

Faculdade de Engenharia da Universidade do Porto



Organização Mundial de Xadrez Bases de Dados



2021/2022 -- Licenciatura em Engenharia Informática e Computação

Carla Teixeira Lopes

Michel Ferreira

Equipa 401:

Docente: Lázaro Gabriel Barros da Costa

Estudantes & Autores:

Francisco Calado Araújo: up201806326@edu.fe.up.pt

Francisco Pimentel Serra: up202007723@edu.fe.up.pt

Pedro João Paixão: up202008467@edu.fe.up.pt

Índice

1. Modelo Conceptual	3
1.1 Descrição do Contexto	3
1.2 Diagrama UML	4
1.3 Diagrama UML Revisto	5
2. Modelo Relacional	6
2.1 Esquema relacional	6
2.2 Dependências Funcionais e Formas Normais	7
2.3 Restrições	10

1. Modelo Conceptual

1.1 Descrição do Contexto

Pretende-se armazenar a informação relativa a todo o tipo de **Partidas/Torneios** que se realizem em xadrez clássico. De cada **Jogador** armazena-se o seu primeiro e último nome, a morada, o contacto de telefone e a data de nascimento. Cada **Jogador** pode pertencer apenas a um **Clube** de xadrez. Do **Clube** é necessário saber o seu nome, a morada, o ranking e o número de membros que se sabe pelo número de **Jogadores** que pertencem ao **Clube**. Através do id de membro é possível saber qual o **Jogador** associado ao **Clube**.

O **Clube** pode criar os seus próprios **Torneios**, sendo que cada **Torneio** tem, uma data inicial e de término, um nome e um vencedor. As **Partidas** têm uma data, um resultado, os detalhes desse resultado, a duração inicial das peças brancas e pretas, que pode ser diferente, o incremento e o número de movimentos. Para além disso, as **Partidas** têm 4 tipos: **Clássico**, que tem inicialmente mais de 60 minutos, **Rápido**, que cada **Partida** tem inicialmente mais de 10 minutos, **Blitz**, em que as **Partidas** têm inicialmente no mínimo 3 minutos e todas as **Partidas** que inicialmente têm um tempo inferior são chamadas de **Bullet**. Cada **Partida** pertence a um único tipo e todas as **Partidas** pertencem a um tipo e têm no mínimo dois **Jogadores**. Obviamente, todos os **Torneios** têm **Partidas** e todos os **Jogadores** estão associados a **Torneios** e a **Partidas**.

Todos os **Jogadores** podem ter várias **Contas** online, em diversos **Websites**, sendo que de cada **Conta** é necessário ter uma alcunha, uma conta email, número de pontos e um título, que é intransmissível. Do **Website** associado, é necessário saber o seu link e nome, sendo que se este desaparecer o mesmo acontece nas **Contas** dos **Jogadores** nessa plataforma. O **Website** pode também criar os seus **Torneios** e **Partidas**.

O **Jogador** pode também ter associado um **Ranking da Federação de Xadrez**, sendo que esta tem pontos ELO, um título e uma categoria (homens, mulheres, juniores, raparigas). Os **Jogadores** e os **Torneios** podem ter **Patrocinadores**, dos quais é preciso saber o nome, telefone, endereço eletrónico e morada, sendo que, de cada patrocinador é lhe dado um nível de patrocínio (ouro, prata ou bronze).

