



**Universidade do Minho**  
Escola de Engenharia  
Licenciatura em Engenharia Informática

## **Unidade Curricular de Bases de dados**

Ano Letivo de 2023/2024

### **I.B.M. Watson**

Diogo Neto (A98197)   Diogo Gonçalves (A101919)   Guilherme Oliveira  
(A95021)   Tiago Guedes (A97369)

20 de julho de 2025

# **B D**

Data de Receção	
Responsável	
Avaliação	
Observações	

# I.B.M. Watson

Diogo Neto (A98197)   Diogo Gonçalves (A101919)   Guilherme Oliveira (A95021)   Tiago Guedes (A97369)

20 de julho de 2025

# Resumo

Este projeto prático foi desenvolvido no contexto da disciplina de Bases de Dados, com o objetivo principal de analisar, planejar, modelar, arquitetar e implementar um Sistema de Base de Dados para a empresa fictícia de investigação "I.B.M. Watson". O trabalho envolveu uma abordagem multifacetada para o levantamento e análise de requisitos, identificação e caracterização das entidades e dos relacionamentos, e a criação de um modelo de dados que cumpre as três formas normais. Este relatório detalha todas as etapas da criação da base de dados e apresenta o modelo lógico resultante, que foi criado a partir de um levantamento sucinto de dados relativos ao tema em questão.

**Área de Aplicação:** Desenho, arquitetura e desenvolvimento de Sistemas de Bases de Dados para uma empresa relacionada com investigações realizadas por detetives.

**Palavras-Chave:** Bases de Dados Relacionais, MySQL, Detetives, Investigação, Gestão de casos, Empresa

# Índice

<b>1</b>	<b>Definição do Sistema</b>	<b>1</b>
1.1	Contextualização . . . . .	1
1.1.1	Apresentação do Caso de Estudo . . . . .	2
1.2	Motivação e Objectivos . . . . .	2
1.3	Análise da Viabilidade do processo . . . . .	3
1.4	Recursos e Trabalho da equipa . . . . .	4
1.5	Plano de Execução do Projeto . . . . .	4
<b>2</b>	<b>Levantamento e Análise de Requisitos</b>	<b>7</b>
2.1	Método de Levantamento e de Análise de Requisitos Adotado . . . . .	7
2.2	Organização dos Requisitos Levantados . . . . .	9
2.2.1	Requisitos de descrição . . . . .	9
2.2.2	Requisitos de manipulação . . . . .	11
2.2.3	Requisitos de controlo . . . . .	13
2.3	Análise e validação dos requisitos . . . . .	14
<b>3</b>	<b>Modelação Conceptual</b>	<b>15</b>
3.1	Apresentação da abordagem de modelação realizada . . . . .	15
3.2	Identificação e caracterização das entidades . . . . .	15
3.3	Identificação e caracterização dos relacionamentos . . . . .	16
3.4	Identificação e Caracterização dos Atributos das Entidades e dos Relacionamentos	20
3.5	Apresentação e explicação do diagrama ER produzido . . . . .	24
<b>4</b>	<b>Modelação Lógica</b>	<b>26</b>
4.1	Construção e Validação do Modelo de Dados Lógico . . . . .	26
4.2	Apresentação e Explicação do Modelo Lógico Produzido . . . . .	30
4.3	Normalização de Dados . . . . .	31
4.4	Validação do Modelo com Interrogações do Utilizador . . . . .	31
<b>5</b>	<b>Implementação Física</b>	<b>34</b>
5.1	Apresentação e explicação da base de dados implementada . . . . .	34
5.2	Criação de utilizadores da base de dados . . . . .	38
5.3	Povoamento da base de dados . . . . .	39
5.4	Plano de segurança e recuperação de dados . . . . .	39
5.5	Cálculo do espaço da base de dados . . . . .	40
5.6	Definição e caracterização das vistas de utilização em sql . . . . .	44
5.7	Tradução das interrogações do utilizador para SQL . . . . .	44

5.8	Indexação do sistema de dados . . . . .	46
5.9	Implementação de procedimentos, funções e gatilhos . . . . .	46
<b>6</b>	<b>Conclusões e Trabalho Futuro</b>	<b>47</b>
	<b>Lista de Siglas e Acrónimos</b>	<b>49</b>
<b>7</b>	<b>Anexos</b>	<b>50</b>
7.1	Script de Povoamento: . . . . .	50
7.2	Script de Backup e Restauo . . . . .	52

# Lista de Figuras

1.1	Tabela - expectativa de execução . . . . .	5
1.2	Diagrama de Gantt - expectativa de execução . . . . .	5
1.3	Tabela - execução . . . . .	6
1.4	Diagrama de Gantt - execução . . . . .	6
3.1	Relacionamento cliente-caso . . . . .	17
3.2	Relacionamento caso-caso . . . . .	17
3.3	Relacionamento caso-evidencia . . . . .	18
3.4	Relacionamento Equipa-Caso . . . . .	18
3.5	Relacionamento Detetive-Equipa . . . . .	19
3.6	Relacionamento Detetive-Evidência . . . . .	19
3.7	Relacionamento Detetive-Equipamento . . . . .	20
3.8	Modelo Conceptual . . . . .	25

# Lista de Tabelas

2.1	Requisitos de descrição - contribuições . . . . .	11
2.2	Requisitos de Manipulação - contribuições . . . . .	13
2.3	Requisitos de controlo - contribuições . . . . .	14
5.1	Tamanho de cada tipo de dados . . . . .	40
5.2	Tamanho dos atributos da tabela Equipamento . . . . .	40
5.3	Tamanho dos atributos da tabela Equipa . . . . .	40
5.4	Tamanho dos atributos da tabela Detetive . . . . .	41
5.5	Tamanho dos atributos da tabela Cliente . . . . .	41
5.6	Tamanho dos atributos da tabela Caso . . . . .	42
5.7	Tamanho dos atributos da tabela Fotografia . . . . .	42
5.8	Tamanho dos atributos da tabela Documento . . . . .	42
5.9	Tamanho dos atributos da tabela Evidencia . . . . .	42
5.10	Tamanho dos atributos da tabela Utilizado . . . . .	43

# 1 Definição do Sistema

## 1.1 Contextualização

Após vários anos a trabalhar para um detetive particularmente prestigiado, o Dr. Watson ambicionava ser responsável pelos seus casos de investigação, pelo que, a 7 de Agosto de 2012 o detetive Watson iniciou a atividade do Instituto de Buscas Metódico Watson, mais conhecido como I.B.M. Watson, uma empresa privada que presta serviços de investigação particulares.

Inicialmente o I.B.M. Watson era constituído apenas pelo Dr. Watson, este focava-se na resolução de casos como investigação de pessoas e bens, procura de pessoas desaparecidas e investigação de casos familiares ou empresariais.

No decorrer dos anos o detetive Watson mostrou a sua perícia na resolução dos casos dos seus clientes, o que levou a um aumento da popularidade dos seus serviços e a um aumento no fluxo de casos a trabalhar. De forma a atender ao maior número de casos possíveis, atender a novas áreas de investigação e manter a qualidade do serviço prestado, o detetive Watson contratou onze novos colaboradores, contudo, informação sobre as investigações em curso, recolhida por membros de cada equipa, começaram a perder-se pelo escritório, atrasando assim a resolução dos casos.

Sempre que havia reabertura de um caso, a procura dos documentos pelo arquivo da empresa era bastante demorada, sendo que, por vezes, os documentos acabavam por não serem encontrados. Após várias reclamações por parte dos seus funcionários e alguns atrasos nas investigações, o que levou a reclamações por parte dos clientes, o Dr. Watson chegou à conclusão que teria de alterar o método de armazenamento e consulta dos casos em que a sua empresa trabalhava, daí ter recorrido à atual equipa de desenvolvimento deste projeto.

Numa reunião inicial o Dr. Watson expôs os problemas que a sua empresa enfrentava a um dos membros da equipa, este, por sua vez, sugeriu que a implementação de uma base de dados seria o suficiente para evitar perdas de informação e melhorar a organização entre equipas de investigação. No fim da reunião, o Dr. Watson concordou com a implementação deste sistema se o mesmo se mostrasse viável fosse ao encontro do processo de digitalização da empresa.



### **1.1.1 Apresentação do Caso de Estudo**

A I.B.M. Watson é uma empresa de investigação fictícia que foi idealizada para elaborar um caso de estudo, no domínio de "Agência de detetives", proposto pelos docentes da unidade curricular de bases de dados no ano letivo de 2023/2024 na Universidade do Minho.

Este caso de estudo foi concebido para aplicar os conceitos teóricos aprendidos em sala de aula a um ambiente empresarial simulado visando o desenvolvimento de um sistema de gestão de bases de dados relacional.

Qualquer cenário apresentado neste relatório é meramente figurativo, ou seja, não representa qualquer empresa ou entidade à data de elaboração do projeto.

## **1.2 Motivação e Objectivos**

O desenvolvimento deste trabalho foi motivado pela necessidade de resolver os desafios enfrentados pela I.B.M. Watson no que diz respeito à gestão ineficiente de informação e documentos dos casos de investigação e ,também, no que diz respeito à organização dos seus funcionários nas respetivas equipas, assim como equipamentos da empresa usados pelos mesmos.

### **Ata da reunião de objetivos a alcançar**

**27 de fevereiro de 2024**

**Participantes:** - Autores do projeto - Dr. Watson - Chefes de equipas - Contabilidade

Nesta primeira reunião os respetivos chefes de equipa começaram por destacar a dificuldade de dedução dos membros das suas equipas, uma vez que, por vezes, alguém recolhe uma evidência porém a mesma não é encontrada nos registos dos casos. Com a frequência destes acontecimentos, as investigações de alguns casos acabam por se atrasarem. A contabilidade referiu que, por vezes, é difícil saber qual detetive pertence a determinada equipa e se o mesmo participou na investigação de um caso, o que dificulta no cálculo do preço da investigação. De modo a facilitar o controlo de equipamentos disponíveis, o Dr. Watson pediu que a plataforma acomodasse uma funcionalidade de gestão de equipamentos dos funcionários. No final da reunião ficou acordado que, embora a contabilidade tivesse opinião na concepção do sistema, esta apenas o utilizaria para verificar o preço dos casos, pelo que, o sistema é focado na gestão de equipas, casos, evidências e equipamentos.

### **Fim da reunião.**

Como descrito anteriormente, a empresa enfrenta dificuldades significativas na organização de casos no seu arquivo para consulta futura, assim como na partilha de novas evidências em casos abertos entre membros da equipa que a compõem.

Também foi relatado pelo Dr. Watson que existe alguma confusão no que diz respeito à divisão de detetives em equipes, bem como na distribuição do seu vasto inventário de equipamentos como carros, câmaras e telemóveis pelos respectivos funcionários.

Diante desses desafios, o principal objetivo deste projeto é eliminar perdas de informação de casos e melhorar certas logísticas internas.

A implementação de um sistema de gestão de bases de dados tem como objetivo reunir e automatizar a gestão de informação e documentos, facilitando o acesso rápido e preciso a dados relevantes para cada caso de investigação, resolvendo assim os problemas identificados anteriormente e melhorando a eficiência operacional da empresa.

Na sequência da reunião onde foram discutidos os objetivos a alcançar, seguem-se os mesmos:

- Arquivo: O primeiro objetivo do projeto é criar um sistema que permite reunir toda a informação relacionada aos casos de investigação da empresa num único local acessível a todos os membros da equipa, isto inclui detalhes dos casos, evidências recolhidas, relatórios de investigação e informações dos clientes.
- Melhorar logística de equipas: Diante do problema proposto, é necessário que a plataforma permita agrupar toda a informação de cada detetive, da equipa a que este pertence, dos casos em que cada uma trabalha e dos equipamentos usados por detetives na recolha de evidências.
- Eficiência Operacional: Por fim, o objetivo principal é aumentar a eficiência operacional da empresa, reduzindo os atrasos nas investigações, melhorando a comunicação e a colaboração entre as equipas e garantindo um serviço de alta qualidade e pontualidade para com os clientes.

Estes objetivos foram definidos com base nos desafios identificados e nas necessidades específicas da empresa, e orientaram todo o processo de desenvolvimento deste sistema de gestão de bases de dados.

## **1.3 Análise da Viabilidade do processo**

A análise da viabilidade do processo de desenvolvimento deste projeto parte de uma motivação intrínseca à resolução dos desafios enfrentados pela I.B.M. Watson no âmbito da gestão de informação e documentos dos casos de investigação, bem como na organização dos seus recursos humanos e materiais, apresentados anteriormente. Após a reunião inicial de definição de objetivos, onde foram identificadas dificuldades como a falta de organização de evidências, tornou-se clara a necessidade de um sistema capaz de centralizar e organizar todas a informação dos casos em investigação. O Dr. Watson salientou ainda a importância de uma gestão eficiente dos equipamentos utilizados pelos detetives, reforçando a urgência na implementação de uma solução que aborde integralmente essas questões.

