

Projet Réseaux : Gestion de comptes bancaires

L'objectif est d'implémenter une application permettant de gérer des comptes bancaires en utilisant une architecture de type client-serveur.

Le serveur maintiendra pour chaque client un ou plusieurs comptes. On supposera que la création des comptes a été déjà faite.

Le serveur devra implanter les requêtes suivantes :

- **AJOUT** <id_client id_compte password somme> permettra d'ajouter la **somme** sur le compte identifié par **id_compte** par le client **id_client** identifié avec **password**.
- **RETRAIT** <id_client id_compte password somme> permettra de retirer la **somme** sur le compte identifié par **id_compte** par le client **id_client** identifié avec **password**.
- **SOLDE** <id_client id_compte password> permettra d'obtenir le solde du compte identifié par **id_compte** par le client **id_client** identifié avec **password**.
- **OPERATIONS** <id_client id_compte password> permettra d'obtenir les 10 dernières opérations effectuées sur le compte identifié par **id_compte**. Cette opération ne peut être demandée que par le client **id_client** identifié avec **password**.

Le client recevra et traitera les réponses du serveur :

- **OK** – informe le client que la commande (AJOUT, RETRAIT) s'est bien déroulée
- **KO** - informe le client que la commande ne s'est pas bien déroulée
- **RES_SOLDE** - *solde date_dernière_opération* : envoi au client le solde actuel du compte et la date de la dernière opération.
- **RES_OPERATIONS** - envoi au client les 10 dernières opérations du compte sous le format *type_opération date_opération montant_opération*

Le document <https://broux.developpez.com/articles/c/sockets/#LV> vous propose la présentation des sockets et leur utilisation dans différentes applications de type client-serveur. En vous inspirant du code fourni dans [1] vous devez proposer :

1. une implantation du serveur et du client en utilisant des sockets TCP.
2. une implantation du serveur et du client en utilisant des sockets UDP.
3. (bonus) Modifier votre serveur afin d'accepter et traiter plusieurs clients.

Travail à rendre via le moodle:

1. le code .c du serveur et du client.
2. un rapport présentant la description du code ainsi que le résultat des tests effectués (copies d'écran).