

Projet : ArtisteManager

Documentation :

Table des matières

I. Schéma de la base de données :.....	2
II. Cas d'usages de l'application :	5
III. UML des entités :	6

I. Schéma de la base de données :

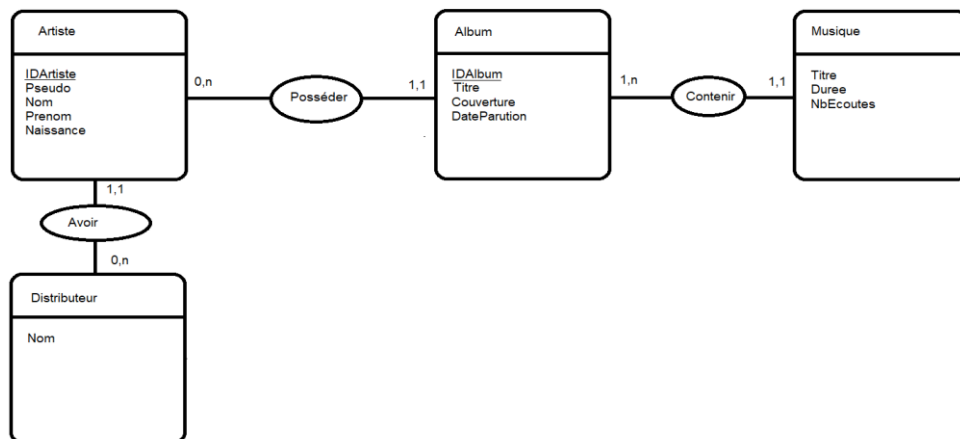


Figure 1 – MCD version 0 du projet ArtisteManager

Pour cette version 0, nous utilisons 4 données différentes :

- Distributeur,
- Artiste,
- Album,
- Musique.

Un **Distributeur** est caractérisé par son nom.

De plus, un **Distributeur** peut être associé à 0 ou plusieurs **Artiste**.

Un **Artiste** est caractérisé par son identifiant, son pseudo, son nom, son prénom et sa date de naissance.

De plus, un **Artiste** peut posséder 0 ou plusieurs **Album**, et a 1 seul **Distributeur**.

Un **Album** est caractérisé par son identifiant, son titre, sa couverture et sa date de parution.

De plus, un **Album** est associé à 1 seul **Artiste**, et peut contenir au moins 1 **Musique**.

Une **Musique** est caractérisée par son titre, sa durée et son nombre d'écoutes.

De plus, une **Musique** est contenue dans 1 seul **Album**.

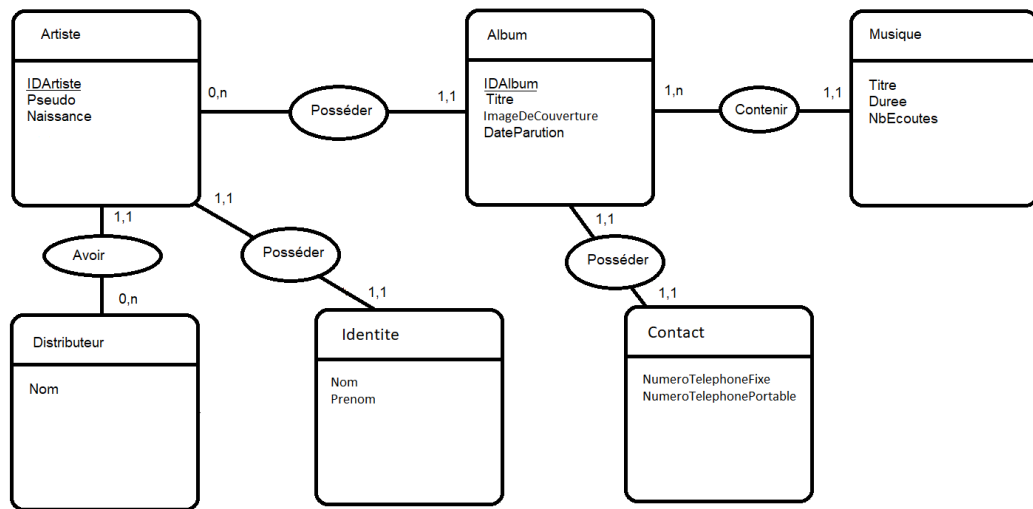


Figure 2 - MCD version 1 de ArtisteManager

Pour la version 1, nous utilisons 6 données différentes :

- Distributeur,
- Artiste,
- Album,
- Musique,
- Identite,
- Contact.

Un **Distributeur** est caractérisé par son nom.

De plus, un **Distributeur** peut être associé à 0 ou plusieurs **Artiste**.

Un **Artiste** est caractérisé par son identifiant, son pseudo, son **Identite** (composé de nom et prénom) et sa date de naissance.

