IxeMelDb

Version 1 du 9 septembre 2022

Techniques avancées de gestion des données
Administration de BD et XML

3ème Bachelier en informatique de gestion

1. Introduction

Vous trouverez dans ce document le contexte général du système informatique qui devra être réalisé. On y détaille ensuite les différentes fonctionnalités demandées. Ce document peut donc être considéré comme votre ”contrat de travail”. Il permettra de mieux appréhender l'organisation du travail de l'année et sa planification au travers des différentes semaines.

Les modalités d'évaluation se trouvent à la fin de cet énoncé reprenant les explications quant à la construction de la cote finale. Il convient d'en prendre connaissance. Elles vous engagent pour toute la durée de l'année scolaire.

Cet énoncé est très fortement inspiré de l'énoncé du projet à destination des étudiants en informatique industrielle, pour ce qui est du contexte (Merci Mr De Dijcker) ainsi que de l'énoncé de l'année passée, pour les fonctionnalités demandées (Merci Mr Vilvens).

Les données des films proviennent de The Movie Database (TMDb).

2. Contexte

La société IxeMelDb, dont le siège central se situe en dehors de l'union européenne pour éviter le plus longtemps possible les procès pour plagiat, a pour objectif de concurrencer d'autres plate-formes de présentation de films, comme IMDb et TMDb.

Leur agent spécial a réussi à voler une partie des données du site de TMDb (grâce à l'API fournie par cette dernière…), mais pour que ce soit utilisable, il leur faut désormais structurer ces données (ils optent judicieusement pour le format XML) effectuer les transformations nécessaires pour pouvoir les présenter sur leur site web (grâce à XSLT).

2.1 Les films

La signalétique d'un film comporte toutes les informations relatives à ce film. Par exemple, son titre, sa durée, ses genres, le public concerné, les acteurs principaux, une affiche, …. Chaque film est une ligne (un enregistrement) stocké dans un fichier texte appelé movies.txt. On retrouve 15 champs par film expliqués dans le tableau page suivante

.
Le format d'un enregistrement de la table externe est le suivant :

```
_id(U+2023)title(U+2023)original_title(U+2023)release_date(U+2023)status(U+2023)
vote_average(U+2023)vote_count(U+2023)runtime(U+2023)certification(U+2023)pos
ter_path(U+2023)budget(U+2023)tagline(U+2023)genres:id(U+2024)name(U+2023)di
rectors:id(U+2024)name(U+2023)actors:id(U+2024)name(U+2024)character
```

Ou encore:

```
_id▸title▸original_title▸release_date▸status▸vote_average▸vote_count▸runtime▸certificati
on▸poster_path▸budget▸tagline▸genres:id.name▸directors:id.name▸actors:id.name.chara
cter.
```

Attention, les caractères séparateurs proviennent de Unicode et sont représentés par leur "code point" Unicode en hexadécimal placé entre parenthèses comme (U+2023).

Certains champs peuvent être absents, ce qui est représenté par deux séparateurs consécutifs (U+2023)(U+2023).

Les séparateurs utilisés pour chaque enregistrement (ou ligne du fichier) texte sont les suivants :

- Le caractère ▸ est utilisé pour séparer les champs qui sont au niveau le plus haut de la hiérarchie (top-level). Il s'agit du caractère Unicode appelé triangular bullet dont le code est U+2023.
- Le nom indiqué devant les deux-points est le nom du champ contenant un tableau comme genres, directors et actors. Notez que le nom du champ et les deux-points ne font pas partie des données. Donc par exemple "genres:" n'est pas présent dans les données textuelles.
- Le caractère ‖ est utilisé comme séparateur des éléments du tableau. Il s'agit du caractère Unicode appelé double vertical line dont le code est U+2016.
- Le caractère . est utilisé pour séparer les champs d'un élément dans un tableau. Il s'agit du caractère Unicode appelé one dot leader dont le code est U+2024.

