Change request log

# Team

Pilot – Diego Navigator - Cade

# Change Request

Change request #4, add the ability to clear directory history

# 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 | *We search though jedit\_gui.props for recent file and recent directories* | *Recent files has similar functionality so maybe we can copy paste code* |
| 2 | *We searched though RecentFilesProvider.java* | *Tring to understand how the clear function works* |
| 3 | *We searched though RecentDirectoriesProvider.java* | *Tring to find where to insert the functionality* |
| 4 | *Search for the phase “clear-recent-files.label” using Ctrl +H in eclipse* | *When the functionality was added no text appeared and “clear-recent-files.label” was a part of the RecentFilesProvider.java so somehow the phase was connected to some data* |

**Time spent (in minutes):** 60 min

# 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 | *We copied code from RecentFilesProvider.java to RecentDirectoriesProvider.java to see if any changes occurred* | *Because both scripts are similar, perhaps we could just copy code.* |
| 2 | *Called HistoryModel.getModel("vfs.browser.path").removeAllElements() to see if it would have any side effects* | *The function seems to do what we want but we have to verify that it doesn’t break anything else.* |

**Time spent (in minutes):** 60 min

# 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 | *Copied code from RecentFilesProvider.java and inserted it into the same place into RecentDirectoriesProvider.java* | *Both scripts are similarly structed and RecentFilesProvider has similar functionality* |
| 2 | *Replaced “clear-recent-files.label” with “clear-recent-directories.label”* | *The string input determines the return string that is used for the UI* |
| 3 | *Replaced “BufferHistory.clear()” with HistoryModel.getModel("vfs.browser.path").removeAllElements()* | *Replace what was originally there with* |
| 4 | *Added “clear-recent-files.label=Clear Recent Directories” to jedit\_en.props* | *jedit\_en.props is what determines what the buttons text say* |

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

# Validation

Use the table below to describe any validation activity (e.g., testing, code inspections, etc.) you performed for this change request. Include the description of each test case, the result (pass/fail) and its rationale.

|  |  |  |
| --- | --- | --- |
| Step # | Description | Rationale |
| 1 | *Open Utilities  Excepted Output: We should see the “No Recent Directories”* | *Make sure the Clear Recent Directories button doesn’t appear by default*  *Test passed* |
| 2 | *Open a file then open Utilities  Excepted Output: We should see the “Clear Recent Directories” button* | *Seeing if the button does appear Test passed* |
| 3 | *Click “Clear Recent Directories” Excepted Output: We should see the “No Recent Directories”* | *Testing the functionality of the code Test passed* |

**Time spent (in minutes):** 2 min

# Timing

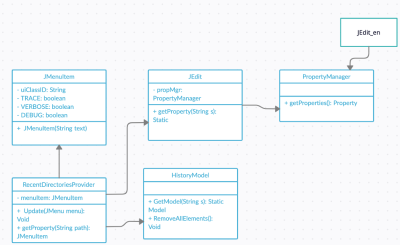
Summarize the time spent on each phase.

|  |  |
| --- | --- |
| Phase Name | Time (in minutes) |
| Concept location | 60 min |
| Impact Analysis | 60 min |
| Actualization | 10 min |
| Verification | 2 min |
| Total | 132 min |

# Reverse engineering

Create a UML sequence diagram (or more if needed) corresponding to the main object interactions affected by your change.

Create a partial UML class diagram of the classes visited while navigating through the code. Include the associations between classes (e.g., inheritance, aggregations, compositions, etc.), as well as the important fields and methods of each class that you learn about. The diagram may have disconnected components. Use the UML tool of your preference. When a significant fact about a class or method is learned, indicate it via annotations on the diagram. **For each change request, start with the diagram produced in the previous change request. For the first, you will start from scratch.**

****

# Conclusions

Perform and analysis of the change requests and the change process. List the major challenges this change request posed.

List all the classes and methods you have changed.

*This change was much easier because we already had knowledge of how UI functionality worked and where it was but also had a similar button to derive code from. Impact analysis was done with Eclipse manually by tracing function calls and the verification was also done by hand.*

*Classes and methods changed:*

* *org/gjt/sp/jedit/menu/RecentDirectoriesProvider.java*
  + *void update(JMenu menu)*

*Other files changed:*

* *bin/org/jedit/localization*
  + *Recent Directories menu*