

ANEXO B – Código-fonte

B.1 CÓDIGO DOS AGENTES

B.1.1 atuador_por_categorias.asl

```

!observe.
2
+!observe : true
4   <- .wait(200);
      ?myTools(A1, A2, Id); // discover the tools
6     println("agente criado");
      !consumeItems(Id).
8
+!consumeItems(Id): true
10  <- .wait(1000);
     get_para_Categorias(Item);
12  !processItem(Item, Id);
     !!consumeItems(Id).
14
+!processItem(null, Id) : true
16  <- .my_name(Me).
     //println(Me," : null").
18
+!processItem(Keyword, Id) : true
20  <- executar_abrangente(Keyword, 30, Result1)[artifact_id(Id
    )];
     executar_especifico(Result1, Result2)[artifact_id(Id)];
22  isEmpty(Result2, Test)[artifact_id("entrada")];
     !decidir(Result1, Result2, Test, Id).
24
+!decidir(Result1, Result2, Test, Id) : Test
26  <- .my_name(Me); println(Me," : vazio");
     executar_getIndividuos(Result1, 20, Result3)[artifact_id(
        Id)];
28  put(Result3).

30 +!decidir(Result1, Result2, Test, Id) : true
    <- .my_name(Me); println(Me," : não vazio");
32  executar_getIndividuos(Result2, 20, Result3)[artifact_id(
        Id)];
    put(Result3).
34
+?myTools(A1, A2, A3): true
36  <- lookupArtifact("entrada",A1);
     lookupArtifact("saida",A2);
38  .my_name(N);
     lookupArtifact(N,A3).
40
-?myTools(A1, A2, A3): true
42  <- .wait(100);

```

```
?myTool(A1, A2, A3).
```

B.1.2 atuador_por_classes.asl

```

1 !observe.

3 +!observe : true
  <- .wait(200);
5   ?myTools(A1, A2, Id); // discover the tools
   println("agente criado");
7   !consumeItems(Id).

9 +!consumeItems(Id): true
  <- .wait(1000);
11  get_para_Classes(Item);
   !processItem(Item, Id);
13  !!consumeItems(Id).

15 +!processItem(null, Id) : true
  <- .my_name(Me).
17  //println(Me," : null").

19 +!processItem(Keyword, Id) : true
  <- executar_abrangente(Keyword, 30, Result1)[artifact_id(Id)
   ];
21  executar_especifico(Result1, Result2)[artifact_id(Id)];
   isEmpty(Result2, Test)[artifact_id("entrada")];
23  !decidir(Result1, Result2, Test, Id).

25 +!decidir(Result1, Result2, Test, Id) : Test
  <- .my_name(Me); println(Me," : vazio");
27  executar_getIndividuos(Result1, 20, Result3)[artifact_id(
   Id)];
   put_less_value(Result3);
29  put(Result3).

31 +!decidir(Result1, Result2, Test, Id) : true
  <- .my_name(Me); println(Me," : não vazio");
33  executar_getIndividuos(Result2, 20, Result3)[artifact_id(
   Id)];
   put_less_value(Result3);
35  put(Result3).

37 +?myTools(A1, A2, A3): true
  <- lookupArtifact("entrada",A1);
39   lookupArtifact("saida",A2);
   .my_name(N);

```

```

41     lookupArtifact(N,A3).
43 -?myTools(A1, A2, A3): true
    <- .wait(100);
45     ?myTool(A1, A2, A3).

```

B.1.3 atuador_por_individuos.asl

```

1  !observe.

3  +!observe : true
    <- .wait(200);
5      ?myTools(A1, A2, Id); // discover the tools
    println("agente criado");
7      !consumeItems(Id).

9  +!consumeItems(Id): true
    <- .wait(1000);
11  get_para_Individuos(Item);
    !processItem(Item, Id);
13  !!consumeItems(Id).

15 +!processItem(null, Id) : true
    <- .my_name(Me).
17  //println(Me," : null").

19 +!processItem(Keyword, Id) : true
    <- executar_abrangente(Keyword, 50, Result)[artifact_id(Id)
    ];
21  executar_especifico(Result, Result2)[artifact_id(Id)];
    isEmpty(Result2, Test)[artifact_id("entrada")];
23  !decidir(Result1, Result2, Test, Id).

25 +!decidir(Result1, Result2, Test, Id) : Test
    <- .my_name(Me); println(Me," : vazio");
27  put_more_value(Result2);
    put(Result2).

29

31 +!decidir(Result1, Result2, Test, Id) : true
    <- .my_name(Me); println(Me," : não vazio");
    put_more_value(Result2);
33  put(Result2).

35 +?myTools(A1, A2, A3): true
    <- lookupArtifact("entrada",A1);
37     lookupArtifact("saida",A2);
    .my_name(N);

```

```

39     lookupArtifact(N,A3).

41 -?myTools(A1, A2, A3): true
    <- .wait(100);
43     ?myTool(A1, A2, A3).

```

B.1.4 inicializador.asl

```

1 !create_and_use.

3 +!create_and_use : true
  <- !setupTools;
5     println("artefatos criados").

7 // create the tools
+!setupTools: true
9     <- makeArtifact("entrada","artifacts.ArtefatoEntrada",[],
    A1);
    makeArtifact("saida","artifacts.ArtefatoSaida",[],A2);

11 //makeArtifact("atuador_por_individuos","artifacts.
    AbordagemPorIndividuo",[],A3);
13 //makeArtifact("atuador_por_classes","artifacts.
    AbordagemPorClasses",[],A4);
    //makeArtifact("atuador_por_categorias","artifacts.
    AbordagemPorCategoria",[],A5)

15 makeArtifact("atuador_por_individuos1","artifacts.
    AbordagemPorIndividuo",[],A3);
17 makeArtifact("atuador_por_individuos2","artifacts.
    AbordagemPorIndividuo",[],A6);
    makeArtifact("atuador_por_individuos3","artifacts.
    AbordagemPorIndividuo",[],A9);
19 makeArtifact("atuador_por_classes1","artifacts.
    AbordagemPorClasses",[],A4);
    makeArtifact("atuador_por_classes2","artifacts.
    AbordagemPorClasses",[],A7);
21 makeArtifact("atuador_por_classes3","artifacts.
    AbordagemPorClasses",[],A10);
    makeArtifact("atuador_por_categorias1","artifacts.
    AbordagemPorCategoria",[],A5);
23 makeArtifact("atuador_por_categorias2","artifacts.
    AbordagemPorCategoria",[],A8);
    makeArtifact("atuador_por_categorias3","artifacts.
    AbordagemPorCategoria",[],A11).

```

B.2 CÓDIGO DOS ARTEFATOS

B.2.1 AbordagemPorCategoria.java

```

1 package artifacts;

3 import cartago.*;
  import com.hp.hpl.jena.query.QueryExecution;
5 import com.hp.hpl.jena.query.QuerySolution;
  import com.hp.hpl.jena.query.ResultSet;
7 import com.hp.hpl.jena.sparql.engine.http.QueryEngineHTTP;
  import java.util.ArrayList;
9 import java.util.Arrays;
  import util.NLPTool;

11
12 public class AbordagemPorCategoria extends Artifact
13 {
14     void init()
15     {
16
17     }

18     @OPERATION
19     void executar_abrangente(String keyphrase, int limit,
20                             OpFeedbackParam<Object> res)
21     {
22         System.out.println("LOG: Abordagem por Categoria
23                             Abrangente STARTED");
24         int t_max = 6;
25         ArrayList<String> result = new ArrayList<String>();
26         boolean success = false;
27         while(!success)
28         {
29             try
30             {
31                 result = executar_abrangente2(keyphrase,
32                                             t_max, limit);
33                 success = true;
34             } catch (Exception e)
35             {
36                 t_max -= 2;
37                 if (t_max == 0)
38                 {
39                     res.set(new ArrayList<String>());
40                     System.out.println("LOG: Abordagem por
41                                     Categoria Abrangente ENDED: 0");
42                     return;
43                 }
44             }
45         }
46     }

```

```

    }
43     res.set(result);
    System.out.println("LOG: Abordagem por Categoria
        Abrangente ENDED: "+result.size());
45 }

47 @OPERATION
void executar_especifico(ArrayList<String> categorias ,
    OpFeedbackParam<Object> res)
49 {
    System.out.println("LOG: Abordagem por Categoria
        Especifico STARTED");
51     int t_max = 6;
    ArrayList<String> result = new ArrayList<String>();
53     boolean success = false;
    while(!success)
55     {
        try
57         {
            result = executar_especifico2(categorias ,
                t_max);
            success = true;
59         } catch (Exception e)
        {
            t_max--;
61             if (t_max == 0)
            {
63                 res.set(new ArrayList<String>());
                System.out.println("LOG: Abordagem por
                    Categoria Especifico ENDED: 0");
65                 return;
            }
67         }
69     }
    res.set(result);
71     System.out.println("LOG: Abordagem por Categoria
        Especifico ENDED: "+result.size());
73 }

75 public ArrayList<String> executar_abrangente2(String
    keyphrase, int t_max, int limit)
    {
77         keyphrase = NLPTool.toLowerCase(keyphrase);
        ArrayList<String> keywords = NLPTool.separateWords(
            keyphrase);
79         String keyword = keywords.remove(0);

81         String queryString = AbordagemPorIndividuo.
            getPrefixes()
                + "select distinct ?Category where {\"
                + "?s dcterms:subject ?Category .\"
83                 + " ?Category rdfs:label ?title .\"

```



```

85         + " ?title <bif:contains> \""+keyword+"\" ."
            + filtrarCategorias(AbordagemPorIndividuo.
                categorias_filtradas , t_max)
87         + filtrarKeyphrase(keywords)
            + "} LIMIT "+limit;
89
QueryExecution qe = new QueryEngineHTTP("http://
            dbpedia.org/sparql", queryString);
91
ArrayList<String> categorias = new ArrayList<String>
    >();
93
try
{
95     ResultSet results = qe.execSelect();
    for (; results.hasNext(); )
97     {
        QuerySolution s = results.next();
99         categorias.add(s.getResource("Category").
            toString());
    }
101
}
finally
{
103     qe.close();
105
}

107     return categorias;
    }
109
@OPERATION
111 ArrayList<String> executar_especifico2(ArrayList<String>
    categorias , int t_max)
{
113     System.out.println("LOG: Categorias , tentativa: "+
        t_max+" STARTED");

115     ArrayList<String> resultado = new ArrayList<String>
        >();

117     int cont = 1;
    for(String categoria: categorias)
119     {
        //System.out.println("LOG: "+cont+" "+categoria)
        ;
121     cont++;
        String queryString = AbordagemPorIndividuo.
            getPrefixes()
            + "SELECT distinct ?Category "
            + "WHERE {"
123
            + "{?s dcterms:subject ?Category FILTER (?
                Category = <"+categoria+">) .}"
125

```

```

127         + filtrarCategorias (AbordagemPorIndividuo.
128             categorias_filtradas_avancado , t_max)
129         + "}"
130         + "LIMIT 5";
131
132     QueryExecution qe = new QueryEngineHTTP("http://
133         dbpedia.org/sparql", queryString);
134
135     ArrayList<String> parcial = new ArrayList<String>
136         >();
137     try
138     {
139         ResultSet results = qe.execSelect();
140         for (; results.hasNext(); )
141         {
142             QuerySolution s = results.next();
143             parcial.add(s.getResource("Category").
144                 toString());
145         }
146     }
147     finally
148     {
149         qe.close();
150     }
151     //System.out.println("parcial: "+parcial);
152     resultado.addAll(parcial);
153 }
154 System.out.println("categorias: "+resultado);
155
156 System.out.println("LOG: Categorias , tentativa: "+
157     t_max+" ENDED");
158
159 return resultado;
160 }
161
162 @OPERATION
163 void executar_getIndividuos (ArrayList<String> categorias
164     , int limit_ind , OpFeedbackParam<Object> result)
165 {
166     System.out.println("LOG: Abordagem por Categorias —
167         getIndividuos STARTED");
168
169     ArrayList<String> individuos = new ArrayList<String>
170         >();
171     for (String classe: categorias)
172         individuos.addAll(getAllIndividuals(classe ,
173             limit_ind));
174
175     System.out.println("individuos—categorias: "+
176         individuos);

```

```

169         result.set(individuos);

171         System.out.println("LOG: Abordagem por Categorias –
                               getIndividuos ENDED: "+individuos.size());
    }

173     public ArrayList<String> getAllIndividuals(String
        category, int limit)
175     {
        String queryString = AbordagemPorIndividuo.
            getPrefixes()
177             + "select distinct ?ind where {"
            + " ?ind dcterms:subject <"+category+"> ."
179             + "} LIMIT "+limit;

181         QueryExecution qe = new QueryEngineHTTP("http://
            dbpedia.org/sparql", queryString);

183         ArrayList<String> ind = new ArrayList<String>();
        try
185         {
            ResultSet results = qe.execSelect();
187             for (; results.hasNext(); )
            {
189                 QuerySolution s = results.next();
                    ind.add(s.getResource("ind").toString());
191             }
        }
193         finally
        {
195             qe.close();
        }
197         return ind;
    }

199     private String filtrarKeyphrase(ArrayList<String>
        keywords)
201     {
        //String filters_keyphrase = "";
203         //for(int i = 0; i < keywords.size(); i++)
        //    filters_keyphrase += "FILTER (CONTAINS(?title ,
        //        \"+keywords.get(i) +"\") .";
205         //return filters_keyphrase;
        return "";
207     }

209     private String filtrarCategorias(ArrayList<String>
        categorias, int t_max)
211     {
        String option = "option(TRANSITIVE, T_DISTINCT,
            t_max("+t_max+"), T_NO_CYCLES, t_shortest_only)"
            ;
    }

```

```

String result = "";
213 for (String s: categorias)
    result += "filter (NOT EXISTS {?Category skos:
        broader dbpedia.category:" + s + " " + option + " })
        .";
215 return result;
    }
217 }

```

B.2.2 AbordagemPorClasses.java

```

package artifacts;

2 import cartago.*;
4 import com.hp.hpl.jena.query.QueryExecution;
import com.hp.hpl.jena.query.QuerySolution;
6 import com.hp.hpl.jena.query.ResultSet;
import com.hp.hpl.jena.sparql.engine.http.QueryEngineHTTP;
8 import java.util.ArrayList;
import java.util.Arrays;
10 import util.NLPTool;

12 public class AbordagemPorClasses extends Artifact
{
14     void init()
    {
16     }

18     @OPERATION
20     void executar_abrangente(String keyphrase, int limit,
        OpFeedbackParam<Object> result)
    {
22         System.out.println("LOG: Abordagem por Classes
            STARTED");

24         keyphrase = NLPTool.toLowerCase(keyphrase);
        ArrayList<String> keywords = NLPTool.separateWords(
            keyphrase);
26         String keyword = keywords.remove(0);

28         String queryString = AbordagemPorIndividuo.
            getPrefixes()
                + "select distinct ?Concept where {"
30                 + "[ ] a ?Concept ."
                + " ?Concept rdfs:label ?title ."
32                 + " ?title <bif:contains> \"" + keyword + "\" ."

```

```

        + filtrarClasses(AbordagemPorIndividuo.
            classes_filtradas , 5)
34         + filtrarKeyphrase(keywords)
        + "} LIMIT "+limit;

36
QueryExecution qe = new QueryEngineHTTP("http://
    dbpedia.org/sparql", queryString);
38
ArrayList<String> classes = new ArrayList<String>();
40
try
42 {
    ResultSet results = qe.execSelect();
44     for (; results.hasNext(); )
    {
46         QuerySolution s = results.next();
        classes.add(s.getResource("Concept").
            toString());
48     }
    }
50 finally
52 {
    qe.close();
54 }

    result.set(classes);

56
    System.out.println("LOG: Abordagem por Classes ENDED
        : "+classes.size());
58 }

@OPERATION
60 void executar_especifico(ArrayList<String> classes ,
    OpFeedbackParam<Object> result)
62 {
    System.out.println("LOG: Abordagem por Classes
        Avancado STARTED");
64
    ArrayList<String> resultado = new ArrayList<String>
        >();
66
    int cont = 1;
68     for(String classe: classes)
    {
70         //System.out.println("LOG: "+cont+" "+classe);
        cont++;
72         String queryString = AbordagemPorIndividuo.
            getPrefixes()
            + "SELECT distinct ?Concept "
74             + "WHERE {"

```

```

76         + "{?object ?p ?Concept FILTER (?Concept = <
           "+classe+">) .}"

78         + filtrarClasses(AbordagemPorIndividuo.
           classes_filtradas_avancado , 5)
       + "}"
80       + "LIMIT 5";

82       QueryExecution qe = new QueryEngineHTTP("http://
           dbpedia.org/sparql", queryString);

84       ArrayList<String> parcial = new ArrayList<String>
           >();
       try
86       {
           ResultSet results = qe.execSelect();
88           for (; results.hasNext());
           {
               QuerySolution s = results.next();
               parcial.add(s.getResource("Concept").
                   toString());
92           }
       }
94       finally
       {
96           qe.close();
       }
98       //System.out.println("parcial: "+parcial);
       resultado.addAll(parcial);
100   }
   System.out.println("classes: "+resultado);
102
   result.set(resultado);
104
   System.out.println("LOG: Abordagem por Classes
       Avancado ENDED: "+resultado.size());
106 }

108 @OPERATION
   void executar_getIndividuos(ArrayList<String> classes ,
       int limit_subclasses , OpFeedbackParam<Object> result
       )
110 {
       System.out.println("LOG: Abordagem por Classes
           Avancado — getIndividuos STARTED");
112
       ArrayList<String> individuos = new ArrayList<String>
           >();
114       for(String classe: classes)
           individuos.addAll(getAllSubClasses(classe ,
               limit_subclasses));
116

```

```

System.out.println("indivíduos-classes: "+indivíduos
);
118
result.set(indivíduos);
120
System.out.println("LOG: Abordagem por Classes
Avançado – getIndivíduos ENDED: "+indivíduos.
size());
122 }

124 public ArrayList<String> getAllSubClasses(String classe ,
int limit)
{
126 String queryString = AbordagemPorIndividuo.
getPrefixes()
+ "select distinct ?subclass where {"
128 + " ?subclass a <"+classe+"> option(
TRANSITIVE, t_distinct , TMAX(4))."
+ "} LIMIT "+limit;

130 QueryExecution qe = new QueryEngineHTTP("http://
dbpedia.org/sparql", queryString);
132
ArrayList<String> subclasses = new ArrayList<String>
>();
134 try
{
136 ResultSet results = qe.execSelect();
for (; results.hasNext(); )
138 {
QuerySolution s = results.next();
140 subclasses.add(s.getResource("subclass").
toString());
}
142 }
finally
{
144 {
qe.close();
146 }
return subclasses;
148 }

150 private String filtrarClasses(ArrayList<String> classes ,
int t_max)
{
152 String option = "option(TRANSITIVE, t_distinct ,
TMAX("+t_max+"))";
String result = "";
154 for(String s: classes)
result += "filter (NOT EXISTS {?Concept rdfs:
subclassOf "+s+" "+option+" } ) .";
156 return result;

```

```

    }
158     private String filtrarKeyphrase ( ArrayList<String>
        keywords)
160     {
        //String filters_keyphrase = "";
        //for(int i = 0; i < keywords.size(); i++)
        //    filters_keyphrase += "FILTER (CONTAINS(?title ,
        //        \"\"+keywords.get(i) +\"\")) .";
162        //return filters_keyphrase;
        return "";
164    }
166 }

```

B.2.3 AbordagemPorIndividuo.java

```

package artifacts;
2
import cartago.*;
4 import com.hp.hpl.jena.query.QueryExecution;
import com.hp.hpl.jena.query.QuerySolution;
6 import com.hp.hpl.jena.query.ResultSet;
import com.hp.hpl.jena.sparql.engine.http.QueryEngineHTTP;
8 import java.util.ArrayList;
import java.util.Arrays;
10 import util.NLPTool;

12 public class AbordagemPorIndividuo extends Artifact
{
14     void init()
    {
16     }

18     @OPERATION
    void executar_abrangente(String keyphrase, int limit,
        OpFeedbackParam<Object> result)
    {
22         System.out.println("LOG: Abordagem por Individuos
            STARTED");
        keyphrase = NLPTool.toLowerCase(keyphrase);
24         ArrayList<String> keywords = NLPTool.separateWords(
            keyphrase);
        String keyword = keywords.remove(0);

26         String queryString = getPrefixes()
            + "SELECT distinct ?object "
28

```



```

30         + "WHERE {"
31
32         + "{?object db-prop:title ?title . ?title <
           bif:contains> \""+keyword+"\" . "+
           filtrarKeyphrase(keywords, "title")+}"
33
34         + " UNION "
35         + "{?object db-prop:name ?name. ?name <bif:
           contains> \""+keyword+"\" . "+
           filtrarKeyphrase(keywords, "name")+}"
36         //+ " UNION "
37         //+ "{?object foaf:name ?label. ?label <bif:
           :contains> \""+keyword+"\" . "+
           filtrarKeyphrase(keywords, "label")+}"
38
39         + filtrarClasses(classes_filtradas)
40         + filtrarClasses(classes_filtradas_avancado)
41         //+ filtrarCategorias()
42         + "}"
43         + "LIMIT "+limit;
44
45     QueryExecution qe = new QueryEngineHTTP("http://
         dbpedia.org/sparql", queryString);
46
47     ArrayList<String> individuos = new ArrayList<String>
         >();
48     try
49     {
50         ResultSet results = qe.execSelect();
51         for (; results.hasNext(); )
52         {
53             QuerySolution s = results.next();
54             individuos.add(s.getResource("object").
                 toString());
55         }
56     finally
57     {
58         qe.close();
59     }
60
61     result.set(individuos);
62     System.out.println("LOG: Abordagem por Indivíduos
         ENDED: "+individuos.size());
63 }
64
65 @OPERATION
66 void executar_especifico(ArrayList<String> individuos ,
        OpFeedbackParam<Object> result)
67 {
68     System.out.println("LOG: Abordagem por Indivíduos
        Avancado STARTED");

```

```

70     ArrayList<String> resultado = new ArrayList<String>
71         >();
72     int cont = 1;
73     for (String individuo: individuos)
74     {
75         //System.out.println("LOG: "+cont+" "+individuo)
76         ;
77         cont++;
78         String queryString = getPrefixes()
79             + "SELECT distinct ?object "
80             + "WHERE {"
81             + "{?object ?p ?o FILTER (?object = <" +
82                 individuo+">) .}"
83             + filtrarCategorias(categorias_filtradas, 4)
84             + filtrarCategorias(
85                 categorias_filtradas_avancado, 3)
86             + "}"
87             + "LIMIT 5";
88
89         QueryExecution qe = new QueryEngineHTTP("http://
90             dbpedia.org/sparql", queryString);
91
92         ArrayList<String> parcial = new ArrayList<String>
93             >();
94         try
95         {
96             ResultSet results = qe.execSelect();
97             for (; results.hasNext(); )
98             {
99                 QuerySolution s = results.next();
100                 parcial.add(s.getResource("object").
101                     toString());
102             }
103         }
104         finally
105         {
106             qe.close();
107         }
108         //System.out.println("parcial: "+parcial);
109         resultado.addAll(parcial);
110     }
111
112     System.out.println("individuos: "+resultado);
113     result.set(resultado);
114     System.out.println("LOG: Abordagem por Indivíduos
115         Avancado ENDED: "+resultado.size());
116 }
117
118 private String filtrarKeyphrase(ArrayList<String>
119     keywords, String campo)

```

```

112     {
113         //String filters_keyphrase = "";
114         //for(int i = 0; i < keywords.size(); i++)
115             filters_keyphrase += "FILTER (CONTAINS(?"+
116             campo+", \""+keywords.get(i) +"\")) .";
117         //return filters_keyphrase;
118         return "";
119     }

120     public static String getPrefixes()
121     {
122         String s = "PREFIX db-prop: <http://dbpedia.org/
123             property/>"
124             + "PREFIX dbpedia-owl:<http://dbpedia.org/
125                 ontology/>"
126             + "PREFIX rdf:<http://www.w3.org/1999/02/22-
127                 rdf-syntax-ns#>"
128             + "PREFIX rdfs: <http://www.w3.org/2000/01/
129                 rdf-schema#>"
130             + "PREFIX dbpedia-category: <http://dbpedia.
131                 org/resource/Category:> "
132             + "PREFIX dcterms: <http://purl.org/dc/terms
133                 /> "
134             + "PREFIX foaf: <http://xmlns.com/foaf/0.1/>
135                 "
136             + "PREFIX yago:<http://dbpedia.org/class/
137                 yago/> "
138             + "PREFIX skos: <http://www.w3.org/2004/02/
139                 skos/core#> ";
140         return s;
141     }

142     public static ArrayList<String> classes_filtradas =
143         new ArrayList(Arrays.asList(new String[] { "
144             dbpedia-owl:PeriodicalLiterature", "dbpedia-
145             owl:Album",
146             "dbpedia-owl:VideoGame", "dbpedia-owl:
147             TelevisionEpisode", "dbpedia-owl:
148             TelevisionShow",
149             "dbpedia-owl:Film", "dbpedia-owl:Band", "<http
150             ://umbel.org/umbel/rc/TVShow-SingleCW>", "
151             yago:LivingPeople",
152             "yago:Publication106589574", "yago:
153             Gathering107975026"}));
154     public static ArrayList<String>
155         classes_filtradas_avancado =
156         new ArrayList(Arrays.asList(new String[] { "yago:
157             Event100029378",
158             "yago:WebSite106359193", "yago:Award106696483",
159             "yago:Professorship100598056",
160             "yago:Medium106254669", "yago:HotSpot108586330",
161             "yago:Contestant109613191",

