

Base de Dados de uma Liga de Futebol

Entrega 3

Turma 1- Grupo 2

Pedro Miguel da Costa Azevedo - up201806728

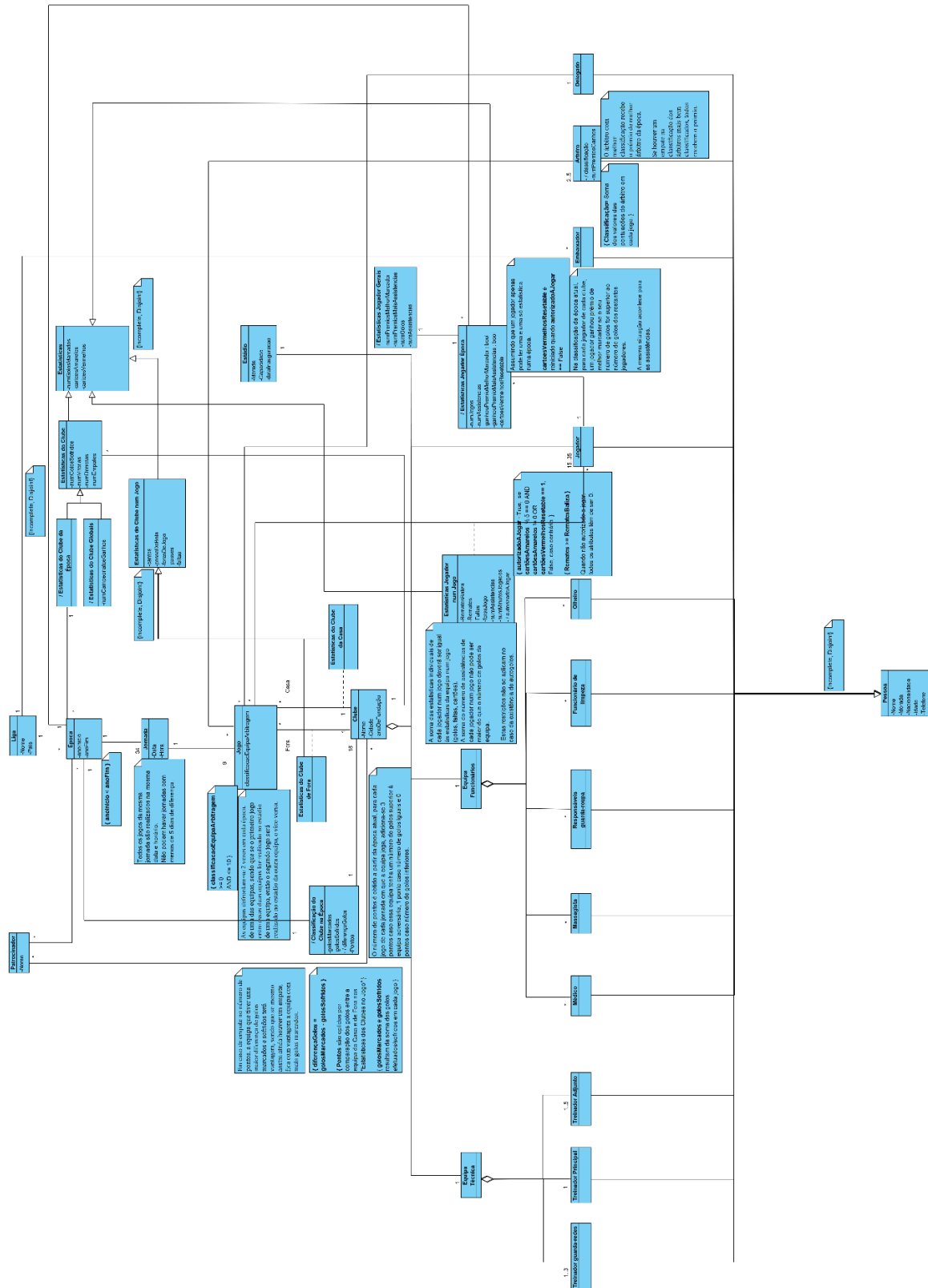
Rui Filipe Mendes Pinto - up201806441

Tiago Gonçalves Gomes – up201806658

Contexto da Base de Dados:

- A liga tem 18 equipas, vários embaixadores, épocas e patrocinadores.
- Cada equipa tem 1 equipa técnica, entre 15 e 35 jogadores (inclusive), 1 estádio, vários patrocinadores, e uma equipa de funcionários.
- Cada equipa técnica é composta por 1 treinador principal, de 1 a 5 treinadores adjuntos e de 1 a 3 treinadores de guarda-redes.
- A equipa de funcionários é composta por vários médicos, massagistas, responsáveis pelo guarda-roupa, funcionários de limpeza e olheiros.
- Para todas as pessoas, é necessário saber o nome completo, a morada, a nacionalidade, a idade e o número de telefone.
- Cada época tem 34 jornadas. Cada jornada tem 9 jogos, realizados na mesma data e hora; as equipas defrontam-se 2 vezes em cada época.
- Cada jogo tem 1 data e horário, 2 equipas, 1 equipa de arbitragem, é realizado em um estádio (de uma das equipas, sendo que se o primeiro jogo entre essas duas equipas for realizado no estádio de uma equipa, então o segundo jogo será realizado no estádio da outra equipa, e vice-versa), tem 1 delegado da liga responsável pela organização do jogo.
- Para cada jogo interessa também saber todas as estatísticas, de cada uma das equipas: golos marcados, posse de bola, número de passes efetuados, remates, remates à baliza, faltas, cartões amarelos e vermelhos, foras de jogo e cantos. Além disso, as seguintes estatísticas têm que estar associadas a um jogador específico: golos, número de assistências, faltas, cartões amarelos e vermelhos. É também atribuída uma classificação de 0 a 10 à atuação da equipa de arbitragem.
- Cada jogador da liga tem associado a si estatísticas, que serão atualizadas a cada jornada: número de jogos, número de golos marcados (não se contam os autogolos), número de cartões amarelos, número de cartões vermelhos, número de faltas, número de assistências. Se o jogador tiver 1 cartão vermelho ou 5 cartões amarelos, fica impedido de jogar na próxima jornada, sendo que depois esse número de cartões é reinicializado.
- Cada jogo é arbitrado por 2 a 5 árbitros. Para cada árbitro interessa saber o nome completo, a morada e o telefone, bem como a sua classificação, calculada pela soma de todas as classificações dos jogos em que participou.
- Existe uma classificação da liga, que é atualizada a cada jornada. Uma vitória equivale a três pontos, um empate a 1 ponto e uma derrota a 0 pontos. Em caso de empate no número de pontos, a equipa que tiver uma maior diferença de golos marcados e sofridos terá vantagem, sendo que se mesmo assim ainda houver um empate, fica com vantagem a equipa com mais golos marcados.
- No fim da época, a equipa que ficou em primeiro lugar sagra-se campeã, e o árbitro com melhor classificação recebe o prémio de melhor árbitro da época. Se houver um empate na classificação dos árbitros mais bem classificados, todos recebem o prémio. É também atribuído o prémio de melhor(es) marcador(es) e de jogador(es) com mais assistências da época.
- Interessa guardar a informação de quantos campeonatos foram ganhos por cada equipa, quantos prémios de melhor árbitro foram ganhos por cada árbitro, quantos prémios de melhor marcador e de jogador com mais assistências foram ganhos por cada jogador.

