



Universidade do Minho
Escola de Engenharia
Licenciatura em Engenharia Informática

Unidade Curricular de Bases de Dados

Ano Letivo de 2025/2026

PortusPesca - Grupo 9

Guilherme Baptista
a111906

Salvador Coelho
a108663

Matteo Murari
a112155

Rui Rebelo
a112411

Dezembro, 2025

BD

| | |
|-----------------|--|
| Data da Receção | |
| Responsável | |
| Avaliação | |
| Observações | |

PortusPesca - Grupo 9

Guilherme Baptista Salvador Coelho Matteo Murari Rui Rebelo
a111906 a108663 a112155 a112411

Dezembro, 2025

Resumo

No âmbito da Unidade Curricular de Bases de Dados, lecionada pelo Professor Orlando Belo, o presente relatório documenta todo o ciclo de desenvolvimento de um Sistema de Base de Dados Relacional. Respondendo ao desafio de “Catalogação de Viagens Virtuais”, o projeto centra-se na PortusPesca, uma empresa do setor marítimo, visando a otimização da gestão da sua frota pesqueira e o controlo detalhado das expedições em alto mar.

A motivação para este sistema surgiu da necessidade de modernizar os processos operacionais da empresa, garantindo maior segurança na integridade dos dados e uma eficiência operacional superior na coordenação das tripulações e capturas.

Metodologicamente, o trabalho evoluiu desde uma fase inicial de abstração e levantamento de requisitos, passando pela modelação conceptual e lógica (com recurso ao BR-Modelo), até culminar na fase de implementação física. Nesta última etapa, materializada sobre o Sistema de Gestão de Bases de Dados (SGBD) MySQL, foi construída a infraestrutura robusta do sistema. Esta construção envolveu não só a definição de tabelas e relações, mas também a criação estratégica de utilizadores e permissões de segurança, vistas para simplificação de leitura, índices para performance, bem como a automatização de regras de negócio através de procedimentos armazenados, funções e gatilhos. Adicionalmente, o projeto contemplou o povoamento da base de dados e o respetivo cálculo de espaço ocupado.

Área de Aplicação: Desenho e arquitetura de Sistemas de Bases de Dados.

Palavras-Chave: Sistema de Base de Dados, PortusPesca, Modelação Conceptual, Modelação Lógica, Implementação Física

Índice

| | |
|---|-----------|
| 1. Definição do Sistema | 1 |
| 1.1. Contexto de aplicação | 1 |
| 1.2. Motivação e Objetivos | 2 |
| 1.3. Análise da Viabilidade do Processo | 3 |
| 1.4. Recursos e Equipa de Trabalho | 4 |
| 1.4.1. Recursos Humanos | 4 |
| 1.4.2. Recursos Físicos | 4 |
| 1.4.3. Recursos Digitais | 4 |
| 1.4.4. Equipa de Trabalho | 4 |
| 1.5. Plano de Execução do Projeto | 5 |
| 2. Levantamento e Análise de Requisitos | 7 |
| 2.1. Método de Levantamento e de Análise de Requisitos Adotado | 7 |
| 2.2. Organização dos Requisitos Levantados | 8 |
| 2.2.1. Requisitos de Descrição (RD) | 8 |
| 2.2.2. Requisitos de Exploração (RE) | 10 |
| 2.2.3. Requisitos de Administração (RA) | 11 |
| 2.3. Análise e Validação Geral dos Requisitos | 12 |
| 3. Modelação Conceptual | 13 |
| 3.1. Apresentação da Abordagem de Modelação Realizada | 13 |
| 3.2. Identificação e Caracterização das Entidades | 14 |
| 3.3. Identificação e Caracterização dos Relacionamentos | 15 |
| 3.4. Identificação e Caracterização dos Atributos das Entidades e dos Relacionamentos | 18 |
| 3.5. Apresentação e Explicação do Diagrama ER Produzido | 23 |
| 4. Modelação Lógica | 24 |
| 4.1. Construção e Validação do Modelo de Dados Lógico | 24 |
| 4.2. Apresentação e Explicação do Modelo Lógico Produzido | 25 |
| 4.3. Normalização de Dados | 31 |
| 4.4. Validação do Modelo com Interrogações do Utilizador | 32 |
| 4.4.1. Listagem geral de viagens | 32 |
| 4.4.2. Identificação de viagens ativas para atualização de estado | 33 |
| 4.4.3. Validação de estado contratual de tripulante, cargo e estado | 34 |
| 4.4.4. Histórico operacional de tripulante (exemplo ID = 1) | 35 |
| 4.4.5. Relatório diário de lucro da viagem com base nas receitas e despesas | 36 |
| 4.4.6. Estatística semanal de capturas por zona e espécie | 39 |
| 4.4.7. Auditoria Integral de Expedição | 40 |
| 5. Implementação Física | 43 |
| 5.1. Apresentação e explicação da base de dados implementada | 43 |
| 5.2. Criação de Utilizadores da Base de Dados | 45 |
| 5.3. Povoamento da Base de Dados | 46 |
| 5.4. Cálculo do Espaço da Base de Dados | 47 |
| 5.4.1. Detalhamento por Entidade | 47 |
| 5.4.2. Resumo e Projeção de Crescimento | 48 |
| 5.5. Definição e Caracterização de Vistas de Utilização em SQL | 49 |

| | | |
|--|---|-----------|
| 5.6. | Tradução das Interrogações do Utilizador para SQL | 50 |
| 5.6.1. | Listagem de viagens | 50 |
| 5.6.2. | Identificação de viagens ativas para atualização de estado | 50 |
| 5.6.3. | Validação de estado contratual de tripulante, cargo e estado (exemplo ID = 1) | 50 |
| 5.6.4. | Histórico operacional de tripulante (exemplo ID=1) | 51 |
| 5.6.5. | Relatório diário de lucro da viagem com base nas receitas e despesas | 51 |
| 5.6.6. | Estatística semanal de capturas por zona e espécie | 52 |
| 5.6.7. | Auditoria Integral de Expedição (exemplo ID = 1) | 52 |
| 5.7. | Indexação do Sistema de Dados | 53 |
| 5.8. | Implementação de procedimentos, funções e gatilhos | 54 |
| 5.8.1. | Procedimentos | 54 |
| 5.8.2. | Funções | 58 |
| 5.8.3. | Gatilhos | 60 |
| 6. | Conclusões e Trabalho Futuro | 62 |
| Bibliografia | 63 | |
| Lista de Siglas e Acrónimos | 64 | |
| Anexos | 65 | |
| [I] | Diagrama de Gantt | 65 |
| [II] | Documento para Requisitos | 65 |
| [III] | Modelo Conceptual | 65 |
| [IV] | Modelo Lógico | 65 |
| [V] | Expressões de AR - ReLaX | 65 |
| [VI] | DDL - Definição de Dados | 65 |
| [VII] | DCL - Controlo de Dados | 65 |
| [VIII] | DML - Povoamento | 65 |
| [IX] | DQL - Interrogações | 65 |
| [X] | Procedimentos Armazenados | 65 |
| [XI] | Funções SQL | 65 |
| [XII] | Gatilhos (Triggers) | 66 |
| [XIII] | Repositório GitHub | 66 |

Índice de Figuras

| | | |
|-------------|--|----|
| Figura 1.1 | Diagrama de Gantt | 6 |
| Figura 3.1 | Diagrama ER Conceptual | 23 |
| Figura 4.1 | Diagrama ER Lógico após a aplicação da primeira regra de derivação | 27 |
| Figura 4.2 | Diagrama ER Lógico após a aplicação da segunda regra de derivação | 28 |
| Figura 4.3 | Diagrama ER Lógico após a aplicação da terceira regra de derivação | 30 |
| Figura 4.4 | Representação gráfica da expressão em Álgebra Relacional 1 | 32 |
| Figura 4.5 | Representação gráfica da expressão em Álgebra Relacional 1 | 33 |
| Figura 4.6 | Representação gráfica da expressão em Álgebra Relacional 3 | 34 |
| Figura 4.7 | Representação gráfica da expressão em Álgebra Relacional 4 | 36 |
| Figura 4.8 | Representação gráfica da expressão em Álgebra Relacional 5 | 37 |
| Figura 4.9 | Representação gráfica da expressão em Álgebra Relacional 6 | 39 |
| Figura 4.10 | Representação gráfica da expressão em Álgebra Relacional 7 | 41 |

Índice de Tabelas

| | |
|--|----|
| Tabela 2.1 Requisitos de Descrição (RD) | 8 |
| Tabela 2.2 Requisitos de Exploração (RE) | 10 |
| Tabela 2.3 Requisitos de Administração (RA) | 11 |
| Tabela 3.1 Identificação e Caracterização dos Atributos da Entidade <i>Viagem</i> | 18 |
| Tabela 3.2 Identificação e Caracterização dos Atributos da Entidade <i>Tripulante</i> | 19 |
| Tabela 3.3 Identificação e Caracterização dos Atributos da Entidade <i>Embarcação</i> | 20 |
| Tabela 3.4 Identificação e Caracterização dos Atributos da Entidade <i>Financeiro</i> | 20 |
| Tabela 3.5 Identificação e Caracterização dos Atributos da Entidade <i>ZonaPesca</i> | 21 |
| Tabela 3.6 Identificação e Caracterização dos Atributos da Entidade <i>Captura</i> | 21 |
| Tabela 3.7 Identificação e Caracterização dos Atributos do Relacionamento <i>Participa</i> | 22 |
| Tabela 5.1 Estimativa de volume: Viagem | 47 |
| Tabela 5.2 Estimativa de volume: Tripulante | 47 |
| Tabela 5.3 Estimativa de volume: Embarcação | 47 |
| Tabela 5.4 Estimativa de volume: Financeiro | 47 |
| Tabela 5.5 Estimativa de volume: Captura | 47 |
| Tabela 5.6 Estimativa de volume: ZonaPesca | 47 |
| Tabela 5.7 Estimativa de volume: TripulanteViagem | 48 |
| Tabela 5.8 Estimativa de volume: Tabelas Auxiliares | 48 |
| Tabela 5.9 Cálculo do Espaço de Armazenamento da BD | 48 |

1. Definição do Sistema

1.1. Contexto de aplicação

O Grupo Portus, S.A. é uma holding portuguesa com forte presença no setor marítimo, abrangendo áreas que vão desde a pesca comercial em alto mar à gestão portuária e logística marítima, passando ainda pela exportação e transformação de pescado. A sede localiza-se na Torreira em Aveiro, uma zona com longa tradição ligada ao mar, e o grupo agrupa várias empresas do ramo, entre as quais se destaca a PortusPesca – Operações Marítimas, Lda. (PPOM), responsável pela coordenação das expedições marítimas em alto mar e pela gestão diária da frota pesqueira.

A PPOM opera um conjunto diversificado de embarcações, desde navios de média dimensão até unidades industriais capazes de processar o pescado a bordo. As suas atividades concentram-se na captura de espécies com elevado valor comercial, como o atum, o carapau, a pescada e o espadarte, sempre em conformidade com as normas da Política Comum das Pescas da União Europeia e outras convenções internacionais sobre sustentabilidade marinha.

A empresa integra equipas de tripulantes experientes especializados nas diversas tarefas associadas ao esforço de pesca. Estes profissionais são responsáveis não só pela condução técnica da embarcação, mas também pela monitorização das capturas, execução de manobras, cumprimento das normas de segurança e reporte das informações essenciais para o acompanhamento das viagens. A diversidade de funções e perfis dentro das tripulações reflete a complexidade das operações marítimas conduzidas pela frota.

1.2. Motivação e Objetivos

A PortusPesca tem enfrentado um crescimento significativo na complexidade e volume das suas operações marítimas, resultado direto da expansão da atividade do Grupo Portus, S.A. e da crescente exigência regulatória associada ao setor das pescas. Embora a empresa possua um grande conhecimento e experiência, os métodos utilizados para registar e gerir a informação operacional, relatórios dispersos, folhas de cálculo independentes e anotações físicas, tornaram-se insuficientes para acompanhar o ritmo atual das expedições em alto mar. Com o aumento do número de viagens, da diversidade das operações e da necessidade de rastreabilidade rigorosa, começaram a surgir dificuldades na consolidação de informação, na consulta histórica e na avaliação de desempenho. Situações como dados duplicados, registo incompletos ou informação perdida têm dificultado o trabalho dos responsáveis pela frota, comprometendo a eficiência operacional e criando receio de que a empresa não consiga manter os padrões de qualidade, fiabilidade e transparência que caracterizam o grupo.

Reconhecendo o impacto destes problemas no funcionamento diário da organização, a equipa de engenharia informática – recentemente reforçada – percebeu que a solução passava necessariamente pela modernização tecnológica. A implementação de um Sistema de Gestão de Base de Dados permitiria substituir processos fragmentados por um modelo integrado, capaz de organizar, armazenar e disponibilizar informação de forma coerente e eficaz. Este projeto procura responder a essas necessidades, dotando a PortusPesca de uma ferramenta que acompanhe o crescimento da empresa, garanta maior rigor nos registo e apoie decisões estratégicas fundamentais para a sua competitividade no setor.

Por conseguinte, os principais objetivos que o Grupo Portus pretende alcançar com o desenvolvimento deste sistema são os seguintes:

- **Centralização da Informação:** A centralização dos registo num único sistema estruturado permitirá um acesso rápido, organizado e uniforme à informação relevante. A consolidação dos dados facilita o acompanhamento das operações, promove a consistência dos registo e elimina a dispersão por múltiplos suportes.
- **Escalabilidade:** À medida que o volume de expedições e de informação operacional aumenta, é essencial dispor de um sistema capaz de crescer juntamente com a empresa. O sistema deve acomodar novas operações, novos tipos de dados e futuras integrações tecnológicas sem comprometer o desempenho ou a fiabilidade.
- **Precisão e Consistência:** Registos físicos e folhas de cálculo isoladas são suscetíveis a erros humanos, perda de informação e duplicação. A adoção de um sistema estruturado aumenta a fiabilidade dos dados e garante uniformidade, evitando confusões e imprecisões que possam comprometer decisões.
- **Segurança da Informação:** Um sistema centralizado permite implementar controlos de acesso adequados, garantindo que apenas utilizadores autorizados podem consultar ou alterar dados sensíveis relacionados com as expedições. Isto reduz riscos de perda, adulteração ou partilha indevida de informação crítica.
- **Transparência e Rastreabilidade:** Um sistema digital facilita o cumprimento das normas europeias e internacionais relativas à rastreabilidade das atividades pesqueiras. Registos rigorosos tornam mais simples demonstrar conformidade e responder a auditorias ou exigências legais.
- **Otimização Operacional:** Ao substituir tarefas manuais e processos descoordenados por um sistema digital, torna-se possível automatizar operações de armazenamento, consulta e atualização de dados. Isso liberta tempo dos responsáveis operacionais, melhora a agilidade da gestão e reduz atrasos na análise das expedições.
- **Gestão financeira informada:** A consolidação da informação permite acompanhar com mais clareza os recursos envolvidos nas operações e os custos associados, evitando omissões e facilitando análises económicas mais rigorosas.

1.3. Análise da Viabilidade do Processo

A viabilidade de um projeto de desenvolvimento de um Sistema de Gestão de Base de Dados depende da capacidade de compreender e satisfazer as necessidades operacionais e estratégicas da organização. Isto requer um planeamento cuidadoso e eficiente para garantir a entrega de uma solução fiável e de alta qualidade. Ao seguir uma abordagem metódica, o projeto pode maximizar as suas hipóteses de sucesso ao responder de forma eficaz às exigências de rastreabilidade, sustentabilidade e eficiência. A PortusPesca considerou que o desenvolvimento de um SGBD para o Sistema de Catalogação Virtual de Viagens de Pesca é bastante viável, pois este garantirá uma série de benefícios para a empresa, nomeadamente:

- **Melhor gestão de tripulação:** Maior facilidade em identificar quem está embarcado, disponível ou em formação, permitindo alocação rápida de capitães e marinheiros às viagens e acompanhamento do tempo embarcado e funções exercidas.
- **Resolver a sobrecarga de operações:** Com processos mais ágeis e relatórios automáticos, equipas conseguem fechar viagens, registar capturas e preparar novas expedições mais rapidamente, aumentando produtividade e resposta à procura sazonal.
- **Melhor gestão operacional da frota:** Visão centralizada do estado das embarcações, planeamento de expedições e prevenção de conflitos (não abrir viagem com outra ativa), aumentando eficiência na rotação de navios.
- **Segurança e conformidade acrescidas:** Controlo de acessos por perfil, rastreabilidade integral de capturas por viagem e zona de pesca, integridade dos registos e alinhamento com normas de sustentabilidade e auditoria.
- **Melhoria da qualidade de serviço e bem-estar:** Redução de buscas manuais e informação dispersa; dados de viagens, capturas e zonas ficam acessíveis e confiáveis, diminuindo carga administrativa e melhorando o fluxo de trabalho a bordo e em terra.

Tendo em conta estes fatores, fica evidente que o projeto de implementação do Sistema de Gestão de Base de Dados é altamente viável e proporcionará benefícios significativos à PortusPesca, sobretudo em termos de organização e centralização da informação operacional, melhoria da eficiência dos processos, segurança e rastreabilidade dos dados, e, a longo prazo, apoio ao crescimento sustentável e à competitividade da empresa no setor marítimo.

1.4. Recursos e Equipa de Trabalho

1.4.1. Recursos Humanos

- Funcionários da Empresa (PPOM e Grupo Portus, S.A.): Gestores operacionais, responsáveis técnicos da frota, capitães de embarcação, contabilistas e pessoal administrativo envolvidos na recolha e validação de dados das viagens de pesca;
- Equipa de Desenvolvimento: Grupo responsável pela análise, modelação, implementação e documentação do Sistema de Catalogação Virtual de Viagens de Pesca;
- Consultores Externos: Especialistas em sistemas de informação marítimos e bases de dados relacionais, com função de apoio técnico e validação do modelo final.

1.4.2. Recursos Físicos

- Computadores portáteis ou de secretária, com capacidade para execução de software de modelação e SGBD;
- Servidor central (local ou virtual), destinado à instalação e manutenção da base de dados e armazenamento de backups;
- Conexão à Internet, para sincronização de dados, partilha de ficheiros e comunicação entre a equipa de desenvolvimento;
- Periféricos complementares, como impressoras, projetores e dispositivos de armazenamento externo (USB / SSD) para cópias de segurança.

1.4.3. Recursos Digitais

- Sistemas Operativos: Windows 11 e Linux (Ubuntu e Arch Linux);
- Plataformas de Colaboração e Armazenamento: Google Drive e Google Sheets (partilha de ficheiros e registo de progresso);
- Controlo de Versões: Git e GitHub (gestão de versões do modelo e scripts SQL);
- Ferramentas de Documentação: LaTeX e Typst (produção de relatórios técnicos e documentação final);
- Modelação Conceptual e Lógica: BR-Modelo e MYSQL

1.4.4. Equipa de Trabalho

• Pessoal Interno (PPOM):

- Gestor Operacional da Frota: Responsável pela coordenação das expedições e pela validação dos dados de viagens e tripulações;
- Capitães e Oficiais das Embarcações: Fornecimento de dados sobre embarcações, zonas de pesca, capturas e condições marítimas;
- Departamento Administrativo-Financeiro: Apoio na disponibilização de informação sobre custos operacionais, receitas e rentabilidade das viagens;
- Departamento de TI do Grupo Portus: Apoio técnico na instalação do servidor e gestão de acessos ao sistema.

