



Universidade de São Paulo
Instituto de Ciências Matemáticas e de Computação

Projeto - Produtora - Parte 3
Sistema de Gerenciamento de Dublagem de Animes

SCC-240 - Banco de Dados
Profª. Elaine Parros Machado de Sousa

Ewerton Patrick Silva do Amaral nº 10346975
Lucas Nobuyuki Takahashi nº 10295670
Marcelo Kiochi Hatanaka nº 10295645
Rodrigo Mendes Andrade nº 10262721

São Carlos, 14 de Junho de 2019

DESCRIÇÃO DO PROBLEMA

Anime (ou animê) se refere à animação que é produzida por estúdios japoneses e é um termo abreviado de “animação”. Seu período de exibição é predeterminado e sua produção é constituída essencialmente por 3 partes: animação, trilha sonora e dublagem.

Uma produtora de animes de renome precisa de um sistema capaz de gerenciar a produção de **dublagens originais** de suas animações, ou seja, um sistema que será utilizado para relacionar dubladores a seus respectivos personagens e faixas de áudio às cenas, além de organizar as reuniões dos elencos em horários e locais determinados para as gravações.

Primeiramente, um anime que necessita de dublagem é cadastrado no sistema, sendo armazenadas informações como nome, ano de produção, autor, número de temporadas e estado de gravação. Em seguida, é cadastrada a temporada a ser produzida juntamente com o diretor responsável, sendo este definido por CPF, nome, endereço, email e telefones para contato. As temporadas são definidas por: número da temporada, período de exibição, estado de gravação e número de episódios. Uma mesma temporada deve ser dirigida apenas por um diretor, mas um mesmo diretor pode dirigir várias temporadas (inclusive de diferentes animes).

Em seguida, são identificados os episódios que compõem a temporada e as cenas contidas nos episódios. Os episódios são definidos por: título (opcional), número do episódio, duração (em minutos), estado de gravação e número de cenas. As cenas são definidas por: número da cena, duração, script, faixa de áudio (que será gravada no processo de dublagem) e estado de gravação.

A seleção do elenco ocorre da seguinte forma: os personagens do anime que precisam ser dublados são identificados por nome, timbre de voz e uma breve descrição. Assim que os dubladores são definidos, suas respectivas Agências de Dublagem são contactadas para firmarem-se os contratos. As agências são definidas por CNPJ, nome, telefones para contato, email e endereço. Por sua vez, os dubladores são identificados por nome, CPF, data de nascimento, timbre de voz e sexo. No decorrer do processo de dublagem, o dublador é avaliado para cada episódio em que participa. Ao final da gravação do anime, o trabalho total do dublador é avaliado em função da performance obtida em cada episódio, proporcionando assim uma referência para selecionar o melhor dublador para determinado personagem.

O processo de dublagem do anime é dividido em sessões de gravação. Em cada sessão, são estabelecidas as cenas a serem gravadas, para assim verificar-se quais dubladores precisam estar presentes. Em seguida, a sessão é agendada com o diretor da temporada do anime e os dubladores necessários em data, horário e local específicos, de modo que 2 sessões que ocorram ao mesmo tempo não sejam na mesma sala, que não gravem a mesma cena ou que não envolvam entidades em comum (Ex: um mesmo diretor precisando comparecer a sessões simultâneas de 2 animes que o mesmo dirige). Um técnico de áudio que trabalha para a própria

produtora também deve ser chamado para auxiliar no manuseio dos equipamentos, sendo este cadastrado com CPF, nome, data de nascimento, endereço, email e telefones para contato. Caso o diretor ou algum dos dubladores não possa comparecer, a sessão poderá ser reagendada. Caso o técnico de áudio designado não possa comparecer, outro pode ser chamado. A sessão também poderá ser reagendada caso não haja técnicos de áudio disponíveis para aquela data.

O local de cada sessão sempre será uma das salas de gravação pertencentes à própria produtora, sendo cada sala identificada por número da sala e quantidade de kits de gravação, que define sua capacidade.