Diagrama UML:



Nota: O diagrama UML encontra-se à parte, com melhor qualidade.

4

Definição do Esquema Relacional:

- **Pessoa**(idPessoa, nome, morada, nacionalidade, idade, telefone)
- **EquipaTecnica**(idEquipaTecnica, idClube->Clube)
- **TreinadorGuardaRedes**(idPessoa->Pessoa, idEquipaTecnica->EquipaTecnica, nome, morada, nacionalidade, idade, telefone)
- **TreinadorPrincipal**(idPessoa->Pessoa, idEquipaTecnica->EquipaTecnica, nome, morada, nacionalidade, idade, telefone)
- **TreinadorAdjunto**(idPessoa->Pessoa, idEquipaTecnica->EquipaTecnica, nome, morada, nacionalidade, idade, telefone)
- **EquipaFuncionarios**(idEquipaFuncionarios, idClube->Clube)
- **Medico**(idPessoa->Pessoa, idEquipaFuncionarios->EquipaFuncionarios, nome, morada, nacionalidade, idade, telefone)
- **Massagista**(idPessoa->Pessoa, idEquipaFuncionarios->EquipaFuncionarios, nome, morada, nacionalidade, idade, telefone)
- **ResponsavelGuardaRoupa**(idPessoa->Pessoa, idEquipaFuncionarios->EquipaFuncionarios, nome, morada, nacionalidade, idade, telefone)
- **FuncionarioDeLimpeza**(idPessoa->Pessoa, idEquipaFuncionarios->EquipaFuncionarios, nome, morada, nacionalidade, idade, telefone)
- **Olheiro**(idPessoa->Pessoa, idEquipaFuncionarios->EquipaFuncionarios, nome, morada, nacionalidade, idade, telefone)
- **Jogador**(idPessoa->Pessoa, idClube->Clube, nome, morada, nacionalidade, idade, telefone)
- **EstatisticaJogadorNumJogo**(idJogador->Jogador, idJogo->Jogo, numMinutosJogados)
- **EstatisticaJogadorEpoca**(idEstatisticaJogadorEpoca, numJogos, numAssistencias, numGolos, ganhouPremioMelhorMarcador, ganhouPremioMelhorAssistencias, cartoesVermelhosResetable, epoca->Epoca, idJogador->Jogador)
- **Clube**(idClube, nome, cidade, anoDeFundacao, idClassificacao->ClassificacaoDoClubeNaEpoca)
- **PatrocinioDoClube**(idClube->Clube, idPatrocinador->Patrocinador)
- **Estadio**(morada, capacidade, dataInauguracao, numLugaresVIP, idClube->Clube)

- **ClassificacaoDoClubeNaEpoca**(idClassificacao, golosMarcados, golosSofridos, diferencaGolos, pontos, numVitorias, numDerrotas, numEmpates, epoca->Epoca)
- **Liga**(nome, país)
- **Embaixador**(idPessoa->Pessoa, nomeLiga->Liga.nome, paisLiga->Liga.pais)
- **Delegado**(idPessoa->Pessoa)
- **Arbitro**(idPessoa->Pessoa, classificacao, numPremiosGanhos, nome, morada, nacionalidade, idade, telefone)
- **ArbitroJogo**(idArbitro->Arbitro, idJogo->Jogo)
- **Patrocinador**(idPatrocinador, nome, rank, valorInvestido)
- **PatrocinadorEpoca**(idPatrocinador->Patrocinador, epoca->Epoca)
- **PatrocinadorClube**(idPatrocinador->Patrocinador, idClube->Clube)
- **Epoca**(anoInicio, anoFim, nomeLiga->Liga.nome, paisLiga->Liga.pais)
- **Jornada**(idJornada, epoca->Epoca)
- **Jogo**(idJogo, data_e_hora, classificacaoEquipaArbitragem, idJornada->Jornada, idDelegado->Delegado, idClubeCasa->Clube, idClubeFora->Clube, terminadoPara)
- **EstatísticasClubeJogo**(idEstatisticaClubeJogo, idClube->Clube, numGolos, numCartoesAmarelos, numCartoesVermelhos, numFaltas, numRemates, numRematesBaliza, numCantos, numForasDeJogo, numAssistencias, idJogo->Jogo)
- **EventoJogo**(idEvento, minuto, idJogo->Jogo)
- **Golo**(idEvento>Evento, idJogador->Jogador, minuto, idJogo->Jogo)
- **Cartao**(idEvento>Evento, cor, idJogador->Jogador, minuto, idJogo->Jogo)
- **Falta**(idEvento>Evento, idJogador->Jogador, minuto, idJogo->Jogo)
- **Remate**(idEvento>Evento, naBaliza, idJogador->Jogador, minuto, idJogo->Jogo)
- **Canto**(idEvento>Evento, idJogador->Jogador, idClube->Clube, minuto, idJogo->Jogo)
- **ForaDeJogo**(idEvento>Evento, idJogador->Jogador, idClube->Clube, minuto, idJogo->Jogo)
- **Assistencia**(idEvento>Evento, idJogador->Jogador, minuto, idJogo->Jogo)

