

PAULINO ALMEIDA CANTANHÊDE FILHO

UM MÉTODO PARA SELEÇÃO DE SOFTWARE NA ENGENHARIA DE REQUISITOS

Dissertação de Mestrado submetida à
Coordenação do curso de Pós-Graduação
em Engenharia de Eletricidade da UFMA
para obtenção do título de Mestre em
Ciência da Computação.

Orientador: Prof. Dr. Zair Abdelouahab.

**São Luís
2005**

UM MÉTODO PARA SELEÇÃO DE SOFTWARE NA ENGENHARIA DE REQUISITOS

PAULINO ALMEIDA CANTANHÊDE FILHO

DISSERTAÇÃO APROVADA EM 14/ 03 / 2005

Prof. Dr. Zair Abdelouahab

(Orientador)

Prof. Dr. José Neuman de Souza

(Membro da Banca Examinadora)

Prof. Dra. Maria Del Rosario Girardi

(Membro da Banca Examinadora)

UM MÉTODO PARA SELEÇÃO DE SOFTWARE NA ENGENHARIA DE REQUISITOS

MESTRADO

Área de Concentração: CIÊNCIA DA COMPUTAÇÃO

PAULINO ALMEIDA CANTANHÊDE FILHO

Orientador: Dr. Zair Abdelouahab

Curso de Pós-Graduação
Em Engenharia de Eletricidade da
Universidade Federal do Maranhão

A Deus, ser maior e supremo, razão da
minha existência.

A meus pais, pelo incentivo e carinho
constantes.

À minha amada esposa, França, pela
compreensão e orações.

Às minhas filhas, Sarah e Ludymila, pela
inspiração.

AGRADECIMENTOS

Ao Prof. Dr. Zair Abdelouahab, meu orientador e amigo, pela paciência, ensinamentos e segura orientação durante o desenvolvimento da pesquisa.

A todos os meus familiares, pelo apoio e incentivo.

Aos meus amigos, Jean, Adroaldo, Silas, Werbeth, Jucinei, Gleisse e Alessandra pelas palavras de estímulo.

Aos meus companheiros de curso, em especial, Edil James, Geovane Bezerra, Ricardo Almeida, Marcelo Almeida, Samyr Beliche, Ozeas Lobato, pelos momentos de estudo, angústia, nervosismo, alegria, descontração e, principalmente, muita amizade no decorrer do curso.

A Profa.Dra.Maria da Guia pelas cobranças que se tornaram como estímulo para a conclusão do trabalho.

A amiga Marise Hassin Rodrigues por ter “patrocinado” e acreditado na conclusão deste trabalho.

A Dra. Filomena Saads minha chefe e amiga pela compreensão nos momentos de ausência do trabalho.

A toda Coordenação do Mestrado, coordenador, funcionários e professores, pelos bons serviços oferecidos e que foram fundamentais para a conclusão do mestrado.

Enfim, a todas as pessoas que, direta ou indiretamente, contribuíram para a concretização deste sonho.

RESUMO

Este trabalho apresenta diretrizes para avaliação e seleção de produtos COTS (*Commercial-Off-The-Shelf*). Apresenta-se uma metodologia para o Desenvolvimento Baseado em COTS (DBC) na Engenharia de Requisitos, abordando-se as fases de avaliação e seleção do DBC. O método está dividido em três fases, que por sua vez estão divididas em atividades direcionadas por ações que visam à qualidade das informações para uma tomada de decisão de forma automatizada e orientada por critérios de avaliação definidos durante a Engenharia de Requisitos. Esta abordagem divide os requisitos em três níveis, são eles: Fundamental, Contratual e Desejável. Os requisitos funcionais e não-funcionais são tratados de forma igualitária.

Palavras-chave: Requisitos, COTS (*Commercial-Off-The-Shelf*), Desenvolvimento Baseado em COTS (DBC).

ABSTRACT

Guidelines for evaluation and selection of COTS (Commercial-Off-The-Shelf) products. Comes a methodology for the Development based on COTS (DBC) in the Requirements Engineering, being approached the evaluation phases and selection of DBC. The method is divided in three phases, that are divided in activities orienting by actions that seek to the quality of the information for a socket of decision in an automated and oriented way for defined evaluation criteria during the Requirements Engineering. This approach divide the requirements in three levels: Fundamental, Contractual and Desirable. The functional and no-functional requirements are treated in an equalitarian way.

Keywords: Requirements, COTS (Commercial-Off-The-Shelf), Development based on COTS (DBC).

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Descrição do Problema.....	15
1.2	Objetivo do Trabalho.....	16
1.3	Estrutura da Dissertação.....	17
2	ESTADO DA ARTE.....	19
2.1	Introdução.....	20
2.2	Desenvolvimento Baseado em COTS.....	21
2.2.1	Características do Desenvolvimento Baseado em COTS.....	21
2.2.2	Avaliação de COTS	23
2.2.3	Seleção de COTS	24
2.2.4	Adaptação de COTS.....	25
2.2.5	Integração de COTS.....	25
2.2.6	Atualização	26
2.3	Engenharia de Requisitos	26
2.3.1	Fases da Engenharia de Requisitos	29
2.3.2	Engenharia de Requisitos para Desenvolvimento Baseado em COTS.....	35
2.4	Tomada de Decisão para Selecionar COTS	37
2.4.1	Multi-Attribute Utility Theory – MAUT	38
2.4.2	Weighted Score Method – WSM.....	39
2.4.3	Analytical Hierarchy Process – AHP	40
2.5	Alguns Métodos de Seleção Proposto no DBC.....	41
2.5.1	Método OTSO.....	41

2.5.2	Método lusWare.....	42
2.5.3	Framework STACE.....	43
2.5.4	Método CRE.....	44
2.5.5	Método PORE.....	45
2.6	Comentários.....	46
3	O MÉTODO PROPOSTO.....	47
3.1	Descrição Geral da Abordagem.....	48
3.2	As Fases do Método ASCRE.....	48
3.2.1	Preparação.....	50
3.2.2	Checagem.....	57
3.2.3	Decisão.....	60
3.3	Comentários.....	66
4	IMPLEMENTAÇÃO.....	68
4.1	Protótipo.....	69
4.2	Comentários.....	83
5	ESTUDO DE CASO.....	85
5.1	Objetivos.....	86
5.1.1	Preparação.....	86
5.1.2	Checagem.....	91
5.1.3	Decisão.....	93
5.2	Comentários.....	99
6	CONCLUSÃO.....	100
6.1	Contribuição do Trabalho.....	101
6.2	Trabalhos Futuros.....	102
	GLOSSÁRIO.....	104

REFERÊNCIAS 105

LISTA DE FIGURAS

FIGURA 2.1 DIMENSÕES DA ELICITAÇÃO DE REQUISITOS	30
FIGURA 3.1 VISÃO GERAL DA METODOLOGIA ASCRE	49
FIGURA 3.2 EXEMPLO DO DOCUMENTO DE VISÃO	51
FIGURA 3.3 RELACIONAMENTO DOS REQUISITOS.....	53
FIGURA 3.4 EXEMPLO BASE FORNECEDOR	56
FIGURA 3.5 RELACIONAMENTO DOS REQUISITOS X FORNECEDOR/PRODUTO	58
FIGURA 3.6 EXEMPLO DE VALORES MÁXIMOS POSSÍVEIS PARA UM PROJETO	64
FIGURA 3.7 EXEMPLO DE DEMONSTRAÇÃO DO RESULTADO	65
FIGURA 4.1 CADASTRO DOS STAKEHOLDERS	70
FIGURA 4.2 CADASTRO DO DOMÍNIO	70
FIGURA 4.3 CADASTRO DO SUBDOMÍNIO	71
FIGURA 4.4 CADASTRO DA APLICAÇÃO	71
FIGURA 4.5 CADASTRO DE REQUISITOS	72
FIGURA 4.6 RELAÇÃO REQUISITO X DOMÍNIO	73
FIGURA 4.7 RELAÇÃO REQUISITO X SUBDOMÍNIO	73
FIGURA 4.8 RELAÇÃO REQUISITO X APLICAÇÃO.....	74
FIGURA 4.9 PARÂMETROS CRÍTICOS DE ELIMINAÇÃO DE CANDIDATOS	74
FIGURA 4.10 CRITÉRIO DE AVALIAÇÃO DO FORNECEDOR	75
FIGURA 4.11 CRITÉRIO DE AVALIAÇÃO DO COTS PELO SEU CLIENTE	76
FIGURA 4.12 TABELA DO TIPO DE SERVIÇO	77
FIGURA 4.13 CRITÉRIO DE AVALIAÇÃO DO REQUISITO.....	77
FIGURA 4.14 CADASTRO DE EMPRESA	78
FIGURA 4.15 CADASTRO DE PRODUTOS CANDIDATOS	79
FIGURA 4.16 CLIENTE AVALIANDO O FORNECEDOR DO COTS	80

FIGURA 4.17 CLIENTE AVALIANDO O COTS CANDIDATO	80
FIGURA 4.18 NEGOCIAÇÃO DO REQUISITO COM O FORNECEDOR	81
FIGURA 4.19 NEGOCIAÇÃO DO REQUISITO COM O COTS.....	82
FIGURA 4.20 VALORES MÁXIMOS POSSÍVEIS PARA UM COTS	83
FIGURA 5.1 DIAGRAMA DO DOCUMENTO DE VISÃO.....	89
FIGURA 5.2 RESULTADO DO CADASTRAMENTO DO DOCUMENTO DE VISÃO	90
FIGURA 5.3 COTS E OS REQUISITOS DA ERS	92
FIGURA 5.4 RESULTADO FINAL DA SELEÇÃO DE PRODUTOS.....	99

LISTA DE TABELAS

TABELA 3.1 COMPARATIVO COM OUTRAS ABORDAGENS	67
TABELA 5.1 CRITÉRIO DE AVALIAÇÃO.....	91
TABELA 5.2 QUANTIDADE DE REQUISITOS POR ÍTEM.....	92
TABELA 5.3 VALOR MÁXIMO PARA PONTUAÇÃO EM CADA ITEM.....	93
TABELA 5.4 PONTUAÇÃO OBTIDA PELOS CANDIDATOS EM CADA ITEM.....	94
TABELA 5.5 VALOR MÁXIMO PARA SELEÇÃO POR ITEM	95
TABELA 5.6 PONTUAÇÃO DOS CANDIDATOS PARA SELEÇÃO EM CADA ITEM	96
TABELA 5.7 CLASSIFICAÇÃO GERAL DO DOMÍNIO SEGURANÇA DE REDE.....	97
TABELA 5.8 CLASSIFICAÇÃO PARA O SUBDOMÍNIO FIREWALL	98
TABELA 5.9 CLASSIFICAÇÃO PARA O SUBDOMÍNIO INSPEÇÃO DE CONTEÚDO	98
TABELA 5.10 CLASSIFICAÇÃO PARA O SUBDOMÍNIO DETECTOR DE INTRUSÃO	98

Capítulo 1

INTRODUÇÃO

O presente capítulo mostra a descrição do problema que motivou a realização deste trabalho. Em seguida, apresenta o objetivo da abordagem e finalizando, a estrutura do trabalho.

1.1 Descrição do Problema

O processo de elaboração de um sistema baseado em computador é denominado de Engenharia de Software e é compreendido pelas fases de Engenharia de Requisitos, Análise, Projeto e Implementação.

Este trabalho apresenta uma pesquisa na área de Engenharia de Software com ênfase na Engenharia de Requisitos. Atualmente, essa primeira fase da Engenharia de Software não trata a especificação de um problema visando apenas a construção de produto computacional, mas também para a compra de software que venham solucionar, total ou parcialmente, problemas comuns. Esses softwares existentes no mercado são chamados de COTS (*Commercial-Off-The-Shelf*) [1]. Exemplos típicos de produtos COTS são sistemas de antivírus, tais como: Norton, Etrust.

A parte da Engenharia de Software que utiliza os COTS é denominada de Desenvolvimento Baseado em COTS (DBC). Dentre as fases do DBC, estão as fases de avaliação e seleção de produtos candidatos, que se constituem em atividades críticas, visto que é nesse momento onde serão analisadas as características de cada COTS candidato para em seguida escolher um ou mais produtos que se mostram adequados. Se um produto for selecionado de forma inadequada irá comprometer o projeto de DBC. O sucesso de sistemas baseado em COTS depende largamente de um preciso entendimento das capacidades e limitações de cada produto [1]. A fim de atingir esse sucesso, a avaliação dos produtos deve ser um processo simultâneo com a aquisição de requisitos do usuário [2].

A comunidade de Engenharia de Software concorda que os erros ocasionados durante a fase de Engenharia de Requisitos estão entre os mais caros para corrigir depois que o sistema está funcionando [16]. Analisando esses sintomas em relação a sistemas baseados em COTS, observa-se que as conseqüências são muito mais serias [40]. Sendo a redução de custos e tempo de desenvolvimento algumas das promessas do DBC, esforços tem sido feito para melhorar o ciclo de vida DBC. Alguns desses esforços se concentram na Engenharia de Requisitos, que passa a ter um papel especial na avaliação e seleção dos COTS.

Como será mostrado neste trabalho a avaliação e seleção de COTS que realmente atendam os requisitos dos *stakeholders* pode contribuir bastante para a redução de custos e tempo no processo DBC.

1.2 Objetivo do Trabalho

O principal objetivo desta dissertação é fornecer um método para assistir os processos de avaliação e seleção de COTS dentro da Engenharia de Requisitos. Esse método enfatizará a automação do processo de tomada de decisão para minimizar o tempo e maximizar a precisão da avaliação/seleção, visando a possibilidade da escolha de múltiplos COTS. Para alcançar esse objetivo este estudo utilizou as vantagens de métodos existentes na literatura e criou mecanismos para suprir suas deficiências.

Dentre as vantagens desses métodos estão:

- ❖ A identificação de produtos;
- ❖ A aquisição de requisitos;
- ❖ Alguns aspectos da avaliação e seleção que proporcionam uma análise de decisão.

Dentre as desvantagens desses métodos estão:

- ❖ Os métodos discriminam entre requisitos funcionais e não-funcionais, priorizando um e desprezando o outro;
- ❖ Complexidade para aplicar as metodologias;
- ❖ Falta de orientação para automação dos procedimentos, que é fundamental para a redução do tempo e custos;
- ❖ A possibilidade de selecionar múltiplos COTS na tomada de decisão.

Nesta abordagem uma seqüência de passos é proposta objetivando a formação de bases de dados que conterá informações suficientes para uma tomada de decisão. Contempla,

também, orientações para a equipe de seleção definir critérios de avaliação para os candidatos de forma justa e criteriosa, objetivando precisão na escolha.

1.3 Estrutura da Dissertação

Esta dissertação é dividida em seis capítulos. Com exceção deste capítulo, são eles:

❖ Capítulo 2 – Estado da Arte

Este capítulo descreve a situação atual do desenvolvimento baseado em COTS (DBC). Em seguida, fornece uma visão geral das fases da Engenharia de Requisitos tradicional e aborda a Engenharia de Requisito para DBC. Depois examina algumas técnicas para tomada de decisão que podem ajudar a definir o critério de avaliação. Para finalizar, serão mostradas outras abordagens para seleção de COTS e são feitos alguns comentários.

❖ Capítulo 3 – O Método Proposto

Este capítulo descreve a principal contribuição deste trabalho, o método ASCRE (Automação da Seleção de COTS na Engenharia de Requisitos). Será mostrada individualmente cada uma das três fases que compõem o método, são elas: Preparação, Checagem e Decisão.

❖ Capítulo 4 – Implementação

Este capítulo descreve uma implementação da base de dados e alguns algoritmos para automação dos resultados sugeridos pela metodologia ASCRE.

❖ Capítulo 5 – Estudo de Caso

Este capítulo fornece um exemplo fictício utilizando um problema de gerenciamento de rede de uma organização, para demonstrar a eficiência do ASCRE no processo de tomada de decisão, considerando múltiplas escolhas para a solução do domínio.

❖ Capítulo 6 – Conclusão

Este capítulo mostra as principais contribuições desta dissertação para o desenvolvimento baseado em COTS e sugere algumas pesquisas que poderão ser realizadas a partir deste trabalho.

Capítulo 2

ESTADO DA ARTE

Este capítulo descreve a situação atual do desenvolvimento baseado em COTS (DBC). Em seguida, fornece uma visão geral das fases da Engenharia de Requisito tradicional e aborda a mesma para o DBC. A seção seguinte examina algumas técnicas para tomada de decisão que podem ajudar na definição do critério de avaliação. Logo depois, serão mostradas outras abordagens para seleção de COTS. No final são feitos alguns comentários.

2.1 Introdução

A Engenharia de Software é uma atividade que vem crescendo a cada ano. Seu objetivo principal é a construção de produtos de software que atenda as necessidades dos *stakeholders*. Essas necessidades aumentam na mesma medida das inovações tecnológicas, surgindo então uma competição mercadológica onde a confiabilidade, estabilidade, tempo e custo são alguns dos requisitos essenciais para a sobrevivência das indústrias de software.

Engenharia de Requisitos, Projeto e Implementação são etapas da Engenharia de Software que irão finalizar com o produto desejado. Entretanto, esses produtos muitas vezes não atendem as necessidades dos clientes, causando problemas no orçamento por causa da manutenção e conseqüentemente influenciando no tempo. Para tentar resolver esses problemas, tem-se investido em pesquisas de métodos, técnicas e ferramentas que auxiliem a Engenharia de Software.

Alguns anos atrás, uma empresa para possuir um software que atendesse suas necessidades, precisaria pagar para desenvolverem tal sistema. A evolução da Tecnologia da Informação proporcionou ao mercado condições de desenvolver software para atender muitas dessas necessidades. Organizações como a *National Aeronautics and Space Administration* (NASA) e o Departamento de Defesa dos Estados Unidos, declararam ter obtido sucesso com esses softwares [2]. Com isso, surgiu a necessidade de metodologias que avaliassem, selecionassem, adaptassem e integrassem esses produtos dentro de um projeto. Surgindo assim um novo paradigma da Engenharia de Software, o Desenvolvimento Baseado em COTS (DBC), que não precisa passar por todas as fases do paradigma tradicional, pois o software já está, praticamente, pronto e não precisa passar pela fase de Projeto e Implementação. Todavia, a fase de Engenharia de Requisitos ainda se torna necessária, também, neste paradigma, porque é através dos requisitos que se faz a avaliação e seleção dos produtos COTS.

A próxima seção mostra a situação atual do DBC. A Engenharia de Requisitos, que é uma etapa essencial no processo DBC, será mostrada na seção seguinte. Em seguida, algumas técnicas para tomada de decisão, que podem auxiliar na seleção de COTS, serão mostradas. Logo depois, serão mostradas outras abordagens para seleção de COTS. No final são feitos alguns comentários.

2.2 Desenvolvimento Baseado em COTS

O reuso de componentes tem sido proposto pela comunidade da Engenharia de Software como uma alternativa para se desenvolver produtos confiáveis e estáveis, num curto intervalo de tempo e com custo reduzido. O uso dessa alternativa é baseado numa longa história de trabalho em sistemas modulares, projeto estruturado e mais recentemente em sistemas orientados a objetos.

O aumento da competição para prover soluções de software com maior qualidade e desenvolvidas rapidamente e o uso dessas soluções em setores maduros como as indústrias de telefonia, aviação e eletrônica tem conduzido um crescente interesse no desenvolvimento de sistemas baseados em COTS.

Outras características que também colaboraram para o aceite dessa nova tendência de desenvolvimento de software são:

- ❖ Dificuldade de desenvolvimento de software complexo e extenso;
- ❖ Disponibilidade de COTS alternativos de domínio específico ou genérico;
- ❖ A maturidade dos componentes COTS;
- ❖ A interoperabilidade dos COTS dentro do padrão de mercado.

A filosofia COTS se concentra na produção de componentes competitivos e fáceis de integrar, muitas vezes usando componentes de outros fornecedores. O aumento considerável no número de organizações que adotam uso de COTS fortalece a credibilidade na obtenção de benefícios com essa nova forma de desenvolvimento. Governo, indústria, comércio e instituições de pesquisa tem demonstrado interesse pelo tema [1].

2.2.1 Características do Desenvolvimento Baseado em COTS

Os termos componentes e COTS criaram uma confusão conceitual, pois não existe uma única definição para cada um deles. Algumas definições para componentes aceitas pela comunidade CBSE (*Component-Based Software Engineering*) são citadas em [22]:

- ❖ Um componente como um elemento arquitetural – este componente deve prover e estar em conformidade com um conjunto de interfaces;
- ❖ Um componente como um elemento implementacional – acessado através de interfaces bem documentadas que podem ser descobertas em tempo de execução;
- ❖ Um componente de negócio – representa a implementação de um conceito autônomo de negocio ou processo.
- ❖ Componentes são artefatos claramente identificáveis, que descrevem ou realizam uma função específica e tem interfaces claras em conformidade com um dado modelo de arquitetura de software, documentação apropriada e um grau de reutilização definido.

Na literatura, também, existem várias definições para o termo COTS. Esta abordagem adota a definição interpretada em [23]: “Software que existe, está disponível para o público, pode ser comprado ou licenciado e que é suportado e atualizado pelo vendedor que retém o direito de propriedade”. Para o melhor entendimento deste trabalho, será usado este conceito tanto para COTS quanto para componente, pois ambos podem ser selecionados no processo DBC.

Existem dois tipos de sistemas baseados em COTS [24]. O primeiro e mais simples é chamado de CSS (*COTS Solution System*), onde somente um produto é usado para prover a solução. Nesse tipo de sistema, a organização deve se adaptar ao produto e toda infra-estrutura e manutenção é provida pelo vendedor. O segundo tipo de sistema baseado em COTS é chamado de CIS (*COTS Intricate System*). Neste caso múltiplos produtos são integrados para prover a funcionalidade do sistema. Eles permitem maior flexibilidade aos processos de mudança organizacional pois a própria organização é responsável pela infra-estrutura e manutenção. Todavia, os processos de seleção, integração e manutenção desses sistemas são muito mais complexos.

A integração de COTS faz com que a definição de uma arquitetura consistente, extensiva e robusta se torne muito complexa no DBC. Dessa forma, a seleção de diferentes pacotes de software tem quase sempre um grande impacto na arquitetura global do sistema. Logo, os engenheiros de software têm que prever alguns requisitos arquiteturais, como:

- ❖ Natureza do componente, incluindo suporte de infra-estrutura que o componente pode necessitar, modelo de controle de sincronização e modelo de dados;
- ❖ Natureza dos conectores, onde problemas podem ser encontrados nos protocolos utilizados e tipos de dados a serem transmitidos pelos conectores;
- ❖ Estrutura da arquitetura, que pode apresentar problemas na forma que um componente prever que a arquitetura se comporta em termos das iterações entre os componentes;

Outra característica diferenciada do DBC é fato do COTS ser considerado como uma caixa preta, pois na maioria dos casos os fornecedores proíbem a análise do código fonte. A falta desse código pode dificultar a avaliação da vulnerabilidade desses componentes e conseqüentemente a avaliação da confiabilidade do produto final. Nesse caso, a interface do componente assume uma importância fundamental, porque através dela é que se conhece a operação e o comportamento do COTS. De posse desse conhecimento é que se pode entender as funcionalidades e restrições desse componente para depois fazer a composição e customização. Um componente pode implementar uma ou mais interfaces e eles também podem usar interfaces provenientes de outros componentes. Para se construir um sistema composto de componentes, é necessário definir um mapeamento através da conexão compatível entre as interfaces desses componentes [25].

A maioria das metodologias de desenvolvimento tem o seu foco voltado para a implementação das funcionalidades do sistema. Entretanto, o desenvolvimento baseado em COTS procura adquirir produtos que tenham a maior parte dessas funcionalidades implementadas. Dessa forma, para o ciclo de vida DBC as fases de avaliação, seleção, adaptação, integração e atualização são consideradas como essenciais [26].

2.2.2 Avaliação de COTS

A fase de avaliação é composta das seguintes atividades:

- ❖ Elicitação de requisitos;
- ❖ Identificação dos COTS;

❖ Avaliação dos COTS em relação aos requisitos elicitados.

Ao longo dessa fase, as propriedades de cada produto devem ser identificadas e analisadas. Tais propriedades incluem uma descrição das funcionalidades e restrições dos possíveis candidatos. Outro aspecto importante é obter uma descrição do comportamento de cada produto no que diz respeito à sua interação com o restante do sistema. Esse tipo de informação diminui o risco de possíveis erros e conseqüentemente torna o processo de avaliação menos ad hoc [27]. Os produtos identificados não devem possuir dependência em relação a outros produtos que não são de interesse da empresa. Logo, esta fase ao mesmo tempo em que identifica, também elimina produtos candidatos, pois nenhum produto pode violar requisito essencial para o projeto.

Segundo Wallnau referenciado em [2], existem duas visões de avaliar COTS, a primeira e mais simples é chamada superficial. Esse tipo de avaliação consiste em adquirir os produtos que pelo menos aparentemente são adequados para uso. Essa abordagem garante custo mínimo durante o esforço de avaliação pois os produtos que falham durante um estágio particular do processo de avaliação são logo eliminados e o primeiro produto que passar por todos os estágios é escolhido para posterior integração. É importante lembrar que os estágios do processo de avaliação são definidos, na maioria das vezes, por cada organização e normalmente não existe um processo padrão. A outra visão de avaliação é chamada de exhaustiva. Nessa abordagem todos os candidatos são avaliados detalhadamente ao longo de todos os estágios do processo de avaliação. O objetivo da avaliação exhaustiva é garantir que um ou mais produtos apropriados e de garantida qualidade são selecionados e posteriormente integrados ao sistema.

2.2.3 Seleção de COTS

O objetivo dessa fase é selecionar um ou mais candidatos que vieram da fase de avaliação e que posteriormente deverão passar para a fase de adaptação do sistema. Normalmente, a seleção de um produto requer uma análise de prioridades e negociação. Para tanto, precisa-se estabelecer critérios que orientem a equipe de seleção na hora de negociar os requisitos com os fornecedores e, não somente requisitos dos *stakeholders*, mas avaliar as condições do fornecedor, como: reputação; maturidade; Garantias de suporte; Estabilidade. O

histórico do produto com outros clientes que tenham objetivos semelhantes ao projeto que está sendo desenvolvido, devem ser levados em consideração na hora de discriminar entre dois produtos.

A avaliação e seleção devem ocorrer paralelamente, como uma única fase. Isso se deve a interatividade e iteratividade que acontece nos processos de identificação, avaliação, priorização e negociação durante o DBC.

2.2.4 Adaptação de COTS

Os produtos COTS por serem genéricos, ou seja, criados para atender clientes que atuam em um determinado domínio, geram, em muitos casos, a necessidade de adaptação a um projeto específico.

Esta fase é para providenciar todo recurso para adaptação e complementos necessários que facilitem a integração do COTS ao ambiente que ele irá exercer sua atividade. Essa adaptação deve ser baseada em regras que buscam eliminar conflitos dentro do ambiente operacional. Uma técnica bastante simples é o uso de scripts, que são descritos como uma proteção entre as entradas e saídas [30]. Entretanto, existem várias outras estratégias mais sofisticadas para fazer a adaptação de COTS, como por exemplo: capas de proteção ou *wrappers*, análise de atenuação de riscos e injeção de falhas.

Uma forma de amenizar os problemas de adaptação é criar uma camada de software no COTS para limitar suas funcionalidades. Essa capa de proteção, chamada Técnica *Wrapper*, é implementada testando as entradas antes de fazer uma chamada ao sistema.

2.2.5 Integração de COTS

A fase de integração consiste em incorporar os COTS selecionados em um único sistema. Podendo ser desenvolvido processos que não foram contemplados pelos COTS adquiridos e que são necessários para atender o objetivo do sistema. Quanto maior for o sistema, maior poderá ser sua complexidade. Com isso, sistemas integrados com muitos

COTS podem ter dificuldades nesta fase, porque durante a integração é que se verificam os possíveis problemas que não foram detectados nas fases anteriores. Sendo assim, realizar uma estimativa consistente de prazos e recursos necessários pode ser uma tarefa muito difícil.

2.2.6 Atualização

As mudanças na tecnologia e nas regras de negócio proporcionam sucessivas atualizações nos sistemas de computador. Logo, os sistemas baseados em COTS também são afetados com tais mudanças. Assim, a fase de atualização tem como objetivo identificar e planejar as atualizações do sistema integrado por COTS.

A atualização de sistemas baseados em COTS está diretamente ligada com a aquisição de novos lançamentos e versões dos produtos. Logo, um sistema que possui inúmeros COTS está fortemente dependente das futuras modificações e atualizações realizadas pelos vendedores desses produtos. Normalmente, é necessário revalidar licenças e contratos com os vendedores [31].

2.3 Engenharia de Requisitos

A etapa de Engenharia de Requisitos se constitui na fase crítica do ciclo de vida de um sistema, pois uma compreensão completa dos requisitos de software é fundamental para um bem-sucedido desenvolvimento do mesmo [12]. A base do Software é seus requisitos, por isso é necessário que o produto final satisfaça todas as necessidades possíveis dos *stakeholders*, se não toda a estrutura poderá ficar comprometida. Quando mais cedo se detecta problemas de requisitos no ciclo de desenvolvimento de um software, mais fácil e menos dispendioso é a sua correção [11].

Para melhor entendimento da Engenharia de Requisitos é necessário definir o que significa requisito. Segundo o IEEE (Instituto de Engenharia Elétrica e Eletrônica) um requisito é definido como[22]:

- ❖ Uma condição ou capacidade necessitada por um usuário para resolver um problema ou atingir um objetivo;
- ❖ Uma condição ou capacidade que deve ser atingida ou possuída por um sistema ou componente de sistema para satisfazer um contrato, padrão, especificação, ou outro documento de formalidade;
- ❖ Uma representação documentada de uma condição ou capacidade como nos dois conceitos anteriores.

Em suma, requisito é a condição necessária à obtenção de alguma coisa ou para atingir certo fim, uma exigência legal.

Tendo-se o entendimento de requisito, pode-se agora definir Engenharia de Requisitos. Existem na literatura diversos conceitos para Engenharia de Requisitos, como:

- “Engenharia de Requisitos é a aplicação disciplinada de demonstrar princípios, métodos, ferramentas e notações para descrever o comportamento pretendido de um sistema e suas restrições associadas” [13];
- “Engenharia de Requisitos é o ramo da Engenharia de Software que se interessa por metas do mundo real com respeito a funções e restrições nos sistemas de software. Preocupa-se também com a relação desses fatores com especificações precisas do comportamento do software, e a sua evolução com o passar do tempo através de famílias de software” [14];
- “A Engenharia de Requisitos consiste de todas as atividades envolvidas no descobrimento, documentação e gerenciamento de um conjunto de requisitos para um sistema baseado em computador” [15].

Segundo o IEEE, referenciado em Alves [2], Engenharia de Requisitos corresponde ao processo de aquisição, representação, refinamento e verificação das necessidades do cliente para um sistema de software, objetivando-se ter uma especificação completa e correta dos requisitos de software. Os conceitos refletem o mesmo significado e

tem como essência o conhecimento do que é desejado por todos os interessados no produto final. Os requisitos devidamente especificados e modelados servem de base para a análise.

Antes de se abordar o processo de Engenharia de Requisitos em maiores detalhes, serão vistos alguns conceitos fundamentais para o melhor entendimento.

® *Stakeholders*

Define-se *stakeholders*, referenciado em [9], como:

- “Qualquer um que é afetado de alguma forma pelo sistema”;
- “Qualquer um que requer interesse na empresa ou sistema”;
- “São pessoas que irão ser afetadas pelo sistema e que tem influência direta ou indireta nos requisitos do sistema”;
- “São indivíduos ou organizações que estão prestes a ganhar ou perder com o sucesso ou fracasso de um sistema”.

Sendo assim, quem paga pelo sistema, quem projeta o sistema e quem utiliza o sistema são considerados *Stakeholders* e a interação entre eles é de fundamental importância para o sucesso da Engenharia de Requisitos. Por esse motivo esta abordagem considera a identificação dos *stakeholders* como a primeira ação da fase da Engenharia de Requisitos.

® *Domínio*

É o nível mais genérico do problema a ser abordado. Refere-se a área na qual o sistema será aplicado, descrito de forma abrangente. Por exemplo: área médica; bancária; recursos humanos; etc. Contudo, o domínio pode ser visto de outras formas, como: o domínio do software, que sustenta uma solução para o problema; o domínio da aplicação, onde o problema está situado. O domínio da aplicação pode ser visto como o que é dado e o domínio do software como o que será construído [17].

® *Documento de Requisitos*

É a declaração formal dos requisitos de um sistema para clientes, usuários finais e desenvolvedores de software [15, 16]. Também conhecido como ‘Documento de

Especificação de Software' [15], ele é a síntese da Engenharia de Requisitos, pois contém os requisitos funcionais e não-funcionais do sistema a ser desenvolvido, validado e acordados pelos *stakeholders*. Este documento deve ser consistente com os documentos existentes, correto e completo com relação a satisfação das necessidades, entendido pelos usuários, clientes, projetistas e provedores, e capaz de servir como base para o projeto [13]. Percebe-se então que um bom documento de requisitos deve ser claro, preciso e sem ambigüidades.

2.3.1 Fases da Engenharia de Requisitos

Engenharia de Requisitos pode ser descrita como um processo sistemático de cinco passos distintos: elicitaco de requisitos, anlise e negociao de requisitos, especificao de requisitos, validao de requisitos, e gerenciamento de requisitos. Estas atividades so intercaladas, interativas, iterativas e podem estender-se sobre todo o ciclo de vida da Engenharia de Software.

□ Elicitao de Requisitos

Esta atividade corresponde ao processo de descoberta das exigncias do sistema. Ela necessita de um entendimento harmnico entre engenheiros, clientes e usurios finais para encontrar informao sobre o problema a ser resolvido, os servios que o sistema deve contemplar, o ambiente que deve funcionar, restrio envolvidas, etc.

A identificao dos *stakeholders*  o primeiro passo neste processo, pois eles so importantes nesta atividade, visto que o produto final deve atend-los.

Sommerville, referenciado em [9], prope quatro dimenses para a elicitaco de requisitos representado pela figura 2.1.

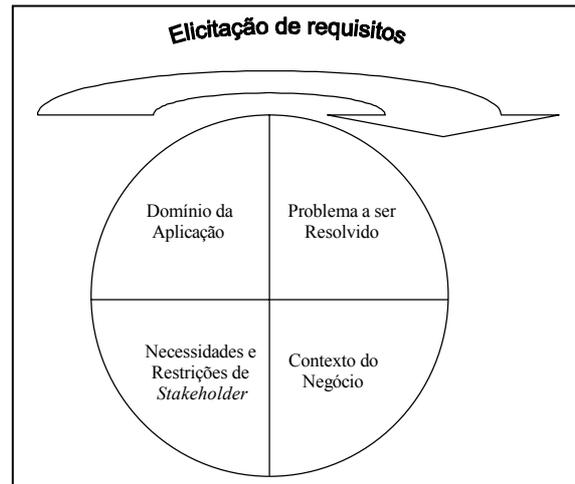


Figura 2.1 – Dimensões da Elicitação de Requisitos

Tais dimensões consistem em:

1. Entendimento do domínio da aplicação: trata-se do conhecimento das áreas gerais onde o sistema é aplicado;
2. Entendimento do problema a ser resolvido: são os detalhes do problema especificado pelo cliente onde o sistema será aplicado que devem ser entendidos;
3. Entendimento do negócio: são as regras que regem a organização na qual o sistema deve contribuir;
4. Entendimento das necessidades e restrições dos *stakeholders* do sistema: são as especificidades de cada entidade (que podem ser clientes, usuários, etc) envolvida com o projeto.

Um dos principais problemas encontrados durante esta atividade é a dificuldade de entender as reais necessidades dos usuários [11]. Em boa parte dos casos, isso é devido à formação distinta entre analista e usuários, e, portanto eles têm ponto de vista diferentes sobre o mesmo problema. A experiência da equipe de Engenharia de Requisitos tem importância relevante para fornecer ao usuário os recursos necessários para que ele possa ter uma idéia precisa e explícita do domínio do problema e assim poder fornecer as informações necessárias do contexto do projeto.

Algumas das técnicas para elicitación de requisitos propostas na literatura são:

- ❖ Técnicas tradicionais – inclui o uso de questionários, entrevistas, análise da documentação existente. O questionário é uma técnica onde perguntas predefinidas são distribuídas para uma amostragem significativa e representativa de *stakeholders* e os resultados são avaliados. São úteis quando a quantidade de *stakeholders* é extremamente grande. Uma vez que todas as questões podem ser predeterminadas, é mais eficiente na avaliação de tendências de opiniões a respeito de requisitos específicos e bem definidos. A entrevista consiste no contato pessoal do engenheiro de requisitos e o *stakeholders*, onde perguntas e respostas devem levar ambos ao conhecimento mais profundo do problema em questão. As entrevistas são úteis quando os *stakeholders* possuem muitos conhecimentos subjetivos e estão dispostos a serem entrevistados. Para ser mais eficiente, o entrevistador deve ser experiente;
- ❖ Técnicas de elicitación em grupo – são técnicas de dinâmica de grupo com o objetivo de entender de forma mais detalhada as necessidades dos usuários, estão incluídas: *brainstorming*, sessões JAD e RAD que estão detalhadas em [14];
- ❖ Técnicas de modelagem – fornece um modelo específico das informações que serão adquiridas, e usa esse modelo para orientar o processo de aquisição. Uma técnica bastante utilizada é o uso de cenários para representar as tarefas que os usuários executam normalmente e aquelas que eles desejam executar. Inclui métodos baseados em metas, tais como: KAOS [18] e métodos baseados em cenários como CREWS [19];
- ❖ Técnicas Cognitivas – inclui uma série de técnicas originalmente desenvolvidas para aquisição de conhecimento para sistemas baseados em conhecimentos, alguns exemplos são: Análise de protocolo, *laddering*, *card sorting*, *repertory grids* [20];
- ❖ Técnicas Contextuais – surgiu como uma alternativa para as técnicas tradicionais e cognitivas, inclui técnicas de etnografia e análise social;
- ❖ Etnografia – observar potenciais usuários em seu ambiente natural. Resulta em uma percepção mais precisa do problema do que perguntar aos usuários o que eles fazem. São úteis para suporte à automação de uma função pouco ou não automatizada,

particularmente quando são disponíveis a observadores treinados sem noções preconcebidas do problema e de sua solução.

A escolha da técnica para adquirir requisitos ou a combinação entre elas, depende de questões específicas do projeto, tais como: tempo e recursos disponíveis; tipo de informação necessária.

□ **Análise e Negociação de Requisitos**

A atividade de análise e negociação tenta resolver os conflitos entre os requisitos sem necessariamente comprometer a satisfação dos objetivos de cada usuário. O engenheiro de sistemas deve harmonizar estes conflitos através de um processo de negociação. Os riscos associados a cada requisito são identificados e analisados. A análise deve verificar o esforço de desenvolvimento e o impacto no custo do projeto e no tempo de entrega. Para tanto, precisa-se usar uma abordagem interativa que provavelmente deve eliminar, combinar e/ou modificar os requisitos com o objetivo de obter o maior nível possível de satisfação dos *stakeholders*.

Em geral, os modelos de análise e negociação identificam as principais necessidades de cada usuário, ou seja, atribuem prioridades aos requisitos, em seguida analisam esses resultados para tentar garantir que os requisitos mais críticos sejam atendidos. Existem algumas técnicas para priorização e negociação de requisitos como AHP (ver seção 2.4.3) e QFD [21]. Assim sendo, problemas podem ser reconhecidos, discutidos e resolvidos para proporcionar no final a documentação de requisito.

□ **Especificação de Requisitos**

Na Engenharia de Software o termo especificação pode significar um documento descrito, um modelo gráfico, um modelo matemático formal, uma coleção de cenários de uso, um protótipo, ou qualquer combinação dos itens citados.

Na Engenharia de Requisitos a especificação de requisitos deve fornecer como produto final informações para as engenharias de hardware, software e banco de dados. Essas informações devem descrever as interfaces com o usuário, entradas, saídas, funções, controles do sistema, manutenção, teste, desempenho requerido de um sistema baseado em computação e as regras que irão guiar seu desenvolvimento.

□ **Validação de Requisitos**

Esta atividade, embora semelhante a de análise e negociação, estrutura os documentos de forma que aspectos importantes não sejam esquecidos. Ela deve focalizar aspectos de qualidade do documento de requisito como um todo, bem como os relacionamentos entre os requisitos. Algumas questões podem ser relacionadas para checar se algum requisito está ausente; se os requisitos estão consistentes e não contraditórios; se o documento está estruturado de forma a facilitar o entendimento; se foi feito o rastreamento dos requisitos; ou se o documento está de acordo com os padrões estabelecidos.

O mecanismo primário de validação de requisitos é a revisão técnica formal. Essa revisão deve ser feita várias vezes durante a especificação dos requisitos, para que sejam validadas pequenas porções de especificação, garantindo que a atenção seja focada em aspectos específicos dos requisitos. Embora não exista uma regra padrão para validação de requisitos, aconselha-se utilizar check list para examinar cada requisito, como por exemplo:

- ❖ O requisito está estruturado claramente?
- ❖ A fonte (pessoa, documento, etc.) foi identificada? Essa fonte validou a estrutura final do requisito?
- ❖ Os requisitos relacionados a este estão claramente identificados? E estão todos?
- ❖ O requisito viola alguma regra de domínio?
- ❖ Os requisitos associados à performance, comportamento e características operacionais foram estruturados claramente? Que requisitos parecem implícitos?

□ **Gerenciamento de Requisitos**

É a fase que gerencia as mudanças sofridas pelos requisitos durante o desenvolvimento. Isto significa dizer que os requisitos devem continuar válidos até o final do ciclo de desenvolvimento. Se mudanças ocorrem na organização, no negócio ou no domínio da aplicação, serão refletidas nos requisitos e estes devem ser atualizados [9].

Segundo Pressman referenciado em [9], gerenciamento de requisitos corresponde ao conjunto de atividades que auxilia a equipe de projeto a identificar, controlar e rastrear os requisitos, bem como as alterações nos requisitos em muitos momentos do projeto.

É essencial que um requisito possua um identificador exclusivo para que seja facilitado o seu rastreamento. Existem algumas abordagens que tratam os problemas de identificação de requisitos. As técnicas mais comuns para identificar requisitos são:

- ❖ Baseada na numeração de capítulo, seção e unidade. Para se identificar o segundo requisito da sétima seção do capítulo três, deve-se codificar o requisito com o número 3.7.2;
- ❖ A identificação simbólica, onde os requisitos são associados ao assunto tratado e recebe um símbolo, como por exemplo, o primeiro requisito de performance ficaria PF-001.

As principais preocupações no gerenciamento de requisitos são:

- ❖ Gerenciar mudanças nos requisitos acordados;
- ❖ Gerenciar os relacionamentos entre os requisitos;
- ❖ Gerenciar as dependências entre o documento de requisitos e outros documentos produzidos ao longo do sistema e do processo de engenharia de software.

Ferramentas de gerenciamento de requisitos podem prover facilidades como:

- ❖ Um sistema de banco de dados para armazenamento de requisitos;
- ❖ Análise de documento e facilidades de geração para ajudar a construir um banco de dados de requisitos e auxiliar na criação dos documentos de requisitos;
- ❖ Facilidades de gerenciamento de mudanças que ajudam a garantir que as mudanças foram validadas e cotadas corretamente;
- ❖ Facilidade de rastreabilidade que auxiliam os engenheiros de requisitos a encontrar dependências entre requisitos.

2.3.2 Engenharia de Requisitos para Desenvolvimento Baseado em COTS

Em princípio a Engenharia de Requisitos tem suas fases iguais tanto para DBC quanto para o desenvolvimento tradicional. Todavia, existem algumas diferenças na forma e nas possibilidades de execução dos processos. Enquanto o desenvolvimento tradicional permite alterações de requisitos no decorrer da Engenharia de Software, o DBC não tem a mesma flexibilidade, visto que, os produtos adquiridos têm características de caixa-preta, ou seja, nem sempre é possível alterar os seus códigos fontes. Geralmente os vendedores fornecem uma boa descrição das funcionalidades dos produtos, mas isso não evita possíveis incompatibilidades entre requisitos dos *stakeholders* e dos produtos.

A informação adquirida na elicitação de requisito no DBC pode ter duas origens. A primeira é baseada no stakeholder, quando este inicialmente irá fornecer informações sobre as suas necessidades principais. Essas informações devem ser modeladas de forma que se identifique no mercado produtos com características semelhantes. Essas características devem ser tanto funcionais como não-funcionais. Vale aqui destacar os requisitos contratuais, que são uma importante categoria de requisitos não-funcionais. Eles estão relacionados com os acordos e negociações que devem ser realizados entre os vendedores de COTS e a organização que está adquirindo esse produto. Kotonya classifica os requisitos contratuais como sendo não-funcionais externos [32].

A segunda origem é baseada no produto, quando a equipe de elicitação recolhe informações sobre o produto para poder negociar com os *stakeholders* sobre requisitos que, a priori, não foram previstos. O fornecedor, muitas das vezes, é uma importante fonte de informação sobre a regra de negócio. Por exemplo, uma empresa que queira adquirir um sistema de recursos humanos pode não conhecer todas as leis que regem esse setor, enquanto um fornecedor experiente na construção desse tipo de produto tem a obrigação de conhecer.

Normalmente, as técnicas para elicitação de requisitos para DBC são as mesmas usadas no desenvolvimento tradicional. Cada técnica é particularmente útil para uma etapa do processo de aquisição de requisitos. Em geral, deve-se iniciar o processo de busca a partir da descrição dos requisitos considerados críticos pelos *stakeholders*. Dependendo da complexidade do problema, nem todos os requisitos adquiridos durante o processo de elicitação poderão ser atendidos em um só COTS ou mesmo no sistema integrado. Sendo

assim, a fase de análise e negociação tem um papel fundamental para avaliar a importância de cada requisito e contribuir para que esses requisitos sejam priorizados de forma consistente com as reais necessidades dos *stakeholders*.

Quando se tem uma quantidade relevante de informações adquiridas na fase de elicitação, já é possível iniciar o processo de busca dos potenciais produtos candidatos. Durante o processo de localização, a análise dos requisitos deve ajudar a discriminar entre dois produtos semelhantes. É importante ressaltar que esta abordagem não faz diferença à nível de prioridade entre requisitos funcionais e os não-funcionais. Considera-se que a prioridade para o cliente são os requisitos fundamentais, que podem ser funcionais ou não-funcionais. Dessa forma, a avaliação entre produtos deve começar a partir da análise dos requisitos fundamentais que cada produto suporta.

No desenvolvimento convencional, a análise dos requisitos é geralmente direta pois os *stakeholders* descrevem o futuro sistema através de modelos que contêm um conjunto de atributos e condições específicas que o sistema deve suportar. Entretanto, a análise dos requisitos é diferente no desenvolvimento de sistemas baseados em COTS, pois a especificação deve ser flexível a ponto de aceitar algumas funcionalidades já presentes nos produtos e que não foram inicialmente descritas [31]. Assim a fase de análise para DBC deve suportar um extenso processo de modificação dos requisitos especificados.

Quando os produtos não atenderem todos os requisitos fundamentais, existem, basicamente, duas decisões a serem tomadas. A primeira sugere que a especificação dos requisitos sofra ligeiras alterações para suportar tais restrições. A segunda propõe que esses requisitos sejam implementados, em caso de requisitos funcionais, ou o ambiente seja adequado ao produto, em caso de requisitos não-funcionais. Sendo assim, deve ser realizado um extenso processo de priorização e negociação de tais requisitos em relação aos produtos disponíveis. Esta abordagem dispõe de um processo de negociação automático onde o critério de seleção deve nortear esse processo.

A documentação de requisitos é, também, uma etapa importante no DBC. A equipe de engenharia deve descrever os requisitos dos usuários num documento que deve conter uma especificação detalhada dos requisitos funcionais e não-funcionais, assim como mostrar as prioridades de cada requisito.

É importante que o documento de requisitos permita um efetivo gerenciamento dos requisitos, pois eles normalmente serão alterados ao longo do processo de seleção de COTS. Um dos principais motivos de mudança na especificação dos requisitos é devido a conflitos entre requisitos. Outro aspecto que precisa ser tratado na fase de documentação é o rastreamento dos requisitos. Uma vez que é fundamental documentar as decisões e modificações feitas nos requisitos durante todo o processo de desenvolvimento baseado em COTS. Em particular, é importante descrever os motivos pelos quais os requisitos tiveram que ser modificados. Além disso, o rastreamento de requisitos deve permitir que os sucessivos refinamentos dos requisitos fiquem disponíveis para auxiliar o processo de seleção de COTS. O processo de rastreamento garante a satisfação dos *stakeholders* disponibilizando um documento que mostra que todos os requisitos estão sendo atendidos [34]. No caso do desenvolvimento baseado em COTS, o rastreamento possibilita um entendimento dos motivos pelos quais os produtos foram priorizados e selecionados.

Outra etapa importante da Engenharia de Requisitos no DBC é a validação dos requisitos. Esta etapa precisa ser feita em paralelo com a atividade de avaliação dos produtos. Normalmente, a validação é diretamente afetada pela disponibilidade dos produtos existentes no mercado e que satisfazem os requisitos especificados. Uma possível atividade dessa etapa é aceitar as funcionalidades existentes nos produtos e que não foram especificadas pelos *stakeholders* ao longo das fases da Engenharia de Requisitos. Obter consenso entre os *stakeholders* em relação aos requisitos que tiveram que ser simplificados ou até mesmo excluídos devido à falta de produtos que atendessem tais requisitos é uma tarefa bastante difícil nesta etapa. Em casos que um número mínimo dos requisitos críticos não puderem ser satisfeitos, os *stakeholders* devem concordar que eles deverão ser implementados da forma tradicional.

2.4 Tomada de Decisão para Selecionar COTS

Uma das etapas crítica da maioria dos processos de seleção de COTS é o processo de tomada de decisão. Com o objetivo de selecionar ou recomendar um produto COTS adequado para integrar o sistema, as alternativas disponíveis precisam ser avaliadas de acordo com sua relativa importância para atender os requisitos dos usuários. Geralmente são usadas as técnicas de tomada de decisão para analisar essas alternativas e fazer um ranking das

melhores opções e assim contribuir para obtenção de um processo de seleção consistente. Em [35] é apresentado uma hierarquia de três níveis para o processo de tomada de decisão. No primeiro nível é apresentada a principal meta para o processo de tomada de decisão. No segundo nível são mostrados alguns critérios relevantes para selecionar o produto, estão entre eles: funcionalidades, garantias do vendedor, requisitos de qualidade como usabilidade, etc. No terceiro nível estão os produtos candidatos que serão avaliados a partir dos critérios definidos no nível 2 para atingir a meta principal. Uma vez que as alternativas disponíveis são avaliadas, uma abordagem comum é necessária para consolidar os resultados da avaliação e fazer o ranking das varias alternativas.

Um problema inerente a qualquer processo de tomada de decisão é a dificuldade para atribuir pesos e pontuação aos atributos que estão sendo examinados. Uma possível solução é formar um grupo de avaliadores que possua experiência em tarefas dessa natureza e já tenha utilizado as técnicas propostas. Outro aspecto importante é que o processo de tomada de decisão seja realizado pelas mesmas pessoas, assim os riscos de inconsistência entre os resultados podem ser reduzidos.

As principais técnicas para tomada de decisão possuem algumas limitações quando utilizadas para selecionar produtos COTS. Além da complexidade de uso, outro aspecto relevante é a falta de uma abordagem que trate o inter-relacionamento entre os atributos dos produtos. A maioria dos requisitos de software interage entre si, de forma que a avaliação de um produto em relação à satisfação de um requisito pode sofrer influencia de outros requisitos que pelo menos aparentemente, não possuam nenhuma ligação.

As técnicas descritas a seguir podem auxiliar na tomada de decisão para selecionar COTS.

2.4.1 Multi-Attribute Utility Theory – MAUT

Esta técnica se propõe auxiliar na escolha da melhor alternativa dentro de um conjunto de opções [36]. Ela negocia tomando como referência as preferências definidas pelos usuários, que pode ser valores de um atributo ou entre atributos.

Uma das vantagens da técnica MAUT é que ela força os usuários a considerarem todos os atributos referentes a uma decisão. Muitas vezes, durante o processo de tomada de decisão pode ocorrer que certos atributos importantes sejam esquecidos. Essa técnica auxilia os tomadores de decisão para criar uma lista detalhada de todos os atributos relevantes [37]. A principal fraqueza da técnica é que ela representa os atributos de forma independente e assim não trata a relação de dependência que ocorre entre muitos atributos.

2.4.2 Weighted Score Method – WSM

Essa técnica é uma das abordagens mais usadas em situações de tomada de decisão [36]. Ela faz um ranking das alternativas em função dos critérios preestabelecidos, onde é definida uma escala que representa a preferência relativa de cada atributo. Essa técnica funciona da seguinte forma:

- i. Vários critérios são descritos para avaliar cada alternativa;
- ii. Cada critério tem um peso associado que está diretamente relacionado com a sua importância;
- iii. As alternativas devem ser avaliadas segundo a observância de cada critério e depois é atribuído um escore numa escala pré-definida;
- iv. Finalmente, cada alternativa receberá uma pontuação que é calculada a partir da seguinte fórmula:

$$Pont_Produto_a = \sum_{j=1}^n (peso * escore_{aj})$$

Onde: *a* representa uma alternativa e *n* o número de critérios

O uso da técnica WSM é simples e apresenta resultados aparentemente lógicos. Todavia, existem diversos problemas com essa técnica que tornam questionável se WSM consegue representar a real preferência entre alternativas [1]. Como o método WSM produz números reais como resultado, tais resultados podem ser facilmente interpretados como se eles representasse a real diferença entre duas alternativas. De fato, os escores resultantes apenas representam o ranking relativo entre as alternativas e as diferenças entre os valores não

dá nenhuma indicação da sua relativa superioridade [6]. Outra restrição de WSM é que a atribuição dos pesos para cada critério pode ser bastante complicada quando o número de critérios é muito grande.

2.4.3 Analytical Hierarchy Process – AHP

A técnica AHP se baseia na decomposição hierárquica de um problema de tomada de decisão com múltiplos critérios [35]. Ela é suportada pelo *Expert Choice* [43], que é um software comercial que auxilia nos cálculos e julgamentos.

A AHP decompõe um problema numa hierarquia de critérios, onde em cada nível a importância relativa de cada fator é avaliada através de comparações. Os principais passos para aplicar a técnica são:

- i. Definir uma hierarquia de fatores que influenciam a decisão, isso resulta numa estrutura hierárquica de fatores que tem as alternativas como nos finais dessa hierarquia;
- ii. Definir a importância dos fatores em cada nível;
- iii. Definir as preferências entre as alternativas;
- iv. Verificar a consistência do ranking e revisar os resultados;
- v. Apresentar o resultado da avaliação, onde a alternativa com maior prioridade será recomendada como a melhor escolha.

O ranking obtido através da comparação aos pares entre as alternativas é convertido para um ranking normalizado usando vetores normalizados, ou seja, o ranking relativo entre as alternativas é representado em valores cuja soma total é um. A técnica evita comparar valores absolutos com a perspectiva de alcançar resultados mais confiáveis na medida que compara as alternativas e critérios aos pares.

Um das principais desvantagens da AHP é a complexidade dos cálculos para fazer as comparações aos pares e realizar as operações entre as matrizes de valores. Quando se trata

de avaliar COTS, essa técnica tem como desvantagem a falta de uma ligação de dependência entre atributos, que é muito comum encontrar na engenharia de requisitos.

2.5 Alguns Métodos de Seleção Proposto no DBC

Existem basicamente três estratégias para seleção de COTS: filtragem progressiva, identificação por características chave e montagem de quebra-cabeça [23]. Filtragem progressiva representa uma estratégia onde um produto é selecionado a partir de um grupo de potenciais candidatos, esse processo começa com uma grande variedade de candidatos e através de mecanismos de filtragem, os produtos considerados inadequados são eliminados progressivamente. Na identificação por características chave, uma característica particular é selecionada antes de iniciar o processo de seleção, essa característica pode ser um determinado vendedor ou uma tecnologia, por exemplo: o critério de seleção inicial pode ser procurar somente produto compatíveis com Linux. O modelo de montagem de quebra-cabeça inicia com a premissa de que uma solução COTS precisa se “encaixar” com os vários componentes do sistema como um quebra-cabeça. Essa abordagem aplica técnicas de prototipagem evolucionaria para construir inúmeras versões que são progressivamente próximas do sistema final. Existem vários trabalhos que propõem métodos para auxiliar o processo de seleção. A seguir, serão descritos alguns desses métodos.

2.5.1 Método OTSO

Uma das primeiras iniciativas para assistir e facilitar a seleção de COTS é o método OTSO (*Off-The-Shelf-Option*) [7]. Esse método suporta a localização, avaliação e seleção de componentes reusáveis, assim como fornece técnicas específicas para definir o critério de avaliação, comparando custos e benefícios das alternativas. Esse método decompõe o critério de avaliação em quatro áreas principais [6]:

- ❖ Requisitos dos usuários;
- ❖ Características de qualidade do produto;

- ❖ Compatibilidade de domínio e arquitetura;
- ❖ Interesses estratégicos (custos, estabilidade do vendedor, etc.).

Uma das fases mais importantes do método é a análise dos resultados, essa fase faz uma estimativa dos custos para adquirir cada candidato COTS e compara custos e benefícios de cada alternativa. Além disso, usa a técnica AHP para consolidar os resultados da avaliação e suporta o processo de tomada de decisão.

Apesar do método OTSO enfatizar que um dos problemas chave do processo de seleção de COTS é a falta de atenção aos requisitos dos *stakeholders*, esse método não prove nenhuma sugestão ou solução para atacar esse problema. O principal foco do método é definir o critério de avaliação, porém ele não oferece orientação de como adquirir e modelar os requisitos. O método assume que os requisitos já existem, pois ele parte da especificação dos requisitos para em seguida utilizá-los durante o processo de seleção.

2.5.2 Método IusWare

O método IusWare [39] é baseado na abordagem Auxílio de Decisão de Critérios Múltiplos (ADCM). O método define um processo de avaliação que consiste de duas fases principais, a primeira é o projeto do modelo de avaliação e a segunda fase é chamada de fase de aplicação. A fase de projeto inclui as seguintes atividades:

- ❖ Identificar os atores relevantes no processo de avaliação, descrevendo seus papéis e objetivos, assim como os recursos disponíveis para esse processo;
- ❖ Identificar o tipo de avaliação necessária, onde pode ser feita uma descrição detalhada dos produtos ou ranking de produtos, indicando a preferência de cada um;
- ❖ Definir uma hierarquia não redundante dos atributos usados durante a avaliação;
- ❖ Associar uma medida, uma escala de critério e uma função para transformar a escala de medida numa escala de critério para cada atributo;
- ❖ Escolher uma técnica de agregação para agregar valores no critério de seleção.

Na fase de aplicação, os atributos dos produtos são medidos, em seguida essas medidas são transformadas em valores e agregadas de acordo com a recomendação dos atores. Apesar do método IusWare ser uma metodologia para avaliar e selecionar produtos de software, ele não trata a questão dos requisitos nesse processo. O método apresenta um modelo de avaliação que envolve a definição de uma hierarquia dos atributos juntamente com um modelo de tomada de decisão que é baseado na abordagem ADCM. A principal fraqueza desse método é que ele assume que os requisitos já estão disponíveis.

2.5.3 Framework STACE

O framework STACE (Social-Technical Approach to COTS Evaluation) tem como objetivo descrever uma abordagem sócio-técnica para selecionar produtos [27]. STACE enfatiza a importância da participação dos usuários no processo de seleção pois essa é a principal forma de trazer a dimensão social ao desenvolvimento de sistemas. Esse método usa a abordagem identificação por características chave e dessa forma sugere selecionar a tecnologia que os produtos candidatos devem se basear antes de iniciar o processo seletivo. O Framework STACE é fortemente influenciado pelos trabalhos de Kontio (método OTSO) e Maiden (método PORE), ele possui alguns princípios fundamentais:

- ❖ Suporta uma abordagem sistemática de avaliação e seleção de COTS, permitindo a reutilização de conhecimento e casos futuros;
- ❖ Suporta a avaliação tanto de produtos COTS como de tecnologia subjacentes;
- ❖ Usa abordagem sócio-técnica para melhorar o processo de seleção;
- ❖ Suporta a avaliação de fatores não técnicos.

STACE recomenda que o uso de um critério de seleção deve ser baseado em fatores sócio-técnico, onde são analisadas características de funcionalidade, qualidade do produto segundo normas ISO e fatores econômicos. Durante o processo de avaliação dos produtos, a participação dos usuários é usada para aumentar a aceitação do produto, como também para resolver problemas de conflito de interesses. Uma das limitações desse método é a ausência de uma avaliação dos requisitos não-funcionais durante o processo de seleção de

COTS. O método STACE se baseia em normas ISO, essas normas definem os requisitos de qualidade como um conjunto de atributos do produto, porém não analisam o impacto que tais atributos proporcionarão em cada produto a ser selecionado.

2.5.4 Método CRE

Esse método tem como principal estratégia explorar a descrição dos requisitos não-funcionais durante o processo de seleção [2]. O método fornece um *template* cujo objetivo é fornecer diretrizes para a realização das suas atividades. Ele está dividido em quatro fases: Identificação; Descrição; Avaliação e Aceitação.

A fase de Identificação tem os seguintes objetivos:

- ❖ Elicitar os requisitos críticos dos *stakeholders*;
- ❖ Elaborar critério de seleção;
- ❖ Guiar o processo de localização de produtos;
- ❖ Adquirir informações sobre os produtos identificados;
- ❖ Atribuir escores de conformidade entre os produtos e os requisitos críticos.

A fase seguinte, Descrição, detalha o critério de seleção com foco nos requisitos não-funcionais. O objetivo dessa etapa é discriminar os produtos utilizando os requisitos não-funcionais e atribuir prioridades a cada requisito.

A fase de Avaliação objetiva avaliar a adequabilidade de cada produto em relação ao critério de seleção e quantificar cada alternativa para ser usado por uma técnica de tomada de decisão existente no mercado. E por fim, a fase de Aceitação realiza uma validação final dos requisitos com os produtos selecionados.

Esse método possui como limitação o tempo para realizar todas as suas fases propostas. Ele usa o NFR *Framework* para tratar os requisitos não-funcionais, isso pode causar um aumento de custo no treinamento da equipe de seleção para usar essa ferramenta.

Outra desvantagem desse método é a não possibilidade de escolher múltiplos COTS dentro de um sistema integrado.

2.5.5 Método PORE

O método PORE (Procurement-Oriented Requirements Engineering) é baseado num processo iterativo de aquisição de requisitos e avaliação/seleção de produtos [1]. A abordagem de PORE consiste de seis processos genéricos. O modelo de processo descreve os processos fundamentais que precisam ser realizados durante a aquisição de produtos COTS, são eles:

- ❖ Processo de gerenciamento de aquisição do produto – onde os objetivos são planejar e controlar o processo de aquisição;
- ❖ Processo de aquisição de requisitos – esse processo adquire e valida os requisitos dos usuários, ele também determina a arquitetura que os produtos deverão seguir;
- ❖ Processo de seleção de fornecedor – o objetivo é estabelecer um critério para selecionar o fornecedor e fazer um ranking de acordo com o critério das melhores alternativas;
- ❖ Seleção de pacote de software – o objetivo é identificar os pacotes candidatos, estabelecer o critério de seleção usando os requisitos dos usuários e fazer um ranking para selecionar o produto que melhor se ajusta aos requisitos chave dos usuários;
- ❖ Produção do contrato – onde são negociados os contratos legais com o fornecedor, além disso são negociados os direitos e garantias;
- ❖ Aceitação do pacote – o objetivo é validar o pacote em relação aos requisitos chave dos usuários.

Uma das principais fraquezas desse método é falta de uma análise mais detalhada de como deve ser realizado o processo de requisitos. Esse método traz uma abordagem nova, porem não descreve como o critério de seleção deve ser obtido e utilizado, como também esquece de considerar aspectos não técnicos durante o processo seletivo, tais como: custo e

treinamento. Outras limitações do método são a sua complexidade de uso e o tempo necessário para realizar todas as fases propostas.

2.6 Comentários

Este capítulo mostrou o estado atual do Desenvolvimento Baseado em COTS (DBC) e os processos que influenciam nas suas atividades. Como influencia principal para o DBC, a Engenharia de Requisitos desenvolve um papel fundamental para o sucesso desse novo paradigma. Praticamente, todas as fases da Engenharia de Requisitos tradicional foram mantidas para o processo DBC. Os requisitos elicitados no DBC podem ser confrontados com os produtos identificados no mercado, isso garante a redução do tempo de desenvolvimento do produto e, provavelmente, a redução de custo do projeto.

As técnicas para tomada de decisão, podem auxiliar no processo de seleção. Todavia, elas fornecem vantagens e desvantagens para o processo DBC. Talvez, a principal desvantagem dessas técnicas seja a falta de uma abordagem que trate o inter-relacionamento entre os atributos, pois os requisitos dos usuários, geralmente, interagem entre si.

As abordagens para DBC descritas neste capítulo, apresentam fraquezas que este trabalho tenta suprir. Mesmo sendo o método mais consistente, o método PORE apresenta um grau de complexidade muito grande nos seus processos, e não fornece nenhuma diretriz para uma possível automação do processo de seleção.

O próximo capítulo fornece uma proposta para o DBC que enfatiza a automação do processo de seleção de COTS a fim de obter ganho no tempo e na exatidão do melhor produto para a instituição.

Capítulo 3

O MÉTODO PROPOSTO

Este capítulo descreve a principal contribuição deste trabalho, o método ASCRE. Será mostrada individualmente cada uma das três fases que compõem o método, são elas: Preparação, Checagem e Decisão.

3.1 Descrição Geral da Abordagem

Esta abordagem propõe um método que orienta a criação de bases de informações para automatizar os processos de avaliação e seleção/rejeição de um ou mais COTS dentro da Engenharia de Requisitos chamado de método ASCRE (Automação da Seleção de COTS na Engenharia de Requisitos). Esse método é orientado a informações e tem como principal estratégia o armazenamento das informações em tabelas relacionais, que irão proporcionar maior segurança e performance na escolha dos componentes.

O método ASCRE é fundamentado no processo iterativo e interativo de aquisição de informações e seleção/rejeição de produtos. Os processos de aquisições de informações alimentam as bases de dados para subsidiar a negociação entre requisitos dos *stakeholders* e requisitos dos fornecedores/produtos. À medida que evolui a negociação, a quantidade de requisitos aumenta e o número dos produtos candidatos tende a diminuir, pelo grau de exigência do processo. Isso demonstra que o ASCRE usa a filtragem progressiva (ver Seção 2.5) como estratégia de seleção e uma avaliação exaustiva (ver Seção 2.2.2). A próxima seção descreve as fases do método ASCRE.

