



UAST
Unidade Acadêmica
de Serra Talhada - PE
Desde 2006



**PROJETO DE DISCIPLINA:
FUNDAMENTOS DE BANCO DE DADOS**

WILLIANY THALITA ALMEIDA VERAS

**SERRA TALHADA, PE
[DEZEMBRO/2018]**

WILLIANY THALITA ALMEIDA VERAS

Projeto desenvolvido para efetivação da [1ª ou 2ª ou 3ª] etapa apresentado para avaliação na Disciplina Projeto de Banco de Dados Relacionais ministrada pelo Prof. Hidelberg Oliveira, período letivo 2018.2.

PROJETO DE BANCO DE DADOS

01. DEFINIÇÃO DA PROBLEMÁTICA (DOMÍNIO DA APLICAÇÃO)

Sistema para Locadora de Veículos

O objetivo do projeto é construir um Sistema para a empresa de locação de veículos “Veículos Pajeú”, que tem matriz em Serra Talhada-PE. A empresa aluga automóveis, camionetas de passageiros e camionetas de carga.

Ela atende a dois mercados: o das pessoas físicas e o das pessoas jurídicas. Para acelerar o atendimento, é importante conhecer os dados de clientes que já tenham usado a locadora no passado. Para cada pessoa física, é necessário conhecer seu nome, sexo, data de nascimento, endereço e CPF. Já para as pessoas jurídicas, é necessário conhecer seu nome, CNPJ, inscrição estadual e endereço. Os clientes são identificados por um código interno à locadora.

A empresa tem uma grande rede de filiais, espalhada pelo Nordeste Brasileiro. Em um momento no tempo, um veículo encontra-se sob responsabilidade de uma filial. Entretanto, como veículos podem ser alugados para viagens em sentidos diferentes (ida e volta, somente ida), os veículos podem mudar de filial. Um veículo é identificado por sua placa, o número do chassi, o número do motor, a categoria, a cor de cada veículo, o torque do motor, tipo de combustível (gasolina, etanol, diesel, ou bicomcombustível – nesse caso, especificar quais) e quilometragem atual.

Seu software deverá registrar:

- os veículos disponíveis em determinada filial na data corrente;
- as reservas para veículos em uma filial, com previsão de quais veículos estarão disponíveis em uma data futura;
- os veículos presentemente alugados pela filial, o ponto de entrega (caso seja diferente do local de locação) e data de entrega prevista.

Os veículos são classificados por uma tabela de categorias. Por exemplo, P3 corresponde a automóveis pequenos, de quatro portas e com ar-condicionado e G4 a grandes automóveis de luxo. As reservas não são feitas para uma marca ou modelo de veículo, mas para um categoria de veículo.

- Todos os veículos deverão possuir informações comuns, como Fabricante, Modelo, Número de passageiros, número de portas, Ano de Fabricação, Ano do Modelo.

- Para tipos de automóveis pequenos, os clientes desejam saber o tamanho (pequeno, médio e grande), bem como se possui os seguintes acessórios: ar-condicionado, rádio, DVD, MP3, direção hidráulica, câmera de ré, e tipo de câmbio (Automático ou Manual).
- Para tipos de camionetas de passageiros, as informações são as mesmas que para automóveis, além de air bag (simples-dianteira, duplo-dianteira, total), direção assistida, cintos de segurança traseiros retráteis (3 pontos), rodas de liga leve, controle de poluição do ar.
- Já para tipos de camionetas de carga, além das informações comuns, os clientes desejam saber a capacidade de carga da camioneta (em toneladas), o acionamento da embreagem (manual ou hidráulico), desempenho do veículo (relação consumo/km), potência do motor, volume de abastecimento de combustível, distância entre eixos.

Para cada tipo de veículo, há um determinado número de horas necessário para limpeza e revisão de entrega, entre uma reserva e outra. O número de horas deve ser definido pelos alunos. Seu sistema deve programar as revisões dos veículos, impedindo que sejam reservados quando há revisões pendentes. Esta programação é feita com base na quilometragem atual do veículo: a cada 5.000km, os veículos de pequeno/médio porte irão para revisão, e a cada 10.000km, os veículos de grande porte irão para as revisões.

O locatário não necessariamente precisa ser o motorista. A seguradora exige que, para cada veículo alugado seja mantida a identificação do motorista, o número de sua habilitação e data de vencimento da mesma. A habilitação não pode vencer dentro do prazo da locação, e o motorista deve ter idade mínima de 21 (vinte e um) anos. É exigido que exista um cadastro para cada motorista como cliente.

As reservas não garantem marca ou modelo do veículo, somente a categoria e, caso a categoria reservada não esteja disponível no momento da retirada, seu sistema irá providenciar um veículo de categoria superior pelo mesmo valor cobrado. As reservas são garantidas pelo prazo de até 1 (uma) hora após o horário previsto para a retirada, desde que esta hora de tolerância esteja no período de funcionamento normal da loja (horário comercial). Para as reservas, serão cobradas a metade da primeira diária antecipadamente, de acordo com a modalidade de locação escolhida.

Existem dois tipos de locação: “Km Livre” e “Km Controle”. A modalidade de “Km Livre” prevê uma cobrança maior da locação, sem cobrança adicional por quilometragem rodada. A modalidade de “Km Controle” prevê a cobrança pela quantidade de quilômetros rodados.

- Os valores das locações deverão ser distintos por categorias do veículo e da categoria da locação.

- O valores definidos para a locação deverão ser cadastrados no banco de dados.

O período mínimo para a locação do veículo é de 1 (uma) diária de 24 (vinte e quatro) horas, com até 1 (uma) hora de tolerância para devolução. Se a devolução do veículo ocorrer em até 4 horas após a hora prevista para o término da diária, serão cobradas horas extras inteiras, à razão de 1/4 da diária contratada, sendo cobrada, inclusive, a hora de tolerância. Após 4 (quatro) horas extras, será cobrada uma nova diária do veículo. A locação pode ser feita diretamente ou aproveitada de uma reserva previamente efetuada.

No momento da entrega do veículo, o cliente deverá entregar o veículo lavado e com tanque cheio. Caso contrário, pagará no ato da devolução do veículo a “taxa de higienização”, na razão de 2% do valor a pagar da locação, e/ou a “taxa de combustível” na razão de 3% do valor a pagar da locação. No momento da finalização da locação, a quilometragem atual do veículo deve ser atualizada.

