



Universidade do Minho
Licenciatura em Engenharia Informática

Unidade Curricular de Bases de Dados

Ano Lectivo de 2023/2024

Mundo Sóbrio

- ❖ **Diogo Fernandes, nº100746;**
- ❖ **Samuel Ferreira, nº100654;**
- ❖ **Guilherme Marques, nº94984;**
- ❖ **Bruno Campos, nº 98639.**

Data de Recepção	
Responsável	
Avaliação	
Observações	

Resumo

O presente relatório descreve o desenvolvimento de um Sistema de Bases de Dados (SBD) para a Agência de Detetives "Mundo Sóbrio", uma organização dedicada ao combate do narcotráfico e uso de substâncias ilícitas. Diante do crescente desafio imposto pela complexidade das operações criminosas, a agência enfrentou dificuldades na gestão e coordenação eficaz das suas investigações, resultando em perdas de evidências e falta de comunicação entre as equipas. Motivados por estas lacunas, os objetivos do trabalho visam fortalecer a capacidade da agência de lidar com estes desafios, promovendo uma gestão mais eficiente de casos, monitoramento em tempo real, análise avançada de dados, coleta e preservação de evidências, colaboração entre as equipas e transparência nas operações.

A análise da viabilidade do projeto destacou a recuperação de prejuízos financeiros, aumento da eficiência operacional, melhoria na tomada de decisões, fortalecimento da relação com clientes e parceiros, e redução de custos a longo prazo como benefícios esperados. Os recursos necessários incluem uma equipa multidisciplinar, hardware adequado e um sistema de gestão da base de dados. A metodologia do levantamento de requisitos adotou entrevistas estruturadas, análise documental e observação de processos para garantir uma compreensão abrangente das necessidades da agência e dos utilizadores finais.

Os requisitos foram organizados e validados, servindo como base para a modelação conceptual e lógica do sistema. A modelação conceptual envolveu a identificação de entidades, relacionamentos e atributos. Posteriormente, o modelo lógico foi construído e validado para garantir integridade e eficiência na manipulação dos dados.

Para além disso também fizemos a implementação física do mesmo em linguagem SQL de modo a poder ser armazenado e manipulado pelo sistema de base de dados.

Área de Aplicação: Desenvolvimento de Sistemas de Informação para Agências de Investigação e Segurança.

Palavras-Chave: Agência de Detetives, Sistema de Bases de Dados, Levantamento de Requisitos, Modelo Conceptual, Modelo Lógico.

Índice

1. Definição do Sistema	1
1.1. Contexto de aplicação (Contextualização)	1
1.2. Motivação e Objetivos do Trabalho (Fundamentação e Objetivos)	2
1.3. Análise da Viabilidade do processo	3
1.4. Recursos e Equipa de Trabalho	4
1.5. Plano de Execução do Projeto	5
2. Levantamento e Análise de Requisitos	6
2.1. Método de Levantamento e de Análise de Requisitos Adotado	6
2.2. Organização dos Requisitos Levantados	7
2.3. Análise e Validação Geral dos Requisitos	8
3. Modelação Conceptual	9
3.1. Apresentação da Abordagem de Modelação Realizada	9
3.2. Identificação e Caracterização das Entidades	9
3.3. Identificação e Caracterização dos Relacionamentos	10
3.4. Identificação e Caracterização dos Atributos das Entidades e dos Relacionamentos	15
3.5. Apresentação e Explicação do Diagrama ER Produzido	17
4. Modelação Lógica	18
4.1. Construção e Validação do Modelo de Dados Lógico	18
4.2. Apresentação e Explicação do Modelo Lógico Produzido	18
4.3. Normalização de Dados	23
4.4. Validação do Modelo com Interrogações do Utilizador	24
5. Implementação Física	25
5.1. Apresentação e explicação da base de dados implementada	25
5.2. Criação de utilizadores da base de dados	28
5.3. Povoamento da base de dados	29
5.4. Cálculo do espaço da base de dados	34
5.5. Definição e caracterização de vistas de utilização em SQL	36
1. Lista dos Detetives Ativos pela Data de Contratação	36
2. Lista de Suspeitos por Caso	36
3. Relatório do Número de Detenções e Apreensões Diárias	37
5.6. Tradução das interrogações do utilizador para SQL	38
1. Visualizar Operações de uma Determinada Localização	38
2. Lista de Todos os Casos em que um Detetive Está Envolvido	38
3. Quantidade Total de Drogas Apreendidas por Cada Origem	39
5.8. Implementação de procedimentos, funções e gatilhos	40
6. Conclusões e Trabalho Futuro	45
Bibliografia	46
Lista de Siglas e Acrónimos	47

Índice de Figuras

Figura 1 - Diagrama de Gantt

Figura 2 - Relacionamento Caso-Suspeito

Figura 3 - Relacionamento Suspeito-Droga

Figura 4 - Relacionamento Suspeito-Testemunha

Figura 5 - Relacionamento Caso-Droga

Figura 6 - Relacionamento Caso-Testemunha

Figura 7 - Relacionamento Detetive-Caso

Figura 8 - Relacionamento Detetive-Detetive

Figura 9 - Modelo Conceptual

Figura 10 - Conversão da Entidade 'Detetive'

Figura 11 - Conversão da Entidade 'Suspeito'

Figura 12 - Conversão da Relação 'Supervisiona'

Figura 13 - Conversão da Relação 'Investiga'

Figura 14 - Conversão da Relação 'Testemunhado por'

Figura 15 - Conversão da Relação 'Identifica'

Figura 16 - Conversão da Relação 'Envolve'

Figura 17 - Conversão da Relação 'De'

Figura 18 - Conversão da Relação 'Envolve'

Figura 19 - Modelo Lógico

Figura 20 - Implementação física da tabela Detetive

Figura 21 - Implementação física da tabela Supervisão

Figura 22 - Esquema físico

Índice de Tabelas

Tabela 1 - Lista de requisitos

Tabela 2 - Caracterização dos atributos de Detetive

Tabela 3 - Caracterização dos atributos de Caso

Tabela 4 - Caracterização dos atributos de Droga

Tabela 5 - Caracterização dos atributos de Testemunha

Tabela 6 - Caracterização dos atributos de Suspeito

Tabela 7 - Tamanho de cada tipo de atributo

Tabela 8 - Tamanho das tabelas criadas

Tabela 9 - Taxa de crescimento anual de cada tabela criada

1. Definição do Sistema

1.1. Contexto de aplicação (Contextualização)

A Agência de Detetives "Mundo Sóbrio" emerge como uma resposta especializada aos desafios impostos pelo tráfico de drogas e o uso ilícito de substâncias em comunidades diversas. Fundada por ex-agentes das forças policiais e especialistas em segurança, esta organização opera com discrição e eficácia, utilizando métodos de investigação avançados para identificar e neutralizar atividades relacionadas ao narcotráfico.

Localizada estrategicamente numa área discreta da cidade, a agência opera de maneira sigilosa e eficiente, utilizando métodos de investigação avançados e tecnologias de alta gama para alcançar os seus objetivos. O objetivo principal é identificar e neutralizar atividades relacionadas ao narcotráfico, trabalhando em colaboração com as autoridades locais e agências governamentais para garantir a segurança e o bem-estar da comunidade.

No núcleo deste contexto, encontramos a necessidade urgente de uma gestão eficiente de informações e coordenação de casos para lidar com a crescente complexidade das operações criminosas. O fracasso numa operação recente contra o narcotráfico devido à falta de comunicação e partilha de informações entre as equipas de investigação destaca a importância crítica dum sistema centralizado de gestão de dados para a agência.

Diante destes desafios e considerando o papel crucial que a agência desempenha na manutenção da segurança pública, surge a necessidade urgente de desenvolver um Sistema de Bases de Dados (SBD) dedicado. Este sistema, projetado especificamente para atender às necessidades únicas da agência, visa fortalecer a sua capacidade de investigação, coordenação de casos e colaboração com outras entidades de segurança.

1.2. Motivação e Objetivos do Trabalho (Fundamentação e Objetivos)

Nos últimos anos, a Agência de Detetives "Mundo Sóbrio" tem enfrentado desafios crescentes na luta contra o narcotráfico e o uso de substâncias ilícitas.

À medida que as operações de narcotráfico se tornaram mais complexas e sofisticadas, a agência começou a sentir a pressão de lidar com uma quantidade cada vez maior de informações dispersas e desorganizadas. Os casos de investigação acumulavam-se, e a falta de um sistema centralizado de gestão de dados dificultava o acompanhamento e a coordenação eficaz das operações.

A situação atingiu um ponto crítico quando uma grande operação contra o narcotráfico fracassou devido à falta de comunicação e partilha de informações entre as equipas de investigação. Evidências valiosas foram perdidas, suspeitos escaparam e a confiança na agência foi abalada.

Percebendo a necessidade urgente de melhorar a sua capacidade de gestão de informações e coordenação de casos, os líderes da Agência de Detetives "Mundo Sóbrio" reuniram-se para discutir uma solução. Foi neste momento que surgiu a ideia de desenvolver um Sistema de Bases de Dados (SBD) dedicado, projetado especificamente para atender às necessidades únicas da agência.

Para a Agência de Detetives "Mundo Sóbrio", os objetivos a serem alcançados com a implementação do Sistema de Bases de Dados (SBD) são fundamentais para fortalecer a sua capacidade de investigação e combate ao narcotráfico.

Abaixo, enumeram-se e explicam-se cada um desses objetivos:

- **Organização e Gestão de Casos:** O principal objetivo do SBD é organizar e gerir eficientemente os casos de investigação em andamento. Isto inclui registar detalhes sobre suspeitos, evidências, testemunhas e progresso das investigações, garantindo que nenhuma informação crucial seja perdida ou negligenciada.
- **Monitoramento em Tempo Real:** O SBD permitirá à agência monitorar em tempo real o status de cada caso, fornecendo uma visão clara do progresso das investigações e identificando áreas que exigem atenção adicional ou recursos adicionais.
- **Análise Avançada de Dados:** Um objetivo-chave é capacitar a agência a realizar análises avançadas de dados, identificando padrões, tendências e conexões entre diferentes casos e indivíduos. Isto proporcionará insights valiosos que podem informar os agentes sobre estratégias de investigação eficientes, bem como ajudar a prever e prevenir atividades criminosas futuras.
- **Coleta e Preservação de Evidências:** O SBD facilitará a coleta e preservação de evidências, garantindo a sua integridade e admissibilidade em processos judiciais.

Incluindo o registo seguro e organizado de todas as evidências coletadas durante as investigações.

- **Colaboração e Partilha de Informações:** O objetivo é promover a colaboração e a partilha eficaz de informações entre os membros da equipa e outras agências de segurança. O SBD proporcionará uma plataforma centralizada para partilhar dados e insights, facilitando uma abordagem coordenada e colaborativa no combate ao narcotráfico.
- **Transparência e Prestação de Contas:** O SBD promoverá a transparência e a prestação de contas dentro da agência. Isto garantirá que todas as operações sejam conduzidas de acordo com os procedimentos estabelecidos e em conformidade com as leis e regulamentos aplicáveis.

Resumindo, os objetivos delineados visam fortalecer a capacidade da Agência de Detetives "Mundo Sóbrio" de enfrentar os desafios do narcotráfico e promover a segurança e justiça na sociedade. O SBD desempenhará um papel crucial no alcance destes objetivos, habilitando a agência a agir de forma proativa e eficaz na luta contra o crime organizado.

1.3. Análise da Viabilidade do processo

A viabilidade do projeto de implementação do Sistema de Bases de Dados (SBD) para a Agência de Detetives "Mundo Sóbrio" é fundamental para garantir que os recursos investidos tragam retornos significativos e sustentáveis.

A implementação do SBD permitirá à agência melhorar o controlo das suas operações e reduzir perdas financeiras derivadas de falhas na gestão de informações. Por exemplo, ao evitar a perda de provas cruciais ou falhas na comunicação entre as equipas de investigação, a agência poderá recuperar rapidamente parte dos prejuízos associados a casos mal sucedidos.

Com um SBD em funcionamento, a agência poderá realizar as suas operações de investigação de forma mais eficiente e coordenada. Isto inclui otimizar o tempo gasto na coleta e análise de dados, bem como facilitar a colaboração entre equipas de investigação e agências parceiras. Como resultado, a agência poderá realizar um maior número de operações bem-sucedidas num período de tempo mais curto, aumentando a sua eficiência global.

Um SBD fornecerá à agência insights valiosos sobre padrões criminais, tendências de mercado, testemunhas e comportamentos suspeitos. Estas informações ajudarão os agentes a

tomar decisões mais informadas e estratégicas durante as investigações, aumentando a probabilidade de sucesso e reduzindo o risco de erros.

Ao demonstrar um compromisso contínuo com a eficácia e a transparência, a implementação do SBD pode melhorar a reputação da agência e fortalecer a sua relação com clientes e parceiros. Por exemplo, ao fornecer relatórios detalhados sobre o progresso das investigações e os resultados alcançados, a agência pode aumentar a confiança e a satisfação dos seus clientes e colaboradores.

Embora o custo inicial do desenvolvimento e implementação do SBD possa ser significativo, os benefícios a longo prazo superam estes custos. Por exemplo, ao evitar perdas financeiras associadas a casos mal geridos ou investigações mal sucedidas, a agência economizará dinheiro a longo prazo e maximizará o seu retorno sobre o investimento no sistema.

A implementação do SBD é não apenas viável, mas também altamente benéfica para a Agência de Detetives "Mundo Sóbrio". Ao melhorar a eficiência operacional, fortalecer a tomada de decisões e fortalecer as relações com clientes e parceiros, o sistema proporcionará um retorno significativo sobre o investimento e garantirá o sucesso contínuo da agência na sua missão de combate ao narcotráfico.

