Blachère Adrien  
Garrido Kevin

Projet Examen Décembre 2013

**1) Descriptif des packages**

- org.jacademie.dao :

Ce package regroupe toutes les classes d’accès aux données

- org.jacademie.dbupdate.ctrl :

Ce package contient la classe qui effectue les appels à toutes les autres classes.

- org.jacademie.domain :

Ce package contient les classes Artistes, Album, Chanson.

- org.jacademie.service.FileReader :

Ce package permet lire les fichiers dans un dossier spécifique.

- org.jacademie.service.parser :

Ce package permet de parser un fichier.

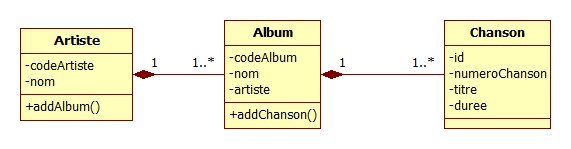
**2) Frameworks choisis**

Nous avons choisi les frameworks suivants :

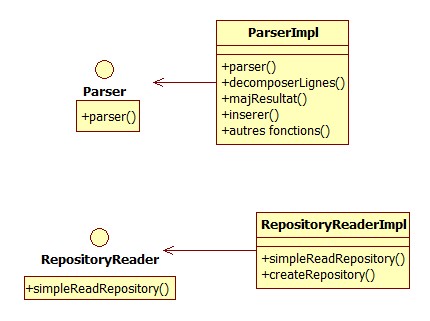
* Junit : Pour pouvoir faire des tests unitaires
* hibernate : Gere la persistance des données
* log4j : Pour pouvoir mettre des loggers et afficher sur la console
* hsqldb : Permet de gerer la base de données relationnelle

**3) Modèle objet**

Modèle objet des entités :



Modèle objet des services :



Modèle objet des DAOs :



**4) Modèle physique de données (tables et colonnes)**

La base contient trois tables : ARTISTE, ALBUM et CHANSON

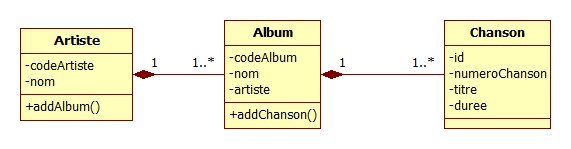
La table ARTISTE contient deux colonnes :

* CODE\_ARTISTE : un entier et une clé primaire de la table,
* NOM : chaîne de caractères.

La table ALBUM contient trois colonnes

* CODE\_ALBUM : clé primaire de la table,
* NOM : chaîne de caractères,
* ARTISTE\_ID : Foreign key sur la colonne CODE\_ARTISTE de la table ARTISTE

La table CHANSON contient cinq colonnes :

* ID : identifiant unique des chansons,
* NUMERO\_CHANSON : entier,
* TITRE : Chaine de caractères,
* DUREE : Entier,
* ALBUM\_ID : Clé étrangère vers CODE\_ALBUM de la table ALBUM.

**5) Fichier de configuration pour lancer l’application**

Le fichier de configuration peut se trouver à la racine du projet. Le code SQL à rentrer est le suivant :

CREATE TABLE ARTISTE(CODE\_ARTISTE INTEGER NOT NULL PRIMARY KEY,NOM VARCHAR(80) NOT NULL);

CREATE TABLE ALBUM(CODE\_ALBUM INTEGER NOT NULL PRIMARY KEY,NOM VARCHAR(80) NOT NULL,ARTISTE\_ID INTEGER,CONSTRAINT FK\_ALBUM FOREIGN KEY(ARTISTE\_ID) REFERENCES ARTISTE(CODE\_ARTISTE));

CREATE TABLE CHANSON(ID INTEGER GENERATED BY DEFAULT AS IDENTITY(START WITH 1) NOT NULL PRIMARY KEY, NUMERO\_CHANSON INTEGER NOT NULL, TITRE VARCHAR(80) NOT NULL, DUREE INTEGER NOT NULL,ALBUM\_ID INTEGER NOT NULL,CONSTRAINT FK\_CHANSON FOREIGN KEY(ALBUM\_ID) REFERENCES ALBUM(CODE\_ALBUM));

De plus pour pouvoir tester le bon fonctionnement du code, il faut modifier le code de la classe App. En effet, le chemin vers le répertoire contenant les fichiers .music correspond au chemin de nos tests.

Le dossier playlist contient plusieurs fichiers qui peuvent être utilisés pour peupler la base de données grâce au programme