

Pente : Protocole

D.Moreaux

30 septembre 2024

1 Déterminer les échanges

La première étape est de décider quels seront les échanges entre le client et le serveur.

Ces échanges commencent à la connexion et se terminent à la déconnexion. Ils doivent donc commencer par un message "d'accueil" qui permettra de savoir si la connexion a été acceptée¹ et se terminera par le message de fin de session²

Il faudra ainsi définir les échanges nécessaires pour le lobby (affichage des parties disponible, rejoindre ou créer une partie, statistiques d'un joueur, changement de mot de passe, . . .) et pour la partie (du début de partie à la fin avec l'annonce du gagnant, en tenant compte de l'abandon de partie).

Cette liste doit être la plus exhaustive possible et inclure un résumé des informations qui sont incluses)

2 Type de paquets

L'échange peut prendre deux formes différentes : un protocole texte (comme les protocoles SMTP ou HTTP) ou un protocole binaire. Ce choix sera déterminant pour la suite et doit concerner l'ensemble des échanges (pas de changement en cours).

Dans le cas d'un protocole texte, la "syntaxe" utilisée devra également être déterminée : comment les différents paramètres seront-ils transmis, comment envoyer des données plus complexes telles que les tableaux, . . .

1. Une connexion peut par exemple être refusée suite à un ban ou si le serveur a atteint le nombre maximal de clients

2. qui peut se produire à l'initiative du client, en réponse à ce dernier, ou du serveur

3 Mise au propre du protocole

Cette étape consiste à la mise dans un document³ de l'entièreté du protocole, chaque paquet étant détaillé ainsi qu'un graphique d'état indiquant les différents états, les paquets acceptés dans chaque situation, les réponses possibles et les transitions vers d'autres états.

Ce document (surtout le graphique d'état) aideront fortement la suite du développement.

Une fois la situation mise sur papier, il conviendra de vérifier que le protocole est correct.

3. Document ODT (Libre Office), HTML, Markdown ou TXT