



Universidade do Minho

Mestrado Integrado em Engenharia Informática
Licenciatura em Ciências da Computação

Unidade Curricular de Bases de Dados

Ano Letivo de 2019/2020

Sports Medical Center

Tiago Magalhães a84485

Hugo Matias a85370

João Abreu a84802

Duarte Oliveira a85517

BD

Data de Recepção	
Responsável	
Avaliação	
Observações	

Clínica de Testes de saúde e performance desportivos

Tiago Magalhães a84485

Hugo Matias a85370

João Abreu a84802

Duarte Oliveira a85517

Resumo

O presente relatório tem por objetivo descrever as várias fases que foram abordadas para o desenvolvimento de uma base de dados relacional e não relacional, expondo fidedignamente a sequência de passos que levou à elaboração deste sistema. O objetivo desta base de dados é o armazenamento de informação digital, ou seja, o armazenamento virtual de informação relativa a todo o processo executivo correlacionado com uma clínica da área de saúde desportiva.

Numa fase inicial é feita a descrição do contexto do problema, fundamentando a necessidade da base de dados e analisada a viabilidade do processo, para além de ser referido todo o contexto que envolvia este projeto. De seguida fez-se o levantamento e análise dos requisitos junto da entidade administrativa da clínica e seus colaboradores, com a qual nos dispusemos a colaborar. Após as formalidades iniciais, é descrito todo o processo científico e prático ligado a este projeto que permitiu a criação, estruturação e desenvolvimento das unidades que foram compondo este sistema de base de dados. É também discriminado todo o processo de associação de modelos e tabelas que justificam e trazem alguma expressão visual à resolução dos problemas que foram sendo acrescentados.

Finda-se com um conjunto de anexos, que, sendo requeridos, procuram acrescentar algo mais a esta exposição.

Área de Aplicação: Criação de Sistemas de Bases de Dados

Palavras-Chave: MYSQL, BASES DE DADOS, POVOAMENTO, RELAÇÕES, MODELAÇÃO, Modelo Conceptual, Modelo Lógico, Base de Dados Relacional, Entidade, Relacionamento, Atributo, Chave Primária, Chave Estrangeira, Transações, Normalização, *Triggers*, Vistas.

Índice

Resumo	i
Índice de Figuras	iv
Índice de Tabelas	v
Introdução	1
1.1. Contextualização	1
1.2. Apresentação do Caso de Estudo	1
1.3. Motivação e Objetivos	2
1.4. Estrutura do Relatório	3
2. Levantamento e Análise de Requisitos	4
2.1. Método de levantamento e de análise de requisitos adotado	4
2.2. Requisitos levantados	5
2.2.1. Requisitos de descrição	5
2.2.2. Requisitos de exploração	6
2.2.2. Requisitos de Control	6
2.3. Análise de requisitos	7
2.4. Apresentação e generalização do diagrama ER	7
3. Modelação conceptual	8
3.1. Identificação e caracterização das entidades	8
3.2. Identificação e caracterização dos relacionamentos	10
3.3. Identificação e Caracterização da Associação dos Atributos com as Entidades e Relacionamentos	12
3.4 Validação do modelo de conceptual com o utilizador	15
4.1 Construção e validação do modelo de dados lógico	16
4.2 Esquematização modelo lógico	18
4.3 Validação formal do modelo com recurso à Normalização	19
4.4 Validação do Modelo com as Interrogações do Utilizador	21
5. Implementação Física	22
5.1. Seleção do sistema de gestão de base de dados	22
5.2. Tradução do esquema lógico para o sistema de gestão de bases de dados escolhido em SQL	22
5.3. Tradução das interrogações do utilizador para o SQL	23
5.4 Tradução das transações estabelecidas para SQL	26
5.5 Definição e caracterização dos mecanismos de segurança em SQL	27
6. Utilização e migração para o Neo4j	30
6.1. Justificação da utilização de um sistema NoSQL	30

6.2 Migração	30
6.3 Tradução das interrogações do utilizador para Cypher	33
Conclusões e trabalho futuro	39
Referências	40
Anexos	41
I. Anexo 1	42
7. Lista de Siglas e Acrónimos	43

Índice de Figuras

Figura 1 - Diagrama ER	7
Figura 2 - Diagrama Lógico	18
Figura 3 - Criação requisito de exploração	23
Figura 4 - Criação de gatilho (" <i>trigger</i> ")	24
Figura 5 - Procedimento	24
Figura 6 - Funções elaboradas	26

Índice de Tabelas

Tabela 1	9
Tabela 2	12
Tabela 3	13
Tabela 4	13
Tabela 5	13
Tabela 6	14
Tabela 7	14
Tabela 8	14

Introdução

1.1. Contextualização

Nesta secção será feita uma contextualização, devidamente fundamentada exibindo uma implementação da base de dados, tendo em mente a viabilidade do projeto. Finda-se esta secção com um resumo da estrutura abrangente do relatório.

1.2. Apresentação do Caso de Estudo

Um aumento populacional aliado a um grande crescimento tecnológico e corporativo traduziu-se numa necessidade imperiosa de organização estrutural de dados. Isto desde cedo atormentou as mais variadas instituições que tiveram de encontrar formas simplistas e eficazes de poderem manipular quantidades incomensuráveis de informação. Nesta variante, uma das secções das mais sensíveis seria o tratamento de informação e dados correlacionados com a área da saúde, sendo que na data de elaboração do presente manuscrito ainda há muitas instituições que necessitam de uma profunda revolução estrutural nos seus sistemas de bases de dados. Isto porque é importante minimizar o tempo de atualização da base de dados anteriormente existente que pressupõe a duplicação de dados e a tardia inserção deles mesmos para que possam ser manipulados mais tarde. Este processo, além de demorado, está sujeito a atrasos e a erros de registo.

Por esta ordem de ideias, este sistema foi formulado com o intuito de facilitar a interação de um serviço de saúde exclusivamente procurado por atletas (profissionais ou semiprofissionais) que necessitam de prestar testes ou exames físicos, muitas vezes para poderem exercerem atividades físicas de forma competitiva ou simplesmente para poderem ser abrangidos por seguros médicos contra os riscos que a prática de desporto acarreta.

Este contexto prova a necessidade e a importância da organização de dados (relativos não só dos utentes da clínica, mas também do tipo de serviço e dos seus colaboradores) de forma intuitiva, simples e eficaz. Poupano-se tempo e mão de obra executiva (como por exemplo serviços de secretariado) e agilizando-se qualquer tipo de atividades não correlacionadas com

esta entidade é possível que se foque a energia laboral para o que é realmente relevante para um bom funcionamento de uma instituição de saúde, facilitando a vida em especial para os funcionários e colaboradores desta, mas também para quem usufrui dos seus serviços.

Assim, a grande desorganização - ainda hoje presente em estabelecimentos de saúde - na vertente de planeamento de consultas e exames médicos, foi uma das questões também postas em foco na elaboração deste trabalho.

1.3. Motivação e Objetivos

A clínica médico-desportiva “*SportsMedicalCenter*” (SPC) apresentou uma proposta bastante aliciante para a atualização da sua base de dados já obsoleta - ainda a papel e guardada em arquivos de escritório. Tendo o nosso grupo de informáticos negociado todas as questões contratuais, e entusiasmados com a ambição que a empresa nos passou, arregaçamos mangas e começamos a trabalhar em prol do desafio apresentado. Tendo em conta a sensibilidade do projeto que nos foi passado era crucial ter uma postura cuja primazia fosse, desde início, a organização e o planeamento. O primeiro passo foi interagir com os nossos clientes. Assim sendo, a primeira etapa passou por haver um estreitamento de relações entre a vertente médica e administrativa e o nosso grupo de informáticos.

Após uma grande deliberação nas quais todas as partes mostraram os seus pontos de vista, acertou-se a implementação de uma base de dados devidamente estruturada e de fácil acesso para colmatar as dificuldades que anteriormente subsistiam na organização e manipulação dos dados referentes às equipas médicas, mas também dos atletas que requeriam os seus serviços. Os exames a que os desportistas se sujeitam também foram postos em foco, procurando-se a agilização da interação entre a clínica e atletas.

Para além disso, a vertente monetária também foi tida em conta, procurando-se verificar a liquidação de contas por parte de cada atleta ou, noutros casos, a entidade desportiva – clube - representada pelo indivíduo que se dirige à clínica.

De forma inovadora, trazendo numerosas vantagens em relação ao primitivo sistema de base de dados que vigorava na instituição, inseriu-se uma base de organização de consultas e testes, evitando complicações em relação ao estado de pagamento ou a reagendamentos por parte da clínica ou mesmo.

