



Universidade do Minho
Licenciatura em Engenharia Informática

Unidade Curricular de Bases de Dados

Ano Letivo de 2023/2024

Detetives renascidos

Diogo Esteves (a104004), Rodrigo Ferreira
(a104531), João Cunha (104611), Gonçalo Alves
(a104079), João Sá (a104612)

03/2024

BD

Data de Recepção	
Responsável	
Avaliação	
Observações	

Detetives renascidos

Diogo Esteves (a104004), Rodrigo Ferreira (a104531), João Cunha (104611), Gonçalo Alves (a104079), João Sá (a104612)

03/2024

Resumo

No âmbito da Unidade Curricular de bases de dados, foi-nos dada a tarefa de criar uma base de dados relacional a ser implementada para uma agência de detetives. Para isso, faremos uma contextualização do trabalho, onde enquadraremos a necessidade da criação de uma base de dados com a história da agência e onde apresentaremos uma justificação da viabilidade e objetivos deste projeto.

Após apresentarmos a contextualização do trabalho, vamos fazer um levantamento dos requisitos necessários à criação da base de dados e faremos a sua caracterização. Além disso, faremos a documentação associada à criação dos sistemas que serão criados com base nos requisitos levantados: sistema concetual e lógico.

Será posteriormente necessário fazer a implementação física, onde implementaremos na prática (com o auxílio das ferramentas associadas ao *MySQL*) os sistemas referidos anteriormente. De seguida, faremos a sua povoação com as fontes de dados necessárias e relevantes ao problema. Acabaremos com a análise dos dados e a definição dos *Dashboards* necessários para suporte dos requisitos de gestão, bem como a sua implementação em *MS Power BI*.

Por fim, finalizaremos com a análise crítica do trabalho realizado junto dos docentes da cadeira, com o objetivo de aperfeiçoar e corrigir possíveis erros cometidos ao longo do desenvolvimento do projeto.

Área de Aplicação: Projeção e desenvolvimento de um sistema de BD aplicado a uma agência de detetives.

Palavras-Chave: Bases de Dados, Análise de Requisitos, Levantamento de Requisitos, Entidades, Atributos, Relações, Modelo Conceptual, Modelo Lógico, Normalização, *MySQL Workbench*, *SQL*, *brModelo*, Agência Fénix, *Chen*

RESUMO.....	I
ÍNDICE DE FIGURAS	IV
ÍNDICE DE TABELAS	V
1. DEFINIÇÃO DO SISTEMA.....	1
1.1. CONTEXTUALIZAÇÃO	1
1.2. MOTIVAÇÃO E OBJETIVOS.....	2
1.3. ANÁLISE DA VIABILIDADE DO PROJETO	2
1.4. RECURSOS E EQUIPA DE TRABALHO	3
1.5. PLANO DE EXECUÇÃO DO PROJETO	4
2. LEVANTAMENTO E ANÁLISE DE REQUISITOS.....	6
2.1. MÉTODO DE LEVANTAMENTO E ANÁLISE DE REQUISITOS ADOTADO.....	6
2.2. ORGANIZAÇÃO DOS REQUISITOS LEVANTADOS.....	7
2.2.1 <i>Requisitos de Descrição</i>	7
2.2.2 <i>Requisitos de Manipulação</i>	9
2.2.3 <i>Requisitos de Controlo</i>	11
2.3. ANÁLISE E VALIDAÇÃO GERAL DOS REQUISITOS.....	12
3.MODELAÇÃO CONCETUAL.....	13
3.1 <i>Apresentação da Abordagem de Modelação Realizada</i>	13
3.2 <i>Identificação e Caracterização das Entidades</i>	14
3.3 <i>Identificação e Caracterização dos Relacionamentos</i>	15
3.4. <i>Identificação e Caracterização dos Atributos das Entidades e dos Relacionamentos</i>	22
3.5 <i>Apresentação e Explicação do Diagrama ER Produzido</i>	31
4. MODELAÇÃO LÓGICA	29
4.1. CONSTRUÇÃO E VALIDAÇÃO DO MODELO DE DADOS LÓGICO	29
4.2. APRESENTAÇÃO E EXPLICAÇÃO DO MODELO LÓGICO	32
4.3. NORMALIZAÇÃO DE DADOS	33
4.4. VALIDAÇÃO DO MODELO COM INTERROGAÇÕES DO UTILIZADOR.....	34
5. IMPLEMENTAÇÃO FÍSICA.....	37
5.1. SELEÇÃO DO SISTEMA DE GESTÃO DE BASES DE DADOS.....	37
5.2. APRESENTAÇÃO E EXPLICAÇÃO DA BASE DE DADOS IMPLEMENTADA	38
5.3. CRIAÇÃO DE UTILIZADORES DA BASE DE DADOS	41
5.4. POVOAMENTO DA BASE DE DADOS	42
5.4.1 <i>Inserção a partir do script SQL</i>	42
5.4.2 <i>Inserção a partir do script C</i>	46
5.5. CÁLCULO DO ESPAÇO DA BASE DE DADOS (INICIAL E TAXA DE CRESCIMENTO ANUAL)	47
5.6. DEFINIÇÃO E CARACTERIZAÇÃO DE VISTAS DE UTILIZAÇÃO EM SQL	48

5.6.1 Vista de Clientes ordenados por Nome	48
5.6.2 Vista de Suspeitos ordenados por Nome	49
5.6.3 Vista dos Casos ordenados pelo Nr_Registo	49
5.7. TRADUÇÃO DAS INTERROGAÇÕES DO UTILIZADOR PARA SQL	49
5.7.1. Projetar o Nº de Casos Atribuídos e Resolvidos de cada Detetive	50
5.7.2. Projetar todos os Detetives por ordem decrescente de Ratio	50
5.7.3. Projetar qual é o detetive que está a investigar cada caso	50
5.7.4. Projetar todos os Clientes por ordem crescente do Nome	50
5.8. INDEXAÇÃO DO SISTEMA DE DADOS	51
5.9. IMPLEMENTAÇÃO DE PROCEDIMENTOS, FUNÇÕES E GATILHOS	51
5.9.1. Função CalculaRatio	51
5.9.2. Procedure DetetivesPorClassificação	52
5.9.3. Trigger AtualizaNrDetetives	52
6. ALTERAÇÕES	54
6.1. Modelo Concetual	54
6.2. Modelo Lógico	54
6.3. Requisitos	54
7. CONCLUSÕES.....	55
BIBLIOGRAFIA.....	56
LISTA DE SIGLAS E ACRÓNIMOS	57
ANEXOS	58

Índice de Figuras

<i>FIGURA 1 - DIAGRAMA DE GANTT</i>	<i>4</i>
<i>FIGURA 3 - RELACIONAMENTO CLIENTE-AGÊNCIA.....</i>	<i>16</i>
<i>FIGURA 4 - RELACIONAMENTO CLIENTE-CASO</i>	<i>17</i>
<i>FIGURA 5 - RELACIONAMENTO CASO-SUSPEITO</i>	<i>18</i>
<i>FIGURA 6 - RELACIONAMENTO AGÊNCIA-DETETIVES.....</i>	<i>19</i>
<i>FIGURA 7- RELACIONAMENTO DETETIVE- DETETIVE</i>	<i>20</i>
<i>FIGURA 8 - RELACIONAMENTO DETETIVE - CASO.....</i>	<i>21</i>
<i>FIGURA 9 - MODELO CONCTUAL.....</i>	<i>28</i>
<i>FIGURA 11 - ÁRVORE CORRESPONDENTE À 1ª INTERROGAÇÃO</i>	<i>34</i>
<i>FIGURA 12 - ÁRVORE CORRESPONDENTE À 2ª INTERROGAÇÃO</i>	<i>35</i>
<i>FIGURA 13 - ÁRVORE CORRESPONDENTE À 3ª INTERROGAÇÃO</i>	<i>36</i>
<i>FIGURA 14 - ÁRVORE CORRESPONDENTE À 4ª INTERROGAÇÃO</i>	<i>37</i>

[OBJ]

Índice de Tabelas

<i>TABELA 1 - CARACTERIZAÇÃO DOS ATRIBUTOS DA AGÊNCIA.....</i>	<i>22</i>
<i>TABELA 2 - CARACTERIZAÇÃO DOS ATRIBUTOS DO DETETIVE</i>	<i>23</i>
<i>TABELA 3 - CARACTERIZAÇÃO DOS ATRIBUTOS DO CLIENTE</i>	<i>24</i>
<i>TABELA 4 - CARACTERIZAÇÃO DOS ATRIBUTOS DO CASO</i>	<i>25</i>
<i>TABELA 5 - CARACTERIZAÇÃO DOS ATRIBUTOS DO SUSPEITO.....</i>	<i>26</i>

1. Definição do Sistema

Este relatório apresenta um caso de estudo sobre a implementação de uma base de dados para modernizar agências de detetives fictícias, no âmbito da unidade curricular de Bases de Dados. Exploraremos os desafios enfrentados pelas agências, seus objetivos, o contexto e a motivação por trás do projeto.

1.1. Contextualização

Num passado não muito distante, fazíamos parte de uma equipa de detetives que fundou uma agência de detetives de renome sediada inicialmente em Matosinhos, Porto (consultar anexo III)- Agência Fénix. A agência dividiu-se em vários polos, espalhados pelo país, para conseguir atender às necessidades das diferentes zonas. Outrora fomos conhecidos por desvendar os casos mais intrigantes e desafiantes que cruzavam o nosso caminho. Ao longo das nossas carreiras como detetives, as nossas habilidades foram aclamadas e a nossa reputação cresceu junta da nossa equipa que sempre procurou ajudar as autoridades locais a trazer justiça à nossa sociedade.

No entanto, à medida que o tempo passava, cada um de nós seguiu caminhos distintos, resultando na separação da equipa. Isto fez com que a agência outrora fundada por nós passasse para as mãos de novos detetives e mais tarde de gestores a quem confiamos o tão aclamado projeto. Parecia que os dias de investigação conjunta haviam ficado para trás, apenas uma memória de um tempo glorioso.

Mas o destino, como sempre, tinha outros planos para a nossa equipa. Após vários anos, a agência viu-se ultrapassada por agências concorrentes que tiraram partido da evolução da tecnologia de bases de dados para aumentar a eficiência das suas equipas. Com isto, a nossa agência ficava cada vez mais para baixo na lista de prioridade das agências a serem selecionadas para investigar casos que, ao longo do tempo, também se foram tornando mais e mais complexos.

Olhando para o estado atual da agência que fundamos, decidimos reunirmo-nos como antigamente, no nosso antigo escritório, agora empoeirado. Olhamos uns para os

outros, vendo os traços da idade e experiência acumulados e decidimos voltar a entrar no projeto. Apesar de as nossas habilidades de detetive já não serem tao afiadas como antigamente, os nossos anos de experiência noutros ramos como a área de bases de dados, ainda poderiam ser uteis para o renascimento da nossa agência, tornando-a uma agência de renome e respeitada como foi nos seus primórdios.

Reconhecemos que, para lidar com o crime moderno e com os desafios da investigação atual, era necessário mais do que apenas instinto e intuição: era necessário o poder da tecnologia e da análise de dados.

Assim, em vez de voltar ao terreno, decidimos direcionar os nossos esforços para um novo tipo de projeto adaptado à agência, construindo uma base de dados.

1.2. Motivação e Objetivos

A motivação deste caso de estudo surge de dois principais objetivos. Primeiramente, busca-se oferecer uma solução prática para os problemas enfrentados pela agência, projetando uma base de dados que possa gerir eficientemente informações sobre casos, clientes, detetives, e suspeitos, além de facilitar análises complexas e apoio à decisão. Em segundo lugar, o projeto proporciona uma oportunidade valiosa para aplicar os conhecimentos adquiridos na unidade curricular, demonstrando a aplicabilidade dos conceitos de bases de dados em situações reais.

O principal objetivo da construção da base de dados será tornar a agência mais competitiva e eficiente. Uma das principais medidas que a agência pode tomar assim que a base de dados esteja concluída está relacionada com a formação das equipas de investigação: ao recolherem-se dados acerca da taxa de sucesso de cada detetive e da sua especialização, por exemplo, será possível a formação de equipas personalizadas a cada um dos casos, bem como serão potencializadas as capacidades de cada detetive, aumentando à partida a sua taxa de sucesso individual. Com isto o número de casos resolvidos pela agência aumentará e esta voltará a tornar-se competitiva no mercado como assim o era aquando da sua fundação.

1.3. Análise da Viabilidade do Projeto

Apesar de não termos acesso a uma Base de Dados de agências de detetives conhecemos as informações necessárias para o funcionamento da mesma.

Este sistema implicará guardar os registos e informações tanto dos clientes como dos detetives, casos e ainda os respetivos suspeitos, para garantir o bom funcionamento das agências e a eficácia dos nossos agentes no terreno. Será necessário fazer com

que o sistema reconheça esses tais dados dos clientes e dos agentes, assim como dos suspeitos e dos casos juntamente com as várias funcionalidades que são acatadas por cada entidade. Por exemplo, a reabertura de casos já encerrados, a alteração da descrição de um caso, entre outros.

Isto implica que o sistema guarde, processe e relacione os dados, e ainda que nos dê a disponibilidade de alterar e pesquisar sobre os mesmos.

Tendo tudo isto em conta, decidimos que um *DBMS (Database management system)* seria o mais indicado para pôr em prática a resolução do problema apresentado, uma vez que permite a realização de várias operações de gestão de dados.

De forma a atingir os objetivos propostos, foram formados vários pontos de foco:

- Treinar os estagiários de forma a garantir o sucesso da Agência no futuro
- Otimizar a investigação dos casos, facilitando o acesso à informação relativa ao mesmo.
- Otimizar a distribuição de recursos, sejam estes humanos e/ou materiais, de modo a aumentar a eficácia no momento.

1.4. Recursos e Equipa de Trabalho

Para o sucesso garantido deste projeto, é crucial a utilização de diversos tipos de recursos estratégicos escolhidos pela nossa equipa de ex-Detetives.

No âmbito dos Recursos Humanos, contaremos com uma equipa de detetives altamente qualificados, cada um ocupando diferentes patentes na agência, desde Estagiário até Chefe (Estagiário, investigador, Sênior, Chefe), contribuindo com as suas habilidades únicas e experiências diversas.

No que diz respeito aos Recursos Materiais, será essencial investir em dispositivos de última geração para suportar o desenvolvimento do DBSM inovador que pretendemos desenvolver. Isso incluirá melhorias significativas nas infraestruturas da sede da Oagência, proporcionando um ambiente adequado para o sucesso do projeto.

1.5. Plano de execução do Projeto

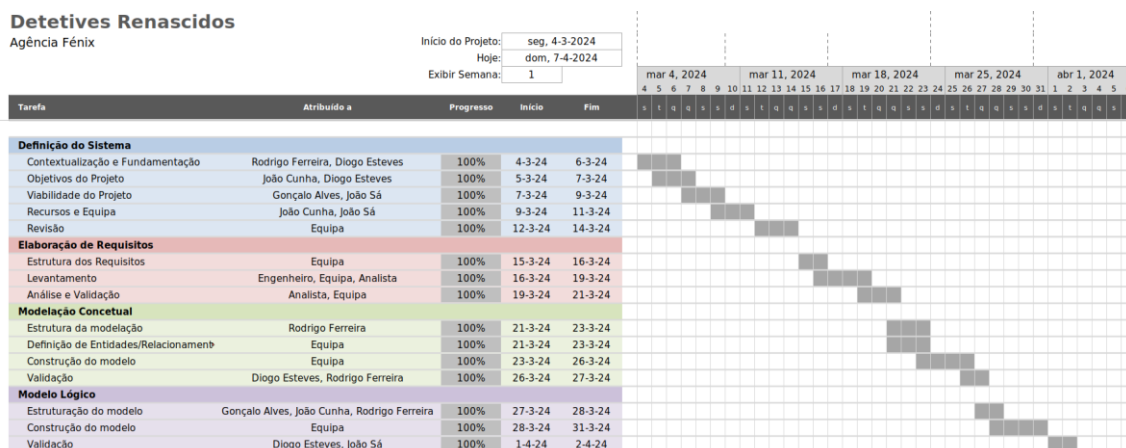


Figura 1 - Diagrama de GANTT

Este é o Diagrama de GANTT que ilustra minuciosamente como cada etapa do projeto foi executada e atribuída aos vários profissionais envolvidos. Utilizamos o termo “Equipa” para representar uma etapa que foi realizada por todos os membros da ex-Agência Fénix envolvidos no desenvolvimento da base de dados (João Cunha, João Sá, Rodrigo Ferreira, Gonçalo Alves, Diogo Esteves).