3.2 As Fases do Método ASCRE

Este método é composto por atividades nas quais são levantadas informações subjetivas e objetivas que servem de subsídios para uma tomada de decisão objetiva. A objetividade da escolha é facilitada pela classificação final dos produtos em relação ao critério de avaliação definido.

O ASCRE possui três fases que interagem entre si, são elas: Preparação, Checagem e Decisão. Esta ordem não é rígida e a atividade que será realizada depende do tipo de informação disponível no momento e do resultado pretendido, com exceção do início do método, que deve ser feita pela fase de preparação, e do final, que é alistar os produtos classificados. A figura 2 fornece uma visão geral do método, mostrando as atividades de cada fase e os relacionamentos entre as mesmas e os repositórios.

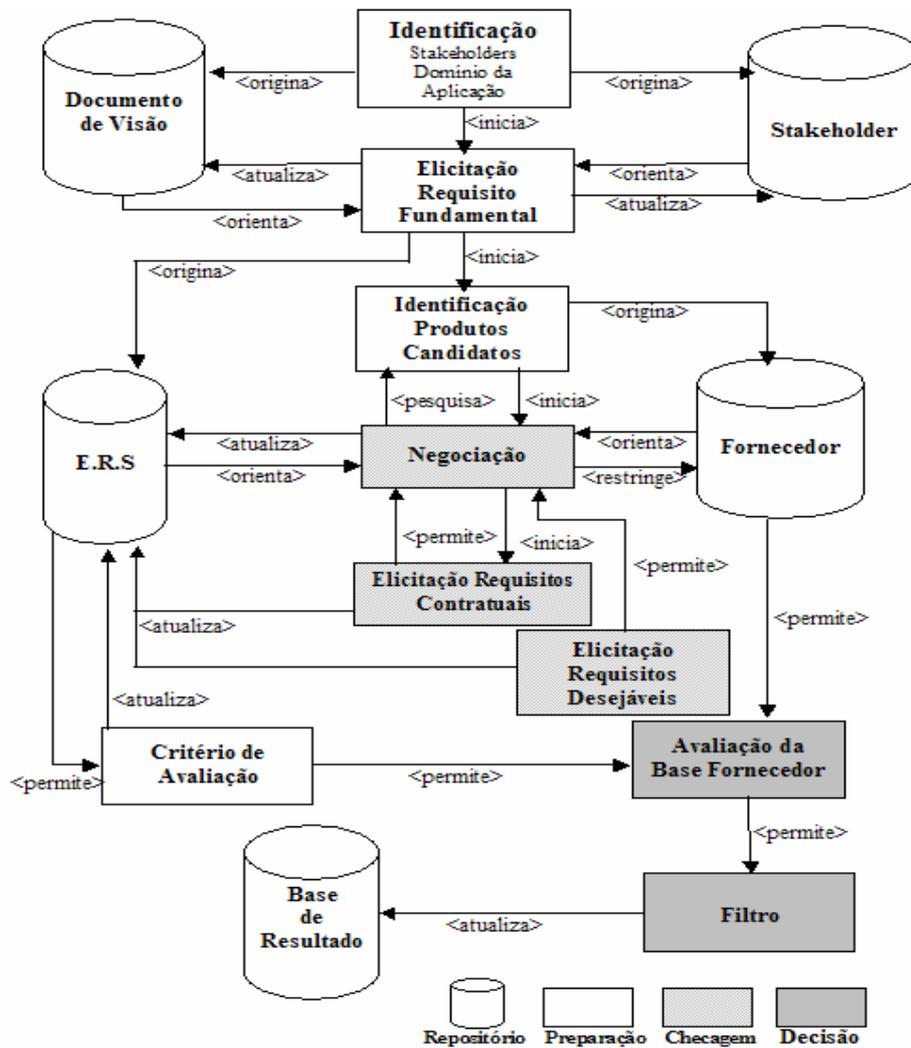


Figura 3.1 Visão Geral da Metodologia ASCRE

As atividades são representadas (figura 3.1) pelos retângulos e as setas representam a relação entre elas (atividades) e as bases de dados. A seta *<origina>* significa o início de armazenamento de informações dentro de uma base de dados. A seta *<inicia>* mostra uma atividade dando início a outra atividade. A seta *<atualiza>* informa que uma atividade está alimentando a base de informação. As informações contidas nas bases de dados servem, também, para orientar a execução das atividades e são representadas pela seta *<orienta>*. A seta *<pesquisa>* indica que uma atividade busca informação na atividade anterior. Uma atividade ou uma base de dados permite que outra atividade seja executada com maior precisão, representada na figura 2 pela seta *<permite>*. A seta *<restringe>* indica que a atividade pode diminuir a quantidade de registros na base de dados.

3.2.1 Preparação

Esta fase inicia a construção dos repositórios e tem como objetivo identificar todos os *stakeholders*, os requisitos fundamentais, os COTS candidatos e estabelecer o critério de avaliação. Para alcançar esse objetivo deve-se passar pelas seguintes atividades:

- ❖ identificar *stakeholders*;
- ❖ identificar o domínio da aplicação;
- ❖ elicitar e analisar os requisitos fundamentais;
- ❖ definir o critério de avaliação;
- ❖ identificar COTS candidatos.

Para começar a identificação dos *stakeholders* é necessário realizar uma reunião com o cliente principal (gestor) do contrato. Ele deve fornecer uma visão geral do problema a ser abordado e identificar as pessoas que vão fornecer os requisitos, dentro do seu ponto de vista. Neste momento pode-se identificar o domínio da aplicação e buscar mais informações com os *stakeholders* identificados. Ao definir o domínio, a equipe de seleção não só restringe o conteúdo dos requisitos, como também realiza um ato de pré-seleção do universo dos possíveis candidatos.

As reuniões com os *stakeholders* podem aumentar o domínio do problema e o domínio pode aumentar o número dos usuários envolvidos no projeto.

A atividade de identificação dos *stakeholders* origina duas bases (a base do Stakeholder e a base do Documento de Visão) que irão orientar a equipe de engenharia de requisitos na busca de informações para satisfazer o objetivo do Documento de Visão.

É comum em projeto de sistema se identificar que o domínio possui subdomínios, e os subdomínios possuem aplicações. Logo, uma instituição que deseja adquirir uma solução para atender um determinado domínio, poderá adquirir mais de um COTS para atender os seus subdomínios e assim compor uma solução de domínio. Sendo assim, O método ASCRE

propõe que o Documento de Visão deva conter informações para o domínio, subdomínio e aplicações separadamente. Essas informações devem contemplar: o objetivo; as necessidades de alto-nível; e as características genéricas do sistema, focando nas potencialidades requeridas pelos *stakeholders*. Um exemplo do Documento de Visão é mostrado na figura 3.

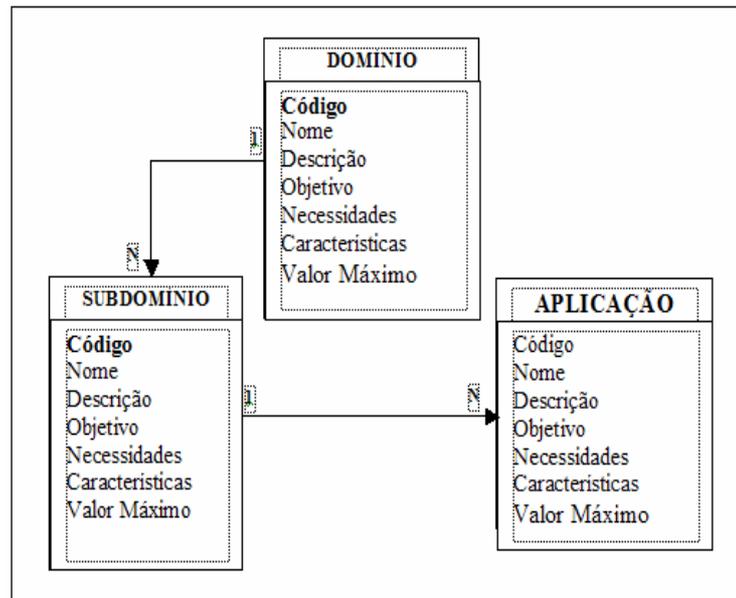


Figura 3.2 - Exemplo do Documento de Visão

A organização pode simplesmente definir o custo máximo para o domínio ou distribuir o orçamento entre os subdomínios e/ou aplicações. Logo o valor máximo da figura 3.2 é o custo que a empresa pode suportar para aquisição da solução. O valor máximo do domínio deve ser igual à soma dos valores dos subdomínios e o valor máximo do subdomínio deve ser a soma dos valores das aplicações.

Esta abordagem considera que todo projeto tem que possuir, obrigatoriamente, um domínio e pelo menos uma aplicação. O subdomínio não é obrigatório, mas uma vez existindo um subdomínio, ele terá que possuir pelo menos uma aplicação subordinada a ele.

A base de *stakeholder* precisa conter informações que auxiliem na engenharia de requisitos, como: profissão; posição que ocupa na instituição; tempo de experiência no domínio do problema (muito importante para resolver conflitos e dá entendimento das regras de negócios); horário de trabalho e telefone de contato (importante para marcar reuniões).

Com o Documento de Visão e a base de stakeholder iniciada, a equipe de elicitação de requisitos pode iniciar o levantamento dos requisitos fundamentais. Esta atividade requer a melhor comunicação possível com os *stakeholders*. Requer, também, um entendimento do que é indispensável para o êxito do projeto. Os requisitos levantados nesta fase são os considerados críticos, são aqueles que os *stakeholders* consideram indispensáveis para atender as suas necessidades. Isso se dá, para que a busca dos produtos candidatos seja mais rápida e objetiva.

Os métodos pesquisados (ver Seção 2.5) discriminam entre requisitos funcionais e não-funcionais. Priorizando um e desprezando o outro, ou não valorizam a elicitação dos mesmos. O ASCRE não faz distinção em nível de prioridade entre requisitos funcionais e os não-funcionais. Entende-se que a funcionalidade pode ser crítica, porém o requisito não-funcional, geralmente, é o que viabiliza a funcionalidade, como por exemplo a arquitetura. Como o objetivo da abordagem DBC é selecionar COTS que atendam o domínio e suas ramificações, o método ASCRE divide os requisitos em três níveis, são eles:

- ❖ Fundamental – é todo requisito crítico, sem o qual se torna impossível alcançar o objetivo do projeto e não será descrito no contrato. Pode ser funcional ou não-funcional;
- ❖ Contratual – é todo requisito que envolve as regras de negócios para aquisição do componente junto ao vendedor. Geralmente consiste de requisito não-funcional;
- ❖ Desejável – é todo aquele que não influencia no objetivo principal da aplicação, mas faz parte do desejo do stakeholder ou é sugerido pelo fornecedor para discriminar a concorrência. Pode ser funcional ou não-funcional.

Sendo assim, a prioridade, nesta abordagem, são os requisitos fundamentais, sem os quais se torna inatingível o objetivo em questão.

Neste estágio de aquisição de requisitos, as técnicas tradicionais (ver Seção 2.3.1) são as mais apropriadas, já que a equipe de elicitação está em constantes reuniões com os *stakeholders*. Aconselha-se também, ter especialistas do domínio fazendo parte da equipe, onde eles assumem um importante papel durante esse processo inicial de elicitação.

Para suportar os requisitos elicitados, esta abordagem propõe a ERS (Especificação de Requisitos de Software), que é base que irá conter os requisitos elicitados, analisados, especificados e validados (ver Seção 2.3.1). Esses requisitos podem ser de domínio ou subdomínio ou aplicação. Quando um requisito pertence a um domínio, significa que todos os seus subdomínios têm que suportar aquele requisito, como por exemplo: o domínio determina que o sistema operacional tem que ser Linux. Um requisito de subdomínio tem que ser suportado por suas aplicações. Uma aplicação pode ter vários requisitos e um requisito pode pertencer a várias aplicações. Logo, ao registrar que um requisito pertence ao nível superior da hierarquia (domínio → subdomínio → aplicação), não se deve registrar para seus níveis inferiores.

O repositório de requisitos deve conter:

- ❖ Identificação – a abordagem que será utilizada para identificação dos requisitos, fica por conta da equipe de elicitação (ver Seção 2.3.1- Gerenciamento de Requisitos);
- ❖ Descrição;
- ❖ Tipo – funcional ou não-funcional;
- ❖ Nível – Fundamental ou Contratual ou Desejável;
- ❖ Justificativa para a solicitação do requisito;
- ❖ Objetivo principal.

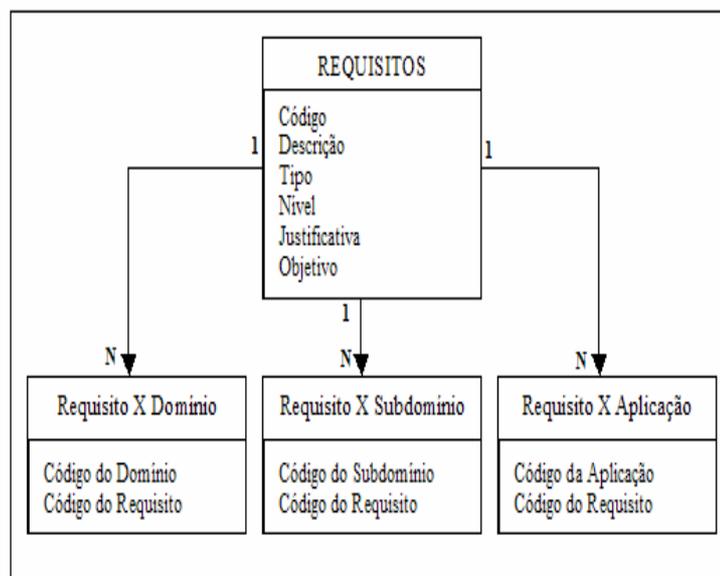


Figura 3.3 - Relacionamento dos Requisitos

A figura 3.4 mostra a tabela de requisitos, onde um requisito pode se relacionar com um ou mais domínio/subdomínio/aplicação.

Outra atividade muito importante neste início de projeto, é definir o critério de avaliação, pois nele deve está definido as regras que satisfaçam a melhor solução possível, ou talvez, quais as melhores soluções para o domínio em questão.

As regras consistem em definir formas de pontuação para os fornecedores/produtos em relação aos requisitos do projeto. A disposição das regras deve favorecer o resultado nos três níveis de requisitos (fundamental, contratual e desejável), ou seja, o critério de avaliação permitirá que o resultado final do certame seja na seguinte ordem:

- ❖ Classificação dos produtos em relação aos requisitos fundamentais;
- ❖ Classificação dos produtos em relação aos requisitos contratuais;
- ❖ Classificação dos produtos em relação aos requisitos desejáveis.

Essa classificação pode definir a melhor (ou as melhores) solução para o domínio. Todavia, existem outros elementos que são importantes na avaliação final, como por exemplo os clientes desses fornecedores. O método ASCRE recomenda que a equipe de projeto levante informações junto a esses clientes para poder avaliar tanto o produto quanto o fornecedor, pois pode haver informações importantes que ajudem a esclarecer melhor o uso do produto e a relação com o fornecedor, como por exemplos:

- ❖ Usabilidade;
- ❖ Compatibilidade;
- ❖ Performance;
- ❖ Responsabilidade do fornecedor quanto ao suporte, manutenção e etc;
- ❖ Confiabilidade no produto e no fornecedor.

Vale ressaltar, que as técnicas para tomada de decisão (ver Seção 2.4) podem servir para orientar a equipe de projeto quando for elaborar o critério de avaliação.

Recomenda-se que os primeiros critérios a serem definidos sejam aqueles que eliminam os candidatos sem haver a mínima possibilidade de negociação, como por exemplo: o valor do COTS não pode ultrapassar o orçamento da aplicação definido pelo gestor do projeto.

Uma sugestão de seqüência para a elaboração do critério de avaliação é:

1. Critério de avaliação de caráter eliminatório;
2. Critério de avaliação para cada nível de requisito – cada nível de requisito (Fundamental, Contratual, Desejável) deve possuir uma pontuação;
3. Critério de avaliação do fornecedor em relação ao cliente – um fornecedor deve ser avaliado pelos clientes que ele já possui no mercado, pois são esses clientes que irão fornecer informações importantes sobre a credibilidade do fornecedor;
4. Critério de avaliação do produto candidato (COTS) em relação ao cliente – quem melhor pode dar informação de um produto, são aqueles que utilizam ou utilizaram tal produto.

Tomando como referência o produto das atividades acima, pode se começar o processo de busca dos produtos candidatos. Esta atividade, identificar produtos candidatos, é informal e muito depende de fatores externos, pois não existe um processo padrão. Ainda assim, existem caminhos que podem ser seguidos, como usar várias fontes de informação para auxiliar a busca. Além disso, é aconselhável documentar as informações coletadas nesse processo. Portanto, o método ASCRE recomenda usar três tabelas para armazenar essas informações, são elas:

- ❖ Dados do fornecedor – informações sobre empresa que fornece ou construiu o produto candidato;
- ❖ Dados do cliente – informações sobre aqueles que já utilizam ou utilizaram os produtos candidatos;
- ❖ Dados do produto – informações sobre COTS candidato.

Um possível relacionamento dentro da base do fornecedor é mostrado na figura 3.4. Essa figura apresenta a seguinte notação [41]:

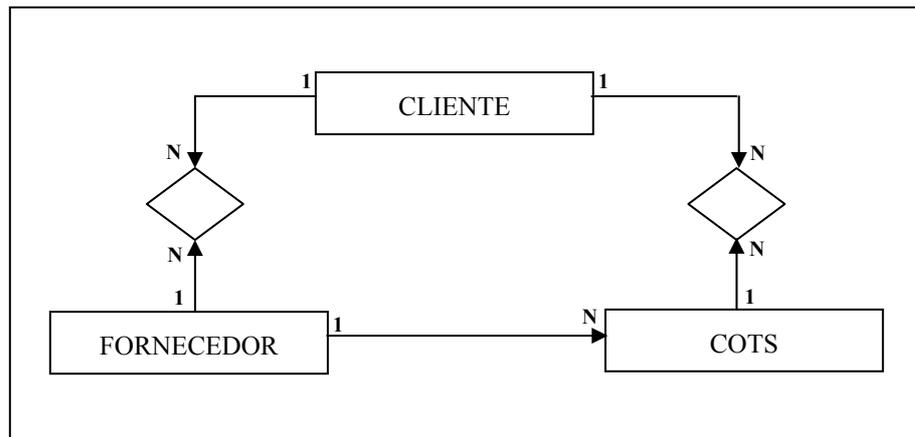
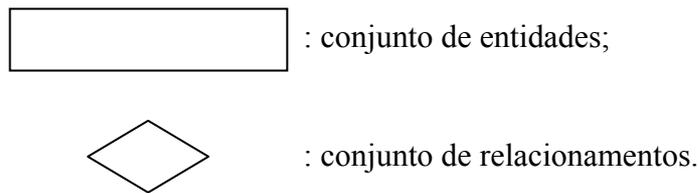


Figura 3.4 - Exemplo Base Fornecedor

A fase de Avaliação (ver Seção 2.2.2) recomenda que as propriedades de cada produto devem ser identificadas e analisadas. Todavia, esta abordagem considera que as funcionalidades, o comportamento e as restrições dos COTS candidatos devem ser avaliados durante o processo de negociação com os requisitos dos *stakeholders*.

Os locais de busca desses candidatos são diversificados. Porém, a internet se tornou a principal fonte de pesquisa, também, para os produtos COTS. Outras fontes de pesquisas são os vendedores, periódicos especializados, consultores, conferências e feiras.

Os responsáveis pela identificação dos COTS devem verificar a frequência com que novas alternativas são descobertas e analisar quando não existir um número considerável de novos achados, para decidir quando deve parar as buscas.

Esta fase do método tem como produto significativo às bases Documento de Visão, Stakeholder, Fornecedor e ERS. Elas serão bastante aproveitadas, para orientar e serem atualizadas, na próxima fase.

3.2.2 Checagem

Esta etapa consiste em três atividades, são elas:

- ❖ Fazer a negociação entre a base ERS e a base do Fornecedor;
- ❖ Elicitação dos requisitos contratuais;
- ❖ Elicitação dos requisitos desejáveis.

A atividade de negociação deve propiciar uma interação entre a Especificação dos Requisitos de Software (ERS) e a base do Fornecedor (Fornecedor, Cliente e COTS). Este processo é a parte principal da metodologia, pois os requisitos do projeto, as características do produto e as condições do fornecedor precisam se relacionar de forma harmônica, para o êxito da seleção do COTS. Para alcançar esse objetivo, o processo de negociação precisa relacionar os requisitos do projeto com cada produto candidato e com fornecedor, permitindo uma flexibilidade das partes envolvidas. A flexibilidade consiste em possíveis mudanças de ambos para atingir o propósito do Documento de Visão, sendo que este também pode sofrer alterações durante o processo de negociação.

Os dois principais relacionamentos da negociação são:

- ❖ Requisito X Fornecedor – onde deve conter o tipo de relacionamento que o fornecedor tem o requisito;
- ❖ Requisito X COTS – deve informar quais os requisitos que o produto candidato se relaciona e qual o tipo de relacionamento.

Existem muitas possibilidades de relacionamentos com um requisito. Um fornecedor/produto pode se relacionar com um requisito de várias maneiras, isso vai depender do tipo de abordagem que a equipe de projeto queira implementar, por exemplo: A equipe decide que um requisito tem quatro status em relação ao fornecedor/produto, ou seja, o fornecedor/produto pode atender totalmente ou atender parcialmente ou atender com serias restrições ou não atender o requisito. A figura 3.5, com a mesma notação da figura 3.4, mostra um possível relacionamento dos requisitos com os fornecedores/produtos, onde o relacionamento serviço identifica a situação do requisito no concorrente.

É importante ressaltar que o método CRE (ver seção 2.5.4) utiliza os serviços mencionados no exemplo (atende, atende parcialmente, atende com serias restrições e não atende) como escore de conformidade e atribui valores a cada um deles baseado na técnica de tomada de decisão WSM (ver seção 2.4.2). Esta abordagem entende que os valores não estão somente ligados ao escore de conformidade (relacionamento) com os requisitos, mas, também, ao nível de requisito (fundamental, contratual, desejável).

