

Trabalho Prático 4 de PM

Pesquisa Eleitoral

Douglas Rodrigues de Almeida

douglasralmeida@live.com

1. Introdução

O objetivo do trabalho foi a criação de um programa com interface gráfica para simular uma pesquisa eleitoral com a função de receber os dados da pesquisa, armazená-las em um repositório e exibi-las posteriormente.

Foram implementadas duas classes, Questionário e Repositório, para armazenar os questionários da pesquisa eleitoral, três classes para representar as janelas da interface gráfica e duas classes extras, Renderizador que é responsável por renderizar o questionário em uma lista de exibição e Aplicacao que é responsável por iniciar a aplicação.

2. Implementação

Enumerações

Alguns dados do questionário são representados por um conjunto finito de elementos. Para implementá-los na aplicação, foi utilizado o tipo de dado *enum* do Java. São eles:

- ◆ Sexo: Utilizado para representar o sexo do eleitor.
 - Constantes: NENHUM, FEMININO; MASCULINO;
- ◆ Escolaridade: Utilizado para representar a escolaridade do eleitor.
 - Constantes: NENHUM; SUPERIOR; MEDIO; FUNDAMENTAL; NAOESCOLARIZADO;
- ◆ Idade: Utilizado para representar a idade do eleitor.
 - Constantes: NENHUM; MENOSDE20; ENTRE20E30; ENTRE30E40; ENTRE40E50; MAISDE50;
- ◆ Renda: Utilizado para representar a renda do eleitor.
 - Constantes: NENHUM; MENOSDE3SM; ENTRE3E5SM; ACIMADE5SM;

A constante NENHUM é utilizada para representar a ausência de opção escolhida durante o preenchimento do formulário.

Classes Questionario e Repositorio

A classe Questionario representa um questionário da pesquisa eleitoral onde estão armazenados os dados do eleitor e o nome do seu candidato. A classe possui os seguintes campos:

- ◆ ID: armazena o ID do questionário no repositório.
- ◆ candidato: armazena o nome do candidato do eleitor.
- ◆ sexo: armazena o sexo do eleitor.
- ◆ escolaridade: armazena a escolaridade do eleitor.

- ◆ idade: armazena a idade do eleitor.
- ◆ renda: armazena a renda do eleitor.

Funções da classe Questionario:

- ◆ getID: Retorna o ID do questionário.
- ◆ getCandidato: Retorna o nome do candidato escolhido pelo eleitor.
- ◆ getEscolaridade: Retorna a escolaridade do eleitor.
- ◆ getIdade: Retorna a idade do eleitor.
- ◆ getRenda: Retorna a renda do eleitor.
- ◆ getSexo: Retorna o sexo do eleitor.
- ◆ setCandidato: Grava o nome do candidato escolhido pelo eleitor no questionário. Gera uma exceção caso o nome seja uma cadeia de caracteres vazia.
- ◆ setEscolaridade: Grava a escolaridade do eleitor no questionário. Gera uma exceção caso a escolaridade não tenha sido escolhida.
- ◆ setIdade: Grava a idade do eleitor no questionário. Gera uma exceção caso a idade não tenha sido escolhida.
- ◆ setRenda: Grava a renda do eleitor no questionário. Gera uma exceção caso a renda não tenha sido escolhida.
- ◆ setSexo: Grava o sexo do eleitor no questionário. Gera uma exceção caso o sexo não tenha sido escolhido.
- ◆ toString: Retorna uma representação em cadeia de caracteres do questionário e de seus campos.

A classe Repositorio representa todos os questionários preenchidos. Ela herda a classe *ArrayList* que utiliza um vetor para representar uma lista de objetos.

A classe possui apenas um único campo, *questionarios* que representa todos os questionários preenchidos. Ela possui as seguintes funções:

- ◆ adicionarQuestionario: Adiciona um questionário no repositório.
- ◆ getQuestionario: Retorna um questionário do repositório conforme o índice informado.
- ◆ getSize: Retorna a quantidade de questionários cadastrados no repositório.
- ◆ removerQuestionario: Remove um questionário do repositório conforme o índice informado.
- ◆ toString: Retorna uma representação em cadeia de caracteres de todos os questionários do repositório e de seus campos.

Foi implementada uma classe chamada *RenderizadorQuestionario*, herdada da interface *ListCellRenderer*, para renderizar a exibição de um questionário numa lista de questionários. A renderização é controlada pela função *getListCellRendererComponent*.

Janelas da aplicação

A aplicação possui uma interface gráfica baseada em janelas que a permite ser operada através de teclado e mouse.

A janela principal é representada pela classe *JanelaPrincipal*, filha da classe *JFrame*. Ela possui três botões, um para criar um questionário, outro para exibir todos os questionários e um para

encerrar a aplicação. Seu construtor exige a passagem do repositório da aplicação como um parâmetro.