1.2 Diagrama UML

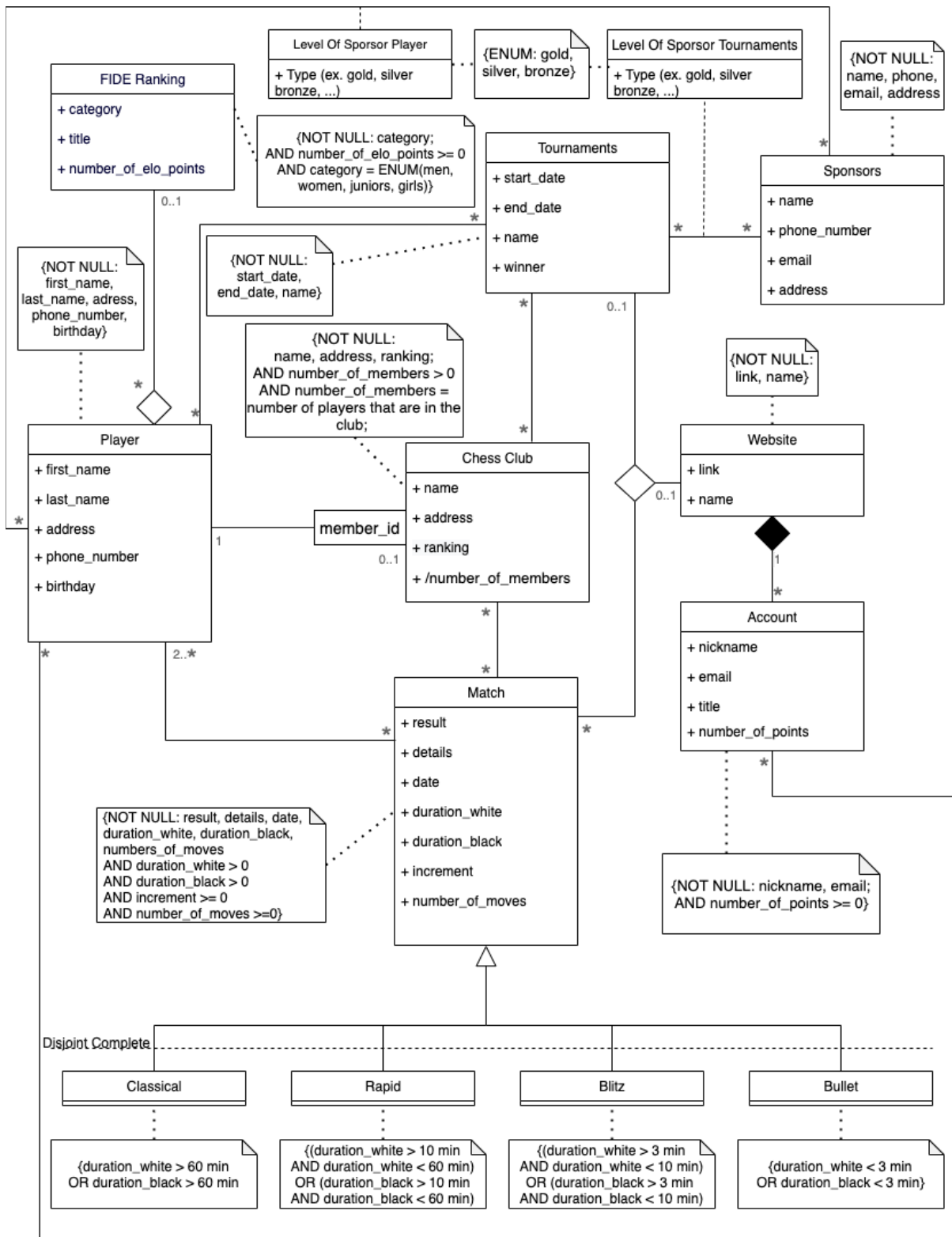


Figura 1: Diagrama UML.

