

CENTRO DE ENSINO UNIFICADO DE TERESINA
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE SOFTWARE

SEBASTIAO VINICIUS ARAUJO DE MOURA

**NEGOCIAÇÃO DE REQUISITOS DE SOFTWARE :
UMA ANÁLISE DA VIABILIDADE DE INTEGRAÇÃO ENTRE
TÉCNICAS QUE ESTIMULAM A PARTICIPAÇÃO DO CLIENTE EM
NEGOCIAÇÕES DE REQUISITOS DE SOFTWARE**

Tersina-PI

2009

CENTRO DE ENSINO UNIFICADO DE TERESINA
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE SOFTWARE

SEBASTIAO VINICIUS ARAUJO DE MOURA

**NEGOCIAÇÃO DE REQUISITOS DE SOFTWARE :
UMA ANALISE DA VIABILIDADE DE INTEGRAÇÃO ENTRE
MÉTODOS QUE ESTIMULAM A PARTICIPAÇÃO DO CLIENTE EM
NEGOCIAÇÕES DE REQUISITOS DE SOFTWARE**

Pré-projeto de pesquisa apresentada à Prof.
Francisca Neusa Farias do Centro de Ensino
Unificado de Teresina como parte dos requisitos
para aprovação na disciplina de Metodologia
Científica.

Tersina-PI

2009

Sumário

1-Introdução	5
2- Justificativa.....	6
4- Revisão Bibliográfica	9
4 Objetivos	12
5 Metodologia	13
6 Cronograma de Atividades de Pesquisa.....	16
7 Referências bibliográficas.....	16

Versões revisadas anteriores

Versão revisada	Comentário	Data
0.1	alteração 4º parágrafo da Justificativa; remoção de colchetes das referências;	09/10/10

1 Introdução

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 requer tempo para 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 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 a fase de requisitos.

Essa pesquisa aborda o problema da não preparação dos processos de desenvolvimento de software para as necessidades e limitações dos clientes e dos engenheiros quanto a visualização das consequências dos impactos provocados pelas alterações nos 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.

2 Justificativa

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 :

“ Mesmo com a execução rigorosa das fases de engenharia de requisitos e de análise, a experiência na construção de sistemas mostra que ainda existe uma lacuna de informações a serem especificadas para prosseguir com a fase de projeto. Isto significa que especificar e modelar o que o sistema deve fazer não é suficiente para saber como o sistema deve ser estruturado e organizado para satisfazer os requisitos funcionais e os atributos de qualidade.”. (VAROTO, 2002)

Nesse cenário, de preocupação em como o software deverá realizar suas funcionalidade, 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 um outra alternativa de solução que satisfaçam as necessidades das partes sem que haja conflito. Mas isso se agrava com o fato do cliente 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 antes da melhora na 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 de 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 dos stakeholders (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 elicitação, análise e negociação de requisitos sua presença é fundamental.

Embora Elicitação, Análise e Negociação de requisitos sejam conceitos que sempre andam juntos durante o processo de desenvolvimento (Sommerville & Sawyer, 1997) este trabalho foca-se na negociação de requisitos. Mas não sugere separação entre elas, 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 e, interesse atuais no qual o cliente se encontra em relação ao desenrolar do processo de engenharia de requisitos.

A engenharia de requisitos esta extremamente ligada à garantia de qualidade e sugere planejamento e prevenção ao invés de correções.

3- Revisão Bibliográfica

Na literatura nacional muitas pesquisas têm proposto soluções que têm como objetivo buscar meios para a antecipação de decisões.

(CYSNEIROS, 2001) propõem uma estratégia que trata da elicitação dos requisitos não funcionais(RNFs) ainda no início do processo de desenvolvimento do software e de como integrar os RNFs elicitados aos modelos conceituais.

(CAMACHO, 2005) apresenta uma estrategia para elicitação de requisitos de Software. O trabalho utiliza a gerencia de conflitos, que através do estímulo aos conflitos funcionais e do controle e tratamento dos conflitos não funcionais visa a criação de idéias e o dinamismo da reunião, aumentando o volume de conhecimento elicitado. A aplicação do método apresentado mostrou-se estimulante aos participantes tornando-os mais participativos.

(RAMIRES, 2004) apresenta uma abordagem para a resolução do conflito na avaliação de requisitos de software. No trabalho é apresentado um sistema que interliga um grupo de pessoas – utilizadores, programadores, chefes de projecto, gestores de topo, etc. e suporta técnicas de negociação na resolução dos conflitos que surgem na avaliação dos requisitos de software.

