



**Universidade de São Paulo Instituto de Ciências
Matemáticas e de Computação**

Projeto - GAME XP- Parte 3

SCC0540 - Banco de Dados, Profa. Elaine Parros Machado de Sousa

Bruner Eduardo Augusto - 9435846

Carlos R Dos Santos Junior - 9435102

Marcelo Foresto Porto da Costa - 8782557

William Luis Alves Ferreira - 9847599

São Carlos - 25/11/2019

1.Descrição do Problema e dos Requisitos de Dados

A GAMES XP com o passar dos anos tem se tornado um dos grandes eventos de jogos na América Latina, atraindo cada vez mais pessoas interessadas no ambiente e tema que a mesma vem proporcionando. Tendo isto em mente, a organização do evento, sediado no Rio de Janeiro, contratou a eSchedule, líder em gestão de pessoas em grandes eventos, através da sua excelência de organização em scheduling e análise da base de dados. Com isso, o sistema proposto consegue extrair informações das quantidades necessárias de recursos humanos, tempo e dinheiro que serão necessários em eventos do mesmo porte.

A eSchedule, trabalha com três entidades cruciais para o scheduling: os **Locais**, as **Atrações**, e as **Pessoas**. Apriori um bom agendamento necessita da organização, distribuição e uma boa relação entre tais entidades. Para tal, utilizamos a entidade **Agenda**, a qual garante a unicidade na ocorrência de uma única atração em um único local, e a relação ternária **Acontece**, sendo esta a fundação do banco de dados, garantindo assim a consistência em alocar recursos e pessoas à uma atração com execução única em todo o evento.

À vista disso, é feito um cadastro no sistema onde as seguintes informações são armazenadas para cada entidade:

- Para os **Locais** temos os atributos Espaço e Nome;
- Para as **Atrações** temos os atributos Id e Tipo;
- Para as **Pessoas** temos os atributos Credencial, Setor, CPF, Sexo, Nome, Endereço (Endereço, Cidade, Estado, País), Idade e Designado;
- Para a **Agenda** temos os atributos Data (dd/mm/aaaa) e Hora (com intervalo de 10min);

A posteriori, temos que o ambiente do maior game-parque do mundo é dividido em vários **Locais**, onde os Espaços disponíveis para sediar as atrações são: GAMEPLAY Arena 1; GAMEPLAY Arena 2; INOVA Arena; OI GAME Arena; Palco Gênesis; Área Externa. Para o contexto de especificidade do ambiente, dentro das arenas ou palcos, temos stands e referências, respectivamente atribuídos à Nome.

Contudo, antes e durante o evento, conseguimos dispender os seguintes **Funcionários** - atribuídos com *Função*, que desempenha na Game-XP, e *reais/horas* pelo serviço prestado - e **Colaboradores** - atribuídos com *Cachê e Tipo*- através da entidade **Agenda** obtendo assim o melhor scheduling. A seguir são definidos cada tipo de **Funcionário** e **Colaborador**, respectivamente:

- **Segurança; Paramédico; Jurado; Narrador; Staff; Assessor; Técnico**
- **Competidores; Influenciadores; Músico; Artistas;**

Não obstante da aquisição e alocação de dados dos **Funcionários** e **Colaboradores**, a **Imprensa** que visita a GAMES XP também tem seus dados guardados, para termos um controle de quem está entrando em contato e quais suas profissões (*Grupo e Função*).

Para uma **Atração** acontecer em determinado local, é necessário a presença de uma equipe formada por uma quantidade, determinada, de **Seguranças**, onde estes estarão incumbidos de *vigiar* o público do evento. Há também a necessidade de uma equipe de **Paramédicos** para *zelar* pelas pessoas no evento, diante de qualquer adversidade, sendo que a equipe deve estar pronta para o atendimento.

Quando se trata do nível de infraestrutura (como palcos para os shows, equipamento de áudio e vídeo, e etc) para a **Atração**, guardamos na base de dados os **Técnicos**, os quais operam qualquer tipo de equipamento(*OperamEquipamento*) e tecnologia em prol da execução da GAMES-XP. Temos que salientar que os **Técnicos** têm diferentes especialidades, que são relacionadas à cada tipo de **Atração**.