De plus, un **Artiste** peut posséder 0 ou plusieurs **Album**, et a 1 seul **Distributeur**.

Un **Album** est caractérisé par son identifiant, son titre, son image de couverture, son **Contact** (composé du numéro de téléphone fixe et du numéro de téléphone portable de l'artiste) et sa date de parution.

De plus, un **Album** est associé à 1 seul **Artiste**, et peut contenir au moins 1 **Musique**.

Une **Musique** est caractérisée par son titre, sa durée et son nombre d'écoutes.

De plus, une **Musique** est contenue dans 1 seul **Album**.

Les besoins couverts par cette modélisation sont :

- Avoir les musiques associées à chaque album,
- Connaître tous les albums d'un artiste,
- Savoir quel est le distributeur de quel artiste,
- Avoir la liste de tous les artistes d'un distributeur,
- Calculer la durée totale d'un album et son nombre de musiques,
- Supprimer des données selon certains critères.

II. Cas d'usages de l'application :

Les différents cas d'usages de l'application sont :

- Pour la classe Artiste :
 - Opérations CRUD simples : Create, Read, Update et Delete
 - Récupérer tous les albums d'un artiste grâce à son identifiant

- Pour la classe Album :
 - Opérations CRUD simples : Create, Read, Update et Delete,
 - Récupérer un ou plusieurs Album grâce à un titre,
 - Récupérer un ou plusieurs Album qui contiennent une chaîne de caractères dans leur titre,
 - Supprimer un ou plusieurs Album depuis un titre,
 - Récupérer les musiques regroupées par Album.

