

Projet 2020-2021

Analyse et Programmation Orientées Objets / C++

Ce projet a pour but de vous faire réaliser un mini logiciel depuis la phase d'analyse jusqu'à la phase d'implémentation. Les choix d'analyse seront exprimés au travers de diagrammes UML et l'implémentation se fera en C++.

Ce projet est à rendre avant le 18 décembre 2020 à 23h59. Vous devrez rendre un dossier compressé (.zip) nommé tel que **Nom1,2_Prenom1,2_Groupe_Drive.zip** et contenant le code, ainsi qu'un rapport (format PDF). L'espace de dépôt se trouve dans l'onglet "Projet" sur la plateforme pédagogique. Nous n'accepterons aucun rendu par mail. Tout retard entraînera un malus de 2 ^(nombre_de_jours_de_retard) points.

Sujet : la gestion des vols

La gestion de vols c'est le suivi en temps réel le déroulement des services offert par une compagnie aérienne. Elle offre une vision globale sur l'organisation et la qualité des services.

On a essayé de concevoir une application qui se rapproche le plus possible des outils de gestion de vol qu'on trouve chez les compagnies aériennes en utilisant un langage qui contient la notion d'orienté objet comme le langage c++

Vous devez établir un logiciel permettant la gestion des vols.

1. Partie Analyse :

Pour commencer notre analyse il faut d'abord trouver tous les relations intérieur et extérieur d'une compagnie aérienne.

Passager : un passager est identifié par les champs suivants :

- Nom
- Prénom
- Age
- Titre (M, Mlle ou Mme)
- N° passeport

Un passager peut réserver un vol.
Un passager peut confirmer une réservation.
Un passager peut annuler une réservation.
Un passager peut afficher la liste des vols.
Un passager peut vérifier l'existence d'un vol.
Un passager peut vérifier l'existence d'une réservation.

Vol : Un vol est identifié par les champs suivants :

- N° vol
- Nombre de place maximal
- Destination
- Date
- Prix

Date : La date est composée de deux champs principaux :

- Date (jour / mois / année)
- Heure (heure : minute)

Destination : La destination est composée de deux champs :

- Ville de départ
- Ville d'arrivée

Réservation : La réservation est composée de cinq champs :

- Numéro de la réservation
- Numéro de passeport
- Numéro de vol
- Confirmation (oui/non)

Administrateur : La destination est composée d'un seul champ :

- Identifiant
- Mot de passe

Un administrateur peut se connecter.
Un administrateur peut ajouter un vol.
Un administrateur peut ajouter un passager.
Un administrateur peut ajouter une réservation.
Un administrateur peut ajouter une destination.

Un administrateur peut modifier l'heure d'un vol.
Un administrateur peut modifier la date d'un vol.
Un administrateur peut afficher la liste des vols.
Un administrateur peut vérifier l'existence d'un vol.
Un administrateur peut afficher la liste des passagers par vol.

Vous pouvez ajouter aussi les fonctionnalités

Supprimer/ modifier un vol, un passager, une réservation, une destination

Afficher la liste des passagers par destination

Ajouter des statistiques (nb passagers, nb destinations, nb passagers par jour, nb vols, nb vols par destination Et plusieurs d'autres)

2. Partie pratique:

Dans la partie pratique on va essayer de traduire les résultats d'analyse a un programme orientée objet a l'aide du langage c++.

Notre projet va comporter 6 classes soient : Passager, Administrateur, Destination, Vol, Réservation, Date.

Le logiciel devra assurer les fonctionnalités suivantes :

Méthode **Recherche** pour traiter l'existence d'un vol.

Méthode **Recherche** pour traiter l'existence d'une réservation.

Méthode pour **Ajouter** un vol.

Méthode **Confirmer** pour confirmer une réservation.

Méthode **Annuler Réservation** pour annuler une réservation.

Méthode **Réserver** pour réserver un vol.

Méthode **Modifier Date** pour modifier la date d'un vol.

Méthode **Modifier Heur** pour modifier l'heure d'un vol.

Méthode **Afficher** pour afficher la liste des vols.

Méthode **Afficher** pour afficher la liste des passagers qui ont réservé dans un vol.

Méthode **Contrôle Date** pour éviter de organiser un vol dans un date qui déjà passer.

Base de données :

Vous pouvez utiliser des fichiers texte comme une base données avec les deux méthodes suivantes :

Méthode **Ecrire fichier** pour stocker toutes les données dans un fichier (.txt).

Méthode **Lire fichier** pour charger toutes les données à partir un fichier (.txt).

Si non :

Vous pouvez utiliser une base de données classique

Travail à effectuer :

Le travail devra se faire en **binôme (bi = 2 !)** et fera l'objet d'un rapport ainsi que d'une présentation sur ordinateur. Le rapport devra contenir :

- ✚ Le diagramme des cas d'utilisations indiquant les fonctionnalités du système.
- ✚ Le diagramme de classe, séquence (3 exemples) avec des explications sur les choix de modélisation effectués ;
- ✚ Le diagramme des composants indiquant la répartition du code dans les fichiers, les relations entre ces fichiers, l'utilisation de bibliothèques, ...
- ✚ Des éléments sur le code : explications de certains algorithmes, choix d'implémentation (choix de conteneurs, utilisation d'algorithmes de la bibliothèque standard, ...) ;

Remarques :

- ✚ L'utilisation des bibliothèques standard C++, de ses conteneurs et des algorithmes (tri, ...) est, bien entendu, très vivement conseillée et constituera

un critère important pour la notation.

✚ Toute tentative de fraude : recopie d'un code d'un autre binôme, recopie du code d'un étudiant de l'an dernier, récupération d'un code sur internet (ici cela va être difficile !) sera sévèrement puni avec un passage en commission de discipline avec risque d'exclusion pour tous diplômes universitaires pendant une durée de 5 ans. Ceci a été effectivement mis en application pour 6 étudiants il y a 2 ans et pour 3 étudiants l'an dernier (Master 1 et Licence 3) ; les conséquences ont été importantes pour ces étudiants : annulation du semestre et de l'année, interdiction pendant 1 an de toutes inscriptions dans le supérieur en France.

Exemple de réalisation :

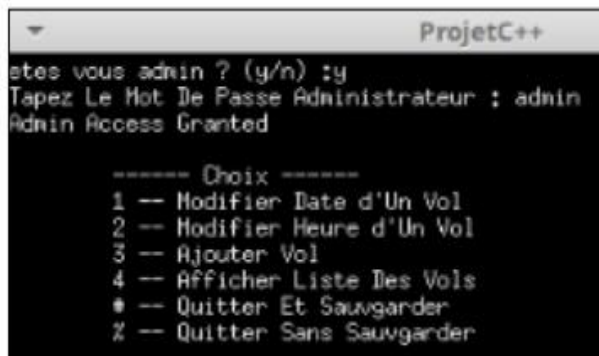


Fig.1 Interface administrateur

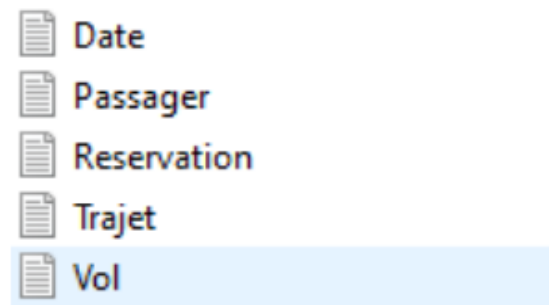


Fig.2 Base de données