Diante desses desafios, o principal objetivo do projeto é desenvolver um sistema de gestão de

bases de dados que possa não só resolver os problemas identificados, mas também otimizar a eficiência operacional da empresa. Este sistema visa reunir e automatizar a gestão de informação e documentos relacionados aos casos de investigação, facilitando o acesso e a partilha entre os membros das equipas. Além disso, a implementação deste sistema procura melhorar a comunicação e colaboração entre os membros das equipas, garantindo a resolução de casos no menor tempo possível e por consequência um serviço de alta qualidade e pontualidade para com os clientes. Estes objetivos, idealizados a partir das necessidades específicas da empresa, servem como guia para todo o processo de desenvolvimento, visando alcançar uma solução eficaz e definitiva para os desafios enfrentados pela I.B.M. Watson.

## **1.4 Recursos e Trabalho da equipa**

A equipa encarregada pelo desenvolvimento do sistema de gestão de bases de dados para a "I.B.M. Watson" é composta por 4 especialistas em Bases de Dados com a colaboração do Dr. Watson para aprovação do projeto.

Durante o desenvolvimento do projeto, de modo a perceber as necessidades da empresa, a equipa reuniu-se com funcionários da empresa como detetives, chefes de equipa e até membros da contabilidade. Nestas reuniões os integrantes disponibilizaram recursos como fichas de recolha de evidências, relatórios de casos e até fichas de clientes, de modo a identificar necessidades da plataforma.

## **1.5 Plano de Execução do Projeto**

O plano de execução para a implementação do sistema de gestão de bases de dados na I.B.M. Watson do Dr. Watson foi elaborado considerando o tempo médio para realização de cada uma das suas etapas.

Para alcançar os objetivos definidos anteriormente, será desenvolvida uma única Base de Dados que abrangerá toda a informação relevante da agência de detetives, esta base de dados será projetada para lidar com o volume de dados esperado.

De modo a cumprir os prazos impostos pela I.B.M. Watson e manter uma boa organização por parte da equipa de desenvolvimento, a mesma elaborou o seguinte diagrama de Gantt.

Prazos definidos pela IBM Watson

Fases	Tópicos	Tarefas	Data de início	Data de conclusão	Duração
Fase 1	1 - Definição do sistema	1.1 - Contextualização	14/fev	17/fev	3
		1.2 - Motivação e Objetivos	18/fev	20/fev	2
		1.3 - Análise da viabilidade do projeto	19/fev	23/fev	4
		1.4 - Recursos e Trabalho da equipa	22/fev	25/fev	3
		1.5 - Plano de execução	26/fev	28/fev	2
	2 - Levantamento e Análise de requisitos	2.1 - Método de Levantamento e de Análise de Requisitos Adotado	29/fev	03/mar	3
		2.2 - Organização dos requisitos levantados	03/mar	10/mar	7
		2.2.1 - Requisitos de descrição	03/mar	10/mar	7
		2.2.2 - Requisitos de exploração	03/mar	10/mar	7
		2.2.3 - Requisitos de administração	03/mar	10/mar	7
		2.3 - Análise e validação de requisitos	10/mar	13/mar	3
		2.3 - Análise e validação de requisitos	10/mar	13/mar	3
	3 - Modelação Conceptual	3.1 - Apresentação da abordagem de modelação realizada	13/mar	14/mar	1
		3.2 - Identificação e caracterização das entidades	15/mar	16/mar	1
		3.3 - Identificação e caracterização dos relacionamentos	15/mar	16/mar	1
		3.4 - Identificação e caracterização da associação dos atributos com as entidades e relacionamentos	15/mar	16/mar	1
		3.5 - Apresentação e explicação do diagrama ER produzido	15/mar	21/mar	6
	4 - Modelação lógico	4.1 - Construção e validação do modelo de dados lógico	21/mar	24/mar	3
		4.2 - Apresentação e Explicação do Modelo Lógico Produzido	21/mar	24/mar	3
		4.3 - Normalização de Dados	21/mar	24/mar	3
		4.4 - Validação do Modelo com Interrogações do Utilizador	24/mar	31/mar	7
		4.4 - Validação do Modelo com Interrogações do Utilizador	24/mar	31/mar	7
Fase 2	5 - Implementação Física	5.1 - Apresentação e explicação da base de dados implementada	21/abr	23/abr	2
		5.2 - Criação de utilizadores da base de dados	24/abr	25/abr	1
		5.3 - Povoamento da base de dados em SQL	26/abr	30/abr	4
		5.4 - Cálculo do espaço da base de dados (inicial e taxa de crescimento anual)	01/mai	03/mai	2
		5.5 - Definição e caracterização de vistas de utilização em SQL	06/mai	13/mai	7
		5.6 - Tradução das interrogações do utilizador para SQL	13/mai	17/mai	4
		5.7 - Indexação do Sistema de Dados	17/mai	22/mai	5
		5.8 - Implementação de procedimentos, funções e gatilhos	24/mai	26/mai	2
		5.8 - Implementação de procedimentos, funções e gatilhos	24/mai	26/mai	2
	6 - Conclusões e Trabalho Futuro		24/mai	26/mai	2

Figura 1.1: Tabela - expectativa de execução

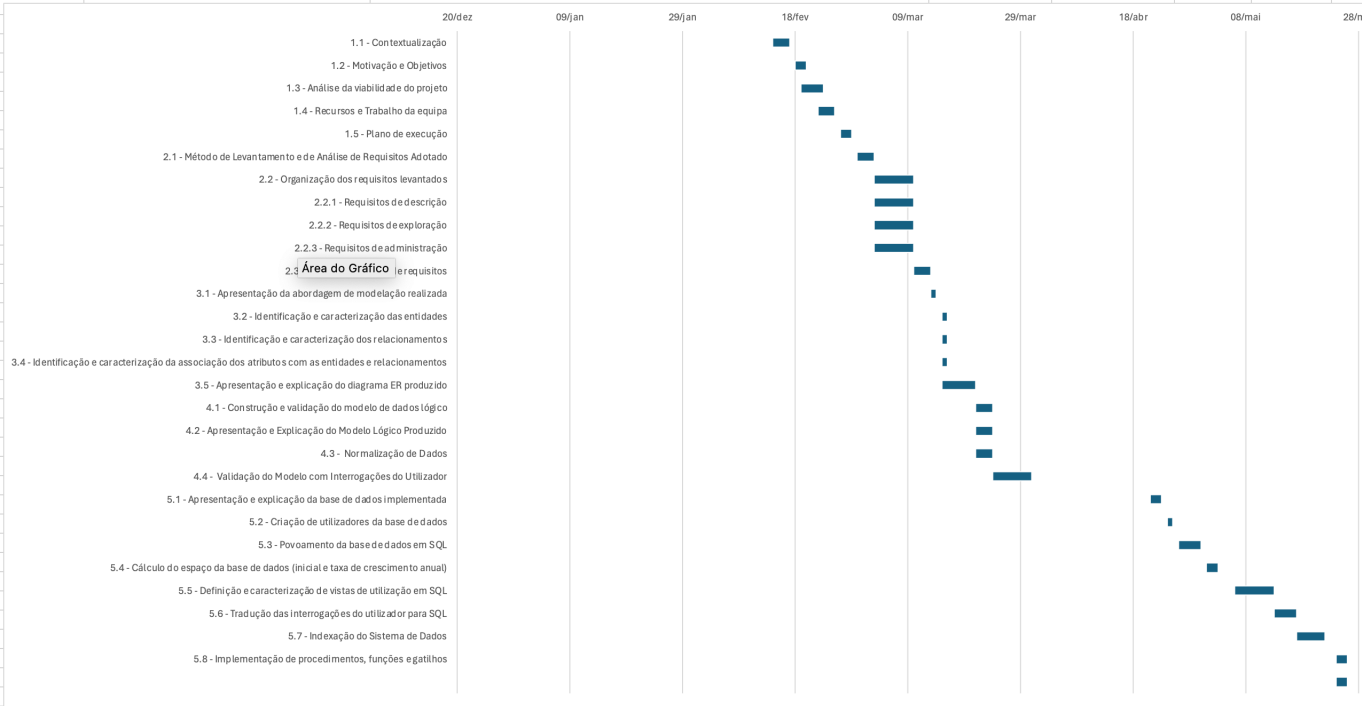


Figura 1.2: Diagrama de Gantt - expectativa de execução

Prazos cumpridos pela Equipa

Fases	Tópicos	Tarefas	Data de início	Data de conclusão	Duração
Fase 1	1 - Definição do sistema	1.1 - Contextualização	19/fev	21/fev	2
		1.2 - Motivação e Objetivos	21/fev	23/fev	2
		1.3 - Análise da viabilidade do projeto	24/fev	26/fev	2
		1.4 - Recursos e Trabalho da equipa	26/fev	27/fev	1
		1.5 - Plano de execução	27/fev	03/mar	5
	2 - Levantamento e Análise de requisitos	2.1 - Método de Levantamento e de Análise de Requisitos Adotado	03/mar	08/mar	5
		2.2 - Organização dos requisitos levantados	07/mar	08/mar	1
		2.2.1 - Requisitos de descrição	04/mar	12/mar	8
		2.2.2 - Requisitos de exploração	04/mar	19/mar	15
		2.2.3 - Requisitos de administração	04/mar	19/mar	15
		2.3 - Análise e validação de requisitos	11/mar	22/mar	11
	3 - Modelação Conceptual	3.1 - Apresentação da abordagem de modelação realizada	22/mar	23/mar	1
		3.2 - Identificação e caracterização das entidades	22/mar	27/mar	5
		3.3 - Identificação e caracterização dos relacionamentos	22/mar	27/mar	5
		3.4 - Identificação e caracterização da associação dos atributos com as entidades e relacionamentos	22/mar	27/mar	5
		3.5 - Apresentação e explicação do diagrama ER produzido	27/mar	28/mar	1
	4 - Modelação lógico	4.1 - Construção e validação do modelo de dados lógico	02/abr	07/abr	5
		4.2 - Apresentação e Explicação do Modelo Lógico Produzido	02/abr	07/abr	5
		4.3 - Normalização de Dados	02/abr	07/abr	5
		4.4 - Validação do Modelo com interações do Utilizador	02/abr	07/abr	5
Fase 2	5 - Implementação Física	5.1 - Apresentação e explicação da base de dados implementada	24/abr	27/abr	3
		5.2 - Criação de utilizadores da base de dados	22/mai	26/mai	4
		5.3 - Povoamento da base de dados em SQL	22/mai	26/mai	4
		5.4 - Cálculo do espaço da base de dados (inicial e taxa de crescimento anual)	22/mai	26/mai	4
		5.5 - Definição e caracterização de vistas de utilização em SQL	22/mai	26/mai	4
		5.6 - Tradução das interrogações do utilizador para SQL	24/mai	26/mai	2
		5.7 - Indexação do Sistema de Dados	24/mai	26/mai	2
		5.8 - Implementação de procedimentos, funções e gatilhos	24/mai	26/mai	2
	6 - Conclusões e Trabalho Futuro		24/mai	26/mai	2

Figura 1.3: Tabela - execução

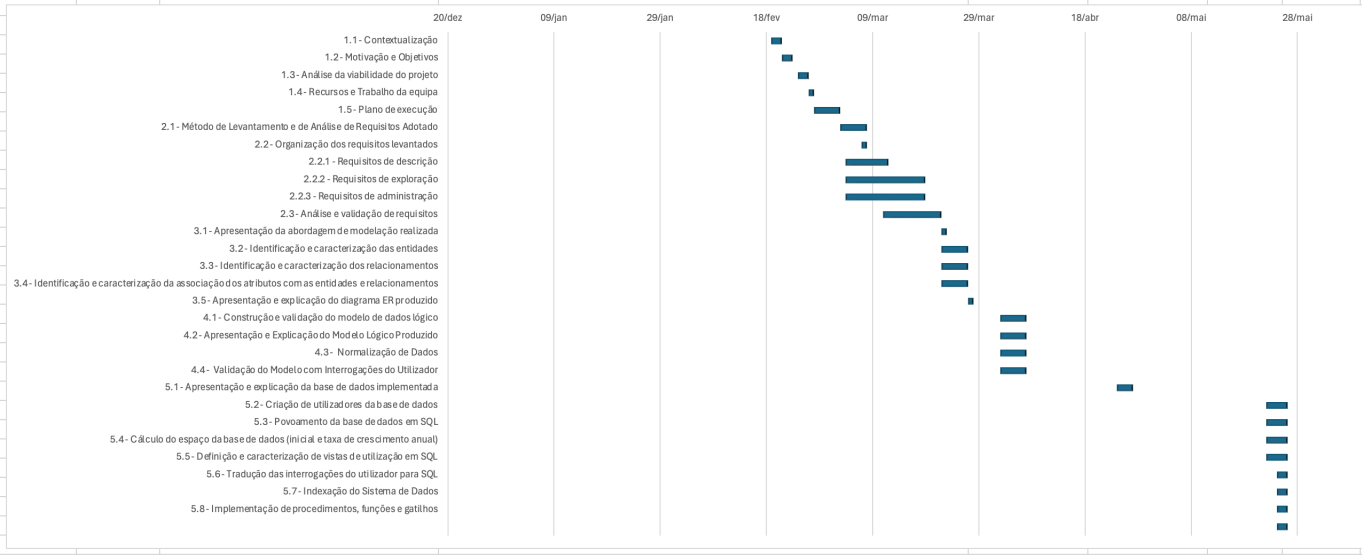


Figura 1.4: Diagrama de Gantt - execução

Como observamos nas tabelas e respectivos diagramas, o prazo inicial estabelecido foi um guia crucial para a equipa de desenvolvimento se orientar e, embora algumas tarefas tenham demorado um pouco mais que o previsto, o prazo de entrega foi cumprido com sucesso.