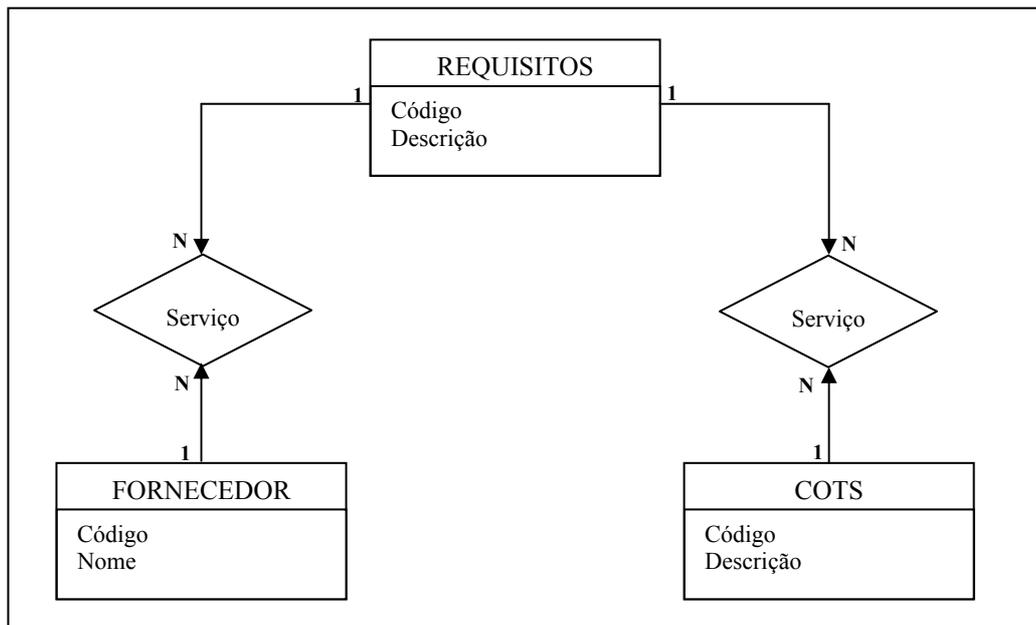


Figura 3.5 - Relacionamento dos Requisitos X Fornecedor/Produto

O processo de negociação permite a restrição de candidatos quando não for possível o ajuste da ERS ao fornecedor, e permite, também, que a atividade de identificação de novos produtos seja executada novamente.

O ASCRE sugere as seguintes etapas para o processo de negociação:

1. Selecionar os critérios de avaliação de caráter eliminatório e verificar se algum fornecedor/produto não atende a esses critérios. Os fornecedores/produtos que não atenderem serão excluídos do certame. A exceção dessa eliminação se dá quando a equipe de projeto resolve modificar o critério de avaliação. Depois da alteração deve ser feita nova verificação em todos os candidatos.

2. Selecionar o fornecedor e armazenar os requisitos que ele atende e que tipo de serviço fornece, na seguinte ordem: requisito de domínio; requisito de subdomínio e requisito de aplicação.
3. Selecionar o produto e armazenar os requisitos com o tipo de serviço que ele suporta. Também, na seqüência de domínio, subdomínio e aplicação.

Em todas as três etapas pode-se consultar o fornecedor na perspectiva de ajuste com o projeto, pois o processo de negociação tem como objetivo principal atender de forma satisfatória os *stakeholders* e não, somente, eliminar candidatos.

A atividade de elicitação de requisitos contratuais pode ser iniciada assim que comece a negociação, pois os *stakeholders* e a equipe de projeto já tem uma visão bem mais ampla do que eles precisam e o que o mercado pode oferecer. Sendo assim, alguns requisitos do contrato já podem ser definidos, como por exemplo:

- ❖ Descrição genérica do objeto do contrato;
- ❖ Prazo de vigência do contrato;
- ❖ Preço máximo e forma de pagamento que a instituição pode suportar;
- ❖ Critério genérico das obrigações das partes envolvidas;
- ❖ Garantia mínima aceitável pela instituição;
- ❖ Descrição do suporte técnico desejável;
- ❖ Período de lançamento de novas versões;
- ❖ Descrever as responsabilidades dos serviços de customização.

Outra atividade que já pode ser iniciada, é a elicitação de requisitos desejáveis, pois eles ajudam a discriminar entre produtos semelhantes (i.e. que já atendem a todos os requisitos fundamentais e contratuais). Os requisitos desejáveis e contratuais devem alimentar a base ERS e fazerem parte da negociação com cada produto candidato.

Neste momento a equipe de projeto deve aceitar sugestões dos fornecedores, pois os mesmos já possuem experiências nas regras de negócios. Os requisitos sugeridos pelos

fornecedores devem ser repassados para os *stakeholders* para serem validados. Se o requisito for aceito, ele deve constar na base da ERS e fazer parte da negociação com todos os candidatos.

Outro elemento importante para fornecimento das informações tanto do fornecedor/produto quanto de requisitos para o projeto, são os clientes desses fornecedores, eles podem ajudar na ampliação da visão do projeto.

O produto desta fase é as bases consolidadas com informações suficientes para uma tomada de decisão. A equipe de projeto deve decidir se as atividades até aqui executadas e os resultados alcançados, já são o suficiente para poder ser executada a fase de Decisão. Se a resposta for negativa, todos os processos podem ser executados novamente, sem perda dos trabalhos anteriores, pois as bases de dados garantem a reutilização do trabalho. O importante é que a fase de Decisão, que é a próxima, tenha informações suficientes para executar suas atividades.

3.2.3 Decisão

É muito importante que esta fase seja executada depois que as outras duas fases tenham sido exaustivamente executadas, pois as informações devem estar esmeradas para o êxito da seleção. A Seção 2.4 descreve técnicas para tomada de decisão.

Esta fase possui dois processos: processo de avaliação da base de Fornecedor e o processo de filtragem ou classificação. O processo de avaliação é composto das seguintes atividades:

1. Identificar o valor máximo possível para o domínio, subdomínio e aplicação;
2. Verificar a pontuação de cada COTS candidato em relação ao domínio, subdomínio e aplicação;
3. Verificar a pontuação de cada fornecedor/produto em relação ao domínio, subdomínio e aplicação.

□ **Identificar o valor máximo possível para o domínio, subdomínio e aplicação**

A identificação dos valores máximos possíveis para o projeto é facilitada pelo critério de avaliação definido na fase de Preparação. Esse critério deverá ter definido uma pontuação para cada um dos três níveis de requisitos (fundamental, contratual e desejável) associados ao tipo de serviço proposto. Então, a primeira ação é localizar a maior pontuação possível para um requisito em cada um dos níveis, ou seja, a maior pontuação possível para um requisito no nível fundamental; a maior pontuação possível para um requisito no nível contratual e a maior pontuação possível para um requisito no nível desejável.

Em seguida, deverá ser lido todos os requisitos de domínio. Para cada requisito lido, deve-se identificar o seu nível e acumular o valor referente a maior pontuação possível para aquele nível, como por exemplo: um domínio que possua 20 (vinte) requisitos no nível fundamental e a maior pontuação para o nível de requisito fundamental é 10 (dez), o valor máximo possível para esse domínio no nível fundamental é de 200 (duzentos) pontos.

Esse procedimento deve ser executado de tal forma que no final se tenha o valor máximo que um candidato poderá obter no domínio para cada nível, como por exemplo: valor máximo do domínio para o nível fundamental é igual a 300; valor máximo do domínio para o nível contratual é igual a 260; valor máximo do domínio para o nível desejável é igual a 100.

O mesmo procedimento deverá ser executado para encontrar os valores máximos possíveis para o subdomínio e para a aplicação.

□ **Verificar a pontuação de cada COTS candidato em relação ao domínio, subdomínio e aplicação**

A pontuação de um produto candidato pode ser encontrada de varias maneiras. A seqüência de passos para se encontrar esses valores, vai depender do modelo da base de dados. Todavia, o método ASCRE sugere os seguintes passos:

- Para o domínio
 1. localizar o domínio em questão;
 2. identificar um requisito desse domínio;

3. identificar o nível desse requisito (fundamental, contratual ou desejável);
 4. localizar um produto que possua esse requisito e qual tipo de serviço ele atende;
 5. identificar no critério de avaliação quantos pontos vale esse nível de requisito para o tipo de serviço proposto pelo produto;
 6. acumular o valor encontrado para o produto no nível de requisito identificado;
 7. executar os passos 4, 5 e 6 até não existir mais produtos que atendem esse requisito;
 8. voltar a executar a seqüência de passos acima, iniciando no passo 2, até não existir mais requisitos para o domínio em questão. Não existindo mais requisitos, finaliza.
- Para o subdomínio e aplicação
1. localizar um subdomínio pertencente ao domínio em questão;
 2. identificar um requisito desse subdomínio;
 3. identificar o nível desse requisito (fundamental, contratual ou desejável);
 4. localizar um produto que possua esse requisito e qual tipo de serviço ele atende;
 5. identificar no critério de avaliação quantos pontos vale esse nível de requisito para o tipo de serviço proposto pelo produto;
 6. acumular o valor encontrado para o produto no nível de requisito identificado;
 7. executar os passos 4, 5 e 6 até não existir mais produto que atende esse requisito;
 8. voltar a executar a seqüência de passos acima, iniciando no passo 2, até não existir mais requisitos para esse subdomínio;
 9. localizar uma aplicação pertencente ao subdomínio que esta sendo processado;
 10. identificar um requisito dessa aplicação;

11. executar o passo 3;
12. executar os passos 4, 5, 6 e 7;
13. voltar a executar a seqüência de passos acima, iniciando no passo 10, até não existir mais requisitos para essa aplicação;
14. voltar a executar a seqüência de passos acima, iniciando no passo 9, até não existir mais aplicação para esse subdomínio;
15. voltar a executar a seqüência de passos acima, iniciando no passo 1, até não existir mais subdomínio para o domínio em questão. Não existindo mais subdomínio, finaliza o procedimento.

□ **Verificar a pontuação de cada fornecedor candidato em relação ao domínio, subdomínio e aplicação.**

A pontuação dos fornecedores pode seguir os mesmos passos já mencionados para os produtos, só que substituindo a localização de produtos por fornecedores, sendo que essa pontuação será computada e armazenada separadamente do produto, porque um fornecedor pode está concorrendo com mais de um produto e no processo de filtragem a classificação será por produto e não por fornecedor, embora se considera para o produto a pontuação do seu fornecedor no item que ele está concorrendo.

Tanto o produto quanto o fornecedor poderão ser pontuados, dependendo do critério de avaliação, em relação aos seus clientes. Essa avaliação é mais complexa, pois, teoricamente, um fornecedor/produto que possua muitos clientes poderá ser beneficiado na hora da decisão. Sendo assim, a decisão de criar critérios que considerem as informações dos clientes fica por conta da equipe de projeto. Todavia, se a equipe decidir por considerar as informações dos clientes, o método ASCRE propõe a mesma abordagem da avaliação do produto.

Com base nas informações da avaliação, o processo de filtragem irá fornecer diversas visões do resultado. Essas visões dependem fundamentalmente da forma que a base de dados foi modelada, por isso que esta abordagem insiste em alertar para que a análise da

base deva ser bastante criteriosa, e considere, sempre, a divisão dos requisitos nos níveis Fundamental, Contratual e Desejável.

A figura 3.6 mostra um exemplo de valores máximos possíveis para um projeto, onde os itens são pontuados separadamente com seus respectivos níveis de requisitos. Todavia, o resultado deverá considera a soma dos valores de cada nível à sua hierarquia superior. Alguns exemplos são demonstrados considerando a figura 3.6:

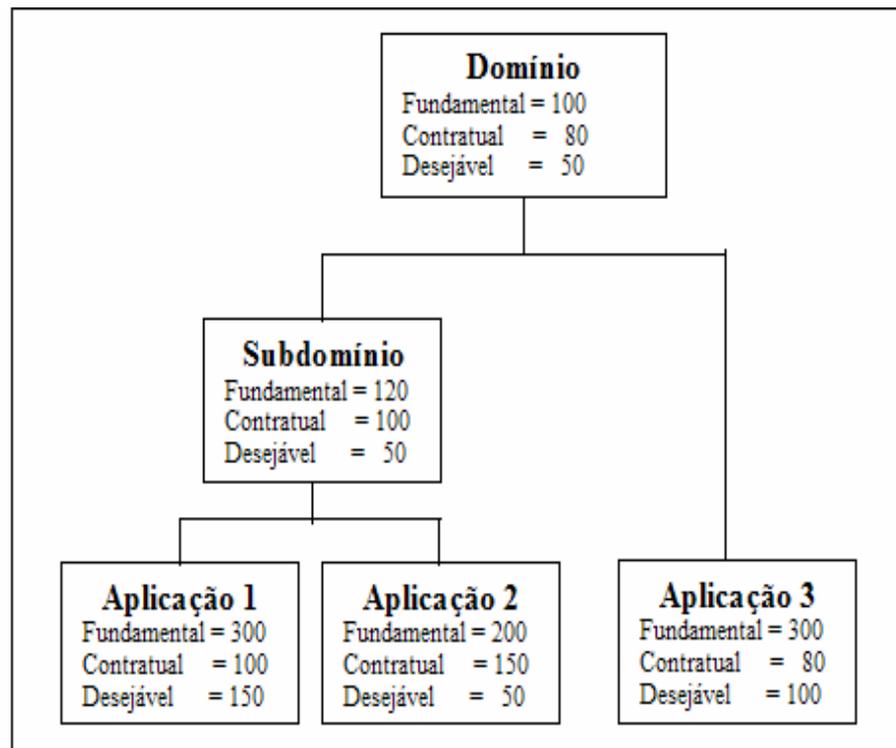


Figura 3.6 - Exemplo de Valores Máximos Possíveis para um Projeto

- ❖ Exemplo 1 – o valor máximo que um COTS pode obter para satisfazer os requisitos da Aplicação 3:
 - para nível de requisito fundamental → $300 + 100 = 400$
 - para nível de requisito contratual → $80 + 80 = 160$
 - para nível de requisito desejável → $100 + 50 = 150$
- ❖ Exemplo 2 – o valor máximo que um produto pode obter para satisfazer os requisitos da Aplicação 2:
 - para nível de requisito fundamental → $200 + 120 + 100 = 420$

- para nível de requisito contratual → $150 + 100 + 80 = 330$

- para nível de requisito desejável → $50 + 50 + 50 = 150$

❖ O valor máximo que um produto pode obter para satisfazer todo o Subdomínio:

- para nível de requisito fundamental → $300 + 200 + 120 + 100 = 720$

- para nível de requisito contratual → $100 + 150 + 100 + 80 = 430$

- para nível de requisito desejável → $150 + 50 + 50 + 50 = 300$

❖ O valor máximo que um produto pode obter para satisfazer o domínio.

- para nível de requisito fundamental → $300+200+300+120+100 = 1.020$

- para nível de requisito contratual → $100+150+80+100+80 = 510$

- para nível de requisito desejável → $150+50+100+50+50 = 400$

Considerando os valores máximos possíveis, o processo de filtragem irá fornecer os resultados, em percentuais, de cada produto. A demonstração do resultado é feita por item (domínio, subdomínio e aplicação) e nível de requisito, onde todos os pontos de cada concorrente (fornecedor/produto) deverão estar somados para o item de concorrência, como mostra a figura 3.7.

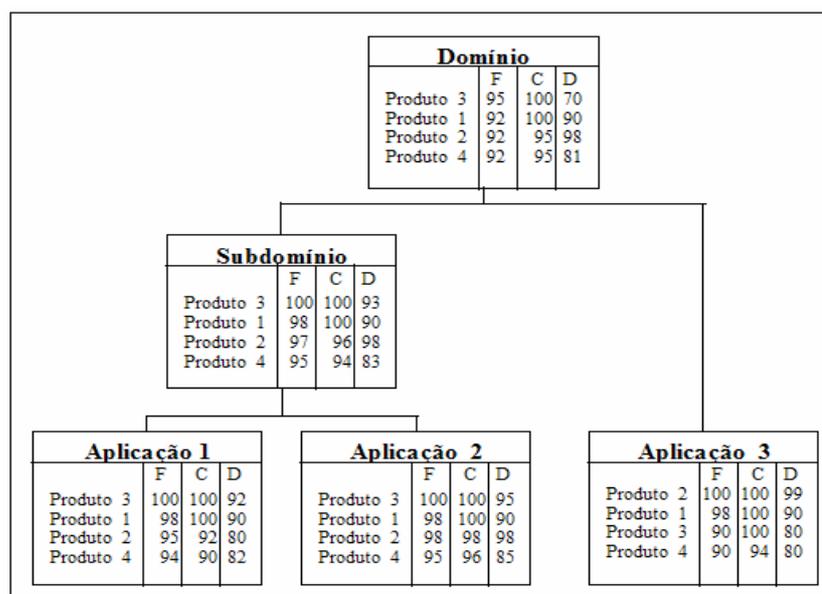


Figura 3.7 - Exemplo de Demonstração do Resultado

Repare na figura 8 que o produto 3 é o que mais atende os requisitos fundamentais de todo o domínio, considerando subdomínio e aplicações. Todavia, ele não pode satisfazer plenamente o projeto. Neste exemplo, o produto 3 pode ser selecionado para satisfazer todo o subdomínio, pois satisfaz plenamente os seus requisitos fundamentais e contratuais. Perceba, também, que o produto 1 é o segundo colocado na classificação geral (domínio), mas não consegue atingir o primeiro lugar nas outras classificações. Embora o produto 2 tenha ficado em terceiro lugar na classificação geral, ele ficou em primeiro na classificação da aplicação 3, podendo ser selecionado para o projeto.

O produto final desta fase é informações quantitativas e qualitativas para a seleção de COTS que pode ser testado em um ambiente mais realístico.

3.3 Comentários

Geralmente, as organizações selecionam produtos de maneira informal, pois se baseiam em informações subjetivas, como por exemplo, observar que produto outras instituições estão usando ou se basear no custo dos produtos no mercado. Isso é muito comum no poder executivo municipal do Brasil, onde empresas que visam lucro, se aproximam do município, para difundir soluções que deram certo em outro município e desconsideram a realidade local. Essa falta de orientação no processo de seleção pode comprometer toda uma instituição.

Para orientar as organizações, o método ASCRE não somente propõe sistematizar o processo de seleção, pois outros métodos já o fazem, como foi mostrado na Seção 2.5, mas, também, automatizar a seleção. Os outros métodos existentes na literatura apenas descrevem um conjunto de diretrizes para sistematizar o processo de seleção, enquanto o método ASCRE descreve diretrizes para sistematizar e automatizar a seleção de produtos COTS. Uma comparação entre os métodos pesquisados e o ASCRE é mostrada na tabela 1.

Diretrizes Métodos	Identificação do Produto	Aquisição dos Requisitos	Avaliação e Seleção	Análise de Decisão	Automação da Tomada de Decisão
IusWare	não trata	não trata	trata	trata	não trata
Framework Stace	trata	trata	trata	trata	não trata
CRE	trata	trata	trata	trata	não trata
PORE	trata	trata	trata	trata	não trata
ASCRE	trata	trata	trata	trata	trata

Tabela 3.1 – Comparativo com outras abordagens

A sistematização e automação são realizadas em três fases, que são: Preparação, Checagem e Decisão. Cada fase possui atividades bem definidas, que são interativas e iterativas numa perspectiva de melhor aproveitamento do tempo e minimização do custo, sem perder de vista a qualidade e satisfação dos requisitos.

A abordagem ASCRE tem como produto final uma classificação de COTS que pode ser suficiente para a empresa decidir qual o melhor ou os melhores produtos que correspondem as suas necessidades. Este poder de decisão é possível, porque o método ASCRE divide sua classificação em domínio, subdomínio e aplicação, considerando que os requisitos dos *stakeholders* podem ser fundamental, contratual ou desejável, para priorizar a satisfação do objetivo principal. Além disso, esta abordagem fornece um conjunto de base de dados que dará a organização condições para fazer uma análise detalhada dos resultados e verificar se a escolha corresponde aos objetivos dos *stakeholders* e se o fornecedor é idôneo.

Capítulo 4

IMPLEMENTAÇÃO

Como foi mencionada em capítulos anteriores, a proposta desta dissertação consiste de um método para automação da seleção de produtos na engenharia de requisitos, o ASCRE. Neste método são descritos processos que interagem com bases de dados que tem como objetivo final alistar produtos classificados para serem selecionados. Como a proposta é automatizar utilizando base de dados, este método não pode ser implementado de forma manual, logo é necessária a implementação de uma ferramenta na forma descrita pelo método.

Este capítulo descreve um protótipo de uma implementação da metodologia ASCRE, denominada de IASCRE – Implementação da Automatização da Seleção de COTS na Engenharia de Requisitos, para demonstrar a viabilidade do método.

A estrutura da base de dados e as telas foram implementadas em Msaccess 2000.

4.1 Protótipo

O protótipo de uma Implementação da Automação da Seleção de COTS na Engenharia de Requisitos – IASCRE foi desenvolvido com intuito de demonstrar a necessidade que se tem de automatizar os processos de avaliação e seleção de COTS dentro da Engenharia de Requisitos.

O IASCRE permite que o engenheiro de requisitos manipule os repositórios de forma fácil e amigável, para poder utilizar as informações da melhor maneira possível e otimizar o processo de avaliação e seleção, pois a redução de custos e tempo de desenvolvimento foram algumas das promessas do DBC apontada na seção 1.1.

No IASCRE os bancos de dados vistos na figura 3.1 consistem de tabelas relacionais que permitem:

- o armazenamento e a recuperação das informações do mapeamento do domínio;
- o armazenamento, a recuperação e a reutilização dos requisitos e dos fornecedores, visto que um requisito pode ser utilizado em vários domínios, e um fornecedor pode concorrer para uma ou mais estrutura de domínio;
- a automatização do processo de tomada de decisão.

Como descrito na seção 3.2.1, a fase de preparação inicia a alimentação dos repositórios e a primeira atividade é a identificação dos stakeholders. As informações sobre os stakeholders serão armazenadas na base Stakeholders que serão acessadas através da tela exibida na figura 4.1, onde o campo *Experiência* corresponde a quantidade de anos de experiência que o usuário tem na regra de negócio do domínio. Os engenheiros de requisitos podem fazer comentários sobre os usuários para facilitar o entendimento entre os interesses pessoais do usuário e o objetivo do projeto como um todo.