Em (GUSMÃO et al, 2008) apresenta uma abordagem de identificação de riscos de requisitos utilizando mapas conceituais. O método serve para expor os possíveis eventos

relacionados aos requisitos do sistema que podem afetar negativamente o projeto.

Em (HAZAN & LEITE) são propostos indicadores para apoiar um gerenciamento efetivo de requisitos funcionais por meio de indicadores gerenciais de estabilidade e rastreabilidade.

Em (LAURINDO e MORAES), é apresentada a aplicação da técnica de Monte Carlo no planejamento de projetos de TI. O artigo faz uma breve revisão sobre técnicas de programação de projetos e sua adequação para as características normalmente encontradas em projetos de TI. E a partir de uma revisão do conceito de simulação é construído um modelo utilizando a planilha eletrônica MS Excel.

Em (KANTORSKI e KROTH, 2004) é apresentada uma ferramenta de workflow que possibilita o controle da qualidade do processo de desenvolvimento de software e, ao mesmo tempo, identifique características de qualidade através de um conjunto de métricas utilizadas desde o levantamento de requisitos até a distribuição do produto de software, indicando possíveis problemas enfrentados pela equipe de desenvolvimento durante o decorrer dos trabalhos e que podem ter afetado o desempenho do processo. Os problemas identificados servirão de base para que os desenvolvedores possam encontrar falhas na execução do processo e propor melhorias para otimizar o seu desempenho. Melhorias essas, que podem ser configuradas na ferramenta de workflow proposta, agilizando o processo de desenvolvimento.

VAROTO (2002) traz uma familiarização de alguns esquemas de visões em arquitetura especificando a competência, a abrangência e os detalhes de cada visão. Também faz uma comparação entre as visões de cada esquema segundo os fatores de qualidade McCall, podendo ser utilizados como um checklist para garantir os níveis de qualidade desejados para cada solução.

(SAYÃO & LEITE) é ressaltado a importância da rastreabilidade no contexto de do processo de desenvolvimento de software, associando-a tanto a aspectos técnicos como gerenciais; No artigo os autores chegam a conclusão de que a rastreabilidade possibilita identificar artefatos de desenho, projeto e implementação afetados por uma solicitação de

mudanças , possibilitando avaliar os impactos das alterações de prazos e custo de desenvolvimento junto ao cliente. Este é um ponto de destaque e cujo o seu nível de maturidade irá determinar a qualidade e agilidade das negociações.

(ALVES, et al 2008) trata sobre o método de estimativa de Ideal Day e uma forma de priorização de trabalho de requisitos relativos a um projeto (release). O método considera uma boa pratica o alinhamento entre o Ideal Day e PERT. O artigo também incentiva o uso de matrizes de rastreabilidade.

(VASQUEZ) define o que é uma estimativa e diferencia de outras ações com as quais ela se confunde, faz uma breve introdução sobre uma séries de técnicas e ferramentas para sua realização e desfaz alguns mal entendidos sobre a dinâmica da analise de pontos de função em estimativas de software, especialmente papel do valor do fator de ajuste. As estimativas são um ponto importante ao se definir o escopo e estarão citadas nas nas argumentações dos negociadores.

##LEMBRETE: Verificar se é pertinente e viável estudar COBIT e ver como ele auxilia as ##negociações.

4 Objetivos

Neste projeto o objetivo é realizar análise das características principais, suas compatibilidades e incompatibilidades, necessidades e possibilidade de adaptações entre as propostas encontradas na literatura da área de Engenharia de Requisitos sob um ponto de vista de como elas ajudam na Negociação de Requisitos. O que se espera é chegar à conclusão sobre a possibilidade de formulação de uma metodologia (compatível com a maioria dos processo de desenvolvimento de software atuais) de negociação com as características de adequação às limitações dos clientes quanto aos fatores do processo de desenvolvimento do software.

Geral:

Elaborar uma monografia sobre o estratégias que facilitem a negociação de requisitos encontradas na literatura atual e verificar possibilidade de integrá-las visando a formação de uma metodologia de negociação.

