jEdit Milestone #4

Name	Student ID
Hamed Kalantari	9411569
Kelvin Iu	4538501
Oualid El Halimi	9385886
Muhammad Farhan Malik	6577792
Youssef Bennis	5570913

Due: Apr 7th, 11 marks

In jEdit milestone #3 [1], we have indicated we will implement the following refactorings in milestone #4:

- God class (see section 2.1.1 of [1])
- InstanceOf (see Section 2.1.2 of [1])
- Long Method (see Section 2.1.3 of [1])

For completeness sake and to make the most from our learning experiences of the project, we have also implemented

- Implicit dependency of Static methods (see section 2.1.4 of [1])

The following section goes into details of what changes were made in each patch in the patchset and the rationale behind their code changes. For the actual code changes 'diff', please refer to the patchset file generated from Git (i.e. Change X/6 in this document maps to [PATCH X/6] in the patchset file, where X >= 1).

Change 0/6: Refactor of *SearchAndReplace* class to improve its maintainability and make it more cohesive.

Change #1 and #2 addressed the issues of implicit dependency of a few static methods exposed from *SearchAndReplace* class. Change #3 and #4 extracted the replace functionality into a separate *Replace* class from the *SearchAndReplace* 'God' class. Change #5 replaced type code by a strategy pattern for handling replacing text in selection in method 'replaceInSelection' of the extracted *Replace* class. Change #6 simplifies the long, complex method 'regexpReplace' of the extracted *Replace* class by extracting methods.

Change 1/6: Dependency of static methods refactoring Create the new hypersearch method that will replace the existing one. This method guarantees assignation of the variables.

By exposing a new public method 'performHyperSearch' to the rest of the system, it guarantees the search string and search file set is initialized before hyper-search is performed.

Change 2/6: Dependency of static methods refactoring Replace the calls from the old method to the new methods that ensures variables are set. Old method that has implicit dependencies has been made private, so no class can call it from outside, only the new method is calling it.

This change also updates the rest of the system to leverage the newly exposed method 'performHyperSearch' from *SearchAndReplace* class to perform hyper-search. The static methods

'setSearchString' and 'setSearchFileSet' is also made private to avoid rest of the system accessing them.

Change 3/6: There is an instance of God Class (Large Class) in "SearchAndReplace"class. The class is large and has several components (methods). This class has two responsibilities: Searching and Replacing content. Hence, there is low cohesion between those distinct functionalities. Therefore, there is no abstraction for the role of the class since it contains different functionalities. There are methods in the "SearchAndReplace" class, that can be extracted to make the class smaller, manageable and also to increase cohesion . In order to improve the cohesion of the methods in "SearchAndReplace" class and to narrow the class objectives to perform a particular task (i.e. Search or Replace), we will extract all related replace functionalities from 'SearchAndReplace" class to a separate class "Replace", which is specialized for replacing content using extract class refactoring. These refactorings are exactly performed according to published jEdit Milestone #3 [1], section 2.1.1 God Class.

A new class Replace is created and the following private static methods were extracted from the SearchAndReplace class: initReplace, replaceInSelection, _replace, replaceOne, regexpBeanShellReplace, regexpReplace, literalBeanShellReplace, getColumnOnOtherLine. The following public static methods were also extracted: replace, replaceAll and they were being delegated from the corresponding methods in the original *SearchAndReplace* class.

Change 4/6: Rename the refactored "SearchAndReplace" class to "Search", in order for the name of the class to match its concrete functionality. These refactorings are exactly performed according to published jEdit Milestone #3 [1], section 2.1.1 God Class.

The change also updates the rest of the system to reference the new class name 'Search'.

Change 5/6: While looking into the God Class code smell (section 2.1.1 of jEdit Milestone #3) in the Replace class, another code smell was identified. This code smell fell under the criteria of "Instance Of " code smell. The method in which "Instance Of" code smell, current changes and refactroing in Replace class and Selection class have removed the InstanceOf code smell. Detailed instructions of steps of the refactorings and the changes made are available in the published jEdit Milestone #3 [1], section 2.1.2 Type Checking (InstanceOf).

The change introduced a new method 'replaceInSelection' in the *Selection* hierarchy of classes. The base implementation will throw a RuntimeException for 'Unsupported' class that correspond to the original behavior of 'replaceInSelection' method of the *SearchAndReplace* class. Implementation of 'replaceInSelection' of the other *Selection* derived classes: *Range* and *Rect*, implement their corresponding logic extracted from 'replaceInSelection' method of the *SearchAndReplace* class.

Change 6/6: The method regexpReplace in Replace class is long and difficult to comprehend due to its

complex flow of execution caused by the loop, switch and if statements. The method can be shorter by extracting less cohesive parts using extract method refactoring. for complete details regarding to this refactoring please refer to the publish JEdit Milestone #3 [1], section 2.1.3 Long Method.

New private static method 'processCharacterInReplaceString' is extracted from method 'regexpReplace' of the extracted *Replace* class to process individual characters in the outer-most for loop. Another private static methods 'processDollarCharacterInReplaceString' and 'processBackSlashCharacterInReplaceString' were extracted to handle the two sub-cases of the characters in the replace string: '$' and '\' characters respectively.

References:

[1] H. Kalantari, K. Lu, O. El Halimi, M. F. Malik, Y. Bennis, "jEdit Milestone #3", http://goo.gl/Q7sFMq, March 2014