

PROJET Air Chance

CHANCE
AIRFRANCE 

TABLE DES MATIÈRES

Introduction

I - Diagrammes

Diagramme UML

II - Schémas des relations

Modèle relationnel

Les contraintes d'intégrités référentielles

Les domaines

Les formes normales

IV - Procédures et Triggers

Introduction

Le projet porte sur la réalisation d'une application JAVA qui communique avec une base de données sur un serveur distant à travers le protocole JDBC. On utilisera pour cela les technologies de programmation JAVA, JDBC pour le partie applicative et ORACLE (SQL, PL/SQL) pour la partie base de données.

L'objectif est de réaliser une application fonctionnelle traçant le service de réservation de vols . Ce service est nommé Air Chance .

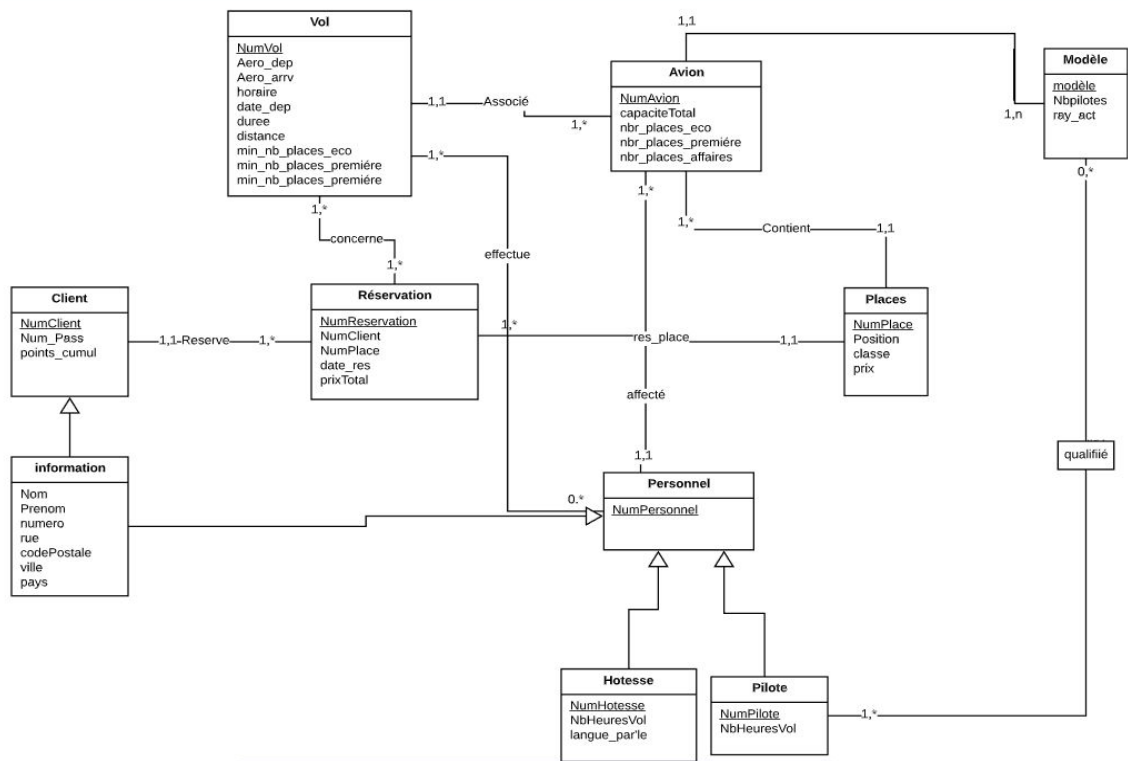
❑ Le client pourra réserver un vol de son choix et bénéficie d'une remise en fonction de ses points de fidélités.

Toutes les transactions, c'est-à-dire les réservations et les locations du client seront sauvegardés dans la base données.

❑ Le superviseur quant à lui pourra planifier les vols .

