



UNIVERSIDADE DO MINHO

MESTRADO INTEGRADO EM ENGENHARIA INFORMÁTICA  
CIÊNCIAS DA COMPUTAÇÃO

BASE DE DADOS

---

## Ginásio do Miguel

---

Adelino Miguel Alves Silva, A82549  
Henrique José Carvalho Faria, A82200  
Luís Filipe da Costa Cunha, A83099  
Miguel Ângelo Moreira Ramos Brandão, A82349

27 de Novembro de 2018

## **Resumo**

Este projeto consiste na criação de uma base de dados para o Sr.Miguel, o dono de um pequeno ginásio que viu um aumento exponencial no número de clientes nas últimas semanas e como tal necessitou de uma estrutura para organizar os seus dados.

Com o intuito, de criar uma base de dados que satisfaça todas as necessidades requeridas pelo nosso cliente, e para que esta consiga suportar e organizar todos os dados necessários, de forma correta e moderada, foi realizada uma pesquisa exaustiva acerca do funcionamento do ginásio do senhor Miguel. Esta pesquisa consistiu em várias observações presenciais, contacto com os operários regulares do ginásio e claro, esclarecimento, com a administração do ginásio, dos principais objetivos que o ginásio pretendia alcançar com a implementação desta base de dados.

No final da análise presencial, e do levantamento de todos os requisitos necessários para cumprir com as expetativas do nosso cliente, começamos a desenvolver a base de dados. Inicialmente elaboramos um modelo conceptual, para confirmar se os requerimentos levantados iam ao encontro do pretendido pelo Sr.Miguel. Depois de todo o modelo estar devidamente corrigido e validado, procedemos para a criação do modelo lógico. Aqui foi onde começamos a dar forma à estrutura lógica da base de dados. Transformamos toda a informação visual do modelo conceptual em estruturação lógica para finalmente, depois da validação e confirmação do modelo lógico, passarmos à criação do modelo Físico. Foi nesta fase que demos origem a todas as funcionalidades requeridas pelo nosso cliente. Finalmente, o cliente pode ver a base de dados em funcionamento, realizando todas as operações por ele requeridas e guardando, organizando e gerindo todos os dados do seu Ginásio.

Toda a metodologia utilizada ao longo do trabalho é a apresentada pelo livro "A Practical Approach to Design, Implementation, and Management."[\[CB15\]](#)

## **Palavras Chave**

Sistema de base de dados relacional, Modelo Conceptual, Modelo Lógico, Modelo Físico, Notação Peter Chen, Modelo ER, Entidades, Atributos, Relações entre entidades.

# Conteúdo

<b>1</b>	<b>Definição do Sistema</b>	<b>6</b>
1.1	Contexto da Aplicação do Sistema	6
1.2	Fundamentação da Implementação da Base de Dados	6
1.3	Análise de Viabilidade do Processo	6
<b>2</b>	<b>Levantamento e análise de requisitos</b>	<b>7</b>
2.1	Métodos de levantamento e de análise de requisitos adotados	7
2.2	Requisitos Levantados	7
2.2.1	Requisitos de Descrição	7
2.2.2	Requisitos de Exploração	8
2.2.3	Requisitos de Controlo	9
2.3	Análise geral dos Requisitos	9
<b>3</b>	<b>Modelação Conceptual</b>	<b>10</b>
3.1	Apresentação da abordagem de modelação realizada	10
3.2	Identificação e caracterização das entidades	10
3.3	Identificação e caracterização dos relacionamentos	11
3.4	Identificação e caracterização das Associação dos Atributos com as Entidades e Relacionamentos.	12
3.5	Detalhe ou generalização de entidades	13
3.6	Apresentação e explicação do diagrama ER	14
3.7	Validação do modelo de dados com o utilizador	14
<b>4</b>	<b>Modelação Lógica</b>	<b>16</b>
4.1	Derivar relações para o modelo de dados lógicos	16
4.2	Desenho do modelo lógico	17
4.3	Validação do modelo através da normalização	18
4.3.1	Dependências Funcionais	18
4.3.2	Primeira Forma Normal (1FN)	18
4.3.3	Segunda Forma Normal (2FN)	18
4.3.4	Terceira Forma Normal (3FN)	19
4.4	Validação do modelo com interrogações do utilizador	19
4.5	Validação do modelo com as transações estabelecidas	20
4.6	Revisão do modelo lógico com o utilizador	21
<b>5</b>	<b>Implementação Física</b>	<b>22</b>
5.1	Seleção do sistema de gestão de bases de dados	22
5.2	Tradução do esquema lógico para o sistema de gestão de bases de dados escolhido em SQL	23
5.3	Tradução das interrogações do utilizador para SQL (alguns exemplos)	23
5.3.1	Interrogação faturação num mês	23
5.3.2	Todos os clientes que fizeram Martelo	23
5.3.3	A que serviços o cliente 1 esta subscrito	24
5.3.4	Faturas correspondentes ao cliente 2	24
5.4	Tradução das transações estabelecidas para SQL (alguns exemplos)	25
5.5	Escolha, definição e caracterização de índices em SQL (alguns exemplos)	26
5.6	Estimativa do espaço em disco da base de dados e taxa de crescimento anual	27

5.7	Definição e caracterização das vistas de utilização em SQL (alguns exemplos) . . .	28
5.8	Definição e caracterização dos mecanismos de segurança em SQL (alguns exemplos)	31
5.9	Revisão do sistema implementado com o utilizador . . . . .	31
<b>6</b>	<b>Conclusões e Trabalho Futuro</b>	<b>32</b>
	<b>Bibliografia</b>	<b>34</b>

# Lista de Figuras

3.1	Modelo Conceptual.	14
4.1	Modelo Lógico	17
5.1	Modelo Lógico representado no MySQL Workbench	23

# Lista de Tabelas

3.1	Identificação e caracterização dos atributos das entidades. . . . .	12
3.2	Identificação e caracterização dos atributos dos relacionamentos. . . . .	13
3.3	Identificação de candidatos a chave primária. . . . .	13

# Capítulo 1

## Definição do Sistema

### 1.1 Contexto da Aplicação do Sistema

O Miguel gosta muito de fazer exercício. Ultrajado com o preço dos ginásios na sua área, ele decide investir no seu próprio equipamento e começar um pequeno ginásio pessoal. Sendo um estudante de Engenharia Informática extremamente popular entre o público feminino, começou a receber diversos pedidos das suas amigas para também poderem treinar no seu ginásio. Sabendo disto, alguns outros estudantes demonstraram interesse em treinar no ginásio do Miguel.

### 1.2 Fundamentação da Implementação da Base de Dados

Com o aumento de clientes, o ginásio teve a necessidade de expandir as suas instalações, comprar novo equipamento e também recrutar novos funcionários, para conseguir satisfazer a necessidade dos seus clientes. Como tal, Miguel concluiu que era conveniente armazenar a informação detalhada dos seus funcionários e equipamentos.

Com a intenção de criar uma ligação mais próxima com os clientes, o Miguel intendeu que se devia guardar também a informação do cliente. Uma ação como mandar um postal pelo Natal, ou pelo aniversário e mandar mensagens para incentivar o treino era impossível sem guardar os dados dos clientes.

### 1.3 Análise de Viabilidade do Processo

Trata-se de um modelo genérico de um sistema de base de dados, desenvolve-se em torno de um fornecedor(O Miguel), serviço(todas as atividades possíveis desenvolver no ginásio, desde personal training a planos de nutrição) e cliente, ou seja, o processo de desenvolvimento desta base de dados é bastante simples e como tal parece-nos um sistema de base de dados fácil de implementar.

## Capítulo 2

# Levantamento e análise de requisitos

### 2.1 Métodos de levantamento e de análise de requisitos adotados

Durante a fase de coleta de requisitos do sistema foram usadas diversas técnicas, como a consulta de documentos utilizados (memorandos, emails, queixas de empregados face ao sistema e críticas/avaliação da performance do sistema), para perceber a origem da necessidade de uma base de dados. O dono, futuro administrador da base de dados, listou as funções que acha necessárias e deu o seu parecer acerca do funcionamento normal do ginásio. Foram realizadas entrevistas adicionais com os funcionários do estabelecimento para verificação dos requisitos já identificados, bem como clarificação dos mesmos e possíveis adições de novos requisitos que fossem de encontro às necessidades destes. Por fim, a equipa de levantamento de requisitos procedeu à coleta de dados por observação do normal funcionamento do ginásio através do sistema de vigilância, visto que o facto de um funcionário saber que está a ser observado pode interferir na forma como uma atividade é executada, e como clientes, de modo a ter uma perceção de como a base de dados poderia influenciar a experiência de um cliente.

### 2.2 Requisitos Levantados

#### 2.2.1 Requisitos de Descrição

Os requisitos de descrição necessários à base de dados recolhidos pela equipa foram :

1. É necessário guardar o nome, sexo, idade, Serviço Prestado, contacto, email e Morada do funcionário.
2. Um Serviço é constituído por um nome e pelo respetivo preço.
3. Temos de guardar todos os dados pessoais do cliente (nome, sexo, idade, contacto, email, morada, Número de Contribuinte) e *IMC*, Peso, Altura, e Limitações Físicas.
4. Os exercícios são compostos pelo seu tipo e nome associado ao mesmo.
5. O equipamento tem nome e um propósito.
6. A fatura é composta pelo numero de contribuinte empresa, data de emissão, descrição, valor, número contribuinte do cliente, serviço prestado.
7. O cliente tem um conjunto de de exercícios, que formam um plano de exercícios.
8. O plano de exercícios é composto pelo número de repetições e número de series.
9. O Cliente antes de ter acesso aos serviços fornecidos no ginásio tem de se registar nos mesmos, registando a data de subscrição.
10. O ginásio tem de ter a funcionalidade de emitir fatura dos serviços fornecidos ao cliente.



11. Após a subscrição de um ou vários serviços, é submetida uma fatura desses serviços, sendo necessário guardar essas relações para futura consulta.
12. Existem funcionários a fornecer serviços específicos.
13. Deve ser registado o momento em que o funcionário se torna responsável pela prestação do serviço.
14. O cliente tem sempre exercícios, a menos q este se subscreve a um serviço de natação ou a um serviço de nutrição, e estes últimos são sempre diferentes de pessoa para pessoa.
15. Para todo o exercício existe também um equipamento adequado para a sua execução.

### **2.2.2 Requisitos de Exploração**

Após a verificação das necessidades dos funcionários e do responsável do ginásio para permitir um bom funcionamento deste, chegamos às seguintes conclusões:

1. Terá de ser possível ao responsável do ginásio adicionar um funcionário.
2. Terá também de ser possível ao administrador do ginásio alterar a informação relativa a um funcionário;
3. Para posterior confirmação dos updates ou adições referentes ao funcionário o administrador deverá poder visualizar o conteúdo da tabela dos funcionários.
4. Tem de ser possível visualizar que serviços são prestados por que funcionários.
5. O responsável do ginásio poderá também consultar os serviços fornecidos pelo ginásio.
6. Outra das funções do administrador do ginásio será adicionar novos serviços que poderão vir a ser disponibilizados aos clientes.
7. Este terá também a obrigação de marcar como não disponíveis os serviços que entretanto forem descontinuados ou os quais o ginásio já não suporte.
8. Terá de ser possível ao funcionário registar um novo cliente;
9. Existe a necessidade de aceder à ficha do cliente pelo número de utilizador.
10. Também deverá ser possível aceder á ficha do cliente através do seu número de telefone, caso este não saiba o seu idCliente;
11. Terá de ser possível ao funcionário consultar os exercícios de um cliente.
12. O funcionário deverá ter permissões para adicionar novos exercícios a um cliente.
13. Para além disso terá de ser também disponibilizada forma de apagar os exercícios associados a um cliente do seu plano de treinos;
14. Deverá ser possível a consulta de todas as faturas emitidas pelo ginásio relativas a um cliente.
15. A consulta de faturas entre duas datas específicas será também uma funcionalidade presente na base de dados;
16. Ao funcionário reserva-se o dever de emitir faturas.
17. Ao funcionário reserva-se também o dever de marcar uma fatura como inválida sempre que esta for mal inserida no sistema.
18. O responsável reserva-se o dever de ter acesso ao total faturado num determinado período de tempo ou durante todo o tempo de vida do ginásio.
19. Cabe ao funcionário designar qual o equipamento a ser utilizado num determinado exercício no momento da inserção deste na base de dados.

20. Quer o administrador quer os rececionistas devem poder ver que serviços foram registados em cada fatura.
21. Os funcionários devem poder também ver quais as subscrições dos clientes.
22. No ginásio é requerida a necessidade de poder adicionar novos equipamentos, isto poderá ser feito quer pelos funcionários quer pelo administrador do ginásio.
23. Um funcionário pode decidir adicionar um novo exercício aos atualmente disponíveis para os clientes.
24. O administrador será responsável por referenciar que funcionários executam que serviço.

### 2.2.3 Requisitos de Controlo

1. Horário de acesso:

Tanto o administrador como os funcionários podem aceder à base de dados durante o horário de funcionamento do ginásio, das 8 da manhã até á meia noite.

2. Funcionários, fornecedores e Serviços:

Apenas o administrador poderá consultar, adicionar ou remover dados referentes a estas entidades.

3. Cliente:

Um funcionário tem total acesso ao cliente podendo adicionar, remover ou modificar os dados de um cliente.

4. Equipamento:

Tanto o funcionário como o administrador podem alterar os dados referentes a um equipamento.

5. Exercícios:

Apenas o funcionário poderá adicionar, modificar ou remover um exercício.

6. Faturas:

Tanto o administrador como o funcionário podem consultar e alterar as faturas. Mas apenas um administrador pode remover uma.

## 2.3 Análise geral dos Requisitos

Para esta base de dados realizou-se uma combinação das visões do administrador, dos funcionários e dos clientes utilizando a *Centralized approach* pois o sistema a modelar não é muito complexo e as vistas dos utilizadores sobrepõem-se em vários aspetos. (Apesar dos clientes não operarem sobre a base de dados é importante saber as suas necessidades como obter a lista de exercícios correspondentes ao seu plano de treino)

Ao realizar uma análise mais profunda dos requisitos podemos inferir que o principal interveniente do sistema é o funcionário visto que este lida com os dados dos clientes, exercícios, faturas e equipamentos do ginásio. Podemos também verificar que a entidade central é o Cliente pois este realiza exercícios, subscreve um ou mais serviços e paga pelos mesmos.

## Capítulo 3

# Modelação Conceptual

### 3.1 Apresentação da abordagem de modelação realizada

A abordagem escolhida para modelar a base de dados do senhor Miguel baseai-se numa modelação conceptual, que é independente de qualquer implementação que envolva aplicações, linguagens de programação, interfaces de hardware, etc. Para isso, iremos utilizar um modelo abstrato de Entidade-Relacionamento (ER) com a notação de **Peter Chen** para elaborar um diagrama de entidades, com os seus atributos, e relacionamentos entre elas.

Primeiramente, fizemos o levantamento dos elementos principais do nosso modelo, as entidades. Para isso, um dos métodos que usamos foi, por exemplo, analisar os requisitos seleccionando os elementos mais destacados.

Em segundo lugar, identificamos de que forma as entidades anteriormente seleccionadas se relacionam entre si para assim, criar-mos relacionamentos entre elas. Mais uma vez, fizemo-lo analisando as expressões usadas nos requerimentos levantados.

Em terceiro lugar, procedemos à identificação de atributos das entidades ou relacionamentos necessários à base de dados. Para isso identificamos características, qualidades e propriedades das varias entidades e relacionamentos, anteriormente levantados, e seleccionamos as mais relevantes para o problema em questão.

Em quarto lugar, identificamos, para cada atributo, o seu domínio.

Em quinto lugar, para cada entidade, seleccionamos, entre todos os seus atributos, a melhor chave primária.

Seguidamente, verificamos a existência de redundância no nosso modelo, ou seja, analisamos entidade a entidade de forma a confirmar que não tínhamos entidades repetidas, depois focamo-nos nos relacionamentos para garantir que todos eles são imprescindíveis, ou seja, existe pelo menos um caso em que apenas aquele relacionamento me poderia dar acesso àquela informação.

Finalmente, procedemos á validação de todo o modelo ao lado do senhor Miguel.

### 3.2 Identificação e caracterização das entidades

Para identificarmos as várias entidades do nosso modelo conceptual, uma das abordagens tomadas foi a análise rigorosa dos requerimentos levantados. Nomes principais referentes a um ginásio ou objetos os quais sejam bastantes referidos ou detalhados nos requerimentos feitos pelo senhor Miguel tendem a ser bons candidatos a entidades.

Para além disso, procuramos ter em atenção entidades auto suficientes, ou seja, entidades que existem por si próprias, independentes de qualquer outra entidade. Exemplo disso é a entidade Funcionário, pois esta existe, independentemente de sabermos quem são, a sua idade, ou o seu cargo. Nesta fase, consideramos que a participação e acompanhamento do senhor Miguel foi essencial sendo que, por vezes, foi necessário um melhor esclarecimento dos requisitos de forma a termos um modelo conceptual mais preciso.

### 3.3 Identificação e caracterização dos relacionamentos

A fim de identificar os vários relacionamentos entre as entidades já anteriormente identificadas iremos mais uma vez recorrer à observação dos requerimentos. Desta vez iremos adotar uma análise diferente, que consiste na observação das expressões usadas, focada nos verbos utilizados para relacionar as nossas identidades.

Assim, os relacionamentos que selecionamos são:

Entidade	Relacionamento	Entidade	Cardinalidade
Cliente	Plano de Exercícios	Exercício	(0,N) - (0,N)
Cliente	Tem	Fatura	(1,1) - (0,N)
Cliente	Subscreve	Serviço	(0,N) - (1,N)
Funcionário	Passa	Fatura	(1,1) - (1,N)
Equipamento	É utilizado para	Exercício	(1,N) - (0,N)
Serviço	É prestado por	Funcionário	(0,N) - (0,N)
Serviço	Constitui	Fatura	(1,N) - (0,N)

### 3.4 Identificação e caracterização das Associação dos Atributos com as Entidades e Relacionamentos.

No final de reunir todas as entidades e relacionamentos relevantes para o nosso modelo, procedemos então à identificação e caraterização de todos os atributo necessários de modo a cumprir com o requerimentos estabelecidos. Na tabela 3.1 é possível visualizar todos os atributos de cada entidade e as caraterísticas associadas a cada um deles.

