Software Documentation : Java Generic HTML Documentation Panel Plugin

This page last changed on Jul 22, 2008 by ghostar.

1. Scope

1.1 Overview

The documentation for elements in the UML Tool is currently entered into a basic text box that has no formatting capabilities, as part of the Documentation Panel component. This component will provide the base for a new HTML-based documentation editor. This initial version will support rendering the HTML and basic editing, with bold, italics and underline supported for text. This component will be able to be configured in the UML Tool through configuration of the Documentation Panel.

1.1.1 Version

1.0

1.2 Logic Requirements

1.2.1 Implementation

The main class of this component should extend the DocumentationEditor class from the Documentation Panel component, allowing for an easy drop in replacement. The JEditorPane SWING class can be used to display the HTML.

1.2.2 Input and Output

The input and output of the component will be XHTML, allowing for easy inclusion into the XMI file-format used by the UML Tool

1.2.3 Invalid input

In the case of invalid XHTML being given to the panel to display, no exception must be thrown. Instead, the component must display the raw XHTML as plain text.

1.2.4 Editing capabilities

This component must support 3 editing operations: bold, italics, and underline. There should be 3 buttons at the top of the editor pane to flag as bold, italics, and underline. These buttons should contain place-holders for icons. These buttons should behave as they do in a standard text editor. For instance, if the bold button is pressed, it must stay pressed when typing. If text is selected and one of the options is toggled, that option should be applied to the selected text. For instance, if the user wants to have a sentence bolded, they can select the entire sentence, and then press the bold button.

1.2.4.1 Whitespace

Whitespace added by the user must be preserved when saving and loading the text in the pane.

1.2.5 Keyboard shortcuts

Keyboard shortcuts must also be defined for the 3 operations. For Windows and Linux platforms, the key combos should be "CTRL+b" for bold, "CTRL+i" for italics, and "CTRL+u" for underline. For Macintosh, the key combos should be "Command+b" for bold, "Command+i" for italics, and "Command+u" for underline.

1.2.6 Undo/Redo

Fine-grained undo and redo must be supported in the editor. For instance, if a user bolds a word and then wants to undo, they can undo just the bolding of the word, assuming it was the last operation done. Undo and redo must be available through keyboard shortcuts ("CTRL/Command + z" and "CTRL/Command + y").

1.2.6 Sample

Some sample code of a similar piece of functionality can be found here:

http://www.artima.com/forums/flat.jsp?forum=1

1.3 Required Algorithms

None

1.4 Example of the Software Usage

This component will provide a richer documentation editor for users of the UML Tool

1.5 Future Component Direction

Colors, fonts, links, tables and lists will all be added to the component in the future. If a designer wants to support any or all of those pieces of functionality, it would be considered very useful additional functionality.

2. Interface Requirements

2.1.1 Graphical User Interface Requirements

This component will be a SWING control.

2.1.2 External Interfaces

None

2.1.3 Environment Requirements

- Development language: Java 1.5
- Compile target: Java 1.5

2.1.4 Package Structure

com.topcoder.gui.panels.documentation.plugins.html

3. Software Requirements

3.1 Administration Requirements

3.1.1 What elements of the application need to be configurable?

None

3.2 Technical Constraints

3.2.1 Are there particular frameworks or standards that are required?

- HTML

3.2.2 TopCoder Software Component Dependencies:

- Documentation Panel http://software.topcoder.com/catalog/c_component.jsp?comp=24671506

3.2.3 Third Party Component, Library, or Product Dependencies:

3.2.4 QA Environment:

- Solaris 7
- RedHat Linux 7.1
- Windows 2000
- Windows 2003

3.3 Design Constraints

The component design and development solutions must adhere to the guidelines as outlined in the TopCoder Software Component Guidelines. Modifications to these guidelines for this component should be detailed below.

3.4 Required Documentation

3.4.1 Design Documentation

- Use-Case Diagram
- Class Diagram
- Sequence Diagram
- Component Specification

3.4.2 Help / User Documentation

- Design documents must clearly define intended component usage in the 'Documentation' tab of the TopCoder UML Tool.