

1 Introdução

Este trabalho apresenta os resultado do projeto de pesquisa **Negociação de Requisitos: uma análise da viabilidade de integração entre técnicas que estimulam a participação do cliente**, que tinha como objetivo fazer uma levantamento e estudo sobre métodos, técnicas ou estratégias que facilitassem a negociação de requisitos de software e verificar sobre a possibilidade de formulação de um abordagem metodológica de negociação que pudesse ser desenvolvida e aprimorada para que pudesse ser adaptada utilizada pelas empresas de Teresina-PI. O nome do projeto sofreu uma pequena alteração para que pudesse refletir o seu objetivo. O projeto passo a ter o título de **Negociação de requisitos de software : uma analise da viabilidade de integração entre técnicas que estimulam a participação do cliente em negociações de requisitos de software**. Procuramos demonstrar a partir de argumentos, práticas e constatações de autores que estudam o assunto e assuntos correlatos bem como as observações e constatações tiradas a partir da análise e de reflexões feitas durante o estudo com o fim de demonstrar a importância do tema Negociação de Requisitos. O projeto também tinha como objetivo específicos: procurar por compatibilidades e incompatibilidade entre as propostas selecionadas, procurar por pontos carentes de adaptações para torna-las compatíveis entre si, bem como identificar os pontos onde tais adaptações seriam mais fáceis e mas trabalhosas.

Estudou-se trabalhos de autores como Ramires (2004), Camacho (2005), Cysneiros (2001), Couto e Martins (2008), Xavier (2009), com a intenção de conseguir uma base de cunho específico e prático para a argumentação a ser usada na monografia e também para delimitação do tema. Os trabalhos de Carvalho(2001), Kantorski e Kroth (2004), Laurindo e Moraes (2004), Gusmão (2008), Germoglio (2008), foram analisados com o fim de se extrair relações com os trabalhos dos autores anteriormente citados e enriquecer a pesquisa bem como: verificar tendências, contextualização de situações, relações entre as abordagens, em fim, verificar pontos ou aspectos interessantes e em comum. Estudou-se também trabalhos mais gerais como Carvalhal (2006), Leite (1994), Lopes (2002), Pressman (2001), (PMBOK, 2004), Lee, Cha e Kwon (1998), Swabok (2004), Sommerville (1997), entre outros, pelos motivos anteriormente citados e, também, para reforçar as argumentações e

alinhar as idéias específicas com idéias gerais, amplamente aceitas e já legitimadas no meio da engenharia de software. Outras obras além das acima citadas (elas representam a base central deste trabalho) também foram consultadas e encontram-se nas referências e no decorrer do texto.

[1] Comentário: Verificar se todas as citações e referências estão de acordo com a abnt
---- 09/01/2011, 11:28 ----

Durante a fase de Requisitos (elicitação, documentação e análise) podem ser identificados conflitos entre os requisitos do cliente. O cliente torna-se “obrigado” a negociação junto aos engenheiros sobre a satisfação dos requisitos que envolvem aspectos, muitas vezes, de difícil compreensão para o cliente. Isso pode dificultar a boa participação do cliente durante o processo de desenvolvimento de software.

O cliente tendo pela frente uma situação de negociação envolvendo aspectos que não conhece necessita de tempo para o entendimento do objeto de negociação, dos riscos por traz da não resolução do conflitos, para que aí sim possa tomar decisões sobre requisitos de forma lúcida.

Geralmente o cliente solicita alterações nos requisitos quando o software já passou pela programação (implementação em linguagem de programação) e uma amostra já foi apresentada. Consertar um defeito relacionado a requisito depois da implementação em uma linguagem de programação é sem dúvida mais dispendioso que fazer isso durante fase de requisitos.

Este trabalho aborda o problema da não preparação dos processos de desenvolvimento de software para as necessidades e limitações dos clientes quanto a visualização das consequências dos impactos entre requisitos. Faz um estudo da viabilidade de formulação de uma metodologia que posiciona o projeto de desenvolvimento de software de forma a suprir as necessidades e limitações dos cliente no que diz respeito ao entendimento dos efeitos positivo e negativos que os requisitos podem causar uns nos outros e nas fases seguintes do desenvolvimento.

Com a evolução acelerada da informática o mercado de trabalho vem se tornando cada vez mais competitivo e exigente. Devido a isso a demanda por softwares que cada vez

mais satisfaçam as necessidades dos clientes e que, ao mesmo tempo custem menos, sejam mais complexos e mais rápidos de serem desenvolvidos aumenta rapidamente.

Ao longo da história da Engenharia de software tem se observado que a importância dada aos requisitos vem aumentando. Porém, o número de trabalhos feitos em que o foco maior se concentra nos requisitos funcionais é muito alto se comparado com os que, além dos funcionais, consideram os atributos de qualidade. Muito avanço já se conseguiu com pesquisas onde o foco maior estava em requisitos funcionais: desenvolvimento de software dirigido por casos de uso, especificação de requisitos com caso de uso, especificação de casos de uso, análise de pontos de função, etc...

A busca pela qualidade faz com que haja uma tendência natural em se preocupar com atributos de qualidade e restrições. Alguns poucos estudos já consideram que a qualidade que o cliente espera que o seu software tenha não é determinada somente com base em requisitos funcionais. Descrever somente visão funcional não é o suficiente. Deve existir esforço para desenvolver modelos em que os atributos de qualidade e restrições possam ser documentados e analisados (VAROTO, 2002).

