Departamento de Engenharia Informática e de Sistemas

Instituto Superior de Engenharia de Coimbra

*Curso Europeu de Informática*

Bases de Dados

2018/2019

Base de Dados

para uma Distribuidora de Combustíveis

Grupo Nº 8 – P3

21270348 – Bruno Miguel Venâncio Rodrigues

21270440 – Diogo Anselmo Henriques

21270422 – João Francisco de Matos Claro

**Índice**

[1. Introdução 2](#_Toc534045426)

[2. Enquadramento da proposta 3](#_Toc534045427)

[2.1. Diagnóstico da Situação Atual 3](#_Toc534045428)

[2.2. Problemas encontrados 3](#_Toc534045429)

[2.3. Descrição da Solução Proposta 3](#_Toc534045430)

[3. Funcionalidades da aplicação 4](#_Toc534045431)

[3.1. Requisitos e/ou Restrições de implementação 4](#_Toc534045432)

[3.2. Estrutura da Aplicação 4](#_Toc534045433)

[3.3. Módulos da aplicação e protótipos 5](#_Toc534045434)

[4. Análise de Dados 16](#_Toc534045435)

[4.1. Entidades 16](#_Toc534045436)

[4.1.1. Entidade Motoristas 16](#_Toc534045437)

[4.1.2. Entidade Viagens 17](#_Toc534045438)

[4.1.3. Entidade Tratores 18](#_Toc534045439)

[4.1.4. Entidade Cisternas 19](#_Toc534045440)

[4.1.5. Entidade Previsões 19](#_Toc534045441)

[4.1.6. Entidade Postos 20](#_Toc534045442)

[4.1.7. Entidade Fornecedores 21](#_Toc534045443)

[4.2. Relacionamentos 22](#_Toc534045444)

[4.2.1. Relacionamento: conduz 22](#_Toc534045445)

[4.2.2. Relacionamento: fazem 23](#_Toc534045446)

[4.2.3. Relacionamento: binómio 23](#_Toc534045447)

[4.2.4. Relacionamento: segue 24](#_Toc534045448)

[4.2.5. Relacionamento: formam-se/instrução 25](#_Toc534045449)

[4.2.6. Relacionamento: recebem 26](#_Toc534045450)

[4.2.7. Relacionamento: entregam 26](#_Toc534045451)

[5. Modelo Físico 28](#_Toc534045452)

[5.1. Tabelas 28](#_Toc534045453)

[5.1.1. Tabela Motoristas 28](#_Toc534045454)

[5.1.2. Tabela Viagens 29](#_Toc534045455)

[5.1.3. Tabela Tratores 29](#_Toc534045456)

[5.1.4. Tabela Cisternas 29](#_Toc534045457)

[5.1.5. Tabela Previsões 30](#_Toc534045458)

[5.1.6. Tabela Postos 30](#_Toc534045459)

[5.1.7. Tabela Fornecedores 30](#_Toc534045460)

[5.1.8. Tabela Binomio 31](#_Toc534045461)

[5.1.9. Tabela Conduz 31](#_Toc534045462)

[5.1.10. Tabela Formação 31](#_Toc534045463)

[5.1.11. Tabela Recebem 32](#_Toc534045464)

[5.1.12. Tabela Segue 32](#_Toc534045465)

[5.2. Diagrama do Modelo Físico 33](#_Toc534045466)

[5.3. Script de criação da Base de Dados 34](#_Toc534045467)

# Introdução

O objetivo deste projeto é implementar um sistema que permita efetuar de uma forma automatizada o processo de carregamento e distribuição de combustíveis, com base na situação do GRUPO ALVES BANDEIRA, SGPS, SA.

O GRUPO ALVES BANDEIRA é uma empresa focada no ramo automóvel que conta já com cerca de 150 postos de abastecimento em Portugal.

Este grupo conta com um total de 16 empresas diferentes, sendo a ALVES BANDEIRA a sub empresa principal, responsável pelo comércio de combustível, onde focámos a criação da base de dados para facilitação do controlo logístico.

No capítulo 2, faz-se um enquadramento da relevância da proposta na distribuição de combustíveis, descreve-se a situação atual, os principais problemas encontrados e, por fim, descreve-se a proposta que irá permitir solucionar esses problemas e atingir o objetivo desejado.

No capítulo 3, são descritas as funcionalidades da aplicação que irá interagir com a base de dados, apresentando-se os seus módulos e detalhando as suas restrições de funcionamento.

No capítulo 4, apresenta-se a análise de dados necessária para a solução proposta, descrevendo-se todas as entidades de dados e os relacionamentos existentes entre elas.

No capítulo 5, apresenta-se o modelo físico da base de dados que vai servir de suporte ao sistema de informação proposto, assim como o respetivo *script* de criação da base de dados.

No capítulo 6, são apresentadas consultas (código SQL) das principais funcionalidades de pesquisas especificadas na solução proposta para a distribuição.

Por fim, no capítulo 7 são apresentadas conclusões do trabalho desenvolvido.

# Enquadramento da proposta

* 1. Diagnóstico da Situação Atual

**Atualmente toda a informação é feita em papel e a partilha da mesma é feita por e-mail.**

Esta forma de partilha de informação é demorada e faltosa, gerando perdas de informação e falhas na comunicação.

A atualização das rotas e a sua otimização são feitas por um grupo de pessoas o que costuma gerar erros.

Atualmente os postos são abastecidos em conformidade com o consumo semanal, o stock tem de ser suficiente para mais de 3 dias.

Os motoristas têm os seus veículos designados perto de uma das 3 refinarias do país (Matosinhos, Aveiras e Sines) para no início do dia de trabalho poderem carregar a cisterna o mais cedo possível e iniciar a sua rota.

* 1. Problemas encontrados
* **A recolha e envio de informação é um processo lento e caro, pois é feito em papel, sendo necessário enviar toda a informação manualmente.**
  + Com o uso da base de dados, a informação, estaria disponível automaticamente bastando para isso estar ligado à rede da empresa
* **O controlo dos funcionários é bastante difícil pois não há forma de verificar a sua entrada e saída.**
  + Esta base de dados conseguiria ter acesso ao pica-ponto dos funcionários bem como os seus salários, prémios e todo o tipo de informação relevante ao mesmo.
* **Algumas rotas não estão otimizadas, o que gera prejuízo.**
  + Com este sistema seria possível analisar mais facilmente cada situação e corrigi-la caso seja necessário.
  1. Descrição da Solução Proposta

A solução proposta baseia-se numa base de dados ligada à rede do departamento de distribuição da empresa que será capaz de:

Relativamente à frota:

A base de dados guardará informação dos veículos, e da sua informação respetiva desde marca até quilómetros andados e viagens feitas.

Relativamente ao pessoal:

Será guardada a informação pessoal de todo o funcionário assim como o seu cargo na empresa e seus salários e o seu pica-ponto

Relativamente às viagens:

A informação das viagens conterá, o seu ponto de partida, paragens e ponto de chegada, assim como o seu horário e o respetivo pessoal.

# Funcionalidades da aplicação

* 1. Requisitos e/ou Restrições de implementação

Nesta secção enumeram-se as restrições que a aplicação deve satisfazer de modo a permitir um correto funcionamento do negócio de venda de livros. As restrições a implementar são as seguintes:

* Não deve ser permitida a venda de um livro que não exista em *stock*, isto é, apenas deve ser possível efetuar vendas de livros cuja quantidade em *stock* (quant\_em\_stock) seja superior a zero.
* Não deve ser permitida a venda de uma quantidade de livros superior à existente em *stock*, isto é, a quantidade da venda deve ser inferior ou igual à quantidade em *stock* (quant\_em\_stock).
* Quando se procede a uma venda, se a escolha do livro for efetuada através do título do livro, apenas deve aparecer a lista dos títulos que existem em *stock* (quant\_em\_stock superior a zero).
* Quando se procede a uma venda, deve ser possível a escolha do livro a vender através do código interno ou do ISBN, de modo a permitir uma futura utilização de leitores de código de barras.
* Quando se insere um novo livro ou se altera a informação do autor e/ou da editora de um livro, a escolha do autor e/ou editora não deve ser diretamente efetuada através dos respetivos códigos internos, mas sim através dos respetivos nomes.
* Quando se insere um novo livro e aquando da seleção do respetivo autor se verifique que não existe o pretendido, deve ser possível a introdução desse novo autor na base de dados, sem cancelar a informação relativo ao livro que eventualmente já tenha sido introduzida.
* Quando se insere um novo livro e aquando da seleção da respetiva editora se verifique que não existe a pretendida, deve ser possível a introdução dessa nova editora na base de dados, sem cancelar a informação relativo ao livro que eventualmente já tenha sido introduzida.
* Quando se regista uma venda e aquando da seleção do cliente que vai comprar o livro, se verifique que esse cliente ainda não existe na base de dados da livraria, deve ser possível a introdução desse novo cliente na base de dados, sem cancelar a informação relativa à venda que eventualmente já tenha sido introduzida
  1. Estrutura da Aplicação

A aplicação deve utilizar uma arquitetura cliente/servidor que irá questionar o servidor de base de dados para aceder à informação desejada. Esta aplicação será instalada em cada um dos *smartphones* dos motoristas, assim como duas páginas web, uma para a central (que tem acesso a informações sensíveis) e outra para os postos, que podem dar *update* aos seus valores, sendo o acesso à base de dados efetuado através da utilização de *drivers* nativos do Oracle.

Para permitir futuras evoluções, esta deve estar estruturada de uma forma modular, que permita facilmente adicionar novos módulos à aplicação. A utilização de uma arquitetura modular irá permitir a implementação faseada dos vários módulos que são propostos, e a possibilidade de futuramente ser possível adicionar novos, além dos já existentes, na mesma interface comum. Esta modularidade permitirá evitar a existência de várias aplicações, com objetivos distintos, mas que necessariamente irão possuir funcionalidades redundantes.

A aplicação móvel é simples e intuitiva para os motoristas.

A página web será o *hub* onde serão feitas as operações mais importantes, e onde importa a modularidade da base de dados. Poderão ser realizadas aqui as mudanças à base de dados nos vários módulos:

* Gestão de Viagens
* Gestão de Tratores
* Gestão de Motoristas
* Gestão de Fornecedores
* Gestão de Postos
* Gestão de Previsões
* Gestão de Cisternas
  1. Módulos da aplicação e protótipos

**Gestão de Viagens**

Após a seleção da opção “Viagens” no menu principal da aplicação deve ser apresentada a interface abaixo (Figura 1):

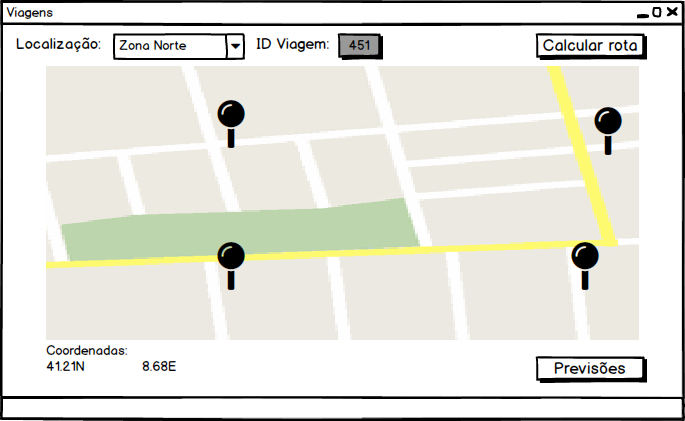


Figura 1 – interface da gestão de Viagens

O módulo de gestão de viagens deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação de viagens. Nomeadamente:

* Verificar a quantidade de postos nas diferentes localizações.
* Geração automática do código interno que será associado a essa viagem. Este código deve ser gerado duma forma sequencial.
* Verificar as coordenadas de um posto selecionado
* Cálculo automático da melhor rota possível (útil em caso de expansão da empresa)
* É ainda possível usar o botão “Previsões” para aceder à interface seguinte (Figura 2):

**Gestão de Previsões**

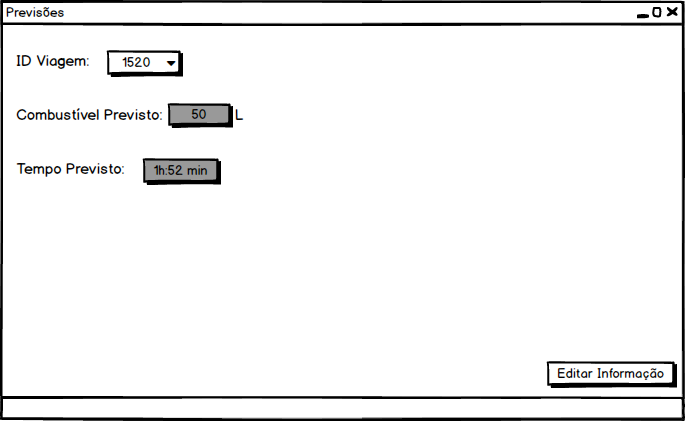


Figura 2 – interface de gestão de previsões

* Depois de aceder às previsões, através do botão no menu principal com esse efeito, chegará a esta interface (Figura 10), que permite verificar as previsões de combustível e tempo para o ID da viagem selecionado (Observação: O ID é o da viagem, pois apesar de existir um campo ID para a previsão, é criada uma previsão para cada viagem, portanto os IDs são iguais)
* Para editar essa informação é necessário carregar em “Editar informação” que o levará à Figura 3:

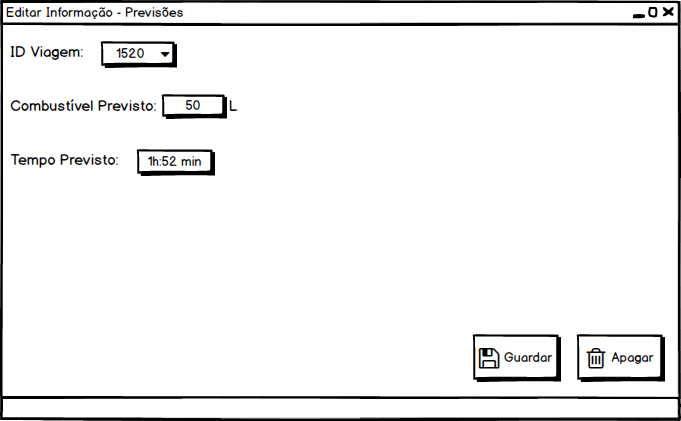


Figura 3 - interface da edição de informação das previsões

* Na edição é possível escolher qual a previsão que precisa de atualização. Por *default* é o mesmo da página anterior
* Aqui pode ser mudado o combustível previsto para essa viagem, assim como o tempo.
* Se necessário, também se pode apagar essa previsão (o que também apagará a entrada com o mesmo ID na tabela Viagens)
* Ao clicar em Guardar, a página é reencaminhada para o menu principal

**Gestão de Motoristas**

Após a seleção da opção “Motoristas” no menu principal da aplicação deve ser apresentada a interface abaixo (Figura 4):

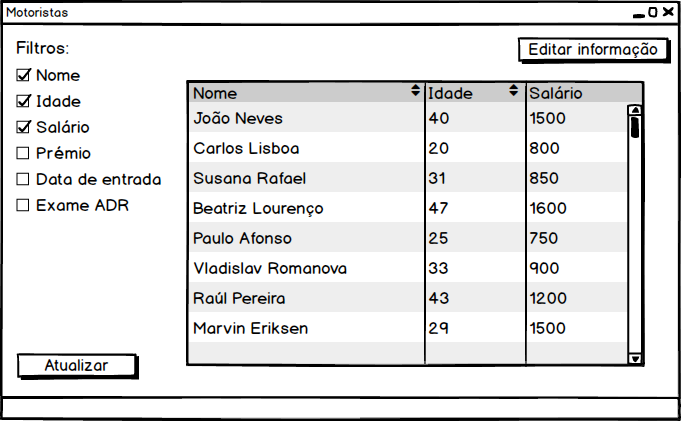


Figura 4 – interface da gestão de motoristas

* Aqui podem ser encontrados todos os motoristas na base de dados, filtrados pelas *checkboxes* e atualizando no botão “Atualizar” que mostrará todos os motoristas com as informações selecionadas.
* É possível ainda editar essa informação, carregando no botão “Editar informação” que o leva à figura 5:

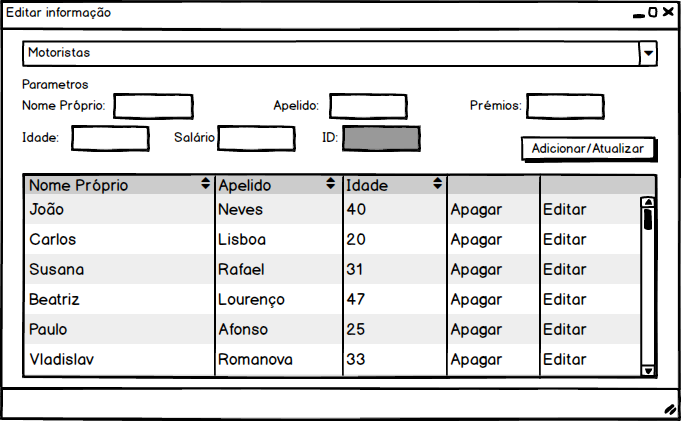


Figura 5 – interface de edição de informação ou remoção de motoristas

* O nome próprio e apelido têm que ser escritos em maiúsculas, não nulo
* O prémio é obrigatoriamente um valor entre 0 e 3000, não nulo
* A idade é um valor obrigatório entre 18 e 65
* O salário é um valor obrigatório e superior a 580 (salário mínimo nacional à data de entrega)
* O ID é um valor que é mostrado automaticamente dependendo do motorista selecionado, um valor de 4 dígitos, não nulo, criado automaticamente pelo código SQL por meio de uma sequência
* Também é possível apagar ou editar os dados de um motorista, carregando no “Apagar” ou “Editar” da linha pretendida, respetivamente. Ao carregar no “Editar”, os campos acima da tabela são preenchidos automaticamente e podem ser alterados, à exceção do ID

**Gestão de Tratores**

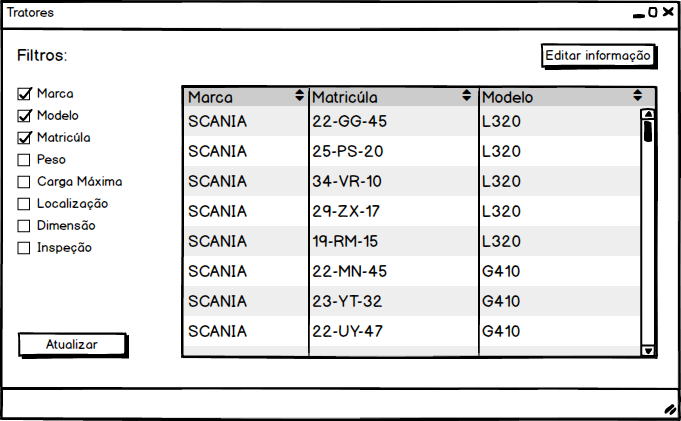


Figura 6 - interface de gestão dos tratores

* Depois da escolha da opção “Tratores” no menu principal, chega-se à Figura 6, que permite filtrar os tratores da frota. Para isso, é possível filtrar por todas as opções presentes na figura, que podem depois ser atualizadas utilizando o botão “Atualizar”
* Ao selecionar o botão “Editar informação” a aplicação leva-o a poder editar as informações, conforme a Figura 7, em baixo

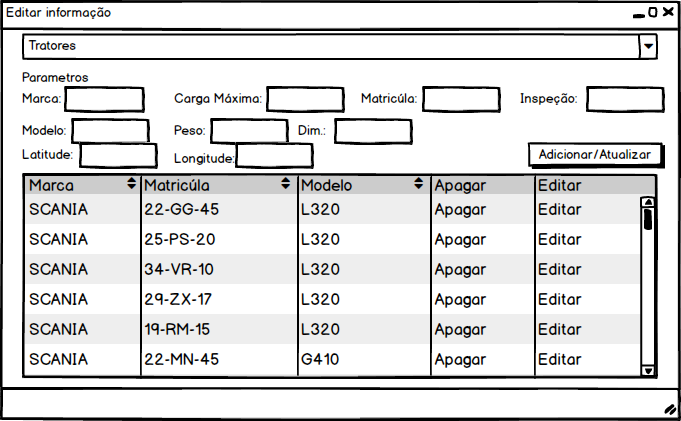


Figura 7 - edição da informação dos tratores

* Esta página, à semelhança da Figura 5, permite adicionar, editar ou apagar todos os tratores da empresa
* A marca é um valor não nulo, obrigatoriamente em maiúsculas
* A matrícula é um valor obrigatório e único, que é o identificador da tabela
* O peso é o do trator, expresso em quilogramas, não nulo
* A inspeção é uma data, não nulo, indicadora da última inspeção realizada ao trator, que tem que ser inferior à data atual (*sysdate*)
* Carga máxima é o peso máximo que o trator pode levar, não nulo e não negativo, expresso em toneladas
* Latitude e longitude são os últimos valores registados pelo motorista para efeitos de localização. São valores não nulos e *varchar,* pois é necessário indicar a orientação
* Dim é a dimensão do trator, não nulo e não negativo
* O modelo, não nulo, é o modelo do trator, que terá que ser inserido em maiúsculas
* Estes valores podem ser introduzidos para inserir um novo trator à base de dados. Para apagar ou editar um trator já existente, é necessário procurar e apagar ou editar, usando os respetivos sítios para o efeito
* Ao utilizar o botão Editar, os valores aparecem automaticamente nas caixas de texto e podem ser editados.

**Gestão de Cisternas**

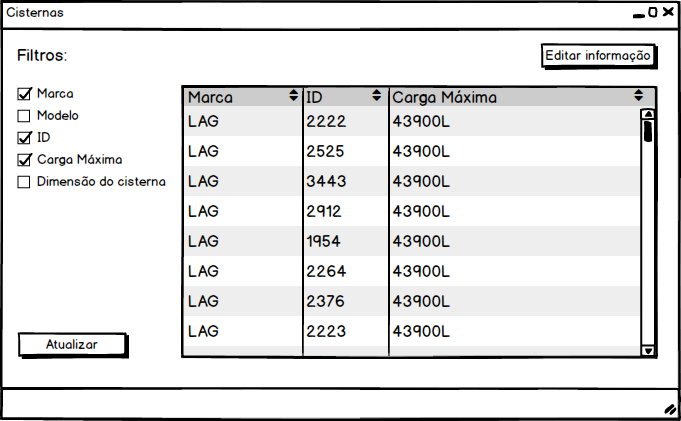


Figura 8 - pesquisa de cisternas

* Nesta página, depois do acesso pelo menu através do botão “Cisternas” é possível verificar todas as cisternas da frota filtrando pelo necessário.
* A Marca é a marca da cisterna, em maiúsculas, que não pode ser nulo
* O Modelo é o modelo da cisterna, em maiúsculas, que não pode ser nulo
* O ID, único e identificador, é o ID atribuído automaticamente através de uma sequência em SQL
* A dimensão da cisterna é não nula e não negativa
* É possível utilizar o botão “Atualizar” para fazer a atualização para outra combinação de filtros. Para além disso, pode ser usado o botão “Editar informação”, que o levará à interface seguinte (Figura 9)

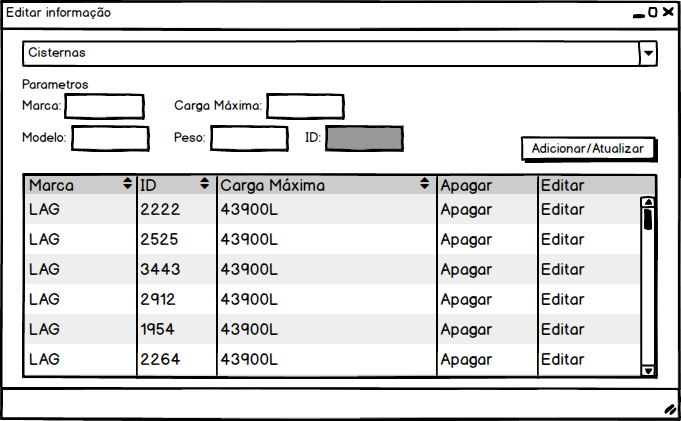


Figura 9 - edição de informação das cisternas

* Nesta interface, à semelhança das já mencionadas, é possível adicionar, apagar ou editar informação existente.
* Ao editar informação já existente, os parâmetros são inseridos automaticamente nas caixas de texto, com possibilidade de os mudar (à exceção do ID)

**Gestão de Postos**

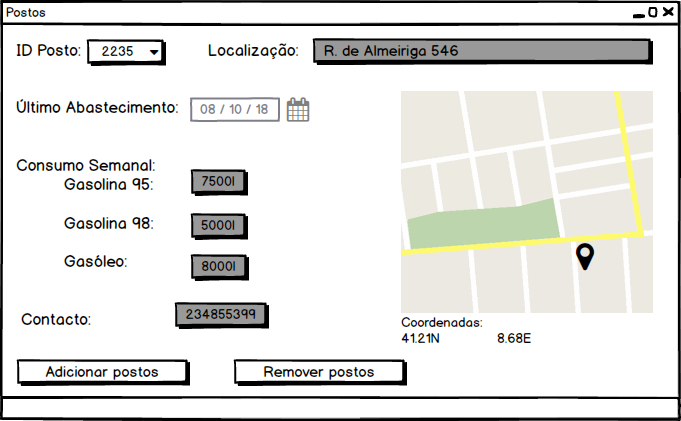


Figura 10 - interface de pesquisa de postos

* Na página “Postos”, após selecionar a opção com o mesmo nome no menu principal, chega-se a esta página, que permite visualizar:
* ID Posto: O identificador da tabela. Não existem 2 postos com o mesmo código
* Gasolina 95, gasolina 98 e gasóleo: Consumo semanal de gasolina 95, gasolina 98 e gasóleo no posto em questão. São obrigatórios e não nulos (cada um tem o seu atributo específico)
* Último abastecimento: É uma data, igual ou inferior à data atual (*sysdate*)
* Longitude e latitude do posto: Um número com um caracter identificador da orientação (traduzido num *varchar* em SQL). Cada um tem um atributo específico
* Contacto: O contacto telefónico do posto, com 9 dígitos
* Nesta página, aparte do ID, nenhum dos atributos pode ser alterado. As alterações são reservadas para a página web de cada posto (não faria sentido a central poder alterar os valores que são intrínsecos a cada posto)
* É possível ainda carregar nos botões “adicionar postos” / “remover postos” que leva às seguintes páginas, respetivamente:



Figura 11 - interface de adição de postos

* Nesta página, é possível ver qual será o ID do novo posto (feito automaticamente através de uma sequência)
* É necessário adicionar a localização do novo posto, que por sua vez irá atualizar o mapa e as coordenadas (coordenadas que serão guardadas)
* Ao carregar em Guardar estas alterações serão guardadas

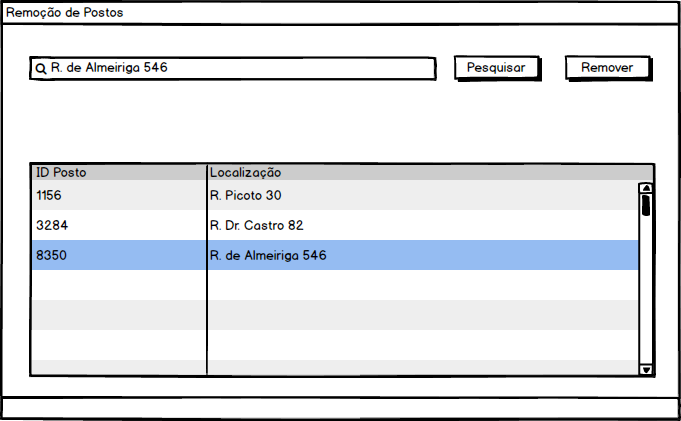


Figura 12 - interface de remoção de postos

* Nesta interface podem-se remover postos
* Ao utilizar a barra de pesquisa pode-se procurar a localização do posto
* Depois de clicar em pesquisar, aparece na tabela abaixo os resultados, onde se pode selecionar um e clicar no botão remover que irá eliminar os dados selecionados da tabela

**Gestão de Fornecedores**



Figura 13 - interface dos fornecedores

* Após selecionar a opção “Fornecedores” no menu inicial, chega-se a esta página, onde é possível verificar informações dos Fornecedores
* Neste caso, o nome é o identificador primário, isto porque apenas existem 3 refinarias, localizadas em 3 sítios diferentes.
* O contacto da refinaria é um número de telefone com 9 dígitos
* A localização indica a morada da refinaria

Em todas as interfaces, à exceção do menu principal, aquando de qualquer alteração final de dados, a página retorna automaticamente ao menu principal.

# 

# Análise de Dados

Neste capítulo pretende-se descrever detalhadamente todas as entidades envolvidas na solução proposta. Na subsecção 4.1 são descritas todas as entidades (respetivos atributos e restrições), na secção 4.2 são descritos todos os relacionamentos existentes entre as diversas entidades, apresentando-se por fim, na secção 4.3, o modelo de Entidade/Relacionamento completo, incluindo todas as entidades e relações relevantes.

* 1. Entidades

Nesta secção vão ser descritas todas as Entidades relevantes para o negócio de venda da distribuidora. Após uma análise aprofundada do modelo de negócio da empresa, constatou-se a necessidade das seguintes Entidades:

* Motoristas
* Viagens
* Tratores
* Cisternas
* Previsões
* Postos
* Fornecedores
  + 1. Entidade Motoristas

A entidade Motoristas representa a informação relativa aos motoristas que existem na empresa. Esta entidade apresenta todos os motoristas que existem. É inserido um novo registo nesta entidade sempre que a empresa pretende adicionar um condutor novo.

Atributos relevantes da Entidade:

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome do atributo** | **Tipo de Dados** | **Aceita Nulos?** | **Descrição** |
| ID\_MOTORISTA | Numérico  (4 dígitos) | N | Código interno único atributo a cada motorista. É um número de 4 dígitos gerado sequencialmente. Ex. 9223 |
| NOME\_PROPRIO | 256 Caracteres | N | Nome do motorista em maiúsculas, tendo no máximo 256 caracteres. Ex. JOSE |
| APELIDO | 256 Caracteres | N | Apelido do motorista em maiúsculas, tendo no máximo 256 caracteres. Ex. VENTURA |
| PREMIOS | Numérico  (24dígitos, 2 decimais) | N | Prémios adquiridos pelos motoristas ao longo da carreira. O valor é expresso em euros. Ex. 12200 |
| SALARIO | Numérico  (24dígitos, 2 decimais) | N | Salário do motorista no inicio de carreira. O valor é expresso em euros. Ex. 12.79 € |
| EXAME\_ADR | Data | S | Data em que o motorista obteve o exame. Ex. 26-11-2003 |
| DATA\_ENTRADA | Data | N | Data em que o motorista iniciou na empresa. Ex. 26-11-2003 |
| IDADE | Numérico  (2 dígitos) | N | Idade do motorista. Ex 50 |

Restrições dos atributos da Entidade:

|  |  |  |
| --- | --- | --- |
| **Nome do atributo** | **Valores Únicos?** | **Observações** |
| ID\_MOTORISTA | S | Identificador (chave primária). Não existem dois motoristas com o mesmo código. |
| NOME\_PROPRIO | N | Só admite maiúsculas. |
| APELIDO | N | Só admite maiúsculas. |
| PREMIOS | N | Valor positivo e inferior a 3000.00 € |
| SALARIO | N | Valor positivo e superior a 580.00 € |
| EXAME\_ADR | N | Data inferior à data atual |
| DATA\_ENTRADA | N | Data inferior à data atual |
| IDADE | N | Valor não nulo superior a 18 e inferior a 65 |

Relacionamentos da Entidade:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nome do relacionamento** | **Cardinalidade** | **Entidade Relacionada** | **Participação**  **Obrigatória** | **Página** |
| Conduz | N : N | TRATORES | NENHUMA | X |
| Fazem | N : N | VIAGENS | NENHUMA | Y |
| Formam-se | 1 : N | FORMACAO | MOTORISTA, FORMACAO | Z |

* + 1. Entidade Viagens

A entidade Viagens representa a informação relativa as viagens. Esta entidade representa todas as viagens efetuadas. Todas as viagens realizadas estão registadas nesta entidade. É inserido uma nova viagem nesta Entidade sempre que a mesma for realizada.

Atributos relevantes da Entidade:

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome do atributo** | **Tipo de Dados** | **Aceita Nulos?** | **Descrição** |
| ID\_VIAGEM | Numérico  (4 dígitos) | N | Código interno único atributo a cada viagem. É um número de 4 dígitos gerado sequencialmente. Ex. 9223 |
| COMBUSTIVEL\_GASTO | Numérico  (4 dígitos, 2 decimais) | N | Quantidade de combustível gasto na viagem. O valor é expresso em litros Ex. 40 |
| TEMPO\_VIAGEM | Tempo | N | Tempo demorado na viagem. Ex. 2:15 |

Atributos com restrições:

|  |  |  |
| --- | --- | --- |
| **Nome do atributo** | **Valores Únicos?** | **Observações** |
| ID\_VIAGEM | S | Identificador (chave primária). Não existem duas viagens com o mesmo código. |
| COMBUSTIVEL\_GASTO | N | Valor não nulo não negativo |
| TEMPO\_VIAGEM | N | Valor não nulo não negativo |

Relacionamentos da Entidade:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nome do relacionamento** | **Cardinalidade** | **Entidade Relacionada** | **Participação**  **Obrigatória** | **Página** |
| Fazem | N : N | MOTORISTAS | NENHUMA | Y |
| Segue | N : N | PREVISOES | NENHUMA | Z |
| Entregam | N : N | POSTOS | POSTOS | X |
| Recebem | N : N | FORNECEDORES | NENHUMA | A |
| Binómio | M : N : P | TRATORES, CISTERNAS | NENHUMA | B |

* + 1. Entidade Tratores

A entidade Tratores representa a informação relativa aos tratores. Esta entidade não representa todos os tratores que existem, mas apenas os que estão preparados para realizarem viagens. É inserido um novo trator nesta Entidade sempre que este estiver preparado.

