

**Universidade do Minho**

Licenciatura em Ciências da Computação

**Unidade Curricular de**

**Bases de Dados**

Ano Letivo de 2024/2025

**Gestão de um Torneio Multimodalidades Com Base em Desportos de Raquetes**

**José loureiro (A96467),**

**Pedro Gonçalves (A101250),**

**Nuno Ni (A102503)**,

**Eduardo Vilaça (A102518),**

Novembro, 2024

<<Mês, Ano>>

**BD**

|  |  |
| --- | --- |
| Data de Recepção |  |
| Responsável |  |
| Avaliação |  |
| Observações |  |

**Gestão de um Torneio Multimodalidades Com Base em Desportos de Raquetes**

**José loureiro (A96467),**

**Pedro Gonçalves (A101250),**

**Nuno Ni (A102503)**,

**Eduardo Vilaça (A102518),**

Novembro, 2024

# Resumo

No presente trabalho, abordamos a análise, planeamento, modelação, arquitetura e implementação de um Sistema de Base de Dados (SBD) relacional, com foco na gestão organização de um torneio multimodalidades com base em desportos de raquetes. A pesquisa começa com uma contextualização do projeto, ressaltando a sua relevância no contexto da gestão de eventos urbanos. Fundamentamos o nosso estudo em conceitos essenciais de bases de dados relacionais.

Os nossos objetivos abrangem a definição, motivação e viabilidade do SBD, com ênfase nas vantagens operacionais que o sistema proporcionará. Detalhamos os recursos necessários, a equipa de trabalho envolvida e um plano de execução do projeto, estabelecendo marcos e cronogramas.

Na etapa seguinte, definimos os requisitos do SBD. Utilizamos um método de levantamento e análise de requisitos, organizando-os em categorias de descrição, exploração e controle. Validamos rigorosamente esses requisitos para garantir sua clareza e detalhamento. A fase subsequente é dedicada à modelação conceitual, na qual criamos o esquema conceitual da base de dados. Apresentamos a abordagem de modelação utilizada, identificamos e caracterizamos as entidades, relacionamentos e atributos envolvidos no sistema. No estado atual do projeto, todas as etapas até a modelação conceitual foram concluídas com êxito.

**Área de Aplicação:** Desenvolvimento e implementação de Sistemas de Bases de Dados

**Palavras-Chave:** Área de Aplicação: Desenvolvimento e implementação de Sistemas de Bases de Dados Bases de Dados, Bases de Dados Relacionais, Bases de Dados Não Relacionais, Análise de Requisitos, Modelo Lógico, Modelo Conceptual, Entidades, Atributos, Normalização, MySQL Workbench, SQL.

# Índice

1. [Introdução 1](#_bookmark1)
   1. [Apresentação do Caso de Estudo 1](#_bookmark2)
   2. [Contextualização 1](#_bookmark3)
   3. [Motivação e Objetivos 2](#_bookmark4)
   4. [Análise da Viabilidade do processo 3](#_bookmark5)
   5. [Recursos 3](#_bookmark6)
   6. [Equipa de Trabalho 3](#_bookmark7)
   7. [Plano de Execução 4](#_bookmark8)
   8. [Estrutura do Relatório 5](#_bookmark11)
2. [Levantamento e Análise de Requisitos 7](#_bookmark12)
   1. [Método de Levantamento e de Análise de Requisitos Adotado 7](#_bookmark13)
   2. [Organização dos Requisitos Levantados 8](#_bookmark14)
   3. [Análise e Validação Geral dos Requisitos 8](#_bookmark16)
3. [Modelação Conceptual 9](#_bookmark17)
   1. [Apresentação da Abordagem de Modelação Realizada 9](#_bookmark18)
   2. [Identificação e Caracterização das Entidades 9](#_bookmark19)
   3. [Identificação e Caracterização dos Relacionamentos 10](#_bookmark21)
   4. Identificação e Caracterização dos Atributos das Entidades e dos Relacionamentos 11
   5. [Apresentação e Explicação do Diagrama ER Produzido 20](#_bookmark33)
4. [Modelação Lógica 21](#_bookmark35)
   1. [Construção e Validação do Modelo de Dados Lógico 21](#_bookmark36)
   2. [Apresentação e Explicação do Modelo Lógico 22](#_bookmark38)
   3. [Normalização de Dados 24](#_bookmark39)
   4. [Validação do Modelo com Interrogações do Utilizador 31](#_bookmark40)
5. Implementação Física 24

5.1 Apresentação do processo de criação da base de dados 27

5.2 Cálculo do espaço da base de dados (inicial e taxa de crescimento anual) 30

5.3 Realização do povoamento da base de dados 31

5.4 Tradução das interrogações do utilizador para SQL 32

5.5 Definição e caracterização das vistas de utilização em SQL 33

5.6 Definição dos perfis de utilização para cada utilizador da base de dados 34

5.7 Indexação do Sistema de Dados 35

5.8 Procedimentos Implementados 36

5.9 Plano de segurança e recuperação de dados 37

1. Conclusões e Trabalho Futuro 39
2. Bibliografia 40

**Índice de Figuras**

Figura 1 - Diagrama de Gantt 3

Figura 2 - Modelo Conceptual 18

Figura 3 - Modelo Lógico 19

Figura 4 - Exemplo de Relação 1:N 21

Figura 5 Exemplo de Auto Relacionamento 22

Figura 6 - Exemplo de Relacionamento N:N 22

Figura 7 - Query 1 23

Figura 8 - Query 2 23

Figura 9 - Query 3 24

*Figura 11 – Criação e Utilização da base de dados nomeadas “Torneio\_EOR”* 25

*Figura 12* ***-*** *Tabela Entidade SQL* 26

*Figura 14* ***-*** *Tabela Contacto em SQL* 27

*Figura 15* **-***Custo de Memória SQL* 28

*Figura 16* *- Custo Tabelas Torneio\_EOR*  29

# Índice de Tabelas

Tabela 1 - Requisitos de descrição sobre o Torneio 5

Tabela 2 - Requisitos de descrição sobre o Staff 5

Tabela 3 - Requisitos de descrição sobre os Campos 6

Tabela 4 - Requisitos de descrição sobre os Atletas 6

Tabela 5 - Requisitos de descrição sobre Partidas e Modalidades 6

Tabela 6 - Requisitos de Manipulação 7

Tabela 7 - Requisitos de Controlo 8

Tabela 8 - Caracterização dos relacionamentos 11

Tabela 9 - Atributos da entidade Torneio 12

Tabela 10 - Atributos da entidade Partida 12

Tabela 11 - Atributos da entidade Atleta 13

Tabela 12 - Atributos da entidade Set 13

Tabela 13 - Atributos da entidade Game 13

Tabela 14 - Atributos da entidade Tiebreaker 14

Tabela 15 - Atributos da entidade Resultado 14

Tabela 16 - Atributos da entidade Campo 14

Tabela 17 - Atributos da entidade Staff 15

Tabela 18 - Atributos da entidade Entrada 15

Tabela 19 - tributos da entidade Modalidade 16

Tabela 20 - Atributos da entidade Pais 16

Tabela 21 - Atributos da entidade Morada 16

1. Introdução
   1. Contextualização

A organização europeia de raquetes (EOR) com sede em Berlim, foi criada em 1984 por Bud Freidman, antigo jogador britânico de ténis e Naomi Bishoff , a renomeada campeã olímpica e pós-presidente do comité alemão de ténis de mesa, propondo organizar e promover eventos e torneios de desportos que servem da utilização de raquetes a nível europeu.

Com isto, a entidade que apenas organizava torneios das modalidades baseadas nos desportos dos fundadores, com o objetivo de crescer e acompanhar as novas tendências das próximas gerações nesta área do desporto, tem vindo a incluir modalidades como badminton, squash e a nova *trend* o *padel*.

Neste momento, a EOR é uma das principais organizadoras de torneios e eventos em torno de raquetes a nível europeu. Com o passar dos anos, o constante crescimento da organização tem levado, perante os novos desafios da atualidade, à necessidade de possuir vários representantes dos inúmeros países europeus.

* 1. Fundamentação

Desde a sua fundação, a organização tem estruturado e realizado diversos eventos e respetivos torneios para os seus desportos, mas em 2024, numa tentativa de combater o declínio da popularidade das suas modalidades e do desporto a tanto nível europeu como mundial, o comité anunciou que vai organizar um torneio, com uma proporção nunca antes vista, de caráter europeu que incluía todas as suas atividades, com objetivo de aumentar a visibilidade das modalidades, assim como, o interesse do público e finalmente, praticantes.

Apesar de já possuir um sistema de base de dados, com a finalidade de organizar e guardar a informação das várias competições realizadas, sendo que esta será a primeira vez onde possuem o desafio de organizar um torneio multimodalidade, o comité decidiu que a melhor abordagem para este tipo de adversidade passará por implementar um novo e melhorado sistema de base de dados, de modo, a auxiliar na organização de eventos desta vigente magnitude.