```

```

        "yago:Trainer110722575", "yago:
            MemoryDevice103744840", "yago:
            Professional110480253", "yago:
            Person100007846",
144         "yago:Music107020895"}));
private String filtrarClasses(ArrayList<String> classes)
146 {
    String result = "";
148     for(String s: classes)
        result += "filter (NOT EXISTS {?object a "+s+"
            }) .";
150     return result;
}
152
public static ArrayList<String> categorias_filtradas =
154     new ArrayList(Arrays.asList(new String[]{ "Sports
        ", "Comics", "Albums", "Television-series",
        "College-football-seasons"}));
public static ArrayList<String>
    categorias_filtradas_avancado =
156     new ArrayList(Arrays.asList(new String[]{ "
        Clubs-and-societies", "Economies-by-country"
        , "Companies",
        "Sports-teams-by-country", "
        University-and-college-sports-clubs-by-country
        "
158         "National-Register-of-Historic-Places-by-county"
        , "National-Register-of-Historic-Places", "
        Hospitals",
        "Musical-compositions", "Hunting", "
        Deaths-from-disease"}));
160 private String filtrarCategorias(ArrayList<String>
    categorias, int t_max)
{
162     String option = "option(TRANSITIVE , T_DISTINCT,
        t_max("+t_max+"))";
    String result = "";
164     for(String s: categorias)
        result += "filter (NOT EXISTS {?object dterms:
            subject ?Category . ?Category skos:broader
            dbpedia-category:"+s+" "+option+" }) .";
166     return result;
}
168 }

```

B.2.4 ArtefatoEntrada.java

```

1 package artifacts;

3 import cartago.*;
  import java.util.ArrayList;
5 import user_interface.Main;

7 public class ArtefatoEntrada extends Artifact
  {
9     private ArrayList<String> items1 = new ArrayList<String>
        >();
    private ArrayList<String> items2 = new ArrayList<String>
        >();
11    private ArrayList<String> items3 = new ArrayList<String>
        >();

13    void init()
    {
15        items1 = new ArrayList<String>();
        items2 = new ArrayList<String>();
17        items3 = new ArrayList<String>();

19        String args[] = new String[2];
        Main.main(args);
21        Main.control.setArtefatoEntrada(this);
    }

23    @OPERATION
25    void get_para_Individuos(OpFeedbackParam<Object> res){
        if(items1.isEmpty())
27        {
            res.set(null);
29            return;
        }
31        Object item = items1.remove(0);
        res.set(item);
33    }

35    @OPERATION
37    void get_para_Classes(OpFeedbackParam<Object> res){
        if(items2.isEmpty())
        {
39            res.set(null);
            return;
41        }
        Object item = items2.remove(0);
43        res.set(item);
    }

45    @OPERATION
47    void get_para_Categorias(OpFeedbackParam<Object> res){
        if(items3.isEmpty())
49        {

```

```

        res.set(null);
51         return;
    }
53     Object item = items3.remove(0);
    res.set(item);
55 }

57 @OPERATION
void isEmpty( ArrayList<String> array , OpFeedbackParam<
    Boolean> res ){
59     if( array.isEmpty() )
    {
61         res.set(true);
        return;
63     }
    res.set(false);
65 }

67 void fornecerInformacoes( ArrayList<String>
    array_keywords)
    {
69         for( String s: array_keywords)
        {
71             items1.add(s);
            items2.add(s);
73             items3.add(s);
        }
75     }
}

```

B.2.5 ArtefatoSaida.java

```

package artifacts;

2
import cartago.*;
4 import com.hp.hpl.jena.query.QueryExecution;
import com.hp.hpl.jena.query.QuerySolution;
6 import com.hp.hpl.jena.query.ResultSet;
import com.hp.hpl.jena.sparql.engine.http.QueryEngineHTTP;
8 import java.util.ArrayList;
import java.util.Collections;
10 import java.util.Comparator;
import java.util.Hashtable;
12 import java.util.List;
import java.util.Map;
14 import user_interface.Main;
import util.NLPTool;

```

```

16 import util.Rank;
   import util.TipoDeSugestao;
18
19 public class ArtefatoSaida extends Artifact
20 {
21     private ArrayList<String> items;
22     private ArrayList<String> items_less_value;
23     private ArrayList<String> items_more_value;
24     private boolean pronto = false;
25     private int cont = 0;
26     private List<String> resultados_por_quant;
27
28     void init()
29     {
30         items = new ArrayList<String>();
31         items_less_value = new ArrayList<String>();
32         items_more_value = new ArrayList<String>();
33         resultados_por_quant = new ArrayList<String>();
34
35         Main.control.setArtefatoSaida(this);
36     }
37
38     @OPERATION
39     void put(Object obj)
40     {
41         ArrayList<String> array = (ArrayList<String>)obj;
42         items.addAll(array);
43
44         cont++;
45         int quantAgentsPorTipo = 3;
46         if(cont == Controladora.quantKeywords*
47             quantAgentsPorTipo)
48         {
49             resultados_por_quant = Rank.rankByQuant(20,
50                 items, items_less_value, items_more_value);
51             System.out.println("rankeado: " +
52                 resultados_por_quant);
53             pronto = true;
54         }
55     }
56
57     @OPERATION
58     void put_less_value(Object obj)
59     {
60         ArrayList<String> array = (ArrayList<String>)obj;
61         items_less_value.addAll(array);
62     }
63
64     @OPERATION
65     void put_more_value(Object obj)
66     {
67         ArrayList<String> array = (ArrayList<String>)obj;

```

```

        items_more_value.addAll(array);
66     }

68     public boolean isReady()
    {
70         return pronto;
    }

72     public List<String> getItemsDevolvidos(TipoDeSugestao
        tipo)
74     {
        String temp;
        String result;
        List<String> list = new ArrayList<String>();
76         if(tipo == TipoDeSugestao.SUGESTAO_TITLE)
78         {
80             for(String s: resultados_por_quant)
            {
82                 result = extrairTitleAndKeyword(s);
                        if(!result.equals(""))
84                 {
                        temp = s+" : "+result;
86                 list.add(temp);
                        }
88             }
            return list;
90         }
        if(tipo == TipoDeSugestao.SUGESTAO_DESCRIPTION)
92         {
            for(String s: resultados_por_quant)
94             {
                result = extrairDescription(s);
96                 if(!result.equals(""))
                {
98                     temp = s+" : "+result;
                        list.add(temp);
100                }
            }
            return list;
102        }
        if(tipo == TipoDeSugestao.SUGESTAO_KEYWORDS)
104        {
            for(String s: resultados_por_quant)
106            {
                result = extrairTitleAndKeyword(s);
108                if(!result.equals(""))
                {
110                    temp = s+" : "+result;
                        list.add(temp);
112                }
            }
114            return list;

```



```

116         }
117         return resultados_por_quant;
118     }

120     public void resetar()
121     {
122         pronto = false;
123         items.clear();
124         resultados_por_quant.clear();
125         cont = 0;
126     }

128     private static String extrairDescription(String
        individuo)
129     {
130         //System.out.println("LOG: Description STARTED");
131         ArrayList<String> individuos;
132         individuos = extrairDescription-parcial(individuo, "
            dbpedia-owl:abstract");

134         if(individuos.isEmpty())
135         {
136             individuos = extrairDescription-parcial(
                individuo, "rdfs:comment");
137         }

138         if(individuos.isEmpty())
139         {
140             return "";
141         }
142         //System.out.println("LOG: Description ENDED");
143         return individuos.get(0);
144     }

146     private static ArrayList<String>
        extrairDescription-parcial(String individuo, String
        prop)
147     {
148         String queryString = AbordagemPorIndividuo.
            getPrefixes()
149             + "SELECT distinct ?title "
150             + "WHERE "
151             + "{<"+individuo+"> "+prop+" ?title . FILTER
                ( lang(?title) = \"en\") .}"
152             + "LIMIT 5";

154         QueryExecution qe = new QueryEngineHTTP("http://
            dbpedia.org/sparql", queryString);

156         ArrayList<String> individuos = new ArrayList<String>
            >();

158         try

```

```

160         ResultSet results = qe.execSelect();
161         for (; results.hasNext(); )
162         {
163             QuerySolution s = results.next();
164             individuos.add(s.getLiteral("title").
165                 getString().toString());
166         }
167     finally
168     {
169         qe.close();
170     }
171
172     return individuos;
173 }
174
175 private static String extrairTitleAndKeyword(String
176     individuo)
177 {
178     //System.out.println("LOG: Title e Keyword STARTED")
179     ;
180     ArrayList<String> individuos;
181     individuos = extrairTitleAndKeyword-parcial(
182         individuo, "db-prop:title");
183
184     if(individuos.isEmpty())
185     {
186         individuos = extrairTitleAndKeyword-parcial(
187             individuo, "db-prop:name");
188     }
189
190     if(individuos.isEmpty())
191     {
192         individuos = extrairTitleAndKeyword-parcial(
193             individuo, "foaf:name");
194     }
195
196     if(individuos.isEmpty())
197     {
198         individuos = extrairTitleAndKeyword-parcial(
199             individuo, "rdfs:label");
200     }
201
202     if(individuos.isEmpty())
203     {
204         return "";
205     }
206
207     //System.out.println("LOG: Title e Keyword ENDED");
208     return individuos.get(0);
209 }

```

```

204     private static ArrayList<String>
        extrairTitleAndKeyword_parcial(String individuo ,
        String prop)
    {
206         String queryString = AbordagemPorIndividuo.
            getPrefixes()
                + "SELECT distinct ?title "
208                + "WHERE "
                + "{ <"+individuo+"> "+prop+" ?title .
                    FILTER ( lang(?title) = \"en\") .}"
210                + "LIMIT 5";

212         QueryExecution qe = new QueryEngineHTTP("http://
            dbpedia.org/sparql", queryString);

214         ArrayList<String> individuos = new ArrayList<String>
            >();
        try
216         {
            ResultSet results = qe.execSelect();
218             for (; results.hasNext(); )
            {
220                 QuerySolution s = results.next();
                individuos.add(s.getLiteral("title").
                    getString().toString());
222             }
        }
224         finally
        {
226             qe.close();
        }

228         return individuos;
230     }
}

```

B.3 CÓDIGO JAVA

B.3.1 Controladora.java

```

1  package artifacts;

3  import java.util.ArrayList;
   import javax.swing.*;
5  import javax.swing.border.PacoteComCampos;
   import javax.swing.util.TipoDeSugestao;

```

```

7 import java.util.List;
import util.NLPTool;
9
11 public class Controladora
12 {
13     ArtefatoEntrada artefatoEntrada;
14     ArtefatoSaida artefatoSaida;
15     public static int quantKeywords;
16
17     public void setArtefatoEntrada(ArtefatoEntrada artefato)
18     {
19         artefatoEntrada = artefato;
20     }
21
22     public void setArtefatoSaida(ArtefatoSaida artefato)
23     {
24         artefatoSaida = artefato;
25     }
26
27     public void obterSugestoes(JTextArea text, ArrayList<
28         PacoteComCampos> valores, TipoDeSugestao tipo)
29     {
30         ArrayList<String> keywords = new ArrayList<String>()
31         ;
32         for(PacoteComCampos p: valores)
33             keywords.addAll(p.processarKeywords());
34
35         System.out.println("keywords: "+keywords);
36
37         quantKeywords = keywords.size();
38         artefatoSaida.resetar();
39         artefatoEntrada.fornecerInformacoes(keywords);
40         TesteThread myRunnable = new TesteThread(text, tipo)
41         ;
42         Thread t = new Thread(myRunnable);
43         t.start();
44     }
45
46     class TesteThread implements Runnable
47     {
48         private JTextArea field;
49         private TipoDeSugestao tipo;
50
51         public TesteThread(JTextArea field, TipoDeSugestao
52             tipo) {
53             this.field = field;
54             this.tipo = tipo;
55         }
56
57         public void run()
58         {
59             try{

```

```

55         while (!artefatoSaida.isReady()) {
                    Thread.yield();
57         }
        } catch (Exception e) {}
59     List<String> items = artefatoSaida.
        getItemsDevolvidos(tipo);
        String t = "";
61     for (String s: items)
        t+=s+"\n";
63     field.setText(t);
        }
65     }
}

```

B.3.2 NLPTool.java

```

1  package util;

3  import java.util.ArrayList;
   import java.util.Arrays;
5  import java.util.Set;
   import java.util.HashSet;
7  import java.io.IOException;
   import org.apache.lucene.util.*;
9  import org.apache.lucene.analysis.*;
   import java.io.StringReader;
11 import org.apache.lucene.analysis.standard.ClassicTokenizer;
   import org.apache.lucene.analysis.standard.StandardTokenizer
   ;
13 import org.apache.lucene.analysis.tokenattributes.*;
   import org.apache.lucene.analysis.en.EnglishAnalyzer;
15
   public class NLPTool
17 {
        public static String toLowerCase(String s)
19     {
            return s.toLowerCase();
21     }

23     public static ArrayList<String> toLowerCase(ArrayList<
        String> array)
        {
25         ArrayList<String> result = new ArrayList<String>();
           for (int i = 0; i < array.size(); i++)
27         {
                result.add(array.get(i).toLowerCase());
29         }

```

```

        return result;
31     }

33     public static ArrayList<String> separateWords(String
        sentence)
    {
35         String [] t = sentence.split(" ");
        return new ArrayList(Arrays.asList(t));
37     }

39     public static void removeEmptyStrings(ArrayList<String>
        words)
    {
41         for(int i=0; i < words.size(); i++)
            if(words.get(i).trim().isEmpty())
43             {
45                 words.remove(i);
                i--;
            }
47     }

49     public static ArrayList<String>
        separateStringsByDotComma(ArrayList<String> words)
    {
51         String [] keywords;
        ArrayList<String> temp = new ArrayList<String>();
53         ArrayList<String> temp2 = new ArrayList<String>();
        for(int i=0; i < words.size(); i++)
55         {
            keywords = words.get(i).split(";+");
57             temp.addAll(Arrays.asList(keywords));
        }

59         for(String s: temp)
61         {
            temp2.add(s.trim());
63         }

65         removeEmptyStrings(temp2);

67         return temp2;
    }

69     public static ArrayList<String> stopWordsRemoval(String
        input)
    {
71         // tokenize the input string
        TokenStream tokenStream = new ClassicTokenizer(
            Version.LUCENE_35, new StringReader(input));
73         // remove stop words
        tokenStream = new StopFilter(Version.LUCENE_35,
75             tokenStream, EnglishAnalyzer.getDefaultStopSet())

```

```

    );

77     // retrieve the remaining tokens
    Set<String> tokens = new HashSet<String>();
79     CharTermAttribute token = tokenStream.getAttribute(
        CharTermAttribute.class);
    try {
81         while (tokenStream.incrementToken()) {
            tokens.add(token.toString());
83         }
    } catch (IOException e) {
85         System.out.println(e);
    }

87     ArrayList<String> array = new ArrayList<String>();
89     for (String s: tokens)
    {
91         if (!s.toLowerCase().equals("my") && !s.
            toLowerCase().equals("has") && !s.
            toLowerCase().equals("was")
                && !s.toLowerCase().equals("does") && !s.
                toLowerCase().equals("goes") && !s.
                toLowerCase().equals("dies")
93         && !s.toLowerCase().equals("yes") && !s.
            toLowerCase().equals("gets") && !s.
            toLowerCase().equals("its")
                && !s.toLowerCase().equals("he") && !s.
                toLowerCase().equals("she") && !s.
                toLowerCase().equals("it")
95         && !s.toLowerCase().equals("we") && !s.
            toLowerCase().equals("they") && !s.
            toLowerCase().equals("you"))
            array.add(stripEnglishPlural(s));

97     }

99     return array;
101 }

103 private static String stripEnglishPlural(String word)
104 {
105     if ( word.endsWith("sses") ||
        word.endsWith("xes") ||
107     word.endsWith("hes") )
    {
109         // remove 'es'
        return word.substring(0,word.length()-2);
111     }
    else if ( word.endsWith("ies") )
113     {
        // remove 'ies', replace with 'y'
115         return word.substring(0,word.length()-3)+'y';
    }
}

```

```

117     }
    else if ( word.endsWith("s") &&
118             !word.endsWith("ss") &&
119             !word.endsWith("is") &&
120             !word.endsWith("us") &&
121             !word.endsWith("pos") ) {
        // remove 's'
122         return word.substring(0,word.length()-1);
123     }
124     return word;
125 }
126 }
127 }

```

B.3.3 Rank.java

```

package util;

2
import java.util.ArrayList;
4 import java.util.Collections;
import java.util.Comparator;
6 import java.util.Hashtable;
import java.util.List;
8 import java.util.Map;

10 public class Rank
{
12     /*public static List<String> rankBySimilarity(int quant,
        ArrayList<String> keywords,  ArrayList<String>
        individuos) // INCOMPLETO
    {
14
15     }*/

16     public static List<String> rankByQuant(int quant,
        ArrayList<String> individuos)
17     {
18         if(individuos.isEmpty())
19             return new ArrayList<String>();
20         Collections.sort(individuos);
21         Hashtable<String, Double> ranked = new Hashtable<
22             String, Double>();
23         double points;
24         for(int i = 0; i < individuos.size(); i++)
25         {
26             ranked.put(individuos.get(i), 0.0);
27         }
28         for(int i = 0; i < individuos.size(); i++)

```



```

30     {
        points = 1;
        ranked.put(individuos.get(i), ranked.get(
32             individuos.get(i)) + points);
    }

34     ArrayList<String> s_ranked = sortValue(ranked, false
        );
    if(quant > s_ranked.size())
36         quant = s_ranked.size();
    return s_ranked.subList(0, quant);
38 }

40 public static List<String> rankByQuant(int quant,
    ArrayList<String> individuos, ArrayList<String>
    less_value, ArrayList<String> more_value)
    {
42         if(individuos.isEmpty())
            return new ArrayList<String>();
44         Collections.sort(individuos);
        Hashtable<String, Double> ranked = new Hashtable<
            String, Double>();
46         double points;
        for(int i = 0; i < individuos.size(); i++)
48         {
            ranked.put(individuos.get(i), 0.0);
50         }
        for(int i = 0; i < individuos.size(); i++)
52         {
            points = 1;
54             if(less_value.contains(individuos.get(i)))
                {
56                 points = 0.8;
                    less_value.remove(individuos.get(i));
58                     //System.out.println("LESS points: "+
                        individuos.get(i));
                }else if(more_value.contains(individuos.get(i)))
60                 {
                    points = 1.3;
62                     more_value.remove(individuos.get(i));
                        //System.out.println("MORE points: "+
                            individuos.get(i));
                }
64             //else
66             //{
                //    System.out.println("REGULAR points: "+
                    individuos.get(i));
68             //}
            ranked.put(individuos.get(i), ranked.get(
                individuos.get(i)) + points);
70         }
    }

```

```

72     ArrayList<String> s_ranked = sortValue(ranked, false
    );
    if(quant > s_ranked.size())
74         quant = s_ranked.size();
    return s_ranked.subList(0, quant);
76 }

78 private static ArrayList<String> sortValue(Hashtable<
    String, Double> t, boolean output){

80     //Transfer as List and sort it
    ArrayList<Map.Entry<String, Double>> l = new
        ArrayList(t.entrySet());
82     Collections.sort(l, new Comparator<Map.Entry<String,
        Double>>(){

84         public int compare(Map.Entry<String, Double> o1,
            Map.Entry<String, Double> o2) {
            return -o1.getValue().compareTo(o2.getValue());
86         }
    });

88     ArrayList<String> s = new ArrayList<String>();
    for(Map.Entry<String, Double> entry: l)
90     {
        if(!output)
92             s.add(entry.getKey().toString());
        else
94             s.add(entry.toString());
    }
96     return s;
    }
98 }

```

B.3.4 enums

```

1 package util;

3 public enum TipoDeCategoria {
    GeneralParaOBAA, LifeCycleFull, RightsFull
5 }

7 package util;

9 public enum TipoDeSugestao {
11     SUGESTAO.TITLE, SUGESTAO.DESCRPTION, SUGESTAO.KEYWORDS
    }

```

B.4 CÓDIGO DA GUI

B.4.1 Main.java

```
1 package user_interface;
3 import artifacts.Controladora;
5 public class Main
6 {
7     public static Controladora control = new Controladora();
9     public static void main(String args[])
10    {
11        /* Set the Nimbus look and feel */
12        //<editor-fold defaultstate="collapsed" desc=" Look
13        and feel setting code (optional) ">
14        /* If Nimbus (introduced in Java SE 6) is not
15        available, stay with the default look and feel.
16        * For details see http://download.oracle.com/javase
17        /tutorial/uiswing/lookandfeel/plaf.html
18        */
19        try {
20            for (javax.swing.UIManager.LookAndFeelInfo info
21                : javax.swing.UIManager.
22                    getInstalledLookAndFeels()) {
23                if ("Nimbus".equals(info.getName())) {
24                    javax.swing.UIManager.setLookAndFeel(
25                        info.getClassName());
26                    break;
27                }
28            }
29        } catch (ClassNotFoundException ex) {
30            java.util.logging.Logger.getLogger(MainFrame.
31                class.getName()).log(java.util.logging.Level
32                    .SEVERE, null, ex);
33        } catch (InstantiationException ex) {
34            java.util.logging.Logger.getLogger(MainFrame.
35                class.getName()).log(java.util.logging.Level
36                    .SEVERE, null, ex);
37        } catch (IllegalAccessException ex) {
38            java.util.logging.Logger.getLogger(MainFrame.
39                class.getName()).log(java.util.logging.Level
40                    .SEVERE, null, ex);
41        }
42    }
43 }
```

```

29         } catch (javax.swing.UnsupportedLookAndFeelException
           ex) {
           java.util.logging.Logger.getLogger(MainFrame.
             class.getName()).log(java.util.logging.Level
               .SEVERE, null, ex);
31       }
           //</editor-fold>
33
           /* Create and display the form */
35       java.awt.EventQueue.invokeLater(new Runnable() {
           @Override
37         public void run() {
           new MainFrame().setVisible(true);
39       }
           });
41     }
  }

```

B.4.2 MainFrame.java

```

1  package user_interface;

3  import java.awt.Color;
   import java.awt.*;
5  import java.awt.event.*;
   import javax.swing.*;
7  import javax.swing.JOptionPane;

9  public class MainFrame extends javax.swing.JFrame {

11     final static String ADEQUACAO_PROFILE_PANEL = "Card with
           Perfil AdequaÃ§Ã£o";
           final static String MINIMUM_OBAA_PROFILE_PANEL = "Card
           with OBAA MÃnimo Profile";
13     final static String IDEAL_OBAA_PROFILE_PANEL = "Card
           with OBAA Ideal Profile";

15     ProfileAdequacao profileAdequacao = new ProfileAdequacao
           ();
           ProfileOBAAMinimo profileOBAAMin = new ProfileOBAAMinimo
           ();
17     ProfileOBAAIdeal profileOBAAIdeal = new ProfileOBAAIdeal
           ();

19     public MainFrame() {
           initComponents();

```

```

21     buttonGroupProfiles.add(
        jRadioButtonMenuItemProfileAdequacao);
    buttonGroupProfiles.add(
        jRadioButtonMenuItemProfileOBAAMinimo);
23     buttonGroupProfiles.add(
        jRadioButtonMenuItemProfileOBAAIdeal);
    jRadioButtonMenuItemProfileOBAAMinimo.setEnabled(
        true);
25     jPanelCards.setLayout(new CardLayout());
    jPanelCards.add(profileAdequacao,
        ADEQUACAO.PROFILE.PANEL);
27     jPanelCards.add(profileOBAAMin,
        MINIMUM.OBAA.PROFILE.PANEL);
    jPanelCards.add(profileOBAAIdeal,
        IDEAL.OBAA.PROFILE.PANEL);
29     jLabelProfileName.setText("Perfil Compatível
        Mãnimo com OBAA (PM-OBAA-CORE)");
    CardLayout cl = (CardLayout)(jPanelCards.getLayout());
31     cl.show(jPanelCards, MINIMUM.OBAA.PROFILE.PANEL);
}

33 @SuppressWarnings("unchecked")
35 // <editor-fold defaultstate="collapsed" desc="Generated
    Code">
    //GEN-BEGIN: initComponents
    private void initComponents() {
37
        buttonGroupProfiles = new javax.swing.ButtonGroup();
39     jLabel1 = new javax.swing.JLabel();
    jPanelCards = new javax.swing.JPanel();
41     jLabelProfileName = new javax.swing.JLabel();
    jMenuItemBar1 = new javax.swing.JMenuBar();
43     jMenuItem = new javax.swing.JMenuItem();
    jMenuItemNewFile = new javax.swing.JMenuItem();
45     jMenuItemOpenFile = new javax.swing.JMenuItem();
    jMenuItemSaveFile = new javax.swing.JMenuItem();
47     jMenuItemSaveFileAs = new javax.swing.JMenuItem();
    jSeparator1 = new javax.swing.JPopupMenu.Separator();
49     jMenuItemGenerateXMLFile = new javax.swing.JMenuItem();
        ();
    jSeparator2 = new javax.swing.JPopupMenu.Separator();
        ();
51     jMenuItemExit = new javax.swing.JMenuItem();
    jMenuItem2 = new javax.swing.JMenuBar();
53     jRadioButtonMenuItemProfileAdequacao = new javax.
        swing.JRadioButtonMenuItem();
    jRadioButtonMenuItemProfileOBAAMinimo = new javax.
        swing.JRadioButtonMenuItem();
55     jRadioButtonMenuItemProfileOBAAIdeal = new javax.
        swing.JRadioButtonMenuItem();
    jMenuItem3 = new javax.swing.JMenuItem();

```

```

57     setDefaultCloseOperation(javax.swing.WindowConstants
        .EXIT_ON_CLOSE);
59     setTitle("Ferramenta de Autoria de Metadados de
        Objetos de Aprendizagem");

61     jLabel1.setFont(new java.awt.Font("Tahoma", 1, 18));
        // NOI18N
        jLabel1.setText("Current Metadata Profile:");

63     javax.swing.GroupLayout jPanelCardsLayout = new
        javax.swing.GroupLayout(jPanelCards);
65     jPanelCards.setLayout(jPanelCardsLayout);
        jPanelCardsLayout.setHorizontalGroup(
67         jPanelCardsLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 1125, Short.MAX_VALUE)
69     );
        jPanelCardsLayout.setVerticalGroup(
71         jPanelCardsLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 653, Short.MAX_VALUE)
73     );

75     jLabelProfileName.setFont(new java.awt.Font("Tahoma"
        , 0, 18)); // NOI18N
        jLabelProfileName.setText("profile_name");

77     jMenu1.setText("File");

79     jMenuItemNewFile.setText("New File");
81     jMenu1.add(jMenuItemNewFile);

83     jMenuItemOpenFile.setText("Open File");
        jMenu1.add(jMenuItemOpenFile);

85     jMenuItemSaveFile.setText("Save File");
87     jMenu1.add(jMenuItemSaveFile);

89     jMenuItemSaveFileAs.setText("Save File As");
        jMenu1.add(jMenuItemSaveFileAs);
91     jMenu1.add(jSeparator1);

93     jMenuItemGenerateXMLFile.setText("Generate XML File"
        );
        jMenu1.add(jMenuItemGenerateXMLFile);
95     jMenu1.add(jSeparator2);

97     jMenuItemExit.setText("Exit");
        jMenuItemExit.addActionListener(new java.awt.event.
            ActionListener() {

