

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

**Luiz Filipe Alves da Silva**

**SERRA TALHADA, PE**

**10/2018**

**Luiz Filipe Alves da Silva**

Projeto desenvolvido para efetivação da 1ª 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)**

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 bicombustível – nesse caso, especificar quais) e quilometragem atual.

O 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.

As divisões em classificações serão definidos pelos aluno.

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 revisão...

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] Manter Cliente |
| ***Descrição:*** | *Inserir e editar pessoas físicas e jurídicas* |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RF02] Manter Filiais |
| ***Descrição:*** | Adicionar e editar filiais de cada cidade |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RF03] Manter veículo |
| ***Descrição:*** | Adicionar, editar, desativar e atualizar estado(aluguel) de veículos |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RF04] Realizar Reserva |
| ***Descrição:*** | reservas para veículos em uma filial, com previsão de quais veículos estarão disponíveis em uma data futura. |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RF05] Buscar veículos disponíveis |
| ***Descrição:*** | Verificar quais veículos estão disponíveis para reserva. Isto inclui estar em bom estado, limpo e sem reserva prevista |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RF06] Manter motorista |
| ***Descrição:*** | inclui manter cliente |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RF07] Manter locação |
| ***Descrição:*** | Adicionar locação |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RF08] Gerar relatorios |
| ***Descrição:*** | Relatórios para clientes, veículos, locações e reservas. |
| ***Prioridade:*** | ( ) Essencial ( ) Importante ( x ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RF09] Manter Usuário |
| ***Descrição:*** | Incluir e editar usuário |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

**2.2. Requisitos Não-Funcionais:**

|  |  |
| --- | --- |
| ***Identificação:*** | [RNF01] Backup |
| ***Tipo:*** | *Segurança* |
| ***RF Relacionado:*** | *Nenhum* |
| ***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 (configurada no banco de dados) |
| ***Prioridade:*** | ( ) Essencial ( x ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RNF02] Log |
| ***Tipo:*** | *Auditoria* |
| ***RF Relacionado:*** | *Nenhum* |
| ***Descrição:*** | O sistema deve mostrar o log de acesso do sistema para os administradores, mostrando as modificações nas tabelas selecionadas por períodos |
| ***Prioridade:*** | ( ) Essencial ( ) Importante ( x ) Desejável |

**2.3. Requisitos de Domínio:**

|  |  |
| --- | --- |
| ***Identificação:*** | [RD01] Multiusuário |
| ***Descrição:*** | O acesso ao sistema será feito em nível multiusuário, subdivididos em três grupos: superusuário, que acesso total e irrestrito ao SW e ao BD; administração, que tem acesso total ao sistema; Atendentes, que efetuam as locações |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RD02] Acesso ao sistema |
| ***Descrição:*** | Todos os usuários devem ser cadastrados como usuários do banco de dados, e só podem ter acesso às tabelas de seu grupo de usuários |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