## 2 Levantamento e Análise de Requisitos

### 2.1 Método de Levantamento e de Análise de Requisitos Adotado

De modo a concluir o levantamento dos requisitos necessários para o desenvolvimento do sistema, optamos por uma abordagem multifacetada que integra várias técnicas e métodos. Inicialmente realizamos uma série de reuniões com os membros que irão utilizar o sistema com o intuito de identificar as necessidades e expectativas em relação ao sistema, recolhendo assim os requisitos de descrição do sistema.

Além disso, organizamos reuniões de brainstorming com a equipa de desenvolvimento e os utilizadores finais para explorar ideias, identificar requisitos específicos e validar conceitos.

Ao longo do processo, também recorremos à análise de documentos existentes, como relatórios, procedimentos internos e documentos de projeto, para entender melhor o contexto operacional da empresa e identificar requisitos adicionais.

A documentação resultante deste processo inclui atas de reuniões e notas de entrevistas que serão incorporadas no processo de desenvolvimento do sistema.

Este método abrangente de levantamento e análise de requisitos permitiu-nos obter uma compreensão das necessidades e expectativas dos utilizadores e das características do sistema, garantindo assim que o mesmo cumpra as exigências da I.B.M. Watson.

#### **Ata-resumo de reunião de levantamento de requisitos**

**6 de março de 2024**

**Participantes:** - Autores do projeto - Dr. Watson - Chefes de equipas - Contabilidade - Recursos humanos

Nesta reunião, os participantes foram discutindo os requisitos necessários para abordar os problemas identificados anteriormente, especialmente em relação à dificuldade de organizar casos e evidências. Os chefes de equipa destacaram a importância de melhorar a documentação e o acompanhamento das investigações para evitar atrasos nos casos. A contabilidade informou

os responsáveis do projeto que seria necessário guardar os dados dos clientes que recorrem aos serviços na empresa para efeitos contratuais. O responsável pelos recursos humanos da empresa mencionou a dificuldade em saber a equipa de cada detetive, assim como os equipamentos que o mesmo utilizava. Cada um dos temas mencionados foi extensamente discutido de modo a obter os requisitos necessários para o funcionamento correto do sistema.

**Fim da reunião.**

## 2.2 Organização dos Requisitos Levantados

### 2.2.1 Requisitos de descrição

Definido o método de recolha e após a reunião da equipa de desenvolvimento com alguns especialistas da I.B.M. Watson, a mesma recolheu os seguintes requisitos de descrição:

RD1 - O detetive tem um número único de funcionário.

RD2 - Um detetive tem um nome, uma especialidade que exerce, NIF e um salário.

RD3 - Um detetive tem um só contacto, sendo ele um email e telemóvel.

RD4 - Um detetive só integra a sua equipa de trabalho.

RD5 - Um detetive é responsável pela recolha das evidências relativas ao caso que a sua equipa investiga.

RD6 - Um detetive pode utilizar vários equipamentos na sua investigação ou não utilizar nenhum.

RD7 - Cada equipamento tem um número de controlo assim que adquirido pela empresa.

RD8 - Um equipamento tem uma marca e um modelo.

RD9 - Para além do identificador do equipamento, dependendo do seu tipo, a plataforma deve permitir armazenar matrícula, IMEI ou SN - opcional.

RD10 - A plataforma deve guardar o estado (avariado, reparado, novo) do equipamento.

RD11 - Assim que um cliente recorre aos serviços da IBM Watson, é-lhe atribuído um número único.

RD12 - O cliente tem na sua ficha um nome e um contacto associados, telemóvel e e-mail.

RD13 - Para efeitos contratuais, a plataforma deve permitir guardar morada e NIF, se o cliente assim o desejar.

RD14 - No caso de o cliente ser uma empresa, deve ser possível guardar o website da mesma, assim como o nome da mesma.

RD15 - A empresa é dividida em equipas, cada uma com um número único de equipa, uma especialidade e um responsável.

RD16 - Uma equipa é composta por vários detetives.

RD17 - Múltiplas equipas podem investigar vários casos.



RD18 - Um caso tem um preço.

RD19 - Um caso apenas pertence a uma equipa.

RD20 - Um caso tem um número único de processo, uma descrição da investigação, estado, prioridade e o detetive responsável pelo mesmo.

RD21 - Um caso tem uma data de abertura e de encerramento, contudo podem existir várias reaberturas de casos.

RD22 - Um caso tem zero ou várias evidências que só a este pertence

RD23 - Todas as evidências têm um número de identificação único.

RD24 - Uma evidência pode ter documentos, fotografias e local de recolha anexados.

RD25 - Um cliente pode entregar vários casos à empresa.

RD26 - Os equipamentos são partilhados pelos detetives.

RD27 - Um caso apenas tem um proprietário

RD28 - Uma evidência é recolhida numa data, contém uma descrição e deve ser possível identificar o seu tipo.

Da reunião de 6 de março de 2024 e posteriores validações surgiu esta tabela de colaborações na recolha de requisitos de descrição:

Nº	Data/Hora	Área	Fonte	Analista
1	5/3/24 17:04	Entidades	Dr. Watson	DN
2	6/3/24 17:06	Entidades	Dr. Watson	DN
3	6/3/24 17:10	Entidades	Dr. Watson	DN
4	6/3/24 17:12	Entidades	Gestor de equipa	DN
5	6/3/24 17:14	Entidades	Gestor de equipa	DN
6	5/3/24 17:16	Entidades	Dr. Watson	DN
7	6/3/24 17:18	Entidades	Dr. Watson	DN
8	6/3/24 17:21	Entidades	Dr. Watson	DN
9	6/3/24 17:30	Entidades	Gestor de equipa	DN
10	6/3/24 17:31	Entidades	Gestor de equipa	DN
11	6/3/24 17:32	Entidades	Gestor do projeto	DN
12	6/3/24 17:34	Entidades	Contabilidade	DN
13	6/3/24 17:35	Entidades	Contabilidade	DN
14	6/3/24 17:36	Entidades	Contabilidade	DN
15	6/3/24 17:38	Entidades	Gestor de equipa	DN
16	6/3/24 17:40	Entidades	Gestor de equipa	DN
17	6/3/24 17:41	Entidades	Gestor de equipa	DN
18	6/3/24 17:43	Entidades	Gestor de equipa	DN
19	6/3/24 17:43	Entidades	Gestor de equipa	DN
20	6/3/24 17:44	Entidades	Gestor de equipa	DN
21	6/3/24 17:44	Entidades	Gestor de equipa	DN
22	6/3/24 17:46	Entidades	Gestor de equipa	DN
23	6/3/24 17:50	Entidades	Gestor de equipa	DN
24	6/3/24 17:55	Entidades	Gestor de equipa	DN
25	11/3/24 14:08	Entidades	Contabilidade	DN
26	18/3/24 14:33	Entidades	Dr. Watson	DN
27	18/3/24 14:35	Entidades	Contabilidade	DN
28	18/3/24 14:38	Entidades	Dr. Watson	DN

Tabela 2.1: Requisitos de descrição - contribuições

## 2.2.2 Requisitos de manipulação

RM1 - Listar e gerir detetives.

RM2 - Listar e gerir equipas.

RM3 - Calcular a quantidade de casos por equipa.

RM4 - Calcular a quantidade de evidências por caso.

RM5 - Calcular o número de equipamentos disponíveis.

RM6 - Visualizar casos de um determinado cliente.

- RM7 - Listar casos ativos por prioridade e data de abertura.
- RM8 - Listar equipamentos utilizados por determinado detetive.
- RM9 - Listar evidências recolhidas por determinado detetive num caso em particular.
- RM10 - Listar casos ativos por equipa.
- RM11 - Listar equipas disponíveis para um novo caso.
- RM12 - Listar evidências de um determinado caso.
- RM13 - Calcular o preço total de um caso com base no número de detetives que participaram e tempo aberto.
- RM14 - Listar casos ativos.
- RM15 - Listar detetives que trabalharam num caso específico.
- RM16 - Visualizar caso, incluindo todas as suas reaberturas.
- RM17 - Listar evidências de um determinado caso, incluindo as suas reaberturas.
- RM18 - Calcular tempo médio de resolução de casos por equipa.
- RM19 - Gerar taxa de resolução de casos por equipa.
- RM20 - Visualizar locais de recolha de evidências de um determinado caso.
- RM21 - Gerar relatório de desempenho de equipas, incluindo número de casos atribuídos, resolvidos e pendentes.
- RM22 - Listar detetives por especialidade.

Nº	Data/Hora	Área	Fonte	Analista
1	6/3/24 17:57	Administração	Dr. Watson	GO
2	6/3/24 18:01	Administração	Dr. Watson	GO
3	6/3/24 18:05	Administração	chefe-equipa	GO
4	6/3/24 18:09	Utilizadores	chefe-equipa	GO
5	6/3/24 18:13	Utilizadores	chefe-equipa	GO
6	6/3/24 18:17	Administração	chefe-equipa	GO
7	6/3/24 18:21	Administração/utilizadores	chefe-equipa	GO
8	6/3/24 18:25	Utilizadores	Utilizadores	GO
9	6/3/24 18:29	Administração/utilizadores	chefe-equipa	GO
10	6/3/24 18:33	Administração	Dr. Watson	GO
11	6/3/24 18:37	Administração	Dr. Watson	GO
12	6/3/24 18:41	Utilizadores	Chefe-equipa	GO
13	6/3/24 18:45	Administração	contabilidade	GO
14	6/3/24 18:49	Administração	chefe-equipa	GO
15	6/3/24 18:53	Administração	Dr. Watson	GO
16	6/3/24 18:57	Utilizadores	chefe-equipa	GO
17	6/3/24 19:01	Utilizadores	chefe-equipa	GO
18	6/3/24 19:05	Administração	Dr. Watson	GO
19	6/3/24 19:09	Administração	Dr. Watson	GO
20	6/3/24 19:13	Utilizadores	chefe-equipa	GO
21	6/3/24 19:17	Utilizadores	Dr. Watson	GO
22	6/3/24 19:20	Administração	chefe-equipa	GO

Tabela 2.2: Requisitos de Manipulação - contribuições

### 2.2.3 Requisitos de controlo

RC1: Os detetives apenas podem visualizar os casos em que a sua equipa trabalha.

RC2: Um detetive responsável de equipa deve poder visualizar todos os casos de todas as equipas.

RC3: Um detetive pode visualizar todas as evidências recolhidas num determinado caso da sua equipa.

RC4: Todos os funcionários devem poder visualizar os equipamentos usados e ocupados.

RC5: Apenas chefes de equipa podem visualizar dados dos clientes.

Nº	Data/Hora	Área	Fonte	Analista
1	6/3/24 19:22	Administração	Dr. Watson	DG
2	6/3/24 19:25	Administração	Dr. Watson	DG
3	6/3/24 19:26	Administração	chefe-equipa	DG
4	6/3/24 19:28	Administração	chefe-equipa	DG
5	6/3/24 19:30	Administração	Dr. Watson	DG

Tabela 2.3: Requisitos de controlo - contribuições

## 2.3 Análise e validação dos requisitos

O processo de análise dos requisitos começou com a divisão em três fases: requisitos de descrição, requisitos de exploração e requisitos de administração. Cada fase foi analisada numa reunião de grupo com a empresa, onde todos os membros do grupo contribuíram com suas perspetivas e ideias.

Na fase de requisitos de descrição, o objetivo foi identificar os principais tópicos necessários para o funcionamento do processo de investigação e empresa, e estabelecer quais deles são mais importantes.

Na fase de requisitos de exploração, foram anotadas as informações mais importantes que precisavam ser extraídas do sistema.

E, por fim, na fase de requisitos de administração, estabeleceram-se as restrições do sistema, para evitar erros e garantir a sua estabilidade.

Após a análise dos requisitos, a equipa de desenvolvimento reuniu-se com os membros responsáveis pela empresa para validar os mesmos.

### **Ata-resumo de reunião de validação de requisitos**

**6 de março de 2024**

**Participantes:** - Autores do projeto - Dr. Watson - Chefes de equipas

- Revisão detalhada dos requisitos. - Identificação de lacunas e ambiguidades na entidade caso.
- Resolução de problema identificado.

### **Fim da reunião.**

Após o debate de ideias, chegou-se a um consenso sobre os requisitos do trabalho, permitindo-nos desenvolver e implementar o sistema de gestão de base de dados de forma eficiente.

## 3 Modelação Conceptual

### 3.1 Apresentação da abordagem de modelação realizada

O diagrama entidade-relacionamento, ou ER, é uma ferramenta comum utilizada para modelação de uma base de dados, este permite representar graficamente as entidades, relacionamentos e atributos envolvidos num sistema de gestão de dados.

