UNIVERSIDADE FEDERAL DE PERNAMBUCO – UFPE

Geração de Testes com Feedback Humano

Proposta de Trabalho de Graduação

Centro de Informática – Cin

Recife, Janeiro de 2013

Aluna

Adriana Libório Fernandes Lins

alfl@cin.ufpe.br

Orientador

Marcelo d'Amorim

damorim@cin.ufpe.br

Índice

1. Contexto	3
2. Objetivo	4
3. Cronograma	5
4. Referências.....	6
5. Possíveis Avaliadores	7
6. Assinaturas.....	8

1. Contexto

Testes de Software é atividade essencial, mas ainda muito cara do desenvolvimento de Software, assim muitas pesquisas tem questionado como automatizar a criação de testes da melhor maneira possível. A criação de testes baseada em busca tem sido proposta como uma possível técnica para produzir testes para programas, resultando em conjuntos de testes que conseguem alta cobertura de código.

Enquanto os seres humanos continuam a participar no desenvolvimento do sistema, erros provavelmente continuarão a serem introduzidos no código e testes de software continuarão a ser uma atividade importante para a engenharia de software. Na indústria, o teste é certamente hoje a técnica mais popular para verificação e validação de software.

Infelizmente, o processo de testes tem suas limitações. Ele pode não cobrir todos os casos necessários e pode consumir energia humana significativa. A automação pode ajudar a melhorar os testes. Técnicas de automação de testes anteriores mostraram alguns bons resultados em algumas configurações de avaliação, quando usando o código controlado. No entanto, um estudo recente mostra que essas técnicas não funcionam bem quando o aplicativo manipula informações externas, como em reais projetos *open-source*. Fraser e Arcuri [1], recentemente conduziram um estudo envolvendo 100 aplicações reais selecionados ao acaso que mostram que a eficácia das técnicas de teste de geração de vários propostos na literatura tem sido superestimada. O estudo mostra que, quando se considera os programas que contém uma quantidade significativa de dependências com o ambiente de teste, a eficácia das técnicas de geração de teste é significativamente mais baixa do que o que foi anunciado. Em particular, o estudo constatou que a geração de teste executa mal para programas com dependências com bancos de dados, arquivos e outros sub-sistemas e, infelizmente, o estudo também mostra que estes programas parecem dominar na prática. Em outro estudo, envolvendo quatro grandes aplicações, Xiao et al. [9, 10] também relatam dificuldades semelhantes que teste de geração de rosto técnicas. Estes dois estudos aplicaram diferentes técnicas de teste (por exemplo, a geração de teste aleatório, evolutivo e simbólico) e, portanto, essas limitações se aplicam a diferentes formas de automação de testes.

Apesar dessas limitações práticas, é importante notar que ele tem sido um senso comum na comunidade acadêmica que a enorme quantidade de contribuições neste campo não foi em vão. Ferramentas de geração de teste ainda podem ser úteis. Por exemplo, eles podem ajudar a encontrar casos específicos em que os testadores teriam dificuldade de encontrar por si próprios.

2. Objetivo

Este projeto investiga maneiras de habilitar os seres humanos e os geradores de teste para interagir, alavancando o poder de cada um individualmente. A intuição é que os resultados dos testes globais de melhorar significativamente quando os testadores participar mais ativamente do processo de teste de geração. Em particular, os testadores podem fornecer orientações para a pesquisa na base das dificuldades de um motor gerador. O principal desafio deste projeto é identificar as interfaces boas de que o usuário da ferramenta e da ferramenta para o usuário. O sucesso deste projeto contribui diretamente para reduzir a lacuna entre a pesquisa e a prática de engenharia de software industrial.

A interação do ser humano e de teste-geradores foi investigado superficialmente no passado [6,10,11]. Uma razão prática para a literatura que descreve os resultados da investigação rasos é que os estudos empíricos que envolvem os seres humanos são geralmente difíceis de avaliar. Este projeto se relaciona fortemente com as pesquisas anteriores sobre o teste interativo. No entanto, com o melhor de nosso conhecimento, centra-se em dois aspectos menos explorados: (a) estuda como melhorar a geração de teste para evitar difíceis de serem satisfazer requisitos de teste, por exemplo, associando métodos de fábrica para os parâmetros do método em particular, e (b) que estuda maneiras de fornecer feedback de qualidade para o usuário, por exemplo, informando o testador que partes de uma sequência de teste de pequeno porte que se aproxima para satisfazer uma exigência de teste têm impacto sobre o teste peças de exigência e que não têm impacto.

3. Cronograma

A Tabela 1 apresenta o cronograma das atividades a serem realizadas para o cumprimento do trabalho proposto neste documento. As linhas representam as principais atividades necessárias à conclusão deste trabalho, enquanto que as colunas representam os meses decorridos até a finalização do mesmo. As colunas estão subdivididas em semanas, para melhor visualizar a execução das atividades. Assim, cada célula representa uma atividade que será executada durante a semana correspondente. Como mostrado na tabela, diferentes atividades poderão ser executadas paralelamente.

Atividade	Dezembro				Janeiro				Fevereiro				Março				
Revisão Bibliográfica e Definição do escopo do projeto	■	■	■	■													
Implementação de Protótipo				■	■	■	■	■	■								
Experimentos									■	■	■	■					
Elaboração do Relatório												■	■	■			
Elaboração de Apresentação														■	■		

Tabela 1: Cronograma de atividades

4. Referências

- [1] Fraser, G. and Arcuri, A., "Sound Empirical Evidence in Software Testing," in *ICSE'12, 2012*, pp. 178-188.
- [2] Bruntink, M., "Testability of Object-Oriented Systems: a Metrics-based Approach," *Master Thesis. The University of Amsterdam*, 2003.
- [3] Fraser, G. and Arcuri, A., "The Seed is Strong: Seeding Strategies in Search-Based Software Testing," in *ICST'12*, pp. 121-130.
- [4] Andrews, J., Groce, A., Weston, M. and Ru-Gang Xu, "Random Test Run Length and Effectiveness," in *ASE'08*, pp. 19-28.
- [5] Groce, A., Zhang, C., Eide, E., Chen, Y. and Regehr, J., "Swarm Testing," *ISSTA'12*, pp. 78-88
- [6] Pavlov, Y. and Fraser, G., "Semi-automatic Search-Based Test Generation," in *ICST'12 (SBST)*, 2012, pp. 777-784.
- [7] Kosorukoff, A., "Human based genetic algorithm," *International Conference on Systems, Man, and Cybernetics*, 2001, pp. 3464-3469.
- [8] Bruntink, M. and Deursen, A., "Predicting Class Testability using Object-Oriented Metrics," in *SCAM '04, Fourth IEEE International Workshop*, 2004, pp. 136-145.
- [9] X. Xiao, T. Xie, N. Tillman, and Johnathan de Halleux. "structural test generation," *ICSE'11*, pp. 611-620
- [10] X. Xiao, T. Xie, N. Tillman, and Johnathan de Halleux. "Covana: precise identification of problems in pex," *ICSE'11*, pp. 1004-1006

5. Possíveis Avaliadores

- Paulo Borba

6. Assinaturas

Marcelo d'Amorim
Orientador

Adriana Libório Fernandes Lins
Aluna