Cadastro Usuário

Código	1
Nome	Paulo de Tarso
Profissão	Analista
Setor trabalho	Centro de Engenharia de Software
Posição	Suporte a rede de comunicação
Telefone contato	221-2121; 9979-7997
Experiência	10
Horário trabalho	13:00 as 19:00
Relação c/projeto	Relacionar requisitos da rede interna da empresa
E-mail	paulotarso@empresa.com.br
Comentário	Stakeholder útil para testar os aplicativos de segurança

Registro 1 de 2

Figura 4.1 – Cadastro dos Stakeholders

Uma vez cadastrado os usuários e identificado, por meio deles, o domínio e as ramificações que compõem a estrutura do projeto, devem ser cadastradas as informações da base do documento de visão através das telas mostradas nas figuras 4.2, 4.3 e 4.4.

Cadastro Domínio

Código	1
Nome	Segurança de Rede
Descrição	Deve prover a segurança completa da rede da empresa
Objetivo	prover segurança para todo e qualquer acesso a rede
Necessidade	Garantir a confiabilidade dos dados da empresa
Característica	Gerenciar toda entrada, saída e movimento interno da rede
Custo Compra	R\$
Despesa Ano	R\$ 2.000,00
Comentário	O custo pode ser revisto de acordo com o tempo do projeto. Pode

Registro 1 de 4

Figura 4.2 – Cadastro do Domínio

IASCRE - Microsoft Internet Explorer
 File Edit View Favorites Tools Help
 Address C:\IASCRE\3 - Subdominio.htm

Cadastro Subdomínio

Código Domínio Segurança de Rede

SUBDOMÍNIO

Código

Nome

Característica

Custo compra

Despesa ano

Comentário

Registro 1 de 9

Figura 4.3 – Cadastro do Subdomínio

Os campos intitulados ‘Custo Compra’ servem para informar o valor máximo que a empresa pode pagar. Se o valor for informado dentro do domínio, a soma dos valores informados nos subdomínios não pode ultrapassar o valor do domínio. Se informar valor para o subdomínio, a soma dos valores informados nas aplicações não pode ultrapassar o valor do subdomínio.

IASCRE - Microsoft Internet Explorer
 File Edit View Favorites Tools Help
 Address C:\IASCRE\4 - Aplicação.htm

Cadastro Aplicação

Código subdomínio

APLICAÇÃO

Código

Nome

Característica

Custo compra

Despesa ano

Comentário

Registro 1 de 14

Figura 4.4 – Cadastro da Aplicação

Os campos intitulados ‘Despesa ano’ servem para informar o valor máximo que a empresa pode gastar por ano com COTS dentro da estrutura de domínio/subdomínio/aplicação.

A metodologia ASCRE orienta que os requisitos fundamentais sejam elicitados em primeiro lugar. Para que se cadastre esses requisitos e se identifique o nível deles (Fundamental, Contratual e Desejável), o IASCRE criou a tela exibida na figura 4.5.

Cadastro de Requisito	
Código	210
Descrição	Limitar o acesso de clientes web a determinadas páginas
Tipo	Fundamental
Nível	Fundamental
Fonte	Instituição
Justificativa	Maximização da produtividade dos funcionários
Objetivo	Não permitir que os usuários utilizem paginas improprias para o
Teste ? (s/n)	Sim
Descrição do teste	Logar em uma pagina não habilitada pela empresa
Testar após requisito	209
Validar com	COTS
Comentário	A medida que o usuário se concentra mais no trabalho a sua

Figura 4.5 – Cadastro de Requisito

Repare na figura acima que o requisito não pertence a um domínio ou subdomínio ou aplicação, isto ocorre para se tentar reutilizar um requisito. Por exemplo: para uma aplicação que consulta o saldo de uma conta corrente, um requisito fundamental é a senha do usuário, mas esse requisito também é necessário para uma aplicação de saque. Portanto, o IASCRE implementa uma tela para cada relacionamento do requisito com estrutura do projeto. As figuras 4.6, 4.7 e 4.8 exibem as telas que relaciona os requisitos, que ficam armazenados na base de Especificação de Requisito de Software (ERS), com a estrutura criada na base do Documento de Visão. Lembrando que, as telas obedecem as regras estabelecidas pelo ASCRE na seção 3.2.1, que trata do relacionamento dos requisitos com a estrutura do Documento de Visão (Domínio/Subdomínio/Aplicação).

IASCRE - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Relação Requisito vs Domínio

Código Domínio

Código Requisito

Comentário

Registro 3 de 5

Done My Computer

Figura 4.6 – Relação Requisito X Domínio

IASCRE - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Relação Requisito vs Subdomínio

Código Subdomínio

Código Requisito

Comentário

Registro 1 de 3

Done My Computer

Figura 4.7 – Relação Requisito X Subdomínio

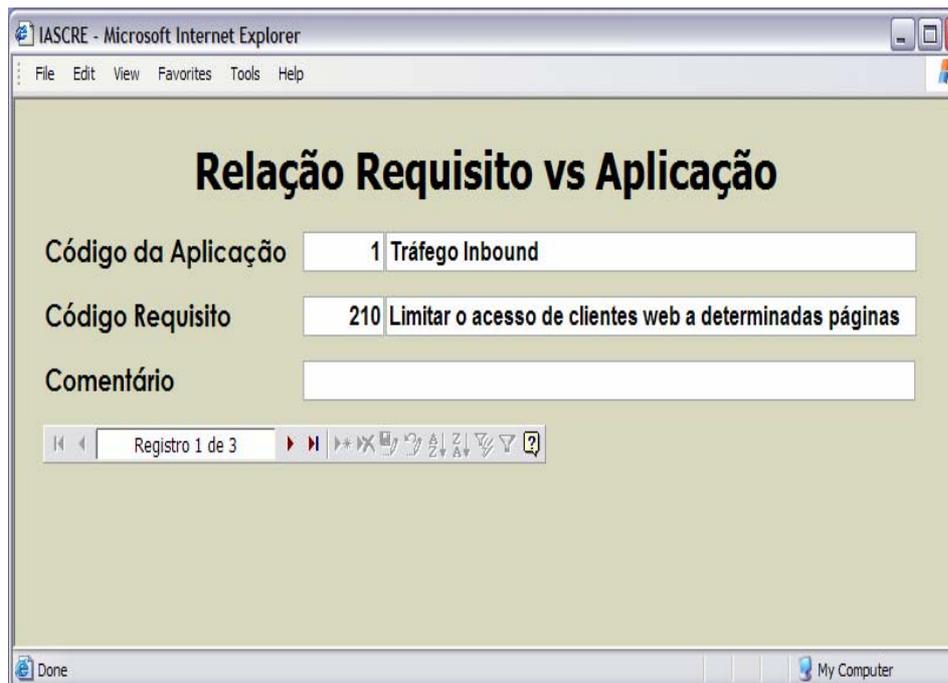


Figura 4.8 – Relação Requisito X aplicação

A próxima atividade do método é definir o critério de avaliação dos concorrentes no domínio em questão. O IASCRE criou quatro telas para se definir os parâmetros de avaliação. A primeira tela é que define a eliminação imediata de um COTS. Logicamente que a atividade de negociação deve ser executada para que se caracterize a eliminação do candidato. A figura 4.9 exhibe os parâmetros críticos para a eliminação de um COTS.

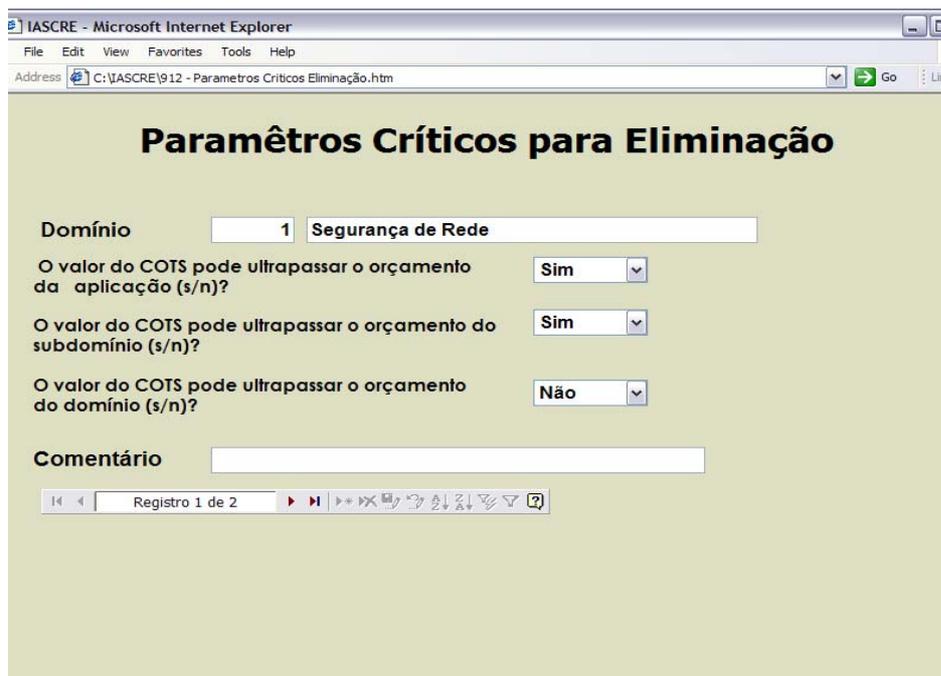


Figura 4.9 – Parâmetros Críticos de Eliminação de Candidatos

Veja que as figuras 4.2, 4.3 e 4.4 exibem os valores de orçamento que se refere a tela da figura acima, no campo intitulado ‘Custo Compra’.

A avaliação do fornecedor é a segunda tela do critério de avaliação implementada pelo IASCRE. Essa tela (figura 4.10) define a pontuação que um fornecedor receberá para cada cliente que o avaliar. No caso do exemplo da figura 4.10, um cliente que avaliar o fornecedor concorrente com satisfação ‘Ótima’, adicionará 10 pontos em sua pontuação de ‘Fornecedor Avaliado pelo Cliente’, para cada mês de contrato entre as partes, ou seja, um fornecedor concorrente que tiver um contrato de um ano com um cliente que o avaliou com satisfação ‘Ótima’, receberá mais 120 pontos. O campo ‘Pontuação inicial’ é utilizado para fornecer uma pontuação para o fornecedor independente da unidade de tempo do contrato do cliente, mas apenas pelo fato de possuir ou ter possuído um contrato com esse cliente.

The screenshot shows a web browser window titled 'IASCRE - Microsoft Internet Explorer'. The address bar shows 'C:\IASCRE\6 - CACFOR.htm'. The main content area has a title 'Critério Avaliação Fornecedor'. Below the title are several form fields:

- Código domínio: 1
- Segurança de Rede: (empty text box)
- Satisfação cliente: Ótima
- Tipo operação: +
- Quantidade pontos: 10
- Unidade de Tempo: Mês
- Pontuação inicial: 0
- Comentário: Para cada mês que o cliente tiver satisfeito o fornecedor

At the bottom of the form, there is a navigation bar with 'Registro 1 de 2' and several icons for navigation.

Figura 4.10 – Critério de Avaliação do Fornecedor

A próxima tela é semelhante a anterior, só que agora o cliente irá avaliar o produto candidato. Acrescentou-se a essa tela, exibida na figura 4.11, a similaridade de uso, pois um COTS pode ser usado de varias formas por seus clientes, por exemplo: um editor de texto é um COTS que pode ser utilizado por um cliente para digitar programas em Java e utilizado por outro cliente para digitar monografia, logo, o editor pode ser muito bom para digitar os programas, mais insuficiente para a monografia. Sendo assim, o campo intitulado

‘Similaridade de uso’ serve para fazer a pontuação diferencial para COTS com similaridade de uso diferente.

Critério Avaliação Cliente X COTS

Código domínio	1	Segurança de Rede
Satisfação cliente	Média	
Similaridade uso	Semelhante	
Tipo operação	+	
Quantidade pontos	5	
Unidade de Tempo	Mês	
Pontuação inicial	0	
Comentário	Para cada mês de uso do COTS pelo cliente, receberá 3	

Registro 2 de 2

Figura 4.11 – Critério de Avaliação do COTS pelo seu Cliente

A última, e talvez a mais importante, tela do critério de avaliação é o critério de avaliação dos requisitos, mas para se abordar essa tela considera-se a seguinte situação: um requisito pode ser contemplado plenamente por um COTS, esse é o estado ideal, todavia, o COTS concorrente pode atender um requisito parcialmente, ou não atender, ou até mesmo não está implementado no momento, mas ser implementado posteriormente pelo fornecedor, ou aberto para o próprio cliente implementar. Por causa dessas variações, o IASCRE implementa a tabela de tipo de serviço que um COTS oferece em relação a um determinado requisito (figura 4.12). Essa tabela é usada tanto no critério de avaliação de um requisito, quanto na relação do requisito com os concorrentes. Sendo assim, a figura 4.13 exibe a tela que define o critério de avaliação dos COTS em relação a um requisito, dependendo do nível e do tipo de serviço implementado. O exemplo mostrado na figura 4.13 define que no domínio ‘Segurança de Rede’ para cada requisito do nível ‘Fundamental’ que já estiver implementado pelo concorrente adicionará vinte pontos a sua pontuação de avaliação de requisitos.

IASCRE - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address C:\IASCRE\S - Servico.htm

Tabela de Tipo de Serviço em Relação ao Requisito

Código

Descrição

Comentário

Registro 1 de 2

Figura 4.12 – Tabela do Tipo de Serviço

IASCRE - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address C:\IASCRE\S - Critério Avaliação Requisito.htm

Critério de Avaliação dos Requisitos

Domínio

Nível de Requisito

Tipo de Serviço

Pontuação

Comentário

Registro 3 de 5

Figura 4.13 – Critério de Avaliação dos Requisitos

Ainda na fase de Preparação (ver seção 3.2.1) o método ASCRE orienta a alimentação da base de Fornecedor, que é alimentada pelo IASCRE com algumas telas. A figura 4.14 exibe a primeira tela, intitulada de ‘Cadastro de Empresa’. O IASCRE considera que o cliente e o fornecedor possuem informações comuns, por isso implementou uma única tela para alimentar esses dados. Para se saber qual é a empresa cliente e qual é a fornecedora, identifica-se primeiro na tela de cadastro dos produtos candidatos, exibida aqui na figura 4.15. Essa tela (figura 4.15) fornecer informações básicas do produto candidato, mas que podem definir a sua eliminação do pleito, pois o ‘Custo Compra’ definido nos parâmetros críticos de eliminação (figura 4.9) pode exigir que o valor seja inferior ao pretendido pelo concorrente. Outra informação importante, para uma avaliação visual, são os tipos de fontes (aberto, semi-aberto, fechado). Repare que um COTS pertence a uma única empresa, logo, algumas informações sobre a empresa podem ser obtidas na tela ‘Cadastro de Empresa’ (figura 4.14).

Código	<input type="text" value="1"/>
Nome	<input type="text" value="ALCA Engenharia de Software Ltda"/>
Tamanho	<input type="text" value="Grande"/>
Ramo atividade	<input type="text" value="Desenvolvimento de Software"/>
CEP	<input type="text" value="65060020"/>
Endereço	<input type="text" value="Rua das aves, 450, Bairro Floresta"/>
Telefone	<input type="text" value="98(xx)3221-2221"/>
Pessoa de contato	<input type="text" value="Pedro Paulo"/>
E-mail	<input type="text" value="pedropaulo@alca.com.br"/>
Data da fundação	<input type="text" value="25/1/1975"/>
Endereço pagina	<input type="text" value="www.alca.es.com.br"/>
Situação atual	<input type="text" value="Ótima"/>
Comentários	<input type="text" value="Já possuímos produtos desta empresa"/>

Figura 4.14 – Cadastro de Empresa

O ‘Cadastro de COTS’ (figura 4.15) armazena a última versão do produto válida no mercado. Essa informação é útil para se perceber a maturidade do produto. O campo intitulado ‘Custo adicional’ serve para informar quanto a instituição precisará gastar para implementar esse produto, por exemplo: o servidor atual da organização precisará expandir a sua memória para suportar o COTS e o custo para essa expansão é tantos reais.

The screenshot shows a web browser window titled 'IASCRE - Microsoft Internet Explorer'. The address bar shows 'C:\IASCRE\COTS.htm'. The main content area has a title 'Cadastro COTS' and a form with the following fields:

Código empresa	1	ALCA Engenharia de Software Ltda
COTS		
Código	1	
Sigla	SISER	
Nome	Sistema de Segurança de Rede	
Versão ofertada	7.3.2	
Custo compra	R\$ 60.000,00	
Despesa ano	R\$ 1.100,00	
Custo adicional	R\$ 10.629,35	
Tipo de fonte	Semi-aberto	
Características	Pacote de segurança de rede	
Comentário	Contempla toda a segurança interna e externa	

At the bottom of the form, there is a navigation bar showing 'Registro 1 de 2' and various icons for navigation and search.

Figura 4.15 – Cadastro de Produtos Candidatos

A Checagem é a segunda fase do ASCRE (ver seção 3.3) e tem como atividade mais importante o processo de negociação dos requisitos com os candidatos. O IASCRE colocou no processo de negociação duas telas para poder pontuar os candidatos em conformidade com os seus clientes, atuais ou não. A figura 4.16 mostra o fornecedor ‘ALCA Engenharia de Software Ltda’ sendo avaliado pelo seu cliente ‘Secretaria de Educação da PMSL’, onde a satisfação já está previamente definida no critério de avaliação (ver figura 4.10) e as datas correspondem ao tempo que as duas empresas se relacionam. Esse tempo de relacionamento influencia na pontuação definida no critério de avaliação. O mesmo acontece na figura 4.17, a ‘Secretaria de Educação da PMSL’ avaliando o COTS 1 com similaridade de uso ‘igual’ a instituição que está a procura do COTS. O tempo entre a instalação e o fim do uso serve, para contabilizar os pontos definidos na tela de critério de avaliação exibida na figura 4.11. Quando o campo intitulado ‘Fim do Uso’ for branco, indica que até a presente data o COTS ainda é utilizado pelo cliente.

IASCRE - Microsoft Internet Explorer
 File Edit View Favorites Tools Help
 Address C:\IASCRE\910 - Avaliação Fornecedor.htm

Avaliação do Fornecedor pelo Cliente

Fornecedor

Cliente

Data Inicial

Data Final

Satisfação

Comentário

CLIForWiz 1 de 2

Figura 4.16 – Cliente Avaliando o Fornecedor do COTS

IASCRE - Microsoft Internet Explorer
 File Edit View Favorites Tools Help
 Address C:\IASCRE\911 - Avaliação COTS.htm

Avaliação COTS pelo Cliente

COTS

Cliente

Similaridade

Instalacao

Fim do Uso

Satisfação

Comentário

Registro 1 de 2

Figura 4.17 – Cliente Avaliando o COTS Candidato

Um requisito pode ser solicitado não para o produto, mas para o fornecedor, como por exemplo: o fornecedor deve estar no mercado a mais de dez anos. Visando isso, O IASCRE colocou no processo de negociação a tela que relaciona os requisitos com o fornecedor do produto (figura 4.18). A figura 4.18 exibe o requisito de numero 9057 sendo atendido pela empresa ‘ALCA Engenharia de software Ltda’ com tipo de serviço pronto, ou seja, a empresa já atende plenamente esse requisito.



Figura 4.18 – Negociação do Requisito com o Fornecedor

A ultima tela do processo de negociação é a relação entre o requisito da aplicação e o COTS candidato. Essa tela, mostrada na figura 4.19, tem as mesmas características da tela da figura acima, sendo que, no processo de tomada de decisão as pontuações obtidas por estas negociações são computadas diretamente para o COTS.

O exemplo da figura 4.19 mostra que o COTS de número 1 não atende o requisito de número 1, mas ele pode ser implementado pela organização que comprar o produto, fazendo uma previsão de seis meses para essa implementação.

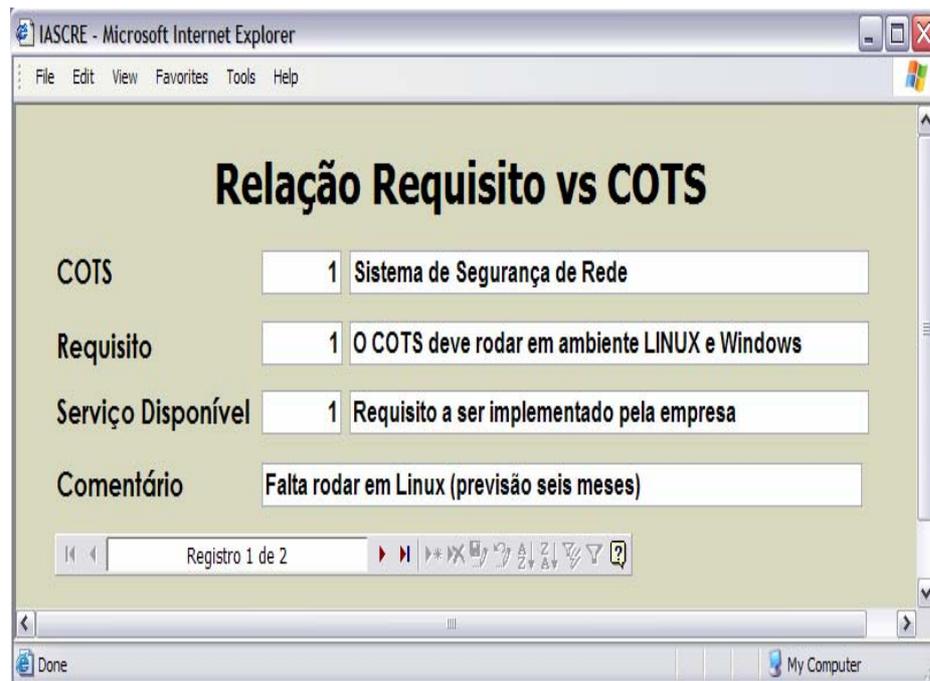


Figura 4.19 – Negociação do Requisito com o COTS

O IASCRE implementa a terceira fase da metodologia ASCRE, fase de Decisão, de forma *batch*, como a própria metodologia sugere, onde o primeiro processo é verificar qual é a pontuação máxima possível que um COTS pode atingir dentro do critério de avaliação definido na tela da figura 4.13. Aqui o processo irá acessar as informações das telas exibidas nas figuras 4.6, 4.7 e 4.8, para consolidar esses valores. A figura 4.20 mostra um exemplo da consolidação dos valores máximos possíveis para um projeto de segurança de rede. Veja que esses valores estão divididos em níveis de requisitos (Fundamental, Contratual e Desejável) e dentro da estrutura definida no Documento de Visão (domínio, subdomínio e aplicação). No exemplo da figura 4.20, fica claro que o subdomínio 2, Inspeção de Conteúdo, não possui requisito diretamente para ele, mas a sua aplicação possui requisitos que totalizam 1.200 pontos para o nível Fundamental, 750 para o nível Contratual e 650 para o nível Desejável. O processo *batch* que atingi esses valores, identifica qual é maior pontuação definida na tela da figura 4.13, dentro de cada nível de requisito, e multiplica esse valor pela quantidade de requisito dentro da estrutura do Documento de Visão.

