

# SAE.S01.S02.2023

**Sujet :** Gestion des vols pour un aéroport - Projet GESTION'AIR

Client : Directeur de l'aéroport Grenoble Alpes Isère

Le directeur de l'aéroport Grenoble Alpes Isère souhaite développer une application de gestion des vols commerciaux au départ de celui-ci. L'application devra proposer plusieurs fonctionnalités :

- Affichage des informations sur les vols sur les tableaux dédiés
- Recherche d'un vol à partir du nom de la compagnie, de la destination ou de l'heure de décollage
- Affichage de la liste des passagers sur les écrans des salles d'embarquement
- Gestion des retards et des annulations des vols
- Maximiser l'utilisation de la piste

Afin d'implémenter ces fonctionnalités, l'application disposera des informations suivantes pour chaque vol :

- Le numéro
- La compagnie
- La destination
- Le numéro du comptoir d'enregistrement
- Les heures de début et de fin de l'enregistrement
- Le numéro de la salle d'embarquement
- Les heures de début et de fin de l'embarquement
- L'heure de décollage
- L'état du vol (à l'heure, retardé [entre 1 mn et 60 mn], annulé)
- La liste des passagers

Pour chaque passager, l'application disposera des informations suivantes :

- Nom
- Prénom
- Date de naissance au format JJ/MM/AAAA
- Numéro du siège
- Prix de son billet

Toutes ces informations sont décrites avec des caractères alphanumériques uniquement (pas d'accents, tiret, caractères spéciaux.) : par exemple un tiret dans un nom sera remplacé par un espace.

L'affichage des informations sur les vols doit indiquer la liste des vols (une ligne par vol) dans les 3 heures à venir triée par ordre croissant sur l'heure de décollage. Chaque ligne sera composée de :

- L'heure de décollage
- Le numéro de vol
- Le nom de la compagnie
- La destination
- Le comptoir d'enregistrement, ainsi que les heures de début et de fin de l'enregistrement si c'est d'actualité (à afficher 30mn avant et 10mn après)
- La salle d'embarquement, ainsi que les heures de début et de fin de l'embarquement (à afficher 30mn avant et 10mn après)
- L'état du vol (à l'heure, retardé (nombre de minutes), annulé)

La recherche d'un vol à partir du nom de la compagnie, de la destination ou de l'heure de décollage, peut éventuellement afficher plusieurs vols vérifiant le critère donné. On peut envisager de donner la possibilité à l'utilisateur d'ajouter un critère ou plusieurs critères pour affiner la recherche au sein de cette liste vols.

Les écrans d'affichage des salles d'embarquement doivent indiquer la liste des passagers triée sur l'ordre d'embarquement. Cet ordre est à calculer en se basant sur :

- l'âge du passager : les enfants de moins de 12 ans d'abord
- le prix du billet dans l'ordre décroissant

En cas d'égalités, on départage les passagers selon un ordre alphabétique basé sur le nom.

La gestion des retards et des annulations consistera à tenter de reprogrammer un vol retardé **au plus tôt**. L'aéroport ne disposant que d'une piste pour le décollage des vols commerciaux, il faudra respecter la contrainte de sécurité qui impose un délai de 5 minutes minimum entre deux décollages. Il faudra également garantir un "couvre feu" pour limiter les nuisances vis à vis des riverains de l'aéroport. Celui-ci impose que les décollages ne peuvent avoir lieu qu'entre 6h et 22h. Ainsi, la reprogrammation d'un vol retardé peut conduire à l'annulation de vols qui ne peuvent pas décoller avant 22h.

La dernière fonctionnalité est non prioritaire. Elle constitue un bonus qui ne sera envisagé que lorsque les autres fonctionnalités sont opérationnelles. Elle consiste à adapter la gestion des retards et des annulations afin de maximiser l'utilisation de la piste. Il s'agit donc de reprogrammer les vols retardés de telle sorte à maximiser le nombre de décollages sur la journée.

