

TP1- POO

Ecrire un Client / Serveur qui fonctionne de la manière suivante

Le serveur fonctionne en multi-thread en créant un nouveau thread pour chaque client qui se connecte.

Le serveur possède une liste d'objets dont la valeur des attributs doit être renseignée.

Lorsque le client se connecte, le serveur lui envoie une série d'objets qu'il retire de sa liste. Le client ne connaît pas le type des objets envoyés. Il réceptionne l'objet et crée dynamiquement une interface de saisie des attributs de l'objet. Attention si un entier est attendu, l'interface doit permettre la saisie d'un entier uniquement.

Lorsqu'il a terminé la saisie des valeurs, il valide et envoie le résultat au serveur.

Lorsque le serveur reçoit la réponse il insère l'objet dans une nouvelle liste puis envoie un autre objet de la liste.

Ecrire le diagramme de séquence entre le client et le serveur et entre les threads du serveur

Vous pouvez utiliser des éditeurs en ligne comme

<https://online.visual-paradigm.com/>

<https://www.lucidchart.com>

<https://www.draw.io/>

<https://creately.com/lp/uml-diagram-tool/>

1. Ecrire le code du client et du serveur

Vous pouvez procéder par étapes et mettre vos codes sur github (<https://github.com/>) en faisant un push pour chaque étape.

Etape 1 : écrire le serveur puis le client et vérifier la connexion entre les deux (envoi d'un message simple entre les deux)

Etape 2 : passer le serveur en multiprocessus en créant un thread à chaque connexion d'un client

Etape 3 : faire la synchronisation entre les processus pour l'accès à la liste des objets à envoyer/réceptionner

Etape 4 : mettre en place la réflexivité en envoyant un objet quelconque au client, celui-ci devra alors afficher la liste des attributs de l'objet avec leur type.

Etape 5 : faire la saisie sur la console des valeurs des attributs et renvoyer l'objet au serveur

Etape 6 : créer l'interface graphique qui permette de saisir la valeur des attributs

Remarque : dans un premier temps on peut considérer que les attributs des objets envoyés ne sont que des types simples (int, float ou double, String) et dans un second temps des types complexes (instances d'autres classes). Dans ce cas il faudra analyser récursivement les attributs.