

INSTALAÇÃO

Espaço de Eventos Culturais

Ricardo Ferreira (201907835)
Sofia Teixeira (201806629)
Tomás Martins (201704976)

Índice

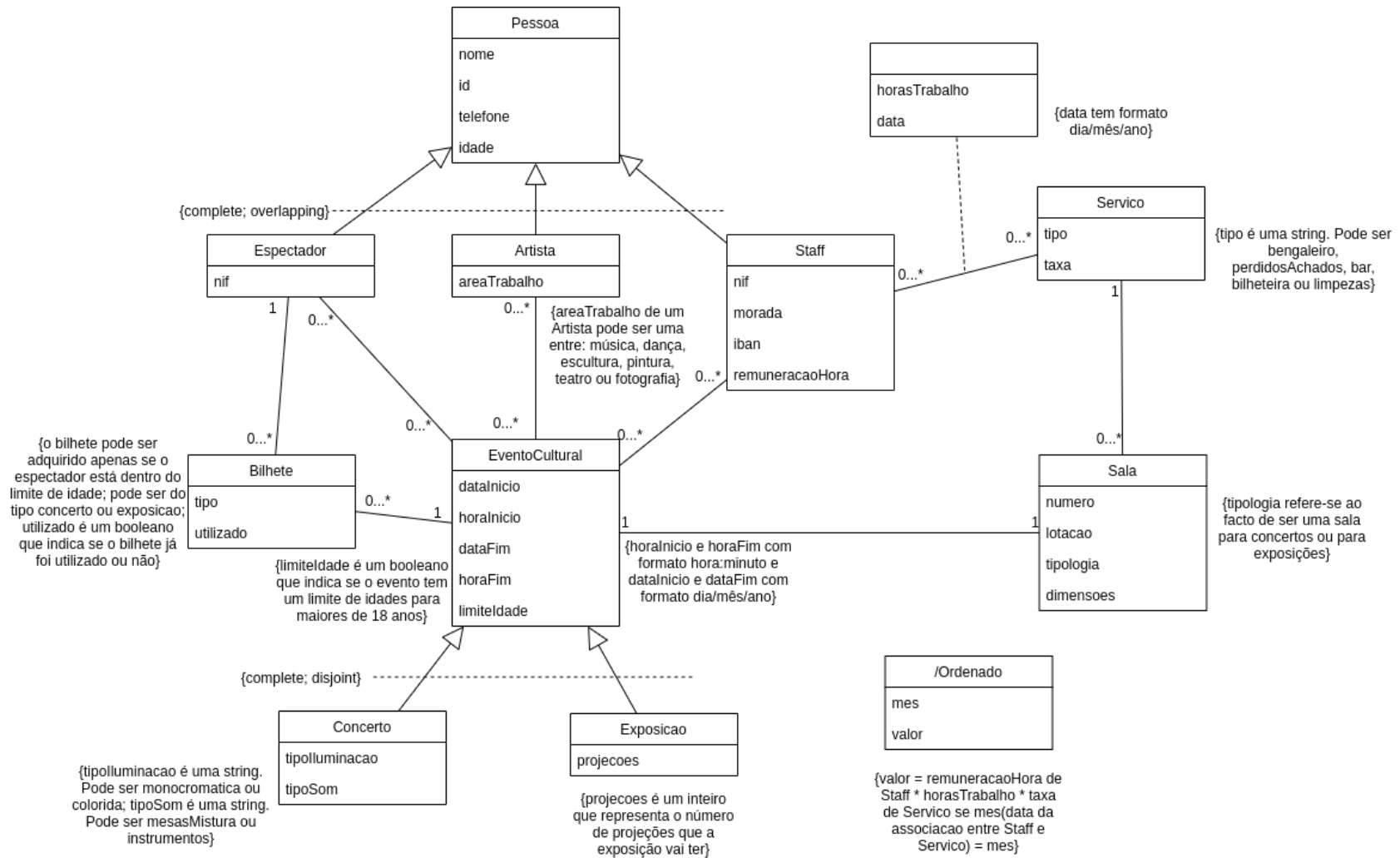
1. Contexto.....	2
2. Diagrama UML.....	3
3. Diagrama UML Revisto.....	4
4. Esquema Relacional.....	5
5. Análise de Dependências Funcionais e Formas Normais.....	7
6. Restrições.....	9