Pour le film Inception (id 27205), on a l'enregistrement ci-dessous dans le fichier texte movies.txt :

```
27205▸Inception▸Inception▸2010-07-16▸Released▸7.5▸5578▸148▸PG-13▸
/tAXARVreJnWfoANIHASmgYk4SB0.jpg▸160000000▸Your mind is the scene of the
crime.▸28.Action‖12.Adventure‖9648.Mystery‖878.Science Fiction‖53.Thriller
▸525.Christopher Nolan▸6193.Leonardo DiCaprio.Cobb‖24045.Joseph Gordon-
Levitt.Arthur ‖27578.Ellen Page.Ariadne‖2524.Tom Hardy.Eames‖3899.Ken
Watanabe.Saito‖2037.Cillian Murphy.Robert Fischer‖8293.Marion
Cotillard.Mal‖3895.Michael Caine.Miles‖95697.Dileep Rao.Yusuf‖13022.Tom
Berenger.Browning‖4935.Pete Postlethwaite.Maurice Fischer‖526.Lukas
Haas.Nash‖66441.Talulah Riley.Blonde
```

Nom	Explications
_id	L'identifiant TMDb du film.
title	Le titre du film. Pour rappel, les données sont encodées en UTF-8.
original_title	Le titre original du film. Si la langue originale du film n'est pas l'anglais, le titre dans sa langue originale se trouvera dans ce champ. Si la langue originale est l'anglais alors on retrouve le titre (title) tel quel dans ce champ. Etant donné que le champ original_language n'est pas présent, toutes les recherches sur le titre du film devront se faire sur les deux champs. Pour rappel, les données sont encodées en UTF-8.
release_date	La date de sortie du film (dans le fichier texte elles sont au format YYYY-MM-DD).
status	L'état de production du film.
vote_average	La cote qu'a obtenu le film suite aux votes des membres TMDb.
vote_count	Le nombre de votes TMDb qui a servi à établir le vote_average.
runtime	La durée du film en minutes.
certification	Le rating qui indique le public cible du film. Les abréviations sont tirées de la notation proposée par l'association MPAA (http ://www.mpaa.org/ film-ratings/) et sont les seules autorisées.
poster_path	Un chemin permettant de construire l'URL vers une image de type poster pour le film. Les données n'étant pas à jour dans cette version, nous ne l'utiliserons pas pour la construction de la page web
budget	Le coût du film en dollars.
tagline	Le texte qui accompagne le film et qui est souvent écrit sur la pochette du film.
genres	Les genres dont fait partie le film. Un tableau d'objets dont chacun possède les informations suivantes : id qui est l'identifiant du genre sous forme d'un nombre entier et name qui est le nom du genre comme "Action", "Comedy" ou "Thriller".
directors	Un tableau de réalisateurs. Chaque réalisateur contient les informations suivantes : id et name.
actors	Un tableau d'acteurs. Chaque acteur contient les informations suivantes : id, name et character qui contient le rôle joué par l'acteur dans le film.

3. Réalisations attendues

Vous utiliserez le fichier movie.txt comme base pour vos données, et effectuerez les opérations suivantes pour obtenir le site web souhaité.

3.1 Conversion vers un document XML

En première partie, les données du fichier movie.txt seront converties en un fichier XML. Vous êtes libres dans le nom des tags, dans la structure utilisée, dans le choix des types d'éléments (avec données ou simple tag), d'attributs etc, **mais il faut au minimum utiliser un attribut dans au moins un élément.**

Le langage de programmation pour cette opération est laissé à votre appréciation parmi les possibilités suivantes : (PL/)SQL, Java, C, C++.

Note : Attention aux API blackbox (que vous aurez le temps d'utiliser plus tard) et qui génèrent parfois des erreurs bien difficiles à débugguer. Préférez une approche « gestion de chaine de caractères », qui est certes plus bas niveau mais sur laquelle vous avez le contrôle total et qui, finalement, ira plus vite à développer.

3.2 Structure du document XML

La structure de votre document devra être déterminée. C'est dans cette partie que vous validerez formellement les choix que vous avez effectués dans la section précédente.

Au minimum : Vous réaliserez un document DTD précisant la structure

Pour les pros : Vous réaliserez un document XSD précisant la structure

Note : Ces structures doivent impérativement être écrites à la main, et pas générées automatiquement à partir d'un outil.

3.3 Validation du fichier XML

Une fois votre document XML et votre document structure en main, il vous faudra vérifier si celui-ci est bien valide. Pour ce faire, il faudra réaliser un *parser*. Celui-ci sera écrit en langage java.