Ao final de uma sessão, o estado da cena no sistema é atualizado para “Gravado”. Além disso, sua faixa de áudio é armazenada de modo que possa ser recuperada posteriormente para sincronizá-la com a animação. Caso a cena escalada para aquela sessão não tenha sido gravada, seu estado deve ser reatualizado para “Não gravado”, de modo que possa ser programada para uma nova sessão. Caso uma cena precise ser regravada, por conta de uma falha ou falta de qualidade, seu estado deverá ser alterado para “Necessita de regravação”.

CONSULTAS À BASE DE DADOS

- (1) Para selecionar o dublador que interpretará determinado personagem, deve ser possível:
 - (a) Consultar personagens com mesmo timbre de voz que o personagem em questão;
 - (b) Consultar quais dubladores interpretaram cada um destes personagens, junto com a avaliação que cada dublador obteve.
- (2) Para agendar uma sessão de gravação, deve ser possível:
 - (a) Consultar quais cenas de determinado anime precisam ser gravadas;
 - (b) Consultar o script de determinada cena;
 - (c) Consultar quais personagens participam de determinada cena;
 - (d) Consultar o dublador que dubla determinado personagem em um episódio do qual uma cena faz parte;
 - (e) Consultar qual o diretor da temporada da qual uma cena faz parte;
 - (f) Consultar informações das salas de gravação: número da sala e capacidade (número de kits de gravação).
 - (g) Consultar informações dos técnicos de áudio que trabalham para a produtora: nome, CPF, telefones para contato, endereço, data de nascimento e email.
 - (h) Consultar informações sobre sessões de gravação já agendadas: sala de gravação, técnico auxiliar, data, horário de início, horário de término e cenas escaladas;
- (3) Deve ser possível recuperar o áudio gravado de uma determinada cena;
- (4) Deve ser possível recuperar informações relativas aos animes registrados pela produtora:

- (a) Informações gerais de todos os animes (ou de um em específico) registrados: nome, ano de publicação, autor da obra original, número de temporadas e estado de gravação;
 - (b) Informações de todos os personagens (ou de um em específico) que um anime possui: nome, descrição e timbre de voz;
 - (c) Informações de todas as temporadas (ou de uma em específico) de determinado anime: número da temporada, número de episódios, diretor, data inicial de exibição e data final de exibição;
 - (d) Informações gerais de todos os episódios (ou de um em específico) de determinada temporada: número do episódio, título, duração, número de cenas;
 - (e) Informações gerais de todas as cenas (ou de uma em específico) de um episódio: número da cena, duração, script, faixa de áudio e estado de gravação;
- (5) Deve ser possível recuperar informações relativas aos dubladores que trabalham com a produtora:
- (a) Informações de todas as agências de dublagem (ou de uma em específico) com as quais a produtora tem contato: nome, CNPJ, telefones para contato, endereço e email;
 - (b) Informações sobre todos os dubladores (ou de um em específico) que trabalham/trabalharam com a produtora (em um anime específico ou não): nome, CPF, sexo, agência à qual é associado, timbre de voz e data de nascimento;
- (6) Deve ser possível recuperar informações relacionadas aos diretores que trabalham na produtora:
- (a) Informações sobre todos os diretores (ou de um em específico): nome, CPF, telefones para contato, endereço, email e temporadas que dirige ou dirigiu;
- (7) Deve ser possível recuperar informações sobre os técnicos de áudio que trabalham na produtora:
- (a) Informações sobre todos os técnicos de áudio (ou de um em específico): nome, CPF, telefones para contato, endereço, data de nascimento e email.

Esta solução permite que várias cenas sejam gravadas em uma mesma sessão, mas não faz o controle para que las cenas sejam do mesmo anime. Desse modo, deixamos para que essa restrição fosse implementada a nível de aplicação (ou como trigger de banco).



Na agregação "Sessão de Gravação", a chave primária deverá ser a chave composta Sala, Data e Horário. A chave secundária deve ser a chave composta CPF, Data e Horário

Avaliação Total: é feito a média das avaliações parciais da relação dubia

O atributo "Estado Anime" está relacionado à quantidade de cenas gravadas. Se todas as cenas de um anime são marcadas como "Gravadas", o anime recebe o estado de "Gravado".

Possíveis problemas/inconsistências do DER

Ciclo: personagem->anime->temporada->episódio->cena->personagem e personagem->episódio->cena->personagem.