02. DEFINIÇÃO DE REQUISITOS

2.1. Requisitos Funcionais:

Identificação:	[RF01] Incluir sede/filial
Caso de Uso Relacionado:	<i>Manter Sede da empresa ou filial</i>
Descrição:	Possibilita que o superusuário efetue o cadastro da filial/sede no sistema.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF02] Alterar sede /filial
Caso de Uso Relacionado:	<i>Manter Sede da empresa ou filial</i>
Descrição:	Possibilita que o superusuário edite os dados da filial/sede no sistema.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF03] Consultar sede /filial
Caso de Uso Relacionado:	<i>Manter Sede da empresa ou filial</i>
Descrição:	Possibilita que o superusuário realize buscas de sedes ou filiais cadastradas no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF04] Excluir sede/filial
Caso de Uso Relacionado:	<i>Manter Sede da empresa ou filial</i>
Descrição:	Possibilita que o superusuário, exclua a sede (se a mesma nunca funcionou) ou mude o status das sede/filial para inativo.

Prioridade:	(<input type="checkbox"/>) Essencial (<input checked="" type="checkbox"/>) Importante (<input type="checkbox"/>) Desejável
--------------------	--

Identificação:	[RF05] Incluir cliente
Caso de Uso Relacionado:	<i>Manter Clientes</i>
Descrição:	Possibilita que os usuários cadastrem os clientes no sistema.
Prioridade:	(<input checked="" type="checkbox"/>) Essencial (<input type="checkbox"/>) Importante (<input type="checkbox"/>) Desejável

Identificação:	[RF06] Alterar cliente
Caso de Uso Relacionado:	<i>Manter Clientes</i>
Descrição:	Possibilita que os usuários edite os dados dos clientes no sistema.
Prioridade:	(<input checked="" type="checkbox"/>) Essencial (<input type="checkbox"/>) Importante (<input type="checkbox"/>) Desejável

Identificação:	[RF07] Consultar cliente
Caso de Uso Relacionado:	<i>Manter Clientes</i>
Descrição:	Possibilita que os usuários realizem buscas de clientes cadastrados no sistema.
Prioridade:	(<input type="checkbox"/>) Essencial (<input checked="" type="checkbox"/>) Importante (<input type="checkbox"/>) Desejável

Identificação:	[RF08] Excluir cliente
Caso de Uso Relacionado:	<i>Manter Clientes</i>
Descrição:	Possibilita que o sistema mude o status dos clientes para inativo, ou o exclua caso o mesmo nunca alocou ou fez reserva na empresa.
Prioridade:	(<input type="checkbox"/>) Essencial (<input type="checkbox"/>) Importante (<input checked="" type="checkbox"/>) Desejável

Identificação:	[RF09] Incluir motorista
Caso de Uso Relacionado:	<i>Manter Motorista</i>
Descrição:	Possibilita que os usuários cadastrem os motorista dos clientes (podendo ser o próprio cliente) no sistema.
Prioridade:	(<input checked="" type="checkbox"/>) Essencial (<input type="checkbox"/>) Importante (<input type="checkbox"/>) Desejável

Identificação:	[RF10] Alterar motorista
Caso de Uso Relacionado:	<i>Manter Motorista</i>
Descrição:	Possibilita que os usuários edite os dados dos motoristas no sistema.
Prioridade:	(<input type="checkbox"/>) Essencial (<input checked="" type="checkbox"/>) Importante (<input type="checkbox"/>) Desejável

Identificação:	[RF11] Consultar motorista
Caso de Uso Relacionado:	<i>Manter Motorista</i>
Descrição:	Possibilita que os usuários realizem buscas de motoristas cadastrados no sistema.
Prioridade:	(<input type="checkbox"/>) Essencial (<input checked="" type="checkbox"/>) Importante (<input type="checkbox"/>) Desejável

Identificação:	[RF12] Excluir motorista
Caso de Uso Relacionado:	<i>Manter Motorista</i>
Descrição:	Possibilita que o sistema mude o status dos motoristas para inativo, ou o exclua caso o mesmo nunca foi utilizado.
Prioridade:	() Essencial () Importante (X) Desejável

Identificação:	[RF13] Incluir funcionário
Caso de Uso Relacionado:	<i>Manter Usuarios</i>
Descrição:	Possibilita que o superusuário ou administradores cadastrem os funcionários, no sistema. Esses funcionários podem ser superusuário (tem acesso total e irrestrito ao SW e ao BD), administradores (tem acesso total ao sistema) ou atendentes (usuários que efetuam as locações).
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF14] Alterar funcionário
Caso de Uso Relacionado:	<i>Manter Usuarios</i>
Descrição:	Possibilita que os superusuário ou administradores edite os seus dados no sistema.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF15] Alterar senha funcionário
Caso de Uso Relacionado:	<i>Manter Usuarios</i>
Descrição:	Qualquer usuário poderá, ele mesmo, alterar sua senha de acesso ao Sistema. Este procedimento poderá ser feito sem restrições.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF16] Consultar funcionário
Caso de Uso Relacionado:	<i>Manter Usuarios</i>
Descrição:	Possibilita que os usuários realizem buscas de funcionários cadastrados no sistema. Entretanto, só podem ter acesso às tabelas de seu grupo de usuários.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF17] Excluir funcionário
Caso de Uso Relacionado:	<i>Manter Usuarios</i>
Descrição:	Possibilita que os superusuários ou administradores mudem o status dos funcionários para inativo no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF18] Validar senha funcionário
Caso de Uso Relacionado:	<i>Validar senha</i>

Descrição:	No cadastro do funcionário o campo senha dos usuários deverá aceitar caracteres alfanuméricos, com tamanho entre 6 e 11, distinguindo caracteres maiúsculos de minúsculos.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF19] Reset do funcionário
Caso de Uso Relacionado:	<i>Reset senha Usuario</i>
Descrição:	Caso o usuário perca os dados, somente o superusuário fará o reset do acesso do usuário no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF20] Incluir veículo
Caso de Uso Relacionado:	<i>Manter Veiculos</i>
Descrição:	Possibilita que o superusuário ou administradores cadastrem os veículos no sistema.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF21] Alterar veículo
Caso de Uso Relacionado:	<i>Manter Veiculos</i>
Descrição:	Possibilita que os superusuário ou administradores edite os dados dos veículos no sistema.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF22] Consultar veículo
Caso de Uso Relacionado:	<i>Manter Veiculos.</i>
Descrição:	Possibilita que os usuários realizem buscas de veículos cadastrados no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF23] Excluir veículo
Caso de Uso Relacionado:	<i>Manter Veiculos</i>
Descrição:	Possibilita que os superusuários ou administradores mudem o status dos veículos para inativo no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF24] Incluir categoria dos veículos
Caso de Uso Relacionado:	<i>Manter Veiculos</i>
Descrição:	Possibilita que o superusuário ou administradores cadastrem as categoria dos veículos no sistema.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF25] Alterar categoria dos veículos
Caso de Uso Relacionado:	<i>Manter Veiculos</i>
Descrição:	Possibilita que os superusuário ou administradores edite os dados das categoria dos veículos no sistema.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF26] Consultar categoria dos veículos
Caso de Uso Relacionado:	<i>Manter Veiculos</i>
Descrição:	Possibilita que os usuários realizem buscas das categoria dos veículos cadastrados no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF27] Excluir categoria dos veículos
Caso de Uso Relacionado:	<i>Manter Veiculos</i>
Descrição:	Possibilita que os superusuários ou administradores mudem o status das categoria dos veículos para inativo no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF28] Incluir reserva
Caso de Uso Relacionado:	<i>Manter Reserva</i>
Descrição:	Possibilita que os usuários façam reservas das categoria dos veículos no sistema.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF29] Alterar reserva
Caso de Uso Relacionado:	<i>Manter Reserva</i>
Descrição:	Possibilita que os usuários edite os dados das reservas no sistema.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF30] Consultar reserva
Caso de Uso Relacionado:	<i>Manter Reserva</i>
Descrição:	Possibilita que os usuários realizem buscas das reservas no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF31] Excluir reserva
Caso de Uso Relacionado:	<i>Manter Reserva</i>
Descrição:	Possibilita que os usuários excluam as reservas no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF32] Efetivar de reserva
-----------------------	----------------------------