Entidade	Atributos	Descrição	Chave Primária	Domínio	Tipo de dados	Nulo	Multivalor	Composto	Derivado
Cliente	ClienteID	Identificação do cliente	Sim	Até 5 dígitos	INT	Não	Não	Não	Não
	Nome	Nome do cliente	Não	Até 75 caracteres	VARCHAR(75)	Não	Não	Não	Não
	Idade	Idade do cliente	Não	Até 3 dígitos	INT	Não	Não	Não	Não
	Morada	Morada do cliente	Não	-	-	Não	Não	Sim	Não
	Localidade	Localidade do cliente	Não	Até 75 caracteres	VARCHAR(75)	Não	Não	Não	Não
	Rua	Rua do cliente	Não	Até 75 caracteres	VARCHAR(75)	Não	Não	Não	Não
	Codigo Postal	Codigo Postal do Cliente	Não	1 caractere "-" + 3 dígitos	VARCHAR(8)	Não	Não	Não	Não
	Contacto	Contacto do cliente	-	-	-	-	Não	Sim	Não
	Telemovel 1	Nº de telefone do cliente	Não	9 dígitos	INT	Não	Não	Não	Não
	Telemovel 2	Segundo nº de telefone do cliente	Não	9 dígitos	INT	Sim	Não	Não	Não
	Email	Email do cliente	Não	Até 75 caracteres	VARCHAR(75)	Não	Não	Não	Não
	IMC	Índice de massa corporal do cliente	Não	Até 4 dígitos (2 decimais)	DECIMAL(4,2)	Não	Não	Não	Sim
	Peso	Peso do cliente	Não	Até 5 dígitos (2 decimais)	DECIMAL(5,2)	Não	Não	Não	Não
	Altura	Altura do cliente	Não	Até 4 dígitos (2 decimais)	DECIMAL(4,2)	Não	Não	Não	Não
Funcionário	Nr Contribuinte	Nº de contribuinte do cliente	Não	9 dígitos	INT	Não	Não	Não	Não
	Sexo	Sexo do cliente	Não	1 caractere (F/M)	VARCHAR(1)	Não	Não	Não	Não
	Dependencias Fisicas	Dependências físicas do cliente	Não	Até 45 caracteres	VARCHAR(45)	Sim	Não	Não	Não
	FuncionárioID	Identificação do funcionário	Sim	Até 4 dígitos	INT	Não	Não	Não	Não
	Nome	Nome do funcionário	Não	Até 45 caracteres	VARCHAR(45)	Não	Não	Não	Não
	Cargo	Cargo que o funcionário ocupa no ginásio	Não	Até 75 caracteres	VARCHAR(75)	Não	Não	Não	Não
	Idade	Idade do funcionário	Não	Até 3 dígitos	INT	Não	Não	Não	Não
	Contacto	Contacto do funcionário	-	-	-	-	-	Sim	Não
	Telemovel 1	Nº do telefone do funcionário	Não	9 dígitos	INT	Não	Não	Não	Não
	Telemovel 2	Segundo nº de telefone do funcionário	Não	9 dígitos	INT	Sim	Não	Não	Não
	Email	Email do funcionário	Não	Até 75 caracteres	VARCHAR(75)	Não	Não	Não	Não
	Morada	Morada do funcionário	Não	-	-	-	-	Sim	Não
	Rua	Rua onde o funcionário habita	Não	Até 75 caracteres	VARCHAR(75)	Não	Não	Não	Não
	Localidade	Localidade onde o funcionário habita	Não	Até 75 caracteres	VARCHAR(75)	Não	Não	Não	Não
Serviço	Codigo Postal	Código Postal do funcionário	Não	4 dígitos + 1 caractere (-) + 3 dígitos	VARCHAR(8)	Não	Não	Não	Não
	Estado	Estado em que se encontra o funcionário (se ainda trabalha ou não no ginásio)	Não	1 caractere (A/N)	VARCHAR(1)	Não	Não	Não	Não
	ServçoID	Identificação do Serviço	Sim	Até 3 dígitos	INT	Não	Não	Não	Não
	Preço	Preço do Serviço	Não	Até 5 dígitos (2 decimais)	DECIMAL(5,2)	Não	Não	Não	Não
Fatura	Designação	Especificação do serviço	Não	Até 45 caracteres	VARCHAR(45)	Não	Não	Não	Não
	Estado	Estado em que se encontra o funcionário (se ainda trabalha ou não no ginásio)	Não	1 caractere (A/N)	VARCHAR(1)	Não	Não	Não	Não
	IdFatura	Identificação da Fatura	Sim	Até 9 dígitos	INT	Não	Não	Não	Não
	Contribuinte Ginásio	Nº de contribuinte do Ginásio	Não	9 dígitos	INT	Não	Não	Não	Não
Equipamento	Data	Data de emissão da fatura	Não	2 dígitos + 1 caractere (/) + 2 dígitos + 1 caractere (/) + 4 dígitos	DATE	Não	Não	Não	Não
	Desconto	Desconto associado à fatura	Não	Até 2 dígitos decimais	DOUBLE	Não	Não	Não	Não
	Descrição	Serviços prestados ao cliente	Não	Até 300 caracteres	VARCHAR(300)	Não	Não	Não	Não
	Valor	Valor a ser cobrado pelos serviços prestados	Não	Até 5 dígitos (2 decimais)	Decimal(5,2)	Não	Não	Não	Sim
	idEquipamento	Identificação de Equipamento	Sim	Até 3 dígitos	INT	Não	Não	Não	Não
Exercício	Nome	Nome do equipamento	Não	Até 45 dígitos	VARCHAR(45)	Não	Não	Não	Não
	Descrição	Descrição do equipamento	Não	Até 75 dígitos	VARCHAR(75)	Não	Não	Não	Não
	Estado	Estado em que se encontra o funcionário (se ainda trabalha ou não no ginásio)	Não	1 caractere (A/N)	VARCHAR(1)	Não	Não	Não	Não
Exercício	ExercícioID	Identificação do exercício	Sim	Até 3 dígitos	INT	Não	Não	Não	Não
	Designação	Nome do exercício	Não	Até 45 dígitos	VARCHAR(45)	Não	Não	Não	Não
	Tipo de Exercício	Tipo de exercício a ser praticado	Não	Até 45 dígitos	VARCHAR(45)	Não	Não	Não	Não

Tabela 3.1: Identificação e caraterização dos atributos das entidades.

Seguidamente, podemos analisar os atributos dos relacionamentos coletados.

Relacionamento	Atributos	Descrição	Domínio	Tipo de dados	Nulo	Multivalor	Composto	Derivado
Plano de Exercícios	Número de repetições	Número de vezes que o cliente deve repetir determinado exercício	3 dígitos	INT	Não	Não	Não	Não
	Número de Séries	Número de vezes que o cliente deve realizar determinada repetição	2 dígitos	INT	Não	Não	Não	Não
Subscreve	Data Submissão	Data em que o cliente subscreve determinado serviço	2 dígitos+ 1 caractere (/)+ 2 dígitos+ 1 caractere(/)+ 4 dígitos	DATE	Não	Não	Não	Não
É prestado por	Data Início	Data em que o funcionário começou a prestar determinado exercício	2 dígitos+ 1 caractere (/)+ 2 dígitos+ 1 caractere(/)+ 4 dígitos	DATE	Não	Não	Não	Não

Tabela 3.2: Identificação e caracterização dos atributos dos relacionamentos.

Depois de todos os atributos identificados, já com os seus domínios classificados, procedemos então à seleção da melhor chave primária para cada entidade. Na tabela 3.3 podemos observar todos os atributos de cada entidade os quais consideramos serem candidatos a chave primárias.

Entidade	Chave Primária	Candidato a chave Primária
Cliente	ClienteID	ClienteID Telemóvel 1 Email Nr Contribuinte
Funcionário	FuncionárioID	FuncionárioID Telemóvel 1 Email
Serviço	ServiçoID	ServiçoID Designação
Fatura	IdFatura	IdFatura
Equipamento	IdEquipamento	IdEquipamento
Exercício	ExercícioID	ExercícioID Designação

Tabela 3.3: Identificação de candidatos a chave primária.

Devido a razões de eficiência, decidimos criar um identificador para cada uma das nossas entidades pois, deste modo conseguimos manipular de forma conveniente o domínio destes, a fim de evitar exageros no número de combinações possíveis referente à chave primária de cada entidade. Exemplo disso é, as entidades cliente, funcionário e fatura possuem um atributo relacionado a um número de contribuinte o qual é um candidato válido a ser chave primária, no entanto, não se justifica ter uma chave primária com 9 dígitos quando o número de clientes esperado do ginásio não ultrapassa as centenas.

### 3.5 Detalhe ou generalização de entidades

Para guardar e gerir toda a informação requerida pelo nosso cliente, o senhor Miguel, depois de analisar cuidadosamente os requisitos levantados, selecionamos 6 entidades fundamentais para o bom funcionamento e gestão da nossa base de dados. Estas 6 entidades são caracterizadas como entidades fortes na medida em que, são todas independentes umas das outras e todas elas possuem sentido de existência separadamente.ginásio do senhor Miguel.

1. **Cliente:** A entidade Cliente refere-se aos clientes que estão afiliados ao ginásio do senhor Miguel.
2. **Funcionário:** Esta entidade representa todos os funcionários que realizam qualquer tipo de serviço no ginásio.