A entidade 'personagem' precisa estar relacionada às entidades anime e cena. A primeira relação é necessária pois 'personagem' é entidade fraca de 'anime', visto que a chave é o nome do personagem e ele não é único sem o respectivo anime (Ex: personagens de animes diferentes mas com mesmo nome). A segunda relação é necessária pois será preciso saber quais personagens estarão presentes na cena, a fim de identificar quais dubladores serão necessários para uma sessão de gravação. Os dubladores são identificados através da relação entre 'cena' e 'episódio' (para saber de que episódio aquela cena faz parte) + a relação "dubla" entre 'dublador', 'personagem' e 'episódio' (para saber que dublador dubla aquele personagem na cena daquele episódio).

Ciclo: Personagem->anime->temporada->episódio->personagem

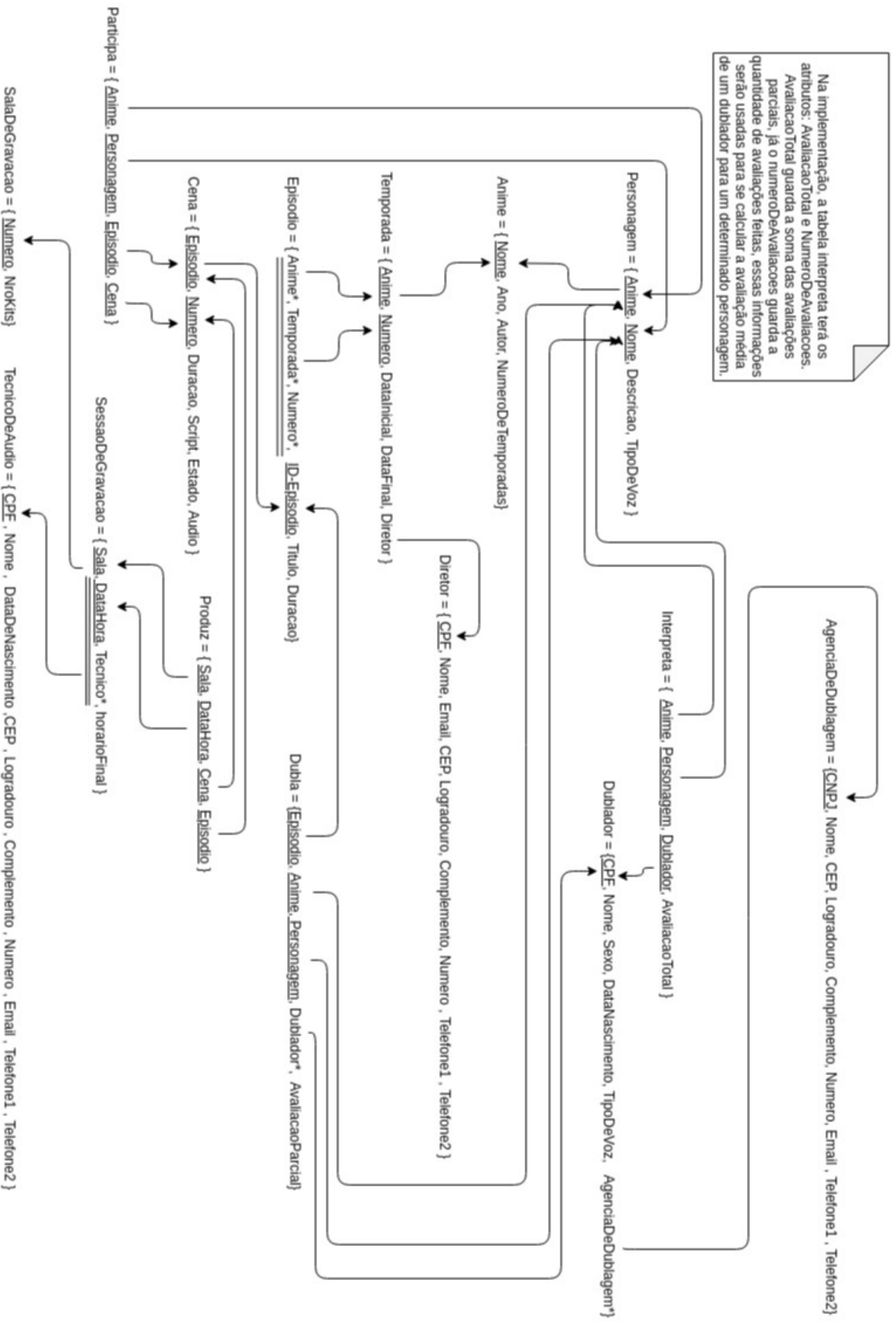
A entidade 'personagem' também precisa estar relacionada à entidade 'episódio' (e com 'dublador') para identificar o dublador que interpretou determinado personagem em determinada cena, visto que é possível que um personagem seja dublado por dubladores diferentes (pode ser que o dublador mude de um episódio para outro).