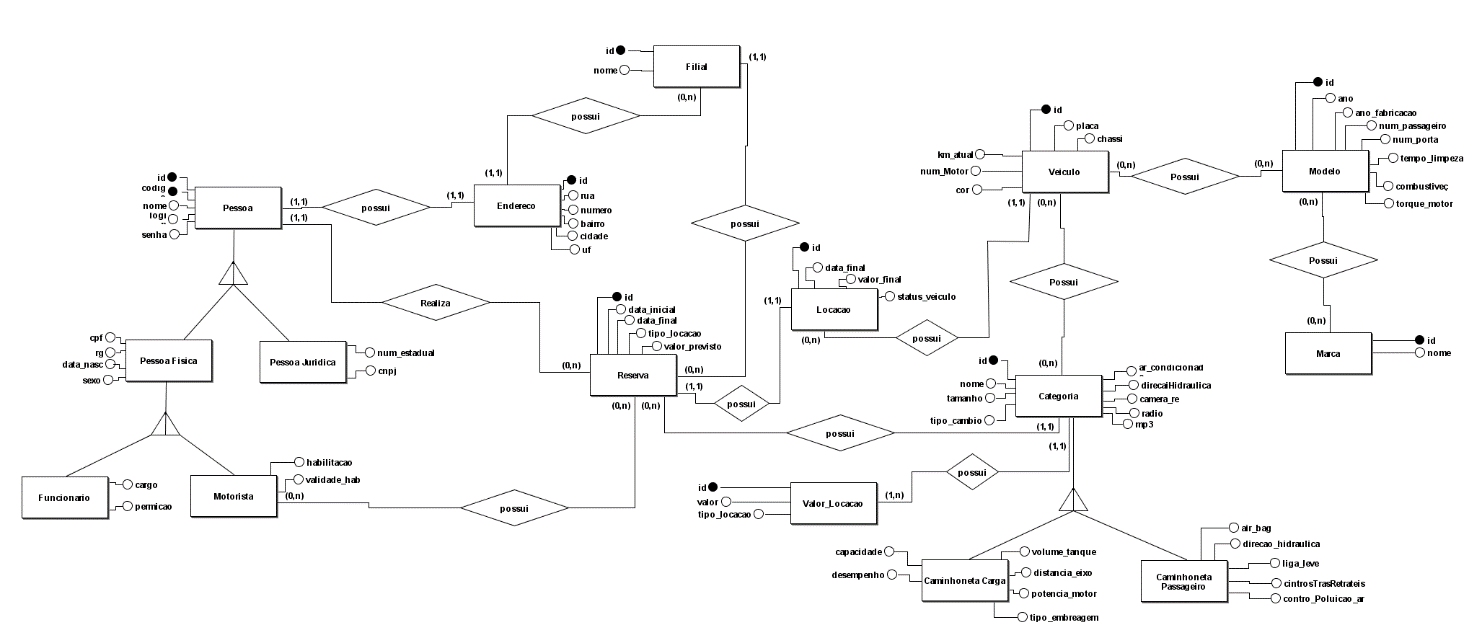
|  |  |
| --- | --- |
| ***Identificação:*** | [RD03] Tamanho da senha |
| ***Descrição:*** | *O campo senha dos usuários deverá aceitar caracteres alfanuméricos, com tamanho entre 6 e 11, distinguindo caracteres maiúsculos de minúsculos* |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RD04] Criptografia |
| ***Descrição:*** | *Todas as senhas devem ser criptografadas no banco de dados* |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RD05] Linguagem |
| ***Descrição:*** | *Deve ser utilizada a linguagem de programação java com jpa* |
| ***Prioridade:*** | ( x ) Essencial ( ) Importante ( ) Desejável |

|  |  |
| --- | --- |
| ***Identificação:*** | [RD06] Banco de dados |
| ***Descrição:*** | *Deve ser utilizado Postgresql* |
| ***Prioridade:*** | ( ) Essencial ( ) Importante ( x ) Desejável |

|  |
| --- |
| **03. DIAGRAMA ENTIDADE-RELACIONAMENTO** |

**

**04. MAPEAMENTO DO ESQUEMA CONCEITUAL PARA O**

**ESQUEMA LÓGICO-RELACIONAL**

Endereco (

id int not null,

bairro varchar(255),

cidade varchar(255),

numero varchar(10),

rua varchar(255),

uf varchar(2),

primary key (id)

);

Categoria (

id int not null,

arCondicionado boolean not null,

cameraRe boolean not null,

direcaoHidraulica boolean not null,

mp3 boolean not null,

nome varchar(75),

radio boolean not null,

tamanho varchar(25),

tipoCambio varchar(25),

capacidade int,

desempenho varchar(255),

distanciaEixo int,

potenciaMotor float,

tipoEmbreagem varchar(255),

volumeTanque int4,

airBag boolean,

cintosTrasRetrateis boolean,

contrPol boolean,

direcaoAssistida boolean,

ligaLeve boolean,

primary key (id),

constraint unique (nome)

);

Marca (

id int not null,

nome varchar(75),

primary key (id)

);

Pessoa (

id int not null,

codigo varchar(50),

login varchar(255),

nome varchar(150),

senha varchar(16),

cpf varchar(15),

dataNascimento timestamp,

rg varchar(25),

sexo varchar(10),

cargo varchar(75),

permissao varchar(75),

cnpj varchar(19),

inscricaoEstadual varchar(25),

habilitacao varchar(75),

validadeHabilitacao timestamp,

endereco\_id int,

primary key (id)

constraint unique (codigo)

constraint unique (login)

constraint unique (cpf)

constraint unique (cnpj)

constraint unique (inscricaoEstadual)

constraint foreign key (endereco\_id) references Endereco

) ;

Filial (

id int not null,

nome varchar(75),

endereco\_id int4,

primary key (id),

constraint unique (nome),

constraint foreign key (endereco\_id) references Endereco

);

