

Rapport Explicatif : Projet de Gestion Bancaire Client-Serveur

Ce rapport présente un projet de gestion de comptes bancaires implémenté en langage C en utilisant une architecture client-serveur. Le projet utilise des sockets pour la communication réseau et supporte des fonctionnalités comme l'ajout, le retrait, la consultation du solde, et l'historique des transactions. Ce document explique les différentes parties du code.

1. Serveur

Le serveur est responsable de gérer les comptes bancaires et de répondre aux requêtes des clients. Il utilise un modèle multi-thread pour gérer plusieurs connexions simultanément. Les principales fonctions incluent :

- Initialisation des comptes : Une fonction configure les comptes avec des identifiants, mots de passe, et soldes initiaux.
- Vérification des comptes : Le serveur vérifie si un compte et son mot de passe correspondent.
- Gestion des commandes : Les commandes comme AJOUT, RETRAIT, SOLDE et OPERATIONS sont traitées pour manipuler ou consulter les données des comptes.
- Réponse au client : Le serveur envoie des réponses indiquant le succès ou l'échec des opérations.

2. Client

Le client permet à un utilisateur d'envoyer des commandes au serveur et d'afficher les réponses reçues. Il se connecte au serveur via des sockets TCP et supporte les commandes suivantes :

- AJOUT : Ajoute une somme d'argent à un compte.
- RETRAIT : Retire une somme d'argent si le solde est suffisant.
- SOLDE : Consulte le solde actuel du compte.
- OPERATIONS : Affiche les 10 dernières opérations effectuées sur le compte.

3. Détails Techniques

Le projet est compatible avec Windows et Linux/macOS. Voici quelques points clés :

- Les threads natifs sont utilisés (pthread pour Linux/macOS, CreateThread pour Windows).

- Les sockets TCP assurent une communication fiable entre client et serveur.
- Un Makefile est fourni pour simplifier la compilation.
- Les identifiants par défaut sont :
 - * Compte 1 : ID = 1, Mot de passe = pass123
 - * Compte 2 : ID = 2, Mot de passe = pass456

4. Instructions pour Tester

1. Compilez les programmes serveur et client à l'aide du Makefile.
2. Démarrez le serveur avec ./serveur (ou serveur.exe sous Windows).
3. Lancez le client avec ./client (ou client.exe sous Windows).
4. Entrez les commandes suivantes dans le terminal client :
 - AJOUT <id_compte> <password> <somme>
 - RETRAIT <id_compte> <password> <somme>
 - SOLDE <id_compte> <password>
 - OPERATIONS <id_compte> <password>
5. Observez les réponses appropriées du serveur.

Ce projet a été conçu pour l'apprentissage des architectures client-serveur en langage C.