Possíveis conflitos em 'sessão de gravação'

A entidade agregada 'sessão de gravação' permite a existência de situações conflituosas, como duas cenas sendo gravadas no mesmo horário e na mesma sala, a mesma cena sendo gravada em duas salas diferentes no mesmo horário, a presença de dubladores ou de um diretor em duas salas ao mesmo tempo, etc. Esses conflitos foram deixados no diagrama pois é preciso que seja possível gravar a mesma cena, na mesma sala, mas em horários diferentes (caso seja preciso regravar uma cena). Para impedir que o sistema armazene informações conflituosas, será preciso tratar esses erros em outro nível de abstração.

Correções e Modificações em relação à parte 1

- Mudamos parte da seção "Descrição do Problema", movendo tudo relacionado às consultas feitas ao banco para uma nova seção "Consultas à base de dados". Adicionamos mais consultas e explicitamos melhor as existentes na seção "Consultas à base de dados".
- Modificamos a "Descrição do Problema" e o MER para que mais de uma cena pudesse ser gravada em cada sessão de gravação.
- Adicionamos um atributo HorárioFim nas sessões de gravação.



MODELO RELACIONAL

JUSTIFICATIVAS DAS SOLUÇÕES NO MODELO RELACIONAL

Atributo Endereço (atributo composto)

Solução Adotada:

No mapeamento do atributo composto 'endereço' de entidades como "Técnico de Áudio", "Diretor" e "Agência de Dublagem", optamos por alocar a informação nas tabelas das próprias entidades.

Vantagens:

- Busca eficiente, pois não há custo de junção de tabelas;
- Não desperdiça tanto espaço, visto que é improvável que os campos sejam nulos.

Alternativa:

Gerar uma nova tabela Endereço para cada uma das entidades (EnderecoDiretor, EnderecoTecnico...) com uma chave estrangeira da tabela da respectiva entidade. Isso seria favorável no cenário em que houvesse muitos campos nulos para o endereço na tabela das entidades, pois economizaria espaço, o que não ocorre no nosso caso. Além disso, a criação de uma nova tabela acrescentaria o custo de junção.

Atributo Telefone (atributo multivalorado)

Solução Adotada:

No mapeamento do atributo 'Telefone' das entidades "Técnico de Áudio", "Diretor" e "Agência de Dublagem", optamos por alocar uma quantidade fixa de telefones (2) na tabela de cada entidade.

Vantagens:

- Acesso direto aos telefones, visto que estão na mesma tabela da entidade

Desvantagens:

- Número limitado de telefones para cada entidade.

Alternativa:

Alocar a informação em tabelas separadas (TelefonesDiretor, TelefonesTecnicoDeAudio e TelefonesAgenciaDeDublagem) contendo o número de telefone e a chave estrangeira referenciando a tabela apropriada como chave primária composta. Isso permitiria que fosse registrado um número indefinido de telefones por entidade, mas acrescentaria o custo de junção. Estimamos que não seria preciso guardar muitos telefones por entidade, por isso deixamos uma quantidade fixa.

Entidade Fraca “Episódio”

Solução Adotada:

No mapeamento da entidade “Episódio”, percebemos que sua chave primária ficaria muito grande/complexa, visto que a entidade é fraca de “Temporada” que, por sua vez, é fraca de “Anime”. Assim, resolvemos gerar uma coluna ‘ID’ para servir de chave primária, enquanto que os campos ‘Anime’, ‘Temporada’ e ‘Numero’ servem de chave secundária.

Vantagens:

- Busca mais eficiente, visto que a chave primária é menor e simples.
- Diminui o custo de memória, visto que a chave primária desta entidade é referenciada por várias outras tabelas, o que implica na replicação do valor inteiro da chave em cada uma

Desvantagens:

- As chaves estrangeiras de tabelas que referenciam “Episódio” não contém mais a informação de “Anime” e “Temporada” do episódio, adicionando o custo de junção com a tabela de “Episódio” para recuperar tais dados.

Alternativa:

Utilizar a chave composta ‘Anime’ + ‘Temporada’ + ‘Numero’. Isso acarretaria em uma pequena economia de espaço na tabela da entidade por haver uma coluna a menos, além de incluir as informações de “Anime” e “Temporada” em tabelas que referenciam “Episódio”, melhorando algumas buscas. No entanto, a busca por episódio (que ocorre com grande frequência) seria muito menos eficiente. Além disso, há uma grande quantidade de chaves estrangeiras que referenciam a chave primária de “Episódio” e, com essa alternativa, o valor das colunas ‘Anime’, ‘Temporada’ e ‘Numero’ seria repetida diversas vezes, ocupando mais espaço do que o que é economizado na ausência do atributo ‘ID’.

Relação “Pertence” entre “Dublador” e “Agência de Dublagem” (relacionamentos 1:N c/ participação total do lado N)

Solução Adotada:

No mapeamento da relação “Pertence”, incluímos um atributo ‘Agencia’ na tabela Dublador como chave estrangeira para a tabela de AgenciaDeDublagem como *not null*.

Vantagens:

- Garante a semântica explicitada (cardinalidade da relação e participação total).

Alternativas:

A criação de uma nova tabela referenciando ambas as entidades não atende à restrição de participação total.

Relação “Dirige” entre “Diretor” e “Temporada” (relacionamento 1:N)**Solução Adotada:**

No mapeamento da relação “Dirige”, incluímos um atributo ‘Diretor’ na tabela Temporada como chave estrangeira para a tabela de Diretor.

Vantagens:

- Não há custo de junção nas consultas pelo diretor de determinada temporada.

Alternativas:

A criação de uma nova tabela referenciando ambas as entidades seria melhor caso houvesse muitas temporadas sem diretor registrado, pois economiza o espaço que seria ocupado por campos nulos na tabela de Temporada, porém acreditamos que não haverá muitas temporadas registradas sem diretor, e caso haja, as mesmas não estarão sem diretor por muito tempo.

Entidade Agregada “Sessão de Gravação”**Solução Adotada:**

No mapeamento da entidade “Sessão de Gravação”, utilizamos como chave primária ‘Sala’ + ‘Data’ + ‘Horario’ para impedir que uma sala fosse agendada para sessões simultâneas e como chave secundária ‘Tecnico’ + ‘Data’ + ‘Horario’ para impedir que um mesmo Técnico de Áudio ficasse responsável por auxiliar em sessões simultâneas.

Vantagens:

- Garante restrições semânticas, impedindo que uma sala de gravação seja agendada para sessões de gravação simultâneas e que um mesmo Técnico de Áudio fique responsável por auxiliar em sessões de gravação simultâneas.

Desvantagens:

- Não é possível atender algumas das restrições semânticas como: Diretores e Dubladores tendo que comparecer a sessões de gravação simultâneas.

Alternativa:

Chave composta ‘Sala’ + ‘Data’ + ‘Horario’ + ‘Tecnico’. Tal solução não cumpre os requisitos listados na “Solução Adotada” (impedir sala e técnico agendados para sessões simultâneas).

Relação “Produz”

Solução Adotada:

No mapeamento da relação “Produz” entre “Sessão de Gravação” e “Cena”, criamos uma nova tabela referenciando `SessaoDeGravacao` e `Cena`, deixando a composição das chaves estrangeiras como chave primária.

Vantagens:

- Garante restrição de cardinalidade.

Desvantagens:

- Não atende à restrição de participação total

Relação “Dubla”

Solução Adotada:

No mapeamento da relação “Dubla” entre “Personagem” e “Dublador” e “Episódio”, criamos uma nova tabela referenciando `Personagem`, `Dublador` e `Episodio`, deixando a composição das chaves estrangeiras de `Personagem` e `Episodio` como chave primária e `Dublador` como atributo *not null*.

Vantagens:

- Garante restrição de cardinalidade.