Modelo (

id int not null,

anoFabricacao int not null,

anoModelo int not null,

combustivel varchar(255),

nome varchar(255),

numPassageiro int,

numPorta int4 not null,

tempoLimpeza int not null,

torqueMotor float not null,

marca\_id int4,

primary key (id)

constraint foreign key (marca\_id) references Marca

);

Reserva (

id int not null,

dataFinalPrevista timestamp,

dataInicial timestamp,

dataReserva timestamp,

tipoLocacao varchar(10),

valorPrevisto float8 not null,

categoria\_id int,

cliente\_id int,

filial\_id int,

filialEntrega\_id int,

motorista\_id int,

primary key (id),

constraint foreign key (categoria\_id) references Categoria

constraint foreign key (cliente\_id) references Pessoa

constraint foreign key (filial\_id) references Filial

constraint foreign key (filialEntrega\_id) references Filial

constraint foreign key (motorista\_id) references Pessoa

)

ValorLocacao (

id int not null,

tipoLocacao varchar(10),

valor float not null,

categoria\_id int,

primary key (id),

constraint foreign key (categoria\_id) references Categoria

);

Veiculo (

id int not null,

chassi varchar(255),

cor varchar(15),

kmAtual float not null,

numMotor int not null,

placa varchar(8),

categoria\_id int,

modelo\_id int,

primary key (id),

constraint unique (chassi),

constraint unique (placa),

constraint foreign key (categoria\_id) references Categoria,

constraint foreign key (modelo\_id) references Modelo

);

Locacao (

id int not null,

dataFinal timestamp,

statusVeiculo varchar(25),

valorFinal float8 not null,

reserva\_id int,

veiculo\_id int,

primary key (id),

constraint foreign key (reserva\_id) references Reserva

constraint foreign key (veiculo\_id) references Veiculo

);

**05. DICIONÁRIO DE DADOS**

|  |  |  |
| --- | --- | --- |
| **Endereço** | | |
| Atributo | Tipo de Dado | Restrições |
| id | int | Primary Key |
| rua | varchar(100) | - |
| numero | varchar(8) | - |
| bairro | varchar(100) | - |
| cidade | varchar(100) | - |

|  |  |  |
| --- | --- | --- |
| **Pessoa** | | |
| Atributo | Tipo de Dado | Restrições |
| id | int | Primary Key |
| codigo | varchar(50) | Unique |
| nome | varchar(150) | - |
| login | vachar(20) | Unique |
| senha | varchar(16) | - |
| endereco\_id | int | FK - Endereco(id) |

|  |  |  |
| --- | --- | --- |
| **PessoaFisica**  **(Extend Pessoa)** | | |
| Atributo | Tipo de Dado | Restrições |
| cpf | varchar(15) | Unique |
| rg | varchar(20) | Unique |
| data\_nasc | date | - |
| sexo | vachar(10) | - |

|  |  |  |
| --- | --- | --- |
| **PessoaJuridica**  **(Extend Pessoa)** | | |
| Atributo | Tipo de Dado | Restrições |
| cnpj | varchar(20) | Unique |
| insc\_estad | varchar(30) | Unique |

|  |  |  |
| --- | --- | --- |
| **Motorista**  **(Extend Pessoa)** | | |
| Atributo | Tipo de Dado | Restrições |
| num\_habilit | varchar(11) | Unique |
| val\_habilit | date | - |

|  |  |  |
| --- | --- | --- |
| **Funcionario**  **(Extend Pessoa)** | | |
| Atributo | Tipo de Dado | Restrições |
| cargo | varchar(25) | - |
| tipo\_permissao | varchar(25) | - |

|  |  |  |
| --- | --- | --- |
| **Filial** | | |
| Atributo | Tipo de Dado | Restrições |
| id | int | Primary Key |
| nome | varchar(100) | Unique |
| endereco\_id | int | FK -Endereço(id) |

|  |  |  |
| --- | --- | --- |
| **Categoria** | | |
| Atributo | Tipo de Dado | Restrições |
| id | int | Primary Key |
| nome | varchar(100) | - |
| tamanho | varchar(30) | - |
| tipo\_cambio | varchar(30) | - |
| ar\_condicionado | boolean | - |
| dir\_hidraulica | boolean | - |
| camera\_re | boolean | - |
| radio | boolean | - |
| mp3 | boolean | - |

|  |  |  |
| --- | --- | --- |
| **Caminhoneta\_Passageiro**  **(Extend Categoria)** | | |
| Atributo | Tipo de Dado | Restrições |
| air\_bag | boolean | **-** |
| dir\_assistida | boolean | **-** |
| liga\_leve | boolean | **-** |
| cintos\_tras\_retra | boolean | **-** |
| contr\_pol | boolean | **-** |

