```
 1 package core;
 2
 3
 4 import java.awt.Component;
10
11 public class ModelManager {
12
13     private DiachronicGraph diachronicGraph=null;
14
15
16     public ModelManager(){}
17
18     public void clear(){
19
20         if(diachronicGraph!=null)
21             diachronicGraph.clear();
22
23     }
24
25     public String getTargetFolder(){
26
27         return diachronicGraph.getTargetFolder();
28
29     }
30
31     public void stopConvergence(){
32
33         diachronicGraph.stopConvergence();
34
35     }
36
37     public void saveVertexCoordinates(String projectIni) throws IOException{
38
39         diachronicGraph.saveVertexCoordinates(projectIni);
40
41     }
42
43     public void setTransformingMode(){
44
45         diachronicGraph.setTransformingMode();
46
47     }
48
49     public void setPickingMode(){
50
51         diachronicGraph.setPickingMode();
52
53     }
54
55
56     public void visualize(VisualizationViewer< String, String> vv,String
   projectIni,String targetFolder,int edgeType) throws IOException {
57
58         diachronicGraph.saveVertexCoordinates(projectIni);
59         diachronicGraph.visualizeIndividualDBVersions(vv,targetFolder,edgeType);
60         diachronicGraph.visualizeDiachronicGraph(vv);
61
62     }
63
64     public Component loadProject(String sql,String xml,String graphml, double
   frameX,double frameY,double scaleX,double scaleY,double centerX,double
   centerY,String targetFolder,int edgeType) throws Exception{
65
66         diachronicGraph = new
```

```
DiachronicGraph(sql,xml,graphml,targetFolder,edgeType,frameX,frameY,scaleX,scaleY,c
enterX,centerY);
67
68          return diachronicGraph.show();
69
70      }
71
72      public void generateVertexDegreeReport(String targetFolder) throws
FileNotFoundException{
73
74          diachronicGraph.generateVertexDegreeReport(targetFolder);
75
76      }
77
78      public void generateVertexInDegreeReport(String targetFolder) throws
FileNotFoundException{
79
80          diachronicGraph.generateVertexInDegreeReport(targetFolder);
81
82      }
83
84      public void generateVertexOutDegreeReport(String targetFolder) throws
FileNotFoundException{
85
86          diachronicGraph.generateVertexOutDegreeReport(targetFolder);
87
88      }
89
90      public void generateVertexBetweennessReport(String targetFolder) throws
FileNotFoundException{
91
92          diachronicGraph.generateVertexBetweennessReport(targetFolder);
93
94      }
95
96      public void generateEdgeBetweennessReport(String targetFolder) throws
FileNotFoundException{
97
98          diachronicGraph.generateEdgeBetweennessReport(targetFolder);
99
100     }
101
102     public void generateGraphDiameterReport(String targetFolder) throws
FileNotFoundException{
103
104         diachronicGraph.generateGraphDiameterReport(targetFolder);
105
106     }
107
108     public void generateVertexCountReport(String targetFolder) throws
FileNotFoundException{
109
110         diachronicGraph.generateVertexCountReport(targetFolder);
111
112     }
113
114     public void generateEdgeCountReport(String targetFolder) throws
FileNotFoundException{
115
116         diachronicGraph.generateEdgeCountReport(targetFolder);
117
118     }
119
120     public Component refresh(double forceMult, int repulsionRange) {
```

```
121
122         return diachronicGraph.refresh(forceMult,repulsionRange);
123
124     }
125
126     public void generateConnectedComponentsCountReport(String targetFolder) throws
    FileNotFoundException{
127
128         diachronicGraph.generateConnectedComponentsCountReport(targetFolder);
129
130     }
131
132     public void generateClusteringCoefficientReport(String targetFolder) throws
    FileNotFoundException{
133
134         diachronicGraph.generateClusteringCoefficientReport(targetFolder);
135
136     }
137
138     public void generateVertexCountReportForGCC(String targetFolder) throws
    FileNotFoundException {
139
140         diachronicGraph.generateVertexCountReportForGcc(targetFolder);
141
142     }
143
144     public void generateEdgeCountReportForGCC(String targetFolder) throws
    FileNotFoundException {
145
146         diachronicGraph.generateEdgeCountReportForGcc(targetFolder);
147
148     }
149
150
151 }
152
```