

FORUM DE DISCUSSION : PROJET DE PROGRAMMATION SYSTÈME INFO 1

L. POINSOT, C. RODRIGUES

Contacts clients :

Laurent POINSOT (laurent.poinsot@lipn.univ-paris13.fr)

Christophe RODRIGUES (christophe.rodrigues@lipn.univ-paris13.fr)

Le but de ce projet est de réaliser un *forum* de diffusion de *messages* (ou *articles*).

Sur le plan technique, nous rappelons quatre objectifs principaux et **obligatoires**.

- (1) **Portabilité** : le système doit fonctionner (au moins) sur les stations GNU/Linux en salle informatique ;
- (2) **Sécurité** : le système devant manipuler des articles de plusieurs utilisateurs, il ne faut pas qu'un utilisateur puisse détruire un article d'un autre utilisateur, ou compromettre la structure de données du système ;
- (3) **Facilité d'administration** : le travail de l'ingénieur système doit être simplifié au maximum : une fois que la configuration est faite, il ne devrait plus avoir à intervenir ;
- (4) **Localisation** : vous ferez le choix d'un fonctionnement **local** (c'est-à-dire que le système client serveur est installé en local sur une machine) ou **à distance** via le réseau (le client et le serveur pourront être installés sur des machines différentes et communiquer à travers de sockets). Vous pouvez éventuellement commencer le projet en mode de fonctionnement local puis, dans un deuxième temps, mettre à jour le système afin de supporter un mode de fonctionnement à distance plus réaliste ¹.

Ci-dessous sont données les fonctionnalités **minimales** supportées par le système ainsi que son architecture globale. **Il vous est bien sûr possible d'ajouter de nouvelles fonctionnalités ou bien de développer une interface graphique.** Néanmoins votre système devra

Date: 26-01-2009 11:03.

1. Le mode de fonctionnement à distance bénéficiera en outre d'un bonus dans la notation finale du fait de la difficulté accrue de programmation.

a minima fonctionner comme cela est indiqué ci-dessous.

Le forum se compose de deux programmes distincts : un **serveur** et un **client**.

SERVEUR.

Le serveur est lancé une fois pour toutes, par l'ingénieur système, lors du démarrage de la machine, pour le compte d'un utilisateur spécial ; vous le lancerez vous-même en phase de test initial, puis vous pourrez faire lancer votre serveur par un collègue.

Le rôle du serveur est de contrôler le stockage des articles. Ceux-ci sont placés dans un répertoire que seul le serveur peut modifier, mais que tous les autres utilisateurs peuvent lire. Chaque article porte un numéro comme nom. Périodiquement (une fois par jour par exemple), le serveur lance une expiration, c'est-à-dire qu'il archive les articles plus vieux qu'une certaine date (passée en argument du serveur par exemple). Ces articles ne feront pas partie de la liste des articles non consultés (voir plus loin) par les utilisateurs.

Les fichiers contenant les articles doivent comporter deux parties séparées par une ligne vide. La première partie est l'*en-tête*, comprenant l'identité de l'auteur, la date de postage de l'article et le sujet fourni par l'auteur. La deuxième partie est le message proprement dit.

Par exemple :

```
From : Zidane (Zinédine Zidane)
Date : Thu Jan 10 12 :34 :00 MET 2009
Subject : Coup de boule
```

Pourquoi ai-je mis ce coup de boule en finale de la coupe du Monde ? Je peux l'avouer maintenant : c'est parce que j'ai souhaité finir ma carrière « à la Marseillaise » !

CLIENT.

Le client est un programme exécuté par les utilisateurs du forum, pour consulter et poster les articles. La consultation s'effectue directement dans le répertoire des articles.

Le client ne doit pas présenter à l'utilisateur des articles qu'il a déjà lus, c'est-à-dire que le programme maintient à jour un fichier « `~/forum` » avec les numéros d'articles déjà lus par l'utilisateur.

L'interface utilisateur de base (que vous pouvez améliorer !) est simplifiée à l'extrême. Le client doit présenter la liste des messages (avec leur numéro, leur auteur et les premiers caractères du sujet) non encore lus à l'utilisateur. Si un utilisateur tape un numéro d'article, celui-ci est affiché à l'aide de l'utilitaire `more` (ou de toute autre commande référencée par la variable d'environnement `PAGER`), même si l'article est déjà lu. Le contenu de l'article doit être fourni à la commande via son entrée standard. Si l'utilisateur tape la commande `d` suivie d'un numéro d'article, celui-ci est considéré comme lu. La commande `D` marque tous les articles comme étant lus. La commande `l` affiche la liste des articles non lus.

Avec la commande `s < sujet >`, le client appelle l'éditeur `emacs` (ou toute autre commande référencée par la variable d'environnement `EDITOR`) pour saisir l'article dans un fichier temporaire dans le répertoire `/tmp` (créé avec `mkstemp`). La transmission de l'article de l'éditeur au client passe par un tube. Une fois l'article composé, le client demande confirmation avant de poster l'article réellement.

Pour poster l'article, le client procède de la manière suivante : tout d'abord, il met en forme l'en-tête et copie l'article dans un fichier temporaire *dans le répertoire du serveur* (créé avec `mkstemp`). Puis, via une *file de messages* (pour un fonctionnement local) ou via une *socket* (pour un fonctionnement à distance) il envoie le nom du fichier au serveur. Le serveur attribue alors un numéro à l'article et renomme le fichier temporaire en conséquence. Vous noterez que le client doit pouvoir écrire dans le répertoire du serveur, et doit donc avoir les droits de l'utilisateur propriétaire du serveur.