1.4. Recursos e Equipa de Trabalho

Recursos:

Humanos:

- Agentes da Agência de Detetives "Mundo Sóbrio": Responsáveis pela execução das investigações, coleta de dados e utilização do sistema.
- Equipa de Desenvolvimento de Software: Profissionais especializados na área de desenvolvimento de sistemas, incluindo arquitetos de software, engenheiros de dados e programadores.
- Utilizadores Finais: Representantes da agência que utilizarão o sistema diariamente para registo de informações e acompanhamento dos casos.

Materiais:

- Hardware: Servidores de alta capacidade para alojar o sistema de base de dados, garantindo o desempenho e segurança adequados. Além disso, serão necessários computadores pessoais para acesso ao sistema por parte dos utilizadores.
- Software: Sistema de Gerência da Base de Dados (SGBD) para armazenamento e manipulação eficiente dos dados. Além disso, serão desenvolvidas aplicações

específicas para interação dos utilizadores com o sistema, incluindo interfaces intuitivas e ferramentas de análise de dados.

Equipa de Trabalho:

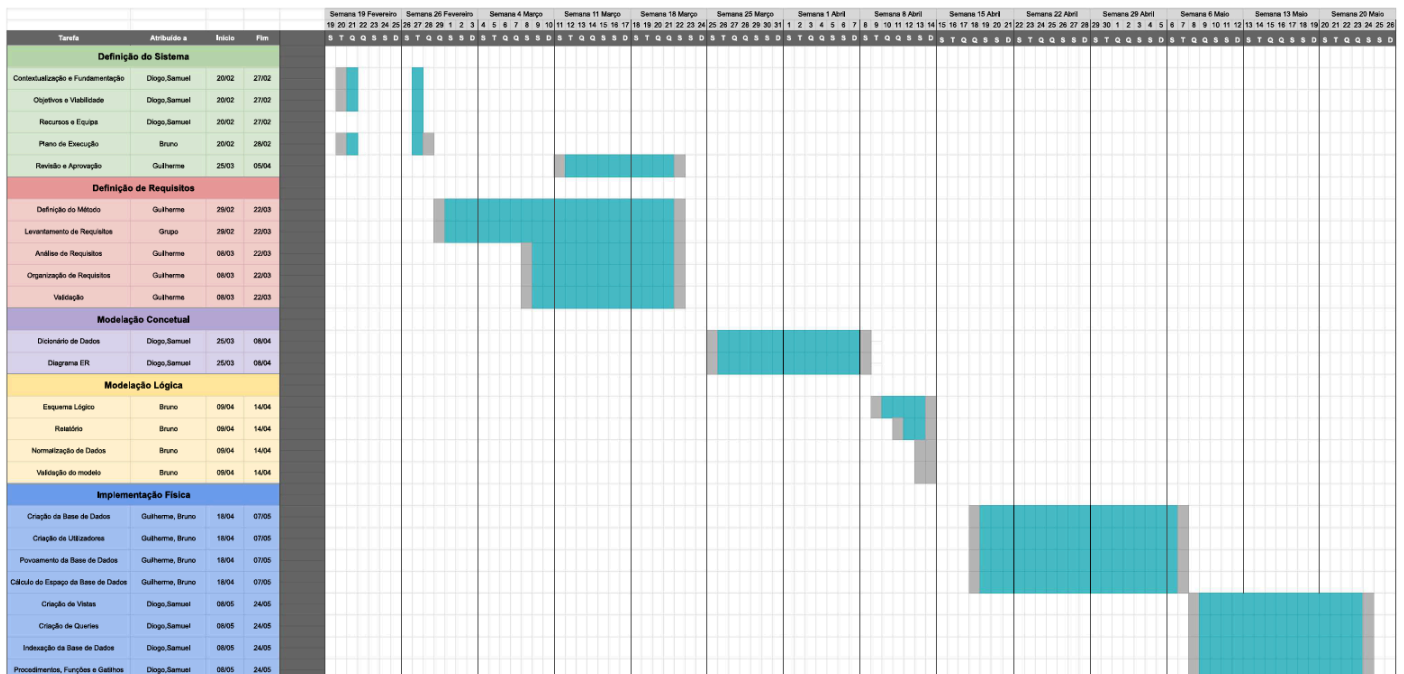
Pessoal Interno:

- Diretores e Gerentes da Agência: Responsáveis pela definição de requisitos, supervisão do projeto e aprovação de decisões importantes.
- Agentes: Envolvidos na coleta de dados e no registo de informações diretamente relacionadas às investigações a decorrer.
- Analistas de Dados: Encarregados de analisar os dados coletados e extrair insights valiosos para apoiar as decisões estratégicas da agência.

Pessoal Externo:

- Equipa de Desenvolvimento de Software: Contratada para projetar, desenvolver e implementar o sistema de bases de dados, incluindo arquitetos de software, engenheiros de dados e programadores.

1.5. Plano de Execução do Projeto



2. Levantamento e Análise de Requisitos

2.1. Método de Levantamento e de Análise de Requisitos Adotado

Para garantir um levantamento abrangente e preciso dos requisitos, foram adotados métodos como: entrevistas estruturadas e questionários onde foram conduzidas entrevistas com a equipa da Agência de Detetives, incluindo gestores e outros funcionários-chave, para entender as suas necessidades e expectativas em relação ao sistema, além disso, foram distribuídos questionários aos detetives e clientes da plataforma, a fim de capturar uma variedade de perspectivas e requisitos essenciais para o desenvolvimento do sistema; para a análise de documentos, foi realizada uma análise detalhada de documentos existentes, como registos de casos anteriores, procedimentos operacionais padrão de agências e regulamentações relevantes, para identificar requisitos e observação; e, finalmente, para a análise de processos, foram feitas observações de processos diários da agência de modo a identificar fluxos de trabalho e requisitos implícitos que não foram expressos diretamente durante as entrevistas e questionários.

Com este método, procuramos garantir que todos os requisitos essenciais sejam identificados e compreendidos de maneira abrangente, permitindo o desenvolvimento de um sistema que atenda às expectativas e necessidades do Mundo Sóbrio, dos detetives e dos clientes da plataforma.

Questionários realizados:

1. Questionário para Detetives:
 - Importância da facilidade do registo de informações no sistema, como detalhes pessoais e histórico de casos?
 - Considera que a lista de suspeitos e testemunhas associadas a um caso é abrangente o suficiente para as suas investigações?
 - Qual é a sua opinião sobre a disponibilidade e clareza das informações fornecidas pelo sistema durante investigações?
2. Questionário para Clientes da Plataforma:
 - Quais as informações que quer que sejam fornecidas pelo sistema sobre casos em andamento?
 - Como avalia a importância da facilidade de acesso ao histórico criminal de suspeitos e evidências relevantes para os casos?

- Acha necessário e relevante que sejam fornecidas notificações sobre eventos importantes relacionados aos casos?

2.2. Organização dos Requisitos Levantados

Tipo	Nº	Data	Descrição do requisito	Fonte	Analista
Descrição	1	21/03/2024	O sistema deve permitir o registo de novos casos, atribuindo um ID único, uma descrição do caso, a data de abertura e o status do caso.	Agência	Diogo
Descrição	2	21/03/2024	O sistema deve permitir o registo de detetives, através do seu ID, nome, data de nascimento, gênero, status, data de contratação, salário.	Agência	Samuel
Descrição	3	21/03/2024	Nos registos dos casos, devem ainda ficar guardados registos dos suspeitos associados, através de um ID, o seu nome, data de nascimento, gênero, endereço, contactos, nível de envolvimento com drogas, o seu histórico criminal se existir e status.	Agentes	Bruno
Descrição	4	21/03/2024	Um caso deve ter um ou vários detetives a investigá-lo e uma lista de suspeitos atribuída.	Agentes	Guilherme
Descrição	5	21/03/2024	Para cada caso, pode ou não haver testemunhas associadas aos suspeitos que testemunham.	Agência	Diogo
Descrição	6	12/03/2024	As testemunhas devem ser registadas através de um ID, nome, data de nascimento, gênero, contactos e o seu relato sobre o caso.	Clientes	Samuel
Descrição	7	21/03/2024	Um detetive pode ou não supervisionar outro detetive.	Clientes	Bruno
Descrição	8	21/03/2024	As evidências, ou seja, as drogas apreendidas pela agência em cada caso, devem estar todas catalogadas no sistema, sendo necessário registar para cada uma, um ID, o nome ou tipo da droga, a quantidade, a origem, data de apreensão e o valor estimado.	Agência	Guilherme
Descrição	9	21/03/2024	Um caso envolve uma ou mais drogas que estão associadas aos suspeitos que as possuem.	Agência	Diogo
Manipulação	10	21/03/2024	O sistema deve listar os detetives pela sua data de contratação.	Agência	Samuel
Manipulação	11	21/03/2024	O sistema deve informar qual a cidade com maior número de suspeitos, ou seja, maior atividade criminal.	Agência	Bruno
Manipulação	12	29/02/2024	O sistema deve fornecer a lista de detetives que trabalham na agência.	Clientes	Guilherme
Manipulação	13	29/02/2024	O sistema apresenta quais os detetives que se encontram atualmente em serviço.	Clientes	Diogo

Manipulação	14	29/02/2024	Visualizar operações de uma determinada localização.	Agência	Samuel
Manipulação	15	29/02/2024	Listar as operações ordenadas por data de abertura e cujo o estado seja ativo.	Agência	Bruno
Controlo	16	29/02/2024	O sistema deve ser capaz de lidar com um grande volume de dados e operações simultâneas, mantendo um bom desempenho.	Clientes	Guilherme
Controlo	17	29/02/2024	O sistema deve garantir a segurança dos dados, com controlo de acesso baseado em funções e encriptação de informações sensíveis.	Agentes	Diogo
Manipulação	18	29/02/2024	A interface do sistema deve ser intuitiva e de fácil utilização para os detetives e utilizadores autorizados.	Agentes	Samuel
Manipulação	19	29/02/2024	O sistema deve estar disponível 24 horas por dia, 7 dias por semana, com um tempo de inatividade mínimo para manutenção.	Agência	Guilherme
Controlo	20	29/02/2024	Deve ser implementado um sistema de backup e recuperação de dados para garantir a integridade das informações em caso de falhas ou desastres.	Clientes	Bruno
Descrição	21	29/02/2024	Elaborar um relatório do número de detenções e apreensões diárias para gestão de recursos e tomada de decisões estratégicas.	Agência	Guilherme

Tabela 1 - Lista de requisitos

2.3. Análise e Validação Geral dos Requisitos

Após o registo dos requisitos, a equipa realizou uma análise abrangente para identificar possíveis inconsistências ou duplicações entre os requisitos levantados. Além disso, foram realizadas reuniões com os gerentes da Agência com o objetivo de garantir a satisfação e a adequação dos requisitos às necessidades da organização.

Após esta fase de análise e validação, os requisitos foram refinados e finalizados, estando prontos para servirem como base para o desenvolvimento do sistema.

3. Modelação Conceptual

3.1. Apresentação da Abordagem de Modelação Realizada

Após a análise dos requisitos e a formulação das necessidades, iniciamos o processo de planeamento da estrutura e design da Base de Dados que será implementada.

Para facilitar o desenvolvimento da nossa base de dados, optamos por iniciar com um Diagrama ER (Entidade-Relacionamento), pois é um tipo de representação gráfica que ilustra como as entidades se relacionam entre si. A sua elaboração é simples, começando pela identificação das entidades envolvidas, seguida dos relacionamentos entre essas entidades e, por fim, a especificação dos atributos das entidades ou dos relacionamentos.

Portanto, uma modelação conceptual deve fornecer um modelo estrutural detalhado das entidades, as suas relações e atributos, de acordo com as exigências de gestão, refletindo um modelo comportamental. No nosso caso, com o objetivo de gerir eficazmente um sistema de informação relacionado à nossa agência de detetives, desenvolvemos esta modelação para permitir uma abordagem próxima da realidade operacional.

Este processo de modelação conceptual proporciona uma visão clara e abrangente da estrutura do sistema de informação, permitindo uma implementação eficiente e uma gestão eficaz dos dados.

3.2. Identificação e Caracterização das Entidades

Ao pensarmos na nossa agência de detetives que se dedica à eliminação de estupefacientes do mundo em que vivemos consideramos que as entidades necessárias seriam: Detetive, Droga, Suspeito, Caso e Testemunha. Para poderem cumprir os requisitos que foram apresentados anteriormente, as entidades possuem os seguintes atributos:

- **Detetive** representa os profissionais encarregados de investigar os casos de tráfico de drogas. Cada detetive é identificado por um ID único e possui informações pessoais como nome, data de nascimento e gênero. O seu *status* pode ser ativo ou inativo, indicando se está atualmente envolvido em atividades de investigação. A data de contratação regista quando o detetive ingressou na agência de detetives e o seu salário reflete a sua remuneração pelo trabalho realizado.

- **Suspeito** representa os indivíduos que estão sob suspeita de envolvimento em atividades criminosas relacionadas com drogas. Cada suspeito é identificado por um ID único e possui informações pessoais como nome, data de nascimento e gênero. O seu endereço é composto por rua, código postal e localidade. Além disso, podem ser registados vários contactos para cada suspeito. O nível de envolvimento com drogas indica o papel que o suspeito desempenha no contexto do crime, podendo variar de usuário a traficante. Caso haja histórico criminal, ele também será registado. O status reflete a situação atual do suspeito no processo da investigação.
- **Droga** representa as substâncias ilícitas apreendidas em operações policiais. Cada droga é identificada por um ID único e pelo nome específico da substância. São registadas informações como a quantidade apreendida, a origem da droga, a data em que foi apreendida e o valor estimado. Estes dados são cruciais para o processo de investigação e o subsequente procedimento legal relacionado ao tráfico de drogas.
- **Caso** representa um caso específico de atividade criminosa relacionada ao tráfico de drogas. Cada caso é identificado por um ID único e possui uma descrição detalhada da situação. O status indica a fase atual do caso, como aberto, em andamento, encerrado, etc. A data de abertura regista quando o caso foi oficialmente iniciado e é fundamental para o gerenciamento do tempo e das ações investigativas.
- **Testemunha** representa indivíduos que podem fornecer informações relevantes para a investigação dos casos. Cada testemunha é identificada por um ID único e possui informações pessoais como nome, data de nascimento e gênero. Os seus contactos podem ser registados para facilitar futuras comunicações. Além disso, cada testemunha fornece um relato detalhado dos eventos ou das observações relacionadas ao caso em questão. Esses relatos são fundamentais para esclarecer os acontecimentos e auxiliar no processo investigativo.

3.3. Identificação e Caracterização dos Relacionamentos

- **Relacionamento Caso - Suspeito**

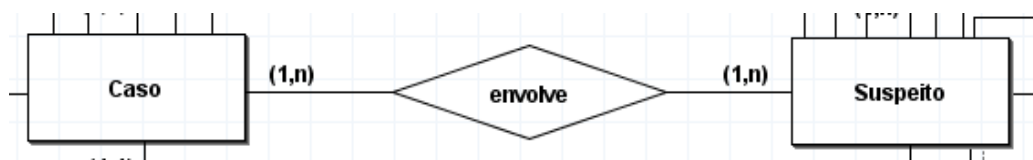


Figura 2 - Relacionamento Caso-Suspeito

Relacionamento: Caso envolve Suspeito

Descrição: Este relacionamento é estabelecido para obter uma ligação entre um caso específico de atividade criminosa mais concretamente o tráfico de drogas, e os indivíduos suspeitos de estarem envolvidos no caso. Isto permite aos detetives associar os suspeitos a casos, o que é crucial para rastrear e analisar a possível rede de indivíduos envolvidos em atividades criminosas. Além disso, fornece uma forma de organizar e gerenciar informações sobre os suspeitos dentro do processo de cada caso, facilitando assim o processo de investigação e tomada de decisões.

Multiplicidade: Caso(1,n) - Suspeito(1,n)

Um caso envolve um ou vários suspeitos assim como cada suspeito tem um ou vários casos.

Atributos: O relacionamento não tem atributos associados

- **Relacionamento Droga-Suspeito**

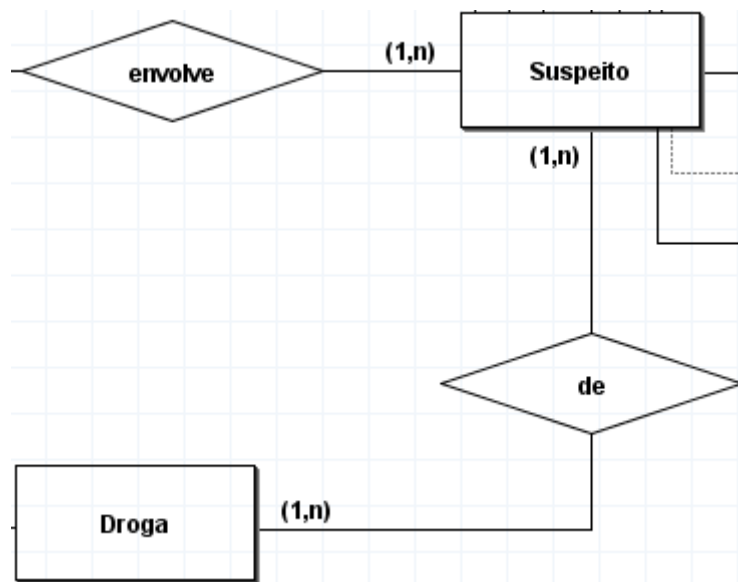


Figura 3 - Relacionamento Suspeito-Droga

Relacionamento: Droga de Suspeito

Descrição: Como o Suspeito pode ter em sua posse vários tipos diferentes de drogas será necessário relacionar estas entidades, de forma a que poder ser registadas as drogas que um suspeito tem em sua posse ou associadas ao mesmo.

Multiplicidade: Droga(1,n) - Suspeito(1,n)

Um suspeito pode possuir várias drogas, e uma droga pode estar associada a vários suspeitos.

Atributos: O relacionamento não tem atributos associados

- **Relacionamento Suspeito-Testemunha**

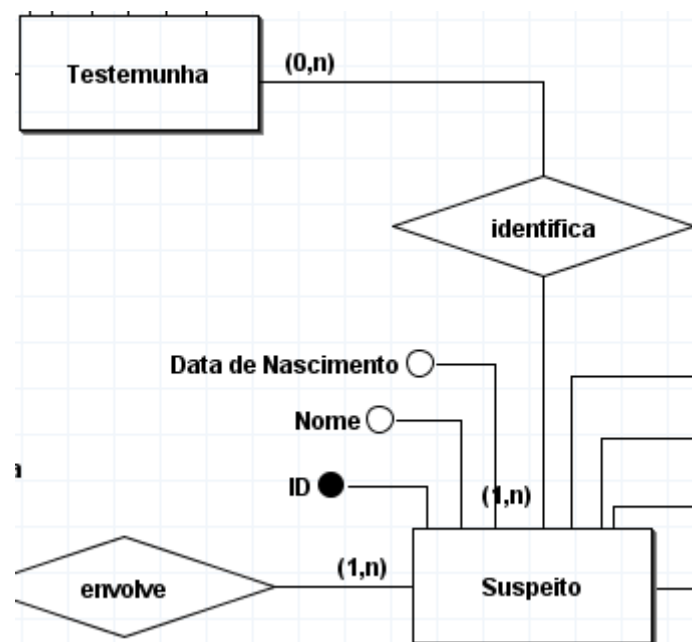


Figura 4 - Relacionamento Suspeito-Testemunha

Relacionamento: Testemunha identifica Suspeito

Descrição: Este relacionamento permite aos investigadores identificar testemunhas que possam fornecer relatos ou evidências que ajudem a esclarecer as circunstâncias em torno das atividades criminosas do suspeito facilitando assim a coleta de informações cruciais para a resolução do caso.

Multiplicidade: Testemunha(0,n) - Suspeito(1,n)

Uma ou várias testemunhas identificam um ou vários suspeitos e um suspeito pode ou não ser identificado por uma ou várias testemunhas.

Atributos: O relacionamento não tem atributos associados

- **Relacionamento Caso-Droga**

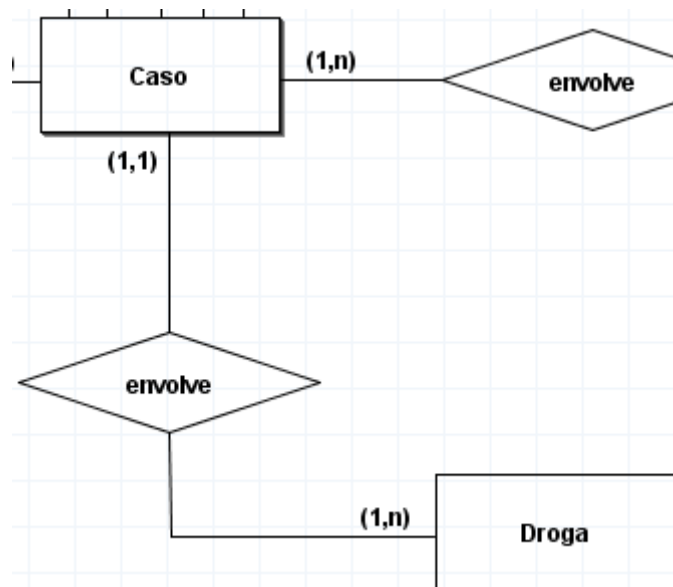


Figura 5 - Relacionamento Caso-Droga

Relacionamento: Caso envolve Droga

Descrição: O relacionamento permite aos detetives rastrear e documentar as drogas específicas envolvidas em cada caso, fornecendo uma visão abrangente da natureza e escala das atividades criminosas relacionadas ao tráfico de drogas. Além disso, ajuda na análise de padrões e tendências, auxiliando na identificação de redes criminosas.

Multiplicidade: Caso(1,1) - Droga(1,n)

Cada caso envolve uma ou mais drogas. No entanto, cada droga está apenas associada a um caso.

Atributos: O relacionamento não tem atributos associados

- **Relacionamento Caso-Testemunha**

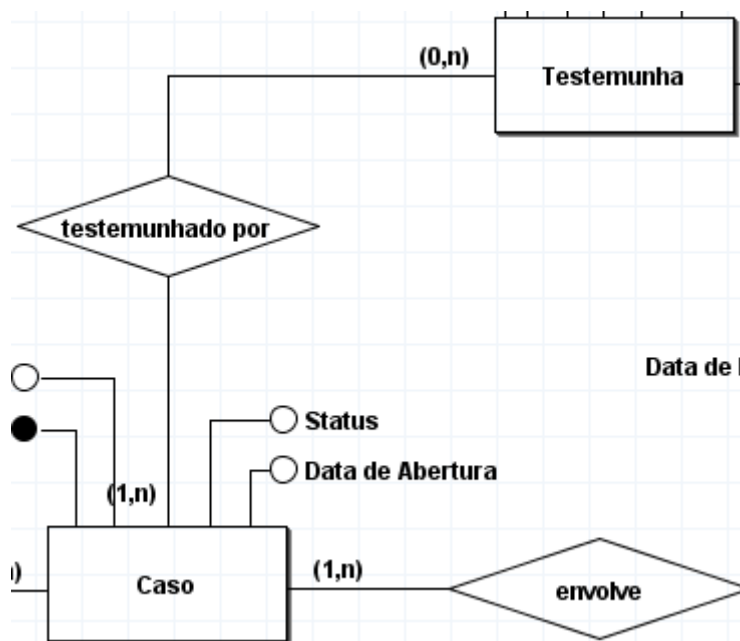


Figura 6 - Relacionamento Caso-Testemunha

Relacionamento: Caso testemunhado por Testemunha

Descrição: Cada caso pode ter ou não testemunhas e no caso de ter é importante que as suas informações fiquem armazenadas desta maneira para depois conseguirmos ter acesso à informação que elas nos fornecem.

Multiplicidade: Caso(1,n) - Testemunha(0,n)

Um caso pode ou não ser testemunhado por uma ou várias testemunhas e uma testemunha está associada a um ou vários casos

Atributos: O relacionamento não tem atributos associados

- **Relacionamento Detetive-Caso**



Figura 7 - Relacionamento Detetive-Caso

Relacionamento: Detetive investiga Caso

Descrição: O relacionamento é essencial no contexto das investigações criminais, permitindo registrar a conexão entre um detetive específico e os casos aos quais ele está atribuído para investigação.

Multiplicidade: Detetive(1,n) - Caso(1,n)

Cada Caso é investigado por um ou vários detetives e cada detetive investiga um a vários casos.

Atributos: O relacionamento não tem atributos associados

- **Relacionamento Detetive-Detetive**

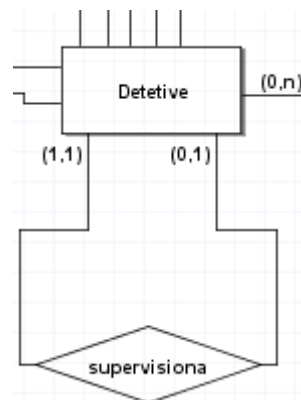


Figura 8 - Relacionamento Detetive-Detetive

Relacionamento: Detetive supervisiona Detetive

Descrição: O relacionamento é estabelecido para refletir a estrutura hierárquica dentro da equipa de investigação, onde detetives mais experientes ou de maior cargo têm a responsabilidade de supervisionar e orientar outros detetives.

Multiplicidade: Detetive(1,1) - Detetive(0,1)

Cada detetive pode ou não ser supervisionado por um outro detetive.

Atributos: O relacionamento não tem atributos associados.

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

Entidade: Detetive

Atributos	Tipo de Dados	Nulo	Composto	Multivalorado	Derivado	Candidato
ID	INT	Não	Não	Não	Não	Sim
Nome	VARCHAR(75)	Não	Não	Não	Não	Não
DataNascimento	DATE	Não	Não	Não	Não	Não
Gênero	CHAR(1)	Não	Não	Não	Não	Não
Status	CHAR(1)	Não	Não	Não	Não	Não
DataContratação	DATE	Não	Não	Não	Não	Não
Salário	DECIMAL(8,2)	Não	Não	Não	Não	Não

Tabela 2 - Caracterização dos atributos de Detetive

Entidade: Caso

Atributos	Tipo de Dados	Nulo	Composto	Multivalorado	Derivado	Candidato
ID	INT	Não	Não	Não	Não	Sim
Descrição	TEXT	Não	Não	Não	Não	Não
Status	CHAR(1)	Não	Não	Não	Não	Não
DataAbertura	DATE	Não	Não	Não	Não	Não

Tabela 3 - Caracterização dos atributos de Caso

Entidade: Droga

Atributos	Tipo de Dados	Nulo	Composto	Multivalorado	Derivado	Candidato
ID	INT	Não	Não	Não	Não	Sim
Nome	VARCHAR(75)	Não	Não	Não	Não	Não
Quantidade	INT	Não	Não	Não	Não	Não
DataApreensão	DATE	Não	Não	Não	Não	Não
Valor	DECIMAL(8,2)	Não	Não	Não	Não	Não
Origem	VARCHAR(75)	Não	Não	Não	Não	Não

Tabela 4 - Caracterização dos atributos de Droga

Entidade: Testemunha

Atributos	Tipo de Dados	Nulo	Composto	Multivalorado	Derivado	Candidato
ID	INT	Não	Não	Não	Não	Sim
Nome	VARCHAR(75)	Não	Não	Não	Não	Não
DataNascimento	DATE	Não	Não	Não	Não	Não
Género	CHAR(1)	Não	Não	Não	Não	Não
Contactos	VARCHAR(75)	Não	Não	Sim	Não	Não
Relato	TEXT	Não	Não	Não	Não	Não

Tabela 5 - Caracterização dos atributos de Testemunha

Entidade: Suspeito

Atributos	Tipo de Dados	Nulo	Composto	Multivalorado	Derivado	Candidato
ID	INT	Não	Não	Não	Não	Sim
Nome	VARCHAR(75)	Não	Não	Não	Não	Não
DataNascimento	DATE	Não	Não	Não	Não	Não
NivelEnvolvimento	ENUM('Suspeito', 'Acusado', 'Condenado')	Não	Não	Não	Não	Não

Género	CHAR(1)	Não	Não	Não	Não	Não
Status	CHAR(1)	Não	Não	Não	Não	Não
Rua	VARCHAR(75)	Não	Sim	Não	Não	Não
CodPostal	VARCHAR(75)	Não	Sim	Não	Não	Não
Localidade	VARCHAR(75)	Não	Sim	Não	Não	Não
HistóricoCriminal	TEXT	Sim	Não	Não	Não	Não
Contactos	VARCHAR(75)	Não	Não	Sim	Não	Não

Tabela 6 - Caracterização dos atributos de Suspeito

3.5. Apresentação e Explicação do Diagrama ER Produzido

Como já explicamos detalhadamente as entidades, os seus respetivos atributos e os relacionamentos definidos entre elas, apresentamos agora o modelo conceptual completo.

