

Travail pratique no.1

INF1010 – Réseaux d'ordinateurs II

Date limite de remise : 26 octobre 2025 à 23h55

Travail en équipe: 4 à 6 étudiant(e)s

Pondération: 15%.

1. Mandat à réaliser

Il s'agit de réaliser, en utilisant **les sockets**, un annuaire accessible à distance. À titre d'exemple, l'annuaire référencera les professeurs du département et les étudiants inscrits au cours INF1010.

Chaque personne inscrite dans l'annuaire est caractérisée par :

- son nom et son prénom;
- sa catégorie (professeur, auxiliaire d'enseignement, ou étudiant);
- son matricule (si étudiant);
- son adresse courriel;
- son téléphone au bureau (si professeur ou auxiliaire d'enseignement);
- son domaine d'activité, et d'autres informations que vous jugerez pertinentes

Ces caractéristiques peuvent se trouver dans une base de données dans une machine (host) donnée. L'annuaire devra être réalisé sous forme d'une application client/serveur.

1.1. Serveur

Le serveur gère la base de données et permet (au moins) de :

1. Lister les membres d'une catégorie donnée;
2. Lister les professeurs dans un domaine d'activité donné;
3. Rechercher un membre;
4. Ajouter un membre;
5. Supprimer un membre;
6. Modifier (mettre à jour) un membre;
7. Mettre un membre sur la liste rouge;
8. Enlever un membre de la liste rouge.

Les services 1 à 3 :

- retournent toutes les informations sur les membres qui ne sont pas sur la liste rouge,
- retournent les noms et prénoms des membres qui sont sur la liste rouge, en mentionnant qu'ils sont sur la liste rouge.

1.2. Clients et administrateur

Les clients (qui ne sont pas forcément sur le même host que le serveur) peuvent faire des requêtes au serveur pour faire appel aux services 1 à 3.

Un client particulier, appelé administrateur, peut faire appel à tous les services offerts par le serveur. L'administrateur devra entrer un mot de passe pour avoir accès aux services 4 à 8.

1.3. Communication entre le client et le serveur

La communication entre le client et le serveur est transparente car elle est gérée par socket.

1.4. Environnement et langages de programmation

Vous pouvez utiliser le langage de programmation de votre choix, mais l'application doit être compatible avec l'environnement Windows.

1.5. Interface utilisateur

L'interface utilisateur devra être conviviale.

1.6. Nombre de clients

Votre système doit supporter plusieurs clients qui accèdent en même temps au serveur.

1.7. Robustesse

Le système ne doit pas planter si l'utilisateur entre une donnée imprévue. De plus, le client ne doit pas planter s'il ne trouve pas le serveur. Il doit plutôt informer l'utilisateur avant de s'arrêter.

2. Livrables

1. Le code source du programme dans un fichier compressé.
2. Un rapport, en format Word ou PDF, qui doit comporter les éléments suivants:
 - Page de couverture;
 - Introduction;
 - Captures d'écran de l'interface utilisateur et les différents cas d'utilisation;
 - Spécification du système réalisé;
 - Distribution des tâches entre les membres de l'équipe;
 - Conclusion:
 - problèmes rencontrés (résolus ou pas);
 - critiques et propositions d'améliorations au travail réalisé;
 - critiques et propositions d'améliorations au travail demandé;
 - Bibliographie (en format IEEE ou APA).

N'oubliez pas de préciser: le nom de chaque fichier source et son contenu, comment générer l'exécutable, comment utiliser l'exécutable, etc. Bref, tout ce qu'on pourrait mettre dans un fichier «ReadMe».