Cada etapa teve um tempo associado. Em relação a parte da “Definição do Sistema” achamos pertinente ter 3 dias para cada sub-etapa e intercalamos a “Contextualização e Fundamentação” com os “Objetivos do Projeto” por serem temas bem claros para todos os 5 membros.

No que toca á “Elaboração de Requisitos”, decidimos atribuir 4 dias á sua realização. Esta foi uma etapa em que todos estruturamos anteriormente o que iria ser feito nos primeiros dois dias. De seguida, dividimos a equipa pelos diferentes tipos de requisitos (Descrição, manipulação e Controlo) para facilitar e otimizar o seu levantamento, que foi concluído entre os dias 16 e 19 de março. Por fim, juntámo-nos todos para analisar e validar o trabalho prévio, para garantir o sucesso futuro do nosso projeto.

O desenvolvimento do modelo conceitual seguiu um cronograma organizado em quatro fases principais. Inicialmente, nos primeiros três dias focamos na estruturação geral e definição dos objetivos, garantindo uma compreensão clara das metas a serem alcançadas. Em seguida, no mesmo período, concentramo-nos pela identificação e definição das entidades e relacionamentos fundamentais. Posteriormente, entre os dias 23/03/2024 e 26/03/2024 traduzimos os conceitos em representações visuais, mais concretamente utilizando o programa *brModelo*, utilizando técnicas como a modelagem entidade-relacionamento (ER) para visualizar a estrutura da base de dados de forma clara e compreensível.

Por fim, reservamos o período de 26/03/2024 a 27/03/2024 para realizar uma revisão minuciosa do modelo, com o objetivo de identificar inconsistências e erros conceituais. Durante esta fase, realizamos iterações necessárias para garantir a integridade e coerência do modelo conceitual, assegurando que estivesse alinhado com as necessidades da Agência Fênix. Essas etapas foram conduzidas de forma diligente e colaborativa, resultando num modelo conceitual sólido e bem elaborado.

Durante o desenvolvimento do modelo lógico para a base de dados, seguimos três fases principais. Primeiramente, entre os dias 27/03/2024 e 28/03/2024, dedicamos tempo para estruturar o modelo, delineando tabelas, chaves primárias e quais chaves estrangeiras deveríamos colocar. Em seguida, de 28/04 até 31/04, convertendo o modelo conceitual para o modelo lógico usando a ferramenta *brModelo* ponto de partida e, em seguida, implementando-o utilizando SQL. Por fim, nos dias 01/04/2024 a 02/04/2024, realizamos a validação do modelo, revisando a sua estrutura e dados para garantir sua conformidade em requisitos do projeto. Realizamos uma validação prática por meio de interações com os usuários finais. Durante essa interação, apresentamos ao usuário um conjunto de expressões previamente definidas nos requisitos de manipulação estabelecidos anteriormente. Qualquer inconsistência ou lacuna identificada durante essa validação foi corrigida para garantir a integridade e eficácia do modelo lógico. Essa abordagem prática e iterativa permitiu que refinássemos o modelo de acordo com o feedback direto dos usuários, assegurando sua adequação às exigências do projeto.

2. Levantamento e Análise de Requisitos

2.1. Método de Levantamento e Análise de Requisitos Adotado

Para uma melhor compreensão e organização dos problemas propostos pela agência passamos ao levantamento de requisitos necessários para o funcionamento da BD que vamos desenvolver. Para isso, fizemos diversas reuniões com os atuais gestores da agência e fizemos observações, no terreno, junto dos detetives para adquirirmos uma ideia mais próxima da realidade dos problemas que a BD se proporá a resolver.

Por fim, após um levantamento inicial dos requisitos, marcamos uma nova reunião com os gestores para proceder à validação e aprimoramento dos mesmos. Dessa reunião resultaram os requisitos finais que a BD deverá respeitar e que estão presentes nos tópicos abaixo.

Com o intuito de recolher informações acerca do funcionamento, na íntegra, de uma agência de detetives, decidimos recorrer a vários métodos de procura e obtenção de informação, de forma a abranger todas as perspectivas e casos possíveis.

Entre esses métodos de recolha de informação estão:

- **Entrevistas:** Foram realizadas entrevistas com detetives da agência para entender, nomeadamente, as suas responsabilidades, metodologias de trabalho, procedimentos de documentação e protocolos de segurança. Deste modo foi possível perceber as necessidades que os detetives (os principais utilizadores da base de dados) têm e como seria a sua interação com o sistema a ser desenvolvido.
(Consultar Anexo I)
- **Observação Participativa e Simulações de Casos:** Participação ativa em simulações de casos (geralmente realizadas como método de treino para detetives) e observação direta das operações diárias da agência com o intuito de entender os fluxos de trabalho, interações entre a equipa de detetives e clientes, e a dinâmica geral do ambiente de trabalho.
- **Análise de gestão de informações:** Foram examinados documentos relacionados ao funcionamento da agência, tais como manuais de procedimentos, relatórios de casos anteriores, políticas de privacidade e regulamentos governamentais pertinentes. Esta análise fornecerá *insights* sobre

os tipos de informações coletadas, armazenadas e utilizadas pela agência de forma a manter o seu funcionamento habitual.
(Consultar Anexo II)

2.2. Organização dos Requisitos Levantados

2.2.1 Requisitos de Descrição

RD1. As agências têm uma morada (com rua, nº da porta e código postal), designação, nº de detetives, polo e identificador associados. O identificador da agência é único. (Data: 17/03/2024 10:23 | Fonte: Inês Rodrigues | Analista: Diogo Esteves)

RD2. Às agências são atribuídos casos a investigar e cada caso é investigado apenas pelas diversas agências.
(Data: 17/03/2024 10:25 | Fonte: Inês Rodrigues | Analista: Diogo Esteves)

RD3. Cada caso é apresentado por um cliente e cada cliente apresenta um ou vários casos às agências.
(Data: 17/03/2024 10:30 | Fonte: Inês Rodrigues | Analista: Diogo Esteves)

RD4. Um caso arquivado (fechado) pode ser reaberto.
(Data: 18/03/2024 12:32 | Fonte: Francisco Santos | Analista: Rodrigo Ferreira)

RD5. A cada cliente está associado um número identificador único, um nome, um número de contribuinte, data nascimento e género.
(Data: 19/03/2024 10:00 | Fonte: Luís Pereira | Analista: João Cunha)

RD6. A cada caso está associado o seu número de registo, a sua descrição, a sua área, estado atual (em andamento, aberto ou fechado), data de início e fecho (opcional) e lista de evidências.
(Data: 18/03/2024 12:37 | Fonte: Francisco Santos | Analista: Rodrigo Ferreira)

RD7. Cada caso pode ser aberto com ou sem suspeitos associados e cada suspeito pode estar associado a um ou mais casos.

(Data: 20/03/2024 15:40 | Fonte: Mariana Costa | Analista: João Sá)

RD8. Cada caso é investigado por um ou mais detetives e um detetive investiga um ou mais casos.

(Data: 18/03/2024 11:12 | Fonte: Francisco Santos | Analista: Rodrigo Ferreira)

RD9. Cada detetive tem um número interno único associado, uma área de especialização, nome, salário, patente e classificação associados (com o número de casos resolvidos, atribuídos e o ratio).

(Data: 21/03/2024 10:45 | Fonte: Diogo Oliveira | Analista: Gonçalo Alves)

RD10. Quando um caso envolve vários detetives, pode haver a supervisão por parte de um detetive de patente "Sénior" ou superior.

(Data: 20/03/2024 11:20 | Fonte: Francisco Santos | Analista: Rodrigo Ferreira)

RD11. Os detetives devem investigar casos relativos à sua área de especialização.

(Data: 21/03/2024 10:49 | Fonte: Diogo Oliveira | Analista: Gonçalo Alves)

RD12. O detetive estagiário apenas deve estar associado a casos onde esteja pelo menos um detetive sénior envolvido. O detetive estagiário não tem qualquer contacto com qualquer suspeito ou cliente.

(Data: 18/03/2024 10:51 | Fonte: Diogo Oliveira | Analista: Gonçalo Alves)

RD13. O detetive investigador apenas deve estar associado a um caso onde esteja pelo menos um detetive sénior associado. O detetive investigador tem contacto com os clientes e com os suspeitos.

(Data: 19/03/2024 10:53 | Fonte: Diogo Oliveira | Analista: Gonçalo Alves)

RD14. O detetive sénior pode estar associado a mais que um caso. É responsável por gerir detetives de patente inferior e são o contacto principal com os suspeitos e com o cliente.

(Data: 21/03/2024 10:56 | Fonte: Diogo Oliveira | Analista: Gonçalo Alves)

RD15. O detetive chefe está responsável pela supervisão de todos os detetives.

(Data: 19/03/2024 10:59 | Fonte: Diogo Oliveira | Analista: Gonçalo Alves)

RD16. Apenas um Detetive de patente "Sénior" (ou superior) tem autoridade para designar casos a outros detetives.

(Data: 21/03/2024 11:05 | Fonte: Diogo Oliveira | Analista: Gonçalo Alves)

RD17. A data de fecho do caso deve ser igual ou posterior à data de início do mesmo.

(Data: 21/03/2024 11:30 | Fonte: Francisco Santos | Analista: Rodrigo Ferreira)

RD18. Apenas os detetives de patente sénior ou superior supervisionam outros detetives.

(Data: 20/03/2024 11:15 | Fonte: Diogo Oliveira | Analista: Gonçalo Alves)

RD19. O salário de cada detetive deve corresponder à sua patente.

(Data: 17/03/2024 11:17 | Fonte: Diogo Oliveira | Analista: Gonçalo Alves)

RD20. Cada suspeito tem associado um identificador único, um nome, histórico, descrição e morada (com rua, nº porta e código postal).

(Data: 20/03/2024 16:00 | Fonte: Mariana Costa | Analista: João Sá)

RD21. As agências têm vários detetives. E cada detetive está associado a uma e uma só agência.

(Data: 17/03/2024 11:20 | Fonte: Inês Rodrigues | Analista: Diogo Esteves)

RD22. A agência tem vários polos espalhados pelo país.

(Data: 17/03/2024 11:30 | Fonte: Inês Rodrigues | Analista: Diogo Esteves)

2.2.2 Requisitos de Manipulação

RM1. Os clientes devem ser inseridos no sistema (caso ainda não constem no mesmo) quando o caso que apresentam é aceite.

(Data: 17/03/2024 18:32 | Fonte: Luís Pereira | Analista: João Cunha)

RM2. Deve ser possível, a cada momento, ter acesso a uma lista ordenada pelo nome, tanto dos clientes como dos suspeitos.

(Data: 17/03/2024 18:55 | Fonte: Luís Pereira | Analista: João Cunha)

RM3. Os casos devem estar todos catalogados no sistema.

(Data: 17/03/2024 19:25 | Fonte: Francisco Santos | Analista: João Cunha)

RM4. A cada momento deve ser possível mudar a descrição de um caso que não esteja encerrado.

(Data: 17/03/2024 20:01 | Fonte: Francisco Santos | Analista: Diogo Esteves)

RM5. Deve ser possível atualizar o histórico de um suspeito.

(Data: 17/03/2024 11:10 | Fonte: Mariana Costa | Analista: Gonçalo Alves)

RM6. Deve ser possível aceder a uma lista de detetives, ordenada pela sua classificação, de uma determinada especialização.

(Data: 18/03/2024 11:36 | Fonte: Diogo Oliveira | Analista: Rodrigo Ferreira)

RM7. Deve-se conseguir saber quais os detetives de uma determinada especialização, que estão disponíveis (sem casos atribuídos) no momento.

(Data: 18/03/2024 11:55 | Fonte: Diogo Oliveira | Analista: João Cunha)

RM8. Deve ser possível saber o Nº de casos atribuídos e resolvidos por detetive.

(Data: 18/03/2024 12:03 | Fonte: Diogo Oliveira | Analista: João Sá)

RM9. Quando um novo detetive é contratado, deve entrar no sistema e deve ser alterado o Nº de detetives da agência.

(Data: 18/03/2024 12:38 | Fonte: Diogo Oliveira | Analista: João Sá)

RM10. A cada momento deve ser possível aceder e alterar as evidências relativas a um caso aberto.

(Data: 18/03/2024 12:44 | Fonte: Francisco Santos | Analista: João Cunha)

RM11. Deve ser sempre possível aceder a uma lista dos casos, tanto encerrados como abertos em todos os momentos.

(Data: 19/03/2024 15:22 | Fonte: Diogo Oliveira | Analista: Gonçalo Alves)

RM12. Todos os detetives devem ser catalogados no sistema, sendo necessário registrar o seu N° interno, nome, especialização, contacto, salário e patente.
(Data: 19/03/2024 15:43 | Fonte: Diogo Oliveira | Analista: Rodrigo Ferreira)

RM13. A classificação dos detetives deve ser atualizada anualmente.
(Data: 19/03/2024 16:19 | Fonte: Diogo Oliveira | Analista: Rodrigo Ferreira)

RM14. Todos os suspeitos devem estar catalogados no sistema, sendo necessário registrar o nome, morada, histórico e descrição.
(Data: 19/03/2024 18:54 | Fonte: Mariana Costa | Analista: João Cunha)

RM15. Deve ser possível alterar o status de um caso, caso seja necessário a reabertura do mesmo.
(Data: 20/03/2024 20:03 | Fonte: Francisco Santos | Analista: João Sá)

RM16. Deve ser possível ver qual o detetive, ou os vários detetives, atribuídos a cada caso.
(Data: 20/03/2024 20:45 | Fonte: Francisco Santos | Analista: João Cunha)

2.2.3 Requisitos de Controlo

RC1. A base de dados está acessível 24 horas por dia, para cada detetive.
(Data: 18/03/2024 09:00 | Fonte: Inês Rodrigues | Analista: João Sá)

RC2. A criação de cada caso é exclusiva a detetives (de patente "Investigador" ou superior) da Agência.
(Data: 18/03/2024 11:30 | Fonte: Francisco Santos | Analista: João Cunha)

RC3. O acesso a cada caso é exclusivo a Detetives (de qualquer patente) da Agência.
(Data: 19/03/2024 10:15 | Fonte: Francisco Santos | Analista: João Cunha)

RC4. Os detetives de patente "Estagiário" apenas podem consultar dados de casos nos quais estão envolvidos, enquanto os detetives de patentes superiores

possuem uma visão mais ampla desses dados.

(Data: 19/03/2024 14:02 | Fonte: Francisco Santos | Analista: João Cunha)

RC5. Nenhum cliente deve conseguir aceder a dados de outros clientes.

(Data: 20/03/2024 08:45 | Fonte: Luís Pereira | Analista: João Sá)

RC6. Nenhum cliente deve ter acesso a dados sobre casos.

(Data: 20/03/2024 13:20 | Fonte: Mariana Costa | Analista: Rodrigo Ferreira)

RC7. Apenas um Detetive de patente "Chefe" tem autoridade para modificar a classificação e a patente de outros detetives.