Uma das entidades importantes para qualquer evento desse porte é o **Assessor** de imprensa, pois o mesmo fica responsável em se *comunicar* com a **Imprensa** especializada, a qual está *cobrindo* uma atração.

Dependendo da **Atração** temos a participação de **Jurados** e **Narradores**, onde o primeiro *fiscaliza* as **Partidas** na **Competição**, como também *julga* no **Concurso Cosplay**. Já o segundo apenas se relaciona com as **Partidas**, *narrando-as*.

Diante da formação de **Equipes** (junção de **Competidores**) guardamos o *NomeEquipe* na base de dados e a *quantidade* de participantes da mesma.

A entidade **Competição**, por possuir a característica de distinguir os grupos de **Partidas** que a compõem, a mesma não interfere na semântica de scheduling, sendo os fatores *NomeJogo*, *Categoria* e *Premiação* os atributos diferenciadores.

A atração que também cativa o público são as palestras, onde **Influenciadores** compõem uma painél também administrado pela **Staff** da Games-Xp. Por conseguinte, como o **Concurso Cosplay** é importante para o público, dividimos o mesmo em *Etapas*, tal com as **Partidas**, (oitavas-de-final;quartas-de-final,...).

A GAME-XP disponibiliza uma série de outras atrações como: a vinda de **artistas** -de variados *estilos* - do mundo inteiro para exposição de suas obras no **ArtStreet**; a apresentação de shows por **Bandas**, as quais possuem um *NomeDoGrupo* e são formadas por vários **Músicos** (*Função*); e atrações **Externas** como ativações de empresas, em espaços abertos, tirolesa e roda gigante, as quais estão distribuídas pelo game parque e estão especificadas no banco de dados através do *Nome*.

Tomando os elementos envolvidos e as relações entre si, é possível estimar o custo/tempo gasto durante o evento e assim conseguir acessar as seguintes informações, e ações, na base de dados:

Consulta

- Para alocar pessoas para uma Atração deve ser possível:
 - Consultar as seguintes pessoas disponíveis: Staff, Segurança, Paramédico, Narrador, Jurado e Assessor;
 - Consultar os técnicos que estão disponíveis e ver a sua especialização;
 - Consultar os colaboradores que irão participar de tal atração;
 - Consultar a Imprensa que estará disponível na atração;
 - Consultar informações sobre a atração.
- Deve ser possível consultar as informações de todas as pessoas que vão participar do evento, pessoas estas: Imprensa, Funcionários e Colaboradores;
- Deve ser possível consultar as Atrações, bem como Local onde ocorrem, tipo de atração, data e horário que vão ocorrer;
- Deve ser possível consultar as informações de todas as atrações disponíveis no evento;
- Deve ser possível consultar dados referentes à competição, tais como as partidas que a compõem, quais os jogadores e suas respectivas equipes, o local e por fim a hora ea data;
- Deve ser possível consultar onde cada funcionário deve estar em uma determinada data e hora;

- Deve ser possível consultar o custo total do evento em relação às pessoas que estão participando;
- Nos funcionários, deve ser possível calcular uma média salarial de cada função desempenhada, como também deve ser possível calcular para cada funcionário individualmente;
- Nos colaboradores deve ser possível calcular o cachê de cada um como também a média do cachê de todos os colaboradores;
- Deve ser possível consultar o número de Pessoas que estão participando do evento.

Inserção

- Deve ser possível inserir **Pessoas** para uma atração.

Remoção

- O desligamento de qualquer **Colaborador**, **Funcionário** ou **Imprensa**, o mesmo deve ser removido das Base de Dados;
- Qualquer mudança de horário das Pessoas, deve ser possível direcioná-las para outras atrações.

Possíveis inconsistências/problemas no DER

- A entidade **Agenda** garante a unicidade na ocorrência de uma única atração em um único local, porém no MER não podemos garantir o overlap entre intervalos válidos armazenados nos atributos da entidade, como: *INOVA Arena - stand 1, atração: Ubisoft, Brasil e inovação agenda: 14:20 dia 25/07/2019* e também no local: *INOVA Arena - stand 1, atração:Shield Maiden agenda: 14:30 dia 25/07/2019* precisa ser tratado com triggers e em último caso em aplicação.
- Consideramos que dado uma **Atração** o funcionário estará presente durante a totalidade da execução da mesma, sendo assim, o monitoramento em tempo real não cabe ao sistema. Com ressalva às atrações externas, como roda gigante, neste caso a totalidade de tempo da atração será fragmentada em turnos de X horas.
- Não cabe ao sistema gerenciar as competições apenas as partidas possuem unicidade de ocorrência e alocação para a semântica do sistema.

