

PLANO DE TRABALHO PARA O MESTRADO

DESENVOLVIMENTO DE UM MODELO CONCEITUAL DE ANÁLISE DE CÓDIGO

Universidade Federal de Santa Catarina – UFSC

Programa de Pós-Graduação em Ciência da Computação – PPGCC

Professora Orientadora: Christiane A. Gresse von Wangenheim

1. Introdução

A qualidade de software é muito importante, no entanto na prática ela costuma ser imperfeita. O SWEBOK apresenta diversas técnicas que podem ser utilizadas para melhorar a qualidade, as quais incluem teste, revisão do código fonte e especificação formal (BOURQUE & FAIRLEY, 2014). Uma forma de análise da qualidade do software é a realização de uma análise estática do código, sem executar o código ou considerar uma entrada específica (AYEWAH et al. 2008). Em vez de tentar provar que o código cumpre suas especificações, estas análises buscam violações de práticas de programação recomendadas e fornecem *feedback* sobre a qualidade do código do software medindo diversas métricas pré-estabelecidas.

A análise estática geralmente é realizada por uma ferramenta automatizada a qual analisa todo o código fonte do programa para um conjunto de tópicos pré-definidos. Estas ferramentas podem ser configuradas para detectar problemas funcionais, tais como, vazamento de memória ou lógica incorreta e problemas de manutenção de código, tais como, desacordo com práticas recomendadas de programação ou violações de convenções de estilo (BELLER et al., 2016). Tradicionalmente as ferramentas automatizadas de análise estática usam técnicas como análise do fluxo de dados e análise do fluxo de controle para detectar defeitos no código fonte (D’SILVA & WEISSENBACHER, 2008). Um exemplo é a ferramenta Dr. Scratch, a qual avalia algumas práticas de programação empregadas na linguagem Scratch (MORENO E ROBLES, 2015).

Algumas ferramentas também realizam a análise estática do código examinando a documentação (BOURQUE & FAIRLEY, 2014). Atualmente, existem diversas ferramentas para a análise de código usando diversas linguagens de programação (BELLER et al., 2016). As ferramentas existentes também variam bastante em termos de métricas que analisam e defeitos de software que detectam (BELLER et al., 2016). E, apesar de existirem várias ferramentas para análise de código para diversas linguagens de programação, não existe um modelo conceitual genérico (independente de uma linguagem de programação) que identifique as características do código que devem ser analisados, nem um processo/arquitetura genérico para o desenvolvimento de um analisador de código estático.

2. Objetivos

O objetivo geral desse trabalho é desenvolver, aplicar e avaliar um modelo conceitual de análise estática de código. O modelo conceitual definirá um conjunto genérico de características/métricas de código a ser analisado. O modelo também definirá uma arquitetura genérica para analisadores estáticos de código independente de uma linguagem de programação específica. O modelo desenvolvido permitirá a criação de analisadores estáticos de código para uma linguagem de programação e finalidade específica.

O objetivo desse trabalho está inserido na linha de pesquisa de Engenharia de Software do PPGCC dentro dos tópicos de Qualidade de Software conforme a definição da área de Engenharia de Software da SBC (SBC, 2005).

Objetivos específicos:

O1. Analisar a fundamentação teórica sobre qualidade de software e análise de código.

O2. Analisar o estado da arte em relação a modelos de análise de código e o levantamento do estado da prática, como ferramentas de análise de código;

O3. Desenvolver um modelo conceitual de avaliação identificando de forma sistemática características do código a serem analisadas e uma arquitetura genérica a partir de uma análise qualitativa da literatura existente considerando os requisitos e necessidades identificadas;

O4. Avaliar o modelo desenvolvido em um estudo de caso utilizando um analisador de código criado a partir do modelo definido na prática para analisar a qualidade de código.

3. Contribuições científicas potenciais

A principal contribuição científica do presente deste trabalho na área de engenharia de software/qualidade de software é o desenvolvimento de um modelo conceitual de análise de código estático.

Além disso, esperam-se as seguintes contribuições científicas:

- Levantamento do estado de arte/prática de forma sistemática fornecendo uma visão geral sobre esta questão de pesquisa;

- Dados e avaliação de analisador de código criado a partir do modelo desenvolvido;

- Um modelo de analisador de código estático, que possa ser utilizado para sistematicamente criar novos analisadores de código para linguagens de programação específicas.

Com base nestes resultados na área de engenharia de software visa-se contribuir também na melhoria da qualidade deste tipo de software.

Referências

AYEWAH, N., et al. **Using Static Analysis to Find Bugs**. IEEE Software, vol. 25, pp. 22-29, Set.-Out. 2008.

BELLER, M. et al. **Analyzing the state of static analysis: A large-scale evaluation in open source software**. In: Proceedings of the 23rd IEEE International Conference on Software Analysis, Evolution, and Reengineering. IEEE, pp. xxx–xxx, 2016.

BOURQUE, P.; FAIRLEY, R. E. **Guide to the Software Engineering Body of Knowledge**. Versão 3.0. IEEE Computer Society, 2014. Disponível em: <www.swebok.org>. Acesso em: out. 2016.

D'SILVA, D. K.; WEISSENBACHER, G. **A survey of automated techniques for formal software verification**. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems , vol. 27, no. 7, pp. 1165–1178, 2008.

MORENO-LEÓN, J.; ROBLES, G. **Computer programming as an educational tool in the English classroom a preliminary study**. In: 2015 IEEE Global Engineering Education Conference (EDUCON). IEEE, p. 961-966, 2015.

SBC. **Currículo de Referência da SBC para Cursos de Graduação em Bacharelado em Ciência da Computação e Engenharia de Computação**. Sociedade Brasileira de Computação, 2005.