* 1. Objetivos

De acordo com os novos desafios da preparação deste evento, o comité estabeleceu vários objetivos que têm de ser satisfeitos pelo novo sistema de base de dados, sendo estes:

* + Organizar o seu modelo de Torneio, bem como melhorar a sua capacidade de gestão e registo dados tanto dos atletas como dos jogos;
  + Divulgar o torneio de forma a abranger novos espetadores com a utilização plataformas *WEB;*
  + Registos dos espetadores no torneio;
  + Facilitar o acesso das informações em tempo real dos jogos ao público;
  + Dar a conhecer novos atletas dos diversos desportos ao público, tanto no evento presencial, como no site do torneio;
  + Permitir ao staff possua informação dos jogos já efetuados, que estejam a decorrer e dos jogos em espera;
  + Armazenar informações relevantes das partidas, como resultados, sets e pontos para mais tarde servirem de base para estatísticas;
  + Armazenar e organizar a informação dos patrocinadores.
  1. Viabilidade

O comité da EOR acredita que com este evento e com o sistema de base de dados implementado será possível aumentar a popularidade e a capacidade de espetadores presenciais no evento e diminuir os custos de administração do torneio:

* Aumentar a realização de novos torneios neste formato;
* Ampliar a eficiência da staff;
* É esperado aumentar a quantidade de espetadores por 16,9% obtendo assim maior lucro;
* Ajudar disponibilizar resultados dos jogos em diversas plataformas;
* Expor estatísticas acerca dos atletas, para despertar o interesse do público;
* Diminuir os custos de organização do evento em 11,2%;
* Levantar a atenção de novos patrocinadores e patrocínios;
* Aumentar a gama de países europeus anfitriões de torneio da EOR.

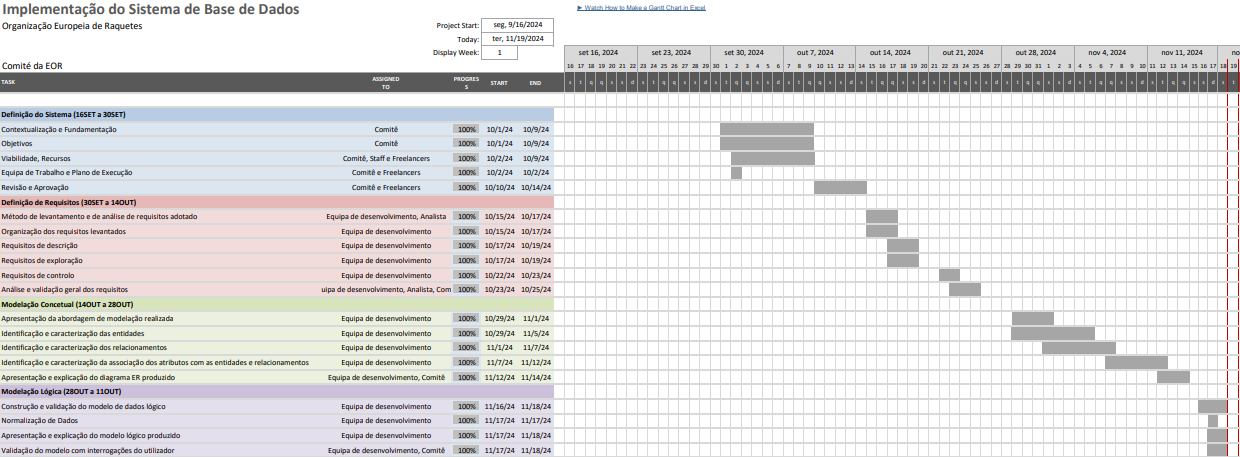
* 1. Recursos
* **Humanos**: Pessoal do staff (organizadores, analistas, árbitros, voluntários) atletas, espetadores, segurança, equipa médica, equipa de desenvolvimento da base de dados;
* **Hardware**: Servidor para alojar a base de dados, 1 computador por campo, estação de computadores para a organização;
* **Software**: SGBD (*mySQLWorkbench*), aplicação de gestão de informação sobre o evento.
  1. Equipa de Trabalho
* **Pessoal Interno:** Staff dentro de campo, responsável pela atualização dos resultados e validação. Staff fora de campo, encarregue de introduzir os dados sobre os jogadores e informações gerais do evento para a plataforma.
* **Pessoal Externo:** Equipa de implementação do sistema de base dados, pessoal de teste, supervisores da plataforma.
* **Outros:** Comité da EOR para fase de testes e supervisão do evento.
  1. Plano de execução do trabalho

Figura 1 - Diagrama de Gantt

1. Definição de Requisitos
   1. Método de levantamento de Requisitos

O processo onde se inicia a seleção dos métodos de levantamento de forma a sustentar a angariação e a organização dos requisitos. Nesse sentido, decidimos que íamos fazer as seguintes atividades:

* **Entrevistas/Reuniões:** Realizar reuniões com os organizadores e membros da staff responsáveis pela organização de Torneios anteriores para tentar entender o seu papel e compreender os seus processos operacionais nos diferentes contextos profissionais. Estas entrevistas/reuniões serão importantes uma vez que os profissionais podem sugerir/indicar ideias que um SGBD os auxiliaria no seu trabalho profissional;
* **Observações:** Ao observar como a equipa da staff realizaram as suas funções durante os Torneios anteriores, podemos tentar observar falhas ou detalhes despercebidos pelos mesmos que poderiam ser corrigidos ou melhorados, levando a um melhor desempenho profissional e maior eficiência para Torneios de maior proporção;
* **Questionários/Inquéritos**: Ao realizar estes questionários ou inquéritos aos membros da staff, podemos entender melhor o que podemos melhorar no SBD, como por exemplo falhas que a organização teve por possivelmente a ausência de uma base de dados anterior ou problemas que uma base de dados resolveria em futuros Torneios;
* **Análise da Documentação:** Ao aceder a informações sobre os Torneios anteriores fornecida pela organização, podemos entender como os dados estão organizados e auxiliar a equipa informática no seu Sistema de Base de Dados durante a sua implementação.
* **Investigação:** estudar outras organizações desenvolvedoras de Torneios a grande escala para poder identificar possíveis soluções ou abordagens mais eficientes;
  1. Organização dos requisitos levantados
     1. **Requisitos de descrição**

Tabela 1 - Requisitos de descrição sobre o Torneio

|  |  |  |  |
| --- | --- | --- | --- |
| Nº | Descrição | Objetivo | Fonte |
| RD1 | O Torneio é constituído por um id, edição e designação. | Permitir a reutilização da BD para várias edições | Documentação |
| RD2 | O torneio possui várias partidas | Associar as partidas a um torneio | Documentação |

Tabela 2 - Requisitos de descrição sobre o Staff

|  |  |  |  |
| --- | --- | --- | --- |
| Nº | Descrição | Objetivo | Fonte |
| RD3 | Um elemento do staff é composto por id, nome, contacto, data de nascimento, sexo, email, área de trabalho | Organizar informação do staff | Documentação |
| RD4 | Elementos do staff podem ter um chefe. | Organizar informação do staff | Documentação |
| RD5 | O contacto é constituído por id do staff a quem pertence e número de telemóvel. | Organizar informação do staff | identificado pelos organizadores |
| RD6 | Um elemento do staff pode ter mais do que um contacto | Organizar informação do staff | Documentação |
| RD7 | Os elementos do staff têm morada | Organizar informação do staff | Observação pela staff |
| RD8 | Cada morada pode estar associada a mais do que um elemento da staff | Organizar informação do staff | Documentação |
| RD9 | Uma morada é constituída por id, rua, número da porta, código postal e cidade e país. | Organizar informação das moradas | Documentação |
| RD10 | As funções do staff são: árbitro, apanha bolas, supervisor da partida, manutenção, organizador. | Diferenciar os elementos do staff consoante as suas funções | Documentação |
| RD33 | Um contacto apenas pode pertencer a um único elemento do staff | Organizar informação das moradas | Documentação |

Tabela 3 - Requisitos de descrição sobre os Campos

|  |  |  |  |
| --- | --- | --- | --- |
| Nº | Descrição | Objetivo | Fonte |
| RD11 | Os campos são constituídos por id, modalidade, lotação | Organizar informação dos campos | Documentação |
| RD12 | Os campos têm uma localização | Organizar informação das moradas | Documentação |
| RD13 | Alguns campos podem estar associados a mesma localização | Organizar informação das moradas | Documentação |
| RD14 | Um campo apenas recebe partidas de uma modalidade | Certificar que não se realizam partidas de uma modalidade diferente da do campo | Documentação |

