Change request log

# Team

Lake Domstead

Neither of my teammates responded to any emails or messages.

# Change Request

Change Request #4

In the File » Recent Files main menu of jEdit, the Clear Recent Files option on the bottom of the recent files allows to clear recent files (see Figure 4). Similarly, in the Utilities » Recent Directories main menu, there should be Clear Recent Directories option on the bottom of the recent directories allows to clear recent directories. You are requested to implement this feature to clear the recent directories list.

# Concept Location

Use the table below to describe each step you follow when performing concept location for this change request. In your description, include the following information when appropriate:

* IDE Features used (e.g., searching tool, dependency navigator, debugging, etc.)
* Queries used when searching
* System executions and input to the system
* Interactions with the system (e.g., pages visited)
* Classes visited
* The first class found to be changed (this is when concept location ends)

When there is a major decision/step in the process, include its rationale, i.e., why that decision/step was taken.

Make sure you time yourselves when going through this process and provide the total time spent below.

The following is an example of a concept location process for the change request "Color student schedule":

|  |  |  |
| --- | --- | --- |
| Step # | Description | Rationale |
| 1 | *I ran the system* |  |
| 2 | *I searched for “Clear Recent Files”* | *The text in the label should appear where the menu is created and thus the functionality I’d need to copy into the directory file* |
| 3 | *1 result was returned, it was the English properties file.* |  |
| 4 | *I then searched for “clear\_recent\_files.label” . Of the 10 results returned only 1 was a java file* | *The searched text was the variable name of the string I previously searched.* |
| 5 | *Inspecting the class the functionality and the menu building are right next to each other. A menu item is created and a listener is added.* | *This means there is also a listener for the Directories equivalent which will also need to be found.* |
| 6 | *I then searched for RecentDirectoriesProvider, of the 3 files the java file was the obvious choice.* | *The two menus already did very similar things are I hoped and was right that they shared a naming convention.* |
| 8 | *The file directory is stored in a data type called recentVector, however this does not exist in the directories equivalent. Instead, the directories are stored in a HistoryModel.* | *The RecentDirectoriesProvider was storing items and iterating through this data structure so I assumed this is where they were stored.* |

**Time spent (in minutes):** 35

# Impact Analysis

Use the table below to describe each step you follow when performing impact analysis for this change request. Include as many details as possible, including why classes are visited or why they are discarded from the estimated impact set.

Do not take the impact analysis of your changes lightly. Remember that any small change in the code could lead to large changes in the behavior of the system. Follow the impact analysis process covered in the class. Describe in details how you followed this process in the change request log. Provide details on how and why you finished the impact analysis process.

|  |  |  |
| --- | --- | --- |
| Step # | Description | Rationale |
| 1 | *I found the files/variables the Clear Recent Files button was using* | *Since the Clear Recent Directories button would do the same thing as the Clear Recent Files I figured they would have about the same impact area* |
| 2 | *The button wasn’t used by any other code. The only files it was using were the language property files. Each of the files was marked as “To Change” as was the mirrored RecentDirectoriesProvider class.* | *I was correct, the mirrored functionality would share mirrored impact area and only required small additional framework in the form of the properties.* |

**Time spent (in minutes):** 10

# Actualization

Use the table below to describe each step you followed when changing the code. Include as many details as possible, including why classes/methods were modified, added, removed, renamed, etc.

|  |  |  |
| --- | --- | --- |
| Step # | Description | Rationale |
| 1 | *I copied the code to create a menu button from the RecentFilesProvider class to the RecentDirectoriesProvider class.* | *Since the functionality was identical I figured the button create would be exactly the same.* |
| 2 | *From there I changed the button label to clear-recent-directories.label* | *This is how the text is displayed for the other button and was scoped out during the impact analysis phase.* |
| 3 | *The RecentFilesProvider stored data in a recentVector variable but the RecentDirectoriesProvider stored this information in a HistoryModel class, this class already contained a ClearAllItems method so it was connected as a listener to the button I just made.* | *The recent files button was connected to a similar “clear all” function in it’s mirrored data structure.* |
| 4 | *I then compiled and ran JEdit and opened two files in different directories. From there I clicked on clear directories and it removed the previously opened direcotries.* | *To ensure it works.* |

**Time spent (in minutes):** 20

# Validation

|  |  |  |
| --- | --- | --- |
| Step # | Description | Rationale |
| 1 | *Test case defined: Open a file, then clear the directory*  *Inputs: Opened a file in C:/Users/Lake/Documents*  *Expected output: A cleared recent directories menu* | *This is the regular expected behavior.*  *The test passed.* |
| 2 | *Test case defined: Open 10 files in all different directories*  *Inputs: Opened files from various locations in C-drive*  *Expected output: A cleared recent directories menu.* | *This is exceptional behavior as the recent directories model holds less than 10 directories to display.*  *The test passed.* |

**Time spent (in minutes):** 15

# Timing

Summarize the time spent on each phase.

|  |  |
| --- | --- |
| Phase Name | Time (in minutes) |
| Concept location | 35 |
| Impact Analysis | 10 |
| Actualization | 20 |
| Verification | 15 |
| Total | 80 |

# Reverse engineering

# 

# 

# Conclusions

*This change request was by far the easiest. Since the functionality already existed in another very similar feature it was mostly copy and paste. This also made impact analysis and location very straightforward. Actualization was also aided by the fact that the data structure implemented already had the feature included it just wasn’t being used. Overall this change request was very simple and straight forward.*