Noutra variante, foi trazida a capacidade de estudo do universo desta instituição de saúde, tendo como exemplo a elaboração de dados estatísticos em relação a exames e testes físicos, de modo a haver um melhor planeamento futuro de gastos em pessoal e infraestruturas. Este mecanismo traz uma clara colaboração científica ao fornecer dados relevantes para casos de estudo científicos na vertente da anatomia.

1.4. Estrutura do Relatório

Este relatório é estruturado de modo a apresentar o que é pretendido com a realização deste trabalho, seguido pelo desenrolar das constituintes e seu uso no projeto feito.

É seguido o plano apresentado pelo docente. Utilizou-se uma linguagem cuidada e o mais cientificamente correta possível.

Inicialmente foram apresentados os “porquês” deste projeto, bem como os objetivos pretendidos com a sua realização.

O que irá ser apresentado de seguida mostra o que é necessário para começar este projeto, as suas modelações, as respetivas validações, progressão de natureza abstrata para definição, e demonstração das entidades constituintes assim como as suas relações.

2. Levantamento e Análise de Requisitos

Nesta secção será abordado o estudo e análise dos vários requisitos levantados e no final será feita uma análise geral dos mesmos.

2.1. Método de levantamento e de análise de requisitos adotado

A nível profissional é necessária uma abordagem bastante aberta e abrangente para entender o que o cliente realmente nos propõe, de modo a que a vertente prática do projeto vá de encontro à visão teórica de quem requer os serviços de criação de uma base de dados. Assim, como referido anteriormente, foi necessário um estreitamento de relações entre a direção administrativa da clínica SPC, os representantes dos seus profissionais e o nosso departamento informático, de modo a que cada parte desse a entender a sua visão e objetivos para o desenvolvimento deste projeto.

Deste modo, entendeu-se que dentro dos planos da clínica, estes procuravam ver estruturada, de forma clara, a informação relativa a cada paciente, visando ter acesso facilitado e simplista da mesma. O mesmo se aplicaria aos seus profissionais de saúde. Para além disso foi imposta a necessidade de armazenamento de dados relativos às consultas das diversas especialidades que constituem esta clínica.

Por conseguinte, seria criado um perfil que seria armazenado de cada atleta – nome, contacto telefónico, número de informação fiscal, género, nacionalidade, data de nascimento, morada e clube a que pertencem. Para além disso frisaram ser necessária a inserção de um “acumulado” neste perfil, no qual esperavam ir colecionando os valores dos testes ou consultas que não fossem imediatamente liquidadas pelo atleta em questão ou, noutros casos, pelo seu clube. Os atletas também teriam a si relacionados a modalidade que praticavam, diferenciada com a sua designação e categoria.

Noutra vertente, uma relação similar, mas menos extensiva, foi exigida para os profissionais de saúde desta clínica: nome, morada, contacto telefónico, data de nascimento e, em especial, um identificador (id) que permitisse individualizar cada colaborador. A especialidade médica de cada um deles também teria a si associada um id específico e a sua designação.

Tanto os médicos como os atletas teriam um código postal associado, obviamente contendo a localidade pertencente e o valor numérico.

Era importante que houvesse um aproximar do contacto entre o atleta e o profissional de saúde. Assim, procurando trazer organização ao sistema, propôs-se a criação de uma estrutura formal que mostrasse as informações relativas às consultas médicas de avaliação. Estas seriam ramificadas pela data de marcação, estado de pagamento, um reagendamento opcional e um id que diferenciase cada consulta. A partir destes diagnósticos poder-se-ia marcar um teste clínico adequado para o problema ou situação do atleta, discriminado pelo nome do teste, preço da especialidade e id que individualizasse o teste em questão.

2.2. Requisitos levantados

De seguida serão apresentados os requisitos que foram levantados e que deverão ser suportados pelo sistema de base de dados a desenvolver para cada um dos diferentes tipos (exploração e administração).

2.2.1. Requisitos de descrição

RD1: Um atleta é caracterizado pelo nome, a morada, código postal, data de nascimento, número de telemóvel, género, nacionalidade.

RD2: Uma modalidade é caracterizada pela sua designação e categoria.

RD3: Um Código postal é caracterizado por Código postal e localidade.

RD4: Um profissional de saúde é caracterizado por nome, morada, número de telemóvel, data de nascimento e id.

RD5: Uma especialidade é caracterizada pelo seu id e sua denominação.

RD6: Um boletim clínico é caracterizado pelo seu id, pela data de marcação, pelo campo que verifica se o pagamento está feito ("Pago") e pelo campo de reagendamento.

RD7: Um teste clínico é caracterizado por id, nome e preço

2.2.2. Requisitos de exploração

- RE1: Permitir a consulta das informações de um atleta, profissional de saúde, boletim clínico.
- RE2: Saber o número de testes que um dado atleta realizou
- RE3: Os profissionais de saúde deverão ser capazes de saber os testes clínicos que têm a realizar num determinado dia
- RE4: Os clubes com protocolo com a clínica deverão saber quanto devem pagar pelos testes clínicos.
- RE5: O funcionário poderá saber quanto faturou a clínica num determinado período
- RE6: O funcionário poderá consultar quais os clubes que têm protocolo com a clínica
- RE7: Um funcionário poderá consultar quais foram os testes clínicos mais realizados
- RE8: Um funcionário deverá ser capaz de consultar que teste clínicos marcados tem um dado atleta
- RE9: O funcionário pode saber se um determinado teste foi reagendado.
- RE10: O funcionário sabe qual o teste mais realizado numa determinada cidade
- RE11: O funcionário sabe qual a especialidade mais responsável por testes

2.2.2. Requisitos de Control

- RC1: O administrador deverá possuir controlo total à base de dados, quer seja apagar, adicionar, seleccionar ou modificar informação.
- RC2: Os funcionários poderão adicionar atletas, boletins clínicos, bem como consultar informações relativos a estes, também poderá consultar profissionais de saúde, e teste clínicos
- RC3: Os profissionais de saúde podem adicionar e editar profissionais de saúde à equipa de um boletim pela qual estão responsáveis, podendo também consultar profissionais de saúde, equipas e boletins clínicos.

2.3. Análise de requisitos

Após o levantamento dos requisitos através de inquéritos, os quatro candidatos a bons e dedicados engenheiros reuniram-se para elaborar uma estratégia face ao problema que havia sido discutido anteriormente. Foram necessárias algumas reuniões até que o conjunto final de requisitos ficasse finalmente estabelecido e tivesse a concordância de todos os membros. O sistema de base de dados em questão deve ter acessos faseados. Portanto, é preciso ter em conta que a clínica dispõe de uma administração, funcionários (serviço de secretariado) e profissionais de saúde, sendo que estes últimos em exames físicos estão dispostos em equipas.

Assim sendo, a administração terá privilégios máximos - acesso pleno a todos os dados, podendo manipulá-los e alterá-los conforme as suas necessidades. Os médicos terão acesso restrito, podendo aceder aos dados dos seus colegas de profissão, do boletim clínico e da equipa médica. Finalmente, os funcionários de secretariado terão acesso a todas as tabelas.

2.4. Apresentação e generalização do diagrama ER

Tendo em conta as entidades, relacionamentos e atributos desenvolvidos anteriormente representados desenvolveu-se o seguinte diagrama conceptual. Apresenta-se de seguida o diagrama de ER que caracteriza o nosso sistema .

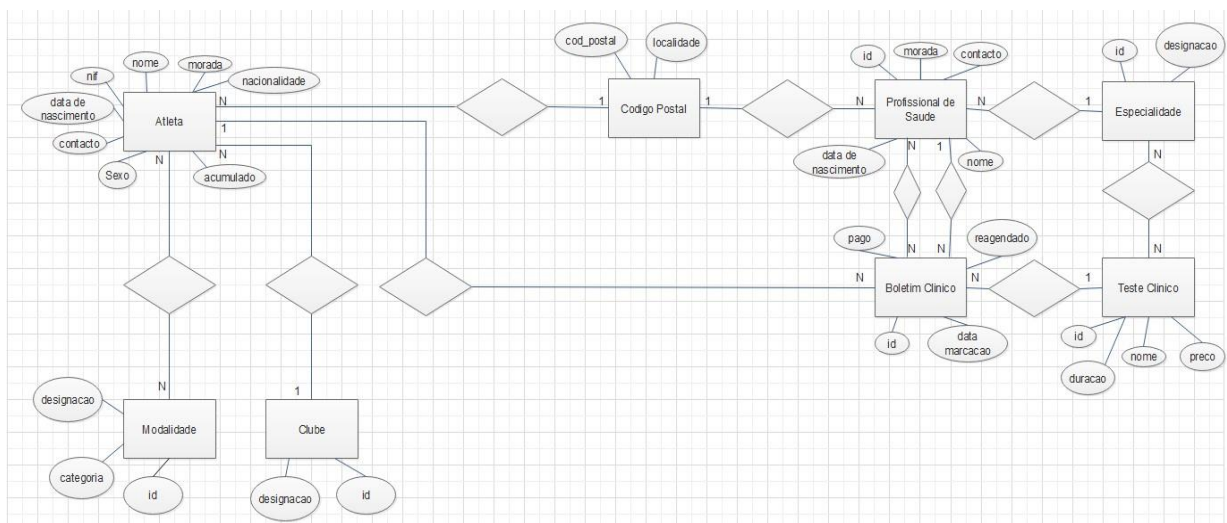


Figura 1: Diagrama ER da Clínica

3. Modelação conceptual

Depois de desenvolvidas todas as tarefas acima referidas, passamos para a conceção do modelo concetual. Este modelo permitiu-nos esquematizar dados necessários para um conjunto de aplicações, visualizando simultaneamente o relacionamento existente entre eles. O modelo produzido visa auxiliar a criação da base de dados pretendida.

