



Universidade do Porto

FEUP Faculdade de
Engenharia

Base de Dados

Campeonato Nacional de Hóquei em Patins

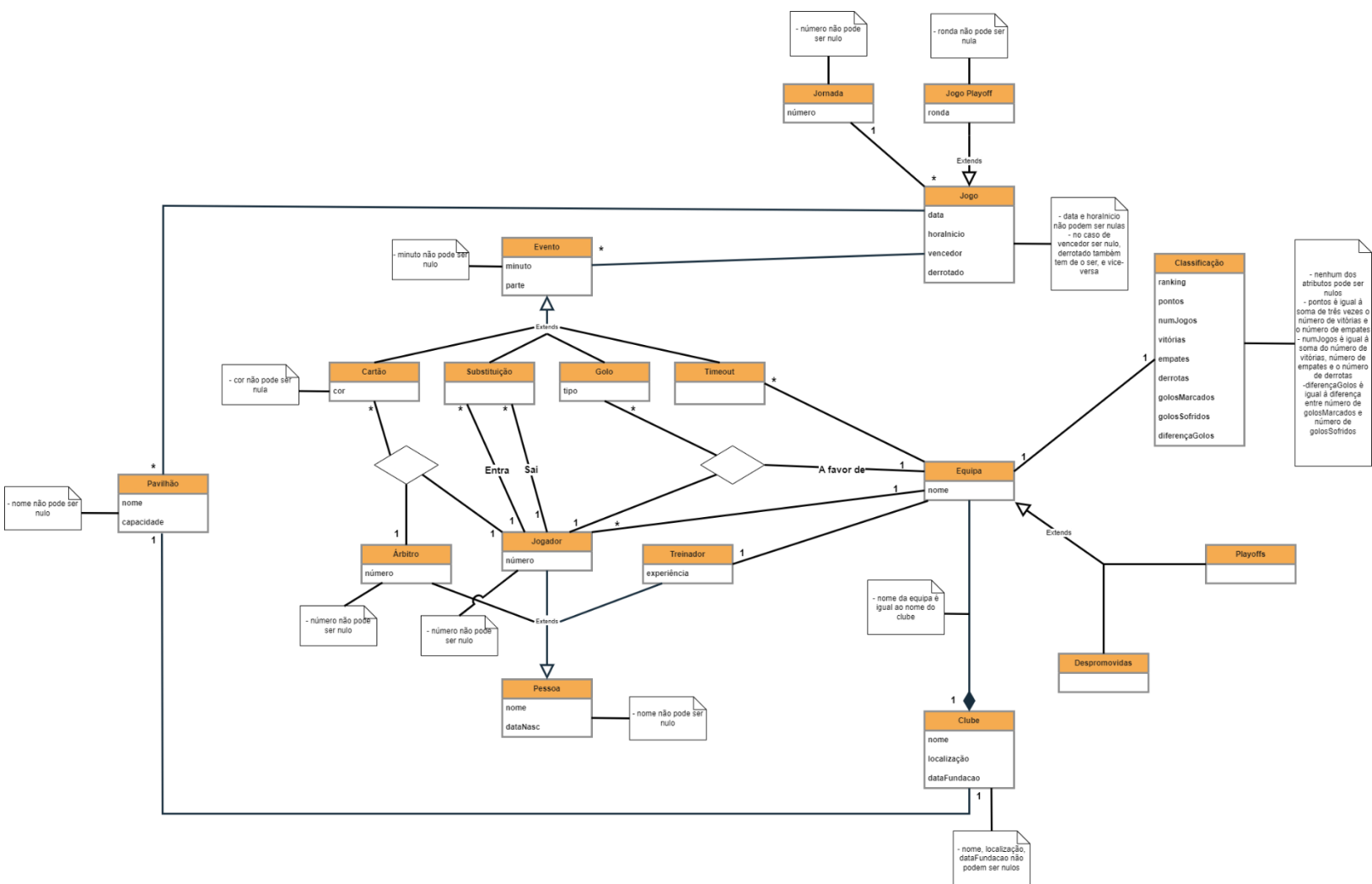
2LEIC02 - Grupo 302

David Cordeiro - up202108820

Diogo Viana - up2021088203

Gabriel Ferreira - up202108722

DIAGRAMA UML



Introdução

Este relatório visa esclarecer o raciocínio do grupo por detrás das decisões tomadas no desenho, criação e povoação de uma base de dados para gerir o campeonato nacional de hóquei em patins, no âmbito da realização de um projeto para a unidade curricular Bases de Dados. Para tal, é incluído, para além da UML da base de dados e o seu modelo relacional, uma descrição de algumas anotações importantes nas classes e associações utilizadas, uma análise das dependências funcionais das relações utilizadas e observações sobre possíveis violações à 3ª Forma Normal ou Forma Normal de Boyce-Codd.