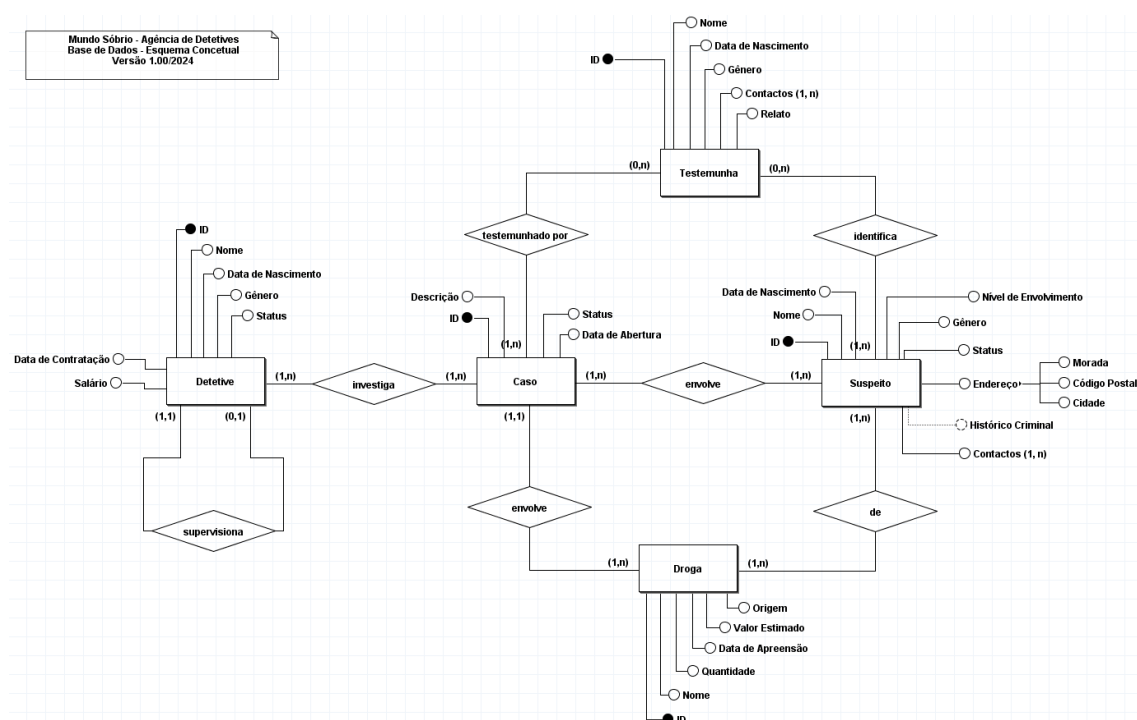


Figura 9 - Modelo Conceptual

4. Modelação Lógica

4.1. Construção e Validação do Modelo de Dados Lógico

Já tendo o modelo conceptual desenvolvido, iniciamos, portanto, o processo de modelação lógica referente à base de dados.

Para o desenvolvimento do modelo lógico são usadas várias operações preliminares à ilustração do diagrama EER através da derivação de relações. Estas operações consistem na normalização dos dados, validação das tabelas e verificação das restrições de integridade.

Recorremos, então, ao uso de um diagrama EER, ou Entidade-Relacionamento Melhorado, que, para além do diagrama ER presente no modelo conceptual que ilustra como as entidades se relacionam entre si, define a organização dos dados e os seus relacionamentos.

Utilizando assim o software de construção de diagramas EER (Entidade-Relação Melhorado) “MySQL Workbench” como ferramenta para a modelação lógica, desenvolvemos o diagrama de acordo com as operações preliminares, assegurando um modelo lógico coeso e organizado.

Este processo de modelação lógica proporciona uma vista clara e organizada do modelo de dados.

4.2. Apresentação e Explicação do Modelo Lógico Produzido

Como explicamos no modelo conceptual o porquê de cada entidade e dos respetivos atributos, vamos só explicar a criação de cada tabela e como é que traduzimos cada atributo das entidades para as tabelas.

As entidades ‘Detetive’, ‘Caso’, ‘Testemunha’, ‘Suspeito’ e ‘Droga’ foram convertidas para tabelas com os respetivos nomes e com os seus atributos como linhas nas tabelas, usando o ID de cada entidade como primary key (PK).

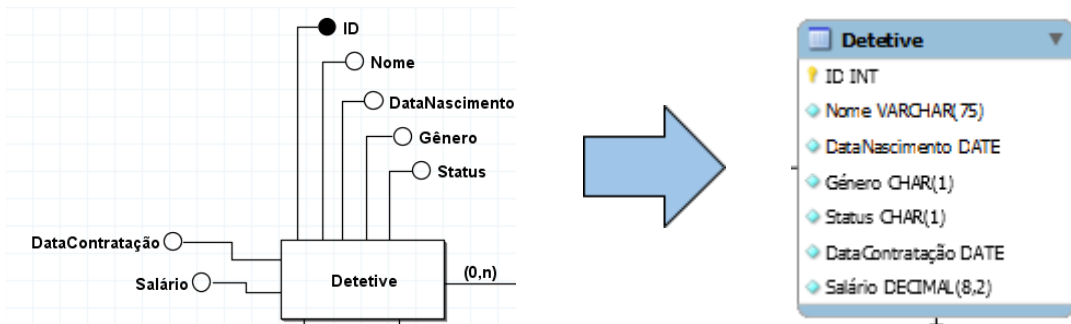


Figura 10 - Conversão da Entidade 'Detetive'

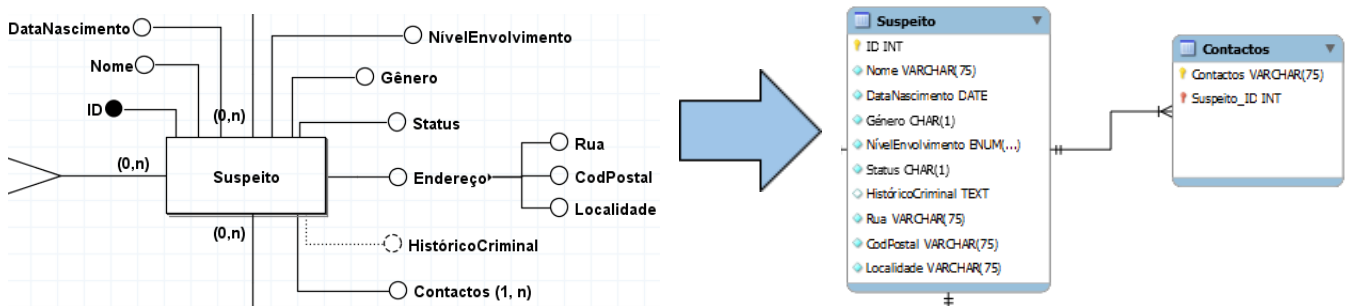


Figura 11 - Conversão da Entidade 'Suspeito'

Como é representado na Figura 11, as entidades com atributos com cardinalidade são representados numa outra tabela contendo como chave primária (PK) o atributo a ser convertido, e como chave estrangeira (FK) o ID da entidade, para que, obtendo a primary key (PK) desta tabela, se consiga fazer uma relação com a entidade assinalada na foreign key (FK).

Já o tópico das relações foi onde encontramos mais dificuldades, porque, ao contrário das entidades que era, muitas das vezes, fazer uma tradução 1:1 para as tabelas, nas relações tivemos de ter em conta a cardinalidade de cada relacionamento bem como as chaves a ser utilizadas nas tabelas referentes.

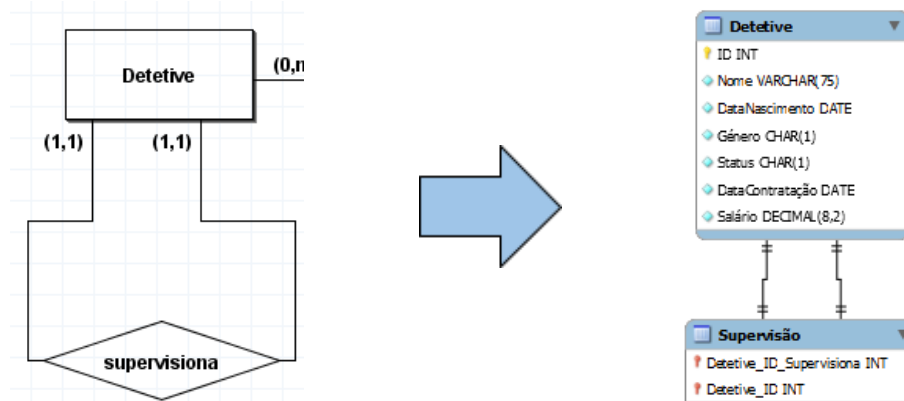


Figura 12 - Conversão da Relação 'Supervisiona'

Esta nova tabela inclui 2 foreign keys (FK) de maneira a ser possível identificar o detetive a supervisionar e o supervisionado e encontra-se relacionada por uma ligação 1:1.

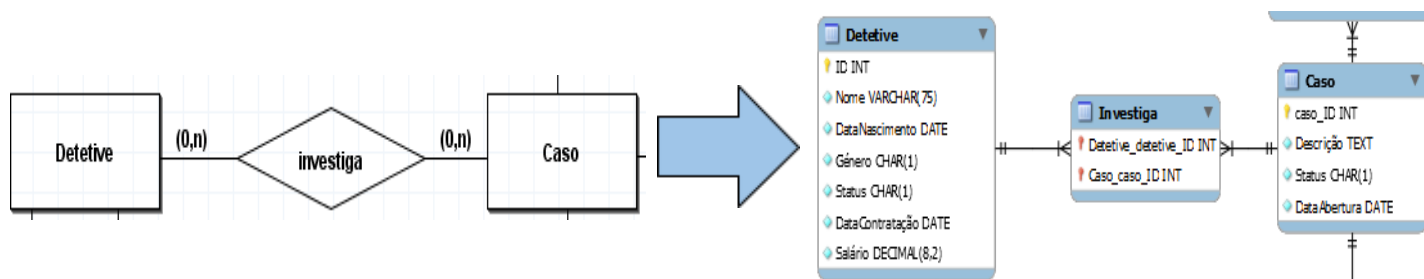


Figura 13 - Conversão da Relação 'Investiga'

Esta nova tabela serve para poder ligar os detetives que estiverem a trabalhar em casos. É formada por uma ligação n:m, já que um detetive pode estar a trabalhar em vários casos, e um caso pode estar a ser investigado por vários detetives.

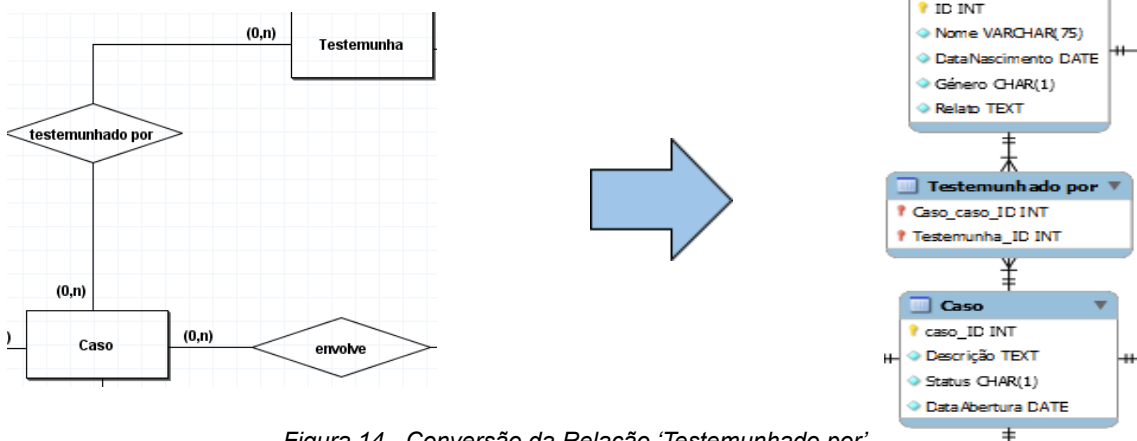


Figura 14 - Conversão da Relação 'Testemunhado por'

A tabela serve para listar as testemunhas que evidenciaram um caso, podendo este ter várias testemunhas, ou até a mesma testemunha evidenciar vários casos, resultando na ligação n:m presente.

