

ANÁLISE DE IMPACTO EM ALTERAÇÕES DE SOFTWARE

Luciani Pizato, Thalís Ruthes

UNISEP

Resumo - O crescente aumento da necessidade de automatização de processos e regras de negócio nas empresas, leva aos desenvolvedores de software uma carga elevada de novos processos a serem desenvolvidos. Novos processos necessitam de muita validação para a garantia do perfeito ou satisfatório funcionamento, e métodos já desenvolvidos necessitam da garantia do funcionamento. Este artigo mostra, através da elaboração de um software, uma sugestão de como pode ser o processo de controle de alterações na área de desenvolvimento e testes de software. Obtém-se, com este aplicativo, uma forma de facilitar e agilizar a rastreabilidade dos requisitos implementados em softwares orientados a objetos, tornando claro o processo de rastreabilidade e os impactos ocasionados pelas mudanças geradas durante os testes e manutenções de software.

Palavras-Chave: Rastreabilidade, Orientação a objeto, Engenharia de software, Testes de software.

IMPACT ANALYSIS IN SOFTWARE CHANGES

Abstract- The increase of the necessity of processes automation and business rules in companies, leads to the software developers a high quantity of new processes to be developed. New processes need much validation for the guarantee of perfect or satisfactory operation, and methods already developed, needs the guarantee of the good operation. This article will show how must proceed the process of control changing in the area of development and tests of software, trying to show a way to facilitate and speed the traceability of implemented requirements. Setting some important areas, to be understood how doing the traceability of implemented requirements with the use of the object oriented programming.

KeyWord: Traceability, Object orientation, Engineering of Software, Software Tests.

1. INTRODUÇÃO

De acordo com Jordão (2007), é raro encontrar uma pessoa que não conviva com a tecnologia, sendo que ela tornou-se um utilitário fundamental na vida de qualquer cidadão, pois ela pode servir tanto para fins de entretenimento quanto para fins profissionais.

Sabe-se que o software baseia-se em requisitos e Filho (2001) determina que dentro de um software os requisitos podem variar muito, definindo assim uma enorme instabilidade, causando um elevado custo para reparos.

Observa-se um crescimento de forma elevada e rápida no desenvolvimento de processos de

software, sendo identificado à necessidade de ferramentas para o auxílio nos processos de desenvolvimento. Para chegar ao desenvolvimento de ferramentas de auxílio à garantia da qualidade de software, parte-se do princípio da utilização de orientação a objetos, levando em consideração as fases de projeto, análise de requisitos e testes de software.

1.1. Desenvolvimento orientado a objeto

A orientação a objetos é um modelo de processo de desenvolvimento de software que orienta o código com alguns conceitos de categorização dos elementos do sistema. Martin (1994) define como qualquer coisa classificada como real ou abstrata,

através da qual é possível armazenar, manipular dados e produzir ações.

No desenvolvimento de sistemas de informação, parte-se do conceito habitual de que objetos são coisas que possuem suas determinadas características e podem fornecê-las para outros objetos que venham a executar outra função específica, possibilitando herdar operação genérica de um objeto principal.

1.2. Fases de desenvolvimento de software

O processo de desenvolvimento de software constitui-se principalmente de três fases, a análise, focada no levantamento de necessidades, o desenvolvimento, destinado para a codificação e os testes, voltados para o controle de qualidade do software. Os requisitos são fundamentais em todas estas fases, sendo que eles devem ser muito bem especificados, bem como suas restrições, pela análise para que possam ser desenvolvidos e testados de acordo com o desejado pelo cliente.

Para Sommerville (2003), os requisitos possuem muitos elos com a sua proposta de funcionamento, por isso, no momento em que se solicita uma mudança no requisito é fácil de compreender que deverá ser avaliado o impacto desta alteração nos outros requisitos do sistema.

Na fase de teste é principalmente onde se detecta a necessidade de uma ferramenta de análise de impacto, que leve em consideração os requisitos implementados ou alterados e suas ligações com os demais requisitos do sistema. Esta integração torna possível um controle mais eficaz de mudanças e a garantia de manter a qualidade dos testes de software.

1.3. Rastreabilidade de requisitos e controle de impactos

A rastreabilidade de requisitos é uma forma de integração entre todos os requisitos que fazem parte do projeto de um software.

