

## Implémentation du Serveur Matou

Le Serveur Matou repose sur 3 classes :

- **ServerCore** : La partie "haut niveau" du serveur. Cette partie est responsable de l'écriture et de la lecture TCP en mode "non bloquant".
- **ServerDataBase** : Il y a une seule instance de cet objet par exécution. Elle contient les informations du serveur en cours d'exécution (informations sur les clients connectés...).
- **ServerSession** : Il y a une instance de cet objet par client connecté. Elle contient les informations relatives à la session d'un client spécifique (son adresse, s'il a un pseudo valide ou non...) et mémorise son état de lecture actuel (lecture d'un en-tête de requête, lecture d'un argument de la requête...).

Le serveur alloue 1 ByteBuffer commun à tous les clients puis 2 ByteBuffer par client (1 pour l'écriture, 1 pour la lecture).

Le serveur positionne la limite du buffer de lecture de manière à lire les requêtes arguments par arguments. Le buffer n'est pas parcouru tant qu'il n'est pas rempli jusqu'à sa limite (afin de garantir l'intégrité de son contenu).

Chaque client est identifié par un pseudo unique. Le pseudo est libéré si le client se déconnecte.

Un pseudo ne peut contenir que des caractères alphanumériques et son encodage ne peut pas contenir plus de 32 octets.

Les pseudos sont insensibles à la casse.

Un message ne peut pas contenir de caractères de contrôles et son encodage ne peut pas contenir plus de 512 octets.

Le serveur gère à la fois les adresses IPv4 et les adresses IPv6.

La capacité du buffer de lecture individuel est générée à partir de la taille maximale des requêtes que le serveur peut recevoir.

La capacité du buffer d'écriture individuel et du buffer d'écriture général sont générées à partir de la taille maximale des requêtes que le serveur peut envoyer.

Cette spécification implique qu'il n'y a pas de taille maximale pour une requête.

Les requêtes sont ajoutées de manière atomique dans les buffers d'écriture. Si un buffer n'a pas la place nécessaire pour stocker la requête complète, alors la requête est ignorée pour ce buffer.

Lorsqu'une erreur survient avec l'un des clients (déconnexion ou non respect du protocole), le serveur ferme la connexion du client.

Le serveur ne doit pas s'arrêter suite à une erreur client.