

Projet : Système de Surveillance Distribué

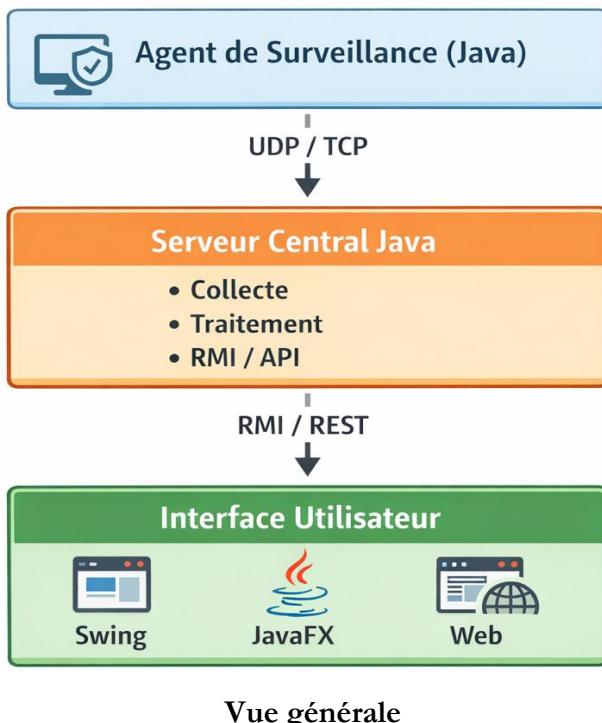
Contexte

La supervision des systèmes distribués nécessite des outils capables de collecter, traiter et visualiser en temps réel des informations provenant de plusieurs machines distantes. Ce projet vise à concevoir une plateforme distribuée permettant la **surveillance des ressources**, la **gestion des alertes**, et la **visualisation graphique** via **différentes technologies d'interface utilisateur** (desktop ou web).

Objectifs pédagogiques

- Implémenter une **architecture distribuée complète**
- Utiliser **Threads, TCP, UDP et Java RMI**
- Concevoir une **interface graphique moderne**
- Comprendre la séparation **Backend / Frontend**
- Appliquer les principes **client/serveur et MVC**

Architecture globale du système



Description des composants

1. Agent de Surveillance (Java)

Rôle :

- Collecter périodiquement les informations système
- Envoyer les données au serveur central

Fonctionnalités

- Surveillance CPU, mémoire, disque
- Envoi périodique via **UDP**
- Alertes critiques via **TCP**
- Identification unique
- Exécution multi-threadée

2. Serveur Central (Java)

Rôle

- Centraliser les données
- Gérer les agents
- Fournir les services d'accès aux données

Fonctionnalités

- Réception UDP multi-agents
- Serveur TCP d'alertes
- Gestion concurrente
- Services RMI
- Exposition d'une **API REST** (**à ajouter à la fin**)

3. Interface Utilisateur (choix technologique à faire : une seule option)

Option 1 : Interface Desktop Java (Swing ou JavaFX)

Technologies

- Swing (par défaut) ou JavaFX (alternative moderne)

Fonctionnalités

- Tableau des agents
- Graphiques temps réel
- Alertes visuelles
- Configuration des seuils
- Rafraîchissement automatique

Architecture

- MVC
- Communication via RMI

Option 2 : Interface Web (Angular / React / Vue)

Technologies

- Angular (ou React / Vue)
- HTML / CSS / TypeScript

Communication

- API REST exposée par le serveur Java
- WebSocket (optionnel)

Fonctionnalités

- Dashboard web
- Graphiques dynamiques
- Notifications temps réel
- Authentification utilisateur

4. Fonctionnalités à implémenter :

- **Historique et statistiques** : Permet l'enregistrement et l'analyse des métriques système sur une période donnée afin de visualiser les tendances, les pics d'utilisation et les performances globales des agents.
- **Alertes configurables** : Offre la possibilité de définir des seuils personnalisés déclenchant automatiquement des alertes afin de détecter et signaler rapidement les situations critiques.
- **Filtrage et recherche** : Facilite la gestion d'un grand nombre d'agents en permettant de rechercher, trier et filtrer les données selon différents critères pertinents.
- **Gestion des utilisateurs** : Assure la sécurité et le contrôle d'accès à la plateforme en gérant l'authentification, les rôles et les droits des utilisateurs.
- **Export des données** : Permet l'extraction des données collectées et des statistiques sous différents formats afin de les exploiter ou les analyser en dehors du système.

5. Livrables du projet

1. **Code source** complet et le guide d'utilisation.
2. **Documentation** de 15-20 pages contenant l'architecture du système, les diagrammes UML, le fonctionnement et l'interface utilisateur.

3. **Présentation** de 10 minutes présentant les principales fonctionnalités du système et les choix technologiques effectués.

6. Travail d'équipe

Le projet pourra être réalisé en **trinôme ou Quadrinôme, avec** une répartition des tâches entre les différents membres