 Garrocho / SIGE

Public

<> Code

Issues

Pull requests 1

Actions


Projects

Wiki

Security


In...

Commit

 This commit does not belong to any branch on this repository, and may belong to a fork outside of the repository.

Extract Method - 14

Browse files

 Ruan-Pablo committed on Jun 27

1 parent c836999 commit c7c2efd

Showing 1 changed file with 128 additions and 112 deletions.

Whitespace

Ignore whitespace

Split

Unified

▼

240

source/src/com/sige/gui/candidato/DialogoCadastrarCandidato.java



61

61

TratadorEventosCadastroCandidato tratadorEventos = new

62

62

TratadorEventosCadastroCandidato(this);

63

63

NumeroAlterar = Integer.parseInt(numero);

64

63

cargoAlterar = cargo;

65

-

private static final int NUMERO_DE_LINHAS = 4;

66

-

private static final int TAMANHO_MAXIMO_NOME = 25;

67

-

private static final int TAMANHO_MAXIMO_DOCUMENTO = 30;

68

-

private static final int TAMANHO_CAMPO_CARGO = 21;

69

64

65

+

final int NUMERO_DE_LINHAS = 4;

66

+

final int TAMANHO_MAXIMO_NOME = 25;

67

+

final int TAMANHO_MAXIMO_DOCUMENTO = 30;

68

+

final int TAMANHO_CAMPO_CARGO = 21;

70

69

71

-

JPanel painelNorte = new JPanel(new GridLayout(NUMERO_DE_LINHAS,0));

70

+

JPanel painelNorte = criarPainelNorte(numero, nome, partido, cargo,

71

+

tratadorEventos, NUMERO_DE_LINHAS, TAMANHO_MAXIMO_NOME,

+

TAMANHO_MAXIMO_DOCUMENTO, TAMANHO_CAMPO_CARGO);

72

72

73

-

73

+

JPanel painelCentro = criarPainelCentro(caminhoFoto);

74

74

75

-

JPanel painelNome = new JPanel();

76

-

fieldName = new JTextField(TAMANHO_MAXIMO_NOME);

77

-

fieldName.setToolTipText("Descreva Aqui o Nome Completo do Candidato.");

78

-

fieldName.setDocument(new TamanhoMaximo(TAMANHO_MAXIMO_DOCUMENTO);

79

-

painelNome.add(new JLabel("Nome "));

```

80      -                painelNome.add(fieldNome);
      75      +                JPanel painelBotoes = criarPainelBotoes(tratadorEventos);
81      76
82      -                // Adiciona o painelNome e todo seu conteudo ao painelNorte.
      77      +                configurarDialogo(painelNorte, painelCentro, painelBotoes);
      78      +
      79      +                pack();
      80      +
      81      +                setIconImage(Toolkit.getDefaultToolkit().getImage(getClass().getResource("/icones/icone.png")));
      82      +                setLocationRelativeTo(null);
      83      +                setModal(true);
      84      +                setDefaultCloseOperation(WindowConstants.DISPOSE_ON_CLOSE);
      85      +                setResizable(false);
      86      +                setVisible(true);
      87      +                }
      88      +                private JPanel criarPainelNorte(String numero, String nome, String partido,
      89      +                String cargo,
      90      +                TratadorEventosCadastroCandidato tratadorEventos,
      91      +                int numLinhas, int
      92      +                tamMaxNome, int tamMaxDocumento, int tamCampoCargo) {
      93      +                JPanel painelNorte = new JPanel(new GridLayout(numLinhas, 0));
      94      +                JPanel painelNome = criarPainelCampo("Nome", nome, tamMaxNome);
      95      +                painelNorte.add(painelNome);
83      96      +                JPanel painelCargo = criarPainelCampoCargo(cargo, tamCampoCargo,
84      97      +                tratadorEventos);
      98      +                painelNorte.add(painelCargo);
      99      +                JPanel painelPartido = criarPainelCampoPartido(partido, tratadorEventos);
100      +                painelNorte.add(painelPartido);
101      +
102      +                JPanel painelNumero = criarPainelCampoNumero(numero);
103      +                painelNorte.add(painelNumero);
104      +
105      +                return painelNorte;
106      +                }
107      +
108      +                private JPanel criarPainelCampo(String labelCampo, String valorInicial, int
      109      +                tamanhoMaximo) {
      110      +                JPanel painelCampo = new JPanel();
      111      +                JTextField textField = new JTextField(tamanhoMaximo);
      112      +                textField.setToolTipText("Descreva aqui o " + labelCampo + " Completo do
      113      +                Candidato.");
      114      +                textField.setDocument(new TamanhoMaximo(tamanhoMaximo));
      115      +                textField.setText(valorInicial);
      116      +                painelCampo.add(new JLabel(labelCampo + "  "));
      117      +                painelCampo.add(textField);
      118      +                return painelCampo;
      119      +                }

