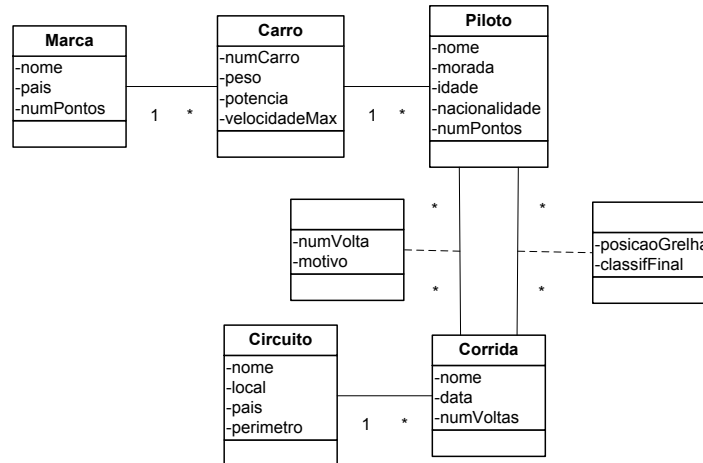


MODELAÇÃO CONCEPTUAL

1. a) Diagrama de classes UML



1. b) Esquema relacional

Marca(idMarca, nome, pais, numPontos)

Carro(idCarro, numCarro, peso, potencia, velocidadeMax, idMarca→Marca)

Piloto(idPiloto, nome, morada, idade, nacionalidade, numPontos, idCarro→Carro)

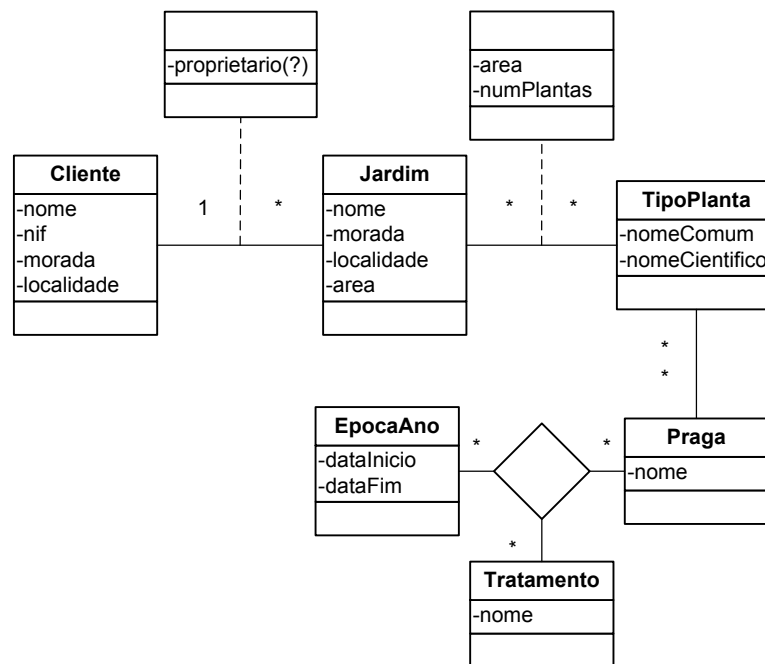
Circuito(idCircuito, nome, local, pais, perimetro)

Corrida(idCorrida, nome, data, numVoltas, idCircuito→Circuito)

Participacao(idPiloto→Piloto, idCorrida→Corrida, posicaoGrelha, classifFinal)

Desistencia(idPiloto→Piloto, idCorrida→Corrida, numVolta, motivo)

2. a) Diagrama de classes UML



2. b) Esquema relacional

Cliente(idCliente, nome, nif, morada, localidade)

Jardim(idJardim, nome, morada, localidade, area)

Proprietario(idCliente→Cliente, idJardim→Jardim, proprietario?)

