|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | Mai 2020  Projet informatique intégré | |
| Enoncé final :  Gestion de vols dans  une compagnie aérienne | | | | |
| Taxi pour Bruxelles. Transfert aéroportuaire à tarif fixe ! | | | | |
| Jonathan SMITH et Pauline LOREA, 2IGA  A l’intention de Mme Françoise DUBISY  et Mme Chantal BERTRAND | |  |  | Département technique de Marche-en-Famenne | HENALLUX |

Table des matières

[Enoncé 3](#_Toc68879993)

[Tâche métier 4](#_Toc68879994)

[Thread supplémentaire 4](#_Toc68879995)

[Recherche numéro 1 5](#_Toc68879996)

[Recherche numéro 2 6](#_Toc68879997)

[Recherche numéro 3 7](#_Toc68879998)

[Schéma EA 8](#_Toc68879999)

# Enoncé

Pour ce projet, nous avons décidé de nous inspirer de la série 5 provenant du cours de conception de base de données.

Le programme que nous vous proposons permettra à un aéroport de gérer les réservations de places sur des vols aériens.

Un vol est identifié par un numéro de vol. Il est caractérisé par un type d’appareil (Airbus, Boeing...), sa durée (prévue), la date et l’heure de départ. Les aéroports de départ et de destination sont mémorisés, ainsi que les (aéroports des) escales éventuelles. La durée prévue des escales est précisée.

On précise également si le vol propose des repas ou non et éventuellement une description de celui-ci

Un des objectifs de ce programme est de pouvoir informer les passagers sur les aéroports qu’ils fréquenteront (départ, destination et escales). A partir d’un code de 3 lettres différent pour chaque aéroport (par exemple, BRU pour Brussel Airport ou CRL pour Charleroi), il faut pouvoir retrouver le nom de l’aéroport correspondant, le pays et le nombre de terminaux. De plus, certains aéroports (régionaux) n’ont pas de services administratifs propres ; ils sont sous la responsabilité administrative d’un aéroport national. Tout aéroport régional est sous la responsabilité d’un (et un seul) aéroport national. Ce type d’information sera utile pour la gestion administrative de l’aéroport.

Un vol dispose d’un certain nombre de places qui peuvent être attribuées à des passagers. Une place a un numéro qui est unique au sein du vol (c’est-à-dire qu’il n’y a qu’une seule place numéro 12b au sein du même vol, mais la place numéro 12b existe sur plus d’un vol. Il y a plusieurs catégories de places, à savoir, charter, normal, business et première classe. Il est précisé également si la place est du coté hublot ou non.

Une place sur un vol ne peut être attribuée qu’à un et un seul passager. Un passager ne se verra attribuer qu’une seule place sur un même vol. Il peut cependant réserver des places sur plus d’un vol. Un passager sera caractérisé par un nom, un prénom, les initiales des prénoms suivants (au total, on ne garde des informations qu’au maximum sur trois prénoms, si du moins le passager en possède plus d’un), sa date de naissance et son sexe. La combinaison du nom, du prénom et de l’initiale du second prénom est identifiante. Par conséquent, l’initiale du second prénom de chaque passager doit être précisée (l’initiale du second prénom est remplacée par un caractère spécial pour les passagers qui ne possèdent qu’un seul prénom).

Les pilotes sont enregistrés dans la base de données. Un pilote a un nom, un numéro de licence et un nombre d’heures de vol. Tout pilote a un numéro de licence qui lui est propre.

Le programme s'ouvrira sur un menu qui permettra de :

- D'ajouter un nouveau vol à la base de données de la compagnie

- Modifier les informations d'un vol déjà existant

- Supprimer un ou plusieurs vol(s)

- Lister tous les vols

- un bouton de recherche

- un formulaire de recherche consiste à :

- rechercher tous les vols à une certaine date (afficher aéroport de départ, arrivé + escale éventuelle + type d’appareil)

- rechercher/lister les passagers d’un vol (spécifier l’aéroport, l’éventuelle escale et les infos de la place de chaque passager)

- rechercher tous les vols d’un pilote ainsi que les aéroports de départ et d’arriver, et le type d’avion

# Tâche métier :

Planification et optimisation des horaires de travail des pilotes sur les vols.

