FACULDADE DE INFORMÁTICA E ADMINISTRAÇÃO PAULISTA

CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

HELP CAR

SOLUÇÃO PARA GESTÃO DE MANUTENÇÃO VEICULAR PROGRAMADA

SÃO PAULO

2024

HELLEN MARINHO CORDEIRO - RM 558841

JÚLIA SOARES FARIAS DOS SANTOS - RM 554609

GABRIEL DIAS MENEZES - RM 55019

HELP CAR

SOLUÇÃO PARA GESTÃO DE MANUTENÇÃO VEICULAR PROGRAMADA

SÃO PAULO

2024

SUMÁRIO

1. DESCRITIVO

1.1. SOLUÇÃO

A solução proposta tem como objetivo proporcionar aos clientes um atendimento de alta qualidade, tratando-se de uma proposta de gestão de serviços de manutenção com foco em melhorar a produtividade e qualidade, através de uma plataforma eficaz para gerenciar dados de veículos e agendamentos de serviços automotivos. O sistema, desenvolvido em Java utilizando Programação Orientada a Objetos (POO), assegura a organização e o gerenciamento de informações de clientes, veículos, seguros e reparos, permitindo uma operação escalável e eficiente.

Os dados de clientes, veículos e seguros são armazenados diretamente no banco de dados, garantindo a persistência das informações, enquanto os agendamentos são enviados para uma lista para organizar as marcações de serviços. Essa estrutura simplificada facilita o gerenciamento das informações e o controle das operações.

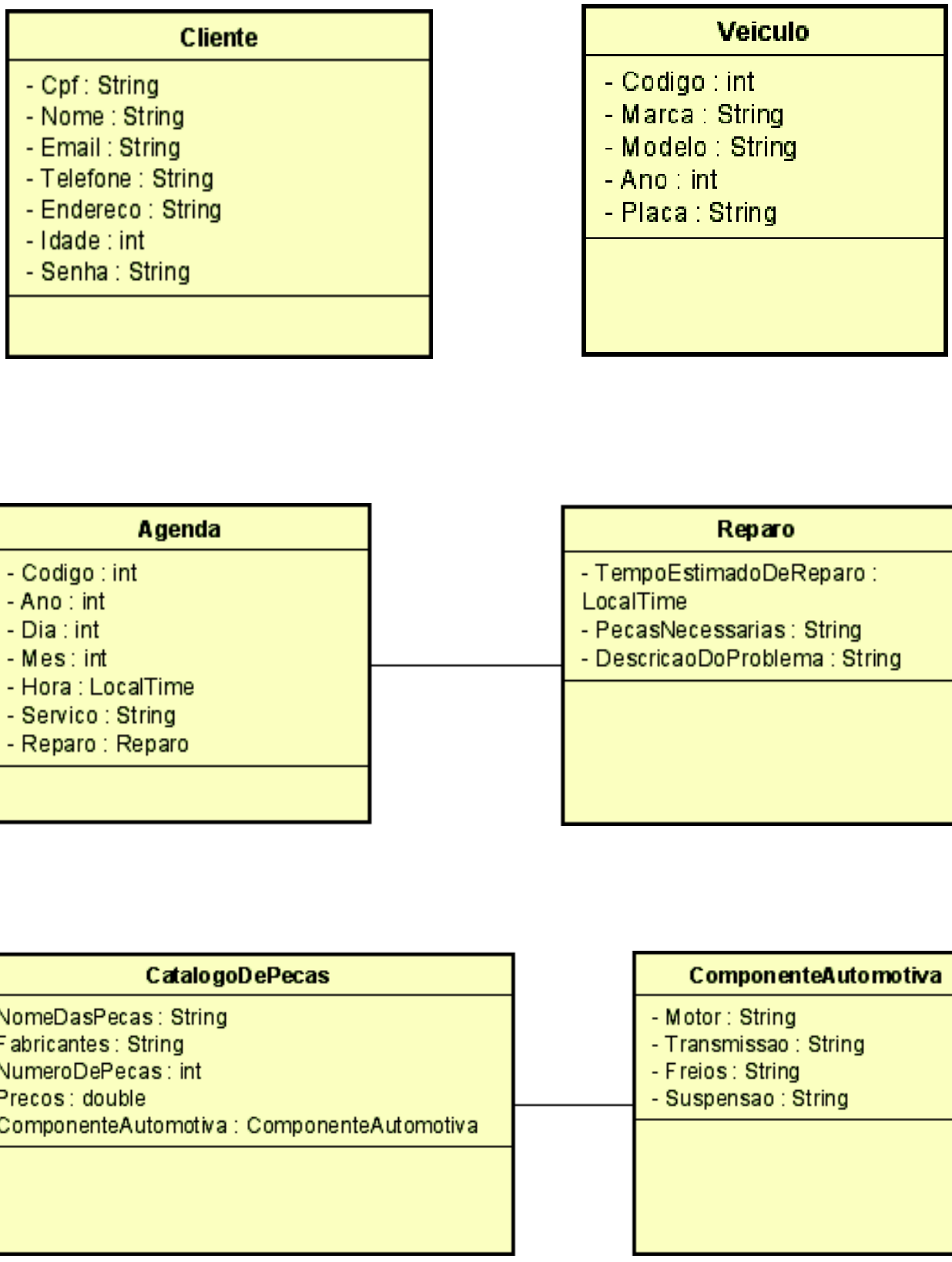
Embora não haja verificação de horários disponíveis, cotação detalhada ou controle de disponibilidade de peças, o foco do projeto está em oferecer uma interface intuitiva para organização das informações automotivas, permitindo uma gestão ágil e eficaz, que atenda às necessidades dos clientes com qualidade e eficiência.