I - Diagrammes

A. Diagramme UML



II - Schémas des relations

A. Modèle relationnel

Pour modéliser cette application, on a défini le schéma relationnel donné ci-dessous. Les identifiants des relations sont les attributs notés en caractères soulignés et les attributs externes sont précédés :

Vol (NumVol, NumAv, Aero_dep, Aero_arrv, horaire, date_dep, duree, distance, min_nb_place_eco, min_nb_place_premiere, min_place_affaire)

Réservation(NumReservation, NumPlace, NumVol, numClient, date_res, prixTotal)

Avion (NumAvion, model, nb_place_eco, nb_place_premiere, nb_place_affaire)

Modèle(modele, Nbpilotes, ray_act)

Place(NumPlace, Position, classe, prix)

Client(NumClient, Num_Pass, points_cumul, nom, prenom, numero, rue, codepostale, ville, pays)

Pilote(NumPilote, NbHeuresVol, nom, prenom, numero, rue, codepostale, ville, pays)

Hotesse(NumHotesse, NbHeuresVol, langues, nom, prenom, numero, rue, codepostale, ville, pays)

Affecte(numAvion, numHotesse, numPilote)

Effectue(numVol, numAvion, numHotesse, numPilote)

Qualifie(MODELE, NUMPILOTE)

B. Les contraintes d'intégrités référentielles

$\text{Vol}[\text{NumAvion}] \subset \text{Avion}[\text{NumAvion}]$
 $\text{Reservation}[\text{NumClient}] \subset \text{Client}[\text{NumClient}]$
 $\text{Reservation}[\text{NumPlace}] \subset \text{Place}[\text{NumPlace}]$
 $\text{Pilote}[\text{NumPilote}] \subset \text{Personnel}[\text{NumPersonnel}]$
 $\text{Hotesse}[\text{NumHotesse}] \subset \text{Personnel}[\text{NumPersonnel}]$
 $\text{Personnel}[\text{NumPersonnel}] = \text{Hotesse}[\text{NumHotesse}] \cup \text{Pilote}[\text{NumPilote}]$
 $\text{Pilote}[\text{NumPilote}] \cap \text{Hotesse}[\text{NumHotesse}] = \emptyset$
 $\text{Adresse}[\text{id}] \subset \text{Client}[\text{NumClient}]$
 $\text{Adresse}[\text{id}] \subset \text{Personnel}[\text{NumPersonnel}]$

C. Les domaines

$\text{dom}(\text{pays}) = \text{dom}(\text{ville}) = \text{dom}(\text{nom}) = \text{dom}(\text{prenom}) = \text{dom}(\text{Aero_dep}) = \text{dom}(\text{Aero_arrv}) =$
 $\text{dom}(\text{langues}) = \text{dom}(\text{model}) = \text{dom}(\text{Position}) = \text{dom}(\text{classe}) = \text{String}$

$\text{dom}(\text{NumVol}) = \text{dom}(\text{NumAv}) = \text{dom}(\text{duree}) = \text{dom}(\text{distance}) = \text{dom}(\text{min_nb_place_eco}) =$
 $\text{dom}(\text{min_nb_place_premiere}) = \text{dom}(\text{min_place_affaire}) = \text{dom}(\text{NumReservation}) =$
 $\text{dom}(\text{NumPlace}) = \text{dom}(\text{NumVol}) = \text{dom}(\text{numClient}) = \text{dom}(\text{prixTotal}) = \text{dom}(\text{NumAvion}) =$
 $\text{dom}(\text{nb_place_eco}) = \text{dom}(\text{nb_place_premiere}) = \text{dom}(\text{nb_place_affaire}) = \text{dom}(\text{Nbpilotes}) =$
 $\text{dom}(\text{NumPlace}) = \text{dom}(\text{Position}) = \text{dom}(\text{prix}) =$
 $\text{dom}(\text{NumClient}) = \text{dom}(\text{Num_Pass}) = \text{dom}(\text{points_cumul}) = \text{dom}(\text{codepostale}) = \text{dom}(\text{NumPilote}) =$
 $\text{dom}(\text{NbHeuresVol}) = \text{dom}(\text{NumHotesse}) = \text{number}$

$\text{dom}(\text{date}) = \text{date}(\text{date_dep}) = \text{date}(\text{date_arr})$

D. Les formes normales

PREMIÈRE FORME NORMALE

Une relation est en première forme normale si chaque valeur de chaque attribut de chaque tuple est une valeur simple (tous les attributs sont simples et monovalués).

