

Review report of a final thesis

Czech Technical University in Prague

Faculty of Information Technology

Student: Jakub Nesveda
Reviewer: Ing. Claus Gittinger
Thesis title: API pro generování kódu a refaktoring
Branch of the study: Software Engineering (Bachelor)

Date: 22. 5. 2015

<i>Evaluation criterion:</i>	<i>The evaluation scale: 1 to 5.</i>
1. Difficulty and other comments on the assignment	1 = extremely challenging assignment, 2 = rather difficult assignment, 3 = assignment of average difficulty, 4 = easier, but still sufficient assignment, 5 = insufficient assignment
<i>Criteria description:</i> Characterize this final thesis in detail and its relationships to previous or current projects. Comment what is difficult about this thesis (in case of a more difficult thesis, you may overlook some shortcomings that you would not in case of an easy assignment, and on the contrary, with an easy assignment those shortcomings should be evaluated more strictly.)	
<i>Comments:</i> The work involves examination and deep understanding of relatively complex frameworks (existing browser and refactory packages) and their relationships. Especially the trouble of integration into the existing tool frameworks make this a much more challenging assignment than many other projects, which „live in their own world“. The assignment involved a lot of „pre-work“ and „pre-experience“ in order to succeed.	
<i>Evaluation criterion:</i>	<i>The evaluation scale: 1 to 4.</i>
2. Fulfilment of the assignment	1 = assignment fulfilled, 2 = assignment fulfilled with minor objections, 3 = assignment fulfilled with major objections, 4 = assignment not fulfilled
<i>Criteria description:</i> Assess whether the thesis meets the assignment statement. In Comments indicate parts of the assignment that have not been fulfilled, completely or partially, or extensions of the thesis beyond the original assignment. If the assignment was not completely fulfilled, try to assess the importance, impact, and possibly also the reason of the insufficiencies.	
<i>Comments:</i> All aspects of the challenge have been addressed and covered in appropriate detail. The author provide a working solution which improves the usability of the software development environment. The availability of a flexible mechanism for custom refactoring and code generators is also a new, innovative and advanced feature in IDEs. Items listed in the future work section at the end are „nice to have“ issues, which do not affect the usability or completeness of the solution. The presence and amount of Unit-Tests and the use of code coverage measurement tools indicate good software engineering practice and discipline of the author.	
<i>Evaluation criterion:</i>	<i>The evaluation scale: 1 to 4.</i>
3. Size of the main written part	1 = meets the criteria, 2 = meets the criteria with minor objections, 3 = meets the criteria with major objections, 4 = does not meet the criteria
<i>Criteria description:</i> Compare the size of the written part with the expected size (without appendices), see the Dean's Directive No. 9/2011, Article 3. To evaluate the thesis it is also important that all parts of the written part are rich on information and necessary for a final thesis. The text should not contain unnecessary parts.	
<i>Comments:</i> All relevant aspects have been covered in enough detail. The encountered problems and the solutions are described in detail and can be followed. The thesis can also be used as a good starting point for similar projects in other languages and/or environments.	
<i>Evaluation criterion:</i>	<i>The evaluation scale: 0 to 100 points (grade A to F).</i>
4. Factual and logical level of the thesis	100 (A)
<i>Criteria description:</i> Assess whether the thesis is correct as to the facts or if there are factual errors and inaccuracies. Evaluate further the logical structure of the thesis, links among the chapters, and the comprehensibility of the text for a reader.	
<i>Comments:</i> I could not find any factual mistakes or omissions. The document is well structured.	
<i>Evaluation criterion:</i>	<i>The evaluation scale: 0 to 100 points (grade A to F).</i>
5. Formal level of the thesis	90 (A)
<i>Criteria description:</i> Assess the correctness of formalisms used in the thesis, the typographical and linguistic aspects, see Dean's Directive No. 9/2011, Article 3.	

Comments:

There are minor grammatical errors (mostly missing articles, which are a consequence of the author's Czech background). However, these do not affect the general readability of the text and are to be considered as „minor typing mistakes“.

Evaluation criterion:

The evaluation scale: 0 to 100 points (grade A to F).

6. Bibliography

90 (A)

Criteria description:

Evaluate the student's activity in acquisition and use of studying materials in his thesis. Characterize the choice of the sources. Discuss whether the student used all relevant sources, or whether he tried to solve problems that were already solved. Verify that all elements taken from other sources are properly differentiated from his own results and contributions. Comment if there was a possible violation of the citation ethics and if the bibliographical references are complete and in compliance with citation standards.

Comments:

The text references the relevant documents on refactoring. However I missed a reference to „A Refactoring Tool for Smalltalk“ by Roberts, Brant and Johnson, which I consider as a milestone in the history of refactoring and development tool technology.

Evaluation criterion:

The evaluation scale: 0 to 100 points (grade A to F).

7. Evaluation of results, publication outputs and awards

100 (A)

Criteria description:

Comment on the achieved level of major results of the thesis and indicate whether the main results of the thesis extend published state-of-the-art results and/or bring completely new findings. Assess the quality and functionality of hardware or software solutions. Alternatively, evaluate whether the software or source code that was not created by the student himself was used in accordance with the license terms and copyright. Comment on possible publication output or awards related to the thesis.

Comments:

The thesis includes working software, which fulfills the stated requirements.

Evaluation criterion:

No evaluation scale.

8. Applicability of the results

Criteria description:

Indicate the potential of using the results of the thesis in practice.

Comments:

The thesis describes and implements customizable refactoring and code generators, which - in this form - are new in the Smalltalk/X environment, and also many other software development environments (both Smalltalk and non-Smalltalk). This does enhance the usability and thus programmer's productivity and will certainly be used in practice.

Evaluation criterion:

No evaluation scale.

9. Questions for the defence

Criteria description:

Formulate any question(s) that the student should answer to the committee during the defence (use a bullet list).

Questions:

* Can you think of an intelligent agent, which monitors the user's ongoing work and tries to extract templates for additional refactorings/code generation automatically? Could the template system be enhanced to create new templates automatically from a set of previous concrete changes? (this is likely to be a question to be really answered in a followup master thesis, as this certainly is a real scientific challenge... so the candidate should not be stressed too much with this question)

* In which way does the Smalltalk way of „living inside the system“ affect the programmer's productivity. Would it be much harder to implement those features in a traditional file-based IDE (such as „Eclipse“)?

* A lot of you work involved understanding the existing code base. Which aspects - in your opinion - have the biggest influence on the readability and especially on the comprehensibility of an existing framework. Discuss test-cases, code examples, charts or diagrams, comments etc. Could your framework be used/adapted to help improving software quality in this area?

Evaluation criterion:

The evaluation scale: 0 to 100 points (grade A to F).

10. The overall evaluation

100 (A)

Criteria description:

Summarize the parts of the thesis that had major impact on your evaluation. The overall evaluation **does not** have to be the arithmetic mean or any other formula with the values from the previous evaluation criteria 1 to 9.

Comments:

The work required a lot of understanding of the general problems with refactoring, change management, undo handling and user interface aspects and of the concrete existing refactoring and browser frameworks. The existing code base was created and maintained by different people over a long time period and as such shows a non homogenous coding style. In addition, some of this existing codebase is poorly documented - especially the refactoring framework. All of this makes this a more challenging assignment, when compared to the average bachelor's task and was therefore also biasing the overall rating. Making refactoring and code generation a customizable feature is not common among software development frameworks and is can thus be seen as an innovative feature of modern IDEs.

The result includes a very usable software component, which fulfills the assignment's requirements and improves the productivity of software developers.

Signature of the reviewer: