

Travail Pratique Java

CALCULATRICE | Boite mail

Mohamed EDDAHBY | 20 décembre 2023

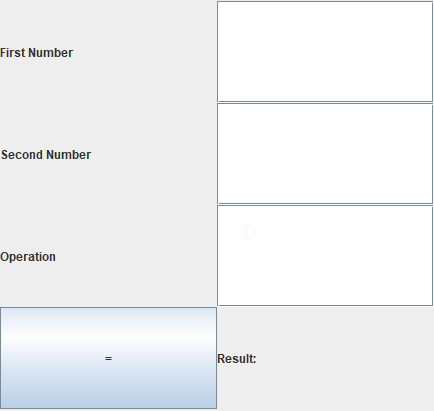
Calculatrice

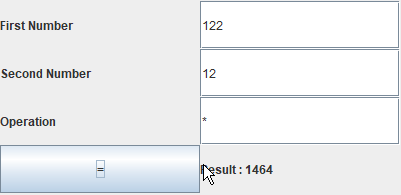
1. Calculatrice

Le but de ce travail pratique est mis en œuvre une interface graphique. Qui permet de calculer de nombre simple grâce à l’opération artistiques.  
le calcul doit être dans un autre Project sous forme d’un server, pour faire une laissant entre ses deux projets.  
le client envoyer le nombre a calculé est l’opération

Le server calculer les résultats.