- Tout attribut doit donc être une valeur atomique
- C'est normalement le cas dans le modèle relationnel
- Toute table est donc normalement en 1FN

DEUXIÈME FORME NORMALE

Pour que une relation des tables soit une relation deuxième forme normale il faut déjà qu'elle soit de première forme normale et si l'on définit une clé primaire PK sur cette table, éventuellement constituée de plusieurs colonnes, alors toutes les colonnes de la table doivent dépendre uniquement de la totalité de PK, et non pas de l'une de ses parties.

TROISIÈME FORME NORMALE

Pour la troisième forme normale , elle est un peu comme la deuxième forme normale mais concerne la dépendance entre attributs non clés. Une relation troisième forme normale doit être de deuxième forme normale et qu'aucun attribut ne faisant pas partie de la clé primaire ne doit dépendre d'une partie des autres attributs

Exemples

- Clients

```
create table Clients
(
  numClient      number(5) not null,
  num_pass       varchar(20) not null,
  point_cumul    number(5) not null,
  nom            varchar(20) not null,
  prenom         varchar(20) not null,
  numero         number(4) not null,
  rue            varchar(30) not null,
  codepostale    number(5) not null,
  ville          varchar(20) not null,
  pays           varchar(20) not null,
  constraint pk_clients primary key (numClient)
);
```

La table Clients respecte bien la première forme normale : tous les attributs sont atomiques et non composé .

Elle respecte aussi la deuxième forme normale : numClient est la clef primaire , les attributs non clés num_pass, point_cumul, nom, prenom, numero, rue, code postale, ville, pays dépendent du client . Un client n'a qu'un nom , un prénom ,un seul numéro de passeport ,une seule adresse et code postale et se localise dans une seule ville .

IV - Procédures et Triggers

Procédures

- ❑ Effectuer un vol

```

CREATE OR REPLACE PROCEDURE CHECK_EFFECTUE (
p_numV IN EFFECTUE.NUMVOL%TYPE,
p_numA IN EFFECTUE.NUMAVION%TYPE,
p_numH IN EFFECTUE.NUMHOTESSE%TYPE,
p_numP IN EFFECTUE.NUMPILOTE%TYPE
)
IS
CURSOR EffectueCur IS SELECT * FROM Effectue;
BEGIN
    FOR EffectueRec in EffectueCur loop
        IF EffectueRec.NUMVOL= p_numV THEN
            --msg:='Vol Déjà Enregister !';
            RAISE_APPLICATION_ERROR(-20205,
            'Vol déjà affecter !') ;
        ELSIF EffectueRec.NUMAVION = p_numA THEN
            --msg:='Avion Déjà Enregister !';
            RAISE_APPLICATION_ERROR(-20205,
            'Avion déjà affecter !');
        ELSIF EffectueRec.NUMHOTESSE = p_numH THEN
            --msg:='Hotesse Déjà Enregister !';
            RAISE_APPLICATION_ERROR(-20205,
            ' Hotesse déjà affecter !');
        ELSIF EffectueRec.NUMPILOTE = p_numP THEN
            RAISE_APPLICATION_ERROR(-20205,
            'Pilote déjà affecter !');
        ELSE
            INSERT INTO Effectue (NUMVOL,NUMAVION,NUMHOTESSE,NUMPILOTE)
            VALUES(p_numV,p_numA,p_numH,p_numP);
            --msg:='Affectation Effectuer !';
        END IF;
    END LOOP;
END;
/