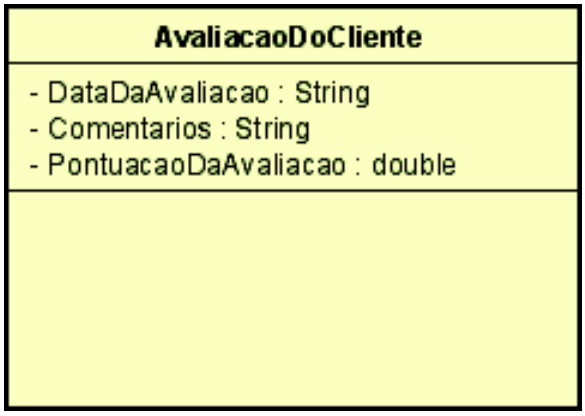
1.2. ALTERAÇÕES NO CÓDIGO

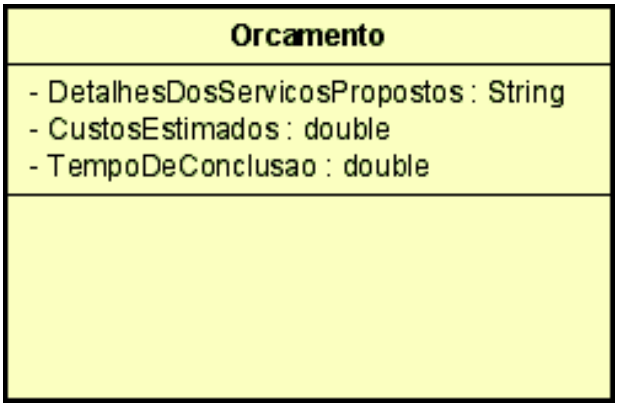
O projeto passou por diversas atualizações significativas para aprimorar sua integração com o banco de dados e a organização das informações. Uma das mudanças foi a adoção de LocalDate e LocalTime, que garantem a compatibilidade com os tipos de dados do banco. Além disso, novos atributos foram adicionados à classe Cliente, como o código (representando o ID do banco de dados), senha e email, aumentando a capacidade de gerenciamento dos dados dos usuários.

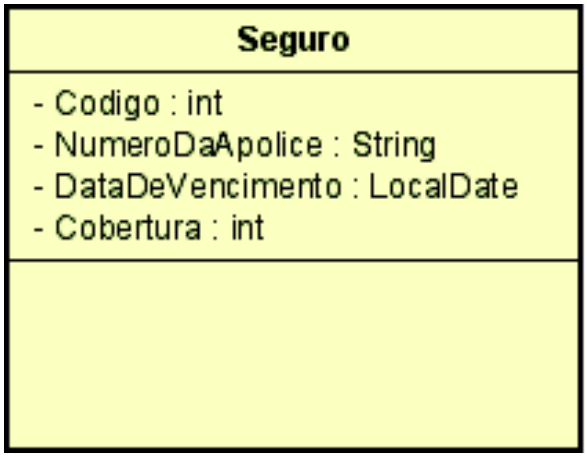
Outra melhoria foi a inserção da conexão com o banco de dados por meio do Conexão Factory, que centraliza e facilita a gestão da conexão de forma eficiente. Também foram implementadas as classes DAO (Data Access Object) e Teste Conexão, que permitem enviar e gerenciar as informações das classes para as respectivas tabelas no banco de dados. As classes DAO organizam a lógica de interação com o banco, permitindo a inserção, atualização e consulta de dados, enquanto a classe Teste Conexão garante que a comunicação entre o sistema e o banco esteja correta.

