

Commit

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

Extract Method - 10, Decompose Conditional - 6

[Browse files](#)

Ruan-Pablo committed on Jun 21

1 parent [1d3b219](#) commit [2375835](#)

Showing 1 changed file with 113 additions and 163 deletions.

[Whitespace](#)[Ignore whitespace](#)[Split](#)[Unified](#)

276 source/src/com/sige/gui/votacao/eventos/TratadorEventosUrnaEletronica.java

```
87      87      * trata os eventos de todos os botoes da urna eletronica.
88      88      */
89      89      public void actionPerformed(ActionEvent evento) {
90          90          +          if (votou) {
91              91              +              return;
92              92              +          }
90      93
91      -          // Verifica se o usuario ja acabou de votar.
92      -          if (!this.votou == true) {
93      -
94      -          // Verifica se o usuario clicou no botao Branco.
95      -          if (evento.getSource() == gui.getBotaoBranco() ) {
96      -
97      -          // Verifica se o label Numero do Candidato e igual a 0,
98      -          se sim modifica o painel para voto em branco.
99      -          if (gui.getNumeroCandidato().getText().trim().length() ==
100      -          0) {
101      -
102      -          gui.getSeuVotoParaLabel().setVisible(true);
103      -          gui.getAperte().setVisible(true);
104      -          gui.getLabelLinha().setVisible(true);
105      -          gui.getLabelLinha2().setVisible(true);
106      -          gui.getLaranjaCorrige().setVisible(true);
107      -          gui.getVerdeConfirma().setVisible(true);
108      -          gui.getNumeroLabel().setText(" VOTO EM");
109      -          gui.getNumeroLabel().setFont(new Font(null,
110      -          Font.PLAIN, 31));
111      -          gui.getNumeroCandidato().setText("BRANCO");
```

```

108 -
109 - gui.getNumeroCandidato().setBorder(BorderFactory.createEmptyBorder());
110 -
111 - gui.getNumeroCandidato().setFont(new Font(null,
112 - Font.PLAIN, 31));
113 -
114 - gui.setCaminhoFoto(gui.TRANSPARENTE);
115 -
116 - }
117 -
118 - }
119 -
120 - // Verifica se o evento aconteceu no votao Corrige. Se sim chama
121 - o metodo corrigeTela().
122 -
123 - else if (evento.getSource() == gui.getBotaoCorrige() ) {
124 -     corrigeTela();
125 - }
126 -
127 - // Verifica se o usuario clicou no botao Confirma.
128 - else if (evento.getSource() == gui.getBotaoConfirma()) {
129 -
130 -     // Verifica se a quantidade de cargos atual e menor que o
131 -     total.
132 -
133 -     if ( gui.getTotalCargos() != gui.getQtdeCargos() ) {
134 -
135 -         try {
136 -             // Caso seja escolhido um candidato
137 -             correto, a imagem do candidato sera trocada.
138 -
139 -             if (
140 - !gui.getCaminhoFoto().equalsIgnoreCase(gui.TRANSPARENTE) ) {
141 -                 tocaSom();
142 -
143 -                 if (evento.getSource() == gui.getBotaoBranco()) {
144 -                     tratarBotaoBranco();
145 -                 } else if (evento.getSource() == gui.getBotaoCorrige()) {
146 -                     corrigeTela();
147 -                 } else if (evento.getSource() == gui.getBotaoConfirma()) {
148 -                     tratarBotaoConfirma();
149 -                 } else {
150 -                     tratarBotoesNumericos(evento.getSource());
151 -                 }
152 -             }
153 -         }
154 -     }
155 -
156 -     dataBaseVotacaoCandidatos.iniciaConexao();
157 -
158 -     int votos =
159 -     dataBaseVotacaoCandidatos.obterVotosCandidato(gui.getNomeCandidato().getText(),
160 - gui.getData_VOTACAO());
161 -
162 -     private void tratarBotaoBranco() {
163 -         if (gui.getNumeroCandidato().getText().trim().length() == 0) {
164 -             exibirPainelVotoBranco();
165 -         }
166 -     }
167 -
168 -     if (votos == 0)
169 -
170 -     dataBaseVotacaoCandidatos.adicionarVotacaoCandidatos(gui.getData_VOTACAO(),
171 - gui.getCargos()[gui.getQtdeCargos()], gui.getNomeCandidato().getText(), votos + 1);
172 -
173 -     else