Atributos relevantes da Entidade:

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome do atributo** | **Tipo de Dados** | **Aceita Nulos?** | **Descrição** |
| MATRICULA | 8 Caracteres | N | Matricula atribuída ao trator. Ex. 00-AA-00 |
| PESO | Numérico  (4 dígitos, 2 decimais) | N | Peso do trator. O valor é expresso em quilogramas Ex. 4000 |
| INSPECOES\_ESPECIFICAS | data | N | Data da última inspeção. Ex. 26-11-2003 |
| CARGA\_MAX | Numérico  (4 dígitos, 2 decimais) | N | Quantidade máxima carregada pelo trator. O valor é expresso em toneladas. Ex. 15 |
| LONGITUDE\_TRATOR | 4 Caracteres | N | Posição do trator. Ex. 40N |
| LATITUDE\_TRATOR | 4 Caracteres | N | Posição do trator. Ex. 30W |
| DIMENSAO\_TRATOR | 15 Caracteres | N | Tamanho do trator. Valor expresso em metros e centímetros, do comprimento, largura e altura. Ex. 4.15C3.50L2.20A |
| MARCA | 20 Caracteres | N | Marca do trator, em maiúsculas. Ex.: SCANIA |
| MODELO | 20 Caracteres | N | Modelo do trator, em maiúsculas. Ex.: L320 |

Atributos com restrições:

|  |  |  |
| --- | --- | --- |
| **Nome do atributo** | **Valores Únicos?** | **Observações** |
| MATRICULA | S | Identificador (chave primária). Não existem duas viagens com o mesmo código. |
| PESO | N | Valor não nulo não negativo |
| INSPECOES | N | Data inferior à data atual |
| CARGA\_MAX | N | Valor não nulo não negativo |
| LONGITUDE\_TRATOR | N | Valor não nulo |
| LATITUDE\_TRATOR | N | Valor não nulo |
| DIMENSAO\_TRATOR | N | Valor não nulo não negativo |
| MARCA | N | Valor não nulo, em maiúsculas |
| MODELO | N | Valor não nulo, em maiúsculas |

Relacionamentos da Entidade:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nome do relacionamento** | **Cardinalidade** | **Entidade Relacionada** | **Participação**  **Obrigatória** | **Página** |
| Conduz | N : N | MOTORISTAS | NENHUMA | Y |
| Binómio | M : N : P | VIAGENS, CISTERNAS | NENHUMA | B |

* + 1. Entidade Cisternas

A entidade Cisternas representa a informação relativa as cisternas. Esta entidade não representa todos as cisternas que existem, mas apenas as que estão preparadas para realizarem viagens. É inserido uma nova cisterna nesta Entidade sempre que esta estiver preparada.

Atributos relevantes da Entidade:

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome do atributo** | **Tipo de Dados** | **Aceita Nulos?** | **Descrição** |
| ID\_CISTERNA | Numérico  (4 dígitos) | N | Código interno único atributo a cada Cisterna. É um número de 4 dígitos gerado sequencialmente. Ex. 9223 |
| CAP\_MAX | Numérico  (4 dígitos) | N | Capacidade máxima da cisterna. O valor é expresso em toneladas Ex. 40 |
| DIMENSAO\_CISTERNA | 20 Caracteres | N | Tamanho do trator. Valor expresso em metros e centímetros, do comprimento e altura. Ex. 15C3.50L2.20A |
| MARCA | 20 Caracteres | N | Marca do trator. Valor expresso em maiúsculas. Ex.: LAG |
| MODELO | 20 Caracteres | N | Modelo do trator. Valor expresso em maiúsculas. Ex.: Model 2 |

Atributos com restrições:

|  |  |  |
| --- | --- | --- |
| **Nome do atributo** | **Valores Únicos?** | **Observações** |
| ID\_CISTERNA | S | Identificador (chave primária). Não existem duas cisternas com o mesmo código. |
| CAP\_MAX | N | Valor não nulo não negativo |
| DIMENSAO\_TRATOR | N | Valor não nulo não negativo |
| MARCA | N | Valor não nulo, em maiúsculas |
| MODELO | N | Valor não nulo, em maiúsculas |

Relacionamentos da Entidade:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nome do relacionamento** | **Cardinalidade** | **Entidade Relacionada** | **Participação**  **Obrigatória** | **Página** |
| Binómio | M : N : P | VIAGENS, TRATORES | NENHUMA | B |

* + 1. Entidade Previsões

A entidade Previsões representa a informação relativa as previsões de viagem. Esta entidade representa as previsões para as viagens. É inserido uma nova previsão nesta Entidade sempre que esta existir.

Atributos relevantes da Entidade:

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome do atributo** | **Tipo de Dados** | **Aceita Nulos?** | **Descrição** |
| ID\_PREV | Numérico  (4 dígitos) | N | Código interno único atributo a cada previsão. É um número de 4 dígitos gerado sequencialmente. Ex. 9223 |
| DURA\_PREV | tempo | N | Tempo previsto demorar na viagem. Ex. 2:15 |
| COMBUSTIVEL\_PREV | data | N | Previsão do combustível a ser gasto na viagem. Valor expresso em litros. Ex. 40 |

Atributos com restrições:

|  |  |  |
| --- | --- | --- |
| **Nome do atributo** | **Valores Únicos?** | **Observações** |
| ID\_PREV | S | Identificador (chave primária). Não existem duas rotas com o mesmo código. |
| DURA\_PREV | N | Valor não nulo não negativo |
| COMBUSTIVEL\_PREV | N | Valor não nulo não negativo |

Relacionamentos da Entidade:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nome do relacionamento** | **Cardinalidade** | **Entidade Relacionada** | **Participação**  **Obrigatória** | **Página** |
| Segue | N : N | VIAGENS | PREVISOES | B |

* + 1. Entidade Postos

A entidade Previsões representa a informação relativa aos postos. Esta entidade representa todos os postos que existem. É inserido um novo posto nesta Entidade sempre que este existir.

Atributos relevantes da Entidade:

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome do atributo** | **Tipo de Dados** | **Aceita Nulos?** | **Descrição** |
| ID\_POSTO | Numérico  (4 dígitos) | N | Código interno único atributo a cada previsão. É um número de 4 dígitos gerado sequencialmente. Ex. 9223 |
| QNT\_GASOLEO | Numérico  (8 dígitos, 2 decimais) | N | Quantidade de gasóleo existente no posto. Valor expresso em litros. Ex. 40 |
| QNT\_GASOLINA95 | Numérico  (8 dígitos, 2 decimais) | N | Quantidade de gasolina 98 existente no posto. Valor expresso em litros. Ex. 40 |
| QNT\_GASOLINA98 | Numérico  (8 dígitos, 2 decimais) | N | Quantidade de gasolina 95 existente no posto. Valor expresso em litros. Ex. 40 |
| ULTIMO\_ABASTECIMENTO | data | S | Data do último abastecimento. Ex. 26-11-2003 |
| LONGITUDE\_POSTO | 4 Caracteres | N | Posição do posto. Ex. 40N |
| LATITUDE\_POSTO | 4 Caracteres | N | Posição do posto. Ex. 30W |
| CONTACTO | Numérico  (9 dígitos) | N | Contacto telefónico do posto. Ex.: 231876537 |
| LOCALIZAÇÃO | 250 Caracteres | N | Localização do posto. Ex.: R. de Almeiriga 546 |

Atributos com restrições:

|  |  |  |
| --- | --- | --- |
| **Nome do atributo** | **Valores Únicos?** | **Observações** |
| ID\_POSTO | S | Identificador (chave primária). Não existem dois postos com o mesmo código. |
| QNT\_GASOLEO | N | Valor não nulo não negativo |
| QNT\_GASOLINA95 | N | Valor não nulo não negativo |
| QNT\_GASOLINA98 | N | Valor não nulo não negativo |
| ULT\_ABASTECIMENTO | N | Data inferior ou igual a data atual |
| LONGITUDE\_POSTO | N | Valor não nulo |
| LATITUDE\_POSTO | N | Valor não nulo |
| CONTACTO | S | Valor não nulo com 9 e apenas 9 dígitos |
| LOCALIZAÇÃO | S | Valor não nulo |

Relacionamentos da Entidade:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nome do relacionamento** | **Cardinalidade** | **Entidade Relacionada** | **Participação**  **Obrigatória** | **Página** |
| Entregam | 1 : N | VIAGENS | POSTOS | B |

* + 1. Entidade Fornecedores

A entidade Previsões representa a informação relativa aos fornecedores. Esta entidade representa todos os fornecedores que existem. É inserido um novo posto nesta Entidade sempre que este existir.

Atributos relevantes da Entidade:

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome do atributo** | **Tipo de Dados** | **Aceita Nulos?** | **Descrição** |
| NOME | 15 Caracteres | N | Nome único atributo a cada fornecedor. Ex. Matosinhos |
| LONGITUDE\_FORN | 4 Caracteres | N | Posição do fornecedor. Ex. 40N |
| LATITUDE\_FORN | 4 Caracteres | N | Posição do fornecedor. Ex. 30W |
| CONTACTO | Numérico  (9 dígitos) | N | Contacto telefónico do posto. Ex.: 229992760 |
| LOCALIZAÇÃO | 256  Caracteres | N | Localização da refinaria. Ex.: Rua de Almeiriga |

Atributos com restrições:

|  |  |  |
| --- | --- | --- |
| **Nome do atributo** | **Valores Únicos?** | **Observações** |
| NOME | S | Identificador (chave primária). Não existem dois fornecedores com o mesmo nome. |
| LONGITUDE\_POSTO | N | Valor não nulo |
| LATITUDE\_POSTO | N | Valor não nulo |
| CONTACTO | S | Valor não nulo com 9 e apenas 9 dígitos |

Relacionamentos da Entidade:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nome do relacionamento** | **Cardinalidade** | **Entidade Relacionada** | **Participação**  **Obrigatória** | **Página** |
| Recebem | N : N | VIAGENS | FORNECEDORES | B |
| Instrução | 1 : 1 | MOTORISTAS | FORNECEDORES | X |

* 1. Relacionamentos

Nesta secção são descritos todos os relacionamentos existentes entre as várias entidades. Após uma análise aprofundada do negócio da distribuição de combustíveis, constatou-se a necessidade dos seguintes relacionamentos:

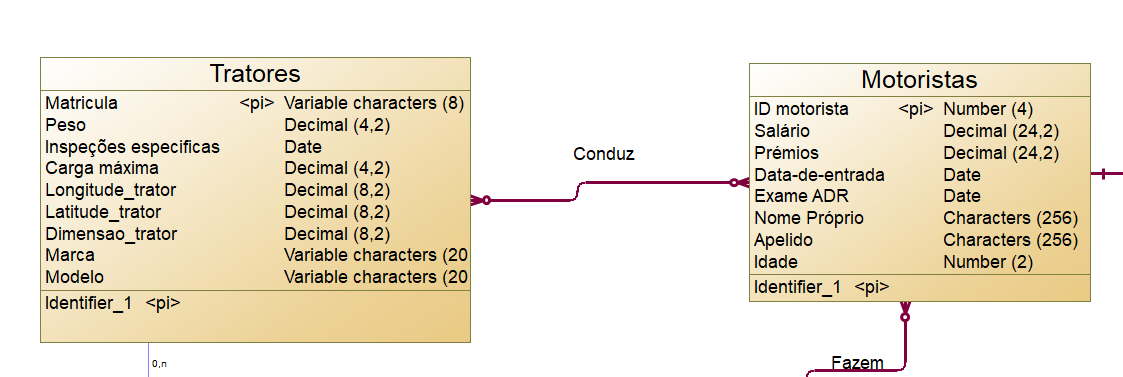
* conduz
* fazem
* binómio
* segue
* formam-se/instrução
* recebem
* entregam
  + 1. Relacionamento: conduz

Este relacionamento pretende expressar o relacionamento existente entre as Entidades TRATORES e MOTORISTAS, no que concerne aos motoristas. O objetivo é expressar quais os MOTORISTAS conduzem quais TRATORES e vice-versa.

Após uma análise do funcionamento da empresa, e as orientações definidas pelo cliente, definiu-se as seguintes condições:

* Um trator é conduzido por apenas um motorista,
* Um motorista pode conduzir vários tratores,
* Não é requerido que haja existência prévia na empresa, de um motorista, antes da sua inserção na base de dados.