3.1. Identificação e caracterização das entidades

Com base na análise dos requisitos que foram apresentados anteriormente, identificamos a necessidade da existência das seguintes oito entidades:

Atleta – A entidade atleta representa os atletas que se registaram na clínica. É caracterizada pelo seu nif, nome, morada, nacionalidade, acumulado (valor total de testes clínicos a pagar), contacto, data de nascimento e sexo.

Profissional de saúde – representa todos os funcionários médicos que exercem funções na clínica. É caracterizado por id, morada, contacto, data de nascimento e nome

Teste Clínico – representa todos os exames clínicos possíveis de realizar na clínica, cada teste clínico tem uma duração, preço, id e nome.

Boletim clínico – representa todos os testes clínicos marcados, cada boletim clínico tem id, data de marcação, pago (saber se já foi pago) e reagendado (saber se o teste foi reagendado).

Clube – A entidade clube representa todos os clubes que têm protocolo com a clínica, podendo estes pagar os testes clínicos dos seus atletas é caracterizador por id e designação.

Modalidade - A entidade modalidade representa as modalidades praticadas pelos atletas registados na clínica, cada modalidade tem um id, designação e categoria.

Especialidade- A entidade especialidade representa as especialidades dos profissionais de saúde da clínica, é caracterizada por id e designação.

Código postal – A entidade código postal representa os códigos postais, dos atletas registados na clínica e é caracterizada por código postal e localidade.

Entidade	Descrição	Ocorrências
Atleta	Termo universal que descreve todos atletas existentes, que usufruem ou já usufruíram dos serviços da clínica	Cada atleta efetua algum tipo de exame físico ou consulta na clínica
Profissional de Saúde	Termo universal que descreve todos especialistas existentes, que laboram na clínica	Cada médico executa algum tipo de exame físico ou consulta na clínica
Teste Clínico	Termo universal que descreve todos os exames e testes físicos já executados na clínica.	Cada teste clínico é referente ao atleta e à equipa de profissionais de saúde que nele estão inseridos
Boletim Clínico	Termo universal que descreve todas as consultas efetuadas na clínica.	Cada teste clínico é referente ao atleta e ao médico a ele atribuído
Clube	Termo universal que descreve o clube a que cada atleta pertence.	Cada clube está ligado a cada atleta, sendo ou não de imperativa associação (atleta pode não ter clube)
Modalidade	Termo universal que descreve as modalidades que os atletas registados praticam.	Cada modalidade está ligada a cada atleta, discriminando a mesma
Especialidade	Termo universal que descreve as especialidades médicas registadas.	Cada especialidade é referente a cada profissional de saúde
Código Postal	Termo universal que descreve o código postal	Cada registo deste atributo é referente ao código postal de cada atleta e profissional de saúde

Tabela 1: Descrição e ocorrências das Entidades

3.2. Identificação e caracterização dos relacionamentos

Tendo em conta o levantamento de requisitos realizado anteriormente, bem como a identificação das entidades foram considerados os seguintes relacionamentos com base em verbos chave:

Relacionamento: Atleta_has_Modalidade (Atleta pratica modalidade)

Com o intuito de saber quais as modalidades praticadas por cada atleta, armazenamos esta informação numa tabela de relacionamento, o facto de se saber a modalidade permite os profissionais de saúde presentes em certos testes e também adapta-los de acordo com a modalidade, de modo a retirarem-se informações mais concretas àcerca da performance do atleta.

Cardinalidade: Atleta N (**pratica**) N Modalidade

Um atleta pode praticar várias modalidades, assim como uma modalidade pode ter vários atletas.

Este relacionamento não possui atributos.

Relacionamento: Atleta_Clube

Este relacionamento existe uma vez que, se o atleta é de algum dos clubes com protocolo com a clínica não necessitará que o próprio pague os seus testes, assim torna-se importante saber qual o clube de um atleta

Cardinalidade: Atleta N (**pertence**) 1..0 Clube

Um atleta pode pertencer ou não a um clube, um clube pode ter vários atletas.

Este relacionamento não possui atributos.

Relacionamento: Atleta Código Postal

Este relacionamento existe uma vez que, por questões estatísticas, é necessário saber o código postal, localidade de um atleta

Cardinalidade: Atleta N (**habita**) 1 Código Postal

Um atleta só pode ter um código postal, mas um código postal pode ter vários atletas

Este relacionamento não possui atributos.

Relacionamento: Profissional de saúde Código Postal

Este relacionamento existe uma vez que, por questões estatísticas, é necessário saber o código postal, localidade de um profissional de saúde

Cardinalidade: Profissional de saúde N (**habita**) 1 Código Postal

Um profissional de saúde só pode ter um código postal, mas um código postal pode ter vários profissionais de saúde

Este relacionamento não possui atributos.

Relacionamento: Atleta Boletim Clínico

Este relacionamento existe uma vez que cada teste é realizado por um atleta

Cardinalidade: Atleta 1 (**marca**) N Boletim Clínico

Um atleta pode marcar vários testes (boletins clínicos), mas um boletim clínico só pode ser relativo a um atleta.

Este relacionamento não possui atributos.

Relacionamento: Boletim Clínico Teste Clínico

Este relacionamento existe uma vez que cada teste é realizado por um atleta

Cardinalidade: Boletim Clínico N (**referente**) 1 Teste Clínico

Um boletim clínico é referente a um teste clínico, mas um teste clínico pode ser referente a vários boletins clínicos.

Este relacionamento não possui atributos.

Relacionamento: Profissional de saúde Boletim Clínico

Este relacionamento existe uma vez que cada boletim tem um profissional de saúde responsável

Cardinalidade: Profissional 1 (**responsável**) N Boletim Clínico

Um boletim clínico só pode ter um profissional de saúde responsável, um profissional de saúde pode ser responsável por vários boletins clínicos.

Este relacionamento não possui atributos.

Relacionamento: Profissional de saúde Boletim Clínico (Equipa)

Este relacionamento existe uma vez que um teste do boletim clínico pode ser realizado por vários profissionais de saúde

Cardinalidade: Profissional N (**realiza**) N Boletim Clínico

Um boletim clínico pode ser realizado por vários profissionais de saúde, um profissional de saúde pode participar em vários boletins clínicos.

Este relacionamento não possui atributos.

Relacionamento: Profissional de saúde Especialidade

Este relacionamento existe uma vez que um profissional de saúde tem uma especialidade

Cardinalidade: Profissional N (é) 1 Especialidade

Um profissional de saúde só pode ser de 1 especialidade, enquanto uma especialidade por ter vários profissionais de saúde

Este relacionamento não possui atributos.

Relacionamento: Teste Clínico Especialidade (Teste Clinico has Especialidade)

Este relacionamento existe uma vez que um profissional de saúde tem uma especialidade

Cardinalidade: Teste Clínico N (pode ser realizado por) N Especialidade

Um teste clínico pode ser realizado por várias especialidades, uma especialidade um realizar vários testes clínicos.

Este relacionamento não possui atributos.