Caso de Uso Relacionado:	<i>Se o aluno utilizar Casos de Uso (extra), preencher. Caso contrário, deverá eliminar esta linha tá tabela.</i>
Descrição:	O sistema deve gerar um comprovante da reservar com os dados do cliente, a categoria reservada e data da locação.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF33] Incluir locação
Caso de Uso Relacionado:	<i>Manter Locação</i>
Descrição:	Possibilita que os usuários cadastrem locação de veículos no sistema. Essas locações pode ser de dois tipos: “Km Livre” e “Km Controle” .
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF34] Alterar locação
Caso de Uso Relacionado:	<i>Manter Locação</i>
Descrição:	Possibilita que os usuários edite os dados das locação dos veículos no sistema.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF35] Consultar locação
Caso de Uso Relacionado:	<i>Manter Locação</i>
Descrição:	Possibilita que os usuários realizem buscas das locação dos veículos cadastrados no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF36] Excluir locação
Caso de Uso Relacionado:	<i>Manter Locação</i>
Descrição:	Possibilita que os usuários mudem o status das locação para cancelada no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF37] Finalizar locação
Caso de Uso Relacionado:	<i>Manter Locação</i>
Descrição:	Possibilita que os usuários finalizem a locação do veículo, assim atualizando a quilometragem atual do veículo.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF38] Gerar Relatórios
Caso de Uso Relacionado:	<i>Gerar Relatórios</i>
Descrição:	Possibilita que os usuários autorizados visualizem relatórios.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF39] Imprimir relatórios
Caso de Uso Relacionado:	<i>Imprimir Relatorios</i>
Descrição:	Possibilita que seja feita a impressão dos relatórios gerados.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF40] Salvar relatórios
Caso de Uso Relacionado:	<i>Salvar Relatorios</i>
Descrição:	Possibilita que os relatórios sejam salvos em PDF.
Prioridade:	() Essencial () Importante (X) Desejável

Identificação:	[RF41] Realizar acesso
Caso de Uso Relacionado:	<i>Fazer Login</i>
Descrição:	O acesso ao Sistema só pode ser feito por usuários autorizados.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RF42] Incluir financeiro
Caso de Uso Relacionado:	<i>Manter financeiro</i>
Descrição:	Possibilita aos superusuários ou administradores incluam conatas a pagar ou a receber no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF43] Alterar financeiro
Caso de Uso Relacionado:	<i>Manter financeiro</i>
Descrição:	Possibilita aos superusuários ou administradores altere dados de uma conta.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF35] Consultar financeiro
Caso de Uso Relacionado:	<i>Manter financeiro</i>
Descrição:	Possibilita aos superusuários ou administradores realizem buscas das contas no sistema.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF36] Excluir financeiro
Caso de Uso Relacionado:	<i>Manter financeiro</i>
Descrição:	Possibilita aos superusuários ou administradores excluam uma conta
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RF36] Gerar log
Caso de Uso Relacionado:	<i>Gerar Log's</i>

Descrição:	Possibilita aos superusuários ou administradores gerem Log's dos usuários.
Prioridade:	() Essencial (X) Importante () Desejável

2.2. Requisitos Não-Funcionais:

Identificação:	[RNF01] Programação
Tipo:	Compatibilidade
RF Relacionado:	Todos
Descrição:	A Linguagem de Programação será obrigatoriamente Java (com JPA) e o SGBD utilizado será o PostgreSQL
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RNF02] Gerenciar dados
Tipo:	Segurança
RF Relacionado:	Todos
Descrição:	Os dados não podem ser corrompidos, por isso devem ser mantidos e gerenciados por um Sistema de Gerenciamento de Banco de Dados (PostgreSQL).
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RNF03] Rápido
Tipo:	Usabilidade
RF Relacionado:	Todos
Descrição:	O sistema deve ser rápido, e fazer o processamento necessário no menor tempo possível
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RNF04] Mensagens de erro
Tipo:	Padrões
RF Relacionado:	Todos
Descrição:	As mensagens de erro do sistema deverão ser precisas e construtivas, fazendo com que o usuário identifique sua origem e como proceder após sua ocorrência.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RNF05] Interface amigável
Tipo:	Usabilidade
RF Relacionado:	Todos
Descrição:	A interface do sistema deverá ser amigável e objetiva, suas funcionalidades e dados deverão estar bem intuitivas.

Prioridade:	() Essencial (X) Importante () Desejável
--------------------	--

Identificação:	[RNF06] Disponibilidade
Tipo:	Disponibilidade
RF Relacionado:	Todos
Descrição:	O sistema deve estar sempre disponível para acesso.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RNF07] Realizar backup
Tipo:	Segurança
RF Relacionado:	Todos
Descrição:	O sistema deve ser configurado para fazer o backup do banco de dados uma vez por dia, forçando o usuário a fazer o backup que sempre que alcançar uma determinada hora do dia.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RNF07] Realizar restauração de dados
Tipo:	Segurança
RF Relacionado:	Todos
Descrição:	O sistema deve ser configurado para recuperar os dados do banco de dados.
Prioridade:	(X) Essencial () Importante () Desejável

2.3. Requisitos de Domínio:

Identificação:	[RD01] Validar motorista
Descrição:	O motorista deve ter idade mínima de 21, e a habilitação não pode vencer entre o período da locação.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RD02] Controlar revisão do veículo
Descrição:	O sistema deve programar as revisões dos veículos, impedindo que sejam reservados quando há revisões pendentes.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RD03] Controlar limpeza do veículo
Descrição:	Possibilitar aos usuários determinar número de horas necessário para limpeza de cada veículo, entre uma reserva e outra.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RD04] Validade da reserva
-----------------------	----------------------------

