

# Relatório Projeto

## Base de Dados – L.EIC012



Trabalho realizado pelo grupo 208, constituído por:

- Guilherme Coutinho – up202108872
- Mário Branco – up202008219
- Luís Sousa – up202005832

## Tópico A

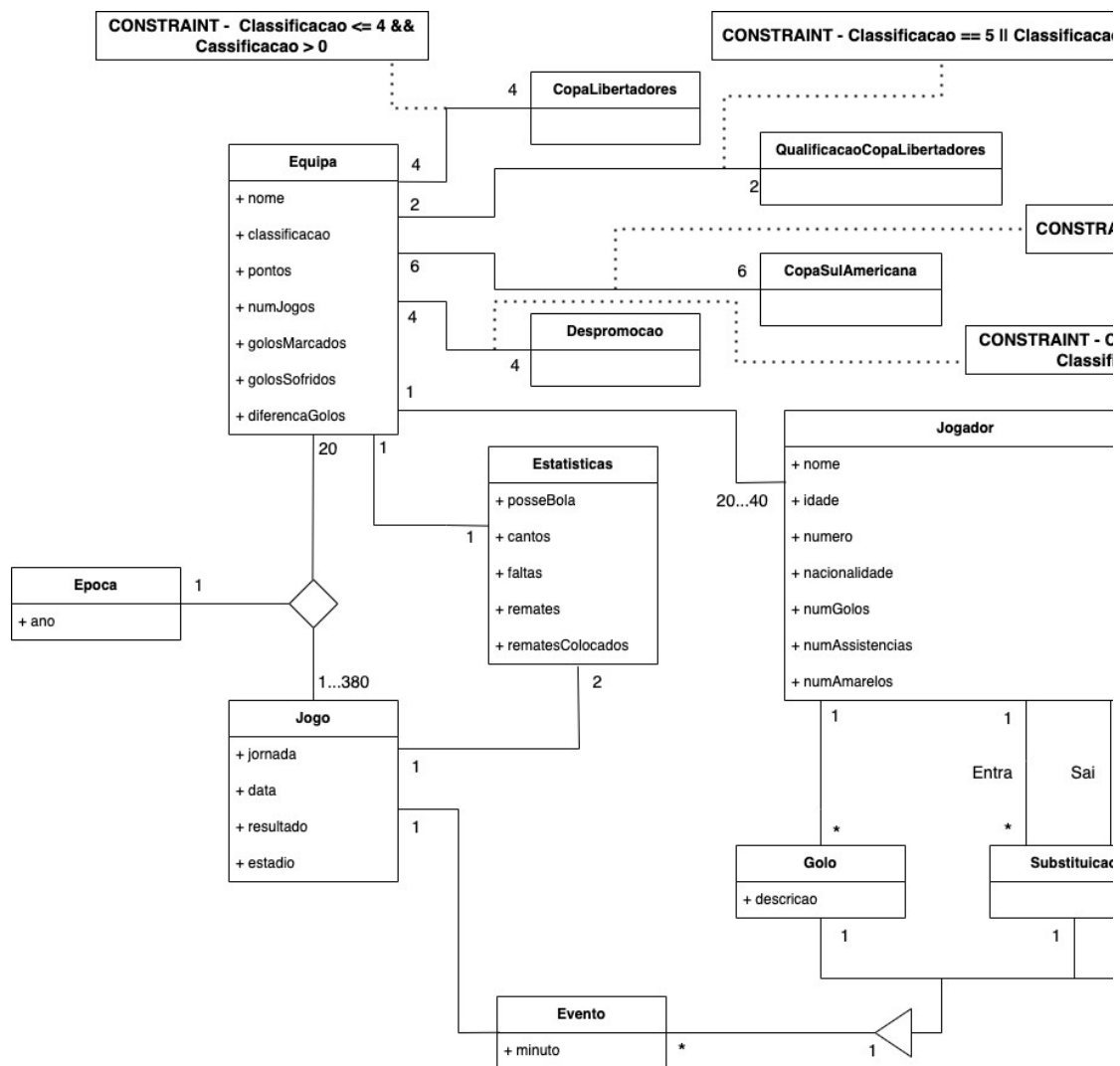
### Descrição do Contexto

O objetivo deste trabalho é desenvolver uma base de dados para gerir os resultados do campeonato brasileiro de futebol, da primeira divisão chamado Brasileirão. Esta tem como função armazenar os resultados dos jogos jornada a jornada, marcadores dos golos, equipas que jogam em casa e como visitante. Também é constituída com informação do estado do campeonato em qualquer jornada, as equipas em condições de ir às competições europeias e em risco de despromoção.