Após a recolha de requisitos, a equipa de trabalho transcreveu as entidades necessárias, assim como os seus atributos e respetivos relacionamentos, garantindo assim que o sistema seja fidedigno às necessidades, requisitos e objetivos propostos pela I.B.M. Watson.

A equipa de trabalho recorreu à ferramenta brmodelo para desenvolver o diagrama em questão, este facilitou a compreensão do sistema e foi essencial para identificar possíveis problemas ou erros antes da implementação física.

### 3.2 Identificação e caracterização das entidades

Com base nos requisitos recolhidos nas reuniões com o cliente, a equipa de desenvolvimento do projeto considera que as entidades essenciais para o sistema são "Cliente", "Caso", "Evidência", "Detetive", "Equipa" e "Equipamento"

De modo a cumprir as exigências do cliente, a equipa utilizou os requisitos recolhidos e chegou à seguinte caracterização das entidades:

#### - **Cliente**

O cliente representa o proprietário de um caso de investigação entregue à empresa, por sua vez, esta entidade tem características como identificador único, nome e contacto, um atributo composto. A mesma também inclui atributos de preenchimento não obrigatório como número de identificação fiscal, empresa e morada, sendo estes dois últimos atributos compostos.

A noção de 'cliente' deriva dos requisitos de descrição com os números 11, 12, 13 e 14.

#### - **Caso**

Um caso de investigação é a entidade central do sistema, este representa uma ficha de identificação relacionada à resolução de um problema. Com base nos requisitos levantados, ficou definido que esta entidade teria características como número único do processo, descrição do caso, estado do processo, detetive responsável, preço, data de abertura e data de encerramento. Esta entidade deriva dos requisitos de descrição com os números 18§, 20 e 21

#### **- Evidência**

Sempre associadas a um caso, as evidências são fundamentais para a resolução do mesmo. Dos requisitos com números 23, 24 e 28, concluí-se que uma evidência tem características como identificados único, descrição, tipo de evidência, data de recolha e ainda, opcionalmente, fotografias anexadas, documentos e local de recolha.

#### **- Equipa**

O sistema interpreta equipa como um conjunto de detetives responsáveis pela investigação de vários casos, esta deriva dos requisitos de descrição com números 14 e 15 e é caracterizada por atributos como número de equipa, especialização e responsável de equipa.

#### **- Detetive**

Os membros da equipa, ou detetives, são responsáveis pela recolha de evidências na investigação de casos. Esta entidade deriva dos requisitos de descrição com os números 1, 2 e 3, estes requisitos descrevem-na com características como número de detetive, nome, especialidade, contacto, número de identificação fiscal e salário.

#### **- Equipamento**

Na investigação de casos, mais propriamente na recolha de evidências, os detetives utilizam equipamentos da empresa. Como descrito anteriormente, é necessário manter um registo de equipamentos da empresa que, segundo os requisitos de descrição com números 7, 8, 9 e 10, têm características como identificados único do equipamento, marca, modelo, estado e identificador único específico para o tipo de equipamento.

### **3.3 Identificação e caracterização dos relacionamentos**

Após a definição das entidades do sistema, é crucial explorar as interações entre estas, identificando os relacionamentos que as conectam. Nesta fase, serão analisados os diversos tipos de relacionamentos presentes no contexto do sistema considerando as suas características e cardinalidades. Este processo é fundamental para assegurar que o modelo de dados resultante seja completo e consistente, reunindo de forma precisa todas as relações relevantes de acordo com os requisitos recolhidos.

### - Relacionamento cliente-caso

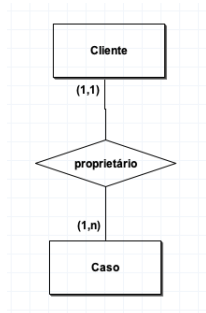


Figura 3.1: Relacionamento cliente-caso

**Relacionamento:** Cliente é proprietário de um caso

**Descrição:** Uma vez que o cliente recorre aos serviços de investigação da I.B.M. Watson, este possui um caso, estas duas entidades relacionam-se e é importante existir o armazenamento das informações do caso, para poderem serem utilizadas e atualizadas ao longo do processo de investigação.

**Multiplicidade:** Cliente(1,1) - Caso(1,N)

Um cliente é proprietário de vários casos e um caso só pertence a um cliente.

**Atributos:** No relacionamento Cliente - Caso não existem atributos associados.

**Requisitos:** Este relacionamento deriva dos requisitos de descrição com números 25 e 27

### - Relacionamento: Caso - Caso

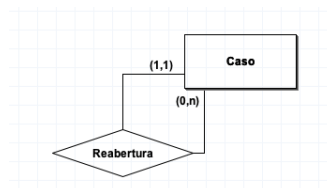


Figura 3.2: Relacionamento caso-caso

**Descrição:** Uma vez que podem ocorrer reaberturas de casos no sistema, um caso reaberto é apenas um caso novo com menção ao anterior.

**Multiplicidade:** Caso(0,1) - Caso(0,N)

Um caso ser reaberto nenhuma ou várias vezes e um caso reaberto deriva, obrigatoriamente, de outro caso

**Atributos:** No relacionamento Caso - Caso não existem atributos associados.



**Requisitos:** Este relacionamento deriva do requisito de descrição com número 21.

- **Relacionamento: Caso - Evidência**

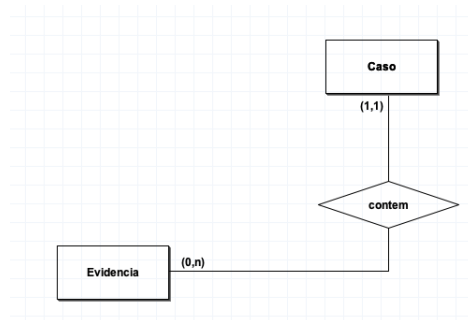


Figura 3.3: Relacionamento caso-evidencia

**Relacionamento: Caso contém evidências**

**Descrição:** No decorrer de uma investigação são recolhidas várias evidências com o objetivo de revolver o mesmos.

**Multiplicidade:** Caso (1,1) - Evidência (0,N)

Um caso pode ter várias evidências recolhidas e estas, por sua vez, só pertencem a um caso.

**Atributos:** No relacionamento Caso-Evidência não existem atributos associados.

**Requisitos:** Este relacionamento deriva do requisito de descrição com o número 22.

- **Relacionamento: Equipe - Caso**



Figura 3.4: Relacionamento Equipa-Caso

**Relacionamento:** Uma equipa investiga casos **Descrição:** Uma equipa investiga vários casos e um caso só pode ser investigado por uma equipa.

**Multiplicidade:** Equipa (1,1) - Caso (0,N)

**Atributos:** No relacionamento Equipa - Caso não existem atributos associados.

**Requisitos:** Este relacionamento deriva do requisitos de descrição com o número 19

- **Relacionamento: Detetive - Equipa**

**Relacionamento:** Detetive integra uma equipa



Figura 3.5: Relacionamento Detetive-Equipa

**Descrição:** A I.B.M. Watson é constituída por equipas no que diz respeito à investigação de casos. Cada detetive pertence a uma equipa e esta é constituída por vários detetives.

**Multiplicidade:** Detetive (1,N) - Equipa (1,1)

**Atributos:** No relacionamento Detetive - Equipa não existem atributos associados.

**Requisitos:** Este relacionamento deriva dos requisitos de descrição com números 4 e 16.

#### - Relacionamento: Detetive - Evidência

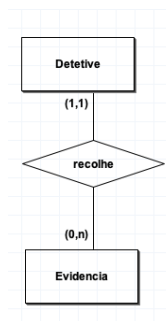


Figura 3.6: Relacionamento Detetive-Evidência

**Relacionamento:** Detetive recolhe evidências

**Descrição:** Como descrito anteriormente no relacionamento Evidência-Caso, um caso tem várias evidências que levam à sua resolução e um detetive é reponsável pela recolha destas várias evidências pertencentes ao caso, enquanto que, uma evidência só é recolhida por um detetive e não pela equipa.

**Multiplicidade:** Detetive (1,1) - Evidência(0,N)

**Atributos:** No relacionamento Detetive - Evidência não existem atributos associados.

**Requisitos:** Este relacionamento deriva do requisito de descrição com número 5.

#### - Relacionamento: Detetive - Equipamento

**Relacionamento:** Equipamento é utilizado por detetive.

**Descrição:**

Durante uma investigação, os detetives podem utilizar os equipamentos fornecidos pela em-



Figura 3.7: Relacionamento Detetive-Equipamento

presa. Um detetive pode utilizar vários equipamentos, estes são partilhados entre detetives, ou seja, o mesmo equipamento pode ser usado por vários detetives.

**Multiplicidade:** Equipamento(0,N)-Detetive(0,N)

**Atributos:** No relacionamento Equipamento - Detetive não existem atributos associados.

**Requisitos:** Este relacionamento deriva dos requisitos de descrição com números 6 e 26

### 3.4 Identificação e Caracterização dos Atributos das Entidades e dos Relacionamentos

Estabelecidas as entidades necessárias ao sistema e os respetivos relacionamentos, é necessário caracterizar estas entidades com atributos de acordo com os requisitos recolhidos na reunião com o cliente. De acordo com requisitos, os relacionamentos não possuem atributos.

## Cliente

### Cliente

- **Número de cliente:** Atributo utilizado para identificar unicamente cada cliente no sistema.
- **Nome:** Nome do cliente.
- **Contacto:** Atributo composto
  - **Telemóvel:** Número de telemóvel do cliente.
  - **E-mail:** E-mail do cliente.
- **Número de Identificação Fiscal - Opcional:** Atributo opcional que guarda o NIF do cliente.
- **Empresa - Atributo composto - Opcional**
  - **Nome da Empresa:** Nome da empresa à qual o cliente está associado.

– **Website:** Website da empresa.

▪ **Morada - Atributo Composto - Opcional**

– **País:** País de residência do cliente.

– **Cidade:** Cidade de residência do cliente.

– **Código Postal:** Código postal da morada do cliente.

– **Localidade:** Localidade onde o cliente reside.

– **Rua:** Nome da rua onde o cliente reside.

– **Porta:** Número da porta onde reside o cliente.

Atributo	Tipo de Dados	Nulo	Composto	Multi-valor	Candidato
Número de Cliente	INT	Não	Não	Não	Sim
Nome	VARCHAR(45)	Não	Não	Não	Não
Telemóvel	INT	Não	Sim	Não	Não
E-mail	VARCHAR(45)	Sim	Não	Não	Não
NIF	VARCHAR(45)	Sim	Não	Não	Não
Nome da Empresa	VARCHAR(45)	Sim	Sim	Não	Não
Website da Empresa	VARCHAR(45)	Sim	Sim	Não	Não

## Caso

### Caso

- **Número do Processo:** Identifica cada caso de forma única no sistema.
- **Descrição:** Permite uma descrição detalhada do caso.
- **Estado do Caso:** Indica o estado atual de cada caso como "Aberto", "Fechado", "Resolvido" etc.
- **Prioridade do Caso:** Nível de importância atribuído ao caso.
- **Detetive Responsável:** Identifica o detetive responsável pelo caso.
- **Data de Abertura:** Data em que o caso foi aberto.
- **Data de Encerramento:** Data em que o caso foi encerrado.
- **Preço:** Valor monetário associado ao caso.
- **Reabertura:** Identificador do caso reaberto a que o novo caso se refere

Atributo	Tipo de Dados	Nulo	Composto	Multi-valor	Candidato
Número do Processo	INT	Não	Não	Não	Sim
Descrição	VARCHAR(45)	Não	Não	Não	Não
Estado do Caso	VARCHAR(45)	Não	Não	Não	Não
Prioridade do Caso	VARCHAR(45)	Não	Não	Não	Não
Detetive Responsável	INT	Não	Não	Não	Não
Data de Abertura	DATETIME	Não	Não	Não	Não
Data de Encerramento	DATETIME	Sim	Não	Não	Não
Preço	DECIMAL	Não	Não	Não	Não

## Equipa

### Equipa

- **Número da Equipa:** Identifica cada equipa de detetives de forma única no sistema.
- **Especialização:** Indica a área de especialização da equipa.
- **Responsável:** Indica o detetive responsável pela equipa.

Atributo	Tipo de Dados	Nulo	Composto	Multi-valor	Candidato
Número da Equipa	INT	Não	Não	Não	Sim
Especialização	VARCHAR(45)	Não	Não	Não	Não
Responsável	INT	Não	Não	Não	Não

## Detetive

### Detetive

- **Número de Detetive:** Identifica cada detetive de forma única no sistema.
- **Nome:** Nome do detetive.
- **Especialidade:** Indica a área de especialização do detetive.
- **Contacto - Atributo Composto:**
  - **Telemóvel:** Número de telemóvel do detetive.
  - **E-mail:** E-mail do detetive.
- **Salário:** Remuneração mensal do detetive.

- **Número de Identificação Fiscal:** NIF do detetive.

