Change request log

Team

gitrams

Change Request

Change Request ID: #2 (JEdit)

Description of the change request: "Currently, JEdit displays the horizontal and vertical scroll bars whenever the content of the opened document exceeds the size of the editor. Implement an option in the View menu that allows to hide the scrollbars."

Concept Location

- IDE Features used: **InstaSearch** and basic Search in Eclipse
- Queries used when searching: scrollbar, scroll, scrolling
- Interactions with the system:
- Classes visited: org.git.sp.jedit.textarea.TextArea
- The first class found to be changed (this is when concept location ends): org.git.sp.jedit.textarea.TextArea

Step #	**Description**	**Rationale**
1	*We compiled and ran JEdit-5-4-0*	
2	*We downloaded and opened an ebook in text format from Project Gutenberg.*	*To have a file with enough content, so as to show up the scroll bars .*
3	*We searched for "scrolling" using the basic search and InstaSearch plugin installed in Eclipse/*	
4	*We found many relevant results: classes and some non java files too.*	
5	*We narrowed down to the org.git.sp.jedit.textarea package and inspected the TextArea class .*	*The name seemed quite related to the white box in which the text of the loaded file is displayed, and this where the scroll bars show up. So it seemed the scroll bars must be generated or controlled in this class.*
6	*We also included non java files like actions.xml, textarea.actions.xml, jedit_en.props and jedit_gui.props*	*As there were many actions and controls related to scroll bar operation in these files.*
7		
8		

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	*We inspected all the classes which were related to the classes we found in the Concept Location phase.*	*To track the classes that could be impacted by the change.*
2	*We did not find any other class which would get affected by the changes we are going to make.*	
3		
4		

Time spent (in minutes): 15

[redacted] 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	*We created two new menu items (Disable Scroll Bars & Enable Scroll Bars) and associated them with actions.*	*This is the change requested.*
2	*We created methods disableScrollbar and enableScrollbar in the class org.git.sp.jedit.textarea.TextArea*	*To use as actions for the newly created GUI menu items*
3	*We created two actions in actions.xml and textarea.actions.xml to call the class methods.*	

Time spent (in minutes): 40

[redacted] Postfactoring (optional)

For this change request we did not require to postfactor any part of the code base except for adding a few comments. This is because the Move Top and Move Bottom functionalities were similar to the Move Up and Down buttons' functionalities that were already present in the code base.

Step #	Description	Rationale
1	*Added comments regarding the newly implemented scroll bar functionality.*	*We wanted to keep a record/documentation in the form of comments in the source code.*

Time spent (in minutes): 10

[redacted] 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	*We used manual testing and used an ebook (Alice in Wonderland) as a test.*	*The file was chosen as it has enough text to enable the scroll bars.*
2	*We opened the file in JEdit and as expected the scrollbars showed up. We then clicked our newly implemented Disable Scroll Bars menu item and the scrollbars disappeared. And then when we clicked on Enable Scroll Bars, the scrollbars showed up.*	*Both the disable and enable scrollbars features worked fine. The test passed.*
3		
4	…	

Time spent (in minutes): 10

[redacted]Timing

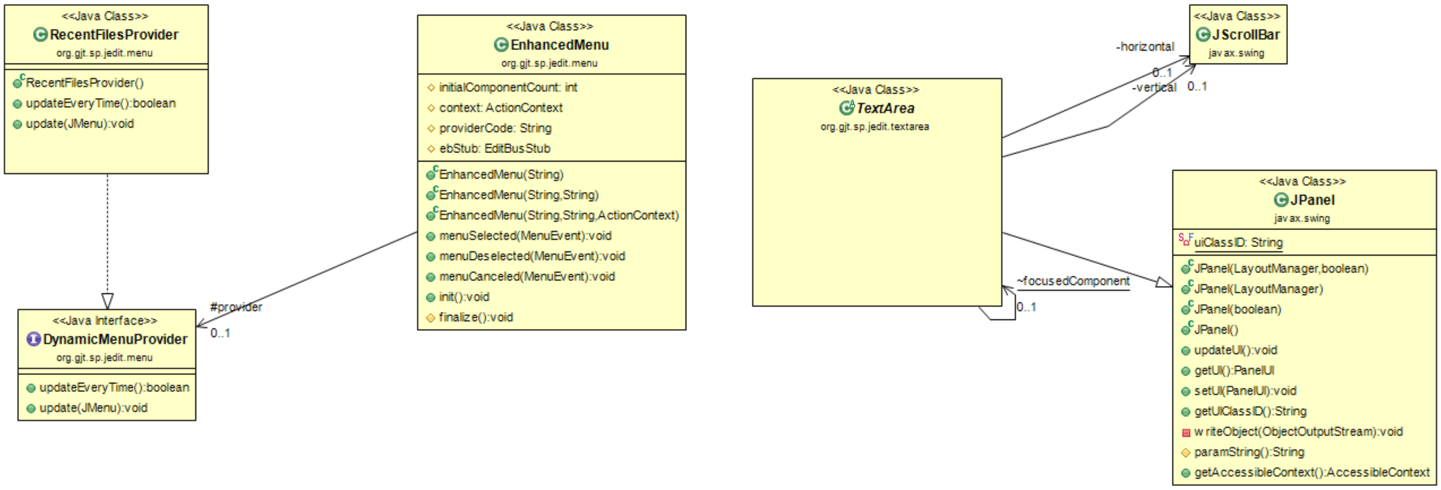
Summarize the time spent on each phase.

Phase Name	Time (in minutes)
Concept location	35
Impact Analysis	15
Prefactoring	0
Actualization	40
Postfactoring	10
Verification	10
Total	110

[redacted]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.**



[redacted]Conclusions

For this change, concept location was slightly hard because the keywords were found in java classes as well as in non-java files, and I was unsure if non-java files should be changed. After having a deeper look in all the searched files, I got to know how the functionality is implemented in the java and non-java (xml, props) files. It was then easy to implement the new GUI buttons, actions and class methods. Also for adding the labels, it took me a while to figure out that the change should be implemented in ALL the language files of the system (contained in org/jedit/localization/jedit_XX.props, where XX is the language). I made the change only in the English language version of the file.

Classes and methods changed:

org/gjt/sp/jedit/textarea/TextArea.java
    void disableScrollbar()
    void enableScrollbar()
org/gjt/sp/jedit/textarea/textarea.actions.xml
org/gjt/sp/jedit/actions.xml
org/gjt/sp/jedit/jedit_gui.props
org/jedit/localization/jedit_en.props