• Pessoal Externo (Equipa de Desenvolvimento do Grupo Portus):

- **Guilherme Baptista:** Análise da Viabilidade do Projeto, Levantamento e Validação de Requisitos, Modelação Conceptual e Lógica;
- **Salvador Coelho:** Levantamento de Requisitos, Modelação Conceptual, Normalização de Dados e Documentação Técnica;
- **Matteo Murari:** Modelação Lógica, Implementação Física da Base de Dados, Criação de Vistas e Interrogações SQL;
- **Rui Rebelo:** Análise da Viabilidade do Projeto, Implementação de Funções, Procedimentos e Gatilhos SQL, e Validação Final do Sistema.

1.5. Plano de Execução do Projeto

Para assegurar uma implementação eficiente e o cumprimento rigoroso dos prazos estipulados para a entrega do Sistema de Catalogação Virtual de Viagens de Pesca, a equipa de desenvolvimento estabeleceu um cronograma de trabalho detalhado. Este plano resultou das reuniões iniciais com a direção da Portus-Pesca e da análise da complexidade técnica dos requisitos levantados.

A execução do projeto foi estruturada em fases sequenciais, permitindo a validação intermédia de cada etapa por parte dos stakeholders. O planeamento temporal, bem como a distribuição de tarefas pelos membros da equipa (identificados na secção 1.4.4), encontra-se sistematizado no Diagrama de Gantt (ver Figura 1.1).

As principais fases definidas para o projeto são:

1. **Definição e Planeamento:** Contextualização do sistema, análise de viabilidade e definição da equipa;
2. **Levantamento e Análise de Requisitos:** Recolha de dados junto das tripulações e gestão, e estruturação dos requisitos funcionais;
3. **Modelação Conceptual:** Criação do Modelo Entidade-Relacionamento (ER) abstrato;
4. **Modelação Lógica:** Conversão para modelo relacional e normalização de dados;
5. **Implementação Física:** Construção da base de dados em SQL, criação de vistas, procedimentos e validação com interrogações reais.

Nota: Consultar anexo [I] Diagrama de Gantt.

Diagrama de Gantt

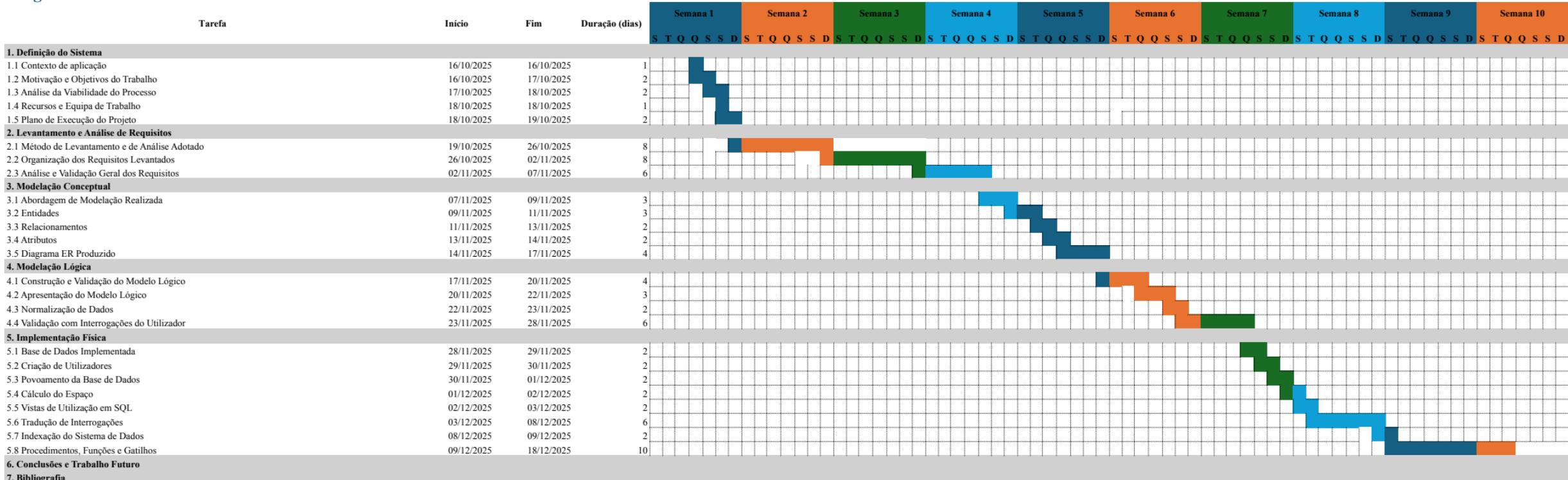


Figura 1.1: Diagrama de Gantt

2. Levantamento e Análise de Requisitos

2.1. Método de Levantamento e de Análise de Requisitos Adotado

O processo de levantamento e análise de requisitos do Sistema de Catalogação Virtual de Viagens de Pesca iniciou-se com uma reunião entre a equipa de desenvolvimento e os responsáveis operacionais da PPOM, com o objetivo de definir as estratégias de recolha de informação mais adequadas ao contexto da empresa e à complexidade dos seus processos marítimos.

Com base nas discussões realizadas, a equipa decidiu adotar uma abordagem mista, combinando métodos qualitativos e documentais, de forma a garantir a recolha de dados completos e fiáveis sobre o funcionamento das expedições, as necessidades dos utilizadores e os objetivos do sistema.

Das várias estratégias avaliadas, foram selecionadas as seguintes:

- **Reuniões** presenciais e online com os gestores da frota, capitães de embarcação e responsáveis administrativos da PPOM, com o propósito de identificar e compreender os diferentes processos envolvidos nas expedições marítimas — desde o planeamento da viagem até ao registo das capturas e contabilização dos custos;
- **Entrevistas** individuais e coletivas com elementos das equipas operacionais, de modo a clarificar fluxos de trabalho, necessidades específicas de informação e eventuais problemas nos métodos atuais de recolha e armazenamento de dados;
- **Emails**, utilizados para esclarecimento de dúvidas pontuais e recolha de contributos adicionais, especialmente de colaboradores que se encontravam embarcados ou ausentes durante o período das reuniões;
- **Análise documental** de relatórios de expedições passadas, registos de captura, folhas de cálculo de custos e receitas, bem como documentação interna relativa à frota, tripulações e zonas de pesca. Esta análise permitiu identificar a estrutura de dados existente e detetar lacunas na consistência e rastreabilidade da informação;
- **Consulta de legislação e normas técnicas**, nomeadamente regulamentos da Política Comum das Pescas da União Europeia e diretrizes nacionais sobre rastreabilidade e sustentabilidade da atividade pesqueira, garantindo que o sistema proposto cumpra as obrigações legais aplicáveis.

Em síntese, a metodologia adotada permitiu reunir um conjunto sólido de informações, que servirão de base para a definição dos requisitos funcionais e não funcionais do sistema, assegurando que a solução final responda de forma eficaz às necessidades operacionais e estratégicas da PPOM.

2.2. Organização dos Requisitos Levantados

O processo de levantamento de requisitos, realizado através de reuniões com os stakeholders e análise documental, resultou num vasto conjunto de necessidades operacionais e estratégicas. De forma a facilitar a modelação e a implementação do sistema, e seguindo rigorosamente a metodologia adotada na unidade curricular, os requisitos foram organizados em três classes distintas:

- **Requisitos de Descrição (RD)**: Definem a estrutura dos objetos de dados, os seus domínios e restrições de integridade;
- **Requisitos de Exploração (RE)**: Estabelecem as necessidades de consulta, relatórios analíticos e processamento de informação;
- **Requisitos de Administração (RA)**: Determinam as regras de controlo de acesso, segurança e consistência do sistema.

2.2.1. Requisitos de Descrição (RD)

| Nr | Descrição | | | |
|------|--|-------------|--------------------|--------------------|
| | Data e Hora | Área | Fonte | Analista |
| RD1 | Cada viagem deve ter um identificador único e sequencial, representado por um número inteiro. | | | |
| | 2025/10/19 12:30 | Viagens | Equipa | Salvador Coelho |
| RD2 | Uma viagem deve ser registada pelos seguintes atributos: identificador único, porto de origem, porto de destino, descrição, um atributo data de partida, um atributo data de chegada, o nome do atributo “estado” e um atributo derivado saldo calculado a partir da soma de todos os valores do movimento financeiro.. | | | |
| | 2025/10/19 16:45 | Viagens | Gestor Operacional | Guilherme Baptista |
| RD3 | O nome do atributo “estado” deve ter um dos seguintes valores: “Em Curso”, “Finalizada” ou “Cancelada”. | | | |
| | 2025/10/19 17:00 | Viagens | Gestor Operacional | Salvador Coelho |
| RD4 | Todas as viagens “Em Curso” contém pelo menos um tripulante. | | | |
| | 2025/10/20 16:00 | Viagens | Capitães | Guilherme Baptista |
| RD5 | Todas as viagens têm associada uma única embarcação. | | | |
| | 2025/10/20 09:00 | Viagens | Gestor Operacional | Salvador Coelho |
| RD6 | Cada viagem tem pelo menos um movimento financeiro. | | | |
| | 2025/10/20 10:00 | Viagens | Dept. Financeiro | Salvador Coelho |
| RD7 | Todas as viagens têm 0 ou mais zonas de pesca. | | | |
| | 2025/10/21 14:15 | Viagens | Capitães | Guilherme Baptista |
| RD8 | Uma viagem pode registrar 0 ou mais capturas. | | | |
| | 2025/10/21 12:15 | Viagens | Capitães | Salvador Coelho |
| RD9 | Cada tripulante deve ter um identificador único e sequencial, representado por um número inteiro. | | | |
| | 2025/10/22 09:30 | Tripulantes | Dept. RH | Salvador Coelho |
| RD10 | Um tripulante deve ser registado pelos seguintes atributos: identificador único, número de cédula marinha, nome completo, data de nascimento, o atributo composto nacionalidade, salário, contacto de emergência (opcional), o nome do atributo “estado”, email (opcional), um atributo composto “morada” - rua, número, código postal e localidade. | | | |
| | 2025/10/22 14:15 | Tripulantes | Dept. RH | Guilherme Baptista |

| Nr | Descrição | | | |
|-------------|--|-------------|--------------------|--------------------|
| | Data e Hora | Área | Fonte | Analista |
| RD11 | O número de cédula marinha é um código formado por duas letras maiúsculas e uma sequência de números inteiros. | | | |
| | 2025/10/22 11:00 | Tripulantes | Capitães | Salvador Coelho |
| RD12 | O nome do atributo estado pode ser “Ativo” ou “Inativo”. | | | |
| | 2025/10/23 17:45 | Tripulantes | Dept. RH | Salvador Coelho |
| RD13 | O nome do atributo cargo deve ter um dos seguintes valores: “Capitão”, “Imediato”, “Oficial de Navegação”, “Engenheiro”, “Mestre de Redes”, “Maquinista”, “Cozinheiro”, “Marinheiro”. | | | |
| | 2025/10/23 16:45 | Tripulantes | Gestor Operacional | Guilherme Baptista |
| RD14 | Cada tripulante só pode desempenhar um único cargo por viagem. | | | |
| | 2025/10/24 16:00 | Tripulantes | Capitães | Salvador Coelho |
| RD15 | Todos os tripulantes participam em 0 ou mais viagens. | | | |
| | 2025/10/24 12:15 | Tripulantes | Dept. RH | Guilherme Baptista |
| RD16 | Cada embarcação deve ter um número de matrícula, representado por duas letras maiusculas e uma sequência de 6 números inteiros. | | | |
| | 2025/10/24 13:30 | Embarcações | Gestor Operacional | Guilherme Baptista |
| RD17 | Uma embarcação deve ser registada pelos seguintes atributos: número de matrícula, nome (opcional), tipo, o atributo multivalorado “portos autorizados” e o atributo composto “dimensões” - altura, largura, comprimento e capacidade de carga. | | | |
| | 2025/10/25 15:00 | Embarcações | Dept. TI Portus | Guilherme Baptista |
| RD18 | Uma embarcação pode ter 0 ou mais viagens associadas. | | | |
| | 2025/10/25 11:15 | Embarcações | Gestor Operacional | Salvador Coelho |
| RD19 | Cada movimento financeiro deve ter um identificador único e sequencial, representado por um número inteiro. | | | |
| | 2025/10/25 17:00 | Financeiro | Dept. Financeiro | Salvador Coelho |
| RD20 | Um movimento financeiro deve ser registado pelos seguintes atributos: identificador único, tipo, valor , categoria e descrição(opcional). | | | |
| | 2025/10/26 17:15 | Financeiro | Dept. Financeiro | Salvador Coelho |
| RD21 | O tipo de um movimento financeiro é “receita” ou “despesa” e o seu valor representa-se por um número de vírgula flutuante positivo ou negativo conforme o tipo de movimento. | | | |
| | 2025/10/26 11:30 | Financeiro | Dept. Financeiro | Salvador Coelho |
| RD22 | Todos os movimentos financeiros estão associados a uma única viagem. | | | |
| | 2025/10/27 16:45 | Financeiro | Dept. Financeiro | Salvador Coelho |
| RD23 | Cada zona de pesca deve ter um identificador único e sequencial, representado por um número inteiro. | | | |
| | 2025/10/27 11:45 | Zonas Pesca | Capitães | Guilherme Baptista |
| RD24 | Uma zona de pesca deve ser registada pelos seguintes atributos: identificador único, nome, descrição (opcional) e pelo atributo composto “coordenadas” - latitude e longitude. | | | |
| | 2025/10/27 12:00 | Zonas Pesca | Oficiais | Salvador Coelho |
| RD25 | Cada zona de pesca está associada a 0 ou mais viagens. | | | |
| | 2025/10/28 16:45 | Zonas Pesca | Capitães | Salvador Coelho |
| RD26 | Cada zona de pesca está associada a 0 ou mais capturas. | | | |
| | 2025/10/28 15:00 | Zonas Pesca | Oficiais | Guilherme Baptista |
| RD27 | Cada captura deve ter um identificador único e sequencial, representado por um número inteiro. | | | |

| Nr | Descrição | | | |
|------|--|----------|----------|--------------------|
| | Data e Hora | Área | Fonte | Analista |
| | 2025/10/29 13:00 | Capturas | Capitães | Guilherme Baptista |
| RD28 | Uma captura deve ser registada pelos seguintes atributos: identificador único, nome comum, nome científico (opcional), quantidade. | | | |
| | 2025/10/29 17:30 | Capturas | Oficiais | Salvador Coelho |
| RD29 | Cada captura pertence a uma única viagem. | | | |
| | 2025/10/29 16:30 | Capturas | Capitães | Guilherme Baptista |
| RD30 | Uma captura está associada a uma única zona de pesca. | | | |
| | 2025/10/30 17:00 | Capturas | Oficiais | Guilherme Baptista |

2.2.2. Requisitos de Exploração (RE)

| Nr | Descrição | | | |
|------|---|-------------|--------------------|--------------------|
| | Data e Hora | Área | Fonte | Analista |
| RE1 | Quando uma viagem é finalizada ou cancelada o seu estado é atualizado respetivamente, assim como a data de chegada e é feita a desassociação da tripulação da viagem. | | | |
| | 2025/10/30 15:45 | Viagens | Gestor Operacional | Salvador Coelho |
| RE2 | Deve ser possível obter uma listagem geral das Viagens com os seus atributos operacionais essenciais: identificador, portos de origem e destino, e estado atual. | | | |
| | 2025/10/31 17:30 | Viagens | Dept. TI Portus | Guilherme Baptista |
| RE3 | O sistema deve permitir pesquisar viagens com base em atributos descritivos, tais como porto de origem, porto de destino, descrição, espécies capturadas ou zonas de pesca visitadas. | | | |
| | 2025/10/31 11:15 | Viagens | Gestor Operacional | Guilherme Baptista |
| RE4 | Dado o identificador de um tripulante, deve ser possível aceder a todas as viagens em que este esteve envolvido, bem como o cargo desempenhado em cada uma. | | | |
| | 2025/10/31 15:00 | Tripulantes | Dept. RH | Salvador Coelho |
| RE5 | Quando um tripulante deixa de exercer a sua função (é promovido ou demitido) o seu atributo “estado” e o “cargo” devem ser atualizados respetivamente. | | | |
| | 2025/11/01 15:15 | Tripulantes | Dept. RH | Salvador Coelho |
| RE6 | Dado o número de matrícula de uma embarcação, o sistema deve permitir consultar todas as viagens realizadas pela mesma, incluindo datas, estados e destino. | | | |
| | 2025/11/01 16:45 | Embarcações | Gestor Operacional | Guilherme Baptista |
| RE7 | Deve ser possível obter todos os movimentos financeiros de uma viagem, incluindo receitas, despesas e categorias. | | | |
| | 2025/11/01 10:45 | Financeiro | Dept. Financeiro | Salvador Coelho |
| RE8 | É permitido pelo sistema obter o custo total de uma viagem, através da soma de todas as despesas relativas à mesma. Salários de tripulantes não estão incluídos. | | | |
| | 2025/11/02 15:45 | Financeiro | Dept. Financeiro | Guilherme Baptista |
| RE9 | É permitido pelo sistema obter o rendimento total de uma viagem, através da soma de todas as receitas relativas à mesma. | | | |
| | 2025/11/02 10:30 | Financeiro | Dept. Financeiro | Salvador Coelho |
| RE10 | É permitido pelo sistema obter o Saldo de uma viagem, através da subtração do valor obtido em RE9 (rendimento) pelo valor obtido em RE8 (custo) | | | |
| | 2025/11/03 13:30 | Financeiro | Dept. Financeiro | Guilherme Baptista |

| Nr | Descrição | | | |
|------|---|-------------|--------------------|--------------------|
| | Data e Hora | Área | Fonte | Analista |
| RE11 | No encerramento de cada dia, o sistema deve gerar automaticamente um relatório contendo todos os movimentos financeiros registados neste dia, agrupados por viagem, incluindo somatórios de receitas, despesas e Saldo diário acumulado. | | | |
| | 2025/11/03 10:30 | Financeiro | Dept. Financeiro | Guilherme Baptista |
| RE12 | No encerramento de cada dia, o sistema deverá gerar um relatório para cada viagem que inclua as novas capturas, zonas de pesca associadas e alterações na tripulação registadas nesse período. | | | |
| | 2025/11/03 11:45 | Viagens | Gestor Operacional | Guilherme Baptista |
| RE13 | No encerramento de cada semana, o sistema deverá gerar um relatório com informações e estatísticas relativas a nomes comuns e quantidades de capturas por zona de pesca. | | | |
| | 2025/11/04 10:15 | Zonas Pesca | Capitães | Salvador Coelho |
| RE14 | O sistema deve permitir a extração de um dossier completo de uma viagem específica para efeitos de fiscalização e auditoria. Este relatório deve cruzar, numa única vista, os dados logísticos da embarcação, a lista detalhada dos tripulantes embarcados (incluindo as suas credenciais) e o registo discriminado de todas as capturas e respetivas zonas de pesca. | | | |
| | 2025/11/04 13:15 | Viagens | Gestor Operacional | Guilherme Baptista |

2.2.3. Requisitos de Administração (RA)

| Nr | Descrição | | | |
|-----|--|---------------|--------------------|--------------------|
| | Data e Hora | Área | Fonte | Analista |
| RA1 | O sistema deve estar operacional 24 horas por dia, 7 dias por semana, sem interrupções. | | | |
| | 2025/11/05 09:15 | Administração | Dept. TI Portus | Salvador Coelho |
| RA2 | O sistema deve garantir que apenas os tripulantes associados a uma determinada viagem e que desempenham cargos de responsabilidade (Capitão ou Imediato) podem alterar os dados relativos à mesma. | | | |
| | 2025/11/05 12:30 | Tripulantes | Gestor Operacional | Guilherme Baptista |
| RA3 | O sistema deve garantir que os tripulantes sem cargos de responsabilidade não podem aceder a dados do movimento financeiro. | | | |
| | 2025/11/05 16:15 | Financeiro | Dept. Financeiro | Guilherme Baptista |
| RA4 | O sistema deve oferecer acesso aos dados de uma viagem em curso apenas aos tripulantes a ela associados e à gerência da empresa. | | | |
| | 2025/11/06 10:00 | Viagens | Dept. TI Portus | Salvador Coelho |
| RA5 | Os administradores devem ser capazes de consultar, inserir, atualizar e remover todas as informações armazenadas na base de dados. | | | |
| | 2025/11/06 12:00 | Administração | Dept. TI Portus | Salvador Coelho |

Nota: Consultar anexo [II] Documentos de Levantamento de Requisitos.

