) permettra de demander la création d'une nouvelle partie (décider de la valeur et la signaler dans le prompt)