3. **Serviço:** A entidade Serviço, como o nome indica, está relacionada com os vários de serviço disponibilizados pelo ginásio os quais são prestados pelos funcionários aos clientes. Exemplo de serviços são natação, nutrição, Personal Trainer.
4. **Exercício:** Esta entidade, como é evidente, corresponde aos de exercício que um cliente do ginásio poderá praticar, com ou sem supervisão por parte de um funcionário.
5. **Equipamento:** A entidade equipamento, refere-se aos equipamentos que o ginásio possui os quais podem ser utilizados pelo cliente para praticar exercícios.
6. **Fatura:** A entidade fatura está relacionada com o pagamento dos serviços prestados no ginásio e como é óbvio, corresponde às faturas que são impressas pelo ginásio aos clientes do mesmo.

### 3.6 Apresentação e explicação do diagrama ER

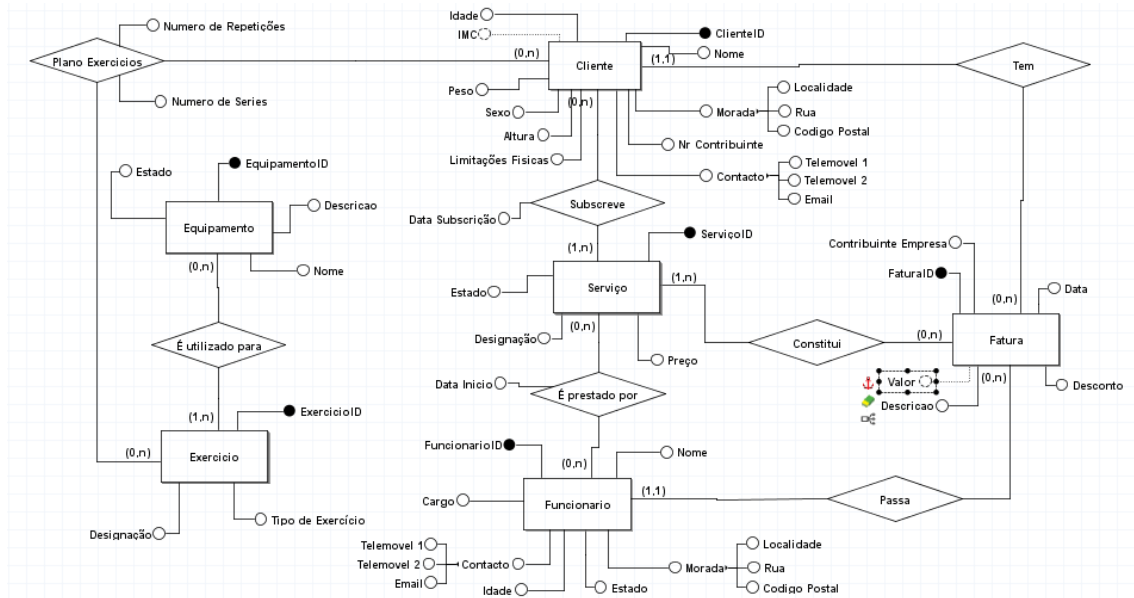


Figura 3.1: Modelo Conceptual.

Como já foi dito anteriormente, com o intuito de modelar todas as entidades da nossa base de dados e os relacionamentos entre elas, criamos um diagrama ER usando notação **Peter Chen** por motivos de organização. Este diagrama é constituído pelas nossas sete entidades as quais estão ligadas entre si através dos relacionamentos já apresentados no grupo 3.3.

Uma das razões de se usar um diagrama deste tipo é que este permite-nos chegar a um entendimento mais preciso em relação ao que o senhor Miguel de facto quer / precisa. De facto, é mais fácil visualizar um diagrama de entidades e relacionamentos complexo, do que ler e entender um longo texto com todos os requerimentos do senhor Miguel. Isto acontece pois, por vezes podemos fazer interpretações erradas de alguns requerimentos ou até, o autor dos mesmos pode não os explicar de forma sucinta.

Assim, o uso de um diagrama ER permite-nos ter uma visualização do problema bastante mais simplificada e organizada proporcionando-nos uma valiosa ferramenta que será a base de todo este projeto.

### 3.7 Validação do modelo de dados com o utilizador

A fim de finalizar todo este processo de modelação, foi necessário que o senhor Miguel o confirmasse e validasse.

No final da criação do modelo ER, juntamo-nos mais uma vez com o nosso cliente para, se necessário, retificar alguma gralha no modelo conceptual. Alteramos e corrigimos vários detalhes e anomalias referentes ao modelo até que o senhor Miguel ficasse satisfeito.

Foi nesta fase que, aquando da aprovação e validação do modelo conceptual como uma verdadeira representação da realidade na qual o modelo incide (o ginásio), por parte do nosso cliente, este o assinou, marcando assim o fim desta etapa fazendo com que pudéssemos avançar para o modelo lógico.



## Capítulo 4

# Modelação Lógica

Nesta capítulo, traduzimos o modelo concetual (fig. 3.1) num modelo de dados lógico adequado aos dados do ginásio. Para tal, seguimos o processo descrito em [CB15].

### 4.1 Derivar relações para o modelo de dados lógicos

Começamos por derivar relações a partir do modelo concetual seguindo as regras adequadas, de modo a representar no modelo lógico as entidades, relacionamentos e atributos identificados previamente. Tal gerou as seguintes relações. No entanto, para as relações *Cliente* e *Funcionário*, decidimos separar os atributos compostos *Contacto* e *Morada* nas suas próprias relações, em vez de colocar os respetivos atributos atómicos nas relações mencionadas, de forma a facilitar a compreensão.

**Cliente** (ClienteID, Nome, Idade, Sexo, Peso, Altura, IMC, Limitações Físicas, NrContribuinte, idContacto, idMorada)

**Primary Key** ClienteID

**Foreign Key** idContacto **references** Contacto(idContacto)

**Foreign Key** idMorada **references** Morada(idMorada)

**Contacto** (idContacto, Telemóvel 1, Telemóvel 2, Email)

**Primary Key** idContacto

**Morada** (idMorada, Rua, Localidade, CodigoPostal)

**Primary Key** idMorada

**Serviço** (ServiçoID, Designação, Preço)

**Primary Key** ServiçoID

**Fatura** (FaturaID, Contribuinte Empresa, Data, Desconto, Descrição, Valor, ClienteID, FuncionárioID)

**Primary Key** FaturaID

**Foreign Key** ClienteID **references** Cliente(ClienteID)

**Foreign Key** FuncionárioID **references** Funcionário(FuncionárioID)

**Funcionário** (FuncionárioID, Nome, Cargo, Idade, idContacto, idMorada)

**Primary Key** FuncionárioID

**Foreign Key** idContacto **references** ContactoFuncionario(idContacto)

**Foreign Key** idMorada **references** MoradaFuncionario(idMorada)

**ContactoFuncionario** (idContacto, Telemóvel 1, Telemóvel 2, Email)

**Primary Key** idContacto

**MoradaFuncionario** (idMorada, Rua, Localidade, CodigoPostal)

**Primary Key** idMorada

**Exercício** (ExercícioID, Designação, Tipo de Exercício)  
**Primary Key** ExercícioID

**Equipamento** (EquipamentoID, Descrição, Nome)  
**Primary Key** EquipamentoID

**Constitui** (ServiçoID, FaturaID)  
**Primary Key** ServiçoID, FaturaID  
**Foreign Key** ServiçoID **references** Serviço(ServiçoID)  
**Foreign Key** FaturaID **references** Fatura(FaturaID)

**Subscreve** (ClienteID, ServiçoID, Data Subscrição)  
**Primary Key** ClienteID, ServiçoID  
**Foreign Key** ClienteID **references** Cliente(ClienteID)  
**Foreign Key** ServiçoID **references** Serviço(ServiçoID)

**E prestado por** (ServiçoID, FuncionárioID, Data Início)  
**Primary Key** ServiçoID, FuncionárioID  
**Foreign Key** ServiçoID **references** Serviço(ServiçoID)  
**Foreign Key** FuncionárioID **references** Funcionário(FuncionárioID)

**Plano Exercícios** (ClienteID, ExercícioID, Numero de Repetições, Numero de Series)  
**Primary Key** ClienteID, ExercícioID  
**Foreign Key** ClienteID **references** Cliente(ClienteID)  
**Foreign Key** ExercícioID **references** Exercício(ExercícioID)

**É utilizado para** (EquipamentoID, ExercícioID)  
**Primary Key** EquipamentoID, ExercícioID  
**Foreign Key** EquipamentoID **references** Equipamento(EquipamentoID)  
**Foreign Key** ExercícioID **references** Exercício(ExercícioID)

## 4.2 Desenho do modelo lógico

A conversão das relações derivadas anteriormente resultou no modelo lógico representado na figura 4.1.