|  |  |  |
| --- | --- | --- |
| **Caminhoneta\_Carga**  **(Extend Categoria)** | | |
| Atributo | Tipo de Dado | Restrições |
| capacidade | int | **-** |
| desempenho | varchar(30) | **-** |
| volume\_tanque | int | **-** |
| dist\_eixo | int | **-** |
| potencia\_motor | real | **-** |
| tipo\_embreagem | varcha(30) | **-** |

|  |  |  |
| --- | --- | --- |
| **Reserva** | | |
| Atributo | Tipo de Dado | Restrições |
| id | int | Primary Key |
| data\_reserva | datetime | - |
| data\_inicial | dateTime | - |
| data\_final\_prev | dateTime | - |
| tipo\_locacao | varchar(20) | - |
| valor\_previsto | real | - |
| cliente\_id | int | FK - Pessoa(id) |
| motorista\_id | int | FK - Motorista(id) |
| filial\_id | int | FK - Filial(id) |
| categoria\_id | int | FK – Categoria(id) |

|  |  |  |
| --- | --- | --- |
| **Marca** | | |
| Atributo | Tipo de Dado | Restrições |
| id | int | Primary Key |
| nome | Varchar(75) | - |

|  |  |  |
| --- | --- | --- |
| **Modelo** | | |
| Atributo | Tipo de Dado | Restrições |
| id | int | Primary Key |
| Nome | Varchar(75) | - |
| torque\_motor | float | - |
| modelo | varchar(30) | - |
| combustivel | varchar(25) | - |
| tempo\_lipeza | int | - |
| num\_passageiro | int | - |
| num\_porta | int | - |
| ano\_fab | int | - |
| ano\_model | int | - |
| Marca\_id | Int | FK – Marca(id) |

|  |  |  |
| --- | --- | --- |
| **Veiculo** | | |
| Atributo | Tipo de Dado | Restrições |
| id | int | Pimary Key |
| placa | varchar(25) | - |
| chassi | varhar(25) | - |
| km\_atual | double | - |
| num\_motor | int | - |
| cor | varchar(25) | - |
| categoria\_id | int | FK - Categoria(id) |
| modelo\_id | int | FK-Modelo(id) |

|  |  |  |
| --- | --- | --- |
| **Locacao** | | |
| Atributo | Tipo de Dado | Restrições |
| id | int | Primary Key |
| data\_final | date | - |
| valor\_final | double | - |
| status\_veiculo | varchar(25) | - |
| vaiculo\_id | int | FK - Veiculo(id) |
| reserva\_id | int | FK - Reserva(id) |

**05. SCRIPTS SQL**

**5.1 DDL**

|  |
| --- |
| **Pessoa** |
| CREATE TABLE public.pessoa  (  dtype character varying(31) NOT NULL,  id integer NOT NULL,  codigo character varying(50),  login character varying(255),  nome character varying(150),  senha character varying(16),  cpf character varying(15),  datanascimento timestamp without time zone,  rg character varying(25),  sexo character varying(10),  cargo character varying(75),  permissao character varying(75),  cnpj character varying(19),  inscricaoestadual character varying(25),  habilitacao character varying(75),  validadehabilitacao timestamp without time zone,  endereco\_id integer,  CONSTRAINT pessoa\_pkey PRIMARY KEY (id),  CONSTRAINT fketlggxage7ixu17gaf2nx53ij FOREIGN KEY (endereco\_id)  REFERENCES public.endereco (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT uk\_1ixtd34q2u4f3uxt4hoovwoqb UNIQUE (inscricaoestadual),  CONSTRAINT uk\_bolqf2tvb1dblmhfh6k5r3fmd UNIQUE (login),  CONSTRAINT uk\_gej40f8jfd5efnwlggtpwjloo UNIQUE (cpf),  CONSTRAINT uk\_hgva89erw7iq5d1718ia26bjh UNIQUE (cnpj)  ) |

|  |
| --- |
| **Categoria** |
| CREATE TABLE public.categoria  (  dtype character varying(31) NOT NULL,  id integer NOT NULL,  arcondicionado boolean NOT NULL,  camerare boolean NOT NULL,  direcaohidraulica boolean NOT NULL,  mp3 boolean NOT NULL,  nome character varying(75),  radio boolean NOT NULL,  tamanho character varying(25),  tipocambio character varying(25),  capacidade integer,  desempenho character varying(255),  distanciaeixo integer,  potenciamotor real,  tipoembreagem character varying(255),  volumetanque integer,  airbag boolean,  cintostrasretrateis boolean,  contrpol boolean,  direcaoassistida boolean,  ligaleve boolean,  CONSTRAINT categoria\_pkey PRIMARY KEY (id),  CONSTRAINT uk\_aq7m0c1f35i5jrrn245usbc1t UNIQUE (nome)  ) |

