

Fonctionnement du serveur JarRet

Choix d'implémentation

Le serveur se décompose en plusieurs sous parties :

- Le corps du serveur (Server.java)
- La définition d'un job (JobDefinition.java)
- La logique d'un job (Job.java)
- Le contexte actuel du serveur (Context.java)
- La tâche à effectuer pour ce contexte (TaskContext.java)

L'idée est de découper un maximum la logique du serveur pour obtenir un résultat clair.

Chargement des jobs

Les jobs sont chargés à partir d'un fichier JSON, chaque job est ajouté plusieurs fois dans une liste des jobs en cours en fonction de leur priorité (une priorité de 3 entraînera la présence du job 3 fois dans la liste).

Les fichiers JSON sont lus grâce à la librairie Jackson.

Logique de traitement des clients

Lorsqu'un client se connecte au serveur il est accepté par le serveur et le code contenu dans le fichier TaskContext.java est exécuté à chaque fois que le serveur le permet.

L'idée est de récupérer les caractères envoyés par le client sachant que ce traitement peut être interrompu, il y a différents états (enum ReadState) représentant l'étape de lecture actuelle.

La chaîne est stockée dans un string builder. A la fin de la lecture de l'en-tête, un HTTPHeader est créé et stocké dans la clé correspondant au client.

Suivant le type de demande (Get ou Post) le serveur génère une réponse pour le client.