3.3. Identificação e Caracterização da Associação dos Atributos com as Entidades e Relacionamentos

Entidade	Atributos	Nulo	Multi-Valor	Derivado	Candidato	Tipo
Profissional de Saúde	Id	Não	Não	Não	Sim	INT
	Morada	Não	Não	Não	Não	VARCHAR (20)
	Contacto	Não	Não	Não	Não	VARCHAR (45)
	Data de Nascimento	Não	Não	Não	Não	DATE
	Nome	Não	Não	Não	Não	VARCHAR (45)
	Codigo postal	Não	Não	Não	Não	VARCHAR (45)
	Especialidade	Não	Não	Não	Não	INT

Tabela 2: Caracterização da entidade “Profissional de Saúde”

Entidade	Atributos	Nulo	Multi-valor	Derivado	Candidato	Tipo
Atleta	Id	Não	Não	Não	Sim	INT(9)
	Nome	Não	Não	Não	Não	VARCHAR(45)
	Morada	Não	Não	Não	Não	VARCHAR(200)
	Data de Nascimento	Não	Não	Não	Não	DATE
	Contacto	Não	Não	Não	Não	INT(9)
	Sexo	Não	Não	Não	Não	ENUM('M','F')
	Nacionalidade	Não	Não	Não	Não	VARCHAR(45)
	Código Postal	Não	Não	Não	Não	VARCHAR(45)
	Acumulado	Não	Não		Não	FLOAT
	idClube	Sim	Não	Não	Sim	INT

Tabela 3: Caracterização da entidade “Atleta”

Entidade	Atributos	Nulo	Multi-valor	Derivado	Candidato	Tipo
Boletim Clínico	Id	Não	Não	Não	Sim	INT
	Data marcação	Não	Não	Não	Não	DATETIME
	Nif atleta	Não	Não	Não	Sim	INT (9)
	Id teste clínico	Não	Não	Não	Sim	INT
	Id profissional de saúde	Não	Não	Não	Sim	INT
	Reagendado	Não	Não	Não	Não	TINYINT
	Pago	Não	Não		Não	TINYINT

Tabela 4: Caracterização da entidade “Boletim Clínico”

Entidade	Atributos	Nulo	Multi-valor	Derivado	Candidato	Tipo
Teste Clínico	Id	Não	Não	Não	Sim	INT
	Nome	Não	Não	Não	Não	VARCHAR(45)
	Preço	Não	Não		Não	FLOAT
	Duração	Não	Não	Não	Não	INT

Tabela 5: Caracterização da entidade “Teste Clínico”

Entidade	Atributos	Nulo	Multi-valor	Derivado	Candidato	Tipo
Modalidade	Id	Não	Não	Não	Sim	INT
	Designação	Não	Não	Não	Não	VARCHAR(45)
	Categoria	Não	Não	Não	Não	VARCHAR(45)

Tabela 6: Caracterização da entidade “Modalidade”

Entidade	Atributos	Nulo	Multi-valor	Derivado	Candidato	Tipo
Clube	Id	Não	Não	Não	Sim	INT
	Designação	Não	Não	Não	Não	VARCHAR(45)

Tabela 7: Caracterização da entidade “Clube”

Entidade	Atributos	Nulo	Multi-valor	Derivado	Candidato	Tipo
Código Postal	Código Postal	Não	Não	Não	Sim	VARCHAR(45)
	Localidade	Não	Não	Não	Não	VARCHAR(100)

Tabela 8: Caracterização da entidade “Código Postal”

Entidade	Atributos	Nulo	Multi-valor	Derivado	Candidato	Tipo
Especialidade	Id	Não	Não	Não	Sim	INT
	Designação	Não	Não	Não	Não	VARCHAR(45)

Tabela 9: Caracterização da entidade “Especialidade”

3.4 Validação do modelo de conceptual com o utilizador

Findado o modelo conceptual, viu-se como essencial apresentar o planeamento dado ao projeto à comissão administrativa da clínica. Por mais insignificante que possa parecer, a percepção da visão do cliente é crucial para que o nosso modelo represente todos os requisitos solicitados por quem nos solicita o trabalho. É importante que nestas reuniões ou apresentações seja tudo extremamente clarificado e que de ambas as partes exista a compreensão pelo lado oposto, de modo a evitar qualquer tipo de mal-entendido. Assim expusemos de forma sucinta cada decisão tomada que vieram a adequar-se ao modelo com formato final.

A administração pediu algum tempo para deliberar se tudo o que fora solicitado inicialmente estava a ser cumprido e, após verificação positiva, aceitaram o modelo, autorizando a continuidade do projeto, sendo necessária então, a elaboração do modelo lógico.

É agora necessária passar a uma fase mais prática deste projeto, ao mesmo tempo aliada a uma compreensão científica bastante sucinta. Nisto foi crucial a consulta do livro de *Connolly e Begg* – “Database Systems A Practical Approach to Design, Implementation, and Management”.

4.1 Construção e validação do modelo de dados lógico

O estruturamento e desenvolvimento do modelo lógico foi feito a partir do modelo conceptual, aplicando-se uma panóplia de passos específicos.

Assim, para terminar a construção do modelo de dados lógicos foram ainda seguidas as regras de derivação dos relacionamentos do modelo de dados relacional, em seguida apresentados:

Relacionamento binário de grau 1:N

São necessárias duas entidades lógicas, uma para cada entidade, e a chave primária da entidade do lado 1 tem de ser usada como atributo na entidade correspondente à entidade do lado N.

No nosso modelo este relacionamento ocorre:

Atleta **N** Clube **1**

Atleta **N** Código Postal **1**

Profissional Saúde **N** Código Postal **1**

Profissional saúde **N** Especialidade **1**

Boletim Clínico **N** Profissional de Saúde **1**

Atleta **1** Boletim Clínico **N**

Teste Clínico **1** Boletim Clínico **N**

Relacionamento binário de grau N:M: São sempre necessárias três entidades lógicas

Neste tipo de relacionamentos, uma para cada entidade e uma para o relacionamento. As chaves primárias das entidades têm de ser atributos na entidade lógica do relacionamento.

No nosso modelo este relacionamento ocorre:

Atleta **N** Modalidade **N**

Boletim Clínico **N** Profissional de Saúde **N**

Especialidade **N** Teste Clínico **N**

Através das regras de derivação definidas anteriormente foi possível identificar chaves estrangeiras e evitar redundâncias de dados.

Em seguida são apresentadas e enumeradas as relações, nas quais se estipulou a representação de chaves primárias com um duplo sublinhado. Chaves estrangeiras e primarias são representadas **assim**.

- **Atleta** – NIF, nome, morada, data de nascimento, contacto, sexo, nacionalidade, código postal, acumulado, idClube;
- **Código Postal** – cod_postal, localidade;
- **Profissional de saúde** – ID, morada, contacto, data de nascimento, nome, código postal, especialidade_id;
- **Especialidade** – ID, designacao;
- **Teste_Clinico** – ID, nome, preco, duracao;
- **Boletim_Clinico** – ID, data_marcacao, atleta_nif, testeclinico_id, profsaude_id, reagendado, pago;
- **Modalidade** – ID, designação, categoria
- **Atleta_has_Modalidade** – modalidade_id, atleta_nif
- **Teste_Clinico_has_Especialidade** – Teste_Clinico_id, Especialidade_id
- **Equipa** – profsaude_id, bolclinico_id
- **Clube** – ID, designação

4.2 Esquematização modelo lógico

Utilizando a plataforma *MySQL Workbench* estruturou-se o seguinte modelo lógico:

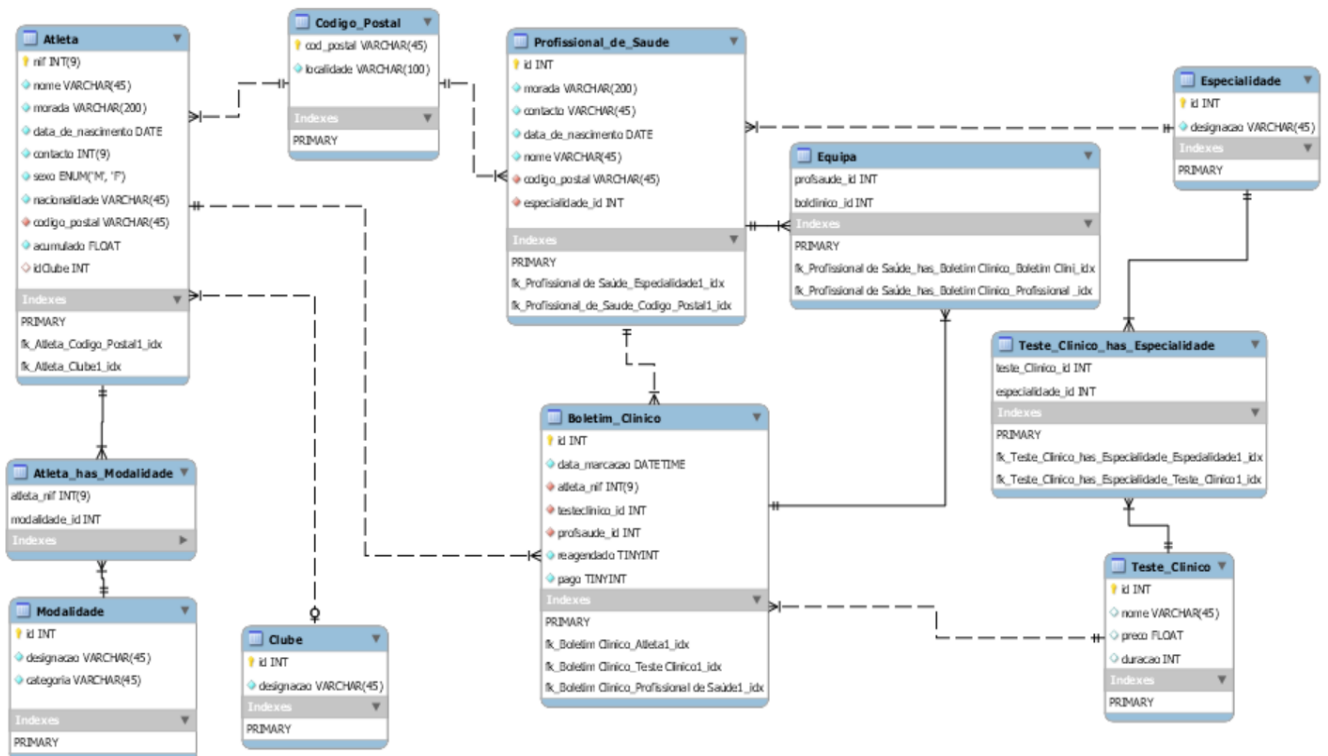


Figura 2 – Modelo lógico

4.3 Validação formal do modelo com recurso à Normalização

Primeira Forma Normal (1FN):