Tabela 4 - Requisitos de descrição sobre os Atletas

|  |  |  |  |
| --- | --- | --- | --- |
| Nº | Descrição | Objetivo | Fonte |
| RD15 | O atleta tem um id, nome, idade, sexo, nacionalidade, ranking, modalidade | Organizar informação dos atletas | Pedido pelos organizadores |
| RD31 | Os atletas só podem participar em partidas da sua modalidade | Assegurar que os atletas só participam na sua modalidade | Documentação |
| RD32 | Os atletas podem participar em ambas os tipos de uma modalidade (individual, duplas) | Permitir a participação dos atletas em ambos os formatos da sua categoria | Pedido pelos organizadores |

Tabela 5 - Requisitos de descrição sobre Partidas e Modalidades

|  |  |  |  |
| --- | --- | --- | --- |
| Nº | Descrição | Objetivo | Fonte |
| RD16 | Uma modalidade é composta por id, designação. | Organizar informação das modalidades | Documentação |
| RD17 | As partidas têm um campo | Associar os campos às respetivas partidas | Documentação |
| RD18 | Uma partida é constituída por um id, data e hora, modalidade, entrada A, entrada B, ronda, id do torneio | Organizar informação constitucional das partidas | Documentação |
| RD19 | Uma partida tem duas entradas | Armazenar os oponentes | Documentação |
| RD20 | Ambas as entradas de uma partida têm atletas da modalidade da partida | Certificar que não há partidas entre entradas de diferentes modalidades | Documentação |
| RD21 | Uma entrada é constituída por 1 ou 2 atletas (entrada individual ou entrada de duplas) | Constituição de uma entrada | Documentação |
| RD22 | Numa entrada de duplas os atletas têm de ter a mesma modalidade | Constituição de uma entrada em duplas | Documentação |
| RD23 | Uma partida tem associado vários membros de staff | Associar staff à partida | Documentação |
| RD24 | A partida é composta por sets | Armazenar pontuação de uma partida | Documentação |
| RD25 | O set é composto por games e tie-breaks | Elementos de um set | Documentação |
| RD26 | O set é constituído po número do set, games do jogador 1, games do jogador 2 | Organizar informações de um set | Documentação |
| RD27 | O game é composto pelo número do set, número do game, pontos do jogador 1, pontos do jogador 2 | Organizar informações de um game | Documentação |
| RD28 | O tie-break é constituído pelo número do set, pontos do tie-break do jogador 1, pontos do tie-break do jogador 2 | Organizar informações de um tiebreaker caso este ocorra | Documentação |
| RD29 | A partida fica decidida por um resultado | Armazenar resultado de uma partida | Documentação |
| RD30 | O resultado é composto pelo vencedor, número de sets jogados e duração. | Composição de um resultado | Documentação |
| RD35 | Cada partida tem associado no mínimo um arbitro | Restrição mínima para ocorrer uma partida | Documentação |
| RD36 | Cada partida tem associado, no mínimo quatro apanha bolas | Restrição mínima para ocorrer uma partida | Documentação |

* + 1. **Requisitos de manipulação**

Tabela 6 - Requisitos de Manipulação

|  |  |  |  |
| --- | --- | --- | --- |
| Nº | Descrição | Objetivo | Fonte |
| RM1 | Todos os elementos do staff têm acesso à visualização dos seus dados pessoais | Transparência dos dados do staff | Documentação |
| RM2 | Deve ser possível gerir os campos | Gestão de dados na BD | Pedido pelos organizadores |
| RM3 | Deve ser possível gerir os atletas | Gestão de dados na BD | Documentação |
| RM4 | Deve ser possível ver o histórico de partidas | Listar informação sobre qualquer partida do torneio | Documentação |
| RM5 | Deve ser possível gerir as partidas | Gestão de dados na BD | pedido pelos organizadores |
| RM7 | Deve ser possível atualizar a pontuação de cada set | Gestão de dados na BD | Documentação |
| RM8 | Deve ser possível atualizar a pontuação de cada game | Gestão de dados na BD | Documentação |
| RM9 | Deve ser possível atualizar as partidas com os tie-breaks para as modalidades que necessitam | Gestão de dados na BD | Documentação |
| RM10 | Deve ser possível ver a duração da partida | Planeamento mais eficiente | Documentação |
| RM11 | Deve ser possível ver a duração da partida mais longa | Estatística para futuro planeamento de torneios | Pedido pelos organizadores |
| RM12 | Deve ser possível gerir o staff | Gestão de dados na BD | Documentação |
| RM13 | Deve ser possível listar o nome dos atletas que praticam uma certa modalidade | Ajuda na organização do evento | Pedido pelos organizadores |
| RM14 | Deve ser possível saber o número de vitórias de um jogador específico | Estatística para futuro planeamento de torneios | Pedido pelos organizadores |
| RM15 | Deve ser possível visualizar os membros do staff que participaram numa certa partida | Ajuda na organização do evento | Pedido pelos organizadores |

* + 1. **Requisitos de Controlo**

Tabela 7 - Requisitos de Controlo

|  |  |  |  |
| --- | --- | --- | --- |
| Nº | Descrição | Objetivo | Fonte |
| RC1 | Apenas um supervisor da partida pode gerir a pontuação das partidas | Para evitar manipulações indevidas | Pedido pelos organizadores |
| RC2 | Apenas os organizadores podem adicionar, remover ou alterar dados de atletas | Para evitar manipulações indevidas | Documentação |
| RC3 | Apenas os organizadores podem adicionar, remover ou alterar dados de elementos do staff | Para evitar manipulações indevidas | Documentação |
| RC4 | A informação da SBD só pode ser alterada conforme a duração do torneio | Proteger informação para a sua utilização futura | Pedido pelos organizadores |
| RC5 | Apenas os organizadores podem adicionar, remover Torneios | Para evitar manipulações indevidas | Documentação |
| RC6 | Apenas os organizadores podem adicionar, remover Campos | Para evitar manipulações indevidas | Documentação |
| RC7 | Os organizadores estão encarregues de associar elementos da staff a cada partida | Organizar a distribuição de tarefas para maior eficiência entre os membros do staff | Pedido pelos organizadores |

* 1. Análise e validação geral dos requisitos

O processo de análise dos requisitos foi segmentado em três etapas distintas: requisitos de descrição, requisitos de manipulação, requisitos de controlo. Cada uma dessas etapas foi examinada em reuniões de grupo, onde os membros contribuíram com suas perspetivas e sugestões. A primeira fase, que abordou os requisitos de descrição, o foco foi identificar os principais tópicos necessários para o funcionamento e realização do Torneio. As descrições que poderiam dar algum benefício à organização e aos seus respetivos membros da staff.

Na segunda fase, dedicada aos requisitos de manipulação, foram mapeadas as informações críticas que deveriam ser obtidas do sistema para ajudarem no objetivo final neste tipo de Torneio. Finalmente, na fase de requisitos de controlo, estabelecemos as restrições da base de dados.

1. Modelação Conceptual
   1. **Apresentação da abordagem de modelação realizada**

A modelação conceptual construída neste projeto, através da ferramenta *brModelo*, segue a esquematização da modelação ER (Entidade-Relacionamento), utilizando a notação de Chen, caracterizando entidades, relacionamentos e atributos que estejam associados aos mesmos. Na construção deste modelo começamos pela criação das entidades, identificando as mesmas e definindo os seus atributos, de seguida estabelecemos os relacionamentos entre as estas. A criação deste modelo conceptual ajudou a visualizar e compreender a estrutura da base de dados, auxiliando a sua implementação assim como deteção de erros antes desta ser implementada.

* 1. **Identificação e caracterização das entidades**
* **Torneio**

A entidade Torneio armazena as informações gerais de um torneio e tem associada as partidas realizadas no mesmo.

* **Partida**

A entidade Partida é essencial para a base de dados pois regista detalhadamente as pontuações e informações relativas a cada partida efetuada no torneio, permitindo assim a gestão dos resultados e vencedores de cada modalidade.

* **Atleta**

Na entidade Atleta é feito o armazenamento dos dados de cada atleta que vai participar no torneio, auxiliando assim na gestão do mesmo

* **Entrada**

A entidade Entrada é responsável por armazenar os jogadores correspondentes a cada entrada do torneio, esta pode ser constituída por um ou dois atletas, permitindo assim a distinção entre duplas e jogadores individuais assim como dar a flexibilidade para jogadores poderem participar em ambos os formatos das partidas.

