

Rapport Projet A&POO

Pour générer l'exécutable, il suffit d'exécuter le makefile présent dans le dossier logicielREE.

I) Fonctionnement du logiciel

Le logiciel utilise un menu console pour fonctionner, l'utilisateur peut naviguer en choisissant le numéro de l'option voulue afin d'ajouter, de modifier ou d'afficher des informations.

À tout moment, lors de la saisie d'informations, si cette dernière est incorrecte une boucle nous permet de faire une nouvelle saisie jusqu'à ce qu'elle soit correcte. Si l'on souhaite sortir de toutes boucles de saisie, il suffit de saisir la chaîne "STOP" pour revenir au menu précédent.

À la fin d'une saisie d'informations, le programme vérifiera que la saisie n'entre pas en conflit avec les choix d'implémentations des objets.

II) Choix d'implémentation

Nous avons décidé de créer un fichier DateHeure qui contient une classe Date et une classe Heure afin de pouvoir les comparer plus facilement. En effet, nous avons surchargé les opérateurs de comparaison pour ces classes et ainsi, il est plus simple pour les autres objets comme rendez-vous, d'utiliser ces comparateurs afin de déterminer si une date en chevauche une autre par exemple.

Pour les rendez-vous, nous utilisons une liste static qui les répertorie tous, mais aussi une liste dans la classe etudiant, qui contient donc tous les rendez-vous de cet étudiant, de même pour les entreprises. Nous utilisons des listes car il est facile d'insérer de nouveaux éléments n'importe où. Avant d'ajouter un rendez-vous, on parcourt la liste déjà existante et pour chaque élément, on regarde si les rendez-vous se chevauchent, si c'est le cas, on renvoie une exception à l'appelant pour signaler l'erreur. Si les rendez-vous ne se chevauchent pas, on regarde si celui que l'on souhaite ajouter se passe avant, si oui, on l'insère dans la liste, sinon on regarde le rendez-vous suivant. Une fois le rendez-vous ajouté à la liste globale de tous les rendez-vous, on sait qu'il est valide, donc on l'ajoute dans la liste de l'étudiant et de l'entreprise concernée dans l'ordre chronologique également. Pour comparer les rendez-vous, on utilise donc les opérateurs surchargés présent dans la classe Date et la classe Heure.

Afin d'avoir une meilleure flexibilité, nous avons utilisé les exceptions. Lors de la création de date ou d'autre objet, le constructeur vérifie que les arguments envoyés sont conformes aux règles que nous nous imposons, par exemple, on ne peut pas créer de date

avec un jour négatif. Ainsi, le constructeur envoie une exception qui est récupérée par l'appelant, et donc, il est possible d'informer l'utilisateur de son erreur et de poursuivre.

III) Les améliorations

Le choix de relier ce projet à une base de donnée aurait été plus adéquat. En effet, cela nous aurait permis de contenir tous les étudiants, entreprises et rendez-vous et donc de pouvoir les réutiliser après la fermeture du logiciel, car dans la version actuelle du projet à chaque fois que nous démarrons le logiciel, il charge les données écrites en dur dans le main, pour avoir une base de test minimale.

Utiliser ce projet avec une interface graphique nous aurait permis de mieux visualiser le logiciel conçu et d'adapter un design pour les options. Cela aurait aussi permis de ne pas se tromper dans la saisie grâce aux listes déroulantes et informations prédéfinies.