2.3. Análise e Validação Geral dos Requisitos

Após o levantamento e a organização dos requisitos descritos no subcapítulo 2.2, procedeu-se à análise e validação geral dos mesmos, garantindo a sua coerência, completude e aderência às necessidades operacionais e estratégicas da PortusPesca.

Inicialmente, a equipa de desenvolvimento realizou uma triagem interna dos requisitos, verificando:

- Consistência lógica: os atributos, relacionamentos e regras definidos para cada entidade foram confrontados entre si, de forma a evitar contradições ou redundâncias;
- Completude: assegurou-se que todas as operações críticas da frota e da gestão de viagens estavam contempladas, incluindo tripulação, embarcações, capturas, zonas de pesca e movimentos financeiros;
- Viabilidade técnica: cada requisito foi avaliado quanto à sua implementação prática no Sistema de Gestão de Base de Dados, considerando os recursos humanos, físicos e digitais disponíveis.

Posteriormente, a lista de requisitos organizada nas categorias RD (Descrição), RE (Exploração) e RA (Administração) foi submetida à revisão com os *stakeholders* da PortusPesca. Para tal, realizou-se uma reunião presencial com a gerência, durante a qual:

- Foram discutidos os requisitos críticos e os fluxos de dados essenciais;
- Alguns requisitos foram ajustados para refletir melhor a realidade operacional da frota, incluindo modificações na forma de registo de capturas e no acompanhamento de movimentos financeiros;
- Esclarecimentos adicionais foram fornecidos sobre regras de acesso, atualização de estados de viagens e vinculação de tripulantes.

Após esta revisão e ajustes finais, a lista de requisitos foi formalmente aprovada pela gerência da Portus-Pesca, permitindo que o projeto avançasse para a fase de modelação conceptual e lógica, com base em informações precisas e validadas, garantindo maior fiabilidade no desenvolvimento do Sistema de Catalogação Virtual de Viagens de Pesca.

3. Modelação Conceptual

3.1. Apresentação da Abordagem de Modelação Realizada

Após a consolidação e validação dos requisitos do sistema, iniciou-se o processo de abstração necessário para a implementação de um Sistema de Gestão de Base de Dados para a PortusPesca.

A metodologia baseou-se na análise semântica dos Requisitos de Descrição (RD) e de Exploração (RE), identificando nomes que deram origem às entidades e verbos que indicaram os relacionamentos. Em seguida, foram caracterizados os atributos das entidades e relacionamentos, bem como as respetivas cardinalidades.

Para formalizar este modelo, utilizou-se um diagrama Entidade-Relacionamento (ER) no Modelo Entidade-Relacionamento (MER), seguindo a notação clássica de Peter Chen, desenvolvido com a ferramenta BR Modelo.

3.2. Identificação e Caracterização das Entidades

Avaliando o domínio funcional da PortusPesca – Operações Marítimas (PPOM), identificaram-se como entidades fundamentais do sistema as seguintes: Viagem, Tripulante, Embarcação, Financeiro, ZonaPesca e Captura.

Para cumprir os requisitos levantados, cada entidade é caracterizada pelo seu papel no domínio, sinónimos relevantes e atributos principais:

- **Viagem:** Representa cada expedição marítima realizada por uma embarcação, envolvendo tripulação, capturas, zonas de pesca visitadas e movimentos financeiros.
 - **Sinónimos:** Expedição, Navegação, Percurso.
 - **Atributos:** ID, Porto de Origem, Porto de Destino, Descrição, Data de Partida, Data de Chegada, Estado, Saldo (derivado).
- **Tripulante:** Representa um membro da equipa marítima que desempenha um cargo específico e contribui para a realização de uma viagem.
 - **Sinónimos:** Navegante, Marujo, Embarcadiço.
 - **Atributos:** ID, Nº Cédula Marinha, Nome, Data de Nascimento, Nacionalidade (multivalorado), Salário, Contacto de Emergência, Estado, Email, Morada (Rua, Número, Código Postal, Localidade).
- **Embarcação:** Representa cada embarcação pertencente à frota, utilizada na realização das viagens.
 - **Sinónimos:** Navio, Barco, Nau.
 - **Atributos:** ID, Nome, Tipo, Portos Autorizados (multivalorado), Dimensões (Altura, Largura, Comprimento, Capacidade de Carga).
- **Financeiro:** Representa movimentos financeiros associados a uma viagem, incluindo despesas operacionais e receitas provenientes da venda do pescado.
 - **Sinónimos:** Registo Económico, Numerário, Transação.
 - **Atributos:** ID, Tipo, Valor, Descrição, Categoria.
- **ZonaPesca:** Representa uma área geográfica legalmente definida onde é permitida a captura.
 - **Sinónimos:** Pescadouro, Banco de Pesca, Zona Marítima.
 - **Atributos:** ID, Nome, Descrição, Coordenadas (Latitude, Longitude).
- **Captura:** Representa um registo individual de uma captura realizada durante uma viagem.
 - **Sinónimos:** Pescado, Marisco, Peixe.
 - **Atributos:** ID, Nome Comum, Nome Científico, Quantidade.

3.3. Identificação e Caracterização dos Relacionamentos

Nesta fase da modelação conceptual identificaram-se seis relacionamentos fundamentais entre as entidades do sistema PPOM, dos quais alguns originam entidades associativas devido à natureza N:M da ligação. Cada relacionamento é descrito de forma a clarificar o seu papel no domínio, cardinalidades, atributos e a sua origem nos requisitos funcionais.

- **Relacionamento Tripulante — Viagem**

- **Relacionamento:** Tripulante participa em Viagem
- **Descrição:** Regista que tripulantes participaram numa viagem específica, bem como o cargo exercido por cada um durante a expedição. Esta informação é essencial para determinar a constituição da tripulação, validar que uma viagem em curso cumpre os requisitos mínimos e identificar o papel desempenhado por cada tripulante ao longo da operação marítima.
- **Cardinalidade:** Tripulante (0,n) — Viagem (0,n). Todos os tripulantes participam em 0 ou mais viagens e todas as viagens em curso possuem pelo menos um tripulante. Cada tripulante só pode exercer um único cargo por viagem; viagens finalizadas não mantêm tripulantes com cargos atribuídos.
- **Baseado nos Requisitos:**
 - RD4 - Todas as viagens “em curso” contém pelo menos um tripulante.
 - RD14 - Cada tripulante só pode desempenhar um único cargo por viagem.
 - RD15 - Todos os tripulantes participam em 0 ou mais viagens.
- **Atributos:** Cargo; Data de Entrada; Data de Saída.
- **Entidade Gerada:** Participa (entidade associativa).

- **Relacionamento Embarcação — Viagem**

- **Relacionamento:** Embarcação executa Viagem
- **Descrição:** Regista qual a embarcação utilizada numa viagem específica. Cada viagem deve estar associada a uma única embarcação, garantindo rastreabilidade operacional, conformidade com autorizações portuárias e verificação de requisitos logísticos (capacidade de carga, tipo). Uma embarcação pode, por sua vez, participar em várias viagens ao longo do tempo.
- **Cardinalidade:** Embarcação (1,1) — Viagem (0,n). Uma embarcação pode executar 0 ou mais viagens; cada viagem está obrigatoriamente associada a uma única embarcação.
- **Baseado nos Requisitos:**
 - RD5 - Todas as viagens têm associada uma única embarcação.
 - RD18 - Uma embarcação pode ter 0 ou mais viagens associadas.
- **Atributos:** Nenhum.
- **Entidade Gerada:** Não gera entidade.

- **Relacionamento Financeiro — Viagem**

- **Relacionamento:** Movimento Financeiro associado à Viagem
- **Descrição:** Permite registar e organizar todos os movimentos financeiros relacionados com uma viagem (despesas e receitas), assegurando que os valores económicos são corretamente atribuídos à viagem onde ocorreram.
- **Cardinalidade:** Financeiro (1,n) — Viagem (1,1). Cada movimento financeiro está associado a uma única viagem; uma viagem pode ter vários movimentos financeiros. Todas as viagens devem possuir pelo menos um movimento financeiro.
- **Baseado nos Requisitos:**
 - RD6 - Cada viagem tem pelo menos um movimento financeiro.
 - RD22 - Todos os movimentos financeiros estão associados a uma única viagem.
- **Atributos:** Nenhum.
- **Entidade Gerada:** Não gera entidade.

- **Relacionamento Viagem — ZonaPesca**

- **Relacionamento:** Viagem ocorre em Zona de Pesca
- **Descrição:** Regista as zonas de pesca visitadas durante a realização de uma viagem. Como uma viagem pode deslocar-se por diferentes áreas marítimas, é necessário identificar em que zonas ocorreu a atividade pesqueira.
- **Cardinalidade:** Viagem (0,n) — ZonaPesca (0,n). Uma viagem pode ocorrer em zero ou mais zonas de pesca; uma zona de pesca pode ser visitada por zero ou mais viagens.
- **Baseado nos Requisitos:**
 - RD7 - Todas as viagens têm 0 ou mais zonas de pesca.
 - RD25 - Cada zona de pesca está associada a 0 ou mais viagens.
- **Atributos:** Nenhum.
- **Entidade Gerada:** ViagemZonaPesca

- **Relacionamento Captura — Viagem**

- **Relacionamento:** Captura pertence à Viagem
- **Descrição:** Identifica todas as capturas realizadas numa determinada viagem e, inversamente, a que viagem pertence cada registo de captura. Existem viagens que podem não registar capturas.
- **Cardinalidade:** Captura (0,n) — Viagem (1,1). Cada captura pertence obrigatoriamente a uma viagem; uma viagem pode ter zero ou mais capturas.
- **Baseado nos Requisitos:**
 - RD8 - Uma viagem pode registrar 0 ou mais capturas.
 - RD29 - Cada captura pertence a uma única viagem.
- **Atributos:** Nenhum.
- **Entidade Gerada:** Não gera entidade.

- **Relacionamento Captura — ZonaPesca**

- **Relacionamento:** Captura pescada em ZonaPesca
- **Descrição:** Identifica em que zona de pesca ocorreu cada captura efetuada durante uma viagem. Dado que espécies distintas podem ser obtidas em áreas diferentes e que as zonas têm regulamentações específicas, é essencial registrar a zona associada a cada captura.
- **Cardinalidade:** Captura (0,n) — ZonaPesca (1,1). Cada captura é obrigatoriamente associada a uma zona de pesca; uma zona de pesca pode ter 0 ou mais capturas.
- **Baseado nos Requisitos:**
 - RD26 - Cada zona de pesca está associada a 0 ou mais capturas.
 - RD30 - Uma captura está associada a uma única zona de pesca.
- **Atributos:** Nenhum.
- **Entidade Gerada:** Não gera entidade.

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

Entidade: Viagem

Tabela 3.1: Identificação e Caracterização dos Atributos da Entidade *Viagem*

| Atributos | Nulo | Composto | Derivado | Multivalorado | Candidato |
|--------------|------|----------|----------|---------------|-----------|
| ID | Não | Não | Não | Não | Sim |
| PortoOrigem | Não | Não | Não | Não | Não |
| PortoDestino | Não | Não | Não | Não | Não |
| Descrição | Não | Não | Não | Não | Não |
| DataPartida | Não | Não | Não | Não | Não |
| DataChegada | Não | Não | Não | Não | Não |
| Estado | Não | Não | Não | Não | Não |
| Saldo | Não | Não | Sim | Não | Não |

Caracterização dos atributos da entidade: Viagem

- **ID:** É único e identifica cada viagem realizada pela frota da PortusPesca;
- **PortoOrigem:** Porto de partida da viagem; identifica o local de onde a embarcação inicia a expedição;
- **PortoDestino:** Porto de chegada da viagem; identifica o destino final da expedição;
- **Descrição:** Texto livre que descreve a viagem, podendo incluir informações como tipo de pesca, objetivos ou observações;
- **DataPartida:** Data e hora de início da viagem;
- **DataChegada:** Data e hora de término da viagem;
- **Estado:** Indica a situação atual da viagem, podendo assumir os valores “Em Curso”, “Finalizada” ou “Cancelada”;
- **Saldo:** Valor derivado que indica o saldo de uma viagem; calculado automaticamente pelo sistema a partir da soma de todos os valores da entidade Financeiro.

Entidade: *Tripulante*

Tabela 3.2: Identificação e Caracterização dos Atributos da Entidade *Tripulante*

| Atributos | Nulo | Composto | Derivado | Multivalorado | Candidato |
|--------------------|------|----------|----------|---------------|-----------|
| ID | Não | Não | Não | Não | Sim |
| NºCédulaMarinha | Não | Não | Não | Não | Sim |
| Nome | Não | Não | Não | Não | Não |
| DataNascimento | Não | Não | Não | Não | Não |
| Nacionalidade | Não | Não | Não | Sim | Não |
| Salário | Não | Não | Não | Não | Não |
| ContactoEmergência | Sim | Não | Não | Não | Não |
| Estado | Não | Não | Não | Não | Não |
| Email | Sim | Não | Não | Não | Não |
| Morada | Não | Sim | Não | Não | Não |
| Morada.Rua | Não | Não | Não | Não | Não |
| Morada.Numero | Não | Não | Não | Não | Não |
| Morada.CodPostal | Não | Não | Não | Não | Não |
| Morada.Localidade | Não | Não | Não | Não | Não |

Caracterização dos atributos da entidade: *Tripulante*

- **ID:** Identificador único de cada tripulante da frota;
- **NºCédulaMarinha:** É o documento oficial que identifica um profissional do mar, composto por duas letras maiúsculas seguidas de números;
- **Nome:** Nome completo do tripulante;
- **DataNascimento:** Data de nascimento do tripulante;
- **Nacionalidade:** Nacionalidade(s) do tripulante;
- **Salário:** Valor monetário pago ao tripulante;
- **ContactoEmergência:** Número de contacto alternativo para emergências;
- **Estado:** Indica se o tripulante está “Ativo” ou “Inativo”;
- **Email:** Endereço eletrónico do tripulante, pode ser nulo;
- **Morada:** Endereço completo do tripulante, composto por:
 - **Morada.Rua:** Nome da rua;
 - **Morada.Numero:** Número da residência;
 - **Morada.CodPostal:** Código postal;
 - **Morada.Localidade:** Cidade ou localidade de residência.

Entidade: Embarcação

Tabela 3.3: Identificação e Caracterização dos Atributos da Entidade *Embarcação*

| Atributos | Nulo | Composto | Derivado | Multivalorado | Candidato |
|-----------------------|------|----------|----------|---------------|-----------|
| ID | Não | Não | Não | Não | Sim |
| Nome | Sim | Não | Não | Não | Não |
| Tipo | Não | Não | Não | Não | Não |
| PortosAut | Não | Não | Não | Sim | Não |
| Dimensões | Não | Sim | Não | Não | Não |
| Dimensões.Altura | Não | Não | Não | Não | Não |
| Dimensões.Largura | Não | Não | Não | Não | Não |
| Dimensões.Comprimento | Não | Não | Não | Não | Não |
| Dimensões.CapCarga | Não | Não | Não | Não | Não |

Caracterização dos atributos da entidade: Embarcação

- **ID:** Identificador único da embarcação, composto por quatro letras maiúsculas seguidas de seis números;
- **Nome:** Nome da embarcação;
- **Tipo:** O tipo da embarcação pode ser: Arrastão, Traineira, Palangreiro, Atuneiro, Emalhador ou Cerqueiro;
- **PortosAut:** Lista de portos em que a embarcação está autorizada a operar;
- **Dimensões:** Medidas da embarcação, composto por:
 - **Dimensões.Altura:** Altura da embarcação em metros;
 - **Dimensões.Largura:** Largura da embarcação em metros;
 - **Dimensões.Comprimento:** Comprimento da embarcação em metros;
 - **Dimensões.CapCarga:** Capacidade máxima de carga em toneladas.

Entidade: Financeiro

Tabela 3.4: Identificação e Caracterização dos Atributos da Entidade *Financeiro*

| Atributos | Nulo | Composto | Derivado | Multivalorado | Candidato |
|-----------|------|----------|----------|---------------|-----------|
| ID | Não | Não | Não | Não | Sim |
| Tipo | Não | Não | Não | Não | Não |
| Valor | Não | Não | Não | Não | Não |
| Descrição | Sim | Não | Não | Não | Não |
| Categoria | Não | Não | Não | Não | Não |

Caracterização dos atributos da entidade: Financeiro

- **ID:** Identificador único de cada movimento financeiro;
- **Tipo:** Indica se o movimento é “Receita” ou “Despesa”;
- **Valor:** Valor monetário do movimento financeiro;
- **Descrição:** Informação adicional sobre o movimento financeiro;
- **Categoria:** Classificação do movimento financeiro que podem ser: Combustível, Manutenção, Portos, Seguros, Venda ou Outro.

Entidade: ZonaPesca

Tabela 3.5: Identificação e Caracterização dos Atributos da Entidade ZonaPesca

| Atributos | Nulo | Composto | Derivado | Multivalorado | Candidato |
|-----------------------|------|----------|----------|---------------|-----------|
| ID | Não | Não | Não | Não | Sim |
| Nome | Não | Não | Não | Não | Não |
| Descrição | Sim | Não | Não | Não | Não |
| Coordenadas | Não | Sim | Não | Não | Não |
| Coordenadas.Latitude | Não | Não | Não | Não | Não |
| Coordenadas.Longitude | Não | Não | Não | Não | Não |

Caracterização dos atributos da entidade: ZonaPesca

- **ID:** Identificador único da zona de pesca;
- **Nome:** Nome da zona de pesca;
- **Descrição:** Informações adicionais sobre a zona de pesca;
- **Coordenadas:** Localização geográfica da zona, composta por:
 - **Coordenadas.Latitude:** Latitude da zona;
 - **Coordenadas.Longitude:** Longitude da zona.

Entidade: Captura

Tabela 3.6: Identificação e Caracterização dos Atributos da Entidade Captura

| Atributos | Nulo | Composto | Derivado | Multivalorado | Candidato |
|----------------|------|----------|----------|---------------|-----------|
| ID | Não | Não | Não | Não | Sim |
| NomeComum | Não | Não | Não | Não | Não |
| NomeCientífico | Sim | Não | Não | Não | Não |
| Quantidade | Não | Não | Não | Não | Não |

Caracterização dos atributos da entidade: Captura

- **ID:** Identificador único da captura;
- **NomeComum:** Nome comum do peixe ou marisco capturado;
- **NomeCientífico:** Nome científico da espécie; pode ser nulo;
- **Quantidade:** Peso total da captura em quilogramas.

Relacionamento: *Participa*

Tabela 3.7: Identificação e Caracterização dos Atributos do Relacionamento *Participa*

| Atributos | Nulo | Composto | Derivado | Multivalorado | Candidato |
|-------------|------|----------|----------|---------------|-----------|
| Cargo | Não | Não | Não | Não | Não |
| DataEntrada | Não | Não | Não | Não | Não |
| DataSaída | Sim | Não | Não | Não | Não |

Caracterização dos atributos do relacionamento *Participa*:

- **Cargo:** A função do tripulante pode ser: Capitão, Imediato, Oficial de Navegação, Engenheiro, Mestre de Redes, Maquinista, Cozinheiro ou Marinheiro;
- **DataEntrada:** Data em que o tripulante se associa à viagem;
- **DataSaída:** Data de saída do tripulante da viagem; pode ser nula.