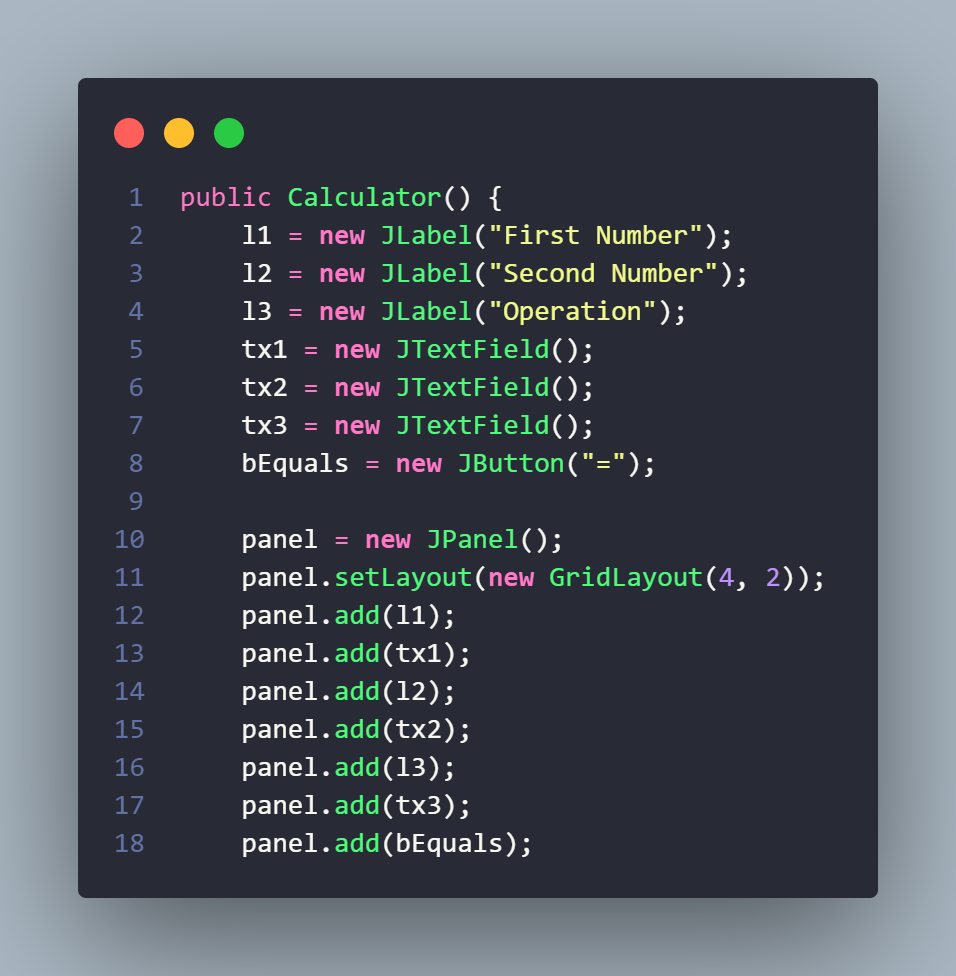
* L’interface



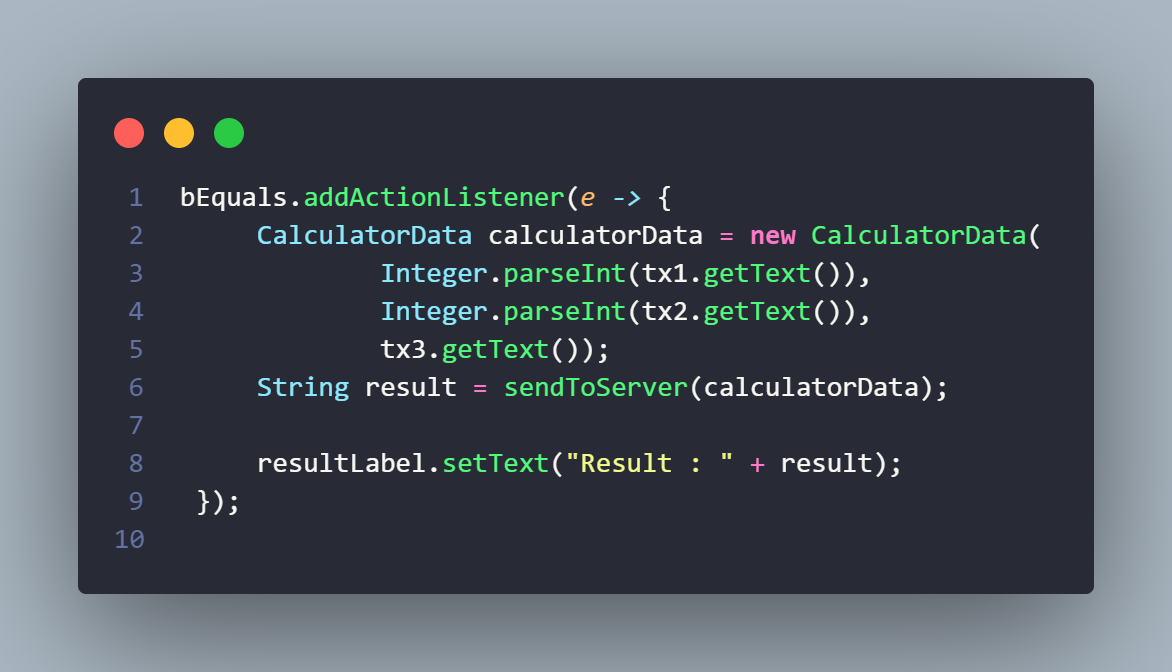


1. Coté Client :

* Les composant de l’interface graphique.



* La fonction qui invoque la fonction Sent to server



* Fonction Send to server



1. Coté Server:

* Le coté server :



* La fonction qui calcule



Cette fonction calculer les deux nombre réservoir dépend de l’opération fait dans la partie Client.

1. Boit Mail

Cette exercice sir a créé une boite mail pour affiche les emails pour chaque personne qui déjà enregistré dans la base de données.

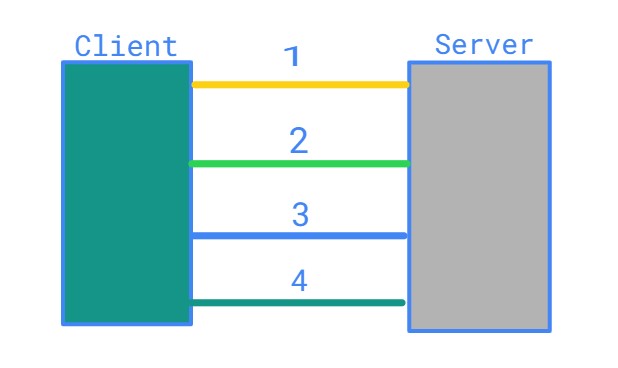
Les utilisateurs vont passer par une phase de autocitations (login) pour voir leur message et pour qu’il peut ajouter des autres.