O Brasileirão é constituído por 20 equipas, sendo que cada uma tem cerca de 40 jogadores. Num jogo, jogam 11 jogadores por cada equipa, fazendo assim 22 jogadores dentro de campo, contamos também com eventuais substituições dos jogadores que estão no banco. São feitos ao todo 380 jogos numa época, sendo assim 38 jornadas com 10 jogos em cada jornada. Duas equipas jogam uma com a outra 2 vezes (como visitante e visitada nos estádios respetivos de quem joga em casa) e têm a chance de pontuar pontos dependendo do resultado do jogo, se uma equipa ganhar, a mesma ganha 3 pontos e a outra não ganha pontos e se o resultado ficar empatado as equipas ganham ambas 1 ponto.

No final da época, existem qualificações e despromoções a fazer, as equipas classificas em:

- 1º, 2º, 3º, 4º, 5º e 6º - estão automaticamente qualificadas para jogar na Copa Libertadores, que é uma competição entre as melhores equipas de futebol profissional da América do Sul;
- 7º e 8º - estão aptas a jogar uma série de jogos com outras equipas sul americanas para disputar um lugar na Copa Libertadores;
- 9º, 10º, 11º, 12º, 13º e 14º - estão automaticamente qualificadas para jogar na Copa Sul-Americana, que é outra competição secundária para as equipas de futebol profissional da América do Sul que não conseguiram entrar na Copa Libertadores;
- 17º, 18º, 19º e 20º - são despromovidas desta liga de futebol e dão lugar a outras 4 equipas de uma liga de futebol brasileira abaixo desta.



UML

## Tópico B

### Modelo Relacional

Jogador ( idJogador, nome, idade, numero, posicao, idEquipa -> Equipa )

Equipa ( idEquipa, nome, classificacao, pontos, numJogos, golosMarcados, golosSofridos, diferencaGolos )

Epoca ( idEpoca, ano )

Jogo ( idJogo, jornada, dia, resultado, estadio)

JogoEquipaEpoca ( idJogo -> Jogo, idEpoca -> Epoca, idEquipaCasa -> Equipa, idEquipaVisitante -> Equipa )

Estatisticas ( idEstatisticas, posseBola, cantos, faltas, remates, rematesColocados, idEquipa -> Equipa)

Golo ( idGolo, minuto, descricao, idJogador -> Jogador, idJogo -> Jogo)

Outro ( idOutro, minuto, descricao, idJogador -> Jogador , idJogo -> Jogo)

Substituicao ( idSubstituicao, minuto, idJogadorEntra -> Jogador, idJogadorSai -> Jogador, idJogo -> Jogo)

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

EntradaCopaLibertadores ( idEntradaCopaLibertadores,  
idEquipa -> Equipa )

QualificacaoCopaLibertadores  
( idQualificacaoCopaLibertadores, idEquipa -> Equipa )

EntradaCopaSulAmericana ( idEntradaCopaSulAmericana,  
idEquipa -> Equipa )

Despromocao ( idDespromocao, idEquipa -> Equipa )

## Tópico C

### Análise de Dependências Funcionais

Jogador ( idJogador, nome, idade, numero, posicao, idEquipa -> Equipa )  
idJogador -> nome, idade, numero, posicao, idEquipa

Equipa ( idEquipa, nome, classificacao, pontos, numJogos,  
golosMarcados, golosSofridos, diferencaGolos )  
idEquipa -> nome, classificacao, pontos, numJogos,  
golosMarcados, golosSofridos, diferencaGolos

Epoca ( idEpoca, ano )  
idEpoca -> ano

Jogo ( idJogo, jornada, dia, resultado, estadio )  
idJogo -> jornada, dia, resultado, estadio

JogoEquipaEpoca ( idJogo -> Jogo, idEpoca -> Epoca, idEquipaCasa ->  
Equipa, idEquipaVisitante -> Equipa )  
idJogo, idEpoca -> idEquipaCasa, idEquipaVisitante