Tomando estas condições em consideração, definiu-se as seguintes características



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entidade** | **Obrigatório** | **Cardinalidade** | **Obrigatório** | **Entidade** |
| **TRATORES** | **NÃO** | **N : N** | **NÃO** | **MOTORISTAS** |

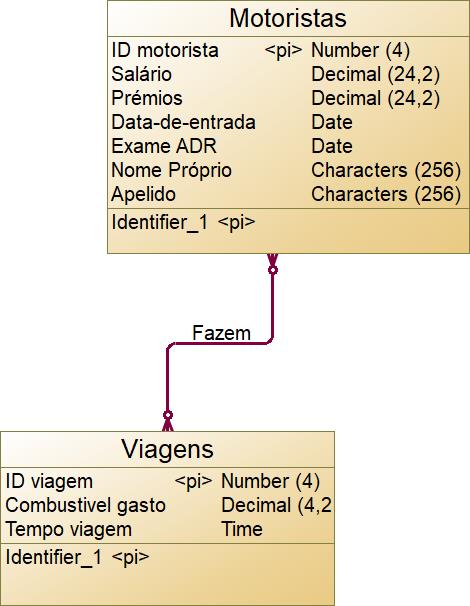
* + 1. Relacionamento: fazem

Este relacionamento pretende expressar o relacionamento existente entre as Entidades VIAGENS e MOTORISTAS, no que concerne às viagens. O objetivo é expressar quais as VIAGENS são feitas por quais MOTORISTAS e vice-versa.

Após uma análise do funcionamento da empresa, e as orientações definidas pelo cliente, definiu-se as seguintes condições:

* Uma viagem é feita por apenas um motorista,
* Um motorista pode fazer várias viagens,
* Não é requerido que haja existência prévia na empresa, de uma viagem, antes da sua inserção na base de dados.

Tomando estas condições em consideração, definiu-se as seguintes características



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entidade** | **Obrigatório** | **Cardinalidade** | **Obrigatório** | **Entidade** |
| **VIAGENS** | **NÃO** | **N : N** | **NÃO** | **MOTORISTAS** |
| **Observações** | | | | |
| * Uma viagem registada na empresa, **obrigatoriamente** foi feita por apenas **um** motorista. * Um motorista **pode** ter feito **várias** viagens que estão registadas na empresa. | | | | |

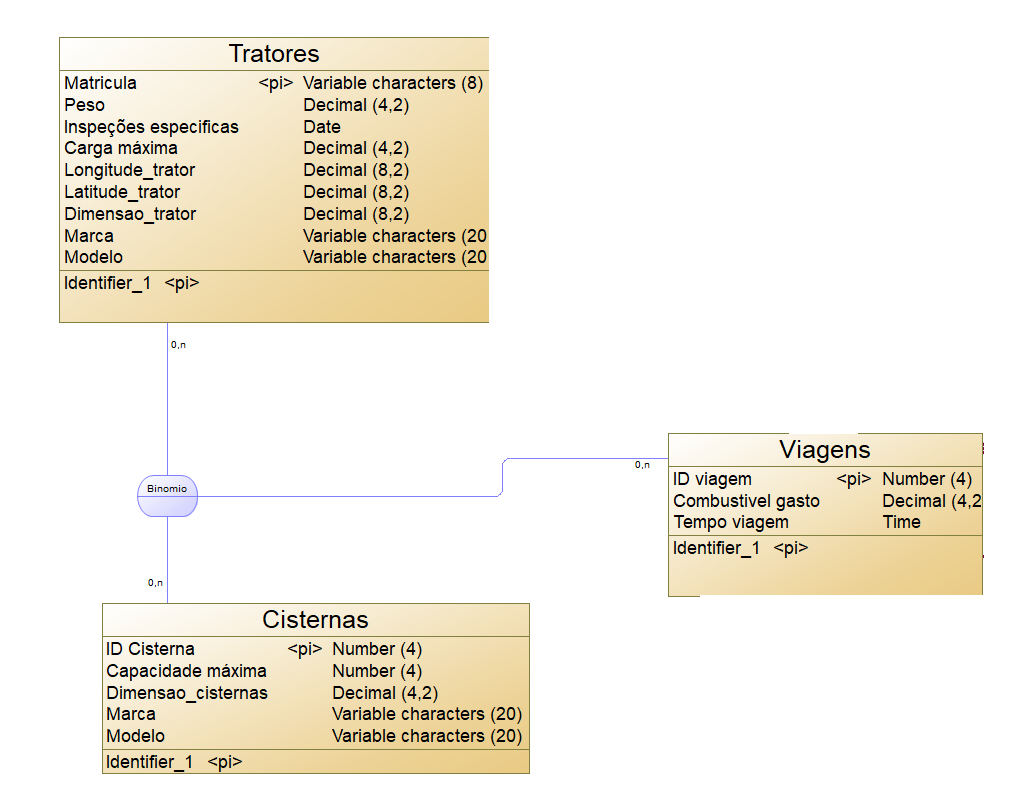
* + 1. Relacionamento: binómio

Este relacionamento pretende expressar o relacionamento existente entre as Entidades VIAGENS, TRATORES E CISTERNAS, no que concerne às viagens. O objetivo é expressar quais os VIAGENS são feitas por quais MOTORISTAS e CISTERNAS, e vice-versa.

Após uma análise do funcionamento da empresa, e as orientações definidas pelo cliente, definiu-se as seguintes condições:

* Uma viagem é feita por apenas um binómio,
* Um binómio pode fazer várias viagens,
* Não é requerido que haja existência prévia na empresa, de um trator e uma cisterna, antes da sua inserção na base de dados.

Tomando estas condições em consideração, definiu-se as seguintes características



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entidade** | **Obrigatório** | **Cardinalidade** | **Obrigatório** | **Entidade** |
| **VIAGENS** | **NÃO** | **M : N : P** | **NÃO** | **MOTORISTAS** |
| **NÃO** | **CISTERNAS** |
| **Observações** | | | | |
| * Uma viagem registada na empresa, **obrigatoriamente** foi levado por apenas **um** binómio * Um binómio **pode** ter sido levado em **várias** viagens que estão registados na empresa | | | | |

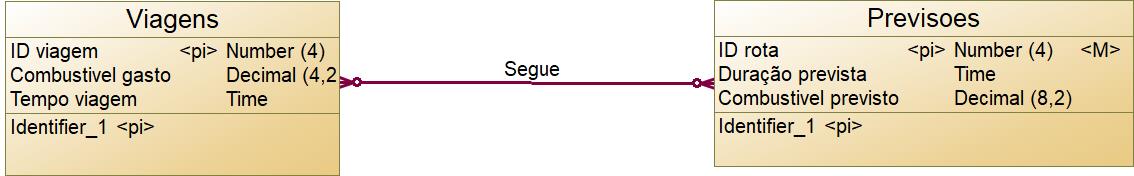
* + 1. Relacionamento: segue

Este relacionamento pretende expressar o relacionamento existente entre as Entidades VIAGENS e PREVISOES, no que concerne às viagens. O objetivo é expressar as PREVISPES das VIAGENS a serem realizadas.

Após uma análise do funcionamento da empresa, e as orientações definidas pelo cliente, definiu-se as seguintes condições:

* Uma previsão é feita para apenas uma viagem,
* Não é requerido que haja existência prévia na empresa, de uma previsão, antes da sua inserção na base de dados.

Tomando estas condições em consideração, definiu-se as seguintes características



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entidade** | **Obrigatório** | **Cardinalidade** | **Obrigatório** | **Entidade** |
| **VIAGENS** | **NÃO** | **N : N** | **NÃO** | **PREVISOES** |
| **Observações** | | | | |
| * Uma viagem **pode** ser seguida por **várias** previsões que estão registadas na empresa. | | | | |

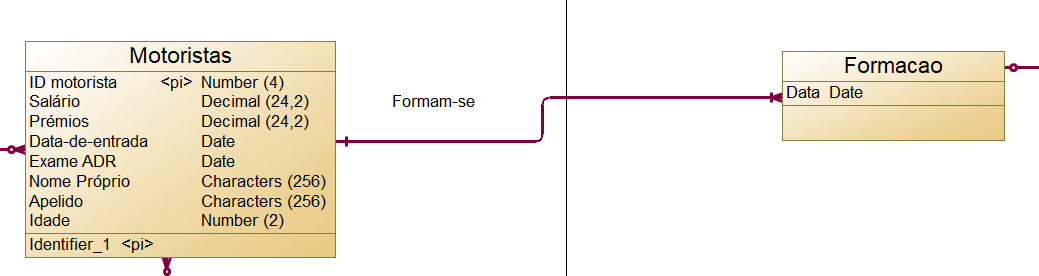
* + 1. Relacionamento: formam-se/instrução

Este relacionamento pretende expressar o relacionamento existente entre as Entidades MOTORISTAS e FORNECEDORES, no que concerne aos fornecedores. O objetivo é expressar quais as formações são instruídas por quais FORNECEDORES para quais MOTORISTAS e vice-versa.

Após uma análise do funcionamento da empresa, e as orientações definidas pelo cliente, definiu-se as seguintes condições:

* Um fornecedor instrui apenas uma formação,
* Um motorista pode fazer várias formações,
* Não é requerido que haja existência prévia na empresa, de uma formação, antes da sua inserção na base de dados.

Tomando estas condições em consideração, definiu-se as seguintes características



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entidade** | **Obrigatório** | **Cardinalidade** | **Obrigatório** | **Entidade** |
| **MOTORISTAS** | **SIM** | **1 : N** | **SIM** | **FORMAÇÃO** |
| **FORMAÇÃO** | **SIM** | **1 : 1** | **SIM** | **FORNECEDORES** |
| **Observações** | | | | |
| * Uma formação registada na empresa, **obrigatoriamente** foi instruída por apenas **um** fornecedor. * Um motorista **pode** ter **várias** formações que estão registadas na empresa. | | | | |

Atributos relevantes do Relacionamento (*eliminar se não existirem*):

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome do atributo** | **Tipo de Dados** | **Aceita Nulos?** | **Descrição** |
| DATA | Data | N | Data em que o motorista recebeu formação. Ex. 26-11-2013 |

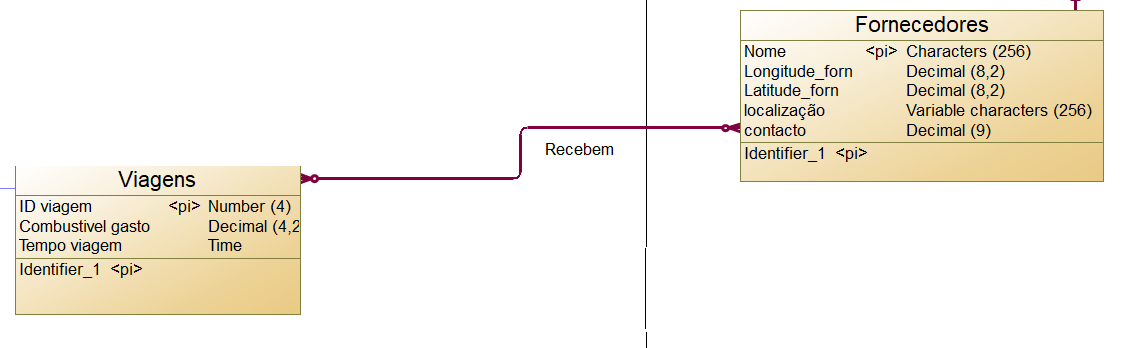
* + 1. Relacionamento: recebem

Este relacionamento pretende expressar o relacionamento existente entre as Entidades FORNECEDORES e VIAGENS, no que concerne às viagens. O objetivo é expressar quais VIAGENS recebem de quais FORNECEDORES e vice-versa.