Correções e Modificações em relação à parte 1

A entidade genérica **Atração**:

- Constatou-se após o mapeamento a necessidade de corrigir o MER desta entidade, pois nossa semântica requer relações comuns entre todas as atrações (específicas) para também garantir a identificação não padronizada entre as mesmas. Contudo, a chave primária da entidade genérica será um id sintético apenas para manter a consistência, tanto das informações das entidades específicas quanto na participação do ternário **Acontece**. Com isso, todas as entidade específicas de **Atração** possuíam suas chaves primárias, onde tais chaves seriam mapeadas como chaves secundárias nas tabelas entidades específicas de atração .

A entidade genérica **Colaborador**:

- Após o mapeamento optou-se em retirar a totalidade da generalização e acrescentar na entidade específica **Influenciador** o atributo *Apelido*. Com essas modificações é possível optar pelo mapeamento alternativa 1 - procedimento padrão 1 visto em aula, o que otimiza o espaço de armazenamento, porém torna mais complexas a distinção e procura na tabela Colaborador, a fim de obter as informações gerais sobre as entidades específicas.

A relação **Acontece**:

- Foi observado que após a correção da primeira parte a participação total na relação acontece não faz sentido semântico, já que queremos possibilitar a inserção de atrações sem vinculá-las imediatamente ao local e hora.

A entidade **Pessoa**:

- Notou-se que não há sentido semântico, em partes do atributo composto endereço sendo este formado por país, estado, cidade, bairro, rua, número e complemento. Tomaremos endereço como atributo composto por país, estado, cidade e endereço, pois não faz sentido no domínio a diferenciação entre número de um endereço entre pessoas. Logo os atributos bairro, rua, número e complemento se tornarão uma única string chamada endereço.

A entidade **Concurso Cosplay**

- Após o mapeamento verificou-se a necessidade de refinar a semântica de identificação desta entidade, portanto definimos que o concurso de cosplay é um acontecimento único durante toda Game Xp, dessa forma, cada etapa do mesmo é atômicamente representado no mapeamento.

Justificativas das soluções do Modelo Relacional

Relação Forma entre Equipe e Competidor (relacionamentos 1:N c/ participação total do lado 1)

- **Solução Adotada:**
 - Como visto em aula, optamos por colocar a chave estrangeira na entidade com participação “N” na relação **-Competidor-** referenciando assim NomeEquipe pertencente a tabela **Equipe**.
- **Observação:**
 - A totalidade na participação da relação, por parte da entidade **Equipe**, não é garantida em nosso mapeamento, pois a chave estrangeira depende da existência de uma instância correspondente ao atributo NomeEquipe que associará a tabela **Competidor** com a tabela **Equipe**, garantindo assim a cardinalidade 1:N. Contudo, trataremos desta dependência no SGBD via trigger, ou caso seja necessário trataremos via aplicativos.

Relações N:N entre as entidades Atração e entidades específicas da entidade genérica de Funcionário

- **Solução Adotada:**
 - Com o intuito de manter a consistência e tornar claro a distribuição dos funcionários entre as atrações do evento, optamos por criar explicitamente cada tabela correspondente às relações tratadas neste tópico.
- **Alternativa:**
 - Conceber uma tabela **ServiçosGerais**={ID,Credencial}, correspondentes às relações *Cobre*, *Cuida* e *Zela*, com ID sendo chave estrangeira de **Atração** e Credencial por sua vez chave estrangeira de **Funcionários**. E conceber uma tabela **ServiçosEspecificos**= {ID,Credencial}, Função/Especialidade} sendo que esta mapeia as relações *Administra* e *OperaEquipamento*.