Aux neveux techniques pour mettre en œuvre cette applications il faut utiliser plusieurs concepts de langage de programmation Java parmi eux :

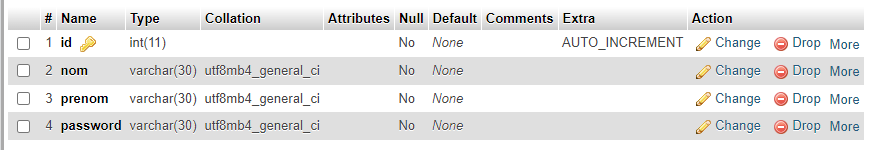
* L’utilisation de JDBC (base de données pour sauvegarder les données des utilisateurs).
* L’utilisation des sockets est les threads pour que l’application soit flexible.
* L’utilisation des design pattern

Pour faciliter l’échange de donnée entre la partie client et la partie serveur j’ai choisi de la faire en quatre étapes.

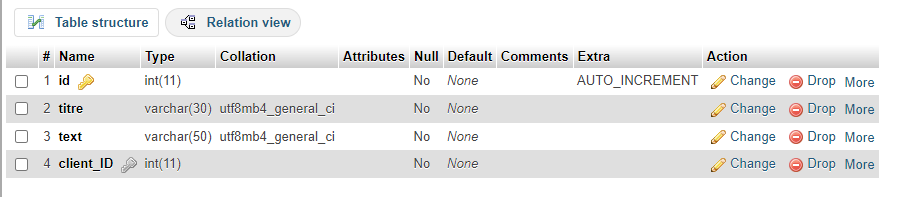
Comme suivant :



* Le client envoyer un nombre a serveur pour l’informé de l’opération qu’il va faire
* Le serveur confirmer l’opération
* Le client envoyer les données
* Les serveurs recevoir les données de client elles les traités puis il les envois de neveux au client
* La table Personne