ORGANISATION ET TRAVAIL ATTENDU.

- (1) **Organisation du travail** : Tous les projets s'effectuent en binôme sauf exception validée par les clients. Le sujet étant le même pour tous, il est fortement conseillé à chaque binôme de faire preuve d'autonomie, d'esprit d'initiative et d'originalité. Chaque binôme doit donc être parfaitement cloisonné de façon à ne souffrir d'aucun transfert de compétences désiré et/ou subi. Le « copiage-collage », l'*espionnage industriel*, la collaboration

ou toute autre preuve explicite - ou jugée comme telle - de diffusion ou fuite d'information au profit d'un autre binôme sera sévèrement punie au niveau de la note finale que ce soit pour la source comme pour le bénéficiaire de la fuite ! Au contraire, imagination et indépendance sont encouragées et valorisées dans la note finale ;

- (2) **Planification du projet** : Ce projet se déroule suivant le cycle en V du développement logiciel consistant dans sa plus grande généralité en sept phases ou étapes intermédiaires (pour ce projet, le nombre d'étapes est réduit). À l'issue de chaque étape (la phase d'implantation mise à part) doit être rendu un document (version papier et/ou électronique) rendant compte de façon synthétique (ce ne sont pas des romans !) de ladite étape. Au terme de l'ultime étape - la livraison finale - aura lieu une présentation orale du travail effectué et le bon fonctionnement du logiciel sera démontré à l'aide d'une batterie de tests. Les binômes sont contraints de suivre le calendrier prévisionnel du déroulement du projet (présenté dans la dernière partie de ce document). Ils sont avertis que la qualité des documents transmis ainsi que la ponctualité de leur livraison prendront une part **prépondérante** dans la note finale (en d'autres termes, les binômes ne doivent pas se contenter de coder au risque d'être déçus de la note finalement attribuée).

CALENDRIER PRÉVISIONNEL.

- (1) **Cahier des charges**. Doivent figurer : mise en forme des demandes du client (quelles sont les fonctionnalités attendues par le client ?), ajout d'éventuelles fonctionnalités supplémentaires pour le forum et description synthétique de l'utilisation du logiciel (comment doit procéder un utilisateur pour installer le client, pour poster des messages ? comment un administrateur installe-t-il et manage-t-il le serveur ? *etc.*). Ce document contient également la description des tests qui seront effectués le jour de la livraison finale afin de démontrer le bon fonctionnement du logiciel. *C'est précisément lors cette phase que les binômes doivent faire preuve d'originalité et d'esprit d'initiative.* Documents à livrer (format papier ou électronique) : le cahier des charges. **Date de livraison**² : **03/03/09** (minuit dernier délai si format électronique) ;

2. Toutes les dates de livraison indiquées sont à respecter sous peine de sanction au niveau de la note finale.

- (2) **Spécifications générales.** Dans ce document on entre dans l'aspect descriptif des fonctionnalités du forum. Néanmoins il ne s'agit pas d'un document de programmation ! Il contient les descriptions de l'architecture générale client-serveur, des algorithmes des principales fonctionnalités du client et du serveur, descriptions des tests de validation de ces fonctionnalités (comment peut-on être certain que chaque fonctionnalité réalise bien ce qu'elle est censée faire ?). Dans ce document, vous ne faites aucune référence au codage dans un langage de programmation. Les algorithmes doivent être décrits soit en pseudo-code (*si ... alors, boucle tant que, etc.*) soit à l'aide de schémas. Documents à livrer : le document des spécifications générales. **Date de livraison : 31/03/09** (minuit dernier délai si format électronique) ;
- (3) **Implémentation (codage).** Le contenu de ce document correspond au code contenu dans l'implémentation du logiciel. Vous y détaillerez - et expliquerez - le code de chacune des fonctionnalités des programmes client et serveur. Ce document contient aussi les tests de validation de chacune des fonctions contenues dans le code. Ce document se rédige en même temps que le codage puisqu'il reprend, en l'expliquant, le code de votre programme. Vous devez rédiger ce document, comme si vous alliez le donner à quelqu'un d'autre pour qu'il effectue le codage. Le code proprement dit est effectué en langage C (en tous cas pour ce qui concerne l'aspect programmation système). Les éventuelles IHM pourront être codées dans le langage qui vous convient le mieux. Documents à livrer : le document d'implémentation. **Date de livraison : 26/05/09** (minuit dernier délai si format électronique) ;
- (4) **Livraison finale.** Présentation orale du fonctionnement du logiciel en mettant l'accent sur les originalités apportées. Réponses aux questions du client (ces questions concerneront les différents documents livrés, l'utilisation et le test du logiciel, la connaissance du code). Aucun document n'est à livrer à cette étape. **Date de soutenance : début juin 2009** (la date exacte sera diffusée ultérieurement).

LE CORRECTEUR (SUSNOMMÉ LE CLIENT) SE RÉSERVE LA LIBERTÉ DE MODIFIER À SA GUISE - MAIS AUSSI APRÈS CONCERTATION AVEC LES ÉTUDIANTS - LE CALENDRIEL PRÉVISIONNEL. BIEN ÉVIDEMMENT TOUTE MODIFICATION SERA ANNONCÉE AUX BINÔMES AVEC UN PRÉAVIS DÉCENT *ad hoc*.