Soluções adotadas em Atributos Derivados

Atributo `AvaliacaoTotal` em `Interpreta`: Incluído como atributo na tabela de `Interpreta`, pois para determinar a média não teremos que realizar junção de tabelas. Ao invés disso, adicionamos na implementação um novo atributo na tabela `Interpreta` que especifica o número total de avaliações e o atributo `AvaliacaoTotal` guarda a soma de todas avaliações parciais feitas por um dublador com um personagem em um anime.

Como a avaliação parcial será incluída ao final de cada episódio, a avaliação total e o número de avaliações na tabela `Interpreta` poderão também ser atualizados.

Vantagens: Para determinar a avaliação total não teríamos que fazer várias consultas para obtermos as informações das avaliações parciais, bastaria consultar a tabela `Interpreta` e calcular a média com as informações já armazenadas.

Atributo `NroDeEpisodios` em `Temporada`: não foi incluído como atributo na tabela, pois essa informação é pouco usada.

Vantagens: Economia de dados e dispensa o uso de trigger. Para determinar o número de episódios deve ser realizado uma junção, essa junção seria uma

desvantagem, contudo acaba sendo vantajoso mantê-la devido a baixa utilização dessa informação.

Atributo **NroDeCenas** em **Episódio**: não foi incluído como atributo na entidade, pois a busca do número de cenas é pouco frequente.

Vantagens: Economia de memória e dispensa o uso de trigger. Para determinar o número de cenas deve ser realizado uma junção, essa junção seria uma desvantagem, contudo acaba sendo vantajoso mantê-la devido a baixa utilização dessa informação.

Atributo **NumeroDeTemporadas** em **Anime**: foi incluído como atributo na tabela, pois ele frequentemente é utilizado como uma informação a ser repassada do anime, além disso ele sofrerá poucas atualizações.

Vantagens: Para determinar NumeroDeTemporadas do Anime não teríamos que fazer consultas na tabela de temporada relacionado ao anime.

IMPLANTAÇÃO DA BASE DE DADOS

Mudanças

- Correções da Parte 2;
- Atributo “timbreDeVoz” de “Personagem” e “Dublador” alterado para “tipoDeVoz” no modelo Relacional;
- Atributos “data” e “hora” de “SessaoDeGravacao” e de “Produz” combinados no atributo “DataHora”;
- Participação Total de “Temporada” em “Dirige” removida. Assim, alterou-se a solução do modelo relacional e sua justificativa.
- No modelo relacional havia observações de alguns detalhes de implantação sobre a avaliação total de dublagem e elas foram alteradas. Agora precisamos armazenar apenas o número de avaliações feitas e a média de avaliações já calculada (informações suficientes para determinar a nova média).

Descrição

A aplicação foi desenvolvida na linguagem Java 8 e pode ser executada em qualquer sistema operacional com o Java 8 instalado. A base de dados foi construída utilizando-se o SGBD Oracle com scripts SQL.

O script de criação das tabelas encontra-se no arquivo *Esquema_Produtora.sql* e o script de alimentação inicial encontra-se no arquivo *Dados_Produtora.sql*. A aplicação pode ser executada através do arquivo *App.jar*.

Há duas categorias de consulta disponíveis: consultas relacionadas a Personagens e consultas relacionadas às Sessões de Gravação.

Em relação aos Personagens, é possível recuperar a tabela de todos os Personagens registrados no sistema que já foram dublados pelo menos 1 vez contendo anime, nome do personagem e tipo de voz. Para isso, o usuário deve clicar no botão “Personagens” no Menu Principal à esquerda. As tuplas da tabela podem ser filtradas por tipo de voz selecionando a opção desejada no canto superior esquerdo e clicando em “Buscar”.

