

Enseignant(s)

PICHARD Félix

Email(s)

fpichard@esgi.fr

Création d'un outil de gestion des clients

1 Matières, formations et groupes

Matière liée au projet :

Formations : -

Nombre d'étudiant
par groupe :**2 à 3**Règles de constitution des groupes: **Libre**Charge de travail
estimée par étudiant : **40,00 h**

2 Sujet(s) du projet

Type de sujet : **Imposé**

Projet de C

Création d'un ERP de suivi des prospects et des clients d'une entreprise. Le logiciel doit permettre une gestion de la part d'un commercial de ses prospects et de ses clients. Il doit être possible de créer, de mettre à jour et de supprimer chaque élément. Il est évidemment également possible de lire chaque élément, soit de manière unitaire, soit selon des conditions, implémentées. Pendant que le programme tourne, tous ces éléments doivent être stockés dans une liste chaînée. Pour que le programme enregistre les données, celles-ci doivent être stockées dans un fichier texte (CSV) ou dans une base de données (SQLite).

Les contraintes techniques sont les suivantes :

- Une liste chaînée « prospects » et une liste chaînée « clients »
 - o Une fonctionnalité doit exister pour transférer un profil de la liste chaînée « prospects » à la liste chaînée « clients »
 - o Une vérification doit exister pour voir si un élément ajouté n'existe pas déjà dans l'une des deux listes chaînées. Cette vérification se fait sur les champs « nom », « prénom » et « entreprise », sans respecter la casse.
 - o Il faut que toutes les opérations CRUD soient implémentées sur ces listes chaînées
 - o Une fonctionnalité de sauvegarde de la liste chaînée doit être implémentée de 3 manières différentes. Cette sauvegarde se fait dans un fichier CSV ou une base de données SQLite :
 - Une commande « save » (ou équivalent) dans l'interface interactive du programme ;
 - Un signal SIGUSR1 envoyé au programme sauvegarde automatiquement ;
 - Un signal SIGINT (CTRL+C) doit demander si l'utilisateur veut :
 - Quitter sans sauvegarder ;
 - Quitter en sauvegardant ;
 - Ne pas quitter.
- Concernant a minima les opérations CRUD, le choix de la fonction à utiliser doit se réaliser via un tableau de pointeurs sur fonction
- Un parser doit évidemment être implémenté pour lire un CSV ou SQLite et le transformer en listes chaînées.
- Attention à la clarté et la propreté de votre code. Un malus pouvant monter jusqu'à 4 points peut être appliqué.
- Implémenter une fonction d'imports de fichier CSV et/ou SQLite, qui serait fusionné avec le fichier CSV ou SQLite existant. Ex : tout est stocké dans une base de données SQLite, mais l'utilisateur peut ajouter du contenu en donnant un fichier CSV contenant de nouvelles entrées au programme. Le programme doit évidemment vérifier qu'une entrée n'existe pas déjà. Un rapport contenant les doublons non ajoutés doit être généré.
- Les listes chaînées sont doublement chaînées et/ou circulaires. Il faut que l'utilisation de ces listes chaînées soit également implémentée.
- Les données au repos doivent être chiffrées avec un algorithme de chiffrement symétrique standard (AES par exemple).

3 Détails du projet

Objectif du projet (à la fin du projet les étudiants sauront réaliser un...)

L'objectif du projet est de créer un outil de gestion de clients en ligne de commande, en C, en utilisant les techniques vues en cours.

Descriptif détaillé

Création d'un ERP de suivi des prospects et des clients d'une entreprise. Le logiciel doit permettre une gestion de la part d'un commercial de ses prospects et de ses clients. Il doit être possible de créer, de mettre à jour et de supprimer chaque élément. Il est évidemment également possible de lire chaque élément, soit de manière unitaire, soit selon des conditions, implémentées. Pendant que le programme tourne, tous ces éléments doivent être stockés dans une liste chaînée. Pour que le programme enregistre les données, celles-ci doivent être stockées dans un fichier texte (CSV) ou dans une base de données (SQLite).

Les contraintes techniques sont les suivantes :

- Une liste chaînée « prospects » et une liste chaînée « clients »
 - o Une fonctionnalité doit exister pour transférer un profil de la liste chaînée « prospects » à la liste chaînée « clients »
 - o Une vérification doit exister pour voir si un élément ajouté n'existe pas déjà dans l'une des deux listes chaînées. Cette vérification se fait sur les champs « nom », « prénom » et « entreprise », sans respecter la casse.
 - o Il faut que toutes les opérations CRUD soient implémentées sur ces listes chaînées
 - o Une fonctionnalité de sauvegarde de la liste chaînée doit être implémentée de 3 manières différentes. Cette sauvegarde se fait dans un fichier CSV ou une base de données SQLite :
 - Une commande « save » (ou équivalent) dans l'interface interactive du programme ;
 - Un signal SIGUSR1 envoyé au programme sauvegarde automatiquement ;
 - Un signal SIGINT (CTRL+C) doit demander si l'utilisateur veut :
 - Quitter sans sauvegarder ;
 - Quitter en sauvegardant ;
 - Ne pas quitter.
- Concernant a minima les opérations CRUD, le choix de la fonction à utiliser doit se réaliser via un tableau de pointeurs sur fonction
- Un parser doit évidemment être implémenté pour lire un CSV ou SQLite et le transformer en listes chaînées.
- Attention à la clarté et la propreté de votre code. Un malus pouvant monter jusqu'à 4 points peut être appliqué.
- Implémenter une fonction d'imports de fichier CSV et/ou SQLite, qui serait fusionné avec le fichier CSV ou SQLite existant. Ex : tout est stocké dans une base de données SQLite, mais l'utilisateur peut ajouter du contenu en donnant un fichier CSV contenant de nouvelles entrées au programme. Le programme doit évidemment vérifier qu'une entrée n'existe pas déjà. Un rapport contenant les doublons non ajoutés doit être généré.
- Les listes chaînées sont doublement chaînées et/ou circulaires. Il faut que l'utilisation de ces listes chaînées soit également implémentée.
- Les données au repos doivent être chiffrées avec un algorithme de chiffrement symétrique standard (AES par exemple).

Ouvrages de référence (livres, articles, revues, sites web...)

Outils informatiques à installer

gcc, vim

4 Livrables et étapes de suivi

5 Soutenance

Durée de présentation
par groupe :

30 min

Audience : **A huis clos**

Type de présentation :

Présentation / PowerPoint - Démonstration

Précisions :