Após uma análise do funcionamento da empresa, e as orientações definidas pelo cliente, definiu-se as seguintes condições:

* Uma viagem recebe de vários fornecedores,
* Um fornecedor só pode ser fornece 1 viagem,
* Não é requerido que haja existência prévia na empresa, de um fornecedor, antes da sua inserção na base de dados.

Tomando estas condições em consideração, definiu-se as seguintes características



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entidade** | **Obrigatório** | **Cardinalidade** | **Obrigatório** | **Entidade** |
| **VIAGENS** | **NÃO** | **N : N** | **NÃO** | **FORNECEDORES** |
| **Observações** | | | | |
| * Uma viagem registada na empresa, **obrigatoriamente** recebe de apenas **um** fornecedor. * Um motorista **pode** ter **várias** formações que estão registadas na empresa. | | | | |

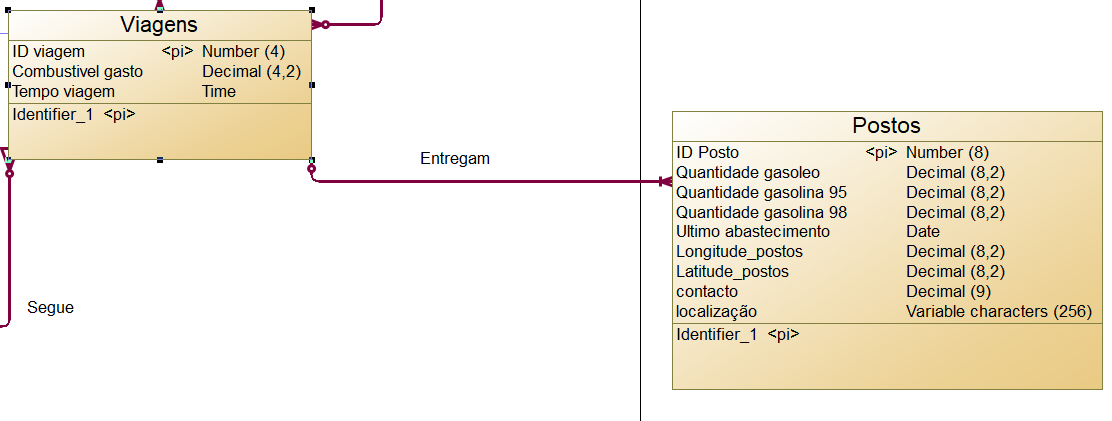
* + 1. Relacionamento: entregam

Este relacionamento pretende expressar o relacionamento existente entre as Entidades POSTOS e VIAGENS, no que concerne às viagens. O objetivo é expressar quais VIAGENS entregam a quais POSTOS e vice-versa.

Após uma análise do funcionamento da empresa, e as orientações definidas pelo cliente, definiu-se as seguintes condições:

* Uma viagem entrega a vários postos,
* Um posto só pode receber de 1 viagem,
* Não é requerido que haja existência prévia na empresa, de uma viagem, antes da sua inserção na base de dados.

Tomando estas condições em consideração, definiu-se as seguintes características



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Entidade** | **Obrigatório** | **Cardinalidade** | **Obrigatório** | **Entidade** |
| **VIAGENS** | **NÃO** | **N : N** | **NÃO** | **FORNECEDORES** |
| **Observações** | | | | |
| * Um posto registado na empresa, **obrigatoriamente** recebe de apenas **uma** viagem. * Um posto **pode** receber **várias** viagens que estão registadas na empresa. | | | | |

# Modelo Físico

Neste capítulo documenta-se detalhadamente todas as tabelas geradas para o Sistema de Gestão de Bases de Dados Relacional (SGBDR) Oracle, a partir do modelo Relacional definido no capítulo 4. Na subsecção 5.1, é descrita cada uma das tabelas (respetivos atributos e restrições de integridade), apresentando-se na secção 5.2 o modelo Físico (ou modelo de tabelas) completo, composto por todas as tabelas e restrições de integridade referencial. A secção 5.3 apresenta o código SQL necessário para criar as tabelas e as restrições definidas.

* 1. Tabelas

Nesta secção são descritas as tabelas necessárias para implementar o negócio de venda de livros da livraria. Estas tabelas foram extraídas, tomando em consideração o modelo Entidade / Relacionamento definido no capítulo 4. As tabelas são as seguintes:

* Motoristas
* Viagens
* Tratores
* Cisternas
* Previsões
* Postos
* Fornecedores
* Binomio
* Conduz
* Formação
* Recebem
* Segue
  + 1. Tabela Motoristas

A tabela Motoristas vai conter todas as informações necessárias para a gestão do pessoal dessa função. É inserida um novo registo sempre que é contratado um novo/a motorista.

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Nome restrição** | **Tipo restrição** | **Observações** |
| ID\_MOTORISTA | ID\_MOTORISTA\_PK | Chave primária | PRIMARY KEY |
| NOME\_PROPRIO | NOME\_PROPRIO\_UP | Verificação | CHECK(NOME\_PROPRIO = UPPER(NOME\_PROPRIO)) |
| APELIDO | APELIDO\_UP | Verificação | CHECK(APELIDO = UPPER(APELIDO)) |
| PREMIOS | PREMIOS\_S0 | Verificação | CHECK(PREMIOS >= 0) |
| PREMIOS | PREMIO\_IM | Verificação | CHECK(PREMIOS <= 3000) |
| SALARIO | PREMIO\_SM | Verificação | CHECK(SALARIO >= 580) |
| EXAME\_ADR | EXAME\_ADR\_IS | Verificação | CHECK(EXAME\_ADR <= SYSDATE) |
| DATA\_ENTRADA | DATA\_ENTRADA\_IS | Verificação | CHECK(DATA\_ENTRADA <= SYSDATE) |
| IDADE | IDADE\_MIN\_MAX | Verificação | CHECK(IDADE BETWEEN 18 AND 65) |

* + 1. Tabela Viagens

A tabela Viagens vai conter informações relativas às viagens programadas que serão realizadas pelos motoristas. É adicionado um novo registo cada vez que é necessário criar uma nova viagem.

|  |  |  |  |
| --- | --- | --- | --- |
| Atributo | **Nome restrição** | **Tipo restrição** | **Observações** |
| ID\_VIAGEM | ID\_VIAGEM\_PK | Chave primária | PRIMARY KEY |
| COMBUSTIVEL\_GASTO | COMBUSTIVEL\_GASTO\_NN | Verificação | CHECK(COMBUSTIVEL\_GASTO >= 0) |
| TEMPO\_VIAGEM | TEMPO\_VIAGEM | Verificação | CHECK(TEMPO\_VIAGEM <= SYSDATE) |

* + 1. Tabela Tratores

A tabela Tratores vai conter as informações que caracterizam cada trator da frota. É adicionado um novo registo cada vez que é adquirido um novo trator para a frota.

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Nome restrição** | **Tipo restrição** | **Observações** |
| MATRICULA | MATRICULA\_PK | Chave primária | PRIMARY KEY |
| PESO | PESO\_S0 | Verificação | CHECK(PESO >= 0) |
| INSPECOES\_ESPECIFICAS | INSPECOES\_IS | Verificação | CHECK(INSPECOES\_ESPECIFICAS <= SYSDATE) |
| CARGA\_MAX | CARGA\_MAX\_NN | Verificação | CHECK(CARGA\_MAX >= 0) |
| DIMENSAO\_TRATOR | DIMENSAO\_TRATOR | Verificação | CHECK(DIMENSAO\_TRATOR > 0) |
| MARCA | MARCA\_T\_UP | Verificação | CHECK(MARCA = UPPER(MARCA)) |
| MODELO | MODELO\_T\_UP | Verificação | CHECK(MODELO = UPPER(MODELO)) |

* + 1. Tabela Cisternas

A tabela Cisternas vai conter informações que caracterizam cada cisterna da frota. É adicionado um novo registo cada vez que é adquirida uma nova cisterna.

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Nome restrição** | **Tipo restrição** | **Observações** |
| ID\_CISTERNA | ID\_CISTERNA\_PK | Verificação | PRIMARY KEY |
| CAP\_MAX | CAP\_MAX\_NN | Verificação | CHECK(CAP\_MAX >= 0) |
| DIMENSAO\_CISTERNA | DIMENSAO\_TRATOR\_NN | Verificação | CHECK(DIMENSAO\_TRATOR > 0) |
| MARCA | MARCA\_C\_UP | Verificação | CHECK(MARCA = UPPER(MARCA)) |
| MODELO | MODELO\_C\_UP | Verificação | CHECK(MODELO = UPPER(MODELO)) |

* + 1. Tabela Previsões

A tabela previsões vai guardar os valores esperados de tempo e combustível para as viagens. Será utilizado para comparar com os resultados que realmente aconteceram para efeitos de avaliação de *performance*. É adicionado um novo registo de cada vez que é inserida uma nova viagem.

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Nome restrição** | **Tipo restrição** | **Observações** |
| ID\_PREV | ID\_PREV\_PK | Verificação | PRIMARY KEY |
| DURA\_PREV | DURA\_PREV\_NN | Verificação | CHECK(DURA\_PREV >= 0) |
| COMBUSTIVEL\_PREV | COMBUSTIVEL\_PREV\_NN | Verificação | CHECK(COMBUSTIVEL\_PREV >= 0) |

* + 1. Tabela Postos

A tabela postos vai guardar todas as informações dos postos da empresa. É adicionado um novo registo de cada vez que é comprado um novo posto.

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Nome restrição** | **Tipo restrição** | **Observações** |
| ID\_POSTO | ID\_POSTO\_PK | Verificação | PRIMARY KEY |
| ID\_VIAGEM | ID\_VIAGEM\_P\_FK | Chave forasteira | FOREGIN KEY REFERENCES VIAGENS(ID\_VIAGEM) |
| QNT\_GASOLEO | QNT\_GASOLEO\_NN | Verificação | CHECK(QNT\_GASOLEO >= 0) |
| QNT\_GASOLINA95 | QNT\_GASOLINA95\_NN | Verificação | CHECK (QNT\_GASOLINA95 >= 0) |
| QNT\_GASOLINA98 | QNT\_GASOLINA98\_NN | Verificação | CHECK(QNT\_GASOLINA98 >= 0) |
| ULT\_ABASTECIMENTO | ULT\_ABASTECIMENTO\_IS | Verificação | CHECK(ULT\_ABASTECIMENTO >= SYSDATE) |
| CONTACTO | CONTACTO\_A9\_P | Verificação | CHECK(LENGTH(CONTACTO) = 9) |

* + 1. Tabela Fornecedores

A tabela fornecedores guarda as informações dos fornecedores portugueses. É adicionado um novo registo se, em Portugal, abrir uma nova refinaria.

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Nome restrição** | **Tipo restrição** | **Observações** |
| NOME | NOME\_PK | Verificação | PRIMARY KEY |
| CONTACTO | CONTACTO\_A9\_F | Verificação | CHECK(LENGTH(CONTACTO) = 9) |

* + 1. Tabela Binomio

A tabela binómio (um relacionamento que também guarda informações) guarda as *primary keys* das tabelas que envolvem o relacionamento. É adicionado um novo registo sempre que for criado um novo registo nas tabelas que envolvem este relacionamento.