A partir da tabela, o usuário pode selecionar um Personagem e clicar no botão “Informações do Personagem” no canto inferior direito para acessar um painel contendo, além dos dados anteriores, informações adicionais do personagem como Descrição e uma tabela de Dubladores que já interpretaram o mesmo. A tabela de Dubladores contém o nome do dublador, avaliação média obtida dublando o personagem e nome, email e telefone de sua agência de dublagem

Em relação às Sessões de Gravação, é possível recuperar uma lista de Sessões de Gravação agendadas no sistema contendo o número da sala, a data, os horários de início e término, o nome do anime cujas cenas serão gravadas, a temporada do anime e o nome do técnico de áudio auxiliar de cada. Além disso, para cada sessão é associada uma lista de cenas que serão gravadas na mesma e, para cada cena, é indicado o episódio a qual pertence. A lista de Sessões de Gravação pode ser filtrada para mostrar apenas sessões agendadas para uma data e um horário iguais ou posteriores à uma data e um horário definidos pelo usuário. Para isso, é possível marcar a opção “Selecionar por data inicial” no canto superior do painel, definir a data de referência e clicar em “Buscar”.

Código fonte das consultas

Código localizado em “*dao/PersonagensDubladosDAO*”, na linha 25, que recupera o anime, o nome e o tipo de voz de todos os personagens dublados:

```
sql = "SELECT DISTINCT pers.anime, pers.nome as personagem,
pers.tipodevoz as tipo_de_voz from interpreta inter" + " join
personagem pers on inter.anime = pers.anime AND inter.personagem =
pers.nome ORDER BY pers.anime";
```

Código localizado em “*dao/PersonagensDubladosDAO*”, na linha 29, que recupera anime, nome e tipo de voz de todos os personagens com um determinado tipo de voz:

```
sql = "SELECT DISTINCT pers.anime, pers.nome as personagem,
pers.tipodevoz as tipo_de_voz from interpreta inter" + " join
personagem pers on inter.anime = pers.anime AND inter.personagem =
pers.nome WHERE pers.tipodevoz = ? ORDER BY pers.anime";
```

Código localizado em “*dao/AvaliacaoDubladorPersonagemDAO.java*”, na linha 21, que busca, para todos os dubladores que dublaram determinado personagem, o seu nome, sua avaliação total, a descrição do personagem e nome, email e telefone da agência de dublagem a qual ele pertence:

```
String sql = "SELECT dub.nome, inter.avaliacaototal,
pers.descricao, ag.nome, ag.email, ag.telefone1 " + "FROM
personagem pers JOIN interpreta inter ON pers.anime = inter.anime
AND pers.nome = inter.personagem " + "JOIN dublador dub ON
inter.dublador = dub.cpf JOIN agenciaDeDublagem ag ON
dub.agenciaDeDublagem = ag.cnpj " + " WHERE inter.personagem = ?
AND pers.anime = ?";
```

Código localizado em “*dao/SessaoGravacao.java*”, na linha 31, que recupera sala, data, horário de início, horário de término, anime, temporada, episódios e as cenas dos episódios que serão gravados de todas as sessões de gravação ordenadas pela data e hora da sessão e a sala:

```
PreparedStatement stmt = Conector.getConector().prepareStatement(
"SELECT P.sala, TO_CHAR(P.data_hora, 'DD/MM/YYYY') as data, "
+ "TO_CHAR(P.data_hora, 'HH24:MI') as hora_inicial,"
+ "TO_CHAR(SES.horariofinal, 'HH24:MI') as hora_final,"
+ "EP.id_episodio, "
+ "EP.anime, EP.temporada, EP.numero as EPISODIO,"
+ "P.cena , C.estado as status, SES.tecnico , Te.nome "
+ "FROM PRODUZ P JOIN SESSAODEGRAVACAO SES ON P.SALA = SES.SALA
AND P.DATA_HORA = SES.DATA_HORA "
```

```

+ "JOIN CENA C ON C.EPISODIO = P.EPISODIO AND C.NUMERO = P.cena
"
+ "JOIN EPISODIO EP ON EP.ID_EPISODIO=C.EPISODIO "
+ "JOIN TECNICODEAUDIO Te ON Te.cpf = SES.tecnico "
+ "ORDER BY P.data_hora, P.SALA");