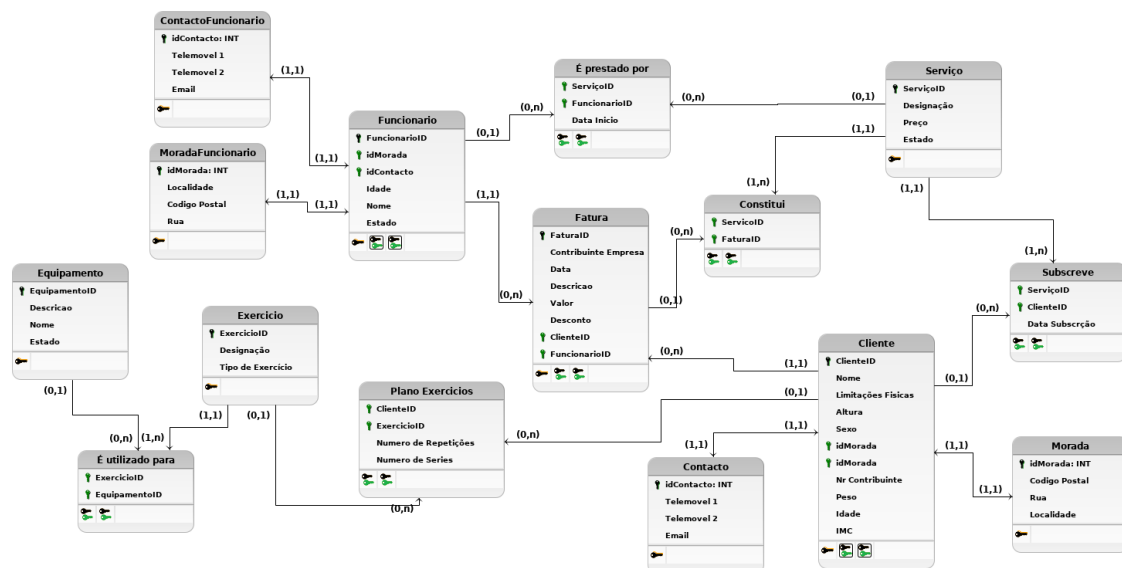


Figura 4.1: Modelo Lógico

## 4.3 Validação do modelo através da normalização

### 4.3.1 Dependências Funcionais

**Cliente:**

ClienteID ® Nome, Idade, Sexo, Peso, Altura, IMC, Limitações Físicas, NrContribuinte, idContacto, idMorada

NrContribuinte ® ClienteID, Nome, Idade, Sexo, Peso, Altura, IMC, Limitações Físicas, idContacto, idMorada

idContacto ® ClienteID, Nome, Idade, Sexo, Peso, Altura, IMC, Limitações Físicas, NrContribuinte, idMorada

**Contacto:**

idContacto ® Telemóvel 1, Telemóvel 2, Email

**Morada:**

idMorada ® Rua, Localidade, CodigoPostal

**Serviço:**

ServiçoID ® Designação, Preço

**Fatura:**

FaturaID ® Contribuinte Empresa, Data, Desconto, Descrição, Valor, ClienteID, FuncionárioID

**Funcionário:**

FuncionárioID ® Nome, Cargo, Idade, idContacto, idMorada

**ContactoFuncionario:**

idContacto ® Telemóvel 1, Telemóvel 2, Email

**MoradaFuncionario:**

idMorada ® Rua, Localidade, CodigoPostal

**Exercício:**

ExercícioID ® Designação, Tipo de Exercício

**Equipamento:**

EquipamentoID ® Descrição, Nome

**Subscreve:**

ClienteID, ServiçoID ® Data Subscrição

**E prestado por:**

ServiçoID, FuncionárioID ® Data Início

**Plano Exercícios:**

ClienteID, ExercícioID ® Numero de Repetições, Numero de Series

### 4.3.2 Primeira Forma Normal (1FN)

Todas as relações estão na Primeira Forma Normal, pois, para cada relação, a interseção de uma linha e uma coluna contém apenas um valor.

### 4.3.3 Segunda Forma Normal (2FN)

As relações estão na Segunda Forma Normal, pois cada relação está na 1FN e todos os atributos não chave primária dessa relação dependem da totalidade da chave primária.

#### 4.3.4 Terceira Forma Normal (3FN)

As relações estão na Terceira Forma Normal, pois cada relação está na 1FN e na 2FN e todos os atributos não chave primária dessa relação e dependem diretamente da chave primária, ou seja, não existem dependências transitivas.

### 4.4 Validação do modelo com interrogações do utilizador

As seguintes interrogações proveem da secção 2.2.2, composta por interrogações e transações. Iremos validar estas interrogações verificando que toda a informação necessária à sua realização (entidades, atributos e relacionamentos) está presente no modelo. As transações serão validadas na secção 4.5.

**Tem de ser possível visualizar que serviços são prestados por que funcionários.**

As entidades Serviço e Funcionário, que compreendem os detalhes dos serviços e dos funcionários, respetivamente, estão representadas no modelo. Podemos usar o relacionamento *Serviço É Prestado Por Funcionário* para extrair a informação requerida.

**O responsável do ginásio poderá também consultar os serviços fornecidos pelo ginásio**

$\pi_{Designacao}(Serviço)$

**Para posterior confirmação dos updates ou adições referentes ao funcionário, o administrador deverá poder visualizar o conteúdo da tabela dos funcionários**

$\pi(Funcionario)$

**Existe a necessidade de aceder à ficha do cliente pelo número de utilizador**

A entidade Cliente representa os dados do cliente, que tem um número de utilizador único (o atributo ClienteID), logo a operação é possível.

**Também deverá ser possível aceder à ficha do cliente através do seu número de telefone, caso este não saiba o seu idCliente**

A entidade Cliente representa os dados do cliente e os contactos dos clientes estão representados na relação Contacto. A entidade Cliente tem como uma *Foreign Key* o identificador do contacto do cliente na tabela Contacto, pelo que é possível associar a um cliente ao seu número de telefone e, assim, aceder aos dados do cliente a partir deste.

**Terá de ser possível ao funcionário consultar os exercícios de um cliente**

A entidade Cliente representa os dados do cliente e a entidade Exercício representa os exercícios no modelo lógico. Utilizando o relacionamento *Plano Exercícios*, é possível consultar a informação requerida.

**Deverá ser possível a consulta de todas as faturas emitidas pelo ginásio relativas a um cliente**

As entidades Cliente e Fatura representam, respetivamente, os clientes (e os seus dados) e as faturas (e os seus dados). Utilizando o relacionamento *Cliente Tem Fatura*, podemos produzir a lista de todas as faturas relativas a um cliente.

**A consulta de faturas entre duas datas específicas será também uma funcionalidade presente na base de dados**

A entidade Fatura, que representa as faturas no modelo lógico, tem um atributo NN *Data*, que guarda a data de emissão da fatura, pelo que a operação é possível.

**O responsável reserva-se o dever de ter acesso ao total faturado num determinado período de tempo ou durante todo o tempo de vida do ginásio**

A entidade Fatura, que representa as faturas no modelo lógico, tem um atributo NN *Data*, que guarda a data de emissão da fatura, e um atributo NN *Valor*, que guarda o valor total da fatura, portanto, a interrogação é válida.

**Quer o administrador quer os rececionistas devem poder ver que serviços foram registados em cada fatura**

As entidades Serviço e Fatura representam, respetivamente, os serviços providenciados pelo ginásio e as faturas emitidas por este no modelo. Utilizando o relacionamento *Serviço Constitui Fatura*, é possível visualizar a informação pretendida.

**Os funcionários devem poder também ver quais as subscrições dos clientes**

As entidades Cliente e Fatura representam, respetivamente, os clientes (e os seus dados) e os serviços do ginásio. Utilizando o relacionamento *Cliente Subscrive Serviço*, é possível saber quais as subscrições dos clientes.

## 4.5 Validação do modelo com as transações estabelecidas

As seguintes transações proveem da secção 2.2.2, composta por interrogações e transações. Iremos validar estas transações verificando que toda a informação necessária à sua realização (entidades, atributos e relacionamentos) está presente no modelo. As interrogações foram validadas na secção 4.4.

**Terá de ser possível ao responsável do ginásio adicionar um funcionário**

A entidade Funcionário, que representa os funcionários e os seus dados, existe no modelo, pelo que é possível adicionar funcionários à BD.

**Terá também de ser possível ao administrador do ginásio alterar a informação relativa a um funcionário**

É possível, por consequência direta da existência da entidade Funcionário.

**Outra das funções do administrador do ginásio será adicionar novos serviços que poderão vir a ser disponibilizados aos clientes**

A entidade Serviços, que representa os serviços oferecidos pelo ginásio, existe, pelo que esta transação é válida.

**Este terá também a obrigação de marcar como não disponíveis os serviços que entretanto forem descontinuados ou os quais o ginásio já não suporte**

A entidade Serviços, que representa os serviços oferecidos pelo ginásio, existe, e contém o atributo Estado, que representa os serviços ativos ou descontinuados. Portanto, a transação é válida.

**Terá de ser possível ao funcionário registar um novo cliente**

A entidade Cliente, que representa os clientes e os seus dados, existe no modelo, pelo que é possível adicionar clientes à BD.

**O funcionário deverá ter permissões para adicionar novos exercícios a um cliente**

As entidades Cliente e Exercício representam, respetivamente, os clientes (e os seus dados) e os exercícios (e os seus dados) e estão ligadas pelo relacionamento binário *Plano Exercícios*, pelo que é possível associar exercícios a um cliente.

**Para além disso terá de ser também disponibilizada forma de apagar os exercícios associados a um cliente do seu plano de treinos**

É possível, como consequência direta da transação anterior ser válida.

**Ao funcionário reserva-se o dever de emitir faturas**

No modelo existe a entidade Fatura, que representa as faturas emitidas pelo ginásio. Portanto, a emissão de faturas é possível.

**Ao funcionário reserva-se também o dever de marcar uma fatura como inválida sempre que esta for mal inserida no sistema**

A entidade Fatura tem o atributo Estado, que guarda se a fatura é válida ou não. Portanto, esta transação é válida.