1. Contexto

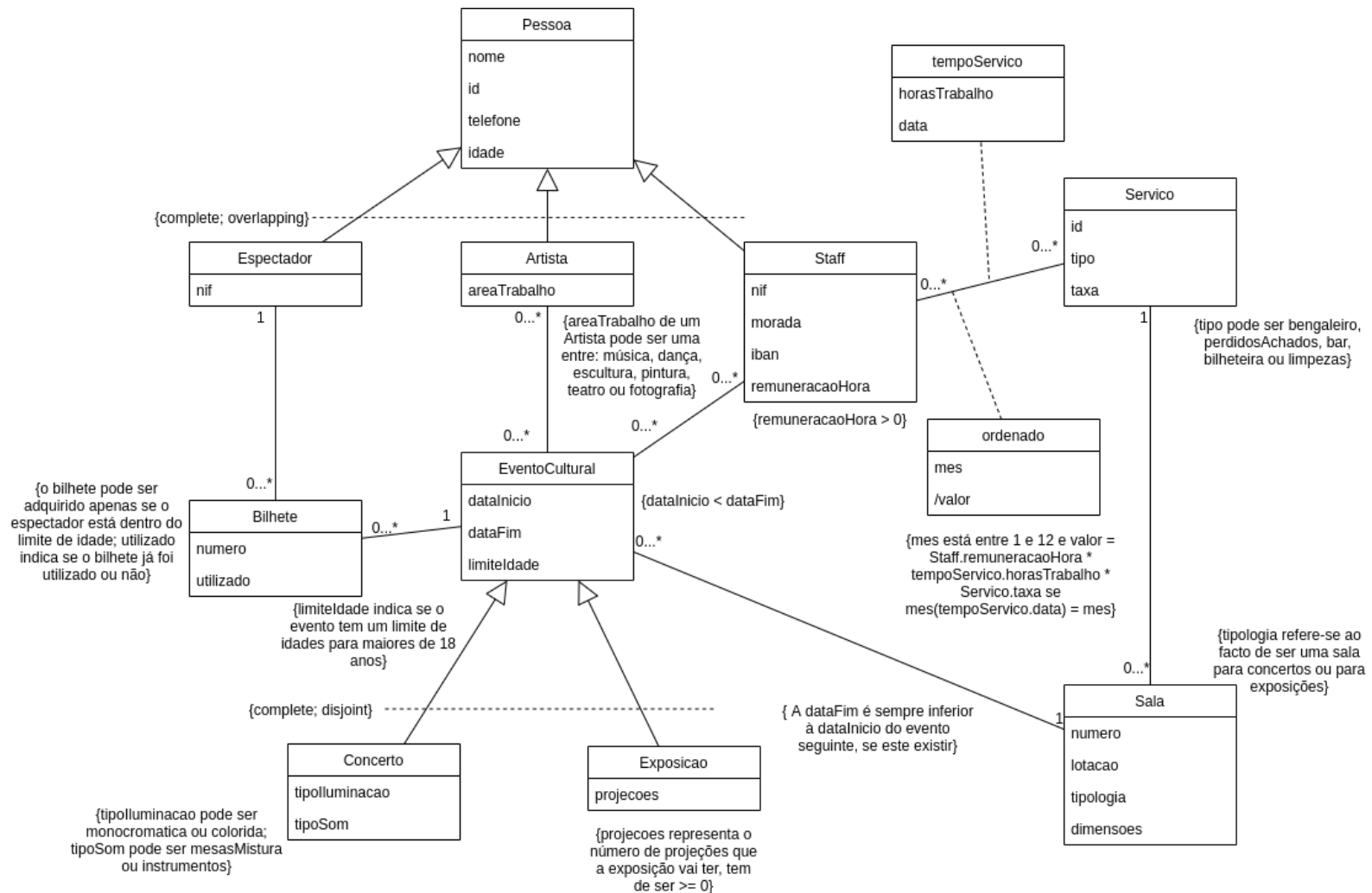
INSTALAÇÃO é um espaço que acolhe eventos culturais e presta serviços a clientes e artistas. Pretendemos, por isso, criar uma base de dados para gerir a logística inerente à sua atividade.

