

Change request #3 log

1. Concept Location

Step #	Description	Rationale
1	We loaded JUnit thru IntelliJ	
2	We examined the application and saw the scrollbars are visible.	This was important so we could get an idea of where the changes should be made.
3	We clicked on the View option in the menu and saw a toggle option for the scroll bar did not exist.	According to the instructions, the View option should have an option to toggle the scroll bar so we knew this was to be implemented.
4	In the code, we went to examine TextArea.java and found a lot of code revolving around the scrolling features.	This class made the most sense since we had the scrolling feature in the text area.
5	After this, we went to find the location of where to implement the changes for the View. After searching the repo, we found the actions.xml file, where we could create an action for the user to toggle the scroll bar.	We searched for instances where the other toggle options were being set. Actions.xml made the most sense since the logic was being called there.

Time spent (in minutes): 45

Classes and methods inspected:

- org/gjt/sp/jedit/textarea/TextArea.java
- org/gjt/sp/jedit/actions.xml

2. Impact Analysis

Step #	Description	Rationale
1	After discovering actions.xml, we saw a bunch of other toggle actions. Each of these had unique variables such as "view.gutter.enabled". When using control-shift-F, we searched the entire repo and found that these were being initialized in jedit.props	We needed to initialize a variable for the scrolling feature to have it appear and function.
2	In jedit_gui.props, we found that the overall "routing" was being done here. This is where the organization of each of the menu features and GUI were being setup.	We determined this was the location to create a toggle option for the scrollbar.
3	In jedit_en.props, we found this was where we could create the label to name the toggle option.	
4	In the TextArea.java class, we had functions for the scrollListeners. These could add or remove listeners for scrolling. There were also sections in the beginning which included initialization for the scrollbars.	Since they were being created in this class, we determined we could make or call a function to add or remove the scrollbars.

Time spent (in minutes): 120

Classes and methods inspected:

- org/gjt/sp/jedit/textarea/TextArea.java

- org/gjt/sp/jedit/actions.xml
- org/gjt/sp/jedit/jedit_gui.props
- org/jedit/localization/jedit_en.props
- org/gjt/sp/jedit/jedit.props

3. Actualization

Step #	Description	Rationale
1	First, in jedit_gui.props, we added a new line in 382 for "toggle-scroll \". This would add a new option for the user.	
2	Next, we went in jedit_en.props to add the proper label in line 276, "toggle-scroll.label=Toggle \$Scroll"	Now, when the user clicks the View option in the menu, they will see an option to Toggle Scroll
3	In the jedit.props, we added a Boolean variable to view the scroll bars in line 387, "view.gutter.scrollBar=true"	
4	Now that we had set everything else up, we had to make the changes in actions.xml. We added a new action for toggling the scrollbar. We used similar logic for what was used in the other toggle options.	
5	In TextArea.java, we created a new Boolean function, toggleScrollBar(). This used logic similar to what was used to initially create the scrollbars when loading the text area. We set it in a conditional to determine whether to remove the scrollbars.	This was important since we would use this function in our actions.xml section to make the function call whenever the user clicked on the toggle scroll option.

Time spent (in minutes): 120

Classes and methods inspected:

- org/gjt/sp/jedit/textarea/TextArea.java
- org/gjt/sp/jedit/actions.xml
- org/gjt/sp/jedit/jedit_gui.props
- org/jedit/localization/jedit_en.props
- org/gjt/sp/jedit/jedit.props

Classes and methods changed:

- org/gjt/sp/jedit/textarea/TextArea.java
- org/gjt/sp/jedit/actions.xml
- org/gjt/sp/jedit/jedit_gui.props
- org/jedit/localization/jedit_en.props
- org/gjt/sp/jedit/jedit.props

4. Validation

Step #	Description	Rationale
1	We ran the program and clicked the toggle option to determine if it was working correctly or not	This is the regular expected behavior. The test passed.
2	We also ran the program in debugger mode to determine if we were getting the correct Boolean values.	This was expected and the test passed.

Time spent (in minutes): 20

5. 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	45	2	0	3	0
Impact Analysis	120	5	0	4	0
Actualization	120	5	5	5	5
Verification	20	1	0	0	0
Total	295	13	5	12	5

6. Conclusions

This change request required a fair bit of time to determine the best strategy to solve the problem. We had to examine multiple files to track the solution down. Once we figured it out, the issue was to create the logic for the conditional statement to toggle the scrollbars. We had issues where the scroll bar would initially be removed when the user clicked the view button in the menu and had to make modifications to prevent that. Overall, this was a good learning experience when it came to solving challenging coding issues.