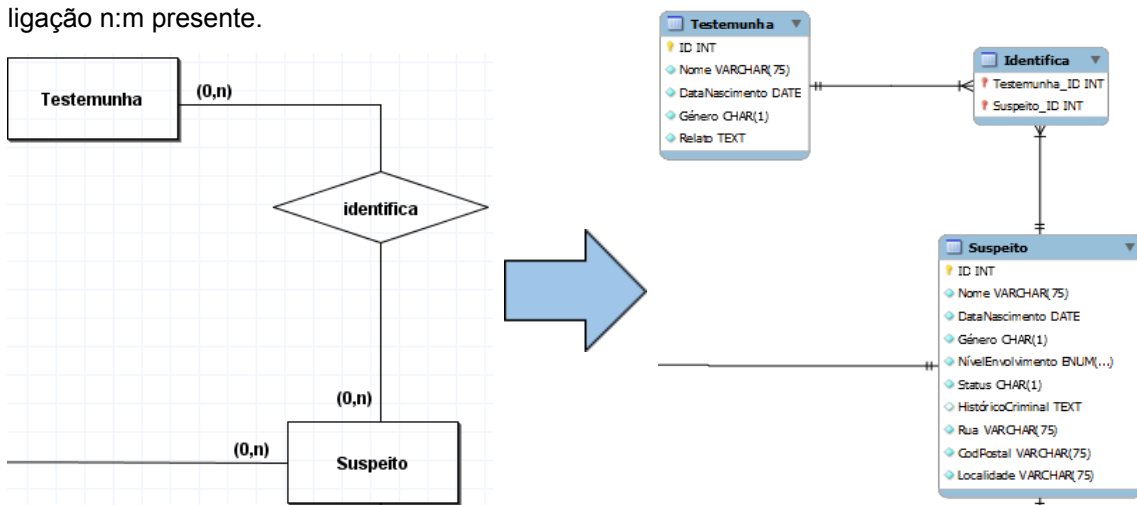


Figura 15 - Conversão da Relação 'Identifica'

A tabela Identifica tem a função de indicar se a testemunha identifica o suspeito como potencial culpado ou não. Está presente uma ligação n:m pois a testemunha pode identificar vários suspeitos, bem como várias testemunhas identificarem o mesmo suspeito.

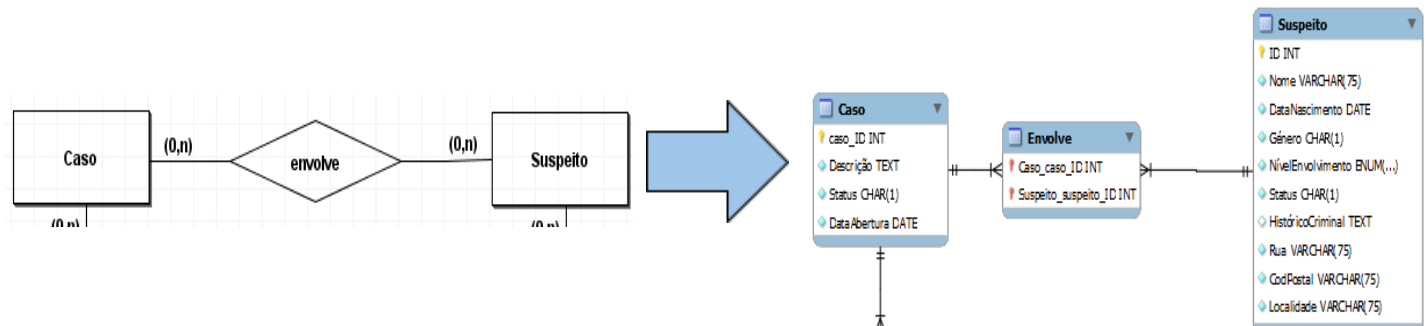


Figura 16 - Conversão da Relação 'Envolve'

Cada caso tem as suas especificações, porém um caso pode envolver apenas um suspeito, como pode envolver múltiplos. Por sua vez, um suspeito pode ser azarado e ser identificado mais do que uma vez sem ter cometido o crime, resultando numa ligação n:m.

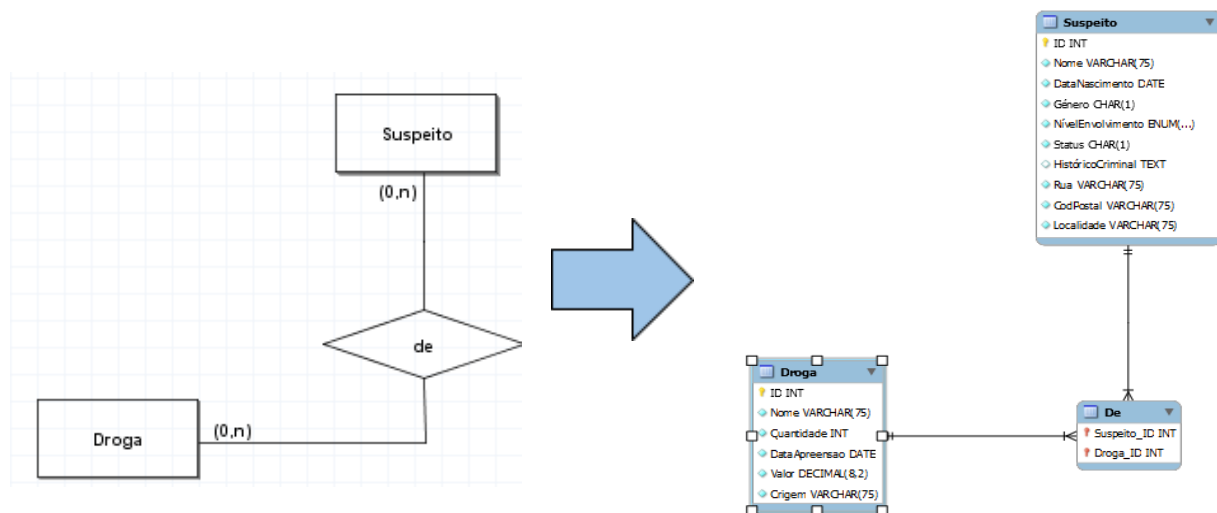


Figura 17 - Conversão da Relação 'De'

A tabela De efetua a relação entre o suspeito carregar uma ou várias drogas, e da mesma maneira, vários suspeitos podem ser detentores de apenas uma droga, visto que podem estar a operar em grupo, resultando assim numa relação n:m.

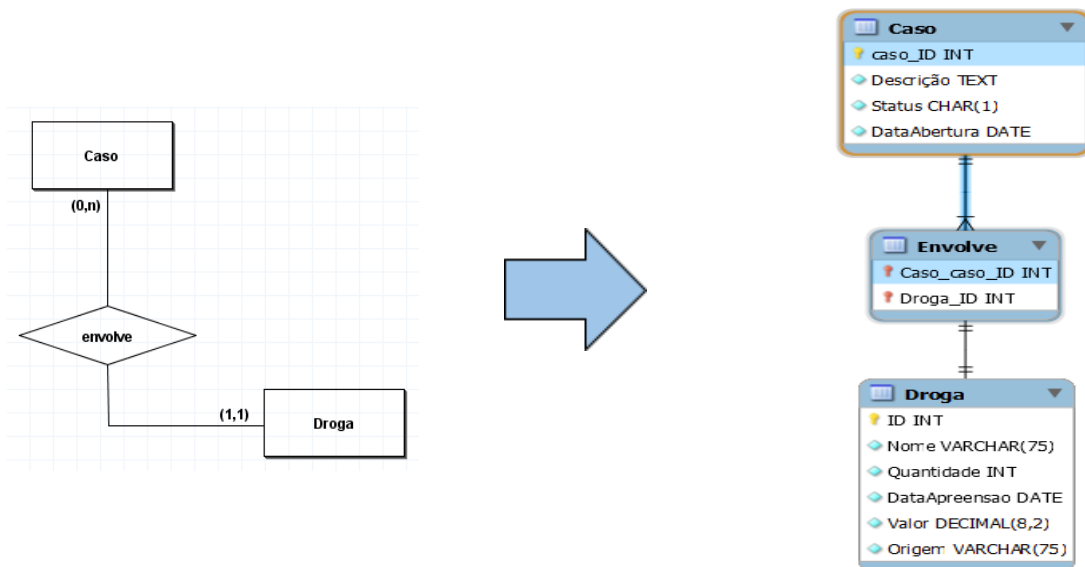


Figura 18 - Conversão da Relação 'Envolve'

No entanto esta relação difere levemente da anterior, pois, muito dificilmente, a mesma droga é apreendida na mesma data e nas mesmas quantidades, mas isso não impossibilita, em cada caso, serem recolhidas diversas drogas. Assim, a última relação desenvolvida é 1:n.

Terminando assim a explicação da criação de cada tabela, segue de seguida a figura com o modelo lógico completo.

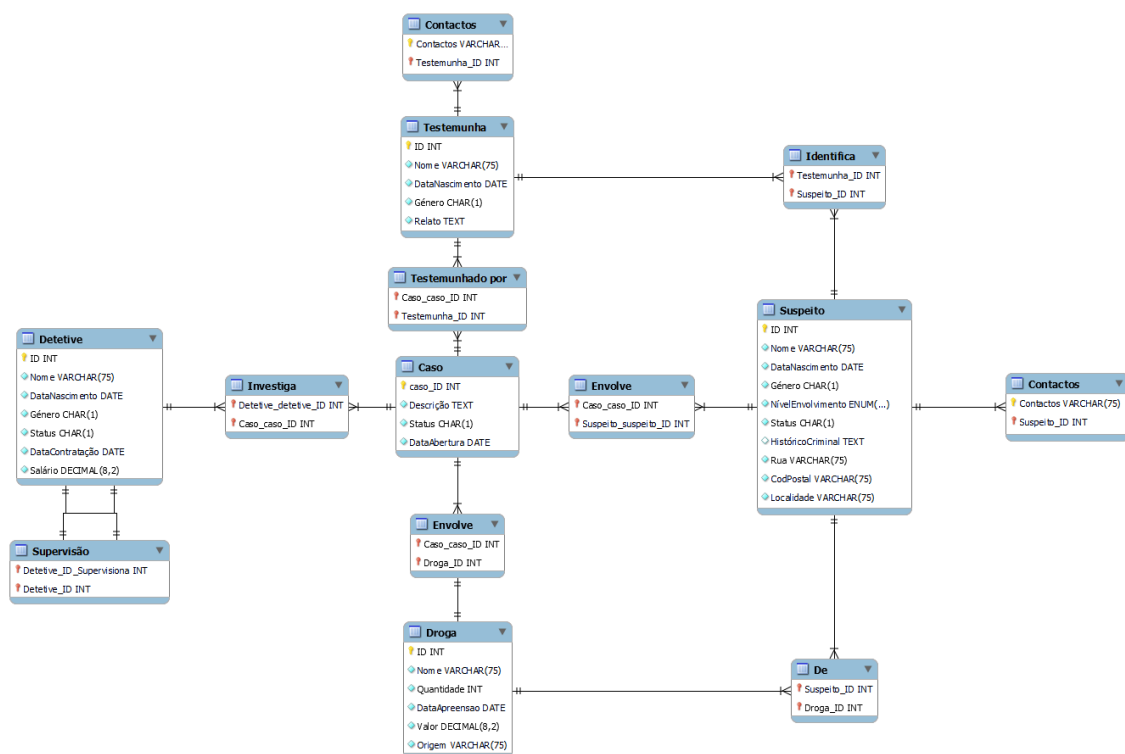


Figura 19 - Modelo Lógico

4.3. Normalização de Dados

O processo de normalização de dados é uma técnica orientada para a organização dos dados numa base de dados relacional. Este método destina-se a assegurar a coesão dos tipos das entidades envolvidas, bem como melhorar os níveis da eficiência de armazenamento e também a integridade e escalabilidade dos dados.

Assim sendo, num processo de normalização de dados, devemos saber que cada registo numa tabela deve representar uma entidade ou instância de um relacionamento; a menção de outras entidades deve ser feita através de foreign keys; o desenvolvimento do esquema pode ser explicado de tabela a tabela e a semântica dos atributos deve ser de fácil interpretação.

A normalização de dados é um processo progressivo que se baseia na execução de uma série de etapas, chamadas de Formas Normais (FN), onde cada uma delas corresponde a critérios concretos de validação, o que torna a base de dados mais robusta e menos vulnerável.

Na primeira forma normal (1FN), o domínio de cada atributo deve ser apenas composto por valores atômicos, ou seja, valores que não podem ser decompostos. Por sua vez, não pode conter grupos de dados repetidos, isto é, não podem existir conjuntos de atributos com valores relacionados.

Segundo o modelo lógico da figura 19, podemos ver que cada tabela tem chave primária, cada atributo é atômico e não existem grupos de dados repetidos.

Na segunda forma normal (2FN) temos de verificar se todos os atributos que são não-primos (que não fazem parte de uma chave primária) são totalmente dependentes da primary key (PK) correspondente.

Neste caso, o que é apresentado vai de encontro com o que a etapa pede, visto que, apenas com o ID das entidades é que é possível descobrir o resto da informação ligada a esse ID.

Na terceira forma normal (3FN) temos de nos certificar que todos os atributos, excetuando a primary key (PK), sejam mutuamente independentes, não havendo dependências funcionais transitivas.

Tal como referimos anteriormente, todas as informações estão dependentes do ID, fazendo com que nenhum dos atributos restantes esteja ligado a outro atributo não-chave.

4.4. Validação do Modelo com Interrogações do Utilizador

A partir do conjunto de requisitos de manipulação estabelecidos, decidimos utilizar os seguintes:

1. O sistema deve listar os detetives pela sua data de contratação. (nº10);
2. O sistema deve fornecer a lista de detetives que trabalham na agência. (nº12);
3. O sistema apresenta quais os detetives atualmente em serviço. (nº13);
4. Visualizar operações de uma determinada localização. (nº14);
5. Listar as operações ordenadas por data de abertura e cujo o estado seja ativo. (nº15).

Para cada um destes requisitos de exploração, apresentamos a sua projeção na álgebra relacional:

1. $\pi_{ID, Nome, DataContratação} (Detetive)$;
2. $Detetive$;
3. $\sigma_{Status = 'A'} (Detetive)$;
4. $\pi_{caso_ID} (\sigma_{Localidade = 'setlocation'} (Suspeito)) \bowtie Caso$;
5. $\tau_{DataAbertura\ asc} (\pi_{caso_ID, Descrição, DataAbertura} (\sigma_{Status = 'A'} (Caso)))$.

Conseguimos, assim, traduzir as expressões referentes aos requisitos de manipulação definidos anteriormente, em álgebra relacional.

5. Implementação Física

5.1. Apresentação e explicação da base de dados implementada

Tendo o modelo lógico desenvolvido, iniciamos, então, a próxima fase que é a implementação física. Nesta etapa apresentamos de forma detalhada a implementação física do sistema de base de dados que foi idealizado durante as fases anteriores. Utilizando a linguagem de programação SQL no MySQL Workbench, conseguimos traduzir o esquema lógico em estruturas físicas concretas. Também detalhamos, abaixo, o processo de criação de cada estrutura, explicando a lógica por trás de cada decisão desta implementação.

Primeiramente é necessário criar o sistema de dados, onde usamos o seguinte conjunto de instruções para poder usar uma estrutura lógica MundoSobrio que agrupa os objetos constituintes da base de dados.

```
DROP SCHEMA IF EXISTS MundoSobrio;  
CREATE SCHEMA MundoSobrio;  
USE MundoSobrio;
```

Após estas preparações, começamos a converter as tabelas do esquema lógico de maneira a que retratassem estas de melhor forma possível.

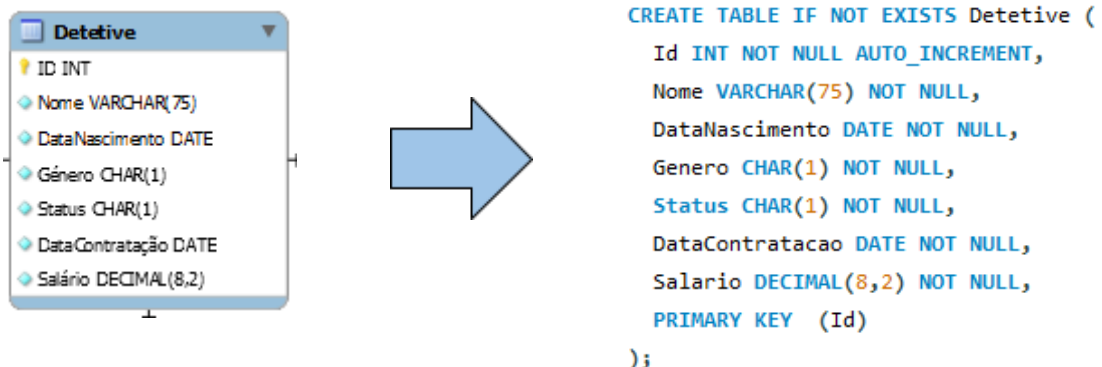


Figura 20 - Implementação física da tabela Detetive

Cada tabela em SQL é uma implementação 1:1 (um por um) da tabela do modelo lógico, onde cada tipo foi implementado exatamente como a sua idealização. As únicas mudanças são referentes à instrução “Auto_Increment” e a adição da instrução “Primary Key”, que referem a, toda a vez que um detetive é adicionado, a base de dados incrementa o ID deste automaticamente, e a identificar qual dos atributos constitui a chave primária (PK), respetivamente.

Já as relações são compostas por 2 chaves estrangeiras, pelo que, o par destas 2 chaves resulta na chave primária da tabela.

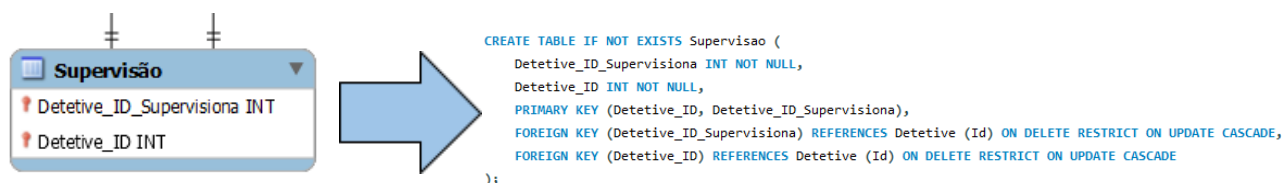


Figura 21 - Implementação física da tabela Supervisão

A sintaxe da aplicação de chaves estrangeiras contém o atributo ao qual identificamos como chave estrangeira, seguido da tabela à qual a chave estrangeira se refere.

Também acrescentamos as cláusulas “ON DELETE RESTRICT” e “ON UPDATE CASCADE”, que impedem a eliminação de registos “pai” caso existam registos que dependem dele, e que, ao atualizar um registo, atualiza automaticamente na tabela relacionada, respetivamente.

Em seguida apresentamos o resto da implementação em linguagem SQL.

```
-- Table Caso
CREATE TABLE IF NOT EXISTS Caso (
    Caso_Id INT NOT NULL AUTO_INCREMENT,
    Descricao TEXT NOT NULL,
    Status CHAR(1) NOT NULL,
    DataAbertura DATE NOT NULL,
    PRIMARY KEY (Caso_Id)
);

-- Table Investiga
CREATE TABLE IF NOT EXISTS Investiga (
    Detetive_ID INT NOT NULL,
    Caso_Id INT NOT NULL,
    PRIMARY KEY (Detetive_ID, Caso_Id),
    FOREIGN KEY (Detetive_ID) REFERENCES Detetive (Id) ON DELETE RESTRICT ON UPDATE CASCADE,
    FOREIGN KEY (Caso_Id) REFERENCES Caso (Caso_Id) ON DELETE RESTRICT ON UPDATE CASCADE
);

-- Table Suspeito
CREATE TABLE IF NOT EXISTS Suspeito (
    Id INT NOT NULL AUTO_INCREMENT,
    Nome VARCHAR(75) NOT NULL,
    DataNascimento DATE NOT NULL,
    Genero CHAR(1) NOT NULL,
    NivelEnvolvimento ENUM('Suspeito', 'Acusado', 'Condenado') NOT NULL,
    Status CHAR(1) NOT NULL,
    HistoricoCriminal TEXT,
    Rua VARCHAR(75) NOT NULL,
    CodPostal VARCHAR(75) NOT NULL,
    Localidade VARCHAR(75) NOT NULL,
    PRIMARY KEY (Id)
);

-- Table Contactos_Suspeito
CREATE TABLE IF NOT EXISTS Contactos_Suspeito (
    Suspeito_ID INT NOT NULL,
    Contactos VARCHAR(75) NOT NULL,
    PRIMARY KEY (Suspeito_ID, Contactos),
    FOREIGN KEY (Suspeito_ID) REFERENCES Suspeito (Id) ON DELETE RESTRICT ON UPDATE CASCADE
);
```



```

-- Table Testemunha
CREATE TABLE IF NOT EXISTS Testemunha (
    Id INT NOT NULL AUTO_INCREMENT,
    Nome VARCHAR(75) NOT NULL,
    DataNascimento DATE NOT NULL,
    Genero CHAR(1) NOT NULL,
    Relato TEXT NOT NULL,
    PRIMARY KEY (Id)
);

-- Table Contactos_Testemunha
CREATE TABLE IF NOT EXISTS Contactos_Testemunha (
    Testemunha_ID INT NOT NULL,
    Contactos VARCHAR(75) NOT NULL,
    PRIMARY KEY (Testemunha_ID, Contactos),
    FOREIGN KEY (Testemunha_ID) REFERENCES Testemunha (Id) ON DELETE RESTRICT ON UPDATE CASCADE
);

-- Table Testemunhado_Por
CREATE TABLE IF NOT EXISTS Testemunhado_Por (
    Caso_Id INT NOT NULL,
    Testemunha_Id INT NOT NULL,
    PRIMARY KEY (Caso_Id, Testemunha_Id),
    FOREIGN KEY (Caso_Id) REFERENCES Caso (Caso_Id) ON DELETE RESTRICT ON UPDATE CASCADE,
    FOREIGN KEY (Testemunha_Id) REFERENCES Testemunha (Id) ON DELETE RESTRICT ON UPDATE CASCADE
);

-- Table Identifica
CREATE TABLE IF NOT EXISTS Identifica (
    Testemunha_Id INT NOT NULL,
    Suspeito_Id INT NOT NULL,
    PRIMARY KEY (Testemunha_Id, Suspeito_Id),
    FOREIGN KEY (Testemunha_Id) REFERENCES Testemunha (Id) ON DELETE RESTRICT ON UPDATE CASCADE,
    FOREIGN KEY (Suspeito_Id) REFERENCES Suspeito (Id) ON DELETE RESTRICT ON UPDATE CASCADE
);

-- Table Envolve_Suspeito
CREATE TABLE IF NOT EXISTS Envolve_Suspeito (
    Caso_Id INT NOT NULL,
    Suspeito_Id INT NOT NULL,
    PRIMARY KEY (Caso_Id, Suspeito_Id),
    FOREIGN KEY (Caso_Id) REFERENCES Caso (Caso_Id) ON DELETE RESTRICT ON UPDATE CASCADE,
    FOREIGN KEY (Suspeito_Id) REFERENCES Suspeito (Id) ON DELETE RESTRICT ON UPDATE CASCADE
);

-- Table Droga
CREATE TABLE IF NOT EXISTS Droga (
    Id INT NOT NULL AUTO_INCREMENT,
    Nome VARCHAR(75) NOT NULL,
    Quantidade INT NOT NULL,
    DataApreensao DATE NOT NULL,
    Valor DECIMAL(8,2) NOT NULL,
    Origem VARCHAR(75) NOT NULL,
    PRIMARY KEY (Id)
);

-- Table De_Droga
CREATE TABLE IF NOT EXISTS De_Droga (
    Suspeito_Id INT NOT NULL,
    Droga_Id INT NOT NULL,
    PRIMARY KEY (Suspeito_Id, Droga_Id),
    FOREIGN KEY (Suspeito_Id) REFERENCES Suspeito (Id) ON DELETE RESTRICT ON UPDATE CASCADE,
    FOREIGN KEY (Droga_Id) REFERENCES Droga (Id) ON DELETE RESTRICT ON UPDATE CASCADE
);

```

```
-- Table Envolve_Droga
CREATE TABLE IF NOT EXISTS Envolve_Droga (
    Caso_Id INT NOT NULL,
    Droga_Id INT NOT NULL,
    PRIMARY KEY (Caso_Id, Droga_Id),
    FOREIGN KEY (Caso_Id) REFERENCES Caso (Caso_Id) ON DELETE RESTRICT ON UPDATE CASCADE,
    FOREIGN KEY (Droga_Id) REFERENCES Droga (Id) ON DELETE RESTRICT ON UPDATE CASCADE
);
```

Figura 22 - Esquema físico

5.2. Criação de utilizadores da base de dados

Criação do Utilizador Administrador

```
DROP USER IF EXISTS admin@localhost;
FLUSH PRIVILEGES;
CREATE USER 'admin'@'localhost' IDENTIFIED BY 'admin1234';
GRANT ALL PRIVILEGES ON MundoSobrio.* TO 'admin'@'localhost';
FLUSH PRIVILEGES;
```

O utilizador 'admin' tem todas as permissões, incluindo gestão de utilizadores, e a capacidade de criar, ler, atualizar e eliminar tabelas e registos.

Começamos por remover qualquer utilizador existente com o nome 'admin' no host 'localhost', garantindo que não haja conflitos. 'FLUSH PRIVILEGES' assegura que as mudanças nas permissões são aplicadas imediatamente.

É criado um novo utilizador chamado 'admin' com a password 'admin1234'.

Finalmente atribuímos todos os privilégios sobre todas as tabelas da base de dados 'MundoSobrio' ao utilizador 'admin'. 'FLUSH PRIVILEGES' aplica novamente as mudanças.

Criação do Utilizador Editor

```
DROP USER IF EXISTS editor@localhost;
FLUSH PRIVILEGES;
CREATE USER 'editor'@'localhost' IDENTIFIED BY 'editor1234';
GRANT SELECT, INSERT, UPDATE, DELETE ON MundoSobrio.* TO 'editor'@'localhost';
FLUSH PRIVILEGES;
```

O utilizador 'editor' pode ler, atualizar e eliminar registos, mas não pode alterar as estruturas das tabelas.

Começamos por remover qualquer utilizador existente com o nome 'editor' no host 'localhost', garantindo que não haja conflitos. 'FLUSH PRIVILEGES' assegura que as mudanças nas permissões são aplicadas imediatamente.

É criado um novo utilizador chamado 'editor' com a password 'editor1234'.

Finalmente, atribuímos os privilégios para a execução de instruções SELECT, INSERT, UPDATE e DELETE nas tabelas da base de dados 'MundoSobrio' ao utilizador 'editor'. 'FLUSH PRIVILEGES' aplica novamente as mudanças.

Criação do Utilizador Viewer

```
DROP USER IF EXISTS viewer@localhost;
FLUSH PRIVILEGES;
CREATE USER 'viewer'@'localhost' IDENTIFIED BY 'viewer1234';
GRANT SELECT ON MundoSobrio.* TO 'viewer'@'localhost';
FLUSH PRIVILEGES;
```

O utilizador 'viewer' apenas poderá ler os registos.

Começamos por remover qualquer utilizador existente com o nome 'viewer' no host 'localhost', garantindo que não haja conflitos. 'FLUSH PRIVILEGES' assegura que as mudanças nas permissões são aplicadas imediatamente.

É criado um novo utilizador chamado 'viewer' com a password 'viewer1234'.

Finalmente atribuímos o privilégio para a execução de instruções SELECT nas tabelas da base de dados 'MundoSobrio' ao utilizador 'viewer'. 'FLUSH PRIVILEGES' aplica novamente as mudanças.

5.3. Povoamento da base de dados

Povoamento com instruções INSERT

```
-- Inserir dados na tabela Detetive
INSERT INTO Detetive (Nome, DataNascimento, Genero, Status, DataContratacao, Salario)
VALUES
('Jorge Costa', '1980-05-15', 'M', 'A', '2022-06-01', 3000.00),
('Mariana Silva', '1985-09-23', 'F', 'A', '2022-03-12', 3200.00),
('Carla Casaca', '1978-11-30', 'F', 'A', '2021-04-23', 2800.00),
('Nuno Miguel', '1990-02-17', 'M', 'A', '2023-01-08', 3500.00);

-- Inserir dados na tabela Supervisao
INSERT INTO Supervisao (Detetive_ID_Supervisiona, Detetive_ID)
VALUES
(1, 2),
(1, 3),
(2, 4);

-- Inserir dados na tabela Caso
INSERT INTO Caso (Descricao, Status, DataAbertura)
VALUES
('Tráfico de cocaína', 'I', '2023-01-15'),
('Tráfico de heroína', 'I', '2023-02-20'),
('Tráfico de MD', 'I', '2023-03-05');
```

A tabela 'Detetive' contém as informações sobre os detetives, incluindo o nome, data de nascimento, gênero, status, data de contratação e salário.