O espaço é composto por várias salas, constituídas pelo número, lotação, tipologia (para exposições ou para concertos) e dimensões. Cada sala pode ter um único evento cultural na mesma altura. Os eventos culturais têm datas de início e de fim. Estes podem, também, ter um limite de idade (para maiores de 18 anos). Os eventos podem ser concertos ou exposições. As exposições podem usar um número de projeções pretendidas e os concertos tipos de iluminação (monocromática ou colorida) e de equipamentos sonoros (mesas de mistura ou instrumentos). Cada sala pode oferecer um serviço, especificado pelo tipo (bengaleiro, perdidos e achados, bar, bilheteira ou limpezas) e uma taxa associada. Cada serviço pode ter vários elementos do staff associado e as horas de trabalho de cada elemento, bem como a data em que o serviço foi realizado. Das pessoas sabe-se o nome, telefone, ID e idade. As pessoas podem ser elementos do staff, dos quais se sabe, também, a morada, IBAN (para transferência do salário), o NIF e a remuneração por hora; espectadores, dos quais se sabe, adicionalmente, o NIF; artistas, dos quais se conhece, também, a área de trabalho (música, dança, escultura, pintura, teatro ou fotografia). Existem bilhetes, cada um com um número diferente, assim como o facto de este ter sido utilizado ou não. O ordenado do staff é calculado automaticamente, para cada mês, a partir da remuneração por hora desse elemento, as horas de trabalho feitas durante um serviço e a taxa deste.

2. Diagrama UML



3. Diagrama UML Revisto



4. Esquema Relacional

- Pessoa(nome, id, telefone, idade)
Chave primária: id
- Espectador(id-> Pessoa.id, nif)
Chave primária: id
Chave estrangeira: id
- Artista(id-> Pessoa.id, areaTrabalho)
Chave primária: id
Chave estrangeira: id
- Staff(id-> Pessoa.id, nif, morada, iban, remuneracaoHora)
Chave primária: id
Chave estrangeira: id
- Servico(id, tipo, taxa)
Chave primária: id
- ordenado(idStaff->Staff.id, idServico->Servico.id, mes, valor)
Chave primária composta: idStaff, idServico
Chave estrangeira composta: idStaff, idServico
- tempoServico(idStaff->Staff.id, idServico->Servico.id, horasTrabalho, data)
Chave primária composta: idStaff, idServico
Chave estrangeira composta: idStaff, idServico
- Sala(numero, lotacao, tipologia, dimensoes)
Chave primária: numero
- EventoCultural(id, dataInicio, horaInicio, dataFim, horaFim, limiteIdade)
Chave primária: id
- Concerto(id-> EventoCultural.id, tipoIluminacao, tipoSom)
Chave primária: id
Chave estrangeira: id
- Exposicao(id-> EventoCultural.id, projecoes)
Chave primária: id

Chave estrangeira: id

- **Bilhete**(numero, utilizado)

