# Serviço de Streaming

Grupo:

Turma 7, Grupo 1

Estudantes:

Bruno Rosendo, up201906334

João Mesquita, up201906682

Rui Alves, up201905853

Disciplina:

Bases de Dados

## Contexto da Base de Dados

Pretende-se armazenar a informação relativa a um Serviço de Streaming (semelhante a Netflix).

Para autenticação no mesmo, existirão dois tipos de **conta**, uma para **utilizadores** e uma para **administradores**. Cada conta é representada por um e-mail e password. O email é característico de cada utilizador e deve ter um formato adequado (de email). Por outro lado, a password tem de ter tamanho maior que 5 e pelo menos uma letra maiúscula, uma minúscula e um número. As contas dos utilizadores são constituídas por no máximo 5 **perfis**, cada um com um nome, avatar e uma linguagem de forma a distinguir a pessoa que o utiliza. Por outro lado, as contas dos administradores são representadas um nome.

Guarda-se ainda informação acerca do tipo de cada conta do utilizador (**premium** ou **standard**) e do pagamento da mesma, isto é, se tem o pagamento em dia e o seu preço. Este depende do número de perfis da conta e do tipo de utilizador.

Os administradores são responsáveis pela alteração de **conteúdo** ao serviço. Este pode ser um **filme**, uma **série** ou **conteúdo especial** (como por exemplo, cenas dos bastidores). As contas premium têm, em comparação com as contas standard, acesso ao conteúdo especial.

Todo o conteúdo tem um título, data de lançamento, estúdio, descrição, idade mínima recomendada e classificação. A classificação de um conteúdo varia entre 0 e 5 estrelas. De forma a representar o tipo de conteúdo, este está associado a no mínimo um e no máximo três **géneros**, dentro de uma lista predefinida (ação, comédia, romance, animação, anime, documentário, drama, ficção, terror).

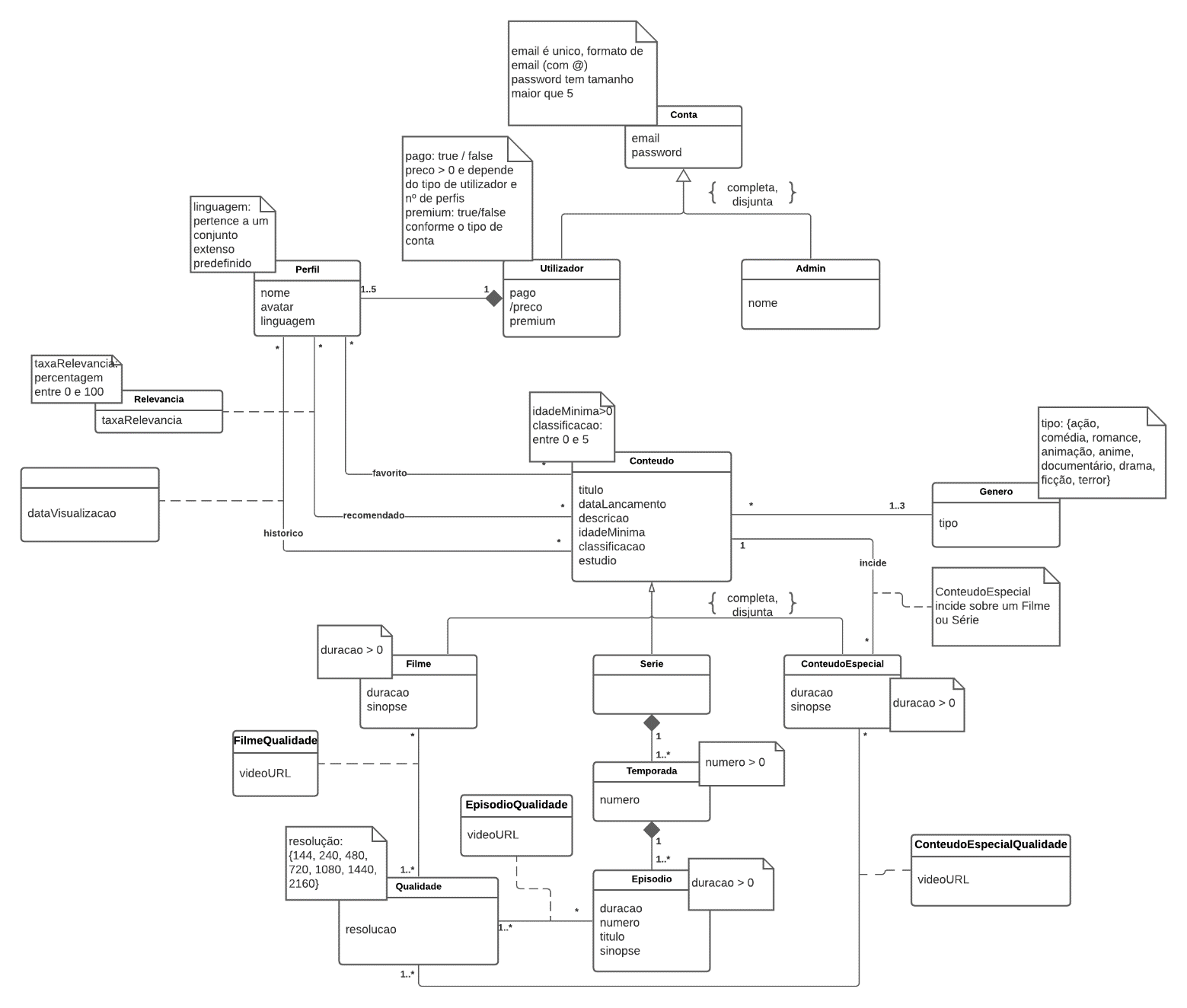
As séries são constituídas por **temporadas**, caracterizadas por um número, sendo que estas estão dividas em **episódios** com uma dada duração, título e sinopse. Por outro lado, os filmes e o conteúdo especial apenas têm informação da duração e sinopse. O conteúdo especial incide sobre um filme ou uma série.

