Uma imagem com logótipo

Descrição gerada automaticamente

**NBA League**

**Base de Dados 22-23**

**André Oliveira – 107637**

**Duarte Cruz – 107359**

**P4– G1**

Índice

[Introdução 3](#_Toc136620374)

[Estrutura da Pasta do Projeto 4](#_Toc136620375)

[Requisitos Funcionais 5](#_Toc136620376)

[Entidades 6](#_Toc136620377)

[Diagramas 7](#_Toc136620378)

[Diagrama Entidade-Relacionamento 7](#_Toc136620379)

[Esquema Relacional 7](#_Toc136620380)

[Esquema Relacional (SGDB) 8](#_Toc136620381)

[Queries SQL 9](#_Toc136620382)

[Data Definition Language (DDL) 9](#_Toc136620383)

[Data Manipulation Language (DML) 9](#_Toc136620384)

[Views 9](#_Toc136620385)

[Indexes 9](#_Toc136620386)

[User Definition Functions (UDF’s) 9](#_Toc136620387)

[Stored Procedures 9](#_Toc136620388)

[Triggers 9](#_Toc136620389)

# **Introdução**

A escolha do nosso projeto foi motivada pelo interesse em comum dos membros do grupo na NBA. A aplicação permitirá o acompanhamento da temporada atual da liga.

A interface foi desenvolvida para o uso administrativo para a manipulação dos diversos dados, mas na perspetiva do utilizador, estes poderiam acompanhar a classificação geral da temporada, informações sobre equipas específicas e seus jogadores assim como, sobre os treinadores. Para além disto, poderão até mesmo consultar os bilhetes restantes para cada jogo.

Além disso, a nossa aplicação oferecerá informações sobre resultados de jogos, estatísticas dos jogadores, etc. Isto permitirá aos usuários se manterem atualizados sobre as últimas novidades da NBA numa única plataforma.

Futuramente, a nossa plataforma poderia permitir que os usuários criem seus próprios perfis e se conectem com outros fãs da NBA, compartilhando opiniões e participando de discussões sobre a liga, tornando a experiência mais interativa e social para os usuários.

Para desenvolver a nossa aplicação, planeamos recolher dados de diversas fontes, incluindo *websites* das diversas equipas.

Em resumo, a nossa aplicação, por agora, forneceria uma solução completa para os fãs da NBA, oferecendo informações atualizadas e relevantes e a possibilidade de busca de informações de bilhetes para jogos, tudo em um só lugar.

Na realização deste projeto foi utilizada uma pasta partilhado no OneDrive como plataforma de organização. O projeto também se encontra disponível no GitHub, mas sublinhamos que os *commits* não correspondem à participação de cada elemento do grupo.

# **Estrutura da Pasta do Projeto**

Dentro do repositório Projeto-BD e do ficheiro .zip é possível encontrar 3 imagens (diagrama DER, diagrama ER e diagrama ER do SGBD), uma pasta SQL com ficheiros .sql divididos de acordo com o tipo de queries (DDL.sql, DML.sql, INDEXES.sql, STORED\_PROCEDURES.sql, TRIGGERS.sql, UDF.sql e VIEWS.sql), a pasta Proposta com os ficheiros entregues anteriormente na proposta do projeto, o pdf Apresentação que corresponde ao powerpoint usado na apresentação do projeto na aula e os ficheiros relativos à interface gráfica (Pasta Projeto e o ficheiro Projeto.sln inicia a interface).

* **Pasta SQL:**
  + **DDL.sql –** *Script SQL* que contém a definição de todas as tabelas necessárias, assim como definições de chaves primárias e estrangeiras e algumas verificações para os atributos de cada tabela;
  + **DML.sql –** *Script SQL* que contém os dados a ser inseridos em cada tabela;
  + **INDEXES.sql –** *Script SQL* que contém os *Indexes*;
  + **STORED\_PROCEDURES.sql –** *Script SQL* que contém os *Stored Procedures*;
  + **TRIGGERS.sql –** *Script SQL* que contém os *Triggers*;
  + **UDF.sql –** *Script SQL* que contém as *User Defined Functions*;
  + **VIEWS.sql –** *Script SQL* que contém as *Views*.

**Nota:** A cada execução de um ficheiro .sql, é tudo refeito, não sendo necessário apagar nada antes.

# **Requisitos Funcionais**

|  |  |
| --- | --- |
| Entidade | Funcionalidades |
| Utilizador | * Procurar informações sobre cada equipa, sobre os respetivos treinadores das equipas e sobre os jogadores * Procurar informações sobre jogos, assim como os bilhetes para os mesmos que ainda não ocorreram * Acompanhar a classificação da liga atual * Procurar estatísticas das equipas e de respetivos jogadores (média e de cada jogo) |
| Administrador | * Inserir/alterar/remover dados sobre jogos, treinadores, jogadores e equipas |

# **Entidades**

**Team –** Representa uma equipa participante da liga. Possui um ID, um nome, uma conferência, um ano de fundação e é de uma cidade. Esta, tem vários jogadores, um treinador e um presidente.

**Person –** Representa a instância mais geral do treinador e do jogador. Possui um nome, uma idade e um número de cartão de cidadão. Esta pode ter um contrato associado.

**Player –** Representa um jogador de uma equipa. Possui um nome, uma idade, um número de cartão de cidadão, uma altura, um peso, uma posição de jogo e um número de equipamento. Este possui uma estatística média.

**Coach –** Representa um treinador de uma equipa. Possui um nome, uma idade um número de cartão de cidadão.

**Contract –** Representa um contrato assinado por um treinador ou jogador ou pessoa que será presidente de uma equipa. Possui um ID, descrição, salário, data de início e data de fim.

**Stadium –** Representa um estádio. Possui um ID, localização, capacidade e nome. Este pertence a uma equipa e nele são realizados vários jogos.

**Average Individual Numbers –** Representa a média da estatística individual de um jogador. Esta possui pontos, abafos, roubos de bola, assistências, ressaltos, percentagem de triplo e percentagem de lançamentos de campo.

**Game –** Representa um jogo entre duas equipas (visitada e visitante). Possui um ID, hora, data e pontos da equipa visitada e da visitante. Este é realizado num estádio.

**Ticket –** Representa os bilhetes que estão a ser vendidos por uma equipa para um determinado jogo. Possui um tipo, um preço e um número de bilhetes restantes.

# **Diagramas**

## **Diagrama Entidade-Relacionamento**

**Uma imagem com diagrama, file, esboço, origami

Descrição gerada automaticamente** Abaixo está ilustrado o diagrama entidade-relacionamento correspondente às entidades referidas acima e relações entre as mesmas.

## 

## **Esquema Relacional**

Uma imagem com texto, recibo, Tipo de letra, file

Descrição gerada automaticamente Abaixo está ilustrado o esquema relacional de acordo com o diagrama entidade-relacionamento desenvolvido.

## **Esquema Relacional (SGDB)**

Uma imagem com diagrama, Esquema, Desenho técnico, file

Descrição gerada automaticamente Abaixo está ilustrado o esquema relacional criado pelo SGBD utilizado.

# **Queries SQL**

## **Data Definition Language (DDL)**

## **Data Manipulation Language (DML)**

## **Views**

## **Indexes**

## **User Definition Functions (UDF’s)**

## **Stored Procedures**

## **Triggers**