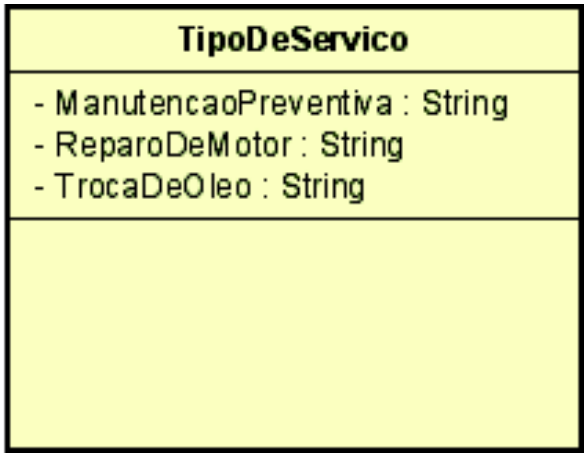
2. MODELAGEM DE CLASSES



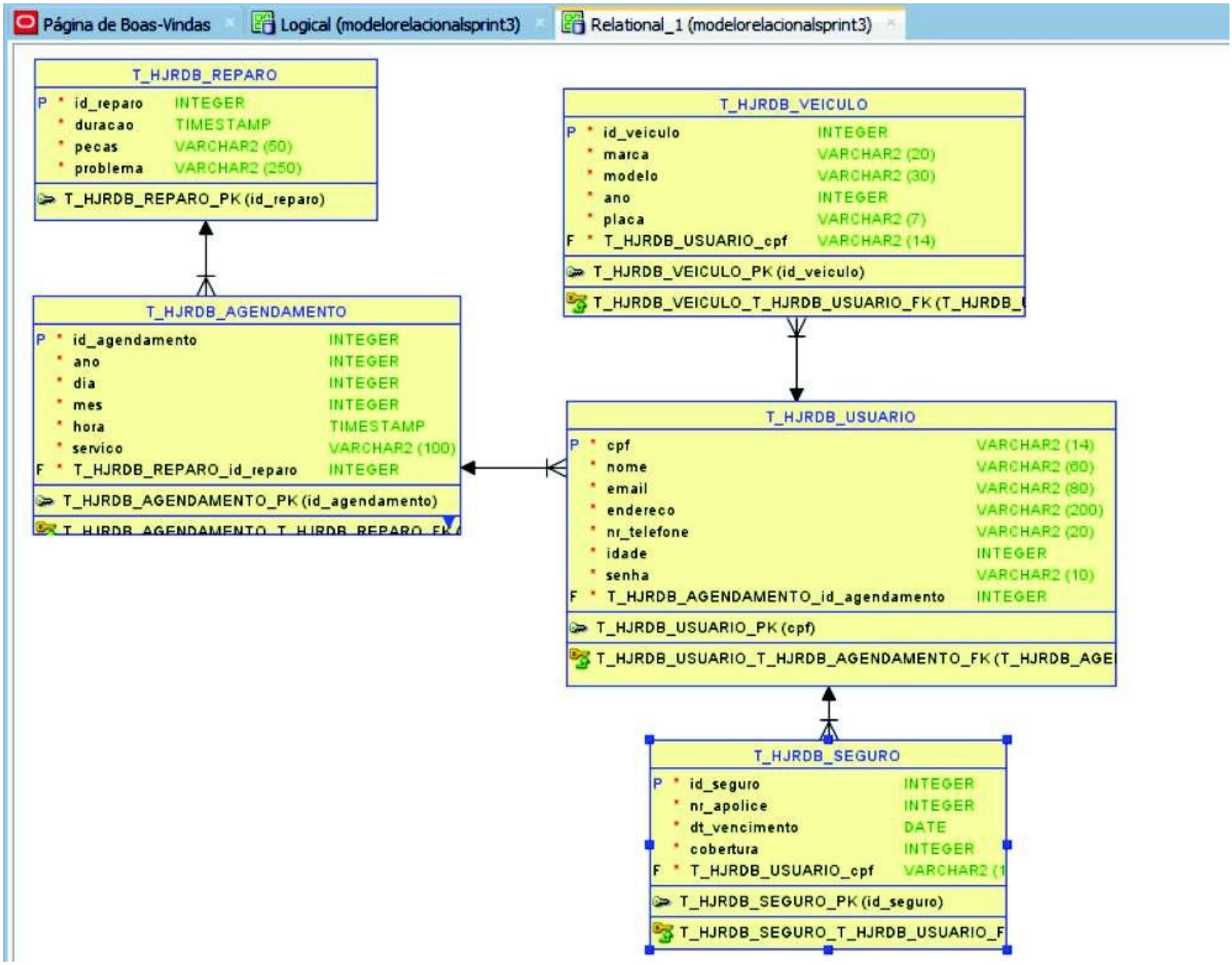

AvaliacaoDoCliente
- DataDaAvaliacao : String
- Comentarios : String
- PontuacaoDaAvaliacao : double


Orcamento
- DetalhesDosServicosPropostos : String
- CustosEstimados : double
- TempoDeConclusao : double


Seguro
- Codigo : int
- NumeroDaApolice : String
- DataDeVencimento : LocalDate
- Cobertura : int


TipoDeServico
- ManutencaoPreventiva : String
- ReparoDeMotor : String
- TrocaDeOleo : String

3. MODELO RELACIONAL - BANCO DE DADOS



4. PROCEDIMENTOS PARA RODAR A APLICAÇÃO

Procedimentos necessários para o funcionamento da aplicação:

- Conexão Factory
- Classes DAO
- Classes Teste

Os caminhos estabelecidos para os end points rodarem são seguintes:

- Cliente resource
- Veículo resource
- Agenda resource

A API que será utilizada no processo será a “VIA CEP”, a qual irá buscar as informações de logradouro, complemento, bairro, cidade e estado, concentram em um único objeto denominado “endereço”, o qual será enviado para o banco de dados.

Diagrama de Classes