



Universidade do Porto

FEUP Faculdade de Engenharia

Gestão de Filmes

Bases de Dados

Turma 3, Grupo 307

José Rebelo	up201303964
Miguel Pereira	up201305998
Ricardo Cerqueira	up201304000

Índice

Introdução	2
Contexto	3
Diagrama de classes UML	4
Modelo Relacional	5
Instruções LMD-SQL.....	6
Gatilhos	6
Interrogações	6
View	6
Conclusão.....	7

Introdução

Com o desenvolvimento e proliferação de plataformas digitais e o resultante aumento da quantidade e variedade de conteúdo cinematográfico, torna-se essencial a construção e manutenção de um sistema eficaz e eficiente de armazenamento de toda a informação referente a este.

Assim pretende-se modelar conceptual e relacionalmente o contexto em causa, assim como desenvolver uma base de dados que suporte uma aplicação web permitindo aos utilizadores a pesquisa e organização de listas de filmes.

Contexto

Para cada filme é guardado o título, a duração, a sinopse, a classificação média dada pelos utilizadores derivada das várias críticas feitas pelos utilizadores ao filme, os vários géneros a que está associado (drama, terror, etc.), o orçamento do filme, assim como os seus lucros.

O filme tem um lançamento e uma classificação etária por cada país onde é exibido. A classificação etária tem uma idade mínima, uma descrição do motivo pelo qual é dada a classificação, assim como o seu nome. Sobre os lançamentos, é mantido a data de estreia do filme naquele país assim como os ganhos nele adquiridos.

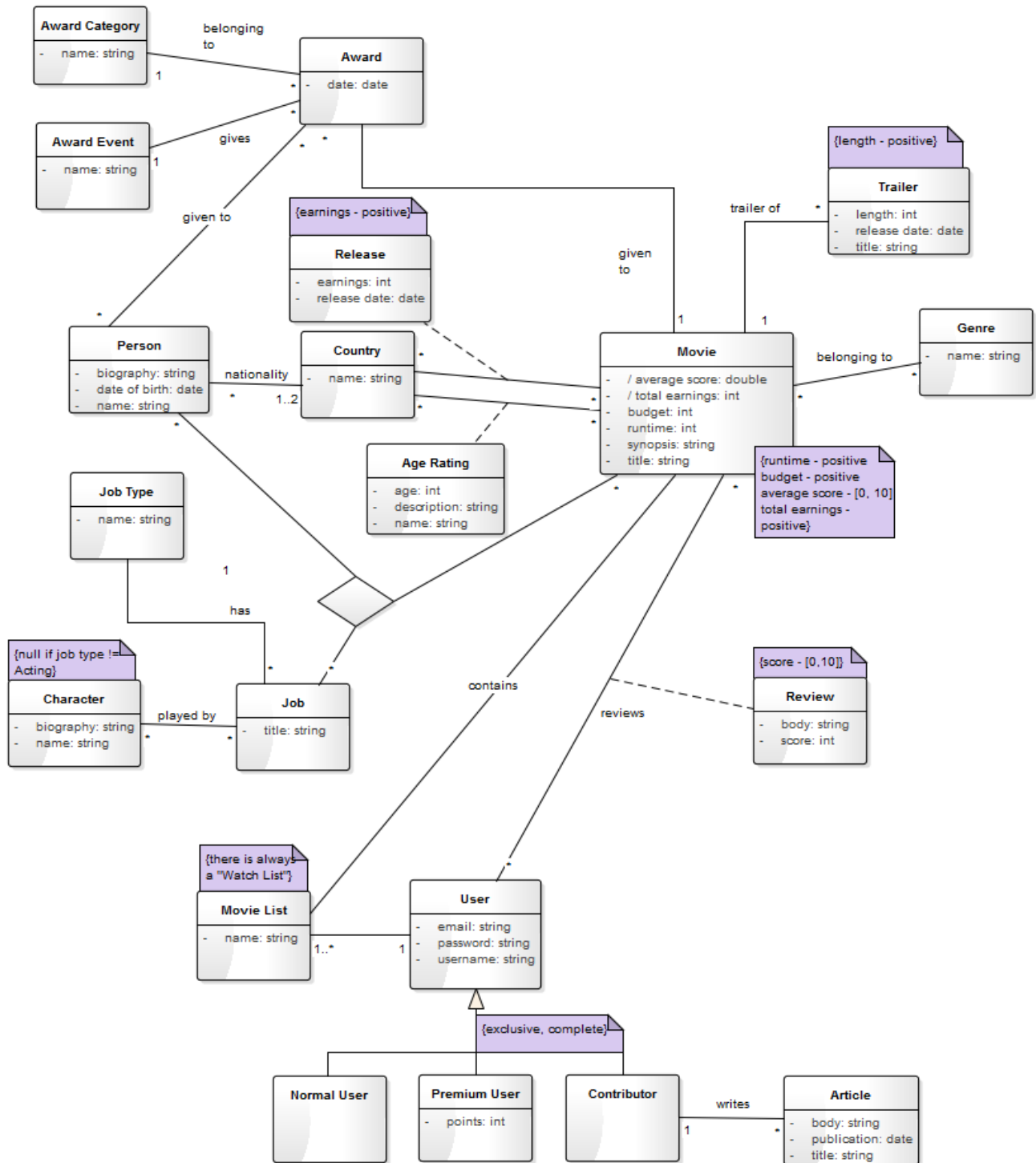
Um filme pode, ainda, ter vários trailers associados que terão um título, uma duração e uma data de lançamento.

