



Relatório sobre o sistema de vendas Concessionária José Motors.

Docentes: *Matheus Gonçalves Ribeiro*

Discentes: *Flávia Narumi Nitto,
Sabrina de Oliveira Guizilini,
Sarah de Oliveira Ferreira,
Miguel Donizeti da Silva e Silva.*

Resumo

Foi solicitado que o grupo criasse um sistema de vendas para uma concessionária de automóveis chamada José Motors, com funcionalidades de cadastro, alteração e exclusão das classes Funcionário, Veículo, Carro, Motocicleta, Cliente, Venda, Data e Hora. Além disso, também foi pedido que houvesse um mecanismo de login a fim de identificar qual o tipo de funcionário (Gerente ou Vendedor). Depois, para melhor usabilidade do programa, foram criadas as classes Endereco, Operacoes, e Pagamento.

Nesse sentido, para possibilitar a recuperação dos dados mesmo após o fechamento do programa, utilizamos vários arquivos .txt com os dados delimitados pelo operador ; assim, no processo de recuperação dos dados usamos o método split para separar os dados.

Um ponto importante é que utilizamos o modelo MVC (Model View Controller) para a divisão dos pacotes, onde os arquivos .txt ficam salvos no Model, as interfaces gráficas no View e as classes java no pacote Controller.

Após amplas discussões entre os integrantes do grupo sobre como iríamos fazer o cadastro da venda, pois precisamos dela para fazer a contagem de quantas vendas foram realizadas por determinado vendedor e qual o tipo de veículo mais vendido, não conseguimos encontrar uma solução sem que a classe Venda fosse instanciada, por isso, optamos por instanciar a classe Venda mesmo sabendo que podemos perder pontos por isso.

Diagrama de Classe UML

O seguinte diagrama de classes foi usado para construir e implementar o sistema da concessionária, nele usamos os conceitos vistos tanto nas aulas de Programação Orientada a Objetos como nas aulas de Laboratório de Programação Orientada a Objetos, são eles: classes abstratas, métodos abstratos, herança, associações, relações de dependência e os conceitos de multiplicidade entre classes.

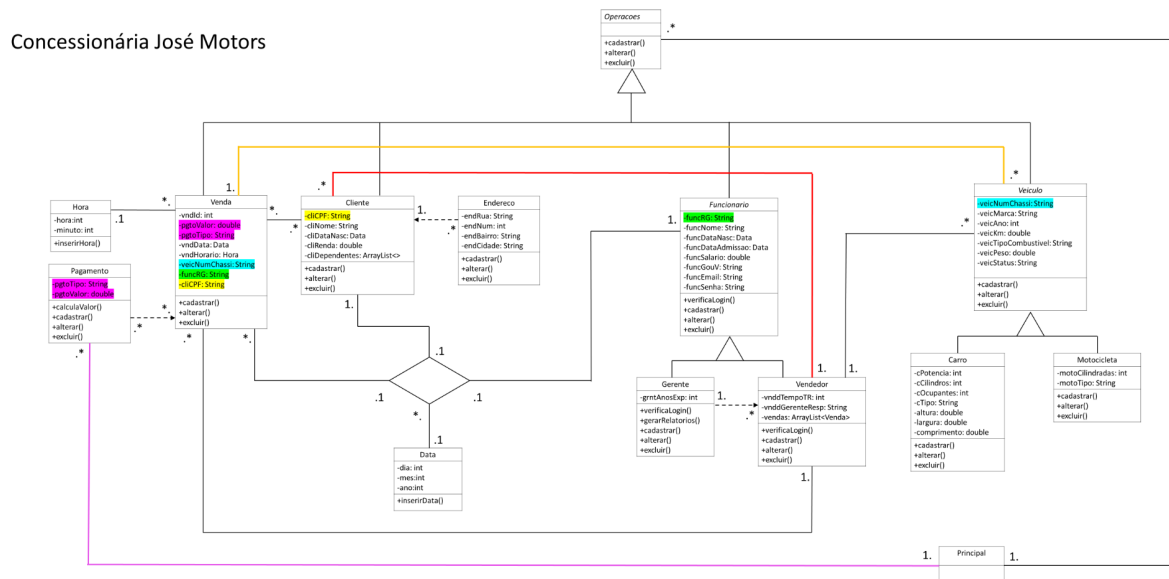
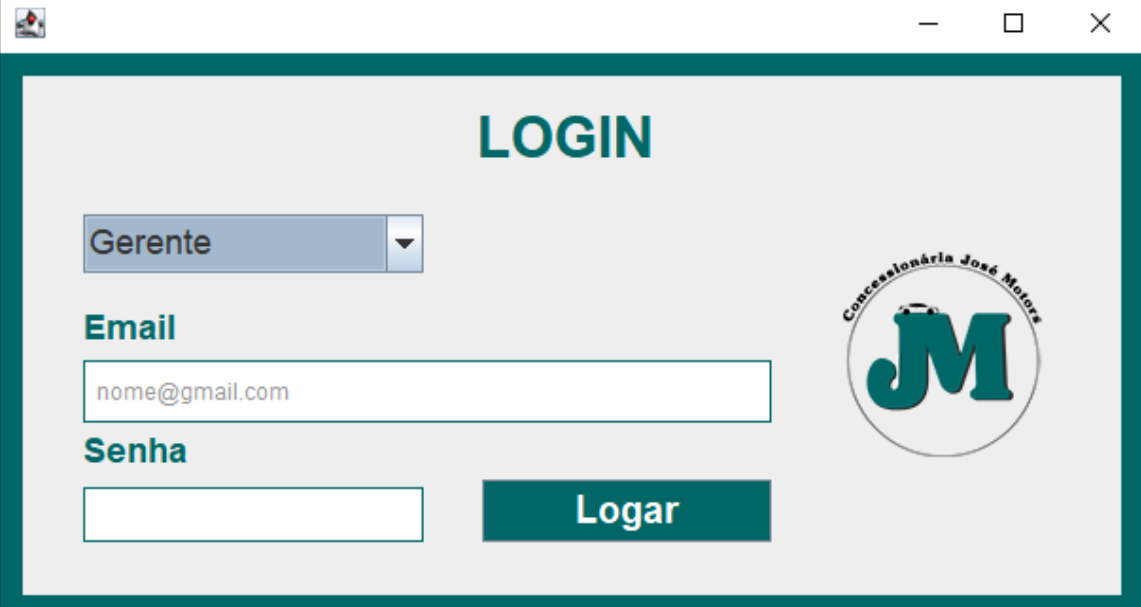