TipoPlanta(idTipoPlanta, nomeComum, nomeCientifico)

JardimPlantas(idJardim→Jardim, idTipoPlanta→TipoPlanta, area, numPlantas)

Praga(idPraga, nome)

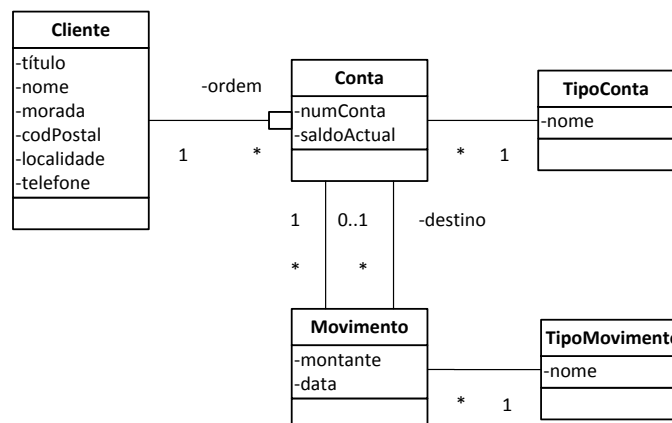
PragaTipoPlanta(idPraga→Praga, idTipoPlanta→TipoPlanta)

Tratamento(idTratamento, nome)

EpocaAno(idEpocaAno, dataInicio, dataFim)

PragaTratamentoEpoca(idPraga→Praga, idTratamento→Tratamento, idEpocaAno→EpocaAno)

3. a) Diagrama de classes UML



3. b) Esquema relacional

Cliente(idCliente, título, nome, morada, codPostal, localidade, telefone)

TipoConta(idTipoConta, nome)

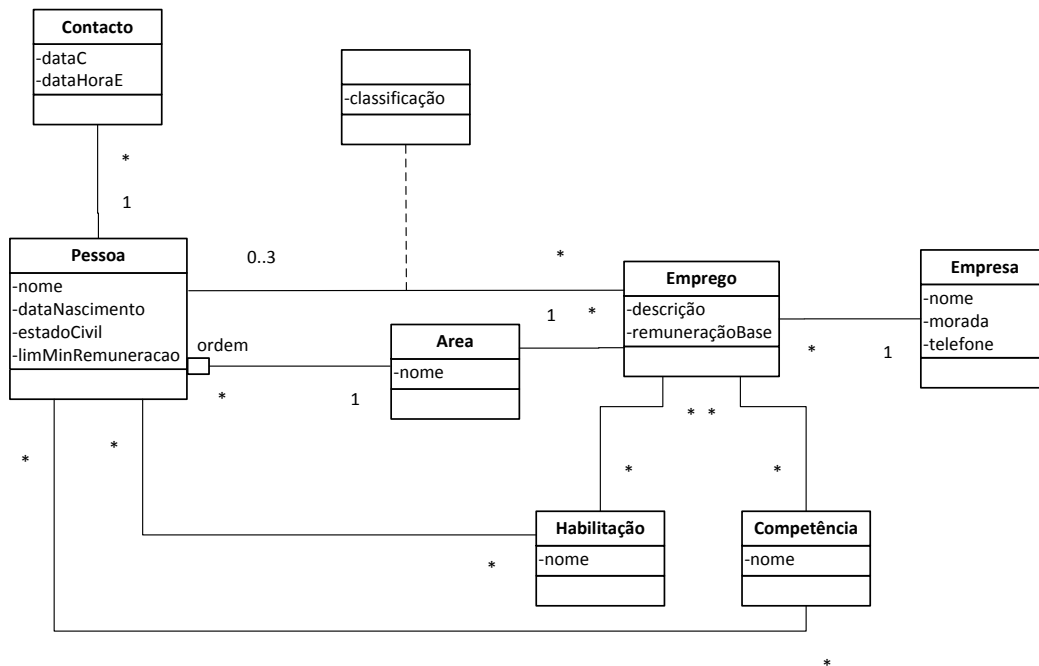
Conta(idConta, numConta, saldoActual, idTipoConta → TipoConta)

Titular(idConta → Conta, ordem, idCliente → Cliente)

TipoMovimento(idTipoMovimento, nome)

Movimento(idMovimento, montante, data, idTipoMovimento → TipoMovimento, origem → Conta, destino → Conta)

4. a) Diagrama de classes UML



4. b) Esquema relacional

Pessoa(idPessoa, nome, dataNascimento, estadoCivil, limMinRemuneracao)