Nesse cenário, de preocupação em como o software deverá realizar suas funcionalidades, temos possibilidade de identificação de possíveis impactos positivos ou negativos em outros requisitos que determinam a qualidade do software. No caso dos impactos negativos surge a necessidade de negociação para que se escolha um requisito em detrimento de outro ou que se contente com a satisfação parcial de um, ou para que se encontre alternativas que satisfaçam a necessidade das partes sem que haja conflitos ou que a ocorrência deles seja a menor possível. Mas isso se agrava com o fato do cliente (leigo) não conseguir entender e, por consequência, não conseguir visualizar esse impacto tornando a negociação algo extremamente difícil para as partes. Isso se deve ao caráter abstrato que o software possui.

O sucesso que é esperado de um projeto de software é altamente relacionado e dependente de uma arquitetura bem definida e apropriada. Mas para isso é necessário que haja sucesso nas negociações, ou seja, que elas sejam realizadas com eficiência e efetividade. Então para que se tenha uma arquitetura consistente e confiável primeiro é necessário que se

tenha o mínimo de requisitos (selecionados lucidamente) necessários à definição da arquitetura e isto depende muito do resultado de negociações. Uma arquitetura mal definida afeta a comunicação da equipe de desenvolvimento, o crescimento do software e decisões sobre as estruturas que formarão o sistema. Afeta também, a estrutura global de controle que será usada, protocolos de comunicação, sincronização e acesso a dados, atribuição de funcionalidades a elementos do sistema ou ainda sobre distribuição física dos elementos do sistema. Além disso, o problema envolverá decisões que impactarão no comportamento do sistema em termos de escala e desempenho. Garlan e Shaw (apud GERMOGLIO, 2009). Mas tão quão a comunicação da equipe de desenvolvimento do software temos que a comunicação nas negociações iniciais, que ocorrem entre cliente e equipe de engenharia de software, gerentes de projeto etc, precisam ser melhoradas, bem como a manipulação ágil e representação dos objetos de negociação, que no caso deste trabalho são os requisitos funcionais e não funcionais.

Mudanças nos requisitos tornam o processo de desenvolvimento cada vez mais caros na medida em que as fases do processo de desenvolvimento vão se passando. Correções nas fases iniciais são mais baratas. E isso obriga os clientes e engenheiros de software a anteciparem as tomadas de decisões quanto a fatores críticos do projeto.

Por isso se faz necessária a negociação de requisitos conflitantes para tomada antecipada de decisões necessárias, afim de que se chegue a um consenso comum entre clientes e engenheiros de software sobre os requisitos principais e, também para que se tenha o mínimo de informação para que se possa prosseguir com a fase de projeto e de desenvolvimento.

Assim qualquer esforço no sentido de diminuir tempo de familiarização do cliente com aspectos inerentes aos impactos dos requisitos durante a negociação dos requisitos com o objetivo de diminuir os riscos de mudanças em fase avançadas do processo de desenvolvimento é bem vindo. Uma idéia nesse sentido seria criar um mecanismo que facilite identificar os requisitos críticos para o sucesso do projeto e facilitar o consenso entre cliente e engenheiros sobre o escopo de requisitos. E que ao mesmo tempo garanta que as principais tomadas de decisão sejam feitas nas fases iniciais, reduzindo assim os riscos referente a custo,

tempo e qualidade.

A necessidade constante de sempre produzir softwares que cada vez mais custem menos, que sejam produzidos mais rapidamente, e de acordo com as expectativas (e necessidades) dos stakeholders (ou seja, que tenham qualidade), revelam outras necessidades mais específicas dos processo de desenvolvimento. A necessidade de que o cliente seja mais participativo no processo de desenvolvimento de software é uma delas. Para a elicitación, análise e negociação de requisitos isso é fundamental.

Embora Elicitación, Análise e negociação de requisitos sejam conceitos que sempre andam juntos durante o processo de desenvolvimento, esse trabalho foca-se na negociação de requisitos. Mas não sugere separação entre essas atividades, por fazerem parte de um processo em espiral (SOMMERVILLE, SAWYER, 1997).

A negociação é um instrumento e, ao mesmo tempo, uma necessidade presente durante o processo de desenvolvimento de software. É essencial e de uso inevitável em qualquer projeto que envolva pelo menos duas partes. Ela também tem suas necessidades próprias. E busca-se produzir instrumentos e desenvolver técnicas, métodos , estratégias etc. com o objetivo de satisfazer suas necessidades.

Nas negociações pode-se verificar o grau de contentamento, necessidade, interesse atuais no qual o cliente se encontra e outras coisas mais.

O capítulo 2 deste trabalho apresenta alguns conceitos básicos de Engenharia de Requisitos necessários ao entendimento deste trabalho.

O capítulo 3 discorre sobre Negociação de uma forma geral e básica para que se possa realizar associações às necessidades do processo de desenvolvimento de software.

O capítulo 4, chamado de Verificação de Viabilidade apresenta o resultado da verificação das análises de compatibilidades entre as propostas estudadas; os pontos sugeridos para adaptações; as facilidades e possíveis dificuldades para realizar as adaptações.

O capítulo 5 apresenta as conclusões tiradas durante a pesquisa e também as idéias

de trabalhos futuros que podem ser realizados a partir deste trabalho.