|  |
| --- |
| **Endereco** |
| CREATE TABLE public.endereco  (  id integer NOT NULL,  bairro character varying(255),  cidade character varying(255),  numero character varying(10),  rua character varying(255),  uf character varying(2),  CONSTRAINT endereco\_pkey PRIMARY KEY (id)  ) |

|  |
| --- |
| **Filial** |
| CREATE TABLE public.filial  (  id integer NOT NULL,  nome character varying(75),  endereco\_id integer,  CONSTRAINT filial\_pkey PRIMARY KEY (id),  CONSTRAINT fkpbwvco1cmulwlwmsjuew2y255 FOREIGN KEY (endereco\_id)  REFERENCES public.endereco (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT uk\_6iufanokhdce2p5gkd1c0x4ji UNIQUE (nome)  ) |

|  |
| --- |
| **Locacao** |
| CREATE TABLE public.locacao  (  id integer NOT NULL,  datafinal timestamp without time zone,  statusveiculo character varying(25),  valorfinal double precision NOT NULL,  reserva\_id integer,  veiculo\_id integer,  status character varying(255),  CONSTRAINT locacao\_pkey PRIMARY KEY (id),  CONSTRAINT fk7ecbnfxee42w33etsijqyp046 FOREIGN KEY (reserva\_id)  REFERENCES public.reserva (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fkhk6p0qn75inply3qqjvuaqk96 FOREIGN KEY (veiculo\_id)  REFERENCES public.veiculo (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION  ) |

|  |
| --- |
| **Locacao2** |
| CREATE TABLE public.locacao2  (  id integer NOT NULL,  datafinal timestamp without time zone,  datamodificacao timestamp without time zone,  status character varying(255),  statusveiculo character varying(25),  valorfinal double precision NOT NULL,  reserva\_id integer,  veiculo\_id integer,  CONSTRAINT locacao2\_pkey PRIMARY KEY (id),  CONSTRAINT fkcsktsv3f2g4t6nxy6ibbmj8vc FOREIGN KEY (veiculo\_id)  REFERENCES public.veiculo (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fkp9l8b6mow2oyaefo4aim9aa3p FOREIGN KEY (reserva\_id)  REFERENCES public.reserva (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION  ) ON UPDATE NO ACTION ON DELETE NO ACTION  ) |

|  |
| --- |
| **Marca** |
| CREATE TABLE public.marca  (  id integer NOT NULL,  nome character varying(75),  CONSTRAINT marca\_pkey PRIMARY KEY (id),  CONSTRAINT uk\_31y3b9y9nthoykx6ukxpgvxw UNIQUE (nome)  ) |

|  |
| --- |
| **Modelo** |
| CREATE TABLE public.modelo  (  id integer NOT NULL,  anofabricacao integer NOT NULL,  anomodelo integer NOT NULL,  combustivel character varying(15),  nome character varying(15),  numpassageiro integer NOT NULL,  numporta integer NOT NULL,  tempolimpeza integer NOT NULL,  torquemotor real NOT NULL,  marca\_id integer,  CONSTRAINT modelo\_pkey PRIMARY KEY (id),  CONSTRAINT fk89uvrlg31p26yvjv1eqvvigdr FOREIGN KEY (marca\_id)  REFERENCES public.marca (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT uk\_co3vjgkjauexyd90x0b7r1dd1 UNIQUE (nome)  ) |

|  |
| --- |
| **Pessoa2** |
| CREATE TABLE public.pessoa2  (  dtype character varying(31) NOT NULL,  id integer NOT NULL,  codigo character varying(50),  datamodificacao timestamp without time zone,  login character varying(255),  nome character varying(150),  senha character varying(11),  cnpj character varying(19),  inscricaoestadual character varying(25),  cpf character varying(15),  datanascimento timestamp without time zone,  rg character varying(25),  sexo character varying(10),  habilitacao character varying(75),  validadehabilitacao timestamp without time zone,  endereco\_id integer,  CONSTRAINT pessoa2\_pkey PRIMARY KEY (id),  CONSTRAINT fkiqegwvtdlbuxtd9w32cx9ivg4 FOREIGN KEY (endereco\_id)  REFERENCES public.endereco (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION  ) REFERENCES public.veiculo (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION  ) |