(Data: 20/03/2024 14:00 | Fonte: Diogo Oliveira | Analista: Gonçalo Alves)

RC8. Apenas um Detetive de patente "Chefe" tem autoridade para alterar o Status de um caso.

(Data: 21/03/2024 16:37 | Fonte: Francisco Santos | Analista: Diogo Esteves)

RC9. Detetives de patente "Chefe" devem fazer um backup dos dados, semestralmente.

(Data: 21/03/2024 16:09 | Fonte: Diogo Oliveira | Analista: Diogo Esteves)

2.3. Análise e Validação Geral dos Requisitos

Após o levantamento dos requisitos, foi essencial realizar uma análise abrangente dos mesmos. Com esse propósito, foram realizadas reuniões entre a nossa equipa e com representantes da Agência com o intuito de discutir sua viabilidade. No decorrer das reuniões, chegamos a um consenso sobre os detalhes e requisitos que merecem maior atenção e desenvolvimento, bem como aqueles que podem ser desvalorizados.

Um dos detalhes que ambas as partes acharam crucial, foi que a hierarquia dos detetives deveria ser um requisito fundamental para a estrutura da BD a ser implementada. Esta decisão foi crucial para garantir uma dinâmica na BD que reflete fielmente o funcionamento padrão das agências.

Após várias reuniões, foram ajustadas diferentes interpretações que tínhamos e chegamos a um consenso sobre os nossos requisitos. Com as visões, de ambos os lados, alinhadas, a BD a desenvolver estava com os seus alicerces solidificados.

3. Modelação Concetual

3.1 Apresentação da Abordagem de Modelação Realizada

De forma a obter uma noção concetual da BD, a desenvolver, foi necessária a criação de um modelo concetual. Este processo iniciou-se após a recolha da informação e da formulação dos requisitos necessários.

Para agilizar o processo de desenvolvimento da BD, foi utilizado um diagrama ER baseado na notação *Chen*. A escolha do diagrama ER e da notação *Chen* derivam da sua simplicidade de representação e transmissão da informação relativa à organização e armazenamento dos dados. O diagrama ER permite ilustrar os relacionamentos entre as diferentes entidades (pessoas, objetos, conceitos, etc), tal como transmitir de uma forma clara e sucinta os atributos das mesmas. A sua construção é simples, uma vez que o primeiro passo é identificar as entidades do problema, de seguida os seus relacionamentos e por fim os atributos, quer seja das entidades ou dos relacionamentos.

Desta forma, a modelação concetual que apresentaremos deve representar um modelo de estrutura de informação relativo à nossa agência de detetives, representando de forma estrutural o relacionamento entre as diferentes entidades relativas à agência, bem como os seus atributos.

3.2 Identificação e Caracterização das Entidades

Após a avaliação do funcionamento das agências de detetives, consideramos que as entidades que melhor se adequam às agências são: agência, cliente, caso, detetive e suspeito.

Para cumprir os requisitos apresentados no capítulo anterior, as entidades possuem os seguintes atributos que são dados elementares que caracterizam as mesmas:

- **Agência:** entidade que representa as agências e todos os dados a elas associados, com o objetivo de permitir ao cliente a apresentação do caso que pretende ver investigado. Esta é caracterizada pelo seu identificador (ID), número de detetives, email, polo, telefone(s) (este é um atributo multivalorado uma vez que a agência pode ter vários telefones) e morada (constituída por rua, porta e código postal, sendo por isso um atributo composto). Esta entidade originou-se a partir do requisito RD1.
- **Cliente:** representa o indivíduo que apresenta o caso que pretende ver investigado. É identificado pelo seu ID e é caracterizado pelo seu nome, contribuinte, email, telefone, data de nascimento e género. É importante o acesso rápido a informações de um cliente de modo a otimizar o contacto entre a agência e o mesmo, aumentando assim a eficiência na resolução do caso. Esta entidade originou-se a partir do requisito RD5.
- **Detetive:** representa os detetives associados à agência e que investigam os casos. Este é identificado pelo seu número interno (Nr) e caracterizado pelo seu nome, salário, especialização, patente, telefone(s) (multivalorado devido à possibilidade da existência de um ou mais telefones) e classificação (constituída por número de casos resolvidos, número de casos atribuídos). É importante a utilização de uma entidade do tipo Detetive para facilitar o processo de atribuição de casos pois as informações, como patente e especialização são cruciais para o mesmo. Esta entidade originou-se a partir do requisito RD9.
- **Caso:** representa o caso que é apresentado pelo cliente e investigado pelos detetives. É identificado pelo seu número de registo (Nº registo) e é caracterizado pela sua descrição, área, status, data de abertura e fecho (opcional) e evidência(s) (atributo multivalorado devido à possibilidade de haver uma ou mais evidências). É importante armazenar os dados relativos aos casos numa entidade pois um detetive deve conseguir de forma rápida e

eficaz aceder a um caso (e aos seus dados) durante a sua investigação. Esta entidade originou-se a partir do requisito RD6.

- **Suspeito:** representa o individuo/organização que será investigado no caso. É identificado pelo seu identificador (ID) e é caracterizado pelo seu nome, descrição, histórico e morada (constituída por rua, porta e código postal, sendo assim um atributo composto). É importante armazenar os dados relativos a um suspeito numa entidade pois um detetive deve conseguir de forma rápida e eficaz aceder aos seus dados durante a investigação. Esta entidade originou-se a partir do requisito RD20.

3.3 Identificação e Caracterização dos Relacionamentos

• Relacionamento Agência-Caso

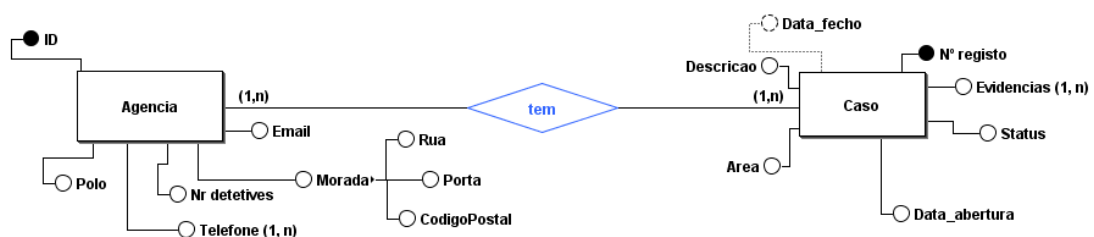


Figura 2- Relacionamento Agência-Caso

Relacionamento: Agências têm casos.

Descrição: Com o objetivo de associar vários casos às agências, é necessário armazenar esta informação para que assim que um caso seja apresentado por um cliente, seja possível aceder aos dados do caso a ser criado. Este relacionamento originou-se a partir do Requisito RD2.

Cardinalidade: Agência (1, n) – Caso (1, n).

Às agências são apresentados um ou mais casos por parte do cliente, visto que as agência pode lidar com vários casos em simultâneo.

Cada caso é investigado por uma ou mais agência.

- **Relacionamento Agência-Cliente**

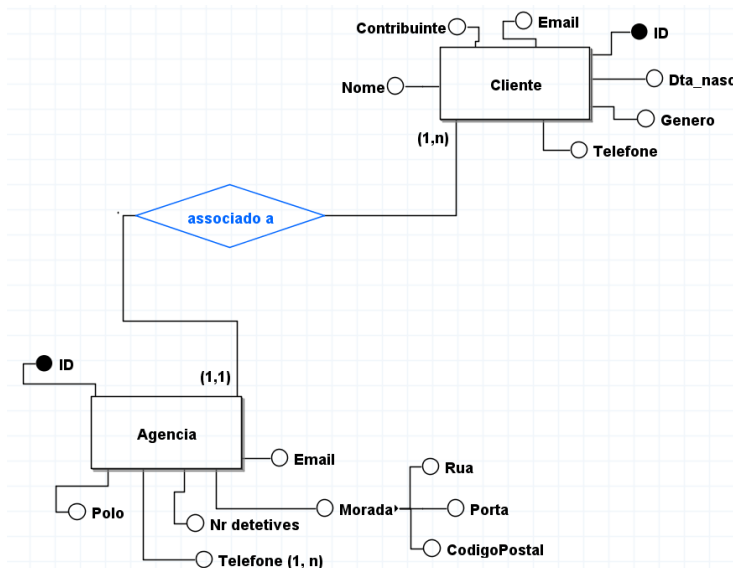


Figura 3 - Relacionamento Cliente-Agência

Relacionamento: Cliente associado a agência.

Descrição: Assim que um cliente apresenta um caso à agência, os seus dados são armazenados para que, na eventualidade da apresentação de um novo caso, a agência tenha acesso à ficha do cliente. Este relacionamento originou-se a partir do Requisito RD3.

Cardinalidade: Cliente (1, n) - Agência (1, 1).

Cada cliente está associado exclusivamente a uma agência, não podendo estar associado a entidades externas à agência.

À agência estão associados um ou mais cliente uma vez que esta pode lidar com múltiplos clientes.

- **Relacionamento Cliente-Caso**

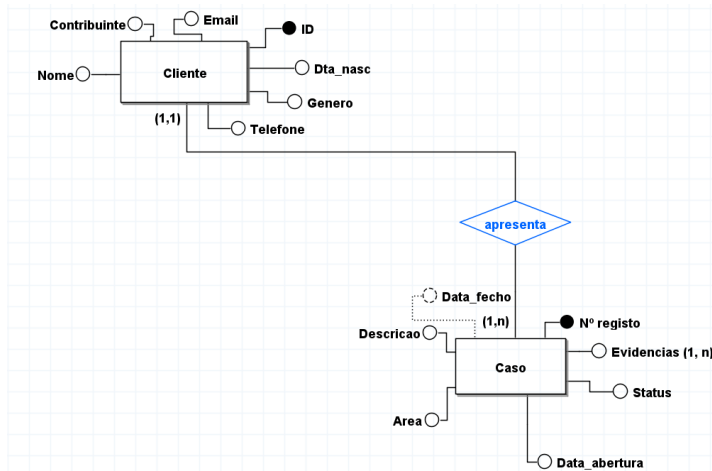


Figura 4 - Relacionamento Cliente-Caso

Relacionamento: cliente apresenta casos.

Descrição: Os clientes apresentam à agência um ou mais casos. Para isso é necessário armazenar a informação associada ao caso apresentado pelo cliente para que, no futuro, acedendo à ficha do cliente, a BD consiga listar os casos que este apresentou. Este requisito originou-se a partir do Requisito RD3.

Cardinalidade: Cliente (1, 1) – Caso (1, n).

Um cliente apresenta à agência um ou mais casos a serem investigados.

Cada caso é apresentado por um único cliente, não podendo ser apresentado por múltiplos clientes.

- **Relacionamento Caso-Suspeito**

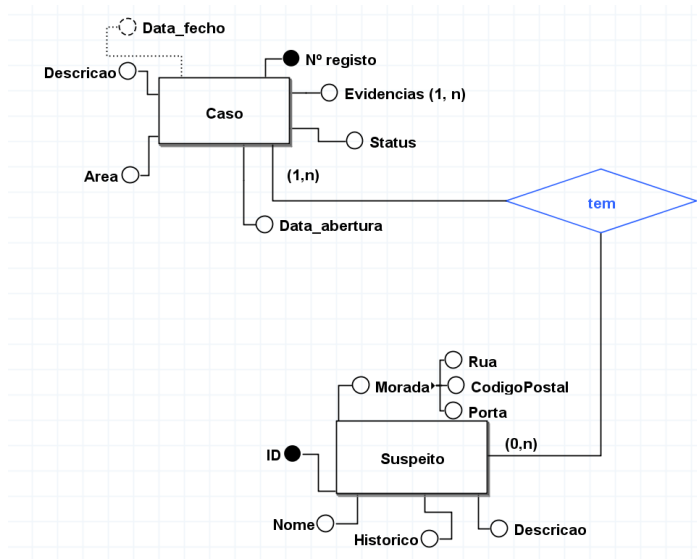


Figura 5 - Relacionamento Caso-Suspeito

Relacionamento: caso tem um suspeito.

Descrição: A cada caso está associado um suspeito a investigar. É importante armazenar as informações relativas ao suspeito uma vez que a BD deve ser capaz de listar os dados relativos ao suspeito aquando do acesso ao caso.

Este relacionamento originou-se a partir do Requisito RD7.

Cardinalidade: Caso (1, n) – Suspeito (0, n).

A cada caso podem estar, ou não, um ou mais suspeito associados e cada suspeito pode estar associado a um ou mais casos.

- **Relacionamento Agencia-Detetive**

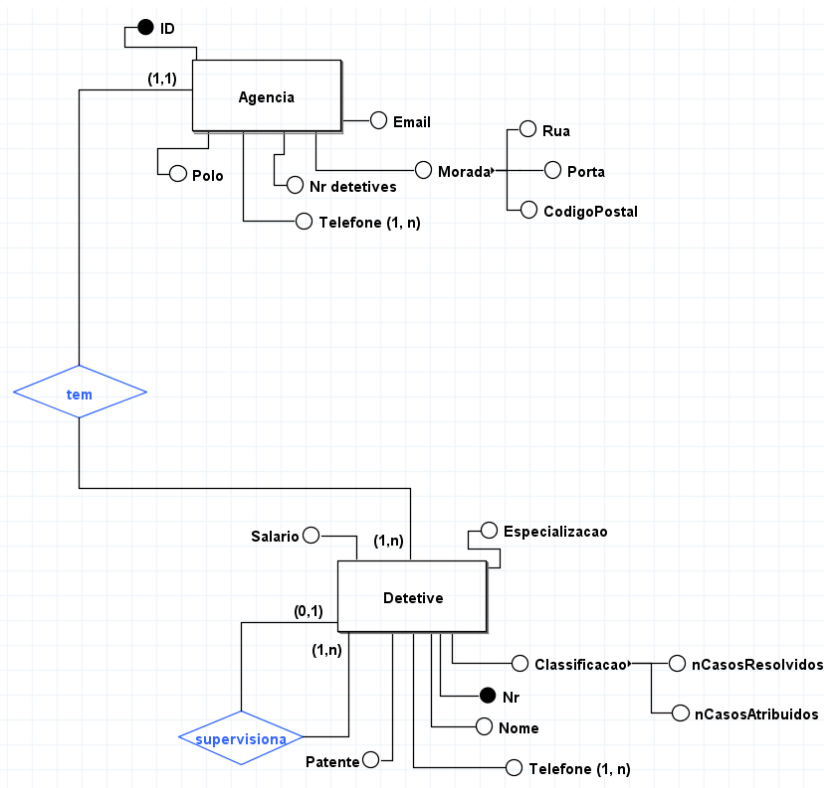


Figura 6 - Relacionamento Agência-Detetives

Relacionamento: Agência tem detetives.

Descrição: À agência estão associados diversos detetives. Por este motivo, é necessário armazenar os dados relativos aos detetives para que esta possa listar as informações relativas aos detetives. Este relacionamento originou-se a partir do Requisito RD21.

Cardinalidade: Agência (1, 1) – Detetive (1, n)

À agência estão associados um ou mais detetives.

A cada detetive está associada uma e uma só agência, não podendo os detetives estar associados a mais do que uma agência.