Figura 1: Diagrama de classe

Obs: Devido a baixa qualidade da imagem no relatório, o arquivo em .pdf estará anexado junto à tarefa no classroom.

Desenvolvimento

O sistema utiliza interfaces gráficas com diversas telas para cada etapa do processo, a Figura 2 representa a tela de login, onde podemos selecionar o tipo de funcionário (Gerente ou Vendedor) e em seguida inserir os dados cadastrais. Um ponto importante é que nosso sistema precisa iniciar com pelo menos um gerente cadastrado para então efetuar o cadastro de vendedores e ir complementando o sistema. Portanto, o projeto já contém um usuário e senha padrão cadastrados no arquivo (email: jose@gmail.com e senha: 123).



A imagem mostra uma janela de aplicativo com o título "LOGIN". No topo, há um menu suspenso com o texto "Gerente". Abaixo dele, há campos para "Email" (contendo "nome@gmail.com") e "Senha". Um botão verde com o texto "Logar" está à direita dos campos de entrada. No canto superior direito, há um logotipo circular com o texto "Concessionária José Motors" e uma grande letra "M" no centro.

Figura 2: Tela de Login

Após logar no sistema, é apresentado a tela inicial com as funcionalidades do sistema, note que se o funcionário logado for do tipo gerente, todos os botões estarão habilitados, já se for do tipo vendedor, apenas os botões para acesso de Veículos (Carro e Moto), Consultas e Vendas estarão habilitados.



Figura 3: Tela inicial para o funcionário do tipo gerente

Clicando no botão Carro, a seguinte janela será aberta. Para cadastrar um novo carro é necessário primeiro clicar no botão de Cadastrar e em seguida preencher os campos de dados. Note que quando clicamos no botão Cadastrar, os botões de Alterar e Excluir são automaticamente desabilitados, evitando que a operação de cadastro seja interrompida pela funcionalidade dos outros botões. Isso se replica para os dois outros botões também.

Design Preview [JFormCadastroCarro]

JM CARRO

Buscar pelo número do chassi

Número do chassi

Marca

Modelo

Tipo

Utilitário

Status

A venda

Ano

Km

Tipo de combustível

Peso

Potência

Número de cilindros

Número de ocupantes

Dimensões

Altura

Comprimento

Largura

Cadastrar **Alterar** **Excluir** **Confirmar** **Cancelar**

Figura 4: Tela de cadastro, alteração e exclusão de carros

Depois de preencher todos os campos, é necessário clicar no botão Confirmar para efetuar o cadastro no arquivo e então uma janela de informação mostra a mensagem de operação efetuada com sucesso.

Para efetuar a alteração ou exclusão de um carro, primeiro é preciso clicar no botão desejado e então digitar o número do chassi na barra superior e clicar no botão com o ícone de busca. Se o carro com o número de chassi digitado for encontrado, os campos serão preenchidos com os dados encontrados e para efetuar a operação de alteração ou exclusão é preciso clicar no botão confirmar.

Na tela de cadastro, alteração e exclusão de motocicletas, vendedores, gerentes, clientes e vendas, o processo é bem parecido com o que foi mostrado anteriormente na tela de cadastro, alteração e exclusão de carros.