Chave primária: numero

5. Análise de Dependências

Funcionais e Formas Normais

- Pessoa(nome, id, telefone, idade)
{id} -> {nome, telefone, idade}
- Espectador(id-> Pessoa.id, nif)
{id} -> {nif}
{nif} -> {id}
- Artista(id-> Pessoa.id, areaTrabalho)
{id} -> {areaTrabalho}
- Staff(id-> Pessoa.id, nif, morada, iban, remuneracaoHora)
{id} -> {nif, morada, iban, remuneracaoHora}
{nif} -> {id}
{iban} -> {id}
- Servico(id, tipo, taxa)
{id} -> {tipo, taxa}
- ordenado(idStaff->Staff.id, idServico->Servico.id, mes, valor)
{idStaff, idServico} -> {mes, valor}
- tempoServico(idStaff->Staff.id, idServico->Servico.id, horasTrabalho, data)
{idStaff, idServico} -> {horasTrabalho, data}
- Sala(numero, lotacao, tipologia, dimensoes)
{numero} -> {lotacao, tipologia, dimensoes}
- EventoCultural(id, dataInicio, horaInicio, dataFim, horaFim, limiteIdade)
{id} -> {dataInicio, horaInicio, dataFim, horaFim, limiteIdade}
- Concerto(id -> EventoCultural.id, tipoIluminacao, tipoSom)
{id} -> {tipoIluminacao, tipoSom}
- Exposicao(id -> EventoCultural.id, projecoes)
{id} -> {projecoes}

- Bilhete(numero, utilizado)
 $\{\text{numero}\} \rightarrow \{\text{utilizado}\}$

Nenhuma das relações anteriores constitui uma violação à Forma Normal Boyce-Codd (BCNF), uma vez que uma relação R está na BCNF se para cada dependência funcional $\bar{A} \rightarrow \bar{B}$, $\bar{A} \rightarrow \bar{B}$ é trivial ou é uma chave/superchave. Como esta forma é um subconjunto da 3ª Forma Normal (3NF), então também respeitam a 3NF.

6. Restrições

- Pessoa
 - nome não pode ser nulo (NOT NULL)
 - id é chave primária (PRIMARY KEY)
 - idade não pode ser nulo (NOT NULL) e tem de ser superior a 0 (CHECK idade > 0)
- Espectador
 - id é chave primária (PRIMARY KEY) e chave estrangeira (FOREIGN KEY)
 - nif tem de ser único para cada espectador (UNIQUE) e não pode ser nulo (NOT NULL)
- Artista
 - id é chave primária (PRIMARY KEY) e chave estrangeira (FOREIGN KEY)
 - areaTrabalho não pode ser nulo (NOT NULL) e tem de ser uma entre música, dança, escultura, pintura, teatro ou fotografia (CHECK (areaTrabalho = "musica" OR areaTrabalho = "danca" OR areaTrabalho = "escultura" OR areaTrabalho = "pintura" OR areaTrabalho = "teatro" OR areaTrabalho = "fotografia"))
- Staff
 - id é chave primária (PRIMARY KEY) e chave estrangeira (FOREIGN KEY)
 - não existem espectadores com o mesmo nif (UNIQUE) e este não pode ser nulo (NOT NULL)
 - morada não pode ser nulo (NOT NULL)
 - não existem espectadores com o mesmo iban (UNIQUE) e este não pode ser nulo (NOT NULL)
 - remuneracaoHora não pode ser nula (NOT NULL) e tem de ser superior a zero (CHECK (renumeracaoHora > 0))
- Servico
 - id é chave primária (PRIMARY KEY)
 - tipo não pode ser nulo (NOT NULL) e tem de ser um entre bengaleiro, perdidos e achados, bar, bilheteira ou limpezas (CHECK (tipo = "bengaleiro" OR tipo = "perdidosAchados" OR tipo = "bar" OR tipo = "bilheteira" OR tipo = "limpezas"))
 - taxa não pode ser nula (NOT NULL) e tem de ser superior a zero (CHECK (taxa > 0))