- **Relacionamento Detetive-Detetive**

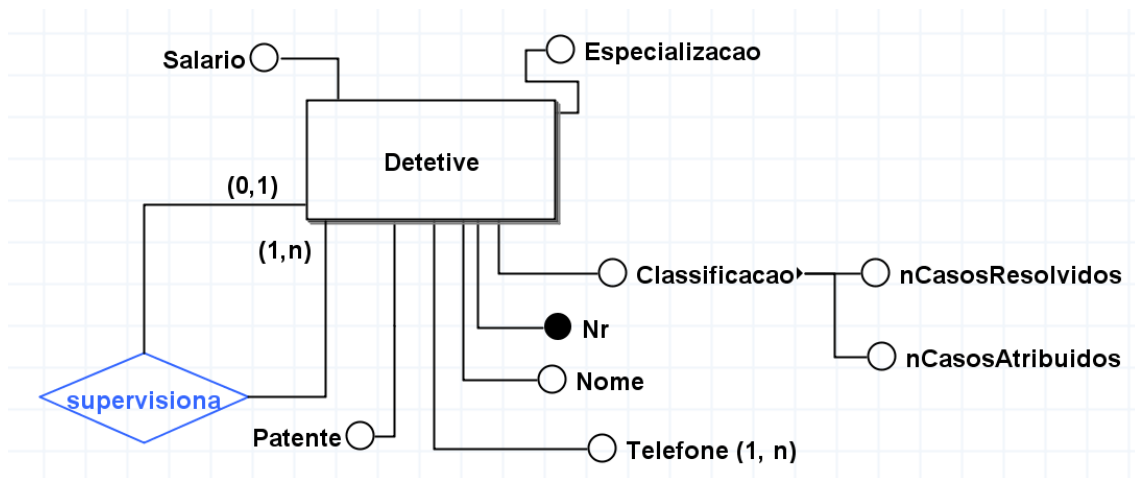


Figura 7- Relacionamento Detetive- Detetive

Relacionamento: Detetive supervisiona detetive

Descrição: Cada detetive, dependendo da sua patente, supervisiona um ou mais detetives. Por este motivo é necessário guardar a informação relativa ao relacionamento unário entre detetives. Este relacionamento originou-se a partir do Requisito RD10.

Cardinalidade: Detetive (1,1) – Detetive (1, n)

Um detetive pode supervisionar um ou mais detetives de patente inferior ao mesmo.

Um detetive de patente inferior a outro, só pode ser supervisionado por um e um só detetive.

- **Relacionamento Detetive-caso**

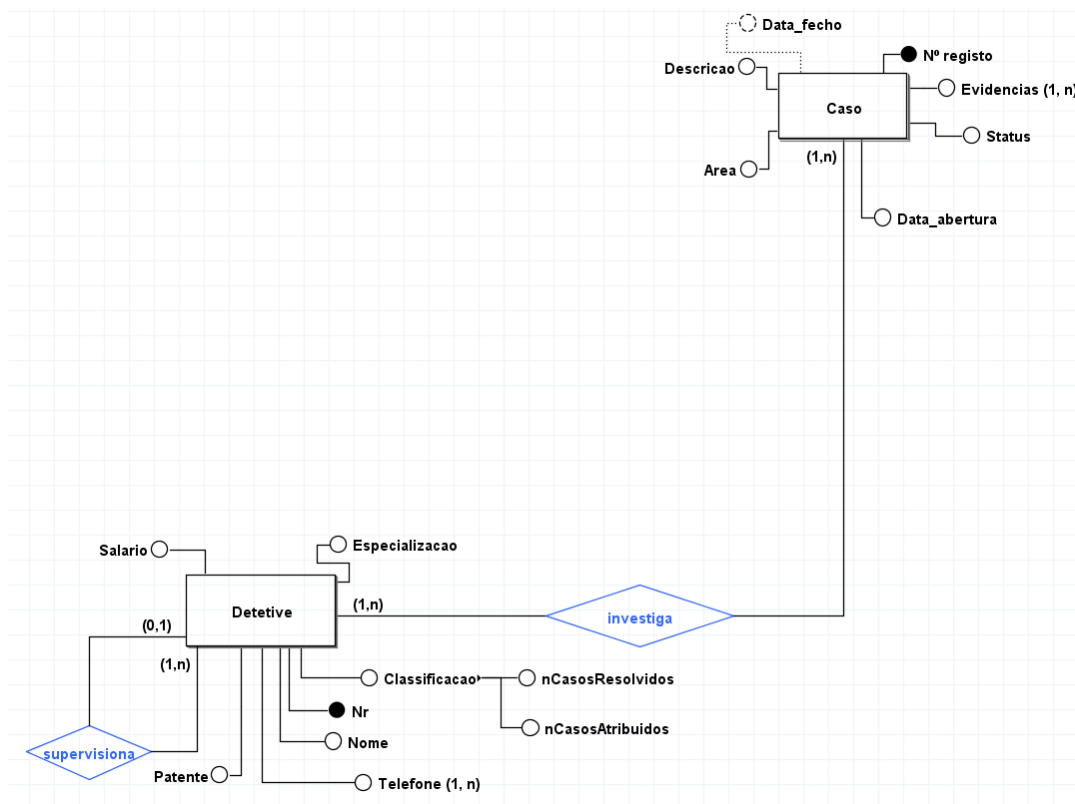


Figura 8 - Relacionamento Detetive - Caso

Relacionamento: Detetive investiga caso

Descrição: Um detetive investiga um caso. Para isso é necessário armazenar a informação relativa ao relacionamento entre o detetive e o caso para que este consiga aceder aos dados do caso de forma rápida e eficaz.

Cardinalidade: Detetive (1, n) – Caso (1, n)

Um ou mais detetives podem investigar um ou mais casos.

Um ou mais casos podem ser investigados por um detetive.

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

ENTIDADE: Agência

Atributos	Tipo de dados	Nulo	Composto	Multivalorado	Derivado	Candidato
ID	INT	Não	Não	Não	Não	Sim
Telefone	INT	Não	Não	Sim	Não	Não
Email	VARCHAR (60)	Não	Não	Não	Não	Não
Rua	VARCHAR (100)	Não	Sim	Não	Não	Não
Código Postal	VARCHAR (20)	Não	Sim	Não	Não	Não
Porta	INT	Não	Sim	Não	Não	Não

Tabela 1 - Caracterização dos atributos da agência

O atributo ID é comumente usado como uma chave primária na tabela da Agência. Fornece uma identificação única para a agência na base de dados, facilitando a recuperação e manipulação de dados relacionados. Identifica de forma legível e significativa para os clientes. O telefone e o email são meios de contato essenciais para as agências. O telefone permite que os clientes e detetives comuniquem-se diretamente com as agências, enquanto o email fornece uma forma de comunicação escrita. Ambos os atributos são fundamentais para facilitar a comunicação e o atendimento ao cliente de forma eficiente. A rua, o código postal e a porta descrevem o endereço físico da Agência. A rua indica o nome da rua ou avenida onde está localizada, o código postal identifica a área geográfica específica e a porta representa o número do prédio ou unidade onde a agência está situada. Juntos, esses atributos fornecem informações precisas sobre o endereço das agências, facilitando a navegação e a localização para os clientes e para outros detetives.

ENTIDADE: Detetive

Atributos	Tipo de dados	Nulo	Composto	Multivalorado	Derivado	Candidato
Nr	INT	Não	Não	Não	Não	Sim
Nome	VARCHAR (100)	Não	Não	Não	Não	Não
Especialização	VARCHAR (30)	Não	Não	Não	Não	Não
Salário	INT	Não	Não	Não	Não	Não
Telefone	INT	Não	Não	Sim	Não	Não
Nº casos atribuídos	INT	Não	Sim	Não	Não	Não
Nº casos resolvidos	INT	Não	Sim	Não	Não	Não
Patente	VARCHAR (50)	Não	Não	Não	Não	Não

Tabela 2 - Caracterização dos atributos do detetive

O atributo Nr é um identificado único atribuído a cada detetive na Agência. Ele serve como chave primária na tabela dos detetives, garantindo a identificação única de cada registro. O atributo Nome representa o nome completo do detetive. Este atributo é importante para identificar e distinguir os detetives dentro da agência. Ele fornece uma referência clara e legível para cada detetive na base de dados.

O atributo especialização indica a área de especialização do detetive. Pode incluir habilidades específicas de investigação, como investigação criminal, investigação corporativa, análise forense, entre outros. A especialização ajuda a atribuir os detetives aos casos mais adequados com base em suas habilidades e experiência.

O atributo Salário representa a remuneração mensal do detetive. Ele reflete o valor monetário que o detetive recebe pela sua contribuição e trabalha na agência. O salário é importante para fins de contabilidade, folha de pagamento e gestão financeira da agência.

O atributo multivalorado Telefone é o número de telefone de contato do detetive. Ele fornece um meio de comunicação direta entre o detetive e a agência, bem como entre o detetive e outras partes envolvidas nos casos em que está trabalhando.

O nº de casos atribuídos e casos resolvidos representam o número de casos atribuídos e resolvidos pelo detetive, respectivamente. Eles fornecem uma medida quantitativa da

carga de trabalho e do desempenho do detetive na resolução de casos. São informações úteis para avaliar a eficiência e a produtividade do detetive nas agências.

ENTIDADE: Cliente

Atributos	Tipo de dados	Nulo	Composto	Multivalorado	Derivado	Candidato
ID	INT	Não	Não	Não	Não	Sim
Nome	VARCHAR (100)	Não	Não	Não	Não	Não
Telefone	INT	Não	Não	Não	Não	Não
Contribuinte	INT	Não	Não	Não	Não	Sim
Email	VARCHAR (60)	Não	Não	Não	Não	Não
Data Nascimento	DATE	Não	Não	Não	Não	Não
Gênero	CHAR (1)	Não	Não	Não	Não	Não

Tabela 3 - Caracterização dos atributos do cliente

O atributo ID é novamente usado como chave primária na tabela de clientes. Fornece uma identificação única para cada cliente na base de dados, garantindo a integridade dos dados e facilitando a recuperação e manipulação de informações relacionadas a cada cliente.

O atributo Nome representa o nome completo do cliente, garantindo uma identificação clara e legível do cliente dentro do sistema, facilitando a comunicação e o reconhecimento dos clientes pela agência.

O atributo Telefone é o nº de telefone de contato do cliente. É um meio de comunicação direta entre a agência e o cliente, permitindo a coordenação de investigações e o acompanhamento de casos.

O atributo candidato a chave primária Contribuinte que representa o número de contribuinte fiscal do cliente. Embora seja único para cada cliente, o contribuinte pode não ser ideal como chave primária devido à possibilidade de mudanças no número de contribuinte ao longo do tempo ou à falta de garantia da sua unicidade em todos os casos.

O atributo Email é o endereço de email do cliente. É um meio adicional de comunicação e contato entre a agência e o cliente, sendo útil para o envio de documentos, atualizações e informações relevantes sobre investigações em andamento.

O atributo Data de Nascimento armazena a data de nascimento do cliente. É importante para verificar a idade do cliente, determinar elegibilidade para determinados serviços ou consentimento legal, e personalizar abordagens de comunicação e atendimento ao cliente com base na faixa etária.

O atributo Gênero indica o gênero do cliente, como masculino ou feminino. É útil para personalizar a comunicação e os serviços oferecidos pela agência com base nas preferências individuais dos clientes.

ENTIDADE: Caso

Atributos	Tipo de dados	Nulo	Composto	Multivalorado	Derivado	Candidato
Nº Registo	INT	Não	Não	Não	Não	Sim
Descrição	TEXT	Não	Não	Não	Não	Não
Área	VARCHAR (70)	Não	Não	Não	Não	Não
Status	VARCHAR (50)	Não	Não	Não	Não	Não
Data abertura	DATETIME	Não	Não	Não	Não	Não
Evidências	TEXT	Não	Não	Sim	Não	Não
Data Fecho	DATETIME	Sim	Não	Não	Não	Não

Tabela 4 - Caracterização dos atributos do caso

O atributo “nº registo” é uma identificação única atribuída a cada caso nas agências fénix. Serve como chave primária na tabela de casos, garantindo a identificação única de cada registo e facilitando a recuperação e manipulação de informações relacionadas ao caso.

O atributo “Descrição” fornece uma narrativa ou resumo detalhado do caso sob investigação. Descreve os eventos, circunstâncias e detalhes relevantes relacionados ao caso, ajudando os detetives e outros envolvidos a entender a natureza do trabalho a ser realizado.

O atributo “Área” indica a área de especialização ou tipo de caso ao qual pertence. Pode incluir categorias como investigação criminal, fraudes corporativas, investigação de infidelidade, entre outros. A área do caso é útil para atribuir casos aos detetives mais adequados.

O atributo “Status” indica o estado atual do caso, que pode ser “Aberto”, “Em Andamento” ou “Fechado”. Ele fornece uma indicação clara do progresso e do estágio de resolução do caso, permitindo que os detetives e gerentes acompanhem e gerenciem o fluxo de trabalho de investigação de forma eficaz.

O atributo “Data de abertura” armazena a data em que o caso foi aberto na Agência Fénix. É importante para armazenar o início da investigação e acompanhar o tempo decorrido desde o início do caso. A data de abertura pode ser usada para estabelecer prazos e prioridades de investigação.

O atributo “Evidências” representa os elementos de prova ou informações relevantes coletadas durante a investigação do caso. Isso pode incluir informações textuais sobre testemunhos, fotos, vídeos, entre outros. As evidências são fundamentais para apoiar as conclusões da investigação e contribuir para a resolução bem-sucedida do caso.

O atributo “Data de Fecho” armazena a data do fim da investigação.

ENTIDADE: Suspeito

Atributos	Tipo de dados	Nulo	Composto	Multivalorado	Derivado	Candidato
ID	INT	Não	Não	Não	Não	Sim
Nome	VARCHAR (100)	Não	Não	Não	Não	Não
Descrição	TEXT	Não	Não	Não	Não	Não
Histórico	TEXT	Não	Não	Não	Não	Não
Rua	VARCHAR (100)	Não	Sim	Não	Não	Não
Código Postal	VARCHAR (20)	Não	Sim	Não	Não	Não
Porta	INT	Não	Sim	Não	Não	Não

Tabela 5 - Caracterização dos atributos do suspeito

O atributo ID é uma identificação única atribuída a cada suspeito. Ele serve como chave primária, garantindo a identificação única de cada registo e facilitando a recuperação e manipulação de informações relacionadas ao suspeito.

O atributo “Nome” representa o nome completo do suspeito. É uma identificação clara e legível do suspeito dentro do sistema, facilitando o reconhecimento do suspeito durante a investigação.

O atributo “Descrição” fornece uma descrição detalhada do suspeito, incluindo características físicas, comportamentais ou outras informações relevantes que possam ajudar na identificação e investigação do suspeito. A descrição pode incluir detalhes como altura, peso, cor dos olhos, tatuagens, entre outros.

O atributo “Histórico” guarda informações sobre o histórico do suspeito, como antecedentes criminais, registos de prisões anteriores, ou outras atividades suspeitas. Essas informações podem ser úteis para avaliar o perfil e o comportamento do suspeito durante a investigação.

A rua, o código postal e a porta descrevem o endereço físico da Agência. A rua indica o nome da rua ou avenida onde está localizada, o código postal identifica a área geográfica específica e a porta representa o número do prédio ou unidade onde a agência está situada. Juntos, esses atributos fornecem informações precisas sobre o endereço das agências Fénix, facilitando a navegação e a localização para os clientes e para outros detetives.