Em seguida, temos o botão para verificar o desempenho dos vendedores do gerente logado. Em teoria, ao clicar no botão Gerar relatório, o programa acessa o arquivo desempenhos.txt e imprime as seguintes informações: RG do vendedor, quantidade de vendas realizadas, quantidade de vendas do tipo a vista, quantidade de vendas do tipo a prazo e qual o tipo de veículo mais vendido. Após verificar o relatório, é possível retornar ao menu principal.

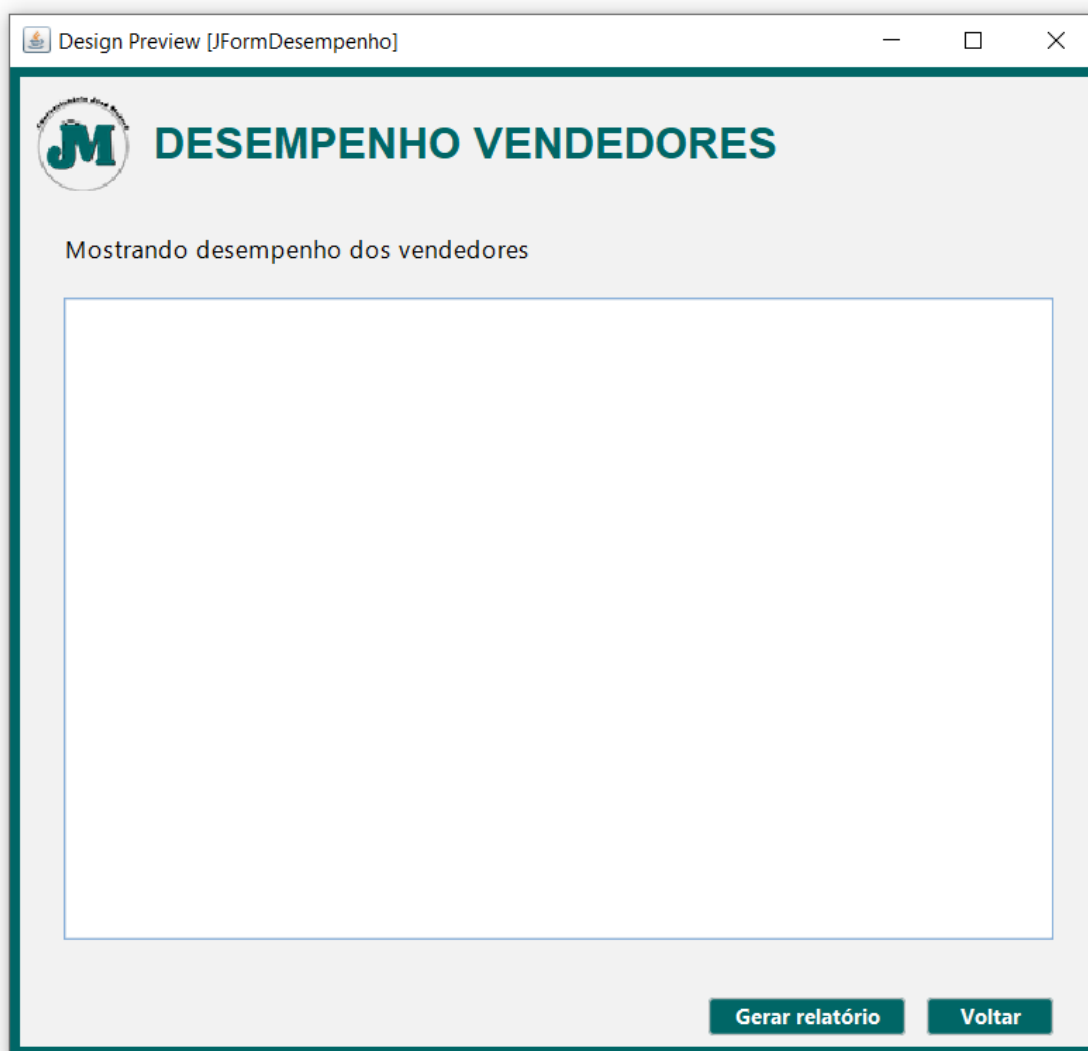


Figura 5: Tela de relatório de desempenho dos vendedores

Ao clicar no botão de consultar, a seguinte janela é aberta e temos 4 opções de consulta de dados: Carro, Moto, Cliente e Vendas. No exemplo mostrado na figura, temos a consulta das motos cadastradas no arquivo motocicletas.txt. Optamos por mostrar apenas alguns dados principais dentro do JTextArea. Logo em baixo da textArea, podemos retornar ao menu principal para usar as outras funcionalidades do sistema.



Figura 6: Tela de consulta de carros, motos, clientes e vendas

Conclusão

Concluimos que o trabalho englobou os assuntos das duas disciplinas (Programação Orientada a Objetos e Laboratório de Programação Orientada a Objetos) muito bem e contribuiu significativamente para o aprendizado e fixação dos conceitos aprendidos. No entanto, um grande desafio foi conseguir programar em conjunto com 4 pessoas e manter todos os integrantes com as versões atualizadas das telas de interface e dos códigos Java.