Atributo	Tipo de Dados	Nulo	Composto	Multi-valor	Candidato
Número de Detetive	INT	Não	Não	Não	Sim
Nome	VARCHAR(45)	Não	Não	Não	Não
Especialidade	VARCHAR(45)	Não	Não	Não	Não
Telemóvel	INT	Não	Sim	Não	Não
E-mail	VARCHAR(45)	Não	Sim	Não	Não
NIF	VARCHAR(45)	Não	Não	Não	Não
Salário	DECIMAL	Não	Não	Não	Não

## Evidência

### Evidência

- **Identificador:** Identifica cada evidência de forma única no sistema.
- **Descrição:** Permite uma descrição detalhada da evidência.
- **Tipo:** Indica o tipo de evidência.
- **Data de Recolha:** Data em que a evidência foi recolhida.
- **Fotografia - Opcional - Multi-Valor:** Fotografias da evidência.
- **Local de Recolha - Atributo Composto - Opcional:**
  - **Latitude:** Coordenada de latitude do local de recolha.
  - **Longitude:** Coordenada de longitude do local de recolha.
- **Documento - Opcional - Multi-Valor:** Documentos relacionados à evidência.

Atributo	Tipo de Dados	Nulo	Composto	Multi-valor	Candidato
Identificador	INT	Não	Não	Não	Sim
Descrição	VARCHAR(45)	Não	Não	Não	Não
Tipo	VARCHAR(45)	Não	Não	Não	Não
Data de Recolha	DATETIME	Não	Não	Não	Não
Fotografia	BLOB	Sim	Não	Sim	Não
Latitude	DECIMAL	Sim	Sim	Não	Não
Longitude	DECIMAL	Sim	Sim	Não	Não
Documento	BLOB	Sim	Não	Sim	Não

# Equipamento

## Equipamento

- **Identificador:** Identifica cada equipamento de forma única no sistema.
- **Marca:** Marca do equipamento.
- **Modelo:** Modelo do equipamento.
- **Estado:** Estado atual do equipamento.
- **Identificador Específico - Opcional - Atributo Composto:**
  - **Matrícula:** Identificador único de um carro.
  - **IMEI:** Identificador único de um telemóvel.
  - **Número de Série:** Número de série do equipamento.

Atributo	Tipo de Dados	Nulo	Composto	Multi-valor	Candidato
Identificador	INT	Não	Não	Não	Sim
Marca	VARCHAR(45)	Não	Não	Não	Não
Modelo	VARCHAR(45)	Não	Não	Não	Não
Estado	VARCHAR(45)	Não	Não	Não	Não
Matrícula	VARCHAR(45)	Sim	Sim	Não	Não
IMEI	VARCHAR(45)	Sim	Sim	Não	Não
Número de Série	VARCHAR(45)	Sim	Sim	Não	Não

## 3.5 Apresentação e explicação do diagrama ER produzido

Após os capítulos anteriores, obtivemos o seguinte modelo conceptual do sistema:

Desta maneira, ficamos com uma melhor perspetiva do modelo a implementar na base de dados para a I.B.M. Watson . O sistema é composto por 6 entidades com os seus respetivos atributos e relacionamentos, explorados com mais detalhe no ponto anterior. Este modelo foi construído com a ferramenta de trabalho "brmodelo". Após a tradução de requisitos de descrição nas entidades, atributos e respetivos relacionamentos, a equipa de trabalho começou a desenhar o modelo tendo sempre em conta possíveis ambiguidades, de seguida seguiu-se a validação deste modelo e concepção do modelo lógico, estamos então em condições de avançar para a sua implementação física.

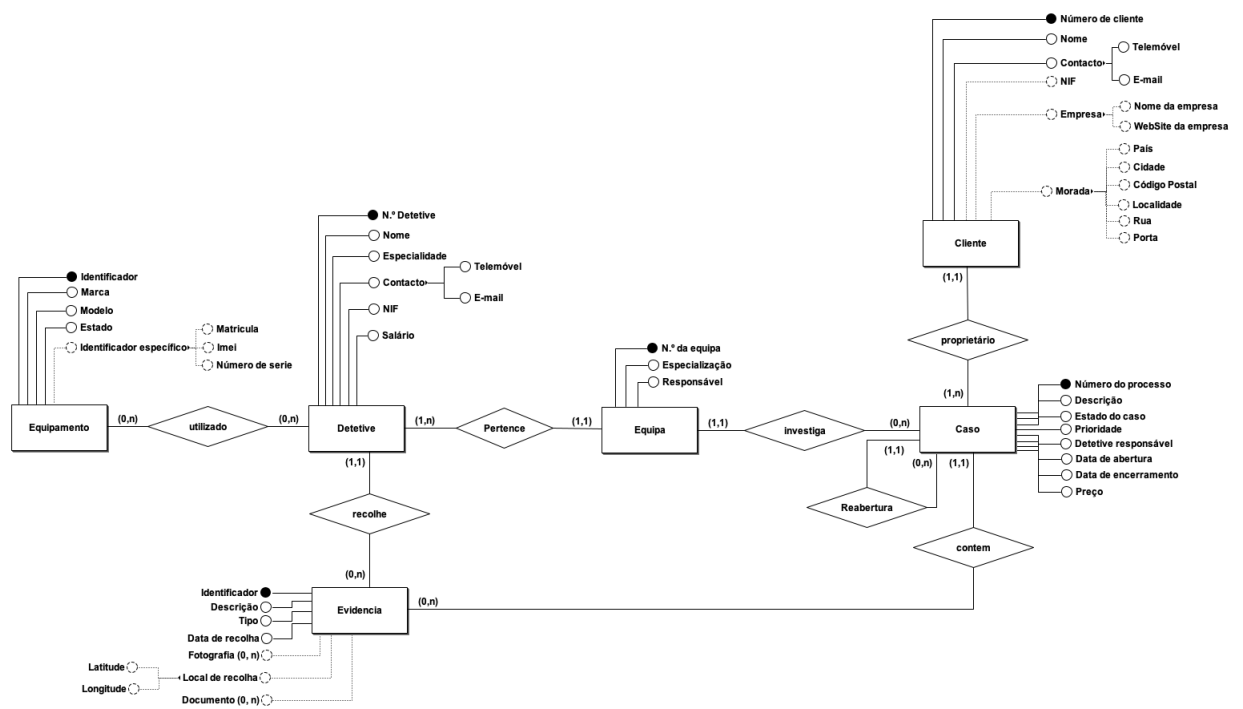


Figura 3.8: Modelo Conceptual



## 4 Modelação Lógica

### 4.1 Construção e Validação do Modelo de Dados Lógico

O modelo conceptual apresentado no pondo anterior serviu como base para a construção do modelo lógico. Para esta modelação, convertemos as entidades do modelo conceptual em tabelas, assim como os relacionamentos e respetivos requisitos associados. Para identificarmos uma entidade utilizámos Primary Keys ou Foreign Keys quando é necessária a associação entre tabelas. Cada entidade é constituída pelas seguintes tabelas:

#### Detetive

- **Primary Key:** Nr Detetive INT
- **Atributos:**
  - Nome VARCHAR
  - Especialidade VARCHAR
  - E-mail VARCHAR
  - Telemóvel INT
  - NIF VARCHAR
  - Salário DECIMAL
- **Foreign Key:** Equipa.Nr Equipa INT

#### Equipa

- **Primary Key:** Nr Equipa INT
- **Atributos:**
  - Especialização VARCHAR
  - Responsável VARCHAR

- **Foreign Key:** Não tem

## Utilizador

- **Primary Key:**
  - Detetive.Nr Detetive INT
  - Equipamento.Identificador INT
- Identificador INT
- **Foreign Key:**
  - Detetive.Nr Detetive INT
  - Equipamento.Identificador INT

## Equipamento

- **Primary Key:** Identificador INT
- **Atributos:**
  - Marca VARCHAR
  - Modelo VARCHAR
  - Estado VARCHAR
  - Matrícula VARCHAR
  - IMEI VARCHAR
  - Número de Série INT
- **Foreign Key:** Não tem

## Evidência

- **Primary Key:** Identificador INT
- **Atributos:**
  - Descrição VARCHAR
  - Tipo VARCHAR
  - Data de Recolha DATETIME

- Latitude INT
- Longitude INT

- **Foreign Key:**

- Detetive.Nr Detetive INT
- Fotografia.idFotografia INT
- Documento.idDocumento INT

## Documento

- **Primary Key:** idDocumento INT

- **Atributos:**

- Documento BLOB

- **Foreign Key:** Não tem

## Fotografia

- **Primary Key:** idFotografia INT

- **Atributos:**

- Fotografia BLOB

- **Foreign Key:** Não tem

## Cliente

- **Primary Key:** Número de Cliente INT

- **Atributos:**

- Nome VARCHAR
- Telemóvel INT
- E-mail VARCHAR
- NIF VARCHAR
- Nome da Empresa VARCHAR
- Website da Empresa VARCHAR

- País VARCHAR
- Cidade VARCHAR
- Rua VARCHAR
- Porta INT
- Código Postal VARCHAR
- Localidade VARCHAR

- **Foreign Key:** Não tem

## Caso

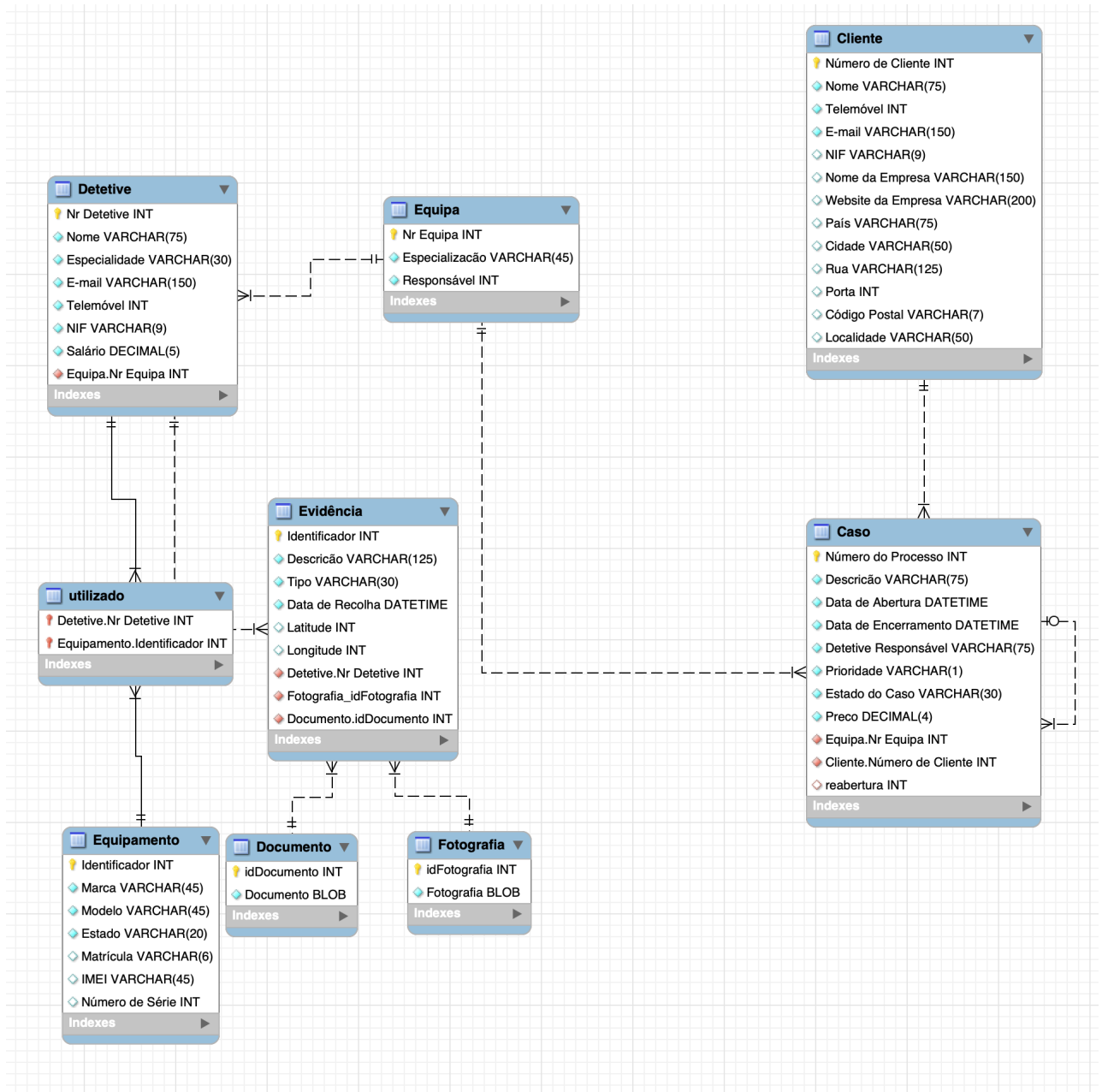
- **Primary Key:** Número do Processo INT
- **Atributos:**
  - Descrição VARCHAR
  - Data de abertura DATETIME
  - Data de Encerramento DATETIME
  - Detetive Responsável VARCHAR
  - Prioridade VARCHAR
  - Estado do Caso VARCHAR
  - País VARCHAR
  - Preço DECIMAL

- **Foreign Key:**
  - Equipa.Nr Equipa INT
  - Cliente.Número de Cliente INT

## Reabertura

- **Primary Key:** Caso.Número do Processo INT
- **Foreign Key:** Caso.Número do Processo INT

## 4.2 Apresentação e Explicação do Modelo Lógico Produzido



Concluído e validado o modelo conceptual, organizamos as entidades e relacionamentos no mysql workbench, com os respetivos tipos de dados necessários a cada atributo. Contudo, uma vez que existem atributos multivalor, surgiram as tabelas "Documento" e "Fotografia" que representam os mesmos. Como explicada anteriormente, existe um relacionamento na entidade "Caso" para a entidade "Caso", este pode ser reaberto gerando assim um novo caso com menção ao anterior, isto é, uma nova chave estrangeira que corresponde ao caso reaberto.