Les données requises par l'application seront fournies dans un fichier au format csv selon le format suivant :

- une ligne par vol :
  - Numéro de vol
  - Compagnie
  - Destination
  - Numero comptoir d'enregistrement
  - Heure debut enregistrement
  - Heure fin enregistrement
  - Numero de la salle d'embarquement
  - Heure debut embarquement
  - Heure fin embarquement
  - Heure decollage
  - Etat vol (A l'heure, Retarde (durée retard), Annulé)
  - Liste des passagers (séparés par le caractère “;”)
    - Nom
    - Prénom
    - Date de naissance
    - Numéro de siège
    - Prix billet

## Partie 1 : Algorithmique

La partie analyse, en amont de l'implémentation en C, constituera la SAÉ d'algorithmique. Il s'agira ici d'identifier / choisir les différents algorithmes qui seront implémentés, de donner leurs spécifications formelles (contrats ou parties statiques), leurs codes, leurs complexités théoriques. On pourra envisager d'évaluer leur efficacité pratique à travers une implémentation en langage python qui se chargera de dénombrer les opérations élémentaires lors d'une exécution. Dans le cas où vous auriez identifié plusieurs algorithmes possibles pour traiter un même problème, vous pouvez vous appuyer sur tous ces éléments pour justifier le choix de l'algorithme retenu.

Sachant que les données sur les vols seront représentées dans des tableaux de structures en langage C, nous vous proposons de simplifier le type des données manipulées lors de l'évaluation de l'efficacité pratique des algorithmes. Vous pourrez donc utiliser des tableaux d'entiers afin de valider vos choix en amont de leur implémentation dans l'application en langage C. Il faudra générer des tableaux d'entiers avec des valeurs se rapprochant le plus possible de la nature des données à traiter dans l'application finale.

Par exemple, on peut envisager de représenter les horaires avec des entiers à 3 ou 4 chiffres au format HHMM concaténant les heures et les minutes. Pour les villes, on peut considérer des entiers qui représentent un numéro d'identification dans une base de données...

Par ailleurs, une évaluation rigoureuse de l'efficacité pratique d'un algorithme nécessite d'avoir des tableaux de différentes tailles et propriétés (triés dans l'ordre croissant / décroissant, non trié). Pour simplifier, nous vous proposons, pour chaque fonctionnalité de l'application, de définir 3 classes de tableaux selon leurs tailles (petite, moyenne, grande) et de créer pour chaque classe 10 tableaux remplis de valeurs entières générées aléatoirement. Ensuite, on pourra se contenter de calculer la moyenne des nombres d'opérations élémentaires effectuées sur les tableaux d'une classe.

Le rendu sera un document d'au plus 15 pages au format pdf à déposer sur Moodle au plus tard le jeudi 11/01 au soir, ainsi que le fichier python (un seul) élaboré pour effectuer l'analyse demandée.

## Partie 2 : Langage C

Pour des raisons évidentes de sécurité aéroportuaire, le directeur de l'aéroport est intransigeant sur la qualité du code de l'application.

Lors de la présentation de l'application, il sera absolument nécessaire de mettre en place tous les éléments nécessaires pour le convaincre que les fonctionnalités sont opérationnelles dans tous les cas de figures envisageables, mais aussi que le code respecte les standards de qualité professionnelle. Vous pouvez envisager une démonstration partant de la compilation à l'exécution du programme sur des fichiers de tests. En outre, vous pourrez vous appuyer sur un diaporama de 5 pages maximum pour démontrer que les fonctionnalités sont opérationnelles et que le code est de qualité professionnelle.

Lors de la démonstration, il faudra envisager la saisie de l'heure actuelle, mais aussi permettre la saisie des informations sur un vol retardé.

L'évaluation se fera donc à l'occasion d'une présentation la semaine du 15 janvier. Le code de l'application est à rendre au plus tard le jeudi 11/01 au soir sur GitLab (projet privé sur <https://forge.univ-lyon1.fr/> en invitant votre enseignant comme développeur).