* **Resultado**

A entidade Resultado é responsável por decidir o vencedor de uma partida, assim como registar alguns dados extra sobre a mesma

* **Modalidade**

Em Modalidade armazenamos todas as modalidades presentes no torneio

* **Set**

Set regista o resultado de um set de uma partida

* **Game**

Game regista o resultado de um Game pertencente a um Set de uma partida

* **Tiebreaker**

Em caso de ocorrer um Tiebreaker em um dado set, Tiebreaker regista o resultado do mesmo

* **Staff**

A entidade Staff armazena os dados de cada elemento do staff, auxiliando a gestão dos recursos humanos disponíveis no torneio, facilitando a sua organização

* **Campo**

Esta entidade é responsável por armazenar a informação dos campos onde serão realizadas as partidas do torneio

* **Morada**

A entidade Morada auxilia no registo das moradas dos elementos da staff, assim como a localização de cada campo

* **País**

Esta entidade armazena vários países

* 1. **Identificação e caracterização dos relacionamentos**

Tabela 8 - Caracterização dos relacionamentos

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Entidade | Cardinalidade | Relacionamento | Cardinalidade | Entidade | Requisito |
| Partida | (1,n)  (1,n)  (1,n)  (0,n)  (0,n)  (0,n)  (1,1)  (0,n) | entradaA  entradaB  realiza  tem  é jogada  participa  contêm  decide | (1,1)  (1,1)  (1,1)  (1,1)  (1,1)  (1,n)  (1,n)  (1,1) | Entrada  Entrada  Torneio  Modalidade  Campo  Staff  Set  Entrada |  |
| Set | (1,1)  (1,1) | contêm  pode ter | (1,n)  (0,1) | Game  Tiebreaker |  |
| Staff | (0,1)  (1,n) | é chefe  possui | (1,n)  (1,1) | Staff  Morada |  |
| Campo | (1,n)  (1,n) | localiza-se  tem | (1,1)  (1,1) | Morada  Modalidade |  |
| Atleta | (1,n)  (0,n)  (1,n) | participa  representa  joga | (0,n)  (1,1)  (1,1) | Entrada  País  Modalidade |  |
| País | (1,1)  (1,1) | tem  realiza-se | (0,n)  (0,n) | Morada  Torneio |  |

* 1. **Identificação e caracterização da associação dos atributos com as entidades e relacionamentos**
     1. **Entidade – Torneio**

Tabela 9 - Atributos da entidade Torneio

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| idTorneio | Identifica as partidas com um id único | INT | Não | Não | Sim | 1  2 |
| designação | Nome do torneio | TEXT | Não | Não | Não | Raquetadas no Estoril  Grande Torneio da Póvoa |
| edição | Identifica a edição do torneio | INT | Não | Não | Não | 20  5 |

* + 1. **Entidade – Partida**

Tabela 10 - Atributos da entidade Partida

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| idPartida | Identifica as partidas com um id único | INT | Não | Não | Sim | 1  2 |
| dataInicio | Guarda a data e hora do começo da partida | DATETIME | Não | Não | Não | 12/10/2024 12:30:00  22/08/2024 09:00:00 |
| ronda | Identifica a fase em que a partida está a decorrer | TEXT | Não | Não | Não | Final  Quartos de Final |

* + 1. **Entidade – Atleta**

Tabela 11 - Atributos da entidade Atleta

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| idAtleta | Identifica as partidas com um id único | INT | Não | Não | Sim | 1  2 |
| nome | Nome do Atleta | TEXT | Não | Não | Não | Danill Medvedev  João Sousa |
| idade | Idade do Atleta | INT | Não | Não | Não | 23  36 |
| sexo | Sexo do Atleta | TEXT | Não | Não | Não | M  F |
| Ranking | Posição no ranking mundial referente à modalidade deste | INT | Sim | Não | Não | 12  115 |

* + 1. **Entidade – Set**

Tabela 12 - Atributos da entidade Set

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| nrSet | Identifica o número do set a jogar | INT | Não | Não | Sim | 1  2 |
| gamesEntradaA | Identifica número de games vencidos da entrada A | INT | Não | Não | Não | 3  2 |
| gamesEntradaB | Identifica número de games vencidos da entrada B | INT | Não | Não | Não | 0  7 |

* + 1. **Entidade – Game**

Tabela 13 - Atributos da entidade Game

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| nrGame | Identifica o número do game associado ao set a jogar | INT | Não | Não | Sim | 1  2 |
| pontosEntradaA | Identifica número de pontos da entrada A relativos ao game | INT | Não | Não | Não | 30 |
| pontosEntradaB | Identifica número de pontos da Entrada B relativos ao game | INT | Não | Não | Não | 15 |

* + 1. **Entidade – Tiebreaker**

Tabela 14 - Atributos da entidade Tiebreaker

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| tiebreakersEntradaA | Guarda informação se houver tiebreaker num determinado set | INT | Não | Não | Não | 5  7 |
| tiebreakersEntradaB | Guarda informação se houver tiebreaker num determinado set | INT | Não | Não | Não | 2  6 |

* + 1. **Entidade – Resultado**

Tabela 15 - Atributos da entidade Resultado

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| totalSets | Número total de sets jogados na partida | INT | Não | Não | Não | 2  5 |
| duração | Tempo em minutos da duração da partida | INT | Não | Não | Não | 120  60 |

* + 1. **Entidade – Campo**

Tabela 16 - Atributos da entidade Campo

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| idCampo | Identifica o campo com um número único. | INT | Não | Não | Sim | 2  1 |
| designação | Nome do campo | TEXT | Não | Não | Não | Campo de Tênis de Real  Campo de Tênis de Silves |
| lotação | Quantidade de lugares disponíveis no campo | INT | Não | Não | Não | 1200  5000 |

* + 1. **Entidade – Staff**

Tabela 17 - Atributos da entidade Staff

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| idStaff | Identifica um membro do staff com um número único. | INT | Não | Não | Sim | 2  1 |
| nome | Nome do funcionário | TEXT | Não | Não | Não | Nuno Ni  Ni Nuno |
| areaTrabalho | Designação da área de trabalho onde o membro do staff trabalha | TEXT | Não | Não | Não | Apanha bolas  Árbitro |
| dataNascimento | Guarda a data de nascimento do membro do staff | Date | Não | Não | Não | 12/02/2004  22/07/1999 |
| contacto | Número de telefone para possível contacto | INT | Não | Sim | Não | 917623123  253142156 |
| email | Email para contacto | TEXT | Não | Não | Não | [eor@gmail.com](mailto:eor@gmail.com)  arbitro@hotmail.com |
| genero | Identifica o género do membro do staff | TEXT | Não | Não | Não | Feminino  Masculino |

* + 1. **Entidade – Entrada**

Tabela 18 - Atributos da entidade Entrada

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| idEntrada | Atribui um identificador único á entrada | INT | Não | Não | Sim | 5  7 |
| tipo | Identifica o tipo da entrada | TEXT | Não | Não | Não | duplas  singulares |

* + 1. **Entidade – Modalidade**

Tabela 19 - Atributos da entidade Modalidade

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| idModalidade | Identifica a modalidade com um número único | INT | Não | Não | Sim | 2  1 |
| designação | Nome da modalidade | TEXT | Não | Não | Não | Tênis  Padel |

* + 1. **Entidade – País**

Tabela 20 - Atributos da entidade Pais

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| idPais | Identifica o país com um identificador único | INT | Não | Não | Sim | 30  22 |
| Pais | Nome do país | TEXT | Não | Não | Não | Portugal  Rússia |