Contacto(idContacto, dataC, dataHoraE, idPessoa→Pessoa)

Empresa(idEmpresa, nome, morada, telefone)

Area(idArea, nome)

PessoaArea(idPessoa→Pessoa, ordem, idArea→Area)

Emprego(idEmprego, descricao, remuneracaoBase, idEmpresa→Empresa, idArea→Area)

EmpregoPessoa(idEmprego→Emprego, idPessoa→Pessoa, classificacao)

Habilitacao(idHabilitacao, nome)

Competencia(idCompetencia, nome)

PessoaHabilitacao(idPessoa→Pessoa, idHabilitacao→Habilitacao)

PessoaCompetencia(idPessoa→Pessoa, idCompetencia→Competencia)

EmpregoHabilitacao(idEmprego→Emprego, idHabilitacao→Habilitacao)

EmpregoCompetencia(idEmprego→Emprego, idCompetencia→Competencia)

5. c) Esquema Relacional

Marca(idMarca, nome)

Modelo(idModelo, nome, idMarca → Marca)

CodPostal(codPostal1, localidade)

Cliente(idCliente, nome, morada, codPostal1 → CodPostal, codPostal2, telefone)

Carro(idCarro, matricula, idModelo → Modelo, idCliente → Cliente)

Reparacao(idReparacao, dataInicio, dataFim, idCliente → Cliente, idCarro → Carro)

Peca(idPeca, codigo, designacao, custoUnitario, quantidade)

ReparacaoPeca(idReparacao → Reparacao, idPeca → Peca, quantidade)

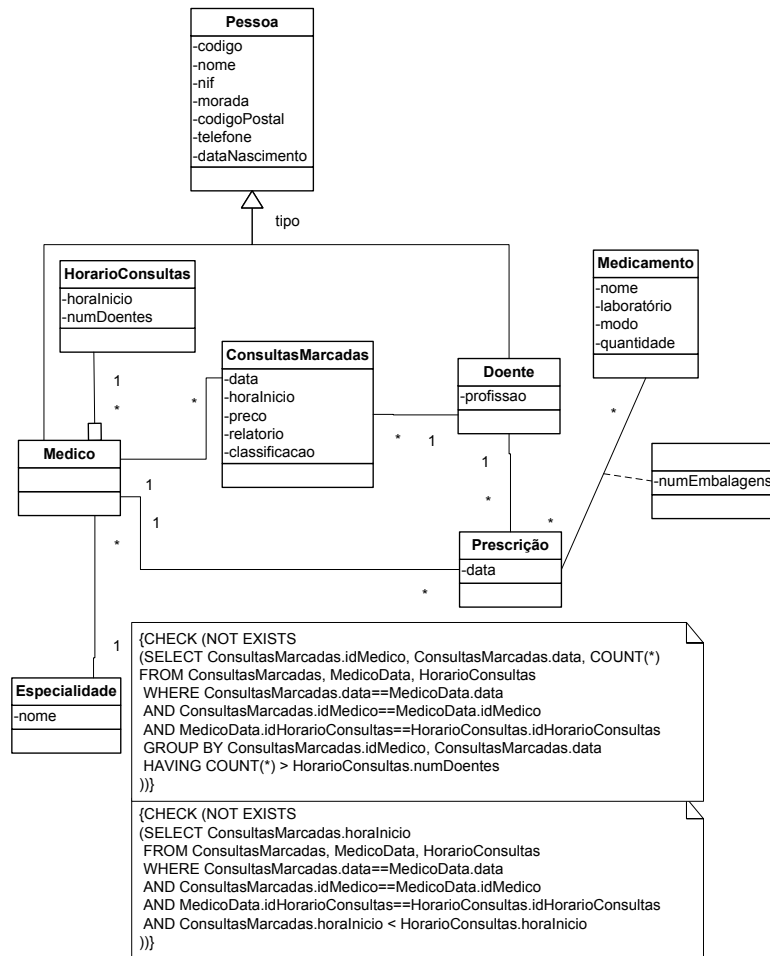
PecaModelo(idPeca → Peca, idModelo → Modelo)

Especialidade(idEspecialidade, nome, custoHorario)

