System Software Project

Text Editor

The goal of this project is to implement a text editor that can handle text files with different fonts and styles. It should be based on the simple editor presented in the course (http://ssw.jku.at/Teaching/Lectures/SSW/Editor.zip). Since this prototype is written in Java the natural choice for the implementation language of your editor is Java as well, but you can also use C# or C++ instead. The project *cannot* be implemented in teams, i.e. you have to implement it alone.

Your editor should have the same basic functionality as the prototype editor, i.e.:

- It should be able to read and display a text file. Long lines are not wrapped but simply clipped at the right margin.
- The user should be able to type and delete characters in the text.
- There should be a caret and a selection.
- The user should be able to scroll.

In addition to that, your editor should have the following features:

- The text data structure should be managed as a piece list (not as a gap text like in the prototype).
- Words should be selectable with a double click.
- The user should be able to set the font, size and style of arbitrary text regions via menu commands. Note that a line can have characters of different height so the line metrics have to be computed on a per-character basis, and also the drawing of the lines becomes more complicated.
- There should be menu commands for Cut, Copy and Paste operations. This requires an application-wide clipboard to be implemented. When text is copied, fonts and styles should be copied as well.
- Fonts and styles should be stored (as meta-data) at the beginning of a text file as it was discussed in the course. Fonts and styles should be restored when the text is reopened. But your editor should also be able to open standard ASCII text files without such a header.
- There should be menu commands for opening and saving text files. If a new text file is opened via a menu command it should be displayed in a new viewer.
- There should be a Find command for searching the next occurrence of a search strings from the caret or position.

Note that you have to implement your own text class and your own viewer class, i.e. it is not allowed to use ready-made text components from a library. The correct handling of fonts and styles is an important part of your project.