Descrição:	O sistema gerar o prazo de validade da reserva.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RD05] Concorrência de reserva
Descrição:	O sistema deve verificar quem tem se tem veículos disponíveis para fazer a reservar na categoria desejada pelo o cliente. E quem tem preferencia em alocar o veiculo primeiro.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RD06] Gerar pagamento inicial da locação
Descrição:	Possibilita que sistema gere um valor a ser pago, este valor é a metade da primeira diária, de acordo com a modalidade de locação escolhida
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RD07] Gerar pagamento final locação
Descrição:	Possibilita que sistema gere um valor total a ser pago pelo cliente. Incluindo as taxas de higienização e taxa de combustível, se no momento da devolução o veiculo não estiver limpo e com o tanque cheio. Ou se a devolução do veículo ocorrer em até 4 horas após a hora prevista para o término da diária, serão cobradas horas extras inteiras, à razão de 1/4 da diária contratada, sendo cobrada, inclusive, a hora de tolerância. Após 4 (quatro) horas extras, será cobrada uma nova diária do veículo.
Prioridade:	(X) Essencial () Importante () Desejável

Identificação:	[RD08] Restrições de locação
Descrição:	O sistema deve verificar se o cliente não possui restrições à locação do automóvel.
Prioridade:	() Essencial () Importante (X) Desejável

Identificação:	[RD09] Realizar auditoria
Descrição:	O sistema deve mostrar o log de acesso do sistema para os administradores, mostrando as modificações nas tabelas selecionadas por períodos.
Prioridade:	() Essencial (X) Importante () Desejável

Identificação:	[RD10] Reconhecer inconsistência de dados
Descrição:	Os dados inseridos no sistema deverão ser validados a fim de evitar que dados errôneos sejam armazenados, prejudicando a corretude e consistência da base de dados.
Prioridade:	() Essencial (X) Importante () Desejável

03. DIAGRAMA ENTIDADE-RELACIONAMENTO

- ✓ Locacao(id, tipo_locacao, data_hora_retirada, data_hora_devolucao, km_inicial, km_final, status, taxa_higienização, taxa_combustivel, *fk_reserva*, *fk_usuario*, *fk_cliente*, *fk_motorista*, *fk_veiculo*)
- ✓ Usuario(id, nome, status, CPF, RG, CTPS, tipo_usuario, login, senha, DN, *fk_endereco*, *fk_sede_filial*)
- ✓ Sede/Filial(id, nome, tipo, CNPJ, status, *fk_endereco*)
- ✓ Motorista (id, nome, status, CNH, data_vencimento_CHN, DN)
- ✓ Cliente (id, nome, status, *fk_endereco*)
- ✓ PessoaFisica(CPF, sexo, DN, *fk_cliente*)
- ✓ PessoaJuridica(CNPJ, insc_estadual, *fk_cliente*)
- ✓ Categoria(id, código, nome, descricao, status, tamanho, ar_condicionado, radio, DVD, MP3, direcao_hidraulica, camera_re, tipo_cambio, valor_locar_categoria)
- ✓ Automoveis(*fk_categoria*)
- ✓ CamionetasCarga(*fk_categoria*, capacidade_carga, tipo_embreagem, desempenho, potencia_motor, volume_abastecimento, distancia_entre_eixos)
- ✓ CamionetasPassageiros(*fk_categoria*, tipo_air_bag, direcao_assistida, cintos_traseiros_retrateis, rodas_liga_leve, cont_poluicao_ar)
- ✓ ControleFinanceiro(id, descrição, valor, data, tipo, formas_pagamento, *fk_locacao*)

05. DICIONÁRIO DE DADOS

Endereço			
Tabela para o cadastro de endereços no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
id	Código de identificação do endereço.	Inteiro	PRIMARY KEY, SERIAL
logradouro	Nome da rua/avenida do endereço	Caractere	NOT NULL
bairro	Bairro do endereço	Caractere	NOT NULL
cidade	Cidade do endereço	Caractere	NOT NULL
uf	Estado do endereço	Caractere	NOT NULL
cep	Código de Endereçamento Postal do endereço	Caractere	NOT NULL
numero	Numero onde se localiza o endereço.	Inteiro	NOT NULL

Veículos Tabela para o cadastro de veículos no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
id	Código de identificação do veículos.	Inteiro	PRIMARY KEY, SERIAL
n_placa	Numero da placa do veículo.	Caractere	NOT NULL
n_chassi	Numero do chassi do veículo.	Caractere	
n_motor	Numero do motor do veículo.	Caractere	
cor	Cor do veículo.	Caractere	
toque_motor	Toque do motor do veículo.	Double	
tipo_combustivel	Tipos de combustíveis que podem ser usados no veículo.	Caractere	
quilometragem_atual	Quilometragem atual do veículo.	Double	
fabricante	Fabricante do veículo.	Caractere	
modelo	Modelo do veículo.	Caractere	
n_passageiros	Numero de Passageiros do veículo.	Inteiro	
n_porta	Numero de portas do veículo.	Inteiro	
ano_fabricacao	Ano de fabricação do veículo.	Inteiro	
ano_modelo	Ano do modelo do veiculo.	Inteiro	
status	Status do veículo.	Double	NOT NULL
fk_categoria	Código da categoria em que o veiculo esta cadastrado.	Inteiro	FOREIGN KEY
fk_sede_filial	Código da sede ou filial em que o veiculo esta no momento.	Inteiro	FOREIGN KEY

Reserva Tabela que cadastra as reservas de veículos no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
id	Código de identificação da reserva.	Inteiro	PRIMARY KEY, SERIAL
data_hora_reserva	Data e hora da reserva do veiculo	Caractere	
data_deseja_locar	Dia em que deseja locar o veiculo	Data	NOT NULL
valor_locacao	Valor da reserva	Float	
status	Status da reserva	Boolean	
fk_usuario	Funcionário que realizou a reserva	Inteiro	FOREIGN KEY

fk_cliente	Cliente que reservou o veiculo	Inteiro	FOREIGN KEY
fk_categoria	Categoria da reserva do veiculo	Inteiro	FOREIGN KEY

Locação Tabela que cadastra as locações de veículos no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
id	Código de identificação da locação.	Inteiro	PRIMARY KEY, SERIAL
tipo_locacao	Tipos de locação: Km Livre e Km Controle.	Caractere	NOT NULL
data_hora_retirada	Data e hora que o veiculo foi locado pelo cliente.	Caractere	NOT NULL
data_hora_devolucao	Data e hora que o veiculo deve ser devolvido a locadora.	Caractere	
km_inicial	Km inicial do veículo a ser locado	Float	
km_final	Km final do veículo a ser locado	Float	
status	Status atual da locação	Boolean	NOT NULL
taxa_higienização	Taxa de cobrança de higienização.	Float	
taxa_combustivel	Taxa de cobrança de combustível.	Float	
fk_reserva	Código de identificação da reserva.	Inteiro	FOREIGN KEY
fk_usuario	Funcionário que realizou a locação	Inteiro	FOREIGN KEY
fk_cliente	Cliente que locou o veiculo	Inteiro	FOREIGN KEY
fk_motorista	Motorista do veiculo locado	Inteiro	FOREIGN KEY
fk_veiculo	Veiculo locado	Inteiro	FOREIGN KEY

Usuários Tabela que cadastra os funcionários no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
id	Código de identificação do funcionário.	Inteiro	PRIMARY KEY, SERIAL
nome	Nome do funcionário	Caractere	NOT NULL
status	Status do usuario	Boolean	NOT NULL
CPF	Numero do CPF do funcionário	Caractere	
RG	Numero do RG do funcionário	Caractere	

CTPS	Numero da carteira de trabalho e previdência social do funcionário	Caractere	
tipo_usuario	Se o funcionário é um superusuário, administrador ou atendente	Caractere	NOT NULL
login	Login do funcionário, para entrar no sistema.	Caractere	NOT NULL
senha	Senha do funcionário, para entrar no sistema.	Caractere	NOT NULL
DN	Data de nascimento do funcionário.	Data	
fk_endereco	Endereço do funcionário	Inteiro	FOREIGN KEY
fk_sede_filial	Sede ou filial ao qual o funcionário trabalha.	Inteiro	FOREIGN KEY

Sede/Filial Tabela que cadastra a sede ou as filiais da empresa no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
id	Código de identificação da Sede/Filial.	Inteiro	PRIMARY KEY, SERIAL
nome	Nome da empresa	Caractere	NOT NULL
Tipo	Se é a sede ou uma filial	Caractere	
CNPJ	Numero do Cadastro Nacional da Pessoa Jurídica	Caractere	
status	Status da sede ou da filial	Boolean	NOT NULL
fk_endereco	Endereço da sede ou da filial.	Inteiro	FOREIGN KEY

Motorista Tabela que cadastra os motoristas dos veículos no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
id	Código de identificação do motorista.	Inteiro	PRIMARY KEY, SERIAL
nome	Nome do motorista	Caractere	NOT NULL
status	Status do motorista	Boolean	NOT NULL
CNH	Numero da CNH do motorista	Caractere	NOT NULL
data_vencimento_CHN	Data de vencimento da CHN do motorista	Data	NOT NULL
DN	Data de nascimento do motorista	Data	NOT NULL

Cliente

Tabela que cadastra os clientes no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
id	Código de identificação dos clientes.	Inteiro	PRIMARY KEY, SERIAL
nome	Nome do cliente	Caractere	NOT NULL
status	Status do cliente	Boolean	
fk_endereco	Endereço dos clientes	Inteiro	FOREIGN KEY

Pessoa Física Tabela faz referência a tabela Cliente			
Atributo	Descrição	Tipo de Dado	Restrições
fk_cliente	Código de identificação dos clientes PF.	Inteiro	FOREIGN KEY
CPF	Numero do CPF do cliente PF	Caractere	NOT NULL
sexo	Sexo do cliente PF	Caractere	
DN	Data de nascimento do cliente PF	Data	

Pessoa Jurídica Tabela faz referência a tabela Cliente			
Atributo	Descrição	Tipo de Dado	Restrições
<u>fk_cliente</u>	Código de identificação dos clientes PJ.	Inteiro	FOREIGN KEY
CNPJ	O Cadastro Nacional da Pessoa Jurídica, do cliente PJ	Caractere	NOT NULL
insc_estadual	Inscrição estadual do cliente PJ	Caractere	

Categoria Tabela que cadastra as categorias no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
id	Código de identificação da categoria.	Inteiro	PRIMARY KEY, SERIAL
codigo	Código da categoria	Caractere	NOT NULL
nome	Nome da categoria	Caractere	NOT NULL
descricao	Descrição da categoria	Caractere	
tamanho	O tamanho do veículo: pequeno, médio ou grande.	Caractere	
Status	Status da categoria	Boolean	NOT NULL
ar_condicionado	Se o veículo possui acessório: ar-condicionado	Boolean	
radio	Se o veículo possui	Boolean	

	acessório: radio		
DVD	Se o veículo possui acessório: DVD	Boolean	
MP3	Se o veículo possui acessório: MP3	Boolean	
direcao_hidraulica	Se o veículo possui acessório: direção hidráulica	Boolean	
camera_re	Se o veículo possui acessório: câmera de ré	Boolean	
tipo_cambio	Se o câmbio é automático ou manual.	Caractere	
valor_locar_categoria	Valor cobrado dessa categoria	Float	NOT NULL

Automóveis Tabela que cadastra os veículos na categoria automóvel no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
fk_categoria	Código de identificação da categoria.	Inteiro	FOREIGN KEY

Camionetas de Carga Tabela que cadastra os veículos na categoria camionetas de carga no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
fk_categoria	Código de identificação da categoria.	Inteiro	FOREIGN KEY
capacidade_carga	Capacidade de carga do veículo (em toneladas).	Float	
tipo_embreagem	O acionamento da embreagem pode ser manual ou hidráulico.	Caractere	
desempenho	Desempenho do veículo (relação consumo/km).	Float	
potencia_motor	potência do motor	Float	
volume_abastecimento	Volume de abastecimento de combustível	Float	
distancia_eixos	Distância entre eixos.	Float	

Camionetas de Passageiros Tabela que cadastra os veículos na categoria camionetas de passageiros no sistema			
Atributo	Descrição	Tipo de Dado	Restrições
fk_categoria	Código de identificação da categoria.	Inteiro	FOREIGN KEY
tipo_air_bag	Se o air bag é simples-	Caractere	

	dianteira, duplo-dianteira ou total.		
direcao_assistida	Se possuem direção assistida.	Boolean	
cintos_traseiros_retrateis	Se possuem cintos de segurança traseiros retráteis	Boolean	
rodas_liga_leve	Se possuem rodas de liga leve	Boolean	
cont_poluicao_ar	Se possuem controle de poluição do ar	Boolean	

Controle Financeiro Tabela de controle financeiro			
Atributo	Descrição	Tipo de Dado	Restrições
id	Código de identificação do financeiro.	Inteiro	PRIMARY KEY, SERIAL
descricao	Descrição das contas ou pagamentos no sistema	Caractere	
valor	Valor das contas ou pagamentos.	Float	NOT NULL
dataFincanceiro	Data das contas ou pagamentos.	Data	NOT NULL
tipo	Tipo das contas.	Caractere	NOT NULL
formas_pagamento	As formas de pagamento: Dinheiro, Cartão de crédito ou débito, boleto...	Caractere	NOT NULL
fk_locacao	Código de identificação dos locação.	Inteiro	FOREIGN KEY
fk_usuario	Código de identificação dos funcionários que estão fazendo o controle.	Inteiro	FOREIGN KEY