Cada filme, episódio ou conteúdo especial tem vários tipos de **qualidade de visualização**, selecionados dentro de uma lista predefinida (144, 240, 480, 720, 1080, 1440, 2160).

O sistema informático deverá atribuir automaticamente **conteúdo recomendado e** a sua taxa de relevância a cada perfil e guardar o seu respetivo **histórico** (quando viu um dado filme).

Por outro lado, o utilizador pode ainda adicionar conteúdo a uma **lista de favoritos**, que é diferente para cada perfil.

## Diagrama UML (revisto)



## Esquema Relacional e Dependências Funcionais

-Conta (email, password);

* email->password;

-Utilizador (email->Conta, pago, premium);

* email -> pago, premium;
* preço é um atributo derivado, pelo que não consta do esquema relacional.

-Admin (email->Conta, nome);

* email ->nome;

-Perfil (id, nome, avatar, linguagem, emailUtilizador->Utilizador);

* id -> nome, avatar, linguagem, emailUtilizador;

-Conteudo (id, titulo, dataLancamento, descricao, idadeMinima, classificacao, estudio);

* id -> titulo, dataLancamento, descricao, idadeMinima, classificacao, estudio;

-Genero (tipo);

-GeneroConteudo (idConteudo->Conteudo, tipo->Genero);

-Filme (idConteudo->Conteudo, duracao, sinopse);

* idConteudo -> duracao, sinopse;

-Serie (idConteudo->Conteudo);

-ConteudoEspecial(idConteudo->Conteudo, duracao, sinopse, idConteudoIncidido -> Conteudo);

* idConteudo -> duracao, sinopse, idConteudoIncidido;

-Temporada (idTemporada, numero, idSerie->Serie);

* idTemporada -> numero, idSerie;
* numero, idSerie -> idTemporada;

-Episodio (id, numero, duracao, titulo, sinopse, idTemporada -> Temporada);

* id -> duracao, titulo, sinopse, idTemporada, numero;

-Qualidade (resolucao);

-ConteudoEspecialQualidade(idConteudo -> ConteudoEspecial, resolucao -> Qualidade, urlVideo);

* idConteudo, resolucao -> urlVideo

-FilmeQualidade (idFilme->Filme, resolucao->Qualidade, urlVideo);

* idConteudo, resolucao -> urlVideo

-EpisodioQualidade (idEpisodio->Episodio, resolucao->Qualidade, urlVideo);

* idConteudo, resolucao -> urlVideo

-Favorito (idPerfil->Perfil, idConteudo->Conteudo);

-Historico (idPerfil->Perfil, idConteudo->Conteudo, dataVisualizacao);

* idPerfil, idConteudo -> dataVisualizacao;

-Recomendado (idPerfil->Perfil, idConteudo->Conteudo, taxaRelevancia);

* idPerfil, idConteudo -> taxaRelevancia;

## Dependências Funcionais e Formas Normais

Para todas as relações descritas na página anterior, o lado esquerdo das suas dependências funcionais é uma (super) chave dessa relação, pelo que se comprova que estão na Boyce-Codd Normal Form, visto que o fecho dos atributos desse lado contém todos os atributos da relação, e, consequentemente, também estão na Terceira Forma Normal, que é um superconjunto da BCNF. De seguida estão explicitados os fechos de todas as relações.