Entidade Competição e Partida

- **Solução Adotada:**
 - Com o intuito de armazenar o mínimo sobre as competições, desta forma, concebe-se a entidade **Competição** para manter consistência entre as etapas existentes na entidade **Partida**, para que seja possível inserir tanto: CS:GO, FEMININO, FINAIS; quanto CS:GO, MASCULINO, FINAIS. Os atributos *NomeJogo*, *Categoria* e *Etapas* são mapeadas como chave secundária composta da entidade **Partida**, no qual estas são chaves estrangeiras vindas da entidade **Competição**.
- **Conflito na Generalização da Entidade Genérica Atração e Entidade Específica Partida:**
 - Como a chave primária é suficiente para identificar a entidade **Partida**, sendo esta herdada de **Atração**, a chave secundária deve apenas distinguir as etapas iguais correspondentes a competições diferentes.

Entidade Banda e as relações Forma e Apresenta

- **Solução Adotada:**
 - Optou-se por convergir os três conjuntos, de modo que a entidade **Show** terá como chaves: primária *NomedaBanda* e secundária *ID* garantindo a cardinalidade 1:N entre **Show** e **Músico**. Por fim adicionamos uma chave estrangeira, default Null, chamada *NomeBanda*, em **Músico**, permitindo que ao longo do evento o mesmo possa ser associado através desta chave estrangeira à banda correspondente, a qual será vinculada à **Atração** através do *ID* na tabela **Show**.
- **Observação:**
 - A entidade específica herda a chave primária da entidade genérica, como definido no procedimento 8 visto em sala. Porém, para manter a cardinalidade correta entre **Show** e **Músico**, 1:N, optamos por utilizar o *ID* como chave secundária na primeira entidade específica, devido ao fato da semântica de se instanciar uma tupla **Show** antes do evento, já que a mesma define o nome da banda que ocupa aquela atração. Posteriormente, até mesmo durante a Game Xp, será possível associar os **Músicos**, da entidade específica **Colaboradores**.

Entidade genérica Funcionário

- **Solução Adotada:**
 - As informações das entidades específicas de funcionário não possuem atributos próprios, por esse motivo optamos por alocar as informações dessas entidades na tabela funcionários, havendo a distinção pelo atributo *função*.
- **Alternativa**
 - Criar uma tabela funcionário com a chave primária e o atributo *função*, que atribui à correspondente entidade específica, devido ao fato de lidarmos com uma generalização total exclusiva.

Entidade genérica Colaborador

- **Solução Adotada:**
 - Devido a semântica do domínio e com o modelo MER corrigido, acrescentando o atributo *Apelido* na entidade específica influenciador e removendo a totalidade da generalização, assim, optamos por seguir a alternativa 1 - procedimento Padrão 1 visto em aula.
- **Desvantagens:**
 - A disposições das tabelas específicas da entidade Colaborador não disponibiliza um sumário por setor que naturalmente se encaixa na proposta do scheduling .
- **Vantagens**
 - Falta de otimização na reutilização de colunas similares entre as tabelas específicas da entidade genérica Colaborador.
- **Alternativa**
 - Devido a semântica do problema e o modelo relacional concebido, tratamos a entidades genérica Colaborador como mutuamente exclusiva e temos como solução o procedimento 8.

Descrição da Implantação da Base de Dados

Diante das partes anteriores corrigidas, tomamos o Modelo Entidade-Relacionamento e o Modelo Relacional como o norte para a criação da base de dados, o e-Scheduling, a qual foi desenvolvida utilizando o SGDB da Oracle com scripts SQL. A conexão com o usuário vem pelo meio da nossa interface, feita em JAVA, a sendo esta funcional em qualquer sistema operacional com JAVA instalado. Para nosso projeto, testamos nos S.O's Windows 10 e Mint Linux.

Os scripts como *criacao_bd.sql* e *feed_bd.sql*, cuidam respectivamente da criação das tabelas e da alimentação inicial das mesmas. Portanto caso o Banco de dados esteja sendo inicializado pela primeira vez, basta que tais scripts sejam processados para que a base seja montada. Já o script de consultas, o qual já está integrado dentro da aplicação e pode ser facilmente encontrado no query_bd.sql (junto com inserções default para observação do funcionamento)l, deixa disponível para o usuário 5 consultas, onde tais são baseadas em informações de funcionários, atrações e outras que são transmitidas do Oracle SQL para a interface com o usuário.