Planification :

Vérifier si un pilote n'est pas inscrit sur un vol et un autre qui a lieu en même temps mais aussi qu'un même pilote ne fasse pas trop d'heures de vols sur une journée (nous avons établi 5h max par jour).

Optimisation :

Faire en sorte que le programme propose pour un certain vol, en priorité le ou les pilote(s) qui se trouvent déjà à l'aéroport de départ (toujours en tenant compte que le vol ne fera pas dépasser le nombre d'heures de travail par pilote par jour).

C'est le pilote qui sera déjà à l'aéroport de départ et qui aura le moins d'heures de travail attribuées sur la journée qui a le plus de chance d'être mis en avant pour effectuer le vol.

# Thread supplémentaire

Le thread supplémentaire permettra la synchronisation de l’animation d’une image à travers la fenêtre de bienvenue.

Une image PNG d’un avion défilera continuellement à travers la fenêtre de bienvenue afin de simuler le vol de celui-ci.

Les classes [Image](https://docs.oracle.com/javase/7/docs/api/java/awt/Image.html), [ImageIcon](https://docs.oracle.com/javase/7/docs/api/javax/swing/ImageIcon.html) et [ImageIO](https://docs.oracle.com/javase/7/docs/api/javax/imageio/ImageIO.html) pourront nous permettrons d’intégrer une image à l’interface graphique du programme et d’en faire un objet pour pouvoir la manipuler.

# Recherche numéro 1

**Titre de la recherche** : Vols prévu à une date.

**Objectif de la recherche** : Donner les informations de tous les vols qui ont lieu à une certaine date dans la compagnie.

**Jointure entre les tables suivantes (au moins 3)** :

Table 1 : Flight Table 2 : Plane Table 3 : Stop Table 4 : Airport

**Entrées :**

**Critère de recherche Format** (type de composant swing)

Date de début JSpinner

Date de fin JSpinner

**Sorties** (dans une JTable) :

**Nom** **de** **la** **colonne** provenant de la table **Nom** **de** **la** **table**

Number Flight

FlightTime Flight

DepartureTime Flight

StopTime Stop

Code Airport

Name Airport

Country Airport

# Recherche numéro 2

**Titre de la recherche** : réservation des sièges d’une catégorie donnée.

**Objectif de la recherche** : Donner les informations de tous les passagers, vols et aéroport de destination correspondant aux réservations de sièges d’une catégorie donnée

**Jointure entre les tables suivantes (au moins 3)** :

Table 1 : Flight Table 2 : Seat Table 3 : Passenger

Table 4 : Airport Table 5 : Stop

**Entrées :**

**Critère de recherche Format** (type de composant swing)

Catégorie de sièges JComboBox proposant les noms des catégories

(reprises dans la table Seat)

**Sorties** (dans une JTable) :

**Nom** **de** **la** **colonne** provenant de la table **Nom** **de** **la** **table**

LastName Passenger

FirstName Passenger

Number Seat

Number Flight

DepartureTime Flight

FlightTime Flight

Code Airport

Name Airport

Country Airport

# Recherche numéro 3

**Titre de la recherche** :Vols d’un pilote.

**Objectif de la recherche** : Afficher tous les vols d’un pilote ainsi que les aéroports de départ et d’arriver et l’avion

**Jointure entre les tables suivantes (au moins 3) :**

Table 1 : Flight Table 2 : Pilot Table 3 : Plane Table 4 : Airport

**Entrées** :

**Critère de recherche Format** (type de composant swing)

Pilote JComboBox proposant les noms des pilotes

**Sorties** (dans une JTable) :

**Nom de la colonne** provenant de la table **Nom de la table**

Number Flight

DepartureTime Flight

FlightTime Flight

Number Plane

TypeName Plane

Départ : Name Airport

Depart : Country Airport

Arrivée : Name Airport

Arrivée : Country Airport

# [Schéma EA](https://app.diagrams.net/#Hpauline-lr%2FPOOA-Projet%2Fmain%2Fsch%C3%A9ma%20conceptuel%20.drawio)