Análise de Dependências Funcionais e Formas Normais:

Pessoa(idPessoa, nome, morada, nacionalidade, idade, telefone) :

- idPessoa -> nome, morada, nacionalidade, idade, telefone
- telefone -> idPessoa, nome, morada, nacionalidade, idade

BCNF: sim

3NF: sim

EquipaTecnica(idEquipaTecnica, idClube->Clube)

- idEquipaTecnica -> idClube

BCNF: sim

3NF: sim

TreinadorGuardaRedes(idPessoa->Pessoa, nome, morada, nacionalidade, idade, telefone, idEquipaTecnica->EquipaTecnica)

- idPessoa -> nome, morada, nacionalidade, idade, telefone, idEquipaTecnica
- telefone -> idPessoa, nome, morada, nacionalidade, idade, idEquipaTecnica

BCNF: sim

3NF: sim

TreinadorPrincipal(idPessoa->Pessoa, nome, morada, nacionalidade, idade, telefone, idEquipaTecnica->EquipaTecnica)

- idPessoa -> nome, morada, nacionalidade, idade, telefone, idEquipaTecnica
- telefone -> idPessoa, nome, morada, nacionalidade, idade, idEquipaTecnica

BCNF: sim

3NF: sim

TreinadorAdjunto(idPessoa->Pessoa, nome, morada, nacionalidade, idade, telefone, idEquipaTecnica->EquipaTecnica)

- idPessoa -> nome, morada, nacionalidade, idade, telefone, idEquipaTecnica
- telefone -> idPessoa, nome, morada, nacionalidade, idade, idEquipaTecnica

BCNF: sim

3NF: sim

EquipaFuncionarios(idEquipaFuncionarios, idClube->Clube)

- idEquipaFuncionarios -> idClube

BCNF: sim

3NF: sim

Medico(idPessoa->Pessoa, nome, morada, nacionalidade, idade, telefone,
idEquipaFuncionarios->EquipaFuncionarios)

- idPessoa-> nome, morada, nacionalidade, idade, telefone, idEquipaFuncionarios
- telefone -> idPessoa, nome, morada, nacionalidade, idade, idEquipaFuncionarios

BCNF: sim

3NF: sim

Massagista(idPessoa->Pessoa, nome, morada, nacionalidade, idade, telefone,
idEquipaFuncionarios->EquipaFuncionarios)

- idPessoa-> nome, morada, nacionalidade, idade, telefone, idEquipaFuncionarios
- telefone -> idPessoa, nome, morada, nacionalidade, idade, idEquipaFuncionarios

BCNF: sim

3NF: sim

ResponsavelGuardaRoupa(idPessoa->Pessoa, nome, morada, nacionalidade, idade,
telefone, idEquipaFuncionarios->EquipaFuncionarios)

- idPessoa-> nome, morada, nacionalidade, idade, telefone, idEquipaFuncionarios
- telefone -> idPessoa, nome, morada, nacionalidade, idade, idEquipaFuncionarios

BCNF: sim

3NF: sim

FuncionarioDeLimpeza(idPessoa->Pessoa, nome, morada, nacionalidade, idade, telefone,
idEquipaFuncionarios->EquipaFuncionarios)

- idPessoa-> nome, morada, nacionalidade, idade, telefone, idEquipaFuncionarios
- telefone -> idPessoa, nome, morada, nacionalidade, idade, idEquipaFuncionarios

BCNF: sim

3NF: sim

Olheiro(idPessoa->Pessoa, nome, morada, nacionalidade, idade, telefone,
idEquipaFuncionarios->EquipaFuncionarios)

- idPessoa-> nome, morada, nacionalidade, idade, telefone, idEquipaFuncionarios
- telefone -> idPessoa, nome, morada, nacionalidade, idade, idEquipaFuncionarios

BCNF: sim

3NF: sim

Jogador(idPessoa->Pessoa, nome, morada, nacionalidade, idade, telefone, idClube->Clube)

- idPessoa -> nome, morada, nacionalidade, idade, telefone, idClube
- telefone -> idPessoa, nome, morada, nacionalidade, idade, idClube

BCNF: sim

3NF: sim

EstatisticaJogadorNumJogo(idJogador->Jogador, idJogo->Jogo, numMinutosJogados)

- idJogador, idJogo -> numMinutosJogados, autorizadoAJogar

BCNF: sim

3NF: sim

EstatisticaJogadorEpoca(idEstatisticaJogadorEpoca, numJogos, numAssistencias,
numGolos, ganhouPremioMelhorMarcador, ganhouPremioMelhorAssistencias,
cartoesVermelhosResetable, epoca->Epoca, idJogador->Jogador)

- idEstatisticaJogadorEpoca -> umJogos, numAssistencias, numGolos,
ganhouPremioMelhorMarcador, ganhouPremioMelhorAssistencias,
cartoesVermelhosResetable, epoca, idJogador

BCNF: sim

3NF: sim

Clube(idClube, nome, cidade, anoDeFundacao,
idClassificacao->ClassificacaoDoClubeNaEpoca)

- idClube -> nome, cidade, anoDeFundacao, idClassificacao

BCNF: sim

3NF: sim

PatrocinioDoClube(idClube->Clube, idPatrocinador->Patrocinador)

- --

BCNF: sim

3NF: sim

Estadio(morada, capacidade, numLugaresVIP, dataInauguracao, idClube->Clube)

- morada -> capacidade, dataInauguracao, idClube
- capacidade -> numLugaresVIP

BCNF: não

3NF: não

Justificação:

- A capacidade não é uma (super)key, logo não respeita a BCNF.
- A capacidade não é uma (super)key e o numLugaresVIP não é um *prime attribute* (não está contido na chave), logo não respeita a 3NF.