Neste processo povoamos a tabela 'Detetive' com quatro registos:

- Jorge Costa: Nascido a 15 de maio de 1980, contratado a 1 de junho de 2022, com um salário de 3000.00.
- Mariana Silva: Nascida a 23 de setembro de 1985, contratada a 12 de março de 2022, com um salário de 3200.00.
- Carla Casaca: Nascida a 30 de novembro de 1978, contratada a 23 de abril de 2021, com um salário de 2800.00.
- Nuno Miguel: Nascido a 17 de fevereiro de 1990, contratado a 8 de janeiro de 2023, com um salário de 3500.00.

Na tabela 'Supervisao' são colocados 3 dados:

- Detetive com ID 1 supervisiona o Detetive com ID 2.
- Detetive com ID 1 supervisiona o Detetive com ID 3.
- Detetive com ID 2 supervisiona o Detetive com ID 4.

Na tabela 'Caso' inserimos três registos:

- Tráfico de cocaína: Aberto a 15 de janeiro de 2023 e atualmente inativo.
- Tráfico de heroína: Aberto a 20 de fevereiro de 2023 e atualmente inativo.
- Tráfico de MD: Aberto a 5 de março de 2023 e atualmente inativo.

Este é apenas um excerto do processo de povoamento da base de dados. O povoamento das outras tabelas é feito de forma semelhante, utilizando o comando `INSERT INTO` seguido pelo nome da tabela, a lista de colunas e finalmente escrevemos os valores a serem inseridos. Cada conjunto de valores é inserido como um novo registo na tabela correspondente.

Povoamento com recurso a programa Python

O grupo decidiu desenvolver um pequeno programa em Python para povoar duas tabelas, 'Detetive' e 'Suspeito', da base de dados com seis registos cada. Foi escolhida esta linguagem devido à sua facilidade com base em bibliotecas existentes que se podem conectar diretamente à base de dados.

IMPORTS Necessários

```
import mysql.connector
from mysql.connector import errorcode
```

- 'mysql.connector': Biblioteca usada para conectar e interagir com o MySQL.
- 'errorcode': Módulo usado para lidar com códigos de erro específicos do MySQL.

Configuração da Conexão

```
# Configuração da conexão
config = {
    'user': 'admin',
    'password': 'admin1234',
    'host': '127.0.0.1',
    'database': 'MundoSobrio',
    'raise_on_warnings': True
}
```

- user: Nome do utilizador para a conexão ao MySQL. Neste caso utilizamos o utilizador admin criado anteriormente por ter todos os privilégios.
- password: Senha do utilizador.
- host: Endereço do servidor MySQL (neste caso, localhost).
- database: Nome da base de dados a ser usada.
- raise_on_warnings: A 'True' de modo a levantar exceções para todos os avisos SQL.

Dados a Serem Inseridos

```
# Dados para a tabela Detetive
detetives = [
    ('Vasco da Gama', '1979-08-15', 'M', 'A', '2005-06-01', 3000.00),
    ('Sophia Breyner', '1983-03-27', 'F', 'A', '2010-03-12', 3200.00),
    ('Luís de Camões', '1988-11-05', 'M', 'A', '2000-07-19', 2700.00),
    ('Amália Rodrigues', '1976-09-20', 'F', 'A', '2015-01-08', 3500.00),
    ('Fernando Pessoa', '1985-05-12', 'M', 'A', '2000-07-19', 1500.00),
    ('Isabel Santos', '1990-07-03', 'F', 'A', '2015-01-08', 2500.00)
]

# Dados para a tabela Suspeito
suspeitos = [
    ('Henrique Martins', '2003-02-10', 'M', 'Suspeito', 'A', 'Tráfico em 2010', 'Rua A', '4560-000', 'Albufeira'),
    ('Floribela Espanca', '2000-11-25', 'F', 'Acusado', 'A', 'Tráfico em 2015', 'Rua B', '0000-000', 'Amadora'),
    ('Pedro Cabral', '2002-07-18', 'M', 'Condenado', 'A', 'Tráfico em 2018', 'Rua C', '1111-111', 'Bragança'),
    ('Catarina Bragança', '2001-09-03', 'F', 'Suspeito', 'A', 'Tráfico em 2020', 'Rua D', '4567-890', 'Braga'),
    ('José Saramago', '1999-04-15', 'M', 'Acusado', 'A', 'Tráfico em 2012', 'Rua E', '5678-901', 'Faro'),
    ('Teodora Gomes', '1998-08-29', 'F', 'Suspeito', 'A', 'Tráfico em 2019', 'Rua F', '6789-012', 'Aveiro')
]
```

Na variável 'detetives' cada linha representa um detetive e os seus atributos: nome, data de nascimento, gênero, status, data de contratação e salário.

Na variável 'suspeitos' cada linha representa um suspeito e os seus atributos: nome, data de nascimento, gênero, nível de envolvimento, status, histórico criminal, rua, código postal e localidade.

Conexão à Base de Dados e Inserir os Dados

```
# Conectar à base de dados
try:
    cnx = mysql.connector.connect(**config)

    # Inserir dados na tabela Detetive
    add_detetive = ("INSERT INTO Detetive "
                    "(Nome, DataNascimento, Genero, Status, DataContratacao, Salario) "
                    "VALUES (%s, %s, %s, %s, %s, %s)")

    with cnx.cursor() as cursor:
        cursor.executemany(add_detetive, detetives)

    # Inserir dados na tabela Suspeito
    add_suspeito = ("INSERT INTO Suspeito "
                    "(Nome, DataNascimento, Genero, NivelEnvolvimento, Status, HistoricoCriminal, Rua, CodPostal, Localidade) "
                    "VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s)")

    with cnx.cursor() as cursor:
        cursor.executemany(add_suspeito, suspeitos)

    # Commit
    cnx.commit()

    print("Sucesso!")

except mysql.connector.Error as err:
    if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
        print("Erro no user ou pass")
    else:
        print(err)
finally:
    cnx.close()
```

Tentativa de Conexão

```
# Conectar à base de dados
try:
    cnx = mysql.connector.connect(**config)
```

Tentamos estabelecer uma conexão com a base de dados utilizando as configurações definidas em cima.

Inserir Dados na Tabela 'Detetive'

```
# Inserir dados na tabela Detetive
add_detetive = ("INSERT INTO Detetive "
                "(Nome, DataNascimento, Genero, Status, DataContratacao, Salario) "
                "VALUES (%s, %s, %s, %s, %s, %s)")

with cnx.cursor() as cursor:
    cursor.executemany(add_detetive, detetives)
```

- add_detetive: Comando SQL para inserir dados na tabela 'Detetive'. Os valores são representados por '%s'.
- cursor.executemany: Executa a instrução SQL para cada linha na lista 'detetives', inserindo os dados na tabela.

Inserir Dados na Tabela 'Suspeito'

```
# Inserir dados na tabela Suspeito
add_suspeito = ("INSERT INTO Suspeito "
               "(Nome, DataNascimento, Genero, NivelEnvolvimento, Status, HistoricoCriminal, Rua, CodPostal, Localidade) "
               "VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s)")

with cnx.cursor() as cursor:
    cursor.executemany(add_suspeito, suspeitos)
```

- add_suspeito: Comando SQL para inserir dados na tabela 'Suspeito'. Os valores são representados por '%s'.
- cursor.executemany: Executa a instrução SQL para cada linha na lista 'suspeitos', inserindo os dados na tabela.

Commit das Transações e Confirmação de Sucesso

```
# Commit
cnx.commit()

print("Sucesso!")
```

cnx.commit(): Confirma as transações, e guarda as alterações na base de dados. Finalmente realizamos um print para confirmarmos o sucesso do try.

Tratamento de Erros

```
except mysql.connector.Error as err:
    if err.errno == errorcode.ER_ACCESS_DENIED_ERROR:
        print("Erro no user ou pass")
    else:
        print(err)
finally:
    cnx.close()
```

- except mysql.connector.Error as err: Encontra os erros relacionados ao MySQL.
- if err.errno == errorcode.ER_ACCESS_DENIED_ERROR: Verifica se o erro é de login negado (utilizador ou password incorretos).
- finally: Fecha a conexão com a base de dados.

5.4. Cálculo do espaço da base de dados

Para calcular a dimensão da base de dados inicial e a sua taxa de crescimento anual, analisamos cada tabela com base nos atributos e respectivos tipos de dados individualmente. Para o armazenamento inicial usamos, como ponto de partida, que cada tabela contém apenas 1 registo. Como forma de nos orientarmos, aceitamos a sugestão de utilizar uma folha de cálculo para nos ajudar, obtendo as seguintes conclusões:

Tipo	Tamanho
INT	4 bytes
DATE	3 bytes
VARCHAR(N)	N bytes
CHAR(N)	N bytes
DECIMAL(8,2)	4 bytes
TEXT	65535 bytes
ENUM	1 byte

Tabela 7 - Tamanho de cada tipo de atributo

A tabela acima indica todos os tipos dos atributos precisos nas tabelas da base de dados, bem como o tamanho de cada um deles.

Sendo assim, adicionamos o tamanho de cada atributo em cada tabela, o que resultou na tabela seguinte:

Tabela	Tamanho Total
Detetive	91 bytes
Supervisão	8 bytes
Investiga	8 bytes
Caso	65543 bytes
Suspeito	66045 bytes
Contactos_Suspeito	79 bytes
Testemunhado_Por	8 bytes
Testemunha	65618 bytes
Contactos_Testemunha	79 bytes

Tabela	Tamanho Total
Identifica	8 bytes
Envolve_Suspeito	8 bytes
Droga	165 bytes
De_Droga	8 bytes
Envolve_Droga	8 bytes

Tabela 8 - Tamanho das tabelas criadas

Adicionando o tamanho de cada tabela, em que cada uma tem pelo menos 1 registo com os atributos completos, temos que a dimensão inicial da base de dados é de 197,676 bytes, ou seja, perto de 193.04 kB.

Em relação à taxa de crescimento anual, chegamos aos valores indicados na tabela abaixo apresentada através de uma abordagem que nos pareceu adequada.

Tabela	Quantidade de aumento / ano	Tamanho adicionado / ano
Detetive	5	0.44 kB
Supervisão	5	0.04 kB
Investiga	100	0.78 kB
Caso	50	3.13 MB
Suspeito	150	9.45 MB
Contactos_Suspeito	300	23.14 kB
Testemunhado_Por	100	0.78 kB
Testemunha	100	6.26 MB
Contactos_Testemunha	200	15.43 kB
Identifica	70	0.55 kB
Envolve_Suspeito	150	1.17 kB
Droga	50	8.06 kB
De_Droga	50	0.39 kB
Envolve_Droga	50	0.39 kB

Tabela 9 - Taxa de crescimento anual de cada tabela criada

Adicionando tudo, podemos concluir que a taxa de crescimento anual da base de dados é de 18.96 MB.

5.5. Definição e caracterização de vistas de utilização em SQL

No desenvolvimento do Sistema de Bases de Dados (SBD) para a Agência de Detetives "Mundo Sóbrio", foram criadas várias vistas (views) para facilitar consultas específicas e recorrentes. Abaixo estão definidas e caracterizadas algumas das vistas de utilização desenvolvidas:

1. Lista dos Detetives Ativos pela Data de Contratação

Descrição: Esta consulta retorna a lista de todos os detetives que estão ativos, ordenados pela data de contratação. É crucial para analisar a experiência dos detetives e para a gestão de recursos humanos.

```
CREATE VIEW vw_DetetivesAtivosPorDataContratacao AS
SELECT Id, Nome, DataContratacao
FROM Detetive
WHERE Status = 'A'
ORDER BY DataContratacao;
```

Utilização: Permite a visualização dos detetives começando pelos mais antigos na agência, ajudando na tomada de decisões sobre promoções, treinamentos e alocação de casos complexos.

2. Lista de Suspeitos por Caso

Descrição: Esta consulta fornece uma lista detalhada de suspeitos associados a cada caso, incluindo detalhes como a descrição do caso e o nível de envolvimento dos suspeitos. É essencial para acompanhar as investigações e manter um registro organizado das pessoas envolvidas em cada caso.

```
CREATE VIEW vw_SuspeitosPorCaso AS
SELECT c.Caso_Id, c.Descricao, s.Id AS Suspeito_Id, s.Nome, s.NivelEnvolvimento
FROM Caso c
JOIN Envolve_Suspeito es ON c.Caso_Id = es.Caso_Id
JOIN Suspeito s ON es.Suspeito_Id = s.Id;
```

Utilização: Facilita a gestão dos casos, ajudando os detetives a rastrear os suspeitos associados a cada investigação, o que é vital para a resolução eficiente dos casos.

3. Relatório do Número de Detenções e Apreensões Diárias

Descrição: Esta consulta gera um relatório diário do número de detenções e apreensões realizadas, agrupando os dados pela data de apreensão. Isso é fundamental para o monitoramento das atividades da agência.

```
CREATE VIEW vw_DetencoesEApreensoesDiarias AS
SELECT d.DataApreensao AS Data, COUNT(DISTINCT s.Id) AS NumeroDeDetencoes, COUNT(DISTINCT d.Id) AS NumeroDeApreensoes
FROM Droga d
JOIN De_Droga dd ON d.Id = dd.Droga_Id
JOIN Suspeito s ON dd.Suspeito_Id = s.Id
GROUP BY d.DataApreensao
ORDER BY d.DataApreensao;
```

Utilização: Essencial para a gestão de recursos e tomada de decisões estratégicas, oferecendo uma visão clara das atividades diárias relacionadas a detenções e apreensões, ajudando a identificar padrões e áreas que necessitam de mais atenção.