```

Código localizado em “*dao/SessaoGravacao.java*”, na linha 68, que recupera sala, data, horário de início, horário de término, anime, temporada, episódios e as cenas dos episódios que serão gravados de todas as sessões de gravação ordenadas pela data e hora da sessão e a sala, porém apenas as sessões com agendamento posterior à data passada são selecionadas

```

PreparedStatement stmt = Conector.getConector().prepareStatement(
"SELECT P.sala, TO_CHAR(P.data_hora, 'DD/MM/YYYY') as data, "
+ "TO_CHAR(P.data_hora, 'HH24:MI') as hora_inicial, "
+ "TO_CHAR(SES.horariofinal, 'HH24:MI') as hora_final, "
+ "EP.id_episodio, "
+ "EP.anime, EP.temporada, EP.numero as EPISODIO, "
+ "P.cena , C.estado as status, SES.tecnico , Te.nome "
+ "FROM PRODUZ P JOIN SESSAODEGRAVACAO SES ON P.SALA = SES.SALA
AND P.DATA_HORA = SES.DATA_HORA "
+ "JOIN CENA C ON C.EPISODIO = P.EPISODIO AND C.NUMERO = P.cena "
+ "JOIN EPISODIO EP ON EP.ID_EPISODIO=C.EPISODIO "
+ "JOIN TECNICODEAUDIO Te ON Te.cpf = SES.tecnico "
+ "WHERE SES.data_hora > TO_DATE(? , 'DD/MM/YYYY HH24:MI') "
+ "ORDER BY P.data_hora, P.SALA");

```

Observação - Atributos Script e Áudio da Tabela de Cenas

Não conseguimos utilizar o tipo BFILE para armazenar os arquivos de script e os arquivos de áudio das cenas na base de dados através do script SQL, pois tivemos problemas com privilégios negados. Fomos orientados, então, a utilizar o tipo BLOB, que armazena grandes arquivos binários nas próprias tabelas. Decidimos usá-lo, no entanto, também não conseguimos criar um script SQL para a inserção automatizada nas colunas correspondentes.

Uma das formas de inserção (dos BLOBs) era criar o script SQL inserindo ‘empty_blob()’ no SQLDeveloper acessando as colunas e depois editar manualmente seus valores, fazendo o upload dos arquivos a serem armazenados. (Cena (tabela) -> dados -> editar valor -> carregar -> escolher arquivo -> abrir -> commit).

Uma outra forma era converter o arquivo inteiro para hexadecimal e colá-lo no valor da coluna do BLOB no comando SQL, usando o comando ‘hextoraw’

```
insert into cena values (at1, at2, ..., hextoraw('todos os bytes do arquivo em hexadecimal')).
```

Para esse método, existem meios por código providos pelos próprios desenvolvedores dos SGBDs que disponibilizam suas bibliotecas para conexão com o banco para várias linguagens de programação.

No script SQL de alimentação da base de dados, usamos um pouco das duas formas. Em algumas tuplas, inserimos apenas uma pequena fração do arquivo convertido para hexadecimal, pois o arquivo inteiro ocuparia muitas linhas, o que tornaria o script desnecessariamente grande (numa criação real da base de dados, deveríamos inserir o arquivo completo, via uma aplicação). Nas outras tuplas, deixamos o script SQL inserindo 'empty_blob()'s nas colunas para que, posteriormente, os arquivos fossem inseridos manualmente.

Observação - Avaliação Total do Dublador na tabela Interpreta

Quando a tupla da tabela Interpreta possui o atributo avaliacaoTotal com valor NULL, a aplicação em Java interpreta o valor como 0. Portanto, definimos o valor "0" de Avaliação Total do dublador como "sem avaliação" e os valores de 1 a 10 como válidos.

Conclusão

O projeto foi de grande importância para a disciplina e para a graduação em si. A especificação de cada etapa e os critérios de avaliação estavam bem detalhados, objetivos e praticamente livres de ambiguidades. A liberdade na escolha do tema tornou o processo de desenvolvimento muito menos maçante e a orientação por parte da professora e do estagiário PAE foram de grande ajuda.

O ponto de maior dificuldade foi a criação da interface gráfica da aplicação, visto que o grupo tinha de pouca a nenhuma experiência com desenvolvimento desta. Sua implementação foi a etapa mais demorada da parte 3.

Em geral, foi possível aprender bastante sobre, pelo menos, o básico de modelagem, projeção e implantação de uma base de dados, além de adquirir prática nesses assuntos.

Acreditamos que seria de grande ajuda, no entanto, a disponibilização de um template para a parte 3 do projeto, assim como foi feito para as partes 1 e 2, como por exemplo, uma aplicação executável com observações de correção.