

TRABALHO PRÁTICO

ANOMALIAS E REFATORAÇÃO

Objetivos

O objetivo deste trabalho é permitir entender como anomalias de código (*code smells*) e de anomalias arquiteturais (*architectural bad smells*) afetam negativamente a qualidade de um sistema de *software* e como técnicas de refatoração podem ser empregadas para solucioná-las.

Tarefas

A tarefa central a ser realizada neste trabalho consiste em identificar anomalias de código e anomalias arquiteturais em um sistema de *software* já desenvolvido ou em estágio de desenvolvimento, bem como que técnica(s) de refatoração poderia(m) ser utilizada(s) para solucioná-las. A identificação de tais anomalias e das técnicas de refatoração deve ser feita com base em catálogos existentes na literatura. Como resultado, deverá ser elaborado um relatório escrito, preferencialmente em formato PDF, contendo, **no mínimo**:

- uma visão geral acerca do sistema, descrevendo (i) o contexto no qual ele se insere, (ii) o problema que ele visa endereçar, (iii) seus objetivos, (iv) os *stakeholders* envolvidos e (v) uma caracterização da sua arquitetura e do seu código fonte;
- uma descrição das anomalias arquiteturais identificadas, como elas se manifestam na arquitetura do sistema de *software* sob análise;
- uma descrição das anomalias de código identificadas e como elas se manifestam na implementação do sistema de *software* sob análise, e;
- uma descrição de que técnicas de refatoração necessitariam ser utilizadas a fim de solucionar as anomalias identificadas e como elas seriam aplicadas sob o sistema de *software* em questão.

Autoria e política de colaboração

O trabalho deverá ser feito **em equipe composta por no máximo dois estudantes**, sendo que, neste último caso, é importante, dentro do possível, dividir as tarefas igualmente entre os integrantes da equipe. O trabalho em cooperação entre estudantes da turma é estimulado, sendo aceitável a discussão de ideias e estratégias. Contudo, tal interação não deve ser entendida como permissão para utilização de (parte de) trabalhos de outras equipes, o que pode caracterizar situação de plágio. Trabalhos copiados em todo ou em parte de outras equipes sumariamente rejeitados e receberão nota zero.

Entrega

A entrega deste trabalho será feita de duas formas, a saber:

- por meio de uma apresentação oral **com duração de no máximo 10 minutos** à turma e aos professores responsáveis pelo componente curricular IMD0122 – Aprendizado Baseado em Projeto 4, sessão que terá início às **11h do dia 24 de agosto de 2018**;
- por meio de entrega do relatório escrito através da opção *Tarefas* da Turma Virtual do SIGAA **até às 23h59 do dia 26 de agosto de 2018, prazo que não será estendido**;

Durante a sessão de apresentação oral, **à qual é obrigatória a presença**, cada equipe deverá responder, com desenvoltura, aos questionamentos que porventura forem levantados pela turma e/ou pelos professores presentes. Nessa apresentação também serão analisadas a autoria do trabalho desenvolvido e a contribuição real de cada integrante da equipe, mesmo que uma determinada parte do trabalho tenha sido conduzida por outro membro da equipe.

Bibliografia sugerida

A título de sugestão, os seguintes materiais (evidentemente não limitados a estes) podem ser utilizados como base para pesquisa e estudo acerca do conteúdo abordado neste trabalho:

- FOWLER, Martin; BECK, Kent; BRANT, John; OPDYKE, William; ROBERTS, Don. *Refactoring: Improving the design of existing code*. USA: Addison-Wesley Longman, Inc., 1999.
- GARCIA, Joshua; POPESCU, Daniel; EDWARDS, George; MEDVIDOVIC, Nenad. Identifying architectural bad smells. In: EUROPEAN CONFERENCE ON SOFTWARE MAINTENANCE AND REENGINEERING, 13., 2009, Kaiserslautern, Germany. *Proceedings...* USA: IEEE Computer Society, 2009, p. 255-258.
- _____. Toward a catalogue of architectural bad smells. In: MIRANDOLA, Rafaella; GORTON, Ian; HOFMEISTER, Christine (Eds.) *Proceedings of the 5th International Conference on Quality of Software Architectures*, East Stroudsburg, USA. *Lecture Notes in Computer Science*, v. 5581. Germany: Springer-Verlag Berlin-Heidelberg, 2009, p. 146-162.
- SOURCEMAKING. *Code smells*. Disponível em: <<https://sourcemaking.com/refactoring/smells>> Acesso em: 17 de agosto de 2018.
- _____. *Refactoring techniques*. Disponível em: <<https://sourcemaking.com/refactoring/refactorings>> Acesso em: 17 de agosto de 2018.
- SURYANARAYANA, Girish; SAMARTHYAM, Ganesh; SHARMA, Tushar. *Refactoring for software design smells: Managing technical debt*. USA: Morgan-Kauffman/Elsevier, Inc., 2015.

Avaliação

A avaliação deste trabalho será feita principalmente com relação à (i) clareza da apresentação oral e (ii) à qualidade do relatório produzido. O trabalho possuirá nota máxima de 10,0 (dez) pontos e será contabilizada para a nota final da disciplina.

Observações

Dúvidas relativas à execução deste trabalho poderão ser sanadas presencialmente em sala ou entrando-se em contato com o professor via *e-mail* através do endereço everton@dimap.ufrn.br.