3.5. Apresentação e Explicação do Diagrama ER Produzido

Através da análise detalhada dos requisitos definidos para o sistema PPOM, e recorrendo ao BR-Modelo desenvolvemos o diagrama Entidade-Relacionamento apresentado a seguir.

Durante o processo de conceção, optámos por iniciar o desenvolvimento a partir da entidade Viagem, dado o seu papel central na coordenação das operações marítimas. A partir desta entidade, foram integradas gradualmente as restantes entidades.

A entidade Viagem constitui o núcleo estrutural do modelo conceptual, servindo como ponto de convergência para a maioria das operações do sistema. Esta entidade é caracterizada por um conjunto de atributos relevantes, totalizando grau 7.

Relativamente às restantes entidades, Tripulante destaca-se pela sua complexidade, apresentando grau 10 e integrando atributos. Esta estrutura reflete a necessidade de representar detalhadamente os membros da tripulação e as informações essenciais à gestão de recursos humanos no domínio marítimo.

O modelo conceptual integra seis relacionamentos fundamentais, representando as interações entre as entidades. Entre os relacionamentos binários (1:N), destacam-se Executa, ligando Embarcação a Viagem, Associa, conectando Financeiro a Viagem, Pertence, ligando Captura a Viagem, e Pescada, associando Captura a ZonaPesca. Quanto aos relacionamentos N:M, identificam-se Participa, que liga Tripulante a Viagem, e Ocorre, que conecta Viagem a ZonaPesca. Ambos originam entidades associativas: Participa acomoda os atributos Cargo, DataEntrada e DataSaída, enquanto Ocorre estabelece a ligação sem atributos adicionais.

Em conjunto, todas estas entidades, atributos e relacionamentos formam um modelo conceptual completo, robusto e coerente com os requisitos do sistema PPOM, assegurando que as operações marítimas, humanas, geográficas e financeiras são representadas de forma precisa e extensível.

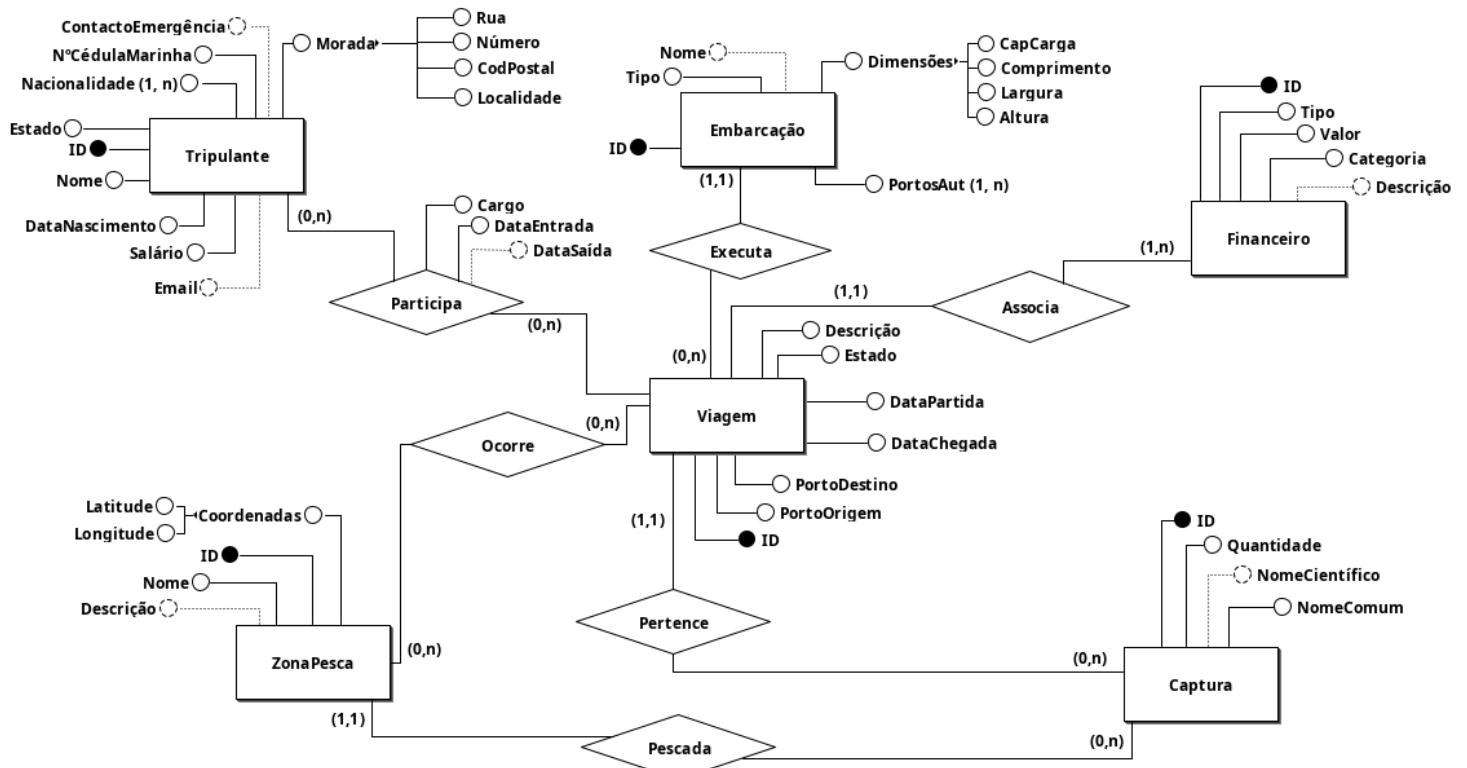


Figura 3.1: Diagrama ER Conceptual

Nota: Consultar anexo [III] Modelo Conceptual.

4. Modelação Lógica

4.1. Construção e Validação do Modelo de Dados Lógico

Num primeiro momento, partiu-se do modelo conceptual, que descreve de forma abstrata os conceitos e relações do sistema. Recorreu-se ao MySQL Workbench para converter esses conceitos em tabelas e estruturas concretas de base de dados, refletindo as entidades, atributos e ligações identificadas no modelo conceptual.

Foram realizados ajustes para otimizar a organização da base de dados e assegurar a sua eficiência e capacidade de expansão, incluindo técnicas de normalização, definição de chaves primárias e estrangeiras, bem como a revisão dos tipos de dados utilizados.

No final deste processo, o modelo lógico constitui uma versão mais precisa e detalhada do sistema, preparada para implementação numa base de dados relacional. Contudo, importa sublinhar que o modelo lógico não é definitivo nem imutável, funcionando como uma etapa intermédia no desenvolvimento de software, sujeita a alterações e melhorias sempre que surjam novos requisitos.

4.2. Apresentação e Explicação do Modelo Lógico Produzido

A construção do modelo lógico baseou-se intrinsecamente no modelo conceptual desenvolvido no capítulo anterior. Para tal, é necessário aplicar as regras de derivação do modelo de dados relacional.

Primeiramente, cada entidade é convertida numa tabela. Deste processo, obtemos as seguintes seis tabelas:

Viagem:

- **Chave primária:**
 - ▶ ID: INT
- **Chaves estrangeiras:**
 - ▶ Embarcação: VARCHAR(8)
- **Atributos:**
 - ▶ PortoOrigem: VARCHAR(50)
 - ▶ PortoDestino: VARCHAR(50)
 - ▶ Descricao: TEXT(300)
 - ▶ DataPartida: DATETIME
 - ▶ DataChegada: DATETIME
 - ▶ Estado: ENUM('Em Curso', 'Finalizada', 'Cancelada')
 - ▶ Saldo: DECIMAL(10,2) (*derivado*)

Tripulante

- **Chave primária:**
 - ▶ ID: INT
- **Chave estrangeira:**
 - ▶ Nenhuma
- **Atributos:**
 - ▶ Estado: ENUM('Ativo', 'Inativo')
 - ▶ CédulaMarinha: VARCHAR(30)
 - ▶ Nome: VARCHAR(150)
 - ▶ DataNascimento: DATE
 - ▶ Salário: DECIMAL(10,2)
 - ▶ Rua: VARCHAR(150)
 - ▶ Número: INT
 - ▶ CodPostal: VARCHAR(10)
 - ▶ Localidade: VARCHAR(150)
 - ▶ ContactoEmergência: VARCHAR(20) (*Nulo*)
 - ▶ Email: VARCHAR(150) (*Nulo*)

Embarcacao

- **Chave primária:**
 - ID: VARCHAR(8)
- **Chave estrangeira:**
 - Nenhuma
- **Atributos:**
 - Tipo: VARCHAR(50)
 - CapCarga: DECIMAL(10,2)
 - Comprimento: DECIMAL(10,2)
 - Altura: DECIMAL(10,2)
 - Largura: DECIMAL(10,2)
 - Nome: VARCHAR(150) (*Nulo*)

Financeiro

- **Chave primária:**
 - ID: INT
- **Chave estrangeira:**
 - Viagem: INT
- **Atributos:**
 - Tipo: ENUM('Receita', 'Despesa')
 - Valor: DECIMAL(10,2)
 - Descricao: TEXT(300) (*Nulo*)
 - Categoria: VARCHAR(50)

ZonaPesca

- **Chave primária:**
 - ID: INT
- **Chave estrangeira:**
 - Nenhuma
- **Atributos:**
 - Nome: VARCHAR(50)
 - Descricao: TEXT(300) (*Nulo*)
 - Latitude: DECIMAL(10,8)
 - Longitude: DECIMAL(10,8)

Captura

- **Chave primária:**
 - ID: INT
- **Chaves estrangeiras:**
 - Viagem: INT
 - ZonaPesca: INT
- **Atributos:**
 - NomeComum: VARCHAR(100)
 - NomeCientifico: VARCHAR(150) (*Nulo*)
 - Quantidade: DECIMAL(10,2)

A aplicação da regra de conversão de uma entidade conceptual para uma tabela resultou no seguinte modelo lógico:

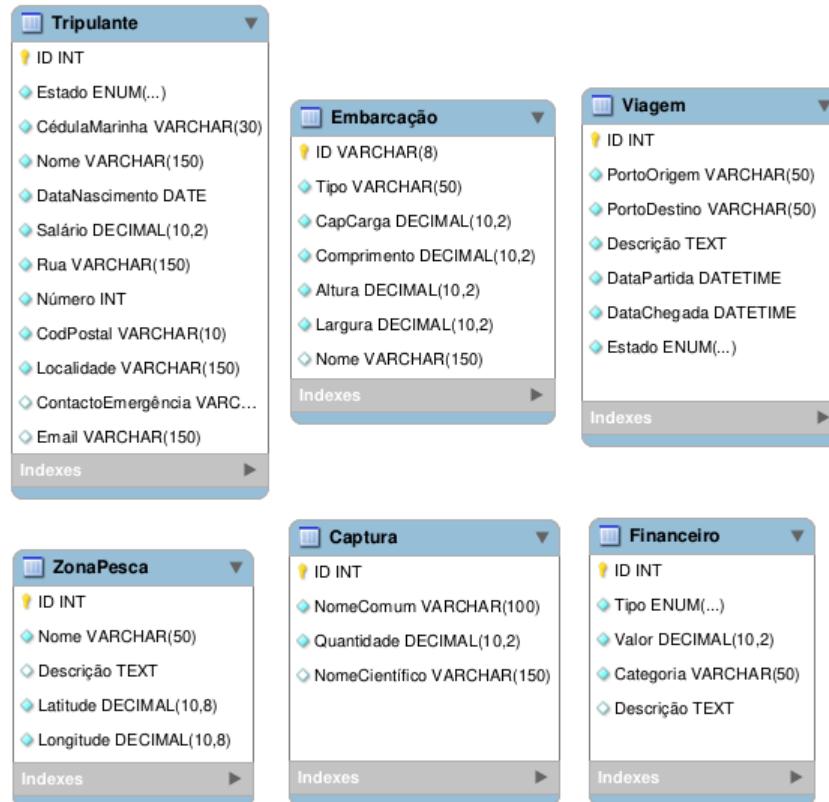


Figura 4.1: Diagrama ER Lógico após a aplicação da primeira regra de derivação

Para a conversão de um Relacionamento Binário de grau N:M, a chave primária de cada entidade é utilizada para a composição da chave primária composta da entidade originada do relacionamento, e esta deve incluir os seus atributos, se existentes.

Para os restantes relacionamentos é necessária a atribuição das chaves estrangeiras respetivas. Para tal, a chave primária da entidade do lado N é usada como chave estrangeira na entidade correspondente do lado 1.

A regra de conversão do relacionamento de grau N:M originou as seguintes entidades:

TripulanteViagem

- Chave primária composta:**
 - ▶ Tripulante: INT
 - ▶ Viagem: INT
 - ▶ Cargo: ENUM('Capitão', 'Imediato', 'Oficial de Navegação', 'Engenheiro', 'Mestre de Redes', 'Maquinista', 'Cozinheiro', 'Marinheiro')
- Chaves estrangeiras:**
 - ▶ Tripulante: INT
 - ▶ Viagem: INT
- Atributos:**
 - ▶ DataEntrada: DATE
 - ▶ DataSaida: DATE (*Nulo*)

ViagemZonaPesca

- **Chave primária composta:**

- Viagem: INT
- ZonaPesca: INT

- **Chaves estrangeiras:**

- Viagem: INT
- ZonaPesca: INT

- **Atributos:**

- Nenhum

Ao concluir a conversão dos relacionamentos binários de grau N:M e a atribuição de chaves estrangeiras aos demais relacionamentos, foi necessário adicionar duas tabelas, obtendo o modelo lógico mostrado na figura seguinte:

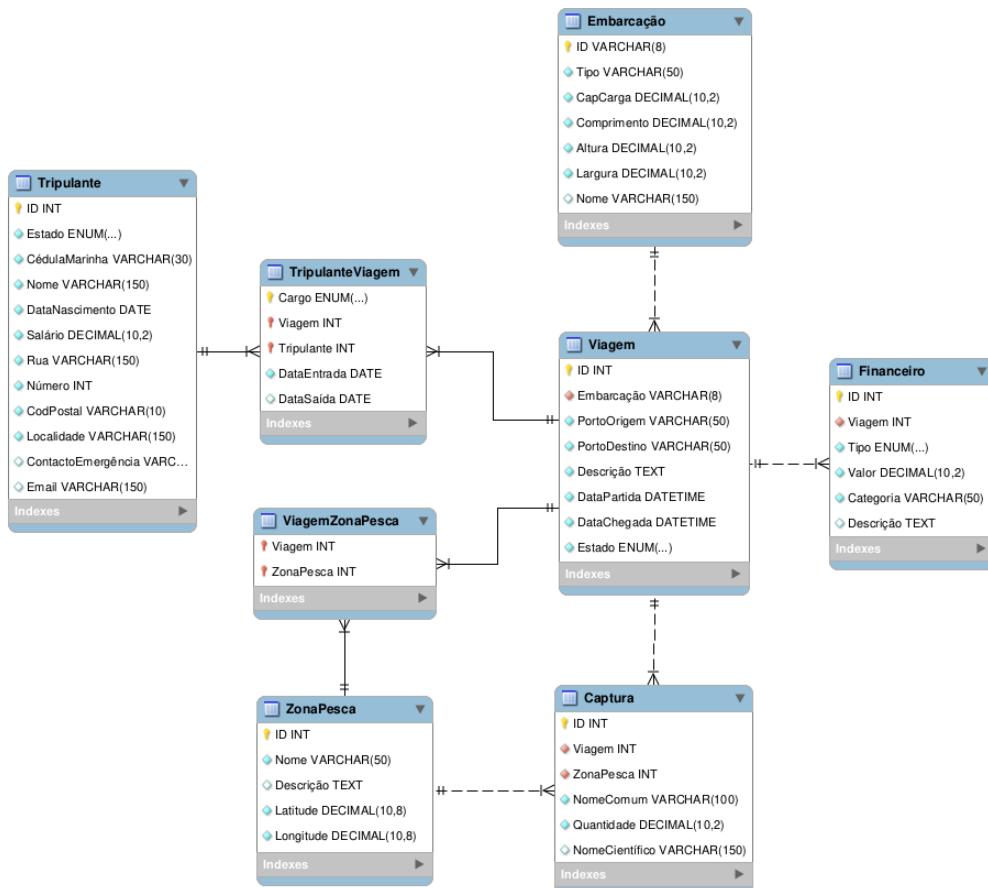


Figura 4.2: Diagrama ER Lógico apóia aplicação da segunda regra de derivação

No modelo conceptual, foram utilizados vários atributos multivalorados nas entidades “Tripulante” e “Embarcação” de forma a registar informações detalhadas relativas às mesmas. Estes atributos especiais resultam na formação de novas entidades lógicas, cujas características são derivadas diretamente dos valores múltiplos que cada entidade pode assumir. Cada uma destas incorpora o atributo como sua chave primária, garantindo a correta associação entre os dados e as entidades principais. As quais:

Nacionalidade

- **Chave primária:**
 - Nacionalidade: VARCHAR(50)
- **Chave estrangeira:**
 - Tripulante: INT
- **Atributos:**
 - Nenhum

PortosAutorizados

- **Chave primária:**
 - PortosAutorizados: VARCHAR(50)
- **Chave estrangeira:**
 - Embarcação: VARCHAR(8)
- **Atributos:**
 - Nenhum

A partir da conversão dos atributos multivalorados presentes no modelo conceptual, foi gerado o seguinte modelo lógico, incluindo a adição de duas novas tabelas. Deste modo fica finalizado o modelo lógico:

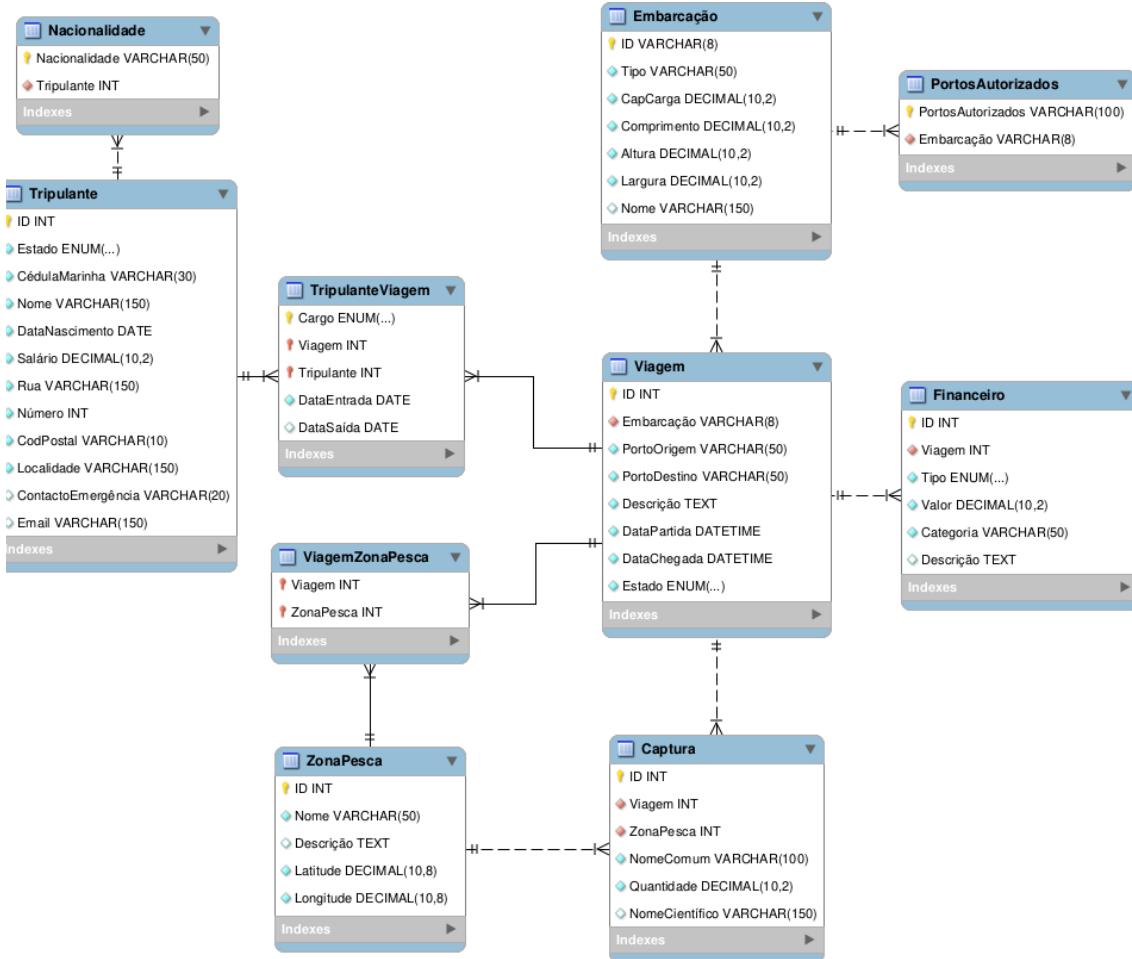


Figura 4.3: Diagrama ER Lógico apóia a aplicação da terceira regra de derivação

Nota: Consultar anexo [IV] Modelo Lógico.