3.5 Apresentação e Explicação do Diagrama ER Produzido

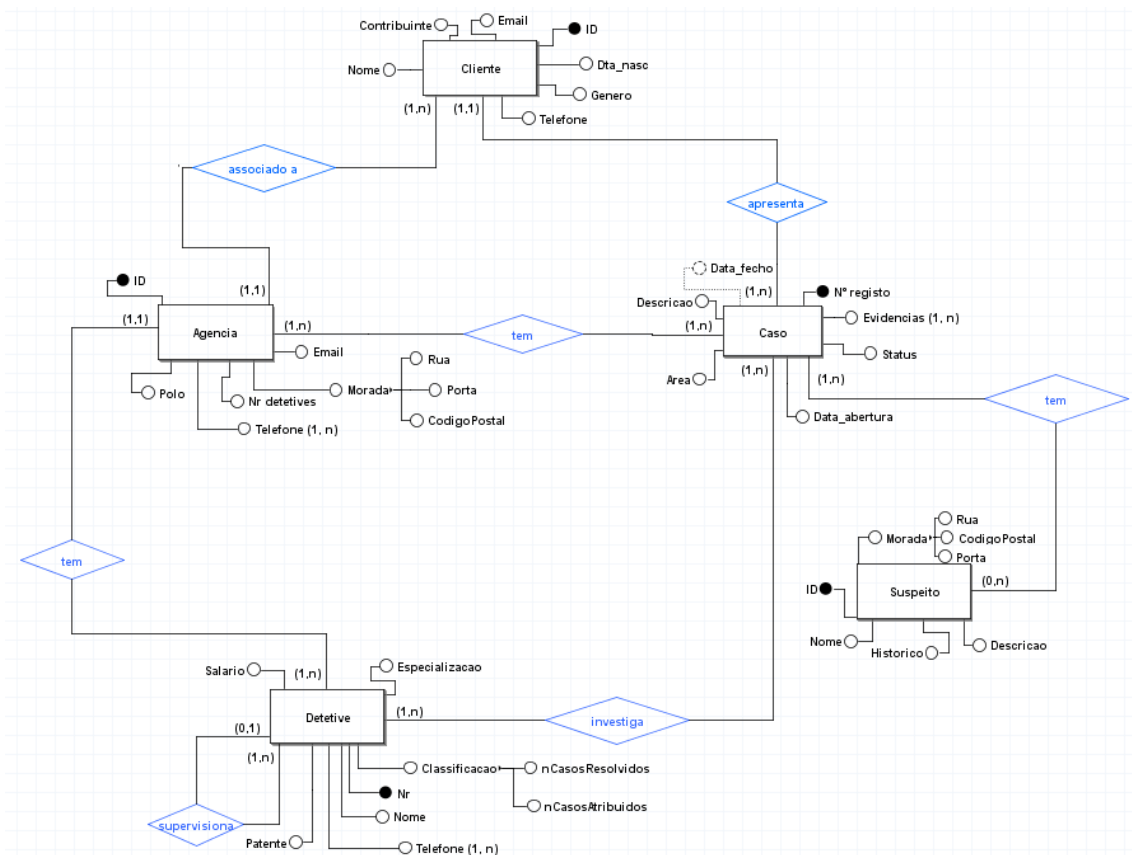


Figura 9 - Modelo Conceitual

O diagrama conceitual foi desenvolvido para representar a estrutura de dados da Agência Fénix, uma agência de detetives, destacando as entidades envolvidas, seus atributos e as relações entre elas. As entidades fundamentais incluem “Agência”, “Detetive”, “Caso”, “Cliente” e “Suspeito”, com atributos específicos que descrevem cada uma detalhadamente. No processo de construção, a identificação das identidades começou com a definição das principais entidades que representam os conceitos-chave da agência: detetives que investigam, casos a serem resolvidos, clientes que contratam a agência e suspeitos que estão relacionados aos casos. Sobre a definição dos atributos, para cada entidade identificada, foram definidos atributos específicos que fornecem uma descrição detalhada. Por exemplo, a entidade “Detetive” inclui atributos como número, nome, especialização, salário, entre outros. A escolha desses atributos visou capturar informações essenciais para a operação da agência.

Relacionamentos significativos entre as entidades foram estabelecidos para refletir as interações dentro do sistema da agência. Por exemplo, um “Caso” está associado a um “Cliente” que o reporta e a um ou mais detetives responsáveis pela investigação. Além disso, casos podem ter um ou mais suspeitos associados.

Cada relacionamento foi caracterizado por cardinalidades que indicam quantidades relativas de instâncias entre entidades relacionadas. Atributos foram detalhadamente definidos para cada entidade, como “ID” sendo a chave primária para a entidade “Agência”, designando um identificador único.

A agência serve como o ponto central do diagrama, mas, neste caso específico, tem um papel mais estrutural do que operacional, devido à singularidade da entidade dentro do contexto do sistema. Já o detetive relaciona-se com os casos através de um relacionamento de atribuição, refletindo a responsabilidade sobre investigações específicas. O caso forma o núcleo das operações de investigação, conectando-se a clientes, detetives e suspeitos, indicando quem reportou o caso, quem o investiga e potenciais envolvidos. Os clientes são a fonte dos casos, e suspeitos são os alvos das investigações. Ambos são essenciais para a narrativa de cada caso.

O modelo concetual foi meticulosamente projetado para capturar a essência das operações da Agências Fénix, enfatizando a importância de cada entidade e relacionamento dentro do sistema. Através da sua construção, foi possível estabelecer uma base sólida para o desenvolvimento subsequente da base de dados, garantindo que todas as informações relevantes sejam capturadas e relacionadas de maneira lógica e eficiente.

4. Modelação Lógica

4.1. Construção e Validação do Modelo de Dados Lógico

O modelo lógico foi construído a partir do modelo conceptual. Aqui as entidades passam a tabelas com os seus atributos simples associados. O mesmo acontece com os relacionamentos que possuem requisitos associados e com os atributos multivalorados. O identificador de uma entidade passa a designar-se por *Primary Key*. ou *Foreign Key* caso o identificador de uma

entidade esteja presente noutra sob a forma de tabela. Após a conversão foram obtidas doze tabelas, das quais duas foram originadas a partir de relacionamentos, três por atributos valorados e uma para evitar redundâncias. Seguem-se essas tabelas:

1. Agencia:

1.1. Primary Key: ID: INT.

1.2. Atributos: Designação: VARCHAR (45), Nr Detetives: INT, Rua: VARCHAR(100),
Codigo Postal: VARCHAR(20), Porta: INT, Email: VARCHAR(60), Polo VARCHAR(2);

2. Cliente:

2.1. Primary Key: ID: INT;

2.2. Atributos: Nome: VARCHAR(200), Contribuinte: VARCHAR(9), Data_nascimento:
DATETIME, Genero: CHAR(1);

2.3. Foreign Key: Agencia_ID: INT;

3. Detetive:

3.1. Primary Key: NR: INT;

3.2. Atributos: Nome: VARCHAR(100), Salario: INT, Especializacao: VARCHAR(30),
nCasosResolvidos: INT, nCasosAtribuidos: INT;

3.3. Foreign Key: Supervisor_ID: INT, Area_ID: INT, Agencia_ID: INT, Patente_ID: INT;

4. Suspeito:

4.1. Primary Key: ID: INT;

4.2. Atributos: Nome: VARCHAR(200), Historico: TEXT, Descricao: TEXT, Rua:
VARCHAR(100), Codigo Postal: VARCHAR(20), Porta: VARCHAR(10);

4.3. Foreign Key: Não tem;

5. Caso:

5.1. Primary Key: ID: INT;

5.2. Atributos: Descricao: TEXT, Area: VARCHAR(70), Data_abertura: DATETIME, Status:
VARCHAR(50);

5.3. Foreign Key: Cliente_id: INT, Agencia_id: INT, Area_ID: INT, Case_status_ID: INT;

Relativamente aos **atributos multivalorados**:

6. TelefoneDetetive (Tabela de atributo multivalorado):

6.1. Primary Key: ID: INT;

6.2. Multivalor: Telefone: INT;

6.3. Foreign Key: Não tem;

7. TelefoneAgencia (Tabela de atributo multivalorado):

7.1. Primary Key: ID: INT;

7.2. Multivalor: Telefone: INT;

7.3. Foreign Key: Não tem;

Relativamente aos **relacionamentos**:

8. Tem_associados:

8.1. Primary Key: Não tem;

8.2. Atributos: Não tem;

8.3. Foreign Key: Caso_nregisto: INT, Suspeito_ID: INT;

9. DetetiveSupervisiona:

9.1. Primary Key: Supervisor: INT;

9.2. Atributos: Não tem;

9.3. Foreign Key: Supervisor: INT, Detetive: INT;

10. investiga:

10.1. Primary Key: Não tem;

10.2. Atributos: Não tem;

10.3. Foreign Key: Detetive_NR: INT, Caso_Nº: INT.

4.2. Apresentação e Explicação do Modelo Lógico

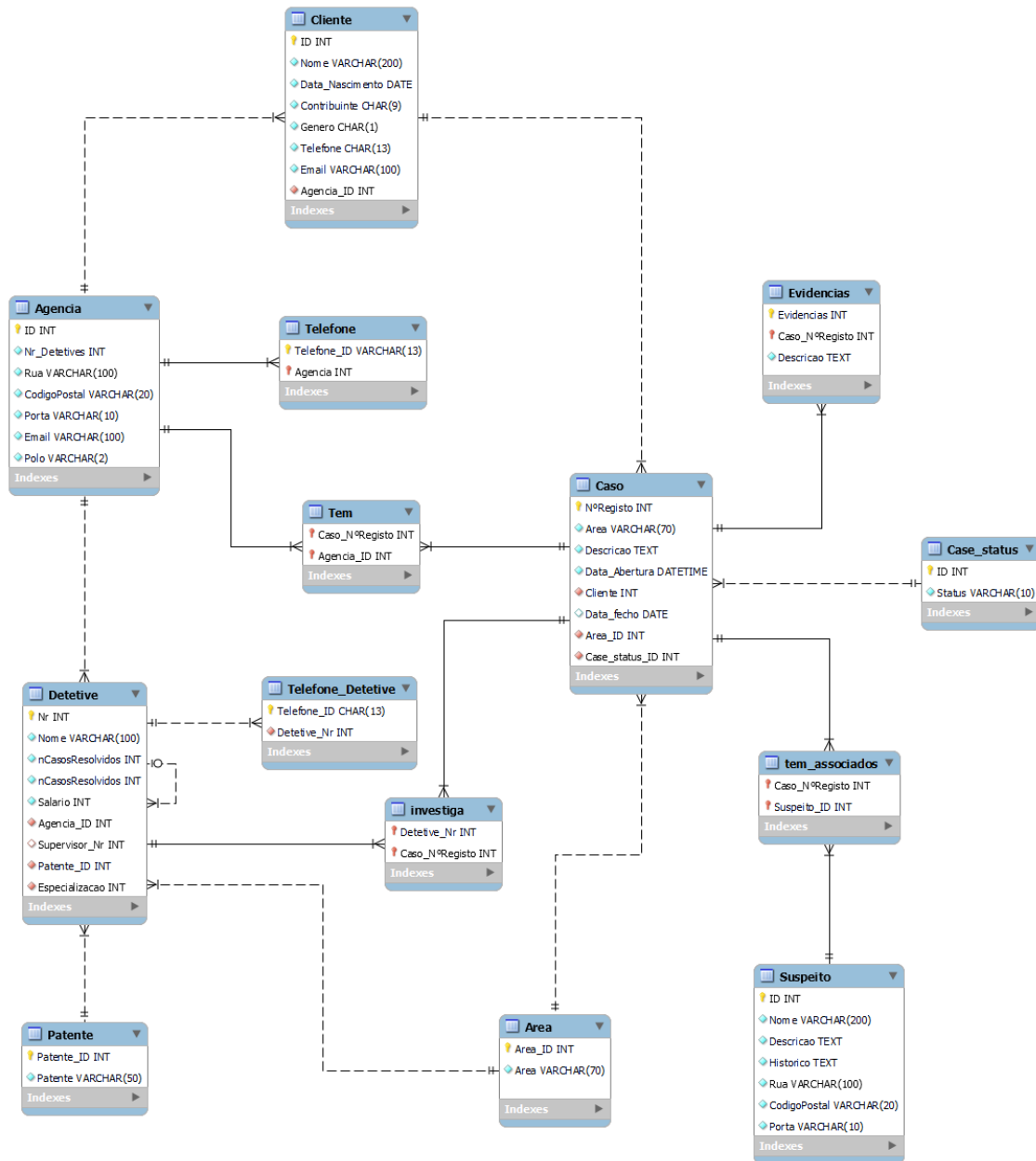


Figura 10 - Modelo Lógico

Este Modelo Lógico foi concebido tendo em conta as regras de conversão que conhecemos, tais como a regra das entidades, dos atributos multivalorados, dos atributos compostos e dos relacionamentos. As mesmas resultaram nas tabelas “Agencia”, “Cliente”, “Detetive”, “Suspeito” e “Caso”, que surgem através das entidades do mesmo nome, contendo os seus atributos, nas tabelas “TelefoneAgencia”, “TelefoneDetetive” e “Evidencias”, que surgem através dos atributos multivalorados, “Telefone(1,n)” da tabela “Agencia”, “Telefone(1,n)” da tabela “Detetive” e “Evidencias(1,n)” da tabela “Caso” e ainda nas tabelas “investiga” e “tem” resultado dos relacionamento binários N:M “investiga” e “tem”, respetivamente.

Realizamos a conversão de atributos compostos como a morada, dividindo respetivamente os seus atributos constituintes (Rua, Porta, Código Postal) e ainda a criação de tabelas “extra”, chamada “Patente”, “Case_status”, “Area” que foram criadas para eliminar possíveis redundâncias nos dados das patentes os detetives (assim evita-se que sejam introduzidos nomes de patentes diferentes para uma mesma patente como, por exemplo, detetive estagiário e estagiário), facilitando o processamento desses mesmos dados e evitando erros de listagem dos detetives por patente.

4.3. Normalização de Dados

O objetivo da normalização de dados da BD é fazer o isolamento dos dados de forma a que eventuais operações de inserção, modificação ou remoção realizadas sobre um dado atributo sejam realizadas numa única tabela e depois propagadas pelo resto da base de dados seguindo os relacionamentos estabelecidos.

Para se fazer a avaliação da normalização do nosso modelo temos de verificar se o mesmo satisfaz as três primeiras formas normais.

A primeira forma normal (1FN) diz que todos os atributos de uma tabela devem ser atômicos, ou seja, as tabelas não devem conter atributos repetidos nem multivalorados. Verificando as tabelas do nosso modelo lógico, verificamos que não existem atributos repetidos nem multivalorados nas mesmas, verificando-se assim a primeira forma normal.

A segunda forma normal (2FN) só pode ser cumprida se a primeira forma normal for satisfeita. Esta diz-nos que os atributos normais devem depender apenas da chave primária da tabela. Olhando para o nosso modelo lógico, e tomando como exemplo a tabela da agência, verificamos que os atributos não-primos (designação, número de detetives, email, rua, código postal e porta) estão dependentes apenas do ID da agência uma vez que sem o mesmo, estes deixam de ter um significado. Esta lógica aplica-se para o restante modelo, pelo que a segunda forma normal é verificada pelo modelo.

Por último, falta verificar a terceira forma normal (3FN). Esta exige que se as duas primeiras formas normais forem verificadas, os atributos que não sejam chaves

primárias devem ser mutuamente independentes, não havendo assim dependências funcionais transitivas. No caso do nosso modelo, não existem atributos que possam ser obtidos a partir de um outro atributo, pelo que se verifica a terceira forma normal.

A partir da segunda forma normal, evitam-se anomalias que podem ocorrer em processos de atualização que envolvam dados redundantes e alguns efeitos colaterais que possam resultar da modificação de uma relação. Assim a segunda forma normal permite aumentar a eficiência no armazenamento de dados e reduzir a repetição dos mesmos.

No que toca à terceira forma normal, como esta é verificada pelo nosso modelo, é possível evitar que dados que não são “desejados” não sejam incluídos nas tabelas.

Com a verificação das três primeiras formas normais, é possível afirmar que o modelo é válido em termos de normalização.