III. UML des entités :

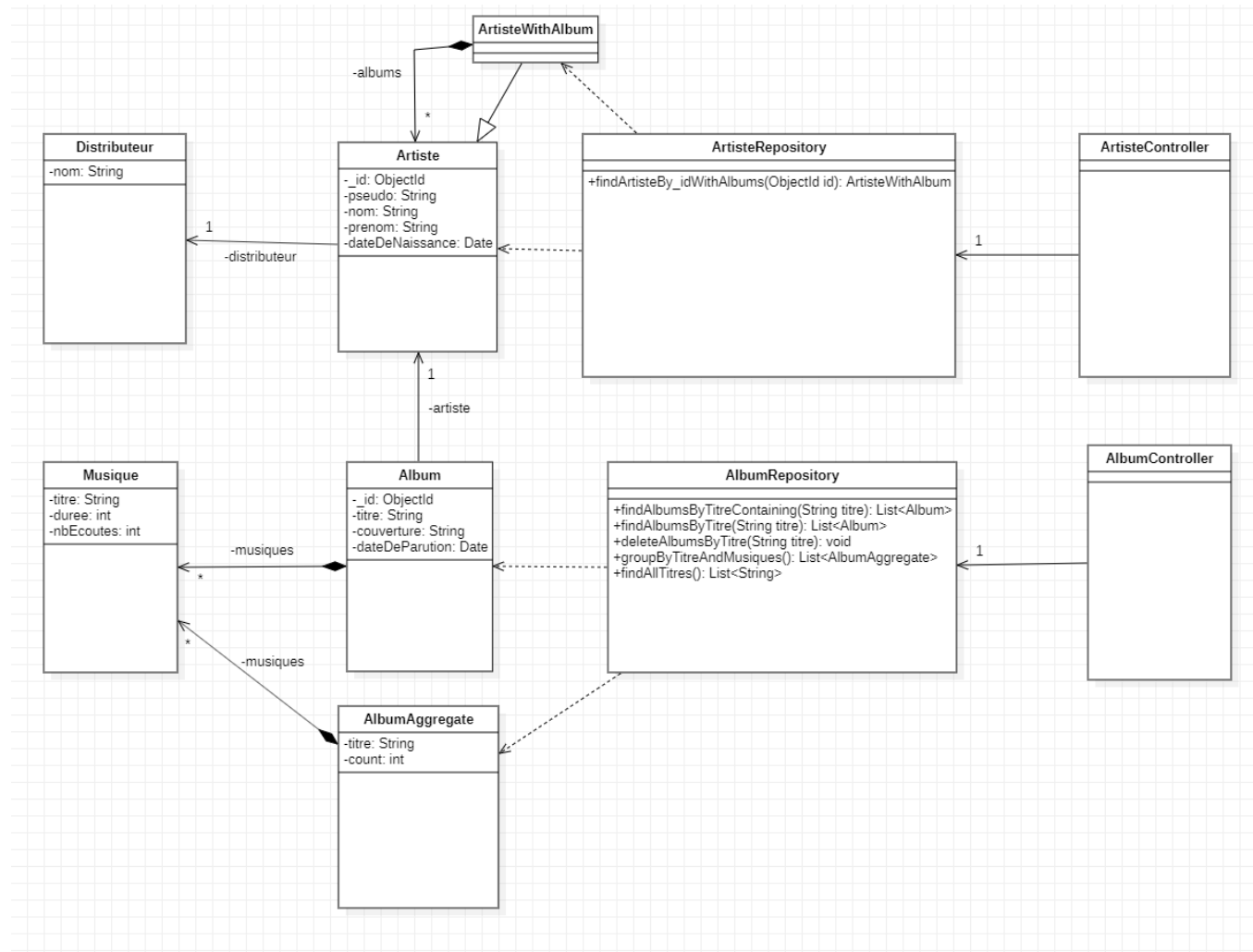


Figure 3 - UML des entités version 0 de ArtisteManager

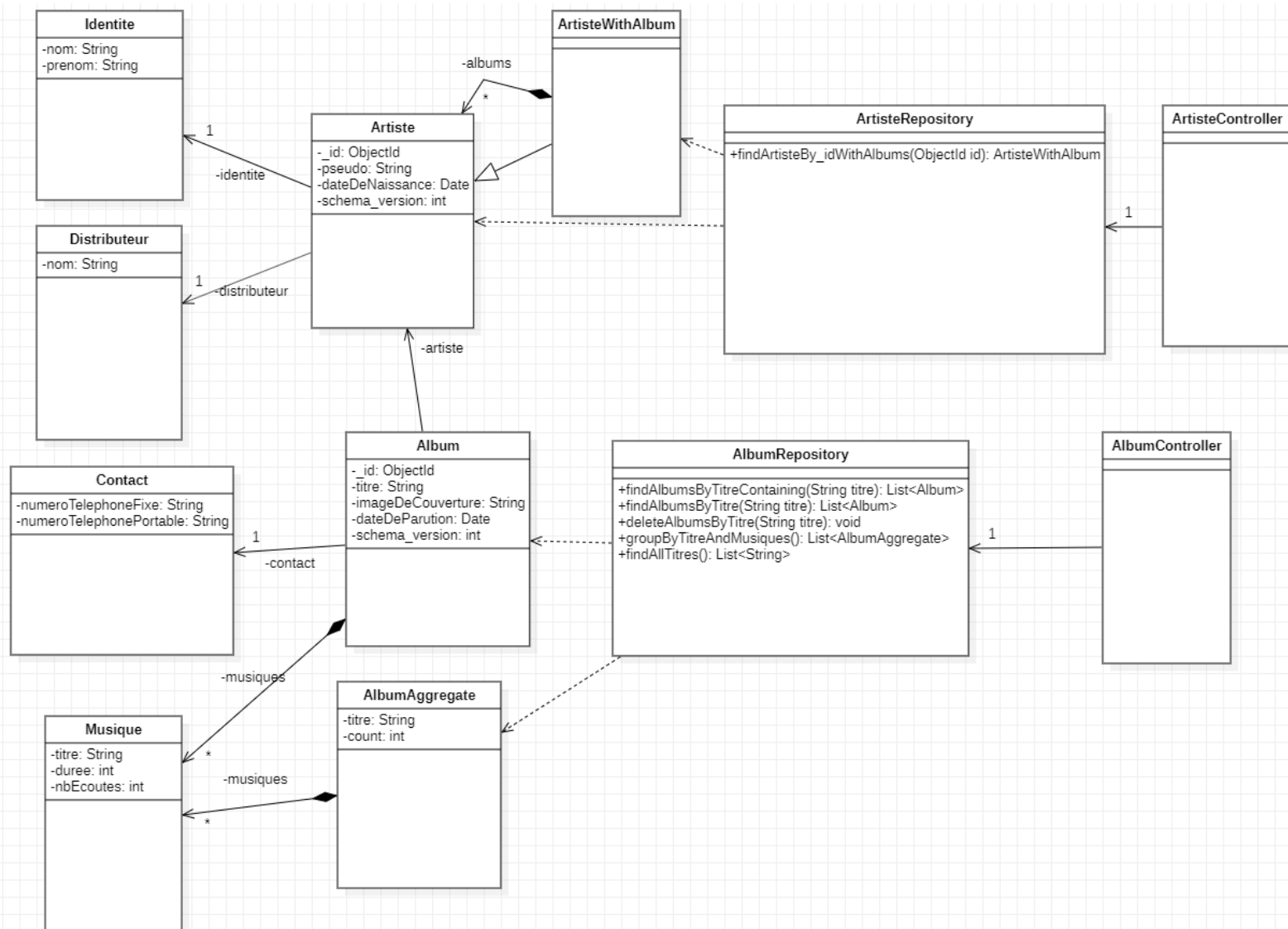


Figure 4 - UML des entités version 1 de ArtisteManager