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Nome restrição** | **Tipo restrição** | **Observações** |
| MATRICULA | MATRICULA\_B\_FK | Chave forasteira | FOREIGN KEY REFERENCES TRATORES(MATRICULA) |
| ID\_VIAGEM | ID\_VIAGEM\_B\_FK | Chave forasteira | FOREIGN KEY REFERENCES  VIAGENS(ID\_VIAGEM) |
| ID\_CISTERNA | ID\_CISTERNA\_B\_FK | Chave forasteira | FOREIGN KEY  REFERENCES CISTERNAS  (ID\_CISTERNA) |

* + 1. Tabela Conduz

A tabela conduz (um relacionamento que também guarda informações) guarda as *primary keys* das tabelas que envolvem o relacionamento. É adicionado um novo registo sempre que for criado um novo registo nas tabelas que envolvem este relacionamento.

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Nome restrição** | **Tipo restrição** | **Observações** |
| MATRICULA | MATRICULA\_\_C\_FK | Chave forasteira | FOREIGN KEY REFERENCES TRATORES(MATRICULA) |
| ID\_MOTORISTA | ID\_MOTORISTA\_C\_FK | Chave forasteira | FOREIGN KEY REFERENCES MOTORISTAS(ID\_MOTORISTA) |

* + 1. Tabela Formação

A tabela formação (um relacionamento que também guarda informações) guarda as *primary keys* das tabelas que envolvem o relacionamento. É adicionado um novo registo sempre que for criado um novo registo nas tabelas que envolvem este relacionamento e guarda a data.

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Nome restrição** | **Tipo restrição** | **Observações** |
| ID\_MOTORISTA | ID\_MOTORISTA\_F\_FK | Chave forasteira | FOREIGN KEY REFERENCES MOTORISTAS(ID\_MOTORISTA) |
| NOME | NOME\_F\_FK | Chave forasteira | FOREIGN KEY REFERENCES FORNECEDORES(NOME) |
| DATA | DATA\_F\_IS | Verificação | CHECK(DATA <= SYSDATE) |

* + 1. Tabela Recebem

A tabela recebem (um relacionamento que também guarda informações) guarda as *primary keys* das tabelas que envolvem o relacionamento. É adicionado um novo registo sempre que for criado um novo registo nas tabelas que envolvem este relacionamento.

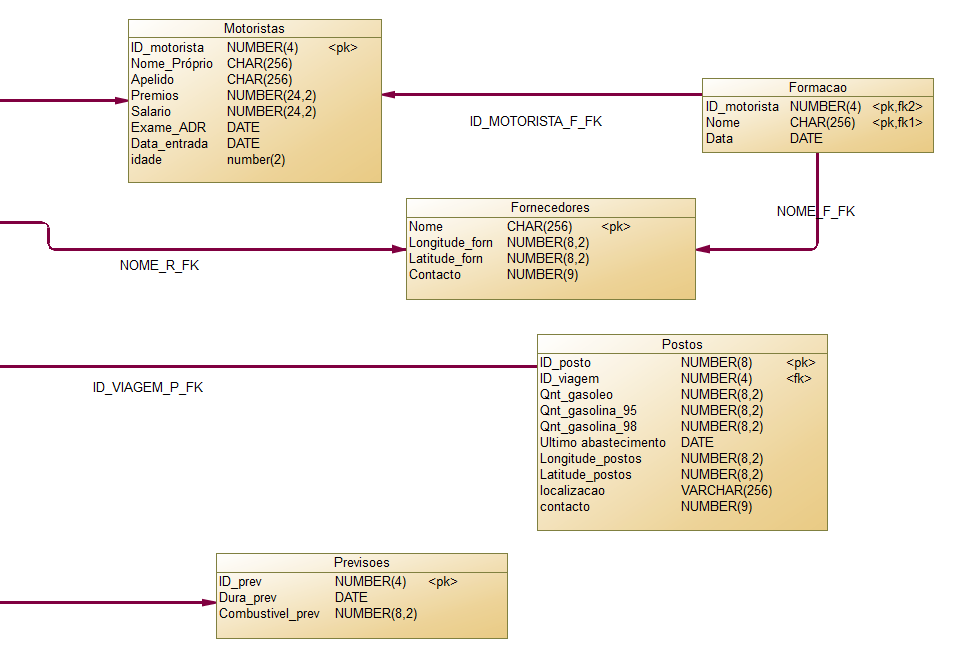
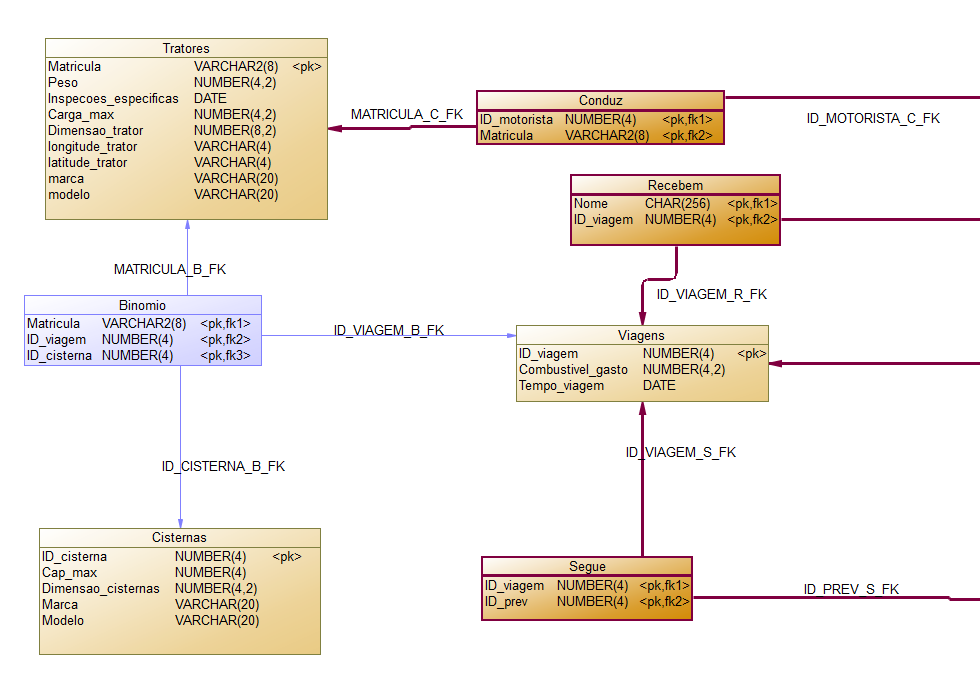
|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Nome restrição** | **Tipo restrição** | **Observações** |
| NOME | NOME\_R\_FK | Chave forasteira | FOREIGN KEY REFERENCES FORNECEDORES(NOME) |
| ID\_VIAGEM | ID\_VIAGEM\_R\_FK | Chave forasteira | FOREIGN KEY REFERENCES  VIAGENS(ID\_VIAGEM) |

* + 1. Tabela Segue

A tabela segue (um relacionamento que também guarda informações) guarda as *primary keys* das tabelas que envolvem o relacionamento. É adicionado um novo registo sempre que for criado um novo registo nas tabelas que envolvem este relacionamento.

|  |  |  |  |
| --- | --- | --- | --- |
| **Atributo** | **Nome restrição** | **Tipo restrição** | **Observações** |
| ID\_VIAGEM | ID\_VIAGEM\_S\_FK | Chave forasteira | FOREIGN KEY REFERENCES  VIAGENS(ID\_VIAGEM) |
| ID\_PREV | ID\_PREV\_S\_FK | Chave forasteira | FOREIGN KEY REFERENCES  PREVISOES(ID\_PREV) |