1.3 Diagrama UML Revisto

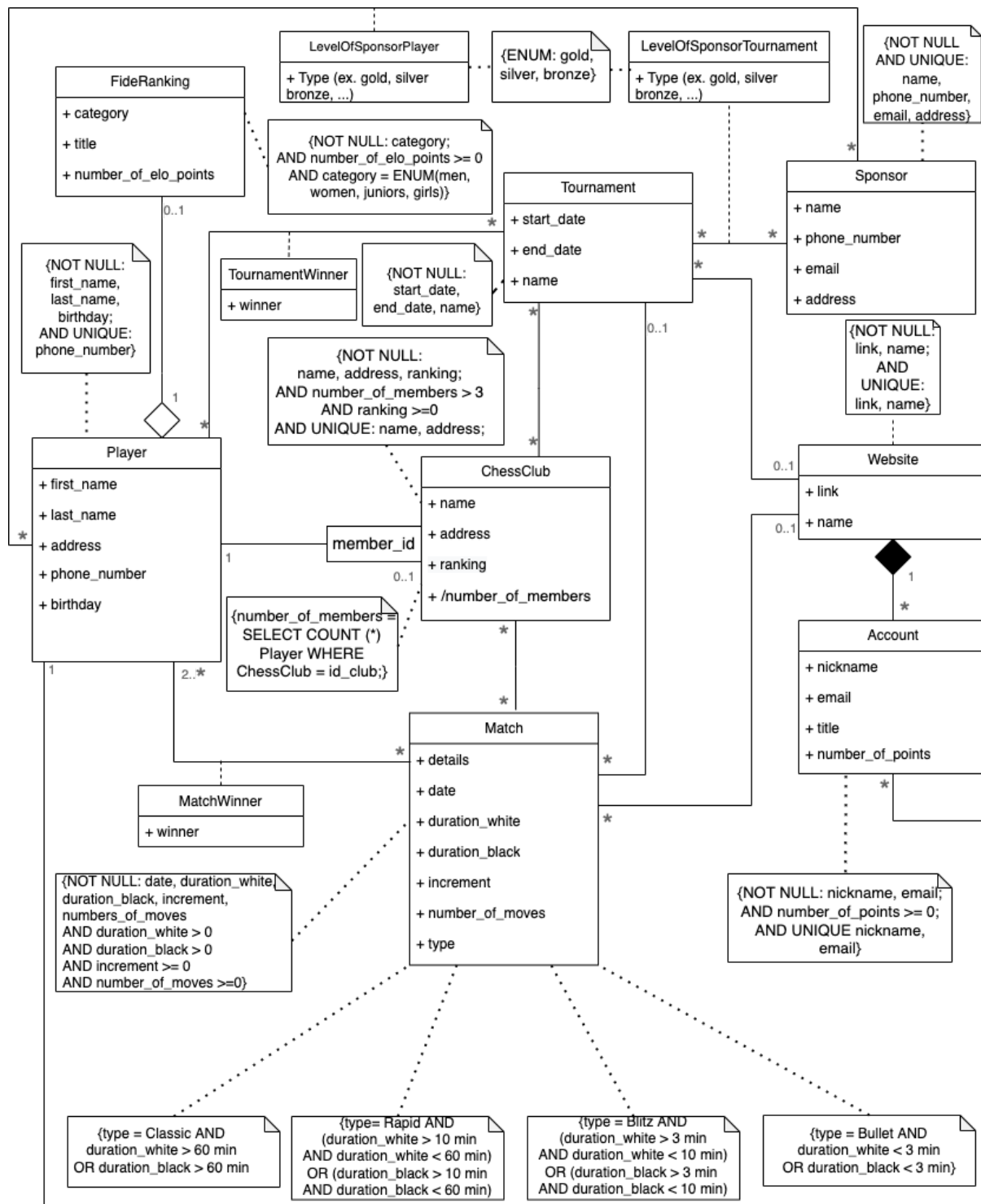


Figura 2: Diagrama UML Revisto, disponível [aqui](#).

2. Modelo Relacional

2.1 Esquema relacional

Player (id_player, first_name, last_name, adress, phone_number, birthday)

FideRanking (id_fide, id_player->Player, category, title, number_of_elo_points)

Website (id_website, link, name)

Account (id_account, nickname, email, title, number_of_points, id_website->Website)

Tournament (id_tournament, start_date, end_date, name, id_website->Website)

Match (id_match, details, date, duration_white, duration_black, increment, number_of_moves, id_tournament->Tournament, id_website->Website)

Sponsor (id_sponsor, name, phone_number, email, address)

PlayerMatch (id_player->Player, id_match->Match, winner)

