

Master 1 Informatique - Réseaux et Communications - Projet

stockage distant et partage de fichiers

Ce projet vous permet de mettre en œuvre des outils vus en cours : communications entre processus sur internet (sockets, TCP/IP), threads, moyens de synchronisation, etc.

L'objectif est de réaliser une application distribuée pour le stockage distant de fichiers et leur partage entre plusieurs utilisateurs. Dans cette application, un utilisateur a la possibilité de stocker (respectivement, télécharger) des fichiers volumineux sur (resp. depuis) un serveur de stockage distant.

Les fonctionnalités offertes par cette application sont décrites dans ce qui suit. A vous de voir comment les réaliser.

- un utilisateur peut exporter un fichier sur le serveur distant. La taille minimum d'un fichier est de 1 Mo.
- un utilisateur peut aussi télécharger des fichiers depuis le serveur.
- pour télécharger un fichier, un utilisateur doit connaître son nom. Ce nom est fourni par l'utilisateur exportant le fichier sur le serveur.
- un fichier stocké sur le serveur peut être partagé (accessible) par plusieurs utilisateurs.
- un utilisateur doit être capable de télécharger et d'exporter des fichiers de manière simultanée.

Les étapes suivantes vous guident dans la réalisation de votre application :

Etape 1 : première réflexion

Définir l'architecture de votre application et le choix des outils à utiliser. Décrire ensuite les étapes à suivre pour la réalisation de votre application. Cette étape devra être résumée dans votre rapport avec justification de vos choix. Ne pas commencer à coder avant que cette étape n'arrive à un stade bien avancé.

La lecture intégrale et attentive de ce sujet vous permettra de guider vos choix. Vous devez notamment être attentifs aux traitements des connexions et des différentes requêtes, aux accès concurrents, etc. Enfin, cette étape vous permet de répartir les tâches à réaliser.

Exemples de questions à traiter durant cette étape :

- discuter l'intérêt d'avoir un stockage de données distant.
- quel est le nombre et rôle des processus constituant votre application ?
- comment les processus se connectent entre eux ?
- comment le serveur doit gérer plusieurs clients ?
- quelle est l'architecture des processus concurrents (s'ils existent) et quel est l'objectif de cette concurrence ? Quel est le rôle de chaque élément concurrent ?
- etc.

Etape 2 : transfert de fichiers

Pour que le transfert, le stockage et le téléchargement de fichiers se réalisent correctement, il faut organiser tous les échanges et stockage de données.

- définir une méthode pour le stockage d'un fichier sur le serveur.

- définir un protocole qui détermine le contenu des messages échangés entre un expéditeur et un destinataire pour la réalisation du transfert d'un fichier (dans les deux sens : exportation et téléchargement).

Etape 3 : protocole d'échange

Compléter le protocole d'échange pour le reste des fonctionnalités offertes par votre application (connexion/déconnexion des utilisateurs, demande d'un fichier, etc.)

Le protocole d'échange doit être décrit dans votre rapport avec justification de vos choix.

Etape 4 : réalisation

Dans un premier temps, programmez votre application avec un minimum de processus. Dans un second temps, ajoutez le support de plusieurs participants.

Quelques précisions :

- vous devez faire attention aux éventuels accès concurrents à des données et à la synchronisation.
- lorsqu'il y a concurrence de traitements, il est nécessaire d'utiliser des threads. A vous de voir comment les utiliser à bon escient.
- nous nous contenterons d'interfaces textes.
- implémenter progressivement les fonctionnalités de votre application en testant le bon fonctionnement de chacune avant de passer à la suivante.
- Toute forme de plagiat (copier-coller sur internet, échange de code entre binômes, etc.) sera sanctionnée et la sanction appliquée sur la note ne sera pas discutable.

Optionnel

A condition que ce qui précède fonctionne correctement, vous pouvez penser à faire des tests de performance de traitement des clients. Dans le cas où les performances se dégradent, revisiter votre conception et implémentation afin d'y remédier.

Document à rendre

Les documents à rendre seront sous forme d'une seule archive *nomEtudiant1-nomEtudiant2.tar.gz*, avec :

- un rapport (au format PDF) avec : description de l'architecture de votre application (justifiée), protocoles d'échange (justifiés), les schémas algorithmiques synthétiques de vos processus et des principales fonctionnalités de l'application, difficultés rencontrées et solutions apportées (les problèmes du genre difficulté de gestion du temps de travail et de programme chargé, etc. n'ont pas à être cités!!), etc. Le rapport ne doit pas excéder 6 pages.
- Le code source accompagné d'un Makefile, avec :
 - un premier répertoire, appelé *partie-obligatoire* contenant les sources sans la partie optionnelle.
 - un deuxième répertoire, appelé *partie-optionnelle* contenant les sources avec la partie optionnelle. Rappel, cette partie ne sera pas prise en compte si la première n'est ni complète ni correcte.

L'archive est à déposer en utilisant l'espace pédagogique. Si vous n'êtes pas encore inscrit dans l'UE FMN104 à partir de cet espace, faites le.

Attention : aucun changement dans le code ne doit être fait par le correcteur avant de pouvoir exécuter vos programmes. Pensez alors au passage de paramètres !

Déroulement, dates et soutenances

Le projet est à réaliser par binômes ! Les binômes devront être annoncés à la première séance du projet (semaine du 24/11).

Votre présence en séance de TP pour la réalisation du projet est très recommandée pour votre suivi. Durant ces séances, vous pouvez poser des questions à votre encadrant(e).

Date limite de dépôt des documents, le 23/12/2014 à 23h00. Après ce délai, AUCUN dépôt ou envoi par email ne sera accepté et la note sera 0/20.

Une soutenance aura lieu courant janvier (les dates vous seront communiquées). Vous aurez 15 minutes pour faire une démonstration de votre projet en utilisant deux à quatre machines de la fac, et pour répondre à des questions (7 minutes de démonstration et 8 minutes de questions).

Pour la soutenance, vous devez préparer des scénarios illustrant le respect du cahier des charges, la gestion de la concurrence et autres fonctionnalités pertinentes de votre choix. Des scénarios posant problème (du fait que l'implémentation ne soit pas aboutie ou en présence de bugs) ne sont pas à exclure de la démonstration !