Au minimum : Vous utiliserez un parser SAX

Pour les pros : Vous utiliserez un parser DOM

Pour les experts : Vous évaluerez les performances de vos programmes en comparant les temps d'exécution ainsi que la consommation mémoire (y compris, le cas échéant, en comparant aussi une validation avec DTD et une validation avec XSD)

Dans tous les cas : En plus de vérifier la validité du fichier, vous effectuerez via le parser les calculs suivants :

- Compter le nombre de films catégorisés PG-13
- Classer les films par leur note moyenne et afficher, en fin de programme, la liste des 10 meilleurs films, triés par cette note. L'optimisation de l'espace mémoire sera pris en compte dans l'évaluation.

3.4 Création du site web avec XSLT

Une fois votre fichier XML en votre possession, vous effectuerez une transformation à l'aide d'un fichier style sheet (XSLT, donc) afin de générer une page HTML affichant proprement les informations.

Puisque nous travaillons avec des gens pressés qui veulent une solution rapide, vous pouvez utiliser un petit serveur apache (par exemple avec XAMPP) où vous stockerez vos fichiers.

Au minimum : Ce sera fonctionnel. Il sera possible de voir toutes les informations (dans une table, par exemple).

Pour les pros : On imaginera un document un peu structuré, joli, ergonomique, sans pour autant faire intervenir des technologies supplémentaires.

Pour les experts : Vous rajouterez des fonctionnalités à l'aide du CSS, du javascript, etc, etc. Vous pouvez laisser libre cours à votre imagination et vos envies. Etonnez-moi ☺

4. Evaluation

Ce projet s'inscrit dans le cadre du cours d'administration de BD et XML. L'UE est répartie de la manière suivante :

- AA Bases de données avancées et XML: 60h Samuel Hiard
- AA Gestion de projet : 45h Souad Serrhini
- AA Développement de projet : 15h
- AA Introduction aux mainframes : 15h

Tout ce qui suit concerne uniquement ma partie (les 60h dont j'ai la charge).

L'évaluation de ma partie sera séparée en deux catégories :

1) Théorie. Evaluée via un QCM. Compte pour 50% des points
2) Laboratoire. Evaluée via deux projets. Compte pour 50% des points

Afin d'évaluer la partie laboratoire, deux projets seront à réaliser.

1) Ce projet XML, qui compte pour 50% de la note de labo
2) Un autre projet, plutôt orienté SGBD, dont l'énoncé sera disponible ultérieurement, et qui comptera pour 50% de la note de labo.

Comme vous savez calculer, vous aurez compris que ce projet compte donc pour 25% de la note concernant ma partie dans l'AA. Les points sont répartis de la manière suivante :

- Conversion vers le document XML 5 points
- Réalisation du DTD et/ou du XSD 4 points
- Parsing et calcul statistique 8 points
- Site web avec XSLT 8 points

Ce projet est à réaliser par binôme. Il sera à présenter le **9 novembre 2022** pendant la séance de laboratoire. Vous pouvez toutefois présenter plus tôt si vous êtes prêts. Dans tous les cas, pour pouvoir présenter le projet, il faut l'avoir envoyé, par mail à samuel.hiard@hepl.be **au plus tard le dimanche précédent le jour du labo** (avec 23h59 comme heure limite).

Il y a beaucoup à faire, donc ne trainez pas.

Pour certaines réalisations, il sera fait mention de « au minimum », « pour les pros » et « pour les experts ». La signification est la suivante :

Au minimum : La base. Ce que j'attends vraiment au minimum du minimum. Arrêtez-vous là si vous visez la réussite simple (ce que vous aurez, mais seulement si c'est bien réalisé).

Pour les pros : Une réalisation un peu plus pro ou demandant un peu plus d'implications et de travail. Ne sera évalué que si l'item « au minimum » est réalisé. Pour ceux qui visent la distinction.

Pour les experts : Pour ceux que le sujet passionne et qui veulent vraiment se spécialiser dans cette discipline. Ou pour ceux qui visent l'excellence et la plus grande distinction.

Bon courage et, surtout, amusez-vous bien ☺