4.4. Validação do Modelo com Interrogações do Utilizador

a. Projetar o Nº de Casos Atribuídos e Resolvidos de cada Detetive

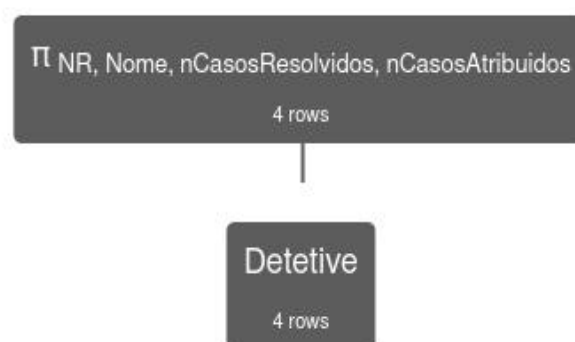


Figura 11 - Árvore correspondente à 1ª interrogação

A operação de projeção é aplicada à tabela Detetive, seleccionando apenas as colunas NR, Nome, nCasosResolvidos e nCasosAtribuidos. Isso significa que estamos interessados apenas nesses atributos específicos de cada detetive. Essa expressão retornará uma tabela que contém apenas as colunas NR, Nome,

nCasosResolvidos e nCasosAtribuidos da tabela Detetive, mostrando informações como número do detetive, seu nome, número de casos resolvidos e número de casos atribuídos. Esta expressão valida o requisito RM8

b. Projetar todos os Detetives por ordem decrescente de Ratio



Figura 12 - Árvore correspondente à 2ª interrogação

Inicialmente, a operação de projeção é aplicada à tabela Detetive, selecionando apenas a coluna Ratio. Isso significa que estamos interessados apenas nos valores de Ratio de cada detetive. Em seguida, os resultados da projeção são ordenados em ordem decrescente com base nos valores de Ratio. O desc indica a ordenação descendente. Finalmente, o operador de projeção ordenada (τ) é usado para apresentar o resultado, que consiste nos valores de Ratio dos detetives da tabela Detetive, organizados em ordem decrescente. Esta expressão valida o requisito RM6.

c. Projetar qual é o detetive que está a investigar cada caso

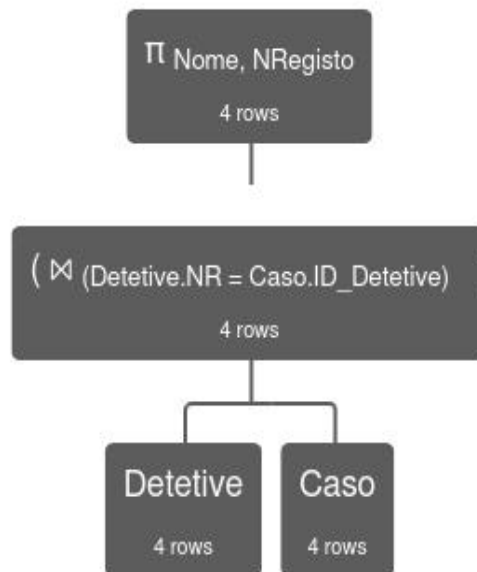


Figura 13 - Árvore correspondente à 3ª interrogação

Esta operação combina informações das tabelas Detetive e Caso com base em uma condição específica. Neste caso, a junção é feita onde o número do detetive (NR na tabela Detetive) é igual ao ID do detetive associado ao caso (ID_Detetive na tabela Caso). Após a junção, a operação de projeção seleciona apenas as colunas de interesse do resultado. Aqui, são selecionados os nomes dos detetives (Nome da tabela Detetive) e os números de registro dos casos (NRegisto da tabela Caso). Esta expressão valida o requisito RM17.

d. Projetar todos os Clientes por ordem crescente do Nome



Figura 14 - Árvore correspondente à 4ª interrogação

Inicialmente, a operação de projeção é aplicada à tabela Cliente, selecionando apenas a coluna Nome. Isso significa que estamos interessados apenas nos nomes dos clientes. Em seguida, os resultados da projeção são ordenados em ordem crescente com base nos nomes dos clientes. O asc indica a ordenação ascendente. Finalmente, o operador de projeção ordenada (τ) é usado para apresentar o resultado final, que consiste nos nomes dos clientes da tabela Cliente, organizados em ordem alfabética crescente. Esta expressão valida o requisito RM2.

5. Implementação Física

5.1. Seleção do sistema de gestão de bases de dados

De forma a realizar a implementação física da nossa base de dados decidimos usar o MySQL como DBMS pois foi este que nos foi introduzido na UC. De forma auxiliar, utilizamos o MySQL Workbench para o desenvolvimento da mesma, pois foi o programa que utilizamos ao longo das aulas (teóricas e práticas). No início deste processo, encontramos outras possibilidades como MariaDB, ferramenta que decidimos utilizar num futuro próximo para expandir o nosso conhecimento nesta área.

5.2. Apresentação e explicação da base de dados implementada

A base de dados implementada resulta de uma implementação do esquema lógico desenvolvido. As tabelas da base de dados foram implementadas recorrendo à definição das mesmas no modelo lógico. As tabelas possuem colunas que derivam dos atributos do esquema lógico. Assim como no modelo lógico, as tabelas tem Primary Keys que podem ser simples (contém apenas uma coluna) ou compostas (contém mais do que uma coluna) e também podem conter Foreign Keys que são usadas para relacionar 2 tabelas.

Tabela Agência

```
CREATE TABLE IF NOT EXISTS Agencia(  
    ID INT AUTO_INCREMENT NOT NULL,  
    Nr_Detetives INT NOT NULL,  
    Rua VARCHAR(100) NOT NULL,  
    CodPostal VARCHAR(20) NOT NULL,  
    Porta VARCHAR(10) NOT NULL,  
    Email VARCHAR(100) NOT NULL,  
    Polo VARCHAR(2),  
    PRIMARY KEY(ID)  
);
```

Tabela Suspeito

```
) CREATE TABLE IF NOT EXISTS Suspeito(  
    ID INT AUTO_INCREMENT,  
    Nome VARCHAR(200) NOT NULL,  
    Historico TEXT NOT NULL,  
    Descricao TEXT NOT NULL,  
    Rua VARCHAR(100) NOT NULL,  
    Codigo_Postal VARCHAR(20) NOT NULL,  
    Porta VARCHAR(10) NOT NULL,  
    PRIMARY KEY(ID)  
- );
```

Tabela Case_Status

```
CREATE TABLE IF NOT EXISTS Case_status (  
    ID INT AUTO_INCREMENT NOT NULL,  
    Status VARCHAR(10) NOT NULL,  
    PRIMARY KEY (ID)  
);
```

Tabela Area

```

) CREATE TABLE IF NOT EXISTS Area(
    Area_ID INT AUTO_INCREMENT NOT NULL
    Area VARCHAR(70) NOT NULL,
    PRIMARY KEY(Area_ID)
);

```

Tabela Patente

```

CREATE TABLE IF NOT EXISTS Patente(
    idPatente INT AUTO_INCREMENT,
    Designacao VARCHAR(50) NOT NULL,
    PRIMARY KEY(idPatente)
);

```

Tabela Cliente

```

CREATE TABLE IF NOT EXISTS Cliente(
    ID INT AUTO_INCREMENT,
    Nome VARCHAR(200) NOT NULL,
    Contribuinte VARCHAR(9) NOT NULL,
    Data_nascimento DATE NOT NULL,
    Genero CHAR(1) NOT NULL,
    Telefone CHAR(13) NOT NULL,
    Email VARCHAR(100) NOT NULL,
    Agencia_ID INT NOT NULL,
    PRIMARY KEY(ID),
    FOREIGN KEY (Agencia_ID) REFERENCES Agencia (ID)
);

```

Tabela Caso

```

) CREATE TABLE IF NOT EXISTS Caso (
    N_Registo INT AUTO_INCREMENT,
    Descricao TEXT NOT NULL,
    Case_Status INT NOT NULL,
    Data_abertura DATETIME NOT NULL,
    Data_fecho DATETIME,
    Cliente_ID INT NOT NULL,
    Area_ID INT NOT NULL,
    PRIMARY KEY(N_Registo),
    FOREIGN KEY (Cliente_ID) REFERENCES Cliente (ID),
    FOREIGN KEY (Area_ID) REFERENCES Area (Area_ID),
    FOREIGN KEY (Case_Status) REFERENCES Case_status (I
);

```

Tabela Evidencias

```

) CREATE TABLE IF NOT EXISTS Evidencias(
    Evidencias_ID INT AUTO_INCREMENT,
    Evidencias TEXT NOT NULL,
    Caso_Nr_Registo INT NOT NULL,
    PRIMARY KEY(Evidencias_ID),
    FOREIGN KEY (Caso_Nr_Registo) REFERENCES Caso (N_Registo)
);

```

Tabela Detetive

```
CREATE TABLE IF NOT EXISTS Detetive(  
    NR INT AUTO_INCREMENT NOT NULL,  
    Nome VARCHAR(100) NOT NULL,  
    Salario INT NOT NULL,  
    Especializacao INT NOT NULL,  
    nCasosResolvidos INT NOT NULL,  
    nCasosAtribuidos INT NOT NULL,  
    Supervisor_ID INT,  
    Agencia_ID INT NOT NULL,  
    Patente_ID INT NOT NULL,  
    PRIMARY KEY(NR),  
    FOREIGN KEY (Patente_ID) REFERENCES Patente (idPatente),  
    FOREIGN KEY (Supervisor_ID) REFERENCES Detetive (NR),  
    FOREIGN KEY (Agencia_ID) REFERENCES Agencia (ID),  
    FOREIGN KEY (Especializacao) REFERENCES Area (Area_ID)  
);
```

Tabela Telefone

```
CREATE TABLE IF NOT EXISTS Telefone(  
    Agencia INT NOT NULL,  
    Telefone_ID char(13) NOT NULL,  
    PRIMARY KEY(Agencia,Telefone_ID),  
    FOREIGN KEY (Agencia) REFERENCES Agencia(ID)  
);
```

Tabela TelefoneDetetive

```
CREATE TABLE IF NOT EXISTS TelefoneDetetive(  
    NR_detetive INT NOT NULL,  
    Telefone char(13) NOT NULL,  
    PRIMARY KEY(Telefone,NR_detetive),  
    FOREIGN KEY (NR_detetive) REFERENCES Detetive(NR)  
);
```

Tabela Tem

```
CREATE TABLE IF NOT EXISTS Tem (  
    Caso_Nr_Registo INT NOT NULL,  
    Agencia_ID INT NOT NULL,  
    PRIMARY KEY (Caso_Nr_Registo, Agencia_ID),  
    FOREIGN KEY (Caso_Nr_Registo) REFERENCES Caso (N_Registo),  
    FOREIGN KEY (Agencia_ID) REFERENCES Agencia (ID)  
);
```

Tabela Investiga

```

CREATE TABLE IF NOT EXISTS Investiga(
    Detetive_Nr INT NOT NULL,
    Caso_Nr_Registo INT,
    PRIMARY KEY (Detetive_Nr,Caso_Nr_Registo),
    FOREIGN KEY (Detetive_Nr) REFERENCES Detetive (NR),
    FOREIGN KEY (Caso_Nr_Registo) REFERENCES Caso (N_Registo)
);

```

Tabela Tem_associados

```

CREATE TABLE IF NOT EXISTS Tem_associados(
    Caso_Nr_Registo INT NOT NULL,
    Suspeito_ID INT NOT NULL,
    PRIMARY KEY (Caso_Nr_Registo,Suspeito_ID),
    FOREIGN KEY (Caso_Nr_Registo) REFERENCES Caso (N_Registo),
    FOREIGN KEY (Suspeito_ID) REFERENCES Suspeito (ID)
);

```

5.3. Criação de utilizadores da base de dados

No que toca a utilizadores, decidimos criar um para elemento do grupo, com todos os privilégios. Relativamente a casos “reais”, criámos o utilizador do Chefe com as devidas permissões (Pode consultar o conteúdo das tabelas, inserir (manualmente ou através de ficheiro), alterar, remover e exportar tabelas como ficheiro.

Criámos um utilizador estagiário que, tal como pretendido, apenas pode consultar dados das tabelas.

```

CREATE USER 'joaosa'@'localhost' IDENTIFIED BY 'admin';
GRANT ALL PRIVILEGES ON *.* TO 'joaosa24'@'localhost';

CREATE USER 'joaocunha'@'localhost' IDENTIFIED BY 'admin';
GRANT ALL PRIVILEGES ON *.* TO 'joaocunha'@'localhost';

CREATE USER 'goncalo'@'localhost' IDENTIFIED BY 'admin';
GRANT ALL PRIVILEGES ON *.* TO 'goncalo'@'localhost';

CREATE USER 'diogo'@'localhost' IDENTIFIED BY 'admin';
GRANT ALL PRIVILEGES ON *.* TO 'diogo'@'localhost';

CREATE USER 'rodrigo'@'localhost' IDENTIFIED BY 'admin';
GRANT ALL PRIVILEGES ON *.* TO 'rodrigo'@'localhost';

CREATE USER 'marianaRibeiro'@'%' IDENTIFIED BY 'chiefFenix';
GRANT SELECT,INSERT,UPDATE,DELETE,FILE, GRANT OPTION ON *.* TO 'marianaRibeiro'@'%';

CREATE USER 'lucasFernandes'@'%' IDENTIFIED BY 'lucasFernandesIntern';
GRANT SELECT ON Fenix.Caso TO 'lucasFernandes'@'%';

FLUSH PRIVILEGES;

```


5.4. Povoamento da base de dados

O povoamento das tabelas SQL presentes na base de dados foi realizado de duas formas: a partir de um script SQL que insere diretamente os dados nas tabelas da base de dados e a partir de um programa em C que acede à base de dados implementada e insere na mesma os dados, utilizando como base um ficheiro 'csv' com os mesmos.

5.4.1 Inserção a partir do script SQL

No que toca ao script SQL utilizado, foram usadas instruções INSERT referenciadas às tabelas correspondentes que se pretende povoar. Um ponto importante a ter em conta é o povoamento das diversas tabelas pela ordem correta. É necessário ter em conta a existência de chaves estrangeiras nas diversas tabelas, uma vez que é preciso povoar primeiro as tabelas que a chave estrangeira referencia. Só depois de o povoamento das tabelas referenciadas pela chave estrangeira estar concluído é que é possível o povoamento da tabela que contém a chave, caso contrário o SQL emitirá uma mensagem de erro a alertar para este mesmo caso.

Após estas considerações, a ordem pelo qual povoamos as tabelas da nossa base de dados foi a seguinte: Agencia > Telefone > Cliente > Area > Case_status > Caso > Evidencias > Patente > Detetive > TelefoneDetetive > Tem > Investiga > Suspeito > Tem_associados. Apesar da existência de chaves estrangeiras em algumas das tabelas, a ordem escolhida poderia variar, desde que fosse respeitada a condição apresentada acima.

O povoamento da tabela Agencia foi feito da seguinte forma:

```
INSERT INTO Agencia (Nr_Detetives, Rua, CodPostal, Porta, Email, Polo) VALUES
(10, 'Rua do Alpinista', '4730-302', '13', 'p1.fenix@fenix.pt', 'P1'),
(10, 'Rua Alto do Monte Largo', '4750-32', '56', 'p2.fenix@fenix.pt', 'P2'),
(10, 'Rua da Universidade', '4780-12', '43', 'p3.fenix@fenix.pt', 'P3');
```

A instrução INSERT preenche os campos da tabela Agencia, com exceção da *primary key*, que é uma chave primária auto-incrementável.

Quanto ao povoamento da tabela Telefone, esta é a instrução INSERT utilizada:

```
INSERT INTO Telefone(Telefone_ID, Agencia) VALUES
('+351912245234',1),
('+351912345234',2),
('+351912342344',3);
```

O povoamento da tabela Telefone é realizado de forma ligeiramente diferente da tabela Agencia, uma vez que esta tabela contém uma chave primária composta por dois atributos: o ID da tabela Agencia e o valor do telefone.

