**Plan du Rapport – Projet Java RPC & RMI : Gestionnaire de Mots de Passe**

**1. Objectifs du Projet**

* Maîtrise du RPC/RMI en Java
* Développement d’une interface graphique (JavaFX ou Swing)
* Intégration de la sécurité (authentification, hachage, validation)
* Compréhension des vulnérabilités réseaux

**2. Partie 1 – Application Java avec RPC**

**3.1 Architecture de l’application**

* Composants (client, serveur)
* Communication RPC

**3.2 Fonctionnalités développées**

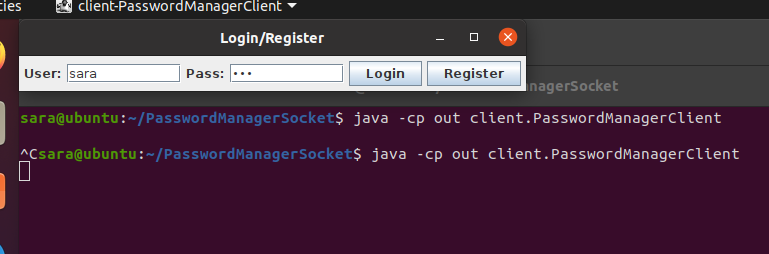
* Authentification
* Ajout, enregistrement, consultation des mots de passe
* Recherche des comptes

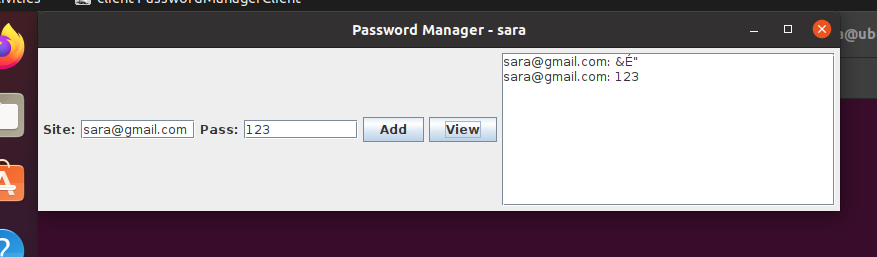
**3.3 Sécurité intégrée**

* Connexion sécurisée simulée (HTTPS basique)
* Validation des entrées
* Gestion de session simple

**3.4 Interface graphique**

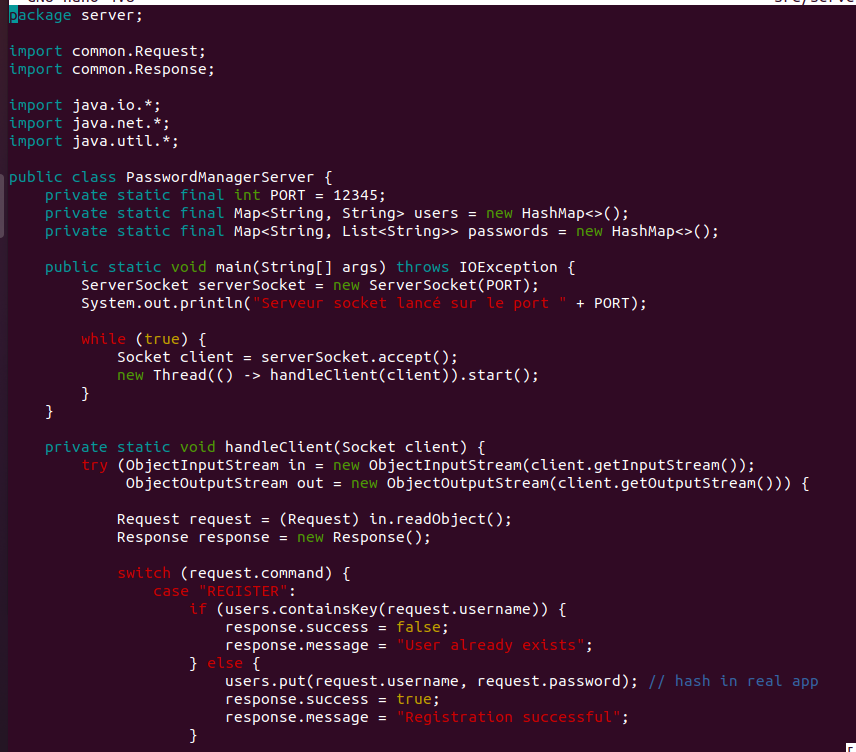
* Présentation des fenêtres (login, tableau, etc.)