4.3. Normalização de Dados

A normalização de dados constitui um processo essencial para garantir a integridade, consistência e eficiência de qualquer sistema de base de dados relacional. No caso da PortusPesca, este processo foi aplicado de forma sistemática ao modelo lógico, assegurando que cada entidade e relacionamento se encontra devidamente estruturado e livre de redundâncias. Através da decomposição das tabelas e da definição clara das dependências funcionais, tornou-se possível eliminar grupos repetitivos e valores multivvalorados, garantindo que cada atributo é atómico e depende diretamente da chave primária da sua tabela.

Um exemplo prático desta aplicação pode ser observado no relacionamento entre tripulantes e viagens. Numa fase inicial, os dados relativos ao cargo e às datas de participação encontravam-se dispersos, o que poderia conduzir a duplicações e inconsistências. Com a criação da entidade associativa Participa, passou a ser possível registar de forma única e controlada o papel de cada tripulante em cada viagem, evitando redundância e assegurando a rastreabilidade. O mesmo princípio foi aplicado às embarcações, cujos portos autorizados foram separados numa tabela própria, garantindo que a informação se mantém organizada e facilmente atualizável.

A aplicação das formas normais permitiu ainda resolver potenciais anomalias de inserção, atualização e remoção. Por exemplo, o saldo das viagens não é armazenado diretamente, mas calculado a partir dos movimentos financeiros associados a cada expedição, prevenindo dependências transitivas e assegurando que qualquer alteração nos registo económicos se reflete automaticamente nos resultados. Esta abordagem promove não só a consistência dos dados, como também a facilidade de manutenção e escalabilidade do sistema, uma vez que novas categorias de movimentos financeiros podem ser introduzidos sem comprometer a estrutura existente.

Em suma, o processo de normalização aplicado ao sistema da PortusPesca garante que os dados se mantêm fiáveis e coerentes, que as operações de consulta e manutenção são realizadas de forma eficiente e que o modelo está preparado para acompanhar o crescimento da organização. Ao evitar redundâncias, estabelecer relações claras entre entidades e assegurar a preservação das dependências funcionais, o sistema demonstra um compromisso firme com a integridade e a sustentabilidade, constituindo uma base sólida para a implementação física da base de dados e para o futuro desenvolvimento da plataforma.

4.4. Validação do Modelo com Interrogações do Utilizador

Para garantir a consistência global do modelo lógico, foram escolhidas expressões em álgebra relacional correspondentes aos requisitos de manipulação de dados, de modo a comprovar o correto funcionamento da maioria das entidades e relacionamentos definidos. Estas serão detalhadas nos parágrafos seguintes.

4.4.1. Listagem geral de viagens

Requisito relacionado:

"RE2 - "Deve ser possível obter uma listagem geral das Viagens com os seus atributos operacionais essenciais: identificador, portos de origem e destino, e estado atual.”"

Expressão em álgebra relacional:

```
1 π ID, PortoOrigem, PortoDestino, Estado (Viagem)
```

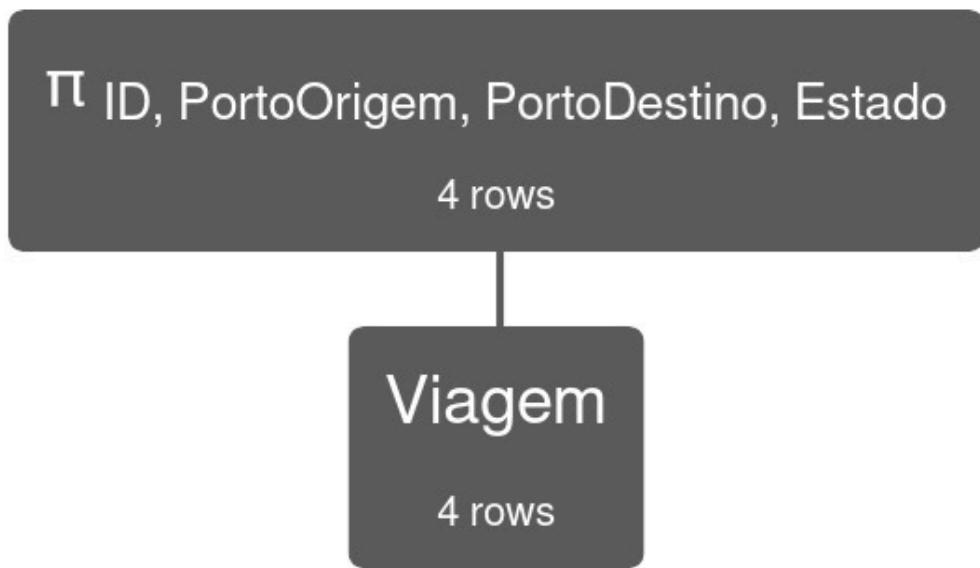


Figura 4.4: Representação gráfica da expressão em Álgebra Relacional 1

Esta interrogação visa fornecer uma visão macroscópica das operações da frota, abstraindo detalhes técnicos excessivos. A expressão aplica um operador de projeção (π) sobre a relação Viagem, selecionando exclusivamente os atributos estruturais essenciais: o identificador único (ID), os portos logísticos (PortoOrigem e PortoDestino) e a situação operacional (Estado). Esta abordagem permite uma consulta rápida e leve, fundamental para painéis de controlo gerais, omitindo dados volumosos como descrições ou datas precisas que não são necessários numa primeira análise.

4.4.2. Identificação de viagens ativas para atualização de estado

Requisito relacionado:

"RE1 - "Quando uma viagem é finalizada ou cancelada o seu estado é atualizado respetivamente, assim como a data de chegada e é feita a desassociação da tripulação da viagem."

Expressão em álgebra relacional:

```
1 π ID, PortoOrigem, PortoDestino, Estado (
2   σ Estado = 'Em Curso' (Viagem)
3 )
```

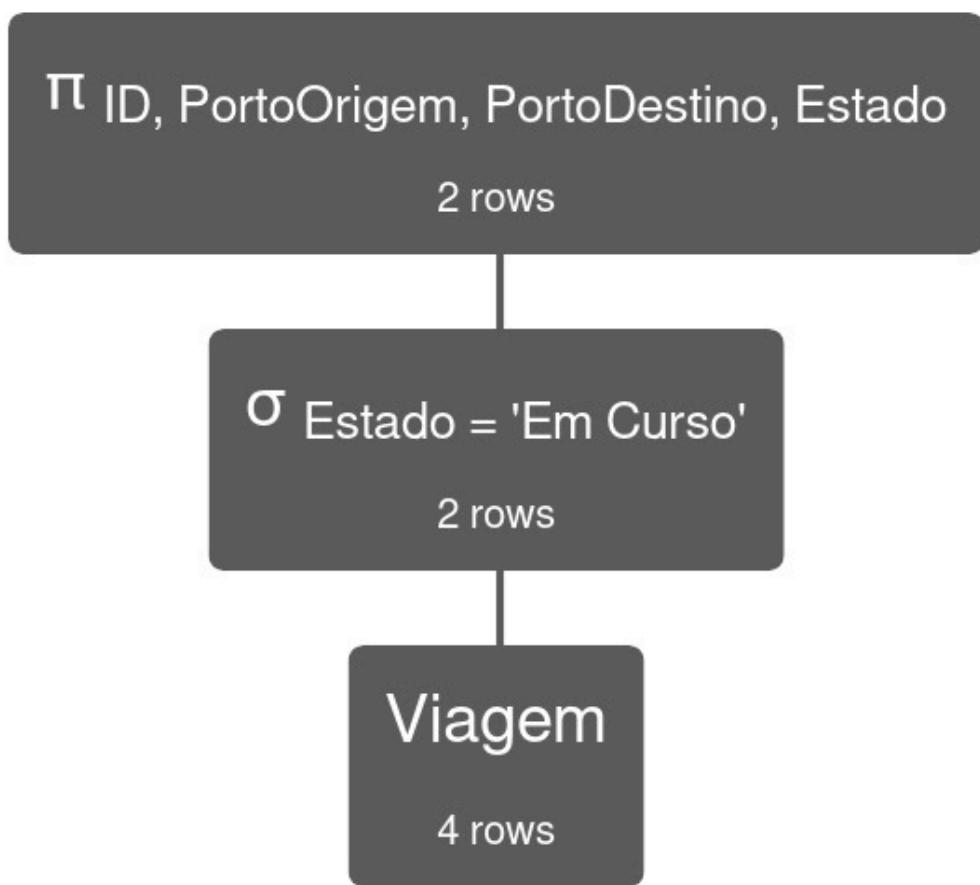


Figura 4.5: Representação gráfica da expressão em Álgebra Relacional 1

Para dar resposta à necessidade de manutenção dos estados das expedições, esta operação isola o subconjunto de viagens que carecem de monitorização ativa. O processo inicia-se com uma seleção (σ) rigorosa na entidade Viagem, filtrando apenas os registo onde o atributo Estado corresponde a "Em Curso". Sobre este resultado, aplica-se uma projeção (π) dos identificadores e rotas. O resultado permite à equipa de gestão focar-se apenas nas expedições abertas, facilitando o processo de atualização para "Finalizada" ou "Cancelada" e prevenindo a alteração acidental de registo históricos já encerrados.

4.4.3. Validação de estado contratual de tripulante, cargo e estado

Requisito relacionado:

"RE5 - "Quando um tripulante deixa de exercer a sua função (é promovido ou demitido) o seu atributo 'estado' e o 'cargo' devem ser atualizados respetivamente."

Expressão em álgebra relacional:

```

1 π Nome, Cargo, Estado (
2   σ (Tripulante.ID = 1 AND DataSaida = 'NULL') (
3     Tripulante ⋈ (Tripulante.ID = TripulanteViagem.Tripulante) TripulanteViagem
4   )
5 )

```

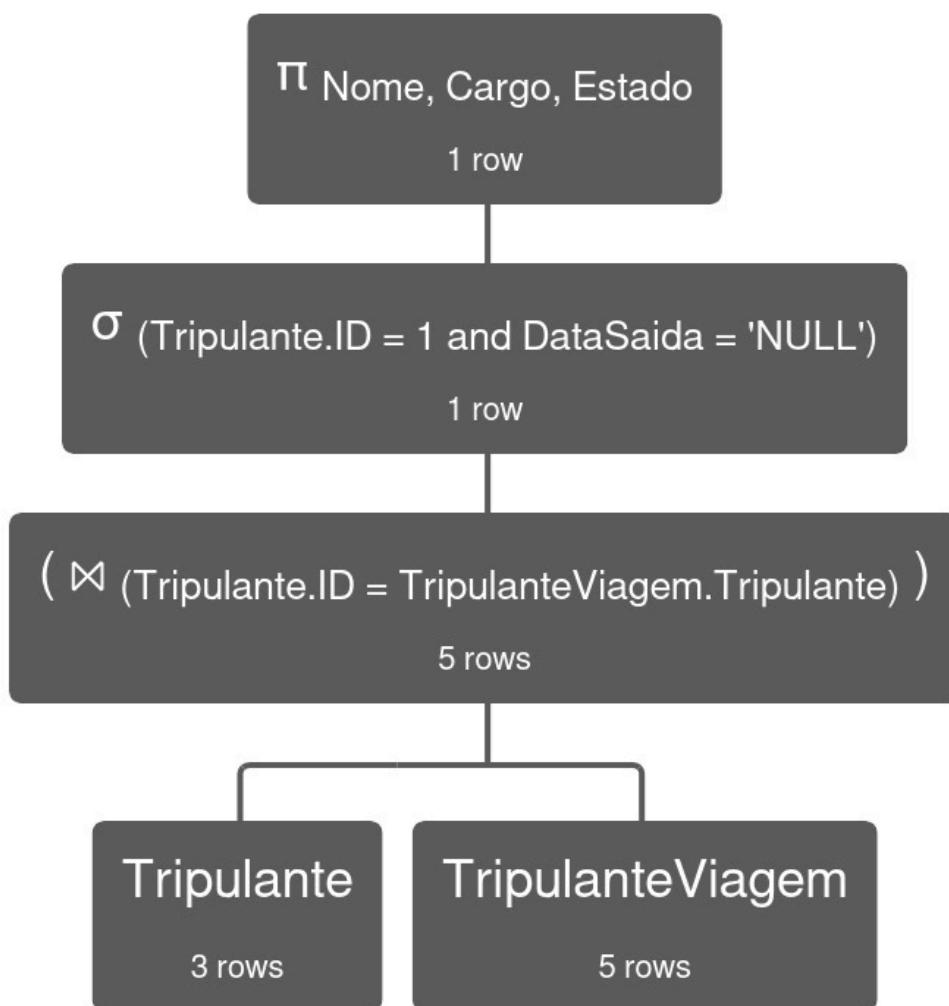


Figura 4.6: Representação gráfica da expressão em Álgebra Relacional 3

Nota: Ver anexo [V] Expressões de AR.

Esta consulta é crítica para garantir a integridade referencial dos Recursos Humanos antes de qualquer alteração contratual (promoção ou demissão). A árvore de execução ilustra uma junção (\bowtie) entre a tabela Tripulante e a entidade associativa TripulanteViagem. Simultaneamente, aplica-se uma seleção composta

(σ) que isola o tripulante específico (pelo ID) e verifica a existência de um vínculo ativo (onde DataSaida é nulo). A projeção final (π) devolve o Cargo e o Estado atuais, assegurando que o gestor possui a informação validada sobre a função corrente do tripulante antes de proceder a modificações administrativas.

4.4.4. Histórico operacional de tripulante (exemplo ID = 1)

Requisito relacionado:

"RE4 - "Dado o identificador de um tripulante, deve ser possível aceder a todas as viagens em que este esteve envolvido, bem como o cargo desempenhado em cada uma.""

Expressão em álgebra relacional:

```
1 π Nome, Cargo, ViagemID, PortoOrigem, PortoDestino, DataPartida (
2     -- Passo 1: Selecionar Tripulante com ID = 1
3     (σ ID = ? (Tripulante))
4
5     -- Passo 2: Juntar com TripulanteViagem
6     ⋈ (ID = TripulanteViagem.Tripulante) TripulanteViagem
7
8     -- Passo 3: Juntar com Viagem
9     ⋈ (TripulanteViagem.Viagem = ViagemID) (ρ ViagemID←ID Viagem)
10 )
```

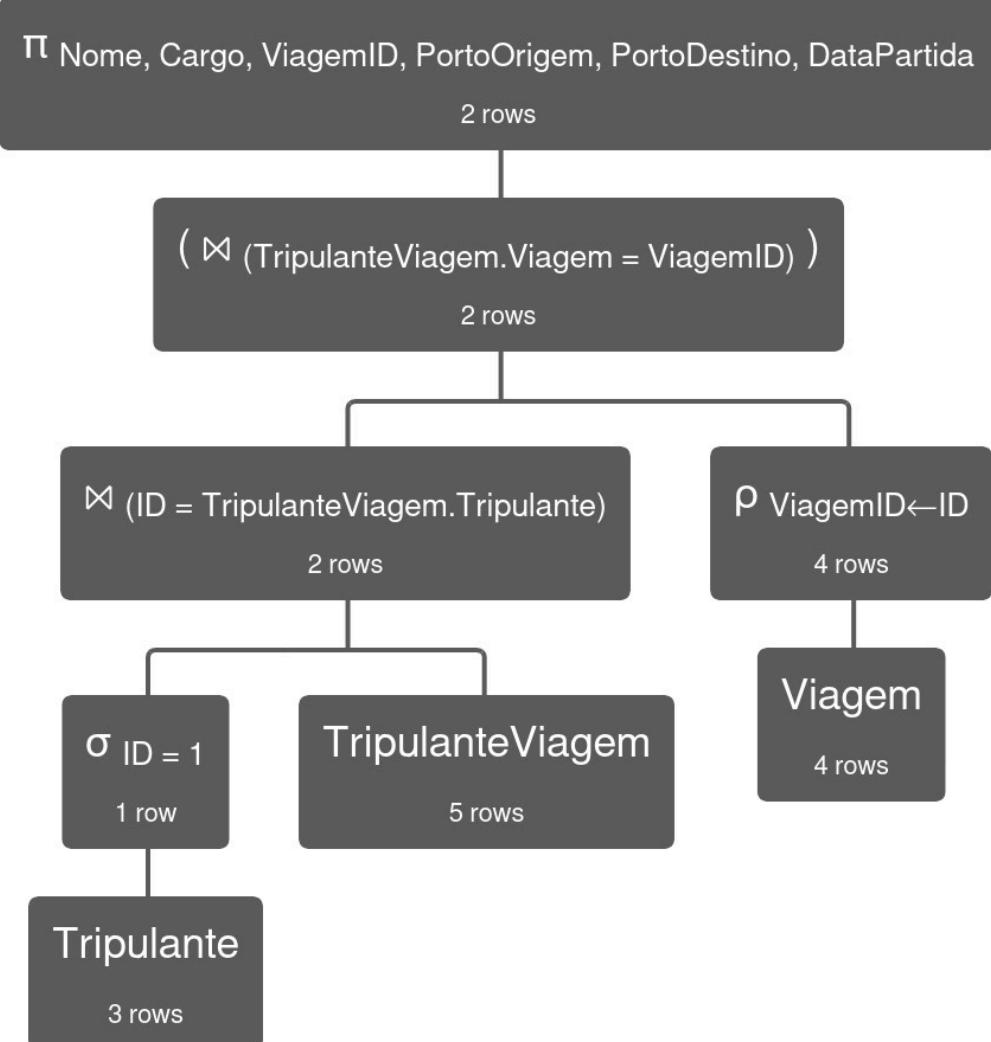


Figura 4.7: Representação gráfica da expressão em Álgebra Relacional 4

A fim de reconstruir a carreira de um tripulante, o sistema executa uma operação ascendente na árvore de execução. Inicialmente, otimiza-se a pesquisa isolando o registo do tripulante alvo através de uma seleção (σ). Este resultado é sequencialmente cruzado (\bowtie) com a tabela associativa TripulanteViagem e, posteriormente, com a tabela Viagem. É de notar a utilização do operador de renomeação (ρ) na tabela de viagens, uma medida técnica necessária para evitar ambiguidades entre os identificadores das diferentes entidades. O resultado final projeta (π) a correlação entre o profissional, o cargo desempenhado e os detalhes logísticos de cada expedição em que participou.