PlayerTournament (id_player->Player, id_tournament->Tournament, winner)

ChessClub (id_club, name, address, ranking, number_of_members)

MemberId (id_club->ChessClub, id_player->Player, member_id)

LevelOfSponsorPlayer (id_player->Player, id_sponsor->Sponsor, type)

LevelOfSponsorTournament (id_tournament->Tournament, id_sponsor->Sponsor, type)

PlayerAccount (id_player->Player, id_account->Account)

MatchChessClub (id_match->Match, id_club->ChessClub)

TournamentChessClub (id_tournament->Tournament, id_club->ChessClub)

2.2 Dependências Funcionais e Formas Normais

Player (id_player, first_name, last_name, adress, phone_number, birthday)

-DFs:

- id_player -> first_name, last_name, address, phone_number, birthday
- phone_number -> id_player, first_name, last_name, address, birthday

-Formas:

- FNBC: Sim
- 3FN: Sim

FideRanking (id_fide, id_player->Player, category, title, number_of_elo_points)

-DF: id_fide->id_player, category, title, number_of_elo_points

-Formas:

- FNBC: Sim
- 3FN: Sim

Website (id_website, link, name)

-DFs:

- id_website -> link, name
- link -> id_website, name
- name-> id_website, link

-Formas:

- FNBC: Sim
- 3FN: Sim

Account (id_account, nickname, email, title, number_of_points, id_website->Website)

-DFs:

- id_account -> nickname, email, title, number_of_points, id_website
- nickname -> id_account, email, title, number_of_points, id_website
- email -> id_account, nickname, title, number_of_points, id_website

-Formas:

- FNBC: Sim
- 3FN: Sim

Tournament (id_tournament, start_date, end_date, name, id_website->Website)

-DF: id_tournament -> start_date, end_date, name, winner, id_website

-Formas:

- FNBC: Sim
- 3FN: Sim

Match (id_match, result, details, date, duration_white, duration_black, increment, number_of_moves, id_tournament->Tournament, id_website->Website, type)

-DF: id_match -> result, details, date, duration_white, duration_black, increment, number_of_moves, id_tournament, id_website, type

-Formas:

- FNBC: Sim
- 3FN: Sim

Sponsor (id_sponsor, name, phone_number, email, address)

-DFs:

- id_sponsor -> name, phone_number, email, address
- name -> id_sponsor, phone_number, email, address
- phone_number -> id_sponsor, name, email, address
- email -> id_sponsor, name, phone_number, address
- address -> id_sponsor, name, phone_number, email

-Formas:

- FNBC: Sim
- 3FN: Sim

PlayerMatch (id_player->Player, id_match->Match, winner)

-DFs: --

PlayerTournament (id_player->Player, id_tournament->Tournament, winner)

-DFs: --

ChessClub(id_club, name, address, ranking, number_of_members)

-DFs:

- id_club -> name, address, ranking, number_of_members
- name -> id_club, address, ranking, number_of_members
- address -> id_clube, name, ranking, number_of_members

-Formas:

- FNBC: Sim
- 3FN: Sim

MemberId (id_club->ChessClub, id_player->Player, member_id)

-DFs:

- id_club, id_player-> member_id
- member_id -> id_club, id_player

-Formas:

- FNBC: Sim
- 3FN: Sim

LevelOfSponsorPlayer (id_player->Player, id_sponsor->Sponsor, type)

-DF: id_player, id_sponsor-> type

-Formas:

- FNBC: Sim
- 3FN: Sim

LevelOfSponsorTournament (id_tournament->Tournament, id_sponsor->Sponsor, type)

-DF: id_tournament, id_sponsor-> type

-Formas:

- FNBC: Sim
- 3FN: Sim

PlayerAccount (id_player->Player, id_account->Account)

-DFs: --

MatchChessClub (id_match->Match, id_club->ChessClub)

-DFs:--

TournamentChessClub (id_tournament->Tournament, id_club->ChessClub)

-DFs: --

Todas as relações da base de dados estão de acordo com a 3ª Forma Normal e a Forma Normal de Boyce-Codd.