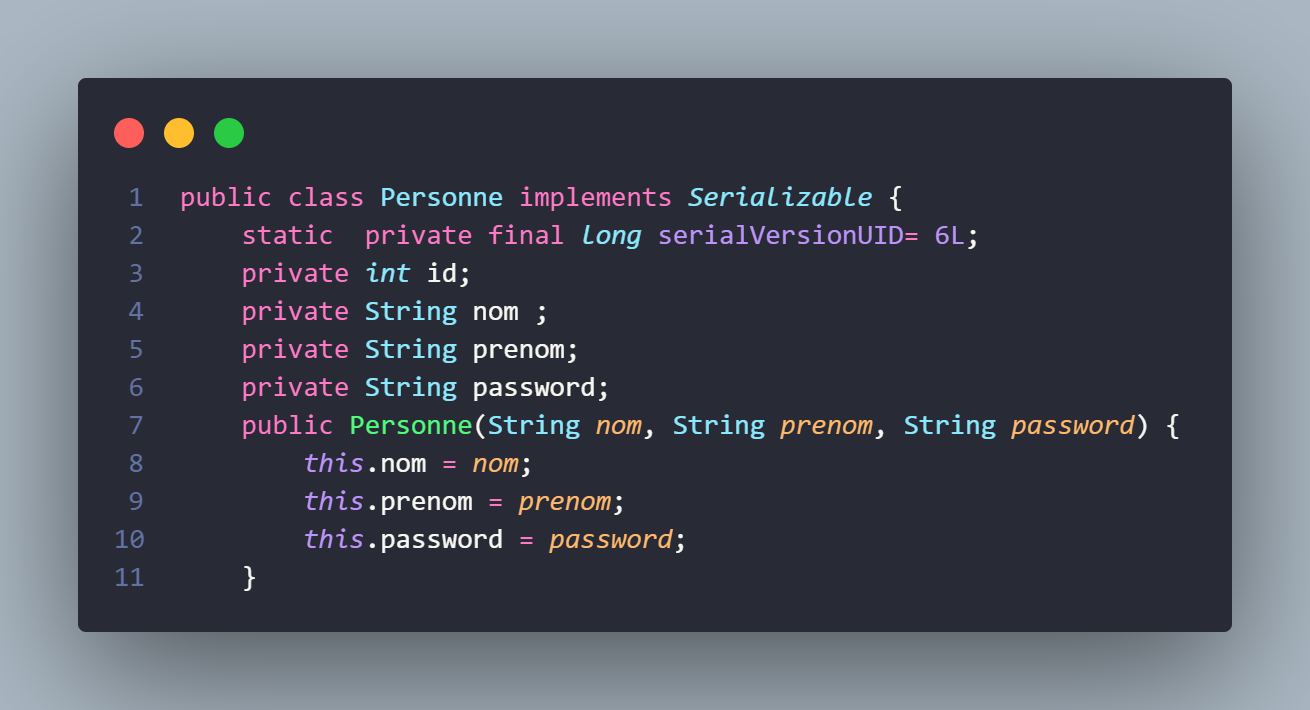


* La table Personne

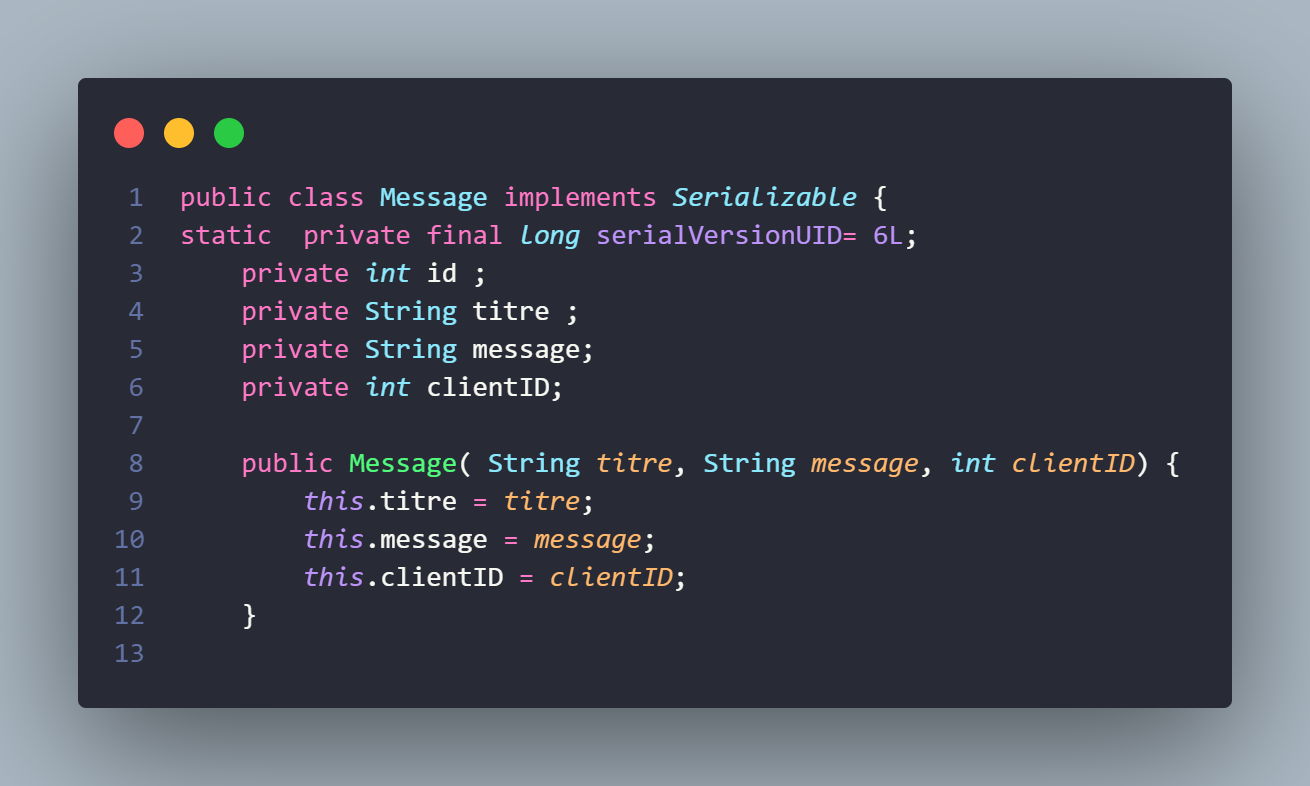


1. Coté Client:

* La classe Personne



* La classe Message

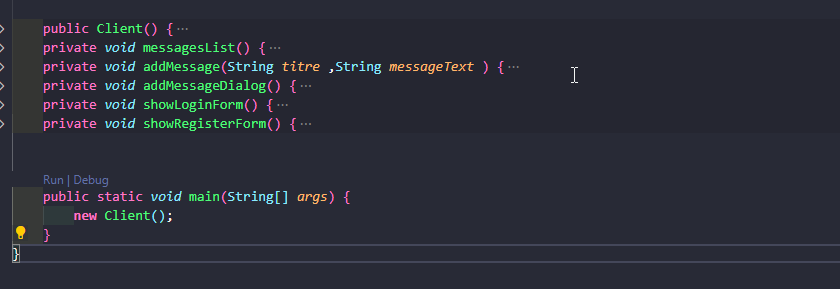


* La classe Client

Cette classe contient les sockets qui va communiquer avec le coté serveur aussi des composant graphique pour construire une interface graphique.



