

Change request log

1. Team

Nick Lawrence and Satya Kalyan

2. Change Request

jEdit Change Request #1

In the File»Recent Files main menu of jEdit, the text box on top of the recent files list allows to highlight recent files names that match with a given string (see Figure 1). The string in the text box should match all the files that contain it anywhere in their name. However, the highlight works only when the string matches the beginning of a file name. You are requested to modify this feature so that the highlight occurs for the cases when the string is contained anywhere in the file name.

3. Concept Location

Step #	Description	Rationale
1	<i>Searched Classes for "RecentFile" and found RecentFilesProvider</i>	
2	<i>In the update method, we located a JTextField called text with an associated KeyAdapter for updating the search term.</i>	<i>To get familiar with some of the features of the system, and identify the screens or graphical elements we had to change.</i>
3	<i>It was determined that here was where the search term was used to locate matching files, and this was where code needed to be changed.</i>	<i>Because we identified a button in the screen called "Add Block Time"</i>

Time spent (in minutes): 15

4. Impact Analysis

Step #	Description	Rationale
1	<i>We found all uses of the RecentFilesProvider.update() method to determine what other classes might need to be changed.</i>	<i>This was the only method that needed to be changed, so anywhere that calls this method may also need some changes.</i>
2	<i>Using a text search for "RecentFilesProvider" revealed that the Recent Files menu item is the only place RecentFilesProvider is used, and will thus only effect that menu item.</i>	<i>We needed to make sure that no other menu items were using the RecentFilesProvider, so that no other areas of the code would be effected by the changes.</i>
3	<i>It was determined the changes would be self-contained to RecentFilesProvider and it's nested subclasses.</i>	

Time spent (in minutes): 20

5. Prefactoring

Step #	Description	Rationale
1	<i>No classes were modified or methods changed in the prefactoring step.</i>	<i>It was determined that the changes would be minimal and that no other areas were needed to be modified.</i>

Time spent (in minutes): 0

6. Actualization

Step #	Description	Rationale
1	<i>A "*" was added to the beginning of the regex string that is passed into the pattern matcher. This way, the string will be found anywhere in the file name, not just at the beginning.</i>	

Time spent (in minutes): 5

7. Postfactoring

Step #	Description	Rationale
1	<i>No classes were modified or methods changed in the postfactoring step.</i>	

Time spent (in minutes): 0

8. Validation

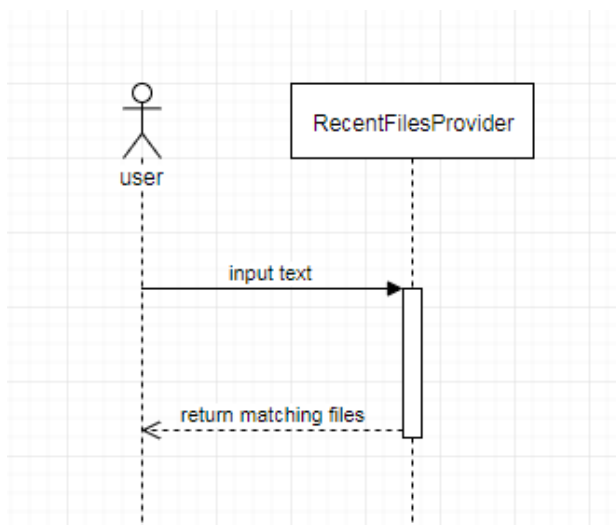
Step #	Description	Rationale
1	<i>All tests provided with jEdit were run as a regression test.</i>	<i>This is the regular expected behavior. The test passed.</i>
2	<i>Manual tests were done by inputting various file names into the text box, and validating the correct output.</i>	<i>This was to verify the changes made had the expected result, which was the case in all manual tests.</i>

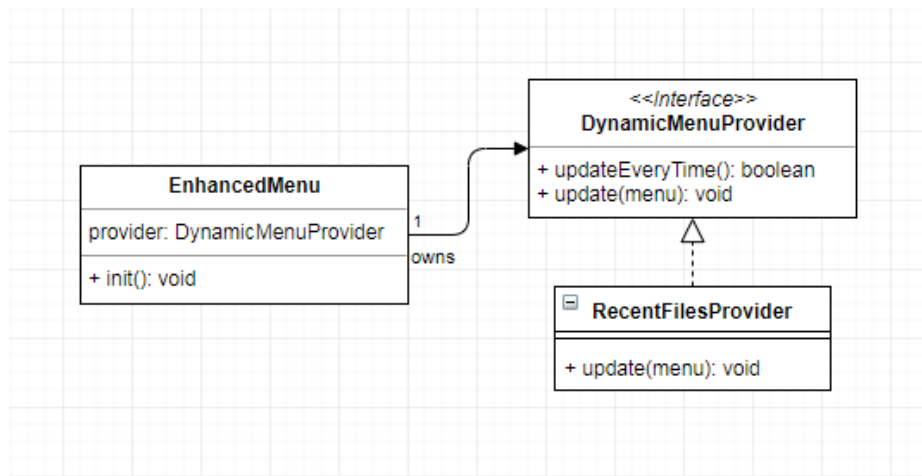
Time spent (in minutes): 40

9. Timing

Phase Name	Time (in minutes)
Concept location	15
Impact Analysis	20
Prefactoring	0
Actualization	5
Postfactoring	0
Verification	40
Total	80

10. Reverse engineering





11. Conclusions

For this change request, all steps were relatively simple due to the easy nature of the change and the layout of the code. The only difficult step was Impact Analysis. Concept location was made easy due to the sensible naming of classes in jEdit. Impact Analysis was a little more difficult as `RecentFilesProvider` is never explicitly referenced in the code of the project, only in a `.props` file that defines the behavior of the menus. From this file, all menu items and their associated providers are loaded, so this text file had to be examined to determine the extent of the changes. The actual modifications to the code were trivial. Testing the changes was also simple due to the limited possible impact and our experience in testing software professionally.

Classes and methods changed:

- `org/gjt/sp/jedit/menu/RecentFilesProvider`
 - `update(JMenu menu)`
 - `new KeyAdapter()` (line 100)
 - `keyReleased(KeyEvent e)`

Final Notes

Nick Lawrence implemented the changes in the code and Satya Kalyan handled the documentation.

In the jEdit repository all the changes are implemented in the master branch.