Cada relação está na Forma Normal Boyce-Codd se para cada $A \rightarrow B$ não é trivial e A é (super) chave.

Para uma relação estar na 3ª Forma Normal $A \rightarrow B$ é não trivial e A é (super) chave ou B consiste apenas de atributos primos (atributos que fazem parte de uma qualquer superchave de cada relação).

2.3 Restrições

Player

- Não pode haver duas classes players como o mesmo ID
 - id_player PRIMARY KEY
- Todas as classes players têm que ter primeiro e último nome tal como uma data de nascimento
 - first_name NOT NULL
 - last_name NOT NULL
 - birthday NOT NULL
- Não podem haver números de telefone repetidos
 - phone_number UNIQUE

FideRanking

- Não pode haver duas classes players como o mesmo ID fide
 - id_fide PRIMARY KEY
- A classe category tem sempre que estar preenchida e os valores podem ser: MEN, WOMEN, JUNIORS e GIRLS
 - category NOT NULL CHECK (category= "MEN" OR
category= "WOMEN" OR
category= "JUNIORS" OR
category= "GIRLS")
- A classe number_of_elo_points tem que ser sempre igual ou superior a zero
 - number_of_elo_points CHECK (number_of_elo_points >=0)

Website

- Não pode haver duas classes websites com o mesmo ID
 - id_website PRIMARY KEY
- Tanto o link como o nome do website não podem ser nulos e têm que ser únicos
 - link NOT NULL UNIQUE
 - name NOT NULL UNIQUE

Account

- Não pode haver duas classes accounts como o mesmo ID
 - id_account PRIMARY KEY

- Não podem existir duas classes **accounts** com o mesmo nickname e email e nenhum dos dois pode ser nulo
 - nickname UNIQUE NOT NULL
 - email UNIQUE NOT NULL
- O number_of_points tem que ser sempre igual ou superior a zero
 - number_of_points CHECK (number_of_points >=0)
- O ID do website tem que pertencer a uma instância da tabela **Website**
 - id_website REFERENCES Website

Tournament

- Não pode haver duas classes **tournaments** com o mesmo ID
 - id_tournament PRIMARY KEY
- Nenhum dos atributos pode ser nulo
 - start_date NOT NULL
 - end_date NOT NULL
 - name NOT NULL
- O ID do website tem que pertencer a uma instância da tabela **Website**
 - id_website REFERENCES Website

Match

- Não pode haver duas classes **matches** como o mesmo ID
 - id_match PRIMARY KEY
- Nenhum dos atributos pode ser nulo
 - date NOT NULL
 - duration_white NOT NULL
 - duration_black NOT NULL
 - increment NOT NULL
 - number_of_moves NOT NULL
- Todos estes atributos têm que ser maiores ou iguais a zero
 - duration_white CHECK (duration_white >=0)
 - duration_black CHECK (duration_black >=0)
 - increment CHECK (increment >=0)
 - number_of_moves CHECK (number_of_moves >=0)
- O ID do torneio e o ID do website têm que pertencer a uma instância das tabelas **Tournament** e **Website** respetivamente
 - id_tournament REFERENCES Tournament
 - id_website REFERENCES Website

Sponsor

- Não pode haver duas classes **sponsors** com o mesmo ID
 - id_sponsor PRIMARY KEY
- Nenhum dos atributos pode ser nulo e todos têm que ser únicos
 - name NOT NULL UNIQUE
 - phone_number NOT NULL UNIQUE
 - email NOT NULL UNIQUE

- address NOT NULL UNIQUE

PlayerMatch

- Tanto o ID do jogador como o ID da partida têm que pertencer a uma instância da tabela **Player** como da tabela **Match**, respetivamente
 - id_player REFERENCES Player
 - id_match REFERENCES Match
- Não pode haver duas instâncias com o mesmo par (id_player, id_match)
 - PRIMARY KEY (id_player, id_match)

