

# Rapport Projet :

## Transactions Bancaires

Elaboré par :

Ines KRICHEN

Fakhri Eddine MARZOUK

Islem BENSALD

Année Universitaire : 2021/2022

Ecole National d'Ingénieurs de Carthage, 45 rue des Entrepreneurs la CHARGUIA II, CP :  
2035, TUNIS, TUNISIE

Téléphone : 71 940.699 / 71 940.775 Fax : 71 941.579

@Web : <http://www.enicarthage.rnu.tn/>

La banque souhaite mettre en place un système informatique de gestion des comptes bancaires des clients .

Au niveau de ce projet on a 2 agents :

- un serveur central
- des postes clients



## Les fonctionnalités Pour Chaque Agent

❖ Le Serveur a pour rôle :

En lui donnant une référence d'un compte

Tout d'abord il assure l'existence du compte dans le fichier « comptes.txt »

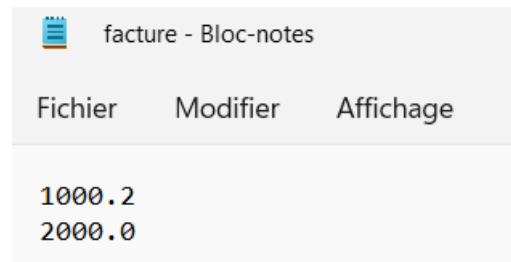
```
comptes - Bloc-notes

Fichier  Modifier  Affichage

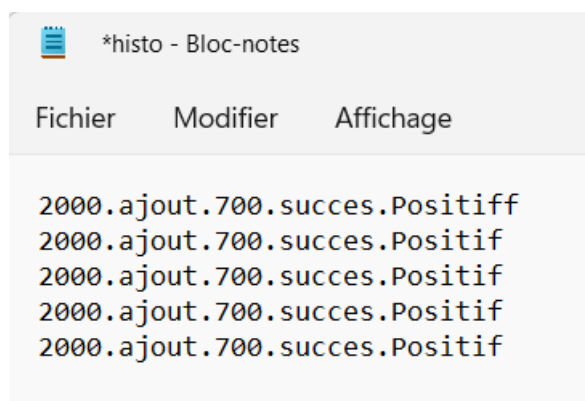
1000.500.Negatif.600
2000.500.Negatif.700
2000.-300.Negatif.700
2000.200.Positif.700
2000.200.Positif.700
2000.200.Positif.700
2000.200.Positif.700
2000.200.Positif.700
```

1. Le serveur peut récupérer les informations relatives au compte défini

2. Le serveur peut consulter la facture du client en cherchant dans le fichier « facture.txt »



3. Le serveur peut récupérer les transactions en interrogeant le fichier « histo.txt »



❖ Pour le Client :

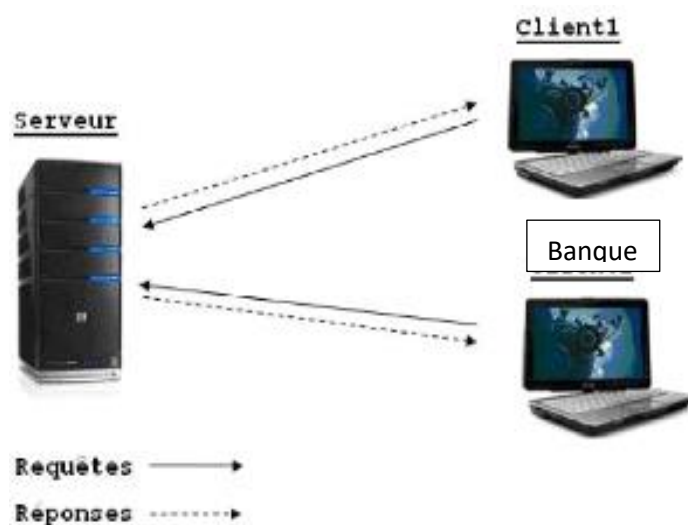
Réaliser les transactions :

- Effectuer un dépôt
- Effectuer un retrait

Recevoir une facture après chaque transaction

Lors d'exécution des transactions une mise a jour est faite pour les fichiers et la préparation et l'envoi des factures des clients.