Pode, também receber diferentes prémios, que estão associados a um evento (Óscares, Globos de Ouro), a uma categoria (melhor filme, melhor realizador) e a uma data de atribuição. Os prémios podem ser atribuídos a uma pessoa pelo seu papel num determinado filme, ou ao próprio filme.

Por último, guardam-se as pessoas que trabalharam nos vários filmes. Uma pessoa tem um nome, uma data de nascimento, uma nacionalidade e uma biografia e está associada aos filmes pelo seu trabalho. Uma pessoa pode ter mais do que um trabalho num determinado filme (ser realizador e escritor, por exemplo). Os trabalhos dos atores estão, ainda, associados às personagens interpretadas, que têm uma biografia e um nome.

Para os utilizadores guardam-se o *username*, o *e-mail* e a palavra-passe. Os utilizadores podem ser normais, *premium* ou contribuidores. Estes últimos podem escrever artigos, que têm um título, um corpo e uma data de publicação e os utilizadores *premium* acumulam pontos. Os utilizadores podem criar listas de filmes (filmes favoritos, por exemplo), que terá um nome e vários filmes, que não se poderão encontrar repetidos. Os utilizadores terão sempre pelo menos uma lista, a *watchlist*, que poderá estar vazia.

Diagrama de classes UML



Modelo Relacional

AgeRating(movieID→Movie, countryID→Country, name, age, description)

Article(articleID, userID→User, title, publication, body)

Award(awardID, awardEventID→AwardEvent, awardCategoryID→AwardCategory, awardDate, movieID→Movie)

AwardCategory(awardCategoryID, name)

AwardEvent(awardEventID, name)

AwardPerson(awardID→Award, personID→Person)

Character(characterID, biography, name)

CharacterJob(characterID→Character, jobID→Job)

Country(countryID, name)

Genre(genreID, name)

Job(jobID, jobTypeID→JobType, title)

JobType(jobTypeID, name)

Movie(movieID, title, synopsis, runtime, budget, averageScore, totalEarnings)

MovieGenre(movieID→Movie, genreID→Genre)

MovieList(movieListID, userID→User, name)

MovieListMovie(movieListID→MovieList, movieID→Movie)

Person(personID, name, date_of_birth, biography)

PersonCountry(countryID→Country, personID→Person)

PersonMovieJob(personID→Person, movieID→Movie, jobID→Job)

Release(movieID→Movie, countryID→Country, releaseDate, earnings)

Review(movieID→Movie, userID→User, score, body)

Trailer(trailerID, title, releaseDate, length, movieID→Movie)

User(userID, username, email, password, points, userType)

Instruções LMD-SQL

Gatilhos

- Ao inserir uma *review*, atualizar a classificação média do filme;
- Ao inserir uma *release*, atualizar os ganhos totais do filme;
- Ao classificar um filme com a pontuação máxima, criar, caso não exista, uma nova lista de filmes onde irão ser colocados todos os filmes que o utilizador classificar com essa pontuação e inserir o filme na lista;
- Ao classificar um filme com a pontuação máxima, inserir o filme numa lista de filmes que o utilizador classificou com a mesma pontuação;
- Evitar que sejam publicados artigos por utilizadores que não sejam contribuidores;
- Para cada utilizador, cria-se, automaticamente, uma lista chamada “Watch List”;
- Ao inserir uma *review*, caso o utilizador seja *premium*, adicionar pontos consoante o número de *reviews* para o filme em questão – quanto menos *reviews* o filme tiver, mais pontos o utilizador recebe.

Interrogações

- Para cada pessoa, calcular a média das *reviews* e a soma dos ganhos totais dos filmes em que participa;
- Lista de personagens e o respetivo filme, ordenadas pelo número de prémios ganhos em categorias distintas;
- Lista de filme com uma classificação etária que é superior à idade de um ou mais atores do filme;
- Para cada filme, somar o número de pessoas de cada nacionalidade;
- Determinar o número de prémios por cada género de filme;
- Para cada utilizador, para cada pessoa, determinar a média das classificações dos filmes a que pertencem;
- Determinar as próximas *releases* num país
- Determinar o quinto país com melhor média de lucros;
- Determinar para cada pessoa qual o género de filme em que participou que tem a melhor média de classificações;
- Determinar o número de artigos que se referem a cada filme;

View

- *View* para os utilizadores públicos.

Conclusão

Um sistema de base de dados requer um estudo aprofundado do contexto em questão, de modo a adequadamente modelar as relações em causa. O modelo acima apresentado constitui meramente uma simplificação do problema.