Específicos:

- procurar e identificar compatibilidades e incompatibilidades entre as propostas selecionadas na disciplina de Engenharia de Requisitos;
- identificar e apontar pontos carentes de adaptações nas propostas afim de ajustá-las em trabalhos futuros;
- identificar os pontos onde as adaptações poderiam se dar de maneira mais amena e os pontos onde elas se tornariam mais trabalhosas;

5 Metodologia

A pesquisa será desenvolvida em três fases: a fase 1, fase 2, fase 3. Na fase 1 a pesquisa terá foco na Área de Engenharia de Requisitos e terá os seguintes objetivos: levantar o estado da arte e da prática em Gerência e negociação de Requisitos funcionais e não funcionais; identificar problemas, lacunas ou possibilidades de aperfeiçoamento na área de Gerência e negociação de requisitos; Elaboração Redação Descritiva sobre os trabalhos lidos; Procurar por formas de representação do conhecimento e fácil assimilação pelos clientes; elaborar um guia para uma pesquisa posterior(mestrado/ doutorado);

Na fase 2 a Pesquisa terá foco na Arquitetura de Software e Gerência de Riscos e Métricas de Softwares e terá os seguintes objetivos: levantar o estado da arte e da prática em especificação de Arquitetura de software; identificar atividades dessas áreas e formas de estimar o tempo e custo; buscar meios de simulação da duração e do custo das atividades; identificar possibilidade de aplicação e aperfeiçoamento na área de Arquitetura de Software e Gerência de Riscos..(porém com o objetivo de, simplesmente, explicitar as limitações que a bibliografia atual nessas áreas impõem a este projeto de pesquisa) na negociação de requisitos; Elaboração de Redação Descritiva sobre os trabalhos lidos; Expor as relações que os assuntos tem com a negociação de requisitos e como esses assuntos podem dar suporte à negociação; Elaborar material de apoio a esta pesquisa e a pesquisa futura.

Na fase três (3) o foco será Produção do trabalho (o relatório final da pesquisa) e terá os seguintes objetivos: elaboração do capítulos introdutórios; elaboração dos capítulos explicativos; elaboração do resumo geral; elaboração da proposta; elaboração do sumário; compilação do trabalho final e entrega do TCC.

Principais atividades das fases 1 e 2

- seleção preliminar e coleta de artigos, teses, monografias, livros, revistas
- leitura e produção de resumo, consolidações, participação de minicursos, palestras.
- Experimentar métodos e técnicas de Engenharia de Requisitos, visando obter uma visão crítica e comparativa dos mesmos, quando isso for importante para o objetivo da pesquisa;
- verificar o que há de comum entre grafos de requisitos não funcionais, o uso do método PERT/CPM , e possibilidade de uso
- Verificar compatibilidades entre A simulação de Monte Carlo e Processos de desenvolvimento de software;
- realizar listagem de lacunas , possibilidades de aperfeiçoamento ou problemas nas áreas de requisitos e arquitetura

6 Cronograma de Atividades de Pesquisa

[illegible]

MORAES, Renato Oliveira; LAURINDO, Fernando José Barbin. **O método de Monte Carlo com MS Excel.** ## Descobrir o ano

LOPES, L.T., MADJENBAUM, A , AUDY, J.L.N. Uma proposta para processo de requisitos em ambientes de desenvolvimento distribuído de software. ## Descobrir o ano

VAROTO, Ane Cristina. **Visões em arquitetura de Software.** USP, 2002. 108p. Dissertação de Mestrado – Instituto de Matemática e Estatística, Universidade de São Paulo.

GERMOGLIO, Guilherme. **Arquitetura de Software.** XXX, ano. 239p. Livro Rice University, Houston, Texas. ## Descobrir o ano

CLEMENTS, Paul, et al. **Documenting Software Architectures.** 2ed, Addison Wesley, 2002.

GUSMÃO,Cristine, et al. Mapas Conceituais: Utilizando-os na Identificação de Riscos de Requisitos. Engenharia de Software Maganize,Brasil, ed 17, ano 2, p.14-21, 2008

LEITE J.C.S.P. , “Engenharia de Requisitos- Notas de Aula”, 1994.

ASTUDILLO , H., O Arquiteto de Software- Apresentação, 2002.

HAZAN, C , Leite, J. C. S. P, “Indicadores Para Gerência de Requisitos”, ## Descobrir o ano

RAMIRES, J.J.C.V, Negociação de Requisitos no Processo de Desenvolvimento de Software,2004, p.199, Dissertação de Mestrado - Departamento de Informática- Faculdade de Ciências da Universidade de Lisboa.

SAYÃO, M., LEITE, J.C.S.P., Rastreabilidade de Requisitos, vol.12, n1,p.30, 2005;

ALVES,F, et al. Ideal Day e Priorização. Engenharia de Software Maganize,Brasil, ed 7, ano 1, p.8-13, 2008

VASQUEZ, C. E. , Estimativas de software- Fundamentos, Técnicas e Modelos. Engenharia de Software Maganize,Brasil, ed 11, ano 1, p.40-48, 2009