```

```

119 +         private JPanel criarPainelCampoCargo(String cargo, int tamanhoCampoCargo,
120 +
TratadorEventosCadastroCandidato tratadorEventos) {
85 121             JPanel painelCargo = new JPanel();
86 122             cargoField = new JTextField(TAMANHO_CAMPO_CARGO);
122 +             cargoField = new JTextField(tamanhoCampoCargo);
87 123             cargoField.setEditable(false);
88 124             botaoPesquisaCargo = new JButton();
89 125             botaoPesquisaCargo.setIcon(new
ImageIcon(getClass().getResource("/icones/pesquisar.png")));
90 126             botaoPesquisaCargo.setPreferredSize(new Dimension(30,30));
91 127             botaoPesquisaCargo.setCursor(new
java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
92 128             botaoPesquisaCargo.setToolTipText("Clique aqui para selecionar um
cargo.");
124 +             botaoPesquisaCargo = criarBotaoPesquisa("/icones/pesquisar.png", "Clique
aqui para selecionar um cargo", tratadorEventos);
93 125             painelCargo.add(new JLabel(" Cargo "));
94 126             painelCargo.add(cargoField);
95 127             painelCargo.add(botaoPesquisaCargo);
96 128             botaoPesquisaCargo.addActionListener(new ActionListener() {
97 129                 public void actionPerformed(ActionEvent arg0) {
98 130                     new LookUpCargo(getThis(),getThis().getCargoField());
99 131                 }
100 132             });
101 133
102 134             // Adiciona o painelCargo e todo seu conteudo ao painelNorte.
103 135             painelNorte.add(painelCargo);
128 +             return painelCargo;
129 +         }
104 130
131 +         private JPanel criarPainelCampoPartido(String partido,
TratadorEventosCadastroCandidato tratadorEventos) {
105 132             JPanel painelPartido = new JPanel();
106 133             partidoField = new JTextField(21);
107 134             partidoField.setEditable(false);
108 135             botaoPesquisaPartido = new JButton();
109 136             botaoPesquisaPartido.setIcon(new
ImageIcon(getClass().getResource("/icones/pesquisar.png")));
110 137             botaoPesquisaPartido.setPreferredSize(new Dimension(30,30));
111 138             botaoPesquisaPartido.setCursor(new
java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
112 139             botaoPesquisaPartido.setToolTipText("Clique aqui para selecionar um
cargo.");
135 +             botaoPesquisaPartido = criarBotaoPesquisa("/icones/pesquisar.png",
"Clique aqui para selecionar um partido", tratadorEventos);
113 140             painelPartido.add(new JLabel(" Partido "));
114 141             painelPartido.add(partidoField);
115 142             painelPartido.add(botaoPesquisaPartido);
116 143             botaoPesquisaPartido.addActionListener(new ActionListener() {
117 144                 public void actionPerformed(ActionEvent arg0) {
118 145                     new LookUpPartido(getThis(),getThis().getPartidoField());
119 146                 }
120 147             });