```

```

136 -
    dataBaseVotacaoCandidatos.atualizarVotacaoCandidatos(gui.getNomeCandidato().getText(),
    votos + 1);
137 -
    dataBaseVotacaoCandidatos.fechaConexao();
111 +     private void exibirPainelVotoBranco() {
112 +         gui.getSeuVotoParaLabel().setVisible(true);
113 +         gui.getAperte().setVisible(true);
114 +         gui.getLabelLinha().setVisible(true);
115 +         gui.getLabelLinha2().setVisible(true);
116 +         gui.getLaranjaCorrige().setVisible(true);
117 +         gui.getVerdeConfirma().setVisible(true);
118 +         gui.getNumeroLabel().setText(" VOTO EM");
119 +         gui.getNumeroLabel().setFont(new Font(null, Font.PLAIN, 31));
120 +         gui.getNumeroCandidato().setText("BRANCO");
121 +         gui.getNumeroCandidato().setBorder(BorderFactory.createEmptyBorder());
122 +         gui.getNumeroCandidato().setFont(new Font(null, Font.PLAIN, 31));
123 +         gui.setCaminhoFoto(gui.TRANSPARENTE);
124 +     }
138 125
139 -
    dataBaseVotacaoCargos.iniciaConexao();
140 -
    dataBaseVotacaoCargos.atualizarEleitores(dataBaseVotacaoCargos.obterQuantidadeEleitores(g
    ui.getCargos()[gui.getQtdeCargos()], gui.getData_VOTACAO()) + 1, gui.getCargos()
    [gui.getQtdeCargos()], gui.getData_VOTACAO());
141 -
    dataBaseVotacaoCargos.fechaConexao();
126 +     private void tratarBotaoConfirma() {
127 +         if (gui.getTotalCargos() != gui.getQtdeCargos()) {
128 +             try {
129 +                 if
    (!gui.getCaminhoFoto().equalsIgnoreCase(gui.TRANSPARENTE)) {
130 +                     realizarVotoCandidato();
131 +                 } else if
    (gui.getPartidoCandidato().getText().equalsIgnoreCase("VOTO NULO")) {
132 +                     realizarVotoNulo();
133 +                 } else if
    (gui.getNumeroCandidato().getText().equalsIgnoreCase("BRANCO")) {
134 +                     realizarVotoBranco();
135 +                 }
136 +             } catch (Exception e) {
137 +                 exibirErroAnalista(e);
138 +             }
139 +         }
142 140
143 -
    if ( gui.getQtdeCargos() <
    gui.getTotalCargos()) {
144 -
    gui.setQtdeCargos(gui.getQtdeCargos() + 1);
145 -
    }
146 -
    corrigeTela();
147 -
    }
141 +     if (gui.getTotalCargos() == gui.getQtdeCargos()) {

```

```

142 +             exibirPainelFimVotacao();
143 +         }
144 +     }
148 145
149 -         // Verifica se o voto e nulo.
150 -         else if
151 -         (gui.getPartidoCandidato().getText().equalsIgnoreCase("VOTO NULO")) {
152 -             tocaSom();
153 -         private void realizarVotoCandidato() throws Exception {
154 -             tocaSom();
155 -         }
156 -         dataBaseVotacaoCargos.iniciaConexao();
157 -         dataBaseVotacaoCargos.atualizarNulo(dataBaseVotacaoCargos.obterQuantidadeNulo(gui.getCargos()[gui.getQtdeCargos()], gui.getDATA_VOTACAO()) + 1, gui.getCargos()[gui.getQtdeCargos()], gui.getDATA_VOTACAO());
158 -         dataBaseVotacaoCargos.atualizarEleitores(dataBaseVotacaoCargos.obterQuantidadeEleitores(gui.getCargos()[gui.getQtdeCargos()], gui.getDATA_VOTACAO()) + 1, gui.getCargos()[gui.getQtdeCargos()], gui.getDATA_VOTACAO());
159 -         dataBaseVotacaoCargos.fechaConexao();
160 -         dataBaseVotacaoCandidatos.iniciaConexao();
161 -         int votos =
162 -         dataBaseVotacaoCandidatos.obterVotosCandidato(gui.getNomeCandidato().getText(),
163 -             gui.getDATA_VOTACAO());
164 -         if ( gui.getQtdeCargos() <
165 -             gui.getTotalCargos()) {
166 -             gui.setQtdeCargos(gui.getQtdeCargos() + 1);
167 -         }
168 -         corrigeTela();
169 -         if (votos == 0)
170 -         dataBaseVotacaoCandidatos.adicionarVotacaoCandidatos(gui.getDATA_VOTACAO(),
171 -             gui.getCargos()[gui.getQtdeCargos()], gui.getNomeCandidato().getText(), votos + 1);
172 -         else
173 -         dataBaseVotacaoCandidatos.atualizarVotacaoCandidatos(gui.getNomeCandidato().getText(),
174 -             votos + 1);
175 -         dataBaseVotacaoCandidatos.fechaConexao();
176 -         }
177 -         dataBaseVotacaoCargos.iniciaConexao();
178 -         dataBaseVotacaoCargos.atualizarVotacaoCargos(gui.getDATA_VOTACAO(),
179 -             gui.getCargos()[gui.getQtdeCargos()], gui.getNomeCandidato().getText(),
180 -             gui.getQtdeCargos() + 1);
181 -         dataBaseVotacaoCargos.fechaConexao();
182 -         // Verifica se o voto e Branco.
183 -         else if
184 -         (gui.getNumeroCandidato().getText().equalsIgnoreCase("BRANCO")) {