Decomposição para BCNF:

1. Key: {morada}
 - a. Violação BCNF: capacidade-> numLugaresVIP
2. EstadioCapacidadeVIP(capacidade, numLugaresVIP)
3. Estadio'(morada, capacidade, dataInauguracao, idClube)
 - a. Analisar *FDs* e *keys* para EstadioCapacidadeVIP:
 - i. capacidade -> numLugaresVIP
 - ii. Key: {capacidade}
 - iii. EstadioCapacidade está na BCNF.
 - b. Analisar *FDs* e *keys* para Estadio':
 - i. morada-> capacidade, dataInauguracao, idClube
 - ii. Key: {morada}
 - iii. Estadio' está na BCNF.

Decomposição para 3NF:

1. *Minimal Basis*
 - a. Lado direito com mais de um atributo. Logo ficamos com:
 - i. morada -> capacidade; morada -> dataInauguracao; morada -> idClube
 - ii. morada -> numLugaresVIP
 - b. A *FD* morada -> numLugaresVIP não é essencial, pois partindo de morada-> capacidade e capacidade -> numLugaresVIP, conseguimos chegar de facto ao lado direito da *FD*. Todas as restantes envolvendo a morada são essenciais.
 - c. {capacidade}⁺ = {capacidade}, não contém numLugaresVIP, logo é essencial.
 - d. Lado esquerdo com apenas um atributo.

2. Para cada $FD X \rightarrow A$ em Estadio, criar uma relação Estadio'' (X, A)
 - a. Estadio' (morada, capacidade, dataInauguracao, idClube)
 - b. EstadioCapacidadeVIP (capacidade, numLugaresVIP)
3. Se nenhuma das relações no 2º passo é uma superkey de Estadio, então adiciona-se outra relação para a chave de Estadio. Não é o caso (Estadio'), portanto o resultado é o do tópico 2.

A conclusão tirada desta decomposição resulta na criação das tabelas: Estadio(morada, capacidade, dataInauguracao, idClube) e EstadioCapacidadeVIP(capacidade, numLugaresVIP).

ClassificacaoDoClubeNaEpoca(idClassificacao, golosMarcados, golosSofridos, diferencaGolos, pontos, numVitorias, numDerrotas, numEmpates, epoca->Epoca)

- idClassificacao -> golosMarcados, golosSofridos, diferencaGolos, pontos, numVitorias, numDerrotas, numEmpates, epoca

BCNF: sim

3NF: sim

Liga(nome, país)

- --

BCNF: sim

3NF: sim

Embaixador(idPessoa->Pessoa, nome, morada, nacionalidade, idade, telefone, nomeLiga->Liga.nome, paisLiga->Liga.pais)

- idPessoa -> nome, morada, nacionalidade, idade, telefone, nomeLiga, paisLiga
- telefone -> idPessoa, nome, morada, nacionalidade, idade, telefone, nomeLiga, paisLiga

BCNF: sim

3NF: sim

Delegado(idPessoa->Pessoa, nome, morada, nacionalidade, idade, telefone)

- idPessoa -> nome, morada, nacionalidade, idade, telefone
- telefone -> idPessoa, nome, morada, nacionalidade, idade

BCNF: sim

3NF: sim

Arbitro(idPessoa->Pessoa, nome, morada, nacionalidade, idade, telefone, classificacao, numPremiosGanhos)

- idPessoa -> nome, morada, nacionalidade, idade, telefone, classificacao, numPremiosGanhos
- telefone -> idPessoa, nome, morada, nacionalidade, idade

BCNF: sim

3NF: sim

ArbitroJogo(idArbitro->Arbitro, idJogo->Jogo)

- --

BCNF: sim

3NF: sim

Patrocinador(idPatrocinador, nome, rank, valorInvestido)

- idPatrocinador -> nome, rank, valorInvestido
- rank -> valorInvestido

BCNF: não

3NF: não

Justificação:

- O rank não é uma (super)key, logo não respeita a BCNF.
- O rank não é uma (super)key e o valorInvestido não é um prime attribute (não está contido na chave), logo não respeita a 3NF.

Decomposição para BCNF:

1. Key: {idPatrocinador}
 - a. Violação BCNF: rank -> valorInvestido
2. Rank(rank, valorInvestido)
3. Patrocinador'(idPatrocinador, nome, rank)
 - a. Analisar *FDs* e *keys* para Rank:
 - i. rank -> valorInvestido
 - ii. Key: {rank}
 - iii. Rank está na BCNF.
 - b. Analisar *FDs* e *keys* para Patrocinador':
 - i. idPatrocinador -> nome, rank
 - ii. Key: {idPatrocinador}
 - iii. Patrocinador' está na BCNF.

Decomposição para 3NF:

1. *Minimal Basis*

- a. Lado direito com mais de um atributo. Logo ficamos com:
 - i. $\text{idPatrocinador} \rightarrow \text{nome} ; \text{idPatrocinador} \rightarrow \text{rank}$
 - ii. $\text{idPatrocinador} \rightarrow \text{valorInvestido}$
 - b. A $FD \text{idPatrocinador} \rightarrow \text{valorInvestido}$ não é essencial, pois partindo de $\text{idPatrocinador} \rightarrow \text{rank}$ e $\text{rank} \rightarrow \text{valorInvestido}$, conseguimos chegar de facto ao lado direito da FD . Todas as restantes envolvendo o idPatrocinador são essenciais.
 - c. $\{\text{rank}\}^+ = \{\text{rank}\}$, não contém valorInvestido , logo é essencial.
 - d. Lado esquerdo com apenas um atributo.
2. Para cada $FD X \rightarrow A$ em Patrocinador , criar uma relação $\text{Patrocinador}' (X, A)$
 - a. $\text{Patrocinador}' (\text{idPatrocinador}, \text{nome}, \text{rank})$
 - b. $\text{Rank} (\text{rank}, \text{valorInvestido})$
 3. Se nenhuma das relações no 2º passo é uma superkey de Patrocinador , então adiciona-se outra relação para a chave de Patrocinador . Não é o caso ($\text{Patrocinador}'$), portanto o resultado é o do tópico 2.