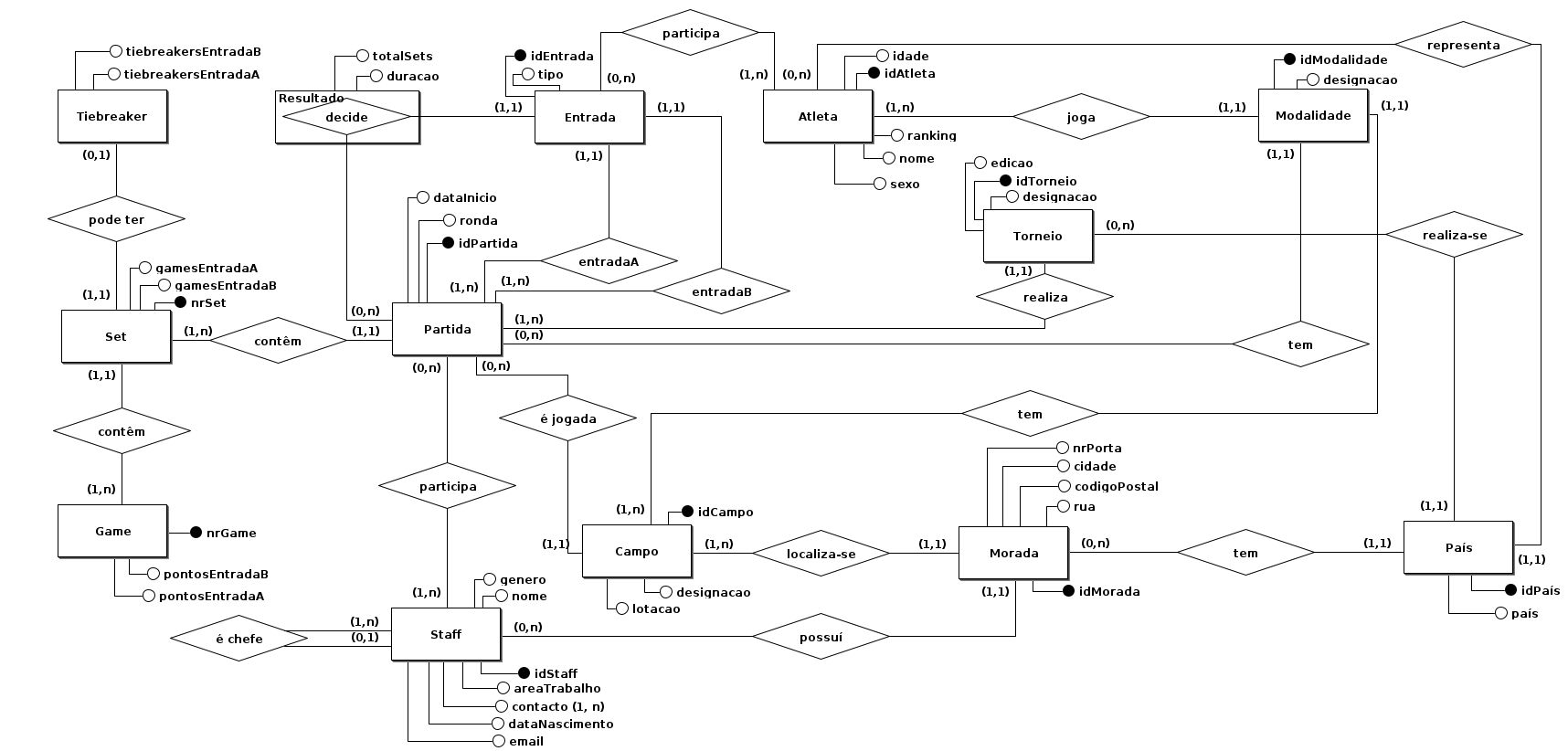
* + 1. **Entidade – Morada**

Tabela 21 - Atributos da entidade Morada

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Atributos | Descrição | Tipo | Nulo | Multi-Valor | Chave | Exemplos |
| idMorada | Identifica a morada com um número único. | INT | Não | Não | Sim | 5  7 |
| rua | Nome da rua relativa á morada | TEXT | Não | Não | Não | Rua de Cima  Rua de Baixo |
| codigoPostal | Código Postal associado á morada | TEXT | Não | Não | Não | 4700-290  4705-306 |
| Cidade | Identificação da cidade referente á morada | TEXT | Não | Não | Não | Braga  Lisboa |
| nrPorta | Número da porta. | INT | Não | Não | Não | 1  3 |

* 1. Apresentação e explicação do diagrama ER produzido

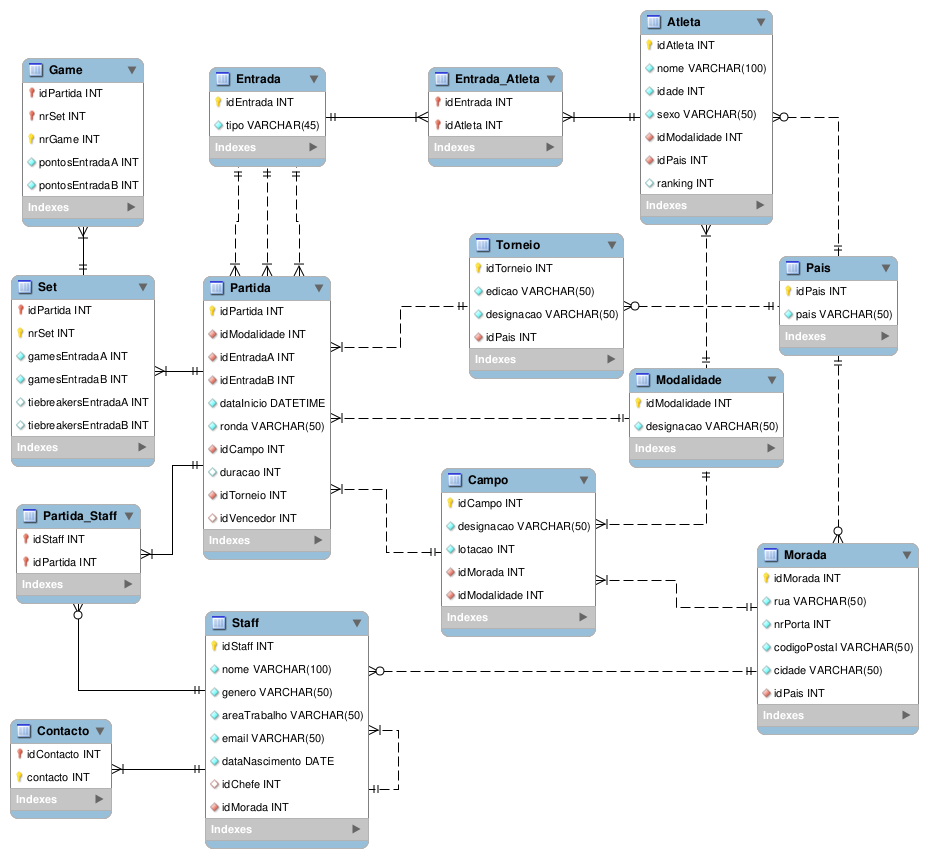
Figura 2 - Modelo Conceptual



Em equipa e após analise, chegamos á conclusão de que este é o diagrama que melhor descreve o sistema que queremos implementar, obedecendo a todos os requisitos estabelecidos pela EOR.

1. Modelação Lógica
   1. **Construção e validação do modelo de dados lógico**

Figura 3 - Modelo Lógico



Com base no Modelo Conceptual foi criado o Modelo Lógico da Figura 2. Neste modelo podemos visualizar as tabelas e as suas respetivas variáveis assim como as relações estabelecidas entre elas. Nos tópicos seguintes iremos explicar o processo de criação deste modelo lógico dando exemplos individuais para a explicação da abordagem tomada no que toca à conversão das relações do modelo conceptual para o modelo lógico, a sua normalização e utilizando álgebra relacional alguns exemplos de queries da nossa base de dados.

* 1. **Normalização de Dados**

Ao longo do processo de criação do nosso modelo lógico, efetuamos a normalização dos dados de modo a garantir que a base de dados seja otimizada, estruturada e evite redundâncias nos seus dados.

De modo a efetuar este processo de normalização seguimos a 3 principais regras de normalização (1FN, 2FN, 3FN), estas regras ajudaram a sucessivamente transformar o nosso modelo lógico para um estado menos vulnerável e mais organizado.

Sendo assim começamos por:

* **1FN:** Estruturamos as nossas relações de modo a não terem dados que possam ser considerados redundantes (repetidos) e garantindo que todos os seus atributos são atómicos.
* **2FN:** Colocamos chaves primárias em todas as nossas tabelas e certificamo-nos que todos os atributos não-primos são totalmente dependentes da sua respetiva chave primária.
* **3FN:** Garantimos que todos os atributos que não sejam chaves primárias sejam independentes entre si, eliminando assim a existência de dependências funcionais transitivas.
  1. **Apresentação e Explicação do Modelo Lógico**

Nesta secção iremos explicar o raciocínio do grupo no processo de conversão do modelo conceptual para o modelo lógico, abordando os métodos utlizados para os diferentes tipos de relações entre tabelas do modelo concetual.

* **Relacionamentos 1:N**

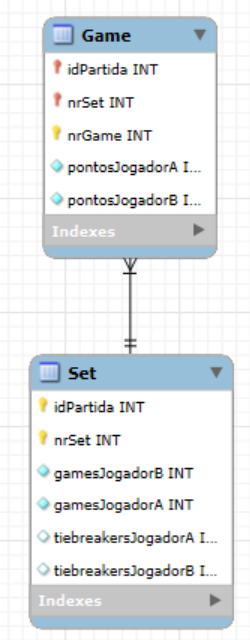
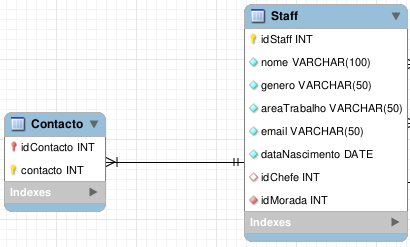
****Para converter as relações de 1:N do modelo conceptual para modelo lógico, adicionamos uma Chave Estrangeira à tabela da entidade que necessita da informação da outra tabela. Na seguinte figura podemos observar o relacionamento entre as entidades **Set** e **Game** (1:N), onde um set é constituído por vários Games mas os Games apenas têm um Set, deste modo é adicionado a Game a chave estrangeira nrSet de Set para identificarmos a que Set pertence cada Game.

Figura 4 - Exemplo de Relação 1:N

* **Multivalorados**

Os atributos multivalorados ganharam uma nova tabela, contendo como chave primária uma chave estrangeira com o id a quem este atributo pertence.