```

```

167         -                                     tocaSom();
162         +                                     proximoCargo();
163         +     }
168         164
169         -
170         -     dataBaseVotacaoCargos.iniciaConexao();
171         -
172         -     dataBaseVotacaoCargos.atualizarBranco(dataBaseVotacaoCargos.obterQuantidadeBranco(gui.get
173         Cargos()[gui.getQtdeCargos()], gui.getData_VOTACAO()) + 1, gui.getCargos()
174         [gui.getQtdeCargos()], gui.getData_VOTACAO());
175         -
176         -     dataBaseVotacaoCargos.atualizarEleitores(dataBaseVotacaoCargos.obterQuantidadeEleitores(g
177         ui.getCargos()[gui.getQtdeCargos()], gui.getData_VOTACAO()) + 1, gui.getCargos()
178         [gui.getQtdeCargos()], gui.getData_VOTACAO());
179         -
180         -     dataBaseVotacaoCargos.fechaConexao();
181         165         +     private void realizarVotoNulo() throws Exception {
182         166         +         tocaSom();
183         167
184         -
185         -                                     if ( gui.getQtdeCargos() <
186         gui.getTotalCargos()) {
187         -
188         -     gui.setQtdeCargos(gui.getQtdeCargos() + 1);
189         -
190         -                                     }
191         -                                     corrigeTela();
192         -                                     }
193         -     } catch (Exception e) {
194         -         new DialogoErro(gui, "Erro", "Informe o
195         Seguinte Erro ao Analista: " + e.toString());
196         -     }
197         -     }
198         168         +     dataBaseVotacaoCargos.iniciaConexao();
199         169         +     dataBaseVotacaoCargos.atualizarVotacaoCargos(gui.getData_VOTACAO(),
200         gui.getCargos()[gui.getQtdeCargos()], "VOTO NULO", gui.getQtdeCargos() + 1);
201         170         +     dataBaseVotacaoCargos.fechaConexao();
202         171
203         -
204         -     // Verifica se o usuario ja votou em todos os cargos. Se
205         -     sim Cria um painelFim e adiciona no painelTela.
206         -     if ( gui.getTotalCargos() == gui.getQtdeCargos() ) {
207         -         gui.getSeuVotoParaLabel().setVisible(false);
208         -         gui.getAperte().setVisible(false);
209         -         gui.getLabelLinha().setVisible(false);
210         -         gui.getLabelLinha2().setVisible(false);
211         -         gui.getLaranjaCorrige().setVisible(false);
212         -         gui.getVerdeConfirma().setVisible(false);
213         -         gui.getNumeroLabel().setVisible(false);
214         -         gui.getNumeroCandidato().setVisible(false);
215         -         gui.getCargoLabel().setVisible(false);
216         -
217         -         JPanel painelFIM = new JPanel();
218         -         JLabel partidoCandidato = new JLabel("FIM");
219         -         partidoCandidato.setFont(new Font(null,
220         Font.BOLD, 150));
221         -         painelFIM.add(partidoCandidato);

```

```

200         -                painelFIM.setPreferredSize(new Dimension(500,
201         -                500));
202         -
203         -                painelFIM.setBackground(Color.LIGHT_GRAY);
204         -
205         -                GridBagConstraints c = new GridBagConstraints();
206         -
207         -                c.insets = new Insets(500, 300, 300, 150);
208         -                c.fill = GridBagConstraints.HORIZONTAL;
209         -                c.anchor = GridBagConstraints.FIRST_LINE_END;
210         -                c.gridx = 0;
211         -                c.gridy = 0;
212         -                gui.setPainelTela(painelFIM, c);
213         -
214         -                JPanel painelVotou = new JPanel();
215         -                painelVotou.setBackground(Color.LIGHT_GRAY);
216         -                JLabel votou = new JLabel("VOTOU");
217         -                votou.setFont(new Font(null, Font.PLAIN, 40));
218         -                painelVotou.add(votou);
219         -
220         -                c.insets = new Insets(10, 500, 187, 33);
221         -                c.fill = GridBagConstraints.HORIZONTAL;
222         -                c.anchor = GridBagConstraints.LAST_LINE_END;
223         -                c.gridx = 0;
224         -                c.gridy = 0;
225         -                gui.setPainelTela(painelVotou, c);
226         -                this.votou = true;
227         -                }
228         -                }
229         -                proximoCargo();
230         -                }
231         -                }
232         -                }
233         -                }
234         -                }
235         -                }
236         -                }
237         -                }
238         -                }
239         -                }
240         -                }
241         -                }