* 1. Diagrama do Modelo Físico



* 1. Script de criação da Base de Dados

**/\*==============================================================\*/**

**/\* Table: BINOMIO \*/**

**/\*==============================================================\*/**

**create table BINOMIO**

**(**

**MATRICULA VARCHAR2(8) not null,**

**ID\_VIAGEM NUMBER(4) not null,**

**ID\_CISTERNA NUMBER(4) not null,**

**);**

**/\*==============================================================\*/**

**/\* Table: CISTERNAS \*/**

**/\*==============================================================\*/**

**create table CISTERNAS**

**(**

**ID\_CISTERNA NUMBER(4) not null,**

**CAP\_MAX NUMBER(4) not null,**

**DIMENSAO\_CISTERNAS VARCHAR2(20) not null,**

**DATA\_DE\_USO DATE not null,**

**MARCA VARCHAR2(20) not null,**

**MODELO VARCHAR2(20) not null,**

**constraint ID\_CISTERNA\_PK primary key (ID\_CISTERNA),**

**constraint CAP\_MAX\_NN check(CAP\_MAX >= 0),**

**constraint DIMENSAO\_TRATOR\_NN check(DIMENSAO\_TRATOR >= 0),**

**constraint MARCA\_C\_UP check(MARCA = upper(MARCA)),**

**constraint MODELO\_C\_UP check(MODELO = upper(MODELO))**

**);**

**/\*==============================================================\*/**

**/\* Table: FORMACAO \*/**

**/\*==============================================================\*/**

**create table FORMACAO**

**(**

**ID\_MOTORISTA NUMBER(4) not null,**

**NOME CHAR(256) not null,**

**DATA DATE not null,**

**constraint DATA\_F\_IS check(DATA <= SYSDATE),**

**);**

**/\*==============================================================\*/**

**/\* Table: FORNECEDORES \*/**

**/\*==============================================================\*/**

**create table FORNECEDORES**

**(**

**NOME CHAR(15) not null,**

**LONGITUDE\_FORN VARCHAR2(4) not null,**

**LATITUDE\_FORN VARCHAR2(4) not null,**

**CONTACTO NUMBER(9) not null,**

**constraint NOME\_PK primary key (NOME)**

**constraint CONTACTO\_9\_F check(length(CONTACTO) = 9)**

**);**

**/\*==============================================================\*/**

**/\* Table: MOTORISTAS \*/**

**/\*==============================================================\*/**

**create table MOTORISTAS**

**(**

**ID\_MOTORISTA NUMBER(4) not null,**

**NOME\_PROPRIO CHAR(256) not null,**

**APELIDO CHAR(256) not null,**

**PREMIOS NUMBER(24,2) not null,**

**SALARIO NUMBER(24,2) not null,**

**EXAME\_ADR DATE,**

**DATA\_DE\_ENTRADA DATE not null,**

**IDADE NUMBER(4) not null,**

**constraint ID\_MOTORISTAS\_PK primary key (ID\_MOTORISTA),**

**constraint NOME\_PRORPIO\_UP check(NOME\_PROPRIO = upper(NOME\_PROPRIO)),**

**constraint APELIDO\_UP check(APELIDO = upper(APELIDO)),**

**constraint PREMIOS\_S0 check(PREMIOS >= 0),**

**constraint PREMIOS\_IM check(PREMIOS <= 3000),**

**constraint SALARIO\_SM check(SALARIO >= 580),**

**constraint EXAME\_ADR\_IS check(EXAME\_ADR <= SYSDATE)**

**constraint DATA\_ENTRADA\_IS check(DATA\_ENTRADA <= SYSDATE)**

**constraint IDADE\_MIN\_MAX check(IDADE between 18 and 65)**

**);**

**/\*==============================================================\*/**

**/\* Table: POSTOS \*/**

**/\*==============================================================\*/**

**create table POSTOS**

**(**

**ID\_POSTO NUMBER(4) not null,**

**ID\_VIAGEM NUMBER(4) not null,**

**QNT\_GASOLEO NUMBER(8,2) not null,**

**QNT\_GASOLINA\_95 NUMBER(8,2) not null,**

**QNT\_GASOLINA\_98 NUMBER(8,2) not null,**

**ULTIMO\_ABASTECIMENTO DATE,**

**LONGITUDE\_POSTO VARCHAR2(4) not null,**

**LATITUDE\_POSTO VARCHAR2(4) not null,**

**CONTACTO NUMBER(9) not null,**

**LOCALIZAO VARCHAR2(256) not null,**

**constraint ID\_POSTO\_PK primary key (ID\_POSTO),**

**constraint QNT\_GASOLEO\_NN check(QNT\_GASOLEO >= 0),**

**constraint QNT\_GASOLINA\_95\_NN check(QNT\_GASOLINA\_95 >= 0),**

**constraint QNT\_GASOLINA\_98\_NN check(QNT\_GASOLINA\_98 >= 0),**

**constraint ULT\_ABASTECIMENTO\_IS check(ULT\_ABASTECIMENTO <= SYSDATE),**

**constraint CONTACTO\_A9\_P check(length(CONTACTO) = 9),**

**constraint ID\_VIAGEM\_FK foreign key (ID\_VIAGEM) REFERENCES VIAGENS(ID\_VIAGEM)**

**);**

**/\*==============================================================\*/**

**/\* Table: PREVISOES \*/**

**/\*==============================================================\*/**

**create table PREVISOES**

**(**

**ID\_PREV NUMBER(4) not null,**

**DURACAO\_PREVI DATE not null,**

**COMBUSTIVEL\_PREV NUMBER(8,2) not null,**

**constraint ID\_PREV\_PK primary key (ID\_PREV)**

**constraint DURA\_PREV\_NN check(DURA\_PREV >= 0)**

**constraint COMBUSTIVEL\_PREV\_NN check(COMBUSTIVEL\_PREV >= 0)**

**);**

**/\*==============================================================\*/**

**/\* Table: TRATORES \*/**

**/\*==============================================================\*/**

**create table TRATORES**

**(**

**MATRICULA VARCHAR2(8) not null,**

**PESO NUMBER(4,2) not null,**

**INSPECOES DATE not null,**

**CARGA\_MAX NUMBER(4,2) not null,**

**LONGITUDE\_TRATOR VARCHAR2(4) not null,**

**LATITUDE\_TRATOR VARCHAR2(4) not null,**

**DIMENSAO\_TRATOR VARCHAR2(15) not null,**

**MARCA VARCHAR2(20) not null,**

**MODELO VARCHAR2(20) not null,**

**constraint MATRICULA\_PK primary key (MATRICULA),**

**constraint PESO\_S0 check(PESO >= 0),**

**constraint INSPECOES\_IS check(INSPECOES <= SYSDATE)**

**constraint CARGA\_MAX\_NN check(CARGA\_MAX > 0)**

**constraint DIMENSAO\_TRATOR\_NN check(DIMENSAO\_TRATOR > 0)**

**constraint MARCA\_T\_UP check(MARCA = upper(MARCA)),**

**constraint MODELO \_T\_UP check(MARCA = upper(MODELO)),**

**);**

**/\*==============================================================\*/**

**/\* Table: VIAGENS \*/**

**/\*==============================================================\*/**

**create table VIAGENS**

**(**

**ID\_VIAGEM NUMBER(4) not null,**

**COMBUSTIVEL\_GASTO NUMBER(4,2) not null,**

**TEMPO\_VIAGEM DATE not null,**

**constraint ID\_VIAGEM\_PK primary key (ID\_VIAGEM),**

**constraint COMBUSTIVEL\_GASTO\_NN check(COMBUSTIVEL\_GASTO >= 0)**

**constraint TEMPO\_VIAGEM check(TEMPO\_VIAGEM <= SYSDATE)**

**);**

**/\*==============================================================\*/**

**/\* Table: CONDUZ \*/**

**/\*==============================================================\*/**

**create table CONDUZ**

**(**

**ID\_MOTORISTA NUMBER(4) not null,**

**MATRICULA VARCHAR2(8) not null**

**);**

**/\*==============================================================\*/**

**/\* Table: RECEBEM \*/**

**/\*==============================================================\*/**

**create table RECEBEM**

**(**

**NOME CHAR(256) not null,**

**ID\_VIAGEM NUMBER(4) not null**

**);**

**/\*==============================================================\*/**

**/\* Table: SEGUE \*/**

**/\*==============================================================\*/**

**create table SEGUE**

**(**

**ID\_VIAGEM NUMBER(4) not null,**

**ID\_PREV NUMBER(4) not null**

**);**

**alter table BINOMIO**

**add constraint MATRICULA\_B\_FK foreign key (MATRICULA)**

**references TRATORES (MATRICULA);**

**add constraint ID\_VIAGEM\_B\_FK foreign key (ID\_VIAGEM)**

**references VIAGENS (ID\_VIAGEM);**

**add constraint ID\_CISTERNA\_B\_FK foreign key (ID\_CISTERNA)**

**references CISTERNAS (ID\_CISTERNA);**

**alter table FORMACAO**

**add constraint ID\_MOTORISTA\_F\_FK foreign key (ID\_MOTORISTA)**

**references MOTORISTAS (ID\_MOTORISTA);**

**add constraint NOME\_F\_FK foreign key (NOME)**

**references FORNECEDORES (NOME);**

**alter table POSTOS**

**add constraint ID\_VIAGEM\_P\_FK foreign key (ID\_VIAGEM)**

**references VIAGENS (ID\_VIAGEM);**

**alter table CONDUZ**

**add constraint ID\_MOTORISTA\_C\_FK foreign key (ID\_MOTORISTA)**

**references MOTORISTAS (ID\_MOTORISTA);**

**add constraint MATRICULA\_C\_FK foreign key (MATRICULA)**

**references TRATORES (MATRICULA);**

**alter table RECEBEM**

**add constraint NOME\_R\_FK foreign key (NOME)**

**references FORNECEDORES (NOME);**

**add constraint ID\_VIAGEM\_R\_FK foreign key (ID\_VIAGEM)**

**references VIAGENS (ID\_VIAGEM);**

**alter table SEGUE**

**add constraint ID\_VIAGEM\_S\_FK foreign key (ID\_VIAGEM)**

**references VIAGENS (ID\_VIAGEM);**

**add constraint ID\_PREV\_S\_FK foreign key (ID\_PREV)**

**references PREVISOES (ID\_PREV); \_PREV);**

# Pesquisas SQL

CREATE SEQUENCE SEQ\_VIAGENS

START WITH 1

INCREMENT BY 1

MAXVALUE 9999

NOMAXVALUE;

Todas as sequências para criação dos IDs de cada tabela seguem a estrutura do código anterior, pelo que não faria sentido repeti-lo para cada tabela, mudando só o nome da sequência.

--PESQUISAS VIAGENS:

--TODAS AS ENTRADAS DA TABELA:

SELECT \*

FROM VIAGENS;

--VERIFICAR AS VIAGENS MAIS LONGAS POR ORDEM:

SELECT ID\_VIAGEM, TEMPO\_VIAGEM

FROM VIAGENS

ORDER BY TEMPO\_VIAGEM DESC;

--VERIFICAR OS TRATORES E CISTERNAS QUE FIZERAM PARTE DE CERTA VIAGEM:

SELECT B.MATRICULA, B.ID\_VIAGEM, B.ID\_CISTERNA

FROM VIAGENS V, BINOMIO B, TRATORES T, CISTERNAS CI

WHERE CI.ID\_CISTERNA = B.ID\_CISTERNA

AND V.ID\_VIAGEM = B.ID\_VIAGEM

AND T.MATRICULA = B.MATRICULA

ORDER BY B.MATRICULA;

--VERIFICAR A QUE FORNECEDOR FORAM NUMA VIAGEM

SELECT R.NOME, R.ID\_VIAGEM

FROM VIAGENS V, RECEBEM R, FORNECEDORES F

WHERE R.NOME = F.NOME

AND R.ID\_VIAGEM = V.ID\_VIAGEM;

--VERIFICAR SE O COMBUSTÍVEL GASTO NA VIAGEM COINCIDE COM A PREVISÃO

SELECT V.COMBUSTIVEL\_GASTO "GASTO", P.COMBUSTIVEL\_PREV "PREVISTO", V.COMBUSTIVEL\_GASTO - P.COMBUSTIVEL\_PREV "DIFERENÇA"

FROM VIAGENS V, PREVISOES P, SEGUE S

WHERE V.ID\_VIAGEM = S.ID\_VIAGEM

AND P.ID\_PREV = S.ID\_PREV;

--VERIFICAR AS PARAGENS DE UMA VIAGEM

SELECT P.LOCALIZACAO

FROM POSTOS P, VIAGENS V

WHERE V.ID\_VIAGEM = P.ID\_VIAGEM;

--PESQUISAS TRATORES:

--TODAS AS ENTRADAS DA TABELA:

SELECT \*

FROM TRATORES;

--VERIFICAR SE AS INSPEÇÕES ESTÃO EM DIA:

SELECT 'HOJE É ' || TO\_CHAR(SYSDATE, 'DD-MM-YYYY') ||'. A ÚLTIMA INSPEÇÃO FOI REALIZADA NO DIA '|| TO\_CHAR(INSPECOES\_ESPECIFICAS, 'DD-MM-YYYY') || '.

A INSPEÇÃO É VÁLIDA POR 1 ANO' "VERIFICAR"

FROM TRATORES;

--QUE MOTORISTA CONDUZ CERTO BINÓMIO:

SELECT 'O MOTORISTA ' || M.NOME\_PROPRIO || M.APELIDO || ' CONDUZ O TRATOR COM A MATRÍCULA ' || T.MATRICULA || ' E A CISTERNA COM O ID ' || CI.ID\_CISTERNA || '.'

FROM MOTORISTAS M, TRATORES T, CONDUZ CO, CISTERNAS CI, BINOMIO B

WHERE CO.ID\_MOTORISTA = M.ID\_MOTORISTA

AND CO.MATRICULA = T.MATRICULA

AND CI.ID\_CISTERNA = B.ID\_CISTERNA

AND B.MATRICULA = T.MATRICULA;

SELECT FORNECEDORES.NOOME, FORMACAO.DATA, MOTORISTAS.NOME

FROM MOTORISTAS, FORNECEDORES, FORMACAO, (SELECT ID\_MOTORISTA MOTO

FROM MOTORISTA

WHERE TO\_CHAR(EXAME\_ADR,'YYYY')<TO\_CHAR(SYSDATE,'YYYY')-3) MOTODATA

WHERE FORNECEDORES.NOME = FORMACAO.NOME

AND MOTORISTAS.ID\_MOTORISTA = FORMACAO.ID\_MOTORISTA

AND MOTORISTAS.ID\_MOTORISTA = MOTODATA.MOTO;

--PESQUISA POSTOS:

--PESQUISA GERAL

SELECT \*

FROM POSTOS;

--LOCALIZAÇÃO DOS POSTOS CUJAS QUANTIDADES DE CADA COMBUSTÍVEL É INFERIOR A 1000L

SELECT LOCALIZACAO, QNT\_GASOLEO, QNT\_GASOLINA95, QNT\_GASOLINA98

FROM POSTOS

WHERE QNT\_GASOLEO < 1000

AND QNT\_GASOLINA95 < 1000

AND QNT\_GASOLINA98 < 1000;

-- QUE FORNECEDORES FORNECERAM O POSTO NO ULTIMO ABASTECIMENTO

SELECT POSTO.LOCALIZACAO, POSTO.ULT\_ABASTECIMENTO, FORNECEDOR.NOME

FROM POSTO, VIAGEM, FORNECEDOR

WHERE FORNECEDOR.NOME = VIAGEM.NOME

AND VIAGEM.ID\_POSTO = POSTO.ID\_POSTO;

--

SELECT CISTERNA.ID\_CISTERNA, CISTERNA.DATA\_USO, TRATOR.MATRICULA

FROM CISTERNAS, TRATORES, VIAGENS, BINOMIO

WHERE CISTERNA.ID\_CISTERNA = BINOMIO.ID\_CISTERNA

AND TRATOR.MATRICULA = BINOMIO.MATRICULA

AND BINOMIO.ID\_VIAGEM = VIAGEM.ID\_VIAGEM;