****

****

**3.5 Code source commenté**

Code serveur :



Code client :



**4. Partie 2 – Adaptation avec RMI**

**4.1 Architecture RMI**

* Interface distante
* Implémentation serveur
* Client graphique

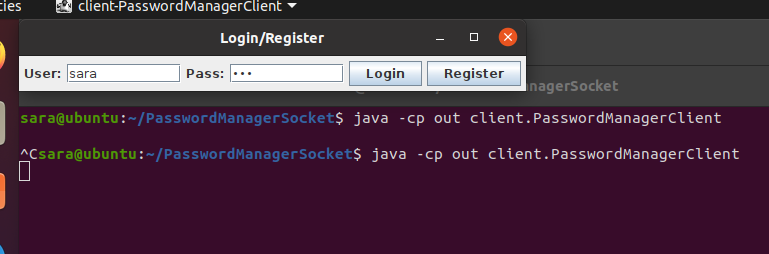
**4.2 Fonctionnalités reprises et différences**

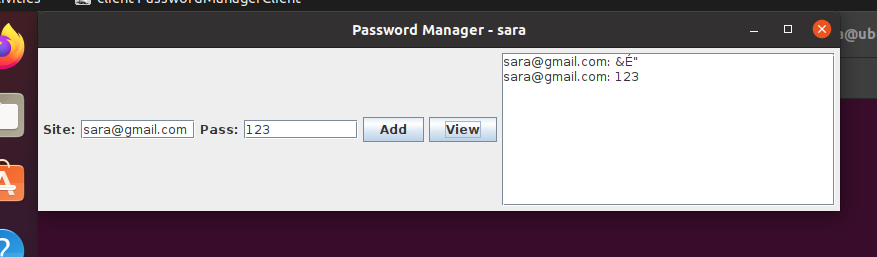
* Reprise des fonctions RPC
* Ajout de gestion des exceptions
* Authentification RMI

**4.3 Problèmes rencontrés & solutions**

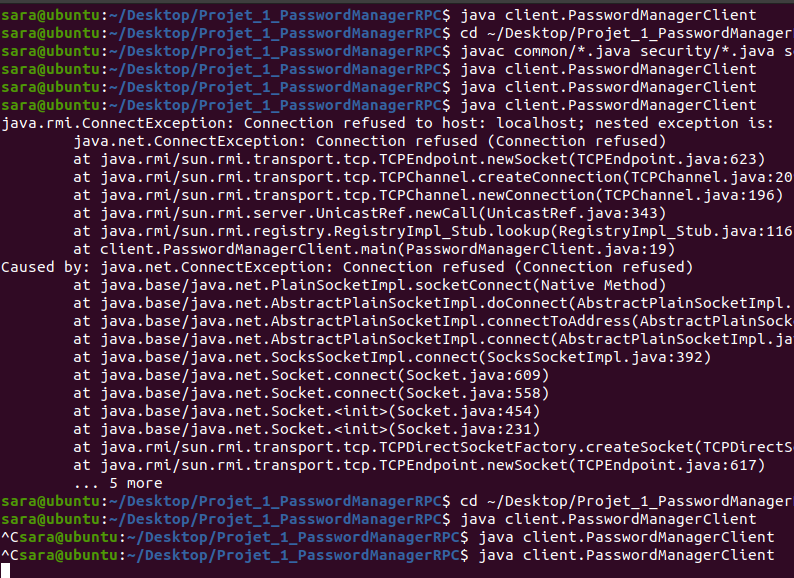
* Sérialisation
* Timeouts
* Connexions instables

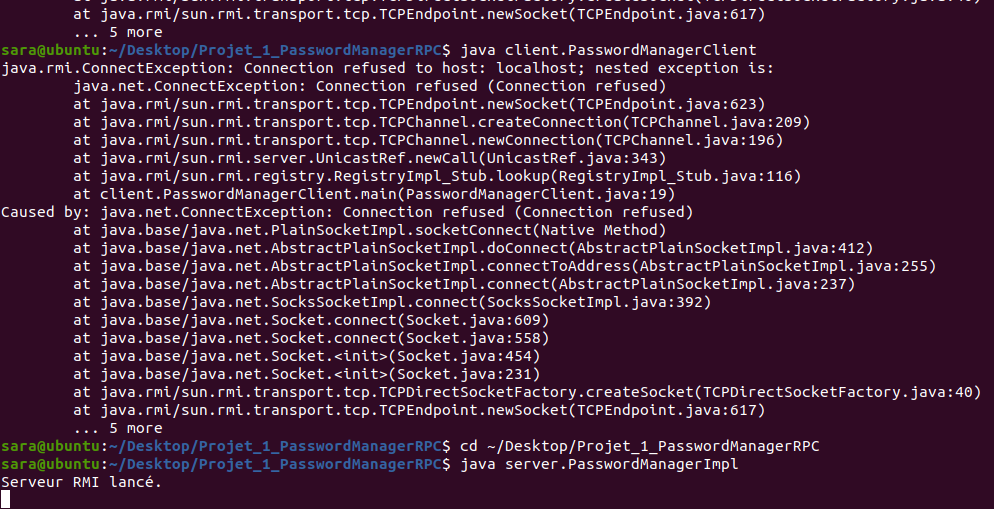
**4.4 Interface graphique Swing/JavaFX**

****

****

**4.5 Code RMI commenté**

****

****

**5. Comparaison entre RPC et RMI**

| **Critère** | **RPC Java** | **RMI Java** |
| --- | --- | --- |
| Type de communication | Basée sur sockets simples | Utilise objets distants |
| Sérialisation | Non native | Native |
| Facilité d’intégration | Plus bas niveau | Plus orienté Java |
| Sécurité | À implémenter manuellement | Partiellement gérable via RMI |
| Interface graphique | Similaire | Similaire |