A normalização de uma tabela na 1.^a Forma Normal (1FN) exige que a tabela tenha uma estrutura bidimensional correta, ou seja, cada linha deve corresponder a um só registo e cada coluna a um só campo. Além disso, cada campo deve conter dados atómicos, ou seja, um só dado por cada registo. Cada dado só deve tomar um valor dentro do domínio definido para esse campo.

Através do modelo desenvolvido chegamos à conclusão de que existem vários casos comprometedores. No caso das relações, identificamos as relações N para N, tendo isto em conta, seguimos as regras de derivação apresentadas anteriormente. Assim o nosso modelo vai de encontro à primeira forma normal.

Segunda Forma Normal (2FN):

A 2.^a **Forma Normal** diz que a tabela tem de estar na 1FN e que cada atributo não chave tem de ser funcionalmente dependente da totalidade da chave primária e não apenas de uma parte dessa chave.

Assim depois de identificarmos a chave primária de uma tabela, pode dar-se um dos dois casos:

1. **A chave primária é constituída por um só atributo** (chave elementar) – neste caso, a **tabela está seguramente na 2FN** (nenhum atributo depende de uma parte da chave, visto que a chave não é composta por partes);
2. **A chave primária é constituída por mais que um atributo** (chave primária composta) – neste caso, se existe algum ou alguns atributos que dependem de uma parte da chave (ou seja, de algum atributo que constitui a chave), então a **tabela não está na 2FN**.

Como no nosso modelo não usamos chaves compostas, com o intuito de evitarmos esses casos, usamos um id que se auto incrementa. Assim, todas os atributos que não chaves principais estão completamente dependentes a nível funcional da chave principal. Deste modo o nosso modelo vai de encontro à segunda forma normal.

Terceira Forma Normal (3FN):

A **3.^a Forma Normal (3FN)** diz que a tabela tem de estar na 2FN e que nenhum atributo não chave pode depender funcionalmente de algum outro atributo que não seja a chave primária.

Portanto, para normalizar uma tabela de acordo com a 3FN, devemos analisar todos os atributos não chave e verificar se existem algumas dependências funcionais entre eles:

- se não existir nenhuma dependência funcional entre os atributos não chave, a tabela está na 3Fn;
- se existir alguma dependência funcional entre atributos não chave, então, é necessário retirar esse conjunto de atributos da tabela e construir com eles uma tabela à parte.

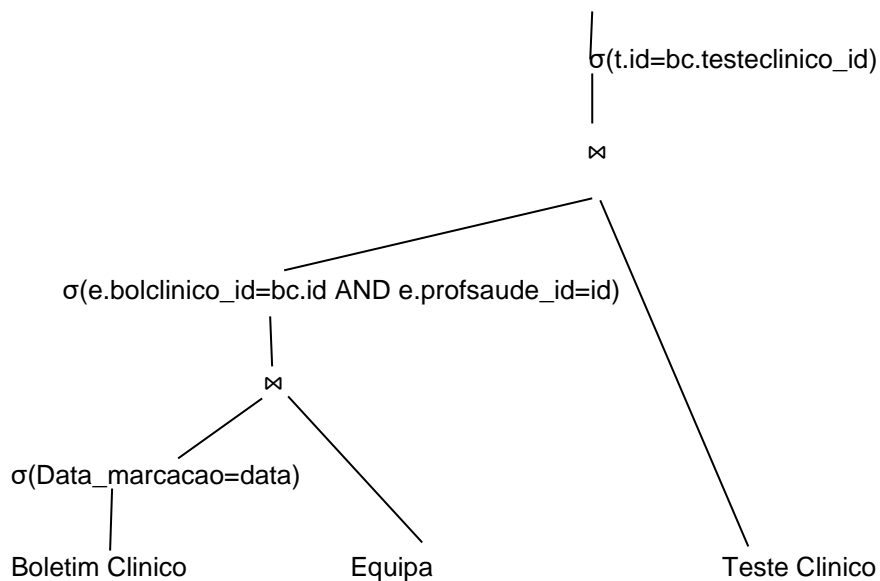
Como o nosso modelo se encontra na primeira e segunda forma normal, visto que no nosso modelo através das chaves não primárias não podemos determinar outras. Então, o nosso modelo vai de encontro à terceira forma normal.

4.4 Validação do Modelo com as Interrogações do Utilizador

Para verificar se o modelo lógico desenvolvido encontra-se válido com base em algumas interrogações do utilizador seleccionamos as interrogações mais relevantes que constavam nos requisitos de exploração da base de dados e explicamos de que forma o nosso modelo lógico consegue adequadamente satisfazê-las.

Interrogação saber os teste que um profissional de saúde realiza num dia

π BoletimClinico.nome,BoletimClinico.datademarcacao,TesteClinico.nome



5. Implementação Física

5.1. Seleção do sistema de gestão de base de dados

Para o sistema de gestão de bases de dados, resolvemos optar pelo *MySQL*. Esta decisão deveu-se ao facto de este se tratar de um software *open source* (ou seja, é livre, apesar de ter sido adquirido pela Oracle) que possui uma consistência, alta *performance*, confiabilidade e é fácil de usar, além de funcionar em mais de 20 plataformas.

5.2. Tradução do esquema lógico para o sistema de gestão de bases de dados escolhido em SQL

Para, de melhor forma, usufruir dos mecanismos que *MySQL* oferece nós decidimos, durante todo o processo, usar a ferramenta de desenho, desenvolvimento e administração de base de dados, *MySQL Workbench*. Dado que o nosso modelo lógico foi criado nesta ferramenta nós pudemos usufruir do *Forward Engineering* que esta oferece. Assim, o código necessário para esta transição foi gerado automaticamente.

5.3. Tradução das interrogações do utilizador para o SQL

Depois de povoada a base de dados (ver anexo), implementamos as interrogações necessárias em MySQL, em seguida apresentadas

Criação do requisito de exploração (RE7)

Um funcionário poderá consultar quais foram os testes clínicos mais realizados

-- 5 testes mais realizados na clinica

```
DROP VIEW IF EXISTS top5testes;
CREATE VIEW top5testes AS
SELECT t.nome,count(bc.testeclinico_id) AS `Num testes realizados` FROM Boletim_Clinico bc
INNER JOIN Teste_Clinico t ON t.id=bc.testeclinico_id
GROUP BY t.nome
ORDER BY `Num testes realizados` DESC
LIMIT 5;

SELECT * FROM top5testes;
```

Figura 3: Criação de um requisito de exploração

Criação do requisito de exploração (RE5)

```
-- Facturação num intervalo de tempo
Delimiter $$
CREATE FUNCTION fatIntTempo (inicio DATE, fim DATE) RETURNS DECIMAL (6,2) DETERMINISTIC
BEGIN
DECLARE valor DECIMAL (6,2);
SELECT sum(t.preco) INTO valor FROM Boletim_Clinico bc,Teste_Clinico t
WHERE
CAST(bc.data_marcacao AS DATE) BETWEEN inicio AND fim AND pago=1 AND t.id=bc.testeclinico_id;
RETURN valor;
END $$
DELIMITER ;
```

Figura 4: Criação de um requisito de exploração

Criação de um gatilho:

Sentimos a necessidade de criação de 2 gatilhos de modo atualizar o que deve pagar o atleta sempre que este pague algum teste cinco o gatilho irá subtrair o preço desse teste ao quanto o atleta deve pagar, sempre que o atleta não pague irá acumular

```
DROP TRIGGER IF EXISTS atualizaPagamento;

DELIMITER //
CREATE TRIGGER atualizaPagamento AFTER UPDATE ON Boletim_Clinico
FOR EACH ROW
BEGIN
    IF(NEW.pago = 1) THEN
        UPDATE Atleta
        SET acumulado = acumulado - (SELECT preco FROM Teste_Clinico WHERE id = new.testeclinico_id)
        WHERE nif = new.atleta_nif;
    END IF;
END //

DELIMITER ;

DROP TRIGGER IF EXISTS atualizaDivida;

DELIMITER //
CREATE TRIGGER atualizaDivida AFTER INSERT ON Boletim_Clinico
FOR EACH ROW
BEGIN
    IF(new.pago = 0) THEN
        UPDATE Atleta
        SET acumulado = acumulado + (SELECT preco FROM Teste_Clinico WHERE id=new.testeclinico_id)
        WHERE nif = new.atleta_nif;
    END IF;
END //
```

Figura 4: Criação de um gatilho

Criação de um procedimento

De modo a saber quanto deve um clube pagar,criamos o seguinte procedimento que recebe o id do clube e determina quanto tem de pagar

```

DROP PROCEDURE IF EXISTS totalapagarClube;
DELIMITER $$
CREATE PROCEDURE totalapagarClube (IN nome VARCHAR(45))
BEGIN
SELECT sum(t.preco) FROM Clube c
INNER JOIN Atleta a ON a.idClube=c.id
INNER JOIN Boletim_Clinico b ON b.atleta_nif=a.nif
INNER JOIN Teste_Clinico t ON t.id=b.testeclinico_id
WHERE c.designacao=nome AND data_marcacao <=NOW();
END$$
DELIMITER ;

```

Figura 5: criação de um procedimento

Criação de funções:

Sentimos a necessidade de criar 2 funções, uma para saber o total faturado num intervalo de tempo e outra para saber quantas horas num determinado dia trabalhou um profissional de saúde.

```

-- Facturação num intervalo de tempo
Delimiter $$
CREATE FUNCTION fatIntTempo (inicio DATE, fim DATE) RETURNS DECIMAL (6,2) DETERMINISTIC
BEGIN
DECLARE valor DECIMAL (6,2);
SELECT sum(t.preco) INTO valor from Boletim_Clinico bc,Teste_Clinico t
WHERE
CAST(bc.data_marcacao AS DATE) BETWEEN inicio AND fim AND pago=1 AND t.id=bc.testeclinico_id;
Return valor;
End $$
DELIMITER ;

-- Numero de horas que profsaude tem de testes clinicos num dia
DROP FUNCTION IF EXISTS numHoras;
DELIMITER $$
CREATE FUNCTION numHoras (id INT,data DATE) RETURNS INT DETERMINISTIC
BEGIN
DECLARE horas INT;
SELECT sum(t.duracao) INTO horas
FROM Boletim_Clinico b
INNER JOIN Equipa e ON DATE(b.data_marcacao)=data AND e.bolclinico_id=b.id AND e.profsaude_id=id
INNER JOIN Teste_Clinico t ON t.id=b.testeclinico_id ;
Return horas;
END$$
DELIMITER ;

```

Figura 6: funções de total faturado num intervalo de tempo e de total de horas de trabalho de um funcionário médico num determinado dia

5.4 Tradução das transações estabelecidas para SQL

Exemplo de uma das transações

Transação para adicionar boletim

```
CREATE PROCEDURE addAtleta(IN nif INT, IN nome VARCHAR(55), IN morada VARCHAR(200), IN data_de_nascimento DATE, IN contacto int(9),
                           IN sexo ENUM('M', 'F'),
                           IN nacionalidade VARCHAR(45), IN cod_postal VARCHAR(45),
                           IN modalidade VARCHAR(45), IN categoria VARCHAR(45), IN idClube int)
BEGIN
    DECLARE v_error BOOL default 0;
    DECLARE CONTINUE HANDLER FOR SQLEXCEPTION SET v_error = 1;

    SET AUTOCOMMIT = OFF;
    START TRANSACTION;
    -- criar atleta
    BEGIN
        DECLARE x INT;
        INSERT INTO Atleta (nif,nome,morada,data_de_nascimento,contacto,sexo,nacionalidade,codigo_postal,acumulado,idClube)
        VALUES (nif,nome,morada,data_de_nascimento,contacto,sexo,nacionalidade,cod_postal,0,idClube);
        -- se não existir modalidade cria registo
        IF NOT EXISTS(SELECT * from Modalidade WHERE designacao=modalidade AND categoria=categoria) THEN
            INSERT INTO Modalidade(id,designacao,categoria)
            VALUES (null,modalidade,categoria);
            SET x= (SELECT (LAST_INSERT_ID()));
        ELSE SET x=(SELECT m.id FROM Modalidade m
                     WHERE m.designacao=designacao AND m.categoria=categoria);
        END IF;
        -- adiciona à relação do atleta com a modalidade
        INSERT INTO Atleta_has_Modalidade(atleta_nif,modalidade_id)
        VALUES (nif,x);
    END;

    IF(v_error) THEN
        ROLLBACK;
    END IF;

    COMMIT;
END $$
DELIMITER ;
```

5.5 Definição e caracterização dos mecanismos de segurança em SQL

Fica claro que um dos pontos fracos e de maior importância em sistemas de base dados é a segurança, e por isso o administrador deve providenciar e implementar estratégias de segurança que impeçam acessos indevidos e indisponibilidades da base de dados.

O MySQL tem determinadas peculiaridades que podem se tornar vulnerabilidades ou pontos fortes na segurança do sistema, isso dependerá do uso de tais recursos pelo administrador.

Assim definimos os seguintes *user* e suas permissões:

```
-- Criação do administrador-----  
DROP USER 'admin'@'localhost';  
CREATE USER 'admin'@'localhost';  
SET PASSWORD FOR 'admin'@'localhost' = 'root';  
  
-- Permissão de acesso a todos os objetos de todas as bases de dados em 'localhost'.  
GRANT ALL PRIVILEGES ON sportsmedicalcenter.* TO 'admin'@'localhost';
```

```

-- Criação do perfil de Funcionário -----

DROP USER 'func'@'localhost';
CREATE USER 'func'@'localhost';
SET PASSWORD FOR 'func'@'localhost' = 'root';

-- Definição de privilégios para o utilizador 'funcionario'.
-- Permissão para a execução de instruções SELECT,INSERT e UPDATE sobre a base de dados
-- em 'localhost'.
-- acessos a tabelas
GRANT SELECT,UPDATE,INSERT ON sportsmedicalcenter.Atleta TO 'func'@'localhost';
GRANT SELECT,UPDATE,INSERT ON sportsmedicalcenter.Boletim_Clinico TO 'func'@'localhost';
GRANT SELECT,UPDATE,INSERT ON sportsmedicalcenter.Codigo_Postal TO 'func'@'localhost';
GRANT SELECT,UPDATE,INSERT ON sportsmedicalcenter.Modalidade TO 'func'@'localhost';
GRANT SELECT ON sportsmedicalcenter.Profissional_de_saude TO 'func'@'localhost';
GRANT SELECT ON sportsmedicalcenter.Especialidade TO 'func'@'localhost';

-- Permissão para a execução de procedimentos .
GRANT EXECUTE ON PROCEDURE sportsmedicalcenter.addAtleta TO 'func'@'localhost';
GRANT EXECUTE ON PROCEDURE sportsmedicalcenter.addBoletim TO 'func'@'localhost';
GRANT EXECUTE ON PROCEDURE sportsmedicalcenter.testereagendado TO 'func'@'localhost';
GRANT EXECUTE ON PROCEDURE sportsmedicalcenter.testeagendados TO 'func'@'localhost';
GRANT EXECUTE ON PROCEDURE sportsmedicalcenter.reagendarTeste TO 'func'@'localhost';
GRANT EXECUTE ON PROCEDURE sportsmedicalcenter.totalapagarClube TO 'func'@'localhost';
GRANT EXECUTE ON PROCEDURE sportsmedicalcenter.testemaisRealCidade TO 'func'@'localhost';
GRANT EXECUTE ON PROCEDURE sportsmedicalcenter.numerotestesAtleta TO 'func'@'localhost';
GRANT EXECUTE ON PROCEDURE sportsmedicalcenter.faturadoMes TO 'func'@'localhost';
-- Permissão para a execução de funções .
GRANT EXECUTE ON FUNCTION sportsmedicalcenter.fatIntTempo TO 'func'@'localhost';

-- Criação do perfil de Profissional Saúde -----
DROP USER 'profsaude'@'localhost';
CREATE USER 'profsaude'@'localhost';
SET PASSWORD FOR 'profsaude'@'localhost' = 'root';

-- Definição de privilégios para o utilizador 'funcionario'.
-- Permissão para a execução de instruções SELECT,INSERT e UPDATE sobre a base de dados
-- em 'localhost'.
-- acessos a tabelas
GRANT SELECT,UPDATE,INSERT ON sportsmedicalcenter.Equipa TO 'profsaude'@'localhost';
GRANT SELECT ON sportsmedicalcenter.Boletim_Clinico TO 'profsaude'@'localhost';
GRANT SELECT ON sportsmedicalcenter.Profissional_de_saude TO 'profsaude'@'localhost';
GRANT SELECT ON sportsmedicalcenter.Especialidade TO 'profsaude'@'localhost';
GRANT SELECT ON sportsmedicalcenter.Teste_Clinico TO 'profsaude'@'localhost';

-- Permissão para a execução de procedimentos e funções .
GRANT EXECUTE ON PROCEDURE sportsmedicalcenter.addProfSaudeEquipaBoletim TO 'profsaude'@'localhost';
GRANT EXECUTE ON PROCEDURE sportsmedicalcenter.TestesProfDia TO 'profsaude'@'localhost';
GRANT EXECUTE ON FUNCTION sportsmedicalcenter.numHoras TO 'profsaude'@'localhost';