* Les fonctions de la classe Client



Chaque un de ces fonction se compose d’un socket et les éléments nécessaire pour envoyer est recevoir des données à partir de server.

On prend par exemple la fonction messageList qui permet de retourner une List des messages.



Cette fonction envoyer au début un nombre au serveur pour qu’il traiter ce cas dépond de ce nombre. Puis il attende que le serveur envoyer un objet. Cette Objet est une sérialisation d’une List des Object de types Message. Une fois il recoin cette objet il le traiter est le mais dans une composant graphique de type TextAea.

La même chose pour les autres fonctions

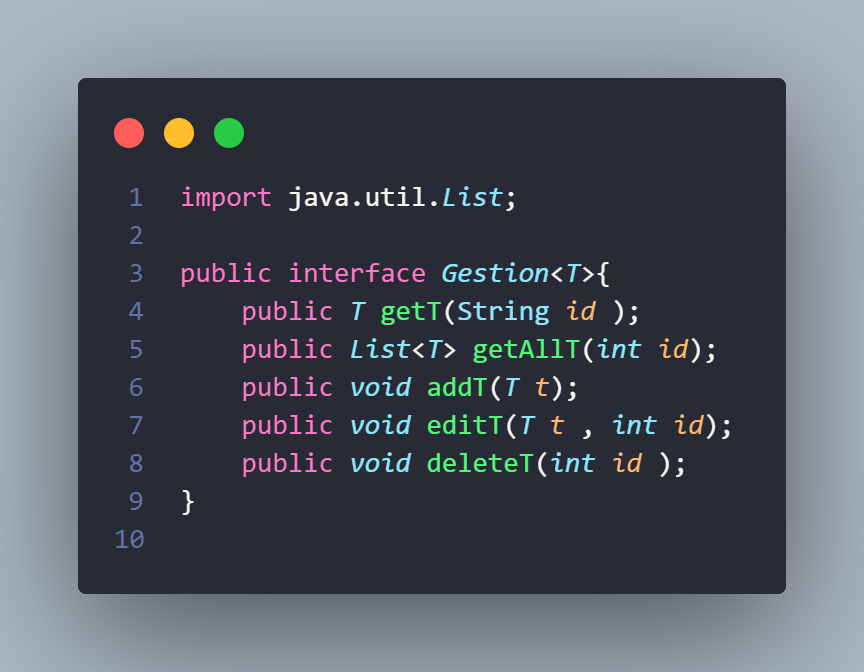
1. Coté Server:

* La classe DBConnection



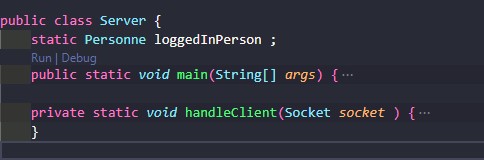
Cette classe permet la connexion avec la base de données on appliquant le design pattern Singleton pour garantir qu’on aura une seul connexion dans notre projet.

* L’interface Gestion



Cette Interface va être implémenté par des classes se qui facilite l’implémentation de ces classes en appliquant les design pattern DAO

* Classe Server



Cette Classe possède deux fonction la fonction appellent (main) et une fonction handleClient qui permet de traiter les données recevoir.

On verra par détails chaque fonction et leur traitement.

* La fonction main



Cette fonction permet de crée un thread pour chaque socket vient du client.

Cette fonction chaque fois une Cockett client se connecter avec e server il appelé la fonction handelClient.

* La fonction handleClient



Cette fonction contient plusieurs éléments qu’ils vont être utilisé par la suite.