**Cabe ao funcionário designar qual o equipamento a ser utilizado num determinado exercício no momento da inserção deste na base de dados**

As entidades Exercício e Equipamento, que representam os exercícios e os equipamentos disponibilizados pelo ginásio, existem no modelo. Utilizando o relacionamento *Equipamento É Utilizado Para Exercício*, é possível associar equipamentos a exercícios. A transação é válida

**No ginásio é requerida a necessidade de poder adicionar novos equipamentos**

A transação é válida, pois a entidade Equipamento representa os equipamentos disponíveis no ginásio, pelo que é possível serem adicionados mais.

**Um funcionário pode decidir adicionar um novo exercício aos atualmente disponíveis para os clientes**

A transação é válida, pois a entidade Exercício representa os exercícios disponibilizados pelo ginásio, pelo que é possível serem adicionados mais.

**O administrador será responsável por referenciar que funcionários executam que serviço**

As entidades Funcionário e Serviço, que representam os funcionários do ginásio e os serviços oferecidos, existem no modelo, e, utilizando o relacionamento *Serviço É Prestado Por Funcionário*, é possível associar um serviço a um funcionário, pelo que a transação é válida.

## 4.6 Revisão do modelo lógico com o utilizador

Após a finalização do modelo lógico, este foi apresentado ao cliente numa reunião, onde foi analisado em conjunto com este, para se determinar se o modelo realmente corresponde à realidade do ginásio. O cliente demonstrou-se bastante satisfeito com o resultado e este modelo lógico foi aprovado.

## Capítulo 5

# Implementação Física

### 5.1 Seleção do sistema de gestão de bases de dados

Optamos pela implementação duma base de dados usando o motor de base de dados MySQL. Trata-se dum motor de base de dados muito renomeado e quase padronizado pela industria, devido a sua elevada segurança dos dados, alto desempenho e pela sua capacidade de otimizar o programa reduzindo o tempo de execução.

Por MySQL ser um motor de base de dados gratuito também contribuiu para a sua escolha pois permite que seja possível a criação duma base de dados dentro dum orçamento mais amigável para o cliente.

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

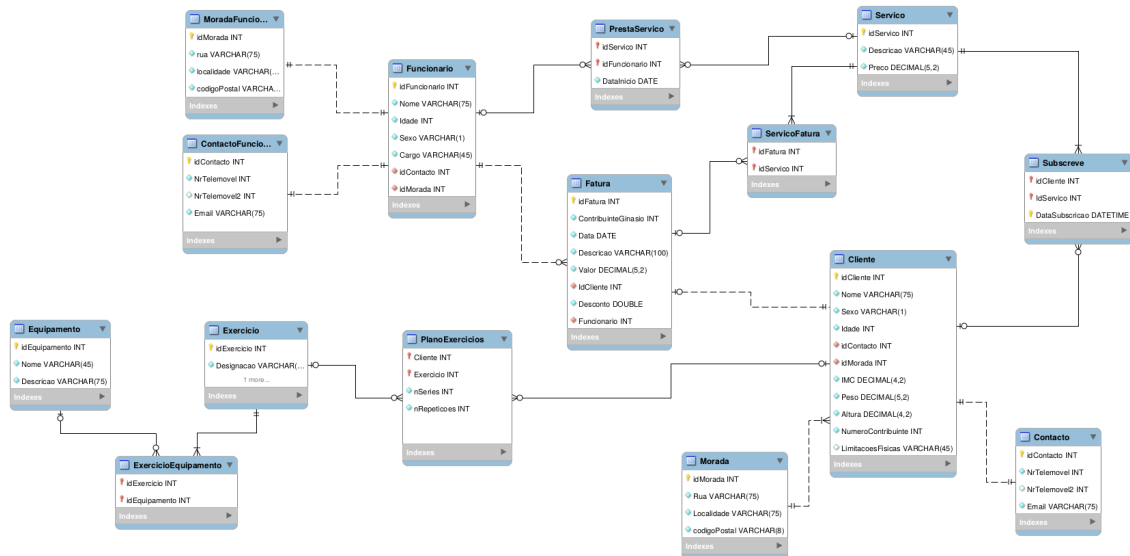


Figura 5.1: Modelo Lógico representado no MySQL Workbench

## 5.3 Tradução das interrogações do utilizador para SQL (alguns exemplos)

### 5.3.1 Interrogação faturação num mês

```
SELECT SUM(Valor) FROM Fatura as F
WHERE MONTH(F.Data)=01 AND YEAR(F.DATA)=2018;
```

	SUM(Valor)
▶	1604.00

### 5.3.2 Todos os clientes que fizeram Martelo

```
SELECT IdCliente, Nome FROM Cliente as C
INNER JOIN planoexercicios as P on P.Cliente=C.idCliente
INNER JOIN exercicio as E on P.Exercicio=E.idExercicio
WHERE E.Designacao='Martelo';
```

	IdCliente	Nome
▶	1	João Paulo Silva Freitas
	2	Joana Amaral Canesi Gomes
	5	José Pedro Campos Silva



### 5.3.3 A que serviços o cliente 1 esta subscrito

---

```
SELECT Descricao,nome FROM Cliente as C
INNER JOIN subscrive as S on C.idCliente=S.idCliente
INNER JOIN servico as Z on S.Idservico=Z.idservico
WHERE (C.idCliente=1);
```

---

	Descricao	nome
▶	Natação, Passe diario	João Paulo Silva Freitas
	Nutrição, ida única	João Paulo Silva Freitas
	Sauna, Passe diario	João Paulo Silva Freitas

### 5.3.4 Faturas correspondentes ao cliente 2

---

```
SELECT idFatura,Valor FROM fatura as F
INNER JOIN Cliente as C on F.idCliente=C.idCliente
WHERE C.idCliente=2;
```

---

	idFatura	Valor
	2	55.00
▶	21	150.00

## 5.4 Tradução das transações estabelecidas para SQL (alguns exemplos)

Quando um cliente é criado, também é criado a morada e o seu contacto, usando as transações os contactos e a morada só são criados se e só se o cliente também for criado.

---

```
CREATE DEFINER='root'@'localhost' PROCEDURE 'Nova Ficha de Cliente'(  
    in nome  
        VARCHAR(75),in idade INT,in sexo VARCHAR(1),in peso INT, in altura INT,in  
        numerocontribuinte INT, in limitacoesfisicas VARCHAR(75),in telemovel1 INT, in  
        telemovel2 INT ,email VARCHAR(75),rua VARCHAR(75),localidade VARCHAR(75),codigopostal  
        VARCHAR(75))  
BEGIN  
    DECLARE x INT;  
    DECLARE y INT;  
    DECLARE z INT;  
    DECLARE EXIT HANDLER FOR SQLEXCEPTION  
        BEGIN  
            ROLLBACK;  
            RESIGNAL;  
        END;  
  
    START TRANSACTION;  
  
    SET z = peso /POWER(altura,2);  
    INSERT INTO morada  
        (Rua,Localidade,codigoPostal)  
        VALUES  
            (rua,localidade,codigopostal);  
    SET x= LAST_INSERT_ID();  
  
    INSERT INTO contacto  
        (NrTelemovel,NrTelemovel2,Email)  
        Values  
            (telemovel1,telemovel2,email);  
    SET y=LAST_INSERT_ID();  
    INSERT INTO cliente  
        (Nome,Sexo,Idade,idContacto,IdMorada,IMC,Peso,Altura,NumeroContribuinte,LimitacoesFisicas)  
        Values  
            (nome,sexo,idade,x,y,z,peso,altura,numerocontribuinte,limitacoesfisicas);  
    COMMIT;  
END
```

---

Tal como nos clientes, nos apenas podemos adicionar o contacto e a morada do funcionario se a adiç o do funcionario tamb m foi um sucesso

---

```
CREATE DEFINER='root'@'localhost' PROCEDURE 'Criacao da Ficha de novo Funcionario'(  
    nome VARCHAR(75),in idade INT,in sexo VARCHAR(1),in cargo VARCHAR(75),in telemovel1  
    INT, in telemovel2 INT ,email VARCHAR(75),rua VARCHAR(75),localidade  
    VARCHAR(75),codigopostal VARCHAR(75))  
BEGIN  
    DECLARE x INT;  
    DECLARE y INT;  
    DECLARE EXIT HANDLER FOR SQLEXCEPTION  
    BEGIN  
        ROLLBACK;  
        RESIGNAL;  
    END;  
    START TRANSACTION;  
    INSERT INTO moradafuncionario  
        (Rua,Localidade,codigoPostal)  
        VALUES  
            (rua,localidade,codigopostal);  
  
    SET x= LAST_INSERT_ID();  
  
    INSERT INTO contactofuncionario  
        (NrTelemovel,NrTelemovel2,Email)  
        Values  
            (telemovel1,telemovel2,email);  
    SET y=LAST_INSERT_ID();  
    INSERT INTO funcionario  
        (Nome,Idade,Sexo,Cargo,idContacto,IdMorada)  
        Values  
            (nome,idade,sexo,cargo,y,x);  
    COMMIT;  
END
```

---

## 5.5 Escolha, defini o e caracteriza o de  ndices em SQL (alguns exemplos)

Implementamos um  ndice no nosso Plano de Exerc cios uma vez que   uma tabela que ao longo do tempo vai ficar extremamente populada, pois para cada cliente vamos preencher v rios exerc cios, como tal para encontrar o plano de exerc cios dum cliente criamos um  ndice na coluna do idCliente para facilitar a sua procura.

Pensamos ainda implementar um  ndice na tabela dos clientes, mas como ainda se trata dum pequeno gin sio, a expectativa de n mero de clientes n o justifica a implementa o dum  ndice.