```

Deste modo procuramos aliar a simplicidade organizacional desta base de dados a um nível mais sofisticado de segurança, crucial para o funcionamento e fidedignidade deste sistema.

6. Utilização e migração para o Neo4j

6.1. Justificação da utilização de um sistema NoSQL

A implementação deste base de dados em neo4j possibilita que a realização de interrogações que envolvem junção de muitas relações se façam de uma forma extremamente rápida, dado que é escalável e rápida no armazenamento e leitura de grandes quantidades de dados ,bem como a visualização de um grafo em algumas interrogações torna mais fácil ao cliente a sua análise.

6.2 Migração

Para migrar a base de dados relacional para uma base dados orientada a grafos. Através da documentação e ferramentas do *neo4j*, tivemos que ter atenção a algumas regras. Portanto, cada tabela com uma chave estrangeira é tratada como *join* e importada como um nodo com relacionamento, uma tabela com duas chaves estrangeiras é tratada como uma tabela *join* e importada como relação, tabelas com mais de 2 chaves estrangeiras são tratadas como um nodo intermediário

Através da ferramenta neo4j etl tool, só tivemos que estabelecer uma conexão através do conector j do *mysql* para o nosso grafo do neo4j. O programa realizou um mapeamento entre tabelas *mysql* para neo4j de acordo com as regras que enunciamos acima e como o mapeamento se encontrava todo correto de acordo com

as regras não foi preciso eliminar nenhuma tabela ou relação (ou mesmo apagar chaves).

Exemplo de ficheiros gerados:

```

[ ] NODE_sportsmedicalcenter.atleta_c6351faa-5aac-4f9e-9ef2-902903d09c5c
[ ] NODE_sportsmedicalcenter.atleta_c6351faa-5aac-4f9e-9ef2-902903d09c5c_headers
[ ] NODE_sportsmedicalcenter.boletimclinico_77d024bb-27ee-4edc-ae02-e9e833b28971
[ ] NODE_sportsmedicalcenter.boletimclinico_77d024bb-27ee-4edc-ae02-e9e833b28971_headers
[ ] NODE_sportsmedicalcenter.clube_66eecadd-b858-4e39-9b9d-9472091e0178
[ ] NODE_sportsmedicalcenter.clube_66eecadd-b858-4e39-9b9d-9472091e0178_headers
[ ] NODE_sportsmedicalcenter.codigopostal_122910c8-e1c4-41d2-a962-c8a489e3438a
[ ] NODE_sportsmedicalcenter.codigopostal_122910c8-e1c4-41d2-a962-c8a489e3438a_headers
[ ] NODE_sportsmedicalcenter.especialidade_dea9abcb-e4d2-4e79-aab8-36c4243f4c0b
[ ] NODE_sportsmedicalcenter.especialidade_dea9abcb-e4d2-4e79-aab8-36c4243f4c0b_headers
[ ] NODE_sportsmedicalcenter.modalidade_a2d73f91-53a2-41c8-af70-4e99dbcc540b
[ ] NODE_sportsmedicalcenter.modalidade_a2d73f91-53a2-41c8-af70-4e99dbcc540b_headers
[ ] NODE_sportsmedicalcenter.profissionaldesaude_607e759a-cb29-4f24-9020-048460ebbcaf
[ ] NODE_sportsmedicalcenter.profissionaldesaude_607e759a-cb29-4f24-9020-048460ebbcaf_he...
[ ] NODE_sportsmedicalcenter.testeclinico_5c6cd1f5-74cf-4f36-a248-c2a1d6ca0d78
[ ] NODE_sportsmedicalcenter.testeclinico_5c6cd1f5-74cf-4f36-a248-c2a1d6ca0d78_headers
[ ] REL_atleta_f0041595-3ab6-4beb-a1fb-79041b8e6f40
[ ] REL_atleta_f0041595-3ab6-4beb-a1fb-79041b8e6f40_headers
[ ] REL_ATLETAHASMODALIDADE_a381fbd1-28d3-4d89-a9d8-6a6ee7c36c85
[ ] REL_ATLETAHASMODALIDADE_a381fbd1-28d3-4d89-a9d8-6a6ee7c36c85_headers
[ ] REL_clube_8fcf756d-0730-4db8-ac7b-4ba80d6c13b6
[ ] REL_clube_8fcf756d-0730-4db8-ac7b-4ba80d6c13b6_headers
[ ] REL_codigopostal_16da9f51-ce06-452e-a529-b34c5289ddb7
[ ] REL_codigopostal_16da9f51-ce06-452e-a529-b34c5289ddb7_headers
[ ] REL_codigopostal_d94e85a0-bf98-4ca8-adbf-654ab7064cc3
[ ] REL_codigopostal_d94e85a0-bf98-4ca8-adbf-654ab7064cc3_headers
[ ] REL_EQUIPA_40528685-ddd0-47d1-a5b0-dd9e22343234
[ ] REL_EQUIPA_40528685-ddd0-47d1-a5b0-dd9e22343234_headers
[ ] REL_especialidade_85e93b14-80bc-4aab-ab7a-237ac8e10447
[ ] REL_especialidade_85e93b14-80bc-4aab-ab7a-237ac8e10447_headers

```

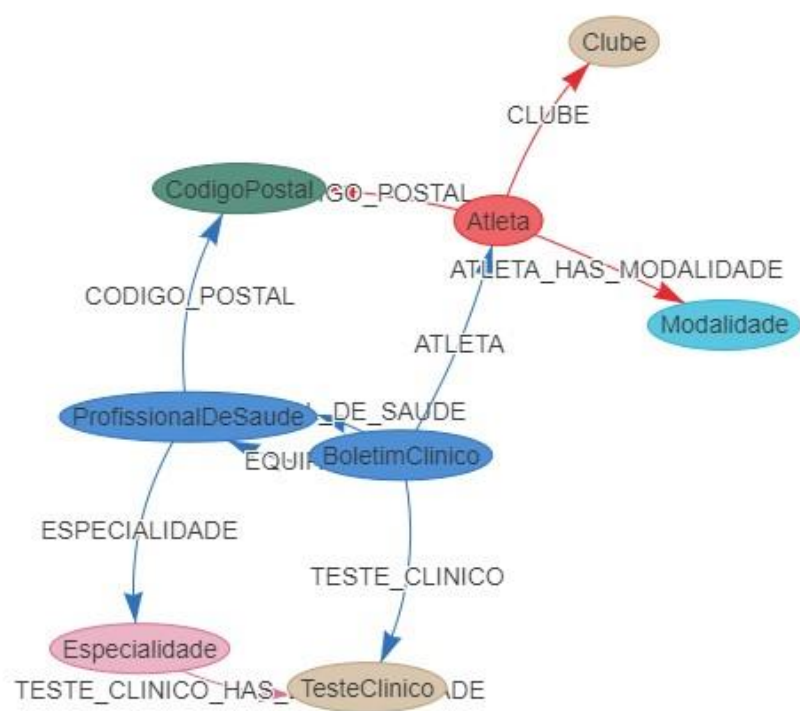
A importação de povoamento que a ferramenta *neo4j etl* utiliza é simples e gerou os csv's das tabelas, sendo gerados da seguinte maneira:

```

USE testeclinico;
| SELECT * FROM Atleta
INTO OUTFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/table.csv'
  FIELDS ENCLOSED BY '"' TERMINATED BY ',' ESCAPED BY ''
  LINES TERMINATED BY '\r\n';

