

## Java SWING Closable Tabs 1.0 Requirements Specification

### 1. Scope

#### 1.1 Overview

The Java SWING Closable Tabs provide a tabbed pane where each tab can be directly closed. Each tab will display an "X" that the user can click to close a tab directly, and a context menu will be provided to perform other manipulations of the open tabs, like "Close all", and "Close other tabs".

#### 1.2 Logic Requirements

##### 1.2.1 Control layout

Prototype images are given that explain the overall layout and functionality of the side menu. Note that the custom look and feel **is not** part of the competition, as that will be handled externally. A placeholder should be used for the "X" image seen in the prototype images.

##### 1.2.2 JTabbedPane

The control for this component must be a JTabbedPane derivative, providing a drop-in replacement.

##### 1.2.3 Close functionality

The tabs must support a close button on each tab. When the mouse is hovered over the close "X" on a tab, there needs to be some visual indication that the "X" is selected. Note that this is not shown in the prototype. This should be a separate image that is settable through a property.

Tabs that are not currently displayed can also be closed by clicking the "X", without having to select or display the contents of the tab.



##### 1.2.4 Context menu

Note that there needs to be a context , right-click, menu for each tab that includes the following options:

- Close tab
  - This functionality is the same as clicking the "X"
- Close other tabs
  - This functionality closes all other tabs, leaving the selected tab open
- Close all tabs
  - This functionality closes every tab

Right clicking on a non-selected tab should not cause it to become visible or selected.

##### 1.2.5 Events

An event handler interface should be defined to allow for an application to perform actions when a tab or set of tabs are closed or added to the pane. This should be in addition to the current ChangeListener functionality of JTabbedPane.

##### 1.2.6 Borders

In the prototype images, it can be seen that there is a blue border drawn around the interior of the individual tab. This needs to be supported in the design. The border width and color must be



allowed to be programmatically configurable, as well as configurable through the Configuration Manager.

#### 1.2.7 Images

The component needs to easily allow the application to set the images shown for the close "X" on each tab, including separate images for selected and non-selected tabs. 4 separate images are needed:

- Selected tab, not hovered
- Selected tab, hover
- Deselected tab, not hovered
- Deselected tab, hover

These images must be allowed to be set both programmatically and through configuration.

### 1.3 Required Algorithms

None

### 1.4 Example of the Software Usage

This component will be used in the TopCoder UML Tool to house the diagram views for multiple diagrams. The user needs to be able to open and close tabs easily, and this component will help with that.

### 1.5 Future Component Direction

None

## 2. Interface Requirements

#### 2.1.1 Graphical User Interface Requirements

The tool should reflect the prototype given, using placeholders for images. The tab styles **do not** have to match the prototype, as the exact look and feel will be updated.

#### 2.1.2 External Interfaces

None

#### 2.1.3 Environment Requirements

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

#### 2.1.4 Namespace

com.topcoder.gui.closabletabbedpane

## 3. Software Requirements

### 3.1 Administration Requirements

#### 3.1.1 What elements of the application need to be configurable?

- The color and width of the internal border
- The images to use for the "X" in each state

### 3.2 Technical Constraints

#### 3.2.1 Are there particular frameworks or standards that are required?

None.

#### 3.2.2 TopCoder Software Component Dependencies:

None

**\*\*Please review the [TopCoder Software component catalog](#) for existing components that can be used in the design.**



### *3.2.3 Third Party Component, Library, or Product Dependencies:*

None

### *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*

XML documentation must provide sufficient information regarding component design and usage.