



# Rapport de Projet Stockage clé-valeur

Réalisé par

FIDAH hanane

Encadré par

JB.besnard  
J.adam

# ***Introduction***

Le serveur de base de données a été développé pour offrir une solution de stockage de clés et de valeurs en temps réel, permettant à plusieurs clients de se connecter simultanément pour exécuter des opérations de lecture et d'écriture. Le serveur utilise des sockets et des threads pour gérer les connexions clientes et offre un ensemble de commandes pour interagir avec les données stockées.

## ***Fonctionnalités Implémentées***

### **1. Déploiement du Serveur**

- Le serveur est opérationnel et écoute les connexions entrantes sur le port défini.
- Il peut gérer plusieurs clients en même temps via des threads dédiés pour chaque connexion.

### **2. Commandes de Base Supportées**

- Prise en charge de commandes telles que PING, SET, GET, DEL, KEYS, EXISTS, FLUSHALL, QUIT, et APPEND.

### **3. Stockage des Données**

- Les données sont stockées en mémoire via des structures de données permettant de stocker les paires clé-valeur pour chaque client.
- Le serveur conserve ces données en mémoire pendant son fonctionnement.

### **4. Persistance des Données**

- Le serveur prend en charge la sauvegarde des données sur le disque dans un fichier texte (`donnees.txt`).
- Les données sont chargées au démarrage du serveur depuis ce fichier pour assurer la persistance des informations.

## ***Points d'Amélioration***

### **1. Persistance Robuste**

- Actuellement, la persistance est réalisée de manière simple via un fichier texte. Une amélioration serait d'implémenter un système plus robuste de stockage des données pour garantir la fiabilité et l'intégrité des informations.

### **2. Gestion des Connexions et des Threads**

- Une gestion plus précise des threads et des connexions pourrait être envisagée pour une meilleure utilisation des ressources système.

## ***Conclusion***

Le serveur de base de données actuel fournit une infrastructure fonctionnelle pour la gestion des données via un réseau. Il prend en charge les principales opérations de manipulation de clés et de valeurs, mais des améliorations peuvent être apportées pour optimiser les performances et renforcer la fiabilité.

### ***Performance Comparative avec Redis***

Une évaluation comparative de performance avec Redis n'a pas encore été réalisée .

## ***Difficultés rencontrées***

J'ai rencontré plusieurs difficultés lors de l'implémentation du code notamment pour la gestion des clients et le chargement des données

## ***Prochaines Étapes***

1. Implémentation de la persistance robuste des données.
2. Optimisation des algorithmes de stockage et de recherche de clés.
3. Évaluation comparative des performances avec Redis.