Comme j’ai déjà mentionné le client envoi un code pour spécifier l’opération il va faire pour le server interagir dépend de code envoyer.  
par exemple dans cette cas le server va revoit le nombre 3, puis il va envoyer une confirmation qu’il attende le client à envoyer les données. Par la suit il les traités et il les retourner au client.

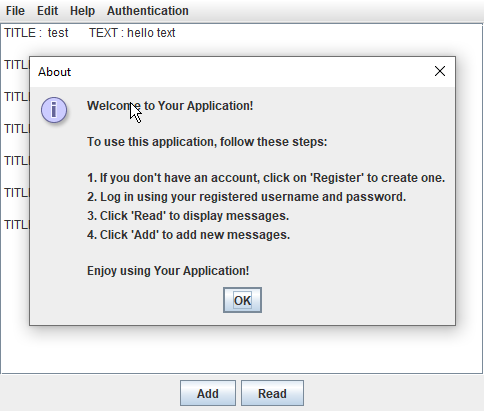
Au moment de traitement la fonctions handleClient appelé des fonctions des classes qui implémente l’interface Gestion et qui sont utilisé l’instance de connexion qui déjà créer au début.

* La suite de fonction handleClient



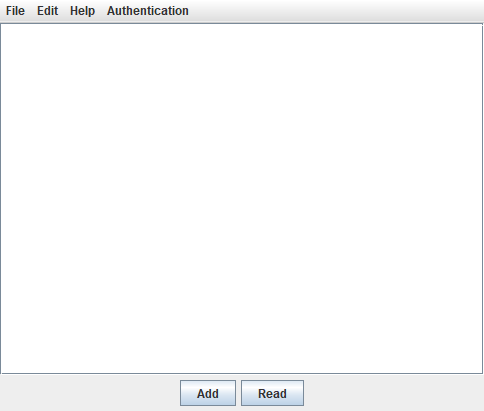
* La suite de fonction handleClient





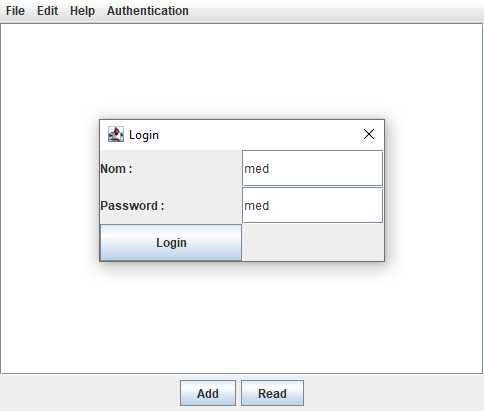
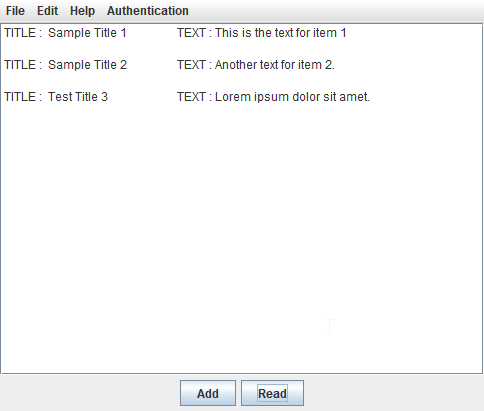
Interface graphique

About

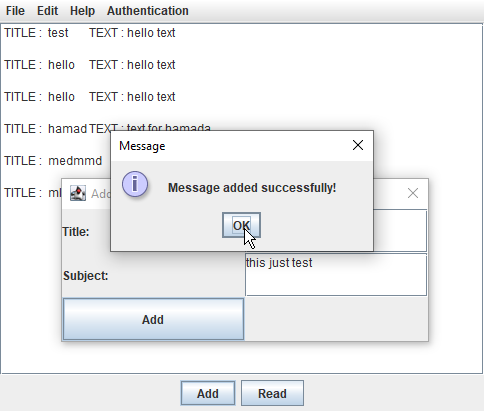
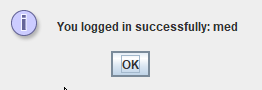


Login

Les messages



Ajouter un message



Conclusion :

En conclusion, le développement de l'interface graphique connectée au serveur pour la création de la boîte mail a été réalisé avec succès.

. Les choix techniques, tels que les design pattern, les threads, les et sockets ont été cruciaux pour atteindre les performances optimales attendues.