A conclusão tirada desta decomposição resulta na criação das tabelas $\text{Patrocinador}(\text{idPatrocinador}, \text{nome}, \text{rank})$ e $\text{Rank} (\text{rank}, \text{valorInvestido})$.

Epoca(anoInicio, anoFim, nomeLiga \rightarrow Liga.nome, paisLiga \rightarrow Liga.pais)

- anoInicio \rightarrow anoFim, nomeLiga, paisLiga

BCNF: sim

3NF: sim

PatrocinadorEpoca(idPatrocinador \rightarrow Patrocinador, epoca \rightarrow Epoca)

- --

BCNF: sim

3NF: sim

PatrocinadorClube(idPatrocinador \rightarrow Patrocinador, idClube \rightarrow Clube)

- --

BCNF: sim

3NF: sim

Jornada(idJornada, epoca->Epoca)

- idJornada -> epoca

BCNF: sim

3NF: sim

Jogo(idJogo, data_e_hora, classificacaoEquipaArbitragem, idJornada->Jornada, idDelegado->Delegado, idClubeCasa->Clube, idClubeFora->Clube, terminadoPara)

- idJogo -> data_e_hora, classificacaoEquipaArbitragem, idJornada, idDelegado, idClubeCasa, idClubeFora, terminadoPara

BCNF: sim

3NF: sim

EstatísticasClubeJogo(idEstatisticaClubeJogo, idClube->Clube, numGolos, numCartoesAmarelos, numCartoesVermelhos, numFaltas, numRemates, numRematesBaliza, numCantos, numForasDeJogo, numAssistencias, idJogo->Jogo)

- idJogo -> idEstatisticaClubeJogo, idClube, numGolos, numCartoesAmarelos, numCartoesVermelhos, numFaltas, numRemates, numRematesBaliza, numCantos, numForasDeJogo, numAssistencias, idJogo

BCNF: sim

3NF: sim

EventoJogo(idEvento, minuto, idJogo->Jogo)

- idEvento -> minuto, idJogo

BCNF: sim

3NF: sim

Golo(idEvento>Evento, idJogador->Jogador, minuto, idJogo->Jogo)

- idEvento -> idJogador

BCNF: sim

3NF: sim

Cartao(idEvento>Evento, cor, idJogador->Jogador, minuto, idJogo->Jogo)

- idEvento -> cor, idJogador

BCNF: sim

3NF: sim

Falta(idEvento→Evento, idJogador→Jogador, minuto, idJogo→Jogo)

- idEvento → idJogador

BCNF: sim

3NF: sim

Remate(idEvento→Evento, naBaliza, idJogador→Jogador, minuto, idJogo→Jogo)

- idEvento → naBaliza, idJogador

BCNF: sim

3NF: sim

Canto(idEvento→Evento, idJogador→Jogador, idClube→Clube, minuto, idJogo→Jogo)

- idEvento → idJogador, idClube

BCNF: sim

3NF: sim

ForaDeJogo(idEvento→Evento, idJogador→Jogador, idClube→Clube, minuto, idJogo→Jogo)

- idEvento → idJogador, idClube

BCNF: sim

3NF: sim

Assistencia(idEvento→Evento, idJogador→Jogador, minuto, idJogo→Jogo)

- idEvento → idJogador

BCNF: sim

3NF: sim

Justificação de todas as relações que seguem a Boyce-Codd Normal Form (BCNF) e a 3ª Forma Normal (3NF):

– Uma relação está na BCNF se, para todo $A \rightarrow B$ não trivial, A é *superkey/key*;

Uma relação está na 3ª Forma Normal se, para todo $A \rightarrow B$ não trivial, A é *superkey/key* ou B é composto apenas por atributos *prime* (membros de pelo menos uma chave da relação). Para todas as *Functional Dependencies (FDs)*, a partir do lado esquerdo de cada *FD*, conseguimos estabelecer uma relação com todos os atributos da relação, o que implica que o lado esquerdo é sempre uma (*super*)*key*. Assim, fica provado que as relações indicadas estão na BCNF e 3NF.

Restrições:

Pessoa:

- Não podem existir duas pessoas com o mesmo ID
 - idPessoa PRIMARY KEY
- Não podem existir telefones repetidos
 - telefone UNIQUE
- Os restantes atributos devem ter sempre um valor atribuído
 - nome NOT NULL
 - morada NOT NULL
 - nacionalidade NOT NULL
 - idade NOT NULL

EquipaTecnica:

- Não podem existir duas equipas técnicas com o mesmo ID
 - idEquipaTecnica PRIMARY KEY
- O ID do clube deverá corresponder ao de um clube na tabela Clube
 - idClube REFERENCES Clube

TreinadorGuardaRedes:

- Não podem existir dois treinadores com o mesmo ID; este ID deverá corresponder a uma pessoa da tabela Pessoa
 - idPessoa PRIMARY KEY REFERENCES Pessoa
- Não podem existir dois treinadores com o mesmo ID de EquipaTécnica; este ID deverá corresponder ao da tabela EquipaTecnica
 - idEquipaTecnica REFERENCES EquipaTecnica

TreinadorPrincipal:

- Não podem existir dois treinadores com o mesmo ID; este ID deverá corresponder a uma pessoa da tabela Pessoa
 - idPessoa PRIMARY KEY REFERENCES Pessoa
- Não podem existir dois treinadores com o mesmo ID de EquipaTécnica; este ID deverá corresponder ao da tabela EquipaTecnica
 - idEquipaTecnica REFERENCES EquipaTecnica

TreinadorAdjunto:

- Não podem existir dois treinadores com o mesmo ID; este ID deverá corresponder a uma pessoa da tabela Pessoa
 - idPessoa PRIMARY KEY REFERENCES Pessoa
- Não podem existir dois treinadores com o mesmo ID de EquipaTécnica; este ID deverá corresponder ao da tabela EquipaTecnica
 - idEquipaTecnica REFERENCES EquipaTecnica

EquipaFuncionarios:

- Não podem existir duas equipas de funcionários com o mesmo ID
 - idEquipaFuncionarios PRIMARY KEY
- O ID do clube deverá corresponder ao de um clube na tabela Clube
 - idClube REFERENCES Clube

Medico:

- Não podem existir dois médicos com o mesmo ID; este deverá corresponder a uma pessoa da tabela Pessoa
 - idPessoa PRIMARY KEY REFERENCES Pessoa
- Não podem existir dois médicos com o mesmo ID de EquipaFuncionarios; este ID deverá corresponder ao da tabela EquipaFuncionarios
 - idEquipaFuncionarios REFERENCES EquipaFuncionarios

Massagista:

- Não podem existir dois massagistas com o mesmo ID; este deverá corresponder a uma pessoa da tabela Pessoa
 - idPessoa PRIMARY KEY REFERENCES Pessoa
- Não podem existir dois massagistas com o mesmo ID de EquipaFuncionarios; este ID deverá corresponder ao da tabela EquipaFuncionarios
 - idEquipaFuncionarios REFERENCES EquipaFuncionarios

ResponsavelGuardaRoupa:

- Não podem existir dois responsáveis por guarda roupa com o mesmo ID; este deverá corresponder a uma pessoa da tabela Pessoa
 - idPessoa PRIMARY KEY REFERENCES Pessoa
- Não podem existir dois responsáveis por guarda roupa com o mesmo ID de EquipaFuncionarios; este ID deverá corresponder ao da tabela EquipaFuncionarios
 - idEquipaFuncionarios REFERENCES EquipaFuncionarios

FuncionarioDeLimpeza:

- Não podem existir dois funcionários de limpeza com o mesmo ID; este deverá corresponder a uma pessoa da tabela Pessoa
 - idPessoa PRIMARY KEY REFERENCES Pessoa
- Não podem existir dois funcionários de limpeza com o mesmo ID de EquipaFuncionarios; este ID deverá corresponder ao da tabela EquipaFuncionarios
 - idEquipaFuncionarios REFERENCES EquipaFuncionarios

Olheiro:

- Não podem existir dois olheiros com o mesmo ID; este deverá corresponder a uma pessoa da tabela Pessoa
 - idPessoa PRIMARY KEY REFERENCES Pessoa
- Não podem existir dois olheiros com o mesmo ID de EquipaFuncionarios; este ID deverá corresponder ao da tabela EquipaFuncionarios
 - idEquipaFuncionarios REFERENCES EquipaFuncionarios

Jogador:

- Não podem existir dois jogadores com o mesmo ID; este deverá corresponder a uma pessoa da tabela Pessoa
 - idPessoa PRIMARY KEY REFERENCES Pessoa
- O ID do clube deverá corresponder ao de um clube na tabela Clube
 - idClube REFERENCES Clube
- Quanto ao número de minutos jogados, o mapeamento é realizado da seguinte forma
 - numMinutosJogados NOT NULL

EstatisticaJogadorNumJogo:

- Não deve haver duas instâncias com o mesmo par (idJogador, idJogo)
 - PRIMARY KEY (idJogador, idJogo)
- Relativamente ao número de minutos jogados temos ainda seguinte restrição
 - CHECK (numMinutosJogados >= 0)

EstatisticaJogadorEpoca:

- Não podem existir duas estatísticas com o mesmo ID
 - idEstatisticaJogadorEpoca PRIMARY KEY
- O ID da época e do jogador associados à estatística deve ser mapeado respetivamente da seguinte forma (estando associados à tabela Época e Jogador)
 - epoca REFERENCES Epoca
 - idJogador REFERENCES Jogador
- Os restantes atributos devem ter sempre um valor atribuído
 - numJogos DEFAULT 0
 - numAssistencias DEFAULT 0
 - numGolos DEFAULT 0
 - ganhouPremioMelhorMarcador DEFAULT '0'
 - ganhouPremioMelhorAssistencias DEFAULT '0'
 - cartoesVermelhosResetable DEFAULT 0
- Para além disso foram definidas as seguintes *constraints*
 - CHECK (numAssistencias >= 0 AND numGolos >= 0 AND (ganhouPremioMelhorMarcador LIKE '0' OR ganhouPremioMelhorMarcador LIKE '1') AND (ganhouPremioMelhorAssistencias LIKE '0' OR ganhouPremioMelhorAssistencias LIKE '1') AND (cartoesVermelhosResetable LIKE '0' OR cartoesVermelhosResetable LIKE '1'))

Clube:

- Não podem existir dois clubes com o mesmo ID
 - idClube PRIMARY KEY
- O ID da classificação do clube na tabela classificativa está referenciado da seguintes forma
 - idClassificacao REFERENCES ClassificacaoDoClubeNaEpoca
- Os restantes atributos devem ter um valor atribuído
 - nome NOT NULL
 - cidade NOT NULL
 - anoDeFundacao CHECK (anoDeFundacao >= 0)

Estadio:

- Não pode ter a mesma morada
 - morada PRIMARY KEY

- O ID do clube deverá corresponder ao de um clube na tabela Clube
 - idClube REFERENCES Clube

EstadioCapacidadeVIP:

- A capacidade é então a chave primária: capacidade PRIMARY KEY
- A capacidade, além de ser maior ou igual a zero, determina o número de lugares VIP, sendo que o número de lugares VIP é igual à capacidade a dividir por 100 (divisão inteira)
 - capacidade NOT NULL CHECK (capacidade >= 0)
 - numLugaresVIP CHECK (numLugaresVIP = capacidade / 100)

Patrocinador:

- Não podem existir dois patrocinadores com o mesmo ID
 - idPatrocinador PRIMARY KEY
- Um patrocinador tem que ter um nome
 - nome NOT NULL

Rank:

- O rank é então a chave primária: rank PRIMARY KEY
- O rank pode ser bronze, prata ou ouro, e definem o valorInvestido, sendo que se o rank for bronze, o valor investido é de 250.000 euros, se o rank for de prata, o valor investido é de 500.000 euros e se o rank for ouro, o valor investido é de 1.000.000 euros.
 - rank CHECK (rank LIKE 'bronze' OR rank LIKE 'prata' OR rank LIKE 'ouro')
 - rank CHECK ((rank LIKE 'bronze' AND valorInvestido = 250000) OR (rank LIKE 'prata' AND valorInvestido = 500000) OR (rank LIKE 'ouro' AND valorInvestido = 1000000))