4.4.5. Relatório diário de lucro da viagem com base nas receitas e despesas

Requisito relacionado:

"RE11 - "No encerramento de cada dia, o sistema deve gerar automaticamente um relatório contendo todos os movimentos financeiros registados neste dia, agrupados por viagem, incluindo somatórios de receitas, despesas e lucro/prejuízo diário acumulado.”"

Expressão em álgebra relacional:

```

1 -- Passo 1: Calcular apenas as Receitas por Viagem
2 TotalReceitas =  $\gamma$  Viagem; sum(Valor)→SomaReceitas (
3      $\sigma$  Tipo = 'Receita' (Financeiro)
4 )
5
6 -- Passo 2: Calcular apenas as Despesas por Viagem
7 TotalDespesas =  $\gamma$  Viagem; sum(Valor)→SomaDespesas (
8      $\sigma$  Tipo = 'Despesa' (Financeiro)
9 )
10
11 -- Passo 3: Juntar as tabelas e calcular o Lucro (Receita - Despesa)
12 RelatorioFinal =  $\pi$  Viagem, SomaReceitas, SomaDespesas, (SomaReceitas -
13     SomaDespesas)→Lucro (
14     TotalReceitas  $\bowtie$  TotalDespesas
15 )
16 -- Executar a query final
17 RelatorioFinal

```

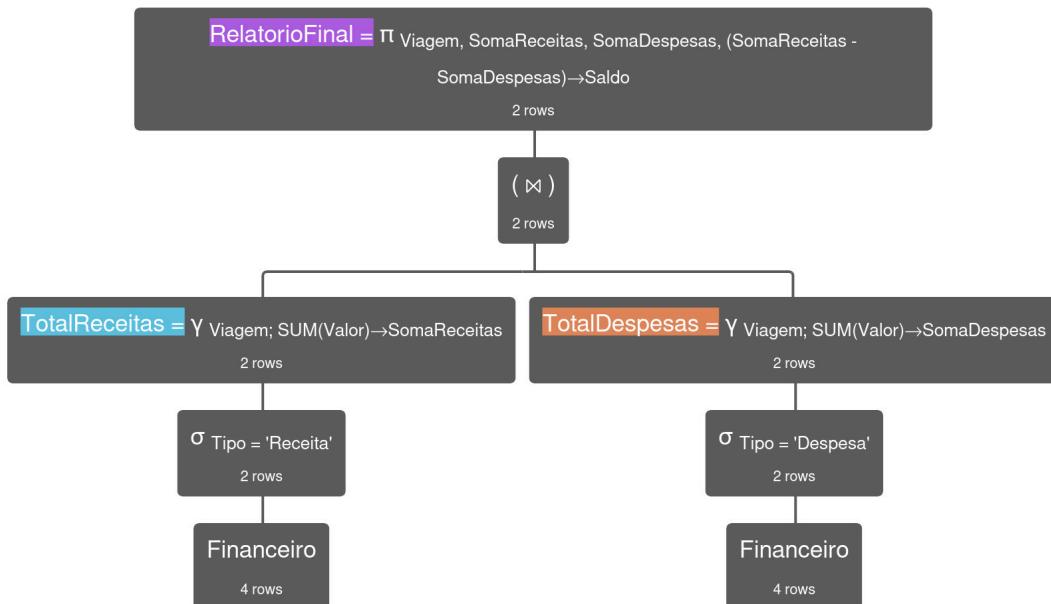


Figura 4.8: Representação gráfica da expressão em Álgebra Relacional 5

A estratégia de resolução para este requisito baseia-se na decomposição dos movimentos financeiros em dois fluxos distintos — Receitas e Despesas — que são agregados isoladamente por viagem e posteriormente unificados para o cálculo aritmético do saldo líquido.

Passo 1: O objetivo é segregar os fluxos financeiros positivos. Através de uma operação de seleção (σ), filtram-se apenas os registos da tabela Financeiro cujo tipo é 'Receita'. De seguida, aplica-se o operador de agrupamento (γ) por Viagem, calculando o somatório dos valores. O resultado é armazenado numa relação temporária (TotalReceitas) que contém o rendimento bruto acumulado para cada expedição.

Passo 2: De forma análoga ao passo anterior, este procedimento foca-se na quantificação dos débitos. O sistema isola os movimentos classificados como ‘Despesa’ e executa uma nova agregação por Viagem. Esta operação gera a relação temporária TotalDespesas, que consolida a soma de todos os encargos operacionais (como combustível, manutenção ou taxas portuárias) associados a cada viagem específica.

Passo 3: A etapa final realiza a integração dos dados através de uma junção natural (\bowtie) entre as relações de receitas e despesas, utilizando o identificador da viagem como chave de ligação. Na projeção final (π), é criado um atributo derivado (‘Lucro’) resultante da subtração aritmética entre a SomaReceitas e a SomaDespesas.

4.4.6. Estatística semanal de capturas por zona e espécie

Requisito relacionado:

"RE13 - "No encerramento de cada semana, o sistema deverá gerar um relatório com informações e estatísticas relativas a nomes comuns e quantidades de capturas por zona de pesca.""

Expressão em álgebra relacional:

```

1 -- Passo 1: Junção Condicional (Associar Capturas às Zonas pelo ID correto)
2 DadosUnificados = Captura ⚡ Captura.ZonaPesca = ZonaPesca.ID ZonaPesca
3
4 -- Passo 2: Estatísticas de Quantidade por Zona e Espécie
5 RelatorioSemanal = γ Nome, NomeComum; sum(Quantidade)→TotalKg,
       count(Captura.ID)→NumRegistros (
6     DadosUnificados
7   )
8
9 RelatorioSemanal

```

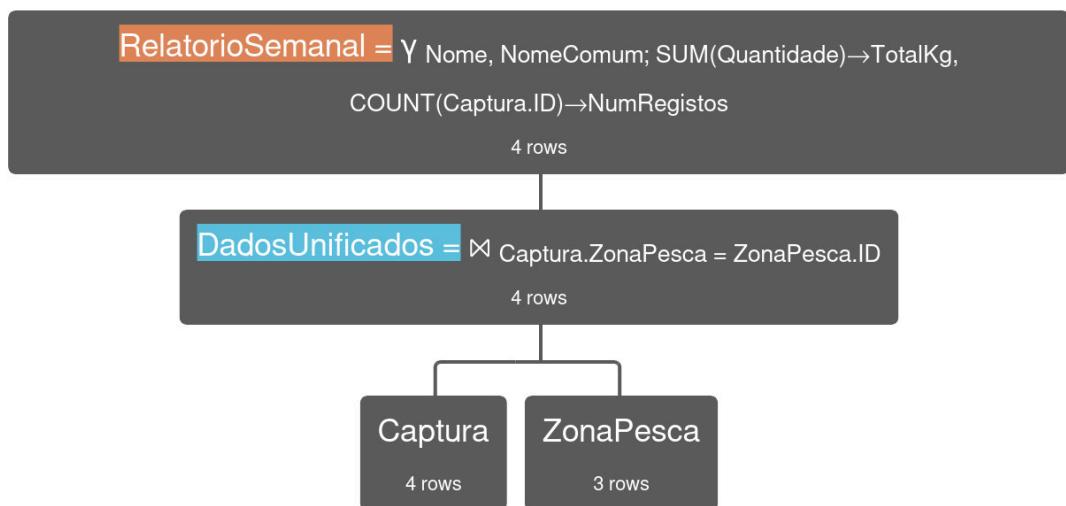


Figura 4.9: Representação gráfica da expressão em Álgebra Relacional 6

A estratégia implementada fundamenta-se no enriquecimento semântico dos registos de captura através da resolução das referências geográficas, seguido de uma agregação multidimensional que processa os dados estatísticos simultaneamente por zona de pesca e espécie.

Passo 1: A tabela Captura referencia a localização da atividade pesqueira apenas através de um código numérico (chave estrangeira). Para enriquecer os dados e tornar o relatório legível, executa-se uma junção condicional ($\bowtie_{Captura.ZonaPesca=ZonaPesca.ID}$) entre as tabelas Captura e ZonaPesca. Esta operação estabelece a ligação semântica entre o registo do pescado e a sua origem geográfica, permitindo o acesso ao atributo Nome da zona, essencial para o agrupamento subsequente.

Passo 2: Sobre a relação unificada, aplica-se o operador de agrupamento (γ) utilizando dois critérios de segmentação simultâneos: o Nome da zona de pesca e o NomeComum da espécie capturada. Para cada combinação única destes fatores, o sistema calcula duas métricas estatísticas fundamentais para a gestão da frota: o somatório da Quantidade, gerando o atributo TotalKg, e a contagem de registos (NumRegistros).

4.4.7. Auditoria Integral de Expedição

Requisito relacionado:

"RE14 - "O sistema deve permitir a extração de um dossier completo de uma viagem específica para efeitos de fiscalização e auditoria. Este relatório deve cruzar, numa única vista, os dados logísticos da embarcação, a lista detalhada dos tripulantes embarcados (incluindo as suas credenciais) e o registo discriminado de todas as capturas e respetivas zonas de pesca."

Expressão em álgebra relacional:

```
1 -- Ramo 1: Dados Logísticos
2 RamoLogistica = π Viagem.ID, DataPartida, PortoOrigem, PortoDestino,
  Nome→NomeNavio, Tipo→TipoNavio (
3   σ Viagem.ID = 1 (Viagem) ⋈ Viagem.Embarcacao = Embarcacao.ID Embarcacao
4 )
5
6 -- Ramo 2: Dados de Recursos Humanos
7 RamoRH = π Cargo, Nome→NomeTripulante, CedulaMarinha, Tripulante, Viagem (
8   σ Viagem = 1 (TripulanteViagem) ⋈ TripulanteViagem.Tripulante = Tripulante.ID
  Tripulante
9 )
10
11 -- Ramo 3: Dados de Produção
12 RamoProducao = π NomeComum→Especie, Quantidade, Nome→NomeZona, Viagem (
13   σ Viagem = 1 (Captura) ⋈ Captura.ZonaPesca = ZonaPesca.ID ZonaPesca
14 )
15
16 -- Junção Intermédia
17 UniaoLogisticaRH = RamoLogistica ⋈ RamoLogistica.ID = RamoRH.Viagem RamoRH
18
19 -- Junção Final
20 DossierFinal = π ID, NomeNavio, DataPartida, NomeTripulante, Cargo, Especie,
  Quantidade, NomeZona (
21   UniaoLogisticaRH ⋈ UniaoLogisticaRH.ID = RamoProducao.Viagem RamoProducao
22 )
23
24 DossierFinal
```

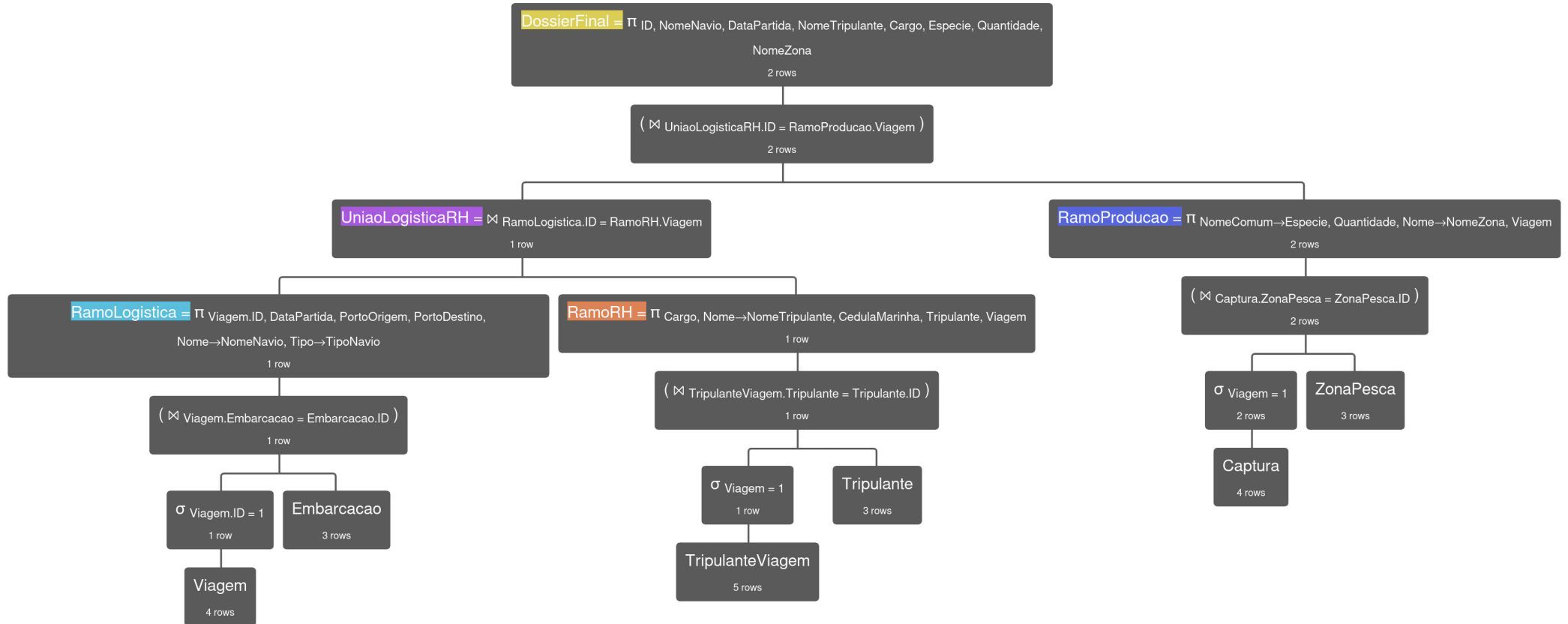


Figura 4.10: Representação gráfica da expressão em Álgebra Relacional 7

A estratégia de resolução para este requisito complexo baseia-se na decomposição do problema em três vetores de informação distintos — Logística, Recursos Humanos e Produção — que são processados isoladamente e posteriormente unificados.

O processamento inicia-se com a construção do **Ramo Logístico**, onde se isola a instância da viagem alvo (neste caso, ID=1) através de uma operação de seleção (σ). O resultado é imediatamente cruzado (\bowtie) com a entidade Embarcacao para obter os detalhes técnicos do navio. É de salientar a aplicação de renomeações (ρ) nos atributos Nome e Tipo (para NomeNavio e TipoNavio), uma medida técnica essencial para prevenir ambiguidades semânticas, visto que estes identificadores se repetem em várias entidades do sistema.

Paralelamente, desenvolve-se o **Ramo de Recursos Humanos**, focado na gestão de pessoal. O sistema filtra a tabela associativa TripulanteViagem para a viagem específica e realiza uma junção com a tabela mestre Tripulante. Aqui, aplica-se novamente a renomeação do atributo Nome para NomeTripulante, resultando numa lista depurada que identifica inequivocamente quem estava a bordo e qual o cargo desempenhado, separando estes dados da identificação do navio.

Num terceiro fluxo, processa-se o **Ramo de Produção**, referente ao resultado económico da expedição. As capturas associadas ao ID da viagem são cruzadas com a tabela ZonaPesca através de uma junção condicional. Esta operação permite substituir o código numérico da zona pela sua designação geográfica descritiva (aqui renomeada para NomeZona), tornando o relatório imediatamente legível para auditores não técnicos.

A fase final do procedimento consiste na **Consolidação Hierárquica** através de junções sequenciais. Primeiramente, unem-se os dados Logísticos aos de Recursos Humanos, criando uma relação intermédia onde cada tripulante fica associado às características do navio. De seguida, este conjunto é cruzado com o Ramo de Produção. O resultado, DossierFinal, é uma relação extensa e desnormalizada que apresenta todas as combinações possíveis entre tripulantes e capturas para aquela viagem.

5. Implementação Física

5.1. Apresentação e explicação da base de dados implementada

5.1.1. Apresentação e Explicação da Base de Dados Implementada

A implementação física do sistema PortusPesca foi materializada através da tradução do modelo lógico para a linguagem SQL (dialecto MySQL 8.0), focando-se na garantia da integridade dos dados, na otimização do armazenamento e na aplicação estrita das regras de negócio.

A estratégia de implementação, cujo script completo se encontra no **anexo [VI] DDL -Definição de Dados**, assentou em três pilares fundamentais:

1. Ordem de Criação e Integridade Referencial para assegurar o cumprimento das restrições de chave estrangeira, estabeleceu-se uma hierarquia de criação estrita, prevenindo erros de dependência:

Entidades Nucleares e Domínios: Criação inicial das tabelas independentes que sustentam o sistema (Embarcacao, ZonaPesca) e tabelas de tripulantes (Tripulante), garantindo que os dados mestres existem antes de qualquer operação.

Entidade Central: Implementação da tabela Viagem, que atua como o eixo central de ligação entre a frota, a tripulação e as operações de pesca.

Entidades Dependentes e Associativas: Por fim, criação das tabelas que dependem da existência de uma viagem (Captura, Financeiro) e das tabelas que materializam as relações N:M (TripulanteViagem, ViagemZonaPesca), fechando a malha de integridade.

2. Refinamento de Domínios e Regras de Negócio uma das principais vantagens da implementação física foi a transferência de validações de negócio para o próprio SGBD. Destaca-se o uso extensivo do tipo enumerado (ENUM) para atributos com domínios fechados, garantindo que o sistema rejeita, ao nível da base de dados, estados inválidos.

Por exemplo, na tabela Viagem, o estado é restrito e possui um valor predefinido para agilizar o registo de novas expedições:

```
1 CREATE TABLE Viagem (
2     (...),
3     Estado ENUM('Em Curso', 'Finalizada', 'Cancelada') NOT NULL DEFAULT 'Em Curso',
4     (...),
5 );
```

Similarmente, na tabela Tripulante, assegura-se que o estado contratual está sempre definido, assumindo “Ativo” por defeito na admissão:

```
1 CREATE TABLE Tripulante (
2     (...),
3     Estado ENUM('Ativo', 'Inativo') NOT NULL DEFAULT 'Ativo',
4     (...),
5 );
```

3. Automatização e Tipos de Dados Para otimizar a inserção de dados e garantir a precisão numérica, foram tomadas decisões específicas de engenharia:

Identificadores Artificiais: Utilização da propriedade AUTO_INCREMENT em chaves primárias numéricas (como em Financeiro e Captura), delegando ao SGBD a gestão da unicidade e sequencialidade dos registos.

Precisão Financeira e Geográfica: Adoção do tipo DECIMAL em detrimento de FLOAT para atributos sensíveis como coordenadas geográficas (ZonaPesca) e valores monetários (Financeiro), eliminando erros de arredondamento em cálculos contabilísticos e de navegação.

```
1 CREATE TABLE Financeiro (
2     ID INT AUTO_INCREMENT PRIMARY KEY,
3     Viagem INT NOT NULL,
4     Valor DECIMAL(10,2) NOT NULL,
5     (...),
6 );
```

SQL

Esta abordagem estruturada garante uma base de dados robusta, onde a integridade é preservada estruturalmente e não apenas por via aplicacional.

5.2. Criação de Utilizadores da Base de Dados

A política de segurança do sistema PortusPesca foi implementada através de instruções DCL (Data Control Language), visando garantir a confidencialidade e integridade da informação. Em vez de atribuir permissões individualmente a cada utilizador, optou-se por uma estratégia baseada em Funções (Roles), facilitando a gestão futura e a escalabilidade da equipa.

