#### MC302

#### Primeiro semestre de 2018

#### Laboratório 9

**Professora:** Esther Colombini (esther@ic.unicamp.br)

PEDs: Nathana Facion (nathanafacion@gmail.com), Rafael Tomazela (sohakes@gmail.com), Luis Fer-

nando Antonioli (luisfernandoantonioli@gmail.com)) **PAD:** Anderson Cotrim (ander.cotrim@gmail.com)

## 1 Descrição Geral

Neste laboratório, continuaremos com o desenvolvimento incremental do sistema de caronas on-line desenvolvido nos laboratórios anteriores.

# 2 Objetivo

Este laboratório tem como objetivo aprimorar o entendimento sobre os conceitos de interface gráfica e com isso finalizar o sistema de carona.

### 3 Atividade

Nessa atividades, faremos uma parte da interface gráfica necessária para o sistema, com algumas funcionalidades implementadas.

#### 3.1 Definições

Para um sistema com interface funcionar bem, devemos considerar que um usuário está "logado" no sistema. Toda a informação disponível na interface deve corresponder à visão de determinado usuário. Mas como a interface gráfica não contempla a criação de usuário, é necessário que esse usuário esteja pré-criado. Recomendamos que os objetos criados no main do laboratório anterior sejam reusados.

A forma de guardar e gerenciar a lista de usuários e grupos do sistema não será especificada. Uma ideia é criar uma classe que tem a função de gerenciar essas estruturas. Não recomendamos que isso seja feito na própria classe da janela, primeiramente porque essa classe é modificada automaticamente pelo eclipse (caso use o editor de janelas) e também por ser má ideia misturar o código da interface gráfica com a lógica de gerenciamento.

A forma de implementar e o layout das janelas pode ser feito como achar melhor. Mas há algumas funcionalidades que precisam estar presentes na interface gráfica:

- Pelo menos algumas informações do usuário e de seu perfil devem ser mostradas (o que achar necessário mostrar);
- Deve ser possível visualizar de maneira simplificada a lista de caronas que o usuário participa, tanto como caroneiro quanto como caronante;
- Deve ser possível criar uma carona;

- Deve ser possível visualizar os grupos que o usuário está;
- Deve ser possível criar um grupo;
- Deve ser possível sair de um grupo;
- A interface deve ter pelo menos duas visões ou janelas diferentes (duas ou mais abas, por exemplo).

#### 3.1.1 Criação de telas

As telas 1, 2 e 3 atendem os requisitos supracitados. O layout não precisa estar assim, e pode ter funcionalidades adicionais. Confiamos que você terá uma melhor noção de estética do que a pessoa que fez essas telas.

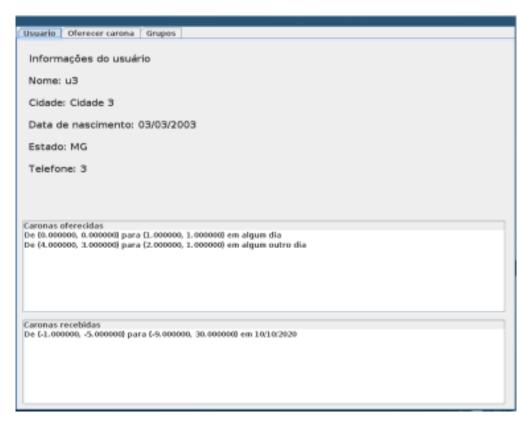


Figura 1: Informações de usuário e carona

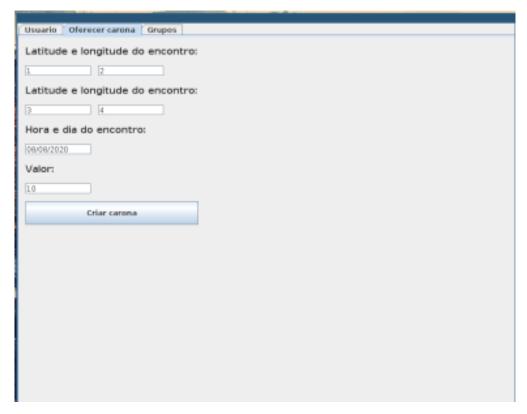


Figura 2: Criação de carona

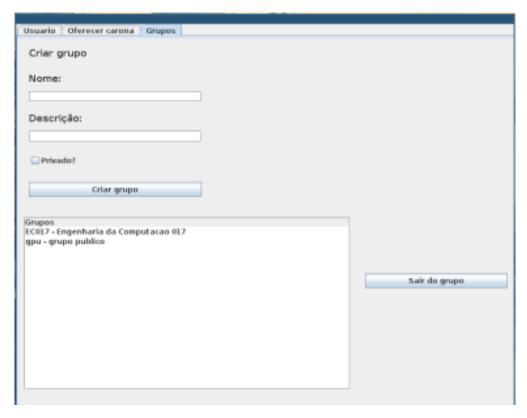


Figura 3: Gerenciamento de grupos

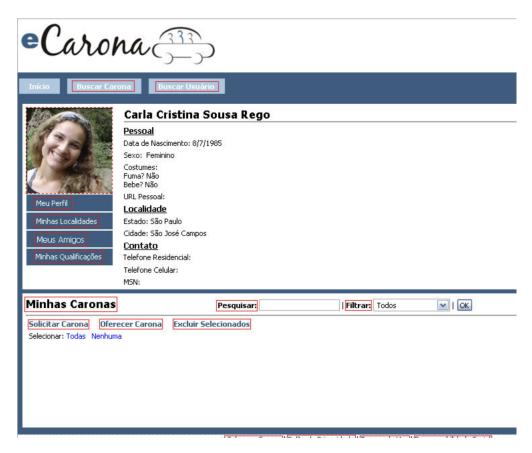


Figura 4: Exemplo de uma tela genrérica com disposição de elementos

#### **3.1.2** Testes

Verifique se é possível criar uma carona e se ela aparece na lista de caronas do usuário. Verifique também se é possível criar e sair de grupos, e se estes também atualizam a lista de grupos.

### 3.2 Observações

- Não peça, em nenhum momento, dados pela entrada padrão. Construa os objetos com valores inseridos diretamente no código ou pela interface gráfica;
- Envie os arquivos .java. Jamais envie somente os .class;
- Verifique se o código não tem nenhum warning antes de enviar;
- Deixe o código organizado e bem indentado. Use comentários quando achar necessário;
- Não coloque acentos no nome das classes. É possível que haja problemas na codificação durante o processo de compressão dos arquivos, e isso vai causar problemas para o compilador;
- Qualquer dúvida, sinta-se livre para pedir ajuda através de e-mail, nos laboratórios, ou na monitoria.

## 4 Questões

Adicione acima do Main as respostas das questões abaixo. É recomendado que antes de responder seja lido o material de sala de aula. Caso você realize o teste para alguma situação abaixo no código, o mesmo deve ficar comentado, pois não faz parte do sistema de Carona proposto.

- 1. Quais as principais diferenças entre Swing e AWT?
- 2. Qual a vantagem de se usar um RootPaneContainer?
- 3. Quando temos um JDialog visível o que ocorre com as outras janelas do sistema?
- 4. Explique as diferenças entre BorderLayout, BoxLayout, FlowLayout e GridLayout. Quais deles você usou na criação das janelas? Por quê?
- 5. Explique o que é uma função callback. Como são implementadas em Java? Não use código, explique com suas palavras.
- 6. Explique com suas palavras como cada nível do framework MVC seria tratado na sua implementação do sistema.

# 5 Requisitos do Sistema que devem estar implementados

- 1. Deve ser possível criar novos Usuários, Perfis, Caroneiros e Caronantes;
- 2. Deve ser possível associar Usuários com Perfis, e Perfis com Caroneiros e Caronantes;
- 3. Um Usuario deve poder criar um Grupo Publico ou um Grupo Privado;
- 4. Um Usuario dono de um grupo privado deve poder adicionar outros Usuarios em tal grupo.
- 5. Um Usuario deve poder se adicionar em um grupo publico
- 6. Um Caronante deve poder criar uma Carona Publica ou uma Carona Privada;
- 7. Um Caronante deve poder adicionar Grupos em Caronas;
- 8. Um Caroneiro deve poder pedir uma Carona existente, dado que este Caroneiro pertença a um Grupo associado à Carona, ou que seja uma Carona Publica sem Grupos;
- 9. Um Caroneiro ou um Caronante deve poder avaliar uma Carona ao qual está associado;
- 10. Deve ser possível calcular a nota/avaliação de um Perfil de Usuario, baseado nas avaliações dadas por outros Usuarios nas Caronas em que participou;
- 11. Deve ser possível ordenar os Perfis dos Usuarios de acordo com suas avaliações;
- 12. Deve ser possível salvar os dados das partes do sistema indicadas em arquivo.

# 6 Questão bônus

Como atividade bônus referente a este laboratório, espera-se que por meio de uma interface gráfica seja calculada a rota entre dois locais. Estes locais devem ser a origem e o destino, para isso use a API do Google Maps (https://bit.ly/2s7QY8c).

O bônus será aplicado sobre a média de laboratório e poderá valer até 0,5 pontos.

# 7 Submissão

Para submeter a atividade utilize o Moodle (https://www.ggte.unicamp.br/ea). Salve os arquivos dessa atividade em um arquivo comprimido no formato .tar.gz ou .zip e nomeie-o **Lab9-00000.zip** trocando '000000' pelo seu número de RA. Submeta o arquivo na seção correspondente para esse laboratório no moodle da disciplina MC302.

## Datas de entrega

• Dia 25 de Junho, Turma ABCD até às 23:55

### Informação importante

• A atividade correspondente ao laboratório 9 contará, para nota, como 2 unidades de laboratório.