Funcionario(idFuncionario, nome, morada, codPostal1 → CodPostal, codPostal2, telefone, idEspecialidade → Especialidade);

FuncionarioReparacao(idFuncionario → Funcionario, idReparacao → Reparacao, numHoras)

6. a) Diagrama de classes UML



6. b) Esquema relacional

Especialidade(idEspecialidade, nome)

Medico(idMedico, codigo, nome, nif, morada, codigoPostal, telefone, dataNascimento, idEspecialidade→Especialidade)

Doente(idDoente, codigo, nome, nif, morada, codigoPostal, telefone, dataNascimento)

HorarioConsulta(idHorarioConsulta, horaInicio, numDoentes)

MedicoConsulta(idMedico→Medico, data, idHorarioConsulta→HorarioConsulta)

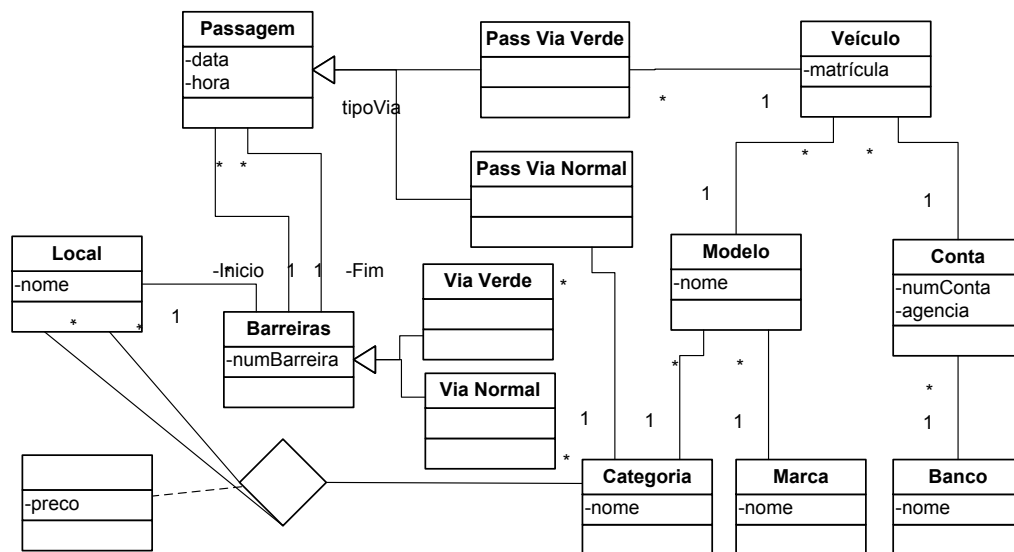
ConsultasMarcadas(idConsultaMarcada, data, horaInicio, preco, relatorio, classificacao, idMedico→Medico, idDoente→Doente)

Medicamento(idMedicamento, nome, laboratorio, modo, quantidade)

Prescricao(idPrescricao, data, idMedico→Medico, idDoente→Doente)

PrescricaoMedicamento(idPrescricao→Prescricao, idMedicamento→Medicamento, numEmbalagens)

7. a) Diagrama de classes UML



Só se devem definir os preços entre locais contíguos na auto-estrada para evitar ambiguidades.

Os registos das passagens são efectuados automaticamente através da detecção do sinal da via verde (Pass Via Verde) ou da indicação do portageiro (Pass Via Normal).

7. b) Esquema relacional

Local(idLocal, nome)

Barreiras(idBarreiras, numBarreira, tipoVia, idLocal→Local)

Categoria(idCategoria, nome)

Preco(idCategoria→Categoria, origem→Local, destino→Local, preco)

Marca(idMarca, nome)

Modelo(idModelo, nome, idMarca→Marca, idCategoria→Categoria)

Banco(idBanco, nome)

Conta(idConta, numConta, agencia, idBanco→Banco)

Veiculo(idVeiculo, matricula, idModelo→Modelo, idConta→Conta)

Passagem(idPassagem, data, hora, inicio→Barreira, fim→Barreira)

PassagemVerde(idPassagem→Passagem, idVeiculo→Veiculo)

PassagemNormal(idPassagem→Passagem, idCategoria→Categoria)

8. a) Diagrama de classes UML