```

```

121 -
122 - // Adiciona o painelPartido e todo seu conteudo ao painelNorte.
123 - painelNorte.add(painelPartido);
139 + return painelPartido;
140 + }
124 141
142 + private JPanel criarPainelCampoNumero(String numero) {
125 143     JPanel painelNumero = new JPanel();
126 144     fieldNumero = new JTextField(13);
127 - fieldNumero.setToolTipText("Descreva Aqui o Numero de Eleio do
    Candidato.");
145 + fieldNumero.setToolTipText("Descreva aqui o Número de Eleição do
    Candidato.");
128 146     fieldNumero.setDocument(new SomenteNumeros());
129 -
130 -     botaoAbrirImagem = new JButton("FOTO", new
    ImageIcon(getClass().getResource("/icones/abrir.png")));
147 +     fieldNumero.setText(numero);
148 +     botaoAbrirImagem = criarBotaoAbrirImagem();
149 +     painelNumero.add(new JLabel("Número"));
150 +     painelNumero.add(fieldNumero);
151 +     painelNumero.add(botaoAbrirImagem);
152 +     return painelNumero;
153 + }
154 +
155 + private JButton criarBotaoPesquisa(String iconePath, String tooltip,
    TratadorEventosCadastroCandidato tratadorEventos) {
156 +     JButton botaoPesquisa = new JButton();
157 +     botaoPesquisa.setIcon(new ImageIcon(getClass().getResource(iconePath)));
158 +     botaoPesquisa.setPreferredSize(new Dimension(30, 30));
159 +     botaoPesquisa.setCursor(new
    java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
160 +     botaoPesquisa.setToolTipText(tooltip);
161 +     botaoPesquisa.addActionListener(tratadorEventos);
162 +     return botaoPesquisa;
163 + }
164 +
165 + private JButton criarBotaoAbrirImagem() {
166 +     JButton botaoAbrirImagem = new JButton("FOTO", new
    ImageIcon(getClass().getResource("/icones/abrir.png")));
131 167     botaoAbrirImagem.setFont(new Font(Font.SANS_SERIF, Font.BOLD, 14));
132 168     botaoAbrirImagem.setCursor(new
    java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
133 -     botaoAbrirImagem.setToolTipText("Selecione Aqui a Foto do Candidato.");
169 +     botaoAbrirImagem.setToolTipText("Selecione aqui a foto do candidato.");
134 170     botaoAbrirImagem.addMouseListener(new MouseAdapter() {
135 171         public void mouseClicked(MouseEvent arg0) {
136 -             String caminhoFoto =
    dialogoAbrirArquivo(getThis(), "Selecionar Foto");
172 +             String caminhoFoto = dialogoAbrirArquivo(getThis(), _
    "Selecionar Foto");
137 173             if (caminhoFoto != null) {
138 174                 addFotoCandidato(caminhoFoto);
139 175             }

```

```

140     176         }
141     177     });
142
143     -
144     -     painelNumero.add(new JLabel("Numero"));
145     -     painelNumero.add(fieldNumero);
146     -     painelNumero.add(botaoAbrirImagem);
147     -     painelNorte.add(painelNumero);
148     -
149     -     // Adiciona o painelNorte no centro do DialogoCadastrarCandidato.
150     -     add(painelNorte, BorderLayout.NORTH);
151     178     +     return botaoAbrirImagem;
152     179     + }
153
154     180
155     181     +     private JPanel criarPainelCentro(String caminhoFoto) {
156     182     -     JPanel painelCentro = new JPanel();
157     -
158     -     JPanel painelBotoes = new JPanel();
159     -     botaoGravar = new JButton("GRAVAR", new
160     ImageIcon(getClass().getResource("/icones/gravar.png")));
161     -     botaoGravar.setFont(new Font(Font.SANS_SERIF, Font.BOLD, 14));
162     -     botaoGravar.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
163     -     botaoGravar.addActionListener(tratadorEventos);
164     -
165     -     painelBotoes.add(botaoGravar);
166     -
167     -     // Verifica se sera um cadastro ou alteracao de candidato.
168     -     if (NumeroAlterar == -1) {
169     -
170     -         // Caso seja um cadastro adiciona a imagem, uma imagem default de
171     -         cadastro e seta o titulo para Cadastro.
172     -         caminhoImagem = "/icones/cadastro.png";
173     -         setTitle("Cadastro de Candidato");
174     -
175     -         botaoCancelar = new JButton("CANCELAR", new
176     ImageIcon(getClass().getResource("/icones/cancelar.png")));
177     -         botaoCancelar.setFont(new Font(Font.SANS_SERIF, Font.BOLD, 14));
178     -         botaoCancelar.setCursor(new
179     java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
180     -         botaoCancelar.addActionListener(tratadorEventos);
181     -
182     -         painelBotoes.add(botaoCancelar);
183     -     }
184     -     else {
185     -
186     -         // Caso seja uma alteracao, adiciona todas as informacoes do
187     -         candidato aos objetos e seta o titulo para alteracao.
188     -         setTitle("Alteracao de Candidato");
189     -         caminhoImagem = caminhoFoto;
190     -         fieldNumero.setText(numero);
191     -         fieldNome.setText(nome);
192     -         cargoField.setText(cargo);
193     -         partidoField.setText(partido);
194     -