-Conta:

* {email}+ = {email, password}

-Utilizador:

* {email}+ = {email, pago, premium}

-Admin:

* {email}+ = {email}

-Perfil:

* {id}+ = {id, nome, avatar, linguagem, emailUtilizador}

-Conteudo:

* {id}+ = {id, titulo, dataLancamento, descricao, idadeMinima, classificacao, estudio}

-Genero:

* {tipo}+ = {tipo}

-GeneroConteudo:

* {idConteudo}+ = {idConteudo, tipo}

-Filme:

* {idConteudo}+ = {idConteudo, duracao, sinopse}

-Serie:

* {idConteudo}+ = {idConteudo}

-ConteudoEspecial:

* {idConteudo}+ = {idConteudo, duracao, sinopse, idConteudoIncidido}

-Temporada:

* {idTemporada}+ = {idTemporada, numero, idSerie}
* {numero, idSerie}+ = {idTemporada, numero, idSerie}

-Episodio:

* {id}+ = {id, numero, duracao, titulo, sinopse, idTemporada}

-Qualidade:

* {resolucao}+ = {resolucao}

-ConteudoEspecialQualidade:

* {idConteudo, resolucao}+ = {idConteudo, resolucao, urlVideo}

-FilmeQualidade:

* {idFilme, resolucao}+ = {idFilme, resolucao, urlVideo}

-EpisodioQualidade:

* {idEpisodio, resolucao}+ = {idEpisodio, resolucao, urlVideo}

-Favorito:

* {idPerfil, idConteudo}+ = {idPerfil, idConteudo}

-Historico:

* {idPerfil, idConteudo}+ = {idPerfil, idConteudo, dataVisualizacao}

-Recomendado:

* {idPerfil, idConteudo}+ = {idPerfil, idConteudo, taxaRelevancia}

## Restrições:

Todas as tabelas usam ON UPDATE CASCADE e ON DELETE CASCADE, de forma a ter todas as tabelas sempre devidamente atualizadas.

-Conta:

* email: chave primária (PRIMARY KEY) e deve estar em formato de e-mail (CHECK e LIKE '%\_@\_\_%.\_\_%')
* password: tamanho é maior ou igual a 5 (CHECK e length())

-Utilizador:

* email: chave primária e estrangeira (PRIMARY KEY E REFERENCES)
* pago: tem que ser um valor booleano (usando INTEGER e limitando a 0 ou 1, com CHECK)
* preco: é calculado, com uma VIEW, a partir do número de perfis e do tipo de conta.
* premium: tem que ser um valor booleano (usando INTEGER e limitando a 0 ou 1, com CHECK)

-Admin:

* email: chave primária e estrangeira (PRIMARY KEY E REFERENCES)
* nome: não pode ser nulo (NOT NULL)

-Perfil:

* id: chave primária (PRIMARY KEY)
* nome: não pode ser nulo (NOT NULL)
* linguagem: não pode ser nulo (NOT NULL) e tem que pertencer a um conjunto predefinido de linguagens (foi usado um conjunto reduzido, para simplificação, com linguagem IN ([LISTA]))
* emailUtilizador: chave estrangeira (REFERENCES)

-Conteudo:

* id: chave primária (PRIMARY KEY)
* titulo: não pode ser nulo (NOT NULL)
* dataLancamento: não pode ser nulo (NOT NULL) e tem que ser anterior à data de inserção na tabela
* idadeMinima: maior que 0 (CHECK)
* classificação: entre 0 e 5 (CHECK)

-Genero:

* tipo: chave primária (PRIMARY KEY) e tem que que pertencer a ["acao", "comedia", "romance", "animacao", "anime", "documentario", "drama", "ficcao", "terror"] (tipo IN ([lista]))

- GeneroConteudo:

* idConteudo: chave estrangeira (REFERENCES)
* tipo: chave estrangeira (REFERENCES)
* Chave primária composta por idConteudo e tipo

-Filme:

* idConteudo: chave estrangeira e primária (PRIMARY KEY E REFERENCES)
* duracao: maior que 0 (CHECK)

-Serie:

* idConteudo: chave primária e estrangeira (PRIMARY KEY E REFERENCES)

-ConteudoEspecial:

* idConteudo: chave primária e estrangeira (PRIMARY KEY E REFERENCES)
* duracao: maior que 0 (CHECK)
* idConteudoIncidido: chave estrangeira (REFERENCES)

-Temporada:

* idTemporada: chave primária (PRIMARY KEY)
* numero: maior que 0 (CHECK)
* idSerie: chave estrangeira (REFERENCES)
* O conjunto (numero, idSerie) é único (UNIQUE)

-Episodio:

* id: chave primária (PRIMARY KEY)
* numero: não pode ser nulo (NOT NULL) e tem que ser maio r que 0 (CHECK)
* duracao: maior que 0 (CHECK)
* idTemporada: chave estrangeira (REFERENCES)

-Qualidade:

* resolucao: chave primária (PRIMARY KEY) e tem que que pertencer a [144, 240, 480, 720, 1080, 1440, 2160] (resolucao in ([lista]))

- ConteudoEspecialQualidade:

* idConteudo: chave estrangeira (REFERENCES)
* resolucao: chave estrangeira (REFERENCES)
* urlVideo: não nulo (NOT NULL)
* Chave primária composta por idConteudo e resolucao

- FilmeQualidade:

* idFilme: chave estrangeira (REFERENCES)
* resolucao: chave estrangeira (REFERENCES)
* urlVideo: não nulo (NOT NULL)
* Chave primária composta por idFilme e resolucao

- EpisodioQualidade:

* idEpisodio: chave estrangeira (REFERENCES)
* resolucao: chave estrangeira (REFERENCES)
* urlVideo: não nulo (NOT NULL)
* Chave primária composta por idEpisodio e resolucao

-Favorito:

* idPerfil: chave estrangeira (REFERENCES)
* idConteudo: chave estrangeira (REFERENCES)
* Chave primária composta por idPerfil e idConteudo

-Historico:

* idPerfil: chave estrangeira (REFERENCES)
* idConteudo: chave estrangeira (REFERENCES)
* dataVisualizacao: não pode ser nulo (NOT NULL) e tem que ser anterior à data de inserção na tabela
* Chave primária composta por idPerfil e idConteudo

-Recomendado:

* idPerfil: chave estrangeira (REFERENCES)
* idConteudo: chave estrangeira (REFERENCES)
* taxaRelevancia: entre 0 e 100, ou seja, percentagem (CHECK)
* Chave primária composta por idPerfil e idConteudo

Interrogações

1. Listagem do título de todos os conteúdos que sejam de um dado género (neste caso, de ação).
2. Listar os perfis que já viram conteúdo de todos os géneros existentes.
3. Busca da qualidade máxima do vídeo de um episódio, de uma dada temporada, de uma dada série específica.
4. Número de episódios por série, com o respetivo título.
5. Listagem do título de todos os filmes que um dado utilizador ainda não viu (neste caso, o utilizador com ID 1) e que tem classificação maior que 4.8.
6. Listagem dos filmes, conteúdos especiais e séries cuja resolução é a resolução máxima de todos os conteúdos (no caso das séries, têm episódios com essa resolução).
7. Contagem do número de géneros de filmes que o perfil de um utilizador (neste caso, perfil com ID 1) viu, mas não adicionou aos favoritos. Para além disso, inclui a percentagem de cada género dentro dos selecionados.
8. Listagem dos conteúdos especiais que obtiveram uma classificação igual ou superior ao conteúdo sobre o qual incidem.
9. Esta interrogação destina-se a retornar os Géneros de Conteúdo e a respetiva frequência que cada Utilizador (com vários perfis) mais visualizou.
   * No caso de haver mais que um conteúdo com frequência máxima, retorna-os a todos.
   * No caso de um utilizador não ter filmes vistos (Histórico vazio), surge com o género mais visto a NULL e frequência 0.

10) Listagem do conteúdo com mais sucesso de cada estúdio e de cada ano, ordenado por ordem crescente de ano por estúdio.

Gatilhos

1. Este gatilho não permite a inserção de conteúdo que um perfil não viu, na sua lista de favoritos. Além disso, a base de dados também não permite a inserção nos recomendados, de conteúdo que um perfil já viu. Nos casos em que isto acontece, é invocado um erro (*raise* com a opção *rollback*).
2. Caso todos os episódios de uma temporada sejam apagados, a respetiva temporada também deve ser apagada. Seguidamente, caso todas as temporadas de uma dada série forem apagadas, a série também o é.
3. Para ajudar o utilizador da base de dados, quando este tenta inserir um tuplo com chave primária (idConteudo e resolucao) repetida numa das tabelas relacionadas com qualidade (ConteudoEspecialQualidade, FilmeQualidade ou EpisodioQualidade), em vez de tentar inserir esse tuplo e lançar o erro de *primary key constraint,* o gatilho vai atualizar o tuplo antigo com a nova informação (ou seja, o videoURL). Isto é útil para que o utilizador possa inserir vídeos atualizados (e.g. reedições) do seu conteúdo sem necessitar de pensar se já inseriu um vídeo com essa resolução (seguindo o exemplo, a reedição pode ter mais opções de resolução do que a edição antiga, pelo que facilita poder apenas adicionar todos os vídeos de novo).