Para que não haja problemas neste povoamento, é necessário que a tabela Agencia esteja criada à *posteriori*. Assim que esta esteja criada, é povoada a tabela com o ID de uma das tabelas Agencia criada e já povoada e com o número de telefone propriamente dito.

De seguida, passamos ao povoamento da tabela Cliente:

```
INSERT INTO Cliente (Nome, Contribuinte, Data_nascimento, Telefone ,Genero, Email, Agencia_ID) VALUES
('João Cunha', 123456789, "2004-06-11", '+351987654321', "M", "joaoCunha@gmail.com",1),
('Gonçalo Silva', 123456789, "2004-06-11", '+351987623456', "M", "goncalosilva@gmail.com",1),
('Diogo Esteves', 123456789, "2004-10-11", '+351923456781', "M", "diogoesteves@gmail.com",2),
('Rodrigo Granja', 123456789, "2004-06-11", '+351912345677', "M", "rodrigogranja@gmail.com",2),
('João Sá', 212132456, "2004-06-11", '+351912653433', "M", "joaosa@gmail.com",3);
```

O povoamento desta tabela não tem qualquer restrição relativa ao povoamento de tabelas posteriores uma vez que esta tabela não contém nenhuma chave estrangeira. Por isso, estamos à vontade para fazer o seu povoamento com os valores escolhidos.

```
INSERT INTO Area (Area) VALUES
('Fraude'),
('Crime Organizado'),
('Furto'),
('Terrorismo'),
('CiberTerrorismo'),
('Vandalismo'),
('Roubo'),
('Homicídios'),
('Agressão'),
('CiberCrime');

INSERT INTO Case_status (Status) VALUES
('aberto'),
('fechado'),
('arquivado');
```

À semelhança da tabela Cliente e Agencia, as tabelas Area e Case_status também não implicam nenhuma dependência relativa a uma chave estrangeira. Assim, o povoamento das mesmas foi feito atribuindo apenas os valores necessários.

O povoamento da tabela Caso, além de ter de ter sido feito após o povoamento das tabelas Cliente, Case_status e Area, teve também que ser feito em duas fases distintas: uma para os casos fechados e outra para os casos que estejam arquivados ou abertos, devido à presença do atributo opcional Data_fecho.

```
INSERT INTO Caso (Descricao, Case_Status, Data_abertura,Data_fecho ,Cliente_ID, Area_ID) VALUES
('Ameaça de bomba na baixa de Lisboa',2,"2022-10-11","2022-10-15",3,4);

INSERT INTO Caso (Descricao, Case_Status, Data_abertura, Cliente_ID, Area_ID) VALUES
('Corpo encontrado no lixo',1,"2024-05-28",1,8),
('Roubo de um banco online',1,"2023-12-20",2,10),
('Roubo de veículo encontrado no Porto',3,"2024-04-30",4,7);
```

O povoamento da tabela de Evidências segue a mesma lógica da tabela Telefone, sendo necessário referenciar as tabelas relativas à chave estrangeira que referencia a tabela Caso.

```
INSERT INTO Evidencias (Evidencias, Caso_Nr_Registo)
VALUES
('Relatório fotográfico detalhado dos danos na cena do crime, incluindo registo de danos materiais e pessoais'),
('Vídeo de vigilância capturado em vários pontos de um posto de gasolina próximo ao local do crime'),
('Depoimentos detalhados de testemunhas oculares, incluindo não apenas o relato da vítima, mas também de vizinhos e comerciantes da área'),
('Análise detalhada das impressões digitais encontradas na arma do crime, incluindo impressões latentes e impressões de transferência'),
('Registos telefónicos pormenorizados, mostrando uma cronologia completa das chamadas e mensagens de texto enviadas e recebidas'),
('Análise completa do ADN que corresponda ao principal suspeito, incluindo amostras de saliva encontradas em objetos pessoais'),
('Cópias autenticadas de e-mails trocados entre as partes envolvidas, incluindo mensagens de texto e mensagens de aplicativos de mensagens instantâneas'),
('Extrato bancário detalhado que revela transações suspeitas, incluindo depósitos e saques em datas próximas ao crime'),
('Gravação de áudio de uma conversa incriminadora, capturando não apenas as partes envolvidas, mas também terceiros que podem ter ouvido a conversa'),
('Cópias autenticadas de documentos financeiros falsificados encontrados no veículo roubado, incluindo cheques e recibos de pagamento').
```

O povoamento da tabela Patente não tem qualquer restrição associada, bastando a atribuição de valores às diferentes colunas.

```
INSERT INTO Patente (Designacao) VALUES
    ('Estagiário'),
    ('Investigador'),
    ('Sênior'),
    ('Chefe');
```

Quanto ao povoamento da tabela de detetives, foi necessário dividir o processo de povoamento em duas fases, devido à presença de um relacionamento unário na mesma. Primeiro é necessário povoar a tabela com os detetives com patente sênior ou chefe, uma vez que estes não necessitam de supervisor. Assim basta atribuir o valor do ID da patente à chave estrangeira respetiva.

De seguida, passamos ao povoamento dos detetives das restantes patentes, uma vez que estes necessitam de um supervisor que precisa de estar anteriormente na tabela.

```
-- NAO PRECISAM DE SUPERVISOR
INSERT INTO Detetive (Nome, Salario, Especializacao, nCasosResolvidos, nCasosAtribuidos, Agencia_ID, Patente_ID)
VALUES
-- Sêniiores
('Carlos Pereira', 7000, 1, 50, 55, 1, 3),
('Ana Costa', 7200, 2, 48, 53, 1, 3),
('Fernando Rocha', 7100, 5, 52, 57, 1, 3),
-- Chefe
('Mariana Ribeiro', 10000, 4, 100, 100, 2, 4);

-- PRECISAM DE SUPERVISOR
INSERT INTO Detetive (Nome, Salario, Especializacao, nCasosResolvidos, nCasosAtribuidos, Agencia_ID, Patente_ID, Supervisor_ID)
VALUES
-- Estagiários
('João Silva', 3000, 10, 5, 10, 1, 1, 1),
('Lucas Fernandes', 3200, 3, 3, 6, 1, 1, 2),
('Gabriela Alves', 3100, 6, 4, 8, 3, 1, 3),
-- Investigadores
('Maria Souza', 5000, 8, 20, 25, 2, 1, 4),
('Carlos Mota', 5200, 7, 18, 23, 3, 2, 3),
('Juliana Lima', 5100, 9, 22, 27, 2, 2, 4);
```

O povoamento das tabelas TelefoneDetetive, Tem e Investiga segue a mesma lógica das tabelas anteriormente povoadas, sendo necessário fazer referência à tabela que é referenciada pelas respetivas chaves estrangeiras.

```

INSERT INTO TelefoneDetetive (Telefone,NR_detetive) VALUES
('+351912345678',1),
('+351912345679',2),
('+351912345680',3),
('+351912345681',4),
('+351912345682',5),
('+351912345683',6),
('+351912345684',7),
('+351912345685',8),
('+351912345686',9),
('+351912345687',10);

INSERT INTO Tem (Caso_Nr_Registo,Agencia_ID) VALUES
(1,2),
(2,3),
(4,1),
(3,1);

INSERT INTO Investiga (Detetive_Nr,Caso_Nr_Registo) VALUES
(1,3),
(2,3),
(5,4),
(6,2),
(2,4),
(5,1),
(10,1),
(8,2),
(1,4);

```

Por fim, para o povoamento das tabelas Suspeito e Tem_associados, é feito através da atribuição de valores às diferentes colunas (para a tabela Suspeito) e atribuição dos valores que a chave estrangeira referencia (para a tabela Tem_associados).

```

INSERT INTO Suspeito (Nome, Historico, Descricao, Rua, Codigo_Postal, Porta) VALUES
('Roberto Carlos','Sem antepaçados criminais','Negro, alto' , 'Rua dos chãos', '4700-234','12'),
('Carlos Roberto','Sem antepaçados criminais','Moreno, baixo', 'Rua dos chãos', '4700-234','12'),
('João Carvalho','Roubo à mão armada','Baixa estatura. Cabelo loiro' , 'Rua Fonte do mundo', '4700-233','10'),
('Gonçalo Reis','Sem antepaçados criminais','Estatura alta, de porte forte' , 'Praça da República', '4700-233','2');

INSERT INTO Tem_associados (Caso_Nr_Registo, Suspeito_ID) VALUES
(1,2),
(2,3),
(3,4),
(4,1);

```

5.4.2 Inserção a partir do script C

Através do conector incluído na biblioteca possível povoar a base de dados, fazendo-se o parse dos dados a partir de um ficheiro csv, como ilustrado na imagem abaixo:

```
D;Cristiano Ronaldo;3000;1;25;30;1;2
D;Artur Cabral;5000;3;15;20;2;1
D;Di Maria;4000;2;20;25;3;3
D;David Neres;4500;4;30;35;1;1
D;Roger Schmit;3500;5;18;28;2;2
D;Morato;3800;1;22;27;3;3
D;João Neves;3600;3;19;29;1;2

C;João Cunha;123456789;2004-06-11;+351987654321;M;joaoCunha@gmail.com;1
C;Pedro Silva;987654321;2000-03-21;+351123456789;M;pedroSilva@gmail.com;2
C;Ana Sousa;456789123;1995-12-15;+351789123456;F;anaSousa@gmail.com;3
C;Maria Ferreira;321654987;1988-07-19;+351654321987;F;mariaFerreira@gmail.com;1
C;Luis Oliveira;789123456;1999-10-05;+351321654789;M;luisOliveira@gmail.com;2
C;Carlos Mendes;654987321;2001-11-11;+351456789123;M;carlosMendes@gmail.com;3
C;Rita Pereira;147258369;1994-09-17;+351258369147;F;ritaPereira@gmail.com;1
C;José Santos;963852741;1993-05-23;+351852741963;M;joseSantos@gmail.com;2
```

Neste ficheiro, o primeiro campo identifica a tabela que se pretende povoar (D remete para detetive e C para cliente) e os restantes campos identificam os campos a povoar.

A implementação do código pode ser consultada no anexo IV.

5.5. Cálculo do espaço da base de dados (inicial e taxa de crescimento anual)

Para fazer o cálculo, utilizamos uma folha de cálculo com todas as tabelas e os devidos bytes para cada atributo.

Tabela Agencia		
Campo	Tipo de Dados	Tamanho (Bytes)
ID	INT	4
Nr Detectives	INT	4
Rua	VARCHAR(100)	102
Cod Postal	VARCHAR(20)	22
Porta	VARCHAR(10)	12
Email	VARCHAR(100)	102
Polo	VARCHAR(2)	4
Total		250

Tabela Suspeito		
Campo	Tipo de Dados	Tamanho (Bytes)
ID	INT	4
Nome	VARCHAR(200)	202
Historico	TEXT	65535
Descricao	TEXT	65535
Rua	VARCHAR(100)	102
Cod Postal	VARCHAR(20)	22
Porta	VARCHAR(10)	12
Total		131412

Tabela Caso		
Campo	Tipo de Dados	Tamanho (Bytes)
N_Registro	INT	4
Descricao	TEXT	65535
Caso_status	INT	4
Data abertura	DATE TIME	8
Data fecho	DATE TIME	8
Cliente ID	INT	4
Area ID	INT	4
Total		65567

Tabela Case Status		
Campo	Tipo de Dados	Tamanho (Bytes)
ID	INT	4
STATUS	VARCHAR(10)	12
Total		16

Tabela Area		
Campo	Tipo de Dados	Tamanho (Bytes)
Area ID	INT	4
STATUS	VARCHAR(70)	72
Total		76

Tabela Patente		
Campo	Tipo de Dados	Tamanho (Bytes)
IdPatente	INT	4
Designacao	VARCHAR(20)	52
Total		56

Tabela Cliente		
Campo	Tipo de Dados	Tamanho (Bytes)
ID	INT	4
Nome	VARCHAR(200)	202
Contribuinte	VARCHAR(9)	11
Data nasc	DATE	3
Genero	CHAR(1)	1
Telefone	CHAR(13)	13
email	VARCHAR(100)	102
agencia id	INT	4
Total		343

Tabela Evidencias		
Campo	Tipo de Dados	Tamanho (Bytes)
Evidencias ID	INT	4
Evidencias	TEXT	65535
Caso Nr Reg	INT	4
Total		65543

Tabela Detetive		
Campo	Tipo de Dados	Tamanho (Bytes)
NR	INT	4
Nome	VARCHAR(100)	102
Salario	INT	4
Especializacao	INT	4
Casos Res	INT	4
Casos Atribui	INT	4
Supervisor ID	INT	4
Agencia ID	INT	4
Patente ID	INT	4
Total		134

Tabela TelefoneDetetive		
Campo	Tipo de Dados	Tamanho (Bytes)
NR Detetive	INT	4
Telefone	CHAR(13)	13
Total		17

Tabela Investiga		
Campo	Tipo de Dados	Tamanho (Bytes)
Detetive Nr	INT	4
Caso Nr Reg	INT	4
Total		8

Tabela Tem Associados		
Campo	Tipo de Dados	Tamanho (Bytes)
Caso Nr Reg	INT	4
Suspeito ID	INT	4
Total		8

Tabela Tem		
Campo	Tipo de Dados	Tamanho (Bytes)
Caso Nr Reg	INT	4
Agencia ID	INT	4
Total		8

Com a soma de todas as tabelas, obtemos o resultado de (aproximadamente) 257, 26 kB.

Para uma previsão do crescimento da Agência decidimos adotar os seguintes valores (ao fim de 5 anos)

- 30 Polos para a Agência
- 300 Suspeitos
- 1800 Clientes
- 900 Casos
- 240 Detetives

Através destes valores, podemos fazer uma previsão para o tamanho da base de dados. Este valor rondaria os 94.56 MB.

5.6. Definição e caracterização de vistas de utilização em SQL

5.6.1 Vista de Clientes ordenados por Nome

Um dos objetivos que tínhamos com esta BD, era ter acesso, sempre que preciso, a uma lista ordenada pelo nome de todos os clientes que fazem parte da Agência, como pedido em um dos

requisitos de manipulação (RM2). A seguinte implementação em SQL permite-nos criar uma *VIEW* que cumpre esse requisito:

```
CREATE VIEW Clientes AS
SELECT *
FROM Cliente
ORDER BY Nome ASC;
```

5.6.2 Vista de Suspeitos ordenados por Nome

O nosso objetivo neste ponto é semelhante ao mencionado acima, mas ao invés de termos uma lista dos Clientes ordenada pelo nome, queremos ter uma lista dos Suspeitos ordenada pelo nome.

```
CREATE VIEW Suspeitos AS
SELECT *
FROM Suspeito
ORDER BY Nome ASC;
```

5.6.3 Vista dos Casos ordenados pelo Nr_Registo

Mais uma vez, semelhante aos outros dois pontos, desejamos ter acesso, sempre que possível, a uma lista ordenada pelo Nr_Registo dos Casos investigados pela nossa Agência

```
CREATE VIEW Casos AS
SELECT *
FROM Caso
ORDER BY N_Registo ASC
```

5.7. Tradução das interrogações do utilizador para SQL

A partir das árvores geradas na secção anterior fomos capazes de converter as interrogações do utilizador para SQL de uma forma rápida e eficaz.