O próximo processo *batch* consolida, com as mesmas características do anterior, as informações armazenadas a partir das telas exibidas nas figuras 4.18 e 4.19 e pontua de acordo com o critério de avaliação visto na figura 4.13.

A ultima atividade é a filtragem que acontece, também, com processo *batch* que irá comparar a pontuação atingida por cada COTS e a pontuação máxima possível (ver figura 4.20). Essa comparação irá trazer resultado em valores percentuais, para se saber o percentual que cada COTS atingiu em relação às necessidades dos usuários (requisitos).

 Universidade Federal do Maranhão – UFMA Centro de Ciências Exatas e Tecnologia - CCET Curso de Pós-Graduação em Engenharia de Eletricidade – CPGE E.			
Demonstração dos Valores Máximos Possíveis			
	Fundamental	Contratual	Desejável
Seguranca de Rede	200	100	50
1 Firewall	300	250	150
2 Tráfego Outbound	300	100	100
1 Tráfego Inbound	800	400	100
2 Inspeção de Conteúdo	0	0	0
3 Gerente de Sites	1.200	750	650
3 Detector de Intrusão	400	200	300
6 Anti-virus	700	500	100
5 Tentava de Invasão	300	100	300
4 Atividade Suspeita	800	950	100

Figura 4.20 – Valores Máximos Possíveis para um COTS

A equipe de engenheiros de requisitos pode usar informações disponíveis nas bases para avaliar o resultado obtido.

Os dados processados pelos processos *batch* são armazenados na base de Resultado (ver figura 3.1).

4.2 Comentários

Este capítulo mostrou o protótipo de uma implementação de uma ferramenta com as diretrizes e funcionalidades do método ASCRE. O protótipo possui vinte e nove tabelas divididas logicamente nas bases de Documento de Visão (3 tabelas), Stakeholders (1 tabela), Fornecedor (4 tabelas), ERS (12 tabelas) e Resultado (9 tabelas). Portanto, a pretensão desta implementação foi demonstrar a viabilidade e eficácia da metodologia.

A implementação do ASCRE traz para uma organização vantagens como:

- ❖ Base de informações sobre o mercado e seus fornecedores;
- ❖ Base de informações sobre empresas que trabalham no mesmo ramo de negócio;
- ❖ Flexibilidade para negociação;
- ❖ Possibilidade de disponibilizar a informação para todos os interessados;
- ❖ Possibilidade de reutilização de dados;
- ❖ Critério de avaliação definido de acordo com a realidade de cada projeto;
- ❖ Melhor performance no processo de avaliação e seleção;
- ❖ Fundamentação criteriosa na hora da tomada de decisão.

Capítulo 5

ESTUDO DE CASO

Este capítulo fornece um exemplo fictício utilizando um problema de gerenciamento de rede de uma organização para demonstrar a eficiência do ASCRE no processo de tomada de decisão, considerando múltiplas escolhas para a solução de um domínio.

5.1 Objetivos

O principal objetivo deste exemplo é demonstrar a eficiência do ASCRE na seleção de COTS. Portanto, não será exibido neste exemplo os processos detalhados de elicitação e análise de requisitos, assim como outras atividades menos relevantes para o objetivo deste estudo de caso.

Tomando como exemplo o domínio na área de segurança de rede, onde uma empresa deseja uma solução integrada de proteção para sua rede, será aplicado o ASCRE neste estudo de caso utilizando a arquitetura implementada no capítulo 4.

5.1.1 Preparação

Um dos objetivos desta fase é alimentar o Documento de Visão, para isso é preciso identificar o domínio do problema, que neste caso é a segurança nos acessos interno e externo da rede da instituição.

Com o domínio identificado, o passo seguinte é elicitar, junto aos *stakeholders* identificados, os requisitos fundamentais.

Como já foi visto, o ASCRE identifica os requisitos como sendo de domínio, subdomínio ou aplicação. Para fazer esse armazenamento a equipe de projeto precisa mapear o domínio a partir das informações adquiridas. Será feito um mapeamento a partir das seguintes informações:

❖ Objetivo

- Aquisição de um Pacote de Software de Segurança de Redes de Computadores.

❖ Características Gerais

- O Pacote deve contemplar sistemas com funcionalidades de Firewall, Detector de Intrusão, Analisador de Conteúdo e Antivírus corporativo.

❖ Firewall

- É um sistema que impõe os requerimentos de tráfego em uma rede Internet/Intranet, para permitir ou impedir o acesso à dados compartilhados na mesma. O firewall baseia-se em uma série de critérios de aplicativos, serviços de rede, endereços de origem e destino e portas de comunicação. Ele filtra os pacotes de entrada e saída na rede e aplica as regras que são previamente definidas em sua configuração.

❖ Tráfego Inbound

- É o tráfego de dados de fora para dentro, geralmente utilizado quando um cliente remoto quer acessar alguma informação dentro da rede, ou é um cliente VPN (Virtual Private Network) ou quando está tentando acessar uma página de Internet que esteja disponível em um Servidor de Página que esteja dentro da rede corporativa. Antes de permitir o acesso inbound na rede é necessário que se aplique regras de tráfego, ou seja, especificar quais são as portas de comunicação, serviços e protocolos que estarão abertas para acesso externo e quais clientes poderão ter acesso à rede interna.

❖ Tráfego Outbound

- É o tráfego de dados de dentro para fora, utilizado quando os clientes da rede corporativa querem acessar dados fora da rede. Por exemplo, acesso à Internet ou aplicativos externos à rede. Também é necessário especificar as portas de comunicação e os serviços que serão disponíveis para acesso à rede externa. Serão especificados os usuários, protocolos permitidos, serviços, etc. Possuem geralmente disponíveis as portas de tráfego HTTP, FTP, SMTP e POP.

❖ Detector de Intrusão

- Com a permissão de acessos externos à rede, tem-se conseqüentemente porta de comunicação aberta, o que cria possibilidades de intrusão. Para inibir que as informações da rede interna possam ser acessadas por pessoas

não autorizadas ou mesmo, que dados sejam rackeados é necessário a utilização de detectores de intrusão. O sistema detector de intrusão tem o objetivo de analisar as atividades de rede e no caso de atividades suspeitas ou tentativas de invasão tomar atitudes de impedimento de tais ações.

❖ Atividades Suspeitas

- Caracteriza-se por atividades de usuários internos da rede que se assemelha com intrusão, como por exemplo, grande quantidade de pacotes SNMP enviados pela rede, várias tentativas de logon sem êxito, etc. Os logs destas atividades devem ser armazenados em uma base de dados SQL Server

❖ Tentativas de Invasão

- Quando as tentativas de acesso ocorrerem de fora para dentro da rede, deve ser avaliado se a rede não está sendo invadida. Para isso deve-se analisar em um banco SQL, modelos de tentativas de invasão para que sejam comparadas com o tráfego inbound. Caso alguma atividade se assemelhe com invasão, medidas de segurança devem ser tomadas para interromper a conexão externa com a rede.

❖ Antivírus

- Para garantir a segurança contra vírus na rede corporativa, é necessário que esta esteja protegida com um antivírus corporativo. Este antivírus deverá fazer consulta das assinaturas dos vírus existentes e efetuar varreduras periódicas nos computadores ligados à rede.

❖ Inspeção de conteúdo

- Como é permitido o acesso à Internet por usuários internos da rede, deve-se garantir que as páginas e os serviços web acessados sejam próprios para o ambiente de trabalho. Ou seja, limitar o acesso de clientes web a determinadas páginas, bem como limitar a largura de banda para cada usuário. Também, deve ser especificado em uma lista, todos os sites

permitidos para acesso, bem como os horários de acesso e os usuários que poderão ou não acessar estas páginas.

- Todos os sites proibidos devem ser especificados em uma lista separada. Ao tentar acessar uma página web proibida, o usuário receberá uma mensagem de acesso negado. Devido o grande número de páginas impróprias existentes e a velocidade com que elas são criadas, o fornecedor do sistema deverá fornecer a possibilidade de downloads periódicos desta lista de proibição em seu site.

A figura 5.1 mostra o mapeamento baseado nessas informações. O Documento de Visão é alimentado seguindo o mapeamento do domínio, onde se têm três subdomínios e seis aplicações.

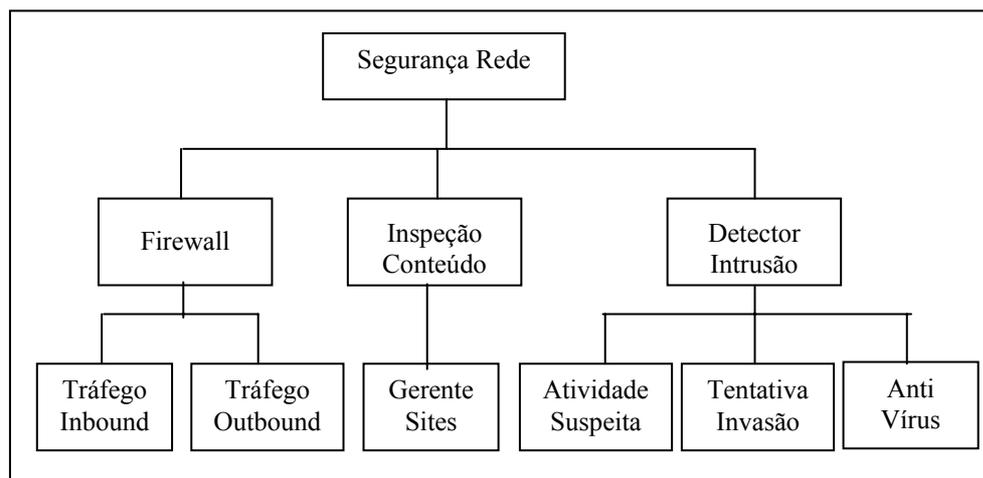


Figura 5.1 – Diagrama do Documento de Visão

A figura 5.2 mostra o resultado do cadastramento do Documento de Visão.

DOM_codigo	DOM_nome	DOM_descrição
1	Segurança de Rede	o pacote deve contemplar sistemas com funcionalidade
	SBD_codigo	SBD_nome
		SBD_características
1	Firewall	impõe os requerimentos de tráfego em uma rede int
	APL_codigo	APL_nome
		APL_características
+	1	Tráfego Inbound
+	2	Tráfego Outbound
*	0	
2	Inspeção de Conteúdo	garantir que as paginas e os serviços web acessad
	APL_codigo	APL_nome
		APL_características
+	3	Gerente de Sites
*	0	
3	Detector de Intrusão	inibir que as informações da rede interna possam s
	APL_codigo	APL_nome
		APL_características
+	4	Atividade Suspeita
+	5	Tentativa de Invasão
+	6	Antivírus
*	0	

Figura 5.2 – Resultado do Cadastramento do Documento de Visão

Como já foi mencionado, este exemplo é fictício e não tem a pretensão de demonstrar todas as fases da engenharia de requisitos. Então, se pode imaginar que os requisitos foram codificados e cadastrados.

A atividade seguinte é definir o critério de avaliação. Para este exemplo não será considerada a avaliação do fornecedor/produto em relação aos seus clientes, pois o exemplo se tornaria muito extenso e complexo. Portanto será definido apenas o critério de avaliação para os requisitos. Antes de definir o critério é necessário definir os tipos de serviços possíveis.

Os serviços possíveis que um fornecedor/produto pode atender, neste exemplo, em relação a um requisito são:

- 1 - pronto;
- 2 - customizado pelo fornecedor; e
- 3 - customizado pela organização.

Com esta informação é definido o critério de avaliação identificando-se a pontuação para cada nível de requisito em relação ao tipo de serviço, conforme mostra a Tabela 5.1. Repare que os níveis têm a mesma pontuação para os tipos de serviços, mas isso não afetará o resultado, porque os níveis são hierarquizados na hora da filtragem.

Tipo Serviço \ Requisito	Fundamental	Contratual	Desejável
1 - Pronto	10	10	10
2 - Customizado pelo Fornecedor	5	5	5
3 - Customizado pela Organização	1	1	1

Tabela 5.1 – Critério de Avaliação

Até aqui as bases do Documento de Visão, Stakeholder e ERS foram alimentadas. Falta agora alimentar a base do fornecedor. Para tanto, imagina-se que foram localizados oito candidatos com características para atender o domínio e suas ramificações. Passa-se então para a fase seguinte.

5.1.2 Checagem

Como já foi visto, nesta fase se faz a negociação e a elicitação dos outros níveis de requisitos. A demonstração deste processo seria extensa, pois se relacionaria uma quantidade muito grande de requisitos com cada um dos candidatos identificados na fase anterior. Sem levar em consideração, que a ERS pode ser modificada para se adequar à realidade de mercado.

A figura 5.3 exibe um COTS se relacionando com os requisitos e o tipo de serviço que o produto atende, onde a empresa 1 com o COTS 1 atende aos requisitos 1,2 e 9 com o tipo de serviço 1 – pronto. O requisito 6 não está pronto, mas pode ser customizado e o requisito 8 não está pronto e nem é de interesse da empresa fazer a customização, mas o aplicativo permite que a instituição faça a customização.

EMP_codigo	COT_codigo	COT_sigla	COT_nome	COT_versao	COT_compr
1	1	COTS 1	produto candidato 1		R\$ 0,00
	REQ_codigo	SVC_codigo	REC_comentario		
	1	1			
	2	1			
	6	2			
	8	3			
	9	1			
2	2	COTS 2	produto candidato 2		R\$ 0,00
	REQ_codigo	SVC_codigo	REC_comentario		
	3	1			
	4	2			
	6	1			
	7	1			
	8	2			
3	3	COTS 3	produto candidato 3		R\$ 0,00
	REQ_codigo	SVC_codigo	REC_comentario		
	1	1			
	2	2			

Figura 5.3 – COTS e os Requisitos da ERS

Após todos os requisitos devidamente cadastros e negociados com cada fornecedor/produto, passa-se para a fase de decisão. A próxima fase receberá as quatro bases de dados (Documento de Visão, Stakeholders, Fornecedor e ERS) alimentadas com informações suficientes para uma tomada de decisão. A tabela 5.2 mostra a quantidade de requisitos para cada item do Documento de Visão, perfazendo um total de 1.020 (mil e vinte requisitos) requisitos.

DOCUMENTO DE VISÃO	QUANTIDADE DE REQUISITOS E.R.S.		
	Fundamental	Contratual	Desejável
DOM 1 – Segurança de Rede	20	10	5
SBD 1 – Firewall	30	25	15
APL 1 - Tráfego Inbound	80	40	10
APL 2 - Tráfego Outbound	30	10	10
SBD 2 - Inspeção Conteúdo	0	0	0
APL 3 - Gerente Sites	120	75	65
SBD 3 - Detector Intrusão	40	20	30
APL 4 - Atividade Suspeita	80	95	10
APL 5 - Tentativa Invasão	30	10	30
APL 6 - AntiVírus	70	50	10

Tabela 5.2 – Quantidade de Requisitos por Item

5.1.3 Decisão

Com as quatro bases consolidadas, a fase de decisão irá executar os processos que determinam o resultado da seleção. Esta fase é totalmente automatizada, fazendo com que as atividades ganhem em tempo e precisão.

Como o resultado final é fornecido em percentual, é preciso identificar os valores máximos possíveis que um produto pode alcançar no certame. Para isso, é executado o primeiro processo *batch* desta fase. O resultado desse processo é a multiplicação da quantidade de requisitos encontrado na relação Documento de Visão versus ERS, pelo maior valor da Tabela 5.1 em cada nível de requisito. A tabela 5.3 mostra o valor máximo possível, onde o domínio no nível fundamental tem pontuação máxima 200, porque a maior pontuação para o nível fundamental é dez e a quantidade de requisitos nesse item são vinte, conforme mostra tabela 5.2.

DOCUMENTO DE VISÃO	VALOR MÁXIMO p/ PONTUAÇÃO		
	Fundamental	Contratual	Desejável
DOM 1 – Segurança de Rede	200	100	50
SBD 1 – Firewall	300	250	150
APL 1 - Tráfego Inbound	800	400	100
APL 2 - Tráfego Outbound	300	100	100
SBD 2 - Inspeção Conteúdo	0	0	0
APL 3 - Gerente Sites	1200	750	650
SBD 3 - Detector Intrusão	400	200	300
APL 4 - Atividade Suspeita	800	950	100
APL 5 - Tentativa Invasão	300	100	300
APL 6 - AntiVírus	700	500	100

Tabela 5.3 – Valor máximo para pontuação em cada item

O próximo passo é executar o segundo processo *batch*, para encontrar os valores de cada produto candidato. Como na primeira fase foram encontrados oito candidatos, eles serão chamados de C1, C2, C3, C4, C5, C6, C7, C8. A tabela 5.4 exhibe uma simulação do resultado da execução desse processo.

ITENS DO PROJETO		VALORES DOS CANDIDATOS							
		C1	C2	C3	C4	C5	C6	C7	C8
DOM 1 – Segurança de Rede	F	200	200	200	180	200	150	200	185
	C	100	100	100	100	90	100	100	95
	D	50	50	50	50	45	45	50	50
SBD 1 – Firewall	F	300	300	280	300	250	300	300	290
	C	250	250	250	250	200	250	250	240
	D	150	150	100	100	150	150	150	130
APL 1 - Tráfego Inbound	F	800	800	650	700	800	790	800	790
	C	400	400	300	350	350	390	400	400
	D	100	100	100	80	100	90	100	90
APL 2 - Tráfego Outbound	F	290	280	290	250	300	290	300	300
	C	100	100	90	80	100	100	100	100
	D	100	80	90	100	100	100	100	100
SBD 2 - Inspeção Conteúdo	F	0	0	0	0	0	0	0	0
	C	0	0	0	0	0	0	0	0
	D	0	0	0	0	0	0	0	0
APL 3 - Gerente Sites	F	1160	1190	1200	1150	1000	1180	1200	1150
	C	740	750	750	720	740	750	730	620
	D	650	630	640	640	600	570	580	530
SBD 3 - Detector Intrusão	F	400	400	400	380	390	400	390	400
	C	200	200	200	180	190	200	200	200
	D	280	300	300	250	290	300	300	300
APL 4 - Atividade Suspeita	F	790	800	800	750	700	800	800	800
	C	950	950	900	750	700	900	950	950
	D	100	100	100	50	100	100	90	100
APL 5 - Tentativa Invasão	F	300	300	250	250	200	250	290	300
	C	90	100	100	50	100	100	100	100
	D	290	300	290	250	200	200	300	300
APL 6 - AntiVirus	F	700	700	690	600	500	600	700	700
	C	450	500	450	400	400	300	500	500
	D	50	100	90	100	100	90	100	100

Tabela 5.4 – Pontuação obtida pelos candidatos em cada item

Com a tabela acima, termina o processo de avaliação da base fornecedor. A próxima atividade é fazer o processo de filtragem do resultado das avaliações. Para executar o filtro é bom que se verifique antes a tabela 5.3, onde o valor máximo de cada item foi determinado pela quantidade de requisitos por item. Todavia, a metodologia ASCRE considera o problema a ser solucionado como uma árvore (domínio, subdomínio e aplicação), sendo assim, a pontuação da raiz deve ser o total da soma de todos os pontos do projeto por nível de requisito, ou seja, a pontuação do domínio é igual a sua pontuação mais a pontuação

dos outros itens no mesmo nível de requisito. Logo, o valor máximo do domínio para o nível Fundamental é a soma dos valores da coluna 2 da tabela 5.3, representado:

$$\text{DOM 1 [Fundamental]} = 200+300+800+300+0+1.200+400+800+300+700=5.000$$

O critério utilizado acima, também é utilizado para o subdomínio e aplicação. O subdomínio tem que atender os requisitos de domínio, logo a sua pontuação também deve considerar os pontos para o domínio. O produto que vai ser selecionado para o atender um subdomínio, tem que satisfazer os requisitos de suas aplicações. Portanto, o valor máximo para o nível Fundamental do subdomínio Firewall (SBD 1) é a soma do valor do DOM 1 mais o SBD 1 mais APL 1 mais APL 2 , da coluna 2 da tabela 5.3, representado :

$$\text{SBD 1 [fundamental]} = 200 + 300 + 800 + 300 = 1.600$$

Obedecendo a esse critério de soma dos pontos, a tabela 5.5 mostra os valores máximos possíveis que um produto pode atingir para ser selecionado como solução.

DOCUMENTO DE VISÃO	VALOR MÁXIMO PARA SELEÇÃO		
	Fundamental	Contratual	Desejável
DOM 1 – Segurança de Rede	5.000	3.350	1.850
SBD 1 – Firewall	1.600	850	400
APL 1 - Tráfego Inbound	1.300	750	300
APL 2 - Tráfego Outbound	800	450	300
SBD 2 - Inspeção Conteúdo	1.400	850	700
APL 3 - Gerente Sites	1.400	850	700
SBD 3 - Detector Intrusão	2.400	1.850	850
APL 4 - Atividade Suspeita	1.400	1.250	450
APL 5 - Tentativa Invasão	900	400	650
APL 6 - AntiVírus	1.300	800	450

Tabela 5.5 – Valor máximo para seleção por item

Repare que na tabela 5.3 o SBD 2, Inspeção de Conteúdo, não possui nenhum requisito, porém, os valores do domínio mais os valores do APL 3, Gerente Sites, influencia sua pontuação. Baseado no mesmo critério matemático que se chegou as pontuações da tabela 5.5, a tabela 5.6 exhibe a pontuação obtida pelos produtos para serem selecionados (confira com a tabela 5.4).