Análise de Classes

Clube

Em primeiro lugar podemos observar a classe horário, que é usada para representar o clube a que pertencem as equipas que participam no campeonato. Um clube tem um único nome e localização (que no contexto da nossa base de dados são diferentes concelhos de Portugal) e ainda uma data de fundação. Um Clube, no contexto do campeonato nacional de hóquei, tem exatamente uma equipa participante e um pavilhão.

Pavilhão

A classe pavilhão foi criada para representar o nome e a capacidade, ou número de lugares disponíveis, dos locais onde decorrem os jogos. É de notar que no mesmo pavilhão é possível decorrerem vários jogos ao longo do campeonato, mas qualquer jogo acontece sempre em exatamente em um pavilhão.

Jogo e subclasses

Usamos a classe Jogo para representar as informações relevantes aos jogos do campeonato, como a data, a hora a que o jogo começa e o local onde decorre (representado pela sua associação com pavilhão). Num jogo participa uma equipa visitada e uma visitante. Os atributos vencedor e derrotado indicam o resultado de um jogo. Na eventualidade de um empate, ambos os valores ficam nulos. É também superclasse da classe Jogo Playoff, que é um jogo que decorre numa determinada ronda (quartos, meias ou finais) e onde as equipas que participam têm de se ter apurado para os playoffs.

Jornada

Uma classe muito simples que representa o número das 26 diferentes jornadas que constitui a fase regular do campeonato. Os jogos que decorrem numa determinada jornada são mostrados pela associação com Jogo.

Equipa e subclasses

Uma equipa é identificada por um nome, igual ao nome do respetivo clube. Cada equipa tem uma única classificação na fase regular. Uma equipa que é apurada para a fase de playoff é uma equipa cujo ranking da sua classificação na fase regular é menor ou igual a 8. Uma equipa despromovida é uma equipa cujo ranking da classificação na fase regular é igual ou superior a 12.

Pessoa e subclasses

Uma pessoa, no contexto da nossa base de dados, pode ser ou um árbitro, um jogador ou um treinador (generalização completa e disjunta). Os atributos comuns de uma pessoa são o nome e a data de nascimento. Um árbitro e um jogador têm um número próprio e um treinador é caracterizado pela sua experiência, ou seja, o número de anos como treinador de hóquei em patins.

Evento e subclasses

Um evento é algo que acontece num minuto de uma das 2 partes de um jogo. Pode ser um cartão, uma substituição, um golo, um timeout ou outra coisa não descrita pela uml (generalização disjunta incompleta). Um cartão é representado pela sua cor, um golo por um de 3 tipos diferentes (Penáti, livre ou diferente). Uma substituição é caracterizada pelas 2 associações com Jogador, uma para o jogador que entra e outra para o que sai. Um timeout é um período de tempo onde o jogo é interrompido a pedido de uma equipa.

Associações n-árias

São ainda utilizadas 2 associações n-árias na uml. Uma delas para representar a atribuição de um cartão por um árbitro a um jogador e outra para representar a marcação de um golo, que é sempre feita por um jogador, em favor da sua equipa, ou, da equipa contrária no caso de autogolo.

Esquema Relacional

Notas: Para as generalizações de Pessoa e Evento utilizou-se o estilo “Object Oriented” por serem disjuntas, sendo que a relação Pessoa foi omitida, dado que a sua generalização é completa.

- Clube (nome, localização, dataFundacao)
- Pavilhão (idPavilhão, nome, capacidade, nomeClube -> Clube)
- Equipa (idEquipa, nomeClube -> Clube, ranking -> Classificação)
- Playoffs (idEquipa, nomeClube -> Clube, ranking -> Classificação)
- Desqualificadas (idEquipa, nomeClube -> Clube, ranking -> Classificação)
- Classificação (ranking, pontos, numJogos, vitórias, empates, derrotas, golosMarcados, golosSofridos, diferençaGolos)
- Jogo (idJogo, data, horaInicio, vencedor, derrotado, idEquipaVisitada -> Equipa, idEquipaVisitante -> Equipa)

*Generalização estilo E/R

- JogoPlayoff (idJogo -> Jogo, ronda)
- Evento (idEvento, minuto, parte, idJogo -> Jogo)
- Cartão (idEvento, minuto, parte, cor, idJogo -> Jogo)
- Substituição (idEvento, minuto, parte, idJogadorSai -> Jogador, idJogadorEntra -> Jogador, idJogo -> Jogo)
- Golo (idEvento, minuto, parte, tipo, idJogo -> Jogo)
- Timeout (idEvento, minuto, parte, idEquipa -> Equipa, idJogo -> Jogo)
- Árbitro (idPessoa, nome, dataNasc, número)
- Treinador (idPessoa, nome, dataNasc, experiência, idEquipa -> Equipa)
- Jogador (idPessoa, nome, dataNasc, número, posição, idEquipa -> Equipa)
- Jornada (numero)