## 4.3 Normalização de Dados

A normalização é um conjunto de regras (1FN, 2FN, 3FN) que visam prevenir a redundância de dados e melhorar o desempenho do modelo. Estas regras analisam os atributos de uma entidade e as relações entre entidades, com o objetivo principal de evitar anomalias na adição, remoção e modificação de registos.

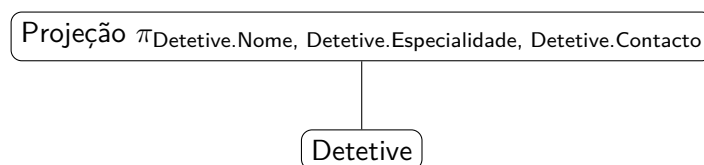
Para confirmar a normalização do modelo, precisamos verificar se ele cumpre as três formas normais:

- **Primeira Forma Normal (1FN):** A 1FN indica que os atributos devem ser atómicos, ou seja, as tabelas não podem ter valores duplicados nem atributos multivalorados. Ao verificar o nosso modelo, podemos confirmar que isso acontece, portanto, a 1FN é cumprida.
- **Segunda Forma Normal (2FN):** A 2FN só pode ser cumprida se a 1FN for satisfeita, e estipula que os atributos não-chave devem depender exclusivamente da chave primária da tabela. Por exemplo, a password do cliente depende unicamente da chave primária, o NIF. O mesmo é observável no resto do modelo, logo, a 2FN é cumprida.
- **Terceira Forma Normal (3FN):** A 3FN indica que, após as duas formas anteriores serem cumpridas, todos os atributos de uma tabela devem ser independentes uns dos outros. Podemos verificar que o nosso modelo não possui nenhum atributo proporcional a outro ou que possa ser gerado a partir de outro, provando que a 3FN é cumprida.

Como podemos verificar no modelo lógico apresentado no capítulo 4.2, este cumpre as três formas normais, portanto, podemos afirmar que o modelo é válido em termos de normalização.

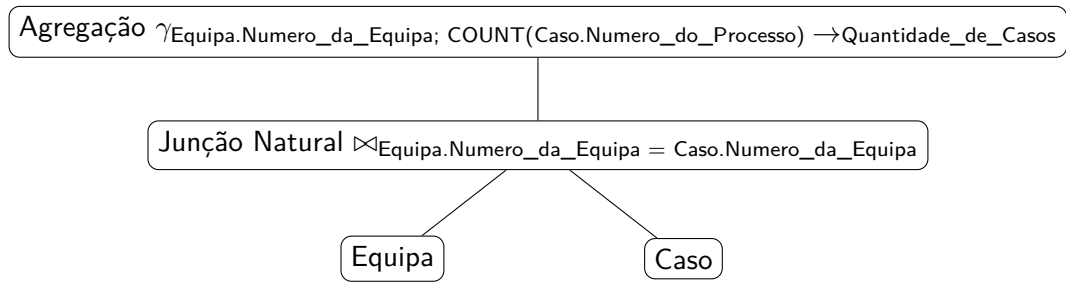
## 4.4 Validação do Modelo com Interrogações do Utilizador

- **Listar e gerir detetives (RM1):**



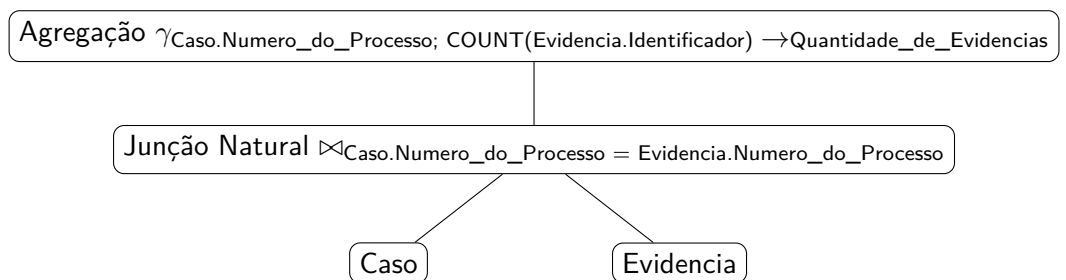
A operação de projeção é usada para selecionar apenas as colunas relevantes da relação Detetive.

- **Calcular a quantidade de casos por equipa (RM3):**



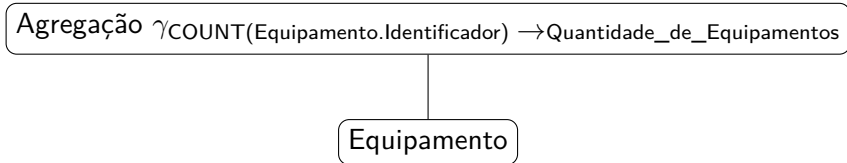
Primeiro, a operação de junção natural é usada para combinar as relações Equipa e Caso com base no número da equipa. Em seguida, a operação de agregação é usada para contar o número de casos para cada equipa.

- **Calcular a quantidade de evidências por caso (RM4):**



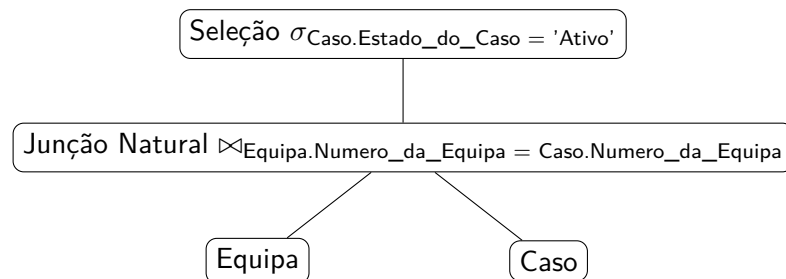
Primeiro, a operação de junção natural é usada para combinar as relações Caso e Evidencia com base no número do processo. Em seguida, a operação de agregação é usada para contar o número de evidências para cada caso.

- **Calcular o número de equipamentos disponíveis (RM5):**



A operação de agregação COUNT é usada para contar o número de identificadores de equipamentos únicos na relação Equipamento.

- **Listar casos ativos por equipa (RM10):**



Primeiro, a operação de junção natural é usada para combinar as relações Equipa e Caso com base no número da equipa. Em seguida, a operação de seleção é usada para filtrar apenas os casos ativos.



## 5 Implementação Física

### 5.1 Apresentação e explicação da base de dados implementada

Neste projeto, optamos por utilizar o MySQL como sistema de gestão de base de dados, pois foi introduzido durante a unidade curricular de Base de Dados.

Após a validação do esquema lógico anteriormente produzido, seguiu-se a validação do modelo com interrogações do utilizador, uma vez concluídas estas etapas passamos então para a criação de um modelo físico. Para traduzir o modelo lógico num sistema MySQL, utilizámos um comando de conversão para o modelo físico. Esse comando gera o código SQL necessário para criar a estrutura da base de dados. O código gerado inclui comandos para criar tabelas, definir chaves primárias, chaves estrangeiras e outros elementos essenciais.

Após a revisão do código SQL, copiámos o mesmo para um script SQL e adicionamos os comandos para criar o schema ou base de dados. Ao executar esse script, a base de dados é criada de acordo com o modelo lógico estabelecido, com todas as tabelas e relacionamentos necessários. Comparando o código de criação das tabelas da base de dados com o ponto 3.4 deste relatório, verificamos que o mesmo vai de encontro ao que foi projetado anteriormente, daí aceitar-mos o código de cada tabela, uma vez que vai de encontro ao esquema lógico e conceptual projetados com fundamentação nos requisitos.

Dessa forma, o modelo lógico é convertido num modelo físico, permitindo a implementação efetiva da base de dados utilizando o MySQL.

O código das tabelas geradas pelo MySQLWorkbench foi o seguinte:

```

CREATE TABLE IF NOT EXISTS `IBMWatson`.`Caso` (
  `NumeroProcesso` INT NOT NULL,
  `Descricao` VARCHAR(75) NOT NULL,
  `EstadoCaso` VARCHAR(30) NOT NULL,
  `Prioridade` VARCHAR(1) NOT NULL,
  `DetetiveResponsavel` VARCHAR(75) NOT NULL,
  `Preco` DECIMAL(4) NOT NULL,
  `DataAbertura` DATETIME NOT NULL,
  `DataEncerramento` DATETIME NULL,
  `NrEquipa` INT NOT NULL,
  `NumeroCliente` INT NOT NULL,
  `Reabertura` INT NULL,
  PRIMARY KEY (`NumeroProcesso`),
  FOREIGN KEY (`NumeroCliente`) REFERENCES Cliente(`NumeroCliente`),
  FOREIGN KEY (`NrEquipa`) REFERENCES Equipa(`NrEquipa`),
  FOREIGN KEY (`Reabertura`) REFERENCES Caso(`NumeroProcesso`)
);

```

Figura 5.1: Código SQL para criar tabela do Caso

```

CREATE TABLE IF NOT EXISTS `IBMWatson`.`Cliente` (
  `NumeroCliente` INT NOT NULL,
  `Nome` VARCHAR(75) NOT NULL,
  `Telemovel` VARCHAR(9) NOT NULL,
  `Email` VARCHAR(150) NOT NULL,
  `NIF` VARCHAR(9) NULL,
  `NomeEmpresa` VARCHAR(150) NULL,
  `WebsiteEmpresa` VARCHAR(200) NULL,
  `Pais` VARCHAR(75) NULL,
  `Cidade` VARCHAR(50) NULL,
  `Rua` VARCHAR(125) NULL,
  `Porta` INT NULL,
  `CodigoPostal` VARCHAR(7) NULL,
  `Localidade` VARCHAR(50) NULL,
  PRIMARY KEY (`NumeroCliente`),
  UNIQUE KEY (`NIF`)
);

```

Figura 5.2: Código SQL para criar tabela do Cliente

```

CREATE TABLE IF NOT EXISTS `IBMWatson`.`Detetive` (
  `NrDetetive` INT NOT NULL,
  `Nome` VARCHAR(75) NOT NULL,
  `Especialidade` VARCHAR(30) NOT NULL,
  `NIF` VARCHAR(9) NOT NULL,
  `Salario` DECIMAL(10, 2) NOT NULL,
  `Email` VARCHAR(150) NOT NULL,
  `Telemovel` VARCHAR(9) NOT NULL,
  `NrEquipa` INT NOT NULL,
  PRIMARY KEY (`NrDetetive`),
  FOREIGN KEY (`NrEquipa`) REFERENCES Equipa(`NrEquipa`),
  UNIQUE KEY (`NIF`)
);

```

Figura 5.3: Código SQL para criar tabela do Detetive

```

CREATE TABLE IF NOT EXISTS `IBMWatson`.`Documento` (
  `IdDocumento` INT NOT NULL,
  `Documento` BLOB NOT NULL,
  PRIMARY KEY (`IdDocumento`)
);

```

Figura 5.4: Código SQL para criar tabela do Documento

```

CREATE TABLE IF NOT EXISTS `IBMWatson`.`Equipa` (
  `NrEquipa` INT NOT NULL,
  `Especialização` VARCHAR(45) NOT NULL,
  `Responsável` INT NOT NULL,
  PRIMARY KEY (`NrEquipa`)
);

```

Figura 5.5: Código SQL para criar tabela da Equipa

```
CREATE TABLE IF NOT EXISTS `IBMWatson`.`Equipamento` (
  `Identificador` INT NOT NULL,
  `Marca` VARCHAR(45) NOT NULL,
  `Modelo` VARCHAR(45) NOT NULL,
  `Estado` VARCHAR(20) NOT NULL,
  `Matricula` VARCHAR(6) NULL,
  `IMEI` VARCHAR(15) NULL,
  `NumeroSerie` VARCHAR(15) NULL,
  PRIMARY KEY (`Identificador`),
  UNIQUE KEY (`Matricula`),
  UNIQUE KEY (`NumeroSerie`)
);
```

Figura 5.6: Código SQL para criar tabela do Equipamento

```
CREATE TABLE IF NOT EXISTS `IBMWatson`.`Evidencia` (
  `Identificador` INT NOT NULL,
  `Descricao` VARCHAR(125) NOT NULL,
  `Tipo` VARCHAR(30) NOT NULL,
  `DataRecolha` DATETIME NOT NULL,
  `Latitude` INT NULL,
  `Longitude` INT NULL,
  `NrDetetive` INT NOT NULL,
  `IdFotografia` INT NOT NULL,
  `IdDocumento` INT NOT NULL,
  `NumeroProcesso` INT NOT NULL,
  PRIMARY KEY (`Identificador`),
  FOREIGN KEY (`NrDetetive`) REFERENCES Detetive(`NrDetetive`),
  FOREIGN KEY (`IdFotografia`) REFERENCES Fotografia(`IdFotografia`),
  FOREIGN KEY (`IdDocumento`) REFERENCES Documento(`IdDocumento`),
  FOREIGN KEY (`NumeroProcesso`) REFERENCES Caso(`NumeroProcesso`)
);
```