PlayerTournament

- Tanto o ID do jogador como o ID do torneio têm que pertencer a uma instância da tabela **Player** como da tabela **Tournament**, respetivamente
 - id_player REFERENCES Player
 - id_tournament REFERENCES Tournament
- Não pode haver duas instâncias com o mesmo par (id_player, id_tournament)
 - PRIMARY KEY (id_player, id_tournament)

ChessClub

- Não pode haver duas classes **clubs** com o mesmo ID
 - id_club PRIMARY KEY
- Nenhum dos atributos pode ser nulo
 - name NOT NULL
 - ranking NOT NULL
 - address NOT NULL
- Tanto o nome como a morada têm que ser únicos
 - name UNIQUE
 - address UNIQUE
- A pontuação tem que ser maior ou igual a zero e o números de membros superior a três
 - ranking CHECK (ranking >=0)
 - number_of_members CHECK (number_of_members > 3)

MemberId

- Tanto o ID do clube como o ID do jogador têm que pertencer a uma instância da tabela **ChessClub** como da tabela **Player**, respetivamente
 - id_club REFERENCES ChessClub
 - id_player REFERENCES Player
- o ID de um membro não pode ser nulo
 - member_id NOT NULL
- Não pode haver duas instâncias com o mesmo par (id_club, id_player)
 - PRIMARY KEY (id_club, id_player)

LevelOfSponsorPlayer

- Tanto o ID do jogador como o ID do patrocinador têm que pertencer a uma instância da tabela **Player** como da tabela **Sponsor**, respetivamente
 - id_player REFERENCES Player
 - id_sponsor REFERENCES Sponsor
- O tipo do nível de um patrocinador do jogador não pode ser nulo e o valor pode ser: GOLD, SILVER ou BRONZE)
 - type NOT NULL CHECK(type="GOLD" OR
type= "SILVER" OR
type = "BRONZE")
- Não pode haver duas instâncias com o mesmo par (id_player, id_sponsor)
 - PRIMARY KEY (id_player, id_sponsor)

LevelOfSponsorTournament

- Tanto o ID do torneio como o ID do patrocinador têm que pertencer a uma instância da tabela **Tournament** como da tabela **Sponsor**, respetivamente
 - id_tournament REFERENCES Tournament
 - id_sponsor REFERENCES Sponsor
- o tipo do nível de um patrocinador do torneio não pode ser nulo e o valor pode ser: GOLD, SILVER ou BRONZE)
 - type NOT NULL CHECK (type="GOLD" OR
type= "SILVER" OR
type = "BRONZE")
- Não pode haver duas instâncias com o mesmo par (id_tournament, id_sponsor)
 - PRIMARY KEY (id_tournament, id_sponsor)

PlayerAccount

- Tanto o ID do jogador como o ID do conta têm que pertencer a uma instância da tabela **Player** como da tabela **Account**, respetivamente
 - id_player REFERENCES Player
 - id_account REFERENCES Account
- Não pode haver duas instâncias com o mesmo par (id_player, id_account)
 - PRIMARY KEY (id_player, id_account)

MatchChessClub

- Tanto o ID da partida como o ID do clube têm que pertencer a uma instância da tabela **Match** como da tabela **ChessClub**, respetivamente
 - id_match REFERENCES Match
 - id_club REFERENCES ChessClub
- Não pode haver duas instâncias com o mesmo par (id_match, id_club)
 - PRIMARY KEY (id_match, id_club)

TournamentChessClub

- Tanto o ID do torneio como o ID do clube têm que pertencer a uma instância da tabela **Tournament** como da tabela **ChessClub**, respetivamente
 - id_tournament REFERENCES Tournament

- id_club REFERENCES ChessClub
- Não pode haver duas instâncias com o mesmo par (id_tournament, id_club)
 - PRIMARY KEY (id_tournament, id_club)

NOTA: Restrições do tipo ON DELETE e ON UPDATE estão implementadas na base de dados em SQL de forma a manter a integridade da mesma.