Estas três vistas foram desenvolvidas para melhorar a eficiência das consultas e facilitar a análise de dados críticos para a operação da Agência de Detetives "Mundo Sóbrio". Ao fornecer uma camada de abstração sobre os dados subjacentes, as vistas facilitam a realização de consultas complexas e recorrentes, simplificando o acesso e a análise dos dados para os utilizadores. As três vistas de utilização apresentadas nesta seção demonstram como é possível organizar e disponibilizar dados de forma mais eficaz, atendendo às necessidades específicas dos utilizadores da base de dados implementada. Essas vistas não apenas melhoram a eficiência das consultas, como também contribuem para uma melhor compreensão e utilização dos dados, promovendo assim uma gestão mais eficaz e informada da base de dados.

5.6. Tradução das interrogações do utilizador para SQL

Na etapa de "Tradução das interrogações do utilizador para SQL", o objetivo é apresentar consultas SQL que correspondam às interrogações previamente definidas e validadas durante a construção do esquema lógico da base de dados. Este processo envolve transformar as necessidades e requisitos do utilizador em comandos SQL que possam ser executados na base de dados.

Aqui, serão apresentadas três consultas de exemplo que demonstram como as interrogações dos utilizadores podem ser convertidas em instruções SQL. Estas consultas não só traduzem as necessidades dos utilizadores para a linguagem SQL, mas também fornecem um exemplo prático de como realizar operações específicas na base de dados, abordando os requisitos funcionais e as características do sistema em questão.

1. Visualizar Operações de uma Determinada Localização

Descrição: Esta consulta lista as operações (casos) para uma localização específica, como "Braga". Ela permite uma análise detalhada das operações em uma determinada área, incluindo detalhes do caso e a localização dos suspeitos.

```
SELECT *  
FROM vw_OperacoesPorLocalizacao  
WHERE Localidade = 'Braga'; -- Substituir 'Braga' pela localização desejada
```

Utilização: Útil para investigações focadas em áreas geográficas específicas, permitindo identificar padrões criminais e tomar decisões estratégicas com base nos dados de localização.

2. Lista de Todos os Casos em que um Detetive Está Envolvido

Descrição: Esta consulta lista todos os casos em que um detetive está envolvido, seja como investigador principal ou supervisor. Ela fornece uma visão abrangente das atividades de cada detetive na agência, destacando seu envolvimento em investigações específicas.

```
SELECT d.Nome AS Detetive_Nome, c.Caso_Id, c.Descricao, i.Detetive_ID AS Investigador_ID
FROM Detetive d
LEFT JOIN Investiga i ON d.Id = i.Detetive_ID
LEFT JOIN Caso c ON i.Caso_Id = c.Caso_Id;
```

Utilização: Importante para monitorar a participação dos detetives em investigações individuais, permitindo uma melhor distribuição de casos e uma análise da carga de trabalho de cada detetive.

3. Quantidade Total de Drogas Apreendidas por Cada Origem

Descrição: Esta consulta calcula a quantidade total de drogas apreendidas agrupadas por origem. Ela fornece uma visão geral da distribuição das drogas apreendidas com base na sua origem, auxiliando na compreensão dos padrões de tráfico de drogas e no direcionamento de esforços de combate ao crime.

```
SELECT Origem, SUM(Quantidade) AS Quantidade_Total_Apreendida
FROM Droga
GROUP BY Origem;
```

Utilização: Importante para a análise de dados relacionados ao tráfico de drogas, ajudando na identificação de rotas de tráfico e origens de drogas ilegais, o que pode informar estratégias de aplicação da lei e políticas de prevenção.

Após a apresentação das três consultas de exemplo na etapa de "Tradução das interrogações do utilizador para SQL", é possível concluir que a construção de consultas SQL eficazes é fundamental para atender às necessidades e requisitos dos utilizadores no contexto de um sistema de base de dados. Essas consultas representam uma ponte entre os requisitos do utilizador e a manipulação de dados na base de dados, permitindo que as operações sejam realizadas de forma precisa e eficiente.

Portanto, a capacidade de compreender as necessidades dos utilizadores e traduzi-las em consultas SQL adequadas é uma habilidade essencial para desenvolvedores e administradores de base de dados, garantindo o funcionamento eficiente e eficaz do sistema de base de dados como um todo.

5.7. Indexação do Sistema de Dados

Ao criar um objeto na base de dados são criados, automaticamente, índices para as colunas com restrições de PRIMARY KEY ou UNIQUE, conhecidos como índices implícitos. No entanto, também podemos definir os nossos próprios índices, chamados de índices explícitos. Existem algumas razões para definir índices nas tabelas da base de dados, estes aceleram as operações de seleção e consulta, como SELECT e WHERE e índices únicos garantem também a unicidade dos valores em colunas específicas. Por outro lado, abrandam operações de inserção e atualização, como INSERT e UPDATE.

Assim sendo, devemos tentar apenas definir índices para as colunas que são consultadas com mais frequências mas que, no entanto, não são atualizadas com frequência. Escolhemos então duas tabelas para definirmos índices. Começamos com a tabela Detetive na coluna Nome para assim facilitar a procura por nome de detetives, que podem ser frequentes em operações de pesquisa e relatórios.

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

Por fim, na tabela Caso e nas colunas Status e DataAbertura para melhorar a velocidade de consulta em situações que é preciso filtrar os casos por status e data de abertura, sendo esta uma situação comum em casos que acabam por estar fechados durante um certo período.

```
CREATE INDEX idx_caso_status_data ON Caso(Status, DataAbertura);
```

5.8. Implementação de procedimentos, funções e gatilhos

O procedimento *RegistrarCaso* foi criado para facilitar a inserção de um novo caso na base de dados e associar automaticamente detetives, suspeitos e testemunhas ao caso. Esta automatização é importante para garantir que todos os relacionamentos necessários são estabelecidos de forma consistente, reduzindo a probabilidade de erros e inconsistências nos dados.

```

CREATE PROCEDURE RegistrarCaso (
    IN pDescricao TEXT,
    IN pStatus CHAR(1),
    IN pDataAbertura DATE
)
BEGIN
    DECLARE vCaso_Id INT;
    -- Iniciar uma transação
    START TRANSACTION;
    BEGIN
        -- Inserir um novo caso
        INSERT INTO Caso (Descricao, Status, DataAbertura)
        VALUES (pDescricao, pStatus, pDataAbertura);
        SET vCaso_Id = LAST_INSERT_ID();
        -- Associar detetives ao caso
        INSERT INTO Investiga (Detetive_ID, Caso_Id)
        SELECT Detetive_ID, vCaso_Id
        FROM TempDetetives;
        -- Associar suspeitos ao caso
        INSERT INTO Envolve_Suspeito (Caso_Id, Suspeito_Id)
        SELECT vCaso_Id, Suspeito_Id
        FROM TempSuspeitos;
        -- Associar testemunhas ao caso
        INSERT INTO Testemunhado_Por (Caso_Id, Testemunha_Id)
        SELECT vCaso_Id, Testemunha_Id
        FROM TempTestemunhas;
        -- Commit da transação
        COMMIT;
    END;
END;

```

Exemplo de Uso e Aplicação de Transações

Consideramos um cenário concreto onde é necessário registrar um novo caso envolvendo vários detetives, suspeitos e testemunhas. Este procedimento assegura que todas as operações sejam feitas de forma integral, ou seja, todas as operações são confirmadas juntas ou nenhuma é confirmada.

Criação das Tabelas Temporárias:

```
-- Criar tabelas temporárias
CREATE TEMPORARY TABLE TempDetetives (Detetive_ID INT);
CREATE TEMPORARY TABLE TempSuspeitos (Suspeito_ID INT);
CREATE TEMPORARY TABLE TempTestemunhas (Testemunha_ID INT);
```

Inserção de Dados nas Tabelas Temporárias:

```
-- Inserir dados nas tabelas temporárias
INSERT INTO TempDetetives (Detetive_ID) VALUES (1), (2), (3);
INSERT INTO TempSuspeitos (Suspeito_ID) VALUES (4), (5);
INSERT INTO TempTestemunhas (Testemunha_ID) VALUES (6), (7);
```

Chamada do Procedimento:

```
-- Chamar o procedimento
CALL RegistrarCaso('Descrição do caso', 'A', '2024-05-27');
```

Explicação

Ao chamar o procedimento *RegistrarCaso*, ocorre o seguinte:

É inserido um novo registo na tabela *Caso* com a descrição, status e data de abertura fornecidos. O ID do caso recém-criado é recuperado. Os detetives (com IDs 1, 2 e 3), suspeitos (com IDs 4 e 5) e testemunhas (com IDs 6 e 7) são associados ao caso usando os dados das tabelas temporárias. Todas estas operações são executadas dentro de uma transação. Se alguma destas operações falhar, a transação inteira será revertida, garantindo assim a integridade dos dados. Desta forma, o uso de transações assegura que o novo caso e todas as suas associações são criados de forma consistente e integral.

A função *NumeroCasosAbertos* foi criada para fornecer de uma maneira eficiente e conveniente de contar o número de casos que estão atualmente abertos no sistema. Isto é útil em situações onde há necessidade de monitorar ou criar relatórios sobre o Status dos casos, permitindo que outros procedimentos, funções ou consultas utilizem essa contagem diretamente sem a necessidade de duplicar lógica ou realizar consultas complexas repetidamente.


```

CREATE FUNCTION NumeroCasosAbertos()
RETURNS INT
BEGIN
    DECLARE vNumCasos INT;

    SELECT COUNT(*)
    INTO vNumCasos
    FROM Caso
    WHERE Status = 'A';

    RETURN vNumCasos;
END;

```

Exemplo Concreto de Uso

Após a criação da função, esta pode ser chamada em consultas para obter o número de casos abertos. Por exemplo:

```
SELECT NumeroCasosAbertos() AS CasosAbertos;
```

Isto retornará o número total de casos que estão atualmente abertos no sistema, fornecendo uma maneira rápida e eficiente de ter acesso a essa informação.

Benefícios

Reutilização: A lógica da contagem está centralizada numa única função, evitando assim a duplicação de código.

Manutenção: Facilita a manutenção, pois qualquer alteração na lógica da contagem precisa apenas de ser feita na função.

Desempenho: Melhora o desempenho em consultas complexas que precisam dessa informação frequentemente, pois encapsula a contagem em uma função simples.

O gatilho *AtualizaStatusDetetive* foi criado para automatizar a atualização do status de um detetive baseado na sua data de contratação. A regra especifica que um detetive deve ser marcado como inativo ('I') se sua data de contratação for superior a 10 anos. Isso ajuda a manter os registos dos detetives atualizados automaticamente, sem a necessidade de que haja uma intervenção manual, garantindo a integridade e a consistência dos dados.

```

CREATE TRIGGER AtualizaStatusDetetive
AFTER UPDATE ON Detetive
FOR EACH ROW
BEGIN
    IF NEW.DataContratacao < DATE_SUB(CURDATE(), INTERVAL 10 YEAR) THEN
        UPDATE Detetive
        SET Status = 'I'
        WHERE Id = NEW.Id;
    END IF;
END //

```

Exemplo Concreto de Uso

Considerando um cenário onde um detetive foi contratado há mais de 10 anos, mas o seu registo na tabela Detetive é atualizado por algum motivo (por exemplo, atualização de salário).

Situação Inicial:

Detetive com ID 3 foi contratado a '2000-07-19' e tem o status 'A' (ativo).

```

('Carla Casaca', '1978-11-30', 'F', 'A', '2000-07-19', 2800.00),

```

Atualização do Registo

```

UPDATE Detetive
SET Salario = 3300.00
WHERE Id = 3;

```

Ação do Gatilho

Após a atualização, o gatilho é acionado. O gatilho verifica se já passaram 10 anos desde a data de contratação. Como desde a data de contratação '2000-07-19' já passaram mais de 10 anos, o gatilho atualiza o status do detetive para 'I'.

6. Conclusões e Trabalho Futuro

O desenvolvimento e a implementação da base de dados representam um marco significativo para a Agência de Detetives "Mundo Sóbrio" no seu esforço contínuo de combate ao narcotráfico e ao uso de substâncias ilícitas.

Ao longo deste trabalho, foram identificados os principais desafios enfrentados pela agência, tais como a falta de coordenação entre as equipas, perda de evidências e deficiências na gestão de informação, pelo que, através da análise detalhada destes problemas e da definição clara de objetivos, bem como a criação do modelo conceptual e lógico, foi possível conceber e implementar um sistema robusto que atende às necessidades específicas da agência.

A implementação deste teve tanto aspetos positivos quanto negativos, pois tivemos, várias vezes, trabalho que tivemos de refazer, porém, o facto de fazermos os modelos e termos os requisitos bem definidos, ajudou muito na organização do projeto inteiro.

Além disso, é importante destacar que a natureza dinâmica do combate ao narcotráfico requer uma abordagem adaptável e em constante evolução. Nesse sentido, a base de dados foi concebida como uma ferramenta flexível e escalável, capaz de se adaptar às mudanças nas operações criminosas e às necessidades da agência ao longo do tempo.

Concluindo, o desenvolvimento e a implementação do sistema de base de dados representam um passo significativo na capacidade da Agência de Detetives "Mundo Sóbrio" de cumprir a sua missão de combate ao narcotráfico. No entanto, o trabalho futuro será fundamental para garantir que o sistema permaneça eficaz e adaptável às necessidades em constante evolução da agência.

Bibliografia

Matéria disponibilizada pelos professores da UC.

MySQL Exemplos e Ajudas -<https://www.w3schools.com/MySQL/default.asp>.

Lista de Siglas e Acrónimos

BD	Base de Dados
SBD	Sistema de Bases de Dados
PK	Primary Key
FK	Foreign Key
FN	Fórmula Normal
SQL	Structured Query Language