Figura 5 - Exemplo de atributos multivalorados



* **Auto-Relacionamentos**

Na conversão de auto-relacionamentos adicionamos à tabela uma chave estrangeira que está ligada à chave primária da própria tabela. Como podemos observar na figura 4, a chave estrangeira idChefe está ligada à chave primária idStaff, deste modo também é necessário tornar esta chave estrangeira nula.

Uma imagem com texto, captura de ecrã, Tipo de letra, número

Descrição gerada automaticamenteFigura 6 Exemplo de Auto Relacionamento

* **Relacionamentos N:N**

Tendo em conta a maior complexidade deste tipo de relações, houve necessidade de criar uma tabela extra que se posiciona entre a relação das duas entidades. Como podemos ver na figura, com esta nova tabela passamos a ter duas relações 1:N entre as entidades **Atleta** e **Entrada\_Atleta** assim como **Entrada** e **Entrada\_Atleta**. Deste modo é fundamental que a nossa tabela extra tenha como chave primaria ambas as chaves estrangeiras necessárias.

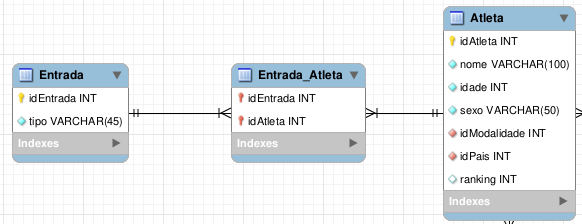
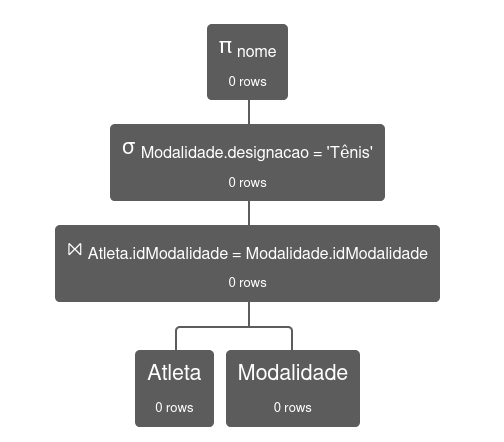
****

Figura 7 - Exemplo de Relacionamento N:N

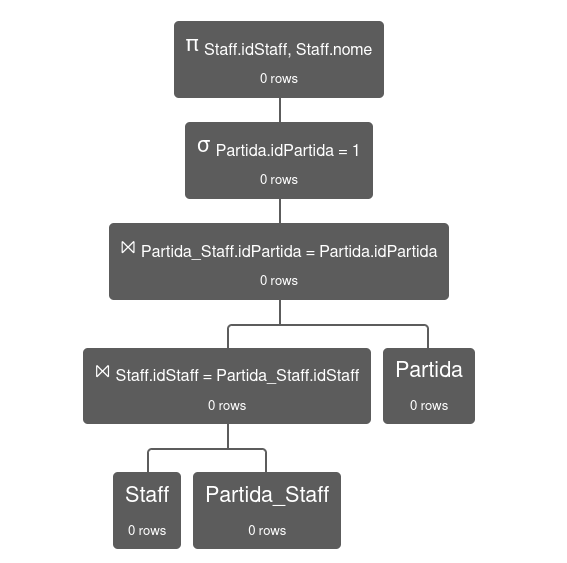
* 1. **Validação do modelo com interrogações do utilizador**
* ***Query* 1:** “Deve ser possível listar os nomes dos atletas que praticam certa modalidade”.

π nome σ Modalidade.designacao = 'Tênis' (Atleta ⨝ Atleta.idModalidade = Modalidade.idModalidade Modalidade)

Figura 8 - Query 1

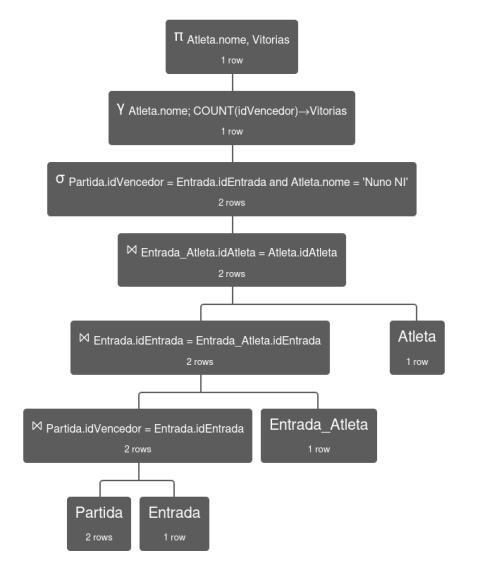
* ***Query* 2:** “Deve ser possível listar os elementos do staff que participaram numa certa partida”.

π Staff.idStaff, Staff.nome σ Partida.idPartida = 1 ( ( Staff ⨝ Staff.idStaff = Partida\_Staff.idStaff Partida\_Staff ) ⨝ Partida\_Staff.idPartida = Partida.idPartida Partida )

Figura 9 - Query 2

* ***Query* 3:** “Deve ser possível saber o número de vitórias de um jogador em específico”.

π Atleta.nome, Vitorias γ Atleta.nome; COUNT(idVencedor)→Vitorias σ Partida.idVencedor = Entrada.idEntrada and Atleta.nome = 'Nuno NI' ( ( ( Partida ⨝ Partida.idVencedor = Entrada.idEntrada Entrada ) ⨝ Entrada.idEntrada = Entrada\_Atleta.idEntrada Entrada\_Atleta) ⨝ Entrada\_Atleta.idAtleta = Atleta.idAtleta Atleta )

Figura 10 - Query 3

1. Implementação Física

**5.1 Apresentação do processo de criação da base de dados**

Neste trabalho, usamos o MySQL como sistema de gestão de bases de dados.

Traduzimos o modelo lógico para um sistema MySQL usando o MySQL Workbench para suportar e correr o nosso código. Para Gerar as tabelas necessárias executamos o código que está nos ficheiros do projeto.

Nesse sentido, recorrendo aos comandos SQL, começamos a implementação física. Com os comandos apresentados abaixo, criamos a Base de dados com o nome ‘Torneio\_EOR’ e começamos a trabalhar nela.

Uma imagem com texto, Tipo de letra, captura de ecrã, branco

Descrição gerada automaticamente

*Figura 11 – Criação e Utilização da base de dados nomeadas “Torneio\_EOR”*

Após a criação da nossa base de dados, seguimos com a criação tabelas que pertencentes à mesma, utilizando o comando “Create Table”, este processo deve tem de ser feito com cuidado e seguindo uma certa ordem de criação, sendo que primeiro de tudo temos de efetuar a criação das tabelas sem dependências, de modo que depois seja possível referenciar estas mesmas tabelas na criação de tabelas que possuem dependências destas tabelas independentes. De seguida temos exemplos da criação de algumas das tabelas criadas na nossa base de dados.

**1. Tabela da Entidade “Partida”**

Uma imagem com texto, menu, captura de ecrã, documento

Descrição gerada automaticamente

*Figura 12* ***-*** *Tabela Entidade SQL*

Nesta Tabela é descrito o id sendo este um valor inteiro, é descrito também todos os outros atributos. A chave primária corresponde ao id da Partida e temos as chaves estrangeiras idModalidade, idCampo, idTorneio, idEntradaA, idEntradaB e idVencedor que referênciam as chaves primárias das respetivas tabelas.

Uma imagem com texto, Tipo de letra, captura de ecrã, número

Descrição gerada automaticamente**2. Tabela da Relação Partida entre e Staff**

*Figura 13* ***-*** *Tabela Entidade Partida\_Staff SQL*

Nesta tabela, é criada de modo a lidar com uma relação de N para N, neste caso a relação entre a tabela staff e a tabela partida. Todas as tabelas com este tipo de objetivo, têm sempre 2 chaves estrangeiras, neste caso, o id do Staff e o id da Partida e estas duas chaves formam a chave primária.

Uma imagem com texto, Tipo de letra, captura de ecrã, file

Descrição gerada automaticamente**3. Tabela do multivalorado Contacto**

*Figura 14* ***-*** *Tabela Contacto em SQL*

Neste tipo de tabela, sendo esta a correspondente aos contactos dos Staff de um evento, é sempre necessário a chave estrangeira da tabela “relacionada” e o atributo correspondente. Estes dois formam sempre a chave primária. Esta estrutura é análoga a todo o tipo de tabelas de multivalorados.

