

Change request log 1

Concept Location

Step #	Description	Rationale
1	We ran the system	
2	We interacted with the system: We located the status bar in the GUI where the requested change had to be done	To understand correctly where the change had to be done
3	We understood the code structure and looked at/listed the classes in the source code	To get an idea of which classes we may have to touch in order to implement the change
4	We used IntelliJ IDE's search feature to locate the StatusBar class in the source code	After inspecting GUI and understanding the code structure we figured the changes has to be done in the StatusBar class
5	We located the file StatusBar.java using the IDE's search file feature	To look into and understand the code in the StatusBar class
6	We looked carefully at the code in the StatusBar.java and figured that the method updateCaretStatus() needs some enhancements in order to incorporate the changes	The method updateCaretStatus() is responsible for updating the status bar with information about the caret position, line number
7	We looked at the method declaration and definition and understood that the method does not return any value and does not accept any parameters	To make sure if we need to look after inputs and output of the method
8	Located the string "buf" which is actually being displayed in the status bar in GUI	To get a final idea of the specific thing to change in source code

Time spent (in minutes): 30

We inspected the following classes and methods :

- org/gjt/sp/jedit/gui/StatusBar.java
 - public void updateCaretStatus()

Impact Analysis

Step #	Description	Rationale
1	We first made a list of all the places the updateCaretStatus() is being used	To check in what all scenarios this method is being called
2	We found that this method is being called from two classes EditPane and View	
3	We inspected these two classes specifically where the method is being called before making any change	To get an idea if we have to make changes in those classes as well after the change in updateCaretStatus() method
4	We understood no changes will be required in the classes where this method is accessed	

Time spent (in minutes): 30

We inspected following classes and methods :

- org/gjt/sp/jedit/gui/StatusBar.java
 - public void updateCaretStatus()
- org/gjt/sp/jedit/EditPane.java
 - handleBufferUpdate(BufferUpdate msg)
- org/gjt/sp/jedit/View.java

Actualization

Step#	Description	Rationale
1	We first understood how the displayed string “buf” in the status bar is being updated	To get an idea as to how to update it
2	We first calculated the number of words from the start of file to the caret position	To display word offset as per the change request
3	We then calculated the total number of words present in the file, that is, the number of words from the start of file to the end	To display total word count as per the change request
4	We looked and understood that the string “buf” is being appended as per the different parameters of the if-else statements	To understand how and where to include word offset and total word count in the string
5	We located the suitable if condition and modified the existing code to append the existing string with the word offset and the total number of words	

Time spent (in minutes): 60

We inspected following classes and methods :

- org/gjt/sp/jedit/gui/StatusBar.java
 - public void updateCaretStatus()

1. Validation

Step#	Description	Rationale
1	<p>We performed manual testing and the steps followed are mentioned below:</p> <ol style="list-style-type: none">1. Opened jEdit and created a new file2. We added some content (paragraphs in the file)3. Verified that the status bar shows the current line number, column position, character offset, and total number of characters4. Moved the caret to a new location in the file and verified that the status bar updates to show the new line number, column position, character offset, and total number of charactersVerified that the existing functionality is not impacted/broken after our change5. Verified that the status bar also shows the new word offset and the total number of words in the file6. Created a new file with a large number of words and verify that the status bar correctly displays the word offset and the total number of words7. Saved the file and verified that the status bar still displays the correct information8. Verified that there are no adverse effects on other features or functionalities of jEdit	The test passed
2	We repeated the above steps for different file types and verified that the status bar works as expected for each type	The test passed

Time spent (in minutes): 20

2. Summary of the change request

Phase	Time (minutes)	No. of classes inspected	No. of classes changed	No. of methods inspected	No. of methods changes
Concept location	30	1	1	3	1
Impact Analysis	30	3	2	3	1
Actualization	60	1	1	1	1
Verification	20	1	1	1	1
Total	140	3	1	1	1

3. Conclusion

We spent around 30 minutes on the concept location where we inspected the GUI and the code structure. We spent another 30 minutes on impact analysis, listing all the places where the `updateCaretStatus()` method was being used and finding that no changes were required in the classes where this method was accessed. During the actualization phase, we spent 60 minutes modifying the `updateCaretStatus()` method in the `StatusBar` class to display the word offset and total number of words in the file. We verified their changes for different file types, and the tests passed. Finally, we spent 20 minutes performing manual testing to verify that the existing functionality was not impacted and the status bar worked as expected.

Overall, we inspected three classes, changed one class and one method, and spent a total of 140 minutes on the change request. The tests passed, indicating that the changes were successfully implemented and did not have any adverse effects on other features or functionalities of jEdit.