

Université Nice Sophia Antipolis

IUT département Informatique

Lisez la description de l'étude de cas en entier et toutes les questions jusqu'au bout avant de commencer.

Le barème est donné à titre indicatif, mais il peut être modifié.

Chaque fois que vous butez sur un manque de spécification, faites un choix et explicitez ce choix par une note.

Contrôle -- M3105- S3T

Conception et Programmation Objet Avancées

janvier 2018 - Durée 1h30

Une page recto/verso au format a4 autorisée

Nowadays recommender systems is a software that is used as an important tool of e-commerce, which helps to analyze users' tastes and provide them with lists of products that they would like to prefer. Development of recommender systems is a result of work of scientists from different fields: Data Mining, Machine Learning, Marketing, Information Technology, Statistics, Human Computer Interaction, Adaptive User Interfaces, Decision Support Systems, Consumer Behaviour. Recommender system is a software which uses history of users' preferences to provide them with a list of items (movies, audio records, articles, books and others) that they would like to prefer. Such systems include all steps of process: from extraction information about the users till presenting the recommendations to users. Two basic strategies of recommender systems are content filtering and collaborative filtering.

Collaborative filtering for music recommender system [Elena Shakirova](#) *Young Researchers in Electrical and Electronic Engineering (EIconRus), 2017 IEEE Conference of Russian*

D'après un sujet de Sébastien Mosser sur SOA

Music Corp est un disquaire en ligne (initialement un disquaire « physique » implanté à Albuquerque) ayant vocation à faire évoluer son marché en proposant à ses clients de découvrir de nouveaux artistes, sur la base de leurs goûts musicaux. La société souhaite automatiser cette découverte musicale à travers un site web de vente en ligne et à l'aide d'algorithmes de *machine learning*, qui permettront aux clients n'ayant pas l'opportunité de visiter la boutique en Arizona de profiter de leur expérience.

*Votre défi est de concevoir l'application pour la société Music Corp¹
(en vous limitant à l'énoncé).*

¹ *MusicCorp est l'exemple fil rouge du livre "Building Fine grained systems with micro-services"*

Pour un internaute, les propositions de musiques reposent sur les grandes tendances du marché. Il peut naviguer soit en fonction de genre soit au hasard des tendances. Il peut ainsi découvrir des artistes et acheter des musiques ou produits en lignes. Lors du premier achat, il devient membre. Un membre peut gérer des « playlists » en ligne qui conjugue des musiques issues de différents fournisseurs, grâce à un accord avec Spotify et Deezer.

Pour un nouveau membre, les propositions de musiques reposent sur une adéquation entre les goûts qu'il déclare au travers de son profil et les informations associées aux artistes.

Pour un membre ayant déjà acheté plus de 10 albums, les propositions de musiques sont faites en utilisant des algorithmes de « machine learning » qui font des propositions en fonction du profil du membre, de ses playlists, de ses achats et des avis qu'il a émis par le passé.

Si on constate qu'un membre ne choisit jamais les recommandations qui lui sont faites et/ou émet des avis négatifs sur les morceaux proposés alors on le considère comme un nouveau membre. Le système doit encore apprendre pour lui faire des recommandations plus pertinentes.

Les administrateurs gèrent les stocks, en particulier, grâce à un accord avec Amazon, qui permet des commandes automatiques.

Une playlist est une composition de musiques. Une musique peut être un morceau, un album ou des playlists. Les caractéristiques d'une playlist consistent en l'agrégation des valeurs des caractéristiques des musiques qui la composent. Les caractéristiques d'un album ou d'un morceau sont le ou les compositeurs, le ou les interprètes., le ou les styles.

Scenario * :

Un membre se connecte en donnant son identifiant et son mot de passe.

Le système récupère le membre correspondant et les propositions de musiques qui lui sont associées et crée une nouvelle session.

Tant que l'utilisateur ne se déconnecte pas, le membre peut sélectionner des musiques. Ses sélections sont automatiquement enregistrées par la session (musique sélectionnée et date) et les musiques sont diffusées.

Extension :** Quand des musiques sont beaucoup écoutées, le nombre de ventes de ces musiques augmente. On aimerait donc au-delà d'un certain seuil d'écoute provisionner nos stocks par avance. De plus également en fonction des musiques écoutées, on voudrait gérer les droits d'auteur (SACEM), améliorer notre base de connaissances, etc.

Question 1 : 3 pts Identifiez les acteurs et cas d'utilisation de haut niveau. Représentez acteurs et cas d'utilisation sur un diagramme en vous basant sur l'ensemble de l'énoncé.

Question 2 : 3 pts Dessinez un diagramme de séquence en conception qui explicite le scénario*. Ne gérez pas les cas d'erreurs, mais faites attention à l'attribution des responsabilités.

Question 3 : 8 pts Définissez un diagramme de classes en conception en vous limitant au cœur du métier. Préciser les éléments mis en jeu dans des design patterns et leur rôle. *Attention il doit être cohérent relativement au scénario précédent sur la partie métier sans pour autant représenter les objets de l'interface graphique.*

Question 4 : 3 pts Dessinez le diagramme de classes correspondant au code qui vous est donné en partie 4. Ce code fait référence à des classes qui ne vous sont pas données. Représentez-les en fonction de ce que vous en savez uniquement.

Question 5 : 1 pt Que proposez-vous pour prendre en compte les exigences de la partie *Extension*** ?

Question 6 : 2 pts Quel code correspond à *Album* relativement au diagramme UML de la Figure 1 ?

3^e partie : Connexion à Spotify

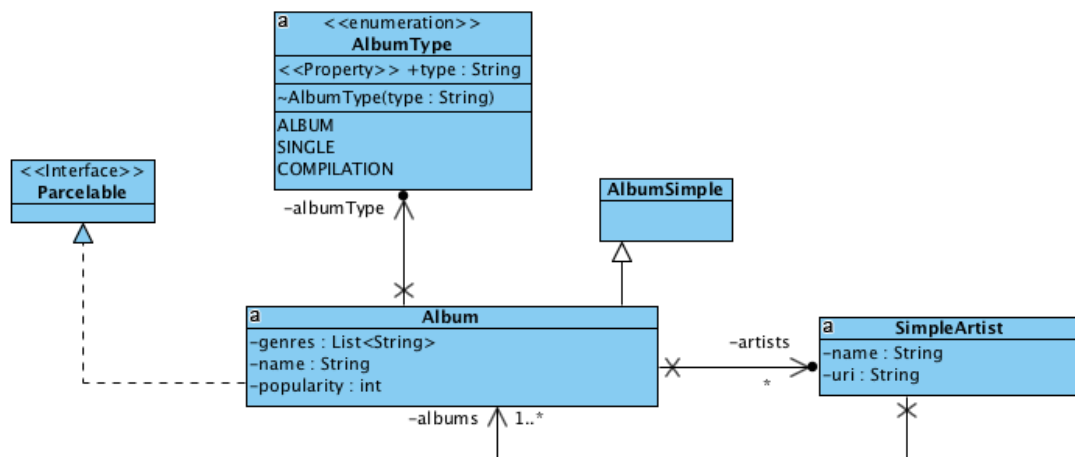


Figure 1 : Modèle UML correspondant à un modèle d'adaptateur vers Spotify

4^e partie : Service de recommandation

Voici des codes permettant de mettre en place un système de recommandation pour les musiques : <https://github.com/guoguibing/librec>

```
public abstract class SocialRecommender extends MatrixFactorizationRecommender
{
    protected SparseMatrix socialMatrix;
    protected float regSocial;
    private ArrayList<User> users;
    protected double predict(int userIdx, int itemIdx, boolean bounded) throws
LibrecException {
    .... }
}
```