ITENS DO PROJETO		VALORES DOS CANDIDATOS							
		C1	C2	C3	C4	C5	C6	C7	C8
DOM 1 – Segurança de Rede	F	4940	4970	4760	4560	4340	4760	4980	4915
	C	3280	3350	3140	2880	2870	3090	3330	3205
	D	1770	1810	1760	1620	1685	1645	1770	1700
SBD 1 – Firewall	F	1590	1580	1420	1430	1550	1530	1600	1565
	C	850	850	740	780	740	840	850	835
	D	400	380	340	330	395	385	400	370
APL 1 - Tráfego Inbound	F	1300	1300	1130	1180	1250	1240	1300	1265
	C	750	750	650	700	640	740	750	735
	D	300	300	250	230	295	285	300	270
APL 2 - Tráfego Outbound	F	790	780	770	730	750	740	800	775
	C	450	450	440	430	390	450	450	435
	D	300	280	240	250	295	295	300	280
SBD 2 - Inspeção Conteúdo	F	1360	1390	1400	1330	1200	1330	1400	1335
	C	840	850	850	820	830	850	830	715
	D	700	680	690	690	645	615	630	580
APL 3 - Gerente Sites	F	1360	1390	1400	1330	1200	1330	1400	1335
	C	840	850	850	820	830	850	830	715
	D	700	680	690	690	645	615	630	580
SBD 3 - Detector Intrusão	F	2390	2400	2340	2160	1990	2200	2380	2385
	C	1790	1850	1750	1480	1480	1600	1850	1845
	D	770	850	830	700	735	735	840	850
APL 4 - Atividade Suspeita	F	1390	1400	1400	1310	1290	1250	1390	1385
	C	1250	1250	1200	1030	980	1000	1250	1245
	D	430	450	450	350	435	445	440	450
APL 5 - Tentativa Invasão	F	900	900	850	810	790	800	880	885
	C	390	400	400	330	380	400	400	395
	D	620	650	640	550	535	545	650	650
APL 6 - AntiVirus	F	1300	1300	1290	1160	1090	1150	1290	1285
	C	750	800	750	680	680	600	800	795
	D	380	450	440	400	435	435	450	450

Tabela 5.6 – Pontuação dos candidatos para seleção em cada item

Agora é possível equacionar os resultados para determinar o percentual que cada candidato atende do Documento de Visão. O candidato C1 atingiu 4.940 (quatro mil novecentos e quarenta) pontos para atender os requisitos fundamentais de todo o domínio (ver tabela 5.6 na linha DOM 1, coluna C1 para nível F), enquanto o valor máximo possível é de 5.000 (cinco mil) pontos (ver tabela 5.5 na linha DOM 1, coluna Fundamental – nível F), sendo assim, o candidato C1 atende o domínio em 98,8% (noventa e oito virgula oito por cento). A tabela 5.7 mostra o resultado da filtragem, em percentual, para o domínio.

CLASSIFICAÇÃO DO DOMÍNIO			
FUNDAMENTAL		CONTRATUAL	
C7	99,60	C2	100,00
C2	99,40	C7	99,40
C1	98,80	C1	97,91
C8	98,30	C8	95,67
C3	95,20	C3	93,73
C6	95,20	C6	92,24
C4	91,20	C4	85,97
C5	86,80	C5	85,67

Tabela 5.7 – Classificação geral do domínio Segurança de Rede

A tabela 5.7 mostra que na classificação geral C7 preencheu a maior parte dos requisitos Fundamentais. Todavia, não preencheu 100% do domínio, logo, não satisfaz plenamente o domínio da solução. Então a solução será vista através dos subdomínios.

A tabela 5.8 mostra que o SBD 1 é totalmente satisfeito pelo COTS C7. No entanto o SBD 1 é formado pelas aplicações APL 1 e APL 2. Na APL 1 os COTS C1 e C2 satisfazem plenamente os requisitos fundamentais, porém a APL 2 só é satisfeita totalmente por C7. Portanto, é selecionado C7 para solucionar o SBD 1 ou verificar a totalização dos dados contratuais para verificar se comprar duas soluções é mais viável que comprar somente uma.

O SBD 2 é satisfeito pelo C3 e C7 (ver tabela 5.9). O desempate é feito no nível contratual, onde o C3 satisfaz plenamente. Neste caso, o ASCRE aconselha a equipe de projeto avaliar os dados contratuais para verificar a possibilidade de comprar C7 para SBD1 e SBD2.

A tabela 5.10 mostra que somente C2 satisfaz plenamente os requisitos fundamentais do SBD 3. Todavia, o SBD 3 é formado pelas aplicações APL 4, APL 5 e APL 6. APL 5 e APL 6 são plenamente satisfeitas pelos COTS C1 e C2, a APL 5 é satisfeita pelos COTS C2 e C3. Deve-se, então, verificar os dados contratuais para decidir qual é a melhor solução: compra somente uma ou duas soluções para o SBD 3.

CLASSIFICAÇÃO											
FUNDAMENTAL						CONTRATUAL					
SBD 1		APL 1		APL 2		SBD 1		APL 1		APL 2	
C7	100	C1	100	C7	100	C1	100	C1	100	C1	100
C1	99,38	C2	100	C1	98,75	C2	100	C2	100	C2	100
C2	98,75	C7	100	C2	97,50	C7	100	C7	100	C6	100
C8	97,81	C8	97,31	C8	96,88	C6	98,82	C6	98,67	C7	100
C5	96,88	C5	96,15	C3	96,25	C8	98,24	C8	98,00	C3	97,78
C6	95,63	C6	95,38	C5	93,75	C4	91,76	C4	93,33	C8	96,67
C4	89,38	C4	90,77	C6	92,50	C3	87,06	C3	86,67	C4	95,56
C3	88,75	C3	86,92	C4	91,25	C5	87,06	C5	85,33	C5	86,67

Tabela 5.8 – Classificação para o subdomínio Firewall

CLASSIFICAÇÃO							
FUNDAMENTAL				CONTRATUAL			
SBD 2		APL 3		SBD 2		APL 3	
C3	100	C3	100	C2	100	C2	100
C7	100	C7	100	C3	100	C3	100
C2	99,29	C2	99,29	C6	100	C6	100
C1	97,14	C1	97,14	C1	98,82	C1	98,82
C8	95,36	C8	95,36	C5	97,65	C5	97,65
C4	95,00	C4	95,00	C7	97,65	C7	97,65
C6	95,00	C6	95,00	C4	96,47	C4	96,47
C5	85,71	C5	85,71	C8	84,12	C8	84,12

Tabela 5.9 – Classificação para o subdomínio Inspeção de Conteúdo

CLASSIFICAÇÃO															
FUNDAMENTAL							CONTRATUAL								
SBD 3		APL 4		APL 5		APL 6		SBD 3		APL 4		APL 5		APL 6	
C2	100	C2	100	C1	100	C1	100	C2	100	C1	100	C2	100	C2	100
C1	99,58	C3	100	C2	100	C2	100	C7	100	C2	100	C3	100	C7	100
C8	99,38	C1	99,29	C8	98,33	C3	99,23	C8	99,73	C7	100	C6	100	C8	99,38
C7	99,17	C7	99,29	C7	97,78	C7	99,23	C1	96,76	C8	99,60	C7	100	C1	93,75
C3	97,50	C8	98,93	C3	94,44	C8	98,85	C3	94,59	C3	96,00	C8	98,75	C3	93,75
C6	91,67	C4	93,57	C4	90,00	C4	89,23	C6	86,49	C4	82,40	C1	97,50	C4	85,00
C4	90,00	C5	92,14	C6	88,89	C6	88,46	C4	80,00	C6	80,00	C5	95,00	C5	85,00
C5	82,92	C6	89,29	C5	87,78	C5	83,85	C5	80,00	C5	78,40	C4	82,50	C6	75,00

Tabela 5.10 – Classificação para o subdomínio Detector de Intrusão

A figura 5.4 mostra o resultado final, onde três produtos são selecionados para serem testados em um ambiente operacional.

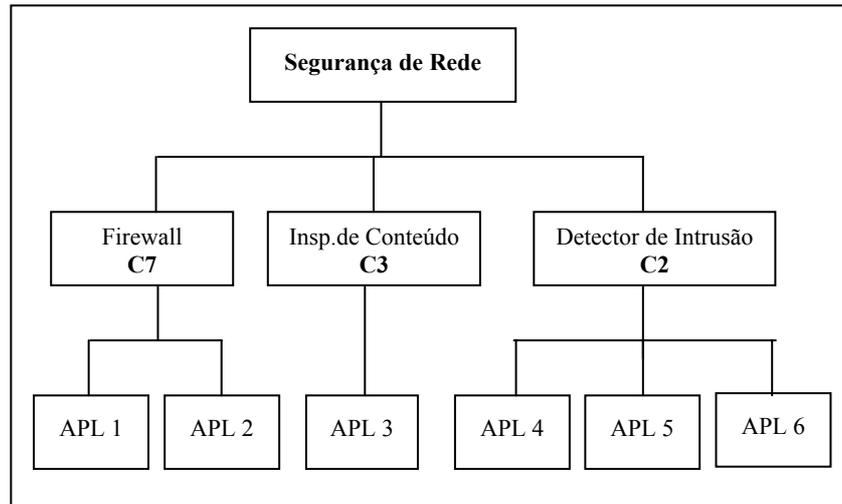


Figura 5.4 – Resultado Final da Seleção de Produtos

5.2 Comentários

Um dos objetivos da metodologia ASCRE é fornecer informações suficientes para uma tomada de decisão. Neste estudo de caso foi demonstrada a eficiência do método na utilização de critérios matematicamente definidos. O ASCRE fornece um resultado lógico dentro da definição da equipe de projeto. Certamente, outras questões precisam de avaliação, como por exemplos:

- ❖ Como integrar os componentes selecionados (C7, C3 e C2);
- ❖ Como gerenciar as versões futuras para não causar incompatibilidade entre elas.

Todavia, essas questões não precisam ser respondidas na fase de seleção, mas na fase posterior aos produtos selecionados, porque o objetivo principal da fase de avaliação/seleção é indicar produtos que preencham o maior número possível de requisitos fundamentais. Neste sentido, a metodologia ASCRE prova que é eficiente para ser usada em um processo de avaliação/seleção de produtos dentro da engenharia de requisitos.

Capítulo 6

CONCLUSÃO

O presente capítulo mostra as principais contribuições desta dissertação para o desenvolvimento baseado em COTS. Depois, sugere algumas pesquisas que poderão ser realizadas a partir deste trabalho.

6.1 Contribuição do Trabalho

A reutilização tem sido muito explorada na Ciência da Computação. Muitas pesquisas estão sendo realizadas para maximizar o reuso em todas as fases da engenharia de software, que vai da engenharia de requisitos à manutenção de software. O desenvolvimento baseado em COTS é uma forma de reutilização, com objetivo de reduzir o tempo e o custo para as organizações. Nesse contexto, este trabalho forneceu importantes contribuições para o DBC, como:

- ✓ Tratamento igualitário para os requisitos funcionais e não-funcionais;
- ✓ Maximização na importância e no tratamento dos requisitos fundamentais, colocando os requisitos desejáveis como forma de diferenciação entre os candidatos;
- ✓ Critérios para definição de bases de dados que facilitam a seleção de COTS na engenharia de requisitos, onde a equipe de seleção tem informações suficientes para encontrar as melhores soluções para o problema;
- ✓ Reuso de dados para novos projetos;
- ✓ Um protótipo com cinco bases de dados que aplicam a metodologia ASCRE;
- ✓ Mapeamento do problema em forma de árvore, possibilitando a seleção múltipla;
- ✓ Automação no processo de decisão, garantindo o ganho de tempo e precisão matemática na hora da escolha.

No capítulo 2 foi visto o estado atual do desenvolvimento baseado em COTS, a forma tradicional da engenharia de requisito, a engenharia de requisito voltada para DBC, técnicas para tomada de decisão que poderão orientar a equipe de seleção na elaboração do critério de avaliação, e métodos propostos no desenvolvimento baseado em COTS que serviram de inspiração para o desenvolvimento da metodologia ASCRE, onde foram absorvidos as vantagens dessas metodologias e implementado procedimentos não encontrados nelas.

O capítulo 3 descreveu o método proposto detalhando as suas três fases. A primeira fase inicia o processo com o mapeamento do domínio e no final fornece para fase seguinte as bases de dados com o critério de avaliação definido e informações fundamentais para o início da negociação. Em seguida, a fase de checagem consolida as bases através da elicitação dos requisitos contratuais e desejáveis, ao mesmo tempo em que faz a negociação. Por fim, a fase de decisão fornecerá o resultado de forma totalmente automatizada.

O capítulo 4 mostrou um protótipo de uma implementação da metodologia ASCRE, denominada de IASCRE – Implementação da Automatização da Seleção de COTS na Engenharia de Requisitos, para confirmar a viabilidade do método.

No capítulo 5 pode-se verificar a viabilidade do método ASCRE num processo de seleção de produtos para gerenciar a rede de uma organização, onde a fase de decisão aplicou o critério de avaliação de forma automatizada, fornecendo ainda possibilidade de múltiplas escolhas em diversas opções.

6.2 Trabalhos Futuros

Esta dissertação forneceu um avanço importante para melhorar o desenvolvimento baseado em COTS na engenharia de requisitos. Todavia, este trabalho pode ser aprofundado com pesquisas mais abrangentes que vão melhorar o ASCRE. Como continuidade são relacionadas as seguintes sugestões:

- ✓ Automação do processo de negociação, onde poderão ser normalizadas as bases ERS e Fornecedor para propiciar uma negociação automatizada. Uma possível saída para essa negociação, pode ser o uso de técnicas da Inteligência Artificial;
- ✓ Implementação completa do ASCRE considerando uma coluna de prioridade na definição do critério de avaliação, um processo de rastreamento dos requisitos e uma relação de influencia entre os próprios requisitos. Com uma base de dados confiável e uma linguagem voltada para a WEB, Fornecer essas informações para o mercado de usuários de COTS;

- ✓ Sistematização do processo de identificação de requisitos, para possibilitar de forma efetiva o reuso de requisitos sem descaracterizar o objetivo principal do ASCRE;
- ✓ Possibilitar o mapeamento do domínio pelo próprio sistema a partir dos requisitos especificados.

Em suma, o ASCRE procurou promover na engenharia de requisitos alguns benefícios para o desenvolvimento baseado em COTS, isto é, economia de tempo, redução de riscos e de custos, além de minimizar os esforços da equipe de seleção.

GLOSSÁRIO

- AHP - Analytical Hierarchy Process é uma técnica de tomada de decisão baseada na decomposição de um problema com critérios múltiplos dentro de uma hierarquia de comparações entre pares para a formação de um ranking.
- COTS - Commercial Off-The-Shelf são produtos de software disponíveis no mercado de informática para serem reutilizados e são atualizados periodicamente pelos seus construtores.
- DBC - Desenvolvimento Baseado em COTS é um paradigma de desenvolvimento baseado em produtos de software já existente no mercado de informática.
- NFR Framework- É uma abordagem orientada a processo onde os requisitos não-funcionais são explicitamente representados como metas a serem obtidas.
- Stakeholder - Qualquer pessoa envolvida em um projeto de sistema de software.
- Template - Modo sistemático de apresentar um processo de alto nível que fornece orientação sobre as atividades a serem executadas.

REFERÊNCIAS

- [1] NCUBE, C. A Requirements Engineering Method for COTS-Based Systems Development. PhD thesis. School of Informatics, City University, London, Mai. 2000.
- [2] ALVES, C., Seleção de Produtos de Software “Utilizando uma Abordagem Baseada em Engenharia de Requisitos”. Dissertação de mestrado, Universidade Federal de Pernambuco, Centro de Informática, 2001.
- [3] NOVELLI Filho, A. Engenharia de Requisitos Norteador por Necessidades de Informação. São Paulo, 1999. 160 p. TESE (Doutorado) - Escola Politécnica, Universidade de São Paulo.
- [4] BOOCH, G., RUMBAUGH, J., JACOBSON, I. Unified Modeling Language user guide. Menlo Park, Addison-Wesley, 1999.
- [5] SOMMERVILLE, Ian. Software Engineering. 5. ed. Harlow, Addison-Wesley, 1995.
- [6] KONTIO, J., A Case Study in Applying a Systematic Method for COTS selection. International Conference on Software Engineering, 18., Berlin, Germany, 1996.
- [7] KONTIO, J., Chen, S., Limperos, K., A COTS Selection Method and Experiences of Its Use. NASA Software Engineering Laboratory, Greenbelt, MD, 1995.
- [8] ZANLORENCI, Edna P.; BURNETT, Robert C. O Tratamento da Informação (Requisitos no Ciclo de Vida do Produto) Caso Prático: Sistema de Informações de Previdência. Workshop de Engenharia de Requisitos, 4., Buenos Aires, Argentina, 2001. p. 55-72.
- [9] FOOK, K. Donato, “Método para Reutilização de Objetivos na Engenharia de Requisitos”. Dissertação de Mestrado, Universidade Federal do Maranhão, 2001.
- [10] CARVALHO, A. E., Uma Estratégia para Implantação de uma Gerência de Requisitos Visando a Melhoria dos Processos de Software. Workshop de Engenharia de Requisitos, 4., Buenos Aires, Argentina, 2001. p. 32-54.

- [11] NUSEIBEH, B. EASTERBROOK, S. Requirements Engineering: A Roadmap. Conferencia Internacional de Engenharia de Software, 22, Limerick, Irland, 2000.
- [12] CARVALHO, M. C. ABDELOUAHAB, Z., Um Método para Elicitação e Modelagem de Requisitos Baseado em Objetivos. Workshop de Engenharia de Requisitos, 4., Buenos Aires, Argentina, 2001. p. 319-337.
- [13] HSIA, Pei, DAVIS, Alan and KUNG, David. Status Report: Requirements Engineering. IEEE Software. November, 1993.
- [14] FINKELSTEIN, Anthony. Requirements Engineering: a review and research agenda. City University, Department of computer Science, London EC1V 0HB. 1995.
- [15] SOMMERVILLE, Ian and SAWYER, Pete. Requirements Engineering: A good practice guide. John Wiley & Sons. 1998.
- [16] SOMMERVILLE, Ian and KOTONYA, Gerald. Requirements Engineering: Processes and Techniques. John Wiley & Sons. 1998.
- [17] ZANLORENCI, Edna P.; BURNETT, Robert C. Engenharia de Requisitos (RE – Requirements Engineering): Conceitos e Fundamentos. BateByte Edição 77/98. 1998.
- [18] LAMSWEERDE, A., DARIMONT, R. E., LETIER, E., “Managing Conflicts in Goal-Driven Requirements Engineering”, IEEE Transactions on Software Engineering, Special Issue on Managing Inconsistency in Software Development. Nov. 1998.
- [19] MAIDEN, N., CREWS-SAVRE: Scenarios for Acquiring and Validating Requirements, Automated Software Engineering, 1998.
- [20] SHAW, M. E., GAINES, B., Requirements Acquisition. Software Engineering Journal, vol 11, 1995.
- [21] KARLSSON, K. E., RYAN, K., Software Requirements Prioritizing, 2nd International Conference on Requirements Engineering. Apr. 1996, pp. 110-116
- [22] BROWN, A. W., WALLNAU, K. C., The current State of CBSE, IEEE Software, Sep/Oct. 1998.

- [23] OBERNDORF, T., COTS and Open Systems – An Overview. 1997, disponível em <http://www.sei.cmu.edu/str/descriptions/cots.html#ndi>.
- [24] FOREMAN, J., On the Front Lines of COTS – Lessons Learned, Speculation for the Future. Software Engineering Institute. Disponível em http://www.sei.cmu.edu/activities/cbs/cbs_slides/stc98/frontlines/sld001.htm.
- [25] BERGNER, K. RAUSCH, A. e SIHLING, M. Componentware- The Big Picture. International Workshop on Component-Based Software Engineering, Apr. 1998.
- [26] BROWN, A. W. e WALLNAU, K. C. Engineering of Component-Based Systems. Software Engineering Institute, IEEE Computer Society Press. 1996
- [27] KUNDA, D. e BROOKS, L. Applying Social-Technical Approach for COTS Selection. Proceeding of the 4 UKAIS Conference, University of York, Apr. 1999.
- [28] TRAN, V. e LIU, D. A Procurement-centric Model for Engineering Component-based Software System. Proceedings of the 5 International Symposium on Assessment of Software Tools. Jun. 1997.
- [29] HAINES, G. CARNEY, D. e FOREMAN, J. Component-Based Software Development/COTS Integration. Software Engineering Institute, Carnegie Mellon University, USA. Jan. 1997.
- [30] BROWN, A. W. e SHORT, K. On Components and Objects: The Foundation of Component-Based Development. Proceedings of the Fifth International Symposium on Assessment of Software Tools. Jun, 1997.
- [31] CARNEY, D. Assembling Large Systems from COTS Components: Opportunities, Cautions, and Complexities. SEI Monographs on the Use of Commercial Software in Government Systems, Jun, 1997.
- [32] KOTONYA, G e SOMMERVILLE, I. A Framework for Integrating Functional and Non-Functional Requirements. International Workshop on Systems Engineering for Real Time Applications. UK. 1993.

- [33] MAIDEN, N. e RUGG, G. ACRE: Selecting Methods for Requirements Acquisition. *Software Engineering Journal*. Vol 11. número 3. 1996. pp. 183-192.
- [34] GOTEL, O e FINKELSTEIN, Anthony. An Analysis of the Requirements Traceability Problem. 1st. International Conference on Requirements Engineering, Colorado Springs, EUA. Apr. 1994.
- [35] SAATY, T. *The Analytic Hierarchy Process*. New York: McGraw-Hill, 1990.
- [36] CLEMEN, P. T. *Making Hard Decisions: An Introduction to Decision Analysis*. Hardcover 2nd edition, 1997.
- [37] YOUNG, K. The MAUT Analysis Case Study, Disponível em <http://www.emunix.emich.edu/~kcyoung/course/DS605/project/furnace/termpapl.html>.
- [38] LEITE, Júlio C.S.P. Viewpoints on Viewpoints. International Workshop on Multiple Perspectives in Software Development, 2., San Francisco, USA, 1996. p. 285-288.
- [39] MORISIO, M. e CARROLL, J.M. A Methodology for the Evaluation and Selection of Software Products. Dipartimento di Automatica e Informatica, Politecnico di Torino, Itália. 1997.
- [40] NCUBE, C e Maiden, N. A. PORE: Procurement-Oriented Requirements Engineering Method for the Component-Based System Engineering Development Paradigm. International Workshop on Component-Based Software Engineering, Mai. 1999.
- [41] DATE, C.J. *Uma Introdução a Sistemas de Banco de Dados*. Tradução da 6^a edição americana: Elza F. Gomide. Editora Edgar Blücher Ltda. São Paulo, 1999.
- [42] Paulino Cantanhêde, Zair Abdelouahab, Simara Vieira Rocha. A Method for Selection of Software with Engineering of Requirements. IADIS International Conference on Applied Computing 2005 (AC2005), Algarve, Portugal, 2005.
- [43] <http://www.expertchoice.com>