O script de definição de acessos, disponível integralmente no anexo [VII] DCL - Controlo de Dados, estabelece três perfis de acesso distintos, alinhados com a hierarquia da empresa:

1. Perfil de Administração (g_administrador) Destinado à equipa de gestão e TI, este perfil possui controlo total sobre a base de dados. Esta permissão responde diretamente ao requisito [RA5], assegurando que os administradores podem efetuar qualquer operação de manutenção, leitura ou escrita.

```
1 CREATE ROLE 'g_administrador';
2 GRANT ALL PRIVILEGES ON PortusPesca.* TO 'g_administrador' WITH GRANT OPTION;
```

2. Perfil de Oficiais (g_oficial) Atribuído a Capitães e Imediatos, este perfil foca-se na gestão operacional. Para cumprir o requisito [RA2], que exige que apenas cargos de responsabilidade alterem dados de viagens, concedeu-se permissão de INSERT e UPDATE nas tabelas operacionais (Viagem, Captura, TripulanteViagem), mantendo o acesso de leitura (SELECT) às tabelas financeiras para controlo de custos.

3. Perfil de Tripulação (g_tripulacao) Destinado a marinheiros e restantes membros, este perfil é estritamente de consulta. A sua implementação destaca-se por duas medidas de segurança granulares:

Bloqueio Financeiro: Em conformidade com o [RA3], a tripulação não possui qualquer privilégio de acesso à tabela Financeiro, garantindo a proteção de dados sensíveis da empresa.

Segurança ao Nível da Coluna: Para proteger a privacidade dos colaboradores, o acesso à tabela Tripulante foi restringido ao nível dos atributos. Um marinheiro pode consultar os contactos dos colegas (para coordenação), mas o SGBD bloqueia o acesso ao atributo Salario.

```
1 -- Exemplo de Granularidade: Acesso a dados de contacto, ocultando o Salário
2 GRANT SELECT(ID, Nome, Email, ContactoEmergencia, Estado)
3 ON PortusPesca.Tripulante TO 'g_tripulacao';
```

Materialização dos Utilizadores Por fim, foram criados utilizadores concretos para testar a eficácia destas políticas (ex: capitao_ahab, marinheiro_ismael), aos quais foram atribuídas as respetivas Roles. Esta abordagem garante que, ao rodar a equipa ou contratar novos elementos, basta associá-los a uma Role existente, sem necessidade de redefinir permissões complexas.

5.3. Povoamento da Base de Dados

O povoamento da base de dados representa a materialização do sistema com dados operacionais, realizada através de scripts DML (Data Manipulation Language). Este processo não consistiu apenas na inserção de registos, mas sim na validação prática de todas as restrições de integridade e domínios definidos anteriormente.

A estratégia de povoamento, cujo script integral se encontra no **anexo [VIII] DML - Povoamento**, obedeceu a uma ordem de precedência estrita para garantir a consistência referencial:

1. Carregamento de Entidades Fortes e Dados Mestres Inicialmente, foram povoadas as tabelas que não dependem de chaves estrangeiras, estabelecendo os dados fundamentais da empresa.

Frota e Logística: Inserção das embarcações (Embarcacao) e respetivas zonas de operação (ZonaPesca).

Recursos Humanos: Registo dos colaboradores na tabela Tripulante. Aqui, destaca-se a normalização dos atributos multivalorados, onde nacionalidades e portos autorizados foram inseridos em tabelas próprias (Nacionalidade, PortosAutorizados) logo após a criação das entidades pai.

2. Registo da Atividade Operacional (Entidade Central) Com os dados mestres assegurados, procedeu-se à inserção das viagens na tabela Viagem. Este passo é crítico pois atua como eixo central para as restantes dependências. Foi contemplada a inserção de viagens com estados distintos para testar a robustez do modelo:

Viagens Finalizadas: Com datas de partida e chegada preenchidas.

Viagens 'Em Curso': Onde a DataChegada é inserida como NULL, validando a capacidade do sistema em lidar com operações a decorrer.

3. Entidades Dependentes e Histórico. Por fim, foram carregados os dados que dependem da existência de uma viagem:

Tripulação em Viagem: A tabela TripulanteViagem associa tripulantes a viagens específicas, atribuindo cargos ('Capitão', 'Marinheiro') e validando as datas de entrada.

Gestão Financeira: A tabela Financeiro foi povoada com uma lógica de dupla validação: utiliza-se o atributo Tipo ('Receita'/'Despesa') e, simultaneamente, o sinal numérico do atributo Valor (negativo para despesas, positivo para receitas) para facilitar cálculos de agregados.

```
1 -- Exemplo de registo financeiro (Despesa vs Receita)
2 INSERT INTO Financeiro (Viagem, Tipo, Valor, Descricao, ...) VALUES
3     (1, 'Despesa', -850.00, 'Combustível...', 'Combustível'),
4     (1, 'Receita', 2700.00, 'Venda de carapau...', 'Venda');
```

SQL

Esta abordagem estruturada permitiu detetar e corrigir incongruências de dependência antes da entrada em produção, garantindo que a base de dados reflete fielmente a realidade operacional da PortusPesca.

5.4. Cálculo do Espaço da Base de Dados

O cálculo volumétrico da base de dados *PortusPesca* baseia-se na análise da estrutura física implementada (DDL) e no volume de dados inicial carregado (DML). Para a determinação do espaço de armazenamento, seguiram-se as especificações do manual oficial do MySQL 8.0, assumindo que os campos de texto utilizam o conjunto de caracteres latin1 (1 byte por caráter) para efeitos de estimativa base.

Abaixo apresenta-se o detalhamento da estrutura de cada tabela, seguido pelo cálculo consolidado do espaço total ocupado.

5.4.1. Detalhamento por Entidade

Tabela 5.1: Estimativa de volume: Viagem

| Viagem | Tipo | Bytes |
|-----------------------|-------------|------------|
| ID | INT | 4 |
| Embarcacao | VARCHAR(8) | 9 |
| PortoOrigem | VARCHAR(50) | 51 |
| PortoDestino | VARCHAR(50) | 51 |
| Descricao | TEXT | 302 |
| DataPartida | DATETIME | 5 |
| DataChegada | DATETIME | 5 |
| Estado | ENUM | 1 |
| Total/Registro | | 428 |

Tabela 5.2: Estimativa de volume: Tripulante

| Tripulante | Tipo | Bytes |
|-----------------------|---------------|------------|
| ID | INT | 4 |
| Cedula | VARCHAR(30) | 31 |
| Nome | VARCHAR(150) | 151 |
| DataNasc | DATE | 3 |
| Salario | DECIMAL(10,2) | 5 |
| Cont.Emerg | VARCHAR(20) | 21 |
| Email | VARCHAR(150) | 151 |
| Estado | ENUM | 1 |
| Morada | Composto | 317 |
| Total/Registro | | 684 |

Nota: O campo **Morada** inclui Rua (151), Número (4), CodPostal (11) e Localidade (151).

Tabela 5.3: Estimativa de volume: Embarcação

| Embarcacao | Tipo | Bytes |
|-----------------------|---------------|------------|
| ID | VARCHAR(8) | 9 |
| Nome | VARCHAR(150) | 151 |
| Tipo | VARCHAR(50) | 51 |
| Dimensoes | 3xDECIMAL | 15 |
| CapCarga | DECIMAL(10,2) | 5 |
| Total/Registro | | 231 |

Tabela 5.4: Estimativa de volume: Financeiro

| Financeiro | Tipo | Bytes |
|-----------------------|---------------|------------|
| ID | INT | 4 |
| Viagem | INT | 4 |
| Tipo | ENUM | 1 |
| Valor | DECIMAL(10,2) | 5 |
| Descricao | TEXT | 302 |
| Categoria | VARCHAR(50) | 51 |
| Total/Registro | | 367 |

Tabela 5.5: Estimativa de volume: Captura

| Captura | Tipo | Bytes |
|-----------------------|---------------|------------|
| ID | INT | 4 |
| Viagem | INT | 4 |
| Zona | INT | 4 |
| NomeComum | VARCHAR(100) | 101 |
| NomeCient | VARCHAR(150) | 151 |
| Qtd | DECIMAL(10,2) | 5 |
| Total/Registro | | 269 |

Tabela 5.6: Estimativa de volume: ZonaPesca

| ZonaPesca | Tipo | Bytes |
|-----------------------|---------------|------------|
| ID | INT | 4 |
| Nome | VARCHAR(50) | 51 |
| Descricao | TEXT | 302 |
| Latitude | DECIMAL(10,8) | 5 |
| Longitude | DECIMAL(10,8) | 5 |
| Total/Registro | | 367 |

Tabela 5.7: Estimativa de volume: TripulanteViagem

| TripulanteViagem | Tipo | Bytes |
|----------------------|------|-----------|
| Tripulante | INT | 4 |
| Viagem | INT | 4 |
| Cargo | ENUM | 1 |
| DataEntrada | DATE | 3 |
| DataSaida | DATE | 3 |
| Total/Registo | | 15 |

Tabela 5.8: Estimativa de volume: Tabelas Auxiliares

| Tabelas Auxiliares | Tipo | Bytes |
|--------------------------|----------|-------|
| PortosAutorizados | | |
| Emb + Porto | VARCHARs | 110 |
| Nacionalidade | | |
| Trip + Nac | INT+VAR | 55 |
| ViagemZonaPesca | | |
| Viagem + Zona | 2xINT | 8 |

5.4.2. Resumo e Projeção de Crescimento

Com base no detalhamento acima e nos registos inseridos durante o povoamento (descrito em 5.3), obtém-se a seguinte dimensão inicial:

| Tabela | Cálculo dos Atributos (Bytes) | B/Reg | Nº | Total |
|---------------------------------------|---|-------|----|---------------|
| Embarcacao | ID(9) + Nome(151) + Tipo(51) + Dims(15) + Carga(5) | 231 | 6 | 1 386 |
| Tripulante | ID(4) + Cedula(31) + Dados(159) + Sal(5) + Morada(317) + Est(1) + Cont(172) | 684 | 12 | 8 208 |
| ZonaPesca | ID(4) + Nome(51) + Desc(302) + Coords(10) | 367 | 5 | 1 835 |
| Viagem | ID(4) + Emb(9) + Portos(102) + Desc(302) + Datas(10) + Est(1) | 428 | 7 | 2 996 |
| Financeiro | ID(4) + Viagem(4) + Tipo(1) + Valor(5) + Desc(302) + Cat(51) | 367 | 26 | 9 542 |
| Captura | ID(4) + Viagem(4) + Zona(4) + Nomes(252) + Qtd(5) | 269 | 11 | 2 959 |
| TripulanteVia- gem | Trip(4) + Viagem(4) + Cargo(1) + Datas(6) | 15 | 25 | 375 |
| ViagemZona- Pesca | Viagem(4) + Zona(4) | 8 | 9 | 72 |
| PortosAutoriza- dos | Emb(9) + Porto(101) | 110 | 14 | 1 540 |
| Nacionalidade | Trip(4) + Nac(51) | 55 | 15 | 825 |
| Dimensão Inicial Total (Bytes) | | | | 29 738 |

Tabela 5.9: Cálculo do Espaço de Armazenamento da BD

Análise de Crescimento: A base de dados inicia a sua operação com aproximadamente **29.74 KB**. Considerando um cenário de expansão operacional com 50 novas viagens anuais, onde cada viagem gera em média 10 capturas e 15 movimentos financeiros, projeta-se um crescimento anual de aproximadamente **350 KB**. Esta arquitetura garante a sustentabilidade do sistema *PortusPesca* a longo prazo.

5.5. Definição e Caracterização de Vistas de Utilização em SQL

Com o objetivo de abstrair a complexidade do modelo físico e fornecer aos utilizadores finais (gestores e oficiais) um acesso direto a informação consolidada, foram implementadas vistas (Views) estratégicas. Estas estruturas virtuais simplificam consultas recorrentes que, de outra forma, exigiriam a escrita repetida de junções complexas.

As principais vistas desenvolvidas, cujo código integral consta no anexo [VI] DDL - Definição de Dados, são:

1. Monitorização da Frota: ViagensEmCurso Esta vista foi desenhada para dar resposta à necessidade de controlo operacional em tempo real. Em vez de consultar a tabela Viagem e obter apenas IDs numéricos, esta vista efetua a junção (INNER JOIN) com a tabela Embarcacao, permitindo visualizar imediatamente o Nome e Tipo do navio associado. A aplicação do filtro WHERE Estado = 'Em Curso' garante que a vista retorna dinamicamente apenas as expedições ativas, servindo como dashboard para a equipa de terra.

```
1 CREATE VIEW ViagensEmCurso AS
2     SELECT V.ID, E.Nome, V.PortoDestino, V.Descricao
3     FROM Viagem AS V
4     INNER JOIN Embarcacao AS E ON V.Embarcacao = E.ID
5     WHERE V.Estado = 'Em Curso';
```

SQL

2. Análise de Produtividade: DetalhesCapturas Para facilitar a análise de rendimento pesqueiro, criou-se uma vista que desnormaliza a informação das capturas. Através da junção de três tabelas (Captura, ZonaPesca e Viagem), esta vista enriquece cada registo de captura com o nome legível da zona geográfica e a data da expedição. Isto permite, por exemplo, que um analista consulte rapidamente "quanto se pescou na Zona Norte" sem precisar de conhecer a estrutura relacional subjacente ou as chaves estrangeiras.

```
1 CREATE VIEW DetalhesCapturas AS
2     SELECT C.NomeComum, C.Quantidade, Z.Nome AS ZonaPesca, V.DataPartida
3     FROM Captura AS C
4     INNER JOIN ZonaPesca AS Z ON C.ZonaPesca = Z.ID
5     INNER JOIN Viagem AS V ON C.Viagem = V.ID;
```

SQL

3. Abstração de Inventário: TodasEmbarcacoes Foi ainda criada uma vista de acesso geral à frota, que serve como camada de abstração sobre a tabela física Embarcacao. Embora atualmente represente uma seleção direta, a sua utilização nas aplicações cliente permite que, no futuro, se possam adicionar filtros de segurança (ex: ocultar embarcações descomissionadas) sem alterar o código das aplicações que consomem esta informação.

```
1 CREATE VIEW TodasEmbarcacoes AS
2     SELECT * FROM Embarcacao;
```

SQL

5.6. Tradução das Interrogações do Utilizador para SQL

Nota: Consultar as queries desenvolvidas na íntegra no anexo [IX] DQL - Interrogações

5.6.1. Listagem de viagens

A primeira query SQL foi obtida através da tradução direta da expressão AR previamente apresentada no subcapítulo Validação do Modelo com Interrogações do Utilizador. Esta efetua a seleção dos atributos estruturais essenciais (identificador, portos e estado) de todos os registo da tabela Viagem, fornecendo uma visão ampla das operações da frota sem aplicação de filtros condicionais.

```
1 SELECT ID, PortoOrigem, PortoDestino, Estado FROM Viagem;
```

 SQL

5.6.2. Identificação de viagens ativas para atualização de estado

Similarmente à query anterior, esta segunda instrução foi obtida através da tradução direta da correspondente expressão em Álgebra Relacional. Esta apresenta o subconjunto de expedições que se encontram a decorrer, através da filtragem da tabela Viagem onde o atributo Estado é igual a 'Em Curso', permitindo à gestão identificar rapidamente quais os registo que podem transitar para "Finalizada" ou "Cancelada".

```
1 SELECT ID, PortoOrigem, PortoDestino, Estado
2 FROM Viagem
3 WHERE Estado = 'Em Curso';
```

 SQL

5.6.3. Validação de estado contratual de tripulante, cargo e estado (exemplo ID = 1)

A tradução da expressão AR permitiu a obtenção desta query, crítica para a integridade dos Recursos Humanos. Esta efetua uma junção entre a tabela Tripulante e a entidade associativa TripulanteViagem, filtrando pelo identificador do tripulante (exemplo ID=1) e garantindo que apenas são devolvidos registo onde a DataSaida é nula (vínculo ativo). O resultado permite validar o cargo atual antes de qualquer alteração contratual.

```
1 SELECT
2     T.Nome,
3     TV.Cargo,
4     T.Estado
5 FROM Tripulante AS T
6 INNER JOIN TripulanteViagem AS TV
7 ON T.ID = TV.Tripulante
8 WHERE
9     T.ID = 1
10    AND TV.DataSaida IS NULL;
```

 SQL

5.6.4. Histórico operacional de tripulante (exemplo ID=1)

Esta instrução reconstrói a carreira de um tripulante específico. Através de junções sequenciais entre Tripulante, TripulanteViagem e Viagem, é possível projetar o nome do profissional, o cargo desempenhado e os detalhes logísticos (portos e datas) de todas as expedições em que participou no passado.

```
1 SELECT
2     T.Nome,
3     TV.Cargo,
4     V.ID AS ViagemID,
5     V.PortoOrigem,
6     V.PortoDestino,
7     V.DataPartida
8 FROM Tripulante AS T
9 INNER JOIN TripulanteViagem AS TV
10    ON T.ID = TV.Tripulante
11 INNER JOIN Viagem AS V
12    ON TV.Viagem = V.ID
13 WHERE
14     T.ID = 1;
```

5.6.5. Relatório diário de lucro da viagem com base nas receitas e despesas

Esta query consolida a lógica de cálculo financeiro definida na Álgebra Relacional. Utiliza funções de agregação condicional (SUM com CASE) para calcular, numa única passagem pela tabela Financeiro, o total de receitas e o total de despesas agrupado por viagem. O atributo Saldo é gerado aritméticamente pela subtração destas duas agregações, cumprindo o requisito de exploração RE11.

```
1 SELECT
2     Viagem,
3     SUM(CASE WHEN Tipo = 'Receita' THEN Valor ELSE 0 END) AS SomaReceitas,
4     SUM(CASE WHEN Tipo = 'Despesa' THEN ABS(Valor) ELSE 0 END) AS SomaDespesas,
5     SUM(Valor) AS Saldo -- Assume-se que despesas estão registadas como valores
6     negativos
6 FROM Financeiro
7 GROUP BY Viagem;
```

5.6.6. Estatística semanal de capturas por zona e espécie

A implementação deste requisito estatístico foi obtida através da junção entre a tabela Captura e a tabela ZonaPesca, permitindo resolver a referência geográfica para o seu nome legível. Posteriormente, aplica-se um agrupamento composto (GROUP BY) pelo nome da zona e pelo nome comum da espécie, calculando-se o somatório da quantidade capturada e a contagem de registos individuais

```
1  SELECT
2      ZP.Nome AS NomeZona,
3      C.NomeComum,
4      SUM(C.Quantidade) AS TotalKg,
5      COUNT(C.ID) AS NumRegistros
6  FROM Captura AS C
7  INNER JOIN ZonaPesca AS ZP
8      ON C.ZonaPesca = ZP.ID
9  GROUP BY
10     ZP.Nome,
11     C.NomeComum;
```

SQL

5.6.7. Auditoria Integral de Expedição (exemplo ID = 1)

Esta query representa a tradução direta da árvore de execução mais complexa do modelo (Auditoria), unificando os três ramos de informação: Logística, Recursos Humanos e Produção. Através de múltiplas junções (INNER JOIN) centradas numa viagem específica (exemplo ID=1), constrói-se um “dossier” plano que cruza os dados da embarcação, a lista de tripulantes a bordo e o registo detalhado das capturas efetuadas, fundamental para efeitos de fiscalização.

```
1  SELECT
2      V.ID AS ViagemID,
3      E.Nome AS NomeNavio,
4      V.DataPartida,
5      T.Nome AS NomeTripulante,
6      TV.Cargo,
7      C.NomeComum AS Especie,
8      C.Quantidade,
9      ZP.Nome AS NomeZona
10 FROM Viagem AS V
11 INNER JOIN Embarcacao AS E
12     ON V.Embarcacao = E.ID
13 INNER JOIN TripulanteViagem AS TV
14     ON V.ID = TV.Viagem
15 INNER JOIN Tripulante AS T
16     ON TV.Tripulante = T.ID
17 INNER JOIN Captura AS C
18     ON V.ID = C.Viagem
19 INNER JOIN ZonaPesca AS ZP
20     ON C.ZonaPesca = ZP.ID
21 WHERE
22     V.ID = 1;
```

SQL

5.7. Indexação do Sistema de Dados

A indexação permite otimizar as operações de consulta e manipulação de dados, reduzindo significativamente o tempo de resposta do sistema. Um dos aspectos destacados na análise de requisitos foi a necessidade de gerar relatórios diários e semanais, bem como a consulta rápida do estado da frota. Estas indexações visam diretamente atender a essas necessidades, melhorando a eficácia operacional da PortusPesca.

Após uma análise dos padrões de acesso, criámos três indexações com os seguintes propósitos: Implementámos uma indexação na coluna Estado, na tabela Viagem. O objetivo é acelerar a filtragem das viagens ativas (estado “Em Curso”), uma vez que esta é a consulta mais frequente realizada pelos gestores de frota e é a base da vista ViagensEmCurso.

```
1 CREATE INDEX idx_estado_viagem ON Viagem (Estado);
```

 SQL

Criámos a indexação idx_zona_captura na tabela Captura, especificamente na coluna ZonaPesca (chave estrangeira). Esta decisão foi tomada para otimizar a geração do relatório semanal de estatísticas (RE13), que exige o agrupamento e junção frequente de milhares de registos de capturas com as respetivas zonas geográficas.

```
1 CREATE INDEX idx_zona_captura ON Captura (ZonaPesca);
```

 SQL

Para justificar a criação do índice idx_nome_tripulante na tabela Tripulante com base na coluna Nome, podemos mencionar a necessidade de facilitar a busca rápida por tripulantes específicos durante a constituição das equipas para novas expedições, evitando a leitura sequencial de toda a tabela de recursos humanos.

```
1 CREATE INDEX idx_nome_tripulante ON Tripulante (Nome);
```

 SQL

Nota: Ver anexo **[VI] DDL - Definição de Dados** para ver script da indexação da BD.

5.8. Implementação de procedimentos, funções e gatilhos

A utilização de procedimentos, funções e gatilhos em sistemas de bases de dados constitui um elemento fundamental para assegurar maior eficiência, integridade e automatização. Enquanto procedimentos e funções promovem a reutilização de código e simplificam a execução de operações complexas ou repetitivas de forma estruturada, os gatilhos permitem a execução automática de ações em resposta a eventos específicos, reforçando a consistência e a fiabilidade dos dados.

5.8.1. Procedimentos

Para otimizar a execução de determinados requisitos de exploração, recorremos à utilização de procedimentos tornando o processo mais ágil.

Nota: Ver anexo [X] Procedimentos Armazenados

Posto isto, alguns dos procedimentos criados são:

ConsultarViagensTripulante

Este procedimento foi criado com o objetivo de agilizar a consulta das viagens associadas a um determinado tripulante, melhorando a eficiência e a automatização da BD. A sua implementação responde ao requisito RE4 – “Dado o identificador de um tripulante, deve ser possível aceder a todas as viagens em que este esteve envolvido, bem como o cargo desempenhado em cada uma.” Tal como referido, este procedimento recebe como parâmetro o identificador do tripulante e, através da junção das tabelas TripulanteViagem e Viagem, permite obter todos os registo das viagens em que o tripulante participou, incluindo os detalhes da viagem e o cargo desempenhado.

```
1  DELIMITER //
2
3
4  CREATE PROCEDURE ConsultarViagensTripulante(IN p_tripulante INT)
5  BEGIN
6      SELECT V.ID AS ViagemID,
7          V.PortoOrigem,
8          V.PortoDestino,
9          V.DataPartida,
10         V.Estado,
11         TV.Cargo
12     FROM TripulanteViagem AS TV
13     INNER JOIN Viagem AS V ON TV.Viagem = V.ID
14     WHERE TV.Tripulante = p_tripulante;
15 END //
16
17
18 DELIMITER ;
```

ConsultarViagensEmbarcacao

Este procedimento foi criado para facilitar a consulta de todas as viagens realizadas por uma embarcação específica, garantindo maior eficiência e automatização da BD. A sua implementação responde ao requisito RE6 – “Dado o número de matrícula de uma embarcação, o sistema deve permitir consultar todas as viagens realizadas pela mesma, incluindo datas, estados e destinos.” O procedimento recebe como parâmetro a matrícula da embarcação. Primeiramente, verifica se a embarcação existe no sistema. Caso não exista, devolve uma mensagem informativa. Caso exista, retorna todas as viagens associadas à embarcação, incluindo os detalhes da viagem (portos, datas, estado e descrição) e os dados da embarcação (nome e tipo).

```
1  DELIMITER //
2
3
4  CREATE PROCEDURE ConsultarViagensEmbarcacao(
5      IN matricula_embarcacao VARCHAR(8)
6  )
7  BEGIN
8      IF NOT EXISTS (SELECT 1 FROM Embarcacao WHERE ID = matricula_embarcacao) THEN
9          SELECT 'Embarcação não encontrada no sistema.' AS Mensagem;
10     ELSE
11         SELECT
12             V.ID AS ViagemID,
13             V.PortoOrigem,
14             V.PortoDestino,
15             V.DataPartida,
16             V.DataChegada,
17             V.Estado,
18             V.Descricao,
19             E.Nome AS NomeEmbarcacao,
20             E.Tipo AS TipoEmbarcacao
21         FROM Viagem AS V
22         INNER JOIN Embarcacao AS E
23             ON V.Embarcacao = E.ID
24         WHERE
25             V.Embarcacao = matricula_embarcacao
26         ORDER BY
27             V.DataPartida DESC;
28
29     END IF;
30 END //
31
32
33 DELIMITER ;
```

CriaTripulanteEViagem

Este procedimento foi criado para automatizar a criação de um novo tripulante e de uma viagem associada, garantindo que ambos os registo são inseridos de forma consistente e que o tripulante fica imediatamente vinculado à viagem com o cargo indicado. A sua implementação responde a requisitos de manipulação relacionados com a gestão de tripulantes e viagens, permitindo agilizar operações que, de outra forma, exigiriam múltiplos comandos SQL independentes.

O procedimento recebe como parâmetros os dados pessoais do tripulante (incluindo a data de nascimento), os dados da viagem e o cargo que o tripulante irá desempenhar. Através de uma transação, insere primeiro o tripulante, depois a viagem e finalmente cria a associação entre ambos na tabela Tripulante-Viagem. Caso ocorra algum erro durante o processo, a transação é revertida e é devolvida uma mensagem informativa.

Este procedimento garante, assim, a correta criação e associação entre os registo de tripulante e viagem, assegurando a conformidade com os RD2 – “Uma viagem deve ser registada com identificador único, porto de origem, porto de destino, descrição, datas e estado” – e RD10 – “Um tripulante deve ser registado com identificador único, número de cédula marinha, nome completo, data de nascimento, nacionalidade, salário, contacto de emergência, estado, email e morada”.

```
1  DELIMITER //
2
3  CREATE PROCEDURE CriaTripulanteEViagem (
4      -- Dados do Tripulante
5      IN p_nome VARCHAR(150),
6      IN p_cedula VARCHAR(30),
7      IN p_data_nascimento DATE,
8      IN p_salario DECIMAL(10, 2),
9
10     -- Dados da Viagem
11     IN p_embarcacao VARCHAR(8),
12     IN p_origem VARCHAR(50),
13     IN p_destino VARCHAR(50),
14     IN p_descricao TEXT,
15
16     -- Associação
17     IN p_cargo ENUM(
18         'Capitão', 'Imediato', 'Oficial de Navegação', 'Engenheiro',
19         'Mestre de Redes', 'Maquinista', 'Cozinheiro', 'Marinheiro'
20     ),
21
22     OUT msg VARCHAR(255)
23 )
24 BEGIN
25     DECLARE v_tripulante_id INT;
26     DECLARE v_viagem_id INT;
27
28     DECLARE EXIT HANDLER FOR SQLEXCEPTION
29     BEGIN
30         ROLLBACK;
31         SET msg = 'Erro ao criar tripulante e viagem.';
32     END;
33
34     START TRANSACTION;
```

```

35
36    -- 1. Criar Tripulante
37    INSERT INTO Tripulante
38        (CedulaMarinha, Nome, DataNascimento, Salario, Estado)
39    VALUES
40        (p_cedula, p_nome, p_data_nascimento, p_salario, 'Ativo');
41
42    SET v_tripulante_id = LAST_INSERT_ID();
43
44    -- Nacionalidade default
45    INSERT INTO Nacionalidade (Tripulante, Nacionalidade)
46    VALUES (v_tripulante_id, 'Portuguesa');
47
48    -- 2. Criar Viagem
49    INSERT INTO Viagem
50        (Embarcacao, PortoOrigem, PortoDestino, Descricao, DataPartida, Estado)
51    VALUES
52        (p_embarcacao, p_origem, p_destino, p_descricao, NOW(), 'Em Curso');
53
54    SET v_viagem_id = LAST_INSERT_ID();
55
56    -- 3. Associar Tripulante à Viagem
57    INSERT INTO TripulanteViagem
58        (Tripulante, Viagem, Cargo, DataEntrada)
59    VALUES
60        (v_tripulante_id, v_viagem_id, p_cargo, CURDATE());
61
62    COMMIT;
63
64    SET msg = CONCAT(
65        'Tripulante ID ', v_tripulante_id,
66        ' e Viagem ID ', v_viagem_id,
67        ' criados com sucesso.'
68    );
69 END //
70
71 DELIMITER ;

```

5.8.2. Funções

Para a execução de certos requisitos de exploração na área financeira, recorremos ao uso de funções, de modo a agilizar os processos e permitir a reutilização das instruções.

Nota: Ver anexo [XI] **Funções SQL**.

Posto isto, algumas das funções criadas são:

CustoTotalViagem

A função CustoTotalViagem foi criada para calcular automaticamente o custo total associado a uma viagem, permitindo obter de forma eficiente o somatório de todas as despesas registadas. A sua implementação responde ao requisito nº RE9 – “É permitido pelo sistema obter o custo total de uma viagem, através da soma de todos os movimentos financeiros classificados como despesas.”. A função recebe como parâmetro o identificador da viagem e, através da filtragem da tabela Financeiro, calcula o somatório dos valores de todas as despesas associadas. Caso a viagem não possua registo financeiro de despesas, a função retorna o valor zero, garantindo consistência e evitando erros de cálculo.

```
1  DELIMITER //  
2  
3  
4  CREATE FUNCTION CustoTotalViagem(viagem_id INT)  
5    RETURNS DECIMAL(10,2)  
6    DETERMINISTIC  
7    BEGIN  
8        DECLARE custo_total DECIMAL(10,2) DEFAULT 0;  
9  
10  
11        -- Calcular o somatório de todas as despesas da viagem  
12        SELECT COALESCE(SUM(ABS(Valor)), 0)  
13        INTO custo_total  
14        FROM Financeiro  
15        WHERE Viagem = viagem_id AND Tipo = 'Despesa';  
16  
17  
18        RETURN custo_total;  
19    END //  
20  
21  
22  DELIMITER ;
```

CalcularSaldoViagem

A função CalcularSaldoViagem exemplifica a reutilização de instruções através do cálculo direto dos valores de receitas e despesas associados a uma viagem, cujo identificador é passado como argumento. Esta função baseia-se no requisito nº RE10 – “É permitido pelo sistema obter o Saldo de uma viagem, através da subtração do valor obtido em RE8 (receitas) pelo valor obtido em RE9 (custos).”. Similarmente às funções apresentadas anteriormente, se a viagem com o identificador fornecido não possuir registo financeiro, a função CalcularSaldoViagem retorna o valor zero, garantindo consistência e evitando erros de cálculo.

```
1  DELIMITER //
2
3
4  CREATE FUNCTION CalcularSaldoViagem(viagem_id INT)
5    RETURNS DECIMAL(10,2)
6    DETERMINISTIC
7    BEGIN
8      DECLARE saldo DECIMAL(10,2) DEFAULT 0;
9      DECLARE total_receitas DECIMAL(10,2) DEFAULT 0;
10     DECLARE total_despesas DECIMAL(10,2) DEFAULT 0;
11
12
13    -- Calcular total de receitas da viagem
14    SELECT COALESCE(SUM(Valor), 0)
15    INTO total_receitas
16    FROM Financeiro
17    WHERE Viagem = viagem_id AND Tipo = 'Receita';
18
19
20    -- Calcular total de despesas da viagem (valor absoluto)
21    SELECT COALESCE(SUM(ABS(Valor)), 0)
22    INTO total_despesas
23    FROM Financeiro
24    WHERE Viagem = viagem_id AND Tipo = 'Despesa';
25
26
27    -- Calcular lucro/prejuízo
28    SET saldo = total_receitas - total_despesas;
29
30
31    RETURN saldo;
32 END //
33
34
35 DELIMITER ;
36
```

5.8.3. Gatilhos

Através do uso de gatilhos em SQL, foi possível automatizar a execução de certos requisitos de manipulação crítica, garantindo a consistência e integridade dos dados sem dependência exclusiva da camada aplicacional.

Nota: Ver anexo [XII] **Gatilhos(Triggers)**.

Deste modo, foram desenvolvidos os seguintes gatilhos:

UpdateViagemEstado

Baseado no requisito RE1 – “Quando uma viagem é finalizada ou cancelada o seu estado é atualizado respetivamente, assim como a data de chegada e é feita a desassociação da tripulação da viagem” –, o gatilho UpdateViagemEstado permite a atualização automática da data de chegada de uma expedição, assim como efetua o registo de saída de todos os tripulantes que se encontravam embarcados. Primeiramente, antes da atualização de um registo da tabela Viagem, verifica-se se o estado antigo era “Em Curso” e se o novo estado é “Finalizada” ou “Cancelada”. Se tal acontecer, a data de chegada (DataChegada) é preenchida automaticamente com a data atual. Simultaneamente, na tabela TripulanteViagem, todos os registos associados a essa viagem que ainda não tenham data de saída (DataSaída nula) são atualizados com a data corrente, libertando a tripulação.

```
1  DELIMITER //
2  CREATE TRIGGER UpdateViagemEstado
3  BEFORE UPDATE ON Viagem
4  FOR EACH ROW
5  BEGIN
6      -- Verifica se o estado da viagem passou de 'Em Curso' para um estado final
7      IF OLD.Estado = 'Em Curso' AND NEW.Estado IN ('Finalizada', 'Cancelada') THEN
8
9          -- Atualiza a data de chegada da viagem automaticamente
10         SET NEW.DataChegada = CURRENT_TIMESTAMP;
11
12
13         -- Desassocia (regista a saída) de todos os tripulantes desta viagem
14         UPDATE TripulanteViagem
15         SET DataSaída = CURRENT_DATE
16         WHERE Viagem = NEW.ID AND DataSaída IS NULL;
17
18     END IF;
19 END //
20 DELIMITER ;
```

UpdateTripulanteEstado

Baseado no requisito RE5 – “Quando um tripulante deixa de exercer a sua função [...] o seu atributo ‘estado’ [...] deve ser atualizado” – e na necessidade de manter a consistência da frota, o gatilho UpdateTripulanteEstado assegura que um tripulante que deixa de estar ativo na empresa não permaneça erroneamente associado a viagens em curso.

Antes da atualização de um registo na tabela Tripulante, verifica-se se o estado foi alterado de “Ativo” para “Inativo”. Se esta condição se verificar, o sistema percorre a tabela TripulanteViagem e fecha qualquer embarque aberto desse tripulante, definindo a DataSaida como a data atual. Isto previne que tripulantes demitidos ou reformados constem como embarcados em relatórios operacionais.

```
1  DELIMITER //
2  CREATE TRIGGER UpdateTripulanteEstado
3  AFTER UPDATE ON Tripulante
4  FOR EACH ROW
5  BEGIN
6      -- Verifica se o tripulante passou para 'Inativo'
7      IF OLD.Estado = 'Ativo' AND NEW.Estado = 'Inativo' THEN
8
9          -- Fecha qualquer participação ativa em viagens
10         UPDATE TripulanteViagem
11         SET DataSaida = CURRENT_DATE
12         WHERE Tripulante = NEW.ID AND DataSaida IS NULL;
13
14     END IF;
15 END //
16 DELIMITER ;
```

6. Conclusões e Trabalho Futuro

Em suma, a equipa de trabalho concluiu com sucesso todas as etapas planeadas para o desenvolvimento do Sistema de Gestão de Base de Dados da PortusPesca - Operações Marítimas, Lda. O projeto permitiu a transição de um sistema de registos fragmentados para uma solução digital integrada, robusta e escalável.

No que diz respeito ao trabalho realizado, foram atingidos os seguintes marcos: Definição do Sistema: Estabelecimento claro do âmbito, motivações e objetivos do sistema, com foco na centralização e rastreabilidade da atividade pesqueira.

- **Levantamento e Análise de Requisitos:** Recolha exaustiva das necessidades operacionais da frota, resultando numa lista validada de requisitos de Descrição, Exploração e Administração.
- **Modelação Conceptual:** Criação de um modelo Entidade-Relacionamento (ER) capaz de representar fielmente a complexidade das viagens, tripulações, capturas e fluxos financeiros.
- **Modelação Lógica:** Refinamento do modelo conceptual para um esquema relacional normalizado, garantindo a integridade dos dados e a eliminação de redundâncias.
- **Implementação Física:** A concretização do sistema no SGBD MySQL, incluindo a criação de tabelas com restrições de integridade, povoamento de dados, e cálculo de volumetria.
- **Otimização e Programação:** Implementação de vistas para facilitar a consulta de gestores, índices para melhorar a performance de pesquisas frequentes, e o desenvolvimento de procedimentos, funções e gatilhos para automatizar regras de negócio críticas (como o fecho de viagens e cálculo de lucros).

A solução apresentada cumpre integralmente os requisitos propostos, mantendo a simplicidade necessária para uma utilização eficiente, sem comprometer a segurança e a consistência da informação.

Trabalho Futuro

Relativamente a trabalhos futuros, e numa ótica de evolução estritamente focada no modelo de dados, identificamos a oportunidade de refinar a entidade Captura. Atualmente, os dados das espécies (nome comum e científico) residem como atributos textuais; a criação de uma entidade autónoma Especie (numa relação 1:N) eliminaria a repetição de dados textuais e permitiria, por exemplo, a associação a uma nova entidade QuotaLegal, possibilitando que a própria base de dados validasse inserções de capturas contra os limites legais de pesca vigentes por zona.

Este projeto estabelece, assim, uma fundação sólida para a modernização tecnológica do Grupo Portus, garantindo a sua competitividade num setor cada vez mais exigente.

Bibliografia

- [1] Connolly, T., & Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management (6th ed.). Pearson Education. London, UK.
- [2] Reis, J., & Housley, M. (2022). Fundamentals of Data Engineering: Plan and Build Robust Data Systems (1st ed.). O'Reilly Media. Sebastopol, CA.
- [3] MySQL 8.0 Reference Manual (2024). MySQL 8.0 Reference Manual: Data Type Storage Requirements. MySQL, Oracle.

Lista de Siglas e Acrónimos

BD Base de Dados

SGBD Sistema de Gestão de Base de Dados

AR Álgebra Relacional

PPOM PortusPesca - Operações Marítimas

RD Requisitos de Descrição

RE Requisitos de Exploração

RA Requisitos de Administração

ER Entidade-Relacionamento

ID Identificador

SQL *Structured Query Language*

DDL *Data Definition Language*

DCL *Data Control Language*

DQL *Data Query Language*

DML *Data Manipulation Language*

DTL *Data Transaction Language*

Anexos

[I] Diagrama de Gantt

[II] Documento para Requisitos

[III] Modelo Conceptual

[IV] Modelo Lógico

[V] Expressões de AR - ReLaX

[VI] DDL - Definição de Dados

[VII] DCL - Controlo de Dados

[VIII] DML - Povoamento

[IX] DQL - Interrogações

[X] Procedimentos Armazenados

[XI] Funções SQL

[XII] Gatilhos (Triggers)

[XIII] Repositório GitHub