Estatisticas ( idEstatisticas, posseBola, cantos, faltas, remates, rematesColocados, idEquipa -> Equipa)

idEstatisticas -> posseBola, cantos, faltas, remates, rematesColocados, idEquipa

Golo ( idGolo, minuto, descricao, idJogador -> Jogador, idJogo -> Jogo)  
idGolo -> minuto, descricao, idJogador, idJogo

Outro ( idOutro, minuto, descricao, idJogador -> Jogador, idJogo -> Jogo)  
idOutro -> minuto, descricao, idJogador, idJogo

Substituicao ( idSubstituicao, minuto, idJogadorEntra -> Jogador, idJogadorSai -> Jogador, idJogo -> Jogo)  
idSubstituicao -> minuto, idJogadorEntra, idJogadorSai, idJogo

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

EntradaCopaLibertadores ( idEntradaCopaLibertadores, idEquipa -> Equipa )  
idEntradaCopaLibertadores ->> idEquipa

QualificacaoCopaLibertadores ( idQualificacaoCopaLibertadores, idEquipa -> Equipa )  
idQualificacaoCopaLibertadores -> idEquipa

EntradaCopaSulAmericana ( idEntradaCopaSulAmericana, idEquipa -> Equipa )  
idEntradaCopaSulAmericana -> idEquipa

Despromocao ( idDespromocao, idEquipa -> Equipa )  
idDespromocao -> idEquipa

As definições de Boyce-Codd Normal Form e 3rd Normal Form são:

Boyce-Codd Normal Form:

Para qualquer dependência não trivial  $A \rightarrow B$ , A é uma superchave da relação.

3rd Normal Form:

Para qualquer dependência não trivial  $A \rightarrow B$ , A é uma superchave da relação OU B é constituído apenas por atributos primos.

Em qualquer relação da base de dados, com a parte da esquerda (A) é possível descobrir todos os atributos, ou seja, A é uma superchave. Portanto, todas as relações cumprem com a Boyce-Codd Normal Form e a 3rd Normal Form.

## Tópico E

### Restrições

NOT NULL – Usamos esta restrição para garantir que os parâmetros nunca iam ser nulos em todas as tabelas;

PRIMARY KEY – Esta restrição foi utilizada em todas as tabelas no parâmetro do id dessa tabela.

```
CREATE TABLE EQUIPA  
(  
    IDEQUIPA INTEGER PRIMARY KEY NOT NULL,
```

Foi também utilizada na Relação entre as tabelas EPOCA, JOGO e EQUIPA pois pode haver um id de um jogo igual ao id de outro jogo, mas noutra época, portanto era necessário a chave primária conter os dois parâmetros.

```
CREATE TABLE JOGOEQUIPAEPOCA
(
  IDJOGO INT NOT NULL,
  IDEPOCA INT NOT NULL,
  IDEQUIPACASA INT NOT NULL,
  IDEQUIPAVISITANTE INT NOT NULL,
  PRIMARY KEY(IDJOGO,IDEPOCA), FOREIGN KEY (IDEQUIPACASA) REFERENCES EQUIPA(IDEQUIPA),
  FOREIGN KEY (IDEQUIPAVISITANTE) REFERENCES EQUIPA(IDEQUIPA));
```

FOREIGN KEY – Usada nas tabelas sempre que foi preciso fazer ligação entre tabelas. As chaves estrangeiras são atributos numa tabela que servem de referência para chaves primárias noutra tabela.

Exemplo: Um jogador pertence a uma equipa, logo tem de ter um atributo que será uma chave estrangeira para uma chave primária do id dessa equipa para haver essa ligação entre os dados.

```
CREATE TABLE JOGADOR
(
  IDJOGADOR INTEGER PRIMARY KEY NOT NULL,
  NOME TEXT NOT NULL,
  IDADE INT NOT NULL,
  NUMERO INT NOT NULL,
  POSICAO TEXT NOT NULL,
  IDEQUIPA TEXT NOT NULL,
  FOREIGN KEY (IDEQUIPA) REFERENCES EQUIPA(IDEQUIPA));
```