```


Assim ficamos com o seguinte grafo



6.3 Tradução das interrogações do utilizador para Cypher

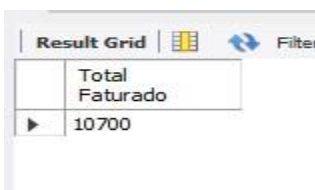
Para esta etapa, usaremos algumas queries propostas pelo utilizador que foram seleccionadas na primeira fase. Essas queries serão agora expressas em Cypher e em SQL para podermos verificar se os resultados são semelhantes em ambos os modelos:

Total Faturado pela clinica:

SQL

```
-- Total faturado pela clinica
DROP VIEW IF EXISTS totFaturado;
CREATE VIEW totFaturado AS
    SELECT sum(t.preco) AS `Total Faturado` FROM Boletim_Clinico bc,Teste_Clinico t
    WHERE t.id=bc.testeclinico_id AND bc.pago=1;

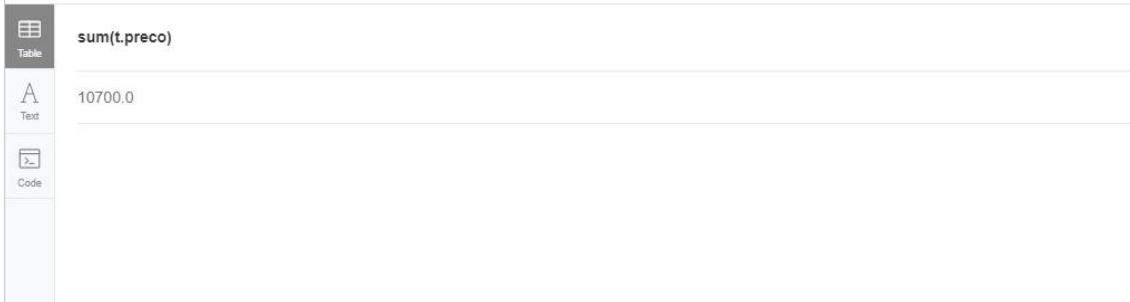
SELECT * FROM totFaturado;
```



Total Faturado
10700

Cypher

```
$ MATCH (b:BoletimClinico)-[TESTE_CLINICO]-(t:TesteClinico) WHERE( b.pago=1 ) return sum(t.preco)
```



sum(t.preco)
10700.0

Testes mais realizados na clinica(top 5)

SQL

```
-- 5 testes mais realizados na clinica
```

```
DROP VIEW IF EXISTS top5testes;  
CREATE VIEW top5testes AS  
SELECT t.nome,count(bc.testeclinico_id) AS `Num testes realizados` FROM Boletim_Clinico bc  
INNER JOIN Teste_Clinico t ON t.id=bc.testeclinico_id  
GROUP BY t.nome  
ORDER BY `Num testes realizados` DESC  
LIMIT 5;
```

```
SELECT * FROM top5testes;
```

	nome	Num testes realizados
►	Exame Médico Desportivo	6
	Eletrocardiograma	4
	Raio X	4
	Ergoespiometria	2
	Baropodometria	2

Cypher

```
$ MATCH (b:BoletimClinico)-[TESTE_CLINICO]-(t:TesteClinico) return count (t),t.nome ORDER BY count(t) DESC LIMIT(5)
```

count (t)	t.nome
6	"Exame Médico Desportivo"
4	"Raio X"
4	"Eletrocardiograma"
2	"Teste de Sangue"
2	"Teste Intolerância alimentar"

Especialidade com mais teste pelos quais foi responsável

SQL

```
SELECT e.designacao,count(e.id) FROM Especialidade e,Boletim_Clinico bc,Profissional_de_Saude ps
WHERE ps.id=bc.profsaude_id AND e.id=ps.especialidade_id
GROUP BY e.designacao
ORDER BY count(e.id) DESC;
```

	designacao	count(e.id)
▶	Enfermagem	8
	Endocrinologia e Nutrição	6
	Medicina Desportiva	3
	Fisioterapeuta	2
	Psiquiatria	2
	Cirurgia Geral	2
	Cardiologia	1
	Oftalmologia	1

Cypher

```
$ MATCH(b:BoletimClinico)-[prof:PROFISSIONAL_DE_SAUDE]- (p:ProfissionalDeSaude)-[ESPECIALIDADE]-(e:Especialidade) with e,count(e.id) as frequency return frequency,e.designacao ORDER BY frequency DESC
```

frequency	e.designacao
8	"Enfermagem"
6	"Endocrinologia e Nutriçã"
3	"Medicina Desportiva"
2	"Fisioterapeuta"
2	"Psiquiatria"
2	"Cirurgia Geral"
1	"Cardiologia"
1	"Oftalmologia"

Started streaming 8 records after 14 ms and completed after 16 ms.

Testes realizados por cada atleta

SQL

```
-- Teste clinico de cada atleta
```

```
SELECT a.nome,t.nome,t.id,t.preco FROM Teste_Clinico t,Boletim_Clinico b,Atleta a
WHERE b.atleta_nif=a.nif AND t.id=b.testeclinico_id;
```

	nome	nome	id	preco
▶	Mariana Pinho de Mendes	Exame Médico Desportivo	1	50
	Ana Figueiredo Anjos	Teste de Sangue	4	200
	Daniel Azevedo Anjos	Teste ADN Desporto	7	750
	Alexandra Débora Teixeira Pacheco	Eletrocardiograma	5	400
	Cláudio Branco Cunha	Ergoespiometria	3	30
	Salvador António de Leal	Raio X	9	750
	Margarida Cardoso	Teste Antidoping	8	900
	Tiago Márcio Gomes de Gomes	Teste intolerância alimentar	6	1450
	Mariana Pinho de Mendes	Exame Médico Desportivo	1	50
	Mariana Pinho de Mendes	Eletrocardiograma	5	400
	Mariana Pinho de Mendes	Exame Médico Desportivo	1	50

Cypher

```
$ MATCH (a:Atleta) -[ATLETA]-(b:BoletimClinico)-[TESTE_CLINICO]-(t:TesteClinico) return a.nome,t.nome,t.id,t.preco;
```

	"a.nome"	"t.nome"	"t.id"	"t.preco"
Table	"Mariana Pinho de Mendes"	"Exame Médico Desportivo"	1	50.0
Text	"Mariana Pinho de Mendes"	"Teste ADN Desporto"	7	750.0
Code	"Mariana Pinho de Mendes"	"Raio X"	9	750.0
	"Mariana Pinho de Mendes"	"Exame Médico Desportivo"	1	50.0
	"Mariana Pinho de Mendes"	"Teste de Sangue"	4	200.0
	"Mariana Pinho de Mendes"	"Eletrocardiograma"	5	400.0
	"Mariana Pinho de Mendes"	"Exame Médico Desportivo"	1	50.0
	"Mariana Pinho de Mendes"	"Exame Médico Desportivo"	1	50.0
	"Ana Figueiredo Anjos"	"Teste de Sangue"	4	200.0
	"Cláudio Branco Cunha"	"Ergoespiometria"	3	30.0
	"Márcio Garcia Fonseca"	"Eletrocardiograma"	5	400.0

Conclusões e trabalho futuro

O nosso grupo avalia de forma positiva todo o trabalho desenvolvido neste projeto. Foi bastante notória a evolução sentida, não só a nível prático como na percepção teórica dos conceitos que a área de Bases de Dados implica. As dificuldades sentidas remeteram-se quase exclusivamente a divergências de interpretação em relação ao problema exposto, mas também na migração para o “*neo4j*”, onde a falta de contacto com a ferramenta mostrou ter algum impacto a nível de eficiência produtiva. A necessidade de elaborar um projeto com fortes tendências profissionais trouxe muita maturidade e percepção do que é ter de trabalhar em grupo e respeitar prazos e etapas. A nível futuro ambicionam-se projetos de maior complexidade, com maiores bases de dados e relações, trazendo assim uma maior maturação das capacidades que futuramente, a nível profissional, se revelarão extremamente úteis. O rigor e seriedade com que este projeto foi abraçado pelos elementos do grupo traz-nos satisfação e felicidade ao ver que o nos fora exigido conseguiu ser implementado de forma eficiente e clara.

Referências

- Connolly, T.M. and Begg C. E., 2005. Database Systems A Practical Approach to Design, Implementation, and Management. 6th Edition, England: Pearson Education Limited.

Anexos

<<Os anexos deverão ser utilizados para a inclusão de informação adicional necessária para uma melhor compreensão do relatório o para complementar tópicos, secções ou assuntos abordados. Os anexos criados deverão ser numerados e possuir uma designação. Estes dados permitirão complementar o Índice geral do relatório relativamente à enumeração e apresentação dos diversos anexos.>>

I. Anexo 1

7. Lista de Siglas e Acrónimos

SQL - Structured Query Language

BD – Bases de Dados

SBD – Sistema de Base de Dados