```

❑ affecter une équipe et un avion à un vol

```

CREATE OR REPLACE PROCEDURE CHECK_AFFECTATION (
p_numA IN AFFECTE.NUMAVION%TYPE,
p_numH IN AFFECTE.NUMHOTESSE%TYPE,
p_numP IN AFFECTE.NUMPILOTE%TYPE
)
IS
CURSOR AffectCur IS SELECT * FROM Affecte;
BEGIN
    FOR AffectA in AffectCur loop
        IF AffectA.NUMAVION = p_numA THEN
            RAISE_APPLICATION_ERROR(-20205,
            'Avion déjà affecter !');
        ELSIF AffectA.NUMHOTESSE = p_numH THEN
            RAISE_APPLICATION_ERROR(-20205,
            'Hotesse déjà affecter !');
        ELSIF AffectA.NUMPILOTE = p_numP THEN
            RAISE_APPLICATION_ERROR(-20205,
            'Pilote déjà affecter !');
        ELSE
            INSERT INTO Affecte (NUMAVION,NUMHOTESSE,NUMPILOTE)
            VALUES(p_numA,p_numH,p_numP);
        END IF;
    END LOOP;
END;
/

CREATE TRIGGER SUPPRESION_MODELE_AVION
AFTER DELETE
ON ModelesAvions
FOR EACH ROW

```

SQL ▼ Larg

Triggers

❏ Suppression d'un modèle d'avion

```

END LOOP;

END;
/

CREATE TRIGGER SUPPRESION_MODELE_AVION
AFTER DELETE
ON ModelesAvions
FOR EACH ROW
BEGIN
DELETE FROM Avions
WHERE modele=:old.modele;
END;
/

CREATE TRIGGER SUPPRESION_PLACES
AFTER DELETE
ON Avions
FOR EACH ROW
BEGIN
DELETE FROM PLACES
WHERE NUMAVION=:old.NUMAVION;
END;
/

```


Les Tests :

ajouter un Pilote :

```
<terminated> Squelette (1) [Java Application] /usr/lib/jvm/java-11-openjdk-amd64/bin/java (31 mars 2020 à 21:32:59)
6-Gestion des Réservations.
7-Gestion des Places d'Avion.
5
-----Gestion des Personnels-----
-----Veuillez choisir une opération-----
1-Gestion des Pilotes
2-Gestion des Hôtes
3-Affecter un Equipage à un Avion
1
-----Gestion des Pilotes-----
-----Veuillez choisir une opération-----
1-Ajouter un Pilote
2-Afficher tous Les Pilotes
3-Supprimer un Pilote
4-Ajouter un Qualife
1
Nombre d'Heures de Vol:
0
Nom:
galmami
Prénom:
mohamed
Numéro:
4
Rue:
St ursule
Code Postale:
3800
Ville:
Grenoble
Pays:
France
Pilote Ajouter avec Succès!
```

* vérification de l'ajout :

```
1
-----Gestion des Pilotes-----
-----Veuillez choisir une opération-----
1-Ajouter un Pilote
2-Afficher tous Les Pilotes
3-Supprimer un Pilote
4-Ajouter un Qualife
2
Liste des Pilotes:
|Numéro Pilote|Nombre d'Heures.Vol|Nom|Prénom|Numéro|rue|Code Postale|Ville|Pays

|921|0|Francois |thomas|5|chinois|3840|st martin|France

|124|0|galmami|mohamed|4|st ursl|3800|Grenoble|France
```

- supprimer un pilote/hôtes:

```

1
-----Gestion des Pilotes-----
-----Veuillez choisir une opération-----
1-Ajouter un Pilote
2-Afficher tous Les Pilotes
3-Supprimer un Pilote
4-Ajouter un Qualife
3
Veuillez sélectionner un Numéro de Pilote:
[124]
[921]
921
Pilote Supprimer avec Succès!

```

*** vérification :**

```

4
Liste des Pilotes:
|Numéro Pilote|Nombre d'Heures.Vol|Nom|Prénom|Numéro|rue|Code Postale|Ville|Pays

|124|0|galmami|mohamed|4|st ursl|3800|Grenoble|France

```