```

```

185         botaoExcluir = new JButton("EXCLUIR", new
        ImageIcon(getClass().getResource("/icones/excluir.png")));
186         botaoExcluir.setFont(new Font(Font.SANS_SERIF, Font.BOLD, 14));
187         botaoExcluir.setCursor(new
        java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
188         botaoExcluir.addActionListener(tratadorEventos);
189
190         painelBotoes.add(botaoExcluir);
191     }
192
193     // Adiciona o painelBotoes ao sul do DialogoCadastrarCandidato.
194     add(painelBotoes, BorderLayout.SOUTH);
195
196     183     Image img;
197     if (caminhoImagem.equalsIgnoreCase("/icones/cadastro.png"))
198         img =
        Toolkit.getDefaultToolkit().getImage(getClass().getResource(caminhoImagem));
184     +     if (caminhoFoto.equalsIgnoreCase("/icones/cadastro.png"))
185     +         img =
        Toolkit.getDefaultToolkit().getImage(getClass().getResource(caminhoFoto));
199     186     else
200         img = Toolkit.getDefaultToolkit().getImage(caminhoImagem);
201     Image menor = img.getScaledInstance(200, 200, Image.SCALE_DEFAULT);
187     +     img = Toolkit.getDefaultToolkit().getImage(caminhoFoto);
188     +     Image menor = img.getScaledInstance(200, 200, Image.SCALE_DEFAULT);
202     189     FotoCandidato = new ImageIcon(menor);
203     190     JLabel labelFoto = new JLabel(FotoCandidato);
204
205     // Adiciona o labelFoto ao painelCentro.
206     191     painelCentro.add(labelFoto);
192     +     return painelCentro;
193     + }
194     +
195     +     private JPanel criarPainelBotoes(TratadorEventosCadastroCandidato
        tratadorEventos) {
196     +         JPanel painelBotoes = new JPanel();
197     +         botaoGravar = criarBotao("GRAVAR", "/icones/gravar.png",
        tratadorEventos);
198     +         painelBotoes.add(botaoGravar);
199     +         botaoCancelar = criarBotao("CANCELAR", "/icones/cancelar.png",
        tratadorEventos);
200     +         painelBotoes.add(botaoCancelar);
201     +         return painelBotoes;
202     +     }
207     203
208     // Adiciona o painelCentro no centro do DialogoCadastrarCandidato.
204     +     private JButton criarBotao(String texto, String iconePath,
        TratadorEventosCadastroCandidato tratadorEventos) {
205     +         JButton botao = new JButton(texto, new
        ImageIcon(getClass().getResource(iconePath)));
206     +         botao.setFont(new Font(Font.SANS_SERIF, Font.BOLD, 14));
207     +         botao.setCursor(new java.awt.Cursor(java.awt.Cursor.HAND_CURSOR));
208     +         botao.addActionListener(tratadorEventos);
209     +         return botao;

```

```

210 +     }
211 +
212 +     private void configurarDialogo(JPanel painelNorte, JPanel painelCentro, JPanel
    painelBotoes) {
213 +         setLayout(new BorderLayout());
214 +         add(painelNorte, BorderLayout.NORTH);
209 215 +         add(painelCentro, BorderLayout.CENTER);
216 +         add(painelBotoes, BorderLayout.SOUTH);
217 +     }
210 218
211 -         pack();
212 -
    setIconImage(Toolkit.getDefaultToolkit().getImage(getClass().getResource("/icones/icone.p
    ng"))));
213 -         setLocationRelativeTo(null);
214 -         setModal(true);
215 -         setDefaultCloseOperation(WindowConstants.DISPOSE_ON_CLOSE);
216 -         setResizable(false);
217 -         setVisible(true);
219 +     private String dialogoAbrirArquivo(Component parent, String titulo) {
220 +         JFileChooser fileChooser = new JFileChooser();
221 +         FileNameExtensionFilter filtro = new FileNameExtensionFilter("Imagens",
    "jpg", "jpeg", "png", "gif");
222 +         fileChooser.setFileFilter(filtro);
223 +         int returnVal = fileChooser.showOpenDialog(parent);
224 +         if (returnVal == JFileChooser.APPROVE_OPTION) {
225 +             File arquivo = fileChooser.getSelectedFile();
226 +             return arquivo.getPath();
227 +         }
228 +         return null;
218 229     }
230 +
231 +     private DialogoCadastrarCandidato getThis() {
232 +         return this;
233 +     }
234 +
219 235
220 236 /**
221 237  * Este metodo retorna a referencia da propria classe.

```

0 comments on commit `c7c2efd`Please [sign in](#) to comment.