Figura 5.7: Código SQL para criar tabela da Evidência

```
CREATE TABLE IF NOT EXISTS `IBMWatson`.`Fotografia` (
  `IdFotografia` INT NOT NULL,
  `Fotografia` BLOB NOT NULL,
  PRIMARY KEY (`IdFotografia`)
);
```

Figura 5.8: Código SQL para criar tabela da Fotografia

```
CREATE TABLE IF NOT EXISTS `IBMWatson`.`Utilizado` (
  `NrDetetive` INT NOT NULL,
  `Identificador` INT NOT NULL,
  PRIMARY KEY (`Identificador`, `NrDetetive`),
  FOREIGN KEY (`NrDetetive`) REFERENCES Detetive(`NrDetetive`),
  FOREIGN KEY (`Identificador`) REFERENCES Equipamento(`Identificador`)
);
```

Figura 5.9: Código SQL para criar tabela Utilizado

## 5.2 Criação de utilizadores da base de dados

De acordo com os requisitos de controlo recolhidos nas reuniões mencionadas no primeiro capítulo, criamos três tipos de utilizadores com diferentes privilégios na base de dados:

- Dr. Watson - Administrador da base de dados com acesso total.

```
CREATE USER 'Watson'@'%' IDENTIFIED BY 'password';
GRANT ALL PRIVILEGES ON IBMWatson.* TO 'Watson'@'%';
FLUSH PRIVILEGES;
```

- Chefe de equipa - Utilizador com permissões específicas ditadas pelos requisitos de controlo.

```
-- Criar chefe de equipa
CREATE USER 'chefe_equipa'@'%' IDENTIFIED BY 'password';

-- Permissões para chefes de equipa
GRANT SELECT ON IBMWatson.caso TO 'chefe_equipa'@'%';
GRANT SELECT ON IBMWatson.cliente TO 'chefe_equipa'@'%';
GRANT SELECT ON IBMWatson.equipamento TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_quantidade_casos_equipa TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_quantidade_evidencias_caso TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_casos_cliente TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_equipamentos_detetive TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_evidencias_detetive_caso TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_evidencias_caso TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_preco_caso TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_detetives_caso TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_caso_e_reaberturas TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_evidencias_caso_e_reaberturas TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_locais_recolha_evidencias_caso TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_detetives_especialidade TO 'chefe_equipa'@'%';
FLUSH PRIVILEGES;
```

- Detetive - Utilizador com permissões específicas ditadas pelos requisitos de controlo.

```

-- Criar detetive
CREATE USER 'detetive'@'%' IDENTIFIED BY 'password';

-- Permissões para detetives
GRANT SELECT ON IBMWatson.caso TO 'detetive'@'%';
GRANT SELECT ON IBMWatson.evidencia TO 'detetive'@'%';
GRANT SELECT ON IBMWatson.equipamento TO 'detetive'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_equipamentos_detetive TO 'detetive'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_evidencias_detetive_caso TO 'detetive'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_evidencias_caso TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_locais_recolha_evidencias_caso TO 'chefe_equipa'@'%';
GRANT EXECUTE ON PROCEDURE IBMWatson.pr_caso_e_reaberturas TO 'chefe_equipa'@'%';
FLUSH PRIVILEGES;

```

## 5.3 Povoamento da base de dados

Neste projeto criamos dois tipos de povoamento para a base de dados:

- Script sql Neste tipo de povoamento, apenas utilizamos instruções INSERT para adicionar registos a cada linha das tabelas.
- Script python Neste script abrimos conexão à base de dados com a biblioteca mysql.connector. A partir desse momento o script lê um ficheiro de dados e linha por linha divide os valores separados por virgulas e determina que tipo de registo é, detetive, cliente, etc. Para cada um destes tipos de registos chama uma função de inserção e por fim efetiva as alterações e fecha a conexão. (O ficheiro seguirá em anexo)

## 5.4 Plano de segurança e recuperação de dados

Um fator extremamente importante no sistema de gestão de base de dados são as cópias de segurança, que garantem confiança e integridade no sistema implementado. No caso do nosso projeto, essas cópias são essenciais para preservar os dados dos clientes e garantir que estejam sempre atualizados a longo prazo, evitando a perda de informações caso ocorra algum incidente.

Com o objetivo de assegurar essa segurança e a recuperação dos dados, decidimos desenvolver um script que permite ao administrador da base de dados agendar backups diários, nos horários desejados, e salvá-los em uma pasta designada. Dessa forma, o administrador pode escolher o momento mais adequado para realizar os backups, de acordo com a disponibilidade do sistema e a demanda operacional da transportadora.

No caso de uma eventual recuperação de dados, o administrador precisará apenas executar o último backup disponível utilizando o script desenvolvido. Esse backup conterá os dados mais recentes da base de dados da transportadora, permitindo a restauração das informações

e garantindo a continuidade das operações sem perda significativa de dados.

(O script seguirá em anexo)

## 5.5 Cálculo do espaço da base de dados

Tabela 5.1: Tamanho de cada tipo de dados

<b>Tipo de Dados</b>	<b>Tamanho (em bytes)</b>
INT	4
VARCHAR(N)	$N + 1$
DATE	3
TINYINT	1
DECIMAL(P, S)	$P + 2$
DATETIME	8
BLOB	variável

Tabela 5.2: Tamanho dos atributos da tabela Equipamento

<b>Atributo</b>	<b>Tipo</b>	<b>Tamanho (em bytes)</b>
Identificador	INT	4
Marca	VARCHAR(45)	46
Modelo	VARCHAR(45)	46
Estado	VARCHAR(20)	21
Matricula	VARCHAR(6)	7
IMEI	VARCHAR(15)	16
NumeroSerie	VARCHAR(15)	16
<b>Total por Entrada</b>		<b>156 bytes</b>

Tabela 5.3: Tamanho dos atributos da tabela Equipa

<b>Atributo</b>	<b>Tipo</b>	<b>Tamanho (em bytes)</b>
NrEquipa	INT	4
Especialização	VARCHAR(45)	46
Responsável	INT	4
<b>Total por Entrada</b>		<b>54 bytes</b>

Para calcular o tamanho inicial da base de dados assumindo 1 registo por tabela, podemos somar os tamanhos de cada registo em todas as tabelas:

- Tabela Equipamento: 156 bytes por registo
- Tabela Equipa: 54 bytes por registo

Tabela 5.4: Tamanho dos atributos da tabela Detetive

<b>Atributo</b>	<b>Tipo</b>	<b>Tamanho (em bytes)</b>
NrDetetive	INT	4
Nome	VARCHAR(75)	76
Especialidade	VARCHAR(30)	31
NIF	VARCHAR(9)	10
Salario	DECIMAL(10, 2)	12
Email	VARCHAR(150)	151
Telemovel	VARCHAR(9)	10
NrEquipa	INT	4
<b>Total por Entrada</b>		<b>298 bytes</b>

Tabela 5.5: Tamanho dos atributos da tabela Cliente

<b>Atributo</b>	<b>Tipo</b>	<b>Tamanho (em bytes)</b>
NumeroCliente	INT	4
Nome	VARCHAR(75)	76
Telemovel	VARCHAR(9)	10
Email	VARCHAR(150)	151
NIF	VARCHAR(9)	10
NomeEmpresa	VARCHAR(150)	151
WebsiteEmpresa	VARCHAR(200)	201
Pais	VARCHAR(75)	76
Cidade	VARCHAR(50)	51
Rua	VARCHAR(125)	126
Porta	INT	4
CodigoPostal	VARCHAR(7)	8
Localidade	VARCHAR(50)	51
<b>Total por Entrada</b>		<b>794 bytes</b>



Tabela 5.6: Tamanho dos atributos da tabela Caso

Atributo	Tipo	Tamanho (em bytes)
NumeroProcesso	INT	4
Descricao	VARCHAR(75)	76
EstadoCaso	VARCHAR(30)	31
Prioridade	VARCHAR(1)	2
DetetiveResponsavel	VARCHAR(75)	76
Preco	DECIMAL(4)	6
DataAbertura	DATETIME	8
DataEncerramento	DATETIME	8
NrEquipa	INT	4
NumeroCliente	INT	4
Reabertura	INT	4
<b>Total por Entrada</b>		<b>226 bytes</b>

Tabela 5.7: Tamanho dos atributos da tabela Fotografia

Atributo	Tipo	Tamanho (em bytes)
IdFotografia	INT	4
Fotografia	BLOB	X
<b>Total por Entrada</b>		<b>X+4 bytes</b>

Tabela 5.8: Tamanho dos atributos da tabela Documento

Atributo	Tipo	Tamanho (em bytes)
IdDocumento	INT	4
Documento	BLOB	Y
<b>Total por Entrada</b>		<b>Y + 4 bytes</b>

Tabela 5.9: Tamanho dos atributos da tabela Evidencia

Atributo	Tipo	Tamanho (em bytes)
Identificador	INT	4
Descricao	VARCHAR(125)	126
Tipo	VARCHAR(30)	31
DataRecolha	DATETIME	8
Latitude	INT	4
Longitude	INT	4
NrDetetive	INT	4
IdFotografia	INT	4
IdDocumento	INT	4
<b>Total por Entrada</b>		<b>189 bytes</b>

Tabela 5.10: Tamanho dos atributos da tabela Utilizado

Atributo	Tipo	Tamanho (em bytes)
NrDetetive	INT	4
Identificador	INT	4
<b>Total por Entrada</b>		<b>8 bytes</b>

- Tabela Detetive: 298 bytes por registo
- Tabela Cliente: 794 bytes por registo
- Tabela Caso: 226 bytes por registo
- Tabela Fotografia: X
- Tabela Documento: Y
- Tabela Evidencia: 189 bytes por registo
- Tabela Utilizado: 8 bytes por registo

Somando os tamanhos de um único registo em todas as tabelas, temos:

$$\text{Tamanho Inicial da Base de Dados} = 156 + 54 + 298 + 794 + 226 + 189 + 8 = 1725 \text{ bytes}$$

Dado que ao fim de um ano temos:

- Cliente: 90 registos
- Detetive: 15 registos
- Caso: 102 registos
- Equipamento: 70 registos
- Equipa: 4 registos
- Evidencia: 500 registos
- Utilizado: 3000 registos
- Fotografia: 200 registos
- Documento: 150 registos

Podemos calcular o tamanho final da base de dados ao fim de um ano:  $\text{Tamanho Final} = 71460 + 4470 + 23052 + 10920 + 216 + 94500 + 24000 + 20480000 + 30720000 = 51256318 \text{ bytes}$

se considerarmos um crescimento de 12% ao ano temos aproximadamente 90.44 MB ao fim de 5 anos.

## 5.6 Definição e caracterização das vistas de utilização em sql

Com o intuito de facilitar a pesquisa e visualização de dados dos casos e das equipas de detetives correspondendo aos casos que investigam, criámos vistas que os agrupam para ser de melhor interpretação pelo utilizador.

```
create view vw_casos_detalhados as
select c.NumeroProcesso, c.Descricao as DescricaoCaso, c.EstadoCaso, c.Prioridade, c.DetetiveResponsavel, c.Preco, c.DataAbertura, c.DataEncerramento, cl.Nome as NomeCliente,
       cl.Email as EmailCliente, cl.Telemovel as TelemovelCliente, e.NrEquipa, e.Especializacao as EspecializacaoEquipa, e.Responsavel as ResponsavelEquipa
from caso c
inner join cliente cl ON c.NumeroCliente = cl.NumeroCliente
inner join equipa e ON c.NrEquipa = e.NrEquipa;
```

Figura 5.16: Código SQL para a visualização dos casos detalhadamente com toda a informação.

```
create view vw_equipas_casos_ativos as
select e.NrEquipa, e.Especializacao as EspecializacaoEquipa, e.Responsavel as ResponsavelEquipa, c.NumeroProcesso, c.Descricao as DescricaoCaso, c.Prioridade, c.DataAbertura, cl.Nome as NomeCliente,
       cl.Email as EmailCliente, cl.Telemovel as TelemovelCliente
from equipa e
inner join caso c ON e.NrEquipa = c.NrEquipa
inner join cliente cl ON c.NumeroCliente = cl.NumeroCliente
where c.EstadoCaso = 'Ativo';
```

Figura 5.17: Código SQL para a visualização das equipas detalhadamente e com toda a informação do/s caso/s que investigam.