Os arquivos estão compactados no arquivo eSchedulin.zip, o qual se encontra no escaninho do aluno Carlos R Dos Santos Junior - 9435102

Código Fonte de Alimentação

```
--Script para a insercao de pelo menos 2 tuplas por tabela:

--ACONTECE
INSERT INTO ACONTECE (ID, DATA, HORAS, ESPACO, LOCAL) VALUES (123, 12/11/1991, 12:12:01, 'SAGAO', 'AUDITORIO');
INSERT INTO ACONTECE (ID, DATA, HORAS, ESPACO, LOCAL) VALUES (12, 12/11/1921, 15:09:01, 'SAGAO', 'AUDITORIO');

--LOCAL
INSERT INTO LOCAL (ESPACO, NOME) VALUES ('AUDITORIO', 'COMPETICAO');
INSERT INTO LOCAL (ESPACO, NOME) VALUES ('AUDITORIO', 'SHOW');

--ATRACAO
INSERT INTO ATRACAO (ID , TIPO) VALUES (1234, 'PARTIDA');
INSERT INTO ATRACAO (ID, TIPO) VALUES (4321, 'EXTERNNA');
```

- O código de alimentação, feed_bd.sql, é basicamente a inserção de duas tuplas por tabelas. Entretanto para o funcionamento correto, colocamos que cada inserção nas Entidades especializadas de **Pessoa** e **Atração**, devem ser inseridas também nas suas respectivas tabelas generalistas, respectivamen. Assim, para duas inserções em **Funcionários**, **Imprensa** e **Colaboradores**, devem ser inseridas com o mesmo identificador em **Pessoa**, totalizando 6 inserções nesta tabela. A localização do código fonte de alimentação encontra-se dentro do eSchedulin.zip .

Código Fonte de Inserção

```
btnProximo.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent actionEvent) {
        bd.query("INSERT INTO ATRACAO VALUES (" + id + ", 'PARTIDA')");

        String s = "INSERT INTO PARTIDA VALUES(" + id +
            ", " + "'" + txtCategoria.getText() + "'" +
            ", " + "'" + txtNomeJogo.getText() + "'" +
            ", " + "'" + txtEtapa.getText() + "'" +
            ", " + "'" + txtEquipe1.getText() + "'" +
            ", " + "'" + txtEquipe2.getText() + "'" + ")";
        bd.query(s);
    }
});
```

- O código acima foi tomado como exemplo para demonstrar como inserimos dados de **Partida** no nosso banco. Todas as outras inserções no banco via usuário são feitas da mesma maneira: pega-se o que o usuário digitou, e dado a atração ou pessoa, inserimos da maneira correta. No caso acima, como o SGBD Oracle não possui de maneira facilitada um id com auto incremento, optamos pela aplicação para fazer essa alteração nos id's. Também vale salientar que estamos inserido na ordem das colunas que compõem a tabela em questão.
- localização do trecho no código fonte (addPartida.java, no projeto trabalho, na linha 39)

Código Fonte de Consulta

- Deve ser possível consultar quantos assessores estavam assessorando um grupo de imprensa em um espaço específico da Game-Xp

```
SELECT A.ESPACO, COUNT(FUN_A.ASSESSOR)
FROM COMUNICA FUN_A JOIN COBRE IMP
ON (FUN_A .IMPrensa = IMP.IMPrensa)
JOIN ACONTECE A
ON (A.ID = IMP.ID)
GROUP BY A.ESPACO
```

- Usando a junção das tabelas comunica e cobre, além da tabela acontece, conseguimos agrupar por espaço e mostrar ao usuário, o número de assessores que estão no local trabalhando com os grupos de imprensa que cobrem a atração.
- O trecho do código acima encontra-se tanto no arquivo query_bd.sql , quanto no método de consulta da aplicação, e ambos estão dentro do eSchedulin.zip

