Relatório Grupo 1

Autor: João Pedro Santos Morais

## **Introdução**

Para o projeto de programação orientada a objeto, nossa equipe se propôs a desenvolver um sistema para solucionar um problema referente ao automobilismo e praticidade na vida das pessoas. Muitas vezes, proprietários de veículos necessitam realizar alguns serviços nos mesmos, que muitas vezes necessitam de um profissional competente para tal. Porém sabemos que, para contratar esses serviços, os proprietários precisam levar seus veículos até o estabelecimento onde o profissional está atuando. Algumas pessoas não se importam em fazer isso, mas outras se incomodam. Se alguém quer, por exemplo, fazer uma lavagem completa em seu carro para poder sair a noite, ela precisa levar seu carro a um lava jato. Mas talvez essa pessoa não tenha tempo ou disposição para fazer isso.

Visando resolver esse problema, idealizamos um sistema que tem por objetivo conectar prestadores de serviços para carros e possíveis clientes. Basicamente, é um programa em que uma pessoa pode entrar, se cadastrar, e então procurar pelo serviço que precisa para o seu veículo. Ao encontrá-lo, ela pode solicitá-lo, e então a empresa responsável pode aceitá-lo. Ao fazer isso, a empresa se propõe a enviar um profissional até a localização indicada pelo cliente para que ele realize o serviço lá mesmo, sem a necessidade de levar o carro até o seu estabelecimento. Os serviços são vários, como uma lavagem completa no veículo, troca de óleo, troca de pneu, instalação de insulfilm, entre outros. As empresas podem entrar nesse sistema e registrar os serviços que disponibiliza. Desse modo, a empresa aumentaria sua captação de clientes, e os clientes resolveriam seus eventuais problemas. O foco principal do sistema, seriam serviços de limpeza para veículos em geral, mas nada impede que outros serviços relacionados a carros sejam oferecidos.

## **Planejamento**

Pretendemos desenvolver a idea base para o projeto e uma visão de como ele funcionaria na prática. O objetivo da realização do projeto é didático, então não há pretensão de colocá-lo em prática na vida real. Portanto, alguns conceitos mais complexos não serão abordados.

Para a primeira parte do projeto, pretendemos desenvolver a ideia, e criar as possíveis telas que estarão no programa, e o seu funcionamento no geral. Elas basicamente ficaram separadas da seguinte maneira:

Geral

* Tela de Login

Lado do Cliente

* Tela de Registro
* Tela Menu Principal
* Tela Agendar Serviços
* Tela Ver Agendamentos
* Tela Editar Perfil

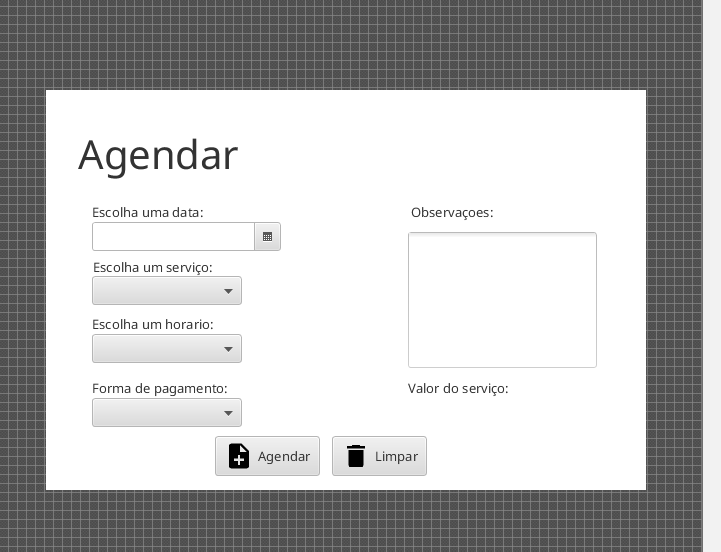
Lado da Empresa

* Tela de Registro
* Tela Menu Principal
* Tela Ver Agendamentos
* Tela Ver Solicitações de Agendamentos
* Tela Editar Perfil

Essas são apenas as telas iniciais que pretendemos colocar. Futuramente serão feitas as adições necessárias conforme o andamento do projeto. Essas telas foram divididas entre os membros da equipe, e cada um deve elaborá-las utilizando o Scene Builder com o JavaFX. Eu, João Pedro Santos Morais fiquei responsável pela criação das interfaces de menu principal e de prototipação do menu de agendamentos.

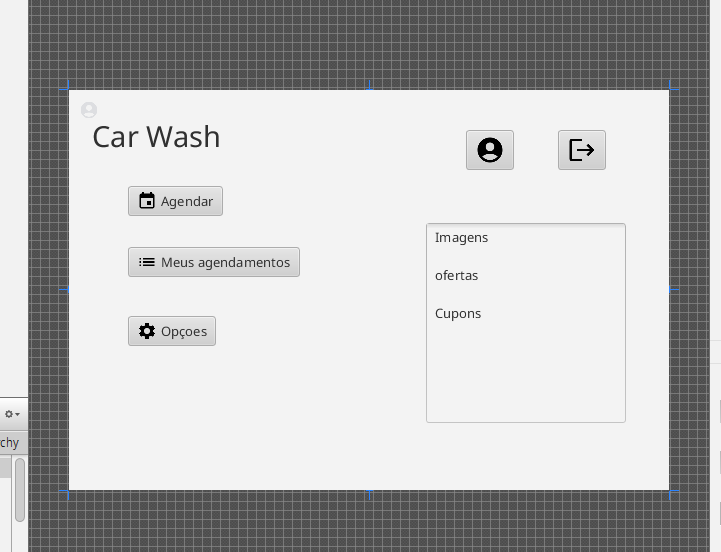
**Execução**

Foi criado, através do scene buider, a prototipação da tela de menu:



O menu de agendamento conta, inicialmente, com formulário somente para as escolhas de datas, horas, serviços e forma de pagamento pois, devido ao preenchimento inicial de dados no registro de usuário, é desnecessário o usuário informar novamente dados mais burocráticos, tais como endereço, cep, cpf, dentre outros.

Em segundo plano foi criado o protótipo do menu principal do usuário, no qual conta com as opçoẽs de agendar e ver os próprios agendamentos.



Do lado direito da imagem está destinado um campo para propagandas e ofertas no qual as próprias empresas poderão, com o devido pagamento, usar para se promoverem.

**Conclusão**

Todas as telas foram feitas com uso de javaFx no Scene Builder e com o auxílio de IDEs como o apache NetBeans.

A prototipação do projeto respeitou o modelo MVC requirido, estruturando o projeto em view (telas de menu, agendamento, perfil, dentre outras), controller, model,(Classes abstratas que vão ser herdadas pelas classes usuário e empresas, dentre outras).