Com a afirmação de Sommerville (2003), no tópico "1.2", torna-se claro que um sistema de informação precisa ser rastreável, para que a fábrica de software tenha a possibilidade de testar os requisitos implementados e saber qual é o impacto que estes implicarão nos objetos já existentes no sistema. No entanto, se estes requisitos não forem controlados, os impactos gerados durante o rastreamento não acrescentam qualidade ao software e ao projeto. É através do controle de impacto, que a fábrica de software torna ágil o processo de correção e melhoria do software.

2. MATERIAL E MÉTODOS

Pressman (1995) prega que o controle de mudanças previne impactos ruins no processo de desenvolvimento. Para isso é necessário que os

pedidos de mudança sejam formalizados, avaliando a natureza e controlando o impacto da mudança de qualquer requisito.

Partindo deste princípio, discutido por Pressman, identificou-se a necessidade de uma ferramenta que auxiliasse no controle dos requisitos utilizados pelos softwares. Através desta ferramenta é possível agilizar o trabalho da equipe de testes e controlar o impacto gerado por mudanças, sempre necessárias, no sistema. O aplicativo foi desenvolvido utilizando a linguagem de programação para Web Php e o banco de dados MySQL, que se mostraram suficientes para o que foi proposto. A idéia de desenvolver focando-se no ambiente Web foi por motivo de acessibilidade.

A ferramenta foi desenvolvida com o objetivo de agilizar a tarefa de rastreabilidade e controle dos mais variados requisitos trabalhados por empresas que desenvolvem software orientado a objetos.

A figura 01 mostra algumas das principais interfaces apresentadas pelo programa para auxiliar no controle de impactos gerados pelas mudanças efetivadas no sistema. Através da ferramenta, o usuário (testador) pode efetuar uma busca pelo relacionamento de um objeto e o software permite identificar os relacionamentos, desde módulos até objetos diretamente influenciados. Sendo assim, o aplicativo facilita o trabalho do usuário, retornando toda a gama de relacionamentos existentes com este objeto no software e garantindo a rastreabilidade e o impacto positivo das mudanças.

MÓDULO SELECIONADO		OBJETO SELECIONADO	
Vendas		w_notas_saida	
Informações do objeto		Operações	
OBJETOS	PLANOS DE TESTE	Operações relacionadas : Converter nota de venda	
w_acc_vendas	1 - Saídas	Informações do objeto	
w_notas_saida	8 - Descontos vendas	Módulo relacionado : Vendas	
		Objetos	
		Diretamente Relacionados: w_acc_vendas	
		tx_fiscal	
		w_notas_saida	
		Ancestral: w_acc_vendas	

Figura 01: Rastreabilidade de objetos

3. RESULTADOS

O desenvolvimento de software exige atenção especial, principalmente em se tratando de manter a qualidade do mesmo. A rastreabilidade de requisitos e o controle de impactos de alterações são meios que proporcionam esta qualidade. Quando a fábrica de software não foca-se em requisitos rastreáveis, o software corre um sério risco de não atender as

necessidades do cliente e de apresentar um nível de falhas ocultas muito grande. O sistema proposto reduz este número de falhas através de um controle mais aprimorado do processo de desenvolvimento, controlando a interligação entre os requisitos e garantindo uma visibilidade ampla de possíveis impactos no momento das alterações.

Através da ferramenta, pode-se verificar que o tempo de manutenção foi reduzido substancialmente e a fase de teste torna-se mais ágil e eficaz, pois garante maior controle no próprio processo. A mesma é utilizada desde o processo de análise, porém mostra-se fundamental nos aspectos

de manutenção de software.

REFERÊNCIAS

JORDÃO, C. Profissões de futuro. **Revista ISTOÉ**, São Paulo, v. 1793, setembro/2007.

MARTIN, J. **Princípios de análise e projetos baseado em objetos**. 4.ed. Editora Campus: Rio de Janeiro, 1994.

SOMMERVILLE, I. **Engenharia de software**. 6. Adisson-Wesley : São Paulo, 2003.

FILHO, W. P. P.. **Engenharia de Software – Fundamentos, Métodos e Padrões**. 2.ed, Editora LTC: Rio de Janeiro, 2001.

PRESSMAN, Roger S. **Engenharia de software**. Makron Books: São Paulo, 1995.