- Deve ser possível consultar os colaboradores da Game Xp, agrupando-os por estado, tipo e disponibilizando quantos existem por estado e a média dos cachês pagos aos mesmo.

```
SELECT NOME, ESTADO, TIPO, SEXO , COUNT(*), AVG(CACHEE) AS MED
FROM COLABORADORES
GROUP BY ESTADO, TIPO, SEXO
HAVING MED BETWEEN 30 AND 40
ORDER BY NOME;
```

- Para apresentar os dados de colaboradores ao usuário, escolhemos sobre os agrupamentos de estado , tipo e sexo àqueles que tem a média de cachê entre 30 e 40. Por fim ordenamos eles pelo nome.
 - O trecho do código acima encontra-se tanto no arquivo query_bd.sql , quanto no método de consulta da aplicação, e ambos estão dentro do eSchedulin.zip
- Deve ser possível consultar onde cada segurança deve estar em uma determinada data e hora

```
SELECT ESPACO, DATA, HORAS
FROM FUNCIONARIOS F JOIN CUIDA S
ON (F.CREDENCIAL = S.SEGURANCA)
JOIN ACONTECE A
ON (S.ID = A.ID)
GROUP BY F.CPF
ORDER BY ESPACO
```

- Uma consulta simples onde conseguimos juntar os dados da tabelas acontece, funcionários e segurança, assim conseguimos mostrar ao usuário as informações dos seguranças , agrupados por cpf e ordenados pelo nome do espaço.
 - O trecho do código acima encontra-se tanto no arquivo query_bd.sql , quanto no método de consulta da aplicação, e ambos estão dentro do eSchedulin.zip
- Deve ser possível consultar as **Atrações** que vão ocorrer bem como local onde ocorrerão, tipo de atração, data e horário

```
SELECT TIPO, ESPACO, LOCAL, DATA, HORAS
FROM ACONTECE H JOIN ATRACAO A
ON(H.ID = A.ID)
WHERE DATA > SYSDATE
```

- De uma maneira sucinta, apenas conferimos quais atrações estão ocorrendo , com o uso do join, e da comparação dos ID, e apenas selecionamos aqueles onde o campo data é maior que a data do sistema. Lembrando que data é relacionada à ano, mês dia.
 - O trecho do código acima encontra-se tanto no arquivo query_bd.sql , quanto no método de consulta da aplicação, e ambos estão dentro do eSchedulin.zip
- Consultar os seguintes funcionários staff disponíveis:

```
SELECT F.CREDENCIAL, F.NOME
FROM ADMINSTRA A , FUNCIONARIOS F
WHERE ( (F.FUNCAO = 'STAFF') AND (F.CREDENCIAL != A.STAFF) )
ORDER BY F.NOME;
```

- Através do filtro que aplicamos na tabela **funcionários**, cuja função são **staff**, apenas verificamos se sua credencial não consta na tabela **administra**(lembrando que só estão aqui a staff que está administrando uma atração, e não os ociosos)
- O trecho do código acima encontra-se tanto no arquivo query_bd.sql , quanto no método de consulta da aplicação, e ambos estão dentro do eSchedulin.zip.

Conclusão

Conforme cada etapa do projeto foi sendo feita, e corrigida, cada vez mais ficava nítido que tanto a aquisição dos requisitos, definição do escopo, modelagem das Entidades-Relacionamento, Modelo Relacional e codificação da base de dados corretamente, tinham tanto valor quanto visualizar em nossos erros base das decisões tomadas, as quais nem sempre são fáceis de se ver que levam à problemas que ofuscam a qualidade e eficiência do banco. Como sugestão, tornar o tema mais livre ou mesmo colocar em algumas categorias como: jogos, esporte, finanças, redes sociais, dentre outras, deixaria o processo de desenvolvimento mais dinâmico, devido ao fato de que como temas muito fechados nem sempre agradam a todos. Por sua vez um tema que seguisse por categorias poderia estimular àqueles que por vezes não se conectam com o tema proposto. Outro ponto, que por vezes não há como controlar, foi a falta de um estagiário PAE para dividir o tempo dispendido pela professora no atendimento aos alunos.

Contudo, a realização do projeto revela-se de tamanha importância para aprender e fundar nossa base de projetistas que não só vêem o papel importante da modelagem mas como também o de aprender de maneira rápida à corrigir os erros da melhor maneira possível, visando a qualidade e eficiência de uma base de dados relacional.