05. SCRIPTS SQL

5.1 DDL

CATEGORIA
<pre> create table CATEGORIA (id int not null, DVD boolean not null, MP3 boolean not null, ar_condicionado boolean not null, camera_re boolean not null, codigo varchar(5) not null, direcao_hidraulica boolean not null, nome varchar(255) not null, </pre>

radio boolean not null,
status boolean not null,
tamanho varchar(50),
descricao varchar(50),
tipo_cambio varchar(50),
valor_locar_categoria float not null,
primary key (id)

)

CATEGORIA_AUTOMOVEL

```
create table CATEGORIA_AUTOMOVEL (  
    id int not null,  
    primary key (id)  
)  
alter table CATEGORIA_AUTOMOVEL  
    add constraint FKmlux6rsvg0m5voaf809l4nnvh  
    foreign key (id)  
    references CATEGORIA
```

CATEGORIA_CAMINONETA_CARGA

```
create table CATEGORIA_CAMINONETA_CARGA (  
    capacidade_carga float not null,  
    desempenho float not null,  
    distancia_eixos float not null,  
    potencia_motor float not null,  
    tipo_embreagem varchar(255),  
    volume_abastecimento float not null,  
    id int not null,  
    primary key (id)  
)  
alter table CATEGORIA_CAMINONETA_CARGA  
    add constraint FKKhgo16v6ldsi3u4dobrt7um3yc  
    foreign key (id)  
    references CATEGORIA
```

CATEGORIA_CAMIONETA_PASSAGEIRO

```
create table CATEGORIA_CAMIONETA_PASSAGEIRO (  
    cintos_traseiros_retrateis boolean not null,  
    cont_poluicao_ar boolean not null,  
    direcao_assistida boolean not null,  
    rodas_liga_leve boolean not null,  
    tipo_air_bag varchar(50),  
    id int not null,  
    primary key (id)  
)  
alter table CATEGORIA_CAMIONETA_PASSAGEIRO  
    add constraint FK10uyyps6pamidomefkk6b5jnnm  
    foreign key (id)  
    references CATEGORIA
```

CLIENTE

```
create table CLIENTE (  
    id int not null,  
    nome varchar(255) not null,  
    status boolean not null,  
    Id_Endereco int,  
    primary key (id)  
)  
alter table CLIENTE  
    add constraint FKguraahagpuqf00rtcmk5e1g7y  
    foreign key (Id_Endereco)  
    references ENDERECO
```

CLIENTE_PESSOA_FISICA

```
create table CLIENTE_PESSOA_FISICA (  
    CPF varchar(15) not null,  
    DN date,  
    sexo varchar(10),  
    id int not null,  
    primary key (id)  
)  
alter table CLIENTE_PESSOA_FISICA  
    add constraint UK_7ebcig87jbdkup47ci773v3oj unique (CPF)  
alter table CLIENTE_PESSOA_FISICA  
    add constraint FK8a6a8b9ddqoo7g18uguca042e  
    foreign key (id)  
    references CLIENTE
```

CLIENTE_PESSOA_JURIDICA

```
create table CLIENTE_PESSOA_JURIDICA (  
    CNPJ varchar(20) not null,  
    insc_estadual varchar(30),  
    id int not null,  
    primary key (id)  
)  
alter table CLIENTE_PESSOA_JURIDICA  
    add constraint FKdwjo01mk4pu8fxbqoc0x6c1qk  
    foreign key (id)  
    references CLIENTE
```

CONTROLE_FINANCEIRO

```
create table CONTROLE_FINANCEIRO (  
    id int not null,  
    dataFincanceiro date not null,  
    descricao varchar(255),  
    tipo varchar(255) not null,  
    valor float not null,  
    formas_pagamento varchar(255) not null,
```



```

        Id_Locacao int,
        Id_Usuario int not null,
        primary key (id)
    )
alter table CONTROLE_FINANCEIRO
    add constraint FKtpsnnlepdd5vkv4mrn25tilx
        foreign key (Id_Locacao)
        references LOCACAO
alter table CONTROLE_FINANCEIRO
    add constraint FKornrxvvpoeih7of9rlrrchw2i
        foreign key (Id_Usuario)
        references USUARIO

```

ENDERECO

```

create table ENDERECO (
    id int not null,
    bairro varchar(50) not null,
    cep varchar(10) not null,
    cidade varchar(100) not null,
    logradouro varchar(100) not null,
    numero int not null,
    uf varchar(2) not null,
    primary key (id)
)

```

LOCACAO

```

create table LOCACAO (
    id int not null,
    data_hora_devolucao varchar(255),
    data_hora_retirada varchar(255),
    km_final float not null,
    km_inicial float not null,
    metade_valor_locacao float not null,
    status boolean not null,
    taxa_combustivel float not null,
    taxa_higienizacao float not null,
    tipo_locacao varchar(50) not null,
    Id_Cliente int,
    Id_Motorista int,
    Id_Reserva int,
    Id_Usuario int,
    Id_Veiculo int,
    primary key (id)
)
alter table LOCACAO
    add constraint FK9njhlbmtuw46m3ruocy1pso6b
        foreign key (Id_Cliente)
        references CLIENTE
alter table LOCACAO
    add constraint FKqdf27rvp8l3n0u4wdexh4nw15

```

```

foreign key (Id_Motorista)
references MOTORISTA
alter table LOCACAO
add constraint FK5tum65mlvhh4ijqqg8tdubnlb
foreign key (Id_Reserva)
references RESERVA
alter table LOCACAO
add constraint FKk9609w7tkqx6fo6iai3cf76jo
foreign key (Id_Usuario)
references USUARIO
alter table LOCACAO
add constraint FK6h8799tbk4y4i2eaiqfwi1cr6
foreign key (Id_Veiculo)
references VEICULO

```

MOTORISTA

```

create table MOTORISTA (
  id int not null,
  CNH varchar(20) not null,
  DN date not null,
  data_vencimento_CHN date not null,
  nome varchar(255),
  status boolean not null,
  primary key (id)
)
alter table MOTORISTA
add constraint UK_onjx8wmb2rm9nx43j3mcnc5eh unique (CNH)

```

RESERVA

```

create table RESERVA (
  id int not null,
  data_deseja_locar date not null,
  data_hora_reserva varchar(255),
  status boolean not null,
  valor_locacao float not null,
  Id_Categoria int,
  Id_Cliente int,
  Id_Usuario int,
  primary key (id)
)
alter table RESERVA
add constraint FK2liwb54txag5p2lm19ltwqto8
foreign key (Id_Categoria)
references CATEGORIA
alter table RESERVA
add constraint FKmcr9iyuqr1t8xi4oh71of8ff1
foreign key (Id_Cliente)
references CLIENTE
alter table RESERVA
add constraint FKk4c1qo9wi1wgyfvicfavchy1