## Architecture Client-Serveur et la communication entre les deux Machines



Les échanges entre les clients et le serveur doivent suivre un protocole bien défini pour que le serveur comprenne les requêtes des clients et pour que les clients comprennent les résultats renvoyés par le serveur.

## ❖ Communication entre 2 machines :

L'échange entre les 2 machines est fait par les Sockets Ceci sont les étapes de communication :

- ✓ La création du socket au niveau de 2 machines
- ✓ Chaque socket est associé à un port
- ✓ Connexion de 2 sockets à l'aide du protocole TCP
- ✓ Chaque machine lit et/ou écrit dans son socket
- ✓ Les données sont envoyées dans le réseau
- ✓ Fermeture de 2 sockets



## Le choix du protocole TCP :

TCP et UDP se distinguent sur deux plans : la qualité et la vitesse.

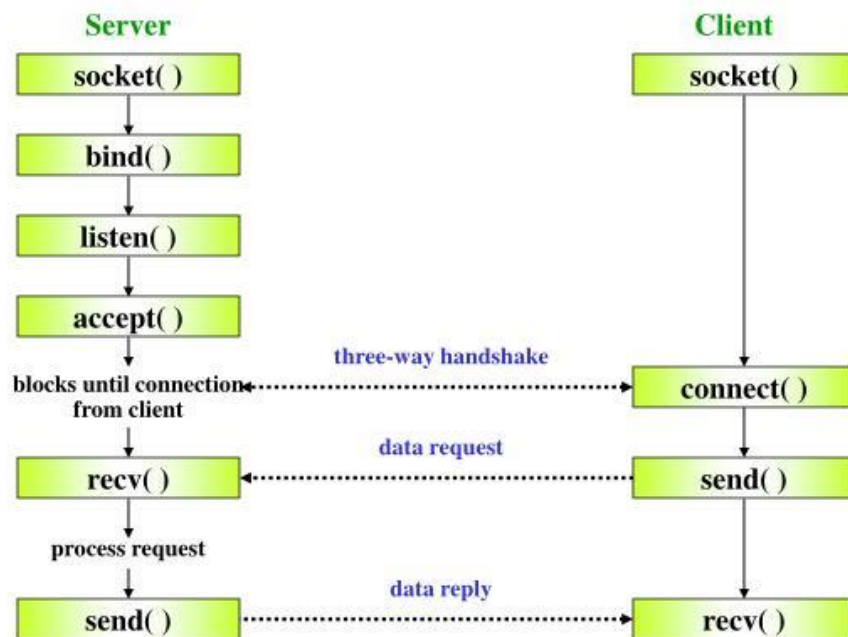
Avec TCP, les deux machines communiquent en permanence. Ainsi, la machine source établit d'abord une connexion avec la machine destinataire

Une fois la connexion établie, la source envoie des paquets numérotés dans un certain ordre au destinataire qui accuse réception à chaque fois après avoir vérifié que les données sont intactes grâce à un mécanisme de contrôle.

Pour faire simple, TCP est ainsi considéré comme un protocole axé sur la qualité de transmission. Qualité qui se

paye en vitesse, car, du fait des échanges permanents entre la source et le destinataire, la transmission des données prend du temps.

## TCP Client-Server Model



## Les Threads :

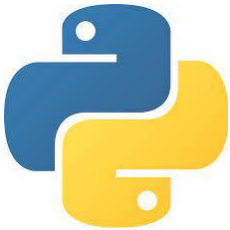
Un thread ou fil ou tâche est similaire à un processus car tous deux représentent l'exécution d'un ensemble d'instructions du langage machine d'un processeur.

On a utilisé les threads dans ce projet pour que chaque client se connecte il y aura lieu de création d'un nouveau thread



Utilisation de Mutex pour assurer l'exclusion mutuelle a une section critique.

## Technologies Utilisées :



Python : Pour le codage comme langage de programmation



GitHub : Collaboration entre les membres d'équipe

Répartition des taches :

Le travail était en groupe en présentiel