|  |
| --- |
| **Reserva** |
| CREATE TABLE public.reserva  (  id integer NOT NULL,  datafinalprevista timestamp without time zone,  datainicial timestamp without time zone,  datareserva timestamp without time zone,  tipolocacao character varying(10),  valorprevisto double precision NOT NULL,  categoria\_id integer,  cliente\_id integer,  filial\_id integer,  filialentrega\_id integer,  motorista\_id integer,  status character varying(255),  CONSTRAINT reserva\_pkey PRIMARY KEY (id),  CONSTRAINT fk7bwtucus42ys9kq6p19whqngp FOREIGN KEY (cliente\_id)  REFERENCES public.pessoa (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fkbipvru2vde7vahrtdsirf4iji FOREIGN KEY (filial\_id)  REFERENCES public.filial (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fkhwuy1ee4mf3migav7vcydr3a8 FOREIGN KEY (motorista\_id)  REFERENCES public.pessoa (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fkp524k6dcqixufjc7iygyhwm7a FOREIGN KEY (filialentrega\_id)  REFERENCES public.filial (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fkp7xg2cj93up7rldrxrbdqckt1 FOREIGN KEY (categoria\_id)  REFERENCES public.categoria (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION  ) |

|  |
| --- |
| **Reserva2** |
| CREATE TABLE public.reserva2  (  id integer NOT NULL,  datafinalprevista timestamp without time zone,  datainicial timestamp without time zone,  datamodificacao timestamp without time zone,  datareserva timestamp without time zone,  status character varying(255),  tipolocacao character varying(10),  valorprevisto double precision NOT NULL,  categoria\_id integer,  cliente\_id integer,  filial\_id integer,  filialentrega\_id integer,  motorista\_id integer,  CONSTRAINT reserva2\_pkey PRIMARY KEY (id),  CONSTRAINT fk16co3vdmveujljqbcelr8xdjf FOREIGN KEY (cliente\_id)  REFERENCES public.pessoa (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fk427wt66ae5pbaaaylfo4rn6u6 FOREIGN KEY (categoria\_id)  REFERENCES public.categoria (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fk4tjslrrx38unnjb11kmt6r194 FOREIGN KEY (filial\_id)  REFERENCES public.filial (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fkc9eyt6a4b7rnntyffuxntm8bp FOREIGN KEY (motorista\_id)  REFERENCES public.pessoa (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fkhkut26s17pwc9oepbg1m84i21 FOREIGN KEY (filialentrega\_id)  REFERENCES public.filial (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION  ) |

|  |
| --- |
| **ValorLocacao** |
| CREATE TABLE public.valorlocacao  (  id integer NOT NULL,  tipolocacao character varying(10),  valor double precision NOT NULL,  categoria\_id integer,  CONSTRAINT valorlocacao\_pkey PRIMARY KEY (id),  CONSTRAINT fk4xa974xsp3qs1wk22ya0oywyd FOREIGN KEY (categoria\_id)  REFERENCES public.categoria (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION  ) ON UPDATE NO ACTION ON DELETE NO ACTION  ) |

|  |
| --- |
| **Veiculo** |
| CREATE TABLE public.veiculo  (  id integer NOT NULL,  chassi character varying(255),  cor character varying(15),  kmanterior double precision NOT NULL,  kmatual double precision NOT NULL,  nummotor integer NOT NULL,  placa character varying(8),  status character varying(10),  categoria\_id integer,  filialatual\_id integer,  modelo\_id integer,  CONSTRAINT veiculo\_pkey PRIMARY KEY (id),  CONSTRAINT fk28ydec49vit7r5r3tt72any6x FOREIGN KEY (filialatual\_id)  REFERENCES public.filial (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fkijgiia7h7nqm8nq8p43qss4ju FOREIGN KEY (modelo\_id)  REFERENCES public.modelo (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT fksxqfq9928iouas1d17w48sk9 FOREIGN KEY (categoria\_id)  REFERENCES public.categoria (id) MATCH SIMPLE  ON UPDATE NO ACTION ON DELETE NO ACTION,  CONSTRAINT uk\_5vjrigbbme5uspvbbcjnecalc UNIQUE (placa),  CONSTRAINT uk\_63ouoyf532uqmpf4rmwmyd5yv UNIQUE (chassi)  ) |