```

```

182 +         proximoCargo();
183 +     }
242 184
243 -         // Verifica se o usuario clicou no botao 3. Se sim chama o
        tratador de botoes passando como argumento o numero dele.
244 -         else if (evento.getSource() == gui.getBotoesNumericos()[3] ) {
245 -             trataBotoes(3);
246 -         }
185 +     private void exibirErroAnalista(Exception e) {
186 +         JOptionPane.showMessageDialog(gui, "Erro ao votar: " + e.getMessage(),
        "Erro", JOptionPane.ERROR_MESSAGE);
187 +     }
247 188
248 -         // Verifica se o usuario clicou no botao 4. Se sim chama o
        tratador de botoes passando como argumento o numero dele.
249 -         else if (evento.getSource() == gui.getBotoesNumericos()[4] ) {
250 -             trataBotoes(4);
251 -         }
189 +     private void exibirPainelFimVotacao() {
190 +         gui.getNumeroLabel().setVisible(false);
191 +         gui.getNumeroCandidato().setVisible(false);
192 +         gui.getNomeCandidato().setVisible(false);
193 +         gui.getPartidoCandidato().setVisible(false);
194 +         gui.getFotoCandidato().setVisible(false);
195 +         gui.getLabelCargos().setVisible(false);
196 +         gui.getLabelCargosVotados().setVisible(false);
197 +         gui.getLabelFim().setVisible(true);
198 +         gui.getLabelCargos().setText("VOTAÇÃO ENCERRADA");
199 +         gui.getBotaoBranco().setEnabled(false);
200 +         gui.getBotaoCorrige().setEnabled(false);
201 +         gui.getBotaoConfirma().setEnabled(false);
202 +         gui.getBotaoSair().setVisible(true);
203 +     }
252 204
253 -         // Verifica se o usuario clicou no botao 5. Se sim chama o
        tratador de botoes passando como argumento o numero dele.
254 -         else if (evento.getSource() == gui.getBotoesNumericos()[5] ) {
255 -             trataBotoes(5);
256 -         }
205 +     private void proximoCargo() {
206 +         gui.zerarControlesTela();
257 207
258 -         // Verifica se o usuario clicou no botao 6. Se sim chama o
        tratador de botoes passando como argumento o numero dele.
259 -         else if (evento.getSource() == gui.getBotoesNumericos()[6] ) {
260 -             trataBotoes(6);
261 -         }
208 +         if (gui.getQtdeCargos() < gui.getTotalCargos() - 1) {
209 +             gui.setQtdeCargos(gui.getQtdeCargos() + 1);
210 +             exibirPainelCargos();
211 +         } else {
212 +             exibirPainelFimVotacao();
213 +         }
214 +     }

```

```

262 | 215 |
263 | - | // Verifica se o usuario clicou no botao 7. Se sim chama o
    |   | tratador de botoes passando como argumento o numero dele.
264 | - |     else if (evento.getSource() == gui.getBotoesNumericos()[7] ) {
265 | - |         trataBotoes(7);
266 | - |     }
    | 216 | +     private void tratarBotoesNumericos(Object source) {
    | 217 | +         JButton button = (JButton) source;
    | 218 | +         String buttonText = button.getText();
267 | 219 |
268 | - |     // Verifica se o usuario clicou no botao 8. Se sim chama o
    |   | tratador de botoes passando como argumento o numero dele.
269 | - |     else if (evento.getSource() == gui.getBotoesNumericos()[8] ) {
270 | - |         trataBotoes(8);
    | 220 | +         if (gui.getNumeroCandidato().getText().length() < 5) {
    | 221 | +             if (buttonText.equals("0") &&
    |   | gui.getNumeroCandidato().getText().equals("0")) {
    | 222 | +                 return;
271 | 223 |         }
272 | 224 |
273 | - |     // Verifica se o usuario clicou no botao 9. Se sim chama o
    |   | tratador de botoes passando como argumento o numero dele.
274 | - |     else if (evento.getSource() == gui.getBotoesNumericos()[9] ) {
275 | - |         trataBotoes(9);
276 | - |     }
    | 225 | +     gui.getNumeroCandidato().setText(gui.getNumeroCandidato().getText() + buttonText);
277 | 226 |     }
278 | 227 | }
279 | 228 |
    | 229 | +
280 | 230 | /**
281 | 231 |  * Este metodo corrige a visualizacao do painel.
282 | 232 |  */

```

0 comments on commit 2375835

Please [sign in](#) to comment.