CHECK – Utilizamos esta restrição para verificar se os valores introduzidos estão dentro do intervalo pretendido.

```
CREATE TABLE ENTRADACOPALIBERTADORES
(
  IDENTRADACOPALIBERTADORES INTEGER PRIMARY KEY NOT NULL,
  IDEQUIPA TEXT NOT NULL,
  CLASSIFICACAO INT NOT NULL CHECK(CLASSIFICACAO <= 4 AND CLASSIFICACAO > 0),
  FOREIGN KEY (IDEQUIPA) REFERENCES EQUIPA(IDEQUIPA),
  FOREIGN KEY (CLASSIFICACAO) REFERENCES EQUIPA(CLASSIFICACAO));
```



## Tópico F

### Carregamento de dados

Para carregar os dados para a Base de dados, utilizamos uma ferramenta do Google Chrome chamada “Web Scraper”. Visitámos o website [zerozero.pt](http://zerozero.pt) para conseguirmos retirar toda a informação sobre a época deste ano do Brasileiro.

//Colocar aqui o processo de dar scrape

Depois de ter os ficheiros csv das relações todas, foi necessário “limpar” as linhas, mas como alguns ficheiros tinham milhares de linhas, fabricamos um ficheiro de código em python para apagar os caracteres desnecessários.

Exemplo de código para um dos ficheiros:

```
# script to remove unwanted stuff from a certain file that has
# been scraped from
# the web and put in a csv file

def cleanFile(readFile, writeFile, integers):
    read = open(readFile, 'r')
    write = open(writeFile, 'w')
    lines = read.readlines()
    aid = 0

    for line in lines:
        i = 0
        count = 0
        outString = ""
        aid += 1
        fields = line.split(',')

        equipaCasa = fields[1]
        equipaVisitante = fields[2]
        equipasFile = open("EQUIPA.csv", 'r')
        equipas = equipasFile.readlines()
        idEquipaCasa, idEquipaVisitante = 0,0

        for e in equipas:
            if equipas.index(e) == 0:
                continue

            fields_equipas = e.split(',')
            if fields_equipas[1] == equipaVisitante:
                idEquipaVisitante = fields_equipas[0]
            elif fields_equipas[1] == equipaCasa:
```

```

idEquipaCasa = fields_equipas[0]

    outString = fields[0] + ",1," + str(idEquipaCasa) + ',' +
str(idEquipaVisitante) + '\n'
    # step ahead first line
    if lines.index(line) == 0:
        outString =
"idJogo,idEpoca,idEquipaCasa,idEquipaVisitante\n"
        write.write(outString)
        continue

    # count = 0
    # outString = ""
    # for i in range(len(fields)):
    #     if i in integers:
    #         #remove " from integer fields
    #         fields[i] = fields[i][2:len(fields[i])-1]

    # # remove last " in line
    # fields[len(fields)-1] = fields[len(fields)-1]
[:len(fields[len(fields)-1])-1]
    # rejoin fields in a string

    # add a newline
    # outString += '\n'
    # write out to the file
    write.write(outString)

# it is necessary to change the file names and the
# indexes of the fields that are integers
# from each file that is going to be cleaned
cleanFile("JOGOEquipaEpocas.csv", "JOGOEQUIPAEPOCA.csv", [4])

```

Assim que acabámos a limpeza dos ficheiros, utilizamos o “DB Browser” uma app que cria um ficheiro de base de dados e guarda toda a informação das tabelas dessa base. Este software permitiu-nos importar os ficheiros csv e colocar a sua informação para as tabelas criadas. Quando a base obteve toda a informação necessária, exportamos a informação toda para um ficheiro sql, neste caso, o “povoar.sql”.

## Tópico G

### Interrogação da base de dados

Interrogação 1 – Listar todos os jogadores da Base de dados (mostrar nome e equipa;

Interrogação 2 – Selecionar as equipas que ficaram no top 5 da

classificação final (mostrar nome da equipa e respetiva qualificação);  
Interrogação 3 – Selecionar os jogos que ficaram empatados na primeira rodada do campeonato (mostrar equipas e número da jornada);  
Interrogação 4 – Selecionar todos os jogadores com mais de 3 cartões amarelos (mostrar nome equipa);  
Interrogação 5 – Selecionar todas as equipas com diferença de golos negativa (mostrar equipa, classificação e a diferença de golos);  
// 6 ... 10;

## Tópico H

### Adição de gatilhos à base de dados

## Participação

Guilherme Coutinho (up202108872) – 37,5%  
Mário Branco (up202008219) – 37,5%  
Luís Sousa (up202005832) – 25%