## 5.6 Estimativa do espaço em disco da base de dados e taxa de crescimento anual

Para a estrutura de base de dados no seu estado atual, e considerando os tamanhos máximos permitidos nas várias estruturas, temos uma ocupação máxima de espaço dividida para cada elemento inserido.

**Exemplo de calculo do espaço ocupado:**

Espaço ocupado pela tabela Cliente	Total em Bytes
Primeiro de tudo, o Cabeçalho, ocupa 4 bytes.	4 bytes
Data Type que tem valor fixo(INT)5x4	20 bytes
Null Block = 2 bytes, pois temos 11 campos	2 bytes
Variable Block = 2 bytes + 12bytes (2 para cada campo com dados variáveis, nos temos 6)	14 bytes
Data Types com valores variáveis (nome varchar(75),sexo varchar(1),IMC decimal(4,2),Peso decimal(5,2), altura(decimal(4,2) e limitações físicas varchar(45)	134 bytes
<b>Total</b>	174 bytes

- Um cliente ocupa 174bytes+96bytes do contacto + 175bytes da morada,ou seja, 445 bytes
- Um funcionário ocupa 154 bytes +96bytes para contacto + 175 bytes para a morada,ou seja,425 bytes
- Uma fatura ocupa 149 bytes
- Um serviço ocupa 68 bytes
- Um exercício ocupa 135 bytes
- Um equipamento ocupa 138 bytes
- Uma relação entre serviço e funcionário ocupa 15 bytes
- Uma relação entre serviço e cliente ocupa 23 bytes
- Uma relação entre serviço e fatura ocupa 15 bytes
- Uma relação entre cliente e exercício ocupa 23 bytes
- Uma relação entre exercício e equipamento ocupa 15 bytes

Tendo em conta que a base de dados no estado atual tem 20 clientes, 8 funcionários,9 serviços, 18 equipamentos, 22 exercícios, 82 exercícios dedicados aos clientes ,6 serviços a ser prestados aos clientes pelos funcionários, existem 35 inscrições dos clientes nos serviços,20 serviços faturados,23 exercícios utilizam equipamento e 20 faturas, temos :

---

20 x 445= 8900 B -> Clientes

8 x 425 = 3400 B -> Funcionário

9 x 68 = 612 B -> Serviço

18 x 138 = 2484 B -> Equipamento

22 x 135 = 2970 B -> Exercícios

82 x 23 =1886 B -> Relacionamento exercício-cliente

6 x 15 =90 B -> Relacionamento serviço-funcionário

35 x 23 =805 B -> Relacionamento serviço-cliente

20 x 15 =300 B -> Relacionamento serviço-fatura

23 x 15 =345 B -> Relacionamento exercício-equipamento

20 x 149 = 2980 B -> Faturas

---

Total: 24 772 b = 24.19141 kB

---

Logo, a base de dados no estado atual tem aproximadamente 24.2kB ocupados.

O Sr. Miguel espera no próximo ano ter mais uma centena de clientes,subscrivendo se no mínimo a 1 serviço, cada um com pelo menos 5 exercícios( no intuito de balancear com os que se inscrevem apenas em nutrição,logo não tem exercícios), e para satisfazer as necessidades dos mesmos ele quer acrescentar 20 novos equipamentos e recrutar 5 novos funcionários, além disso, planeia ter 2000 faturas ao final do ano.

---

100 x 445 = 44500 B-> Clientes  
 5 x 425 = 2125 B-> Funcionários  
 20 x 138 = 2760 B-> Equipamento  
 2000 x 149 = 298000 B-> Fatura  
 100 x 805 = 80500 B-> Relacionamento serviço-cliente  
 5 x 100 x 23 = 11500 B -> Relacionamento cliente-exercício  
 Total = 450885 B = 440.32 kB

---

Com base nestes dados no próximo ano a base de dados ocupará  $24.1 + 440.32 = 463.33$  kB

## 5.7 Definição e caracterização das vistas de utilização em SQL (alguns exemplos)

Código da implementação da view que lista todos os clientes do Ginásio

---

```

CREATE VIEW 'view_cliente' AS
SELECT
  C.idCliente,
  C.Nome,
  C.Sexo,
  C.Idade,
  C.IMC,
  C.Peso,
  C.Altura,
  C.NumeroContribuinte,
  C.LimitacoesFisicas,
  CC.NrTelemovel,
  CC.NrTelemovel2,
  CC.Email,
  M.Rua,
  M.Localidade,
  M.CodigoPostal
FROM
  Cliente AS C
  INNER JOIN
  Contacto AS CC ON CC.idContacto = C.idContacto
  INNER JOIN
  Morada AS M ON M.idMorada = C.idMorada;
  
```

---

idCliente	Nome	Sexo	Idade	IMC	Peso	Altura	NumeroContribuinte	LimitacoesFisicas	NrTelemovel	NrTelemovel2	Email	Rua	Localidade	CodigoPostal
1	João Paulo Silva Freitas	M	20	23.20	80.70	1.86	384968373	XXXX	965499765	XXXX	quantosques@outlook.com	Rua dos peoes	Vila Verde	4730-490
2	Joana Amaral Canesi Gomes	F	21	25.30	75.00	1.75	384757394	XXXX	911457895	937652981	joaninha@hotmail.com	Rua das alminhas	Braga	4700-005
3	Luís Vitor Costa	M	32	32.60	94.00	1.81	485963744	XXXX	916150165	XXXX	carnerinho@hotmail.com	Rua das prezes	Esposende	4740-007
4	Pedro Simões Cardoso	M	21	20.70	79.00	1.87	586734747	XXXX	916150166	XXXX	eraumavez@gmail.com	Travessa principal	Braga	4700-100
5	José Pedro Campos Silva	M	19	27.00	87.20	1.78	485737263	XXXX	912153165	916789541	cocacola@gmail.com	Rua do Farto	Braga	4700-426
6	Ana Salva Gatore Resende	F	18	26.10	69.20	1.53	457463635	nanismo	916220165	XXXX	anabelinha@gmail.com	Rua da Sé	Braga	4700-426
7	Silvia Domingues Santos	F	19	28.20	76.00	1.49	584736463	nanismo	916152265	XXXX	silviame@gmail.com	Alameda da Fonte	Braga	4700-426
8	Salvador Sabrosa Vidal	M	45	26.10	86.40	1.70	567362635	reumatismo	916159865	XXXX	pepsivinscoca@hotmail.com	Rua 1º de Maio	Espinho	4715-604
9	Nunu Uzumaki Tristão	M	23	29.40	84.50	1.78	505795847	asma	933178987	XXXX	mentoscomcola@gmail.com	Rua 25 de Abril	Espinho	4715-602
10	Rafael Penedo Feio	M	20	23.20	86.30	1.97	458677947	esclerose lateral	911567972	965816791	mordidelasnostetos@outlook.com	Rua 8 de Dezembro	Espinho	4715-605
11	Jonas Soares Sousa	M	22	22.10	76.30	1.86	346678645	XXXX	928765915	XXXX	escacarpedra@yahoo.pt	Rua da Igreja	Padim da ...	4700-658
12	Carlos Manuel Cartola	M	22	21.50	73.20	1.76	968372621	XXXX	969167845	XXXX	wafuflord@outlook.pt	Rua da Fontinha d...	Espinho	4700-670
13	Filipe Peso Pesado	M	22	31.50	75.40	1.85	485769485	XXXX	938752875	938456101	mylitlepony@gmail.com	Rua do 1º Cabo Ba...	ADAÚFE	4710-586
14	Hugo Pena pesada	M	21	36.40	83.60	1.89	485768473	XXXX	916125955	XXXX	lolisdragon@outlook.com	Travessa 1º de Maio	Adaúfe	4710-470
15	Jacoto Leite Portugal	M	20	25.10	79.90	1.71	485977373	XXXX	96789555	XXXX	speciallod@gmail.com	Largo da Igreja Velha	Arentim	4705-014
16	Parvalhoti Pampam Pam	M	38	26.90	82.90	1.78	495686743	asma	916199998	921560165	origams@hotmail.com	Rua Monte de Baixo	Arentim	4705-015

Código da implementação da view que mostra todos os planos de exercícios.

---

```
CREATE VIEW 'view_exerciciosCliente' AS

SELECT C.Nome AS NomeCliente, E.Designacao AS NomeExercicio, PE.nSeries, PE.nRepeticoes
FROM PlanoExercicios AS PE
INNER JOIN Exercicio AS E ON E.idExercicio = PE.Exercicio
INNER JOIN Cliente AS C ON C.idCliente = PE.Cliente;
```

---

Código da implementação da view que mostra Os serviço(serviços que o cliente subscreveu) que cada funcionário faz

---

```
CREATE
ALGORITHM = UNDEFINED
DEFINER = 'root'@'localhost'
SQL SECURITY DEFINER
VIEW 'view_prestaservico' AS
SELECT
    'S'.Descricao AS 'Descricao',
    'F'.Nome AS 'Nome',
    'F'.Cargo AS 'Cargo',
    'S'.Preco AS 'Preco'
FROM
    (('prestaservico' 'PS'
    JOIN 'servico' 'S' ON (('S'.idServico' = 'PS'.idServico')))
    JOIN 'funcionario' 'F' ON (('F'.idFuncionario' = 'PS'.idFuncionario')))
ORDER BY 'S'.Descricao'
```

---

Descricao	Nome	Cargo	Preco
Ginasio, Passe mensal com PT	Genghis Khan	Personal Traineer	120.00
Ginasio, Passe mensal com PT	Usain Bolt	Personal Trainner	120.00
Natação, Passe diario	Fidel Castro	Professor de Natação	5.00
Natação, Passe mensal	Fidel Castro	Professor de Natação	30.00
Nutrição, ida única	Joseph Goebbels	Nutricionista	30.00
Nutrição, 2x mês	Joseph Goebbels	Nutricionista	50.00

Código da implementação da view que mostra a que serviços os clientes estão subscritos.

