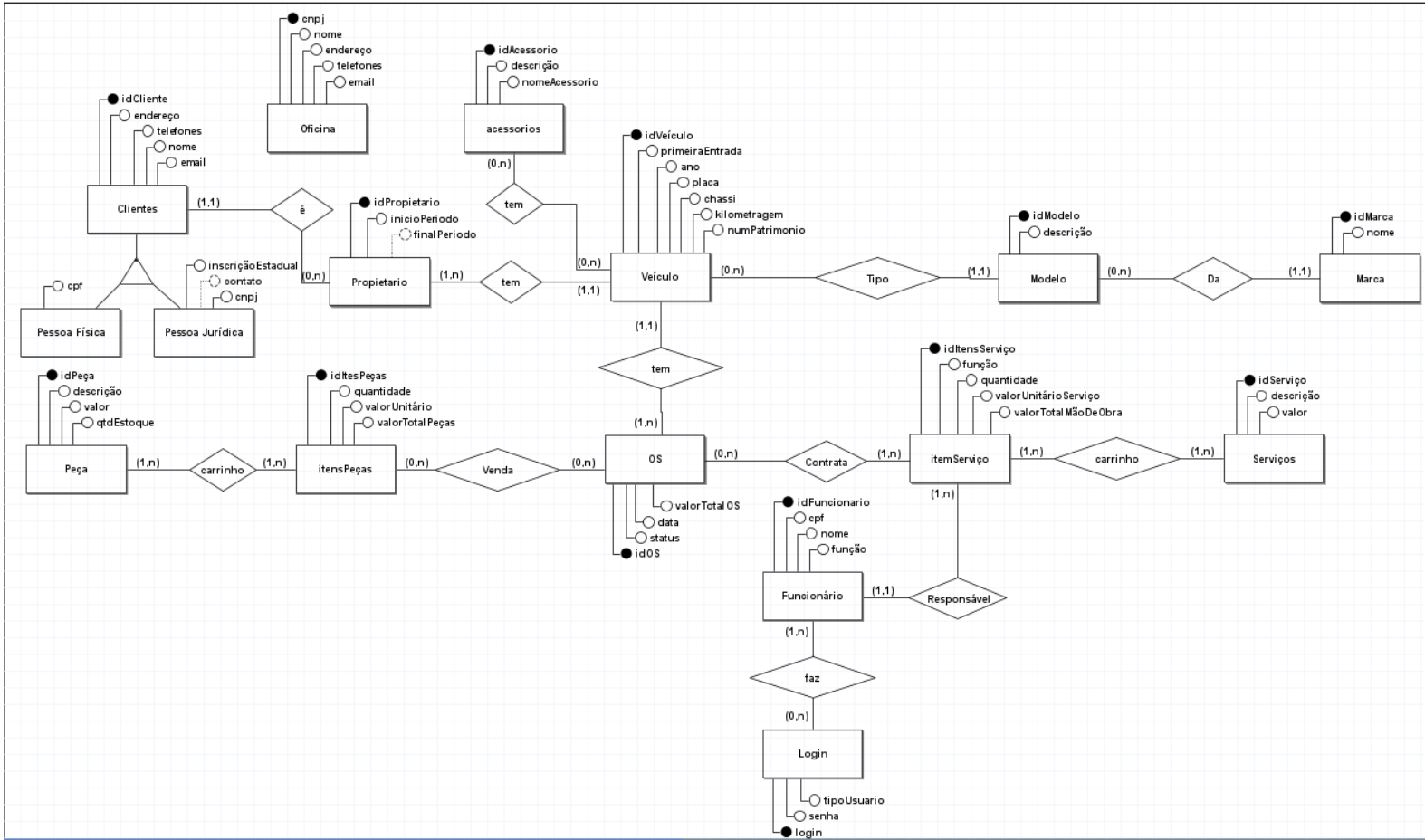


Modelo Conceitual – MER



Modelo Lógico

Oficina(cnpj, nome, endereco_logradouro, endereco_numero, telefones_ddi, telefones_ddd, telefones_numero, email)

Funcionario(idFuncionario, cpf, nome, funcao)

Clientes(idCliente, nome, endereco_logradouro, endereco_numero, telefones_ddi1, telefones_ddd1, telefones_numero1, telefones_ddi2, telefones_ddd2, telefones_numero2, email)