- ordenado
 - idStaff é chave primária (PRIMARY KEY) e chave estrangeira (FOREIGN KEY)
 - idServico é chave primária (PRIMARY KEY) e chave estrangeira (FOREIGN KEY)
 - mes não pode ser nulo (NOT NULL) e tem de estar compreendido entre 1 e 12 (CHECK (mes BETWEEN 1 AND 12))
 - valor não pode ser nulo (NOT NULL) e tem de ser superior a zero (CHECK (valor > 0))

- tempoServico
 - idStaff é chave primária (PRIMARY KEY) e chave estrangeira (FOREIGN KEY)
 - idServico é chave primária (PRIMARY KEY) e chave estrangeira (FOREIGN KEY)
 - horasTrabalho não pode ser nulo (NOT NULL)
 - data não pode ser nulo (NOT NULL)

- Sala
 - numero é chave primária (PRIMARY KEY)
 - lotacao não pode ser nulo (NOT NULL)
 - tipologia não pode ser nulo e tem de ser uma entre concertos ou exposições (CHECK (tipologia = "concertos" OR tipologia = "exposicoes"))
 - dimensoes não pode ser nulo (NOT NULL) e tem de ser superior a zero (CHECK (dimensoes > 0))

- EventoCultural
 - id é chave primária (PRIMARY KEY)
 - dataInicio não pode ser nulo (NOT NULL)
 - dataFim não pode ser nulo (NOT NULL)
 - limiteldade não pode ser nulo (NOT NULL) e cada evento pode ter limiteldade ou não (CHECK (limiteldade = 0 OR limiteldade = 1))
 - dataFim tem de ser depois da data de início (CHECK (dataInicio < dataFim))

- Concerto
 - id é chave primária (PRIMARY KEY) e chave estrangeira (FOREIGN KEY)
 - tipolluminacao não pode ser nulo (NOT NULL) e tem de ser uma entre monocromática ou colorida (CHECK (tipolluminacao = "monocromatica" OR tipolluminacao = "colorida"))

- tipoSom não pode ser nulo e tem de ser um entre mesas de mistura ou instrumentos (CHECK (tipoSom = "mesasMistura" OR tipoSom = "instrumentos"))
- Exposicao
 - id é chave primária (PRIMARY KEY) e chave estrangeira (FOREIGN KEY)
 - projecoes não pode ser nulo (NOT NULL) e tem de ser maior ou superior a zero (CHECK (projecoes >= 0))
- Bilhete
 - numero é chave primária (PRIMARY KEY)
 - utilizado não pode ser nulo (NOT NULL) e tem de referir se o bilhete já foi utilizado ou não (CHECK (utilizado = 0 OR utilizado = 1))

Adicionalmente, às restrições do tipo FOREIGN KEY de ordenado e tempoServico foi adicionada a restrição no caso de eliminação da chave, para que esta seja considerada nula nas restantes referências (ON DELETE SET NULL), já no caso das FOREIGN KEYS de Espectador, Artista, Staff, Concerto e Exposicao é utilizada a restrição no caso de eliminação nos locais onde estas referências acontecem inicialmente (ON DELETE CASCADE). Em adição, foi adicionada, em todas as FOREIGN KEYS, a restrição para o caso de serem feitas alterações da chave, estas serem repercutidas onde são referidas (ON UPDATE CASCADE).

Existem algumas restrições que serão apenas referidas na próxima fase do projeto, uma vez que vão ser implementadas com a utilização de gatilhos:

- O bilhete apenas poderá ser adquirido no caso do espectador cumprir com o limite de idade imposto pelo evento associado;
- Não pode existir mais que um evento associado a uma sala ao mesmo tempo, isto é, a dataFim é sempre anterior à dataInicio do evento seguinte, se este existir;
- O valor do ordenado é calculado a partir da seguinte fórmula:

$$\text{Staff.renumeracaoHora} * \text{tempoServico.horasTrabalho} * \text{Servico.taxa, se mes (tempoServico.data) = mes.}$$