PatrocinadorEpoca:

- Não podem haver duas instâncias com o par (idPatrocinador, epoca)
 - PRIMARY KEY (idPatrocinador, epoca)
- O idPatrocinador deve de corresponder a um ID da tabela Patrocinador
 - idPatrocinador REFERENCES Patrocinador(idPatrocinador)
- A epoca deve de corresponder a uma data de início da tabela Epoca
 - epoca REFERENCES Epoca(epoca)

PatrocinadorClube:

- Não podem haver duas instâncias com o par (idPatrocinador, idClube)
 - PRIMARY KEY (idPatrocinador, idClube)
- O idPatrocinador deve de corresponder a um ID da tabela Patrocinador
 - idPatrocinador REFERENCES Patrocinador(idPatrocinador)
- O idClube deve de corresponder a um ID da tabela Clube
 - idClube REFERENCES Clube(idClube)

PatrocinioDoClube:

- Não deve haver duas instâncias com o mesmo par (idClube, idPatrocinador)
 - PRIMARY KEY (idJogador, idJogo)

ClassificacaoDoClubeNaEpoca:

- Não podem existir duas classificações com o mesmo ID
 - idClassificacao PRIMARY KEY
- O número de pontos é igual a 3 vezes o número de vitórias mais o número de empates
 - pontos CHECK (pontos = 3 * numVitorias + numEmpates)
- Relativamente à diferença de golos de cada equipa, a restrição é estabelecida da seguinte forma
 - CHECK(diferencaGolos = (golosMarcados – golosSofridos))
- Os restantes atributos devem ter sempre um valor atribuído
 - golosMarcados DEFAULT 0
 - golosSofridos DEFAULT 0
 - diferencaGolos DEFAULT 0
 - pontos DEFAULT 0
 - numVitorias DEFAULT 0
 - numDerrotas DEFAULT 0
 - numEmpates DEFAULT 0

Liga:

- Não podem haver ligas com o mesmo nome e país
 - PRIMARY KEY(nome, país)

ArbitroJogo:

- Não podem haver duas instâncias com o par (idArbitro, idJogo)
 - PRIMARY KEY (idArbitro, idJogo)

- O idArbitro deve de corresponder a um ID da tabela Pessoa
 - idArbitro REFERENCES Pessoa(idPessoa)
- O idJogo deve de corresponder a um ID da tabela Jogo
 - idJogo REFERENCES Jogo(idJogo)

Arbitro:

- A classificação tem que ser maior ou igual a zero
 - classificacao DEFAULT 0 CHECK (classificacao >= 0)
- O número de prémios ganhos tem que ser maior ou igual a zero
 - numPremiosGanhos DEFAULT 0 CHECK (numPremiosGanhos >= 0)

Epoca:

- Só existe uma época por cada ano de início
 - anoInicio PRIMARY KEY
- Só existe uma época por cada ano de fim
 - anoFim UNIQUE
- O par (nomeLiga, paisLiga) referencia uma liga
 - FOREIGN KEY (nomeLiga, paisLiga) REFERENCES Liga(nomeLiga, paisLiga)

Jornada:

- Não podem existir duas jornadas com o mesmo ID
 - idJornada PRIMARY KEY
- Uma Jornada só tem uma época
 - epoca REFERENCES Epoca

Jogo:

- Não podem existir dois jogos com o mesmo ID
 - idJogo PRIMARY KEY
- A classificação da Equipa de Arbitragem varia entre 0 e 10
 - classificacaoEquipaArbitragem CHECK (classificacaoEquipaArbitragem >= 0 AND classificacaoEquipaArbitragem <= 10)
- O idJornada deve de corresponder a um ID da tabela Jornada
 - idJornada REFERENCES Jornada(idJornada)
- O idDelegado deve de corresponder a um ID da tabela Delegado
 - idDelegado REFERENCES Delegado(idDelegado)
- O idCubeCasa deve de corresponder a um ID da tabela Clube

- idClubeCasa REFERENCES Clube(idClubeCasa)
- O idCubeFora deve de corresponder a um ID da tabela Clube
 - idClubeCasa REFERENCES Clube(idCubeFora)
- Relativamente à data, ela será do formato “YYYY-MM-DD HH:MM” (algo alcançado através da função strftime) e deverá ser NOT NULL.

EstatisticasClubeJogo:

- Não podem existir duas estatísticas do clube num jogo
 - idEstatisticasClubeJogo PRIMARY KEY
- O número de golos tem de ser sempre positivo ou zero
 - numGolos DEFAULT 0 CHECK (numGolos >= 0)
- O número de cartões amarelos tem de ser sempre positivo ou zero
 - numCartoesAmarelos DEFAULT 0 CHECK (numCartoesAmarelos >= 0)
- O número de cartões vermelhos varia entre 0 e 5
 - numCartoesVermelhos DEFAULT 0 CHECK (numCartoesVermelhos >= 0 AND numCartoesVermelhos <= 5)
- O número de faltas tem de ser sempre positivo ou zero
 - numFaltas DEFAULT 0 CHECK (numFaltas >= 0)
- O número de remates tem de ser sempre positivo ou zero
 - numRemates DEFAULT 0 CHECK (numRemates >= 0)
- O número de remates à baliza varia entre 0 e o número de remates
 - numRematesBaliza DEFAULT 0 CHECK (numRematesBaliza >= 0 AND numRematesBaliza <= numRemates)
- O número de cantos é sempre positivo ou zero
 - numCantos DEFAULT 0 CHECK (numCantos >= 0)
- O número de foras de jogo é sempre positivo ou zero
 - numForasDeJogo DEFAULT 0 CHECK (numForasDeJogo >= 0)
- O número de assistências varia entre 0 e o número de golos
 - numAssistencias DEFAULT 0 CHECK (numAssistencias >= 0 AND numAssistencias <= numGolos)
- Uma estatística só pode estar associada a um jogo
 - idJogo REFERENCES Jogo