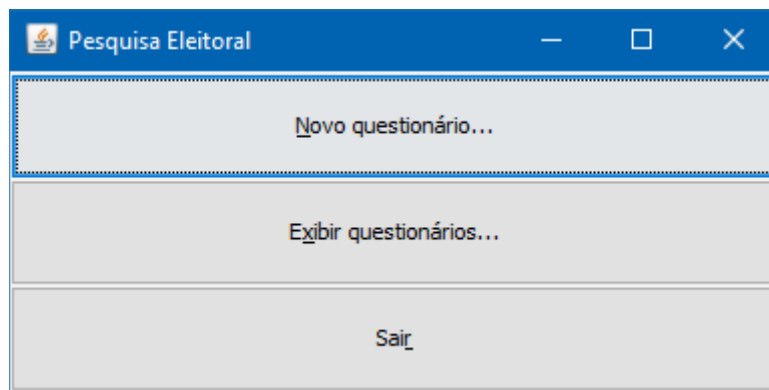


Figura 1: Janela principal da aplicação

A janela para criar um questionário é representada pela classe *JanelaNovoQuestionario*, filha da classe *JDialog*. A janela contém um campo de edição para informar o nome do candidato e botões de opção para escolher os dados do eleitor. No rodapé, há três botões: o primeiro salva o questionário no repositório, o segundo exibe os dados do questionário usando a função *toString()* da classe *Questionario* conforme a Figura 3 e o terceiro encerra a caixa de diálogo sem realizar nenhuma operação.

A classe possui tratamento para as exceções geradas no armazenamento dos dados na classe *Questionario* instanciada. Se os dados do questionário estiverem incompletos, será exibida uma mensagem de erro.

A janela 'Novo Questionário' possui uma barra de título azul com o texto 'Novo Questionário' e um botão de fechar. O conteúdo principal é um formulário com o título 'Pesquisa espontânea para Presidente'. O formulário contém os seguintes campos e opções: 'Nome do candidato:' com um campo de texto; 'Sexo do eleitor:' com botões de opção 'Feminino' e 'Masculino'; 'Nível de escolaridade:' com botões de opção 'Superior', 'Ens. Médio', 'Ens. Fundamental' e 'Não escolarizado'; 'Idade do eleitor:' com botões de opção 'Menos de 20 anos', 'Entre 20 e 30 anos', 'Entre 30 e 40 anos', 'Entre 40 e 50 anos' e 'Mais de 50 anos'; e 'Renda familiar do eleitor:' com botões de opção 'Menos de 3 salários mínimos', 'Entre 3 e 5 salários mínimos' e 'Mais de 5 salários mínimos'. No rodapé, há três botões: 'Salvar', 'Imprimir' e 'Cancelar'.

Figura 2: Janela para criar um novo questionário

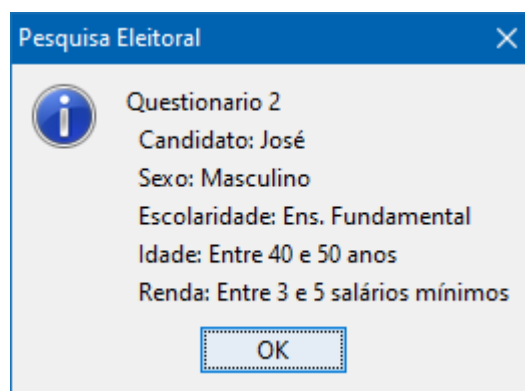


Figura 3: Questionário exibido

A janela para exibir os questionários também é derivada da classe *JDialog*. Os questionários são exibidos em uma lista da classe *JList* que utiliza a classe *RenderizadorQuestionario* para renderizar cada questionário. Ao dar um clique duplo em um questionário da lista, ele é exibido usando o mesmo método da Figura 3.

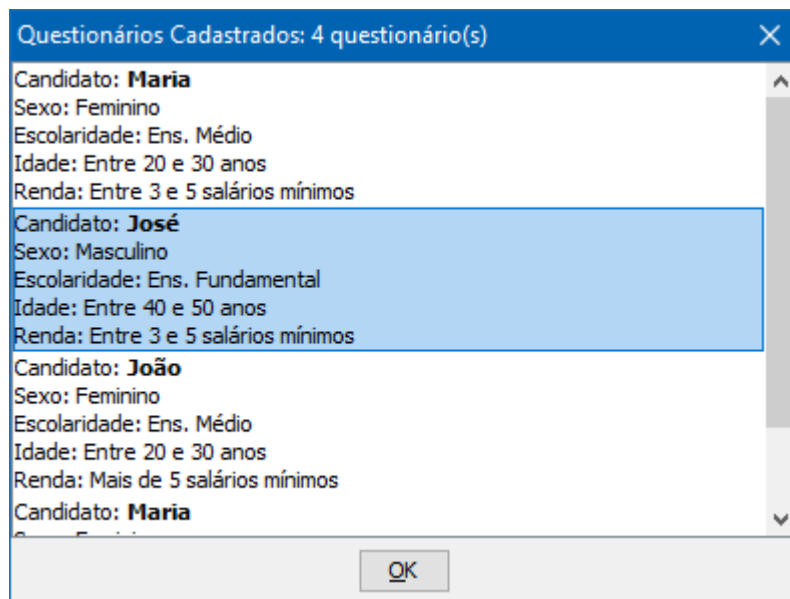


Figura 4: Janela para exibir todos os questionários

3. Diagrama de Classes