```

foreign key (Id_Usuario)
references USUARIO

SEDE_FILIAL

```
create table SEDE_FILIAL (  
    id int not null,  
    CNPJ varchar(20),  
    nome varchar(255) not null,  
    tipo varchar(20),  
    status boolean not null,  
    Id_Endereco int,  
    primary key (id)  
)  
alter table SEDE_FILIAL  
    add constraint FKe6p41v9h047ue555w3f4ocdj7  
    foreign key (Id_Endereco)  
    references ENDERECO
```

USUARIO

```
create table USUARIO (  
    id int not null,  
    status boolean not null,  
    CPF varchar(30),  
    CTPS varchar(30),  
    DN date,  
    RG varchar(30),  
    login varchar(30) not null,  
    nome varchar(255) not null,  
    senha varchar(30) not null,  
    tipo_usuario varchar(30) not null,  
    Id_Endereco int,  
    Id_Sede_Filial int,  
    primary key (id)  
)  
alter table USUARIO  
    add constraint FKodhd56sisgsowxse12bxc3dff  
    foreign key (Id_Endereco)  
    references ENDERECO  
alter table USUARIO  
    add constraint FKp8hvv7miw8jqt8wbhc3v1w4ni  
    foreign key (Id_Sede_Filial)  
    references SEDE_FILIAL
```

VEICULO

```
create table VEICULO (  
    id int not null,  
    ano_fabricacao int not null,  
    ano_modelo int not null,  
    cor varchar(100),
```

```

    fabricante varchar(100),
    modelo varchar(100),
    n_chassi varchar(17),
    n_motor varchar(17),
    n_passageiros int not null,
    n_placa varchar(10) not null,
    n_porta int not null,
    quilometragem_atual float not null,
    status boolean not null,
    tipo_combustivel varchar(100),
    toque_motor float not null,
    Id_Categoria int,
    Id_Sede_Filial int,
    primary key (id)
)
alter table VEICULO
    add constraint FKjekxwj397rr09dsv3wh7h6qc2
    foreign key (Id_Categoria)
    references CATEGORIA
alter table VEICULO
    add constraint FKri28x0to3u4c5bqku77iowarv
    foreign key (Id_Sede_Filial)
    references SEDE_FILIAL

```

5.2. DML

persist Função que irá inserir (persist) uma categoria
<pre> insert into CATEGORIA (DVD, MP3, ar_condicionado, camera_re, codigo, status, direcao_hidraulica, nome, radio, tamanho, tipo_cambio, valor_locar_categoria, id) values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?) </pre>

merge Função que irá editar (merge) uma categoria
<pre> update CATEGORIA set DVD=?, MP3=?, ar_condicionado=?, camera_re=?, codigo=?, status=?, direcao_hidraulica=?, nome=?, radio=?, tamanho=?, </pre>

```
tipo_cambio=?,
valor_locar_categoria=?
where
id=?
```

remove
Função que irá excluir(remove) uma categoria

```
delete from CATEGORIA where id=?
```

cadastrarAutomovel
Função que irá inserir (persist) um automóvel

```
insert into CATEGORIA_AUTOMOVEL
(id)
values
(?)
```

editarAutomovel
Função que irá editar (merge) um automóvel

```
select automovel from CATEGORIA_AUTOMOVEL automovel inner join CATEGORIA
automovel on automovel.id=automovel.id
where
automovel.id=?
```

excluirAutomovel
Função que irá excluir(remove) um automóvel

```
select automovel from CATEGORIA_AUTOMOVEL automovel inner join CATEGORIA
automovel on automovel.id=automovel.id
where
automovel.id=?
delete from CATEGORIA_AUTOMOVEL where id=?
```

cadastrarCamionetasCarga
Função que irá inserir (persist) uma Camioneta de Carga

```
insert into CATEGORIA_CAMINONETA_CARGA
(capacidade_carga, desempenho, distancia_eixos, potencia_motor, tipo_embreagem,
volume_abastecimento, id)
values
(?, ?, ?, ?, ?, ?, ?)
```

editarCamionetasCarga
Função que irá editar (merge) uma Camioneta de Carga

```
select camionetas from CATEGORIA_CAMINONETA_CARGA camionetas inner join
CATEGORIA camionetas on camionetas.id=camionetas.id
where
camionetas.id=?
update CATEGORIA_CAMINONETA_CARGA set
capacidade_carga=?
```

```
desempenho=?,
distancia_eixos=?,
potencia_motor=?,
tipo_embreagem=?,
volume_abastecimento=?
where
id=?
```

excluirCamionetasCarga
Função que irá excluir(remove) uma Camioneta de Carga

```
select camionetas from CATEGORIA_CAMIONETA_CARGA camionetas inner join
CATEGORIA camionetas on camionetas.id=camionetas.id
where
camionetas.id=?
delete from CATEGORIA_CAMIONETA_CARGA where id=?
```

cadastrarCamionetasPassageiros
Função que irá inserir (persist) uma Camioneta de Passageiro

```
insert into CATEGORIA_CAMIONETA_PASSAGEIRO
(cintos_traseiros_retrateis, cont_poluicao_ar, direcao_assistida, rodas_liga_leve,
tipo_air_bag, id)
values
(?, ?, ?, ?, ?, ?)
```

editarCamionetasPassageiros
Função que irá editar (merge) uma Camioneta de Passageiro

```
select camionetas from CATEGORIA_CAMIONETA_PASSAGEIRO camionetas inner join
CATEGORIA camionetas on camionetas.id=camionetas.id
where
camionetas.id=?
update CATEGORIA_CAMIONETA_PASSAGEIRO set
cintos_traseiros_retrateis=?,
cont_poluicao_ar=?,
direcao_assistida=?,
rodas_liga_leve=?,
tipo_air_bag=?
where
id=?
```

excluirCamionetasPassageiros
Função que irá excluir(remove) uma Camioneta de Passageiro

```
select camionetas from CATEGORIA_CAMIONETA_PASSAGEIRO camionetas inner join
CATEGORIA camionetas on camionetas.id=camionetas.id
where
camionetas.id=?
delete from CATEGORIA_CAMIONETA_PASSAGEIRO where id=?
```

cadastrarEndereco

<p>Função que irá inserir (persist) um Endereço</p> <pre> insert into ENDERECO (bairro, cep, cidade, logradouro, numero, uf, id) values (?, ?, ?, ?, ?, ?, ?) </pre>
<p>editarEndereco Função que irá editar (merge) um Endereço</p> <pre> update ENDERECO set bairro=?, cep=?, cidade=?, logradouro=?, numero=?, uf=? where id=? </pre>
<p>excluirEndereco Função que irá excluir(remove) um Endereço</p> <pre> delete from ENDERECO where id=? </pre>
<p>cadastrarMotorista Função que irá inserir (persist) um Motorista</p> <pre> insert into MOTORISTA (CNH, DN, data_vencimento_CHN, nome, status, id) values (?, ?, ?, ?, ?, ?) </pre>
<p>editarMotorista Função que irá editar (merge) um Motorista</p> <pre> update MOTORISTA set CNH=?, DN=?, data_vencimento_CHN=?, nome=?, status=? where id=? </pre>
<p>excluirMotorista Função que irá excluir(remove) um Motorista</p> <pre> delete from MOTORISTA where id=? </pre>
<p>persist Função que irá inserir (persist) um Cliente</p> <pre> insert into CLIENTE </pre>