5.7.1. Projetar o Nº de Casos Atribuídos e Resolvidos de cada Detetive

```
CREATE VIEW Nrcasos AS
SELECT NR, Nome, nCasosAtribuidos, nCasosResolvidos
FROM Detetive
ORDER BY NR ASC;
```

5.7.2. Projetar todos os Detetives por ordem decrescente de Ratio

```
CREATE VIEW DetetivesPorRatio AS
SELECT NR, Nome, CalculaRatio(Detetive.nCasosResolvidos, Detetive.nCasosAtribuidos) AS Ratio
FROM Detetive
ORDER BY Ratio DESC;
```

5.7.3. Projetar qual é o detetive que está a investigar cada caso

```
CREATE PROCEDURE DetetivesPorCaso(IN CasoID INT)
) BEGIN
    SELECT *
    FROM Detetive
    INNER JOIN Caso ON Detetive.ID = CasoID
    ORDER BY Detetive.ID ASC;
- END$$
```

5.7.4. Projetar todos os Clientes por ordem crescente do Nome

```
CREATE VIEW Clientes AS
SELECT *
FROM Cliente
ORDER BY Nome ASC;
```

5.8. Indexação do Sistema de Dados

Ao criar uma tabela, são criados índices para as colunas com a restrição PRIMARY KEY e UNIQUE. Estes índices são criados automaticamente e são chamados de implícitos. Porém, também é possível definir outros índices, estes serão chamados de explícitos. Existem vantagens, nomeadamente a aceleração de operações de seleção e consulta, porém também acatam algumas desvantagens, como abrandamento das operações de inserção e de atualização. Logo, devemos tentar apenas definir índices para colunas que são consultadas com frequência (e com várias entradas), mas que não sejam muito atualizadas.

Um bom exemplo para um índice é a coluna Polo da tabela Agencia. Ao procurar por entradas da tabela da Agencia, uma tabela irá conter várias entradas, logo uma possibilidade é procurar por um determinado polo da Agencia, em vez de procurar pelo ID (chave primária). Para a criação deste índice, utilizamos o seguinte comando:

```
CREATE INDEX Polo_Agencia ON Agencia(Polo);
```

Outro índice, que nos pareceu correto implementar, foi na coluna Nome da tabela Detetive, visto que existem vários detetives, pode ser útil ter a possibilidade de acessar a dados de um detetive diretamente pelo nome do mesmo. A implementação deste índice é a seguinte:

```
CREATE INDEX Nome_Detetive ON Detetive(Nome);
```

5.9. Implementação de procedimentos, funções e gatilhos

5.9.1. Função CalculaRatio

Esta função aceita dois parâmetros de entrada: nCasosResolvidos e nCasosAtribuidos.

A função `CalculaRatio` divide o número de casos resolvidos pelo número de casos atribuídos, multiplica o resultado por 100 para colocar em percentagem, e retorna esse valor como `FLOAT`. Decidimos criar esta função para podermos calcular a eficiência de um detetive na resolução de casos.

```
CREATE FUNCTION CalculaRatio(nCasosResolvidos INT, nCasosAtribuidos INT)
RETURNS FLOAT
DETERMINISTIC
BEGIN
    DECLARE Ratio FLOAT;
    SET Ratio = (nCasosResolvidos / nCasosAtribuidos) * 100;
    RETURN (Ratio);
END //
```

5.9.2. Procedure `DetetivesPorClassificação`

Neste procedure recebemos como parâmetro de entrada a identificador relativo à especialização de um detetive. Seguidamente seleciona todas as colunas do `Detetive` e usando a function acima referido (`CalculaRatio`) calcula a razão entre os casos resolvidos e os casos atribuídos. Para além disso no `'WHERE'` filtra os detetives pela especialização especificada e ordena os detetives pelo valor da razão calculada em ordem crescente.

Este procedure é útil para obter uma lista de detetives de uma determinada especialização, ordenados pela eficiência na resolução de casos.

```
CREATE PROCEDURE DetetivesPorClassificacao(IN p_Especializacao INT)
BEGIN
    SELECT Detetive.*, CalculaRatio(Detetive.nCasosResolvidos, Detetive.nCasosAtribuidos) AS Ratio
    FROM Detetive
    WHERE Detetive.Especializacao = p_Especializacao
    ORDER BY Ratio ASC;
END$$
```

5.9.3. Trigger `AtualizaNrDetetives`

O trigger `AtualizaNrDetetives` é acionado automaticamente após a inserção de uma nova linha na tabela `Detetive`. Incrementa o campo `'Nr_Detetives'` na tabela `Agencia` em 1, para refletir o aumento do número de detetives associado à agência. A atualização é aplicada apenas à agência cujo `'ID'` corresponde ao `'Agencia_ID'` da nova linha inserida na tabela `Detetive`.

Este trigger é útil para manter automaticamente o número de detetives atualizados em cada agência quando novos detetives são adicionados à base de dados.

```
CREATE TRIGGER AtualizaNrDetetives
AFTER INSERT ON Detetive
FOR EACH ROW
BEGIN
    UPDATE Agencia
    SET Nr_Detetives = Nr_Detetives + 1
    WHERE Agencia.ID = NEW.Agencia_ID;
END $$
```

6. Alterações

Na defesa da 1ª fase do projeto, através do feedback da professora, chegámos a conclusão de que existiam aspetos a melhorar, nomeadamente no modelo concetual e lógico.

6.1. Modelo Concetual

No modelo concetual, decidimos manter a entidade “Agência” adicionando um atributo “Polo”. Este atributo nasceu de uma adição na contextualização, sendo esta, a agência tinha vários polos espalhados pelo país. Outra mudança, foi na possibilidade do atributo “Data Fecho” ser nulo. No primeiro modelo, este atributo não podia ser nulo, ou seja, iria sempre existir uma data de fecho e este facto não ia de encontro com a nossa visão da utilidade da base de dados (Um detetive deve ser capaz de aceder a informação de um caso em investigação). Por fim, decidimos também retirar o atributo “Ratio” da tabela “Detetive” visto que não cumpria a normalização pois iria depender de outros atributos, que não eram a chave primária da tabela. O “Ratio” de cada detetive não foi eliminado por completo, este foi implementado recorrendo a uma função (em SQL) que calcula o mesmo.

6.2. Modelo Lógico

O modelo lógico inicial foi um dos pontos fracos da 1ª fase. Este facto tornou-se bastante evidente na defesa visto que existiam pontos a melhorar no mesmo, ficando aquém relativamente ao resto do trabalho. No entanto, decidimos realizar, quase do zero, um novo modelo lógico que representasse, de uma maneira correta, o nosso trabalho. Usando como recurso, o material disponibilizado na blackboard e o livro do Professor Orlando Belo, construímos o modelo lógico que achamos que está devidamente desenvolvido, possibilitando uma implementação mais fidedigna do que é a nossa visão relativamente à base de dados.

6.3. Requisitos

Ao fazer uma revisão do relatório inicial, reparámos que existem requisitos que não fariam muito sentido estar presentes no projeto. Por isso, decidimos fazer uma pequena “limpeza” na listagem de requisitos que tínhamos.

7. Conclusões

Inicialmente, começamos por analisar quais seriam as necessidades que o sistema teria que ser capaz de suportar (i.e. aquilo que seria importante para garantir que o sistema seria benéfico e funcional para a empresa).

Como primeiros passos, levantamos os requisitos através de várias técnicas e organizamo-los em grupos de descrição, manipulação e controlo. A partir destes, foi possível desenvolver um modelo conceptual, com um maior nível de abstracção, que revela as nossas entidades, atributos e relacionamentos.

Após o conceptual, passámos para a modelação lógica, um dos “alicerces” mais importantes do trabalho, pois serviria como base para a implementação física. Tal como referido, pelos docentes da UC, foram estes passos que nos consumiram mais tempo, pois para uma base de dados ser útil, deve estar bem construída e atender a todas as necessidades. Reconhecemos que existem algumas eventuais falhas e aspectos a melhorar (otimizar a alocação de recursos nos atributos do tipo TEXT) mas de uma perspectiva geral, achamos que foram bem concebidos.

Por fim, na implementação física, foi uma fase mais “atrativa” pois estávamos a trabalhar com algo mais “palpável” e era possível testar com vários exemplos. Decidimos realizar um número considerável de *procedures* / *views* / *functions* pois tínhamos de cumprir vários requisitos de manipulação (nomeadamente).

No quesito de trabalho futuro, reconhecemos que existem formas de otimizar o tamanho da base de dados, nomeadamente na escolha do tipo de dados para os atributos. Se a empresa tiver um crescimento exponencial, a base de dados tornar-se-ia demasiado pesada, obrigando a alocação a novos recursos como por exemplo um servidor para “alojar” a base de dados.

De uma perspectiva geral, estamos contentes com o resultado final e achamos que a nossa base de dados seria capaz de suportar o funcionamento de uma Agência de Detetives, objetivo inicial do projeto.

Bibliografia

Belo, Orlando - **Bases de Dados Relacionais**: FCA, 2021. 9789727229215.

Lista de Siglas e Acrónimos

<<Apresentar uma lista com todas as siglas e acrónimos utilizados durante a realização do trabalho. O formato base para esta lista deverá ser da forma como abaixo se apresenta.>>

BD	Base(s) de Dados
SQL	Structed Query Language
DBMS	<i>Database Management System</i>
UC	Unidade Curricular
RD	Requisito de Descrição
RM	Requisito de Manipulação
RC	Requisito de Controlo
FN	Fórmula Normal
ER	Entidade Relação

Anexos

I. Anexo 1 – Entrevista com um representante da Agência

Olá, obrigado por concordar em reunir para discutir alguns detalhes sobre a estrutura da base de dados. Tenho algumas questões sobre como devemos categorizar as especializações dos detetives na base de dados. Qual é a sua opinião sobre isso?



Diogo Oliveira Fénix

Olá, é um prazer ajudar. Na minha opinião, é fundamental incluir a especialização de cada detetive na base de dados. Isso evitará enviar apenas detetives de uma especialização para todos os casos, garantindo uma distribuição equilibrada das habilidades e experiência da nossa equipa.

Concordo totalmente. Seria prejudicial para a eficiência das vossas investigações se não considerarmos a especialização de cada detetive ao designar casos. Tens alguma sugestão sobre como podemos implementar isso de forma eficiente na base de dados?



Diogo Oliveira Fénix

Uma abordagem eficaz seria criar categorias para as especializações mais comuns entre os detetives, como investigação criminal, vigilância, investigação digital, entre outras. Em seguida, podemos associar cada detetive a uma ou mais dessas categorias, permitindo que o detetive responsável por atribuir os casos selecione detetives com as habilidades mais adequadas para cada situação.

Isso faz todo o sentido! Ao incluir a especialização de cada detetive na base de dados, poderemos garantir que os casos sejam atribuídos aos detetives mais qualificados para lidar com eles. Isso certamente aumentará a eficiência e a qualidade de nossas investigações. Muito obrigado pela sua perspicácia e sugestões valiosas



Diogo Oliveira Fénix

Estou feliz por poder contribuir. Estamos todos a trabalhar para o mesmo objetivo: garantir que a Fénix seja a mais eficiente e bem-sucedida possível. Se precisar de mais alguma coisa, estou à disposição para ajudar no que conseguir.

II. Anexo 2 – Regulamento Interno relativo à Hierarquia



Normas de Organização e Hierarquia

A hierarquia é essencial para assegurar a eficiência das operações e o sucesso das investigações. Cada patente de detetive desempenha um papel crucial dentro da organização, contribuindo com as suas habilidades e experiência para alcançar os objetivos comuns. Abaixo, apresentamos as diferentes patentes de detetives e suas responsabilidades específicas:

1. Detetive Estagiário:

- Adquire experiência sob supervisão.
- Associado a casos com detetives sêniores.
- Sem contato direto com suspeitos ou clientes.

2. Detetive Investigador:

- Conduz investigações sob supervisão.
- Associado a casos com detetives sêniores.
- Contato limitado com suspeitos e clientes.

3. Detetive Sênior:

- Lidera investigações autonomamente.
- Gere detetives de patentes inferiores.
- Principal ponto de contato com suspeitos e clientes.

4. Detetive Chefe:

- Supervisiona todas as operações.
- Autoridade para alterar classificações de detetives.
- Resolve problemas operacionais e estabelece políticas.

É crucial que todos os detetives compreendam e respeitem a hierarquia estabelecida, seguindo as diretrizes operacionais e contribuindo para um ambiente de trabalho colaborativo e produtivo. Em caso de dúvidas ou preocupações, os detetives devem recorrer aos seus superiores imediatos para orientação e esclarecimento.

Qualquer violação deste código, seja direta ou indireta, será minuciosamente investigada e medidas apropriadas serão tomadas em relação aos envolvidos

Atenciosamente,
Comitê Executivo

III. Anexo 3 – Sede da Agência



IV. Anexo 4 – Implementação do script C

```

1  int main(int argc, char** argv) {
2      MYSQL* con = mysql_init(NULL);
3
4      if (con == NULL) {
5          fprintf(stderr, "mysql_init() failed\n");
6          exit(1);
7      }
8
9      if (mysql_real_connect(con, "localhost", "Diogo", "ogoid2004",
10         "fenix", 0, NULL, 0) == NULL) {
11          finish_with_error(con);
12      }
13
14      FILE* file_fd = fopen("povoamento.csv", "r");
15      if (file_fd == NULL) {
16          perror("Erro ao abrir o arquivo");
17          exit(1);
18      }
19
20      char line[1024];
21
22      while (fgets(line, 1024, file_fd)) {
23          char* line_cpy = strdup(line);
24
25          char* recordType = getfield(line_cpy, 0);
26          if (strcmp(recordType, "D") == 0) { // povoar detetives
27              free(line_cpy); // Liberar line_cpy usado pelo getfield
28              line_cpy = strdup(line); // Duplicar novamente para evitar modificações
29
30              char* Nome = getfield(line_cpy, 1);
31              char* Salario = getfield(line_cpy, 2);
32              char* Especializacao = getfield(line_cpy, 3);
33              char* nCasosResolvidos = getfield(line_cpy, 4);
34              char* nCasosAtribuidos = getfield(line_cpy, 5);
35              char* Agencia_ID = getfield(line_cpy, 6);
36              char* Patente_ID = getfield(line_cpy, 7);
37
38              char query[1024];
39              sprintf(query, sizeof(query),
40                  "INSERT INTO Detetive (Nome, Salario, Especializacao, nCasosResolvidos, nCasosAtribuidos, Agencia_ID, Patente_ID) VALUES ('%s', %s, '%s', %s, %s, %s, %s)",
41                  Nome, Salario, Especializacao, nCasosResolvidos, nCasosAtribuidos, Agencia_ID, Patente_ID);
42
43              if (mysql_query(con, query)) {
44                  finish_with_error(con);
45              }
46
47              free(Nome);
48              free(Salario);
49              free(Especializacao);
50              free(nCasosResolvidos);
51              free(nCasosAtribuidos);
52              free(Agencia_ID);
53              free(Patente_ID);
54          } else if (strcmp(recordType, "C") == 0) { // povoar clientes
55              char* Nome = getfield(line_cpy, 1);
56              char* Contribuinte = getfield(line_cpy, 2);
57              char* Data_nascimento = getfield(line_cpy, 3);
58              char* Telefone = getfield(line_cpy, 4);
59              char* Genero = getfield(line_cpy, 5);
60              char* Email = getfield(line_cpy, 6);
61              char* Agencia_ID = getfield(line_cpy, 7);
62
63              char query[1024];
64              sprintf(query, sizeof(query),
65                  "INSERT INTO Cliente (Nome, Contribuinte, Data_nascimento, Telefone, Genero, Email, Agencia_ID) VALUES ('%s', '%s', '%s', '%s', '%s', '%s', '%s')",
66                  Nome, Contribuinte, Data_nascimento, Telefone, Genero, Email, Agencia_ID);
67
68              if (mysql_query(con, query)) {
69                  finish_with_error(con);
70              }
71
72              free(Nome);
73              free(Contribuinte);
74              free(Data_nascimento);
75              free(Telefone);
76              free(Genero);
77              free(Email);
78              free(Agencia_ID);
79          }
80          free(recordType); // Liberar recordType
81          free(line_cpy);
82      }
83
84      fclose(file_fd);
85      mysql_close(con);
86      return 0;
87  }
88

```