```

```

99         public void actionPerformed(java.awt.event.
           ActionEvent evt) {
           jMenuItemExitActionPerformed(evt);
101     }
    });
103 jMenuItem1.add(jMenuItemExit);

105 jMenuItemBar1.add(jMenuItem);

107 jMenuItem2.setText(" Profiles");

109 jMenuItemButtonMenuItemProfileAdequacao.setText(" Perfil
           AdequaÃ§Ã£o");
jRadioButtonMenuItemProfileAdequacao.
    addActionListener(new java.awt.event.
        ActionListener() {
111         public void actionPerformed(java.awt.event.
           ActionEvent evt) {
           jMenuItemButtonMenuItemProfileAdequacaoActionPerformed
               (evt);
113     }
    });
115 jMenuItem2.add(jRadioButtonMenuItemProfileAdequacao);

117 jMenuItemButtonMenuItemProfileOBAAMinimo.setSelected(
           true);
jRadioButtonMenuItemProfileOBAAMinimo.setText("OBAA
           MÃnimo");
119 jMenuItemButtonMenuItemProfileOBAAMinimo.
    addActionListener(new java.awt.event.
        ActionListener() {
           public void actionPerformed(java.awt.event.
               ActionEvent evt) {
121             jMenuItemButtonMenuItemProfileOBAAMinimoActionPerformed
                (evt);
           }
    });
123 jMenuItem2.add(jRadioButtonMenuItemProfileOBAAMinimo);

125 jMenuItemButtonMenuItemProfileOBAAIdeal.setText("OBAA
           Ideal");
127 jMenuItemButtonMenuItemProfileOBAAIdeal.setToolTipText(
           "");
jRadioButtonMenuItemProfileOBAAIdeal.
    addActionListener(new java.awt.event.
        ActionListener() {
129         public void actionPerformed(java.awt.event.
           ActionEvent evt) {
           jMenuItemButtonMenuItemProfileOBAAIdealActionPerformed
               (evt);
131     }
    });

```

```

133 jMenu2.add(jRadioButtonMenuItemProfileOBAAIdeal);
135 jMenuBar1.add(jMenu2);
137 jMenu3.setText("About");
139 jMenuBar1.add(jMenu3);
141
142 setJMenuBar(jMenuBar1);
143
144 javax.swing.GroupLayout layout = new javax.swing.
145     GroupLayout(getContentPane());
146 getContentPane().setLayout(layout);
147 layout.setHorizontalGroup(
148     layout.createParallelGroup(javax.swing.
149         GroupLayout.Alignment.LEADING)
150         .addGroup(layout.createSequentialGroup()
151             .add(layout.createParallelGroup(
152                 javax.swing.
153                     GroupLayout.Alignment.LEADING)
154                 .addComponent(jPanelCards, javax.swing.
155                     GroupLayout.DEFAULT_SIZE, javax.
156                         swing.GroupLayout.DEFAULT_SIZE,
157                             Short.MAX_VALUE)
158                 .addGroup(layout.createSequentialGroup()
159                     .addComponent(jLabel1)
160                     .addPreferredGap(javax.swing.
161                         LayoutStyle.ComponentPlacement.
162                             RELATED)
163                     .addComponent(jLabelProfileName)
164                     .addGap(0, 0, Short.MAX_VALUE)))
165             .addContainerGap())
166         );
167 layout.setVerticalGroup(
168     layout.createParallelGroup(javax.swing.
169         GroupLayout.Alignment.LEADING)
170         .addGroup(layout.createSequentialGroup()
171             .add(layout.createParallelGroup(
172                 javax.swing.
173                     GroupLayout.Alignment.BASELINE)
174                 .addComponent(jLabel1)
175                 .addComponent(jLabelProfileName))
176             .addPreferredGap(javax.swing.
177                 LayoutStyle.
178                     ComponentPlacement.RELATED)
179             .addComponent(jPanelCards, javax.swing.
180                 GroupLayout.PREFERRED_SIZE, javax.swing.
181                     GroupLayout.DEFAULT_SIZE, javax.swing.
182                         GroupLayout.PREFERRED_SIZE)
183             .addContainerGap(49, Short.MAX_VALUE))
184         );
185
186 java.awt.Dimension screenSize = java.awt.Toolkit.
187     getDefaultToolkit().getScreenSize();

```



```

        setBounds(( screenSize.width-1167)/2, ( screenSize.
            height-815)/2, 1167, 815);
171 }// </editor-fold> //GEN-END: initComponents

173 private void
    jButtonMenuItemPerfilAdequacaoActionPerformed(
        java.awt.event.ActionEvent evt) { //GEN-FIRST:
        event_jButtonMenuItemPerfilAdequacaoActionPerformed

        CardLayout cl = (CardLayout)(jPanelCards.getLayout()
            );
175 cl.show(jPanelCards, ADEQUACAO.PROFILE.PANEL);
        jLabelProfileName.setText(" Perfil AdequaÃ§Ã£o (PM-
            ADEQ-BASE)");
177 } //GEN-LAST:
        event_jButtonMenuItemPerfilAdequacaoActionPerformed

179 private void
    jButtonMenuItemPerfilOBAAMinimoActionPerformed(
        java.awt.event.ActionEvent evt) { //GEN-FIRST:
        event_jButtonMenuItemPerfilOBAAMinimoActionPerformed

        CardLayout cl = (CardLayout)(jPanelCards.getLayout()
            );
181 cl.show(jPanelCards, MINIMUM.OBAA.PROFILE.PANEL);
        jLabelProfileName.setText(" Perfil CompatÃvel
            MÃnimo com OBAA (PM-OBAA-CORE)");
183 } //GEN-LAST:
        event_jButtonMenuItemPerfilOBAAMinimoActionPerformed

185 private void jButtonMenuItemExitActionPerformed( java.awt.event
        .ActionEvent evt) { //GEN-FIRST:
        event_jButtonMenuItemExitActionPerformed
        System.exit(0);
187 } //GEN-LAST: event_jButtonMenuItemExitActionPerformed

189 private void
    jButtonMenuItemPerfilOBAAIdealActionPerformed(
        java.awt.event.ActionEvent evt) { //GEN-FIRST:
        event_jButtonMenuItemPerfilOBAAIdealActionPerformed

        CardLayout cl = (CardLayout)(jPanelCards.getLayout()
            );
191 cl.show(jPanelCards, IDEAL.OBAA.PROFILE.PANEL);
        jLabelProfileName.setText(" Perfil CompatÃvel Ideal
            com OBAA (PM-OBAA-FULL)");
193 } //GEN-LAST:
        event_jButtonMenuItemPerfilOBAAIdealActionPerformed

```

```

195 // Variables declaration — do not modify//GEN-BEGIN:
    variables
196 private javax.swing.ButtonGroup buttonGroupProfiles;
197 private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabelProfileName;
199 private javax.swing.JMenu jMenuItem1;
    private javax.swing.JMenu jMenuItem2;
201 private javax.swing.JMenu jMenuItem3;
    private javax.swing.JMenuBar jMenuBar1;
203 private javax.swing.JMenuItem jMenuItemExit;
    private javax.swing.JMenuItem jMenuItemGenerateXMLFile;
205 private javax.swing.JMenuItem jMenuItemNewFile;
    private javax.swing.JMenuItem jMenuItemOpenFile;
207 private javax.swing.JMenuItem jMenuItemSaveFile;
    private javax.swing.JMenuItem jMenuItemSaveFileAs;
209 private javax.swing.JPanel jPanelCards;
    private javax.swing.JRadioButtonMenuItem
        jMenuItemProfileAdequacao;
211 private javax.swing.JRadioButtonMenuItem
        jMenuItemProfileOBAAIdeal;
    private javax.swing.JRadioButtonMenuItem
        jMenuItemProfileOBAAMinimo;
213 private javax.swing.JPopupMenu.Separator jSeparator1;
    private javax.swing.JPopupMenu.Separator jSeparator2;
215 // End of variables declaration//GEN-END:variables
}

```

B.4.3 Profile.java

```

1 package user_interface;

3 import util.TipoDeSugestao;

5 public interface Profile {
    public void gerarSugestoes(TipoDeSugestao tipo);
7 }

```

B.4.4 ProfileAdequacao.java

```

package user_interface;
2 import java.awt.Color;

```

```

import javax.swing.JOptionPane;
4 import javax.swing.table.DefaultTableModel;
import util.TipoDeSugestao;

6 public class ProfileAdequacao extends javax.swing.JPanel
    implements Profile
8 {

10     public ProfileAdequacao() {
        initComponents();
12         jTableIdentifier.getColumnModel().getColumn(0).
            setMinWidth(270);
        jTableIdentifier.getColumnModel().getColumn(0).
            setMaxWidth(270);
14         jTableIdentifier.setShowGrid(true);
        jTableIdentifier.setGridColor(new Color(0,0,153));
16         jTableIdentifier.setTableHeader(null);
    }

18     @Override
20     public void gerarSugestoes(TipoDeSugestao tipo){}

22     @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated
        Code">
24     private void initComponents() {

26         jTabbedPane = new javax.swing.JTabbedPane();
        jPanel1 = new javax.swing.JPanel();
28         jLabel7 = new javax.swing.JLabel();
        jLabel8 = new javax.swing.JLabel();
30         jTextField1 = new javax.swing.JTextField();
        jTextFieldGeneralIdentifierAgents = new javax.swing.
            JTextField();
32         jLabel11 = new javax.swing.JLabel();
        jLabel12 = new javax.swing.JLabel();
34         jLabel13 = new javax.swing.JLabel();
        jLabel14 = new javax.swing.JLabel();
36         jScrollPane2 = new javax.swing.JScrollPane();
        jTableIdentifier = new javax.swing.JTable();
38         jButtonTableIdentifierAdd = new javax.swing.JButton
            ();
        jButtonTableIdentifierDel = new javax.swing.JButton
            ();
40         jLabel15 = new javax.swing.JLabel();
        jPanel2 = new javax.swing.JPanel();
42         jLabel9 = new javax.swing.JLabel();
        jTextFieldTechLocation = new javax.swing.JTextField
            ();
44         jLabel10 = new javax.swing.JLabel();
        jPanel3 = new javax.swing.JPanel();
46         jLabel1 = new javax.swing.JLabel();

```

```

jLabel2 = new javax.swing.JLabel();
48  jLabel3 = new javax.swing.JLabel();
jLabel4 = new javax.swing.JLabel();
50  jScrollPane1 = new javax.swing.JScrollPane();
jTextArea1 = new javax.swing.JTextArea();
52  jLabel5 = new javax.swing.JLabel();
jSeparator1 = new javax.swing.JSeparator();
54  jLabel6 = new javax.swing.JLabel();
jLabel17 = new javax.swing.JLabel();
56  jLabel16 = new javax.swing.JLabel();
jLabel18 = new javax.swing.JLabel();
58  jSeparator2 = new javax.swing.JSeparator();
jLabel19 = new javax.swing.JLabel();
60  jLabel20 = new javax.swing.JLabel();

62  setPreferredSize(new java.awt.Dimension(1148, 744));

64  jPanel1.setToolTipText("");
jPanel1.setFont(new java.awt.Font("Tahoma", 0, 14));
    // NOI18N
66
jLabel7.setFont(new java.awt.Font("Tahoma", 1, 14));
    // NOI18N
68  jLabel7.setText("1. Identifier.");

70  jLabel8.setIcon(new javax.swing.ImageIcon(getClass()
    .getResource("/img/help.png"))); // NOI18N
jLabel8.setToolTipText("<html>A globally unique
    label that identifies this learning object.</
    html>");
72

jTextFieldGeneralIdentifierAgents.setEditable(false)
    ;
74  jTextFieldGeneralIdentifierAgents.setBackground(new
    java.awt.Color(204, 204, 204));
jTextFieldGeneralIdentifierAgents.setText("sem
    sugestoes");
76

jLabel11.setFont(new java.awt.Font("Tahoma", 1, 14))
    ; // NOI18N
78  jLabel11.setText("1.1. Catalog");

80  jLabel12.setFont(new java.awt.Font("Tahoma", 1, 14))
    ; // NOI18N
jLabel12.setText("1.2. Entry");
82

jLabel13.setIcon(new javax.swing.ImageIcon(getClass()
    .getResource("/img/help.png"))); // NOI18N
84  jLabel13.setToolTipText("<html>The name or
    designator of the identification or cataloging\
    nscheme for this entry. A namespace scheme.<br>
    <br>\ne.g: \"ISBN\", \"ARIADNE\", \"URI\"</html>")

```

```

    );

86     JLabel14.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
    JLabel14.setToolTipText("<html>\nThe value of the
        identifier within the identification<br>\n
        cataloging scheme that designates or identifies<
        br>\nthis learning object. A namespace specific
        string.<br>\n<br>\ne.g:  \"2-7342-0318\", \"
        LEAO875\",<br>\n\" http://www.ieee.org/documents
        /1234\"<n</html>");

88
    jTableIdentifier.setBorder(javax.swing.BorderFactory
        .createLineBorder(new java.awt.Color(0, 0, 153))
        );
90     jTableIdentifier.setFont(new java.awt.Font("Tahoma",
        0, 14)); // NOI18N
    jTableIdentifier.setModel(new javax.swing.table.
        DefaultTableModel(1,2));
92     jTableIdentifier.setAutoResizeMode(javax.swing.
        JTable.AUTO_RESIZE_LAST_COLUMN);
    jTableIdentifier.setRowHeight(24);
94     jTableIdentifier.setSelectionBackground(new java.awt
        .Color(199, 230, 251));
    jTableIdentifier.setSelectionForeground(new java.awt
        .Color(0, 0, 0));
96     jTableIdentifier.getTableHeader().
        setReorderingAllowed(false);
    jScrollPane2.setViewportViewView(jTableIdentifier);
98     jTableIdentifier.getColumnModel().getColumn(0).
        setResizable(false);
    jTableIdentifier.getColumnModel().getColumn(1).
        setResizable(false);

100
    jButtonTableIdentifierAdd.setText("+");
102     jButtonTableIdentifierAdd.addActionListener(new java
        .awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.
            ActionEvent evt) {
104         jButtonTableIdentifierAddActionPerformed(evt
            );
        }
106     });

108     jButtonTableIdentifierDel.setText("-");
    jButtonTableIdentifierDel.addActionListener(new java
        .awt.event.ActionListener() {
110         public void actionPerformed(java.awt.event.
            ActionEvent evt) {
            jButtonTableIdentifierDelActionPerformed(evt
            );
112     }

```



```

138         .addComponent(
                jLabel13)
        .addGap(168, 168,
                168)
140        .addComponent(
                jLabel12)
        .addPreferredGap(
                javax.swing.
                LayoutStyle.
                ComponentPlacement
                .RELATED)
142        .addComponent(
                jLabel14)
        .addGap(310, 310,
                310))))))
144        .addPreferredGap(javax.swing.
                LayoutStyle.ComponentPlacement.
                RELATED)
        .addGroup(jPanel1Layout.
                createParallelGroup(javax.swing.
                GroupLayout.Alignment.LEADING,
                false)
146                .addComponent(
                        jButtonTableIdentifierAdd,
                        javax.swing.GroupLayout.
                        DEFAULT_SIZE, javax.swing.
                        GroupLayout.DEFAULT_SIZE,
                        Short.MAX_VALUE)
                        .addComponent(
                                jButtonTableIdentifierDel,
                                javax.swing.GroupLayout.
                                DEFAULT_SIZE, javax.swing.
                                GroupLayout.DEFAULT_SIZE,
                                Short.MAX_VALUE))
148                .addContainerGap(358, Short.
                MAX_VALUE))
        .addGroup(jPanel1Layout.
                createSequentialGroup())
150        .addComponent(jTextField1, javax.
                swing.GroupLayout.PREFERRED_SIZE
                , 257, javax.swing.GroupLayout.
                PREFERRED_SIZE)
        .addPreferredGap(javax.swing.
                LayoutStyle.ComponentPlacement.
                RELATED)
152        .addComponent(
                jTextFieldGeneralIdentifierAgents
                , javax.swing.GroupLayout.
                PREFERRED_SIZE, 257, javax.swing
                .GroupLayout.PREFERRED_SIZE)
        .addContainerGap(javax.swing.
                GroupLayout.DEFAULT_SIZE, Short.

```

```

MAX.VALUE))
154 .addGroup(jPanel1Layout .
      createSequentialGroup()
      .addComponent(jLabel15)
156 .addGap(0, 0, Short.MAX.VALUE)))
);
158 jPanel1Layout.setVerticalGroup(
      jPanel1Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
160 .addGroup(jPanel1Layout.createSequentialGroup()
      .addGroup(jPanel1Layout.createParallelGroup(
        javax.swing.GroupLayout.Alignment.
162 LEADING)
      .addGroup(jPanel1Layout .
        createSequentialGroup()
        .addGap(18, 18, 18)
164 .addComponent(jLabel7))
      .addGroup(jPanel1Layout .
        createSequentialGroup()
166 .addContainerGap()
        .addComponent(jLabel8)))
168 .addPreferredGap(javax.swing.LayoutStyle.
      ComponentPlacement.RELATED)
      .addGroup(jPanel1Layout.createParallelGroup(
        javax.swing.GroupLayout.Alignment.
170 TRAILING)
      .addGroup(jPanel1Layout .
        createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
        .addComponent(jLabel11)
172 .addComponent(jLabel13))
      .addGroup(jPanel1Layout .
        createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
174 .addComponent(jLabel12)
        .addComponent(jLabel14)))
176 .addPreferredGap(javax.swing.LayoutStyle.
      ComponentPlacement.RELATED)
      .addGroup(jPanel1Layout.createParallelGroup(
        javax.swing.GroupLayout.Alignment.
178 LEADING)
      .addComponent(jScrollPane2, javax.swing.
        GroupLayout.PREFERRED.SIZE, 75,
        javax.swing.GroupLayout.
        PREFERRED.SIZE)
      .addGroup(jPanel1Layout .
        createSequentialGroup()
180 .addComponent(
          jButtonTableIdentifierAdd)
      .addPreferredGap(javax.swing.
        LayoutStyle.ComponentPlacement.
        RELATED))

```



```

182         .addComponent(
                jButtonTableIdentifierDel)))
        .addGap(259, 259, 259)
184        .addComponent(jLabel15)
        .addPreferredGap(javax.swing.LayoutStyle.
                ComponentPlacement.RELATED)
186        .addGroup(jPanel1Layout.createParallelGroup(
                javax.swing.GroupLayout.Alignment.
                BASELINE)
                .addComponent(jTextField1, javax.swing.
                        GroupLayout.PREFERRED_SIZE, javax.
                        swing.GroupLayout.DEFAULT_SIZE,
                        javax.swing.GroupLayout.
                        PREFERRED_SIZE)
188                .addComponent(
                        jTextFieldGeneralIdentifierAgents,
                        javax.swing.GroupLayout.
                        PREFERRED_SIZE, javax.swing.
                        GroupLayout.DEFAULT_SIZE, javax.
                        swing.GroupLayout.PREFERRED_SIZE))
                .addContainerGap(268, Short.MAX_VALUE))
190    );

192    jTabbedPane.addTab("<html><b>1. General</b></html>",
        jPanel1);

194    jPanel2.setFont(new java.awt.Font("Tahoma", 0, 14));
        // NOI18N

196    jLabel9.setFont(new java.awt.Font("Tahoma", 2, 14));
        // NOI18N
    jLabel9.setText("3. Location:");

198    jLabel10.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
200    jLabel10.setToolTipText("<html>LocalizaÃ§Ã£o fÃsica
        do objeto e seus conteÃºdos (ex.: URL, URI).<br
        ><br>\n Caso haja mais de uma localizaÃ§Ã£o
        separe os valores com ;</html>");

202    javax.swing.GroupLayout jPanel2Layout = new javax.
        swing.GroupLayout(jPanel2);
    jPanel2.setLayout(jPanel2Layout);
204    jPanel2Layout.setHorizontalGroup(
        jPanel2Layout.createParallelGroup(javax.swing.
                GroupLayout.Alignment.LEADING)
206        .addGroup(jPanel2Layout.createSequentialGroup()
                .addContainerGap()
208                .addComponent(jLabel9)
                .addPreferredGap(javax.swing.LayoutStyle.
                        ComponentPlacement.RELATED)
                .addComponent(jLabel10)
210

```

```

        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
212     .addComponent(jTextFieldTechLocation, javax.
            swing.GroupLayout.PREFERRED_SIZE, 544,
            javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(485, Short.MAX_VALUE))
214 );
jPanel2Layout.setVerticalGroup(
216     jPanel2Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup())
218     .addContainerGap()
        .addGroup(jPanel2Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.
                LEADING)
220     .addComponent(jLabel10)
        .addComponent(jTextFieldTechLocation,
            javax.swing.GroupLayout.
                PREFERRED_SIZE, javax.swing.
                GroupLayout.DEFAULT_SIZE, javax.
                swing.GroupLayout.PREFERRED_SIZE)
222     .addComponent(jLabel9))
        .addContainerGap(679, Short.MAX_VALUE))
224 );

jTabbedPane.addTab("<html><i>4. Technical</i></html>
226     ", jPanel2);

jLabel1.setFont(new java.awt.Font("Tahoma", 0, 18));
228     // NOI18N
jLabel1.setText("Legenda:");

230
jLabel2.setFont(new java.awt.Font("Tahoma", 2, 13));
232     // NOI18N
jLabel2.setText("– campo opcional");

234
jLabel3.setFont(new java.awt.Font("Tahoma", 1, 13));
236     // NOI18N
jLabel3.setText("– campo obrigatorio");

jLabel4.setFont(new java.awt.Font("Tahoma", 1, 18));
238     // NOI18N
jLabel4.setText("Perfil AdequaÃ§Ã£o (PM-ADEQ-BASE)")
    ;

240
jScrollPane1.setHorizontalScrollBarPolicy(javax.
    swing.ScrollPaneConstants.
        HORIZONTAL_SCROLLBAR_NEVER);

242
jTextArea1.setEditable(false);
jTextArea1.setColumns(20);
244
jTextArea1.setLineWrap(true);

```

```

jTextArea1.setRows(5);
246 jTextArea1.setText("Este perfil consiste em uma
    quantidade minima de informacoes que podem ser
    registradas a respeito de um determinado objeto.
    O uso pretendido deste perfil e apenas para
    situacoes onde nada mais se conhece a respeito
    de um dado objeto alem de seu identificador
    unico e sua localizacao (URL) em um dado
    repositorio. Essas situacoes podem ocorrer, por
    exemplo, no caso do armazenamento (possivelmente
    temporario) de objetos legados que ainda nao
    foram analisados.");
jTextArea1.setWrapStyleWord(true);
248 JScrollPane.setViewportView(jTextArea1);

jLabel5.setFont(new java.awt.Font("Tahoma", 1, 13));
// NOI18N
jLabel5.setText("Referencia:");

jLabel6.setText("Relatorio Tecnico RT-OBAA-01");

254 JLabel17.setFont(new java.awt.Font("Tahoma", 1, 14))
; // NOI18N
jLabel17.setText("1. General");

jLabel16.setIcon(new javax.swing.ImageIcon(getClass
().getResource("/img/help.png"))); // NOI18N
jLabel16.setToolTipText("<html>This category groups
the general information<br>\nthat describes this
learning object as a whole.</html>");
260

jLabel18.setFont(new java.awt.Font("Tahoma", 1, 18))
; // NOI18N
262 JLabel18.setText("Categorias:");

jLabel19.setFont(new java.awt.Font("Tahoma", 1, 14))
; // NOI18N
264 JLabel19.setText("4. Technical");

jLabel20.setIcon(new javax.swing.ImageIcon(getClass
().getResource("/img/help.png"))); // NOI18N
268 JLabel20.setToolTipText("<html>This category
describes the technical<br>\nrequirements and
characteristics of this<br>\nlearning object.</
html>");

270 javax.swing.GroupLayout jPanel3Layout = new javax.
swing.GroupLayout(jPanel3);
jPanel3.setLayout(jPanel3Layout);
272 jPanel3Layout.setHorizontalGroup(
jPanel3Layout.createParallelGroup(javax.swing.
GroupLayout.Alignment.LEADING)

```

```

274         .addGroup(jPanel3Layout.createSequentialGroup()
275             .addContainerGap()
276             .addGroup(jPanel3Layout.createParallelGroup(
277                 javax.swing.GroupLayout.Alignment.
278                 LEADING)
279             .addGroup(jPanel3Layout.
280                 createParallelGroup(javax.swing.
281                     GroupLayout.Alignment.LEADING, false
282                 )
283             .addComponent(jLabel1)
284             .addComponent(jLabel3)
285             .addComponent(jScrollPane1, javax.
286                 swing.GroupLayout.DEFAULT_SIZE,
287                 626, Short.MAX_VALUE)
288             .addComponent(jLabel4)
289             .addComponent(jSeparator1, javax.
290                 swing.GroupLayout.PREFERRED_SIZE
291                 , 379, javax.swing.GroupLayout.
292                 PREFERRED_SIZE)
293             .addComponent(jLabel2, javax.swing.
294                 GroupLayout.PREFERRED_SIZE, 116,
295                 javax.swing.GroupLayout.
296                 PREFERRED_SIZE)
297             .addGroup(jPanel3Layout.
298                 createSequentialGroup()
299                 .addGroup(jPanel3Layout.
300                     createParallelGroup(javax.
301                         swing.GroupLayout.Alignment.
302                         TRAILING, false)
303                     .addGroup(jPanel3Layout.
304                         createSequentialGroup()
305                         .addComponent(jLabel17)
306                         .addPreferredGap(javax.
307                             swing.LayoutStyle.
308                             ComponentPlacement.
309                             RELATED, javax.swing.
310                                 GroupLayout.
311                                 DEFAULT_SIZE, Short.
312                                 MAX_VALUE)
313                             .addComponent(jLabel16))
314                         .addComponent(jLabel5))
315                     .addPreferredGap(javax.swing.
316                         LayoutStyle.
317                         ComponentPlacement.RELATED)
318                     .addComponent(jLabel6))
319                     .addComponent(jLabel18)
320                     .addComponent(jSeparator2))
321             .addGroup(jPanel3Layout.
322                 createSequentialGroup()
323                 .addComponent(jLabel19)
324                 .addPreferredGap(javax.swing.
325                     LayoutStyle.ComponentPlacement.

```

```

298         RELATED)
        .addComponent(jLabel20)))
        .addContainerGap(505, Short.MAX_VALUE))
300    );
jPanel3Layout.setVerticalGroup(
302    jPanel3Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
        .addGroup(jPanel3Layout.createSequentialGroup())
304        .addGap(27, 27, 27)
        .addComponent(jLabel11)
306        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addComponent(jLabel3)
308        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addComponent(jLabel2)
310        .addGap(72, 72, 72)
        .addComponent(jSeparator1, javax.swing.
            GroupLayout.PREFERRED_SIZE, 10, javax.
                swing.GroupLayout.PREFERRED_SIZE)
312        .addGap(18, 18, 18)
        .addComponent(jLabel4)
314        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addComponent(jScrollPane1, javax.swing.
            GroupLayout.PREFERRED_SIZE, 243, javax.
                swing.GroupLayout.PREFERRED_SIZE)
316        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addGroup(jPanel3Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.
                BASELINE)
            .addComponent(jLabel5)
            .addComponent(jLabel6))
318        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.UNRELATED)
        .addComponent(jSeparator2, javax.swing.
            GroupLayout.PREFERRED_SIZE, 10, javax.
                swing.GroupLayout.PREFERRED_SIZE)
320        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addComponent(jLabel18)
322        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addGroup(jPanel3Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.
                TRAILING)
            .addComponent(jLabel17)
            .addComponent(jLabel16))
324        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
326        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
328        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)

```

```

        .addGroup(jPanel3Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.
330         TRAILING)
            .addComponent(jLabel19)
            .addComponent(jLabel20))
332         .addContainerGap(124, Short.MAX_VALUE))
    );

334     jTabbedPane.addTab("?", jPanel3);

336     javax.swing.GroupLayout layout = new javax.swing.
        GroupLayout(this);
338     this.setLayout(layout);
    layout.setHorizontalGroup(
340         layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
        .addComponent(jTabbedPane, javax.swing.
            GroupLayout.Alignment.TRAILING, javax.swing.
            GroupLayout.DEFAULT_SIZE, 1148, Short.
            MAX_VALUE)
342     );
    layout.setVerticalGroup(
344         layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
        .addComponent(jTabbedPane, javax.swing.
            GroupLayout.Alignment.TRAILING)
346     );

348     jTabbedPane.getAccessibleContext().setAccessibleName
        ("<html><b>?</b></html>");
    }// </editor-fold> //GEN-END: initComponents

350     private void jButtonTableIdentifierAddActionPerformed(
        java.awt.event.ActionEvent evt) { //GEN-FIRST:
        event_jButtonTableIdentifierAddActionPerformed
352         if(jTableIdentifier.getRowCount() == 10)
        {
354             JOptionPane.showMessageDialog(this, "VocÃª nÃ£o
                pode ter mais de 10 linhas na tabela!", "
                Aviso", JOptionPane.WARNING_MESSAGE);
            return;
356         }
        DefaultTableModel model = (DefaultTableModel)
            jTableIdentifier.getModel();
358         model.addRow(new String[2]);
    } //GEN-LAST:
        event_jButtonTableIdentifierAddActionPerformed

360     private void jButtonTableIdentifierDelActionPerformed(
        java.awt.event.ActionEvent evt) { //GEN-FIRST:
        event_jButtonTableIdentifierDelActionPerformed
362         if(jTableIdentifier.getRowCount() == 1)

```

```

364         {
            JOptionPane.showMessageDialog(this, "Você não
                pode ter deletar a última linha de uma
                tabela!", "Aviso", JOptionPane.
                WARNING.MESSAGE);
            return;
366     }
368     if (jTableIdentifier.getSelectedRow() == -1)
    {
        JOptionPane.showMessageDialog(this, "Você
            precisa selecionar a linha que deseja
            deletar!", "Aviso", JOptionPane.
            WARNING.MESSAGE);
370         return;
    }
372
    DefaultTableModel model = (DefaultTableModel)
        jTableIdentifier.getModel();
374     model.removeRow(jTableIdentifier.getSelectedRow());
} //GEN-LAST:
    event_jButtonTableIdentifierDelActionPerformed
376
// Variables declaration - do not modify //GEN-BEGIN:
    variables
378     private javax.swing.JButton jButtonTableIdentifierAdd;
380     private javax.swing.JButton jButtonTableIdentifierDel;
382     private javax.swing.JLabel jLabel1;
384     private javax.swing.JLabel jLabel10;
386     private javax.swing.JLabel jLabel11;
388     private javax.swing.JLabel jLabel12;
390     private javax.swing.JLabel jLabel13;
392     private javax.swing.JLabel jLabel14;
394     private javax.swing.JLabel jLabel15;
396     private javax.swing.JLabel jLabel16;
400     private javax.swing.JLabel jLabel17;
402     private javax.swing.JLabel jLabel18;
404     private javax.swing.JLabel jLabel19;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel20;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JLabel jLabel8;
    private javax.swing.JLabel jLabel9;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JPanel jPanel2;
    private javax.swing.JPanel jPanel3;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JScrollPane jScrollPane2;
    private javax.swing.JSeparator jSeparator1;

```

```

406     private javax.swing.JSeparator jSeparator2;
407     private javax.swing.JTabbedPane jTabbedPane;
408     private javax.swing.JTable jTableIdentifier;
409     private javax.swing.JTextArea jTextArea1;
410     private javax.swing.JTextField jTextField1;
411     private javax.swing.JTextField
        jTextFieldGeneralIdentifierAgents;
412     private javax.swing.JTextField jTextFieldTechLocation;
        // End of variables declaration//GEN-END:variables
414 }

```

B.4.5 ProfileOBAAIdeal.java

```

package user_interface;

2
import java.awt.GridLayout;
4 import user_interface.categorias.JPanelGeneralParaOBAA;
import user_interface.categorias.JPanelLifeCycleFull;
6 import user_interface.categorias.JPanelRightsFull;
import util.TipoDeSugestao;
8

public class ProfileOBAAIdeal extends javax.swing.JPanel
    implements Profile{

10
    public ProfileOBAAIdeal() {
12         initComponents();
        jPanelLifeCycle.setLayout(new GridLayout());
14         jPanelLifeCycle.add(new JPanelLifeCycleFull(this));
        jPanelGeneral.setLayout(new GridLayout());
16         jPanelGeneral.add(new JPanelGeneralParaOBAA(this));
        jPanelRights.setLayout(new GridLayout());
18         jPanelRights.add(new JPanelRightsFull(this));
    }

20
    public void gerarSugestoes(TipoDeSugestao tipo)
22     {
        // INCOMPLETO
24     }

26     @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated
        Code">//GEN-BEGIN: initComponents
28     private void initComponents() {

30         jTabbedPane = new javax.swing.JTabbedPane();
        jPanelGeneral = new javax.swing.JPanel();
32         jPanelLifeCycle = new javax.swing.JPanel();

```



```

jPanel8 = new javax.swing.JPanel();
34  jPanel3 = new javax.swing.JPanel();
jPanel4 = new javax.swing.JPanel();
36  jPanelRights = new javax.swing.JPanel();
jPanel6 = new javax.swing.JPanel();
38  jPanel7 = new javax.swing.JPanel();
jPanel9 = new javax.swing.JPanel();
40  jPanel10 = new javax.swing.JPanel();
jPanel12 = new javax.swing.JPanel();
42  jLabel1 = new javax.swing.JLabel();
jLabel2 = new javax.swing.JLabel();
44  jLabel3 = new javax.swing.JLabel();
jLabel4 = new javax.swing.JLabel();
46  jScrollPane1 = new javax.swing.JScrollPane();
jTextArea1 = new javax.swing.JTextArea();
48  jLabel5 = new javax.swing.JLabel();
jSeparator1 = new javax.swing.JSeparator();
50  jLabel6 = new javax.swing.JLabel();
jPanel11 = new javax.swing.JPanel();
52  jLabel7 = new javax.swing.JLabel();
jLabel8 = new javax.swing.JLabel();
54  jLabel9 = new javax.swing.JLabel();
jLabel10 = new javax.swing.JLabel();
56  jScrollPane2 = new javax.swing.JScrollPane();
jTextArea2 = new javax.swing.JTextArea();
58  jLabel11 = new javax.swing.JLabel();
jSeparator2 = new javax.swing.JSeparator();
60  jLabel12 = new javax.swing.JLabel();
jSeparator3 = new javax.swing.JSeparator();
62  jLabel18 = new javax.swing.JLabel();
jLabel17 = new javax.swing.JLabel();
64  jLabel19 = new javax.swing.JLabel();
jLabel16 = new javax.swing.JLabel();
66  jLabel20 = new javax.swing.JLabel();
jLabel21 = new javax.swing.JLabel();
68  jLabel22 = new javax.swing.JLabel();
jSeparator4 = new javax.swing.JSeparator();
70  jLabel25 = new javax.swing.JLabel();
jLabel26 = new javax.swing.JLabel();
72  jLabel27 = new javax.swing.JLabel();
jLabel28 = new javax.swing.JLabel();
74  jLabel29 = new javax.swing.JLabel();
jLabel30 = new javax.swing.JLabel();
76  jLabel31 = new javax.swing.JLabel();
jLabel32 = new javax.swing.JLabel();
78  jLabel33 = new javax.swing.JLabel();
jLabel34 = new javax.swing.JLabel();
80  jLabel35 = new javax.swing.JLabel();
jLabel36 = new javax.swing.JLabel();
82  jLabel37 = new javax.swing.JLabel();
jLabel38 = new javax.swing.JLabel();
84  jSeparator5 = new javax.swing.JSeparator();

```

```

86     javax.swing.GroupLayout jPanelGeneralLayout = new
        javax.swing.GroupLayout(jPanelGeneral);
88     jPanelGeneral.setLayout(jPanelGeneralLayout);
        jPanelGeneralLayout.setHorizontalGroup(
            jPanelGeneralLayout.createParallelGroup(javax.
                swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanelGeneralLayout.createSequentialGroup()
                    .addGap(0, 1167, Short.MAX_VALUE)
                )
            );
92     jPanelGeneralLayout.setVerticalGroup(
        jPanelGeneralLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanelGeneralLayout.createSequentialGroup()
                .addGap(0, 740, Short.MAX_VALUE)
            )
        );
96     jTablebedPane.addTab("<html><b>1. General</b></html>",
        jPanelGeneral);
98
        javax.swing.GroupLayout jPanelLifeCycleLayout = new
            javax.swing.GroupLayout(jPanelLifeCycle);
100     jPanelLifeCycle.setLayout(jPanelLifeCycleLayout);
        jPanelLifeCycleLayout.setHorizontalGroup(
            jPanelLifeCycleLayout.createParallelGroup(javax.
                swing.GroupLayout.Alignment.LEADING)
                .addGroup(jPanelLifeCycleLayout.createSequentialGroup()
                    .addGap(0, 1167, Short.MAX_VALUE)
                )
            );
104     jPanelLifeCycleLayout.setVerticalGroup(
        jPanelLifeCycleLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanelLifeCycleLayout.createSequentialGroup()
                .addGap(0, 740, Short.MAX_VALUE)
            )
        );
108
        jTablebedPane.addTab("<html><b>2. LifeCycle</b></html>
            ", jPanelLifeCycle);
110
        javax.swing.GroupLayout jPanel8Layout = new javax.
            swing.GroupLayout(jPanel8);
112     jPanel8.setLayout(jPanel8Layout);
        jPanel8Layout.setHorizontalGroup(
            jPanel8Layout.createParallelGroup(javax.swing.
                GroupLayout.Alignment.LEADING)
                .addGroup(jPanel8Layout.createSequentialGroup()
                    .addGap(0, 1167, Short.MAX_VALUE)
                )
            );
116     jPanel8Layout.setVerticalGroup(
        jPanel8Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
            .addGroup(jPanel8Layout.createSequentialGroup()
                .addGap(0, 740, Short.MAX_VALUE)
            )
        );
120
        jTablebedPane.addTab("<html><i>3. Meta-Metadate</i></
            html>", jPanel8);
122
124

```

```

javax.swing.GroupLayout jPanel3Layout = new javax.
    swing.GroupLayout(jPanel3);
126 jPanel3.setLayout(jPanel3Layout);
    jPanel3Layout.setHorizontalGroup(
128        jPanel3Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
            .addGap(0, 1167, Short.MAX_VALUE)
130        );
    jPanel3Layout.setVerticalGroup(
132        jPanel3Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
            .addGap(0, 740, Short.MAX_VALUE)
134        );
136 jPanel3.addTab("<html><b>4. Technical</b></html>
    ", jPanel3);

javax.swing.GroupLayout jPanel4Layout = new javax.
    swing.GroupLayout(jPanel4);
138 jPanel4.setLayout(jPanel4Layout);
    jPanel4Layout.setHorizontalGroup(
140        jPanel4Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
            .addGap(0, 1167, Short.MAX_VALUE)
142        );
    jPanel4Layout.setVerticalGroup(
144        jPanel4Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
            .addGap(0, 740, Short.MAX_VALUE)
146        );
148 jPanel4.addTab("<html><b>5. Educational</b></
    html>", jPanel4);
150
javax.swing.GroupLayout jPanelRightsLayout = new
    javax.swing.GroupLayout(jPanelRights);
152 jPanelRights.setLayout(jPanelRightsLayout);
    jPanelRightsLayout.setHorizontalGroup(
154        jPanelRightsLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 1167, Short.MAX_VALUE)
156        );
    jPanelRightsLayout.setVerticalGroup(
158        jPanelRightsLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 740, Short.MAX_VALUE)
160        );
162 jPanelRights.addTab("<html><i>6. Rights</i></html>",
    jPanelRights);

```

```

164 javax.swing.GroupLayout jPanel6Layout = new javax.
    swing.GroupLayout(jPanel6);
    jPanel6.setLayout(jPanel6Layout);
166 jPanel6Layout.setHorizontalGroup(
    jPanel6Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
168     .addGap(0, 1167, Short.MAX_VALUE)
    );
170 jPanel6Layout.setVerticalGroup(
    jPanel6Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
172     .addGap(0, 740, Short.MAX_VALUE)
    );
174
jTabbedPane.addTab("<html><i>7. Relation </i></html>"
    , jPanel6);
176
javax.swing.GroupLayout jPanel7Layout = new javax.
    swing.GroupLayout(jPanel7);
178 jPanel7.setLayout(jPanel7Layout);
jPanel7Layout.setHorizontalGroup(
    jPanel7Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
180     .addGap(0, 1167, Short.MAX_VALUE)
    );
182 jPanel7Layout.setVerticalGroup(
    jPanel7Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
184     .addGap(0, 740, Short.MAX_VALUE)
    );
186
jTabbedPane.addTab("<html><i>8. Annotation </i></html>"
    , jPanel7);
188
javax.swing.GroupLayout jPanel9Layout = new javax.
    swing.GroupLayout(jPanel9);
190 jPanel9.setLayout(jPanel9Layout);
jPanel9Layout.setHorizontalGroup(
    jPanel9Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
192     .addGap(0, 1167, Short.MAX_VALUE)
    );
194 jPanel9Layout.setVerticalGroup(
    jPanel9Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
196     .addGap(0, 740, Short.MAX_VALUE)
    );
198
jTabbedPane.addTab("<html><b>10. Acessibility </b></html>"
    , jPanel9);
200
202

```

```

javax.swing.GroupLayout jPanel10Layout = new javax.
    swing.GroupLayout(jPanel10);
204 jPanel10.setLayout(jPanel10Layout);
    jPanel10Layout.setHorizontalGroup(
206         jPanel10Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
            .addGroup(0, 1167, Short.MAX_VALUE)
208     );
    jPanel10Layout.setVerticalGroup(
210         jPanel10Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
            .addGroup(0, 740, Short.MAX_VALUE)
212     );

214 jTablebedPane.addTab("<html><i>11. Segmentation</i></html>", jPanel10);

216 jLabel1.setFont(new java.awt.Font("Tahoma", 0, 18));
    // NOI18N
    jLabel1.setText("Legenda:");
218

    jLabel2.setFont(new java.awt.Font("Tahoma", 2, 13));
    // NOI18N
220    jLabel2.setText("— campo opcional");

    jLabel3.setFont(new java.awt.Font("Tahoma", 1, 13));
    // NOI18N
222    jLabel3.setText("— campo obrigatorio");
224

    jLabel4.setFont(new java.awt.Font("Tahoma", 1, 18));
    // NOI18N
226    jLabel4.setText("Perfil Compativel Ideal com OBAA (
        PM-OBAA-FULL)");

228 jScrollPane1.setHorizontalScrollBarPolicy(javax.
    swing.ScrollPaneConstants.
        HORIZONTALSCROLLBAR_NEVER);

230    jTextArea1.setEditable(false);
    jTextArea1.setColumns(20);
232    jTextArea1.setLineWrap(true);
    jTextArea1.setRows(5);
234    jTextArea1.setText("O Perfil PM-OBAA-FULL consiste
        na quantidade ideal de informacoes que devem ser
        registradas sobre um objeto para que o OA
        contemple as diversas aplicacoes para TV Digital
        , dispositivos moveis e Web, garantindo
        conversoes de conteudo e interoperabilidade.");
    jTextArea1.setWrapStyleWord(true);
236    jScrollPane1.setViewportView(jTextArea1);

```

```

238     jLabel5.setFont(new java.awt.Font("Tahoma", 1, 13));
        // NOI18N
jLabel5.setText("Referencia:");
240
242     jLabel6.setText("Relatorio Tecnico RT-OBAA-01");
244
246     jLabel7.setFont(new java.awt.Font("Tahoma", 0, 18));
        // NOI18N
jLabel7.setText("Legenda:");
248
246     jLabel8.setFont(new java.awt.Font("Tahoma", 2, 13));
        // NOI18N
jLabel8.setText("- campo opcional");
248
248     jLabel9.setFont(new java.awt.Font("Tahoma", 1, 13));
        // NOI18N
jLabel9.setText("- campo obrigatorio");
250
252     jLabel10.setFont(new java.awt.Font("Tahoma", 1, 18))
        ; // NOI18N
jLabel10.setText("Perfil Compativel Minimo com
        OBAA (PM-OBAA-CORE)");
254
jScrollPane2.setHorizontalScrollBarPolicy(javax.
        swing.ScrollPaneConstants.
        HORIZONTAL_SCROLLBAR_NEVER);
256
jTextArea2.setEditable(false);
258     jLabel2.setColumns(20);
jTextArea2.setLineWrap(true);
260     jLabel2.setRows(5);
jTextArea2.setText("O perfil PM-OBAA-CORE consiste
        no conjunto de metadados minimos considerados
        essenciais para todo objeto de aprendizagem
        descrito conforme o padrao OBAA. A utilizacao
        desse conjunto visa integracao com os OAs
        legados, com reduzido esforco humano.");
262     jLabel2.setWrapStyleWord(true);
jScrollPane2.setViewportView(jTextArea2);
264
264     jLabel11.setFont(new java.awt.Font("Tahoma", 1, 13))
        ; // NOI18N
jLabel11.setText("Referencia:");
266
268     jLabel12.setText("Relatorio Tecnico RT-OBAA-01");
270
270     jLabel18.setFont(new java.awt.Font("Tahoma", 1, 18))
        ; // NOI18N
jLabel18.setText("Categorias:");
272
jLabel17.setFont(new java.awt.Font("Tahoma", 1, 14))
        ; // NOI18N

```

```

274     jLabel17.setText("1. General");

276     jLabel19.setFont(new java.awt.Font("Tahoma", 1, 14))
        ; // NOI18N
    jLabel19.setText("2. LifeCycle");

278     jLabel16.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
280     jLabel16.setToolTipText("<html>This category groups
        the general information<br>\nthat describes this
        learning object as a whole.</html>");

282     jLabel20.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
    jLabel20.setToolTipText("<html>This category
        describes the history and<br>\ncurrent state of
        this learning object and those<br>\nentities
        that have affected this learning object<br>\
        nduring its evolution.</html>");

284     jLabel21.setFont(new java.awt.Font("Tahoma", 1, 14))
        ; // NOI18N
286     jLabel21.setText("4. Technical");

288     jLabel22.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
    jLabel22.setToolTipText("<html>This category
        describes the technical<br>\nrequirements and
        characteristics of this<br>\nlearning object.</
        html>");

290     jSeparator4.setOrientation(javax.swing.
        SwingConstants.VERTICAL);

292     jLabel25.setFont(new java.awt.Font("Tahoma", 1, 14))
        ; // NOI18N
294     jLabel25.setText("5. Educational");

296     jLabel26.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
    jLabel26.setToolTipText("<html>This category
        describes the key educational<br>\nor pedagogic
        characteristics of this learning<br>\nobject.</
        html>");

298     jLabel27.setFont(new java.awt.Font("Tahoma", 1, 14))
        ; // NOI18N
300     jLabel27.setText("6. Rights");

302     jLabel28.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N

```

```

jLabel28.setToolTipText("<html>This category
    describes the intellectual<br>\nproperty rights
    and conditions of use for this<br>\nlearning
    object.</html>");
304
jLabel29.setFont(new java.awt.Font("Tahoma", 1, 14))
    ; // NOI18N
306
jLabel29.setText("7. Relation");

jLabel30.setIcon(new javax.swing.ImageIcon(getClass
    ().getResource("/img/help.png"))); // NOI18N
jLabel30.setToolTipText("<html>This category defines
    the relationship<br>\nbetween this learning
    object and other<br>\nlearning objects, if any.<
    br><br>\nTo define multiple relationships, there
    may<br>\nbe multiple instances of this category
    . If there<br>\nis more than one target learning
    object, then<br>\neach target shall have a new
    relationship<br>\ninstance.</html>");
310
jLabel31.setFont(new java.awt.Font("Tahoma", 1, 14))
    ; // NOI18N
312
jLabel31.setText("8. Annotation");

jLabel32.setIcon(new javax.swing.ImageIcon(getClass
    ().getResource("/img/help.png"))); // NOI18N
jLabel32.setToolTipText("<html>This category
    provides comments on the<br>\neducational use of
    this learning object, and<br>\ninformation on
    when and by whom the<br>\ncomments were created
    .<br><br>\nThis category enables educators to
    share their<br>\nassessments of learning objects
    , suggestions<br>\nfor use, etc.</html>");
316
jLabel33.setFont(new java.awt.Font("Tahoma", 1, 14))
    ; // NOI18N
318
jLabel33.setText("3. Meta—Metadata");

jLabel34.setIcon(new javax.swing.ImageIcon(getClass
    ().getResource("/img/help.png"))); // NOI18N
jLabel34.setToolTipText("<html>This category
    describes this metadata record<br>\nitself (
    rather than the learning object that this<br>\n
    record describes).<br><br>\nThis category
    describes how the metadata<br>\ninstance can be
    identified, who created this<br>\nmetadata
    instance, how, when, and with what<br>\n
    references.</html>");
322
jLabel35.setFont(new java.awt.Font("Tahoma", 1, 14))
    ; // NOI18N
324
jLabel35.setText("10. Acessibility");

```



```

326     jLabel36.setFont(new java.awt.Font("Tahoma", 1, 14))
        ; // NOI18N
jLabel36.setText("11. Segmentation");
328
jLabel37.setIcon(new javax.swing.ImageIcon(getClass
    ().getResource("/img/help.png"))); // NOI18N
330 jLabel37.setToolTipText("<html>Grupo de metadados
        com informacoes de<br>\nsegmentacao dos
        conteudos do objeto.</html>");

332 jLabel38.setIcon(new javax.swing.ImageIcon(getClass
    ().getResource("/img/help.png"))); // NOI18N
jLabel38.setToolTipText("<html>Grupo de metadados
        com informacoes<br>\nde acessibilidade do objeto
        .</html>");
334
jSeparator5.setOrientation(javax.swing.
    SwingConstants.VERTICAL);
336
javax.swing.GroupLayout jPanel11Layout = new javax.
    swing.GroupLayout(jPanel11);
338 jPanel11.setLayout(jPanel11Layout);
jPanel11Layout.setHorizontalGroup(
340     jPanel11Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
        .addGroup(jPanel11Layout.createSequentialGroup()
            .addContainerGap()
            .addGroup(jPanel11Layout.createParallelGroup(
                javax.swing.
342                 GroupLayout.Alignment.LEADING)
                .addComponent(jLabel7)
                .addComponent(jLabel9)
                .addComponent(jLabel8, javax.swing.
                    GroupLayout.PREFERRED_SIZE, 116,
                    javax.swing.GroupLayout.
                        PREFERRED_SIZE)
                .addComponent(jLabel10)
                .addComponent(jScrollPane2, javax.swing.
                    GroupLayout.PREFERRED_SIZE, 626,
                    javax.swing.GroupLayout.
                        PREFERRED_SIZE)
                .addGroup(jPanel11Layout.
                    createSequentialGroup()
350                     .addComponent(jLabel11)
                     .addPreferredGap(javax.swing.
                         LayoutStyle.ComponentPlacement.
                             RELATED)
                     .addComponent(jLabel12))
                .addComponent(jSeparator2, javax.swing.
                    GroupLayout.PREFERRED_SIZE, 379,
                    javax.swing.GroupLayout.

```



```

368                                     ()
                                     .
                                     .addComponent
                                     (
                                     .jLabel19
                                     )
                                     .
                                     .addPreferredGap
                                     (javax.
                                     swing.
                                     LayoutStyle
                                     .
                                     ComponentPlacemen
                                     .RELATED
                                     )
370                                     .
                                     .addComponent
                                     (
                                     .jLabel20
                                     ))
                                     .addGroup(
                                     .jPanel11Layout
                                     .
                                     .createSequentialGroup
                                     ()
372                                     .
                                     .addComponent
                                     (
                                     .jLabel21
                                     )
                                     .
                                     .addPreferredGap
                                     (javax.
                                     swing.
                                     LayoutStyle
                                     .
                                     ComponentPlacemen
                                     .RELATED
                                     )
374                                     .
                                     .addComponent
                                     (
                                     .jLabel22
                                     ))
                                     .addGroup(
                                     .jPanel11Layout
                                     .
                                     .createSequentialGroup
                                     ()
376                                     .
                                     .addComponent
                                     (

```

```

jLabel33
)
.
addPreferredGap
(javax.swing.
LayoutStyle
.
ComponentPlacement
.RELATED
)
378 .
addComponent
(
jLabel34
))
.addPreferredGap(
javax.swing.
LayoutStyle.
ComponentPlacement
.RELATED, javax.swing.
GroupLayout.
DEFAULT_SIZE,
Short.MAX_VALUE)
))
380 .addComponent(jSeparator4,
javax.swing.GroupLayout.
PREFERRED_SIZE, 16,
javax.swing.GroupLayout.
PREFERRED_SIZE)
.addPreferredGap(javax.swing
.LayoutStyle.
ComponentPlacement.
RELATED)
382 .addGroup(jPanel11Layout.
createParallelGroup(
javax.swing.GroupLayout.
Alignment.LEADING)
.addGroup(jPanel11Layout
.
createSequentialGroup
())
384 .addComponent(
jLabel25)
.addPreferredGap(
javax.swing.
LayoutStyle.
ComponentPlacement
.RELATED)
386 .addComponent(
jLabel26))

```

```

        .addGroup(jPanel11Layout
            .createSequentialGroup
            ()
388         .addComponent(
            jLabel27)
        .addPreferredGap(
            javax.swing.
            LayoutStyle.
            ComponentPlacement
            .RELATED)
390         .addComponent(
            jLabel28))
        .addGroup(jPanel11Layout
            .createSequentialGroup
            ()
392         .addComponent(
            jLabel29)
        .addPreferredGap(
            javax.swing.
            LayoutStyle.
            ComponentPlacement
            .RELATED)
394         .addComponent(
            jLabel30))
        .addGroup(jPanel11Layout
            .createSequentialGroup
            ()
396         .addComponent(
            jLabel31)
        .addPreferredGap(
            javax.swing.
            LayoutStyle.
            ComponentPlacement
            .RELATED)
398         .addComponent(
            jLabel32)))
        .addGap(11, 11, 11)))
400 .addGap(18, 18, 18)
        .addComponent(jSeparator5, javax.
            swing.GroupLayout.PREFERRED_SIZE
            , 16, javax.swing.GroupLayout.
            PREFERRED_SIZE)
402 .addPreferredGap(javax.swing.
            LayoutStyle.ComponentPlacement.
            RELATED)
        .addGroup(jPanel11Layout.
            createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)

```

```

404         .addGroup(jPanel11Layout .
        createSequentialGroup ()
        .addComponent (jLabel35)
406         .addPreferredGap (javax . swing
        . LayoutStyle .
        ComponentPlacement .
        RELATED)
        .addComponent (jLabel38))
408     .addGroup (jPanel11Layout .
        createSequentialGroup ()
        .addComponent (jLabel36)
410         .addPreferredGap (javax . swing
        . LayoutStyle .
        ComponentPlacement .
        RELATED)
        .addComponent (jLabel37)))))
412     .addContainerGap (505 , Short . MAX . VALUE))
);
414 jPanel11Layout . setVerticalGroup (
    jPanel11Layout . createParallelGroup (javax . swing .
        GroupLayout . Alignment . LEADING)
416     .addGroup (jPanel11Layout . createSequentialGroup ()
        .addGap (27 , 27 , 27)
418         .addComponent (jLabel7)
        .addPreferredGap (javax . swing . LayoutStyle .
            ComponentPlacement . RELATED)
420         .addComponent (jLabel9)
        .addPreferredGap (javax . swing . LayoutStyle .
            ComponentPlacement . RELATED)
422         .addComponent (jLabel8)
        .addGap (72 , 72 , 72)
424         .addComponent (jSeparator2 , javax . swing .
            GroupLayout . PREFERRED . SIZE , 10 , javax .
            swing . GroupLayout . PREFERRED . SIZE)
        .addGap (18 , 18 , 18)
426         .addComponent (jLabel10)
        .addPreferredGap (javax . swing . LayoutStyle .
            ComponentPlacement . RELATED)
428         .addComponent (jScrollPane2 , javax . swing .
            GroupLayout . PREFERRED . SIZE , 243 , javax .
            swing . GroupLayout . PREFERRED . SIZE)
        .addPreferredGap (javax . swing . LayoutStyle .
            ComponentPlacement . RELATED)
430         .addGroup (jPanel11Layout . createParallelGroup
            (javax . swing . GroupLayout . Alignment .
            BASELINE)
            .addComponent (jLabel11)
            .addComponent (jLabel12))
432         .addPreferredGap (javax . swing . LayoutStyle .
            ComponentPlacement . UNRELATED)
        .addComponent (jSeparator3 , javax . swing .
            GroupLayout . PREFERRED . SIZE , 10 , javax .

```

```

        swing.GroupLayout.PREFERRED_SIZE)
    .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.RELATED)
436    .addComponent(jLabel18)
    .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.RELATED)
438    .addGroup(jPanel11Layout.createParallelGroup
        (javax.swing.GroupLayout.Alignment.
            LEADING)
        .addGroup(jPanel11Layout.
            createParallelGroup(javax.swing.
                GroupLayout.Alignment.LEADING, false
440            )
            .addComponent(jSeparator4)
            .addGroup(jPanel11Layout.
                createSequentialGroup()
442                .addGroup(jPanel11Layout.
                    createParallelGroup(javax.
                        swing.GroupLayout.Alignment.
                            TRAILING)
                    .addComponent(jLabel25)
444                    .addComponent(jLabel26))
                .addPreferredGap(javax.swing.
                    LayoutStyle.
                        ComponentPlacement.RELATED)
446                .addGroup(jPanel11Layout.
                    createParallelGroup(javax.
                        swing.GroupLayout.Alignment.
                            TRAILING)
                    .addComponent(jLabel27)
448                    .addComponent(jLabel28))
                .addPreferredGap(javax.swing.
                    LayoutStyle.
                        ComponentPlacement.RELATED)
450                .addGroup(jPanel11Layout.
                    createParallelGroup(javax.
                        swing.GroupLayout.Alignment.
                            TRAILING)
                    .addComponent(jLabel29)
452                    .addComponent(jLabel30))
                .addPreferredGap(javax.swing.
                    LayoutStyle.
                        ComponentPlacement.RELATED,
                        javax.swing.GroupLayout.
                            DEFAULT_SIZE, Short.
                                MAX_VALUE)
454                .addGroup(jPanel11Layout.
                    createParallelGroup(javax.
                        swing.GroupLayout.Alignment.
                            TRAILING)
                    .addComponent(jLabel31)
456                    .addComponent(jLabel32)))

```

```

    .addGroup(jPanel11Layout .
        createSequentialGroup ()
458    .addGroup(jPanel11Layout .
        createParallelGroup (javax .
            swing . GroupLayout . Alignment .
                TRAILING)
            .addComponent (jLabel17)
460            .addComponent (jLabel16))
        .addPreferredGap (javax . swing .
            LayoutStyle .
                ComponentPlacement .RELATED)
462    .addGroup(jPanel11Layout .
        createParallelGroup (javax .
            swing . GroupLayout . Alignment .
                TRAILING)
            .addComponent (jLabel19)
464            .addComponent (jLabel20))
        .addPreferredGap (javax . swing .
            LayoutStyle .
                ComponentPlacement .RELATED)
466    .addGroup(jPanel11Layout .
        createParallelGroup (javax .
            swing . GroupLayout . Alignment .
                TRAILING)
            .addComponent (jLabel33)
468            .addComponent (jLabel34))
        .addPreferredGap (javax . swing .
            LayoutStyle .
                ComponentPlacement .RELATED,
                javax . swing . GroupLayout .
                    DEFAULT_SIZE, Short .
                        MAX_VALUE)
470    .addGroup(jPanel11Layout .
        createParallelGroup (javax .
            swing . GroupLayout . Alignment .
                TRAILING)
            .addComponent (jLabel21)
472            .addComponent (jLabel22))))
    .addGroup(jPanel11Layout .
        createParallelGroup (javax . swing .
            GroupLayout . Alignment .LEADING, false
        )
474    .addComponent (jSeparator5)
        .addGroup(jPanel11Layout .
            createSequentialGroup ()
476            .addGroup(jPanel11Layout .
                createParallelGroup (javax .
                    swing . GroupLayout . Alignment .
                        TRAILING)
                    .addComponent (jLabel35)
478                    .addComponent (jLabel38))
            )
        )

```



```

        .addPreferredGap(javax.swing.
            LayoutStyle.
                ComponentPlacement.RELATED)
480        .addGroup(jPanel11Layout.
            createParallelGroup(javax.
                swing.GroupLayout.Alignment.
                    TRAILING)
                .addComponent(jLabel36)
482                .addComponent(jLabel37))
            .addGap(48, 48, 48)))
484        .addContainerGap(76, Short.MAX_VALUE))
    );
486
    javax.swing.GroupLayout jPanel12Layout = new javax.
        swing.GroupLayout(jPanel12);
488    jPanel12.setLayout(jPanel12Layout);
    jPanel12Layout.setHorizontalGroup(
490        jPanel12Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
            .addGroup(jPanel12Layout.createSequentialGroup()
492                .addContainerGap()
                .addGroup(jPanel12Layout.createParallelGroup
                    (javax.swing.GroupLayout.Alignment.
                        LEADING)
                    .addComponent(jLabel1)
                    .addComponent(jLabel3)
494                    .addComponent(jLabel2, javax.swing.
                        GroupLayout.PREFERRED_SIZE, 116,
                        javax.swing.GroupLayout.
                            PREFERRED_SIZE)
                    .addComponent(jLabel4)
496                    .addComponent(jScrollPane1, javax.swing.
                        GroupLayout.PREFERRED_SIZE, 626,
                        javax.swing.GroupLayout.
                            PREFERRED_SIZE)
                    .addGroup(jPanel12Layout.
                        createSequentialGroup()
                        .addComponent(jLabel5)
                        .addPreferredGap(javax.swing.
                            LayoutStyle.ComponentPlacement.
                                RELATED)
                        .addComponent(jLabel6))
500                    .addComponent(jSeparator1, javax.swing.
                        GroupLayout.PREFERRED_SIZE, 379,
                        javax.swing.GroupLayout.
                            PREFERRED_SIZE))
                .addContainerGap(529, Short.MAX_VALUE))
502        .addGroup(jPanel12Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel12Layout.
504                createSequentialGroup()
                .addContainerGap()
506                .addGroup(jPanel12Layout.
                    createSequentialGroup()
                    .addContainerGap()

```

```

508         .addComponent(jPanel11, javax.swing.
            GroupLayout.PREFERRED_SIZE, javax.
            swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.
            PREFERRED_SIZE)
        .addContainerGap(javax.swing.GroupLayout
            .DEFAULT_SIZE, Short.MAX_VALUE)))
510     );
    JPanel12Layout.setVerticalGroup(
512         JPanel12Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
        .addGroup(jPanel12Layout.createSequentialGroup())
514         .addGap(27, 27, 27)
        .addComponent(jLabel1)
516         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addComponent(jLabel3)
518         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addComponent(jLabel2)
520         .addGap(72, 72, 72)
        .addComponent(jSeparator1, javax.swing.
            GroupLayout.PREFERRED_SIZE, 10, javax.
            swing.GroupLayout.PREFERRED_SIZE)
522         .addGap(18, 18, 18)
        .addComponent(jLabel4)
524         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addComponent(jScrollPane1, javax.swing.
            GroupLayout.PREFERRED_SIZE, 243, javax.
            swing.GroupLayout.PREFERRED_SIZE)
526         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addGroup(jPanel12Layout.createParallelGroup
            (javax.swing.GroupLayout.Alignment.
            BASELINE)
528         .addComponent(jLabel5)
        .addComponent(jLabel6))
530         .addContainerGap(250, Short.MAX_VALUE))
        .addGroup(jPanel12Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.LEADING)
532         .addGroup(jPanel12Layout.
            createSequentialGroup())
        .addContainerGap()
534         .addComponent(jPanel11, javax.swing.
            GroupLayout.PREFERRED_SIZE, javax.
            swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.
            PREFERRED_SIZE)
        .addContainerGap(javax.swing.GroupLayout
            .DEFAULT_SIZE, Short.MAX_VALUE)))
536     );

```

```

538         jTabbedPane.addTab("?", jPanel12);

540         javax.swing.GroupLayout layout = new javax.swing.
            GroupLayout(this);
        this.setLayout(layout);
542         layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.
                GroupLayout.Alignment.LEADING)
544             .addComponent(jTabbedPane, javax.swing.
                GroupLayout.Alignment.TRAILING)
        );
546         layout.setVerticalGroup(
            layout.createParallelGroup(javax.swing.
                GroupLayout.Alignment.LEADING)
548             .addComponent(jTabbedPane, javax.swing.
                GroupLayout.Alignment.TRAILING)
        );
550     }// </editor-fold>//GEN-END:initComponents
    // Variables declaration - do not modify//GEN-BEGIN:
    variables
552     private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel10;
554     private javax.swing.JLabel jLabel11;
    private javax.swing.JLabel jLabel12;
556     private javax.swing.JLabel jLabel16;
    private javax.swing.JLabel jLabel17;
558     private javax.swing.JLabel jLabel18;
    private javax.swing.JLabel jLabel19;
560     private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel20;
562     private javax.swing.JLabel jLabel21;
    private javax.swing.JLabel jLabel22;
564     private javax.swing.JLabel jLabel25;
    private javax.swing.JLabel jLabel26;
566     private javax.swing.JLabel jLabel27;
    private javax.swing.JLabel jLabel28;
568     private javax.swing.JLabel jLabel29;
    private javax.swing.JLabel jLabel3;
570     private javax.swing.JLabel jLabel30;
    private javax.swing.JLabel jLabel31;
572     private javax.swing.JLabel jLabel32;
    private javax.swing.JLabel jLabel33;
574     private javax.swing.JLabel jLabel34;
    private javax.swing.JLabel jLabel35;
576     private javax.swing.JLabel jLabel36;
    private javax.swing.JLabel jLabel37;
578     private javax.swing.JLabel jLabel38;
    private javax.swing.JLabel jLabel4;
580     private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
582     private javax.swing.JLabel jLabel7;

```

```

private javax.swing.JLabel jLabel8;
584 private javax.swing.JLabel jLabel9;
private javax.swing.JPanel jPanel10;
586 private javax.swing.JPanel jPanel11;
private javax.swing.JPanel jPanel12;
588 private javax.swing.JPanel jPanel13;
private javax.swing.JPanel jPanel14;
590 private javax.swing.JPanel jPanel16;
private javax.swing.JPanel jPanel17;
592 private javax.swing.JPanel jPanel18;
private javax.swing.JPanel jPanel19;
594 private javax.swing.JPanel jPanelGeneral;
private javax.swing.JPanel jPanelLifeCycle;
596 private javax.swing.JPanel jPanelRights;
private javax.swing.JScrollPane jScrollPane1;
598 private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JSeparator jSeparator1;
600 private javax.swing.JSeparator jSeparator2;
private javax.swing.JSeparator jSeparator3;
602 private javax.swing.JSeparator jSeparator4;
private javax.swing.JSeparator jSeparator5;
604 private javax.swing.JTabbedPane jTabbedPane;
private javax.swing.JTextArea jTextArea1;
606 private javax.swing.JTextArea jTextArea2;
// End of variables declaration//GEN-END:variables
608 }

```

B.4.6 ProfileOBAAMinimo.java

```

1 package user_interface;

3 import java.awt.*;
import java.util.ArrayList;
5 import user_interface.categorias.Categoria;
import user_interface.categorias.JPanelGeneralParaOBAA;
7 import user_interface.categorias.JPanelLifeCycleFull;
import user_interface.categorias.JPanelRightsFull;
9 import user_interface.categorias.PacoteComCampos;
import util.TipoDeSugestao;

11 public class ProfileOBAAMinimo extends javax.swing.JPanel
implements Profile {

13     ArrayList<Categoria> categorias;

15     public ProfileOBAAMinimo() {
17         initComponents();

```

```

19      categorias = new ArrayList<Categoria>();

21      jPanelGeneral.setLayout(new GridLayout());
      JPanelGeneralParaOBAA generalParaOBAA = new
          JPanelGeneralParaOBAA(this);
23      jPanelGeneral.add(generalParaOBAA);
      categorias.add(generalParaOBAA);

25

27      jPanelLifeCycle.setLayout(new GridLayout());
      JPanelLifeCycleFull lifeCycleFull = new
          JPanelLifeCycleFull(this);
      jPanelLifeCycle.add(lifeCycleFull);
29      categorias.add(lifeCycleFull);

31      jPanelRights.setLayout(new GridLayout());
      JPanelRightsFull rightsFull = new JPanelRightsFull(
          this);
33      jPanelRights.add(rightsFull);
      categorias.add(rightsFull);
35  }

37  @Override
      public void gerarSugestoes(TipoDeSugestao tipo)
39  {
          ArrayList<PacoteComCampos> valores_para_sugestoes =
              new ArrayList<PacoteComCampos>();
41          for(Categoria c: categorias)
              valores_para_sugestoes.add(c.
                  retornarCamposGeradoresDeSugestoes());

43          boolean valido = false;
45          for(PacoteComCampos pacote: valores_para_sugestoes)
          {
47              for(String s: pacote.valores)
              {
49                  if(!s.isEmpty())
                  {
51                      valido = true;
                      break;
53                  }
              }
              if(valido)
55                  break;
57          }

59          if(!valido)
          {
61              javax.swing.JOptionPane.showMessageDialog(this,
                  "At least one of the fields must be filled!"
                  , "Warning", javax.swing.JOptionPane.
                      ERROR_MESSAGE);

```

```

        return;
    }

    TelaSugestoes tela = new TelaSugestoes(tipo, this);

    Main.control.obterSugestoes(tela.getTextArea(),
        valores_para_sugestoes, tipo);
}

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated
    Code">//GEN-BEGIN: initComponents
private void initComponents() {

    JTabbedPane = new javax.swing.JTabbedPane();
    JPanelGeneral = new javax.swing.JPanel();
    JPanelLifecycle = new javax.swing.JPanel();
    JPanel3 = new javax.swing.JPanel();
    JLabel7 = new javax.swing.JLabel();
    JLabel8 = new javax.swing.JLabel();
    JTextFieldTechFormat = new javax.swing.JTextField();
    JPanelRights = new javax.swing.JPanel();
    JPanel5 = new javax.swing.JPanel();
    JPanel6 = new javax.swing.JPanel();
    JLabel1 = new javax.swing.JLabel();
    JLabel2 = new javax.swing.JLabel();
    JLabel3 = new javax.swing.JLabel();
    JLabel4 = new javax.swing.JLabel();
    JScrollPane1 = new javax.swing.JScrollPane();
    JTextArea1 = new javax.swing.JTextArea();
    JLabel5 = new javax.swing.JLabel();
    JSeparator1 = new javax.swing.JSeparator();
    JLabel6 = new javax.swing.JLabel();
    JSeparator2 = new javax.swing.JSeparator();
    JLabel18 = new javax.swing.JLabel();
    JLabel17 = new javax.swing.JLabel();
    JLabel19 = new javax.swing.JLabel();
    JLabel16 = new javax.swing.JLabel();
    JLabel20 = new javax.swing.JLabel();
    JLabel21 = new javax.swing.JLabel();
    JLabel22 = new javax.swing.JLabel();
    JLabel23 = new javax.swing.JLabel();
    JLabel24 = new javax.swing.JLabel();

    setPreferredSize(new java.awt.Dimension(1148, 744));

    javax.swing.GroupLayout JPanelGeneralLayout = new
        javax.swing.GroupLayout(JPanelGeneral);
    JPanelGeneral.setLayout(JPanelGeneralLayout);
    JPanelGeneralLayout.setHorizontalGroup(
        JPanelGeneralLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)

```

```

        .addGap(0, 1143, Short.MAX_VALUE)
111    );
    JPanelGeneralLayout.setVerticalGroup(
113        JPanelGeneralLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 714, Short.MAX_VALUE)
115    );

117    JTabbedPane.addTab("<html><b>1. General</b></html>",
        JPanelGeneral);

119    javax.swing.GroupLayout jPanelLifeCycleLayout = new
        javax.swing.GroupLayout(jPanelLifeCycle);
    JPanelLifeCycle.setLayout(jPanelLifeCycleLayout);
121    JPanelLifeCycleLayout.setHorizontalGroup(
        jPanelLifeCycleLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
123        .addGap(0, 1143, Short.MAX_VALUE)
    );
125    JPanelLifeCycleLayout.setVerticalGroup(
        jPanelLifeCycleLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
127        .addGap(0, 714, Short.MAX_VALUE)
    );

129    JTabbedPane.addTab("<html><b>2. LifeCycle</b></html>
        ", jPanelLifeCycle);

131    JLabel7.setFont(new java.awt.Font("Tahoma", 1, 14));
        // NOI18N
133    JLabel7.setText("1. Format:");

135    JLabel8.setIcon(new javax.swing.ImageIcon(getClass()
        .getResource("/img/help.png"))); // NOI18N
    JLabel8.setToolTipText("<html>MIME type(s) of (all
        the components of) this learning object.<br><br>
        >\nEx.: application/pdf; image/jpeg; text/html<
        br>\nLista de MIME types: http://www.iana.org/
        assignments/media-types<br><br>\nCaso haja mais
        de uma informaÃ§Ão separe os valores com ;\n</
        html>");

137    javax.swing.GroupLayout jPanel3Layout = new javax.
        swing.GroupLayout(jPanel3);
139    jPanel3.setLayout(jPanel3Layout);
    jPanel3Layout.setHorizontalGroup(
141        jPanel3Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
        .addGroup(jPanel3Layout.createSequentialGroup())
143        .addContainerGap()
        .addComponent(JLabel7)

```

```

145         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addComponent(jLabel8)
147        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addComponent(jTextFieldTechFormat, javax.
            swing.GroupLayout.PREFERRED_SIZE, 412,
            javax.swing.GroupLayout.PREFERRED_SIZE)
149        .addContainerGap(618, Short.MAX_VALUE))
    );
151    jPanel3Layout.setVerticalGroup(
        jPanel3Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
153        .addGroup(jPanel3Layout.createSequentialGroup())
        .addContainerGap()
155        .addGroup(jPanel3Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.
                LEADING)
            .addComponent(jTextFieldTechFormat,
                javax.swing.GroupLayout.
                    PREFERRED_SIZE, javax.swing.
                    GroupLayout.DEFAULT_SIZE, javax.
                    swing.GroupLayout.PREFERRED_SIZE)
157            .addGroup(jPanel3Layout.
                createSequentialGroup())
                .addGap(5, 5, 5)
                .addComponent(jLabel7))
            .addComponent(jLabel8))
        .addContainerGap(679, Short.MAX_VALUE))
    );
163
165    jTabbedPane.addTab("<html><b>4. Technical</b></html>
        ", jPanel3);
167
169    javax.swing.GroupLayout jPanelRightsLayout = new
        javax.swing.GroupLayout(jPanelRights);
171    jPanelRights.setLayout(jPanelRightsLayout);
    jPanelRightsLayout.setHorizontalGroup(
173        jPanelRightsLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 1143, Short.MAX_VALUE)
175        );
    jPanelRightsLayout.setVerticalGroup(
        jPanelRightsLayout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 714, Short.MAX_VALUE)
177        );
    jTabbedPane.addTab("<html><i>6. Rights</i></html>",
        jPanelRights);

```



```

179     jLabel1.setFont(new java.awt.Font("Tahoma", 0, 18));
        // NOI18N
jLabel1.setText("Legenda:");
181
jLabel2.setFont(new java.awt.Font("Tahoma", 2, 13));
        // NOI18N
183     jLabel2.setText("— campo opcional");
jLabel3.setFont(new java.awt.Font("Tahoma", 1, 13));
        // NOI18N
185     jLabel3.setText("— campo obrigatorio");
187
jLabel4.setFont(new java.awt.Font("Tahoma", 1, 18));
        // NOI18N
189     jLabel4.setText("Perfil Compativoel Minimo com OBAA
        (PM-OBAA-CORE)");
191
jScrollPane1.setHorizontalScrollBarPolicy(javax.
        swing.ScrollPaneConstants.
        HORIZONTALSCROLLBAR_NEVER);
193
jTextArea1.setEditable(false);
jTextArea1.setColumns(20);
195     jTextArea1.setLineWrap(true);
jTextArea1.setRows(5);
197     jTextArea1.setText("O perfil PM-OBAA-CORE consiste
        no conjunto de metadados minimos considerados
        essenciaes para todo objeto de aprendizagem
        descrito conforme o padrao OBAA. A utilizacao
        desse conjunto visa integracao com os OAs
        legados, com reduzido esforco humano.");
jTextArea1.setWrapStyleWord(true);
199     jScrollPane1.setViewportView(jTextArea1);
201
jLabel5.setFont(new java.awt.Font("Tahoma", 1, 13));
        // NOI18N
jLabel5.setText("Referencia:");
203
jLabel6.setText("Relatorio Tecnico RT-OBAA-01");
205
jLabel18.setFont(new java.awt.Font("Tahoma", 1, 18))
        ; // NOI18N
207     jLabel18.setText("Categorias:");
209
jLabel17.setFont(new java.awt.Font("Tahoma", 1, 14))
        ; // NOI18N
jLabel17.setText("1. General");
211
jLabel19.setFont(new java.awt.Font("Tahoma", 1, 14))
        ; // NOI18N
213     jLabel19.setText("2. LifeCycle");

```

```

215     jLabel16.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
jLabel16.setToolTipText("<html>This category groups
        the general information<br>\nthat describes this
        learning object as a whole.</html>");
217
jLabel20.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
219     jLabel20.setToolTipText("<html>This category
        describes the history and<br>\ncurrent state of
        this learning object and those<br>\nentities
        that have affected this learning object<br>\n
        nduring its evolution.</html>");

221     jLabel21.setFont(new java.awt.Font("Tahoma", 1, 14))
        ; // NOI18N
jLabel21.setText("4. Technical");
223
jLabel22.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
225     jLabel22.setToolTipText("<html>This category
        describes the technical<br>\nrequirements and
        characteristics of this<br>\nlearning object.</
        html>");

227     jLabel23.setFont(new java.awt.Font("Tahoma", 1, 14))
        ; // NOI18N
jLabel23.setText("6. Rights");
229
jLabel24.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
231     jLabel24.setToolTipText("<html>This category
        describes the intellectual<br>\nproperty rights
        and conditions of use for this<br>\nlearning
        object.</html>");

233     javax.swing.GroupLayout jPanel6Layout = new javax.
        swing.GroupLayout(jPanel6);
jPanel6.setLayout(jPanel6Layout);
235     jPanel6Layout.setHorizontalGroup(
        jPanel6Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
237         .addGroup(jPanel6Layout.createSequentialGroup()
            .addGap(10, 10, 10)
            .addGroup(jPanel6Layout.createParallelGroup(
                javax.swing.GroupLayout.Alignment.LEADING)
239                 .addComponent(jLabel1)
                .addComponent(jLabel3)
                .addComponent(jLabel2, javax.swing.
                    GroupLayout.PREFERRED_SIZE, 116,
                    javax.swing.GroupLayout.
241

```

```

                                PREFERRED_SIZE)
243 .addComponent(jLabel4)
    .addComponent(jScrollPane1, javax.swing.
        GroupLayout.PREFERRED_SIZE, 626,
        javax.swing.GroupLayout.
            PREFERRED_SIZE)
245 .addGroup(jPanel6Layout.
    createSequentialGroup())
    .addComponent(jLabel5)
247 .addPreferredGap(javax.swing.
    LayoutStyle.ComponentPlacement.
        RELATED)
    .addComponent(jLabel6))
249 .addComponent(jSeparator1, javax.swing.
    GroupLayout.PREFERRED_SIZE, 379,
    javax.swing.GroupLayout.
        PREFERRED_SIZE)
    .addGroup(jPanel6Layout.
        createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING, false
        )
251 .addGroup(jPanel6Layout.
    createSequentialGroup())
    .addComponent(jLabel17)
253 .addPreferredGap(javax.swing.
    LayoutStyle.
        ComponentPlacement.RELATED,
        javax.swing.GroupLayout.
            DEFAULT_SIZE, Short.
                MAX_VALUE)
    .addComponent(jLabel16)
255 .addGap(182, 182, 182))
    .addComponent(jLabel18)
257 .addComponent(jSeparator2, javax.
    swing.GroupLayout.PREFERRED_SIZE
        , 275, javax.swing.GroupLayout.
            PREFERRED_SIZE))
    .addGroup(jPanel6Layout.
        createSequentialGroup())
259 .addComponent(jLabel19)
    .addPreferredGap(javax.swing.
        LayoutStyle.ComponentPlacement.
            RELATED)
    .addComponent(jLabel20))
261 .addGroup(jPanel6Layout.
    createSequentialGroup())
263 .addComponent(jLabel21)
    .addPreferredGap(javax.swing.
        LayoutStyle.ComponentPlacement.
            RELATED)
265 .addComponent(jLabel22))

```

```

        .addGroup(jPanel6Layout.
            createSequentialGroup())
267         .addComponent(jLabel23)
            .addPreferredGap(javax.swing.
                LayoutStyle.ComponentPlacement.
                    RELATED)
269         .addComponent(jLabel24)))
        .addContainerGap(505, Short.MAX_VALUE))
271 );
jPanel6Layout.setVerticalGroup(
273     jPanel6Layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
        .addGroup(jPanel6Layout.createSequentialGroup())
275         .addGap(27, 27, 27)
            .addComponent(jLabel1)
277         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
            .addComponent(jLabel3)
279         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
            .addComponent(jLabel2)
281         .addGap(72, 72, 72)
            .addComponent(jSeparator1, javax.swing.
                GroupLayout.PREFERRED_SIZE, 10, javax.
                swing.GroupLayout.PREFERRED_SIZE)
283         .addGap(18, 18, 18)
            .addComponent(jLabel4)
285         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
            .addComponent(jScrollPane1, javax.swing.
                GroupLayout.PREFERRED_SIZE, 243, javax.
                swing.GroupLayout.PREFERRED_SIZE)
287         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
            .addGroup(jPanel6Layout.createParallelGroup(
                javax.swing.GroupLayout.Alignment.
                    BASELINE)
289                 .addComponent(jLabel5)
                    .addComponent(jLabel6))
291         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.UNRELATED)
            .addComponent(jSeparator2, javax.swing.
                GroupLayout.PREFERRED_SIZE, 10, javax.
                swing.GroupLayout.PREFERRED_SIZE)
293         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
            .addComponent(jLabel18)
295         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
            .addGroup(jPanel6Layout.createParallelGroup(
                javax.swing.GroupLayout.Alignment.
                    TRAILING)

```

```

297         .addComponent(jLabel17)
        .addComponent(jLabel16))
299     .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.RELATED)
    .addGroup(jPanel6Layout.createParallelGroup(
        javax.swing.GroupLayout.Alignment.
        TRAILING)
301         .addComponent(jLabel19)
        .addComponent(jLabel20))
303     .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.RELATED)
    .addGroup(jPanel6Layout.createParallelGroup(
        javax.swing.GroupLayout.Alignment.
        TRAILING)
305         .addComponent(jLabel21)
        .addComponent(jLabel22))
307     .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.RELATED)
    .addGroup(jPanel6Layout.createParallelGroup(
        javax.swing.GroupLayout.Alignment.
        TRAILING)
309         .addComponent(jLabel23)
        .addComponent(jLabel24))
311     .addContainerGap(76, Short.MAX_VALUE))
    );
313
    javax.swing.GroupLayout jPanel5Layout = new javax.
        swing.GroupLayout(jPanel5);
315    jPanel5.setLayout(jPanel5Layout);
    jPanel5Layout.setHorizontalGroup(
317        jPanel5Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
        .addGap(0, 1143, Short.MAX_VALUE)
319        .addGroup(jPanel5Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel5Layout.
                createSequentialGroup()
321                .addGap(0, 0, Short.MAX_VALUE)
                .addComponent(jPanel6, javax.swing.
                    GroupLayout.PREFERRED_SIZE, javax.
                    swing.GroupLayout.DEFAULT_SIZE,
                    javax.swing.GroupLayout.
                    PREFERRED_SIZE)
323                .addGap(0, 0, Short.MAX_VALUE)))
        );
325    jPanel5Layout.setVerticalGroup(
        jPanel5Layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
327        .addGap(0, 714, Short.MAX_VALUE)
        .addGroup(jPanel5Layout.createParallelGroup(
            javax.swing.GroupLayout.Alignment.LEADING)

```

```

329         .addGroup(jPanel5Layout .
        createSequentialGroup ()
        .addGap(0, 0, Short.MAX.VALUE)
331         .addComponent(jPanel6, javax.swing.
            GroupLayout.PREFERRED.SIZE, javax.
            swing.GroupLayout.DEFAULT.SIZE,
            javax.swing.GroupLayout.
            PREFERRED.SIZE)
        .addGap(0, 0, Short.MAX.VALUE)))
333     );

335     jTabbedPane.addTab("?", jPanel5);

337     javax.swing.GroupLayout layout = new javax.swing.
        GroupLayout(this);
    this.setLayout(layout);
339     layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
341     .addComponent(jTabbedPane, javax.swing.
        GroupLayout.Alignment.TRAILING)
    );
343     layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
345     .addComponent(jTabbedPane, javax.swing.
        GroupLayout.Alignment.TRAILING)
    );
347 }// </editor-fold>//GEN-END: initComponents
// Variables declaration - do not modify//GEN-BEGIN:
    variables
349     private javax.swing.JLabel jLabel1;
351     private javax.swing.JLabel jLabel16;
353     private javax.swing.JLabel jLabel17;
355     private javax.swing.JLabel jLabel18;
357     private javax.swing.JLabel jLabel19;
359     private javax.swing.JLabel jLabel2;
361     private javax.swing.JLabel jLabel20;
363     private javax.swing.JLabel jLabel21;
365     private javax.swing.JLabel jLabel22;
367     private javax.swing.JLabel jLabel23;
369     private javax.swing.JLabel jLabel24;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JLabel jLabel8;
    private javax.swing.JPanel jPanel3;
    private javax.swing.JPanel jPanel5;
    private javax.swing.JPanel jPanel6;
    private javax.swing.JPanel jPanelGeneral;

```

```

371     private javax.swing.JPanel jPanelLifeCycle;
372     private javax.swing.JPanel jPanelRights;
373     private javax.swing.JScrollPane jScrollPane1;
374     private javax.swing.JSeparator jSeparator1;
375     private javax.swing.JSeparator jSeparator2;
376     private javax.swing.JTabbedPane jTabbedPane;
377     private javax.swing.JTextArea jTextArea1;
378     private javax.swing.JTextField jTextFieldTechFormat;
379 // End of variables declaration//GEN-END:variables
}

```

B.4.7 TelaSugestoes.java

```

package user_interface;

2
import util.TipoDeSugestao;

4
public class TelaSugestoes extends javax.swing.JDialog {

6
    public TelaSugestoes(TipoDeSugestao campoName, javax.
        swing.JPanel panel)

8
    {
        initComponents();
10        jLabelCampo.setText(campoName.toString());
11        this.setVisible(true);
12        this.setLocationRelativeTo(panel);
13        // modal?
14    }

16    public javax.swing.JTextArea getTextArea()
    {
17        return jTextArea1;
18    }

20    @SuppressWarnings("unchecked")
21    // <editor-fold defaultstate="collapsed" desc="Generated
    Code">//GEN-BEGIN: initComponents
22    private void initComponents() {

24        jScrollPane2 = new javax.swing.JScrollPane();
25        jTextArea1 = new javax.swing.JTextArea();
26        jButtonAceitar = new javax.swing.JButton();
27        jLabel1 = new javax.swing.JLabel();
28        jLabelCampo = new javax.swing.JLabel();

30        setDefaultCloseOperation(javax.swing.WindowConstants
            .DISPOSE_ON_CLOSE);

```



```

72         .addGroup(layout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel1)
74             .addComponent(jLabelCampo))
        .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
76         .addComponent(jScrollPane2, javax.swing.
            GroupLayout.DEFAULT_SIZE, 322, Short.
            MAX_VALUE)
        .addGap(13, 13, 13)
78         .addComponent(jButtonAceitar)
        .addContainerGap();
80     );

82     pack();
    }// </editor-fold>//GEN-END: initComponents

84     private void jButtonAceitarActionPerformed(java.awt.
        event.ActionEvent evt) {//GEN-FIRST:
            event_jButtonAceitarActionPerformed
86             this.setVisible(false);
        }//GEN-LAST:event_jButtonAceitarActionPerformed

88     // Variables declaration - do not modify//GEN-BEGIN:
        variables
90     private javax.swing.JButton jButtonAceitar;
    private javax.swing.JLabel jLabel1;
92     private javax.swing.JLabel jLabelCampo;
    private javax.swing.JScrollPane jScrollPane2;
94     private javax.swing.JTextArea jTextArea1;
    // End of variables declaration//GEN-END:variables
96 }

```

B.4.8 Categoria.java

```

1 package user_interface.categorias;

3 public interface Categoria
    {
5         public PacoteComCampos
            retornarCamposGeradoresDeSugestoes();
    }

```

B.4.9 JPanelContribute.java

```

package user_interface.categories;

2 public class JPanelContribute extends javax.swing.JPanel{
3
4     public JPanelContribute() {
5         initComponents();
6     }
7
8     @SuppressWarnings("unchecked")
9     // <editor-fold defaultstate="collapsed" desc="Generated
10        Code"> //GEN-BEGIN: initComponents
11     private void initComponents() {
12
13         jLabel11 = new javax.swing.JLabel();
14         jLabel13 = new javax.swing.JLabel();
15         jLabel12 = new javax.swing.JLabel();
16         jLabel14 = new javax.swing.JLabel();
17         jLabel15 = new javax.swing.JLabel();
18         jLabel16 = new javax.swing.JLabel();
19         jComboBox1 = new javax.swing.JComboBox();
20         jComboBox2 = new javax.swing.JComboBox();
21         jComboBox3 = new javax.swing.JComboBox();
22         jComboBox4 = new javax.swing.JComboBox();
23         jLabel1 = new javax.swing.JLabel();
24         jLabel2 = new javax.swing.JLabel();
25         jLabel3 = new javax.swing.JLabel();
26         jLabel4 = new javax.swing.JLabel();
27         jLabel5 = new javax.swing.JLabel();
28         jTextField1 = new javax.swing.JTextField();
29         jTextField2 = new javax.swing.JTextField();
30         jTextField3 = new javax.swing.JTextField();
31         jTextField4 = new javax.swing.JTextField();
32         jTextField5 = new javax.swing.JTextField();
33         jSeparator1 = new javax.swing.JSeparator();
34         jSeparator2 = new javax.swing.JSeparator();
35
36         jLabel11.setFont(new java.awt.Font("Tahoma", 1, 14))
37             ; // NOI18N
38         jLabel11.setText("3.1. Role");
39
40         jLabel13.setIcon(new javax.swing.ImageIcon(getClass
41             ().getResource("/img/help.png"))); // NOI18N
42         jLabel13.setToolTipText("<html>Kind of contribution
43             .</html>");
44
45         jLabel12.setFont(new java.awt.Font("Tahoma", 1, 14))
46             ; // NOI18N
47         jLabel12.setText("3.2. Entity");

```

```

44      jLabel14.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
46      jLabel14.setToolTipText("<html>\nThe identification
        of and information about<br>\nentities (i.e.,
        people, organizations)<br>\ncontributing to this
        learning object. The<br>\nentities shall be
        ordered as most relevant first.<br>\n</html>");

48      jLabel15.setFont(new java.awt.Font("Tahoma", 1, 14))
        ; // NOI18N
      jLabel15.setText("3.3. Date");

50      jLabel16.setIcon(new javax.swing.ImageIcon(getClass
        ().getResource("/img/help.png"))); // NOI18N
52      jLabel16.setToolTipText("<html>The date of the
        contribution.\n</html>");

54      jComboBox1.setModel(new javax.swing.
        DefaultComboBoxModel(new String[] { "author", "
        publisher", "unknown", "initiator", "terminator"
        , "validator", "editor", "graphical designer", "
        technical implementer", "content provider", "
        technical validator", "educational validator", "
        script writer", "instructional designer", "
        subject matter expert" }));

56      jComboBox2.setModel(new javax.swing.
        DefaultComboBoxModel(new String[] { "1", "2", "3
        ", "4", "5", "6", "7", "8", "9", "10", "11", "12
        ", "13", "14", "15", "16", "17", "18", "19", "20
        ", "21", "22", "23", "24", "25", "26", "27", "28
        ", "29", "30", "31" }));

58      jComboBox3.setModel(new javax.swing.
        DefaultComboBoxModel(new String[] { "jan", "fev"
        , "mar", "abr", "maio", "jun", "jul", "ago", "
        set", "out", "nov", "dez" }));

60      jComboBox4.setModel(new javax.swing.
        DefaultComboBoxModel(new String[] { "2020", "
        2019", "2018", "2017", "2016", "2015", "2014", "
        2013", "2012", "2011", "2010", "2009", "2008", "
        2007", "2006", "2005", "2004", "2003", "2002", "
        2001", "2000", "1999", "1998", "1997", "1996", "
        1995", "1994", "1993", "1992", "1991", "1990", "
        1989", "1988", "1987", "1986", "1985", "1984", "
        1983", "1982", "1981", "1980" }));
      jComboBox4.setSelectedIndex(7);

62      jLabel1.setText("First Name:");

64

```

```

jLabel2.setText("Middle Name:");
66
jLabel3.setText("Last Name:");
68
jLabel4.setText("E-mail:");
70
jLabel5.setText("Organization:");
72

jSeparator2.setOrientation(javax.swing.
    SwingConstants.VERTICAL);
74

javax.swing.GroupLayout layout = new javax.swing.
    GroupLayout(this);
76
this.setLayout(layout);
layout.setHorizontalGroup(
78
    layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
        .addGroup(layout.createParallelGroup(javax.
80
            .addComponent(jSeparator1)
            .addGroup(layout.createParallelGroup(javax.
                .addComponent(jLabel15)
                .addPreferredGap(javax.swing
                    .LayoutStyle.
                        ComponentPlacement.
                            RELATED)
82
                .addComponent(jLabel16)
                .addPreferredGap(javax.swing
                    .LayoutStyle.
                        ComponentPlacement.
                            RELATED)
84
                .addComponent(jComboBox2,
                    javax.swing.GroupLayout.
                        PREFERRED_SIZE, javax.
                        swing.GroupLayout.
                            DEFAULT_SIZE, javax.
                                swing.GroupLayout.
                                    PREFERRED_SIZE)
                .addPreferredGap(javax.swing
                    .LayoutStyle.
                        ComponentPlacement.
                            RELATED)
86
                .addComponent(jComboBox3,
                    javax.swing.GroupLayout.
90
                    .addPreferredGap(javax.swing
                        .LayoutStyle.
                            ComponentPlacement.
                                RELATED)
92
                    .addComponent(jComboBox3,
                        javax.swing.GroupLayout.

```

```

        PREFERRED_SIZE, javax .
        swing . GroupLayout .
        DEFAULT_SIZE, javax .
        swing . GroupLayout .
        PREFERRED_SIZE)
    .addPreferredGap ( javax . swing
        . LayoutStyle .
        ComponentPlacement .
        RELATED)
94    .addComponent ( jComboBox4 ,
        javax . swing . GroupLayout .
        PREFERRED_SIZE, javax .
        swing . GroupLayout .
        DEFAULT_SIZE, javax .
        swing . GroupLayout .
        PREFERRED_SIZE) )
    .addGroup ( layout .
        createSequentialGroup ( )
96    .addComponent ( jLabel12 )
        .addPreferredGap ( javax . swing
            . LayoutStyle .
            ComponentPlacement .
            RELATED)
98    .addComponent ( jLabel14 ) )
    .addGroup ( layout .
        createParallelGroup ( javax .
        swing . GroupLayout . Alignment .
        TRAILING, false )
100    .addGroup ( javax . swing .
        GroupLayout . Alignment .
        LEADING, layout .
        createSequentialGroup ( )
            .addComponent ( jLabel5 )
102    .addPreferredGap ( javax .
        swing . LayoutStyle .
        ComponentPlacement .
        RELATED)
        .addComponent (
            jTextField5 ) ) )
104    .addGroup ( javax . swing .
        GroupLayout . Alignment .
        LEADING, layout .
        createSequentialGroup ( )
            .addComponent ( jLabel4 )
106    .addGap ( 41 , 41 , 41 )
            .addComponent (
                jTextField3 , javax .
                swing . GroupLayout .
                PREFERRED_SIZE, 194 ,
                javax . swing .
                GroupLayout .
                PREFERRED_SIZE) ) )

```

```

108 .addGroup(layout .
    createSequentialGroup ()
    .addGroup(layout .
        createParallelGroup (
            javax.swing.GroupLayout .
            Alignment.LEADING)
110 .addComponent (
            jTextField2 , javax .
            swing.GroupLayout .
            PREFERRED_SIZE, 150 ,
            javax.swing .
            GroupLayout .
            PREFERRED_SIZE)
        .addComponent (jLabel1))
112 .addPreferredGap (javax.swing
    .LayoutStyle .
    ComponentPlacement .
    RELATED)
    .addGroup(layout .
        createParallelGroup (
            javax.swing.GroupLayout .
            Alignment.LEADING)
114 .addComponent (
            jTextField1 , javax .
            swing.GroupLayout .
            PREFERRED_SIZE, 148 ,
            javax.swing .
            GroupLayout .
            PREFERRED_SIZE)
        .addComponent (jLabel2))
116 .addPreferredGap (javax.swing
    .LayoutStyle .
    ComponentPlacement .
    RELATED)
    .addGroup(layout .
        createParallelGroup (
            javax.swing.GroupLayout .
            Alignment.LEADING)
118 .addComponent (jLabel3)
        .addComponent (
            jTextField4 , javax .
            swing.GroupLayout .
            PREFERRED_SIZE, 156 ,
            javax.swing .
            GroupLayout .
            PREFERRED_SIZE)))
120 .addGroup(layout .
    createSequentialGroup ()
    .addComponent (jLabel11)
122 .addPreferredGap (javax.swing
    .LayoutStyle .
    ComponentPlacement .

```

```

RELATED)
124 .addComponent(jLabel13)
    .addPreferredGap(javax.swing
        .LayoutStyle.
        ComponentPlacement.
        UNRELATED)
    .addComponent(jComboBox1,
        javax.swing.GroupLayout.
        PREFERRED_SIZE, javax.
        swing.GroupLayout.
        DEFAULT_SIZE, javax.
        swing.GroupLayout.
        PREFERRED_SIZE)))
126 .addGap(18, 18, 18)
    .addComponent(jSeparator2, javax.
        swing.GroupLayout.PREFERRED_SIZE
        , javax.swing.GroupLayout.
        DEFAULT_SIZE, javax.swing.
        GroupLayout.PREFERRED_SIZE)))
128 .addContainerGap(javax.swing.GroupLayout.
        DEFAULT_SIZE, Short.MAX_VALUE))
);
130 layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
132 .addGroup(layout.createSequentialGroup())
    .addContainerGap()
134 .addGroup(layout.createParallelGroup(javax.
        swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup())
136 .addGap(0, 11, Short.MAX_VALUE)
    .addGroup(layout.createParallelGroup
        (javax.swing.GroupLayout.
        Alignment.LEADING)
138 .addComponent(jLabel11)
    .addComponent(jLabel13)
140 .addComponent(jComboBox1, javax.
        swing.GroupLayout.
        PREFERRED_SIZE, javax.swing.
        GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.
        PREFERRED_SIZE)))
    .addGap(23, 23, 23)
142 .addGroup(layout.createParallelGroup
        (javax.swing.GroupLayout.
        Alignment.LEADING)
    .addComponent(jLabel12)
    .addComponent(jLabel14))
144 .addPreferredGap(javax.swing.
        LayoutStyle.ComponentPlacement.
        RELATED)

```

```

146 .addGroup(layout.createParallelGroup
    (javax.swing.GroupLayout.
      Alignment.LEADING)
    .addGroup(layout.
      createParallelGroup(javax.
        swing.GroupLayout.Alignment.
          BASELINE)
148      .addComponent(jLabel2)
      .addComponent(jLabel3))
150      .addComponent(jLabel1))
    .addPreferredGap(javax.swing.
      LayoutStyle.ComponentPlacement.
        RELATED)
152 .addGroup(layout.createParallelGroup
    (javax.swing.GroupLayout.
      Alignment.BASELINE)
    .addComponent(jTextField2, javax
      .swing.GroupLayout.
        PREFERRED.SIZE, javax.swing.
        GroupLayout.DEFAULT.SIZE,
        javax.swing.GroupLayout.
          PREFERRED.SIZE)
154      .addComponent(jTextField1, javax
        .swing.GroupLayout.
          PREFERRED.SIZE, javax.swing.
          GroupLayout.DEFAULT.SIZE,
          javax.swing.GroupLayout.
            PREFERRED.SIZE)
        .addComponent(jTextField4, javax
          .swing.GroupLayout.
            PREFERRED.SIZE, javax.swing.
            GroupLayout.DEFAULT.SIZE,
            javax.swing.GroupLayout.
              PREFERRED.SIZE))
156      .addPreferredGap(javax.swing.
        LayoutStyle.ComponentPlacement.
          RELATED)
    .addGroup(layout.createParallelGroup
      (javax.swing.GroupLayout.
        Alignment.BASELINE)
158      .addComponent(jLabel4)
      .addComponent(jTextField3, javax
        .swing.GroupLayout.
          PREFERRED.SIZE, javax.swing.
          GroupLayout.DEFAULT.SIZE,
          javax.swing.GroupLayout.
            PREFERRED.SIZE))
160      .addPreferredGap(javax.swing.
        LayoutStyle.ComponentPlacement.
          RELATED)
    .addGroup(layout.createParallelGroup
      (javax.swing.GroupLayout.

```



```

162         Alignment.BASELINE)
        .addComponent(jLabel5)
        .addComponent(jTextField5, javax
            .swing.GroupLayout.
                PREFERRED_SIZE, javax.swing.
                GroupLayout.DEFAULT_SIZE,
                javax.swing.GroupLayout.
                PREFERRED_SIZE))
164    .addPreferredGap(javax.swing.
        LayoutStyle.ComponentPlacement.
        UNRELATED)
    .addGroup(layout.createParallelGroup
        (javax.swing.GroupLayout.
            Alignment.LEADING)
166        .addComponent(jLabel15)
        .addComponent(jLabel16)
168        .addGroup(layout.
            createParallelGroup(javax.
                swing.GroupLayout.Alignment.
                BASELINE)
            .addComponent(jComboBox2,
                javax.swing.GroupLayout.
                PREFERRED_SIZE, javax.
                swing.GroupLayout.
                DEFAULT_SIZE, javax.
                swing.GroupLayout.
                PREFERRED_SIZE)
170            .addComponent(jComboBox3,
                javax.swing.GroupLayout.
                PREFERRED_SIZE, javax.
                swing.GroupLayout.
                DEFAULT_SIZE, javax.
                swing.GroupLayout.
                PREFERRED_SIZE)
            .addComponent(jComboBox4,
                javax.swing.GroupLayout.
                PREFERRED_SIZE, javax.
                swing.GroupLayout.
                DEFAULT_SIZE, javax.
                swing.GroupLayout.
                PREFERRED_SIZE)))
172    .addComponent(jSeparator2, javax.swing.
        GroupLayout.Alignment.TRAILING))
    .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.RELATED)
174    .addComponent(jSeparator1, javax.swing.
        GroupLayout.PREFERRED_SIZE, 10, javax.
        swing.GroupLayout.PREFERRED_SIZE))

    );
176 }// </editor-fold>//GEN-END:initComponents
// Variables declaration - do not modify//GEN-BEGIN:
    variables

```

```

178     private javax.swing.JComboBox jComboBox1;
179     private javax.swing.JComboBox jComboBox2;
180     private javax.swing.JComboBox jComboBox3;
181     private javax.swing.JComboBox jComboBox4;
182     private javax.swing.JLabel jLabel1;
183     private javax.swing.JLabel jLabel11;
184     private javax.swing.JLabel jLabel12;
185     private javax.swing.JLabel jLabel13;
186     private javax.swing.JLabel jLabel14;
187     private javax.swing.JLabel jLabel15;
188     private javax.swing.JLabel jLabel16;
189     private javax.swing.JLabel jLabel2;
190     private javax.swing.JLabel jLabel3;
191     private javax.swing.JLabel jLabel4;
192     private javax.swing.JLabel jLabel5;
193     private javax.swing.JSeparator jSeparator1;
194     private javax.swing.JSeparator jSeparator2;
195     private javax.swing.JTextField jTextField1;
196     private javax.swing.JTextField jTextField2;
197     private javax.swing.JTextField jTextField3;
198     private javax.swing.JTextField jTextField4;
199     private javax.swing.JTextField jTextField5;
200     // End of variables declaration//GEN-END:variables
}

```

B.4.10 JPanelGeneralParaOBAA.java

```

package user_interface.categorias;

2
import java.awt.Color;
4 import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
6 import java.util.ArrayList;
import util.TipoDeSugestao;
8 import user_interface.Main;
import user_interface.Profile;
10
public class JPanelGeneralParaOBAA extends javax.swing.
    JPanel implements Categoria{
12
    Profile profile;
14
    public JPanelGeneralParaOBAA(Profile profile) {
16         initComponents();
        this.profile = profile;
18         jTableIdentifier.getColumnModel().getColumn(0).
            setMinWidth(270);

```

```

        jTableIdentifier.getColumnModel().getColumn(0).
            setMaxWidth(270);
20     jTableIdentifier.setShowGrid(true);
        jTableIdentifier.setGridColor(new Color(0,0,153));
22     jTableIdentifier.setTableHeader(null);
    }

24
    @SuppressWarnings("unchecked")
26     // <editor-fold defaultstate="collapsed" desc="Generated
        Code">//GEN-BEGIN: initComponents
    private void initComponents() {

28
        jLabel7 = new javax.swing.JLabel();
30     jLabel8 = new javax.swing.JLabel();
        jLabel11 = new javax.swing.JLabel();
32     jLabel13 = new javax.swing.JLabel();
        jScrollPane2 = new javax.swing.JScrollPane();
34     jTableIdentifier = new javax.swing.JTable();
        jLabel12 = new javax.swing.JLabel();
36     jLabel14 = new javax.swing.JLabel();
        jButtonTableIdentifierAdd = new javax.swing.JButton
            ();
38     jButtonTableIdentifierDel = new javax.swing.JButton
            ();
        jLabel9 = new javax.swing.JLabel();
40     jSeparator1 = new javax.swing.JSeparator();
        jTextFieldLanguage = new javax.swing.JTextField();
42     jComboBoxTitleLang = new javax.swing.JComboBox();
        jLabel10 = new javax.swing.JLabel();
44     jSeparator2 = new javax.swing.JSeparator();
        jLabel15 = new javax.swing.JLabel();
46     jLabel16 = new javax.swing.JLabel();
        jTextFieldTitle = new javax.swing.JTextField();
48     jSeparator3 = new javax.swing.JSeparator();
        jLabel1 = new javax.swing.JLabel();
50     jScrollPane1 = new javax.swing.JScrollPane();
        jTextAreaDescription = new javax.swing.JTextArea();
52     jComboBoxDescriptionLang = new javax.swing.JComboBox
            ();
        jLabel17 = new javax.swing.JLabel();
54     jSeparator4 = new javax.swing.JSeparator();
        jLabel2 = new javax.swing.JLabel();
56     jLabel18 = new javax.swing.JLabel();
        jTextFieldKeywords = new javax.swing.JTextField();
58     jComboBoxKeywordsLang = new javax.swing.JComboBox();
        jButtonSugestoesParaTitle = new javax.swing.JButton
            ();
60     jButtonSugestoesParaDescription = new javax.swing.
            JButton();
        jButtonSugestoesParaKeywords = new javax.swing.
            JButton();

```

62

```

jLabel7.setFont(new java.awt.Font("Tahoma", 1, 14));
// NOI18N
64  jLabel7.setText("1. Identifier:");

66  jLabel8.setIcon(new javax.swing.ImageIcon(getClass()
    .getResource("/img/help.png"))); // NOI18N
jLabel8.setToolTipText("<html>A globally unique
    label that identifies this learning object.</
    html>");

68

jLabel11.setFont(new java.awt.Font("Tahoma", 1, 14))
    ; // NOI18N
70  jLabel11.setText("1.1. Catalog");

72  jLabel13.setIcon(new javax.swing.ImageIcon(getClass()
    .getResource("/img/help.png"))); // NOI18N
jLabel13.setToolTipText("<html>The name or
    designator of the identification or cataloging\
    nscheme for this entry. A namespace scheme.<br>
    ne.g: \"ISBN\", \"ARIADNE\", \"URI\"</html>
    ");

74

jTableIdentifier.setBorder(javax.swing.BorderFactory
    .createLineBorder(new java.awt.Color(0, 0, 153))
    );
76  jTableIdentifier.setFont(new java.awt.Font("Tahoma",
    0, 14)); // NOI18N
jTableIdentifier.setModel(new javax.swing.table.
    DefaultTableModel(1,2));
78  jTableIdentifier.setAutoResizeMode(javax.swing.
    JTable.AUTO_RESIZE_LAST_COLUMN);
jTableIdentifier.setRowHeight(24);
80  jTableIdentifier.setSelectionBackground(new java.awt
    .Color(199, 230, 251));
jTableIdentifier.setSelectionForeground(new java.awt
    .Color(0, 0, 0));
82  jTableIdentifier.getTableHeader().
    setReorderingAllowed(false);
jScrollPane2.setViewportViewView(jTableIdentifier);

84

jLabel12.setFont(new java.awt.Font("Tahoma", 1, 14))
    ; // NOI18N
86  jLabel12.setText("1.2. Entry");

88  jLabel14.setIcon(new javax.swing.ImageIcon(getClass()
    .getResource("/img/help.png"))); // NOI18N
jLabel14.setToolTipText("<html>\nThe value of the
    identifier within the identification<br>\nno
    cataloging scheme that designates or identifies<
    br>\nthis learning object. A namespace specific
    string.<br>\nne.g: \"2-7342-0318\", \"
    LEAO875\",<br>\n\" http://www.ieee.org/documents

```

```

    /1234\"\\n</html>");
90
jButtonTableIdentifierAdd.setText("+");
92
jButtonTableIdentifierAdd.addActionListener(new java
    .awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.
94            ActionEvent evt) {
                jButtonTableIdentifierAddActionPerformed(evt
                    );
            }
96    });

jButtonTableIdentifierDel.setText("-");
98
jButtonTableIdentifierDel.addActionListener(new java
    .awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.
100            ActionEvent evt) {
                jButtonTableIdentifierDelActionPerformed(evt
                    );
            }
102    });

jLabel9.setFont(new java.awt.Font("Tahoma", 1, 14));
104
// NOI18N
jLabel9.setText("2. Title:");
106

jComboBoxTitleLang.setEditable(true);
jComboBoxTitleLang.setModel(new javax.swing.
    DefaultComboBoxModel(new String[] { "en", "pt"
108    }));

jLabel10.setIcon(new javax.swing.ImageIcon(getClass
    ().getResource("/img/help.png"))); // NOI18N
110
jLabel10.setToolTipText("<html>Name given to this
    learning object and the language that the title
112    is writen.</html>");

jLabel15.setIcon(new javax.swing.ImageIcon(getClass
    ().getResource("/img/help.png"))); // NOI18N
jLabel15.setToolTipText("<html>The primary human
    language or languages used within this learning<
114    br>\nobject to communicate to the intended user
    .<br>\nEx.: en; en-GB; fr-CA; pt-BR; pt-PT
    <br>\nCaso haja mais de uma informaÃ§Ã£o
    separe os valores com ;</html>");
116

jLabel16.setFont(new java.awt.Font("Tahoma", 1, 14))
    ; // NOI18N
jLabel16.setText("3. Language:");
118

jTextFieldTitle.setBackground(new java.awt.Color
    (204, 255, 204));
120

```

```

122    jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
        // NOI18N
jLabel1.setText("4. Description:");
124
jTextAreaDescription.setBackground(new java.awt.
    Color(204, 255, 204));
126    jTextAreaDescription.setColumns(20);
jTextAreaDescription.setLineWrap(true);
128    jTextAreaDescription.setRows(5);
jTextAreaDescription.setWrapStyleWord(true);
130    jScrollPane1.setViewportView(jTextAreaDescription);

132    jComboBoxDescriptionLang.setEditable(true);
jComboBoxDescriptionLang.setModel(new javax.swing.
    DefaultComboBoxModel(new String[] { "en", "pt"
        }));
134

jLabel17.setIcon(new javax.swing.ImageIcon(getClass
    ().getResource("/img/help.png"))); // NOI18N
136    jLabel17.setToolTipText("<html>A textual description
        of the content of this learning object.<br><br>
        >\n\nEx.: \"In this video clip, the life and
        works of Leonardo da Vinci are briefly presented
        . (...)</html>");

138    jLabel2.setFont(new java.awt.Font("Tahoma", 1, 14));
        // NOI18N
jLabel2.setText("5. Keywords:");
140

jLabel18.setIcon(new javax.swing.ImageIcon(getClass
    ().getResource("/img/help.png"))); // NOI18N
142    jLabel18.setToolTipText("<html>A keyword or phrase
        describing the topic of this learning object.<br>
        ><br>\n\nEx.: \"Mona Lisa\"<br><br>\n\nCaso haja
        mais de uma informaÃ§Ã£o, separe os valores com
        ;</html>");

144    jTextFieldKeywords.setBackground(new java.awt.Color
        (204, 255, 204));

146    jComboBoxKeywordsLang.setEditable(true);
jComboBoxKeywordsLang.setModel(new javax.swing.
    DefaultComboBoxModel(new String[] { "en", "pt"
        }));
148

jButtonSugestoesParaTitle.setBackground(new java.awt
    .Color(204, 255, 204));
150    jButtonSugestoesParaTitle.setFont(new java.awt.Font(
        "Tahoma", 1, 13)); // NOI18N
jButtonSugestoesParaTitle.setForeground(new java.awt
    .Color(0, 102, 51));

```

```

152 jButtonSugestoesParaTitle.setText("Help me!");
jButtonSugestoesParaTitle.addActionListener(new java
    .awt.event.ActionListener() {
154     public void actionPerformed(java.awt.event.
        ActionEvent evt) {
        jButtonSugestoesParaTitleActionPerformed(evt
            );
156     }
    });
158
jButtonSugestoesParaDescription.setBackground(new
    java.awt.Color(204, 255, 204));
160 jButtonSugestoesParaDescription.setFont(new java.awt
    .Font("Tahoma", 1, 13)); // NOI18N
jButtonSugestoesParaDescription.setForeground(new
    java.awt.Color(0, 102, 51));
162 jButtonSugestoesParaDescription.setText("Help me!");
jButtonSugestoesParaDescription.addActionListener(
    new java.awt.event.ActionListener() {
164     public void actionPerformed(java.awt.event.
        ActionEvent evt) {
        jButtonSugestoesParaDescriptionActionPerformed
            (evt);
166     }
    });
168
jButtonSugestoesParaKeywords.setBackground(new java.
    awt.Color(204, 255, 204));
170 jButtonSugestoesParaKeywords.setFont(new java.awt.
    Font("Tahoma", 1, 13)); // NOI18N
jButtonSugestoesParaKeywords.setForeground(new java.
    awt.Color(0, 102, 51));
172 jButtonSugestoesParaKeywords.setText("Help me!");
jButtonSugestoesParaKeywords.addActionListener(new
    java.awt.event.ActionListener() {
174     public void actionPerformed(java.awt.event.
        ActionEvent evt) {
        jButtonSugestoesParaKeywordsActionPerformed(
            evt);
176     }
    });
178
javax.swing.GroupLayout layout = new javax.swing.
    GroupLayout(this);
180 this.setLayout(layout);
layout.setHorizontalGroup(
182     layout.createParallelGroup(javax.swing.
        GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
184            .addGroup(layout.createParallelGroup(javax.
                swing.GroupLayout.Alignment.LEADING)

```

```

186         .addGroup(layout.createSequentialGroup())
        .addGroup(layout.createParallelGroup
            (javax.swing.GroupLayout.
              Alignment.LEADING)
188         .addComponent(jSeparator4)
        .addComponent(jSeparator3)
190         .addComponent(jSeparator2)
        .addComponent(jSeparator1)
192         .addGroup(layout.
            createSequentialGroup())
        .addGroup(layout.
            createParallelGroup(
                javax.swing.GroupLayout.
                  Alignment.LEADING)
194         .addGroup(layout.
            createSequentialGroup
            ())
        .addComponent(
            jLabel7)
196         .addPreferredGap(
            javax.swing.
              LayoutStyle.
                ComponentPlacement.
                  RELATED)
        .addComponent(
            jLabel8))
198     .addGroup(layout.
        createSequentialGroup
        ())
        .addGap(10, 10, 10)
200         .addGroup(layout.
            createParallelGroup
            (javax.swing.
              GroupLayout.
                Alignment.
                  LEADING)
        .addComponent(
            jScrollPane2
            )
202         .addGroup(layout.
            .
            createSequentialGroup
            ())
        .
        .addComponent
        (
            jLabel11
        )
204         .
        .addPreferredGap
        (javax.
          swing.

```



```

LayoutStyle
.
ComponentPlacement
.RELATED
)
.
    addComponent
    (
        jLabel13
    )
206 .addGap(168,
        168,
        168)
.
    addComponent
    (
        jLabel12
    )
208 .
    addPreferredGap
    (javax.
    swing.
    LayoutStyle
    .
    ComponentPlacement
    .RELATED
    )
.
    addComponent
    (
        jLabel14
    )
210 .addGap(310,
        492,
        Short.
        MAX.VALUE
        )))))
.addPreferredGap(javax.swing
    .LayoutStyle.
    ComponentPlacement.
    RELATED)
212 .addGroup(layout.
    createParallelGroup(
    javax.swing.GroupLayout.
    Alignment.LEADING, false
    )
    .addComponent(
        jButtonTableIdentifierAdd
        , javax.swing.
        GroupLayout.
        DEFAULT_SIZE, javax.
        swing.GroupLayout.

```

```

                DEFAULT_SIZE, Short.
                MAX_VALUE)
214         .addComponent(
                JButtonTableIdentifierDel
                , javax.swing.
                GroupLayout.
                PREFERRED_SIZE, 41,
                javax.swing.
                GroupLayout.
                PREFERRED_SIZE))))
        .addGap(214, 214, 214))
216     .addGroup(layout.createSequentialGroup())
        .addGroup(layout.createParallelGroup
        (javax.swing.GroupLayout.
        Alignment.LEADING)
218         .addGroup(layout.
        createSequentialGroup())
            .addGap(10, 10, 10)
220             .addComponent(
                JTextFieldKeywords ,
                javax.swing.GroupLayout.
                PREFERRED_SIZE, 674,
                javax.swing.GroupLayout.
                PREFERRED_SIZE)
            .addPreferredGap(javax.swing
            .LayoutStyle.
            ComponentPlacement.
            UNRELATED)
222         .addComponent(
            JComboBoxKeywordsLang,
            javax.swing.GroupLayout.
            PREFERRED_SIZE, 54,
            javax.swing.GroupLayout.
            PREFERRED_SIZE)
            .addGap(18, 18, 18)
224         .addComponent(
            JButtonSugestoesParaKeywords
            ))
        .addGroup(layout.
        createSequentialGroup())
226         .addComponent(jLabel2)
            .addPreferredGap(javax.swing
            .LayoutStyle.
            ComponentPlacement.
            RELATED)
            .addComponent(jLabel18)))
        .addGap(0, 0, Short.MAX_VALUE))
230     .addGroup(layout.createSequentialGroup())
        .addGroup(layout.createParallelGroup
        (javax.swing.GroupLayout.
        Alignment.LEADING)

```

```

232     .addGroup(layout .
        createSequentialGroup()
        .addComponent(jLabel9)
234     .addPreferredGap(javax.swing
        .LayoutStyle .
        ComponentPlacement .
        RELATED)
        .addComponent(jLabel10))
236 .addGroup(layout .
        createSequentialGroup()
        .addComponent(jLabel16)
238     .addPreferredGap(javax.swing
        .LayoutStyle .
        ComponentPlacement .
        RELATED)
        .addComponent(jLabel15))
240 .addGroup(layout .
        createSequentialGroup()
        .addComponent(jLabel1)
242     .addPreferredGap(javax.swing
        .LayoutStyle .
        ComponentPlacement .
        RELATED)
        .addComponent(jLabel17))
244 .addGroup(layout .
        createSequentialGroup()
        .addGap(10, 10, 10)
246     .addGroup(layout .
        createParallelGroup(
        javax.swing.GroupLayout .
        Alignment.LEADING)
        .addGroup(layout .
            createSequentialGroup
            ()
248         .addComponent(
            jTextFieldTitle ,
            javax.swing .
            GroupLayout .
            PREFERRED_SIZE,
            523, javax.swing
            .GroupLayout .
            PREFERRED_SIZE)
        .addPreferredGap(
            javax.swing .
            LayoutStyle .
            ComponentPlacement
            .RELATED)
250     .addComponent(
        jComboBoxTitleLang
        , javax.swing .
        GroupLayout .
        PREFERRED_SIZE,

```

```

54, javax.swing.
    GroupLayout.
    PREFERRED_SIZE)
252 .addPreferredGap(
    javax.swing.
    LayoutStyle.
    ComponentPlacement
    .UNRELATED)
    .addComponent(
        jButtonSugestoesParaTitle
        ))
    .addComponent(
        jTextFieldLanguage,
        javax.swing.
        GroupLayout.
        PREFERRED_SIZE, 523,
        javax.swing.
        GroupLayout.
        PREFERRED_SIZE)
254 .addGroup(layout.
    createSequentialGroup
    ())
    .addComponent(
        jScrollPane1,
        javax.swing.
        GroupLayout.
        PREFERRED_SIZE,
        678, javax.swing.
        .GroupLayout.
        PREFERRED_SIZE)
256 .addPreferredGap(
    javax.swing.
    LayoutStyle.
    ComponentPlacement
    .UNRELATED)
    .addComponent(
        jComboBoxDescriptionLang
        , javax.swing.
        GroupLayout.
        PREFERRED_SIZE,
        54, javax.swing.
        GroupLayout.
        PREFERRED_SIZE)
258 .addPreferredGap(
    javax.swing.
    LayoutStyle.
    ComponentPlacement
    .UNRELATED)
    .addComponent(
        jButtonSugestoesParaDescription
        )))))

```

```

260         .addContainerGap(javax.swing.
           GroupLayout.DEFAULT_SIZE, Short.
           MAX_VALUE)))
    );
262    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.
           GroupLayout.Alignment.LEADING)
264        .addGroup(layout.createSequentialGroup())
           .addContainerGap()
266           .addGroup(layout.createParallelGroup(javax.
           swing.GroupLayout.Alignment.LEADING)
           .addGroup(layout.createSequentialGroup())
268               .addGap(5, 5, 5)
               .addComponent(jLabel7))
270               .addComponent(jLabel8))
           .addPreferredGap(javax.swing.LayoutStyle.
           ComponentPlacement.RELATED)
272           .addGroup(layout.createParallelGroup(javax.
           swing.GroupLayout.Alignment.TRAILING)
           .addGroup(layout.createParallelGroup(
           javax.swing.GroupLayout.Alignment.
           LEADING)
274               .addComponent(jLabel11)
               .addComponent(jLabel13))
           .addGroup(layout.createParallelGroup(
           javax.swing.GroupLayout.Alignment.
           LEADING)
276               .addComponent(jLabel12)
               .addComponent(jLabel14)))
           .addPreferredGap(javax.swing.LayoutStyle.
           ComponentPlacement.RELATED)
280           .addGroup(layout.createParallelGroup(javax.
           swing.GroupLayout.Alignment.LEADING)
           .addComponent(jScrollPane2, javax.swing.
           GroupLayout.PREFERRED_SIZE, 75,
           javax.swing.GroupLayout.
           PREFERRED_SIZE)
282           .addGroup(layout.createSequentialGroup())
               .addComponent(
           JButtonTableIdentifierAdd)
284           .addPreferredGap(javax.swing.
           LayoutStyle.ComponentPlacement.
           RELATED)
           .addComponent(
           JButtonTableIdentifierDel)))
286        .addGap(18, 18, 18)
        .addComponent(jSeparator1, javax.swing.
           GroupLayout.PREFERRED_SIZE, 10, javax.
           swing.GroupLayout.PREFERRED_SIZE)
288        .addPreferredGap(javax.swing.LayoutStyle.
           ComponentPlacement.RELATED)

```

```

290 .addGroup(layout.createParallelGroup(javax.
    swing.GroupLayout.Alignment.LEADING)
    .addComponent(jLabel9)
    .addComponent(jLabel10))
292 .addPreferredGap(javax.swing.LayoutStyle.
    ComponentPlacement.RELATED)
    .addGroup(layout.createParallelGroup(javax.
    swing.GroupLayout.Alignment.BASELINE)
294 .addComponent(jComboBoxTitleLang, javax.
    swing.GroupLayout.PREFERRED_SIZE,
    javax.swing.GroupLayout.DEFAULT_SIZE
    , javax.swing.GroupLayout.
    PREFERRED_SIZE)
    .addComponent(jTextFieldTitle, javax.
    swing.GroupLayout.PREFERRED_SIZE,
    javax.swing.GroupLayout.DEFAULT_SIZE
    , javax.swing.GroupLayout.
    PREFERRED_SIZE)
296 .addComponent(jButtonSugestoesParaTitle)
    )
    .addPreferredGap(javax.swing.LayoutStyle.
    ComponentPlacement.UNRELATED)
298 .addComponent(jSeparator2, javax.swing.
    GroupLayout.PREFERRED_SIZE, 10, javax.
    swing.GroupLayout.PREFERRED_SIZE)
    .addPreferredGap(javax.swing.LayoutStyle.
    ComponentPlacement.RELATED)
300 .addGroup(layout.createParallelGroup(javax.
    swing.GroupLayout.Alignment.LEADING)
    .addComponent(jLabel16)
302 .addComponent(jLabel15))
    .addPreferredGap(javax.swing.LayoutStyle.
    ComponentPlacement.UNRELATED)
304 .addComponent(jTextFieldLanguage, javax.
    swing.GroupLayout.PREFERRED_SIZE, javax.
    swing.GroupLayout.DEFAULT_SIZE, javax.
    swing.GroupLayout.PREFERRED_SIZE)
    .addPreferredGap(javax.swing.LayoutStyle.
    ComponentPlacement.UNRELATED)
306 .addComponent(jSeparator3, javax.swing.
    GroupLayout.PREFERRED_SIZE, 10, javax.
    swing.GroupLayout.PREFERRED_SIZE)
    .addPreferredGap(javax.swing.LayoutStyle.
    ComponentPlacement.RELATED)
308 .addGroup(layout.createParallelGroup(javax.
    swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup())
310 .addComponent(jLabel1)
    .addPreferredGap(javax.swing.
    LayoutStyle.ComponentPlacement.
    RELATED)

```

```

312         .addGroup(layout.createParallelGroup
            (javax.swing.GroupLayout.
             Alignment.LEADING)
            .addComponent(jScrollPane1 ,
                javax.swing.GroupLayout.
                 PREFERRED_SIZE, 79, javax.
                 swing.GroupLayout.
                 PREFERRED_SIZE)
314         .addGroup(layout.
            createParallelGroup(javax.
             swing.GroupLayout.Alignment.
             BASELINE)
            .addComponent(
                JComboBoxDescriptionLang
                , javax.swing.
                 GroupLayout.
                 PREFERRED_SIZE, javax.
                 swing.GroupLayout.
                 DEFAULT_SIZE, javax.
                 swing.GroupLayout.
                 PREFERRED_SIZE)
316         .addComponent(
                JButtonSugestoesParaDescription
                )))
            .addComponent(jLabel17))
318     .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.UNRELATED)
    .addComponent(jSeparator4 , javax.swing.
        GroupLayout.PREFERRED_SIZE, 10, javax.
        swing.GroupLayout.PREFERRED_SIZE)
320     .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.RELATED)
    .addGroup(layout.createParallelGroup(javax.
        swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel2)
        .addComponent(jLabel18))
322
324     .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.RELATED)
    .addGroup(layout.createParallelGroup(javax.
        swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jComboBoxKeywordsLang ,
            javax.swing.GroupLayout.
             PREFERRED_SIZE, javax.swing.
             GroupLayout.DEFAULT_SIZE, javax.
             swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextFieldKeywords , javax.
            swing.GroupLayout.PREFERRED_SIZE,
            javax.swing.GroupLayout.DEFAULT_SIZE
            , javax.swing.GroupLayout.
             PREFERRED_SIZE)
326
328     .addComponent(
        JButtonSugestoesParaKeywords))

```

```

        .addContainerGap(195, Short.MAX_VALUE))
    );
} // </editor-fold> // GEN-END: initComponents
332
@Override
334 public PacoteComCampos
    retornarCamposGeradoresDeSugestoes()
{
336     ArrayList<String> campos_valores = new ArrayList<
        String>();
    campos_valores.add(jTextFieldTitle.getText());
338     campos_valores.add(jTextAreaDescription.getText());
    campos_valores.add(jTextFieldKeywords.getText());
340
    return new PacoteComCampos(util.TipoDeCategoria.
        GeneralParaOBAA, campos_valores);
342 }

344 private void jButtonTableIdentifierAddActionPerformed(
    java.awt.event.ActionEvent evt) { // GEN-FIRST:
    event_jButtonTableIdentifierAddActionPerformed
    if(jTextFieldIdentifier.getRowCount() == 10)
346     {
        JOptionPane.showMessageDialog(this, "Você não
            pode ter mais de 10 linhas na tabela!", "
            Aviso", JOptionPane.WARNING.MESSAGE);
348         return;
    }
    DefaultTableModel model = (DefaultTableModel)
        jTextFieldIdentifier.getModel();
    model.addRow(new String[2]);
352 } // GEN-LAST:
    event_jButtonTableIdentifierAddActionPerformed

354 private void jButtonTableIdentifierDelActionPerformed(
    java.awt.event.ActionEvent evt) { // GEN-FIRST:
    event_jButtonTableIdentifierDelActionPerformed
    if(jTextFieldIdentifier.getRowCount() == 1)
356     {
        JOptionPane.showMessageDialog(this, "Você não
            pode ter deletar a última linha de uma
            tabela!", "Aviso", JOptionPane.
            WARNING.MESSAGE);
358         return;
    }
    if(jTextFieldIdentifier.getSelectedRow() == -1)
360     {
        JOptionPane.showMessageDialog(this, "Você não
            precisa selecionar a linha que deseja
            deletar!", "Aviso", JOptionPane.
            WARNING.MESSAGE);
362         return;
    }

```



```

364     }

366     DefaultTableModel model = (DefaultTableModel)
        jTableIdentifier.getModel();
        model.removeRow(jTableIdentifier.getSelectedRow());
368 }//GEN-LAST:
        event_jButtonTableIdentifierDelActionPerformed

370 private void jButtonSugestoesParaTitleActionPerformed(
        java.awt.event.ActionEvent evt) { //GEN-FIRST:
        event_jButtonSugestoesParaTitleActionPerformed
        profile.gerarSugestoes(TipoDeSugestao.SUGESTAO_TITLE
        );
372 }//GEN-LAST:
        event_jButtonSugestoesParaTitleActionPerformed

374 private void
        jButtonSugestoesParaDescriptionActionPerformed(java.
        awt.event.ActionEvent evt) { //GEN-FIRST:
        event_jButtonSugestoesParaDescriptionActionPerformed
        profile.gerarSugestoes(TipoDeSugestao.
        SUGESTAO_DESCRIPTION);
376 }//GEN-LAST:
        event_jButtonSugestoesParaDescriptionActionPerformed

378 private void jButtonSugestoesParaKeywordsActionPerformed
        (java.awt.event.ActionEvent evt) { //GEN-FIRST:
        event_jButtonSugestoesParaKeywordsActionPerformed
        profile.gerarSugestoes(TipoDeSugestao.
        SUGESTAO_KEYWORDS);
380 }//GEN-LAST:
        event_jButtonSugestoesParaKeywordsActionPerformed

382 // Variables declaration - do not modify//GEN-BEGIN:
        variables
private javax.swing.JButton
        jButtonSugestoesParaDescription;
384 private javax.swing.JButton jButtonSugestoesParaKeywords
        ;
private javax.swing.JButton jButtonSugestoesParaTitle;
386 private javax.swing.JButton jButtonTableIdentifierAdd;
private javax.swing.JButton jButtonTableIdentifierDel;
388 private javax.swing.JComboBox jComboBoxDescriptionLang;
private javax.swing.JComboBox jComboBoxKeywordsLang;
390 private javax.swing.JComboBox jComboBoxTitleLang;
private javax.swing.JLabel jLabel1;
392 private javax.swing.JLabel jLabel10;
private javax.swing.JLabel jLabel11;
394 private javax.swing.JLabel jLabel12;
private javax.swing.JLabel jLabel13;
396 private javax.swing.JLabel jLabel14;
private javax.swing.JLabel jLabel15;

```

```

398     private javax.swing.JLabel jLabel16;
399     private javax.swing.JLabel jLabel17;
400     private javax.swing.JLabel jLabel18;
401     private javax.swing.JLabel jLabel2;
402     private javax.swing.JLabel jLabel7;
403     private javax.swing.JLabel jLabel8;
404     private javax.swing.JLabel jLabel9;
405     private javax.swing.JScrollPane jScrollPane1;
406     private javax.swing.JScrollPane jScrollPane2;
407     private javax.swing.JSeparator jSeparator1;
408     private javax.swing.JSeparator jSeparator2;
409     private javax.swing.JSeparator jSeparator3;
410     private javax.swing.JSeparator jSeparator4;
411     private javax.swing.JTable jTableIdentifier;
412     private javax.swing.JTextArea jTextAreaDescription;
413     private javax.swing.JTextField jTextFieldKeywords;
414     private javax.swing.JTextField jTextFieldLanguage;
415     private javax.swing.JTextField jTextFieldTitle;
416     // End of variables declaration//GEN-END:variables
}

```

B.4.11 JPanelLifeCycleFull.java

```

package user_interface.categorias;

2
import java.awt.GridLayout;
4 import java.awt.Dimension;
import java.util.ArrayList;
6 import javax.swing.JOptionPane;
import javax.swing.JScrollPane;
8 import user_interface.Profile;

10 public class JPanelLifeCycleFull extends javax.swing.JPanel
    implements Categoria {

12     Profile profile;

14     public JPanelLifeCycleFull(Profile profile)
    {
16         initComponents();
        this.profile = profile;
18         jPanelContribute.setLayout(new GridLayout(10,1));
        jPanelContribute.add(new JPanelContribute());
20     }

22     @Override

```

```

public PacoteComCampos
    retornarCamposGeradoresDeSugestoes()
24 {
    return new PacoteComCampos(util.TipoDeCategoria.
        LifecycleFull, new ArrayList<String>());
26 }

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated
    Code">//GEN-BEGIN: initComponents
30 private void initComponents() {

32     jLabel1 = new javax.swing.JLabel();
    jLabel8 = new javax.swing.JLabel();
34     jTextFieldVersion = new javax.swing.JTextField();
    jComboBoxVersionLang = new javax.swing.JComboBox();
36     jLabel2 = new javax.swing.JLabel();
    jComboBoxStatus = new javax.swing.JComboBox();
38     jLabel3 = new javax.swing.JLabel();
    jLabel9 = new javax.swing.JLabel();
40     jSeparator1 = new javax.swing.JSeparator();
    jSeparator2 = new javax.swing.JSeparator();
42     jScrollPane1 = new javax.swing.JScrollPane();
    jPanelContribute = new javax.swing.JPanel();
44     jButtonAddContribute = new javax.swing.JButton();

46     jLabel1.setFont(new java.awt.Font("Tahoma", 1, 14));
        // NOI18N
    jLabel1.setText("1. Version:");
48

    jLabel8.setIcon(new javax.swing.ImageIcon(getClass().
        getResource("/img/help.png"))); // NOI18N
50     jLabel8.setToolTipText("<html>The edition of this
        learning object.<br><br>\nEx.: 1.2.alpha</html>"
        );

52     jTextFieldVersion.setBackground(new java.awt.Color
        (204, 255, 204));

54     jComboBoxVersionLang.setEditable(true);
    jComboBoxVersionLang.setModel(new javax.swing.
        DefaultComboBoxModel(new String[] { "en", "pt"
        }));

56     jLabel2.setFont(new java.awt.Font("Tahoma", 1, 14));
        // NOI18N
58     jLabel2.setText("2. Status:");

60     jComboBoxStatus.setModel(new javax.swing.
        DefaultComboBoxModel(new String[] { "draft", "
        final", "revised", "unavailable" }));
    jComboBoxStatus.setSelectedIndex(1);

```

```

62     jLabel3.setFont(new java.awt.Font("Tahoma", 1, 14));
        // NOI18N
64     jLabel3.setText("3. Contribute:");

66     jLabel9.setIcon(new javax.swing.ImageIcon(getClass()
        .getResource("/img/help.png"))); // NOI18N
        jLabel9.setToolTipText("<html>Those entities (i.e.,
        people, organizations)<br>\nthat have
        contributed to the state of this<br>\nlearning
        object during its life cycle <br>\n(e.g.,
        creation, edits, publication).</html>");

68     jPanelContribute.setBackground(new java.awt.Color
        (204, 204, 204));

70     javax.swing.GroupLayout jPanelContributeLayout = new
        javax.swing.GroupLayout(jPanelContribute);
72     jPanelContribute.setLayout(jPanelContributeLayout);
        jPanelContributeLayout.setHorizontalGroup(
74         jPanelContributeLayout.createParallelGroup(javax.
        swing.GroupLayout.Alignment.LEADING)
        .addGroup(0, 1096, Short.MAX_VALUE)
76     );
        jPanelContributeLayout.setVerticalGroup(
78         jPanelContributeLayout.createParallelGroup(javax.
        swing.GroupLayout.Alignment.LEADING)
        .addGroup(0, 555, Short.MAX_VALUE)
80     );

82     jScrollPane1.setViewportView(jPanelContribute);
        jPanelContribute.setAccessibleContext().
        setAccessibleDescription("");

84     jButtonAddContribute.setText("+");
86     jButtonAddContribute.addActionListener(new java.awt.
        event.ActionListener() {
        public void actionPerformed(java.awt.event.
            ActionEvent evt) {
88         jButtonAddContributeActionPerformed(evt);
        }
90     });

92     javax.swing.GroupLayout layout = new javax.swing.
        GroupLayout(this);
        this.setLayout(layout);
94     layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
96     .addGroup(layout.createSequentialGroup()
        .addGap(0, 0, 0)

```

```

98         .addGroup(layout.createParallelGroup(javax.
           swing.GroupLayout.Alignment.LEADING)
           .addGroup(layout.createParallelGroup(
             javax.swing.GroupLayout.Alignment.
100             TRAILING, false)
             .addComponent(jSeparator2)
             .addComponent(jSeparator1)
102             .addGroup(layout.createParallelGroup
              (javax.swing.GroupLayout.
                Alignment.LEADING)
                .addGroup(layout.
                  createSequentialGroup())
104                  .addComponent(jLabel1)
                  .addPreferredGap(javax.swing
                    .LayoutStyle.
                      ComponentPlacement.
                        RELATED)
106                  .addComponent(jLabel8)
                  .addPreferredGap(javax.swing
                    .LayoutStyle.
                      ComponentPlacement.
                        RELATED)
108                  .addComponent(
                    jTextFieldVersion, javax
                      .swing.GroupLayout.
                        PREFERRED_SIZE, 202,
                        javax.swing.GroupLayout.
                          PREFERRED_SIZE)
                    .addPreferredGap(javax.swing
                      .LayoutStyle.
                        ComponentPlacement.
                          RELATED)
110                  .addComponent(
                    jComboBoxVersionLang,
                    javax.swing.GroupLayout.
                      PREFERRED_SIZE, 54,
                    javax.swing.GroupLayout.
                      PREFERRED_SIZE))
                .addGroup(layout.
                  createSequentialGroup())
112                  .addComponent(jLabel2)
                  .addPreferredGap(javax.swing
                    .LayoutStyle.
                      ComponentPlacement.
                        UNRELATED)
114                  .addComponent(
                    jComboBoxStatus, javax.
                      swing.GroupLayout.
                        PREFERRED_SIZE, javax.
                        swing.GroupLayout.
                          DEFAULT_SIZE, javax.
                        swing.GroupLayout.

```

```

        PREFERRED_SIZE))
    .addGroup(layout .
        createSequentialGroup()
116      .addComponent(jLabel3)
        .addPreferredGap(javax.swing
            .LayoutStyle .
            ComponentPlacement .
            RELATED)
118      .addComponent(jLabel9)
        .addPreferredGap(javax.swing
            .LayoutStyle .
            ComponentPlacement .
            UNRELATED)
120      .addComponent(
        jButtonAddContribute)))
    .addComponent(jScrollPane1 , javax.swing .
        GroupLayout.PREFERRED_SIZE, 710,
        javax.swing .GroupLayout .
        PREFERRED_SIZE))
122      .addContainerGap(421 , Short.MAX_VALUE))
    );
124    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing .
            GroupLayout.Alignment.LEADING)
126      .addGroup(layout.createSequentialGroup()
        .addContainerGap()
128      .addGroup(layout.createParallelGroup(javax .
            swing .GroupLayout.Alignment.LEADING)
        .addComponent(jLabel8)
130      .addComponent(jLabel11)
        .addGroup(layout.createParallelGroup(
            javax.swing .GroupLayout.Alignment .
            BASELINE)
132      .addComponent(jTextFieldVersion ,
            javax.swing .GroupLayout .
            PREFERRED_SIZE, javax.swing .
            GroupLayout.DEFAULT_SIZE, javax .
            swing .GroupLayout.PREFERRED_SIZE
            )
        .addComponent(jComboBoxVersionLang ,
            javax.swing .GroupLayout .
            PREFERRED_SIZE, javax.swing .
            GroupLayout.DEFAULT_SIZE, javax .
            swing .GroupLayout.PREFERRED_SIZE
            )))
134      .addPreferredGap(javax.swing .LayoutStyle .
        ComponentPlacement.UNRELATED)
    .addComponent(jSeparator2 , javax.swing .
        GroupLayout.PREFERRED_SIZE, 10, javax .
        swing .GroupLayout.PREFERRED_SIZE)
136      .addPreferredGap(javax.swing .LayoutStyle .
        ComponentPlacement.RELATED)

```

```

        .addGroup(layout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.BASELINE)
138         .addComponent(jLabel2)
            .addComponent(jComboBoxStatus, javax.
                swing.GroupLayout.PREFERRED_SIZE,
                javax.swing.GroupLayout.DEFAULT_SIZE
                , javax.swing.GroupLayout.
                    PREFERRED_SIZE))
140         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addComponent(jSeparator1, javax.swing.
            GroupLayout.PREFERRED_SIZE, 10, javax.
            swing.GroupLayout.PREFERRED_SIZE)
142         .addPreferredGap(javax.swing.LayoutStyle.
            ComponentPlacement.RELATED)
        .addGroup(layout.createParallelGroup(javax.
            swing.GroupLayout.Alignment.LEADING)
144         .addComponent(jLabel9)
            .addComponent(jLabel3)
146         .addComponent(jButtonAddContribute))
        .addGap(18, 18, 18)
148         .addComponent(jScrollPane1, javax.swing.
            GroupLayout.PREFERRED_SIZE, 451, javax.
            swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(109, Short.MAX_VALUE))
150     );
151 }// </editor-fold>//GEN-END: initComponents

private void jButtonAddContributeActionPerformed(java.
    awt.event.ActionEvent evt) { //GEN-FIRST:
152     event_jButtonAddContributeActionPerformed
154     if (jPanelContribute.getComponentCount() == 30)
        {
156         JOptionPane.showMessageDialog(this, "VocÃª nÃ£o
            pode ter mais de 30 itens!", "Aviso",
            JOptionPane.WARNING_MESSAGE);
            return;
158         }
        jPanelContribute.add(new JPanelContribute());
160 }//GEN-LAST: event_jButtonAddContributeActionPerformed

// Variables declaration - do not modify//GEN-BEGIN:
162     variables
private javax.swing.JButton jButtonAddContribute;
164 private javax.swing.JComboBox jComboBoxStatus;
private javax.swing.JComboBox jComboBoxVersionLang;
166 private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
168 private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel8;
170 private javax.swing.JLabel jLabel9;
private javax.swing.JPanel jPanelContribute;

```

```

172     private javax.swing.JScrollPane jScrollPane1;
173     private javax.swing.JSeparator jSeparator1;
174     private javax.swing.JSeparator jSeparator2;
175     private javax.swing.JTextField jTextFieldVersion;
176     // End of variables declaration//GEN-END:variables
}

```

B.4.12 JPanelRightsFull.java

```

package user_interface.categorias;

2
import user_interface.Profile;
4 import java.util.ArrayList;

6 public class JPanelRightsFull extends javax.swing.JPanel
    implements Categoria{

8     Profile profile;

10     public JPanelRightsFull(Profile profile) {
        initComponents();
12         this.profile = profile;
    }

14     @Override
16     public PacoteComCampos
        retornarCamposGeradoresDeSugestoes()
    {
18         return new PacoteComCampos( util.TipoDeCategoria.
            RightsFull, new ArrayList<String>());
    }

20     @SuppressWarnings("unchecked")
22     // <editor-fold defaultstate="collapsed" desc="Generated
        Code"> //GEN-BEGIN: initComponents
    private void initComponents() {

24         jLabel1 = new javax.swing.JLabel();
26         jLabel8 = new javax.swing.JLabel();
        jComboBoxCost = new javax.swing.JComboBox();
28         jSeparator1 = new javax.swing.JSeparator();
        jLabel2 = new javax.swing.JLabel();
30         jLabel9 = new javax.swing.JLabel();
        jComboBoxCopyright = new javax.swing.JComboBox();
32         jSeparator2 = new javax.swing.JSeparator();
        jLabel3 = new javax.swing.JLabel();
34         jLabel10 = new javax.swing.JLabel();
    }

```



```

jScrollPane1 = new javax.swing.JScrollPane();
jTextAreaDescription = new javax.swing.JTextArea();
jComboBoxDescriptionLang = new javax.swing.JComboBox()
    );
jLabel1.setFont(new java.awt.Font("Tahoma", 2, 14));
    // NOI18N
jLabel1.setText("1. Cost:");

jLabel8.setIcon(new javax.swing.ImageIcon(getClass()
    .getResource("/img/help.png"))); // NOI18N
jLabel8.setToolTipText("<html>Whether use of this
    learning object requires payment.</html>");

jComboBoxCost.setModel(new javax.swing.
    DefaultComboBoxModel(new String[] { "no", "yes"
    }));

jLabel2.setFont(new java.awt.Font("Tahoma", 2, 14));
    // NOI18N
jLabel2.setText("2. Copyright and Other Restrictions
    :");

jLabel9.setIcon(new javax.swing.ImageIcon(getClass()
    .getResource("/img/help.png"))); // NOI18N
jLabel9.setToolTipText("<html>Whether copyright or
    other restrictions apply<br>\nto the use of this
    learning object.</html>");

jComboBoxCopyright.setModel(new javax.swing.
    DefaultComboBoxModel(new String[] { "no", "yes"
    }));

jLabel3.setFont(new java.awt.Font("Tahoma", 2, 14));
    // NOI18N
jLabel3.setText("3. Description:");

jLabel10.setIcon(new javax.swing.ImageIcon(getClass()
    .getResource("/img/help.png"))); // NOI18N
jLabel10.setToolTipText("<html>Comments on the
    conditions of use of this learning object.<br>
    <br>\nEx.: \"Use of this learning object is only
    permitted after (...)\"</html>");

jTextAreaDescription.setColumns(20);
jTextAreaDescription.setLineWrap(true);
jTextAreaDescription.setRows(5);
jTextAreaDescription.setWrapStyleWord(true);
jScrollPane1.setViewportView(jTextAreaDescription);

jComboBoxDescriptionLang.setEditable(true);

```

```
68 jComboBoxDescriptionLang.setModel(new javax.swing.  
DefaultComboBoxModel(new String[] { "en", "pt"  
}));  
  
70 javax.swing.GroupLayout layout = new javax.swing.  
GroupLayout(this);  
this.setLayout(layout);  
72 layout.setHorizontalGroup(  
    layout.createParallelGroup(javax.swing.  
        GroupLayout.Alignment.LEADING)  
74 .addGroup(layout.createSequentialGroup()  
        .addGap(10, 10, 10)  
76 .addGroup(layout.createParallelGroup(javax.swing.  
            GroupLayout.Alignment.LEADING)  
            .addComponent(jSeparator1, javax.swing.  
                GroupLayout.DEFAULT_SIZE, 1019,  
                    Short.MAX_VALUE)  
78 .addGroup(layout.createParallelGroup(javax.swing.  
                GroupLayout.Alignment.LEADING)  
                .addGroup(layout.createSequentialGroup()  
50 .addGroup(layout.createParallelGroup(javax.swing.  
                        GroupLayout.Alignment.LEADING)  
                        .addGroup(layout.  
                            createSequentialGroup()  
82 .addComponent(jLabel1)  
                            .addPreferredGap(javax.swing.  
                                ComponentPlacement.RELATED)  
84 .addComponent(jLabel8)  
                            .addPreferredGap(javax.swing.  
                                ComponentPlacement.UNRELATED)  
86 .addComponent(jComboBoxCost, javax.swing.  
                                    GroupLayout.PREFERRED_SIZE, javax.  
                                        swing.GroupLayout.  
                                            DEFAULT_SIZE, javax.  
                                                swing.GroupLayout.  
                                                    PREFERRED_SIZE))  
                            .addGroup(layout.  
                                createSequentialGroup()  
88 .addComponent(jLabel2)  
                                .addPreferredGap(javax.swing.  
                                    ComponentPlacement.RELATED)  
90 .addComponent(jLabel9)
```

```

        .addPreferredGap(javax.swing
            .LayoutStyle.
            ComponentPlacement.
            UNRELATED)
92    .addComponent(
        JComboBoxCopyright,
        javax.swing.GroupLayout.
        PREFERRED_SIZE, javax.
        swing.GroupLayout.
        DEFAULT_SIZE, javax.
        swing.GroupLayout.
        PREFERRED_SIZE))
    .addGroup(layout.
        createSequentialGroup())
94    .addComponent(jLabel3)
        .addPreferredGap(javax.swing
            .LayoutStyle.
            ComponentPlacement.
            RELATED)
96    .addComponent(jLabel10))
    .addGroup(layout.
        createSequentialGroup())
98    .addComponent(jScrollPane1,
        javax.swing.GroupLayout.
        PREFERRED_SIZE, 386,
        javax.swing.GroupLayout.
        PREFERRED_SIZE)
        .addPreferredGap(javax.swing
            .LayoutStyle.
            ComponentPlacement.
            UNRELATED)
100    .addComponent(
        JComboBoxDescriptionLang
        , javax.swing.
        GroupLayout.
        PREFERRED_SIZE, 54,
        javax.swing.GroupLayout.
        PREFERRED_SIZE)))
        .addGap(0, 0, Short.MAX_VALUE)))
102    .addContainerGap())
    );
104    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.
            GroupLayout.Alignment.LEADING)
106    .addGroup(layout.createSequentialGroup())
        .addContainerGap())
108    .addGroup(layout.createParallelGroup(javax.
        swing.GroupLayout.Alignment.LEADING)
        .addComponent(jComboBoxCost, javax.swing
            .GroupLayout.PREFERRED_SIZE, javax.
            swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.

```

```

        PREFERRED.SIZE)
110     .addComponent(jLabel8)
        .addComponent(jLabel1))
112     .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.UNRELATED)
    .addComponent(jSeparator1, javax.swing.
        GroupLayout.PREFERRED.SIZE, 10, javax.
        swing.GroupLayout.PREFERRED.SIZE)
114     .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.RELATED)
    .addGroup(layout.createParallelGroup(javax.
        swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel2)
        .addComponent(jLabel9)
118        .addComponent(jComboBoxCopyright, javax.
        swing.GroupLayout.PREFERRED.SIZE,
        javax.swing.GroupLayout.DEFAULT.SIZE
        , javax.swing.GroupLayout.
        PREFERRED.SIZE))
    .addGap(18, 18, 18)
120     .addComponent(jSeparator2, javax.swing.
        GroupLayout.PREFERRED.SIZE, 10, javax.
        swing.GroupLayout.PREFERRED.SIZE)
    .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.RELATED)
122     .addGroup(layout.createParallelGroup(javax.
        swing.GroupLayout.Alignment.LEADING)
        .addComponent(jLabel3)
124        .addComponent(jLabel10))
    .addPreferredGap(javax.swing.LayoutStyle.
        ComponentPlacement.RELATED)
126     .addGroup(layout.createParallelGroup(javax.
        swing.GroupLayout.Alignment.LEADING)
        .addComponent(jComboBoxDescriptionLang,
        javax.swing.GroupLayout.
        PREFERRED.SIZE, javax.swing.
        GroupLayout.DEFAULT.SIZE, javax.
        swing.GroupLayout.PREFERRED.SIZE)
128        .addComponent(jScrollPane1, javax.swing.
        GroupLayout.PREFERRED.SIZE, 75,
        javax.swing.GroupLayout.
        PREFERRED.SIZE))
    .addContainerGap(493, Short.MAX_VALUE))
130 );
} // </editor-fold> // GEN-END: initComponents
132 // Variables declaration - do not modify // GEN-BEGIN:
    variables
    private javax.swing.JComboBox jComboBoxCopyright;
134     private javax.swing.JComboBox jComboBoxCost;
    private javax.swing.JComboBox jComboBoxDescriptionLang;
136     private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel10;

```

```

138     private javax.swing.JLabel jLabel2;
139     private javax.swing.JLabel jLabel3;
140     private javax.swing.JLabel jLabel8;
141     private javax.swing.JLabel jLabel9;
142     private javax.swing.JScrollPane jScrollPane1;
143     private javax.swing.JSeparator jSeparator1;
144     private javax.swing.JSeparator jSeparator2;
145     private javax.swing.JTextArea jTextAreaDescription;
146     // End of variables declaration//GEN-END:variables
}

```

B.4.13 PacoteComCampos.java

```

package user_interface.categorias;

2
import util.TipoDeCategoria;
4 import java.util.ArrayList;
import java.util.List;
6 import util.NLPTool;
import util.Rank;
8
public class PacoteComCampos
10 {
    public TipoDeCategoria categoria;
12     public ArrayList<String> valores;

14     public PacoteComCampos(TipoDeCategoria categoria ,
        ArrayList<String> campos_valores)
    {
16         this.categoria = categoria;
        this.valores = campos_valores;
18     }

20     public List<String> processarKeywords() // INCOMPLETO:
        Apenas para GeneralParaOBAA
    {
22         if (categoria == TipoDeCategoria.GeneralParaOBAA)
        {
24             ArrayList<String> title;
            ArrayList<String> description;
26             ArrayList<String> keywords = new ArrayList<
                String>();
            ArrayList<String> keywords_final = new ArrayList<
                String>();

28             //System.out.println("para processar: "+valores)
                ;

```

```

30         title = NLPTool.stopWordsRemoval(NLPTool.
           toLowerCase(valores.get(0)));
32
           description = NLPTool.stopWordsRemoval(NLPTool.
           toLowerCase(valores.get(1)));
34
           keywords.add(valores.get(2));
36         keywords = NLPTool.separateStringsByDotComma(
           keywords);

           keywords_final.addAll(title);
           keywords_final.addAll(description);
           keywords_final.addAll(keywords);
           NLPTool.removeEmptyStrings(keywords_final);
42
           //System.out.println(" title: "+title);
           //System.out.println(" description: "+description
           );
           //System.out.println(" keywords: "+keywords);
           //System.out.println(" keywords_final: "+
           keywords_final);
46
           keywords_final = NLPTool.toLowerCase(
           keywords_final);

           return Rank.rankByQuant(8, keywords_final);
50     }
           else
52     {
           return new ArrayList<String>();
54     }
56 }
}

```
