

Change request log

1 Team

BugHunters

Members:

Prashant K Thakur

Sadaf Ghaffari

Bibek Raj Shrestha

2 Change Request

Jedit Change Request-2 : 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 2). 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

The following is the detail on a concept location process for the change request "Highlight recent files name with matching string":

Step #	Description	Rationale
1	<i>We ran the system</i>	
2	<i>We tried to find the word "Recent*File" in IDE to discover files.</i>	<i>Because we identified a button in the screen called "Recent Files".</i>
3	<i>From 14 results, there were only 3 distinct classes. So we started from the java file "GeneralOptionPanel" which used it a lot.</i>	
4	<i>We inspected the class GeneralOptionPanel but it did not seem to have a place where the regex match was performed. So we checked other files "RecentFileProvider" as it seemed more closer than "BufferHistory".</i>	<i>We noticed that there was a method "update" inside RecentFileProvider which listens for the input and matches the regex pattern.</i>
5	<i>We marked the class RecentFileProvider as "located".</i>	<i>We confirmed this class had to be modified.</i>

Time spent (in minutes): 30

4 Impact Analysis

The following table summarizes the impact analysis we did for this change request.

Step #	Description	Rationale
1	We found the usage of the method update from the RecentFileProvider and found out that it is only been used in one class EnhancedMenu inside init method.	To track the classes that could be impacted by the change.
2	We inspected the class RecentFileProvider and marked it as "to change".	We realized this class has to be changed because it handles the implementation of actual regex match.
3	The class EnhancedMenu was discarded from the list of change.	Because the classes uses the menu event handling and the only change that we did was a pattern (string addition) which does not have any impact on this class.

Time spent (in minutes): 15

5 Actualization

The table below summarizes the actualization we performed to implement this change.

Step #	Description	Rationale
1	We made change to the method "update" inside RecentFileProvider to update the regex to be used for matching the list of files.	We realized that the regex being used was 'regex + "*" ' which would only match the word (typed) and anything after that. So we changed the regex so that it matches anything from the start with the typed characters and followed by anything. So the final regex was ' "*" + regex + "*" '.
2	We compiled the program and try to verify it with different inputs manually in the field.	To make sure everything works.

Time spent (in minutes): 5

6 Validation

The table below summarizes the validation phase of the change request.

Step #	Description	Rationale
1	Test case defined: Inputs: titled Expected output: Untitled-1, Untitled-2,Untitled1	This is the regular expected behavior. As the files matching the pattern are displayed while the other are masked. The test passed.
2	Test case defined: Inputs: Un Expected output: Untitled-1, Untitled-2, Untitled1	The test passed.

3	<i>Test Case defined: Inputs: Present Expected output: All masked</i>	<i>The test passed.</i>
4	<i>Test Case defined: Inputs: 1 Expected output: Untitled-1, Untitled1</i>	<i>The test passed.</i>

Time spent (in minutes): 10

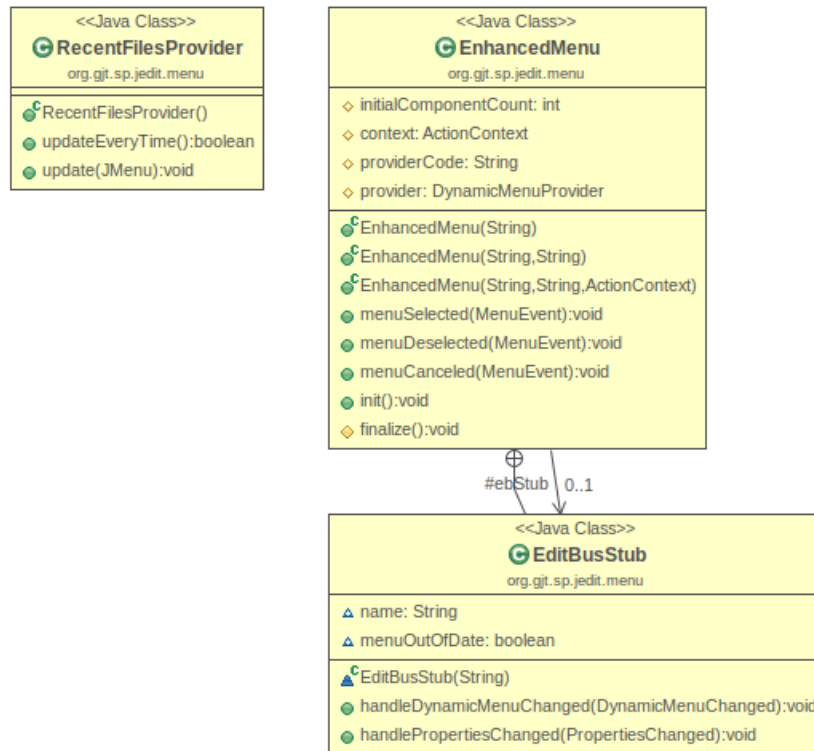
7 Timing

Summarize the time spent on each phase.

Phase Name	Time (in minutes)
Concept location	30
Impact Analysis	15
Prefactoring	-
Actualization	5
Postfactoring	-
Verification	10
Total	60

8 Reverse engineering

UML diagram of the class that were visited.



9 Conclusions

The change was easier for this feature as the footprint for this change was small. The files needed to be modified was few. And they do not show any dependencies between each other in the UML diagram. However, we went through the code of `EnhancedMenu` to verify if the change has any impact. Based on the analysis and the nature of change we assured that the change does not have any influence.

Classes and methods changed:

- `org/git/sp/jedit/menu/RecentFilesProvider.java`
 - `void update(JMenu menu)`

GitHub Repository: <https://github.com/prashantkthakur/cs515-s19-a2-BugHunters-jedit550>

Some screenshot for this implementation:

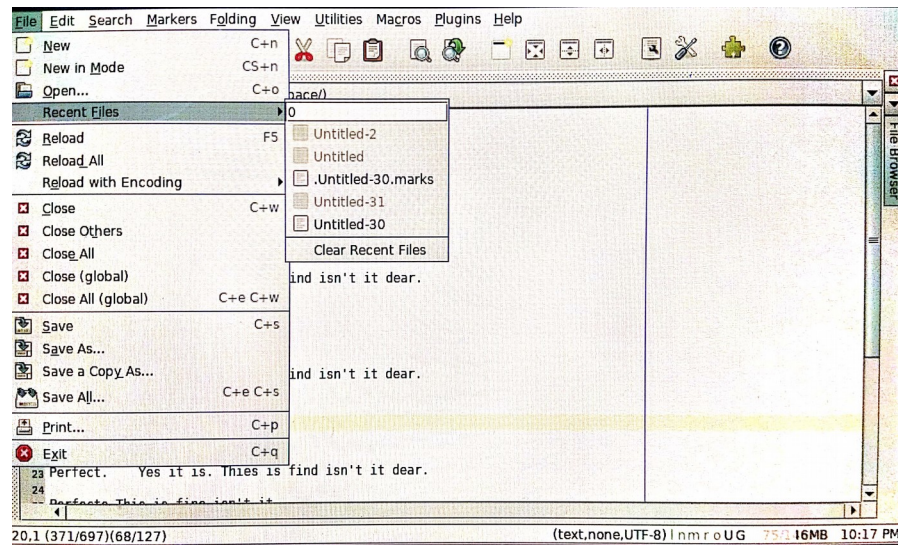


Figure: GUI displaying search in Recent Files

Note: The keyboard are captured while working with JEdit so we have to improvise and take the screenshot from external device.