Pontifícia Universidade Católica
do Rio de Janeiro

Roberta Lopes Arcoverde

Prioritization of Code Anomalies Based on Architecture Sensitiveness

DISSERTAÇÃO DE MESTRADO

Dissertation presented to the Programa de Pós-Graduação em Informática of the Departamento de Informática, PUC-Rio as partial fulfillment of the requirements for the degree of Mestre em Informática.

Advisor: Alessandro Fabricio Garcia

Rio de Janeiro

September 2012

PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO

Roberta Lopes Arcoverde

Prioritization of Code Anomalies Based on Architecture Sensitiveness

Dissertation presented to the Programa de Pós-Graduação em Informática, of the Departamento de Informática do Centro Técnico Científico da PUC-Rio, as partial fulfillment of the requirements for the degree of Mestre

Prof. Alessandro Fabricio Garcia

Advisor

Departamento de Informática – PUC-Rio

Prof. Carlos José Pereira de Lucena

Departamento de Informática – PUC-Rio

Profa. Simone Diniz Junqueira Barbosa

Departamento de Informática – PUC-Rio

Prof. José Eugenio Leal

Coordinator of the Centro Técnico Científico da PUC-Rio

Rio de Janeiro, September 11th, 2012

Roberta Lopes Arcoverde

Graduated in Computer Science from Universidade Federal de Pernambuco (2008, Brazil, Pernambuco). She is a member of the OPUS research group at the Laboratório de Engenharia de Software of PUC-Rio (LES / PUC-Rio). Her main studies are related to Software Engineering, more specifically to Software Architecture and Design.

Bibliographic data

Arcoverde, Roberta Lopes

Prioritization of code anomalies based on architecture sensitiveness / Roberta Lopes Arcoverde ; advisor: Alessandro Fabricio Garcia – 2012.

v., 89 f: II. ; 29,7 cm

Dissertação (Mestrado em Informática)–Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2012.

Inclui referências bibliográficas

1. Informática – Teses. 2. Engenharia de software. 3. Anomalias de Código. 4. Refatoração. 5. Degradação Arquitetural. I. Garcia, Alessandro Fabricio. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

Acknowledgments

This work was the result of many sleepless nights and hard-working days. As a part-time student, I was often overwhelmed with deadlines from projects milestones, calls for papers, classes and exams. Sometimes it felt like it would be impossible to get good grades, carry out some quality research and, at the same time, meet my employer's expectations. And it would indeed have been impossible, if it wasn't for the company of a significant group of wonderful people who walked this path along with me.

Above all, I would like to thank Turah, for being there with me when I needed her to be, and for understanding when I was absent, either attending to conferences or spending countless days at the labs. Her limitless support and care made this possible (and even enjoyable).

This dissertation would not have been possible without the help, patience and wisdom of my kind advisor, Prof. Alessandro Garcia. His deep knowledge and enthusiasm inspired me to become a better person, student and researcher.

I would also like to express my deepest gratitude to the members of the OPUS Research Group, for their guidance, support and friendship. More specifically, to my dear friend Isela Macia, who relentlessly revised my studies, for her generosity, for inspiring this research, and for giving the best advice I could have ever asked. I am also very grateful to my friend Chico, for helping me getting through the "finish" line when I had no more energy to move forward.

I am most grateful to my peers at Radix, for keeping up the good work. I know my absence let you guys in trouble a couple of times, and I deeply appreciate your kindness and professionalism. I would like to specifically acknowledge my manager Paulo Armando for supporting my Master's studies, regardless of how many work days I would miss.

Finally, I would like to thank my parents Waldemar and Marilda, my brother Guilherme, my sister Renata and my aunt Zuca for their love.

Abstract

Arcoverde, Roberta Lopes; Garcia, Alessandro Fabricio (Advisor)**. Prioritization of Code Anomalies Based on Architecture Sensitiveness.** Rio de Janeiro, 2012. 89p. MSc. Dissertation – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

The progressive manifestation of code anomalies in a software system is a key symptom of its architecture quality decline. When those anomalies are not detected and removed early, the maintainability of software projects can be compromised irreversibly, and, eventually, a complete redesign is inevitable. Despite the existence of many techniques and tools for code anomaly detection, identifying anomalies that are more likely to cause architecture problems remains a challenging task. In fact, studies performed in the context of this dissertation show that even when there is tool support for detecting code anomalies, developers seem to invest more time refactoring those that are not related to architectural problems. Moreover, we also found that developers frequently prioritize refactoring of code elements that do not contribute to a better adherence to the intended software architecture. In this context, this dissertation proposes a prioritization approach for identifying which anomalies in a system implementation are more harmful to the architecture. The proposed approach is composed of heuristic strategies that exploit several software project factors to identify and rank code anomalies by their architecture relevance. These factors range from the change characteristics to the potential architecture roles of software modules. Furthermore, we implemented tool support for applying our prioritization approach in Java projects. We also evaluated the prioritization approach on 4 software projects from different application domains. Our evaluation revealed that software maintainers could benefit from the recommended rankings for identifying which code anomalies are harming architecture the most, helping them investing their refactoring efforts into solving the architecturally relevant problems.

Keywords

Code anomalies; refactoring; architecture degradation.

Resumo

Arcoverde, Roberta Lopes; Garcia, Alessandro Fabricio. (Orientador). **Priorização de Anomalias de Código Sensível a Arquitetura.** Rio de Janeiro, 2012. 89p. Dissertação de Mestrado - Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Um dos principais sintomas de declínio da qualidade arquitetural em projetos de software é a manifestação contínua de anomalias de código. Quando estas anomalias não são detectadas e removidas com antecedência, a capacidade de evoluir e manter estes sistemas pode ser comprometida, e, eventualmente, uma reestruturação completa de suas arquiteturas é inevitável. Apesar da existência de diversas técnicas e ferramentas para detecção automática de anomalias de código, a identificação de anomalias que efetivamente causam problemas arquiteturais é ainda uma tarefa desafiadora e não trivial. Ademais, estudos realizados no contexto desta dissertação mostraram que desenvolvedores tendem a refatorar mais frequentemente anomalias que não causam problemas arquiteturais. Em especial, percebeu-se que desenvolvedores priorizam a refatoração de elementos de código que não afetam a arquitetura dos sistemas, como métodos privados ou módulos internos de um componente arquitetural. Neste contexto, o presente trabalho propõe uma abordagem para priorização de anomalias de código. Esta abordagem é composta por heurísticas que exploram diferentes fatores para identificar e ordenar as anomalias detectadas de acordo com suas relevâncias arquiteturais. Tais fatores compreendem desde a quantidade de mudanças realizadas no código ao longo da evolução dos sistemas, até os papéis arquiteturais por ele desempenhados. Foi ainda implementada uma ferramenta para aplicar tais heurísticas de priorização automaticamente em projetos Java. A abordagem proposta foi avaliada em 4 projetos de software de diferentes domínios. Tal avaliação revelou que mantenedores de software poderiam ser beneficiados pelas recomendações de priorização produzidas pela ferramenta, de modo a investir seus esforços de refatoração na solução de problemas arquiteturalmente relevantes.

Palavras-chave

Anomalias de código; refatoração; degradação arquitetural.

Summary

Figures

Tables