**5.2 Cálculo do espaço da base de dados (inicial e taxa de crescimento anual)**

Utilizando informação encontrada na documentação oficial do MySQL, construímos a tabela abaixo que inclui os custos de memória dos diferentes domínios utilizados neste trabalho:

|  |  |
| --- | --- |
| **Custos de Memória** | |
| **Tipo** | **Custo** |
| *INT* | 4 Bytes |
| *VARCHAR(size), 0 < size < 255* | size + 1 Bytes |
| *DATETIME* | 8 Bytes |
| *DATE* | 3 Bytes |

*Figura 15* **-***Custo de Memória SQL*

Consoante os valores acima apresentados, agora é necessário fazer o cálculo de todos os atributos presentes na base de dados correspondentes a cada tabela, assim avançámos para a sua contabilização. Seguindo a seguinte raciocínio, obtemos:

C𝑢𝑠𝑡𝑜 = A \* 4 bytes + B \* (size +1) bytes + C \* *3* bytes + D \* 8 bytes

A = Número de *INTs;*

B = Número de *VARCHAR(size),* Com *size* entre [0,255] bytes

C = Quantidade de *DATEs;*

D = Quantidade de *DATETIMEs;*

(Dados apresentados conforme a tabela X)

|  |  |
| --- | --- |
| **Domínios Da Base de Dados** | |
|  |
| **Tabela** | **Custo** |  |
| Modalidade | (1\*4) + (1\* 51) = 4 + 51 = 55 bytes |  |
| País | (1\*4) + (1\* 51) = 4 + 51 = 55 bytes |  |
| Atleta | (5\*4) + (1\*101) + (1\*51) = 20 + 101 +51 = 172 bytes |  |
| Morada | (3\*4) + (3\*51) = 12 + 153 = 165 bytes |  |
| Campo | (4\*4) + (1\*51) = 16 + 51 = 67 bytes |  |
| Staff | (3\*4) + (3\*51) + (1\*101) + (1\*3) = 12 + 153 + 101 + 3 = 269 bytes |  |
| Contacto | (2\*4) = 8 bytes |  |
| Entrada | (1\*4) + (1\*46) = 4 + 46 = 50 bytes |  |
| Entrada\_Atleta | (2\*4) = 8 bytes |  |
| Torneio | (2\*4) + (2\*51) = 8 + 102 = 110 bytes |  |
| Partida | (8\*4) + (1\*51) + (1\*8) = 32 + 51 +8 = 91 bytes |  |
| Set | (6\*4) = 24 bytes |  |
| Game | (5\*4) = 20 bytes |  |
| Partida\_Staff | (2\*4) = 8 bytes |  |
| **Totalidade da Soma** | 1102 bytes |  |

*Figura 16* *- Custo Tabelas Torneio\_EOR*

Se todas as tabelas da base de dados do Torneio\_EOR tiverem exatamente o mínimo de preenchimento, o custo da base de dados é de 1102 Bytes.

Com esta base de custo apresentado, uma projeção é muito difícil de se fazer devido às diversas variáveis que todo o processo está depende, de forma geral e tendo em conta um dos pontos explorados no tópico 1.4 sobre a Viabilidade deste projeto sendo ele “Aumentar a realização de novos torneios neste formato”, tomando este tópico como principal fator da realização de mais Torneios num ano.

Com isto, depois de realizar este cálculo para todas as tabelas da base de dados, foi feita uma nova tabela na folha de cálculo, na qual constava, para cada tabela da base de dados, a dimensão por registo da tabela (em bytes), a sua taxa de crescimento anual expectável e o tamanho anual adicional. Para a taxa de crescimento anual expectável.

|  |  |  |  |
| --- | --- | --- | --- |
| **Tabela** | **Tamanho por Registro (bytes)** | **Crescimento Anual (%)** | **Tamanho Anual Adicional (bytes)** |
|  |
| Modalidade | 55 bytes | 0% | 55 bytes |  |
| País | 55 bytes | 5% | 57,5 bytes |  |
| Atleta | 172 bytes | 10% | 189,2 bytes |  |
| Morada | 165 bytes | 10% | 181,5 bytes |  |
| Campo | 67 bytes | 40% | 93,8 bytes |  |
| Staff | 269 bytes | 20% | 322,8 bytes |  |
| Contacto | 8 bytes | 20% | 10 bytes |  |
| Entrada | 50 bytes | 100% | 100 bytes |  |
| Entrada\_Atleta | 8 bytes | 100% | 16 bytes |  |
| Torneio | 110 bytes | 120% | 233 bytes |  |
| Partida | 91 bytes | 100% | 182 bytes |  |
| Set | 24 bytes | 100% | 48 bytes |  |
| Game | 20 bytes | 100% | 40 bytes |  |
| Partida\_Staff | 8 bytes | 100% | 8 bytes |  |
| **Total:** | **1102 bytes** | **39%** | **1536,8 bytes** |  |

*Figura 17 – Tabela de Crescimento Anual*

Inicialmente, para a taxa de crescimento anual expectável, espera-se que a realização de mais Torneios ao longo de um ano desportivo aumente em 30%, o que implica que aquilo que deriva do Torneio aumente proporcionalmente, como é o caso das tabelas País, em 5%, devido ao surgimento de atletas de países “estreantes” na competição que serão adicionados à Base de Dados, os atletas e respetivas moradas de toda a força de trabalho em 10%, assim como a adição ou contratação de novo pessoal de staff e respetivo contactos de toda a força de trabalho que simboliza os 20%, com a realização de Torneios em diversos países espera-se futuramente a utilização de novos campos ainda não presentes no sistema, num aumento significativo de 40%, as tabelas Entrada, Entrada\_atleta, Torneio, Partida, Set, Game e Partida staff que aumenta em 100% uma vez que em cada novo Torneio haverá uma nova totalidade de informação nestas suas entradas e para finalizar, como todas as novas modalidades já estarão registradas no sistema, não haverá um aumento previsto para esta tabela

Após a revisão de toda informação apresentada e estudada por todos os envolvidos, a última coluna da tabela, Tamanho Anual Adicional calcula, assumindo que a base de dados começa com um registo por tabela, e tendo em conta o seu crescimento anual, quanto mais ela ocuparia ao final de um ano. A fórmula utilizada é, portanto, Tamanho por Registo \* (1 + Crescimento Anual). Este cálculo é efetuado para cada tabela e, de seguida, calculado o seu total. Chegamos, assim, à conclusão de que a base de dados da Agência Apollo tem uma dimensão inicial (com 1 registo por tabela) de **1102 bytes** e que, ao fim de um ano, este valor aumentaria em **1536,8 bytes**,

**5.3 Realização do povoamento da base de dados**

Para utilizarmos a Base de Dados, vamos precisar de a povoar, usando dados fornecidos e obtidos por toda a equipa de desenvolvimento. O povoamento da Base de Dados pode ser realizado da seguinte forma usando scripts de MySQL para inserir os dados diretamente na Base de Dados com instruções **INSERT**, tal como no processo de criação das nossas tabelas, neste processo de povoamento é primeiro necessário povoar as tabelas sem dependências, e só depois povoar as tabelas que possuem dependências de outras tabelas de modo a que quando efetuarmos a inserção dos dados de uma tabela e esta possua uma ou mais chaves estrangeiras, os valores correspondentes a estas chaves já existam na nossa base de dados.

Temos de seguida como exemplo a povoação dos nossos atletas**:** Uma imagem com texto, captura de ecrã, Tipo de letra, documento

Descrição gerada automaticamente

*Figura 18 – Exemplo do processo de povoamento da base de dados via INSERT dos atletas*

Aqui temos um exemplo de como a inserção é feita. Primeiro selecionamos qual a tabela onde queremos inserir os dados (Neste caso usamos o ‘Torneio\_EOR.Atleta’ onde ‘Torneio\_EOR’ é o nome da Base de Dados e o ‘Atleta’ é a tabela dos Atletas que participam do Torneio), depois indicamos quais os atributos a serem atualizados (indicados pela linha ‘(id, nome, pais, ...)’) e por fim metemos quais os dados, respeitando as condições das tabelas, ou seja, não meter um VARCHAR onde devia estar um INT

**5.4 Tradução das interrogações do utilizador para SQL**

Para a tradução das interrogações do utilizador para SQL, criamos queries. Abaixo serão apresentadas 3 delas. O resto das queries estão nos scripts do projeto.

Query 1: Esta query apresenta todos os países registados na base de dados e o número de atletas que representam cada um desses países. Após o pedido do comitê da EOR, optamos por criar esta Query, de modo a ter facilidade de acesso ao número de participantes e a sua diversidade.

**Uma imagem com texto, Tipo de letra, captura de ecrã

Descrição gerada automaticamente**

*Figura 19 – Query nº de representantes de cada país*

Query 2: Nesta Query obtemos o número de atletas participantes de cada modalidade.

Uma imagem com texto, Tipo de letra, captura de ecrã

Descrição gerada automaticamente

**5.5 Definição e caracterização das vistas de utilização em SQL**

Uma vista é uma tabela virtual derivada de uma ou várias tabelas/vistas existentes. É definida por uma query MySQL e aparenta ser uma tabela, mas não armazena nenhum dado. Assim, de modo a facilitar a utilização da base de dados com e focando nos dados principais a representar-se, foram criadas algumas vistas (views) em SQL. Iremos agora apresenta o código implementado em algumas delas.

Vista 1: Esta vista permite-nos ver os dados de cada atleta na base de dados, esta é útil para ajudar a organização das partidas e das entradas.

Uma imagem com texto, captura de ecrã, Tipo de letra

Descrição gerada automaticamente

*Figura 20 – Vista dos Atletas.*

Uma imagem com texto, captura de ecrã, Tipo de letra, número

Descrição gerada automaticamente

*Figura 20 – Output da Vista dos Atletas*

Vista 2: Com esta vista temos maior facilidade de acesso à informação correspondente às partidas já registadas na nossa base de dados, esta vista pode ser especialmente útil para numa fase final, os utilizadores poderem obter informações de quando e onde se realizam as partidas.

**Uma imagem com texto, captura de ecrã, Tipo de letra

Descrição gerada automaticamente**

*Figura 21 - Vista Partida*

**Uma imagem com texto, Tipo de letra, número, captura de ecrã

Descrição gerada automaticamente**

*Figura 22 – Output da Vista Partida*

**5.6 Definição dos perfis de utilização para cada utilizador da base de dados**

Para a resolução deste problema, criamos alguns utilizadores e adicionamos as permissões necessárias seguindo os requisitos levantados pelo comitê do Torneio. Nos tópicos a seguir é mostrado um excerto do script que permite criarmos utilizadores, garantir e remover permissões, o resto do script encontra-se nos ficheiros do projeto.

Assim, como exemplo, criamos as categorias de “Admin”, “Supervisores” e “Organizadores”

**1. Definição de utilizadores**

Uma imagem com texto, captura de ecrã, Tipo de letra

Descrição gerada automaticamente

*Figura 23 – criação de “Utilizadores” na base de dados*

Conforme a necessidades envolventes na realização/atualização de informação durante um torneio, demos as seguintes permissões, para o exemplo do “supervisor” (membro chefe do grupo staff durante uma partida/game/set) e para o “admin” de:

**2. Dar todas permissões a certos utilizadores**

Uma imagem com texto, captura de ecrã, Tipo de letra

Descrição gerada automaticamente

*Figura 24 – Permissão definidas para um Admin e supervisor*

**5.7 Indexação do Sistema de Dados**

Decidimos criar um conjunto de alguns índices de modo a facilitar a procura em algumas tabelas. Estes índices são utilizados maioritariamente para uma melhoria na velocidade das consultas, pois permitem que a base de dados localize os dados de uma maneira mais rápida sem precisar correr a tabela toda. Também fazem com que as pesquisas sejam otimizadas, diminuindo assim o tempo de procura dos dados. Com base nestas informações decidimos cria os seguintes índices:

Uma imagem com texto, captura de ecrã, Tipo de letra

Descrição gerada automaticamente

*Figura 25 – Criação de Index no My SQL*

Criamos o “index\_idAtleta” com o objetivo de ter acesso rápido a tabela correspondente ao mesmo;

Criamos o “index\_idPartida” com o objetivo de ter acesso rápido a tabela correspondente a Partida;

Criamos o “index\_idStaff” com o objetivo de ter acesso rápido a tabela correspondente ao Staff com o seu id para, caso necessário, fazer a sua modificação;

Criamos o “index\_idEntrada” com o objetivo de ter acesso rápido a tabela correspondente aos Atletas;

No entanto, a equipa de desenvolvimento concluiu que não há necessidade, no estado atual da base de dados, de criar índices visto que a base de dados não tem qualquer problema de desempenho devido à baixa quantidade de dados que esta, atualmente, apresenta. Porém, a criação destes índices pode vir a ser útil no futuro consoante a evolução desta base de dados.

**5.8 Procedimentos Implementados**

Um procedimento é um conjunto de instruções mySQL pré-compiladas armazenadas na Base de Dados que podem ser invocadas. Existem vários procedures no script “Queries” da nossa base de dados. Estas procedures foram implementadas de modo a cumprir os nossos requisitos de manipulação.

Procedure 1: Esta procedure foi criada no intuito de listar os elementos da staff que participaram numa dada partida, como é pedido no requisito de manipulação 15.

Uma imagem com texto, captura de ecrã, Tipo de letra, número

Descrição gerada automaticamente

*Figura 26 – Procedure Correspondente ao RM15*

*Uma imagem com texto, captura de ecrã, Tipo de letra, número

Descrição gerada automaticamente*

*Figura 27 – Output da Procedure staffNaPartida*

Procedure 2: Consoante o nosso requisito de manipulação 14, criamos uma procedure que nós permite saber o número de vitórias do jogador que escolhermos.

Uma imagem com texto, captura de ecrã, Tipo de letra

Descrição gerada automaticamente

*Figura 28 – Procedure Correspondente ao RM14*

**Uma imagem com texto, captura de ecrã, Tipo de letra, file

Descrição gerada automaticamente**

*Figura 29 – Output da Procedure numWins*

**5.9 Plano de segurança e recuperação de dados**

Um fator extremamente importante num sistema de gestão de base de dados são as cópias de segurança, que permitem confiança e integridade no sistema implementado. No caso do nosso projeto, estas cópias são essenciais para manter os dados de todos os docentes que compõem os Torneios salvos e atualizados a longo prazo e em tempo real.

Com isto, a equipa de desenvolvimento achou por bem fazer uma política de segurança completa para esta base de dados, assim ensinou a todos Utilizadores com permissões de administrador os seguintes comandos (numa máquina com sistema operativo Windows 10/11)

de como fazer um backup da Base de dados:

1. Abrir CMD com permissões de administrador

2. >> cd.. (até chegar ao topo do disco)

3. >> cd Program Files

4. >> cd MySQL

5. >> cd MySQL Server 8.0 (8.0 é a versão atual do server)

6. >> cd bin

7. >> mysqldump -u USER -p PASSWORD Torneio\_EOR > backupTorneio\_EOR.sql

Após esta execução sequenciada, é possível gerar o backup da base de dados no seu todo, com isto, e seguindo a recomendação da equipa de desenvolvimento, é fundamental guardar o backup num local de memória desconectado de qualquer sistema informático online para máxima segurança, uma Pen-Drive ou um Disco Externo seriam o ideal para tal.

Foi aconselhado também fazer um backup todos os dias durante o decorrer de um Torneio para evitar ao máximo perdas de dados. Com esta pequena medida deixamos os administradores do sistema mais confiantes possível com o sistema implementado.

Uma imagem com texto, captura de ecrã, software, Software de multimédia

Descrição gerada automaticamenteEm baixo, uma possível execução de comandos para a realização de futuras copias de segurança:

*Figura 30* ***-*** *Execução de Comandos numa Máquina Windows Com as permissões de administrador*

Assim, demonstramos também uma possível execução, no MySQL Workbench, de como guardar a base de dados após a finalização de um dia de torneio ou até mesmo na duração do próprio, seguindo os seguintes passos:

Uma imagem com captura de ecrã, software, texto

Descrição gerada automaticamente

*Figura 31 – passos 1,2 e 3*

Uma imagem com captura de ecrã, texto, software, computador

Descrição gerada automaticamente

*Figura 32 – passo 4*

**6 Conclusões e trabalho futuro**

Este projeto envolveu alguns desafios tanto a nível de grupo quanto na execução do projeto em si. Confirmamos que a comunicação eficaz, a organização de tarefas e uma boa apresentação são elementos fulcrais para o sucesso de qualquer projeto.

A apresentação da primeira parte do projeto e as considerações finais por parte dos docentes que supervisionaram o mesmo permitiram-nos constatar nuances no trabalho desenvolvido até então culminando num ponto de viragem e motivação adicional para o desenvolvimento do produto final.

Durante a realização deste trabalho tivemos um claro crescimento no conhecimento inerente às matérias lecionadas na cadeira de Bases de Dados. Achamos que experiencia de fazer um trabalho a partir de requisitos foi fundamental para o nosso desenvolvimento a nível de projeção de um trabalho coeso e lógico foi bastante positiva. O facto deste trabalho ser baseado numa necessidade real dum evento que trabalha com algum nível de abstração fez com que abordássemos o problema duma maneira diferente do habitual, a possibilidade da liberdade criativa foi algo que promoveu bastante o nosso esforço.

Para finalizar, vamos enumerar alguns pontos que podem ser melhorados numa futura versão desta base de dados. O desenvolvimento de um modelo conceptual bastante melhor. Nós deparamo-nos muitas vezes a editar o modelo conceptual porque alguns atributos claramente não faziam muito sentido.

Apresentamos diversas dificuldades criação de funções uteis para o nossa base de dados, levando mais tempo que tínhamos planeado ao criar o processo de raciocínio das mesmas conforme as adversidades apresentadas

Com isto, nossa dificuldade em sentido coletivo claramente foi a divergência criativa que levou muitas vezes ao ‘roteiro’ a não sair como o esperado na altura e levar nos bastante vezes a refazer certas partes.

**7 Bibliografia**

Conteúdo disponibilizado na blackboard na Unidade Curricular “Base de Dados.