Login(login, senha, tipoUsuario, *idFuncionario*)
idFuncionario relaciona com **Funcionario**

PessoaFisica(cpf, *idCliente*)
idCliente se relaciona com **Clientes**

PessoaJuridica(cnpj, razaoSocial, inscricaoEstadual, *idCliente*)
idCliente se relaciona com **Clientes**

Acessorios(idAcessorio, descricao, nomeAcessorio)

Marca(idMarca, nome)

Modelo(idModelo, descricao, *idMarca*)
idMarca se relaciona com **Marca**

Veiculo(idVeiculo, primeiraEntrada, ano, placa, chassi, kilometragem, numPatrimonio, *idModelo*)
idModelo se relaciona com **Modelo**

VeiculoAcessorio(idVeiculo, idAcessorio)

idVeiculo se relaciona com **Veiculo**

idAcessorio se relaciona com **Acessorio**

Proprietario(idProprietario, inicioPeriodo, finalPeriodo, idVeiculo, idCliente)

idVeiculo se relaciona com **Veiculo**

idCliente se relaciona com **Clientes**

OrdemDeServiço(idOS, status, data, valorTotalOS, valorPago, desconto, idVeiculo)

idVeiculo se relaciona com **Veiculo**

Servicos(idServico, descricao, valor)

itemServico(idItemServico, quantidade, valorUnitarioServico, valorTotalMaoDeObra, idServico, idOS, idFuncionario)

idServico se relaciona com **Servicos**

idOS se relaciona com **OrdemDeServiço**

idFuncionario se relaciona com **Funcionario**

Peca(idPeca, descricao, valor, qtdEstoque)

itensPecas(idItensPecas, quantidade, valorUnitario, valorTotalPecas, idPeca, idOS)

idPeca se relaciona com **Peca**

idOS se relaciona com **OrdemDeServiço**

Modelo Físico

```
-- Tabelas independentes
CREATE TABLE Oficina (
  cnpj CHAR(14) PRIMARY KEY,
  nome VARCHAR(100) NOT NULL,
  endereco_logradouro VARCHAR(255) NOT NULL,
  endereco_numero VARCHAR(20) NOT NULL,
  telefones_ddi INT NOT NULL,
  telefones_ddd INT NOT NULL,
  telefones_numero VARCHAR(20) NOT NULL,
  email VARCHAR(100) NOT NULL UNIQUE
);

CREATE TABLE Funcionario (
  idFuncionario SERIAL PRIMARY KEY,
  cpf CHAR(11) UNIQUE NOT NULL,
  nome VARCHAR(100) NOT NULL,
  funcao VARCHAR(50) NOT NULL
);

CREATE TABLE Clientes (
  idCliente SERIAL PRIMARY KEY,
  nome VARCHAR(100) NOT NULL,
  endereco_logradouro VARCHAR(255) NOT NULL,
  endereco_numero VARCHAR(20) NOT NULL,
  telefones_ddi1 INT NOT NULL,
  telefones_ddd1 INT NOT NULL,
  telefones_numero1 VARCHAR(20) NOT NULL,
  UNIQUE (telefones_ddi1, telefones_ddd1, telefones_numero1),
  telefones_ddi2 INT NULL,
  telefones_ddd2 INT NULL,
  telefones_numero2 VARCHAR(20) NULL,
  UNIQUE (telefones_ddi2, telefones_ddd2, telefones_numero2),
  email VARCHAR(100) NOT NULL UNIQUE
);

CREATE TABLE Acessorios (
  idAcessorio SERIAL PRIMARY KEY,
  descricao TEXT NULL,
  nomeAcessorio VARCHAR(100) NOT NULL
);

CREATE TABLE Marca (
  idMarca SERIAL PRIMARY KEY,
  nome VARCHAR(50) NOT NULL
);

-- Tabelas relacionadas às tabelas independentes

CREATE TABLE Usuario (
  idUsuario SERIAL PRIMARY KEY,
  cpf CHAR(11) NOT NULL UNIQUE,
  tipoUsuario VARCHAR(20) NOT NULL CHECK (tipoUsuario IN
('Administrador', 'Funcionario')),
  dataCriacao TIMESTAMP DEFAULT CURRENT_TIMESTAMP, -- Data de
criação do registro
  FOREIGN KEY (cpf) REFERENCES Funcionario(cpf),
  idFuncionario INT NULL REFERENCES Funcionario(idFuncionario)
);

CREATE TABLE Login (
  login VARCHAR(20) PRIMARY KEY,
  senha VARCHAR(255) NOT NULL,
  tipoLogin VARCHAR(50) NOT NULL CHECK(tipoLogin IN ('Admin',
'Usuario')),
  idFuncionario INT NULL REFERENCES Funcionario(idFuncionario)
);

CREATE TABLE PessoaFisica (
  cpf CHAR(11),
  idCliente INT NOT NULL REFERENCES Clientes(idCliente)
);

CREATE TABLE PessoaJuridica (
  cnpj CHAR(14),
  razaoSocial VARCHAR(100) NOT NULL,
  inscricaoEstadual VARCHAR(20) NOT NULL,
  idCliente INT NOT NULL REFERENCES Clientes(idCliente)
);

CREATE TABLE Modelo (
  idModelo SERIAL PRIMARY KEY,
  descricao VARCHAR(100) NOT NULL,
  idMarca INT NOT NULL REFERENCES Marca(idMarca)
);

CREATE TABLE Veiculo (
  idVeiculo SERIAL PRIMARY KEY,
  primeiraEntrada DATE NOT NULL,
  ano INT NOT NULL,
  placa VARCHAR(10) UNIQUE NOT NULL,
  chassi VARCHAR(20) NOT NULL,
  kilometragem INT NOT NULL,
```

```

        numPatrimonio VARCHAR(20) NOT NULL,
        idModelo INT NOT NULL REFERENCES Modelo(idModelo)
    );

CREATE TABLE VeiculoAcessorio (
    idVeiculo INT NOT NULL REFERENCES Veiculo(idVeiculo),
    idAcessorio INT NOT NULL REFERENCES Acessorio(idAcessorio)
);

-- Tabelas dependentes de Veiculo e Clientes
CREATE TABLE Proprietario (
    idProprietario SERIAL PRIMARY KEY,
    inicioPeriodo DATE NOT NULL,
    finalPeriodo DATE NULL,
    idVeiculo INT NOT NULL REFERENCES Veiculo(idVeiculo),
    idCliente INT NOT NULL REFERENCES Clientes(idCliente)
);

-- Tabelas dependentes de Veiculo
CREATE TABLE OrdemDeServico (
    idOS SERIAL PRIMARY KEY,
    status VARCHAR(20) NOT NULL CHECK(status
IN('Orçamento', 'Aprovado', 'Finalizado', 'Em Andamento',
'Pago'))),
    data DATE NOT NULL,
    valorTotalIOS MONEY NOT NULL,
    valorPago MONEY NULL,
    desconto MONEY NULL,
    idVeiculo INT NOT NULL REFERENCES Veiculo(idVeiculo)
);

-- Tabelas associadas a OrdemDeServico
CREATE TABLE Servicos (
    idServico SERIAL PRIMARY KEY,
    descricao VARCHAR(100) NOT NULL,
    valor MONEY NOT NULL
);

CREATE TABLE itemServico (
    idItemServico SERIAL PRIMARY KEY,
    quantidade INT NOT NULL,
    valorUnitarioServico MONEY NOT NULL,
    valorTotalMaoDeObra MONEY NOT NULL,
    idServico INT NOT NULL UNIQUE REFERENCES Servicos(idServico),
    idOS INT NOT NULL UNIQUE REFERENCES OrdemDeServico(idOS),
    idFuncionario INT NULL REFERENCES Funcionario(idFuncionario)
);

-- Tabelas relacionadas a Peças
CREATE TABLE Peca (
    idPeca SERIAL PRIMARY KEY,
    descricao VARCHAR(100) NOT NULL,
    valor MONEY NOT NULL,
    qtdEstoque INT NOT NULL
);

CREATE TABLE itensPecas (
    idItensPecas SERIAL PRIMARY KEY,
    quantidade INT NOT NULL,
    valorUnitario MONEY NOT NULL,
    valorTotalPecas MONEY NOT NULL,
    idPeca INT NOT NULL UNIQUE REFERENCES Peca(idPeca),
    idOS INT NOT NULL UNIQUE REFERENCES OrdemDeServico(idOS)
);

```