(Id_Endereco, nome, status, id) values (?, ?, ?, ?)

merge Função que irá editar (merge) um Cliente

update CLIENTE set nome=?, status=?, Id_Endereco=? where id=?
--

remove Função que irá excluir(remove) um Cliente

delete from CLIENTE where id=?

cadastrarPessoaFisica Função que irá inserir (persist) uma Pessoa Física

insert into CLIENTE_PESSOA_FISICA (CPF, DN, sexo, id) values (?, ?, ?, ?)
--

editarPessoaFisica Função que irá editar (merge) uma Pessoa Física

update CLIENTE_PESSOA_FISICA set CPF=?, DN=?, sexo=? where id=?
--

excluirPessoaFisica Função que irá excluir(remove) uma Pessoa Física

delete from CLIENTE where id=?

cadastrarPessoaJuridica Função que irá inserir (persist) uma Pessoa Juridica

insert into CLIENTE_PESSOA_JURIDICA (CNPJ, insc_estadual, id) values (?, ?, ?)

editarPessoaJuridica Função que irá editar (merge) uma Pessoa Juridica

```
update CLIENTE_PESSOA_JURIDICA set
    CNPJ=?,
    insc_estadual=?
where
    id=?
```

excluirPessoaJuridica
Função que irá excluir(remove) uma Pessoa Juridica

```
delete from CLIENTE_PESSOA_JURIDICA where id=?
```

cadastrarVeiculo
Função que irá inserir (persist) um Veiculo

```
insert into VEICULO
    (ano_fabricacao, ano_modelo, cor, fabricante, Id_Categoria, Id_Sede_Filial, modelo,
    n_chassi, n_motor, n_passageiros, n_placa, n_porta, quilometragem_atual, status,
    tipo_combustivel, toque_motor, id)
values
    (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
```

editarVeiculo
Função que irá editar (merge) um Veiculo

```
update VEICULO set
    ano_fabricacao=?,
    ano_modelo=?,
    cor=?,
    fabricante=?,
    Id_Categoria=?,
    Id_Sede_Filial=?,
    modelo=?,
    n_chassi=?,
    n_motor=?,
    n_passageiros=?,
    n_placa=?,
    n_porta=?,
    quilometragem_atual=?,
    status=?,
    tipo_combustivel=?,
    toque_motor=?
where
    id=?
```

excluirVeiculo
Função que irá excluir(remove) um Veiculo

```
delete from VEICULO where id=?
```

cadastrarLocacao
Função que irá inserir (persist) uma Locação

```
insert into LOCACAO
```

<pre>(data_hora_devolucao, data_hora_retirada, Id_Cliente, Id_Motorista, Id_Reserva, Id_Usuario, Id_Veiculo, km_final, km_inicial, metade_valor_locacao, status, taxa_combustivel, taxa_higienizacao, tipo_locacao, id) values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)</pre>
--

<p>editarLocacao Função que irá editar (merge) uma Locação</p> <pre>update LOCACAO set data_hora_devolucao=?, data_hora_retirada=?, Id_Cliente=?, Id_Motorista=?, Id_Reserva=?, Id_Usuario=?, Id_Veiculo=?, km_final=?, km_inicial=?, metade_valor_locacao=?, status=?, taxa_combustivel=?, taxa_higienizacao=?, tipo_locacao=? where id=?</pre>
--

<p>excluirLocacao Função que irá excluir(remove) uma Locação</p> <pre>delete from LOCACAO where id=?</pre>
--

<p>cadastrarReserva Função que irá inserir (persist) uma Reserva</p> <pre>insert into RESERVA (data_deseja_locar, data_hora_reserva, Id_Categoria, Id_Cliente, Id_Usuario, status, valor_locacao, id) values (?, ?, ?, ?, ?, ?, ?, ?)</pre>

<p>editarReserva Função que irá editar (merge) uma Reserva</p> <pre>update RESERVA set data_deseja_locar=?, data_hora_reserva=?, Id_Categoria=?, Id_Cliente=?, Id_Usuario=?, status=?, valor_locacao=? where</pre>
--

id=?

excluirReserva

Função que irá excluir(remove) uma Reserva

delete from RESERVA where id=?

cadastrarControleFinanceiro

Função que irá inserir (persist) um Controle Financeiro

insert into CONTROLE_FINANCEIRO
 (dataFincanceiro, descricao, Id_Locacao, Id_Usuario, forma_pagamento, tipo, valor, id)
values
 (?, ?, ?, ?, ?, ?, ?, ?)

editarControleFinanceiro

Função que irá editar (merge) um Controle Financeiro

update CONTROLE_FINANCEIRO set
 dataFincanceiro=?,
 descricao=?,
 Id_Locacao=?,
 Id_Usuario=?,
 forma_pagamento=?,
 tipo=?,
 valor=?
where
 id=?

excluirControleFinanceiro

Função que irá excluir(remove) um Controle Financeiro

delete from CONTROLE_FINANCEIRO where id=?

cadastrarSedeFilial

Função que irá inserir (persist) uma Sede ou Filial

insert into SEDE_FILIAL
 (CNPJ, Id_Endereco, nome, status, tipo, id)
values
 (?, ?, ?, ?, ?, ?)

editarSedeFilial

Função que irá editar (merge) uma Sede ou Filial

update SEDE_FILIAL set
 CNPJ=?,
 Id_Endereco=?,
 nome=?,
 status=?,
 tipo=?
where
 id=?

excluirSedeFilial Função que irá excluir(remove) uma Sede ou Filial
--

delete from SEDE_FILIAL where id=?

cadastrarUsuario Função que irá inserir (persist) um Usuário

insert into USUARIO (CPF, CTPS, DN, RG, Id_Endereco, Id_Sede_Filial, login, nome, senha, status, tipo_usuario, id) values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
--

editarUsuario Função que irá editar (merge) um Usuário

update USUARIO set CPF=?, CTPS=?, DN=?, RG=?, Id_Endereco=?, Id_Sede_Filial=?, login=?, nome=?, senha=?, status=?, tipo_usuario=? where id=?

Função que irá excluir(remove) um Usuário
--

delete from USUARIO where id=?

06. ANEXOS

6.1 Caso de Uso:

