

@startuml

left to right direction

actor "Any User = Author/Editor or Reader" as u

actor "Author/Editor" as e

package "Non-interactive and interactive systems (Manager/Editor/...)" {

usecase "XXX Layout whole package without considering visual models" as UC1

usecase "XXX Layout concrete semantic model" as UC2

usecase "XXX Layout concrete visual model" as UC3

usecase "Set layout configuration" as UC9

usecase "High-level" as UC9a

usecase "Layout for Specification" as UC9a1

usecase "Layout for Slides" as UC9a2

usecase "Memorable layout" as UC9a3

usecase "Layout for Overview (for example only main classes showed normal and the rest minified or no detail)" as UC9a4

usecase "Layout for familiarization" as UC9a5

usecase "(Don't) Mix semantic models" as UC9a6

usecase "Enhance clustered (meaning in layout sense) classes" as UC9a7

usecase "Enhance main entities" as UC9a8

usecase "Preferred aspect ratio" as UC9a9

usecase "Layout everything/only visible part" as UC9a10

usecase "Thoroughness - Time/Effort spend on layout" as UC9a11

usecase "Set target demographic technical knowledge" as UC9a12

usecase "Set detail level" as UC9a12a

usecase "Set technical conventions enforcement" as UC9a12b

usecase "Mental map preservation" as UC9a13

usecase "Mid-Level (graph aesthetic criteria)" as UC9b

usecase "Orthogonality" as UC9b1

usecase "Minimize edge crossings" as UC9b2

usecase "Maximize angles at edge crossings" as UC9b3

usecase "Minimize area" as UC9b4

usecase "Symmetry" as UC9b5

usecase "Minimize longest edge length" as UC9b6

usecase "Uniform node distribution" as UC9b7

usecase "Edge uniformity" as UC9b8

usecase "Clusterization (node density)" as UC9b9

usecase "Low-Level" as UC9c

```
usecase "Minimal node distance" as UC9c1
usecase "Profiled class' position against source class" as UC9c2
usecase "Profiled class' minimal distance from source class" as UC9c3
usecase "Draw Generalization in layered style" as UC9c4
usecase "Preferred direction of generalization" as UC9c5
}
u --> UC1
u --> UC2
u --> UC3
u --> UC9
```

```
UC9 --> UC9a : <<includes>>
UC9 <-- UC9b : <<extends>>
UC9 <-- UC9c : <<extends>>
```

```
UC9a --> UC9a1 : <<includes>>
UC9a --> UC9a2 : <<includes>>
UC9a --> UC9a3 : <<includes>>
UC9a --> UC9a4 : <<includes>>
UC9a --> UC9a5 : <<includes>>
UC9a --> UC9a6 : <<includes>>
UC9a --> UC9a7 : <<includes>>
UC9a --> UC9a8 : <<includes>>
UC9a --> UC9a9 : <<includes>>
UC9a --> UC9a10 : <<includes>>
UC9a --> UC9a11 : <<includes>>
UC9a --> UC9a12 : <<includes>>
UC9a --> UC9a13 : <<includes>>
```

```
UC9a12 --> UC9a12a : <<includes>>
UC9a12 --> UC9a12b : <<includes>>
```

```
UC9b --> UC9b1 : <<includes>>
UC9b --> UC9b2 : <<includes>>
UC9b --> UC9b3 : <<includes>>
UC9b --> UC9b4 : <<includes>>
UC9b --> UC9b5 : <<includes>>
UC9b --> UC9b6 : <<includes>>
UC9b --> UC9b7 : <<includes>>
UC9b --> UC9b8 : <<includes>>
UC9b --> UC9b9 : <<includes>>
```

```
UC9c --> UC9c1 : <<includes>>
UC9c --> UC9c2 : <<includes>>
UC9c --> UC9c3 : <<includes>>
```

UC9c --> UC9c4 : <<includes>>  
UC9c --> UC9c5 : <<includes>>

```
package "Interactive system (Editor)" {  
  usecase "XXX Get cluster suggestions (maybe based on configuration?)" as UC4  
  usecase "XXX Create manual cluster by choosing classes (Both textual and graphical  
form?)" as UC5  
  usecase "XXX Interactive layout by marking checkbox - addition of classes/edges moves  
existing classes as little as possible" as UC6
```

```
  usecase "Set Constraints inside editor to enforce on every relayout" as UC7  
  usecase "(Set relative class positions)" as UC7a  
  usecase "Align classes horizontally/vertically" as UC7b  
  usecase "Anchor class" as UC7c  
  usecase "Put selected classes close" as UC7d  
  usecase "(Select hub class - neighbors should be in the circle around it)" as UC7e
```

```
  usecase "Semi-automatic layouting in form of one-time only constraints" as UC8
```

```
  usecase "XXX Semi-automatic layouting - Put classes into grid" as UC10  
  usecase "Create diagram with only main classes" as UC11
```

```
  usecase "Create diagram with main class + chosen neighborhood (neighborhood = edge  
distance + manually selected classes)" as UC12  
  usecase "Create and layout diagram of profiles and near neighborhood" as UC13  
  usecase "(Once upon a time try completely relayout and give options if better)" as UC14  
  usecase "XXX Create  $2 \cdot n + 2$  diagrams of  $n$  clusters" as UC15
```

```
  usecase "(Set detail level on class manually)" as UC16
```

```
  usecase "(Show problematic classes based on graph aesthetics criteria)" as UC17
```

```
  usecase "Layout multiple diagrams as one (either creates new one or changes selected)"  
as UC18
```

```
  usecase "Create multiple layouts of same diagram with different level of detail" as UC19  
  usecase "Mark class as main" as UC20
```

```
}
```

UC7 --> UC7a : <<includes>>  
UC7 --> UC7b : <<includes>>  
UC7 --> UC7c : <<includes>>  
UC7 --> UC7d : <<includes>>  
UC7 --> UC7e : <<includes>>

UC7b <-- UC8 : <<includes>>  
UC7d <-- UC8 : <<includes>>  
UC7e <-- UC8 : <<includes>>

UC4 --> UC15 : <<includes>>

e --> UC4  
e --> UC6  
e --> UC7  
u --> UC8  
e --> UC10

e --> UC14  
e --> UC16  
e --> UC17  
u --> UC18  
e --> UC19  
e --> UC20

e --> UC5  
e --> UC11  
e --> UC12  
e --> UC13

```
package "Test Suite" {  
    usecase "Generate multiple layouts let user choose the best one" as UC21  
}  
@enduml
```