EventoJogo:

- Não podem existir dois eventos num jogo com o mesmo ID
 - idEvento PRIMARY KEY
- O idJogo corresponde a um id da tabela Jogo

- idJogo REFERENCES Jogo(idJogo)
- Relativamente ao número de minutos jogados é feito um CHECK(minuto >= 0)

Golo:

- Não podem existir dois golos com o mesmo ID
 - idEvento PRIMARY KEY
- O idJogador corresponde a um id da tabela Jogador
 - idJogador REFERENCES Jogador(idJogador)
- Relativamente ao número de minutos jogados é feito um CHECK(minuto >= 0)

Falta:

- Não podem existir duas faltas com o mesmo ID
 - idEvento PRIMARY KEY
- O idJogador corresponde a um id da tabela Jogador
 - idJogador REFERENCES Jogador(idJogador)
- Relativamente ao número de minutos jogados é feito um CHECK(minuto >= 0)

Canto:

- Não podem existir dois cantos com o mesmo ID
 - idEvento PRIMARY KEY
- O idJogador corresponde a um id da tabela Jogador
 - idJogador REFERENCES Jogador(idJogador)
- O idClube corresponde a um id da tabela Clube
 - idClube REFERENCES Jogador(idClube)
- Relativamente ao número de minutos jogados é feito um CHECK(minuto >= 0)

ForaDeJogo:

- Não podem existir dois foras de jogo com o mesmo ID
 - idEvento PRIMARY KEY
- O idJogador corresponde a um id da tabela Jogador
 - idJogador REFERENCES Jogador(idJogador)
- O idClube corresponde a um id da tabela Clube
 - idClube REFERENCES Jogador(idClube)
- Relativamente ao número de minutos jogados é feito um CHECK(minuto >= 0)

Assistência:

- Não podem existir duas assistências com o mesmo ID
 - idEvento PRIMARY KEY
- O idJogador corresponde a um id da tabela Jogador
 - idJogador REFERENCES Jogador(idJogador)
- Relativamente ao número de minutos jogados é feito um CHECK(minuto >= 0)

Cartao:

- Não podem existir dois cartões com o mesmo ID
 - idEvento PRIMARY KEY
- A cor do cartão só pode ser amarelo ou vermelho
 - cor CHECK (cor LIKE 'amarelo' OR cor LIKE 'vermelho')
- Relativamente ao número de minutos jogados é feito um CHECK(minuto >= 0)

Remate:

- Não podem existir dois remates com o mesmo ID
 - idEvento PRIMARY KEY
- Um remate entra na baliza ou não
 - naBaliza CHECK (naBaliza = 0 OR naBaliza = 1)
- Um remate é efetuado por um único jogador
 - idJogador REFERENCE Jogador(idJogador)
- Relativamente ao número de minutos jogados é feito um CHECK(minuto >= 0)

Nota: Todas as restrições de atributos de classes mães se aplicam também às sub-classes. Por exemplo, o atributo telefone da classe Pessoa é NOT NULL UNIQUE, pelo que também será NOT NULL UNIQUE na classe jogador.

As outras restrições anunciadas previamente serão implementadas através de *triggers* na 3ª entrega. Referimo-nos claro, às *constraints* que comparam atributos de diferentes relações. Por fim, no ato da criação da base de dados em SQL, foram utilizadas restrições respeitantes ao ON UPDATE e ON DELETE, de modo a manter a integridade referencial dos dados armazenados.

Nota relativa à 3ª Entrega: Foram adicionados mais alguns dados, de modo a mostrar mais facilmente que as interrogações e gatilhos estão a funcionar devidamente.

Interrogações:

1. Jogos em que houve pelo menos um cartão vermelho (e respetivo número de cartões vermelhos nesse jogo).
2. Patrocinadores que patrocinam um clube e uma época ao mesmo tempo.
3. Árbitros que arbitraram o primeiro jogo do FCPorto (idClube = 7) na época de 2019.
4. Prémio melhor árbitro(s) da época 2019 (pode haver mais do que um árbitro a ganhar o prémio).
5. Menor intervalo de tempo (em dias) entre dois jogos da mesma equipa (e respetivas datas desses mesmos jogos).
6. Clube com a melhor eficácia de golos (número de golos / número de remates), (podem haver clubes com a mesma eficácia).
7. Jogadores que, em média, marcam mais do que um golo por jogo.
8. Resultados de todos os jogos dos clubes na época de 2019.
9. Top 3 clubes com o maior número de golos marcados (em caso de empate vê o(s) que tiverem menos golos sofridos).
10. Classificação das equipas: por cada vitória, a equipa arrecada 3 pontos, por cada empate, 1 ponto, e por cada derrota, 0 pontos. Em caso de empate, tem vantagem a equipa com maior diferença de golos marcados e golos sofridos; em caso de novo empate tem vantagem a equipa com mais golos marcados.

Gatilhos:

1. Um árbitro não pode apitar mais do que um jogo na mesma jornada; além disso, a sua classificação global (soma de todas as classificações nos jogos) é atualizada a cada jogo.
2. Numa época, uma equipa não pode jogar duas vezes no mesmo local contra uma mesma equipa; além disso, duas equipas não se podem defrontar mais do que duas vezes.
3. Para cada golo inserido na base de dados, atualizar a classificação do clube na época. Enquanto o jogo está a decorrer, atualizar apenas os dados relativos ao número de golos (golos marcados, sofridos, e diferença entre estes). Quando o jogo terminar, calcular os pontos ganhos pela equipa no jogo (em caso de vitória, a equipa arrecada 3 pontos, em caso de empate, 1 ponto, e em caso de derrota, 0 pontos), e adicioná-los ao total de pontos da época. Também ao longo do jogo, pode ser inserida informação sobre outros eventos de jogo (Cartões, Faltas, Remates, Foras de Jogo...) que serão também devidamente registadas na Estatísticas do Clube no Jogo, relação essa que será posteriormente usada na atualização do número de vitórias, empates, derrotas e cálculo de pontos.