## 5.7 Tradução das interrogações do utilizador para SQL

Com base nos requisitos de manipulação, criámos queries SQL para os resolver.

```
-- Calcular quantidade de evidências por caso RM4
delimiter $$
create procedure pr_quantidade_evidencias_caso (IN nr_processo int, OUT res int)
begin
select count(*) as 'Quantidade' into res from evidencia e
       inner join caso c on c.NumeroProcesso = e.NumeroProcesso
       where e.NumeroProcesso = nr_processo;
end $$
delimiter ;
```

Figura 5.10: RM4 - Calcular a quantidade de evidências por caso.

```
-- Visualizar casos de um determinado cliente
delimiter $$
create procedure pr_casos_cliente (IN numero_cliente int, OUT res int)
begin
select * into res from casos c
    inner join cliente cl on c.NumeroCliente = cl.NumeroCliente
    where c.NumeroCliente = numero_cliente;
end $$
delimiter ;
```

Figura 5.11: RM6 - Visualizar casos de um determinado cliente.

```
-- Listar evidências recolhidas por determinado detetive num caso em particular RM9
delimiter $$
create procedure pr_evidencias_detetive_caso (IN nr_detetive INT, IN nr_processo INT, OUT res INT)
begin
select count(*) into res
    from evidencia e
    inner join detetive d ON e.NrDetetive = d.NrDetetive
    inner join caso c ON e.NumeroProcesso = c.NumeroProcesso
    where e.NrDetetive = nr_detetive
    and e.NumeroProcesso = nr_processo;
end $$
delimiter ;
```

Figura 5.12: RM9 - Listar evidências recolhidas por determinado detetive em um caso particular.

```
-- Calcular o preço total de um caso com base no número de detetives que participaram e tempo aberto RM13
delimiter $$
create procedure pr_preco_caso (IN nr_processo int, OUT preco_total int)
begin
select nr_processo, (datediff(c.DataEncerramento, c.DataAbertura) * sum(d.Salario)) as 'Preco Total' from caso c
    inner join detetive d on d.NrEquipa = c.NrEquipa
    where c.NumeroProcesso = nr_processo
    group by c.NumeroProcesso;
end $$
delimiter ;
```

Figura 5.13: RM13 - Calcular o preço total de um caso com base no número de detetives que participaram em tempo aberto.

```
-- Visualizar caso, incluindo todas as suas reaberturas RM16
delimiter $$
create procedure pr_caso_e_reaberturas (IN nr_processo int)
begin
select * from caso c
    where c.NumeroProcesso = nr_processo
union all
select * from caso c
    inner join CasoReabertura cr on c.Reabertura = cr.numeroProcesso;
end $$
delimiter ;
```

Figura 5.14: RM16 - Visualizar caso, incluindo todas as suas reaberturas.

```
-- Visualizar locais de recolha de evidências de um determinado caso RM20
delimiter $$
create procedure pr_locais_recolha_evidencias_caso (in nr_processo int)
begin
select e.Latitude, e.Longitude
    from evidencia e
    inner join detetive d on e.NrDetetive = d.NrDetetive
    inner join equipa eq ON d.NrEquipa = eq.NrEquipa
    where e.NumeroProcesso = nr_processo;
end $$
delimiter ;
```

Figura 5.15: RM20 - Visualizar locais de recolha de evidências de um determinado caso

## 5.8 Indexação do sistema de dados

A indexação das tabelas no sistema da IBM Watson pode trazer diversos benefícios, principalmente no que diz respeito à otimização do desempenho das consultas, por exemplo, ao criar índices nas colunas-chave das tabelas, como o identificador nas tabelas Equipamento e Detetive, podemos acelerar operações de procura e junção de dados, tornando as consultas mais eficientes.

No entanto é importante considerar os custos associados à criação de índices, como o espaço ocupado por entrada e o impacto nas operações de inserção, atualização e exclusão de dados.

## 5.9 Implementação de procedimentos, funções e gatilhos

Com o intuito de facilitar parte do trabalho de nós desenvolvedores, desenvolvemos vários procedimentos úteis para a manipulação da nossa base de dados, no entanto, foi necessária também a implementação das seguintes queries para tornar mais simples e eficaz a utilização da nossa base de dados.

```
delimiter $$
create function CalcularTempoResolucao(numero_processo int)
returns int deterministic
begin
declare TempoResolucao int;
select datediff(ifnull(DataEncerramento, now()), DataAbertura) into TempoResolucao
from caso c
where c.NumeroProcesso = numero_processo;
return TempoResolucao;
end $$
delimiter ;
```

Figura 5.18: Código SQL para uma função que calcula o tempo de resolução de um determinado caso.

```
delimiter $$
create trigger AtualizarEstadoEquipamento
after insert on utilizado
for each row
begin
update equipamento e
set e.Estado = 'Indisponível'
where e.Identificador = new.Identificador;
end $$
delimiter ;
```

Figura 5.19: Código SQL para um trigger que atualiza um certo equipamento quando este é utilizado por um detetive, passando a estar indisponível.

## 6 Conclusões e Trabalho Futuro

O trabalho realizado neste projeto apresenta pontos fortes significativos. A abordagem multifacetada para o levantamento e análise de requisitos permitiu uma compreensão abrangente das necessidades e expectativas dos utilizadores. A identificação e caracterização cuidadosa das entidades e dos relacionamentos garantem que o sistema seja capaz de lidar eficientemente com a gestão de informações e documentos, bem como com a organização de funcionários e equipamentos.

A implementação física da base de dados foi concluída com sucesso, permitindo a validação prática do modelo de dados proposto. A eficácia do sistema em melhorar a eficiência operacional da empresa foi confirmada, demonstrando a viabilidade e utilidade do sistema desenvolvido.

No entanto, como em qualquer projeto, existem áreas para melhoria e expansão. Uma das dificuldades encontradas durante o desenvolvimento do projeto foi a possível falta de alguns requisitos de controlo. Isso pode ter impacto na segurança e na eficácia do sistema, e é uma área que requer atenção adicional no futuro.

Em suma, o projeto foi um sucesso, cumprindo os objetivos estabelecidos e proporcionando uma solução eficaz para os desafios enfrentados pela “I.B.M. Watson”. O sistema de gestão de bases de dados desenvolvido representa um passo significativo para melhorar a gestão de informações e documentos, a organização de funcionários e equipamentos, e a eficiência operacional da empresa. A equipa está ansiosa para continuar a melhorar e expandir o sistema no futuro.

### **Trabalho Futuro:**

Futuramente o sistema pode acomodar novas queries baseadas unicamente nos atributos das tabelas. Como parte do plano de backup, a equipa tem as condições necessárias para melhorar as permissões de cada utilizador na base de dados, assim como estudar e implementar um serviço de backup automático.

# Referências

Connolly, T., & Begg, C. (Year). Database Systems: A Practical Approach to Design, Implementation, and Management (6th ed.). Casos de estudo apresentados em sala de aula

# Lista de Siglas e Acrónimos

**BD** Base de Dados

**RD** Requisitos de descrição

**RM** Requisitos de manipulação

**RC** Requisitos de controlo

**I.B.M. Watson** Instituto de buscas metódico Watson



# 7 Anexos

## 7.1 Script de Povoamento:

```
1 import datetime
2 import mysql.connector
3 from mysql.connector import errorcode
4
5 # Função para conectar ao banco de dados
6 def conectar():
7     config = {
8         'user': 'root',
9         'password': 'basededados24',
10        'host': 'localhost', # Alterar para o endereço do servidor de
        banco de dados
11        'database': 'IBMWatson',
12        'port': '3306'
13    }
14    conn = mysql.connector.connect(**config)
15    return conn
16
17 # Função para inserir um detetive na tabela de detetives
18 def inserir_detetive(cursor, detetive):
19     query = "INSERT INTO Detetive (NrDetetive, Nome, Especialidade, NIF
20     , Salario, Email, Telemovel, NrEquipa) VALUES (%s, %s, %s, %s, %s,
21     %s, %s, %s)"
22     cursor.execute(query, detetive)
23
24 # Função para inserir um equipamento na tabela de equipamentos
25 def inserir_equipamento(cursor, equipamento):
26     query = "INSERT INTO Equipamento (Identificador, Marca, Modelo,
27     Estado, Matricula, IMEI, NumeroSerie) VALUES (%s, %s, %s, %s, %s, %
28     s, %s)"
29     cursor.execute(query, equipamento)
30
31 # Função para inserir um cliente na tabela de clientes
32 def inserir_cliente(cursor, cliente):
33     query = "INSERT INTO Cliente (NumeroCliente, Nome, Telemovel, Email
34     , NIF, NomeEmpresa, WebsiteEmpresa, Pais, Cidade, Rua, Porta,
35     CodigoPostal, Localidade) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %
36     s, %s, %s, %s, %s)"
37     cursor.execute(query, cliente)
38
39 # Função para inserir um caso na tabela de casos
```

```

33 def inserir_caso(cursor, caso):
34     # Converta 'NULL' para None
35     caso = list(caso)
36     if caso[2] == 'NULL':
37         caso[2] = None
38
39     if caso[3] == 'NULL':
40         caso[3] = None
41         caso = list(caso)
42
43     if caso[10] == 'NULL':
44         caso[10] = None
45     caso = tuple(caso)
46
47     query = "INSERT INTO Caso (NumeroProcesso, Descricao, DataAbertura,
48         DataEncerramento, DetetiveResponsavel, Prioridade, EstadoCaso,
49         Preco, NrEquipa, NumeroCliente, Reabertura) VALUES (%s, %s, %s, %s
50         , %s, %s, %s, %s, %s, %s, %s)"
51     cursor.execute(query, caso)
52
53 # Função para inserir uma evidencia na tabela de evidencias
54 def inserir_evidencia(cursor, evidencia):
55     query = "INSERT INTO Evidencia (Identificador, Descricao, Tipo,
56     DataRecolha, Latitude, Longitude, NrDetetive, IdFotografia,
57     IdDocumento) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s)"
58     cursor.execute(query, evidencia)
59
60 def inserir_fotografia(cursor, fotografia):
61     # id_fotografia, path_to_image = fotografia
62     # with open(path_to_image, 'rb') as file:
63     #     blob_data = file.read()
64     query = "INSERT INTO Fotografia (IdFotografia, Fotografia) VALUES
65     (%s, %s)"
66     cursor.execute(query, fotografia)
67
68 def inserir_documento(cursor, documento):
69     # id_documento, path_to_document = documento
70     # with open(path_to_document, 'rb') as file:
71     #     blob_data = file.read()
72     query = "INSERT INTO Documento (IdDocumento, Documento) VALUES (%s,
73     %s)"
74     cursor.execute(query, documento)
75
76 # Função para inserir uma equipa na tabela de equipas
77 def inserir_equipa(cursor, equipa):
78     query = "INSERT INTO Equipa (NrEquipa, Especialização, Responsável
79     ) VALUES (%s, %s, %s)"
80     cursor.execute(query, equipa)
81
82 # Função principal
83 def main():
84     conn = conectar()
85     cursor = conn.cursor()

```

```

78
79     # Abrir o arquivo de dados
80     with open('C:/Users/diogo/Desktop/BD/Populacao/dados.txt', 'r',
81               encoding='utf-8') as arquivo:
82         linhas = arquivo.readlines()
83         for linha in linhas:
84             # Fazer o parsing dos valores separados por vírgula
85             valores = linha.strip().split(',')
86
87             # Verificar o tipo de registro (detetive, equipamento,
88             cliente, caso, evidencia)
89             tipo_registro = valores[0]
90             dados_registro = tuple(valores[1:])
91
92             if tipo_registro == 'detetive':
93                 inserir_detetive(cursor, dados_registro)
94             elif tipo_registro == 'equipamento':
95                 inserir_equipamento(cursor, dados_registro)
96             elif tipo_registro == 'cliente':
97                 inserir_cliente(cursor, dados_registro)
98             elif tipo_registro == 'caso':
99                 inserir_caso(cursor, dados_registro)
100            elif tipo_registro == 'evidencia':
101                inserir_evidencia(cursor, dados_registro)
102            elif tipo_registro == 'fotografia':
103                inserir_fotografia(cursor, dados_registro)
104            elif tipo_registro == 'documento':
105                inserir_documento(cursor, dados_registro)
106            elif tipo_registro == 'equipa':
107                inserir_equipa(cursor, dados_registro)
108
109            # Efetivar as alterações no banco de dados
110            conn.commit()
111            cursor.close()
112            conn.close()
113
114 if __name__ == '__main__':
115     main()

```

## 7.2 Script de Backup e Restauo

```

1 import subprocess
2 import time
3
4 def backup_db(host, user, password, db_name, backup_path):
5     # Cria um timestamp
6     timestamp = time.strftime('%Y%m%d%H%M%S')
7
8     # Define o comando mysqldump
9     dumpcmd = f"mysqldump -h {host} -u {user} -p{password} {db_name} >
10    {backup_path}/IBMWatson_{timestamp}.sql"

```

```
10
11     # Executa o comando
12     subprocess.call(dumpcmd, shell=True)
13
14 def restore_db(host, user, password, db_name, backup_file):
15     # Define o comando mysql
16     restorecmd = f"mysql -h {host} -u {user} -p{password} {db_name} < {
17         backup_file}"
18
19     # Executa o comando
20     subprocess.call(restorecmd, shell=True)
21
22 # Exemplo de uso
23 # backup_db('localhost', 'root', 'password', 'IBMWatson', '/path/to/
24     backup/directory')
25 # restore_db('localhost', 'root', 'password', 'IBMWatson', '/path/to/
26     backup/file.sql')
```