- JornadaJogo (idJornada -> Jornada, idJogo -> Jogo)
- PavilhãoJogo (idPavilhão -> Pavilhão, idJogo -> Jogo)
- MarcaGolo (idGolo -> Golo, idJogador -> Jogador, idEquipa -> Equipa)
- AtribuiCartão (idCartão -> Cartão, idÁrbitro -> Árbitro, idJogador -> Jogador)

Dependências Funcionais

- Clube (nome, localização, dataFundacao)
idClube -> nome, localização, dataFundacao
nome -> idClube, localização, dataFundacao
localização -> idClube, nome, dataFundacao
- Pavilhão (idPavilhão, nome, capacidade, idClube -> Clube)
idPavilhão -> nome, capacidade, idClube
nome -> idPavilhão, capacidade, idClube
idClube -> idPavilhão, nome, capacidade
- Equipa (idEquipa, nomeClube -> Clube, idClassificação -> Classificação)
idEquipa -> nomeClube, idClassificação
idClube -> idEquipa, idClassificação
- Playoffs (idEquipa, nomeClube -> Clube, idClassificação -> Classificação)
idEquipa -> nomeClube, idClassificação
idClube -> idEquipa, idClassificação
- Desqualificadas (idEquipa, nomeClube -> Clube, idClassificação -> Classificação)
idEquipa -> nomeClube, idClassificação
idClube -> idEquipa, idClassificação
- Classificação (ranking, pontos, numJogos, vitórias, empates, derrotas, golosMarcados, golosSofridos, diferençaGolos)
ranking -> pontos, numJogos, vitórias, empates, derrotas, golosMarcados, golosSofridos, diferençaGolos
- Jogo (idJogo, data, horaInicio, vencedor, derrotado, idEquipaVisitada -> Equipa, idEquipaVisitante -> Visitante)
idJogo -> data, horaInicio, vencido, derrotado, idEquipaVisitada, idEquipaVisitante.
{ data horaInicio vencedor derrotado } -> idJogo, idEquipaVisitada, idEquipaVisitante
{ data horaInicio idEquipaVisitada idEquipaVisitante } -> idJogo, vencedor, derrotado
- JogoPlayoff (idJogo, ronda)
idJogo -> ronda
- Evento (idEvento, minuto, parte, idJogo -> Jogo)
idEvento -> minuto, idJogo, parte

- Cartão (idEvento, minuto, parte, cor, idJogo -> Jogo)
idEvento -> minuto, cor, idJogo, parte
- Substituição (idEvento, minuto, parte, idJogadorSai -> Jogador, idJogadorEntra -> Jogador, idJogo -> Jogo)
idEvento -> minuto, parte, idJogadorSai, idJogadorEntra, idJogo
idJogo idJogadorSai idJogadorEntra -> minuto, parte, idEvento
- Golo (idEvento, minuto, parte, tipo, idJogo -> Jogo)
idEvento -> minuto, idJogo, parte, tipo
- Timeout (idEvento, minuto, parte, idEquipa -> Equipa, idJogo -> Jogo)
idEvento -> minuto, parte, idEquipa, idJogo
- Pessoa (idPessoa, nome, dataNasc)
idPessoa -> nome, dataNasc
- Árbitro (idPessoa, nome, dataNasc, número)
idPessoa -> nome, dataNasc, número
- Treinador (idPessoa, nome, dataNasc, experiência, idEquipa -> Equipa)
idPessoa -> nome, dataNasc, experiência, idEquipa
idEquipa -> idPessoa, nome, dataNasc, experiência
- Jogador (idPessoa, nome, dataNasc, número, posição, idEquipa -> Equipa)
idPessoa -> nome, dataNasc, número, posição, idEquipa
numero idEquipa -> nome, dataNasc, posição
- Jornada (numero)
- MarcaGolo (idGolo -> Golo, idJogador -> Jogador, idEquipa -> Equipa)
idGolo -> idJogador, idEquipa
- AtribuiCartão (idCartão -> Cartão, idÁrbitro -> Árbitro, idJogador -> Jogador)
idCartão -> idÁrbitro, idJogador
- JornadaJogo (idJornada -> Jornada, idJogo -> Jogo)
- PavilhãoJogo (idPavilhão -> Pavilhão, idJogo -> Jogo)

Violações á 3ª Forma Normal ou Forma Normal de Boyce-Codd

Como se pode observar, nenhuma das relações da base de dados representa uma violação á 3ª Forma Normal ou Forma Normal de Boyce-Codd.

Uma relação está em BCNF se, para todo $A \rightarrow B$ não trivial, A é uma superkey/key e em 3NF se, para todo $A \rightarrow B$ não trivial, A é uma superkey/key ou B consiste apenas em atributos primos (atributos que são um subconjunto próprio de pelo menos uma chave da relação)

Analisando todas as dependências funcionais nas relações descritas acima, verificamos que para cada uma delas conhecendo o conjunto de atributos no lado esquerdo, conseguimos conhecer todos os atributos da relação, verificando assim as definições.