---

```
CREATE VIEW 'view_subscricao' AS

SELECT C.Nome AS NomeCliente, S.Descricao AS Servico, SUB.DataSubscricao FROM Subscreve
AS SUB
INNER JOIN Servico AS S ON S.idServico = SUB.idServico
INNER JOIN Cliente AS C ON C.idCliente = SUB.idCliente;
```

---

	NomeCliente	Servico	DataSubscricao
▶	Joana Amaral Canesi Gomes	Natação,Passe diario	2018-05-21 00:00:00
	Nunu Uzumaki Tristão	Natação,Passe diario	2018-10-20 00:00:00
	Filipe Peso Pesado	Natação,Passe diario	2018-08-10 00:00:00
	João Paulo Silva Freitas	Natação,Passe mensal	2018-04-20 00:00:00
	Pedro Simões Cardoso	Natação,Passe mensal	2018-06-01 00:00:00
	Nunu Uzumaki Tristão	Natação,Passe mensal	2018-11-01 00:00:00
	Jonas Soares Sousa	Natação,Passe mensal	2018-02-08 00:00:00
	Rafael Penedo Feio	Ginasio,Passe diario sem acompanhamento	2018-05-06 00:00:00
	Rafael Penedo Feio	Ginasio,Passe diario sem acompanhamento	2018-05-07 00:00:00
	Rafael Penedo Feio	Ginasio,Passe diario sem acompanhamento	2018-05-08 00:00:00
	Rafael Penedo Feio	Ginasio,Passe diario sem acompanhamento	2018-05-09 00:00:00
	Carlos Manuel Cartola	Ginasio,Passe diario sem acompanhamento	2018-05-10 00:00:00
	Parvalhoti Pampam Pam	Ginasio,Passe diario sem acompanhamento	2018-09-07 00:00:00
	Luís Vitor Costa	Ginasio,Passe mensal sem acompanhamento	2018-05-21 00:00:00
	Silvia Domingues Santos	Ginasio,Passe mensal sem acompanhamento	2018-10-01 00:00:00
	Salvador Sabrosa Vidal	Ginasio,Passe mensal sem acompanhamento	2018-11-01 00:00:00
	Rafael Penedo Feio	Ginasio,Passe mensal sem acompanhamento	2018-05-10 00:00:00
	Jonas Soares Sousa	Ginasio,Passe mensal sem acompanhamento	2018-02-07 00:00:00
	Jacito Leite Portugal	Ginasio,Passe mensal sem acompanhamento	2018-03-07 00:00:00
	Sousa Silva Sozinho	Ginasio,Passe mensal sem acompanhamento	2018-05-07 00:00:00
	Alvaro Campos	Ginasio,Passe mensal sem acompanhamento	2018-10-07 00:00:00
	José Pedro Campos Silva	Ginasio, Passe mensal com PT	2018-11-02 00:00:00
	Filipe Peso Pesado	Ginasio, Passe mensal com PT	2018-08-10 00:00:00
	Manuel Machado Costa	Ginasio, Passe mensal com PT	2018-05-01 00:00:00
	Manuel Machado Costa	Ginasio, Passe mensal com PT	2018-06-07 00:00:00
	Jacito Leite Portugal	Nutrição, 2x mês	2018-03-07 00:00:00
	João Paulo Silva Freitas	Nutrição, ida única	2018-04-20 00:00:00
	Pedro Simões Cardoso	Nutrição, ida única	2018-06-23 00:00:00
	Ana Salva Gatore Resende	Nutrição, ida única	2018-10-11 00:00:00
	Hugo Pena pesada	Nutrição, ida única	2018-05-07 00:00:00
	Luís Vitor Costa	Sauna,Passe mensal	2018-05-21 00:00:00
	Salvador Sabrosa Vidal	Sauna,Passe mensal	2018-11-20 00:00:00
	Salvio Salvio Salvio	Sauna,Passe mensal	2018-09-10 00:00:00
	João Paulo Silva Freitas	Sauna, Passe diario	2018-04-20 00:00:00

## 5.8 Definição e caracterização dos mecanismos de segurança em SQL (alguns exemplos)

Foram criados 2 users, um para cada rececionista, com apenas permissões de adicionar e dar update as linhas das tabelas, tenho como segurança que nenhum destes destrói uma tabela sem querer.

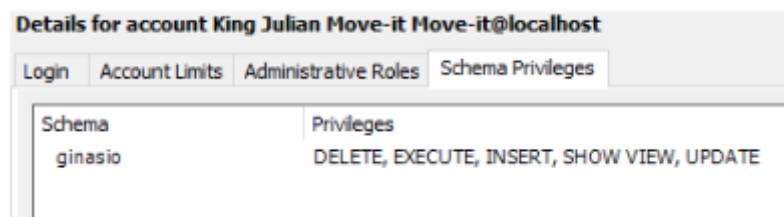
---

```
CREATE USER 'Jafar Strogonof'@'localhost' IDENTIFIED BY 'Jafar';

CREATE USER 'King Julian Move-it Move-it'@'localhost' IDENTIFIED BY 'Julian';

GRANT INSERT,UPDATE,EXECUTE, SHOW VIEW ON Ginasio.* TO 'Jafar Strogonof'@'localhost';
GRANT INSERT,UPDATE,EXECUTE, SHOW VIEW ON Ginasio.* TO 'King Julian Move-it
Move-it'@'localhost';
```

---



The screenshot shows the 'Details for account King Julian Move-it Move-it@localhost' window in MySQL. The 'Schema Privileges' tab is selected. It displays a table with two columns: 'Schema' and 'Privileges'. The 'Schema' column lists 'ginasio', and the 'Privileges' column lists 'DELETE, EXECUTE, INSERT, SHOW VIEW, UPDATE'.

Schema	Privileges
ginasio	DELETE, EXECUTE, INSERT, SHOW VIEW, UPDATE

Sendo assim, o Sr. Miguel a única pessoa com acesso administrativo a base de dados.

## 5.9 Revisão do sistema implementado com o utilizador

Após a implementação do sistema de base de dados, foi realizada uma reunião com o Sr. Miguel para ver se o produto final lhe agradava e receber algum feedback caso não estivesse do seu agrado.

Primeiramente o Sr. Miguel disse que não estávamos a cumprir completamente o seu pedido, ou seja, não tínhamos todos os procedimentos que este desejava, como por exemplo, adicionar um funcionário e um equipamento. Alargou-nos o prazo de entrega para a semana seguinte com o objetivo de adicionarmos essas funções e para dar mais uns retoques onde fosse necessário.

Posteriormente, após a instalação e demonstração da execução destes novos procedimentos, o Sr. Miguel ficou bastante satisfeito com o sistema de Base de Dados.



## Capítulo 6

# Conclusões e Trabalho Futuro

Neste trabalho construímos uma base de dados para o ginásio do Sr. Miguel, cumprindo todos os requisitos que o mesmo nos apresentou. De referir, que este trabalho se tornou mais extenso e trabalhoso do que o expectado, todavia o resultado final compensou a nossa dedicação uma vez que conseguimos alcançar a satisfação do nosso cliente e das metas que nos foram colocadas. Gostaríamos de realçar ainda que este trabalho contribuiu para que fossemos capazes de desenvolver uma melhor capacidade de trabalho em equipa e de delegação de tarefas.

Durante o desenvolvimento deste projeto, percebemos que o auxílio do motor de base de dados MySQL foi de grande importância visto que nos proporcionou ferramentas fundamentais para facilitar várias das nossas tarefas e garantir toda a consistência dos nossos dados.

O nosso maior desafio, mas também uma conquista foi conseguir chegar a um consenso no nosso modelo conceptual. Numa primeira etapa do desenvolvimento do projeto, o modelo parecia-nos correto, contudo tal não se verificou o que nos levou a investir uma grande quantidade de tempo a remodelar o mesmo para que tudo estivesse de acordo com o que o nosso cliente desejava e de acordo com as regras de normalização e das etapas da modelação lógica.

Para além disso, é de salientar que todo o processo de modelação, desde modelo conceptual até ao físico, é imprescindível na criação de uma base de dados para promover a organização e consistência de todo o processo. De facto, este método contribuiu imenso para o nosso sucesso.

Numa perspetiva futura, temos como objetivo implementar uma entidade Fornecedor que estaria encarregue da distribuição dos equipamentos e no caso de um equipamento avariar seria possível chegar ao fornecedor do mesmo e comunicar a necessidade da reparação do equipamento. Além disso queremos expandir a base de dados para mais ginásios, ou seja, criar uma entidade ginásio, permitindo nos comercializar facilmente a base de dados.

# Lista de Siglas e Acrónimos

BD	Base de dados
IMC	Índice de Massa Corporal
1FN	Primeira Forma Normal
2FN	Segunda Forma Normal
3FN	Terceira Forma Normal
NN	Não Nulo

# Bibliografia

- [CB15] Thomas M. Connolly and Carolyn E. Begg. *Database Systems - A Practical Approach to Design, Implementation, and Management*. Pearson Education Limited, 6 edition, 2015.